Legal Notices

The information in this document is subject to change without notice.

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

U.S. Government License

Confidential computer software. Valid license from HP required for possession, use, or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Additional Copyright Notices

This document and the software it describes may also be protected under one or more of the following copyrights. Additional copyrights are acknowledged in some individual manpages.


© Copyright 1985, 1986, 1988 Massachusetts Institute of Technology

© Copyright 1986-2000 Sun Microsystems, Inc.

© Copyright 1988 Carnegie Mellon University

© Copyright 1989-1991 The University of Maryland

© Copyright 1989-1993 The Open Software Foundation, Inc.

© Copyright 1990 Motorola, Inc.

© Copyright 1990-1992 Cornell University

© Copyright 1991-2003 Mentat, Inc.
Trademark Notices

Intel and Itanium are registered trademarks of Intel Corporation in the US and other countries and are used under license.

Java is a US trademark of Sun Microsystems, Inc.

Microsoft and MS-DOS are U.S. registered trademarks of Microsoft Corporation.

OSF/Motif is a trademark of The Open Group in the US and other countries.

UNIX is a registered trademark of The Open Group.

X Window System is a trademark of The Open Group.
**Preface**

HP-UX is the Hewlett-Packard Company’s implementation of a UNIX® operating system that is compatible with various industry standards. It is based on the System V Release 4 operating system and includes important features from the Fourth Berkeley Software Distribution.

The ten volumes of this manual contain the system reference documentation, made up of individual entries called manpages, named for the man command (see man (1)) that displays them on the system. The entries are also known as manual pages or reference pages.

### General Introduction

For a general introduction to HP-UX and the structure and format of the manpages, please see the introduction (9) manpage in volume 10.

### Section Introductions

The manpages are divided into sections that also have introduction (intro) manpages that describe the contents. These are:

- **intro (1)**  
  *Section 1: User Commands*  
  (A-M in volume 1; N-Z in volume 2)

- **intro (1M)**  
  *Section 1M: System Administration Commands*  
  (A-M in volume 3; N-Z in volume 4)

- **intro (2)**  
  *Section 2: System Calls*  
  (in volume 5)

- **intro (3C)**  
  *Section 3: Library Functions*  
  (A-M in volume 6; N-Z in volume 7)

- **intro (4)**  
  *Section 4: File Formats*  
  (in volume 8)

- **intro (5)**  
  *Section 5: Miscellaneous Topics*  
  (in volume 9)

- **intro (7)**  
  *Section 7: Device (Special) Files*  
  (in volume 10)

- **intro (9)**  
  *Section 9: General Information*  
  (in volume 10)

### Index

- **Index, All Volumes**  
  (in volume 10)
## Typographical Conventions

**audit (5)** An HP-UX manpage reference. For example, *audit* is the name and *5* is the section in the *HP-UX Reference*. On the web and on the Instant Information CD or DVD, it may be a hyperlink to the manpage itself. From the HP-UX command line, you can enter “`man audit`” or “`man 5 audit`” to view the manpage. See *man (1)*.

**Book Title** The title of a book. On the web and on the Instant Information CD or DVD, it may be a hyperlink to the book itself.

**Command** A command name or qualified command phrase.

**ComputerOutput** Text displayed by the computer.

**Emphasis** Text that is emphasized.

**Emphasis** Text that is strongly emphasized.

**ENVIRONVAR** The name of an environment variable.

**[ERRORNAME]** The name of an error number, usually returned in the *errno* variable.

**KeyCap** The name of a (usually) nonprinting keyboard key, such as *Ctrl-X* or *Tab*. Note that *Return* and *Enter* both refer to the same key.

**Replaceable** The name for a value that you replace in a command or function, or information in a display that represents several possible values.

**Term** The defined use of an important word or phrase.

**UserInput** Commands and other text that you type.

**$** User command prompt.

**#** Superuser (*root*) command prompt.
**Command Syntax**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Literal</strong></td>
<td>A word or character that you enter literally.</td>
</tr>
<tr>
<td><strong>Replaceable</strong></td>
<td>A word or phrase that you replace with an appropriate value.</td>
</tr>
<tr>
<td><strong>-chars</strong></td>
<td>One or more grouped command options, such as <code>-ikx</code>. The <em>chars</em> are usually a string of literal characters that each represent a specific option. For example, the entry <code>-ikxx</code> is equivalent to the individual options <code>-i</code>, <code>-k</code>, and <code>-x</code>. The plus character (+) is sometimes used as an option prefix.</td>
</tr>
<tr>
<td><strong>-word</strong></td>
<td>A single command option, such as <code>-help</code>. The <em>word</em> is a literal keyword. The difference from <code>-chars</code> is usually obvious and is clarified in an Options description. The plus character (+) and the double hyphen (--) are sometimes used as option prefixes.</td>
</tr>
<tr>
<td><code>[ ]</code></td>
<td>The bracket metacharacters enclose optional content in formats and command descriptions.</td>
</tr>
<tr>
<td><code>{ }</code></td>
<td>The brace metacharacters enclose required content in formats and command descriptions.</td>
</tr>
<tr>
<td>`</td>
<td>`</td>
</tr>
<tr>
<td><code>...</code></td>
<td>The ellipsis metacharacter after a token (<code>abc...</code>) or a right bracket (<code>[...]</code>) or a right brace (<code>{...}</code>) metacharacter indicates that the preceding element and its preceding whitespace, if any, may be repeated an arbitrary number of times.</td>
</tr>
<tr>
<td><code>...</code></td>
<td>Ellipsis is sometimes used to indicate omitted items in a range.</td>
</tr>
</tbody>
</table>
**Function Synopsis and Syntax**

HP-UX functions are described in a definition format rather than a usage format. The definition format includes type information that is omitted when the function call is actually included in a program.

The function syntax elements are the same as for commands, except for the options; see “Command Syntax” on page 7.

<table>
<thead>
<tr>
<th>Function General Definition</th>
<th>The general definition form is:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>typefunc(typeparam[,...])</code>;</td>
</tr>
<tr>
<td>For example:</td>
<td><code>int setuname(const char *name, size_t namelen);</code></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Function Usage</th>
<th>The usage form is:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>func(param[,...])</code>;</td>
</tr>
<tr>
<td>For example:</td>
<td><code>setuname(name[,...]);</code></td>
</tr>
</tbody>
</table>
Publishing History

Revisions of the HP-UX Reference are published with each initial version release and at significant update milestones for each release. The contents are current as of the publication dates. Since manpages are often updated in software patches, you can find the latest version of a manpage on an appropriately patched system, using the `man` command.

The list below is in reverse order of the date of publication.

<table>
<thead>
<tr>
<th>Part Number</th>
<th>Release; Date; Format; Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>B3921-60631</td>
<td>HP-UX 11i Version 2; December 2007 Update; one volume HTML; <a href="http://docs.hp.com">http://docs.hp.com</a> and Instant Information.</td>
</tr>
<tr>
<td>B2355-92066-75</td>
<td>HP-UX 11i Version 2; December 2007 Update; ten volumes PDF; <a href="http://docs.hp.com">http://docs.hp.com</a> and print.</td>
</tr>
<tr>
<td>B2355-60130</td>
<td>HP-UX 11i Version 3; February 2007 Release; one volume HTML; <a href="http://docs.hp.com">http://docs.hp.com</a> and Instant Information.</td>
</tr>
<tr>
<td>B2355-60127</td>
<td>HP-UX 11i Version 1; September 2005 Update; one volume HTML; <a href="http://docs.hp.com">http://docs.hp.com</a> and Instant Information.</td>
</tr>
<tr>
<td>B2355-90902-11</td>
<td>HP-UX 11i Version 1; September 2005 Update; ten volumes PDF; <a href="http://docs.hp.com">http://docs.hp.com</a> and print.</td>
</tr>
<tr>
<td>B2355-60105</td>
<td>HP-UX 11i Version 2; September 2004 Update; one volume HTML; <a href="http://docs.hp.com">http://docs.hp.com</a> and Instant Information.</td>
</tr>
<tr>
<td>B2355-90839-48</td>
<td>HP-UX 11i Version 2; September 2004 Update; ten volumes PDF; <a href="http://docs.hp.com">http://docs.hp.com</a> and print.</td>
</tr>
<tr>
<td>B2355-60103</td>
<td>HP-UX 11i Version 2; August 2003 Release; one volume HTML; <a href="http://docs.hp.com">http://docs.hp.com</a> and Instant Information.</td>
</tr>
<tr>
<td>B2355-90779-87</td>
<td>HP-UX 11i Version 2; August 2003 Release; nine volumes PDF; <a href="http://docs.hp.com">http://docs.hp.com</a> and print.</td>
</tr>
<tr>
<td>B9106-90010</td>
<td>HP-UX 11i Version 1.6; June 2002 Release; one volume HTML; <a href="http://docs.hp.com">http://docs.hp.com</a> and Instant Information.</td>
</tr>
</tbody>
</table>
B9106-90007-13  HP-UX 11i Version 1.5; June 2001 Release; seven volumes HTML; http://docs.hp.com and Instant Information.

B2355-90689-97  HP-UX 11i Version 1; December 2000 Release; nine volumes PDF and HTML; http://docs.hp.com, Instant Information and print.

B2355-90680-84  HP-UX 11.0; October 1997 Release; five volumes HTML; http://docs.hp.com.

B2355-90166  HP-UX 11.0; October 1997 Release; five volumes PDF; http://docs.hp.com.


B2355-90052  HP-UX 10.0; July 1995 Release; four volumes PDF.

**HP Encourages Your Comments**

HP encourages your comments regarding this document. We are committed to providing documentation that meets your needs. Send any errors found, suggestions for improvement, or compliments to:

feedback@fc.hp.com

Include the document title, manufacturing part number, and your comment, error found, or suggestion for improvement that you have concerning this document.
Volume Three
Table of Contents

Section 1M
Volume Three
Table of Contents

Section 1M
### Section 1M: System Administration Commands

<table>
<thead>
<tr>
<th>Entry Name(Section): name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>intro(1M): intro</td>
<td>introduction to system maintenance commands and application programs</td>
</tr>
<tr>
<td>accept(1M): accept, reject</td>
<td>allow or prevent LP printer queuing requests</td>
</tr>
<tr>
<td>acct(1M): acctdisk, acctduusg, accton, acctwtmp</td>
<td>overview of accounting and miscellaneous accounting commands</td>
</tr>
<tr>
<td>acctcom(1M): acctcom1, acctcom2</td>
<td>search and print process accounting files</td>
</tr>
<tr>
<td>acctcon(1M): acctcon, acctcon2</td>
<td>connect-time accounting</td>
</tr>
<tr>
<td>acctdisk</td>
<td>miscellaneous accounting command</td>
</tr>
<tr>
<td>acctduusg</td>
<td>miscellaneous accounting command</td>
</tr>
<tr>
<td>acctmerg(1M): acctmerg</td>
<td>merge or add total accounting files</td>
</tr>
<tr>
<td>accton</td>
<td>miscellaneous accounting command</td>
</tr>
<tr>
<td>acctprcl(1M): acctprcl, acctprcl2</td>
<td>process accounting</td>
</tr>
<tr>
<td>acctprc2</td>
<td>summarize process accounting</td>
</tr>
<tr>
<td>acctwtmp</td>
<td>shell procedures for accounting</td>
</tr>
<tr>
<td>bootpd</td>
<td>Internet Boot Protocol server</td>
</tr>
<tr>
<td>bootpquery</td>
<td>send BOOTREQUEST to BOOTP server</td>
</tr>
<tr>
<td>cachefstat(1M): cachefstat</td>
<td>cache file system statistics</td>
</tr>
<tr>
<td>captoinfo(1M): captoinfo</td>
<td>convert a termcap description into a terminfo description</td>
</tr>
<tr>
<td>cfsadmin</td>
<td>administer disk space used for caching file systems with CacheFS statistics</td>
</tr>
<tr>
<td>chargefee</td>
<td>change system configuration file</td>
</tr>
<tr>
<td>chargemf</td>
<td>charge fee to user</td>
</tr>
<tr>
<td>chargemf(1M): chargemf</td>
<td>shell procedures for accounting</td>
</tr>
<tr>
<td>cfsadmin(1M): cfsadmin</td>
<td>create the cat files for the on-line manpages</td>
</tr>
<tr>
<td>ch_re(1M): ch_re</td>
<td>change system configuration file</td>
</tr>
<tr>
<td>boot(1M): boot</td>
<td>bootstrap process</td>
</tr>
<tr>
<td>bootpd(1M): bootpd</td>
<td>Internet Boot Protocol server</td>
</tr>
<tr>
<td>bootpquery</td>
<td>send BOOTREQUEST to BOOTP server</td>
</tr>
<tr>
<td>cachefstat(1M): cachefstat</td>
<td>cache file system statistics</td>
</tr>
<tr>
<td>captoinfo(1M): captoinfo</td>
<td>convert a termcap description into a terminfo description</td>
</tr>
<tr>
<td>catman</td>
<td>control access to HP-UX Audio</td>
</tr>
<tr>
<td>catman</td>
<td>change or display event or system call audit status</td>
</tr>
<tr>
<td>amconfig(1M): amconfig</td>
<td>manage LUN configuration on disk array</td>
</tr>
<tr>
<td>amdlad(1M): amdlad</td>
<td>download new firmware to disks and controllers</td>
</tr>
<tr>
<td>amdsp(1M): amdsp</td>
<td>display status and configuration of disk array</td>
</tr>
<tr>
<td>amfmt(1M): amfmt</td>
<td>disk format</td>
</tr>
<tr>
<td>amlog(1M): amlog</td>
<td>display controller entries for disk array</td>
</tr>
<tr>
<td>ammgr(1M): ammgr</td>
<td>manage operating characteristics of disk array</td>
</tr>
<tr>
<td>amutil(1M): amutil</td>
<td>disk array management functions</td>
</tr>
<tr>
<td>arp(1M): arp</td>
<td>address resolution display and control</td>
</tr>
<tr>
<td>atm(1M): atm</td>
<td>control access to HP-UX Audio</td>
</tr>
<tr>
<td>auditadm(1M): auditadm</td>
<td>non-interactive editing of the authorization information in the RBAC databases</td>
</tr>
<tr>
<td>authck(1M): authck</td>
<td>check internal consistency of Authentication database</td>
</tr>
<tr>
<td>autofs(1M): autofs</td>
<td>initial system configuration/DHCP support command</td>
</tr>
<tr>
<td>auditd(1M): auditd</td>
<td>system configuration drift analyzer</td>
</tr>
<tr>
<td>auditlog(1M): auditlog</td>
<td>audit system configuration file</td>
</tr>
<tr>
<td>automountd(1M): automountd</td>
<td>autosoft mount/unmount daemons</td>
</tr>
<tr>
<td>automount(1M): automount</td>
<td>manage system database of automatically pushed STREAMS modules</td>
</tr>
<tr>
<td>backup(1M): backup</td>
<td>backup or archive file system</td>
</tr>
<tr>
<td>bastille(1M): bastille</td>
<td>system lockdown tool</td>
</tr>
<tr>
<td>bastille_drift(1M): bastille_drift</td>
<td>system configuration drift analyzer</td>
</tr>
<tr>
<td>bdf(1M): bdf</td>
<td>report number of free disk blocks (Berkeley version)</td>
</tr>
<tr>
<td>boot(1M): boot</td>
<td>bootstrap process</td>
</tr>
<tr>
<td>bootpd(1M): bootpd</td>
<td>Internet Boot Protocol server</td>
</tr>
<tr>
<td>bootpquery</td>
<td>send BOOTREQUEST to BOOTP server</td>
</tr>
<tr>
<td>cachefstat(1M): cachefstat</td>
<td>cache file system statistics</td>
</tr>
<tr>
<td>captoinfo(1M): captoinfo</td>
<td>convert a termcap description into a terminfo description</td>
</tr>
<tr>
<td>catman</td>
<td>control access to HP-UX Audio</td>
</tr>
<tr>
<td>cfsadmin(1M): cfsadmin</td>
<td>create the cat files for the on-line manpages</td>
</tr>
<tr>
<td>ch_re(1M): ch_re</td>
<td>change system configuration file</td>
</tr>
<tr>
<td>chargefee</td>
<td>shell procedures for accounting</td>
</tr>
</tbody>
</table>

**Table of Contents**

_Volumes Three and Four_

---

HP-UX 11i Version 2: December 2007 Update  
Hewlett-Packard Company
Table of Contents
Volumes Three and Four

Entry Name(Section): name

check_patches(1M): check_patches ........................................... HP-UX 11i patch check utility
chnlspath(1M): chnlpasqh .............................................................. configure message catalog path
cpiacct: shell procedures for accounting, check size of accounting file see acctsh(1M)
cleanup(1M): cleanup ................................................................. HP-UX patch cleanup utility
clear_locks(1M): clear_locks ....................................................... clear locks held on behalf of an NFS client
crlt(1M): crlt ........................................................................... clear inode
crlsv2c(1M): crlsnc ................................................................. clean x25 switched virtual circuit
cmdprivadm(1M): cmdprivadm .................................................... noninteractive editing of a command's authorization and privilege information in the privrun database
cmt_tune(1M): cmt_tune .............................................................. query, enable, or disable compartmentalization feature
convert awk(1M): convert awk ..................................................... converts old sendmail.cf files to new format
convertfs(1M): convertfs ................................................................ convert a file system to allow long file names
cpilxnyi(1M): cpilxnyi ................................................................ modify an attribute of a system complex
cpset(1M): cpset ........................................................................ install object files in binary directories
crashconf(1M): crashconf .......................................................... configure System crash dumps
crashutil(1M): crashutil ............................................................. manipulate crash dump data
crcate_sysfile(1M): crcate_sysfile ............................................. create a kernel system file
cron(1M): cron .......................................................................... timed-job execution daemon
dcopy(1M): dcopy .................................................................... copy HFS file system with compaction
devnm(1M): devnm ..................................................................... device name
df(1M): df ................................................................................ report number of free file system disk blocks
df: report number of free CDFS, HFS, or NFS file system disk blocks see df_hfs(1M)
df_hfs(1M): df ............................................................................. report number of free CDFS, HFS, or NFS file system disk blocks
df_vxfs(1M): df ........................................................................... report number of free disk blocks on VxFS file system
dhcpcrlent(1M): dhcpcrlent .......................................................... Client for Dynamic Host Configuration Protocol Server
dhcpcrlb2conf(1M): dhcpcrlb2conf .............................................. DHCP client database converter
dhcptools(1M): dhcptools ........................................................... command line tools for DHCP elements of bootpd
dhcpcrl6clientd(1M): dhcpcrl6clientd .......................................... DHCPv6 client daemon
dhcpcrl6d(1M): dhcpcrl6d .......................................................... Dynamic Host Configuration Protocol Server daemon for IPv6
dhcpcrl6db2conf(1M): dhcpcrl6db2conf ...................................... DHCPv6 client database converter
dig(1M): dig .............................................................................. domain information groper
diskinfo(1M): diskinfo ............................................................. describe characteristics of a disk device
disksect(1M): disksect ................................................................ calculate default disk section sizes
diskusg(1M): diskusg .............................................................. generate disk accounting data by user ID
dmesg(1M): dmesg ..................................................................... collect system diagnostic messages to form error log
dodisk: shell procedures for accounting, perform disk accounting see acctsh(1M)
dpp(1M): dpp ........................................................................ dedicated ports parser used by DDFSA software
drd(1M): drd ........................................................................... manage an inactive system image
drd-activate(1M): drd ................................................................ activate mode sets the inactive system image to be the primary boot disk the next time the system is booted
drd-clone(1M): drd ................................................................... clone mode clones the root volume group
drd-deactivate(1M): drd ............................................................. deactivate mode sets the active system image to be the primary boot disk the next time the system is booted
drd-mount(1M): drd ................................................................... mount mode mounts the inactive system image
drd-runcmd(1M): drd ................................................................. runcmd mode runs a command on the inactive system image that will not make any changes to the booted system, the kernel, or the process space
drd-unmount(1M): drd ................................................................. umount mode unmounds the inactive system image
drd_register_mirror(1M): drd_register_mirror ......................... notify DRD that a system image has been manually mirrored using LVM or VxVM commands
drd_unregister_mirror(1M): drd_unregister_mirror ................. notify DRD that an LVM or VxVM mirror of a system image is about to be manually removed
dump(1M): dump, rdump ........................................................... incremental file system dump
dumpfs(1M): dumpfs ................................................................ dump file system information
edquota(1M): edquota ............................................................. edit user quotas
efi_cp(1M): efi_cp ....................................................................... copy to or from EFI file
efi_fsinit(1M): efi_fsinit ............................................................ write an EFI file system header on a device file
efi_ls(1M): efi_ls ........................................................................ list EFI file information or contents of an EFI directory
efi_mkdir(1M): efi_mkdir ........................................................... make an EFI directory
efi_rm(1M): efi_rm ....................................................................... remove an EFI file
efi_rmdir(1M): efi_rmdir ........................................................... remove an EFI directory
Table of Contents
Volumes Three and Four

Entry Name (Section): name
Description

envd(1M): envd ... system physical environment daemon
exportfs(1M): exportfs ... export and unexport directories to NFS clients
extendfs(1M): extendfs ... extend a file system size (generic)
extendfs_hfs(1M): extendfs ... extend HFS file system size
extendfs_vxfs(1M): extendfs ... extend VxFS file system size
fbackup(1M): fbackup ... selectively back up files
fcmsutil(1M): fcmsutil ... fibre channel diagnostic utility
fdetach(1M): fdetach ... detach a STREAMS-based file descriptor from a file
ff(1M): ff ... list file names and statistics for file system
ff_hfs(1M): ff ... list file names and statistics for HFS file system
ff_vxfs(1M): ff ... fast find: list file names and statistics for VxFS file system
fingerd(1M): fingerd ... remote user information server
fixman(1M): fixman ... fix manpages for faster viewing with man command
format(1M): format ... format an HP SCSI disk array LUN
frecover(1M): frecover ... selectively recover files
frupower(1M): frupower ... turn on/off power or display status for cells and I/O chassis
fsadm(1M): fsadm ... file system administration command
fsadm_hfs(1M): fsadm_hfs ... HFS file system administration command
fsadm_vxfs(1M): fsadm ... resize or reorganize a VxFS file system
fsat_vxfs(1M): fsat ... cat a VxFS file system
fsc(1M): fsc ... file system consistency check and interactive repair
fsck: HFS file system consistency check and interactive repair ... see fsck_hfs(1M)
fsck_hfs(1M): fsck ... HFS file system consistency check and interactive repair
fsck_vxfs(1M): fsck ... check and repair VxFS file systems
fsckclean(1M): fsckclean ... determine shutdown status of specified file system
fsdb(1M): fsdb ... file system debugger (generic)
fsdb_hfs(1M): fsdb ... HFS file system debugger
fsdb_vxfs(1M): fsdb ... VxFS file system debugger
fsirand(1M): fsirand ... install random inode generation numbers
fstyp(1M): fstyp ... determine file system type
ftpd(1M): ftpd ... file transfer protocol server
fuser(1M): fuser ... list processes using a file or file structure
fwtmp(1M): fwtmp, wtmpfix ... manipulate connect accounting records
gated(1M): gated ... gateway routing daemon
gdc(1M): gdc ... operations user interface for gated
geocustoms(1M): geocustoms ... configure system language on multi-language systems
getext(1M): getext ... get VxFS extent attributes
getfilexsec(1M): getfilexsec ... display security attributes of binary executables
getx25(1M): getx25 ... set terminal type, modes, speed, and line discipline
getx25(1M): getx25 ... get x25 line
groupadd(1M): groupadd ... add a new group to the system
groupdel(1M): groupdel ... delete a group from the system
groupmod(1M): groupmod ... modify a group on the system
grpwck: group file checker ... see pwck(1M)
hotplug(1M): hotplug ... PCI I/O hotplug (attention button) events daemon
hpux(1M): hpux ... HP-UX bootstrap and installation utility
hpux.efi(1M): hpux.efi ... HP-UX bootstrap for Itanium-based systems
id(1M): id ... display security attributes of a process
identd(1M): identd ... display protected password databases
getrules(1M): getrules ... display compartment rules
getty(1M): getty ... set terminal type, modes, speed, and line discipline
getx25(1M): getx25 ... get x25 line
infocmp(1M): infocmp ... enable or disable secure internet services
inet(1M): inet ... process control initialization
insects(1M): insects ... install special (device) files
inetsvc(1M): inetsvc ... enable or disable secure internet services
infocmp(1M): infocmp ... compare or print out terminfo descriptions
init(1M): init ... process control initialization
insects(1M): insects ... install special (device) files

HP-UX 11i Version 2: December 2007 Update
Hewlett-Packard Company
# Table of Contents

Volumes Three and Four

<table>
<thead>
<tr>
<th>Entry Name/Section: name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>install(1M): install</td>
<td>install commands</td>
</tr>
<tr>
<td>intctl(1M): intctl</td>
<td>manage interrupt configuration of system</td>
</tr>
<tr>
<td>intrbald(1M): intrbald</td>
<td>daemon for balancing interrupts</td>
</tr>
<tr>
<td>iolnit(1M): iolnit</td>
<td>initialize I/O system</td>
</tr>
<tr>
<td>ioscan(1M): ioscan</td>
<td>scan the I/O system</td>
</tr>
<tr>
<td>isl(1M): isl</td>
<td>initial system loader</td>
</tr>
<tr>
<td>itemap(1M): itemap</td>
<td>load a keymap into the Internal Terminal Emulator (ITE)</td>
</tr>
<tr>
<td>kclog(1M): kclog</td>
<td>manage kernel configuration log file</td>
</tr>
<tr>
<td>kcmodule(1M): kcmodule</td>
<td>manage kernel modules and subsystems</td>
</tr>
<tr>
<td>kconfig(1M): kconfig</td>
<td>manage kernel configurations</td>
</tr>
<tr>
<td>kctune(1M): kctune</td>
<td>print kernel configuration pathnames</td>
</tr>
<tr>
<td>keyenvoy(1M): keyenvoy</td>
<td>manage kernel tunable parameters</td>
</tr>
<tr>
<td>keyserv(1M): keyserv</td>
<td>server for storing private encryption keys</td>
</tr>
<tr>
<td>killall(1M): killall</td>
<td>kill all active processes</td>
</tr>
<tr>
<td>killsm(1M): killsm</td>
<td>kills the sendmail daemon</td>
</tr>
<tr>
<td>kl(1M): kl</td>
<td>control kernel logging</td>
</tr>
<tr>
<td>kmpath(1M): kmpath</td>
<td>retrieve kernel name and associated kernel configuration information</td>
</tr>
<tr>
<td>kmtune(1M): kmtune</td>
<td>query, set, or reset system parameters</td>
</tr>
<tr>
<td>krs_flush(1M): krs_flush</td>
<td>Flushes kernel registry services daemon</td>
</tr>
<tr>
<td>labelit: copy file systems with label checking</td>
<td>see volcopy(1M)</td>
</tr>
<tr>
<td>labelit: copy file systems with label checking</td>
<td>see volcopy_hfs(1M)</td>
</tr>
<tr>
<td>labelit: label for VxFS file system</td>
<td>see volcopy_vxfs(1M)</td>
</tr>
<tr>
<td>lanadmin(1M): lanadmin</td>
<td>local area network administration</td>
</tr>
<tr>
<td>lanscan(1M): lanscan</td>
<td>display LAN device configuration and status</td>
</tr>
<tr>
<td>lastlogin: shell procedures for accounting, show last login date</td>
<td>see acctsh(1M)</td>
</tr>
<tr>
<td>ldapcinfo(1M): ldapcinfo</td>
<td>programmatically provides LDAP-UX information to non-interactive applications</td>
</tr>
<tr>
<td>ldapclientd(1M): ldapclientd</td>
<td>LDAP client daemon process</td>
</tr>
<tr>
<td>ldapugadd(1M): ldapugadd</td>
<td>add new accounts or groups to an LDAP directory server</td>
</tr>
<tr>
<td>ldapugdel(1M): ldapugdel</td>
<td>remove existing POSIX accounts or groups in an LDAP directory server</td>
</tr>
<tr>
<td>ldapuglist(1M): ldapuglist</td>
<td>display and enumerate POSIX-like account and group entries in an LDAP directory server</td>
</tr>
<tr>
<td>ldapugmod(1M): ldapugmod</td>
<td>modify existing POSIX accounts or groups in an LDAP directory server</td>
</tr>
<tr>
<td>libcadmin(1M): libcadmin</td>
<td>libc administration command</td>
</tr>
<tr>
<td>link(1M): link, unlink</td>
<td>execute link() and unlink() system calls without error checking</td>
</tr>
<tr>
<td>linkloop(1M): linkloop</td>
<td>verify LAN connectivity with link-level loopback</td>
</tr>
<tr>
<td>localedef(1M): localedef</td>
<td>generate a locale environment file</td>
</tr>
<tr>
<td>lockd(1M): lockd, rpc.lockd</td>
<td>network lock daemon</td>
</tr>
<tr>
<td>logins(1M): logins</td>
<td>display system and user login data</td>
</tr>
<tr>
<td>ldapadmin(1M): ldapadmin</td>
<td>configure the LP spooling system</td>
</tr>
<tr>
<td>ipana(1M): ipana</td>
<td>print LP spooler performance analysis information</td>
</tr>
<tr>
<td>lpmove: move LP scheduler requests</td>
<td>see lpsched(1M)</td>
</tr>
<tr>
<td>lpmovelsched: lpmovelsched</td>
<td>start/stop the LP request scheduler and move requests</td>
</tr>
<tr>
<td>lpsched(1M): lpsched</td>
<td>start/stop the LP scheduler requests</td>
</tr>
<tr>
<td>lsdev(1M): lsdev</td>
<td>list device drivers in the system</td>
</tr>
<tr>
<td>lssf(1M): lssf</td>
<td>list special file</td>
</tr>
<tr>
<td>lugadmin(1M): lugadmin</td>
<td>long user and group name enablement and display</td>
</tr>
<tr>
<td>lvchange(1M): lvchange</td>
<td>change LVM logical volume characteristics</td>
</tr>
<tr>
<td>lvcreate(1M): lvcreate</td>
<td>create logical volume in LVM volume group</td>
</tr>
<tr>
<td>lvdisplay(1M): lvdisplay</td>
<td>display information about LVM logical volumes</td>
</tr>
<tr>
<td>lvextend(1M): lvextend</td>
<td>stripe, increase space, increase mirrors for LVM logical volume</td>
</tr>
<tr>
<td>lvlnboot(1M): lvlnboot</td>
<td>prepare LVM logical volume to be root, swap, or dump volume</td>
</tr>
<tr>
<td>lvmovechk(1M): lvmovechk</td>
<td>check if disk volume is under HP Logical Volume Manager (LVM) control</td>
</tr>
<tr>
<td>lvmerge(1M): lvmerge</td>
<td>merge two LVM logical volumes into one logical volume</td>
</tr>
<tr>
<td>lvmmigrate(1M): lvmmigrate</td>
<td>prepare root file system for migration from partitions to LVM logical volumes</td>
</tr>
<tr>
<td>lvreduce(1M): lvreduce</td>
<td>decrease physical extents allocated to LVM logical volume</td>
</tr>
<tr>
<td>lvremove(1M): lvremove</td>
<td>remove logical volumes from LVM volume group</td>
</tr>
<tr>
<td>lvrmboot(1M): lvrmboot</td>
<td>remove LVM logical volume link to root, primary swap, or dump volume</td>
</tr>
</tbody>
</table>
### Table of Contents

#### Volumes Three and Four

<table>
<thead>
<tr>
<th>Entry Name (Section): name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>lvsplit(1M): lvsplit</td>
<td>split mirrored LVM logical volume into two logical volumes</td>
</tr>
<tr>
<td>lsync(1M): lsync</td>
<td>synchronize stale mirrors in LVM logical volumes</td>
</tr>
<tr>
<td>lvsed(1M): lvsed</td>
<td>lightweight resolver daemon</td>
</tr>
<tr>
<td>makedbm(1M): makedbm</td>
<td>make a Network Information System database</td>
</tr>
<tr>
<td>makemap(1M): makemap</td>
<td>creates database maps for sendmail</td>
</tr>
<tr>
<td>map-mbone(1M): map-mbone</td>
<td>multicast router connection mapper</td>
</tr>
<tr>
<td>mc(1M): mc</td>
<td>media changer manipulation utility</td>
</tr>
<tr>
<td>mk_kernel(1M): mk_kernel</td>
<td>load a kernel configuration from a system file</td>
</tr>
<tr>
<td>mkboot(1M): mkboot, cmboot</td>
<td>install, update, or remove boot programs from a disk device</td>
</tr>
<tr>
<td>mkfs(1M): mkfs</td>
<td>construct a file system (generic)</td>
</tr>
<tr>
<td>mkfs_hfs(1M): mkfs</td>
<td>construct an HFS file system</td>
</tr>
<tr>
<td>mkfs_vxfs(1M): mkfs</td>
<td>construct VxFS file system</td>
</tr>
<tr>
<td>mklost+found(1M): mklost+found</td>
<td>make a lost+found directory for fsck(1M)</td>
</tr>
<tr>
<td>mknod(1M): mknod</td>
<td>create special and FIFO files</td>
</tr>
<tr>
<td>mksf(1M): mksf</td>
<td>make a special (device) file</td>
</tr>
<tr>
<td>modprpw(1M): modprpw</td>
<td>modify protected password database</td>
</tr>
<tr>
<td>monacct: shell procedures for accounting, create accounting summary</td>
<td>see acctsh(1M)</td>
</tr>
<tr>
<td>mount(1M): mount, umount</td>
<td>mount and unmount a file system</td>
</tr>
<tr>
<td>mount: mount and unmount MemFS file systems</td>
<td>see mount_memfs(1M)</td>
</tr>
<tr>
<td>mount: mount and unmount NFS file systems</td>
<td>see mount_nfs(1M)</td>
</tr>
<tr>
<td>mount_cachefs(1M): mount_cachefs</td>
<td>mount CacheFS file systems</td>
</tr>
<tr>
<td>mount_cdfs(1M): mount, umount</td>
<td>mount and unmount CDFS file systems</td>
</tr>
<tr>
<td>mount_hfs(1M): mount, umount</td>
<td>mount and unmount HFS file systems</td>
</tr>
<tr>
<td>mount_ufs(1M): mount</td>
<td>mount an UFS file system</td>
</tr>
<tr>
<td>mount_memfs(1M): mount, umount</td>
<td>mount and unmount MemFS file systems</td>
</tr>
<tr>
<td>mount_nfs(1M): mount, umount</td>
<td>mount and unmount NFS file systems</td>
</tr>
<tr>
<td>mount_vxfs(1M): mount, umount</td>
<td>mount and unmount VxFS file system</td>
</tr>
<tr>
<td>mountall(1M): mountall, umountall</td>
<td>mount and unmount multiple file systems</td>
</tr>
<tr>
<td>mountd(1M): mountd</td>
<td>NFS mount request server</td>
</tr>
<tr>
<td>mrinto(1M): mrinto</td>
<td>multicast routing configuration information tool</td>
</tr>
<tr>
<td>mrouted(1M): mrouted</td>
<td>IP multicast routing daemon</td>
</tr>
<tr>
<td>mtail(1M): mtail</td>
<td>displays the last part of the mail log</td>
</tr>
<tr>
<td>mvdir(1M): mvdir</td>
<td>move a directory</td>
</tr>
<tr>
<td>naagt(1M): naagt</td>
<td>Native Agent Adapter for SNMP</td>
</tr>
<tr>
<td>named(1M): named</td>
<td>Internet domain name server</td>
</tr>
<tr>
<td>ncheck(1M): ncheck</td>
<td>generate path names from inode numbers</td>
</tr>
<tr>
<td>ncheck_vxfs(1M): ncheck</td>
<td>generate pathnames from inode numbers for VxFS file system</td>
</tr>
<tr>
<td>ndd(1M): ndd</td>
<td>network tuning</td>
</tr>
<tr>
<td>ndp(1M): ndp</td>
<td>IPv6 Neighbor Discovery cache display and control</td>
</tr>
<tr>
<td>netfmt(1M): netfmt</td>
<td>format tracing and logging binary files</td>
</tr>
<tr>
<td>netl(1M): netl</td>
<td>control network tracing and logging</td>
</tr>
<tr>
<td>netlconf(1M): netlconf</td>
<td>configure tracing and logging commands</td>
</tr>
<tr>
<td>newaliases(1M): newaliases</td>
<td>rebuilds the database for the mail aliases file</td>
</tr>
<tr>
<td>newfs(1M): newfs</td>
<td>construct a new file system</td>
</tr>
<tr>
<td>newfs_hfs(1M): newfs</td>
<td>construct a new HFS file system</td>
</tr>
<tr>
<td>newfs_vxfs(1M): newfs_vxfs</td>
<td>construct new VxFS file system</td>
</tr>
<tr>
<td>newkey(1M): newkey</td>
<td>create a new key pair in the publickey database</td>
</tr>
<tr>
<td>nfsd(1M): bsd, nfsd</td>
<td>NFS daemons</td>
</tr>
<tr>
<td>nfstat(1M): nfstat</td>
<td>Network File System statistics</td>
</tr>
<tr>
<td>nis_cachemgr(1M): nis_cachemgr</td>
<td>maintains a cache containing location information about NIS+ servers</td>
</tr>
<tr>
<td>nisaddcred(1M): nisaddcred</td>
<td>create NIS+ credentials</td>
</tr>
<tr>
<td>nisaddent(1M): nisaddent</td>
<td>create NIS+ tables from corresponding /etc files or NIS maps</td>
</tr>
<tr>
<td>nisclient(1M): nisclient</td>
<td>initialize NIS+ credentials for NIS+ principals</td>
</tr>
<tr>
<td>nisd: NIS+ service daemon</td>
<td>see rpc.nisd(1M)</td>
</tr>
<tr>
<td>nisd_resolv: NIS+ service daemon</td>
<td>see rpc.nisd(1M)</td>
</tr>
<tr>
<td>nisinit(1M): nisinit</td>
<td>NIS+ client and server initialization utility</td>
</tr>
<tr>
<td>nislog(1M): nislog</td>
<td>display the contents of the NIS+ transaction log</td>
</tr>
<tr>
<td>nispasswd(): NIS+ password update daemon</td>
<td>see rpc.nispasswd(1M)</td>
</tr>
<tr>
<td>nisping(1M): nisping</td>
<td>send ping to NIS+ servers</td>
</tr>
<tr>
<td>nispopulate(1M): nispopulate</td>
<td>populate the NIS+ tables in a NIS+ domain</td>
</tr>
<tr>
<td>nisservers(1M): nisservers</td>
<td>set up NIS+ servers</td>
</tr>
</tbody>
</table>
Table of Contents
Volumes Three and Four

<table>
<thead>
<tr>
<th>Entry Name(Section): name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>nissetup(1M): nissetup</td>
<td>initialize a NIS+ domain</td>
</tr>
<tr>
<td>nisshowcache(1M): nisshowcache</td>
<td>NIS+ utility to print out the contents of the shared cache file</td>
</tr>
<tr>
<td>nisstat(1M): nisstat</td>
<td>report NIS+ server statistics</td>
</tr>
<tr>
<td>nisupdp(1M): nisupdpkeys</td>
<td>update the public keys in a NIS+ directory object</td>
</tr>
<tr>
<td>ndate(1M): ndate</td>
<td>set time and date via NTP</td>
</tr>
<tr>
<td>ntp(1M): ntpq</td>
<td>Network Time Protocol query program</td>
</tr>
<tr>
<td>nulladm</td>
<td>shell procedures for accounting, create null file</td>
</tr>
<tr>
<td>odb(1M): odb</td>
<td>outbound connection daemon used by DDFA software</td>
</tr>
<tr>
<td>ocd(1M): ocd</td>
<td>outbound connection daemon debug utility used by DDFA software</td>
</tr>
<tr>
<td>olrad(1M): olrad</td>
<td>command for online addition/replacement of PCI IO cards</td>
</tr>
<tr>
<td>opx25(1M): opx25</td>
<td>execute HALGOL programs</td>
</tr>
<tr>
<td>ospf_monitor(1M): ospf_monitor</td>
<td>monitor OSPF gateways</td>
</tr>
<tr>
<td>owners(1M): owners</td>
<td>lists owners of outgoing network connections</td>
</tr>
<tr>
<td>parcreate(1M): parcreate</td>
<td>create a new partition</td>
</tr>
<tr>
<td>parmodify(1M): parmodify</td>
<td>modify an existing partition</td>
</tr>
<tr>
<td>parremove(1M): parremove</td>
<td>remove an existing partition</td>
</tr>
<tr>
<td>parunlock(1M): parunlock</td>
<td>unlock Complex or Partition Configuration Data</td>
</tr>
<tr>
<td>pcf(1M): pcf</td>
<td>processor-dependent code (firmware)</td>
</tr>
<tr>
<td>pfs_exportfs(1M): pfs_exportfs</td>
<td>export and unexport directories to PFS clients</td>
</tr>
<tr>
<td>pfs_mount(1M): pfs_mount</td>
<td>mount and unmount CD-ROM file systems</td>
</tr>
<tr>
<td>pfs_mountd(1M): pfs_mountd</td>
<td>PFS mount request server</td>
</tr>
<tr>
<td>pfs_mountd.rpc</td>
<td>PFS mount request server</td>
</tr>
<tr>
<td>pfs_umount</td>
<td>unmount CD-ROM file systems</td>
</tr>
<tr>
<td>pfmt(1M): pfmt</td>
<td>PFS daemon</td>
</tr>
<tr>
<td>ping</td>
<td>send echo request packets to a network host; test host availability</td>
</tr>
<tr>
<td>portmap</td>
<td>universal addresses to RPC program number mapper</td>
</tr>
<tr>
<td>power_ontoff(1M): power_ontoff</td>
<td>timed, system power on/off</td>
</tr>
<tr>
<td>ppp(1M): ppp</td>
<td>PPPoE (Point to Point Protocol over Ethernet) relay</td>
</tr>
<tr>
<td>pppoe(1M): pppoe</td>
<td>PPPoE (Point-to-Point Protocol over Ethernet) server daemon</td>
</tr>
<tr>
<td>pppoe-d(1M): pppoe-d</td>
<td>shell procedures for accounting, print session record file</td>
</tr>
<tr>
<td>pfstat(1M): pfstat</td>
<td>Basic Serial and HP AdvanceLink server</td>
</tr>
<tr>
<td>printd(1M): printd</td>
<td>shell procedures for accounting, print daily report</td>
</tr>
<tr>
<td>prset(1M): prset</td>
<td>let authorized users edit files that are under access control</td>
</tr>
<tr>
<td>privl(1M): privl</td>
<td>invoke another application with privileges after performing appropriate authorization checks and optionally reauthenticating the user</td>
</tr>
<tr>
<td>psmtp</td>
<td>shell procedures for accounting, print accounting file</td>
</tr>
<tr>
<td>psmtp-check</td>
<td>shell procedures for accounting, print accounting file</td>
</tr>
<tr>
<td>pswl(1M): pswl</td>
<td>change characteristics and access path of physical volume in LVM volume group</td>
</tr>
<tr>
<td>pswl-d(1M): pswl-d</td>
<td>create and manage processor sets</td>
</tr>
<tr>
<td>pswl-check(1M): pswl-check</td>
<td>check or repair a physical volume in LVM volume group</td>
</tr>
<tr>
<td>pswl-check-d(1M): pswl-check-d</td>
<td>create physical volume for use in LVM volume group</td>
</tr>
<tr>
<td>pswl-check-check(1M): pswl-check-check</td>
<td>display information about physical volumes in LVM volume group</td>
</tr>
<tr>
<td>pswl-check-check-d(1M): pswl-check-check-d</td>
<td>move physical extents from one LVM physical volume to other physical volumes</td>
</tr>
<tr>
<td>pswl-check-check-check(1M): pswl-check-check-check</td>
<td>remove an LVM physical volume</td>
</tr>
<tr>
<td>pswl-check-check-check-check(1M): pswl-check-check-check-check</td>
<td>password/group file checkers</td>
</tr>
<tr>
<td>pswl-check-check-check-check-check(1M): pswl-check-check-check-check-check</td>
<td>install, update or check the /etc/shadow file</td>
</tr>
<tr>
<td>ramutil(1M): ramutil</td>
<td>utility to create or destroy RAM disk volumes</td>
</tr>
<tr>
<td>rarpc(1M): rarpc</td>
<td>Reverse Address Resolution Protocol client</td>
</tr>
</tbody>
</table>
Table of Contents
Volumes Three and Four

Entry Name(Section): name Description
rarpd(1M): rarpd ........................................................ Reverse Address Resolution Protocol daemon
rhcacbchk(1M): rhcacbchk .............. verify the syntax of the Role-Based Access Control (RBAC) database files
c(1M): c ......................... general purpose sequencer invoked upon entering new run level
rcancel(1M): rcancel .................. remove requests from a remote line printer spooling queue
routed(1M): routed ..................... router discovery protocol daemon
rdump: incremental file system dump across network .................................................. see dump(1M)
reboot(1M): reboot ........................................... reboot the system
reject: prevent LP printer queuing requests ................................................................. see accept(1M)
remshd(1M): remshd .................. remote shell server
renice(1M): renice ........................................alter priority of running processes
report(1M): report ....................... summarize file system quotas
restore(1M): restore, rrestore .................. restore file system incrementally, local or over a network
rsexec(1M): rsexec ..................... remote execution server
rlp(1M): rlpprinter request to a remote system
rlpdaemon(1M): rlpprinter daemon ........................................ line printer daemon for LP requests from remote systems
rlpstat(1M): rlpprinter status print status of LP spooler requests on a remote system
rmboot - install, update, or remove boot programs from a disk device .................. see mkboot(1M)
rmt(1M): rmt ........................ remove a special (device) file
rmtadm(1M): rmtadm ............ non-interactive editing of role-related information in RBAC databases
route(1M): route .................. manually manipulate routing tables
rpc.lockd: network lock daemon .................................................. see lockd(1M)
rpc.nisd(1M): rpc.nisd_resolv, nisd_resolv ..................... NIS+ service daemon
rpc.nisd_resolv: NIS+ service daemon .................................................. see rpc.nisd(1M)
rpc.nispasswordd(1M): rpc.nispasswordd(), nispasswordd() ..................... NIS+ password update daemon
rpc.portmap: PC-NFS authentication and print request server ............................. see portmap(1M)
rpcinfo(1M): rpcinfo .......................... report RPC information
rquotad(1M): rquotad .................. remote quota server
rrestore: restore file system incrementally over a network ......................................... see restore(1M)
rstatd(1M): rstatd ..................... kernel statistics server
rutildvd(1M): rutildvd .............. Router Advertisement daemon for IPv6
runcnt(1M): runcnt ....................... run daily accounting
rusersd(1M): rusersd ............. network username server
rvxdump: incremental file system dump across network .............................................. see vxdump(1M)
rxrrestore: restore file system incrementally across network ..................................... see rxrrestore(1M)
rwall(1M): rwall ........................ write to all users over a network
rwalld(1M): rwalld .................... network rwall server
rwhod(1M): rwhod .................. system status server
sa1(1M): sa1, sa2, sadc ............... system activity report package
sa2: system activity report package .................................................. see sa1(1M)
sadc: system activity report package .................................................. see sa1(1M)
sam(1M): sam ............................................. system administration manager
sar(1M): sar ............................................. system activity reporter
sasmgr(1M): sasmgr ... utility to manage HP-UX Serial Attached SCSI (SAS) Mass Storage I/O cards / HBAs
savercrash(1M): savercrash .................. save a crash dump of the operating system
scsictl(1M): scsictl .................. control a SCSI device
sec: create and monitor jobs ........................................................................ see swjob(1M)
security_patch_check(1M): security_patch_check .............. check compliance with HP-UX security bulletins
sendmail(1M): sendmail ....................... send mail over the Internet
set_parms(1M): set_parms ................. set system initial identity parameters; host name, date, time, root password, networking
setboot(1M): setboot ...................... display and modify boot variables in stable storage
setext(1M): setext ...................... display and modify boot variables in stable storage
setfilesex(1M): setfilesex ................. set binary executable file security attributes
Table of Contents
Volumes Three and Four

Entry Name(Section): Description
--- | ---
setmemwindow(1M): setmemwindow set window id of a running process or start a program in a particular memory window
setmem(1M): setmem establish file-system mount table, /etc/mnttab
setoncenv(1M): setoncenv NFS environment configuration command
setprivgrp(1M): setprivgrp set special privileges for group
setrules(1M): setrules set compartment rules
setuname(1M): setuname change machine information
showmount(1M): showmount show all remote mounts
shutacct(1M): shell procedures for accounting, turn off accounting see acctsh(1M)
shutdown(1M): shutdown terminate all processing
signamed(1M): sig_named send signals to the domain name server
slpd(1M): slpd SLP Protocol Daemon
slpdc(1M): slpdc send signals to the slpd
smsh(1M): smsh restricted shell for sendmail
smnpd(1M): smnpd daemon that responds to SNMP requests
softpower(1M): softpower determine if softpower hardware is installed
spray(1M): spray spray packets
sprayd(1M): sprayd spray server
st(1M): st shared tape administration
startup(1M): startup shell procedures for accounting, start up accounting see acctsh(1M)
statd(1M): statd, rpc.statd network status monitor
strace(1M): strace STREAMS verification tool
strace(1M): strace strace(1M) write STRMIES event trace messages to standard output
strch(1M): strch, strconf change or query stream configuration
strclean(1M): strclean remove outdated STREAMS error log files
strconf: query stream configuration
strerr(1M): strerr receive error messages from the STREAMS log driver
strvf(1M): strvf STREAMS verification tool
swa(1M): swa HP-UX Software Assistant
swa-clean(1M): swa remove files created by SWA
swa-get(1M): swa download software from HP to resolve issues and make a depot
swa-report(1M): swa report software and security issues, and resolutions.
swa-step(1M): swa advanced control of Software Assistant execution steps
swacl(1M): swaci view or modify Access Control Lists
swagent: perform software management tasks as the agent of an SD command see swagentd(1M)
swagentd(1M): swagentd, swagent serve local or remote SD-UX software management tasks
swapinfo(1M): swapinfo system paging space information
swapon(1M): swapon enable device or file system for paging
swask(1M): swask ask for user response for SD-UX
swconfig(1M): swconfig configure, unconfigure, or reconfigure installed software
sycopy(1M): sycopy, swinstall copy software products for subsequent installation or distribution see swinstall(1M)
swinstall(1M): swcopy, swinstall install and configure software products, copy software products
swjob(1M): swjob, sd display job information, remove jobs, create and monitor jobs
swlist(1M): swlist display information about software products
swmodify(1M): swmodify modify software products in a target root or depot
swpackage(1M): swpackage package software products into a target depot or tape
swreg(1M): swreg register or unregister depots and roots
swremove(1M): swremove unconfigure and remove software products
svrverif(1M): svrverif verify software products
sync(1M): sync synchronize file systems
syncer(1M): syncer periodically sync for file system integrity
sysdef(1M): sysdef display system definition
syslogd(1M): syslogd log system messages
talkd(1M): talkd remote user communication server
tcpd(1M): tcpd access control facility for Internet services
telnetd(1M): telnetd TELNET protocol server
tftp(1M): tftp, tftp server terminfo compiler
termcap(1M): termcap service for accounting, turn on or off process accounting see acctsh(1M)
Table of Contents

Volumes Three and Four

Entry Name(Section): name

Description

udpublickey(1M): udpublickey .................................................. updates the publickey database file and the NIS map

umount: mount and unmount CDFS file systems ........................................ see mount_cdfs(1M)

umount: mount and unmount HPFS file systems ........................................ see mount_hdfs(1M)

umount: mount and unmount MemFS file systems .................................... see mount_memfs(1M)

umount: unmount a file system ....................................................... see mount(1M)

unlink: execute unlink() system call without error checking ..................... see link(1M)

umount: unmount VxFS file system ................................................ see mount_vxfs(1M)

update-ux(1M): update-ux ................................................................ updates the HP-UX operating system

update(1M): updaters ........................................................................ configuration file for NIS updating

ups_mond(1M): ups_mond ..................................................................... Uninterruptible Power System monitor daemon

useradd(1M): useradd ........................................................................ add a new user login to the system

usertdbck(1M): usertdbck .......................................................... verify or fix information in the user database, /var/adm/userdb

userdbget(1M): userdbget ....................................................... display information residing in the user database, /var/adm/userdb

userdsbset(1M): userdsbset .................................................. modify information in the user database, /var/adm/userdb

userdel(1M): userdel .............................................................. delete a user login from the system

usermod(1M): usermod ................................................................ modify a user login on the system

userstat(1M): userstat ............................................................ check status of local user accounts

utmpd(1M): utmpd ........................................................................ user accounting database daemon

uusnaps(1M): uusnaps ............................................................... list spooled uucp transactions grouped by transaction

uusub(1M): uusub ....................................................................... transfer files for the uucp system

uucico(1M): uucico ........................................................................... uucp spool directory clean-up

uucleanup(1M): uucleanup .......................................................... uucp spool directory clean-up

uucdp(1M): uucdp ........................................................................... UUCP over TCP/IP server daemon

uugetty(1M): uugetty ..................................................................... set terminal type, modes, speed and line discipline

uuln(1M): uuln ............................................................................. list spooled uucp transactions grouped by transaction

uusched(1M): uusched .............................................................. schedule uucp transport files

uusnap(1M): uusnap .............................................................. show snapshot of the UUCP system

uusnaps(1M): uusnaps .............................................................. sort and embellish uusnap output

uusub(1M): uusub ....................................................................... monitor uucp network

uuxqt(1M): uuxqt ............................................................................. execute remote uucp or uuxqt command requests

vgcfgbackup(1M): vgcfgbackup .................................................. create or update LVM volume group configuration backup file

vgcfgrestore(1M): vgcfgrestore ..................................................... restore volume group configuration

vgchange(1M): vgchange ........................................................ set LVM volume group availability

vgchgid(1M): vgchgid ..................................................................... set LVM volume group availability

vgcreate(1M): vgcreate ........................................................... create LVM volume group

vgdisplay(1M): vgdisplay ................................................................ display, display, volume group

vgexport(1M): vgexport .......................................................... export an LVM volume group and its associated logical volumes

vgextend(1M): vgextend ......................................................... extend an LVM volume group by adding physical volumes

vgimport(1M): vgimport .......................................................... import an LVM volume group onto the system

vgmodify(1M): vgmodify .............................................................. modify physical volume size changes and modify configuration parameters of an existing LVM volume group

vgrename(1M): vgrrename ........................................................ remove LVM volume group definition from the system

vgscan(1M): vgscan ........................................................................... scan physical volumes for LVM volume groups

vgsync(1M): vgsync ........................................................................... synchronize stale logical volume mirrors in LVM volume groups

vhardlinks(1M): vhardlinks ......................................................... checks the consistency of compartment rules for files with multiple hardlinks

vipw(1M): vipw ............................................................................. edit the password file

volcopy(1M): volcopy, labelit ..................................................... copy file systems with label checking

volcopy_hfs(1M): volcopy, labelit ..................................................... copy file systems with label checking

volcopy_vxfs(1M): volcopy, labelit ..................................................... copy VxFS file system with label checking

vtdaemon(1M): vtdaemon ........................................................ respond to vt requests

vxdiskkusg(1M): vxdiskkusg ..................................................... generate disk accounting data of VxFS file system by user ID

vxdump(1M): vxdump, vxdump ..................................................... incremental file system dump, local or across network

vxenablef(1M): vxenablef ........................................................ enable VxFS DMAP, OnLineIFS, or full VxFS functionality in the kernel

vxfsconvert(1M): vxfsconvert ..................................................... convert file system to vxfs file system

vxfsstat(1M): vxfsstat ............................................................. display VxFS file system statistics

vxlicense(1M): vxlicense ............................................................. VxFS licensing key utility

vxrestore(1M): vxrestore, rvxrestore ................................................ restore file system incrementally, local or across network
### Table of Contents

**Volumes Three and Four**

<table>
<thead>
<tr>
<th>Entry Name (Section)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vxtunefs(1M)</td>
<td>tune a VxFS file system</td>
</tr>
<tr>
<td>vxumount(1M)</td>
<td>unmount a VxFS file system</td>
</tr>
<tr>
<td>vxupgrade(1M)</td>
<td>upgrade the disk layout of a mounted VxFS file system</td>
</tr>
<tr>
<td>wall(1M)</td>
<td>write message to all users</td>
</tr>
<tr>
<td>whodo(1M)</td>
<td>which users are doing what</td>
</tr>
<tr>
<td>wtmpfix</td>
<td>manipulate connect accounting records</td>
</tr>
<tr>
<td>xntpd(1M)</td>
<td>Network Time Protocol daemon</td>
</tr>
<tr>
<td>xntpdc(1M)</td>
<td>special NTP query program</td>
</tr>
<tr>
<td>ypbind</td>
<td>Network Information Service (NIS) binder processes</td>
</tr>
<tr>
<td>ypinit(1M)</td>
<td>build and install Network Information Service databases</td>
</tr>
<tr>
<td>ypmake(1M)</td>
<td>create or rebuild Network Information Service databases</td>
</tr>
<tr>
<td>yppasswd(1M)</td>
<td>daemon for modifying Network Information Service passwd database</td>
</tr>
<tr>
<td>yppoll(1M)</td>
<td>query NIS server for information about an NIS map</td>
</tr>
<tr>
<td>yppush(1M)</td>
<td>force propagation of a Network Information Service database</td>
</tr>
<tr>
<td>ypset(1M)</td>
<td>bind to particular Network Information Service server</td>
</tr>
<tr>
<td>ypupdated(1M)</td>
<td>server for changing NIS information</td>
</tr>
<tr>
<td>ypxfr(1M)</td>
<td>transfer NIS database from NIS server to local node</td>
</tr>
<tr>
<td>ypxfdr</td>
<td>Network Information Service (NIS) transfer processes</td>
</tr>
</tbody>
</table>

---

*Hewlett-Packard Company HP-UX 11i Version 2: December 2007 Update*
Section 1M
Part 1

System Administration Commands
A-M
Section 1M
Part 1

System Administration Commands
A-M
NAME
intro - introduction to system maintenance commands and application programs

DESCRIPTION
This section describes commands that are used chiefly for system maintenance and administration purposes. The commands in this section should be used in conjunction with other sections of this manual, as well as the HP-UX System Administration manuals for your system.

Command Syntax
Unless otherwise noted, commands described in this section accept options and other arguments according to the following syntax:

```
name [ option ( s )] [ cmd_arg ( s )]
```

where the elements are defined as follows:

- **name**
  Name of an executable file.

- **option**
  One or more `option`s can appear on a command line. Each takes one of the following forms:
  - `no_arg_letter`
    A single letter representing an option without an argument.
  - `no_arg_letters`
    Two or more single-letter options combined into a single command-line argument.
  - `arg_letter<>opt_arg`
    A single-letter option followed by a required argument where:
    - `arg_letter`
      is the single letter representing an option that requires an argument,
    - `opt_arg`
      is an argument (character string) satisfying the preceding `arg_letter`,
    - `<>` represents optional white space.

- **cmd_arg**
  Path name (or other command argument) not beginning with `-`, or `-` by itself indicating the standard input. If two or more `cmd_arg`s appear, they must be separated by white space.

RETURN STATUS
Upon termination, each command returns two bytes of status, one supplied by the system giving the cause for termination, and (in the case of “normal” termination) one supplied by the program (for descriptions, see `wait(2)` and `exit(2)`). The system-supplied byte is 0 for normal termination. The byte provided by the program is customarily 0 for successful execution and non-zero to indicate errors or failure such as incorrect parameters in the command line, or bad or inaccessible data. Values returned are usually called variously “exit code”, “exit status”, or “return code”, and are described only where special conventions are involved.

WARNINGS
Some commands produce unexpected results when processing files containing null characters. These commands often treat text input lines as strings and therefore become confused upon encountering a null character (the string terminator) within a line.

SEE ALSO
`getopt(1)`, `exit(2)`, `wait(2)`, `getopt(3C)`, `hier(5)`, `introduction(9)`.

Web access to HP-UX documentation at `http://docs.hp.com`.
accept(1M)

NAME
accept, reject - allow/prevent LP destination queuing requests

SYNOPSIS
/usr/sbin/accept destination ...
/usr/sbin/reject [-r(reason)] destination ... [-r(reason) destination ...] ...

DESCRIPTION
The accept command permits the lp command (see lp(1)) to accept printing requests for each named LP printer or printer class destination queue.

The reject command causes the lp command to reject subsequent printing requests for each named destination queue. Requests already queued will continue to be processed for printing by the lpsched scheduler (see lpsched(1M)).

Use the lpstat command (see lpstat(1)) to find the status of destination queues.
For an overview of LP command interactions, see lp(1).

Options
The reject command can have the following option.

- r(reason)
  Specifies a string that is used to explain why the lp command is not accepting requests for a destination. reason applies to all queues mentioned up to the next -r option. If reason or -r(reason) is omitted, the default reason is "reason unknown". The maximum length of reason is 80 bytes. reason message of length greater than 80 bytes is truncated to 80 bytes.

reason is reported by the lpstat command and by the lp command when users direct requests to a rejected destination.

EXTERNAL INFLUENCES
Environment Variables
The LANG variable determines the language in which messages are displayed. If LANG is not specified or is set to the empty string, it defaults to "C" (see lang(5)).

If any internationalization variable contains an invalid setting, all internationalization variables default to "C" (see environ(5)).

International Code Set Support
Single- and multibyte character code sets are supported.

EXAMPLES
These examples assume you have a system with two printers named laser1 and jet2, and one class named lj that includes both printers.

Example 1
To allow all destinations to accept print requests:

accept laser1 jet2 lj

Example 2
To reject requests to the lj class destination, requiring users to choose a printer:

reject lj

Example 3
To reject requests to the individual printer destinations, requiring all requests to go through the class destination:

accept lj
reject -r"use the lj destination" laser1 jet2

WARNINGS
accept and reject operate on the local system only.
accept(1M)

FILES
/etc/lp Directory of spooler configuration data
/var/adm/lp Directory of spooler log files
/var/spool/lp Directory of LP spooling files and directories

SEE ALSO
enable(1), lp(1), lpstat(1), lpadmin(1M), lpsched(1M), rcancel(1M), rlp(1M), rlpdaemon(1M), rlpstat(1M).
NAME
acctdisk, acctdusg, acton, acctwtmp, closewtmp, utmp2wtmp - overview of accounting and miscellaneous accounting commands

SYNOPSIS
/usr/sbin/acct/acctdisk
/usr/sbin/acct/acctdusg [-u file] [-p file]
/usr/sbin/acct/accton [file]
/usr/sbin/acct/acctwtmp [-X] reason
/usr/sbin/acct/closewtmp
/usr/sbin/acct/utmp2wtmp

DESCRIPTION
Accounting software is structured as a set of tools (consisting of both C programs and shell procedures) that can be used to build accounting systems. The shell procedures, described in acctsh(1M), are built on top of the C programs.

Connect time accounting is handled by various programs that write records into the utmps database. The programs described in acctcon(1M) convert this file into session and charging records which are then summarized by acctmerg (see acctmerg(1M)).

Process accounting is performed by the HP-UX system kernel. Upon termination of a process, one record per process is written to a file (normally /var/adm/pacct). The programs in acctprc(1M) summarize this data for charging purposes; acctcms is used to summarize command usage (see acctcms(1M)). Current process data can be examined using acctcom (see acctcom(1M)).

Process accounting and connect time accounting (or any accounting records in the format described in acct(4)) can be merged and summarized into total accounting records by acctmerg (see acctmerg format in acct(4)). prtacct is used to format any or all accounting records (see acctsh(1M)).

acctdisk reads lines that contain user ID, login name, and number of disk blocks, and converts them to total accounting records that can be merged with other accounting records.

acctdusg reads its standard input (usually from find -print) and computes disk resource consumption (including indirect blocks) by login. Only files found under login directories (as determined from the password file) are accounted for. All files under a login directory are assumed to belong to that user regardless of actual owner. If -u is given, records consisting of those file names for which acctdusg charges no one are placed in file (a potential source for finding users trying to avoid disk charges). If -p is given, file is the name of the password file. This option is not needed if the password file is /etc/passwd. (See diskusg(1M) for more details.)

accton turns process accounting off if the optional file argument is omitted. If file is given, it must be the name of an existing file, to which the kernel appends process accounting records (see acct(2) and acct(4)).

acctwtmp writes a utmp record to its standard output if the -X option is not used. If the -X option is used, acctwtmp writes a wtmps-like record to stdout. The record contains the current time and a string of characters that describe the reason for writing the record. A record type of ACCOUNTING is assigned (see utmp(4) and utmps(4)). The string argument reason must be 11 or fewer characters, numbers, $, or spaces if -X option is not used. Otherwise, it must be 63 or fewer characters, numbers, $, or spaces. For example, the following are suggestions for use in reboot and shutdown procedures, respectively:

acctwtmp 'uname' >> /var/adm/wtmp
acctwtmp "file save" >> /var/adm/wtmp
acctwtmp -X 'uname' >> /var/adm/wtmps
acctwtmp -X 'uname' >> /var/adm/wtmps

closewtmp writes a DEAD_PROCESS record, for each user currently logged in, to the file /var/adm/wtmps. This program is invoked by runacct to close the existing wtmp file before creating a new one.

utmp2wtmp writes a USER_PROCESS record, for each user currently logged in, to the file /var/adm/wtmps. This program is invoked by runacct to initialize the newly created wtmps file.
acct(1M)

FILES
/usr/sbin/acct
/var/adm/pacct
/etc/passwd
/var/adm/wtmp
/var/adm/wtmps

Holds all accounting commands listed in section (1M) of this manual.
Current process accounting file.
Used for converting login name to user ID
Login/logoff history file.
New login/logoff history database.

SEE ALSO
acctcms(1M), acctcom(1M), acctcon(1M), acctmerg(1M), acctpre(1M), acctsh(1M), diskusg(1M), fwtmp(1M), runacct(1M), acct(2), acct(4), utmp(4), utmps(4), wtmps(4).

STANDARDS CONFORMANCE
acctdisk: SVID2, SVID3
accton: SVID2, SVID3
acctwtmp: SVID2, SVID3
NAME
acctcms - command summary from per-process accounting records

SYNOPSIS
/usr/sbin/acct/acctcms [options] files

DESCRIPTION
acctcms reads one or more files, normally in the form described in acct(4). It adds all records for processes that executed identically-named commands, sorts them, and writes them to the standard output, normally using an internal summary format.

Options
acctcms recognizes the following options:
-a Print output in ASCII rather than in the internal summary format. The output includes command name, number of times executed, total kcore-minutes, total CPU minutes, total real minutes, mean size (in K), mean CPU minutes per invocation, "hog factor", characters transferred, and blocks read and written, as in acctcom(1M). Output is normally sorted by total kcore-minutes.
-c Sort by total CPU time, rather than total kcore-minutes.
-j Combine all commands invoked only once under ***other.
-n Sort by number of command invocations.
-s Any file names encountered hereafter are already in internal summary format.
-t Process all records as total accounting records. The default internal summary format splits each field into prime- and non-prime-time parts. This option combines the prime and non-prime time parts into a single field that is the total of both, and provides upward compatibility with old (i.e., UNIX System V) style acctcms internal summary format records.

The following options can be used only with the -a option.
-p Output a prime-time-only command summary.
-o Output a non-prime- (offshift) time only command summary.

When -p and -o are used together, a combination prime and non-prime time report is produced. All the output summaries are total usage except number of times executed, CPU minutes, and real minutes which are split into prime and non-prime.

EXAMPLES
A typical sequence for performing daily command accounting and for maintaining a running total is:
acctcms file ... >today
cp total previoustotal
cacctms -s today previoustotal >total
cacctms -a -s today

WARNINGS
Unpredictable output results if -t is used on new-style internal-summary-format files, or if it is not used with old style internal summary format files.
The process mean memory size may overflow for values greater than MAXINT.

SEE ALSO
acct(1M), acctcom(1M), acctcon(1M), acctmerg(1M), acctpre(1M), acctsh(1M), fwtmp(1M), runacct(1M), acct(2), acct(4), utmp(4).

STANDARDS CONFORMANCE
acctcms: SVID2, SVID3
NAME
acctcom - search and print process accounting files

SYNOPSIS
/usr/sbin/acct/acctcom [option] ... [file] ... 

DESCRIPTION
The acctcom command reads file, standard input, or /var/adm/pacct, in the form described in acct(4) and writes selected records to standard output. Each record represents the execution of one process. The output has the following column titles:

   COMMAND NAME
   USER
   TTYNAME
   START TIME
   END TIME
   REAL (SECS)
   CPU (SECS)
   MEAN SIZE(K)

Optionally, the following can be displayed:

   F          fork()/exec() flag: 1 for fork() without exec()
   STAT        System exit status
   HOG FACTOR  System exit status
   KCORE MIN   Total blocks read and written
   CPU FACTOR  PRM process resource group ID
   CHARSTRNSFD
   BLOCKS READ
   PRMID

The command name is preceded by a # if a privileged user is required to execute the command.

For example, if a user is logged in as root, and executes the date command to check the time, this does not require a privileged user, and will be shown by acctcom without the # character on the line. If the user executes the command date 0731180092 to set the time, this requires a privileged user, and so will be marked with a # by acctcom.

If a process is not associated with a known terminal, a ? is printed in the TTYNAME field.

The system exit status STAT is 0 if the process terminated by calling exit. If it is not 0, it is the signal number that caused the process to terminate. If a core file image was produced as a result of the signal (see signal(5)), the value is the signal number plus 0200.

If no files are specified, and if standard input is associated with a terminal or /dev/null (as is the case when using & in a shell), acctcom reads /var/adm/pacct. Otherwise, it reads standard input.

If any file arguments are given, they are read in their respective order. Each file is normally read forward, that is, in chronological order by process-completion time. The file /var/adm/pacct is usually the current file to be examined. A busy system may need several such files of which all but the current file are found in /var/adm/pacct[1-9].

Options
acctcom recognizes the following values for the option argument. Listing options together has the effect of a logical AND.

   -a     Show some average statistics about the processes selected. Statistics are printed after the output records.
   -b     Read backwards, showing latest commands first. This option has no effect when standard input is read.
   -f     Print in octal the F flag and system exit status columns in the output.
   -h     Instead of mean memory size, MEAN SIZE(K), show the fraction of total available CPU time consumed by the process during its execution. This HOG FACTOR is computed as:
acctcom(1M)

- **i** Print columns containing the I/O counts in the output.
- **k** Instead of memory size, show total kcore-minutes.
- **m** Show mean core size (the default).
- **p** Show the PRM process resource group ID (PRMID) of each process. See DEPENDENCIES.
- **r** Show CPU factor:
  \[ \text{user-time} / (\text{system-time} + \text{user-time}) \]
- **t** Show separate system and user CPU times.
- **v** Exclude column headings from the output.
- **l** Show only processes belonging to terminal /dev/line.
- **u** Show only processes belonging to user, specified as: a user ID, a login name that is then converted to a user ID, a # which designates only those processes executed by a privileged user, or ? which designates only those processes associated with unknown user IDs. The # and ? characters should be preceded by a backslash (\) and typed as \# and \? to prevent the shell from interpreting the # as the start of a comment, or the ? as a pattern.
- **g** Show only processes belonging to group, specified as either the group ID or group name.
- **s** time Select processes existing at or after time, given in the format:
  \[ \text{hour}[:\text{minute}[:\text{second}]] \]
- **e** time Select processes existing at or before time; see **s**.
  Using the same time for both **s** and **e** shows the processes that existed at time; see **s**.
- **S** time Select processes starting at or after time; see **s**.
- **E** time Select processes ending at or before time; see **s**.
- **n** pattern Show only commands matching pattern, where pattern is a regular expression as in ed(1) except that * means one or more occurrences.
- **q** Do not print any output records. Just print the average statistics as with the **a** option.
- **o** ofile Copy selected process records in the input data format to ofile. Suppress standard output printing.
- **H** factor Show only processes that exceed factor, where factor is the "hog factor" as explained in option **h**.
- **O** time Show only those processes with operating system CPU time exceeding time; see **s**.
- **C** sec Show only processes with total CPU time, system plus user, exceeding sec seconds.
- **I** chars Show only processes transferring more characters than the cut-off number given by chars.
- **R** prmgp Show only processes belonging to process resource group prmgp, specified as either process resource group name or ID number. See DEPENDENCIES.

**WARNINGS**

acctcom only reports on processes that have terminated. For active processes, use the ps command (see ps(1)).

If time exceeds the current system clock time, time is interpreted as occurring on the previous day.

The accounting flag is not cleared when one processes exec's another, but only when one process forks another. One side-effect of this is that some processes will be marked with #, when users do not expect them to be.
For example, the `login` command requires a privileged user to assume the identity of the user who is logging-in, setting the ASU bit in the accounting flag (which ultimately causes the # symbol in the `acctcom` output). After assuming the user's identity, `login` exec's the user's shell. Since the exec does not clear the ASU flag, the shell will inherit it, and be marked with a # in the `acctcom` output.

The mean memory size may overflow for values greater than `MAXINT`.

**DEPENDENCIES**

**HP Process Resource Manager**

The `-P` and `-R` options require the optional HP Process Resource Manager (PRM) software to be installed and configured. See `prmconfig(1)` for a description of how to configure HP PRM, and `prmconf(4)` for the definition of process resource group.

**FILES**

`/etc/group`  
`/etc/passwd`  
`/var/adm/pacct`

**SEE ALSO**

`ps(1)`, `su(1)`, `acct(1M)`, `acctms(1M)`, `acctcom(1M)`, `acctmerg(1M)`, `acctsh(1M)`, `fwtmp(1M)`, `runacct(1M)`, `acct(2)`, `wait(2)`, `acct(4)`, `utmp(4)`, `signal(5)`.


**STANDARDS CONFORMANCE**

`acctcom`: SVID2, SVID3
NAME
acctcon, acctcon1, acctcon2 - connect-time accounting

SYNOPSIS
/usr/sbin/acct/acctcon [options]
/usr/sbin/acct/acctcon1 [options]
/usr/sbin/acct/acctcon2

DESCRIPTION
The acctcon1 command converts a sequence of login/logoff records read from its standard input to a sequence of records, one per login session. Its input should normally be redirected from /var/adm/wtmp or /var/adm/wtmps. Its output is ASCII, giving device, user ID, login name, prime connect time (seconds), non-prime connect time (seconds), session starting time (numeric), and starting date and time. Prime connect time is defined as the connect time within a specific prime period on a non-holiday weekday (Monday through Friday). The starting and ending time of the prime period and the year's holidays are defined in file /etc/acct/holidays.

acctcon2 expects as input a sequence of login session records, produced by acctcon1, and converts them into total accounting records (see tacct format in acct(4)).

acctcon combines the functionality of acctcon1 and acctcon2 into one program. It takes the same input format as acctcon1 and writes the same output as acctcon2.

acctcon1 recognizes the following options:
-\p  Print input only, showing line name, login name, and time (in both numeric and date/time formats).
-\t  acctcon1 maintains a list of lines on which users are logged in. When it reaches the end of its input, it emits a session record for each line that still appears to be active. It normally assumes that its input is a current file, so that it uses the current time as the ending time for each session still in progress. The -t flag causes it to use, instead, the last time found in its input, thus ensuring reasonable and repeatable numbers for non-current files.

acctcon1 and acctcon recognize the following options:
-\l file  file is created to contain a summary of line usage showing line name, number of minutes used, percentage of total elapsed time used, number of sessions charged, number of logins, and number of logoffs. This file helps track line usage, identify bad lines, and find software and hardware oddities. Hang-up, termination of login (see login(1)), and termination of the login shell each generate logoff records, so that the number of logoffs is often three to four times the number of sessions. See init(1M) and utmp(4).
-\o file  file is filled with an overall record for the accounting period, giving starting time, ending time, number of reboots, and number of date changes.
-\W  When this option is used, the records of the type found in /var/adm/wtmps, are read from the specified input.

EXAMPLES
These commands are typically used as shown below. The file ctmp is created only for the use of commands described by the acctprec(1M) manual entry:

acctcon1 -t -l lineuse -o reboots < wtmp | sort +1n +2 > ctmp
acctcon2 < ctmp | acctmerg > ctacct

or
acctcon -t -l lineuse -o reboots < wtmp | acctmerg > ctacct

With -W option:

acctcon1 -W -t -l lineuse -o reboots < wtmps | sort +1n +2 > ctmp
acctcon2 < ctmp | acctmerg > ctacct

or
acctcon(1M) acctcon(1M)

acctcon -W -t -l lineuse -o reboots < wtmps | acctmerg > ctacct

Note:
The file wtmps can be either /var/adm/wtmps or a file containing records of the type found in /var/adm/wtmps.

WARNINGS
The line usage report is confused by date changes. Use wtmpfix (see fwtmp(1M)) to correct this situation.

FILES
/var/adm/wtmp
/var/adm/wtmps
/etc/acct/holidays

SEE ALSO
login(1), acct(1M), acctcms(1M), acctcom(1M), acctmerg(1M), acctpret(1M), acctsh(1M), fwtmp(1M), init(1M), utmpd(1M), runacct(1M), acct(2), getbwent(3C), acct(4), utmp(4).

STANDARDS CONFORMANCE
acctcon1: SVID2, SVID3
acctcon2: SVID2, SVID3
NAME
acctmerg - merge or add total accounting files

SYNOPSIS
/usr/sbin/acct/acctmerg [options] [file] ...

DESCRIPTION
acctmerg reads its standard input and up to nine additional files, all in the tacct format (see acct(4)) or an ASCII version thereof. It merges these inputs by adding records whose keys (normally user ID and name) are identical, and expects the inputs to be sorted on those keys.

Options
acctmerg recognizes the following options:

- a Produce output in ASCII version of tacct.
- i Input files are in ASCII version of tacct.
- p Print input with no processing.
- t Produce a single record that totals all input.
- u Summarize by user ID, rather than user ID and name.
- v Produce output in verbose ASCII format, with more precise notation for floating point numbers.

EXAMPLES
The following sequence is useful for making "repairs" to any file kept in this format:

acctmerg -v < file1 > file2
edit file2 as desired ...
acctmerg -i < file2 > file1

SEE ALSO
acct(1M), acctms(1M), acctcom(1M), acctcon(1M), acctprc(1M), acctsh(1M), fwtmp(1M), runacct(1M), acct(2), acct(4), utmp(4).

STANDARDS CONFORMANCE
acctmerg: SVID2, SVID3
NAME
acctprc, acctprc1, acctprc2 - process accounting

SYNOPSIS
/usr/sbin/acct/acctprc
/usr/sbin/acct/acctprc1 [ctmp]
/usr/sbin/acct/acctprc2

DESCRIPTION
acctprc1 reads input in the form described by acct(4), adds login names corresponding to user IDs, then
writes for each process an ASCII line giving user ID, login name, prime CPU time (tics), non-prime CPU time
(tics), and mean memory size (in memory segment units). If ctmp is given, it is expected to contain a list
of login sessions in the form described in acctcon(1M), sorted by user ID and login name. If this file is not
supplied, it obtains login names from the password file. The information in ctmp helps it distinguish
among different login names that share the same user ID.

acctprc2 reads records in the form written by acctprc1, summarizes them by user ID and name, then
writes the sorted summaries to the standard output as total accounting records.

acctprc combines the functionality of acctprc1 and acctprc2 into one program. It takes the same
input format as acctprc1 (but does not accept the ctmp argument) and writes the same output as
acctprc2.

These commands are typically used as shown below:

    acctprc1 ctmp < /var/adm/pacct | acctprc2 > ptacct

or

    acctprc < /var/adm/pacct > ptacct

EXTERNAL INFLUENCES
Environment Variables
For the output of acctprc2, if the user IDs are identical, LC_COLLATE determines the order in which
the user names are sorted.

    If LC_COLLATE is not specified in the environment or is set to the empty string, the value of LANG is
used as a default. If LANG is not specified or is set to the empty string, a default of "C" (see
lang(5)) is used instead of LANG. If any internationalization variable contains an invalid setting, acctprc2 behaves
as if all internationalization variables are set to "C" (see environ(5)).

WARNINGS
Although it is possible to distinguish among login names that share user IDs for commands run normally, it
is difficult to do this for those commands run from cron for example (see cron(1M)). More precise conver-
sion can be done by faking login sessions on the console via the acctwtmp program in acct(1M).

A memory segment of the mean memory size is a unit of measure for the number of bytes in a logical
memory segment on a particular processor.

The mean memory size may overflow for values greater than MAXINT.

FILES
/etc/passwd

SEE ALSO
acct(1M), acctcms(1M), acctcom(1M), acctcon(1M), acctmerg(1M), acctsh(1M), cron(1M), fwtmp(1M),
runacct(1M), acct(2), acct(4), utmp(4).

STANDARDS CONFORMANCE
acctprc1: SVID2, SVID3
acctprc2: SVID2, SVID3
NAME
chargefee, ckpacct, dodisk, lastlogin, monacct, nulladm, prctmp, prdaily, prtacct, shutacct, startup, turnacct - shell procedures for accounting

SYNOPSIS
/usr/sbin/acct/chargefee login-name number
/usr/sbin/acct/ckpacct [blocks]
/usr/sbin/acct/dodisk [-o] [files ...]
/usr/sbin/acct/lastlogin
/usr/sbin/acct/monacct number
/usr/sbin/acct/nulladm file
/usr/sbin/acct/prctmp
/usr/sbin/acct/prdaily [-l] [-c] [mmdd]
/usr/sbin/acct/prtacct file [heading]
/usr/sbin/acct/shutacct [reason]
/usr/sbin/acct/startup
/usr/sbin/acct/turnacct on | off | switch

DESCRIPTION
chargefee Can be invoked to charge a number of units to login-name. A record is written to /var/adm/fee to be merged with other accounting records during the night.

ckpacct Should be initiated via cron(1M). It periodically checks the size of /var/adm/pacct. If the size exceeds blocks, 1000 by default, turnacct is invoked with argument switch. If the number of free disk blocks in the /var file system falls below 500, ckpacct automatically turns off the collection of process accounting records via the off argument to turnacct. When at least this number of blocks is restored, the accounting will be activated again. This feature is sensitive to the frequency at which ckpacct is executed, usually by cron.

dodisk Should be invoked by cron to perform the disk accounting functions. By default, it will do disk accounting on the special files in /etc/fstab. If the -o flag is used, it does a slower version of disk accounting by login directory. files specifies the one or more filesystem names where disk accounting is to be done. If files is used, disk accounting will be done on these filesystems only. If the -o flag is used, files should be mount points of mounted filesystems. If omitted, they should be the special file names of mountable filesystems.

lastlogin Invoked by runacct to update /var/adm/acct/sum/loginlog which shows the last date on which each user logged in (see runacct(1M)).

monacct Should be invoked once each month or each accounting period. number indicates which month or period it is. If number is not given, it defaults to the current month (01 through 12). This default is useful if monacct is to be executed via cron on the first day of each month. monacct creates summary files in /var/adm/acct/fiscal and restarts summary files in /var/adm/acct/sum.

nulladm Creates file with mode 664 and ensures that owner and group are adm. It is called by various accounting shell procedures.

prctmp Can be used to print the session record file normally /var/adm/acct/nite/ctmp created by acctcon1(see acctcon(1M)).

prdaily Invoked by runacct (see runacct(1M)) to format a report of the previous day’s accounting data. The report resides in /var/adm/acct/sum/xprtmmdd where mmdd is the month and day of the report. The current daily accounting reports may be printed by typing prdaily. Previous days’ accounting reports can be printed by using the mmdd option and specifying the exact report date desired. The -l flag prints a report of exceptional usage by login id for the specified date. Previous daily reports are cleaned up and therefore inaccessible after each invocation of monacct. The -c flag prints a report of exceptional

38 Hewlett-Packard Company − 1 − HP-UX 11i Version 2: December 2007 Update
acctsh(1M)

acctsh(1M) can be used to format and print any total accounting file.

`acct` Can be used to format and print any total accounting file.

`shutacct` Should be invoked during a system shutdown to turn process accounting off and append a "reason" record to `/var/adm/wtmp`.

`startup` Should be called by system startup scripts to turn the accounting on whenever the system is brought up.

`turnacct` An interface to `accton` (see sect(1M)) to turn process accounting on or off. The `switch` argument turns accounting off, moves the current `/var/adm/pacct` to the next free name in `/var/adm/pacct` and then turns accounting back on again. (incr is a number starting with 1 and incrementing by one for each additional `pacct` file.) `turnacct` is called by `ckpacct`, and thus can be run under cron and used to keep `pacct` to a reasonable size.

FILES

`/usr/sbin/acct` holds all accounting commands listed in section (1M) of this manual.

`/var/adm/fee` accumulator for fees

`/var/adm/acct/nite` working directory

`/var/adm/pacct` current file for per-process accounting

`/var/adm/pacct*` used if `pacct` gets large, and during execution of daily accounting procedure

`/usr/sbin/acct/ptecms.awk` contains the limits for exceptional usage by command name

`/usr/sbin/acct/ptelus.awk` contains the limits for exceptional usage by login id

`/var/adm/acct/sum` summary directory, should be saved

`/var/adm/wtmp` login/logoff summary

SEE ALSO

`acct(1M), acctcms(1M), acctcom(1M), acctcon(1M), acctmerg(1M), acctprc(1M), cron(1M), diskusg(1M), fwtmp(1M), runacct(1M), acct(2), acct(4), utmp(4).`

STANDARDS CONFORMANCE

chargefee: SVID2, SVID3

ckpacct: SVID2, SVID3

dodisk: SVID2, SVID3

lastlogin: SVID2, SVID3

monacct: SVID2, SVID3

pctmp: SVID2, SVID3

prdaily: SVID2, SVID3

protacct: SVID2, SVID3

shutacct: SVID2, SVID3

startup: SVID2, SVID3

turnacct: SVID2, SVID3

NAME
AM60Srvr - disk array server daemon

SYNOPSIS
/opt/hparray/bin

DESCRIPTION
AM60Srvr is the server portion of the Array Manager 60 management software. It monitors the operation and performance of the disk array, and services external requests from clients executing disk array commands. AM60Srvr monitors disk array performance and status, maintains disk array logs, and allows clients to examine and change disk array configuration.

AM60Srvr must be running to allow management of the disk array using the command line utilities. Host I/Os are not dependent on AM60Srvr and are serviced regardless of whether it is running or not. Because of its importance in managing the disk arrays, AM60Srvr is launched automatically when the system is booted.

SECURITY CONFIGURATION
This command is modified for all security configurations.

Security Behavior/Restrictions
Use of this command is restricted to authorized users only.

Command Authorizations
This command requires the sysadmin authorization to successfully execute.

Privileges
The command has been modified to support least privilege. The potential privileges possessed by the command and their uses include:

allowdacread This privilege is raised to provide discretionary read access to the devices.
allowdacwrite This privilege is raised to provide discretionary write access to the devices.
allowmacread This privilege is raised to provide mandatory read access to the devices.
allowmacwrite This privilege is raised to provide mandatory write access to the devices.
filesysops This privilege is raised to allow the mknod(2) system call to succeed.
writeaudit The command generates its own audit records and submits these directly to the system audit trail. This privilege is raised whenever the command needs to write an audit record.

DIAGNOSTICS
All significant changes in disk array status detected by AM60Srvr are entered in /var/adm/syslog/syslog.log. These entries form a history of disk array operation and can be used to track operation.

AUTHOR
AM60Srvr was developed by HP.

FILES
The following files support the operation of AM60Srvr. These files are typically located in /opt/hparray/lib/nls/msg/C/.

AM60Srvr.cat Message catalog file
am60cl.cat Message catalog file for all command line clients
fwerrcod.cat Message catalog file for command line clients
oemmsg01.cat Message catalog for OEM-specific messages

SEE ALSO
amdsp(1M), amcfg(1M), amutil(1M), amlog(1M), ammgr(1M).
amcfg(1M)

NAME
amcfg - manages LUN (logical drive) configuration on the disk array

SYNOPSIS
amcfg -L cntrlrID:LUN -d channel:ID,channel:ID,...
{-r RAIDlevel [-c capacity] [-s SegmentSize] [-force] [-V] [-?] ArrayID
amcfg -R cntrlrID:LUN -d channel:ID,channel:ID,...
{-r RAIDlevel [-c capacity] [-s SegmentSize] [-force] [-V] [-?] ArrayID
amcfg -C -d channel:ID,channel:ID,... [ {-r RAIDlevel [-s SegmentSize]} [-V] [-?] ArrayID
amcfg -D LUN [-V] [-?] ArrayID
amcfg -M LUN -c cntrlrID [-V] [-?] ArrayID

DESCRIPTION
amcfg manages the LUN configuration on the disk array identified by ArrayID. This involves creating, deleting, replacing, and assigning ownership of all LUNS on the disk array.

WARNING: Deleting a LUN will destroy all the data on the LUN. Backup all vital data before deleting a LUN.

The ArrayID used to address the disk array can be the disk array serial number or name, if one has been assigned to the disk array.

Identifying Disk Modules
Disk modules are identified within Array Manager 60 using a numbered pair of the form n.n. The first number identifies the SCSI channel (or bus) connecting the array controller to the enclosure containing the disk module. The channel number is indicated on the back of the array controller enclosure. The second number is the disk module SCSI ID. The SCSI ID is determined by the slot in which the disk module is installed, but is not the same as the physical slot number (0-9).

For example, the numbered pair 2:1 identifies the disk module on channel 2 with a SCSI ID of 1. Refer to the Disk Array FC/60 User’s Guide for more information on disk module addressing.

Options
amcfg supports the following options:
- -c capacity Indicates the capacity of the LUN being created. The capacity can be specified in megabytes (M), or gigabytes (G) by appending the appropriate letter to the value. If not specified, the default is gigabytes.
By default, the LUN will use the entire capacity available from all the disks. It is possible to specify a lower value for LUN capacity, but this will result in unused disk capacity. Any capacity not included in the LUN will be inaccessible and is essentially wasted capacity.
- -C Calculate the capacity of a LUN using the specified disks, RAID level, and stripe segment size. This command does not actually create the LUN. It simply returns the capacity available if a LUN is created using the specified values.
- -d channel:ID,channel:ID Identifies the disks used for the LUN. Multiple disks can be specified. Each disk is identified by channel number (1-6) and SCSI ID (0-4, 8-12). Note that multiple disks must be specified with no spaces between each disk.
To ensure high-availability, each disk should be in a different disk enclosure. This protects the LUN against an enclosure failure.
NOTE: When selecting disks for a RAID 0/1 LUN, the order in which you specify disks is important. The first half of the disks you specify will be the primary data disks, and the second half of the disks will be the disk mirrors. To maintain data availability, the disk mirrors must be in a different enclosure than the data disks.
For example, assume you are creating a 4-disk RAID 0/1 LUN using one disk enclosure on channel 1, and a second disk enclosure on channel 2. Specifying the disks in the order 1:2, 1:3, 2:2, 2:3 would result in mirrored pairs of 1:2/2:2 and 1:3/2:3. This would ensure availability because the data disks are on channel 1,
and the mirror disks are on channel 2. However, specifying disks in the order 1:2, 2:2, 1:3, and 2:3 would result in mirrored pairs of 1:2/1:3 and 2:2/2:3. This would put the data disk and mirror disk of each pair in the same enclosure, making the LUN vulnerable to an enclosure failure.

-D LUN Delete the LUN identified by LUN.

-force Allows a LUN to be created using two or more disks in the same enclosure. This option allows you to override the high-availability protection designed into the LUN binding process. Using this option you can specify more than one disk per enclosure. You can also use this option to create a RAID 5 LUN that includes more than six disks.

-L cntrlID:LUN Create a LUN with the number specified by LUN. The LUN will be owned by the controller identified by cntrlID.

-M LUN -c cntrlID Change the ownership of the LUN specified by LUN to the controller specified by cntrlID.

-r RAIDlevel Identifies the RAID level used for the LUN. RAID levels that can be specified are 0, 1, and 5. A RAID 0/1 LUN is created by selecting RAID 1 with more than two disks. RAID 0 is available only on firmware HP07 and later.

CAUTION: RAID 0 does not provide data redundancy. It should only be used in situations where high performance is more important than data protection. The failure of any disk within a RAID 0 LUN will cause the loss of all data on the LUN. RAID 0 should only be used for non-critical data that could be lost in the event of a hardware failure.

-R cntrlID:LUN Replace (unbind and rebind) the LUN with the number specified by LUN on the controller identified by cntrlID.

-s SegmentSize Identifies the stripe segment size used for the LUN. The stripe segment size is specified in Kbytes and must be a multiple of the current cache page size setting. For example, if the cache page size is set to 4 Kbytes, valid settings would be 4, 8, 16, 64, etc.

-V Verbose mode displays additional command execution, state, and/or status messages.

-? Display expanded usage message. This option overrides all others.

SECURITY CONFIGURATION
This command is modified for all security configurations.

Security Behavior/Restrictions
Use of this command is restricted to authorized users only.

Command Authorizations
This command requires the sysadmin authorization to successfully execute.

Privileges
The command has been modified to support least privilege. The potential privileges possessed by the command and their uses include:

allowdacread This privilege is raised to provide discretionary read access to the devices.
allowdacwrite This privilege is raised to provide discretionary write access to the devices.
allowmacread This privilege is raised to provide mandatory read access to the devices.
allowmacwrite This privilege is raised to provide mandatory write access to the devices.
filesysops This privilege is raised to allow the mknod(2) system call to succeed.
writeaudit  The command generates its own audit records and submits these directly to the system audit trail. This privilege is raised whenever the command needs to write an audit record.

EXTERNAL INFLUENCES
Environment Variables
LC_MESSAGES determines the language in which messages are displayed. The current language settings can be checked with `locale(1)`.

RETURN VALUE
amcfg returns the following values:
0  Successful completion.
1  An error in execution (I/O, subsystem, security, etc.) occurred.
2  An error in command syntax occurred: for example, an unknown command-line option was passed.
3  Timeout in communication to server. May indicate AM60Srvr is not running.

DIAGNOSTICS
The following messages can be generated by amcfg:
Usage:amcfg { -L | -R } <CntrlrID>:<LUN>
amcfg -C -d <Channel:Id>[,<Channel:Id>...] -r <RAIDLevel> [-s <SegmentSize>] [-V] [-S] <ArrayID>
amcfg -D <LUN> [-V] [-S] <ArrayID>
amcfg -M <LUN> -c <CntrlrID> [-V] [-S] <ArrayID>

Extended help: amcfg -?
An error in command syntax has occurred. Re-enter the command with all necessary arguments.
amcfg: The <ArrayID> entered does not identify a known, supported array
The specified ArrayID does not exist or does not identify a device that is communicating with the system. Verify the array exists and is operational.
amcfg: The specified controller is not physically installed.
A controller-specific command was issued to a non-existent or inaccessible controller.
amcfg: The specified LUN does not exist in this array.
The specified LUN does not exist.
amcfg: The capacity specified exceeds total available for disks.
LUN not created.
The capacity requested exceeds available capacity for the disks requested.
amcfg: Unknown capacity subscript: x
A capacity subscript of M or G was expected, or none at all. Found x instead. Re-enter command with correct subscript.
amcfg: Capacity must be >= 10 MB
An attempt was made to create a LUN with a capacity less than 10 MB. A minimum size of 10 MB is required. Re-issue the command with a larger capacity specification.
amcfg: LUN already owned by specified controller
An attempt was made to change the ownership of a LUN to the same controller that already owns it. No operation is performed.
amcfg: LUN 0 may not be deleted. Use -R option to replace LUN 0 instead.
An attempt was made to delete LUN 0, which is not allowed. LUN 0 may, however, be reconfigured using amcfg -R. No operation is performed.
amcfg: LUN’s owning controller not responding.
   Check controller status or bind LUN to the other controller.
   An attempt was made to bind a LUN to a controller which is not responding. No operation is performed.

amcfg: LUN’s new owning controller not responding.
   Check controller status or bind LUN to the other controller.
   An attempt was made to replace (unbind and rebind) a LUN, but the new owning controller does not respond. No operation is performed.

amcfg: The specified operation cannot be performed on a passive controller.
   An attempt was made to bind or replace a LUN on a passive controller.

amcfg: Error in command execution, <Additional Error Info>:
   <Error Info Decode>
   The command failed due to a device error, an internal error, or a system error. The Additional Error Info and Error Info Decode fields will hold specifics about the failure and its cause.

amcfg: A RAID 5 LUN with more than one disk on a channel is not a recommended high availability configuration. Use -force to override.

amcfg: A RAID 5 LUN with more than one disk in an enclosure is not a recommended high availability configuration. Use -force to override.

amcfg: A RAID 5 LUN with more than six disks in this array is not a recommended high availability configuration. Use -force to override.

amcfg: A RAID 0/1 LUN with both mirrors residing on the same channel is not a recommended high availability configuration. Use -force to override.

amcfg: A RAID 0/1 LUN with both mirrors residing in the same enclosure is not a recommended high availability configuration. Use -force to override.

amcfg: A RAID 0/1 LUN may contain no more than 30 disks.
   The absolute maximum number of disks that the array will support in a RAID 0/1 LUN has been exceeded.

amcfg: A RAID 5 LUN may contain no more than 20 disks.
   The absolute maximum number of disks that the array will support in a RAID 5 LUN has been exceeded.

EXAMPLES
   Bind a 5-disk, RAID 5 LUN on disk array RACK_51. The disk array includes five disk enclosures, each on its own channel. The capacity of the LUN will default to the total capacity available from the five disks. The LUN is owned by controller A, is assigned number 2, and uses a stripe segment size of 16 Kbytes. Note that each disk is in a different enclosure for high availability, and that there are no spaces between the individual disk parameters.
   amcfg -L A:2 -d 1:1,2:2,3:1,4:4,5:3 -r 5 -s 16 RACK_51

   Bind a RAID 0/1 LUN on disk array RACK_51. The LUN is owned by controller B, is assigned number 4, and uses a stripe segment size of 4 Kbytes. Note that the disks selected create mirrored pairs that are in separate enclosures (1:3, 2:3 and 1:4, 2:4). This maintains high availability. Although RAID 1 is specified, the inclusion of more than two disks causes the disk array to create a RAID 0/1 LUN.
   amcfg -L B:4 -d 1:3,1:4,2:3,2:4 -r 1 -s 4 RACK_51

   Calculate the available capacity from a five-disk RAID 5 LUN on disk array RACK_51.
   amcfg -C -d 1:2,2:2,3:2,4:6 -r 5 -s 32 RACK_51

   Delete LUN 3 on the disk array RACK_51:
   amcfg -D 3 RACK_51
Change the ownership of LUN 0 on disk array RACK_51 to controller B:

```
amcfg -M 0 -c B RACK_51
```

**DEPENDENCIES**

`AM60Srvr` must be running to execute this command. See `AM60Srvr(1M)`.  

**AUTHOR**

`amcfg` was developed by HP.

**SEE ALSO**

`amdsp(1M), ammgr(1M), amutil(1M), amlog(1M), AM60Srvr(1M)`.
NAME
amdload - downloads new firmware to the disk array controllers, the disk system BCC controllers, and
disks

SYNOPSIS
amdload -D [ all | channel:ID,channel:ID,...] codefile ArrayID
amdload -C [ all | ctrlrID] codefile ArrayID
amdload -S [ all | channel,channel, ...] [-force] codefile ArrayID
amdload -i ArrayID
amdload -?

Remarks
Downloading firmware should only be performed by service-trained personnel. If firmware is not down-
loaded properly, it may cause the disk array or some its components to become inoperative.

DESCRIPTION
amdload copies new firmware code to the disk array controllers, the disk system BCC controllers, or the
disks in the disk array identified by ArrayID. The new code is stored in the file identified by codefile.
The ArrayID used to address the disk array can be the disk array serial number or name, if one has been
assigned to the disk array.

Identifying Disk Modules
Disk modules are identified within Array Manager 60 using a numbered pair of the form n:n. The first
number identifies the SCSI channel (or bus) connecting the array controller to the enclosure containing the
disk module. The channel number is indicated on the back of the array controller enclosure. The second
number is the disk module SCSI ID. The SCSI ID is determined by the slot in which the disk module is
installed, but is not the same as the physical slot number (0-9).

For example, the numbered pair 2:1 identifies the disk module on channel 2 with a SCSI ID of 1. Refer to the
Disk Array FC/60 User’s Guide for more information on disk module addressing.

Options
amdload supports the following options:

-C
Download new firmware to the disk array controllers. The new code is stored in the file
identified by codefile.

If the all option is included, download firmware to both disk array controllers.
If the ctrlrID option is included, download firmware to the specified controller (A or B).

There are three controller firmware files: bootware, firmware, and NVSRAM. The files
must be downloaded in the proper sequence. When upgrading firmware, download the
bootware code file first, followed by the firmware code file, and finally the NVSRAM file.
When converting to an earlier version of firmware (downgrading), the correct sequence is
firmware first, then bootware, followed by NVSRAM.

NOTE: On firmware version HP07 and higher, the bootware and firmware files are avail-
able in a single "packaged" file. The packaged file should be downloaded first, followed by the
NVSRAM file. This order applies for both upgrades and downgrades.

Typical code filename extensions are as follows:
*.bwd       bootware
#.apd       firmware (or "appware")
#.dlp       packaged bootware and firmware
#.dl or *.dlp NVSRAM

-D
Download new firmware to the disks. The new code is stored in the file identified by
codefile.

If the all option is included, download firmware to all disks on the disk array.
If the channel:ID option is included, download firmware only to the specified disks. Disk
are identified by channel (1-6) and SCSI ID (0-4, 8-12).
-i  Display the firmware revisions for all hardware components of the specified disk array.
-S  Download new firmware to the disk system BCC controllers. The new code is stored in the file identified by codefile.

If the all option is included, download firmware to all disk system BCC controllers on the disk array.
If the channel option is included, download firmware only to the specified disk systems.
Disk systems are identified by channel (1-6). Firmware will be downloaded to both BCC controllers on the disk system.
If the -force option is included, non-optimal disk states will be ignored.
-?  Display expanded usage message. This option overrides all others.

SECURITY CONFIGURATION
This command is modified for all security configurations.

Security Behavior/Restrictions
Use of this command is restricted to authorized users only.

Command Authorizations
This command requires the sysadmin authorization to successfully execute.

Privileges
The command has been modified to support least privilege. The potential privileges possessed by the command and their uses include:
- allowdacread  This privilege is raised to provide discretionary read access to the devices.
- allowdacwrite This privilege is raised to provide discretionary write access to the devices.
- allowmacread  This privilege is raised to provide mandatory read access to the devices.
- allowmacwrite This privilege is raised to provide mandatory write access to the devices.
- filesysops   This privilege is raised to allow the mknod(2) system call to succeed.
- writeaudit   The command generates its own audit records and submits these directly to the system audit trail. This privilege is raised whenever the command needs to write an audit record.

EXTERNAL INFLUENCES
Environment Variables
- LC_MESSAGES determines the language in which messages are displayed. The current language settings can be checked with locale(1).

RETURN VALUE
amdload returns the following values:
0  Successful completion.
1  An error in execution (I/O, subsystem, security, etc.) occurred.
2  An error in command syntax occurred: for example, an unknown command-line option was passed.
3  Timeout in communication to server. May indicate AM60Srvr is not running.

DIAGNOSTICS
The following messages can be generated by amdload:

Usage: amdload -D {all | Chan:ID,Chan:ID,...} <codefile> <ArrayID>
amdload -C { all | <cntrlrID> } codefile <ArrayID>
amdload -S { all | <Chan>,<Chan>,...,} [-force] <codefile> <ArrayID>
amdload -i <ArrayID>
Extended help: amdload -?
amdload(1M) amdload(1M)

amdload -?
An error in command syntax has occurred. Re-enter the command with all necessary arguments.

amdload: Arg out of range
One of the arguments has exceeded its maximum or minimum size, or is incorrect in form. Check the size and form of each argument.

amdload: The <ArrayID> entered does not identify a known, supported array
The specified ArrayID does not exist or does not identify a device that is communicating with the system. Verify the array exists and is operational.

amdload: Firmware download has been aborted.
Disk <disk> does not exist in the subsystem.
The specified disk does not exist in the subsystem. Re-enter the list of disks, making sure all disks are physically present. A similar message will appear for channels that do not exist.

amdload: Firmware download has been aborted.
There is more than one type of disk mechanism among the disks selected for update.
Any particular firmware file is only valid for one type of disk mechanism. This message will appear if the disks specified for update are of more than one mechanism type. Re-enter the list of disks, only including disks of one mechanism type.

amdload: The disk <disk> was repeated in your list.
Re-enter the command, listing each item once.
This message will appear if a disk or channel is repeated in the input list. Be sure to list each item once.

amdload: Firmware download has been aborted.
LUN <lun> is not in an Optimal state. Wait until all LUNs are in an Optimal state before attempting further firmware downloads.
LUNs must be in an Optimal state before attempting disk or array controller downloads. This message will appear if any LUNs are not Optimal prior to a download, or if a LUN enters a non-Optimal state between disk downloads.

amdload: Error in command execution, <Additional Error Info>:
<Error Info Decode>
The command failed due to a device error, an internal error, or a system error. The Additional Error Info and Error Info Decode fields will hold specifics about the failure and its cause.

EXAMPLES
Download new firmware from file coderev2.1 to both disk array controllers on disk array RACK_51:
 amdload -C all coderev2.1 RACK_51
Download new firmware from file diskrev3.2 to the specified disks (channel 6, SCSI ID 2 and channel 6, SCSI ID 4) in disk array RACK_51:
 amdload -D 6:2,6:4 diskrev3.2 RACK_51
Download new firmware from file bccrev14 to the BCC controllers on all the disk systems on disk array RACK_51:
 amdload -S all bccrev14 RACK_51

DEPENDENCIES
AM60Srvr must be running to execute this command. See AM60Srvr(1M).

AUTHOR
amdload was developed by HP.

SEE ALSO
amdsp(1M), amcgt(1M), amutil(1M), amlog(1M), ammgr(1M), AM60Srvr(1M).
NAME
amdsp - display the status and operating configuration of the disk array

SYNOPSIS
amdsp -p [-V] [-S] [-?] DeviceFile
amdsp -i [-V] [-S] [-?]
amdsp -R [-V] [-?]  

DESCRIPTION
amdsp displays status and configuration information for the disk array identified by ArrayID. Logical configuration, physical configuration, and current status can all be displayed using amdsp. A list of all the disk arrays recognized by the host can also be displayed.

The ArrayID used to address the disk array can be the disk array serial number or name, if one has been assigned to the disk array.

Identifying Disk Modules
Disk modules are identified within Array Manager 60 using a numbered pair of the form n:n. The first number identifies the SCSI channel (or bus) connecting the array controller to the enclosure containing the disk module. The channel number is indicated on the back of the array controller enclosure. The second number is the disk module SCSI ID. The SCSI ID is determined by the slot in which the disk module is installed, but is not the same as the physical slot number (0-9).

For example, the numbered pair 2:1 identifies the disk module on channel 2 with a SCSI ID of 1. Refer to the Disk Array FC/60 User’s Guide for more information on disk module addressing.

Options
amdsp supports the following options:

- none
  Display general information about the disk array. This includes product and vendor information, array state, and capacity usage.

- a
  Display all information presented by the -c, -d, -g, -h, -s, -l, and -r options. This is a quick way of displaying all configuration and status information about the disk array. This option will display information for all LUNs.

- A
  Display information related to the interface between the server and the specified disk array.

- c
  Display information for both disk array controllers and the BCC disk system controllers.

- d
  Display information for all disks installed in the array. Each disk is identified by a numbered pair of the form channel:ID. The enclosure and slot in which the disk is installed are also displayed.

- g
  Display disk group information. Information will be displayed for each disk group on the disk array. There is a disk group for each LUN.

- h
  Display information about the disk array hardware. This includes the hardware components in the disk array subsystem and each disk system.

- i
  Display the ID of all disk arrays currently connected to and recognized by the server. If the disk array has been assigned a name, it will also be displayed. Because this option is device-independent, it does not use ArrayID.

- l [LUN]
  Display information for the LUN identified by LUN. If LUN is not specified, display information for all LUNs on the disk array.

- p DeviceFile
  Display hardware path information for the controller corresponding to the specified device file.

- r
  Display the progress of all rebuilds currently in progress on the disk array.

- R
  Rescan for disk arrays. The host will scan for all supported disk arrays and update the current list. This may be useful if a new disk array has been added but does not yet show up in the disk array list returned by the -i option.
amdsp(1M)

Display disk array state information. This includes cache settings for the disk array.

Raw output display. Data is output as a colon-delimited ASCII text string. Raw output format is near the end of this page.

Verbose mode displays additional command execution, state, and/or status messages.

Display extended usage message. This option overrides all others.

SECURITY CONFIGURATION
This command is modified for all security configurations.

Security Behavior/Restrictions
Use of this command is restricted to authorized users only.

Command Authorizations
This command requires the sysadmin authorization to successfully execute.

Privileges
The command has been modified to support least privilege. The potential privileges possessed by the command and their uses include:

- allowdacread: This privilege is raised to provide discretionary read access to the devices.
- allowdacwrite: This privilege is raised to provide discretionary write access to the devices.
- allowmacread: This privilege is raised to provide mandatory read access to the devices.
- allowmacwrite: This privilege is raised to provide mandatory write access to the devices.
- filesysops: This privilege is raised to allow the mknod(2) system call to succeed.
- writeaudit: The command generates its own audit records and submits these directly to the system audit trail. This privilege is raised whenever the command needs to write an audit record.

EXTERNAL INFLUENCES
Environment Variables
LC_MESSAGES determines the language in which messages are displayed. The current language settings can be checked with locale(1).

RETURN VALUE
amdsp returns the following values:

0  Successful completion.
1  An error in execution (I/O, subsystem, security, etc.) occurred.
2  An error in command syntax occurred: for example, an unknown command-line option was passed.
3  Timeout in communication to server. May indicate AM60Srvr is not running.

DIAGNOSTICS
The following messages can be generated by amdsp:


amdsp {-i | -R} [-V] [-S]

amdsp -p [-V] [-S] <DeviceFile>

Extended help: amdsp -?
An error in command syntax has occurred. Reenter the command with all necessary arguments.

amdsp: Arg out of range
One of the arguments has exceeded its maximum or minimum size, or is incorrect in form. Check the size and form of each argument.

amdsp: The <ArrayID> entered does not identify a known, supported array
The specified ArrayID does not exist or does not identify a device that is communicating with the system. Verify the array exists and is operational.

**amdsp**: The specified controller is not physically installed. A controller-specific command was issued to a non-existent or inaccessible controller.

**amdsp**: The specified LUN does not exist in this array. The specified LUN does not exist.

**amdsp**: No array controllers installed The software was unable to identify an installed controller in the array. This may be due to controller failure, or cabling problems. Correct the problem and re-issue the command.

**amdsp**: No controller hardware path was found for the given device file A controller-specific command was issued to a non-existent or inaccessible controller.

**amdsp**: Error in command execution, <Additional Error Info>

The command failed due to a device error, an internal error, or a system error. The Additional Error Info and Error Info Decode fields will hold specifics about the failure and its cause.

**EXAMPLES**

Display general information about disk array RACK_51:

```
amdsp RACK_51
```

Display information for LUN 2 on disk array named RACK_51:

```
amdsp -l 2 RACK_51
```

Display information for all LUNs on disk array RACK_51:

```
amdsp -l RACK_51
```

Display information for all disks installed in disk array named RACK_51:

```
amdsp -d RACK_51
```

List the serial numbers of all of the disk arrays recognized by the host:

```
amdsp -i
```

**RAW OUTPUT FORMAT**

The contents of the raw output text string for each display option are listed here. Note that the raw output for the -a option comprises the individual strings displayed by the -l, -d, -c, -s, -g, -A, -r, and -h options. For clarification when evaluating Boolean expressions, TRUE=1 and FALSE=0.

**OPTION RAW OUTPUT DESCRIPTION**

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The leading D is literal to identify this as a DEFAULT record.</td>
</tr>
<tr>
<td></td>
<td>All capacities are in GB where 1GB=1024^3 bytes, except LUN capacity, which is always in MB in raw output mode, where 1MB=1024^2 bytes.</td>
</tr>
</tbody>
</table>

```
-c:
```

The following is output for each disk array controller:

```
```

The following is output for each disk enclosure controller:

```
```
The leading C is literal to identify this as a CONTROLLER record. The above will be displayed for each controller installed. The first type of record shown above will be displayed for each Subsystem whose value is 0. The second record type will be displayed when the Subsystem is greater than 0.

Quiesced is a boolean field indicating whether the controller is quiescent. CntrlrMode, Quiesced, CntrlrDate, CntrlrTime, and CacheBattAge are only defined when the Subsystem is 0 (array controller enclosure). Otherwise they are undefined.

EncSerNum is the serial number of the disk subsystem enclosure. CntrlrTime is a colon delimited field having a format of HH:MM:SS.

If CompState is not GOOD (1), or CntrlrMode is FAILED (4), information in the remaining fields may be undefined or absent.

ALPA and PreferredALPA are displayed in hexadecimal. CacheBattAge is the time in days (rounded to the nearest 90 days) since the cache battery age was last reset.

The following output is provided for each installed disk:

```
PD:Channel:ID:EnclosureID:Slot:ID:DiskState:DiskGroupID:DGType:
Capacity:Manufacturer:Model:FWRevision:SerialNum:ArrayID
```

The leading PD is literal to identify this as a PHYSICAL DISK record. The above will be displayed for each disk slot. For disks which have been bound to a LUN and then removed, or assigned as a hot spare and then removed, the Manufacturer, Model, FWRevision, and SerialNum fields may be undefined.

Unassigned disks which are removed will have no PD record.

Capacity is displayed in GB, rounded up to the nearest GB. If DGType is 1 (HOT SPARE), and if this disk is currently sparing another disk, then two extra fields will be printed, designating the channel and SCSI ID of the spared disk.

One or more LUN records will be displayed in the following format:

```
G:VendorID:ProductID:DiskGroupID:DGType:
NumLUNs:LUN...:RemCapacity:RAIDLevel:SegmentSize:nDisks:
Channel-ID,Channel-ID...:ArrayID
```

The leading G is literal to identify this as a Disk Group record. The above fields will be displayed for each disk group.

The following fields are undefined when DGType is 1 or 2: NumLUNs, LUN, RemCapacity, RAIDLevel, and SegmentSize.

Note that capacity will always be expressed in MB for raw output.

One or more "component state records" will be displayed in the following format:

```
H:VendorID:ProductID:SubSystem:CntrlrAStatus:CntrlrBStatus:
PwSupplyAStatus:PwSupplyBStatus:FanAStatus:FanBStatus:
TsensorStatus:BattStatus:EncSerNum:ArrayID
```

The leading H is literal to identify this as a Hardware Status record.
When the `SubSystem` is greater than 0 (disk enclosure), the `BattStatus` is undefined.

When the `SubSystem` is greater than 0 (disk enclosure), the power supplies and fans are designated as A and B, rather than 1 and 2. The left to right orientation of the disk enclosure power supplies may not match that of the array controller power supplies.

`EncSerNum` is the disk enclosure serial number, and is only defined when the `SubSystem` is greater than 0; otherwise, it is undefined.

The following output includes a separate field (`ArrayID:Alias`) for each disk array identified:

```
I:ArrayID:Alias:ArrayID:Alias:...
```

The leading `I` is literal to identify this as an Identify record.

```
L:VendorID:ProductID:LunNumber:Present:LunState:
  CacheState:Capacity:Owner:RaidLevel:SegmentSize:
  DiskGroup:nDisks:Channel-ID,Channel-ID,...:ArrayID:
```

The leading `L` is literal to identify this as a LUN record.

`Present` is a Boolean, TRUE if the LUN is configured, FALSE otherwise.

If no option is specified, the above will be displayed for all LUNs, both present and not present.

`CacheState` is a two-bit field in which the high order bit is the boolean `CacheMirroringActive` (CMA) state of the LUN's owning controller, and the low order bit is the boolean `WriteCacheEnabled` (WCE) state for the controller.

`CME` (Cache Mirroring Enabled), `CWOB` (Cache Without Batteries), `RCD` (Read Cache Disabled), `WCA` (Write Cache Active), and `RCA` (Read Cache Active) indicate the state of the caching parameters.

Capacities displayed in the raw output are always shown in MB, where 1MB = 1024^2 bytes.

`IsUtm` is a boolean that indicates whether the current LUN is the Universal Transport Mechanism (UTM). When `IsUtm` is TRUE (1), only the following additional fields are valid: `VendorID`, `ProductID`, `LunNumber`, `Present`, and `UtmLunState`. The remaining fields are undefined.

When `IsUtm` is TRUE (1), `UtmLunState` indicates the status of the UTM LUN, as defined in the UTM Specification. Otherwise this field is undefined.

If `nDisks` is "0", the drive list (Chan-ID,Chan-ID,...) will also be "0".

```
P:CntrlrID:SerNumber:ArrayID
  P:AltCntrlrID:AltSerNumber:ArrayID
```

The leading `P` is literal to identify this as a HARDWARE PATH record.

The first hardware path record is for the controller that corresponds to the given device file. The second record is for the alternate controller. If the alternate controller is not present, its serial number will be 0.

```
R:VendorID:ProductID:CntrlrID:LUN:RebuildProgress:Freq:
  Amt:ArrayID
```

The leading `R` is literal, and identifies this as a LUN rebuild record.

One rebuild record is returned for each LUN that is currently rebuilding. If no LUN is rebuilding, no record will be returned.

`RebuildProgress` above is percent complete.

`Freq` is the rebuild frequency in tenths of a second.

`Amt` is the rebuild amount in blocks per rebuild command.

```
S:VendorID:ProductID:ValidData-A:ValidData-B:
  CacheBlkSize-A:CacheBlkSize-B:CacheFlushThresh-A:
    CacheFlushThresh-B:CacheFlushLim-A:CacheFlushLim-B:
      CacheSize-A:CacheSize-B:ArrayState:ArrayID
```

The leading `S` is literal to identify this as an ARRAY STATE record.
DEPENDENCIES

`AM60Srvr` must be running to execute this command. See `AM60Srvr(1M)`.

AUTHOR

`amdsp` was developed by HP.

SEE ALSO

`ammgr(1M), amcfg(1M), amutil(1M), amlog(1M), AM60Srvr(1M)`.
NAME
amfmt - disk format, not supported

DESCRIPTION
The amfmt command is a data destructive command and is not supported. If an attempt is made to execute the amfmt command, data may be destroyed.

AUTHOR
amfmt was developed by HP.

SEE ALSO
ammgr(1M), amutil(1M), amcfg(1M), amdsp(1M), amlog(1M), amdload(1M), AM60Srvr(1M).
NAME
amlog - displays host-based controller log entries for a disk array

SYNOPSIS
amlog [-s StartTime] [-e EndTime] [-t <RecordType,RecordType...>] [-c] [-d LogDir] [-S]
     [-V] [-a ArrayID]
amlog -?

DESCRIPTION
amlog displays the contents of the disk array controller logs maintained by the host. These logs contain
information useful for diagnosing and troubleshooting the disk array. The host maintains multiple log files
containing entries for each disk array.

The ArrayID used to address the disk array must be the disk array ID. An alias name cannot be used
because alias names are not recorded in the log.

Identifying Disk Modules
Disk modules are identified within Array Manager 60 using a numbered pair of the form n:n. The first
number identifies the SCSI channel (or bus) connecting the array controller to the enclosure containing the
disk module. The channel number is indicated on the back of the array controller enclosure. The second
number is the disk module SCSI ID. The SCSI ID is determined by the slot in which the disk module is
installed, but is not the same as the physical slot number (0-9).

For example, the numbered pair 2:1 identifies the disk module on channel 2 with a SCSI ID of 1. Refer to
the Disk Array FC/60 User’s Guide for more information on disk module addressing.

Options
amlog supports the following options:
- a ArrayID Identify a specific disk array for which the logs will be displayed. The default is to
display the logs for all disk arrays.
- c Limits major event log output to events of critical priority. This option is ignored
   unless specified in conjunction with -t mel.
- d LogDir Specify the path name of the log directory. The default is
   /var/opt/hparray/log.
- e EndTime Specify the ending date and time. Log records with a later date and time will not be
   printed. The default is the time of the last log record. Uses the same format as Start-
   Time.
- s StartTime Specify the starting date and time. Log records with an earlier date and time will not
   be printed. The default is the time of the oldest log record. The format for entering
   the date and time is MMddhhmm[yy], where
   MM = Month (01-12)
   dd = Day (01-31)
   hh = Hour (00-23)
   mm = Minute (00-59)
   yy = Year (00-99) [optional]. Years earlier than 90 are interpreted as 2000 + YY.
- S Raw output display. Data is output as a colon-delimited ASCII text string. Raw output
   format is described below.
- t RecordType Specify the log record types to be displayed. Valid record types are ctrlr (controller
   log sense) and mel (major event log). The default selection is ctrlr. Major event
   logging is available in firmware versions HP07 and later.
- V Verbose output, displays additional "raw" major event log data, which can only be
   interpreted via the MEL Specification. This option is ignored unless specified in con-
   junction with -t mel.
- ? Display expanded usage message. This option overrides all others.
SECURITY CONFIGURATION

This command is modified for all security configurations.

Security Behavior/Restrictions

Use of this command is restricted to authorized users only.

Command Authorizations

This command requires the sysadmin authorization to successfully execute.

Privileges

The command has been modified to support least privilege. The potential privileges possessed by the command and their uses include:

- **allowdacread**: This privilege is raised to provide discretionary read access to the devices.
- **allowdacwrite**: This privilege is raised to provide discretionary write access to the devices.
- **allowmacread**: This privilege is raised to provide mandatory read access to the devices.
- **allowmacwrite**: This privilege is raised to provide mandatory write access to the devices.
- **filesysops**: This privilege is raised to allow the mknod(2) system call to succeed.
- **writeaudit**: The command generates its own audit records and submits these directly to the system audit trail. This privilege is raised whenever the command needs to write an audit record.

EXTERNAL INFLUENCES

Environment Variables

- **LC_MESSAGES**: Determines the language in which messages are displayed. The current language settings can be checked with locale(1).
- **AM60_MAX_LOG_SIZE_MB**: Defines the maximum size (in megabytes) allocated for storage of the disk array log files. The log files are stored as individual files in /var/opt/hparray/log. All log files will be maintained until the specified maximum value is exceeded. At that point, the oldest log file will be deleted to make room for the new file (FIFO).

The valid range for this variable is 100 to 4096 (100 MB to 4 GB). If a value outside of this range is specified, it will be ignored, and the default value will be used. The default value for allocated log storage area is 100 MB. When setting a value, make sure the /var file system is large enough to accommodate the logs files. If it is not, the log files may overflow the file system.

RETURN VALUE

- **amlog** returns the following values:
  - 0: Successful completion.
  - 1: An error in execution (I/O, subsystem, security, etc.) occurred.
  - 2: An error in command syntax occurred: for example, an unknown command-line option was passed.

DIAGNOSTICS

The following message can be generated by amlog:


Extended help: amlog -?

An error in command syntax has occurred. Re-enter the command with all necessary arguments.

EXAMPLES

Display the controller log for disk array 000200A0B805E798. Display the log entries entered since 6/1/99.

```
   amlog -s 0601000099 -a 000200A0B805E798
```

Display the controller log for all disk arrays on the host. Use the default settings to display all log entries.

```
   amlog
```
RAW OUTPUT FORMAT

The contents of the raw output text string for log entries is as follows:

**Default Log Output**

```
AL:LogFilename:AddSenAvailable:SenseDate:SenseTime:FRUCode:
FRUCodeQual:SK:ASC:ASCQ:ArrayID:LogDate:LogTime:LUN
```

One record will be printed for each log entry between the start and end times.

*LogFilename* is the name of the server log file where the record is found.

The leading **AL** is literal, and identifies this as an AM60 LOG record.

*AddSenAvailable* is a boolean that indicates whether the *SenseDate*, *SenseTime* and *FRUCodeQual* are available from the SCSI sense data that generates these fields. If FALSE (0), these fields will be undefined.

*SenseDate* and *SenseTime* correspond to the time of the reported event, while *LogDate* and *LogTime* correspond to the time when the event was written to the log file. The format of the *SenseDate* is **MMDDYY**, while that of *LogDate* is **MMDDYYYY**. The format for the *SenseTime* and *LogTime* is **HHMMSS**.

The SCSI sense data fields, *FRUCode*, *SK*, *ASC*, *ASCQ*, are one byte hexadecimal; the *FRUCodeQual* field is two bytes hexadecimal.

The LUN field will be appended to the output only if the LUN information in the sense data is valid.

**Major Event Log Output**

```
AM:LogFilename:ArrayID:LogMonth:LogDay:LogYear:LogHour:LogMin:
LogSec:SeqNum:EventType:EventMonth:EventDay:EventYear:
EventHour:EventMin:EventSec:EventCategory:CompType:
LocationValid:CompLocation:EventPriority:SK:ASC:ASCQ:FRUCode:
FRUCodeQual:LUN
```

One record will be printed for each log entry between the start and end times.

*LogFilename* is the name of the server log file where the record is found.

The leading **AM** is literal, and identifies this as an AM60 LOG record.

The *Log* timestamp indicates when this event was written to the log file.

The *Event* timestamp indicates when this event occurred.

*LocationValid* is a boolean that indicates whether the *CompLocation* field is valid.

If *LocationValid* is TRUE (1), *CompLocation* indicates an enclosure and, if applicable, a slot. Otherwise, this field is undefined. If both an enclosure and slot are indicated, they will be separated by a comma. The *CompType* field determines whether the *CompLocation* field will include a slot number.

*SK*, *ASC*, and *ASCQ* are SCSI sense data, each one byte hexadecimal.

If `-c` is specified, only MEL events with an *EventPriority* of CRITICAL will be displayed.

The final three fields (*FRUCode*, *FRUCodeQual*, and *LUN*) are only displayed if the *EventType* is 0x3101 (Asynchronous Event Notification or AEN). *FRUCode* and *FRUCodeQual* are one and two bytes hexadecimal, respectively. The *LUN* field is only displayed if valid LUN information is available in the AEN sense data.

**DEPENDENCIES**

There are no dependencies for this command.

**AUTHOR**

`amlog` was developed by HP.

**SEE ALSO**

`amdspl(1M), amcfg(1M), amutil(1M), ammgr(1M), AM60Srvr(1M)`.
NAME
ammgr - manages the operating characteristics of the disk array

SYNOPSIS
ammgr -D ArrayAlias [-V] [-?] ArrayID
ammgr [Options] [-V] [-?] ArrayID

DESCRIPTION
ammgr manages the operating characteristics of the disk array by providing access to the settings used to
control disk array operation. In most cases, these settings control the operation of the entire disk array;
consequently, every LUN on the disk array will be affected by any changes made using this command. The
exception is a parity scan, which impacts only the specified LUN.

The ArrayID used to address the disk array can be the disk array serial number or name, if one has been
assigned to the disk array.

Identifying Disk Modules
Disk modules are identified within Array Manager 60 using a numbered pair of the form n:n. The first
number identifies the SCSI channel (or bus) connecting the array controller to the enclosure containing the
disk module. The channel number is indicated on the back of the array controller enclosure. The second
number is the disk module SCSI ID. The SCSI ID is determined by the slot in which the disk module is
installed, but is not the same as the physical slot number (0-9).

For example, the numbered pair 2:1 identifies the disk module on channel 2 with a SCSI ID of 1. Refer to
the Disk Array FC/60 User's Guide for more information on disk module addressing.

Options
ammgr supports the following options:

- Reset battery age to zero. This should be done when the battery is replaced.
- c AA Set the controller mode to dual-active.
- d channel:ID Remove the role of hot spare for the disk at address channel:ID. The disk is identified
  by channel number (1-6) and SCSI ID (0-4, 8-12).
- D ArrayAlias Assigns the text string specified by ArrayAlias to the disk array. This name is
  another mechanism for identifying the disk array when executing a command. The
  name can be up to 16 characters in length and can include letters, numbers, number
  sign (#), underscore (_), and period (.).

  Disk names can be used in a variety of ways to help identify disk arrays in large sys-
  tems. For example, by assigning numbers to racks and to the shelf positions within
  the racks, each disk array can be uniquely identified using an appropriate alias. If a
  rack is assigned number 12, the disk array installed on shelf 3 of the rack could be
  identified using an alias of 12_03. This technique simplifies locating the disk array
  should it need service.

- h channel:ID Assign the role of hot spare to the disk at address channel:ID. The disk is identified
  by channel number (1-6) and SCSI ID (0-4, 8-12).
- H LUN Halt a parity scan currently in progress on the specified LUN. If a parity scan is not
  in progress on the specified LUN, clear the status of the last completed parity scan for
  the LUN.

- L cntrlrID:percent Sets the cache flush limit for the controller identified by cntrlrID to the value specified
  by percent. When flushing write cache pages to the disk media, the flush operation
  will stop when the indicated percentage is reached. The value is expressed as a per-
  centage of the current cache flush threshold.

- p [4 | 16] Set the cache page size to 4 KB or 16 KB. The cache page size is set for both disk
  array controllers. NOTE: The cache page size may not be set to 16 KB unless the
  segment sizes of all existing LUNs are multiples of 16 KB.

- P LUN Perform a parity scan on the specified LUN.
NOTE: If errors are detected during a parity scan, it is recommended that you contact your Hewlett-Packard service representative immediately. The occurrence of parity errors may indicate a potential problem with the disk array hardware.

-s LUN
Show the status of the parity scan in progress on the specified LUN. The final status continues to be displayed until cleared using the -H option.

-t
Set the date and time stamp for both disk array controllers to match that of the host.

-T cntrlrID:percent
Sets the cache flush threshold for the controller identified by cntrlrID to the value specified by percent. When the contents of the write cache reaches the indicated percentage, pages are flushed (written) to the disk media.

-V
Verbose mode displays additional command execution, state, and/or status messages.

-?
Display extended usage message. This option overrides all others.

SECURITY CONFIGURATION
This command is modified for all security configurations.

Security Behavior/Restrictions
Use of this command is restricted to authorized users only.

Command Authorizations
This command requires the sysadmin authorization to successfully execute.

Privileges
The command has been modified to support least privilege. The potential privileges possessed by the command and their uses include:

allowdacread This privilege is raised to provide discretionary read access to the devices.
allowdacwrite This privilege is raised to provide discretionary write access to the devices.
allowmacread This privilege is raised to provide mandatory read access to the devices.
allowmacwrite This privilege is raised to provide mandatory write access to the devices.
filesysops This privilege is raised to allow the mknod(2) system call to succeed.
writeaudit The command generates its own audit records and submits these directly to the system audit trail. This privilege is raised whenever the command needs to write an audit record.

EXTERNAL INFLUENCES
Environment Variables
LC_MESSAGES determines the language in which messages are displayed. The current language settings can be checked with locale(1).

RETURN VALUE
ammgr returns the following values:

0 Successful completion.
1 An error in execution (I/O, subsystem, security, etc.) occurred.
2 An error in command syntax occurred: for example, an unknown command-line option was passed.
3 Timeout in communication to server. May indicate AM60Srvr is not running.

DIAGNOSTICS
The following messages can be generated by ammgr:

Usage: ammgr (-D <ArrayAlias> | -b | -c AA | -h <Channel:ID> | -d <Channel:ID> | -T <CntrlrID>:<percent> | -L <CntrlrID>:<percent> | -p <pagesize> | -P <LUN> | -H <LUN> | -s <LUN> | -t )
[-V] [-S] <ArrayID>

Extended help: ammgr -?
An error in command syntax has occurred. Re-enter the command with all necessary arguments.

ammgr: Arg out of range
One of the arguments has exceeded its maximum or minimum size, or is incorrect in form. Check the size and form of each argument.

ammgr: The <ArrayID> entered does not identify a known, supported array
The specified ArrayID does not exist or does not identify a device that is communicating with the system. Verify the array exists and is operational.

ammgr: The specified controller is not physically installed.
A controller-specific command was issued to a non-existent or inaccessible controller.

ammgr: The specified LUN does not exist in this array.
The specified LUN does not exist.

ammgr: Invalid Array alias. Alias name characters must be alphanumeric, pound (#), underscore (_), or dot (.)
Illegal syntax for <ArrayID>.

ammgr: Invalid array alias. Another array already has the specified alias.
Another array on the host already has the specified alias. Duplicate aliases are not allowed.

ammgr: Invalid array alias. Alias name cannot be longer than 16 characters.
An attempt was made to set an array alias with more than the maximum allowed number of characters.

ammgr: A parity scan is already in progress on the specified LUN.
An attempt was made to initiate a parity scan on a LUN which already has a parity scan in progress.

ammgr: No parity scan in progress on the specified LUN.
An attempt was made to halt a parity scan on a LUN which has no parity scan in progress.

ammgr: Cache page size may not be set to 16 KB unless the segment sizes of all existing LUNs are multiples of 16 KB.
Illegal cache page size requested. Segment sizes of existing LUNs must be multiples of the requested cache page size.

ammgr: Error in command execution, <Additional Error Info>:
<Error Info Decode>
The command failed due to a device error, an internal error, or a system error. The Additional Error Info and Error Info Decode fields will hold specifics about the failure and its cause.

EXAMPLES
Assign a name to the disk array identified with ID 0000005EBD20. Use an alias that identifies the rack the disk array is installed in (51 for this example), and the shelf position the disk array is on (03 for this example):

ammgr -D RACK_51 0000005EBD20

Add a hot spare to disk array RACK_51 using the disk on channel 3 with SCSI ID 4:

ammgr -h 3:4 RACK_51

Initiate a parity scan of LUN 1 on disk array RACK_51:

ammgr -P 1 RACK_51

Set the cache flush threshold to 80% for controller A on disk array RACK_51:

ammgr -T A:80 RACK_51

DEPENDENCIES
AM60Srvr must be running to execute this command. See AM60Srvr(1M).

ammgr(1M)

AUTHOR

ammgr was developed by HP.

SEE ALSO

amdsp(1M), amcfg(1M), amutil(1M), amlog(1M), AM60Srvr(1M).
NAME
amutil - controls various disk array management functions

SYNOPSIS
amutil [ [ -f channel:ID,channel:ID... ] | [ -F ] | [ -s ] | [ -R LUN -f Freq -a Amt ] ]
[ -p ] | [ -l ] | [ -V ] ArrayID
amutil -?

DESCRIPTION
amutil controls a variety of management operations on the disk array identified by ArrayID. Locating
hardware components and managing the rebuild process are done using amutil.
The ArrayID used to address the disk array can be the disk array serial number or name, if one has been
assigned to the disk array.

Identifying Disk Modules
Disk modules are identified within Array Manager 60 using a numbered pair of the form n:n. The first
number identifies the SCSI channel (or bus) connecting the array controller to the enclosure containing the
disk module. The channel number is indicated on the back of the array controller enclosure. The second
number is the disk module SCSI ID. The SCSI ID is determined by the slot in which the disk module is
installed, but is not the same as the physical slot number (0-9).

For example, the numbered pair 2:1 identifies the disk module on channel 2 with a SCSI ID of 1. Refer to
the Disk Array FC/60 User's Guide for more information on disk module addressing.

Options
amutil supports the following options:
- f channel:ID,channel:ID...
  Flash the amber LED on the specified disks. Each disk is identified by channel:ID.
  Channel:ID is determined by the disk enclosure channel number (1-6) and disk SCSI
  ID (0-4, 8-12).
- F
  Flash the amber LEDs on all the disks in the disk array.
- l
  Flush server log file. This will retrieve the current log entries from the disk array
  controller. The controller logs will be cleared.
- p
  Purge the oldest log file (delete the log file and remove its entry from the log file cata-
  log). Always use this option to delete log files. Using a system command such as
  rm to delete log files will cause log catalog errors.
- R LUN -f req -a Amt
  Set the rebuild priority settings of the specified LUN. The freq value identifies the
  rate at which the disk array attempts to execute rebuild commands. Specified in
tenths of a second, this value can be 1 to 50, or 0.1 seconds to 5.0 seconds. A low set-
ing increases the frequency at which rebuild commands are issued, giving higher
priority to the rebuild but reducing I/O performance. A high value reduces the rebuild
command frequency, giving higher priority to host I/Os. The default value for this set-
ing is 1, or 0.1 seconds.
  The Amt value identifies the number of blocks to rebuild at a time. This value can be
  from 1 to 64K and specifies the number of 512-byte blocks processed during each
  rebuild command. The higher the setting the more blocks which will be processed,
  thus reducing I/O performance. A lower setting gives priority to host I/Os, delaying
  the completion of the rebuild. The default value for this setting is 64 blocks, or 32
  Kbytes of data.
- s
  Stop flashing disk activity lights. This option is used to stop the flashing on all disks.
- V
  Verbose mode displays additional command execution, state, and/or status messages.
- ?
  Display extended usage message. This option overrides all others.

SECURITY CONFIGURATION
This command is modified for all security configurations.
Security Behavior/Restrictions
Use of this command is restricted to authorized users only.

Command Authorizations
This command requires the sysadmin authorization to successfully execute.

Privileges
The command has been modified to support least privilege. The potential privileges possessed by the command and their uses include:
- allowdacread: This privilege is raised to provide discretionary read access to the devices.
- allowdacwrite: This privilege is raised to provide discretionary write access to the devices.
- allowmacread: This privilege is raised to provide mandatory read access to the devices.
- allowmacwrite: This privilege is raised to provide mandatory write access to the devices.
- filesysops: This privilege is raised to allow the mknod(2) system call to succeed.
- writeaudit: The command generates its own audit records and submits these directly to the system audit trail. This privilege is raised whenever the command needs to write an audit record.

EXTERNAL INFLUENCES
Environment Variables
LC_MESSAGES determines the language in which messages are displayed. The current language settings can be checked with locale(1).

RETURN VALUE
amutil returns the following values:
- 0 Successful completion.
- 1 An error in execution (I/O, subsystem, security, etc.) occurred.
- 2 An error in command syntax occurred: for example, an unknown command line option was passed.
- 3 Timeout in communication to server. May indicate the AM60Srvr is not running.

DIAGNOSTICS
The following messages can be generated by amutil:
Usage: amutil {-f <Channel:ID>[,<Channel:ID>...] | -F | -l | -s | -R <LUN> -f <Freq> -a <Amt>} [-V] <ArrayID>}
amutil -p [-V]
Extended help: amutil -?
An error in command syntax has occurred. Reenter the command with all necessary arguments.

amutil: Arg out of range
One of the arguments has exceeded its maximum or minimum size, or is incorrect in form. Check the size and form of each argument.

amutil: The <ArrayID> entered does not identify a known, supported array
The specified ArrayID does not exist or does not identify a device that is communicating with the system. Verify the array exists and is operational.

amutil: The specified controller is not physically installed.
A controller-specific command was issued to a non-existent or inaccessible controller.

amutil: The specified LUN does not exist in this array.
The specified LUN does not exist.

amutil: Error in command execution, <Additional Error Info>:
<Error Info Decode>
The command failed due to a device error, an internal error, or a system error. The Additional Error Info and Error Info Decode fields will hold specifics about the failure and its cause.
amutil(1M)

EXAMPLES
Flash the amber LEDs on all the disks in disk array RACK_51:

   amutil -F RACK_51

Stop flashing the amber LEDs on disk array RACK_51:

   amutil -s RACK_51

The following example alters the rebuild priority settings. It assigns a value of 5 seconds to the rebuild command rate, and sets the data block amount to 16 blocks on LUN 4 on disk array RACK_51. This gives host I/Os higher priority than the default settings.

   amutil -R 4 -f 50 -a 16 RACK_51

DEPENDENCIES
AM60Srvr must be running to execute this command. See AM60Srvr(1M).

AUTHOR
amutil was developed by HP.

SEE ALSO
ammgr(1M), amcfg(1M), amdsp(1M), amlog(1M), AM60Srvr(1M).
arp(1M)

NAME
arp - address resolution display and control

SYNOPSIS

arp
arp hostname
arp -a
[system]
arp -a [system [core]] # PA only
arp [-d | -D] hostname
arp -f filename
arp -s hostname hw_address [temp] [pub] [rif rif_address]
arp -sfc hostname nport_id

DESCRIPTION
The arp command displays and modifies the Internet-to-Ethernet and Internet-to-Fibre Channel address translation tables used by the Address Resolution Protocol (ARP).

Options

arp has the following keyletter options:

hostname
Display the current ARP entry for hostname, which must appear in the hostname database (see hosts(4)), or for the DARPA Internet address expressed in Internet standard “dot” notation.

-a[n]
Display all current ARP entries by reading the table from /dev/kmem based on the kernel file system (default /stand/vmunix). On PA systems only: if a core argument is supplied, read the table from core instead of /dev/kmem. The -an option provides the same information as the -a option except that network addresses are not displayed symbolically.

-d
If an ARP entry exists for the host called hostname, delete it. This option cannot be used to delete a permanent ARP entry whose IP address is an interface on the local system.

-D (Not recommended). Delete a permanent ARP entry whose IP address is an interface on the local system. The removal of such an ARP entry may result in loss or limitation of network connectivity with remote machines. The local system will no longer respond to ARP requests for this IP address. Consequently, communication with remote systems is possible only when that communication is initiated by the local system. This option should be used with extreme caution.

-f
Read file filename and set multiple entries in the ARP tables. Fibre Channel entries in the file should be of the form:

-sfc hostname nport_id

Other entries in the file should be of the form:

hostname hw_address
[temp]
[pub]
[rif
rif_address]

The argument meanings are the same as for the -s option.

-s
Create an ARP entry for the host called hostname with the hardware station address hw_address. The hardware station address is given as six hexadecimal bytes separated by colons. If an ARP entry already exists for hostname, the existing entry is updated with the new information.

The entry is permanent unless the word temp is given in the command.

If the word pub is specified, the entry is published, which means that this system will act as an ARP server responding to requests for hostname even though the host address is not its own.

The word rif specifies source routing information used for token ring networks. This information allows you to specify the particular bridge route which the token ring packet
should be delivered. rif_address is given as an even number of hexadecimal bytes separated by colons, up to a maximum of 16 bytes.

- sfc  Create a permanent ARP entry for the Fibre Channel host called hostname with the N_Port address nport_id. The N_Port address is given as three hexadecimal bytes separated by colons. If an ARP entry already exists for hostname, the existing entry is updated with the new information.

You need superuser privilege to use the -d, -D, -f, -s and -sfc options.

See the Neighbor Discovery Protocol (ndp(1M)) for IPv6.

AUTHOR
arp was developed by HP and the University of California, Berkeley.

SEE ALSO
ifconfig(1M), inet(3N), hosts(4), arp(7P), ndp(1M).
NAME
asecure - control access to Audio on a workstation

SYNOPSIS
/opt/audio/bin/asecure [-C | -d | +h host | -h host | +p user | -p user | +u user | -u user | +b host, user | -b host, user]

DESCRIPTION
On Series 700 workstations, audio is secured so that only the user on the local workstation can access audio. You use the asecure command to modify audio security. This command does not apply to X stations; on an X station, access to audio is unrestricted.

To modify audio security, become root on the local workstation where you want make a change. Then, use asecure as follows:

/opt/audio/bin/asecure -C
When prompted, enter any meaningful password. Issuing asecure -C creates the Audio Security File (ASF). The ASF contains information that determines which hosts and users can access the Aserver, and which users (other than the superuser) can modify the ASF.

If needed, you can allow unrestricted access to audio on this workstation. To remove audio security, issue this command:

/opt/audio/bin/asecure -d
If instead, you wish to modify security, you use asecure to make changes to the information in the ASF. (Because the ASF is a binary file, we do not recommend using an editor on this file.) You can use asecure to make these types of changes:

- Allow all clients from a remote host to access the server.
- Allow specific users from all other hosts to access the server.
- Allow a specific user from a specific host to access the server.
- Disable access control, allowing complete unrestricted access to the server, but leaving the ASF intact.

Every operation that creates, reinitializes, or changes the contents of the ASF is logged in the /var/adm/audio/asecure_log file, so that you can track any changes to the ASF.

Options
asecure supports the following options:

- Add/delete hostname,username pair. You must be either superuser or a privileged user to do this. You can supply more than one hostname,username pair separated by blanks.
     +b | -b host, user
To use either the +b or -b options, you MUST supply at least one hostname,username pair. This option will not work without a pair.

- Create a new ASF file, called the audio.sec file. Access control default is enabled with no entries in the access list. Aserver can now be accessed only by local users on the host machine. If an audio.sec file already exists, it is re-initialized.
     -C
You must be superuser to execute this option. This option is mutually-exclusive of all other options.
This option requires a password. This is an extra layer of protection for the contents of the ASF. It is designed to prevent surreptitious manipulation of the ASF. If you are creating a new ASF, you are prompted for a password and an encrypted copy of that password is stored in the new ASF.
If the ASF already exists, you are prompted for the password. If your password matches the password stored in the ASF, the ASF is then re-initialized.

- Disable access control to the Aserver. This allows unrestricted access by all clients.
     -d

- Enable access control to the Aserver. This restricts access to clients listed in the ASF. Enabled is the default state.
     -e
asecure(1M)

+h|-h host  Add/delete hostnames for ALL users. You must be either superuser or a privileged user to do this. You can supply more than one hostname separated by blanks.

-l  List the contents of the ASF. This option shows a list of the hostnames and/or usernames that have access to the Aserver.

-P  Change password for audio.sec file. You must be superuser to do this. You are prompted once for the old password, then prompted twice for the new password.

+p|-p user  Add/delete privileged users. You must be superuser to do this and must enter the password given when the ASF was created (see -C option). To see a list of privileged users, you must be superuser and use the -l option.

+u|-u user  Add/delete usernames for ALL hosts. You must be either superuser or a privileged user to do this. You can supply more than one username separated by blanks.

EXAMPLES
List entries in access list.

/opt/audio/bin/asecure -l

Disable access control. This means anyone can connect to Aserver without restriction.

/opt/audio/bin/asecure -d

Add moonbeam host for all users to access list. Remove pluto host for all users from access list.

[opt/audio/bin/asecure +h moonbeam -h pluto

Add user comet for hosts saturn and mercury to access list.

/opt/audio/bin/asecure +b saturn,comet mercury,comet

Add user comet to access list for all hosts. Remove users venus and neptune from access list for all hosts.

/opt/audio/bin/asecure +u comet -u venus neptune

Create new access list.

/opt/audio/bin/asecure -C

AUTHOR
asecure was developed by HP.

FILES
/var/opt/audio/asecure_log  asecure log pathname
/etc/opt/audio/audio.sec  ASF pathname

SEE ALSO
audio(5), asecure(1M), aserver(1M), attributes(1), convert(1), send_sound(1).

Using the Audio Developer's Kit
aserver(1M)  aserver(1M)

NAME
  Aserver - start the audio server

SYNOPSIS
  /opt/audio/bin/Aserver -f

DESCRIPTION
  The Aserver command starts the HP-UX Audio server, which can run on a system with audio hardware. See Audio(5) for information about which systems have audio hardware. The -f option forces the starting of the Audio server; this option is only needed if the Aserver has problems starting.

  The Audio Server
  Before using any audio tools such as the Audio Editor, the system or X station must be running two audio server processes, called Aserver. On a Series 700, the Remote Procedure Call daemon (rpcd) must also be running.

  Normally, the Aserver processes and rpcd start automatically when the system is booted. If problems occur on an ENTRIA or ENVIZEX X station, see the X station owner's manual. On a Series 700 Audio hardware, first check if rpcd is running. Type the following:

  ps -e | grep rpcd

  If it is running, you see a line similar to the following.

  604 ? 0:36 rpcd

  If it is not running, see HP 9000/DCE documentation for information on restarting it. If rpcd is running, verify that the Aserver is running. Type:

  ps -e | grep Aserver

  If the Aserver is running you will see lines similar to the following, which indicate the presence of the two Aserver processes:

  1 ? 0:00 Aserver
  224 ? 0:00 Aserver

  If it is not running, become root and restart it as follows:

  /opt/audio/bin/Aserver

  If it fails to start, reissue the command with the -f option:

  /opt/audio/bin/Aserver -f

Using Audio over the Network
  From a workstation, you can also use the Audio Editor and Control Panel over the network. However, the remote system is where the actual playback and recording occur.

  The local workstation (or audio client) can be any Series 700 system. The remote system (or audio server) can be a Series 700 or an X station with audio hardware and must have the Aserver processes running. If the server is a workstation, it must also allow access from remote clients (see asecure(1M)) and must have rpcd running.

  To make the system an audio client, set the AUDIO variable by modifying the $HOME/.vueprofile file as follows:

  Korn and POSIX Shells:

  AUDIO=system_name; export AUDIO

  C Shell:

  setenv AUDIO system_name

  For system_name, identify the workstation or X Station running the Aserver.

  If the AUDIO variable is not set, the Audio Library attempts to use the Aserver on the system defined by the DISPLAY variable. If neither DISPLAY nor AUDIO is set, the Aserver on the local machine is used.

DEPENDENCIES
  The Audio Server must run on a system that has audio hardware. Note that HP-UX for the 8MB 705 System does not include audio software.
**aserver(1M)**

**AUTHOR**
The Audio Server was developed by HP.

**SEE ALSO**
audio(5), asecure(1M), attributes(1), convert(1), send_sound(1).

*Using the Audio Developer's Kit*
audevent(1M)  audevent(1M)

NAME
audevent - change or display event or system call audit status

SYNOPSIS
audevent [ -P -p ] [ -F -f ] [ -E ] [ [-e event ] ... ] [ -S ] [ [-s syscall ] ... ]
audevent [ -l ]

DESCRIPTION
audevent changes or displays the auditing status of the given events or system calls. The event is used to specify names associated with certain self-auditing commands; syscall is used to select related system calls.

If neither -P, -p, -F, nor -f is specified, the current status of the selected events or system calls is displayed.

If the -E option is supplied, it is redundant to specify events with the -e option. This also applies to the -S and -s options. If no event is specified, all events are selected. If no system call is specified, all system calls associated with the selected events are selected.

audevent takes effect immediately. However, the events and system calls specified are audited only when called by a user currently being audited (see audusr(1M)).

If -l is specified, a list of valid events and their associated system calls (if any) are displayed. This option may be helpful when deciding which -e or -s options to use.

Note: The set of audited system calls and corresponding audit events varies frequently as HP-UX evolves. The system call name referred to by the auditing system usually matches the real system call name, but with a few exceptions. Some important known exceptions are provided in System Call Name Mapping Exceptions.

Only the super-user can change or display audit status.

Options
audevent recognizes the following options and command-line arguments:

- P
  Audit successful events or system calls.
- p
  Do not audit successful events or system calls.
- F
  Audit failed events or system calls.
- f
  Do not audit failed events or system calls.
- E
  Select all events for change or display.
- e event
  Select event for change or display.
- S
  Select all system calls for change or display.
- s syscall
  Select syscall for change or display.
- l
  Display a list of valid events and their associated system calls. This option should not be used with any other options.

The following is a list of the valid event types or categories:

create
  Object creation. For example, file creation, directory creation, and other object creation.
delete
  Object deletion. For example, file deletion, directory deletion, and other object deletion.
readdac
  Discretionary access control (DAC) information reading events.
moddac
  DAC modification events.
modaccess
  Non-DAC modification events.
open
  Object opening. For example, file open and other object open.
close
  Object closing. For example, file close and other object close.
process
  Process operations.
**removable**  Removable media events. For example, mounting and unmounting events.

**login**  Login and logout events not related to any particular system call.

**admin**  All administrative and privileged events.

**ipccreat**  Interprocess Communication (IPC) object creation.

**ipcopen**  IPC object opening.

**ipcclose**  IPC object deletion.

**ipcdgram**  IPC Datagram transactions.

**uevent1**  User-defined event 1 (for self-auditing records).

**uevent2**  User-defined event 2 (for self-auditing records).

**uevent3**  User-defined event 3 (for self-auditing records).

**System Call Name Mapping Exceptions**

The following are some important known system call name mapping exceptions:

- **sem_open()** is referred to as **ksem_open()**.
- **sem_unlink()** is referred to as **ksem_unlink()**.
- **sem_close()** is referred to as **ksem_close()**.
- **gethostname()**, **sethostname()**, **uname()**, **ustat()**, **setuname()** are all referred to as **utssys()** by the auditing system.

**WARNINGS**

All modifications made to the auditing system are lost upon reboot.

To make the changes permanent, set **AUDEVENT_ARGS1**, **AUDEVENT_ARGS2**, or **AUDEVENT_ARGS3** in **/etc/rc.config.d/auditing**.

**AUTHOR**

**audevent** was developed by HP.

**SEE ALSO**

**audisp(1M)**, **audomon(1M)**, **audsys(1M)**, **audusr(1M)**, **getevent(2)**, **setevent(2)**, **audit(4)**, **audit(5)**.
NAME

audisp - display the audit information as requested by the parameters

SYNOPSIS

[-s stop_time] [-y2] [-y4] audit_filename ...

DESCRIPTION

audisp analyzes and displays the audit information contained in the specified audit_filename audit files. The audit files are merged into a single audit trail in time order. Although the entire audit trail is analyzed, audisp allows you to limit the information displayed, by specifying options. This command is restricted to privileged users.

Any unspecified option is interpreted as an unrestricted specification. For example, a missing -u username option causes all users' audit information in the audit trail to be displayed as long as it satisfies all other specified options. By the same principle, citing -t start_time without -s stop_time displays all audit information beginning from start_time to the end of the file.

audisp without any options displays all recorded information from the start of the audit file to the end. Specifying an option without its required parameter results in error. For example, specifying -e without any eventname returns with an error message.

Options

-u username Specify the login name (username) about whom to display information. If no (username) is specified, audisp displays audit information about all users in the audit file.

-e eventname Display audit information of the specified event types. The defined event types are admin, close, create, delete, ipcclose, ipccreat, ipcopen, login, modaccess, moddac, open, process, readdac, removable, uevent1, uevent2, and uevent3 (see audevent(1M)).

-c syscall Display audit information about the specified system calls.

-p Display only successful operations that were recorded in the audit trail. No user event that results in a failure is displayed, even if username and eventname are specified. The -p and the -f options are mutually exclusive; do not specify both on the same command line. To display both successful and failed operations, omit both -p and -f options.

-f Display only failed operations that are recorded in the audit trail.

-l ttyid Display all operations that occurred on the specified terminal (ttyid) and were recorded in the audit trail. By default, operations on all terminals are displayed.

-t start_time Display all audited operations occurring since start_time, specified as mmddhhmm[yy] (month, day, hour, minute, year). If the year is specified and is greater than 70, it is interpreted as in the twentieth century. Otherwise, it is interpreted as in the twenty-first century. If no year is given, the current year is used. No operation in the audit trail occurring before the specified time is displayed.

-s stop_time Display all audited operations occurring before stop_time, specified as mmddhhmm[yy] (month, day, hour, minute, year). If the year is specified and is greater than 70, it is interpreted as in the twentieth century. Otherwise, it is interpreted as in the twenty-first century. If no year is given, the current year is used. No operation in the audit trail occurring after the specified time is displayed.

-y2 | -y4 The year is displayed as a two digit number (with -y2), or as a four digit number (with -y4). The default is -y2. Note that start_time and stop_time must still be specified as two digit numbers.

AUTHOR

audisp was developed by HP.

SEE ALSO

audevent(1M), audit(4), audit(5).
NAME
audomon - audit overflow monitor daemon

SYNOPSIS

DESCRIPTION
audomon monitors the capacity of the current audit file and the file system on which the audit file is located, and prints out warning messages when either is approaching full. It also checks the audit file and the file system against 2 switch points: FileSpaceSwitch (FSS) and AuditFileSwitch (AFS) and if either is reached, audit recording automatically switches to the backup audit file if it is available.

The FileSpaceSwitch (FSS) is specified as a percentage of the total disk space available. When the file system reaches this percentage, audomon looks for a backup audit file. If it is available, recording is switched from the audit file to the backup file.

The AuditFileSwitch (AFS) is specified (using audsys(1M)) by the size of the audit file. When the audit file reaches the specified size, audomon looks for a backup audit file. If it is available, recording is switched from the audit file to the backup file (see audsys(1M) for further information on use of this parameter).

If either switch point is reached but no backup file is available, audomon issues a warning message.

audomon is typically spawned by /sbin/init.d/auditing (as part of the init(1M) start-up process) when the system is booted up. Once invoked, audomon monitors, periodically sleeping and "waking up" at intervals. Note that audomon does not produce any messages when the audit system is disabled.

audomon is restricted to privileged users.

Options
-p fss Specify the FileSpaceSwitch by a number ranging from 0 to 100. When the audit file's file system has less than fss percent free space remaining, audomon looks for a backup file. If available, the backup file is designated as the new audit file. If no backup file is available, audomon issues a warning message.

The fss parameter should be a larger number than the min_free parameter of the file system to ensure that the switch takes place before min_free is reached. By default, fss is 20 percent.

-t sp_freq Specify the wake-up switch-point frequency in minutes. The wake-up frequency at any other time is calculated based on sp_freq and the current capacity of the audit file and the file system. The calculated wake-up frequency at any time before the switch points is larger than sp_freq. As the size of the audit file or the file system’s free space approaches the switch points, the wake-up frequency approaches sp_freq. sp_freq can be any positive real number. Default sp_freq is 1 (minute).

-w warning Specify that warning messages be sent before the switch points. warning is an integer ranging from 0 through 100. The higher the warning, the closer to the switch points warning messages are issued. For example, warning = 50 causes warning messages to be sent half-way before the switch points are reached. warning = 100 causes warning messages to be sent only after the designated switch points are reached and a switch is not possible due to a missing backup file. By default, warning is 90.

-v Make audomon more verbose. This option causes audomon to also print out the next wake-up time.

-o output_tty Specify the tty to which warning messages are directed. By default, warning messages are sent to the console. Note that this applies only to the diagnostic messages audomon generates concerning the status of the audit system. Error messages caused by wrong usage of audomon are sent to the standard output (where audomon is invoked).

WARNINGS
All modifications made to the audit system are lost upon reboot. To make the changes permanent, set AUDOMON_ARGS in /etc/rc.config.d/auditing.

AUTHOR
audomon was developed by HP.
SEE ALSO
audsys(1M), audit(5).
NAME
audsys - start or halt the auditing system and set or display audit file information

SYNOPSIS
audsys [-nf] [-c file -s cafes] [-x file -z xafs]

DESCRIPTION
audsys allows the user to start or halt the auditing system, to specify the auditing system "current" and "next" audit files (and their switch sizes), or to display auditing system status information. This command is restricted to super-users.

The "current" audit file is the file to which the auditing system writes audit records. When the "current" file grows to either its Audit File Switch (AFS) size or its File Space Switch (FSS) size (see audomon(1M)), the auditing system switches to write to the "next" audit file. The auditing system switches audit files by setting the "current" file designation to the "next" file and setting the new "next" file to NULL. The "current" and "next" files can reside on different file systems.

When invoked without arguments, audsys displays the status of the auditing system. This status includes information describing whether auditing is on or off, the names of the "current" and "next" audit files, and a table listing their switch sizes and the sizes of file systems on which they are located, as well as the space available expressed as a percentage of the switch sizes and file system sizes.

Options
audsys recognizes the following options:

- **-n**
  Turn on the auditing system. The system uses existing "current" and "next" audit files unless others are specified with the -c and -x options. If no "current" audit file exists (such as when the auditing system is first installed), specify it by using the -c option.

- **-f**
  Turn off the auditing system. The -f and -n options are mutually exclusive. Other options specified with -f are ignored.

- **-c file**
  Specify a "current" file. Any existing "current" file is replaced with the file specified; the auditing system immediately switches to write to the new "current" file. The specified file must be empty or nonexistent, unless it is the "current" or "next" file already in use by the auditing system.

- **-s cafes**
  Specify cafes, the "current" audit file switch size (in kbytes).

- **-x file**
  Specify the "next" audit file. Any existing "next" file is replaced with the file specified. The specified file must be empty or nonexistent, unless it is the "current" or "next" file already in use by the auditing system.

- **-z xafs**
  Specify xafs, the "next" audit file switch size (in kbytes).

If -c but not -x is specified, only the "current" audit file is changed; the existing "next" audit file remains. If -x but not -c is specified, only the "next" audit file is changed; the existing "current" audit file remains. The -c option can be used to manually switch from the "current" to the "next" file by specifying the "next" file as the new "current" file. In this instance, the file specified becomes the new "current" file and the "next" file is set to NULL.

In instances where no next file is desired, the -x option can be used to set the "next" file to NULL by specifying the existing "current" file as the new "next" file.

The user should take care to select audit files that reside on file systems large enough to accommodate the Audit File Switch (AFS) desired. audsys returns a non-zero status and no action is performed, if any of the following situations would occur:

The Audit File Switch size (AFS) specified for either audit file exceeds the space available on the file system where the file resides.

The AFS size specified for either audit file is less than the file's current size.

Either audit file resides on a file system with no remaining user space (exceeds minfree, see tunefs(1M)).
WARNINGS
All modifications made to the audit system are lost upon reboot. To make the changes permanent, set AUDITING, PRI_AUDFILE, PRI_SWITCH, SEC_AUDFILE, and SEC_SWITCH in /etc/rc.config.d/auditing.

A user process will be blocked in the kernel if all of the following events occurs:
- the file system containing current audit file is full,
- there is no next audit file or the next audit file is removed, and
- the user process makes an auditable system call or generates an auditable event.

To recover from the resulting deadlock, the session leader of the console is killed so that the the administrator can login. Hence sensitive applications should not be run as session leaders on the console.

AUTHOR
audsys was developed by HP.

FILES
/.secure/etc/audnames File maintained by audsys containing the "current" and "next" audit file names and their switch sizes.

SEE ALSO
audit(5), audomon(1M), audctl(2), audwrite(2), audit(4), setsid(2).
**NAME**

audusr - select users to audit

**SYNOPSIS**

```bash
audusr [-a user] ... [-d user] ... [-A] [-D]
```

**DESCRIPTION**

`audusr` is used to specify users to be audited or excluded from auditing. The `audusr` command only works for systems that have been converted to trusted mode.

To select users to audit on systems that have not been converted to trusted mode, the TrustedMigration product needs to be installed and the `userdbset` command is used. See also `audit(5)`, `userdbset(1M)`, `userdb(4)`, and `AUDIT_FLAG` in `security(4)`.

If no arguments are specified, `audusr` displays the audit setting of every user. `audusr` is restricted to superusers.

**Options**

`audusr` recognizes the following options:

- `-a user` Audit the specified user. The auditing system records audit records to the "current" audit file when the specified user executes audited events or system calls. Use `audevent` to specify events to be audited (see `audevent(1M)`).

- `-d user` Do not audit the specified user.

- `-A` Audit all users.

- `-D` Do not audit any users.

The `-A` and `-D` options are mutually exclusive: that is, if `-A` is specified, `-d` cannot be specified; if `-D` is specified, `-a` cannot be specified.

Users specified with `audusr` are audited (or excluded from auditing) beginning with their next login session, until excluded from auditing (or specified for auditing) with a subsequent `audusr` invocation. Users already logged into the system when `audusr` is invoked are unaffected during that login session; however, any user who logs in after `audusr` is invoked is audited or excluded from auditing accordingly.

**AUTHOR**

`audusr` was developed by HP.

**FILES**

```
/tcb/files/auth/** File containing flags to indicate whether users are audited.
```

**SEE ALSO**

`audevent(1M)`, `userdbset(1M)`, `setaudproc(2)`, `audswitch(2)`, `audwrite(2)`, `security(4)`, `userdb(4)`, `audit(5)`.
authadm(1M)

NAME
authadm - non-interactive command for administrating the authorization information in the RBAC databases.

SYNOPSIS
authadm add operation [object [comments]]
authadm delete operation [object]
authadm assign role operation [object]
authadm roleassign role subrole
authadm revoke role=name [operation=name [object=name]]
authadm rolerevoke role=name subrole=name
authadm list [role=name] [operation=name] [object=name] [subrole=name] [sys]

DESCRIPTION
authadm is a non-interactive command that allows users with the appropriate privileges to modify and list authorization information in the /etc/rbac/roles and /etc/rbac/auths RBAC databases files.

HP recommends using only the authadm, cmdprivadm, and roleadm commands to edit and view the RBAC databases -- do not edit the RBAC files without these commands.

See rbac(5) for more information on these RBAC databases.

Options
With the exception of the list option, all options recognize a default object. If the parameter RBAC_DEFAULT_OBJECT is specified with a non-empty value in the security default file, /etc/default/security, then the value of this parameter will be the default object. However, if the parameter RBAC_DEFAULT_OBJECT does not exist or is set to an empty value, then the default object will be set to a wild card (*).

Here is how to specify a value to the RBAC_DEFAULT_OBJECT parameter in /etc/default/security:

    RBAC_DEFAULT_OBJECT=value

For example: In /etc/default/security, RBAC_DEFAULT_OBJECT=lj8 sets the default object to lj8. If line RBAC_DEFAULT_OBJECT is not present or is commented out, then the default object will be set to "*".

authadm recognizes the following options:

add operation [object|comments]
   Adds an authorization pair (operation, object) to the system list of valid authorizations by appending a line to the /etc/rbac/auths file.
   If object is not specified, then a default object will be assigned. The default object will either be a wild card (*) or the object specified in the security default configuration file, /etc/default/security. A comment may not be specified when adding an entry that refers to the default object in /etc/default/security. The only way to add a comment to an entry with the add option is to specify the object explicitly.

delete operation [object]
   Deletes an authorization from the system list of valid authorizations. If object is not specified, then a default object will be assumed. The default object will either be a wild card (*) or the object specified in the security default configuration file, /etc/default/security.
   If the authorization exists in /etc/rbac/auths, authadm deletes the entry. If the specified authorization is assigned to any roles in /etc/rbac/role_auth, authadm will remove the authorization from the role. If the specified authorization exists in an entry in /etc/rbac/cmd_priv, authadm will remove the entire entry.
   If the authorization does not exist in /etc/rbac/auths, authadm returns an error message. See the RETURN VALUE section below for more information.

assign role operation [object]
   Assigns an authorization pair (operation, object) to a role. authadm verifies the role exists in /etc/rbac/roles before verifying the authorization pair (operation, object) exists in /etc/rbac/auths. authadm appends the authorization to the role to authorization mapping in /etc/rbac/role_auth if the role and authorization pair exists.
If `object` is not specified, then a default `object` will be assigned. The default `object` will either be a wild card (*) or the `object` specified in the security default configuration file, `/etc/default/security`.

### roleassign role subrole

Assigns a role to another different role. The role being assigned to the other different role is referred to as a **subrole**. A subrole is any valid role defined in the `/etc/rbac/roles` database.

The `roleassign` option allows hierarchical role definition (one role can inherit other subrole). After assigning a subrole to another role, that role will also have all the authorizations of the subrole, and any of its subroles. More than one subrole can be assigned to an other different role. `authadm` verifies the role and subrole exist in `/etc/rbac/roles`. It also verifies that there is no recursive definitions of the role and subrole. (If "role1" has a subrole of "role2", and if you try to `roleassign"role1" to "role2", this will cause a recursive definition of both "role1" and "role2"). `authadm` appends the subrole to the role to authorization mapping in `/etc/rbac/role_auth`.

### revoke role= role subrole= operation= object=

Revokes an authorization from the specified role in `/etc/rbac/role_auth`. If no authorization is specified, `authadm` revokes all the authorizations for the given role. If `object` is not specified, then a default `object` will be assumed. The default `object` will either be a wild card (*) or the `object` specified in the security default configuration file, `/etc/default/security`.

Note: The `/etc/rbac/role_auth` file will be modified by the `authadm revoke` command.

### rolerevoke role= role subrole= operation= object=

Revokes a subrole from the specified role in `/etc/rbac/role_auth`. Note that the role specified as the subrole is not revoked from the database, just the subrole assignment is revoked.

For instance, if these entries are in the database:

```
role1: (operation1, object1) role2
role2: role3 (operation2, object2), role4
```

`authadm rolerevoke role=role1 subrole=role2` will modify the line to:

```
role1: (operation1, object1)
role2: role3 (operation2, object2), role4
```

`authadm` revokes specified the authorizations and/or subrole for the given role.

Note: The `/etc/rbac/role_auth` file will be modified by the `authadm rolerevoke` command.

### authadm list role= role subrole= operation= object= sys

Invoking the `authadm list` command without any parameters lists every entry in `/etc/rbac/auths` database. Specifying a role name lists all the roles witch have that role name. Specifying an operation name lists all the roles which have that operation name. Specifying a subrole name lists all the roles which have that subrole name. Specifying `sys` lists all the authorizations in the `/etc/rbac/role_auth` database.

### Authorizations

In order to invoke `authadm`, the user must either be root, (running with effective uid of 0), or have the appropriate authorization(s). The following is a list of the required authorizations for running `authadm` with particular options:

- **hpux.security.access.auth.add,**
  - Allows user to run `authadm` with `add` option.
- **hpux.security.access.auth.delete,**
  - Allows user to run `authadm` with `delete` option.
- **hpux.security.access.auth.assign,**
  - Allows user to run `authadm` with `assign` or `roleassign` option.
- **hpux.security.access.auth.revoke,**
  - Allows user to run `authadm` with `revoke` or `rolerevoke` option.
- **hpux.security.access.auth.list,**
  - Allows user to run `authadm` with `list` option.
EXTERNAL INFLUENCES

Environment Variables

LC_MESSAGES determines the language in which messages are displayed.

International Code Set Support

Single-byte character code set is supported.

RETURN VALUE

Success If authadm is successful, then the return value from authadm is 0.

Failure authadm returns a value of 1 and an appropriate error message will be printed to stderr.

EXAMPLES

The following commands each add an authorization (operation, object) entry in the /etc/rbac/auths database file:

   # authadm add hpux.printer.job.cancel printer6
   # authadm add hpux.mount
   # authadm add hpux.printer.bld3

The following commands each delete an authorization (operation, object) entry from the /etc/rbac/auths database file:

   # authadm delete hpux.printer.job.cancel printer6
   # authadm delete hpux.printer.bld3

The following commands each assign an authorization (operation, object) pair to a role in /etc/rbac/role_auth database file:

   # authadm assign administrator hpux.printer.job.cancel printer6
   # authadm assign accountant hpux.db.admin

The following commands each assign a subrole to a role in /etc/rbac/role_auth database file:

   # authadm roleassign administrator printAdmin
   # authadm roleassign administrator diskAdmin

The following commands each revokes an operation for the specified operation from a role in the /etc/rbac/role_auth file:

   # authadm revoke role=administrator operation=hpux.printer.job.cancel object=printer6
   # authadm revoke role=accountant operation=hpux.db.admin

The following commands each revokes a subrole from the specified role in the /etc/rbac/role_auth database file:

   # authadm rolerevoke role=administrator subrole=printAdmin
   # authadm rolerevoke role=administrator subrole=diskAdmin

The following command lists all the authorizations for the administrator role:

   # authadm list role=administrator

The following command lists all the entries with operation hpux.db.admin:

   # authadm list operation=hpux.db.admin

The following command lists all the entries with object /etc/passwd.

   # authadm list object=/etc/passwd

The following command lists all the roles with their authorizations in /etc/rbac/auths database:

   # authadm list sys

FILES

/etc/rbac/roles Database containing valid definitions of all roles.

/etc/rbac/auths Database containing definitions of all valid authorizations.

/etc/rbac/user_role Database specifying the roles allowed for each specified user.
authadm(1M)

/etc/rbac/role_auth  Database defining the authorizations for each specified role.

SEE ALSO
    cmdprivadm(1M), privrun(1M), rbacdbchk(1M), roleadm(1M), rbac(5).
NAME
authck - check internal consistency of Authentication database

SYNOPSIS

DESCRIPTION
authck checks both the overall structure and internal field consistency of all components of the Authentication database. It reports all problems it finds. Only users who have the superuser capability can run this command. When pwck is used with the -s option, authck is run with the -p option automatically.

Options
authck recognizes the following options and tests:

- **p**  Check the Protected Password database. The Protected Password database and /etc/passwd are checked for completeness such that neither contains entries not in the other. The cross references between the Protected Password database and /etc/passwd are checked to make sure that they agree. However, if Nis+ is configured in your system, the password table is also checked before reporting a discrepancy. This means that a discrepancy would not be reported for a user that does NOT exist in /etc/passwd but exists in the Protected Password database as well as the Nis+ passwd table. Fields in the Protected Password database are then checked for reasonable values. For example, all time stamps of past events are checked to make sure that they have times less than the times returned by time(2).

- **t**  Fields in the Terminal Control database are checked for reasonable values. All time stamps of past events are checked to make sure they have times less than those returned by time(2).

- **a**  Shorthand equivalent of using the -p and -t options together in a single command.

- **v**  Provide running diagnostics as the program proceeds. Produce warnings when unusual conditions are encountered that might not cause program errors in login, password and su programs.

- **d**  Removes Protected Password database entries that are not found in the Nis+ passwd table. Nis+ users may have an entry in the Protected database and not in /etc/passwd. Thus, this option removes orphaned Protected database entries: orphaned entries can exist for deleted Nis+ users. The optional domainname specifies the desired Nis+ domain to use for the passwd table. If domainname is not specified, the local domain name is used.

FILES
/etc/passwd  System password file
/tcb/files/auth/*/  Protected Password database
/tcb/files/ttys  Terminal Control database
/tcb/files/auth/system/default  System Defaults database
/usr/sbin/authck

AUTHOR
authck was developed by HP.

SEE ALSO
getprpwent(3), getprtcent(3), getprdfent(3), authcap(4).
auto_parms(1M)

NAME
auto_parms - initial system configuration plus DHCP support command

SYNOPSIS
auto_parms

DESCRIPTION
The auto_parms command is a system initialization command that handles first-boot configuration, that is, the setting of unique system “initial identity parameters”, and ongoing management of DHCP lease(s).

The auto_parms command is invoked at boot time by the /sbin/rc command. Initially it loads a list of available ethernet interfaces and requests a DHCP lease on each interface. It stops when a valid lease is secured or the list is exhausted.

While checking for availability of a lease on each interface, auto_parms also consults /etc/rc.config.d/netconf and examines the variable DHCP_ENABLE[index]. If it is set to “1”, auto_parms attempts to request a lease on the the interface designated by index. If DHCP_ENABLE[index] is set to “0” (the default case) or is absent from /etc/rc.config.d/netconf, auto_parms does not attempt a DHCP request on that interface.

Once a lease is secured, the information supplied with the lease is used to initialize key networking parameters; see dhcpdb2conf(1M).

If auto_parms detects that the system is undergoing “first-boot” (meaning the system’s hostname is not yet set), it invokes set_parms and geocustoms to verify the DHCP-supplied parameters and to collect any parameters not supplied by DHCP.

For all subsequent boots, the data supplied by a DHCP lease is assumed by auto_parms to be correct. Note that in a (non-mobile) environment where DHCP is used for IP address management, under normal conditions the lease information should not change between reboots. This is accomplished by auto_parms placing dhcpclient in "lease maintenance mode" prior to exiting.

EXAMPLES
See /sbin/rc for invocation context.

FILES
/sbin/auto_parms
The command itself.
/sbin/set_parms.util
Common subroutines used by auto_parms, set_parms, and setParms sub-area commands.
/etc/auto_parms.log
/etc/auto_parms.log.old
Logfiles written and saved by auto_parms.

AUTHOR
The auto_parms command was developed by HP.

SEE ALSO
set_parms(1M), geocustoms(1M), dhcpdb2conf(1M).
NAME
automount - install automatic mount points

SYNOPSIS
/usr/sbin/automount [-f master-file] [-t duration] [-v]

DESCRIPTION
The automount command installs autofs mount points and associates an automount map with each
mount point. The autofs filesystem monitors attempts to access directories within it and notifies the
automountd daemon (see automountd(1M)). The daemon uses the map to locate a filesystem, which it
then mounts at the point of reference within the autofs filesystem. You can assign a map to an autofs
mount using an entry in the /etc/auto_master map or a direct map.

If the filesystem is not accessed within an appropriate interval (10 minutes by default), the automountd
daemon unmounts the file system.

The file /etc/auto_master determines the locations of all autofs mount points. By default, this file
contains the following entry:

```
# Master map for automounter
# /net -hosts -nosuid,soft,nobrowse
```

The first field in the master file specifies a directory on which an autofs mount will be made, and the
second field specifies the automounter map to be associated with it. Mount options may be supplied as an
optional third field in the entry. These options are used for any entries in the map that do not specify
mount options explicitly. The automount command is usually run without arguments. It compares the
entries /etc/auto_master with the current list of autofs mounts in /etc/mnttab and adds, removes, or updates autofs mounts to bring the /etc/mnttab up to date with the
/etc/auto_master. At boot time, it installs all autofs mounts from the master map. Subsequently,
it may be run to install autofs mounts for new entries in the master map or a direct map, or to perform
unmounts for entries that have been removed.

If the first field specifies the directory as /-, automount treats the second field as the name of a direct
map. In a direct map, each entry associates the full path name of a mount point with a remote file system
to mount.

If the first field is a path name, the second field names an indirect map or a special map (described below).
An indirect map contains a list of the subdirectories within the indicated directory. With an indirect map,
it is these subdirectories that are mounted automatically.

The automounter maps, including the auto_master map, may be distributed by NIS, NIS+, or LDAP. The
Name Service Switch configuration file, /etc/nsswitch.conf, determines where the automount
command will look for the maps.

Options
- **f** master-file Specify a local master file for initialization.

  When the -f option is used and the master file specified is not found, then automount
defaults to /etc/auto_master and then to the NIS auto_master map.

- **t** duration Specify a duration, in seconds, that a file system is to remain mounted when not in use.
The default is 600 (10 minutes).

- **v** Verbose mode. Notify of autofs mounts, unmounts, or other non-essential information.
  Messages are written to standard error.

Map Entry Format
A simple map entry (mapping) takes the form:

```
key [-mount-options] location...
```

where key is the full path name of the directory to mount when used in a direct map, or the simple name of
a subdirectory in an indirect map. mount-options is a comma-separated list of mount options, and location
specifies a filesystem from which the directory may be mounted. In the case of a simple NFS mount, the
options that can be used are as specified in mount_nfs(1M), and location takes the form:

```
host:pathname
```
host is the name of the host from which to mount the file system and pathname is the path name of the directory to mount.

Default mount options can be assigned to an entire map when specified as an optional third field in the master map. These options apply only to map entries that have no mount options.

**Replicated Filesystems**

Multiple location fields can be specified for replicated NFS filesystems, in which case the information is used to try to increase availability. The server chosen for the mount is the one with the strongest preference based on a sorting order. Note that autofs does not monitor the status of mounts when dealing with replicated filesystems and does not select alternate servers.

The sorting order used gives strongest preference to servers on the same local subnet with servers on the local net given the second strongest preference. Among servers equally far away, response times will determine the order if no weighting factors are used (see below).

If the list of locations contains some servers using the NFS Version 2 Protocol and some using the NFS Version 3 Protocol, the automount will choose a subset of the list having only servers with the same protocol. This subset is formed of servers using the NFS Version 3 Protocol unless there are no such servers on the list or there is a server using the NFS Version 2 Protocol that has the strongest preference as described previously. If each location in the list shares the same pathname then a single location may be used with a comma-separated list of hostnames:

```
hostname,hostname...:pathname
```

Requests for a server may be weighted, with the weighting factor appended to the server name as an integer in parentheses. Servers without a weighting default to a value of zero (most likely to be selected). Progressively higher values decrease the chance of being selected. In the example,

```
man -ro alpha,bravo,charlie(1),delta(4):/usr/share/man
```

hosts alpha and bravo have the highest priority; host delta has the lowest priority.

**NOTE:**

Server proximity takes priority in the selection process. In the example above, if the server delta is on the same network segment as the client, but the others are on different network segments, then delta will be selected; the weighting value is ignored. The weighting has effect only when selecting between servers with the same network proximity.

In cases where each server has a different export point, you can still apply the weighting. For example:

```
man -ro alpha:/usr/man bravo,charlie(1):/usr/share/man \
delta(3):/export/man
```

A mapping can be continued across input lines by escaping the newline with a backslash (\). Comments begin with a number sign (#) and end at the subsequent newline.

**Map Key Substitution**

The ampersand (&) character is expanded to the value of the key field for the entry in which it occurs. In this case:

```
amy rowboatserver:/home/\n```

the & expands to amy.

**Wildcard Key**

The asterisk (*) character, when supplied as the key field, is recognized as the catch-all entry. Such an entry will match any key not previously matched. For instance, if the following entry appeared in the indirect map for /config:

```
* &:/export/config/\n```

it would allow automatic mounts in /config of any remote file system whose location could be specified as:

```
hostname:/export/config/hostname
```
Variable Substitution
Client specific variables can be used within an automount map. For instance, if $HOST appeared within a map, automount would expand it to its current value for the client’s host name. Supported variables are:

- **CPU** The processor type. The possible values are: **IA64, PA10, PA11, PA20**.
- **HOST** The output of `uname -n`. The host name. For example, `rowboat`.
- **OSNAME** The output of `uname -s`. The OS name. For example, `HP-UX`.
- **OSREL** The output of `uname -r`. The OS release name. For example, `B.11.00`.
- **OSVERS** The output of `uname -v`. The OS version. For example, `C`.

If a reference needs to be protected from neighboring characters, you can surround the variable name with braces (`{}`).

Multiple Mounts
A multiple mount entry takes the form:

```
key [ -mount-options ] [ [mountpoint] [ -mount-options ] location ... ]
```

The initial `/mountpoint` is optional for the first mount and mandatory for all subsequent mounts. The optional `mountpoint` is taken as a path name relative to the directory named by `key`. If `mountpoint` is omitted in the first occurrence, a `mountpoint` of `/` (root) is implied.

Given an entry in the indirect map for `/src`:

```
beta -ro /
    /svr1,svr2:/export/src/beta /
    /1.0 svr1,svr2:/export/src/beta/1.0 /
    /1.0/man svr1,svr2:/export/src/beta/1.0/man
```

All offsets must already exist on the server under `beta`. automount would automatically mount `/src/beta`, `/src/beta/1.0`, and `/src/beta/1.0/man`, as needed, from either `svr1` or `svr2`, whichever host is nearest and responds first.

The `autofs` mount points must not be hierarchically related. automount does not allow an `autofs` mount point to be created within another `autofs` mount.

Other Filesystem Types
The automounter assumes NFS mounts as a default filesystem type. Other filesystem types can be described using the `-fstype` mount option. Other mount options specific to this filesystem type can be combined with the `-fstype` option. The location field must contain information specific to the filesystem type. If the location field begins with a slash, a colon character must be prepended, for instance, to mount a CD filesystem:

```
cdrom -fstype=hsfs,ro :/dev/sr0
```

or to perform an `autofs` mount:

```
src -fstype=autofs auto_src
```

Mounts using CacheFS are most useful when applied to an entire map as map defaults (see `cfsadmin(1M)`). The following entry in the master map describes cached home directory mounts. It assumes the default location of the cache directory, `/cache`.

```
/home auto_home -fstype=cachefs,backfstype=nfs
```

Indirect Maps
An indirect map allows you to specify mappings for the subdirectories you wish to mount under the directory indicated in the `/etc/auto_master` map. In an indirect map, each `key` consists of a simple name that refers to the subdirectory of one or more filesystems that are to be mounted as needed.

Entries in both direct and indirect maps can be modified at any time. The new information is used when `automountd` next uses the map entry to do a mount.

Direct Maps
Entries in a direct map are associated directly with `autofs` mount points. Each `key` is the full path name of an `autofs` mount point. The direct map as a whole is not associated with any single directory.
Since each direct map entry results in a new `autofs` mount, such maps should be kept short.

If a directory contains direct map mount points, then an `ls -l` in the directory will force all the direct map mounts to occur.

Entries in both direct and indirect maps can be modified at any time. The new information is used when `automountd` next uses the map entry to do a mount.

New entries added to a master map or direct map will not be useful until the `automount` command is run to install them as new `autofs` mount points. New entries added to an indirect map may be used immediately.

Included Maps

The contents of another map can be included within a map with an entry of the form:

```
+mapname
```

If `mapname` begins with a slash then it is assumed to be the path name of a local file. Otherwise the location of the map is determined by the policy of the name service switch according to the entry for the automounter in `/etc/nsswitch.conf`, such as

```
automount: nis files
```

If the name service is `files` then the name is assumed to be that of a local file in `/etc`. If the key being searched for is not found in the included map, the search continues with the next entry.

Special Maps

There are two special maps available: `-hosts` and `-null`. The `-hosts` map is used with the `/net` directory and assumes that the map key is the hostname of an NFS server. The `automountd` daemon dynamically constructs a map entry from the server's list of exported filesystems. References to a directory under `/net/hermes` will refer to the corresponding directory relative to `hermes` root.

The `-null` map cancels a map for the directory indicated. This is most useful in the `/etc/auto_master` map for cancelling entries that would otherwise be inherited from the `+auto_master` include entry. To be effective, the `-null` entries must be inserted before the included map entry.

Executable Maps

Local maps that have the execute bit set in their file permissions will be executed by the automounter and provided with a key to be looked up as an argument. The executable map is expected to return the content of an automounter map entry on its standard output or no output if the entry cannot be determined. A direct map cannot be made executable.

Configuration and the auto_master Map

When initiated without arguments, `automount` consults the master map for a list of `autofs` mount points and their maps. It mounts any `autofs` mounts that are not already mounted, and unmounts `autofs` mounts that have been removed from the master map or direct map.

The master map is assumed to be called `auto_master` and its location is determined by the name service switch policy. Normally the master map is located initially as a local file, `/etc/auto_master`.

Browsing

Browsing of indirect maps allows all of the potential mount points to be visible for that map regardless of whether they are mounted or not. The `-nobrowse` option can be added to any indirect `autofs` map to disable browsing. For example:

```
/net -hosts -nosuid,soft,nobrowse
```

In this case, any host names would only be visible in `/net` after they are mounted. The `-browse` option enables browsing of `autofs` file systems. This is the default for all indirect maps, although it is suggested that the `-hosts` entry contain the `-nobrowse` option.

Note that, although a listing of the `autofs` directory associated with an indirect map shows all potential mountable entries, the attributes associated with those entries are temporary until the actual filesystem attributes can be retrieved once the filesystem has been mounted.
Network Information Service (NIS) and Yellow Pages (YP)
The Network Information Service (NIS) was formerly known as Sun Yellow Pages (YP). The functionality of the two remains the same.

RETURN VALUE
automount returns the following values:
- 0  Success.
- 1  Failure.
- 3  Map not found.

AUTHOR
automount was developed by Sun Microsystems, Inc.

FILES
/etc/auto_master  Master automount map.
/etc/nsswitch.conf  Name service switch configuration file.
/usr/sbin/automount  autosfs automount command.

SEE ALSO
automountd(1M), cfsadmin(1M), mount(1M).
NAME
automountd - autofs mount/unmount daemon

SYNOPSIS
automountd [-TvnL] [-D name=value]

DESCRIPTION
automountd is an RPC server that answers file system mount and unmount requests from the autofs filesystem. It uses local files or name service maps to locate filesystems to be mounted. These maps are described with the automount command (see automount(1M)).

The automountd daemon is automatically invoked if the AUTOFS variable is set to 1 in /etc/rc.config.d/nfsconf.

Options
- T Trace. Expand each RPC call and display it to /var/adm/automount.log.
- v Verbose. Log status messages to /var/adm/syslog/syslog.log.
- n Turn off browsing for all autofs mount points. This option overrides the -browse autofs map option on the local host.
- L Force all mounts to the local host to be NFS mounts instead of the default LOFS mounts. This is necessary for highly available NFS mounts.
- D name=value Assign value to the indicated automount map substitution variable. These assignments cannot be used to substitute variables in the master map auto_master.

Tracing
While automountd is running, the signal SIGUSR2 can be sent to automountd to turn tracing on. It logs messages in /var/adm/automount.log. To turn tracing off, send the signal SIGUSR2 again.

RETURN VALUE
automountd returns:
  0 successful
  1 failure

EXAMPLES
To turn on tracing, enter:
/usr/sbin/automountd -T
or
automountd -T

No messages are displayed by the trace. Check the results in /var/adm/automount.log.

WARNINGS
Any filesystems that are being managed by autofs should never be manually mounted or unmounted. Even if the mount or unmount operation appears to complete successfully, the resulting change of state to the autofs managed filesystem can lead to disruptive or unpredictable results, including but not limited to: commands hanging or not returning expected results, and applications failing due to their dependencies on those mounted filesystems. A reboot may be necessary to resolve these issues.

AUTHOR
automountd was developed by Sun Microsystems, Inc.

FILES
/etc/auto_master master map for automounter
/var/adm/automount.log log file for automountd
/usr/sbin/automountd autofs daemon

SEE ALSO
automount(1M).

HP-UX 11i Version 2: December 2007 Update Hewlett-Packard Company 91
NAME
autopush - manage system database of automatically pushed STREAMS modules

SYNOPSIS
autopush -f file
autopush -g -M major -m minor
autopush -r -M major -m minor

DESCRIPTION
autopush manages the system database that is used for automatic configuration of STREAMS devices.
The command is used in three different ways as dictated by the -f, -g, and -r command-line options
described below.

Options
autopush recognizes the following command-line options and arguments:

-f file
Using the configuration information contained in file, load the system database with
the names of the STREAMS devices and a list of modules to use for each device.
When a device is subsequently opened, the HP-UX STREAMS subsystem pushes the
modules onto the stream for the device.
file must contain one or more lines of at least four fields separated by a space as
shown below:

major minor lastminor module1 module2 ... moduleN

The first field major can be either an integer or a device name. The device name is
the name for the device used in the master file. The next two fields are integers. If
minor is set to -1, then all minor devices for the specified major are configured and
lastminor is ignored. If lastminor is 0, then only a single minor device is configured.
To configure a range of minor devices for a major device, minor must be less then last-
minor. The remaining field(s) list one or more module names. Each module is pushed
in the order specified. A maximum of eight modules can be pushed. Any text after a #
character in file is treated as a comment for that line only.
This option is also used to restore device configuration information previously
removed by autopush -r. However, when used in such a manner, the entire data-
base is restored, not just the information that was previously removed.

-g -M major -m minor
Display current configuration information from the system database for the
STREAMS device specified by the major device number (or device name for the device
from the master file) and minor number.
If a range of minors has been previously configured then autopush -g returns the
configuration information for the first minor in the range, in addition to other infor-
mation.

-r -M major -m minor
Remove configuration information from the system database for the STREAMS device
specified by the major device number (or device name for the device from the master
file and minor number. Removal is performed on the database only, not on the origi-
nal configuration file. Therefore, the original configuration can be restored by using the
-f file option. To permanently exclude a STREAMS device from the database, its
information must be removed from the configuration file.
If minor matches the first minor of a previously configured range then autopush
-r removes the configuration information for the entire configured range.

EXAMPLES
If the file /tmp/autopush.example contains:

75 -1 0 modA modB
test 0 5 modC modA

92 Hewlett-Packard Company - 1 - HP-UX 11i Version 2: December 2007 Update
Then `autopush -f /tmp/autopush.example` will cause `modA` and `modB` to be pushed whenever major device # 75 is opened, and `modC` and `modA` to be pushed for the first six opens of device `test`.

This next example lists information about the stream for major device 75 and its minor device -2:

```
    autopush -g -M 75 -m -2
```

**FILES**

`/usr/lib/nls/msg/C/autopush.cat` NLS catalog for `autopush`.

**SEE ALSO**

`sad(7), streamio(7)`.  

---

**autopush(1M)**

**files**: `/usr/lib/nls/msg/C/autopush.cat` NLS catalog for `autopush`.

**SEE ALSO**

`sad(7), streamio(7)`.  

---

**autopush(1M)**
backup(1M)

NAME
backup - backup or archive file system

SYNOPSIS
/usr/sbin/backup [-A] [-archive] [-fsck]

DESCRIPTION
The backup command uses find(1) and cpio(1) to save a cpio archive of all files that have been modified since the modification time of /var/adm/archivedate on the default tape drive (/dev/update.src). backup should be invoked periodically to ensure adequate file backup.

The -A option suppresses warning messages regarding optional access control list entries. backup(1M) does not backup optional access control list entries in a file's access control list (see acl(5)). Normally, a warning message is printed for each file having optional access control list entries.

The -archive option causes backup to save all files, regardless of their modification date, and then update /var/adm/archivedate using touch(1).

backup prompts you to mount a new tape and continue if there is no more room on the current tape. Note that this prompting does not occur if you are running backup from cron(1M).

The -fsck option causes backup to start a file system consistency check (without correction) after the backup is complete. For correct results, it is important that the system be effectively single-user while fsck is running, especially if -fsck is allowed to automatically fix whatever inconsistencies it finds. backup does not ensure that the system is single-user.

You can edit /usr/sbin/backup to customize it for your system. Several local values are used that can be customized:

- BACKUPDIRS specifies which directories to back up recursively (usually /, meaning all directories);
- BACKUPLOG file name where start and finish times, block counts, and error messages are logged;
- ARCHIVE file name whose date is the date of the last archive;
- REMIND file name that is checked by /etc/profile to remind the next person who logs in to change the backup tape;
- FSCKLOG file name where start and finish times and fsck output is logged.

You may want to make other changes, such as whether or not fsck does automatic correction (according to its arguments), where cpio output is directed, other information logging, etc.

In all cases, the output from backup is a normal cpio archive file (or volume) which can be read using cpio with the c option.

File Recovery
backup creates archive tapes with all files and directories specified relative to the root directory. When recovering files from an archive tape created by backup, you should be in the root directory and specify the directory path names for recovered files relative to the root directory (/). When specifying the directory path name for file recovery by cpio, do not precede the leading directory name with a slash. If you prefer, you can also use cpio with a -t option to determine how files and directories are named on the archive tape before attempting recovery.

WARNINGS
Refer to WARNINGS in cpio(1).

When cpio runs out of tape, it sends an error to standard error and demands a new special file name from /dev/tty.

To continue, rewind the tape, mount the new tape, type the name of the new special file at the system console, and press Return.

If backup is being run unattended from cron(1M) and the tape runs out, backup terminates, leaving the find process still waiting. Kill this process when you return.

FILES
/var/adm/archivedate parameterized file names
SEE ALSO

cpio(1), find(1), touch(1), cron(1M), fbackup(1M), frecover(1M), fsck(1M), acl(5).
NAME
bastille - system lockdown tool

SYNOPSIS
Path (Linux): /usr/sbin
Path (HP-UX): /opt/sec_mgmt/bastille/bin
bastille [-b | -c | -x] [-f alternate_config_file] [-os [version]]
bastille [-l | -r | --assess | --assessnobrowser]

DESCRIPTION
Bastille is a system-hardening/lockdown program that enhances the security of a Unix host. It configures
daemons, system settings and firewalls to be more secure. It can shut off unneeded services and r-tools like
rcp and rlogin, and helps create "chroot jails" that help limit the vulnerability of common Internet ser-
VICES, like Web servers and DNS. This tool currently hardens Red Hat 6.0-8.0, Mandrake 6.0-8.1, HP-UX
11i v1, HP-UX 11i v2, and HP-UX 11i v3. It is currently being tested on Debian, SuSE, and Turbo Linux.
The utility includes a policy/configuration-selection interface, a configuration engine and a reporting
module. The primary profile-building interface is an X interface via Perl/Tk. There is also a text-based
Perl/Curses interface for Linux. The tool can be used interactively and noninteractively (when the policy-
application engine is used directly). Used interactively, to build system-security configurations, Bastille has
been designed to explain security issues to system administrators, then let them decide how to let the tool
handle them. This both secures the system and educates the administrator. When the configuration
engine is used directly, the utility is useful for duplicating a security configuration on multiple machines.

When used interactively (bastille, bastille -x, or bastille -c), the user interface guides you through a
series of questions. Each step contains a description of a security decision involved in hardening a Unix
system. Each question describes the cost/benefit of each decision. The Tk interface gives you the option to
skip to another question module and return to the current module later. The X interface provides "Com-
pleted Indicators" to show you which question modules are complete. After you have answered all of the
questions, the interface then provides automated support in performing lockdown steps. After performing
the steps Bastille can perform automatically, the utility produces a "to-do" list that describes remaining
actions you must perform manually to ensure the system is secure.

Security hardening can also be performed directly through the configuration engine (bastille -b) using the
default or an alternate configuration (bastille -b -f file) (see the config file in the FILES section below for
the default location). This method is useful for duplicating a particular security configuration on multiple
machines. Before using the configuration engine directly, a configuration file must be created by using Bas-
tille interactively. After the configuration file is created, copy it to the other systems, install Bastille Unix
on those systems, then run the configuration engine on those systems.

Bastille draws from many major reputable sources on Unix Security. The initial development integrated
Jay Beale’s existing O/S hardening experience for Solaris and Linux with most major points from the SANS’
Securing Linux Step by Step and Kurt Seifried’s Linux Administrator’s Security Guide. Later versions
incorporated suggestions from the HP-UX Bastion Host White-paper, Center for Internet Security, and
other sources.

To ensure that Bastille is used as safely as possible, please:

1) Let the developers know about any impacts you discover which aren’t mentioned in the question text
   for possible inclusion in future revisions of the questions text.

2) Test Bastille configurations in a nonproduction environment first, with the application stack fully func-
tionally tested after lockdown before deployment in a production environment. The characterization
   of consequences is known to be incomplete, especially for general purpose systems.

Options
bastille recognizes the following options

-b Run in batch mode. This option takes the answers that were created interactively and applies them to
the machine.

c Linux Only. Bring up the text interface of the interactive portion of Bastille. It is implemented with
the Perl/Curses module, which must be installed separately if it did not come with your version of
Perl.
-f alternate_config_file
  Use an alternate config file versus the default location.

-l List applied configuration files. List the configuration files in the configuration file directory that
  matches the one last used.

-r Revert Bastille-modified system files to the state they were in before Bastille was run. Note that, if
  any changes to the system configuration were made in the interim, those changes should be reviewed
  again to make sure they (1) still work, and (2) have not broken the system or compromised its secu-
  rity.

-x The default option. Run the Bastille X interface. It is implemented with the Perl/Tk module, which
  must be installed separately if it did not come with your version of Perl.

--assess
  Run Bastille in assessment-only mode so that it investigates the state of hardening, reports on such
  and generates a score. No changes are made to the system. It generates HTML and text reports and
  a Bastille configuration file.

  For each question, Bastille generates one of the following results:

  Yes  The associated Bastille lockdown has been applied to the product or service shipped with HP-
        UX. Bastille may not always correctly detect the status of products or services that are not
        shipped with the HP-UX OE. Also, Bastille may not detect all variations of the possible ways
        to disable or enable a service or feature. It will detect if Bastille did so, and will likely detect
        configuration made in accepted, standard ways.

  No   The question configuration has not been applied.

User Action Pending
  Bastille had performed a partial configuration; leaving the user with some actions needed to
  complete the configuration. These actions are listed in the TODO file listed below.

Inconsistent
  Bastille can not tell the status. Usually, this is do to the system being in an inconsistent state.
  For example, Bastille would return this status of a service running in the process list, but
  configured on disk to be off. Note, there are some cases where inconsistent states that Bastille
  can not detect could be created on the system, so if the administrator has made changes to the
  system, and needs to rely on Bastille results, the system should be rebooted first to ensure the
  configuration is consistent. This caveat does not apply to Bastille initiated actions.

N/A: S/W Not Installed
  This indicates that the relevant software is not installed, so there is no need to lock down the
  given item, but care should be taken when the software is installed to lock it down at that
  point.

Needed S/W Missing
  This indicates that the item is not locked down since it needs software that is currently not
  available on the system.

Set to value
  This indicates a nonboolean setting.

Not Defined
  This indicates a nonboolean setting that has not been set yet. Thus the system default settings
  apply. In the case of later HP-UX versions, default account security settings are often found in
  the /etc/security.dse file.

  See the FILES section for location. The HTML version of the report is shown in a browser if either a
  graphical or text browser can be found.

--assessnobrowser
  Same as --assess, except that the report is not displayed in a browser.

--os [version]
  Explicitly set the operating system version while generating a configuration file. By setting the
  operating system version, all questions valid for that operating system will be asked and configuration
  files can be generated for any version Bastille recognizes. For a complete list of operating system
  versions type bastille -x --os.


**DIAGNOSTICS**

$DISPLAY not set, cannot use X interface...
You explicitly asked for the X interface using the -x option, but the DISPLAY environment variable was not set.  Set the environment variable to the desired display to correct the problem.

System is in original state...
You attempted to revert the files that Bastille changes with the -r option, but there were no changes to revert.

Must run Bastille as root
Bastille must run as the root user, since the changes it makes configure the machine.

**Troubleshooting**

Error messages that cite problems with opening, copying, or reading files usually relate to NFS file systems that do not trust the root user on the local machine.  Please see the options parameter in the /fstab(4) manpage for details.  Errors that complain about individual configuration files indicate that a system has been too heavily modified for Bastille to make effective changes, or that the files, locations, or permissions of the Bastille installation directories have been changed.

If Bastille is unable to complete a lockdown, you should receive errors or warnings.  Analyze the errors or warnings to determine if your lockdown was successfully applied.  You may use the --assess option to aid in this diagnostic.  Once the system state that caused the abort is fixed, run bastille again to complete the lockdown.  This helps avoid cases where an incomplete lockdown can contribute to an inconsistent system configuration.

**EXAMPLES**

Example 1
Run the Bastille X interface.  This will create a configuration file which can be run either immediately by Bastille after you have answered all of the questions, or saved for later use in a config file.  See the FILES section below.

```
bastille
```

Example 2
Run Bastille in batch mode.  This will take the answers that were created interactively and apply them to the machine.

```
bastille -b
```

Example 3
Perform an audit of the system to determine the state of the security settings on it, and place it in the audit log locations (specified below).

```
bastille --assessnobrowser
```

**DEPENDENCIES**

- Perl version 5.8.0 or greater, but recommend 5.8.8 or greater for best performance
- Perl/Tk version 8.00.23 or greater
- Perl/Curses version 1.06 or greater (on Linux only)

**FILES**

/etc/Bastille/config  (Linux)
/etc/opt/sec_mgmt/bastille/config  (HP-UX)

The config file contains the answers to the most recently saved session

/var/log/Bastille/error-log  (Linux)
/var/opt/sec_mgmt/bastille/log/error-log  (HP-UX)

The error log contains any errors that Bastille encountered while making changes to the system.

/var/log/Bastille/action-log  (Linux)
/var/opt/sec_mgmt/bastille/log/action-log  (HP-UX)
The action log contains the specific steps that Bastille took when making changes to the system.

/var/log/Bastille/TODO (Linux)
/var/opt/sec_mgmt/bastille/TODO.txt (HP-UX)

The to-do list contains the actions that remain for you to do to ensure the machine is secure.

/var/log/Bastille/Assessment/assessment-report.html (Linux)
/var/log/Bastille/Assessment/assessment-report.txt (Linux)
/var/log/Bastille/Assessment/assessment-report-log.txt (Linux)
/var/opt/sec_mgmt/bastille/log/Assessment/assessment-report.html (HP-UX)
/var/opt/sec_mgmt/bastille/log/Assessment/assessment-report.txt (HP-UX)
/var/opt/sec_mgmt/bastille/log/Assessment/assessment-report-log.txt (HP-UX)

These are the assessment report locations. They are formatted HTML, text, and a Bastille "config-file" respectively.

/var/log/Bastille/Assessment/Assessment/Drift.txt (Linux)
/var/opt/sec_mgmt/bastille/log/Assessment/Drift.txt (HP-UX)

This contains information about any configuration drift the system had experienced since the last Bastille run. This file will only be created when there has been an earlier Bastille-configuration applied to the system.

SEE ALSO

perl(1), bastille_drift(1M), fstab(4)

Here are some other references used during Bastille's development. Note that the websites and content are maintained by their domain owners. The domain owners are solely responsible for their own sites and contents.

The Linux Security HOWTO
Available at:
One of the best references regarding general Linux Security.

Security Quick-Start HOWTO for Linux
Available at:
This is also a very good starting point for novice users (both to Linux and security).

The Linux Security Administrator's Guide
Available at:
http://seifried.org/lasg/

Securing and Optimizing Linux: RedHat Edition
Available at:
http://www.linuxdoc.org/links/p_books.html#securing_linux

Securing Debian Manual
Available at:
It is provided for offline reading in several formats (Text, HTML and PDF) by installing the harden-doc package in Debian systems.
NAME
bastille_drift - system configuration drift analyzer

SYNOPSIS
Path (Linux): /usr/sbin
Path (HP-UX): /opt/sec_mgmt/bastille/bin
bastille_drift [--from_baseline [baseline]]
bastille_drift [--save_baseline [baseline]]
bastille_drift

DESCRIPTION
bastille_drift is a program for creating Bastille-configuration baselines and comparing the current state of
the system to a saved baseline. This enables the user to see what, if any, changes had occurred relative to
a saved baseline.

Note: When first run successfully, Bastille automatically saves a baseline in the default location (see
FILES below).

Here are the different operations for bastille_drift:
bastille_drift [--from_baseline [baseline]]
    Compare system state to specified (or default) baseline.
bastille_drift [--save_baseline [baseline]]
    Establish or update specified (or default) baseline.
bastille_drift
    Compare system state to default baseline.

DIAGNOSTICS
The following are diagnostics for bastille_drift:

No Baseline exists with which to compare current state.
The default or specified baseline file doesn't exist. Either save a baseline to the named location, if
you'd specified one, or save a baseline to the default location.

Note: bastille will save a baseline to the default location on its first successful run.

Attempt to establish system state not successful.

bastille_drift ran bastille --assessnobrowser to establish system state, but the operation did not
succeed. The bastille_drift error log should contain enough detail to give the user sufficient informa-
tion to prevent reoccurrence.

Note that bastille_drift only detects a state change with regard to a configuration option manipu-
lated Bastille, at the same granularity as that covered by the original Bastille question. Also, in
a number of cases the input config will differ from the saved baseline. This is normal, and most often
involved either manual-action-required questions, questions that don't affect the system state, or cases
where no change was requested of Bastille, but bastille was able to detect and baseline the initial
state of the system.

Also, note that bastille baselines detect the configured state of the system. If only Bastille, SMH, or
SAM are used to configure the system, those will, usually coincide with the dynamic state of the
affected processes as well. In some cases, especially in the case of a manual file edit or configuration
change, bastille_drift may note a state different than the daemon.

Example: A user changed inetd.conf, but forgot to run inetd -c to ask inetd to reread its
configuration file.

If you need to be certain that the dynamic state matches the configured one, reboot the system.

DEPENDENCIES
Perl version 5.8.0 or greater, but 5.8.8 or greater is recommended for best performance.
FILES
/var/opt/sec_mgmt/bastille/baselines (HP-UX)
/etc/Bastille/baselines (Linux)
   Default location for baselines if path not specified.
/var/opt/sec_mgmt/bastille/baselines/default_baseline (HP-UX)
   Default location for baseline if file not specified. This is also where Bastille stores an initial baseline here on its first successful run.
/var/opt/sec_mgmt/bastille/log/Assessment/Drift.txt (HP-UX)
   Location of drift report/diff resulting from assessment.

SEE ALSO
perl(1), bastille(1M).
NAME
bdf - report number of free disk blocks (Berkeley version)

SYNOPSIS
/usr/bin/bdf [-b] [-i] [-l] [-s] [-t type] [filesystem | file] ...

DESCRIPTION
The bdf command displays the amount of free disk space available either on the specified filesystem (/dev/dsk/c0d0s0, for example) or on the file system in which the specified file (such as $HOME), is contained. If no file system is specified, the free space on all of the normally mounted file systems is printed. The reported numbers are in kilobytes.

Options
The bdf command recognizes the following options:
- b Display information regarding file system swapping.
- i Report the number of used and free inodes.
- l Display information for local file systems only (for example, HFS and CDFS file systems).
- s Do not sync the file system data on the disk before reporting the usage.
- t type Report on the file systems of a given type (for example, nfs or hfs).

RETURN VALUE
The bdf command returns 0 on success (able to get status on all file systems), or returns 1 on failure (unable to get status on one or more file systems).

WARNINGS
If file system names are too long, the output for a given entry is displayed on two lines.

The bdf command does not account for any disk space reserved for swap space, or used for the HFS boot block (8 KB, 1 per file system), HFS superblocks (8 KB each, 1 per disk cylinder), HFS cylinder group blocks (1 KB - 8 KB each, 1 per cylinder group), and inodes (currently 128 bytes reserved for each inode). Non-HFS file systems may have other items not accounted for by this command.

AUTHOR
bdf was developed by the University of California, Berkeley.

FILES
/etc/fstab Static information about the file systems.
/etc/mnttab Mounted file system table.
/dev/dsk/* File system devices.

SEE ALSO
df(1M), fstab(4), mnttab(4).
NAME
  boot - bootstrap process

DESCRIPTION

Itanium-based Hardware
  A Itanium®-based system's bootstrap process involves the execution of four software components:
  - CMOS
  - option ROM
  - EFI
  - Boot Manager
  - hpux.efi (see hpux.efi(1M))

After the processor is reset, firmware initializes and tests processors and platform. During initialization, the firmware lets a user interrupt and configure CMOS and option ROMs. It then transfers control to EFI, the Extensible Firmware Interface. EFI, in turn, initializes EFI boot and runtime services and launches the Boot Manager. The Boot Manager, which allows loading of EFI application or drivers from EFI defined file system, loads and transfers control to hpux.efi, the HP-UX-specific bootstrap loader. hpux.efi then loads the HP-UX kernel object file from the HP-UX file system to memory and transfers control to the loaded kernel image.

PA-RISC Hardware
  The PA-RISC server and workstation bootstrap process involves the execution of three software components:
  - pdc (see pdc(1M))
  - isl (see isl(1M))
  - hpux (see hpux(1M))

After the processor is RESET, pdc, the processor-dependent code (firmware), performs a self-test and initializes the processor. It then loads and transfers control to isl, the operating-system-independent initial system loader. isl, in turn, loads and transfers control to the hpux utility, the HP-UX-specific bootstrap loader. hpux then downloads the HP-UX kernel object file from an HP-UX file system and transfers control to the loaded kernel image.

SEE ALSO
  hpux(1M), hpux.efi(1M), isl(1M), pdc(1M), efi(4).
NAME
bootpd - Internet Boot Protocol server

SYNOPSIS
/usr/sbin/bootpd [-d debuglevel] [-p ping-timeout] [-t timeout] [-P] [configfile [dumpfile]]

DESCRIPTION
The bootpd daemon implements three functions: a Dynamic Host Configuration Protocol (DHCP) server as defined in RFC1541, an Internet Boot Protocol (BOOTP) server as defined in RFC951 and RFC1395, and a DHCP/BOOTP relay agent as defined in RFC1542. It also contains some of the useful fields as defined in RFC2132.

bootpd is run through inetd (see inetd(1M)). It is run by /etc/inetd when the following line (or equivalent) is included in the file /etc/inetd.conf:

    bootps dgram udp wait root /usr/sbin/bootpd bootpd

bootpd starts when a boot request arrives. If it has not received another boot request after 500 minutes, bootpd exits. The -t option can be used to specify a different timeout value in minutes (such as -t20). With a timeout value of zero (-t0), bootpd never exits.

The -d option sets the verbosity level (1–3) of the logging emitted by the daemon via syslog (see syslog(3C)). For improved performance, this option should not be used. If this option is not used, no logging is done by syslog except for fatal errors.

By default, the bootpd daemon pings the IP address before assigning the address to a client to check if the IP Address is already in use. The -P option suppresses bootpd from pinging this address.

The -p option can be used to specify the ping timeout period. The server pings for this duration of time to check if the IP address is already in use. The ping-timeout period is specified in milliseconds and the maximum value is 3000 milliseconds. When the -P option is used, the -p option has no effect, since bootpd never pings the IP address.

When bootpd receives a DHCP/BOOTP request, it first checks if the hardware address of the client is listed in the /etc/dhcpdeny database. If yes, this client is denied lease. If the client is not listed in the dhcpdeny database, it checks whether the client information is in the /etc/bootptab database. If the client information is available, bootpd sends back the reply. Otherwise, it checks whether there is any matched relay information for the client in the /etc/bootptab database. If so, bootpd goes through a series of checks to see if it should relay the request. If no matched relay information was found, bootpd checks whether the client information is matched by a pool or device group in the /etc/dhcptab database. If a match is found, bootpd sends back a reply. The request is dropped if no matched group information is found.

To reply to a DHCP or BOOTP request the server puts together a BOOTREPLY message and does a number of checks to ensure the message is sent to the correct destination.

bootpd first checks the ciaddr (client IP address) field of the DHCP/BOOTP packet. If this field is nonzero, the BOOTREPLY message is sent to the IP address identified in ciaddr.

If the ciaddr field is zero, bootpd checks the giaddr field. If this field is not zero, bootpd sends the BOOTREPLY message to the relay agent specified in giaddr field and the relay agent delivers the BOOTREPLY message to the client. If the giaddr field is zero, bootpd sends the BOOTREPLY message to the client. In both cases, the BOOTREPLY will either be sent to the IP address specified in the yiaddr (your IP address) field or as a broadcast message. On HP-UX, there are two ways to specify that the BOOTREPLY should be sent as a broadcast message.

1. The client sets the broadcast flag bit in the flag field (bit 0) of the DHCP/BOOTP request packet.
2. Define the ba tag in the bootptab file (see Tags for client entries below)

For the case where the bootpd has matched a relay entry in /etc/bootptab, it attempts to forward the request to the configured DHCP/BOOTP server.

bootpd first checks whether the relay function is enabled for the requesting client. The relay capability is configurable. If the relay function is disabled, then the request packet is dropped.

Before bootpd relays the request, it also examines the giaddr (gateway IP address) field. The client sets the giaddr field to zero when it sends out the request. If the relay agent finds this field is zero, it fills this field with the primary IP address of the interface on which the request was received; otherwise, the relay agent does not change this field. Then bootpd increments the value of the hops field, and
bootpd(1M)

relays the request to the DHCP/BOOTP servers that have been configured for this client.

If the relay function is enabled for this client, bootpd checks the hop field of the DHCP/BOOTP request
packet. The client sets the hop field to 0 when it sends out the DHCP/BOOTP request. The hop value
is increased every time the request packet is relayed by a relay agent. The maximum hop number can be
configured. The maximum possible hop number allowed is 16. The default maximum is set to 4. The
request packet is dropped if the hop value exceeds the configured maximum.

Then bootpd compares the value of the secs (seconds since the client began booting) field of the
DHCP/BOOTP packet to the threshold value. The client sets the secs field to zero when it first sends
out the request. The client repeats the request if it does not receive a reply. When the client repeats the
requests, it sets the secs value to the number of seconds since the first request was sent. bootpd does
not relay the request if the value of the secs field is less than the threshold value. The threshold
value can be configured. The default value is 0.

Configuration

Upon startup, bootpd reads its configuration files to build its internal database, then listens for boot
request packets. The default configuration files are, /etc/dhcpdeny, /etc/bootptab, and
/etc/dhcptab. The bootptab file can be specified in the command line. bootpd rereads its
configuration file when it receives a hangup signal, SIGHUP, or when it receives a boot request packet and
detects that the configuration file has been updated. If hosts are added, deleted, or modified, their entries
in the bootptab internal database are updated accordingly when the configuration files are reread. The
/etc/dhcpdeny database contains the list of hardware addresses of the clients that will not be served by
this server.

If bootpd receives a SIGUSR1 signal, it dumps its memory-resident database to the file
/var/tmp/bootpd.dump or the dumpfile specified in the command line.

The configuration file can contain two types of host entries:

1. The client entries, which contains the client information.

2. The relay entries, which contains the configuration to relay DHCP/BOOTP requests for one or more
   clients.

The configuration uses two-character, case-sensitive tag symbols to represent host parameters. These
parameter declarations are separated by colons (:`). The general format is:

    hostname tg=value:...tg=value:...tg=value:...

where hostname is the actual name of a DHCP/BOOTP client in the client entries, and in the case of a relay
entry, it can be the actual name of a client if it is an individual relay entry, or it can be a name for a group
of clients if it is a group relay entry. tg is a two-character tag symbol. Most tags must be followed by an
equals-sign, and a value as above. Some can appear in a boolean form with no value (that is, `tg`).

Blank lines and lines beginning with `#` are ignored in the configuration file. Host entries are separated
from one another by newlines; a single host entry can be extended over multiple lines if the lines end with a
backslash (`\`). It is also acceptable for lines to be longer than 80 characters. Tags can appear in any order
with the following exceptions: The host name must be the very first field in an entry, and the hardware
type tag, ht, must precede the hardware address tag, ha, and the hardware mask tag, hm.

IP addresses are specified in standard Internet dot notation, and can use decimal, octal, or hexadecimal
numbers (octal numbers begin with 0, hexadecimal numbers begin with 0x or 0X). Certain tags accept a
list of one or more IP addresses (ip_address_list). When more than one IP address is listed, the addresses
must be separated by whitespace.

The types of tags can be grouped into three categories:

1. The tags that can be used for both the client and the relay entries.

2. The tags that can only be used in the relay entries.

3. The tags that can only be used in the client information entries.

Tag ip is used to differentiate a client entry from a relay entry. An entry with tag ip defined is treated as
a client entry. A relay entry can contain the relay configuration for an individual client, also a hardware
address mask mechanism is provided to configure the relay entry for a group of clients. The group client
relay entries are kept in a linear sorted table by bootpd. When a client does not have an individual relay
specification, the linear table is searched to see if there is a match for the client. If there are multiple
matched entries in the sorted table, only the first one is used. Tag ha is used to differentiate an individual
client relay entry from a group relay entry. The linear sorted table is sorted on the value of tag \texttt{hm}. The search and match mechanism is explained in the discussion of tag \texttt{hm}.

**Tags for both kinds of entries**

\texttt{ha=hardware-address}

This tag specifies the hardware address of the client. The \textit{hardware address} must be specified in hexadecimal; optional periods and/or a leading \texttt{0x} can be included for readability. The \texttt{ha} tag must be preceded by the \texttt{ht} tag (either explicitly or implicitly; see \texttt{tc} below).

\texttt{ht=hardware-type}

This tag specifies the hardware type code. \textit{hardware-type} can be an unsigned decimal, octal, or hexadecimal integer corresponding to one of the ARP Hardware Type codes specified in RFC1010. It can also be specified by the symbolic names \texttt{ethernet} or \texttt{ether} for 10-Mb Ethernet; \texttt{ethernet3} or \texttt{ether3} for 3-Mb experimental Ethernet; \texttt{ieee802}, \texttt{tr}, or \texttt{token-ring} for IEEE 802 networks; \texttt{pronet} for Proteon ProNET Token Ring; \texttt{chaos}, and \texttt{arcnet}, for Chaos and ARCNET, respectively.

\texttt{tc=template-host}

This tag indicates a table continuation. Often, many host entries share common values for certain tags (such as domain servers, etc.). Rather than repeatedly specifying these tags, a full specification can be listed for one host entry and shared by others via the \texttt{tc} mechanism.

The \texttt{template-host} is a dummy host that does not actually exist and never sends boot requests. Information explicitly specified for a host always overrides information implied by a \texttt{tc} tag symbol. The value of \texttt{template-host} can be the host name or IP address of any host entry previously listed in the configuration file.

Sometimes it is necessary to delete a specific tag after it has been inferred via \texttt{tc}. This can be done using the construction \texttt{tag@} which removes the effect of \texttt{tag}. For example, to completely undo an RFC1034 domain name server specification, use \texttt{:ds@:} at an appropriate place in the configuration entry. After removal with \texttt{@}, a tag is eligible to be set again through the \texttt{tc} mechanism.

**Tags for relay entries**

\texttt{bp=bootp-servers}

This tag specifies the BOOTP servers that DHCP/BOOTP requests will be relayed to. The value of \texttt{bootp-servers} can be one or more individual IP addresses, and/or one or more network broadcast addresses. A relay entry with this tag configured indicates that the relay function is on for the clients specified in this entry. A relay entry missing this symbol means that the relay function is off for the clients specified in this entry.

\texttt{th=threshold}

This tag specifies the \textit{threshold} value in seconds for the entry. The default value is 0.

\texttt{hp=hops}

This tag specifies the maximum \textit{hops} value. If the \texttt{hops} value exceeds 16, it is set to 16. The default value is 4.

\texttt{hm=hardware-address-mask}

This tag specifies the mask for the hardware address \texttt{ha}. \textit{hardware-address-mask} must be specified in hexadecimal. An optional leading \texttt{0x} can be included for readability. The \texttt{hm} tag must be preceded by the \texttt{ht} tag (either explicitly or implicitly; see \texttt{tc} above). Each \texttt{0} bit in \texttt{hm} specifies that the corresponding bit in \texttt{ha} is a “don’t-care” bit, each \texttt{1} bit in \texttt{hm} specifies that the corresponding bit in the \texttt{ha} value is ANDed with the \texttt{hm} value. If the result is the same and also the hardware type matches, then a match is found. For example,

\begin{verbatim}
if (((hm & ha)==(client_hw_addr & hm))
  && (ht == client_hw_type))
  then a match is found
else continue the search
\end{verbatim}

**Tags for client entries**

\texttt{ba}

This tag specifies that \texttt{bootpd} should broadcast the boot reply to the client. As a boolean tag, it causes \texttt{bootpd} to send the boot reply on the configured broadcast address of each network interface. You can also assign the tag an IP-address value, which specifies the specific IP or broadcast address for the boot reply.
bf=filename
This tag specifies the filename of the bootfile that the client should download. The client’s boot request, and the values of the hd (see below) and bf symbols, determine the contents of the bootfile field in the boot reply packet.

If the client specifies an absolute path name (in its boot request), and that file is accessible on the server machine (see below), bootpd returns that path name in the reply packet. If the file is not accessible, the request is discarded; no reply is sent. If the client specifies a relative path name, bootpd constructs a full path name by appending the relative path name to the value of the hd tag, and tests to determine if the full path name is accessible. If the full path name is accessible, it is returned in the boot reply packet; if not, the request is discarded.

Clients that do not specify boot files in their boot requests always elicit a reply from the server. The exact reply depends on the values of the hd and bf tags. If the bf tag specifies an absolute path name, and the file is accessible, that path name is returned in the reply packet. Otherwise, if the hd and bf tags together specify an accessible file, that file name is returned in the reply.

If a complete file name cannot be determined, or the file is not accessible publicly, the reply contains a zeroed-out bootfile field.

If the tftp pseudo-user exists, bootpd treats all path names (absolute or relative) as being relative to the home directory of tftp and checks there first. If the file is not accessible under the tftp home directory or the tftp pseudo-user does not exist, bootpd checks for the file relative to /.

For a file to be available, it must exist, and be publicly readable.

All file names are first tried as filename.hostname and then simply as filename. However, in the case when the tftp pseudo-user exists, but filename.hostname and filename are not accessible under the tftp home directly, only filename is checked relative to /.

Note that a file considered to be accessible relative to / might not actually be accessible via tftp if the command line arguments to tftpd disallow that path.

bs=size
This tag specifies the size of the bootfile. The parameter size can be either a decimal, octal, or hexadecimal integer specifying the size of the bootfile in 512-octet blocks, or the keyword auto, which causes the server to automatically calculate the bootfile size at each request. Specifying the bs symbol as a boolean has the same effect as specifying auto as its value.

ci=client_ID
This tag specifies the client identifier of the client. The parameter client_ID can be either a hexadecimal integer, or a string contained in double quotes. The client_ID is a unique identifier that the DHCP client may use to identify itself to the server. If present, the client identifier supersedes the hardware address, so a client and an entry will only match in one of two situations: one, they both have the same client identifier, or two they both have the same hardware address and neither has a client identifier. If a request has a client identifier, then that is used to match the client up with an entry in the bootp configuration file. One common client ID used is to concatenate the hardware type (e.g. 0x01 for ethernet) with the hardware address.

cs=ip_address_list
This tag specifies the IP addresses of RFC865 Quote of the Day (cookie) servers.

dn=domain_name
This tag specifies the domain name of the client for Domain Name Server resolution (see RFC1034).

ds=ip_address_list
This tag specifies the IP addresses of RFC1034 Domain Name servers.

ef=filename
Specifies the name of an extensions file. The file, retrievable via TFTP, contains information which can be interpreted in the same way as the 64-octet vendor-extension field within the BOOTP response. The maximum length of the file is unconstrained. All references to an extensions filename within the file are ignored.

gw=ip_address_list
This tag specifies the IP addresses of gateways for the client’s subnet. If one of multiple gateways is preferred, it should be listed first.
**hd**=
This tag specifies a directory name to which the bootfile is appended (see the **bf** tag above). The default value of the **hd** tag is "/

**hn**
The presence of this tag indicates that the client's host name should be sent in the boot reply. The **hn** tag is a boolean tag. **bootpd** attempts to send the entire host name as it is specified in the configuration file or hosts database. The configuration file is checked first, if the host name is not found, the hosts(4) database is then checked. If the hostname cannot fit into the reply packet, an attempt is made to shorten the name to just the host field (up to the first period, if present) and then tried. In no case is an arbitrarily truncated host name sent. If nothing reason-able can fit, nothing is sent.

**im**=
This tag specifies the IP addresses of Impress network image servers.

**ip**=
This tag specifies the IP address of the DHCP/BOOTP client.

**lg**=
This tag specifies the IP addresses of MIT-LCS UDP log servers.

**lp**=
This tag specifies the IP addresses of Berkeley 4BSD printer servers.

**md**=
This tag specifies the name of a file to dump the core of a client.

**ms**=
This tag specifies the IP address(es) of SMTP servers available to the client (RFC2132).

**na**=
This tag specifies the IP address(es) of RFC 1001/1002 NetBIOS name server(s) in order of preference.

**nb**=
This tag specifies the IP address(es) of RFC 1001/1002 NetBIOS datagram distribution server(s) in order of preference.

**nc**=
Specifies the NetBIOS node type code. Allows NetBIOS over TCP/IP clients to be configured as described in RFC1001/1002. The **NetBIOS_node_type** can be an unsigned decimal, octal, or hexa-decimal integer corresponding to one of the client types as follows:

- **0x1** or **B-node** for B-node;
- **0x2** or **P-node** for P-node;
- **0x4** or **M-node** for M-node;
- **0x8** or **H-node** for H-node.

**nd**=
This tag specifies the NetBIOS over TCP/IP scope parameter for the client as specified in RFC 1001/1002.

**ns**=
This tag specifies the IP addresses of IEN-116 name servers.

**nt**=
This tag specifies the IP addresses of Network Time Protocol servers. Servers should be listed in order of preference.

**pd**=
This tag specifies the name of clients NIS+ domain name (RFC2132).

**ps**=
This tag specifies the IP address(es) of NIS+ servers available to the client (RFC2132).

**rl**=
This tag specifies the IP addresses of RFC887 Resource Location Protocol servers.

**rp**=
This tag specifies a path name to be mounted as a root disk.
**bootpd(1M)**

**sa=tftp_server**
This tag specifies the IP address of the TFTP server where the client's bootfile resides. When this option is enabled, `bootpd` uses the IP address specified in this tag for the `siaddr` field in a BOOTP/DHCP packet header. Otherwise, the IP address of the BOOTP/DHCP server is used in the `siaddr` field. The `sa` tag allows the BOOTP/DHCP server and the TFTP server to be two different systems, if desired.

**sm=subnet-mask**
This tag specifies the client's subnet mask. `subnet-mask` is specified as a single IP address.

**sr=destination_ip_address gateway_ip_address ...**
This tag specifies a list of static routes that the client should put in its routing cache. Each route consists of a pair of IP addresses. The first address is the destination address, and the second is the router. Use the `gw=` option to specify the default route (0.0.0.0) as it is not a legal destination address.

**ss=ip_address**
This tag specifies the IP address of a swap server.

**Tnnn=generic-data**
This is a generic tag where `nnn` is an RFC1533 option field tag number. Use this option to configure RFC1533 options not currently supported with `bootpd` tag names. This option allows one to immediately take advantage of future extensions to RFC1533. The `generic-data` data can be represented as either a stream of hexadecimal numbers or as a quoted string of ASCII characters. The length of the generic data is automatically determined and inserted into the proper fields of the RFC1541-style boot reply.

**to=offset**
This tag specifies the client’s time zone offset in seconds from UTC. The time `offset` can be either a signed decimal integer or the keyword `auto` which uses the server's time zone offset. Specifying the `to` symbol as a boolean has the same effect as specifying `auto` as its value.

**ts=ip_address_list**
This tag specifies the IP addresses of RFC868 Time Protocol servers.

**yd=NIS-domain-name**
Specifies the name of the client's NIS domain.

**ys=ip_address_list**
Specifies the IP address(es) of NIS servers available to the client. Servers should be listed in order of preference.

**vm=magic-cookie**
This tag specifies the RFC1048 vendor information magic cookie. `magic-cookie` can be one of the following keywords: `auto` (indicating that vendor information is determined by the client's request), `rfc1048` (which always forces an RFC1048-style reply), or `cmu` (which always forces a CMU-style reply).

**Vnnn=generic-data**
This is a generic tag for vendor specific information where `nnn` is a vendor defined option field tag number. The `generic-data` data can be represented as either a stream of hexadecimal numbers or as a quoted string of ASCII characters. The length of the generic data is automatically determined and inserted into the vendor specific field of the RFC1541-style boot reply.

**xd=ip_address_list**
This tag specifies the IP addresses of systems that are running the X Window System Display Manager and are available to the client. Addresses should be listed in order of preference.

**xf=ip_address_list**
This tag specifies the IP addresses of X window System font servers available to the client. Servers should be listed in order of preference.

**Dhcpdeny Configuration**
The configuration file `/etc/dhcpdeny` contains the list of hardware addresses, one address per line, for clients that will not be served by our server. If we know about some bad clients in the network and we don't want to serve them, add the hardware address of those clients in this file. This file, like other configuration files, takes `#` character as the starting of a comment.
Dhcptab Configuration
The configuration file /etc/dhcptab defines groups of IP addresses that to be leased out to clients. It also specifies certain general behaviors of the server, such as whether or not to give addresses from these groups to bootp clients or only to DHCP clients.

The configuration file has a format similar to the /etc/bootptab configuration file, with a keyword followed by one or more tag symbols. These tag symbols are separated by colons (:). The general format is:

```
keyword: tg=value: ...:tg=value: ...
```

where `keyword` is one of four allowed (non-case-sensitive) symbols and `tg` is a two or more (case-sensitive) character tag symbol. Most tags must be followed by an equals-sign and a value as above. Some can also appear in a boolean form with no value (i.e. `tg:`).

Blank lines and lines beginning with `#` are ignored in the configuration file. Keyword entries are separated from one another by newlines; a single host entry may be extended over multiple lines if each continued line ends with a backslash (`\`). Lines may be longer than 80 characters. Tags can appear in any order.

IP addresses must be specified in standard Internet “dot” notation, and can use decimal, octal, or hexadecimal numbers (octal numbers begin with `0`, hexadecimal numbers begin with `0x` or `0X`). Certain tags accept a list of one or more IP addresses (`ip_address_list`). When more than one IP address is listed, they must be separated by white space.

The currently recognized keywords are:

- dhcp_pool_group
  This keyword is followed by tags defining a group of IP addresses to give out to clients on the same subnet, and the characteristics of that group. In addition to the tags defined for DHCP groups, all of the two-letter tags for bootp entries may also be used (except for `ht`, the hardware type tag, `ha`, the hardware address tag, or `ci`, the client ID tag. Required tags are: `subnet-mask`, `addr-pool-start-address`, and `addr-pool-last-address`.

- dhcp_device_group
  This keyword is used to define a group of IP addresses on a subnet much like `dhcp_pool_group`, but with one exception: all clients in a device group must have the same client class (specified with tag `class-id`). This allows different types of clients to receive different parameters from the server. Required tags are: `class-id`, `subnet-mask`, `addr-pool-start-address`, and `addr-pool-last-address`.

- dhcp_default_client_settings
  This keyword is followed by tags to be applied to all groups. These tag values can be overridden for a specific group if that tag is defined for that specific group. This keyword simply saves one from entering the same tag for every group. Thus most tags that may be used for `dhcp_pool_group`, and `dhcp_device_group`, may be used here. The tag descriptions specify if a tag may not be used here.

- dhcp_server_settings
  This keyword is followed by tags that specify a few general behaviors for the dhcp server as a whole.

The currently supported tags for `dhcp_server_settings` are:

- `dhcpdb-write-perf`:
  This parameter takes a small integer (like 2 or 5) as input. If set, the write to the `/etc/dhcpdb` file will be delayed by the server. This will increase performance for busy servers. If set to a value greater than 2, the server will spawn a new process to do the writing, which will be a considerable performance improvement.

- `callback-style=OLD | NEW`
  Callbacks are a powerful feature that allow the system administrator to customize the operation of the server. A user-supplied executable file (typically a shell script) is executed each time one of the main server actions is performed (example: granting a lease). An argument list is passed in with information about the individual client and the lease. The `callback-style` tag specifies whether the `old` (and confusing) argument list should be used with the `call-on-xxx` feature described below. The `new` (and recommended) argument list is much simpler to use, and is identical for all of the `call-on-xxx` functions. The `new` style simply inserts a value of "00" for fields that are not sensible for a particular callback. The `new` argument list is:
The old argument list is described for each of the individual callbacks below.

**call-on-unrequited=filename**

This tag specifies an executable file `filename` that will be called when the server receives a request to which it cannot send a response. Certain arguments will be passed in; the call executed will be:

```
filename: client-id htype haddr [gateway]
```

where `client-id` is the client ID in hex if present, or 00 if there is no client ID. `htype` is the hardware type as per the ARP section of the "Assigned Numbers" RFC. `haddr` is the hardware address in hex. `gateway` is the IP address of the `bootp` relay agent. If the packet was not relayed, then this field is absent.

The currently supported tags for `dhcp_pool_group`, `dhcp_device_group`, and `dhcp_default_client_settings` are:

**call-on-assignment=filename**

This tag specifies the fully qualified `filename` to be called when an IP address has been assigned to a new client. Some arguments will be passed in, the call will be made as follows:

```
filename: client-id htype haddr ipaddr subnet-mask lease-expiration [hostname]
```

where `client-id` is the client ID in hex if present, or 00 if there is no client ID. `htype` is the hardware type as per the ARP section of the "Assigned Numbers" RFC. `haddr` is the hardware address in hex. `ipaddr` is the IP address that was assigned to the client. `subnet-mask` is the subnet mask of the client represented as an IP address. `lease-expiration` is the `bootpd` internal representation of when the lease will expire (based on a C call to time()), a value of `ffffffff` represents an infinite lease. If there is a `hostname` associated with this address, then it is the final argument.

**call-on-decline=filename**

This tag specifies the fully qualified `filename` to be called when an IP address has been declined by a new client. Some arguments will be passed in, the call will be made as follows:

```
filename: client-id htype haddr ipaddr subnet-mask
```

where `client-id` is the client ID in hex if present, or 00 if there is no client ID. `htype` is the hardware type as per the ARP section of the "Assigned Numbers" RFC. `haddr` is the hardware address in hex. `ipaddr` is the IP address that was declined by the client. `subnet-mask` is the subnet mask of the client represented as an IP address.

**call-on-discard=filename**

This tag specifies the fully qualified `filename` to be called when an IP address has been discarded due to a conflict. Some arguments will be passed in, the call will be made as follows:

```
filename: client-id htype haddr ipaddr subnet-mask
```

where `client-id` is the client ID in hex if present, or 00 if there is no client ID. `htype` is the hardware type as per the ARP section of the "Assigned Numbers" RFC. `haddr` is the hardware address in hex. `ipaddr` is the IP address that was declined by the client. `subnet-mask` is the subnet mask of the client represented as an IP address.

**call-on-release=filename**

This tag specifies the fully qualified `filename` to be called when an IP address has been released by a client. Some arguments will be passed in, the call will be made as follows:

```
filename: client-id htype haddr ipaddr lease-expiration
```

where `client-id` is the client ID in hex if present, or 00 if there is no client ID. `htype` is the hardware type as per the ARP section of the "Assigned Numbers" RFC. `haddr` is the hardware address in hex. `ipaddr` is the IP address that was released by the client. `lease-expiration` is the `bootpd` internal representation of when the lease would have expired, a value of `ffffffff` represents an infinite lease.

**call-on-lease-extend=filename**

This tag specifies the fully qualified `filename` to be called when an IP address lease for a client has been extended. Some arguments will be passed in, the call will be made as follows:
filename: client-id htype haddr ipaddr subnet-mask lease-expiration

where client-id is the client ID in hex if present, or 00 if there is no client ID. htype is the hardware type as per the ARP section of the "Assigned Numbers" RFC. haddr is the hardware address in hex. ipaddr is the IP address that was assigned to the client. subnet-mask is the subnet mask of the client represented as an IP address. lease-expiration is the bootpd internal representation of when the lease will expire (based on a C call to time()), a value of ffffffff represents an infinite lease.

call-on-discover=filename
This tag specifies the fully qualified filename to be called when the server receives a discover. It should be noted that this callback can only be used when callback-style is set to new. The format of the arguments passed to this callback is same as the format specified for callback-style=new. If a particular parameter is not known or not required, 00 can be used in it's place.

call-on-offer=filename
This tag specifies the fully qualified filename to be called when the server sends an offer to a client. It should be noted that this callback can only be used when callback-style is set to new. The format of the arguments passed to this callback is same as the format specified for callback-style=new. If a particular parameter is not known or not required, 00 can be used in it's place.

class-name=classname
This tag specifies a name to refer to a device group by. It is only applicable to dhcp_device_group. The only use that bootpd makes of this field is in logging errors found in the configuration of the group.

pool-name=poolname
This tag specifies a name to refer to a pool group by. It is only applicable to dhcp_pool_group. The only use that bootpd makes of this field is in logging errors found in the configuration of the group.

class-id=client-class
This tag specifies the client-class that clients must have to be assigned to this group. This tag is required for dhcp_device_group and is inappropriate for any other keyword. Some DHCP clients send out a client-class that identifies a class that a client belongs to. For an IP address to be assigned from a device group address pool, not only must the client be on the right subnet, it must send a request with a client-class that matches that defined for the class-id. This may be specified in either hex or in ASCII (an ASCII string must be enclosed in double quotes).

cpyid
This is a boolean tag that instructs bootpd not to send the class-id back to the client. This tag is applicable only for dhcp_device_group.

re
This is a boolean tag that instructs the bootpd to match the class-id in the client's request with the class-id in any dhcp_device_group, that contains the re tag using any basic regular expression. This tag is applicable only for dhcp_device_group.

subnet-mask=mask
This tag specifies the subnet mask for the addresses in the group being defined. It is specified as an IP address. This tag is required for both dhcp_device_group and dhcp_pool_group, and is inappropriate for dhcp_default_client_settings.

addr-pool-start-address=ip-address
This tag specifies the lowest address in the pool group to be assigned. This tag is required for both dhcp_device_group and dhcp_pool_group, and is inappropriate for dhcp_default_client_settings.

addr-pool-last-address=ip-address
This tag specifies the highest address in the pool group to be assigned. This address and the addr-pool-start-address define a range of addresses that can be assigned to clients. For the server, no two group address ranges may overlap.

reserved-for-other=ip-address-list
This tag is followed by one address that falls in the range of the group. This address is reserved, and will not be assigned to any clients by the DHCP server. Alternatively, a range of addresses may be defined by giving 2 addresses, with the range being the addresses from the first address up to the second address, inclusively. This tag may be repeated to reserve more addresses.
addresses in the same group. It is not appropriate for dhcp_default_client_settings.

lease-time=seconds
This tag specifies the time in seconds that a lease should be given to each client. The word "infinite" may be used to specify leases that never expire. The default is "infinite." Note that if a client asks for a shorter lease than is configured for it, will get that shorter lease time. A lease time shorter than 120 seconds will be silently upgraded to 120.

lease-grace-period=percent
This tag specifies the time after a lease expires during which that lease will not be assigned to a new client. percent is the percentage of the configured lease time that this grace period lasts. The default is 5%.

tr=percent
This tag specifies the DHCP IP lease renewal time (T1). This is the time interval from lease assignment until when the client attempts to renew the lease. RFC1541 states that T1 defaults to half the lease duration. The minimum value is 40 percent. T1 must always be smaller than T2.

tv=percent
This tag specifies the DHCP IP lease rebind time (T2). This is the time interval from lease assignment until when the client attempts to obtain a new lease from any server. RFC1541 states that T2 defaults to 0.875 times the lease duration. The minimum value is 50 percent. T2 must always be greater than T1.

lease-policy=policy
This tag specifies whether or not the assigning of new leases can be done. If policy is set to reject-new-clients then no new clients can get a lease, and only clients with existing leases will get a response. accept-new-clients is the default.

allow-bootp-clients=boolean
This tag specifies whether or not bootp clients can be members of the group being defined. The default is false. If boolean is TRUE, then an IP address may be assigned to a client that doesn't have an entry in the bootptab file and that is on the same subnet as the group being defined. This address is treated as an infinite lease, and a boot reply is sent to the client. This tag is not appropriate for dhcp_device_group, since bootp clients don't have a client class (and therefore a bootp client would be incapable of matching the client class of the device group). If this tag is used for dhcp_default_client_settings, then it is only applicable to pool groups.

ddns=ip-address
This tag specifies the IP address of the Domain Name Server (DNS) to which dynamic update requests are sent.

pcsn
This tag specifies that the name sent by client should be given preference. As a boolean tag, if set it causes bootpd to accept the name sent by the client (if any). If name is not sent by the client, bootpd tries to find one.

sp
As a boolean tag, if set it causes bootpd to not use pre-requisite section in the update request when an update request is to be sent to DNS.

DHCP/BOOTP Packet
The DHCP/BOOTP packet has the following format:

```c
struct dhcp {
    unsigned char op;        /* packet opcode type */
    unsigned char htype;     /* hardware addr type */
    unsigned char hlen;      /* hardware addr length */
    unsigned char hops;      /* gateway hops */
    unsigned long xid;       /* 4 bytes transaction ID */
    unsigned short secs;     /* seconds since boot began */
    unsigned short flags;    /* if giaddr!=0,client flags*/
    struct in_addr ciaddr;   /* client IP address */
    struct in_addr yiaddr;   /* 'your' IP address */
    struct in_addr siaddr;   /* server IP address */
};
```
DHCP Option Numbers

The DHCP/BootP options discussed above correspond to the option numbers in RFC1533 as follows:

<table>
<thead>
<tr>
<th>Number</th>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>sm</td>
<td>Subnet Mask</td>
</tr>
<tr>
<td>2</td>
<td>to</td>
<td>Time Offset</td>
</tr>
<tr>
<td>3</td>
<td>gw</td>
<td>Gateways</td>
</tr>
<tr>
<td>4</td>
<td>ts</td>
<td>Time Servers</td>
</tr>
<tr>
<td>5</td>
<td>ns</td>
<td>IEN 116 Name Servers</td>
</tr>
<tr>
<td>6</td>
<td>ds</td>
<td>Domain Name Servers</td>
</tr>
<tr>
<td>7</td>
<td>lg</td>
<td>Log Servers</td>
</tr>
<tr>
<td>8</td>
<td>cs</td>
<td>Cookie Servers</td>
</tr>
<tr>
<td>9</td>
<td>lp</td>
<td>LPR Servers</td>
</tr>
<tr>
<td>10</td>
<td>im</td>
<td>Impress Servers</td>
</tr>
<tr>
<td>11</td>
<td>rl</td>
<td>Resource Location Servers</td>
</tr>
<tr>
<td>12</td>
<td>hm</td>
<td>Send Host Name in reply</td>
</tr>
<tr>
<td>13</td>
<td>bs</td>
<td>Boot File Size</td>
</tr>
<tr>
<td>14</td>
<td>md</td>
<td>Merit Dump File</td>
</tr>
<tr>
<td>15</td>
<td>dn</td>
<td>Domain Name</td>
</tr>
<tr>
<td>16</td>
<td>ss</td>
<td>Swap Server</td>
</tr>
<tr>
<td>17</td>
<td>rp</td>
<td>Root Path</td>
</tr>
<tr>
<td>18</td>
<td>ef</td>
<td>Extensions Path</td>
</tr>
<tr>
<td>28</td>
<td>ba</td>
<td>Broadcast Address</td>
</tr>
<tr>
<td>33</td>
<td>sr</td>
<td>Static Routes</td>
</tr>
<tr>
<td>40</td>
<td>yd</td>
<td>NIS Domain</td>
</tr>
<tr>
<td>41</td>
<td>ys</td>
<td>NIS Servers</td>
</tr>
<tr>
<td>42</td>
<td>nt</td>
<td>NTP Servers</td>
</tr>
<tr>
<td>43</td>
<td>v###</td>
<td>Vendor Specific Information</td>
</tr>
<tr>
<td>44</td>
<td>na</td>
<td>NetBIOS Name Servers</td>
</tr>
<tr>
<td>45</td>
<td>nb</td>
<td>NetBIOS Datagram Distribution Servers</td>
</tr>
<tr>
<td>46</td>
<td>nc</td>
<td>NetBIOS Node Type</td>
</tr>
<tr>
<td>47</td>
<td>nd</td>
<td>NetBIOS Scope</td>
</tr>
<tr>
<td>48</td>
<td>xf</td>
<td>X Font Servers</td>
</tr>
<tr>
<td>49</td>
<td>xd</td>
<td>X Display Manager</td>
</tr>
<tr>
<td>51</td>
<td>lease-time</td>
<td>IP Address Lease Time</td>
</tr>
<tr>
<td>52</td>
<td>tr</td>
<td>Lease Renewal Time (T1)</td>
</tr>
<tr>
<td>59</td>
<td>tv</td>
<td>Lease Rebinding Time (T2)</td>
</tr>
<tr>
<td>60</td>
<td>class-id</td>
<td>Class Identifier</td>
</tr>
<tr>
<td>61</td>
<td>ci</td>
<td>Client Identifier</td>
</tr>
<tr>
<td>64</td>
<td>pd</td>
<td>NIS+ Domain</td>
</tr>
<tr>
<td>65</td>
<td>ps</td>
<td>NIS+ Servers</td>
</tr>
<tr>
<td>69</td>
<td>ms</td>
<td>SMTP Servers</td>
</tr>
</tbody>
</table>

EXAMPLES

This is an example of a /etc/bootptab file:

```bash
# Common entry

global.defaults:\
  bf=C2300A:\
  hd=/usr/lib/X11/:\
  hn:\
  ht=ether:\
  vm=rfc1048

# Now the actual individual entries
```
xterm1:\
  tc=global.defaults:
  ha=08000903212F:
  ip=190.40.101.22

xterm2:\
  tc=global.defaults:
  ha=0800090324AC:
  ip=190.40.101.35

# Common relay entry.
relay-default:\
  ht=ethernet:
  bp=15.4.3.136 15.13.6.192:
  th=2:
  hp=5:

# Relay entry for node2
node2:\
  tc=relay-default:
  ha=08000902CA00:

# Group relay entry
group-machines:\
  tc=relay-default:
  ha=080009000000:
  hm=080009000000:

# Turn the relay off (block the relay) for the following machines.
blocked-machines:\
  ht=ethernet:
  ha=07000A000000:
  hm=07000A000000:

# Relay definition for all other machines.
all:\
  tc=relay-default:
  ha=000000000000:
  hm=000000000000:

This is an example of a /etc/dhcptab file:

# The first entry is for options which define the server’s operation.

DHCP_SERVER_SETTINGS:\
  call-on-unrequited="/tmp/unrequited.script" :
  dhcpdb-write-perf= 3 :

# The next entry is for options that will be applied to all groups.
# Individual options may be overridden for a specific group if the group
# also configures the option.

DHCP_DEFAULT_CLIENT_SETTINGS:\
  hn:
  lease-time=10080:

# The next entry defines an address pool for devices with the class
# id "xterminal" on subnet 15.14.128. Address leases will be granted
# for up to 1 week. The server will use a broadcast message to
# respond to all client requests.

DHCP_DEVICE_GROUP:
  ba:
    class-name=SUBNET_128_XTERMINAL_GROUP:
    class-id="xterminal:" 
    subnet-mask=255.255.255.0 : 
    addr-pool-start-address= 15.14.128.1 :
    addr-pool-last-address= 15.14.128.254 :
    lease-time=604800 :
    lease-grace-period=5 :

# The next entry grants IP leases to any device on subnet
# 15.13.128. The script /usr/local/bin/assignment.script will be
# run whenever a new lease is granted.

DHCP_POOL_GROUP:
  pool-name=RED_SUBNET_POOL:
  call-on-assignment="/usr/local/bin/assignment.script" :
  subnet-mask=255.255.255.0 :
  addr-pool-start-address= 15.13.128.100 :
  addr-pool-last-address= 15.13.128.254 :
  gw=15.13.128.1 :

This is an example of a /etc/dhcpdeny file:

# Sample /etc/dhcpdeny file
080009000001
0x080009000002
# Above two clients will not be served!

WARNINGS
  Individual host entries must not exceed 1024 characters.

AUTHOR
  bootpd was developed by Carnegie Mellon University, Stanford University, and HP.

FILES
  /etc/bootptab
  /etc/dhcptab
  /etc/services

SEE ALSO
  bootpquery(1M), dhcptools(1M), inetd(1M), tftpd(1M), syslog(3C), hosts(4).

DARPA Internet Requests For Comments: RFC865, RFC868, RFC887, RFC951, RFC1010, RFC1034,
NAME
bootpquery - send BOOTREQUEST to BOOTP server

SYNOPSIS
/usr/sbin/bootpquery haddr [htype] [options]

DESCRIPTION
bootpquery is a diagnostic function used to check the configuration of the Internet Bootstrap Protocol (BOOTP) server, bootpd(1M). This function can only be run by the superuser, since it uses reserved ports.

bootpquery constructs a boot request with the supplied parameters to send to the BOOTP server, and prints the contents of the BOOTP server reply (as shown in EXAMPLES below). Note that bootpquery formats and prints RFC-1048 or CMU-style vendor information included in the BOOTREPLY.

The BOOTREQUEST packet is broadcast on the BOOTP server port, bootps. If a BOOTP server is configured to respond to the request, it returns a BOOTREPLY packet on the BOOTP client port, bootpc.

bootpquery can only display BOOTREPLY packets when the BOOTP server broadcasts the reply on the client port or when the hardware address and IP address supplied in the BOOTREQUEST are those of the host on which bootpquery is run.

The following options provide the information for the BOOTREQUEST:

- haddr Hardware address of the BOOTP client; used in the BOOTREQUEST. A BOOTP server responds if it has configuration information for a host with this link-level address.
- htype Type of address specified as haddr; may be ether or ieee802. The default address type is ether.
- ipaddr Specify the internet address of the BOOTP client to be used in the BOOTREQUEST. If the BOOTP client does not know its IP address, the BOOTP server supplies it in the BOOTREPLY. Otherwise, the server returns the BOOTREPLY directly to ipaddr.
- server Specify the name of the BOOTP server to receive BOOTREQUEST. When the BOOTP server is known, the BOOTREQUEST is not broadcast.
- vendor Specify a vendor name to include vendor information in the BOOTREPLY. vendor can be specified as rfc1048 or cmu. For any other vendor specification, the first four characters of the parameter are used as the vendor magic cookie.
- f Specify that bootpd should broadcast the reply back. This option is only valid for bootpd on the HPUX 10.0 (or later) release(s).
- bootfile Specify a boot file needed by the BOOTP client. If a boot file is specified in the BOOTREQUEST, the BOOTP server responds only if the server host can make the file available.

EXAMPLES
/usr/sbin/bootpquery 02608cee018e ether -s hpserver
Received BOOTREPLY from hpserver.hp.com (15.9.18.119)
Hardware Address: 02:60:8c:ee:01:8e
Hardware Type: ethernet
IP Address: 15.9.18.113
Boot file: /export/tftpdir/hp-gw2-confg
RFC 1048 Vendor Information:
   Subnet Mask: 255.255.248.0
   Bootfile Size: 6512 byte blocks
   Domain Name Server: 15.9.18.119
   Host Name: hp-gw2

AUTHOR
bootpquery was developed by HP.
SEE ALSO

bootpd(1M), tftp(1), tftpd(1M).

DARPA Internet Request For Comments RFC951, RFC1048, RFC1084, RFC1395, RFC1542 Assigned Numbers.
NAME
cachefsstat - Cache File System statistics

SYNOPSIS
   cachefsstat [ -z ] [ path... ]

DESCRIPTION
The cachefsstat command displays statistical information about the cache filesystem mounted on path. The statistical information includes cache hits and misses, consistency checking, and modification operations. If path is not specified, all mounted cache filesystems are used. cachefsstat can also be used to reinitialize this information (see -z option). The statistical information has the following format:

   <cache hit rate>
   <consistency checks>
   <modifies>

where:

   cache hit rate
   The percentage of cache hits over the total number of attempts, followed by the actual numbers of hits and misses.

   consistency checks
   The number of consistency checks performed, followed by the number that passed, and the number that failed.

   modifies
   The number of modify operations, including writes, creates, etc.

Options
The following option is supported:

   -z     Zero (reinitialize) statistics. Execute cachefsstat -z before executing cachefsstat again to gather statistics on the cache performance. This option can only be used by the super-user. The statistics printed reflect those just before the statistics are re-initialized.

EXAMPLES
example% cachefsstat /home/sam
   cache hit rate: 73% (1234 hits, 450 misses)
   consistency checks: 700 (650 pass, 50 fail)
   modifies: 321

EXIT STATUS
The following exit values are returned:

   0     Success.
   non-zero     An error has occurred.

AUTHOR
   cachefsstat was developed by Sun Microsystems, Inc.

SEE ALSO
   cfsadmin(1M).
NAME
captoinfo - convert a termcap description into a terminfo description

SYNOPSIS
captoinfo [-1v] [-wn] [filenames]

DESCRIPTION
captoinfo looks in filenames for termcap(3X) descriptions. For each one found, an equivalent terminfo(4) description is written to standard output along with any comments found. The short two letter name at the beginning of the list of names in a termcap entry, a holdover from Version 6 UNIX, is removed. Any description that is expressed relative to another description (as specified in the termcap tc= field) is reduced to the minimum superset before output.

If no filename is given, the environment variable TERMCAP is used for the filename or entry. If TERMCAP is a full path name to a file, only the terminal whose name is specified in the environment variable TERM is extracted from that file. If the environment variable TERMCAP is not set, the file /usr/share/lib/termcap is read.

Options
captoinfo recognizes the following options:

-1 Print one field per line. If this option is not selected, multiple fields are printed on each line up to a maximum width of 60 characters.

-v Print (verbose) tracing information as the program runs. Additional -v options print more information (for example -v -v -v or -vvv).

-wn Change the output width to n characters.

DIAGNOSTICS
tgetent failed with return code n (reason).
The termcap entry is not valid. In particular, check for an invalid ‘tc=’ entry.

unknown type given for the termcap code ‘cc’.
The termcap description had an entry for ‘cc’ whose type was not boolean, numeric or string.

wrong type given for the boolean (numeric, string) termcap code ‘cc’.
The boolean termcap entry ‘cc’ was entered as a numeric or string capability.

the boolean (numeric, string) termcap code ‘cc’ is not a valid name.
An unknown termcap code was specified.

tgetent failed on TERM=term.
The terminal type specified could not be found in the termcap file.

TERM=term: cap cc (info ii) is NULL: REMOVED
The termcap code was specified as a null string. The correct way to cancel an entry is with an @, as in :bs@: . Giving a null string could cause incorrect assumptions to be made by any software that uses termcap or terminfo.

a function key for ‘cc’ was specified, but it already has the value ‘vv’.
When parsing the ‘ko’ capability, the key ‘cc’ was specified as having the same value as the capability ‘cc’, but the key ‘cc’ already had a value assigned to it.

the unknown termcap name ‘cc’ was specified in the ‘ko’ termcap capability.
A key that could not be handled was specified in the ‘ko’ capability.

the vi character ‘v’ (info ‘ii’) has the value ‘xx’, but ‘ma’ gives ‘n’.
The ‘ma’ capability specified a function key with a value different from that specified in another setting of the same key.

the unknown vi key ‘v’ was specified in the ‘ma’ termcap capability.
A vi key unknown to captoinfo was specified in the ‘ma’ capability.

Warning: termcap sg (nn) and termcap ug (nn) had different values.
terminfo assumes that the sg (now xmc) and ug values were the same.

Warning: the string produced for ‘ii’ may be inefficient.
The parameterized string being created should be rewritten by hand.
Null termname given.
The terminal type was null. This occurs when $TERM is null or not set.

cannot open "file" for reading.
The specified file could not be opened.

Warning: cannot translate capability (unsupported in terminfo).
This termcap capability is no longer supported in terminfo, and therefore cannot be translated.

WARNINGS
Certain termcap defaults are assumed to be true. For example, the bell character (terminfo bel) is
assumed to be ^G. The linefeed capability (termcap nl) is assumed to be the same for both
cursor_down and scroll_forward (terminfo cud1 and ind, respectively). Padding information is
assumed to belong at the end of the string.

The algorithm used to expand parameterized information for termcap fields such as
cursor_position (termcap cm, terminfo cup) sometimes produces a string which, though techni-
cally correct, may not be optimal. In particular, the rarely used termcap operation %n produces strings
that are especially long. Most occurrences of these less than optimal strings are flagged with a warning
message, and may need to be recoded by hand.

HP supports only terminals listed on the current list of supported devices. However, the terminfo database
contains both supported and nonsupported terminals. If you use nonsupported terminals, they may not
work correctly.

AUTHOR
 captoinfo was developed by AT&T.

SEE ALSO
tic(1M), untic(1M), termcap(3X), terminfo(4), infocmp(1M).
NAME
catman - create the cat files for the on-line manpages

SYNOPSIS
/usr/sbin/catman [-A alt-path] [-p] [-m] [-n] [-w] [-z] [sections]

DESCRIPTION
The catman command creates the formatted versions of the online manual pages from nroff-compatible source files. Each manpage in the man*.Z and man* directories is examined, and those whose formatted versions are missing or out of date are recreated. catman formats the most recent of the entries, compresses it, and puts it into the appropriate cat*.Z directory.

If any changes are made, catman recreates the /usr/share/lib/whatis database. By default, the /usr/share/lib/whatis database is overwritten. If the MANPATH environment variable is set to a non-default set of paths, the old database file is saved in /usr/share/lib/whatis.old so that, if desired, the system administrator may merge them together.

By default, catman searches the man*.Z and man* subdirectories under the following man directories:
- /usr/share/man
- /usr/contrib/man
- /usr/local/man

If MANPATH is set in the environment, the directories given in MANPATH are checked instead of the default. See environ(5) for a description of the MANPATH environment variable.

Before running catman, remove any existing cat* directories. If the -z option is used, cat*.Z directories should be removed instead. If both cat*.Z and cat* directories exist, man(1) updates both directories and more space is used.

Any command-line parameters not starting with - are interpreted as a list of manpage sections (directories) to search. For example:

catman 123

restricts updating to manpage sections 1, 2, and 3 (directories man1, man2, and man3).

Options

- m Create a merged /usr/share/lib/whatis database; i.e., information on new manpage entries (added since the last time catman was run) is merged into the current database rather than overwriting it. Ignored if selected with the -n option.
- n Prevents creation of /usr/share/lib/whatis.
- p Prints what would be done instead of doing it.
- w Causes only the /usr/share/lib/whatis database to be created. No manpage reformatting is done.
- z Puts the formatted entries in the cat* directories rather than in the cat*.Z directories.
- A alt-path Perform actions based on the given alternate root. With this option, alt-path will be prepended to all directory paths, including default paths, the paths defined by MANPATH, and the path to /usr/share/lib/whatis.

EXTERNAL INFLUENCES

Environment Variables

MANPATH defines parent directories to be used when searching man* and man*.Z directories.

WARNINGS

If unformatted manpages (those in the ../man* subdirectories) have been removed since the last time catman was run, information in the /usr/share/lib/whatis database may be lost. The -m option may be used to override this, but may result in repeated lines in the database for the same manpage.

EXAMPLES

Create uncompressed cat* files for sections 1 and 1m of the manual, but don't create the /usr/share/lib/whatis database:


```
catman -z -n 11m

Run `catman` from a server to create `cat*` entries for a diskless client under the alternate root

```
/export/shared_roots/OS_700:
```

```
catman -A /export/shared_roots/OS_700
```

This will create `cat*` manpages under:

```
/export/shared_roots/OS_700/usr/share/man/
/export/shared_roots/OS_700/usr/contrib/man/
/export/shared_roots/OS_700/usr/local/man/
```

and a `whatis` file in:

```
/export/shared_roots/OS_700/usr/share/lib/whatis
```

Create `cat*` entries for an application and merge the information with the

```
/export/shared_roots/OS_700/usr/share/lib/whatis
```

Create `cat*` entries for an application and merge the information with the

```
MANPATH=/opt/langtools/man
```

```
catman -m
```

Note that you may wish to save `MANPATH` before doing this, so as not to lose your current `MANPATH`.

AUTHOR

catman was developed by HP and the University of California, Berkeley.

FILES

```
/usr/share/man/man*[.Z]/*  Unformatted (nroff-compatible source) manpage files
/usr/share/man/cat*[.Z]/*  [compressed].
/usr/local/man/man*[.Z]/*  Formatted manpages [compressed].
/usr/local/man/cat*[.Z]/*  /
/usr/contrib/man/man*[.Z]/*  /usr/contrib/man/cat*[.Z]/*
/usr/share/lib/whatis       Database of manpage entry summaries; utilized by the `man -k`
```

```
/usr/sbin/mkwhatis          Command to make `whatis` database.
```

SEE ALSO

`compress(1), fixman(1M), man(1), environ(5).`
NAME
cfsadmin - administer disk space used for caching file systems with the Cache File-System (CacheFS)

SYNOPSIS
cfsadmin -c [ -o cacheFS-parameters ] cache_directory
cfsadmin -d [ cache_ID | all ] cache_directory
cfsadmin -l cache_directory
cfsadmin -s [ mntpt1... | all ]
cfsadmin -u [ -o cacheFS-parameters ] cache_directory

DESCRIPTION
The cfsadmin command provides the following functions:
- cache creation
- deletion of cached file systems
- listing of cache contents and statistics
- resource parameter adjustment when the file system is unmounted.

For each form of the command, with the exception of the -s form, you must specify a cache directory, that
is, the directory under which the cache is actually stored. A path name in the front file system identifies the
cache directory. For the -s form of the command, you must specify a mount point.

You can specify a cache ID when you mount a file system with CacheFS, or you can let the system generate
one for you. The -l option includes the cache ID in its listing of information. You must know the cache ID
to delete a cached file system.

Options
- c Create a cache under the directory specified by cache_directory. This directory must not exist
  prior to cache creation.
- d Remove the file system whose cache ID you specify and release its resources, or remove all file sys-
  tems in the cache by specifying all. After deleting a file system from the cache, you must run the
  fsck_cachefs(1M) command to correct the resource counts for the cache.
- l List file systems stored in the specified cache, as well as statistics about them. Each cached file sys-
  tem is listed by cache ID. The statistics document resource utilization and cache resource parameters.
- s Request a consistency check on the specified file system (or all cachefs mounted file systems). The
  -s option will only work if the cache file system was mounted with demandconst enabled (see
  mount_cachefs(1M)). Each file in the specified cache file system is checked for consistency with its
  corresponding file in the back file system. Note that the consistency check is performed file by file
  as files are accessed. If no files are accessed, no checks are performed. Use of this option will not
  result in a sudden "storm" of consistency checks.
- u Update resource parameters of the specified cache directory. Parameter values can only be
  increased. To decrease the values, you must remove the cache and recreate it. All file systems in
  the cache directory must be unmounted when you use this option. Changes will take effect the next
  time you mount any file system in the specified cache directory. The -u option with no -o option
  sets all parameters to their default values.

Operands
  cache_directory The directory under which the cache is actually stored.
  mntpt1 The directory under which the CacheFS is mounted.

CacheFS Resource Parameters
You can specify the following cacheFS resource parameters as arguments to the -o option. Separate multi-
ple parameters with commas.
  maxblocks=n Maximum amount of storage space that CacheFS can use, expressed as a percentage of
  the total number of blocks in the front file system. If CacheFS does not have exclusive
  use of the front file system, there is no guarantee that all the space the maxblocks
  parameter allows will be available. The default is 90.
**minblocks=\*n** Minimum amount of storage space, expressed as a percentage of the total number of blocks in the front file system, that CacheFS is always allowed to use without limitation by its internal control mechanisms. If CacheFS does not have exclusive use of the front file system, there is no guarantee that all the space the `minblocks` parameter attempts to reserve will be available. The default is 0.

**threshblocks=\*n** A percentage of the total blocks in the front file system beyond which CacheFS cannot claim resources once its block usage has reached the level specified by `minblocks`. The default is 85.

**maxfiles=\*n** Maximum number of files that CacheFS can use, expressed as a percentage of the total number of inodes in the front file system. CacheFS may automatically increase this parameter when being used on a VxFS file system. If CacheFS does not have exclusive use of the front file system, there is no guarantee that all the inodes the `maxfiles` parameter allows will be available. The default is 90.

**minfiles=\*n** Minimum number of files, expressed as a percentage of the total number of inodes in the front file system, that CacheFS is always allowed to use without limitation by its internal control mechanisms. If CacheFS does not have exclusive use of the front file system, there is no guarantee that all the inodes the `minfiles` parameter attempts to reserve will be available. The default is 0.

**threshfiles=\*n** A percentage of the total inodes in the front file system beyond which CacheFS cannot claim inodes once its usage has reached the level specified by `minfiles`. The default is 85.

**maxfilesize=\*n** Largest file size, expressed in megabytes, that CacheFS is allowed to cache. The default is 3.

Currently, `maxfilesize` is ignored by CacheFS; therefore, setting it will have no effect.

Note: You cannot decrease the block or inode allotment for a cache. To decrease the size of a cache, you must remove it and create it again with different parameters.

**EXAMPLES**

The following example creates a cache directory named `/cache`:

```
  cfsadmin -c /cache
```

The following example creates a cache named `/cache1` that can claim a maximum of 60 percent of the blocks in the front file system, can use 40 percent of the front file system blocks without interference by CacheFS internal control mechanisms, and has a threshold value of 50 percent. The threshold value indicates that after CacheFS reaches its guaranteed minimum, it cannot claim more space if 50 percent of the blocks in the front file system are already used.

```
  cfsadmin -c -o maxblocks=60,minblocks=40,threshblocks=50 /cache1
```

The following example changes the `maxfilesize` parameter for the cache directory `/cache2` to 2 megabytes:

```
  cfsadmin -u -o maxfilesize=2 /cache2
```

The following example lists the contents of a cache directory named `/cache3` and provides statistics about resource utilization:

```
  cfsadmin -l /cache3
```

The following example removes the cached file system with cache ID 23 from the cache directory `/cache3` and frees its resources (the cache ID is part of the information returned by `cfsadmin -l`):

```
  cfsadmin -d 23 /cache3
```

The following example removes all cached file systems from the cache directory `/cache3`:

```
  cfsadmin -d all /cache3
```

The following example checks for consistency all filesystems mounted with `demandconst` enabled. No errors will be reported if no `demandconst` filesystems were found.
cfsadmin(1M)

  cfsadmin -s all

AUTHOR
  cfsadmin was developed by Sun Microsystems, Inc.

SEE ALSO
  fsck_cachefs(1M), mount_cachefs(1M).
NAME
ch_rc - change system configuration file

SYNOPSIS
/usr/sbin/ch_rc -a | -r | -l [ -v ] [ -A ] [ -R root ] [ -p parameter | parameter=value ] ... [ file ... ]

DESCRIPTION
ch_rc manages the addition, modification, removal, and retrieval of information stored in files having the format of those in the /etc/rc.config.d directory.

Parameter names are treated as strings. Thus, X[0] has no special meaning to ch_rc in relation to other parameters named X[1] or X.

Options
- file Specify the file(s) to be used as the configuration database. If no file is specified, the set of files used by ch_rc defaults to /etc/TIMEZONE and all files in the /etc/rc.config.d directory.

Modification and deletion of configuration parameters occurs in the file where the parameter is found.
- -a Add or modify a parameter definition. For each parameter specified on the command line, if the parameter is found in the specified (or default) files, it is modified to reflect the specified value. If the parameter is not found, it is added to the specified file(s).

If a new parameter is being defined, one or more files must be specified on the command line; the specified files are those in which the parameter will be defined.
- -r Remove a parameter definition. For each parameter name specified on the command line, remove any occurrence of that parameter from the specified file(s).
- -l List configuration values. For each parameter specified on the command line, output every definition of the parameter from the specified file(s). Output consists of only the values, one per line.
- -p Specify a parameter name or name/value pair. If a name and value is expected, but only a name is specified, the value will be set to the empty string. For example, specifying FOO or FOO= will result in FOO and FOO= respectively.

Due to shell quoting rules, if you need a quoted parameter value, you must protect the quotes from the shell. For example,

```
ch_rc -a -p VALUE="a b c" <file>
```

yields:

```
VALUE=a b c
```

which is an error, whereas,

```
ch_rc -a -p VALUE='"a b c"' <file>
```

yields:

```
VALUE="a b c"
```

- -v Verbose. When used with the -l option, the -v option causes a verbose listing to be output. This listing includes a filename followed by the entire line containing the specified parameter for each occurrence of the parameter.

- -A The -A option is used to list all occurrences of array parameters matching the parameters specified on the command line.

For example,

```
ch_rc -l -A -v -p ZZZ file
```

may emit the following output:

```
file: ZZZ[0]=zero
file: ZZZ[5]=five
file: ZZZ[9]=fred
```
-R root  Normally, the files specified on the command line are used as specified. By specifying a root directory with the -R option, all files (including the default files if none are specified) will be interpreted relative to root.

For example, if root is specified as /foo and /etc/TIMEZONE is specified on the command line, it will be interpreted as /foo/etc/TIMEZONE.

RETURN VALUE

ch_rc exits with one of the following values:

0  add/delete/list successful
1  command line syntax/usage error
2  can not access one or more of the listed (or default) files
3  can not open/create/write file
4  memory error
5  no files specified on command line for add option

EXAMPLES

Files in the /etc/rc.config.d directory have the following format:

# Comments are preceded by pound signs and
# are always on a line of their own.
# Blank lines are allowed.
VARIABLE=value
VARIABLE_2=value2
VARIABLE_3[1]=value3
VARIABLE_3[2]=value4
# All parameters are defined on a single line
# Parameters must not be exported

WARNINGS

ch_rc does not interpret configuration files; it only does pattern matching. As a result, if comments appear on lines containing parameter definitions, the comments will also appear in output when using the -l option.

ch_rc cannot parse multiple parameter definitions which occur on the same line of a file. Also, only certain formats are supported and whole lines are replaced.

Parameters must not be exported. If the parameters are exported, only the parameter value (name=value) pair is stored and the rest of the data like an export special command in front of a name value pair, whitespace, and comments are deleted from the line.

AUTHOR

ch_rc was developed by HP.

FILES

/etc/rc.config  system configuration database driver file
/etc/rc.config.d  directory containing system configuration files

SEE ALSO

rc.config(4).
check_patches(1M)

NAME
check_patches - HP-UX 11i patch check utility

SYNOPSIS
check_patches [-imopsv]

DESCRIPTION
The check_patches utility checks for installation problems and issues related to patches on HP-UX 11i. The utility checks for patches missing the SD-UX patch attributes, missing patch filesets, patch object modules missing from archive libraries, patch filesets with the incorrect patch_state, patch filesets not in the configured state, and patch filesets that fail swverify (see swverify(1M)).

The check_patches utility logs all information to /tmp/check_patches.report.

Options
If no options are specified, check_patches performs all checks. The check_patches utility recognizes the following options:

- `i` Check for invalid patches (patches missing the SD-UX patch attributes)
- `m` Check for missing patch filesets in the currently active patches. This will determine if a product fileset was installed, or reinstalled, after a currently active patch.
- `o` Verify the checksum values of the object modules delivered by the currently active patches
- `p` Check all patch filesets for correct SD-UX patch_state (applied, committed, committed/superseded, superseded)
- `s` Check the SD-UX state to identify all unconfigured patch filesets
- `v` Run the swverify command on all patch filesets

AVAILABILITY
This command is standard on HP-UX 11i v3. On 11i v2 it is available in patch PHCO_32220 or a superseding patch. On 11i v1 it is available in patch PHCO_27780 or a superseding patch.

RETURN VALUE
Upon completion, check_patches returns one of the following values:

0 No problems were found.
1 One or more patch-related problems were logged.
2 Syntax or usage error.

AUTHOR
check_patches was developed by HP.

SEE ALSO
show_patches(1), sysdiff(1), cleanup(1M), swlist(1M), swremove(1M), swverify(1M).
NAME
chnlspath - configure message catalog path

SYNOPSIS
chnlspath [-adlsc] [pseudo-pathname]

DESCRIPTION
chnlspath is used to modify the contents of the configuration file /etc/default/nlspath.
chnlspath recognizes the following options and command-line arguments:

Options
-1 List the contents of the configuration file /etc/default/nlspath.
-c Switch to compatible mode.
-s Switch to secure mode.
-a Add an entry to the configuration file /etc/default/nlspath.
-d Delete a pseudo-pathname from the configuration file /etc/default/nlspath. If multiple
instances of the same pseudo-pathname exists in /etc/default/nlspath, only the first instance
will be removed.

RETURN VALUE
chnlspath returns the following values:
  0 Operation successful.
 >0 Error condition occurred.

SEE ALSO
catopen(3C), nlspath(4), environ(5).
NAME
chroot - change root directory for a command

SYNOPSIS
/usr/sbin/chroot newroot command

DESCRIPTION
The chroot command executes command relative to the newroot. The meaning of any initial slashes (/) in path names is changed for command and any of its children to newroot. Furthermore, the initial working directory is newroot.

Note that command suffixes that affect input or output for the chroot command use the original root, not the new root. For example, the command:

    chroot newroot command > x

locates file x relative to the original root, not the new one.

The command variable includes both the command name and any arguments.

The new root path name is always relative to the current root. Even if a chroot is currently in effect, the newroot argument is relative to the current root of the running process.

This command is restricted to users with appropriate privileges.

EXTERNAL INFLUENCES
International Code Set Support
Single- and multibyte character code sets are supported.

WARNINGS
command cannot be in a shell script.

Exercise extreme caution when referring to special files in the new root file system.

chroot does not search the PATH environment variable for the location of command, so the absolute path name of command must be given.

When using chroot to establish a new environment, all absolute path name references to the file system are lost, rendering shared libraries inaccessible. If continued access to shared libraries is needed for correct operation, the shared libraries and the dynamic loader must be copied into the new root environment.

SEE ALSO
chdir(2), chroot(2).

STANDARDS CONFORMANCE
chroot: SVID2, SVID3, XPG2, XPG3
NAME
cleanup - HP-UX patch cleanup utility

SYNOPSIS
cleanup [-n|-p] [-c number]
cleanup [-n|-p] -d depot [-o {10|11}] [-e {true|false}]
cleanup [-n|-p] -i
cleanup [-n|-p] -s

DESCRIPTION
The cleanup command provides functions useful when dealing with HP-UX patches on HP-UX.
The cleanup command logs all information to /var/adm/cleanup.log.

Options
cleanup recognizes the following options:
- p
  Preview the cleanup task but do not actually perform the requested action.
- n
  Notify the user of cleanup tasks and request confirmation before performing the requested action.
- c number
  Commit patches superseded at least number times. This option is a front-end to the patch_commit option of swmodify (see swmodify(1M)) that provides an easier interface to commit multiple patches at once.
- d
  Remove patches in the specified software depot that have been superseded by patches also available from the depot.
- o {10|11}
  Specifies whether the software depot contains patches for HP-UX 10.X or 11.X releases. This option is only valid with the -d option. If the -o option is not specified, it defaults to HP-UX 11.X patches.
- e {true|false}
  Specifies whether patches in the software depot that contain a corequisite or prerequisite attribute specifying one of the superseded patches as a dependency should also be removed. This option is valid only with the -d option. If the -e option is not specified, it defaults to true.
- i
  Remove HP-UX 10.X patches that remain in the Installed Product Database after an upgrade to HP-UX 11.X. These patches are removed from the IPD so that they are no longer displayed in the output of the swlist command (see swlist(1M)). The HP-UX 10.X patch files are also removed from /var/adm/sw/patch.
- s
  Correct the patch_state attribute for HP-UX 11.X patches.

Recommendations
The cleanup -i command should be executed after updating to HP-UX 11.X from HP-UX 10.20. It is not necessary to execute cleanup -i after updating from one version of HP-UX 11.X software to another because SD-UX will properly remove 11.X patch information from the IPD in these situations.
The cleanup -d command should be executed whenever patches are added to a software depot. It is not a requirement that superseded 11.X patches be removed from a software depot, but it is a recommendation to conserve disk space and to avoid confusion. If superseded patches exist in the same depot as patches that supersede them, SD-UX on HP-UX 11.X will handle the situation properly.
The cleanup -c command can be executed to commit all patches that have been superseded at least the specified number of times. The only benefit from committing patches is that disk space will be recovered in the /var/adm/sw/save directory.
The cleanup -s command can be executed to insure that the patch_state attribute of all patch filesets is set correctly. There are several known situations where the patch_state attribute will be set incorrectly. This command can be run at any time, but will be most useful after a system has been installed using Ignite-UX.

AVAILABILITY
This command is standard on HP-UX 11i v3. On 11i v2 it is available in patch PHCO_32220 or a superseding patch. On 11i v1 it is available in patch PHCO_27780 or a superseding patch.
WARNINGS

Extreme care should be used when committing patches. Once a patch has been committed, it cannot be
removed from the system. Patches should only be committed when you are satisfied with their behavior.
Recovering disk space should not be the only factor considered when committing patches.

AUTHOR

cleanup was developed by HP.

SEE ALSO

show_patches(1), sysdiff(1), check_patches(1M), swlist(1M), swmodify(1M), swremove(1M), sd(4).
clear_locks(1M)

NAME
clear_locks - clear locks held on behalf of an NFS client

SYNOPSIS
/usr/sbin/clear_locks [-s] hostname

DESCRIPTION
The clear_locks command removes all file, record, and share locks created by the hostname and held on the current host, regardless of which process created or owns the locks.

This command can be run only by the super-user.

This command should only be used to repair the rare case of a client crashing and failing to clear held locks. Clearing locks held by an active client may cause applications to fail in an unexpected manner.

Options
-s hostname Remove all locks created by the current machine and held by the server, hostname. hostname is the name of host server.

EXIT STATUS
clear_locks returns:
0 Successful operation.
1 If not root.
2 Usage error.
3 If unable to contact server (RPC).

EXAMPLES
Remove all locks on local NFS server for client, client1:
clear_locks client1

Remove all locks created by the local client system and held by the remote NFS server, server1:
clear_locks -s server1

SEE ALSO
lockd(1M), statd(1M).
NAME
cetri - clear inode

SYNOPSIS
/usr/sbin/clri special i-number ...

DESCRIPTION
The cetri command clears the inode i-number by filling it with zeros. special must be a special file name referring to a device containing a file system. For proper results, special should not be mounted (see WARNINGS below). After cetri is executed, all blocks in the affected file show up as "missing" in an fsck of special (see fsck(1M)). This command should only be used in emergencies.

Read and write permission is required on the specified special device. The inode becomes allocatable.

WARNINGS
The primary purpose of this command is to remove a file that for some reason does not appear in any directory. If it is used to clear an inode that does appear in a directory, care should be taken to locate the entry and remove it. Otherwise, when the inode is reallocated to some new file, the old entry in the directory will still point to that file. At that point, removing the old entry destroys the new file, causing the new entry to point to an unallocated inode, so the whole cycle is likely to be repeated again.

If the file system is mounted, cetri is likely to be ineffective.

DEPENDENCIES
cetri operates only on file systems of type hfs.

SEE ALSO
fsck(1M), fsdb(1M), ncheck(1M).

STANDARDS CONFORMANCE
cetri: SVID2, SVID3
NAME
crlsvc - clear x25 switched virtual circuit

SYNOPSIS
crlsvc line pad-type

DESCRIPTION
The uucp commands, including crlsvc, are targeted for removal from HP-UX; see the WARNINGS
below.
crlsvc clears any virtual circuit that might be established on the specified line. pad-type indicates to
crlsvc what opx25 script to run from /usr/lbin/uucp/X25.

DEPENDENCIES
HP 2334A is the only PAD supported at this time, and results in an opx25 execution of HP2334A.clr.

EXAMPLES
A typical invocation is:
/usr/lbin/uucp/X25/crlsvc /dev/x25.1 HP2334A

WARNINGS
Use of uucp commands, including crlsvc, is discouraged because they are targeted for removal from
HP-UX. Use ftp(1) or rcp(1) instead.

AUTHOR
crlsvc was developed by HP.

SEE ALSO
login(1), uucp(1), getty(1M), getx25(1M), opx25(1M).
cmdprivadm(1M)
cmdprivadm(1M)

NAME
cmdprivadm - noninteractive editing of a command's authorization and privilege information in the
privrun database

SYNOPSIS
  cmdprivadm add option=value [option=value]...
  cmdprivadm delete option=value [option=value]...

DESCRIPTION
cmdprivadm is a noninteractive command that allows user with appropriate permission to add or delete
a command and its privileges in the Role-Base Access Control (RBAC) database,
/etc/rbac/cmd_priv. See privrun(1M) for more details on this file.

When adding a line to the database, cmdprivadm sets fields that are not specified a default value. When
deleting a line, the lines matching all the given option=value pairs will be deleted. That is, if all fields
specified match, the entry will be deleted.

  cmdprivadm add option=value [option=value]...
    Appends a line as specified in option=value pairs in the /etc/rbac/cmd_priv file.
  cmdprivadm delete option=value [option=value]...
    Deletes a line as specified in option=value pairs from /etc/rbac/cmd_priv file.

HP recommends that only the authadm, cmdprivadm, and roleadm commands be used to edit and
view the RBAC databases; do not edit the RBAC files directly.

See rbac(5) for information on the RBAC databases.

Options
The following options are valid option=value pairs for cmdprivadm.

  cmd=command    command should include the full path name of the command. There can be one or
                  more arguments following the command.
  file=filename  filename should specify the full path name of a file name.
  op=operation   Specifies the operation.
  object=object  Specifies the object.
  ruid=ruid      Specifies the real user ID (ruid).
  euid=euid      Specifies the effective user ID (euid).
  rgid=rgid      Specifies the real group ID (rgid).
  egid=egid      Specifies the effective group ID (egid).
  compartment=compartment_label
                  Specifies the compartment.
  privs=comma_separated_privilege_list
                  Specifies the privileges.
  re-auth=pam_service
                  Specifies the PAM service name to reauthenticate under. See pam.conf(4) for a list of
                  PAM services.
  flags=comma_separated_flags_list
                  Specifies the flags.

Note: You must enclose values that contain the space character, or any characters that may be interpreted
by the shell, with single quotes. For example, if the cmd has one or more arguments, enclose them with
single quotes:

    cmd='mount -a'

Authorizations:
In order to invoke cmdprivadm, the user must either be root, (running with effective UID of 0), or have
the appropriate authorizations. The following is a list of the required authorizations for running
cmdprivadm(1M) with particular options:

- **hpux.security.access.privrun.add,***
  - Allows user to run `cmdprivadm` with **add** options.
- **hpux.security.access.privrun.delete,***
  - Allows user to run `cmdprivadm` with **delete** options.

### EXTERNAL INFLUENCES

#### Environment Variables

- **LC_MESSAGES** determines the language in which messages are displayed.

#### International Code Set Support

Single-byte character code set is supported.

### RETURN VALUE

Upon completion, `cmdprivadm` returns one of the following values:

- **0** Success.
- **1** Failure. An appropriate error message is printed on standard error.

### EXAMPLES

The following commands add entries into the `/etc/rbac/cmd_priv` file:

```bash
# cmdprivadm add cmd='/sbin/mount -a' op=hpux.adm.mount \
> object='*' ruid=0 euid=0 rgid=0 egid=0 compartment=testcomp \
> privs=BASICROOT,CHANGECMPT re-auth=passwd
# cmdprivadm add cmd=/sbin/mount op=hpux.printer.add object='*' 
```

The following commands delete entries from the `/etc/rbac/cmd_priv` file:

```bash
# cmdprivadm delete cmd='/sbin/mount -a'
# cmdprivadm delete cmd=/sbin/mount 
```

### FILES

- **/etc/rbac/roles** Database containing valid definitions of all roles.
- **/etc/rbac/auths** Database containing definitions of all valid authorizations.
- **/etc/rbac/user_role** Database specifying the roles allowed for each specified user.
- **/etc/rbac/role_auth** Database defining the authorizations for each specified role.
- **/etc/rbac/cmd_priv** Database containing the authorization to execute specified commands, and the privileges to alter UID and GID for command execution.

### SEE ALSO

- `authadm(1M)`, `privrun(1M)`, `rbacdbchk(1M)`, `roleadm(1M)`, `rbac(5)`.
NAME
cmpt_tune - query, enable, or disable compartmentalization feature

SYNOPSIS
   cmpt_tune  
    cmpt_tune [-q] [-s]
   cmpt_tune [-Q] [-s] [-n boot_image]
   cmpt_tune [-Q] [-s] -n boot_image
   cmpt_tune [-d] [-e] [-r] [-s] [-n boot_image]

DESCRIPTION
   cmpt_tune queries, enables, or disables the compartmentalization feature. Compartmentalization is not
   a dynamic feature; enabling or disabling the feature requires a reboot. If you make a change and do not
   specify the -r flag, cmpt_tune reports a reboot reminder message. If no options are specified, the -q
   option is assumed.

   If no compartments have been defined when compartmentalization is enabled, the network interfaces
   currently installed on the system are assigned to a new compartment ifaces, and the administrator is
   given the opportunity to reassign these interfaces (see getrules(1M)).

   The system initially boots into a predefined compartment, INIT. A process in the INIT compartment can
   access all objects (that is, all processes, files, IPC objects, etc., are accessible from the INIT compartment).
   See compartments(5) for more information. Using the setfilexsec command (see setfilexsec(1M)), an
   administrator can set specific binaries to start automatically in other compartments; that is, when a process
   executes the binary, it may find its compartment modified as a side-effect. This concept is similar to a
   setuid binary changing a process's euid.

   When the -e or -d option is specified without the -n option, the current running configuration is modified.
   If -e or -d is specified with the -n option and boot_image does not exist, it is created as though the
   administrator ran the following command:

     kconfig -s boot_image

   In any case, boot_image is marked for use on the next boot.

Options
   The cmpt_tune command recognizes the following options:

   -d  Disables compartments.
   -e  Enables compartments.
   -h  Prints a help message.
   -n boot_image  Makes changes to or queries the specified boot_image. If this option is not specified,
                  boot_image defaults to nextboot. If no other options are specified, the -Q option is
                  assumed.
   -q  Queries the current state of compartments.
   -Q  Queries the state of compartments after the next reboot.
   -r  Reboots after making changes. You can only use this option with the -d or -e options.
   -s  Sets silent mode. Only the exit status is set.

RETURN VALUE
   cmpt_tune returns the following values:

   0  When querying, the compartmentalization feature is enabled. When making changes, the
      changes are successfully applied.
   1  An option processing error occurred. When querying, the compartmentalization feature is dis-
      abled. When making changes, and -r is specified, the reboot option is ignored (for example, to
      allow for editing of compartment configuration files).
When querying, the kernel configuration specified does not exist or has no support for compartmentalization.

WARNINGS
A network interface that is not assigned to any compartment cannot be accessed by any process and effectively cannot be used. Assign at least one network interface to a compartment so that network communications can function.

If the \texttt{-e} or \texttt{-d} option is used in conjunction with the \texttt{-n} option, any prior changes pending to the current configuration are lost.

If the compartments feature is enabled on a kernel configuration that does not reflect the required patch levels (for example, patch PHKL_32798 is missing), the system may not boot properly or may not have network connectivity.

SEE ALSO
authadm(1M), kconfig(1M), getrules(1M), setfilexsec(1M), setrules(1M), compartments(4), compartments(5).
NAME
correct_awk - converts old sendmail.cf files to new format

SYNOPSIS
/usr/newconfig/etc/mail/convert_awk

DESCRIPTION
correct_awk is an awk program that will convert pre-HP-UX 10.20 sendmail.cf files into the format required by sendmail 8.7 and up.
To run it, use:
    awk -f correct_awk < old.cf > new.cf
Note that the new sendmail.cf files offer a wealth of new options and features. You should STRONGLY consider making a new sendmail.cf file from the distribution version or from the m4 macros, which are provided in /usr/newconfig/etc/mail/cf.

SEE ALSO
sendmail(1M).
convertfs(1M)

NAME
convertfs - convert an HFS file system to allow long file names

SYNOPSIS
/usr/sbin/convertfs [-q] [special-file]

DESCRIPTION
The `convertfs` command converts an existing HFS file system supporting the default maximum file name length of 14 characters into one that supports file names up to 255 characters long. Once an HFS file system is converted to long file names, it cannot be restored to its original state, since the longer file names require a directory representation that is incompatible with the default HFS directory format. Since this is an irreversible operation, `convertfs` prompts for verification before it performs a conversion.

`convertfs` forces the system to reboot if the root file system is converted. When converting the root file system, the system should be in single-user mode, with all unnecessary processes terminated and all non-root file systems unmounted. Except for the root file system, `convertfs` requires that the file system to be converted be unmounted.

If invoked without arguments, `convertfs` interactively prompts the user with a list of the HFS file systems from `/etc/fstab`. One or more or all of the listed file systems can be selected for conversion. Typically, it is desirable to convert all of the file systems in `/etc/fstab` to avoid inconsistencies between two file systems mounted on the same system.

`convertfs` can also be invoked with an argument of either a block or character special-file of a file system to be converted. Only the block special file should be specified for a mounted root file system.

As part of the conversion process, `convertfs` performs an `fsck` on each file system (see `fsck(1M)`).

Options
- `-q` Do quietly. `convertfs` will perform the conversions without querying the user. Normally `convertfs` prompts the user before converting a file system.

RETURN VALUE
`convertfs` returns the following values:
- `0` Success. Either `convertfs` successfully converted the file system, or the file system already allowed long file names.
- `non-0` Failure. `convertfs` was not able to convert the file system due to some failure in processing.

AUTHOR
`convertfs` was developed by HP.

FILES
/etc/fstab Default list of file systems to check.

SEE ALSO
fsck(1M), mkfs(1M), newfs(1M), fstab(4).
NAME
cplxmodify - modify an attribute of a system complex

SYNOPSIS
cplxmodify [-N ComplexName]
    [-u username: [passwd] -h IPaddress|hostname ]
    | [ -g [passwd] -h IPaddress|hostname ]

DESCRIPTION
The cplxmodify command modifies attributes of the complex of a partitionable system. The command is
two steps, the equivalent, at the complex level, of the parmodify command, which modifies attributes of an nPartition.

The complex to be modified defaults to the local complex, the one on which the command is run. A remote
complex can be identified by specifying either the -u or -g option in combination with the -h option.

Root permission is required to run this command on the local partition. If the -u or -g option is used to
access a remote partition or complex, root permission is not required on the local system, and the local system
need not exist on an nPartition. If the -u option is specified, username on the remote host must have
root permission or the command will fail.

By the nature of its operation, this command can modify the configuration of its target complex. Operation
can be affected by the state of the target complex’s nPartition Configuration Privilege. If this privilege is
unrestricted (the default), or the complex is accessed using the -g option, all operations are allowed. Oth-
erwise the command fails. The -g option is unaffected by the state of the nPartition Configuration
Privilege. Note: This privilege state can only be changed at the service processor’s Command menu.

Refer to the HP System Partitions Guide for a description of the partition management terms used in this
man page. Refer to partition(5) for a list of partition management commands.

Options and Arguments
The cplxmodify command recognizes the following command line options and arguments:

- N ComplexName
Changes the name of the target complex to ComplexName. This option may also be used to
name a complex for the first time.

    The characters which can appear in a valid complex name are a-zA-Z, 0-9, - (dash), _
(underscore), " " (space) and . (period). If the complex name includes space, then the
name should be enclosed within double quotes. The complex name can have a maximum of
20 characters.

- u username: [passwd]
Specifies the required authorization to access a partition other than the local system (but
can also be used as a loopback access to the local partition). The complex to be modified is
the one in which this target partition resides.

    The -h option is required if this option is used.

    username specifies a configured user name on the target partition.

    passwd specifies the password associated with the username. If this field is empty, the
command prompts for the password.

    Note: This command is a Web-Based Enterprise Management (WBEM) Client Application.
The -u option accesses the target partition using a Secure Sockets Layer (SSL) connection.
If errors are reported, check that the conditions described in the DEPENDENCIES section
are satisfied.

    SECURITY WARNING: Specifying the password directly on the command line may pose a
security risk in your environment. The ps or other related commands can be invoked in
such a way as to display the command line of a process. In this situation, any authenti-
cated user on the system can potentially see the password while the process is executing.
Hence, it is highly recommended not to specify the password on the command line, and
instead allow the command to prompt for the password.

    Note: Specifying passwd on the command line will not be supported in future releases.

- h IPaddress | hostname
This option should only be used in combination with either the -u or -g option.
cplxmodify(1M)

IPaddress | hostname specifies the IP address or hostname of the target partition (-u) or complex (-g).

- **g** [passwd]

Allows access to the complex specified by the -h option. The accessed complex is then considered the target complex. Access is via the service processor’s LAN port.

The -h option is required if this option is used.

passwd specifies the IPMI password of the service processor. If this field is omitted, the command prompts for the password.

If an error is reported when you attempt to connect using this option, check to see that IPMI LAN access has not been disabled on the remote service processor. Access to the complex via IPMI over LAN can be enabled or disabled by logging on to the service processor and using the SA command from the Command Menu.

The -u and -g options are mutually exclusive.

SECURITY WARNING: Specifying the password directly on the command line may pose a security risk in your environment. The **ps** or other related commands can be invoked in such a way as to display the command line of a process. In this situation, any authenticated user on the system can potentially see the password while the process is executing. Hence, it is highly recommended not to specify the password on the command line, and instead allow the command to prompt for the password.

**Note:** Specifying **passwd** on the command line will not be supported in future releases.

**RETURN VALUE**

The cplxmodify command exits with one of the following values:

0 Successful completion.

1 Error condition occurred.

**EXAMPLES**

Change the name of the local complex to "Basingstoke".

```
cplxmodify -N Basingstoke
```

Change the name of a remote complex to "Ploverleigh". The command prompts for the password.

```
cplxmodify -N Ploverleigh -g -h RemoteComplex
```

**WARNINGS**

HP-UX 11i Version 2 is the last HP-UX release on which passing passwd for the -u and -g options is supported. Specifying passwd on the command line will not be supported in future releases. The command will prompt for the password.

**DEPENDENCIES**

This command uses the Web-Based Enterprise Management (WBEM) product and certain of its configuration settings. If you encounter connection errors when using the -u option, check that the following two conditions are satisfied:

- Use the **cimconfig** command (see **cimconfig(1M)** in the WBEM product documentation) to verify (and correct if necessary) the setting of the following two variables:
  - `enableRemotePrivilegedUserAccess=true`
  - `enableHttpsConnection=true`

- You must have appended the target partition's digital certificate to the local partition's Trust Store file.
  For the npartition commands, the Trust Store file is `/var/opt/wbem/client.pem`.
  **Note:** You must have appended the target partition's digital certificate to the local partition's Trust Store file. For the npartition commands, the Trust Store file is `/var/opt/wbem/client.pem`. This file is used by the commands that come with WBEM installation. Hence, if the commands that come with the WBEM installation trust a target partition, then npartition commands will also trust the target partition.

Refer to the WBEM documents specified in the SEE ALSO section below for further information.
AUTHOR

cplxmodify was developed by the Hewlett-Packard Company.

SEE ALSO

fruled(1), parstatus(1), frupower(1M), parcreate(1M), parmgr(1M), parmodify(1M), parremove(1M), parunlock(1M), partition(5).

HP System Partitions Guide on http://docs.hp.com,

HP WBEM Services for HP-UX System Administrator's Guide on http://docs.hp.com,

HP WBEM Services for HP-UX 11i v2.0 on Integrity Servers Version A.01.05 Release Notes on http://docs.hp.com.
NAME
cpset - install object files in binary directories

SYNOPSIS
cpset [-o] object directory [-mode [-owner [-group]]]

DESCRIPTION
The cpset command installs the specified object file in the given directory. The mode, owner, and group, of the destination file can be specified on the command line. If this data is omitted, two results are possible:

- If you have administrative permissions (that is, your numerical ID is less than 100), the following defaults are provided:
  
  mode 0555
  owner bin
  group bin

- If you do not have administrative permissions, the default mode, owner, and group of the destination file are the same as yours.

The -o option forces cpset to move object to OLDobject in the destination directory before installing the new object.

cpset reads the /etc/src/destinations file to determine the final destination of the file to be installed. The destinations file contains pairs of path names separated by spaces or tabs. The first name is the "official" destination (for example: /usr/bin/echo). The second name is the new destination. If echo is moved from /usr/bin to /usr/local/bin, the entry in destinations would be:

/usr/bin/echo /usr/local/bin/echo

When the actual installation happens, cpset verifies that the "old" pathname does not exist. If a file exists at that location, cpset issues a warning and continues.

This file does not exist on a distribution tape; it is used by sites to track local command movement. The procedures used to build the source are responsible for defining the "official" locations of the source.

Cross Generation
The environment variable ROOT is used to locate the destination file (in the form $ROOT/etc/src/destinations). This is necessary in the cases where cross generation is being done on a production system.

EXAMPLES
If you are an administrator, all of the following examples have the same effect. They copy file echo into /usr/bin with mode, owner, and group set to 0555, bin, bin, respectively:

    cpset echo /usr/bin 0555 bin bin
    cpset echo /usr/bin
    cpset echo /usr/bin/echo

If you are not an administrator, the last two examples set mode, owner, and group to your current values.

SEE ALSO
chacl(1), make(1), install(1M), acl(5).
NAME

crashconf - configure system crash dumps

SYNOPSIS

/sbin/crashconf [-artv] [-i | -e class] [-c mode] ... [device...]

DESCRIPTION

crashconf displays and/or changes the current system crash dump configuration. The crash dump configuration consists of:

- The crash dump device list. This list identifies all devices that can be used to store a crash dump.
- The included class list. This list identifies all system memory classes that must be included in any crash dump.
- The excluded class list. This list identifies all system memory classes that should not be included in a crash dump.
- The compression mode selection. This selection is used to turn compression ON or OFF, before dumping.

Most system memory classes are in neither the included class list nor the excluded class list. Instead, the system determines whether or not to dump those classes of memory based on the type of crash that occurs.

Note the system operator may request a full crash dump at the time the dump is taken. In this case, a full dump will be performed regardless of the contents of the excluded class list.

Turning compression mode ON will result in faster and smaller dumps. If the dump is compressed, savecrash will also copy over the dump faster since the dump will be smaller in size.

Since compressed dump requires additional processors and memory to do the compression, the system may fall back on uncompressed dump if it could not identify the processing resources required to do compressed dump.

Any changes to the configuration take effect immediately and remain in effect until the next system reboot, or until changed with a subsequent invocation of crashconf. Using the option -t, changes to the include and exclude class lists and compression mode can be made persistent across system reboots. But the changes do not persist across kernel rebuilds. Use SAM or kctune(1M) to do this.

device specifies a block device file name of a device that is a valid destination for crash dumps. All such devices listed on the command line will be added to the end of the current list of crash dump devices, or will replace the current list of crash dump devices, depending on whether -r is specified.

class is the name (or number) of a system memory class which should be added to the appropriate class list. The list of system memory classes can be obtained using crashconf -v. The memory page size is 4Kb.

class may also be the word all, in which case all classes are added to the appropriate list. (The effect of adding all classes to the included class list is to force full crash dumps under all circumstances. The effect of adding all classes to the excluded class list is to disable crash dumps.)

mode, either ON or OFF, will turn compression ON or OFF, in the dump path.

Options

-a The file /etc/fstab is read, and all dump devices identified in it will be added to (or will replace) the current list of crash dump devices. This is in addition to any crash dump devices specified on the command line. See fstab(4) for information on the format of /etc/fstab.

-c The mode specified with -c will be used to set the compression mode. If the system is not able to identify enough processing resources to do compressed dump, a warning message will be issued.

-e The classes specified with -e will be added to (or will replace) the list of excluded (i.e., should not dump) classes. If any of those classes are present in the current included class list, they will be removed from it.

-i The classes specified with -i will be added to (or will replace) the list of included (i.e., must dump) classes. If any of those classes are present in the current excluded class list, they will be removed from it.

-r Specifies that any changes should replace, rather than add to, the current configuration. Thus, if devices or -a are specified, the current crash dump device list is replaced with new contents; if classes are specified with -e, they replace the list of currently excluded classes, and if classes are specified with -i, they replace the list of currently included classes.
When used with the -i, -e or -c options, sets the dump tunables alwaysdump, dontdump and dump_compress_on respectively, to make the changes persistent across system reboots.

Displays the current crash dump configuration. This is the default option if no arguments are specified. If any changes to the current configuration are specified on the same command line as -v, the configuration will be displayed after the requested changes are made.

RETURN VALUE
Upon exit, crashconf returns the following values:

0  Success.
1  The requested configuration changes could not be made.

WARNINGS
On systems running VxVM 3.5, the swap volumes to be configured for system crash dumps should be created with the usage type as swap during the creation of the swap volume. Not doing so will cause dump corruption. You could use the -U option of vxassist(1M) to do the same.

The output of crashconf is not designed to be parsed by applications or scripts, but only to be read by humans. The output format may change without notice. Applications which require crash dump configuration information should retrieve that information using pstat(2).

Dump devices created by lvcreate(1M) must be contiguous (-Cy option) with bad block relocation turned off (-rn option).

High Availability for HP’s A-A Dump Devices
The product, DUMPAAEnable Ver B.11.23.0609.01, enables high-availability features for HP’s Active-Active (A-A) Disk Array devices when configured as dump devices.

The product Securepath version A.3.0F.02F.00F or higher is also required to be installed along with this product on the system, for enabling these features.

This product provides the following features for HP’s A-A devices

1. Active path configuration: An Active-Active device can be configured as a dump device through any of the paths to the device. If the path through which dump configuration is being attempted is currently offline, then an alternate active path will be used for configuration. Use -l to display the actual path used for configuration. If all the paths to the device are offline, then configuration will fail.

2. Failover/Path change: If the path that was used for dump configuration of an Active-Active device goes offline, then an alternate active path will be automatically configured for dump. Use -l to display the actual path used for dump.

3. Duplicate device identification: An Active-Active device will have multiple device special files (/dev/dsk/c#t#d# files), one corresponding to each path to that device. Configuration of an existing Active-Active dump device through an alternate device special file (c#t#d#) is considered as a duplicate device and configuration will not be allowed.

A-A Dump Option
crashconf recognizes the following new options:

- Displays the current crash dump configuration similar to -v. The -l option displays actual path (the path used for dump configuration) in addition to user configured path (the path specified by the user) for A-A dump devices. The actual path is displayed in brackets right below the user configured path, if the actual path is different from the user configured path. The paths are displayed using the corresponding device special files (c#t#d#). If there are no paths available to a dump device it is shown as OFFLINE.

Actual path information will be shown only for A-A dump devices that are full devices.

AUTHOR

crashconf was developed by HP.

SEE ALSO

lvcreate(1M), vxassist(1M), crashconf(2), pstat(2), fstab(4).
crashutil - manipulate crash dump data

**SYNOPSIS**
```
/usr/sbin/crashutil [-q] [-v version] source [destination]
```

**DESCRIPTION**
`crashutil` copies and preserves crash dump data, and performs format conversions on it. Common uses of `crashutil` include:

- Copying portions of a dump that still reside on a raw dump device into a crash dump directory.
- Converting between different formats of crash dumps.
- Copying crash dumps from one directory, or medium, to another.

`crashutil` will write to its `destination` the crash dump it reads from its `source`. The crash dump format used to write the `destination` is specified with `-v`; if `-v` is not specified, the `destination` will have the same format as the `source`. If no `destination` is specified, `source` is used; the format conversion will be done in place in the `source`, without copying. When `crashutil` completes successfully, the entire contents of the crash dump will exist at `destination`; any portions that had still been on raw dump devices will have been copied to `destination`.

There are three known dump formats:

- **COREFILE** (Version 0) This format, used up through HP-UX 10.01, consists of a single file containing the physical memory image, with a 1-to-1 correspondence between file offset and memory address. Normally there is an associated file containing the kernel image. `sources` or `destinations` of this type must be specified as two pathnames to plain files, separated by whitespace; the first is the core image file and the second is the kernel image file.

- **COREDIR** (Version 1) This format, used in HP-UX 10.10, 10.20, and 10.30, consists of a `core.n` directory containing an `INDEX` file, the kernel (`vmunix`) file, and numerous `core.n.m` files, which contain portions of the physical memory image. `sources` or `destinations` of this type should be specified as the pathname to a core directory.

- **CRASHDIR** (Version 2) This format, used in HP-UX 11.00 and later, consists of a `crash.n` directory containing an `INDEX` file, the kernel and all dynamically loaded kernel module files, and numerous `image.m.p` files, each of which contain portions of the physical memory image and metadata describing which memory pages were dumped and which were not. `sources` or `destinations` of this type should be specified as the pathname to a crash directory.

- **PARDIR** (Version 3) This format is used in HP-UX Release 11i Version 1.0 and later. It is very similar in structure to the CRASHDIR format in that it consists of a `crash.n` directory containing an `INDEX` file, the kernel and all dynamically loaded kernel module files, and numerous `image.m.p` files, each of which contain portions of the physical memory image and metadata describing which memory pages were dumped and which were not. In addition to the primary `INDEX` file, there are auxiliary index files, that contain metadata describing the image files containing the memory pages. This format will be used when the dump is compressed. See `crashconf(1M)`.

Other formats, for example tape archival formats, may be added in the future.

When the `source` and `destination` are different types of files — for example, when `source` is a directory and `destination` is a pair of plain files — both must be specified.

**Options**

- `-q` *(Quiet)* Disables the printing of progress messages. Warning and error messages are still printed.

- `-v version` Specifies the version of the destination format. Allowed values are `COREFILE`, `COREDIR`, `CRASHDIR`, `PARDIR`, 0, 1, 2 or 3. Also allowed is the keyword `CURRENT`, which specifies that the destination format should be the same as the current source format. `CURRENT` is the default if `-v` is not specified. If the destination format is `PARDIR`, then the source format should also be `PARDIR`. Conversion to `PARDIR` from older formats is not supported.

**RETURN VALUE**
Upon exit, `crashutil` returns the following values:

HP-UX 11i Version 2: December 2007 Update -- 1 -- Hewlett-Packard Company 149
crashutil(1M)

0 The operation was successful.
1 The operation failed, and an appropriate error message was printed.

EXAMPLES
An HP-UX 11.00 crash dump was saved by savecrash(1M) to /var/adm/crash/crash.2. The -p flag was specified to savecrash, specifying that only those portions of the dump which were endangered by swap activity should be saved; the rest are still resident in the raw dump devices. To save the remainder of the dump into the crash dump directory, use:

    crashutil /var/adm/crash/crash.2

If preferred, the completed crash dump directory could be in a different location — perhaps on another machine via NFS:

    crashutil /var/adm/crash/crash.2 /nfs/remote/otherdir

To debug this crash dump using tools which do not understand the most current crash dump format, convert it to the older core directory format:

    crashutil -v COREDIR /var/adm/crash/crash.2 /tmp/oldcoredir

or the even older "core file and kernel" format:

    crashutil -v COREFILE /var/adm/crash/crash.2 /tmp/corefile /tmp/kernfile

AUTHOR
crashutil was developed by HP.

SEE ALSO
savecrash(1M), crashconf(1M).
create_sysfile(1M)

NAME
create_sysfile - create a kernel system file

SYNOPSIS
/usr/sbin/sysadm/create_sysfile [outfile]

DESCRIPTION
The create_sysfile command creates a kernel configuration description file (system file) which can be used as input to the commands kconfig(1M) or mk_kernel(1M). The system file that is generated is of type version 1 (see system(4)) and is built according to the drivers required by the current system hardware.

This command is intended for use during the install process when the system does not have a system file.

The create_sysfile command uses a template file as a starting point and then it scans the system hardware and adds all drivers it can identify to run the existing hardware. If outfile is specified, the resulting system file is sent to outfile. If outfile is not specified, the output is placed in the file /stand/system.

RETURN VALUE
Upon completion, create_sysfile returns with one of the following exit values:

 0  Successful.
 1  One or more errors were reported.

DIAGNOSTICS
Errors are sent to stderr. Most of the diagnostic messages from create_sysfile are self-explanatory. Errors cause create_sysfile to halt immediately.

AUTHOR
create_sysfile was developed by HP.

FILES
/usr/conf/gen/templates/*

SEE ALSO
kconfig(1M), mk_kernel(1M), system(4), kconfig(5).
cron(1M)

NAME
cron - timed-job execution daemon

SYNOPSIS
/usr/sbin/cron

DESCRIPTION
cron executes commands at specified dates and times. Regularly scheduled commands can be specified according to instructions placed in crontab files. Users can submit their own crontab files with a crontab command (see crontab(1)). Users can submit commands that are to be executed only once with an at or batch command.

Since cron never exits, it should be executed only once. This is best done by running cron from the initialization process with the startup script /sbin/init.d/cron (see init(1M)).

cron only establishes a schedule for crontab files and at/batch command files during process initialization and when it is notified by at, batch, or crontab that a file has been added, deleted, or modified.

When cron executes a job, the job's user and group IDs are set to those of the user who submitted the job.

Spring and Autumn Time Transitions
On the days of daylight savings (summer) time transition (in time zones and countries where daylight savings time applies), cron schedules commands differently from normal.

In the following description, an ambiguous time refers to an hour and minute that occurs twice in the same day because of a daylight savings time transition (usually on a day during the Autumn season). A nonexistent time refers to an hour and minute that does not occur because of a daylight savings time transition (usually on a day during the Spring season). DST-shift refers to the offset that is applied to standard time to result in daylight savings time. This is normally one hour, but can be any combination of hours and minutes up to 23 hours and 59 minutes (see tztab(4)).

When a command is specified to run at an ambiguous time, the command is executed only once at the first occurrence of the ambiguous time.

When a command is specified to run at a nonexistent time, the command is executed after the specified time by an amount of time equal to the DST-shift. When such an adjustment would conflict with another time specified to run the command, the command is run only once rather than running the command twice at the same time.

Commands that are scheduled to run during all hours (there is a * in the hour field of the crontab entry) are scheduled without any adjustment.

EXTERNAL INFLUENCES
Environment Variables
LANG determines the language in which messages are displayed.

If LANG is not specified or is set to the empty string, it defaults to “C” (see lang(5)). If any internationalization variable contains an invalid setting, all internationalization variables default to “C” (see environ(5)).

DIAGNOSTICS
A history of all actions taken by cron is recorded in /var/adm/cron/log.

EXAMPLES
The following examples assume that the time zone is MST7MDT. In this time zone, the DST transition occurs one second before 2:00 a.m. and the DST-shift is 1 hour.

Consider the following entries in a crontab file:

<table>
<thead>
<tr>
<th># Minute</th>
<th>Hour</th>
<th>MonthDay</th>
<th>Month</th>
<th>Weekday</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>01</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>Job_1</td>
</tr>
<tr>
<td>0</td>
<td>02</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>Job_2</td>
</tr>
<tr>
<td>0</td>
<td>03</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>Job_3</td>
</tr>
<tr>
<td>0</td>
<td>04</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>Job_4</td>
</tr>
<tr>
<td>0</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>Job_hourly</td>
</tr>
<tr>
<td>0</td>
<td>2,3,4</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>Multiple_1</td>
</tr>
<tr>
<td>0</td>
<td>2,4</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>Multiple_2</td>
</tr>
</tbody>
</table>
For the period of 1:00 a.m. to 4:00 a.m. on the days of DST transition, the results will be:

<table>
<thead>
<tr>
<th>Job</th>
<th>Times Run in Fall</th>
<th>Times Run in Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job_1</td>
<td>01:00 MDT</td>
<td>01:00 MST</td>
</tr>
<tr>
<td>Job_2</td>
<td>02:00 MDT</td>
<td>03:00 MDT</td>
</tr>
<tr>
<td>Job_3</td>
<td>03:00 MST</td>
<td>03:00 MDT</td>
</tr>
<tr>
<td>Job_4</td>
<td>04:00 MST</td>
<td>04:00 MDT</td>
</tr>
<tr>
<td>Job_hourly</td>
<td>01:00 MDT</td>
<td>01:00 MST</td>
</tr>
<tr>
<td></td>
<td>02:00 MDT</td>
<td>02:00 MST</td>
</tr>
<tr>
<td></td>
<td>03:00 MST</td>
<td>03:00 MDT</td>
</tr>
<tr>
<td></td>
<td>04:00 MST</td>
<td>04:00 MDT</td>
</tr>
<tr>
<td>Multiple_1</td>
<td>02:00 MDT</td>
<td>03:00 MDT</td>
</tr>
<tr>
<td></td>
<td>04:00 MDT</td>
<td>04:00 MDT</td>
</tr>
<tr>
<td>Multiple_2</td>
<td>02:00 MDT</td>
<td>03:00 MDT</td>
</tr>
<tr>
<td></td>
<td>04:00 MDT</td>
<td>04:00 MDT</td>
</tr>
</tbody>
</table>

**WARNINGS**
In the Spring, when there is a nonexistent hour because of daylight savings time, a command that is scheduled to run multiple times during the nonexistent hour will only be run once. For example, a command scheduled to run at 2:00 and 2:30 a.m. in the MST7MDT time zone will only run at 3:00 a.m. The command that was scheduled at 2:30 a.m. will not be run at all, instead of running at 3:30 a.m.

**DEPENDENCIES**
**HP Process Resource Manager**
If the optional HP Process Resource Management (PRM) software is installed and configured, jobs are launched in the initial process resource group of the user that scheduled the job. The user's initial group is determined at the time the job is started, not when the job is scheduled. If the user's initial group is not defined, the job runs in the user default group (PRMID=1). See `prmconfig(1)` for a description of how to configure HP PRM, and `prmconf(4)` for a description of how the user's initial process resource group is determined.

**AUTHOR**
cron was developed by AT&T and HP.

**FILES**
/var/adm/cron Main cron directory
/var/spool/cron/atjobs Directory containing at and batch job files
/var/spool/cron/crontabs Directory containing crontab files
/var/adm/cron/log Accounting information

**SEE ALSO**
at(1), crontab(1), sh(1), init(1M), queuedefs(4), tztab(4).


**STANDARDS CONFORMANCE**
cron: SVID2, SVID3
NAME
dcopy - copy HFS file system with compaction.

SYNOPSIS
source_fs destination_fs

DESCRIPTION
The dcopy command copies an existing HFS file system (source_fs) to a new HFS file system
(destination_fs), appropriately sized to hold the reorganized results. For best results, the source file system
should be a raw device, and the destination file system should be a block device. Always run dcopy on
unmounted file systems. (In the case of the root file system, copy it to a new minidisk.)

If no options are specified, dcopy copies files from source_fs, compressing directories by removing vacant
entries and spacing consecutive blocks in a file by the optimal rotational gap. If options such as -f or -s
are specified, the destination file system structure will be different from that of the source file system.

dcopy makes the destination file system identical to the source file system and preserves the pack and
volume labels. Thus, to compress a file system without moving it, use dcopy to copy the files to another
file system and the dd command to copy the file back (see dd(1)).

Directory compression is accomplished by running dcopy on the primary copy of the file system and allowing
the modified directories to propagate to the other copies of the file system in the normal manner.

Options
dcopy recognizes the following options:
- d
  Move subdirectories to the beginning of directories.
- f size[:size]
  Specify the file system size (fsize) and inode-list size (isize) in blocks. If this option is
  not specified, the source file-system value is used.
- F hfs
  Specify the HFS file system type. The type of a file system can be determined with
  the fstyp command (see fstyp(1M)). See DEPENDENCIES.
- s cyl:skip
  Supply device information for creating the best organization of blocks in a file. cyl is
  the number of block per cylinder; skip is the number of blocks to skip.
- v
  Report size of source and destination file system.
- V
  Echo the completed command line, but performs no other actions. The command line
  is generated by incorporating the user-specified options and other information derived
  from /etc/fstab. This option allows you to verify the command line.

EXAMPLES
dcopy can be executed with or without options. If no options are specified as in this example, the source
and destination file systems are identical. Any differences between the two file systems lie only in the
available disk space.

dcopy /dev/rdsk/c2d0s4 /dev/dsk/c2d0s5

If options are specified, expect a major difference between the source and destination file system structure:

dcopy -F hfs -f40960:260 -s45:5 -d /dev/rdsk/c2d0s4 /dev/dsk/c2d0s5

WARNINGS
dcopy produces invalid results if run on a mounted file system.

The figures specified in option arguments cannot be smaller than corresponding figures in the source file
system.

DEPENDENCIES
dcopy only operates on HFS file systems.

AUTHOR
dcopy was developed by HP.

SEE ALSO
dd(1), fstyp(1M).

154 Hewlett-Packard Company - 1 - HP-UX 11i Version 2: December 2007 Update
dcopy(1M)

STANDARDS CONFORMANCE
dcopy: SVID3
NAME
devnm - device name

SYNOPSIS
/usr/sbin/devnm [name ...]

DESCRIPTION
For each name specified, the devnm command identifies the special file associated with the mounted file system where the named file or directory resides.

EXAMPLES
The command:
/usr/sbin/devnm /usr
produces:
/dev/dsk/c1d0s9 /usr
if /usr is mounted on /dev/dsk/c1d0s9.

FILES
/dev/dsk/*
/etc/mnttab Mounted file system table.

STANDARDS COMPLIANCE
devnm: SVID2, SVID3
NAME
df - report number of free file system disk blocks

SYNOPSIS
/usr/bin/df [-F FStype] [-befgiklnsv] [-t|-P] [-o specific_options] [-V]

DESCRIPTION
The df command displays the number of free 512-byte blocks and free inodes available for file systems by examining the counts kept in the superblock or superblocks. If a special or a directory is not specified, the free space on all mounted file systems is displayed. If the arguments to df are path names, df reports on the file systems containing the named files. If the argument to df is a special of an unmounted file system, the free space in the unmounted file system is displayed.

Options
df recognizes the following options:

- `b` Report only the number of kilobytes (KB) free.
- `e` Report the number of files free.
- `f` Report only the actual count of the blocks in the free list (free inodes are not reported).
- `F FStype` Report only on the FStype file system type (see fstyp(1M)).
- `g` Report the entire structure described in statvfs(2).
- `i` Report the total number of inodes, the number of free inodes, number of used inodes, and the percentage of inodes in use.
- `k` Report the allocation in kilobytes (KB).
- `l` Report on local file systems only.
- `n` Report the file system name. If used with no other options, display a list of mounted file system types.
- `o specific_options` Specify options specific to each file system type. specific_options is a comma-separated list of suboptions intended for a specific FStype module of the command. See the file-system-specific manual entries for further details.
- `P` Report the name of the file system, the size of the file system, the number of blocks used, the number of blocks free, the percentage of blocks used and the directory below which the file system hierarchy appears.
- `s` Do not sync the file system data on the disk before reporting the usage.
- `t` Report the total allocated block figures and the number of free blocks.
- `v` Report the percentage of blocks used, the number of blocks used, and the number of blocks free. This option cannot be used with other options.
- `V` Echo the completed command line, but perform no other action. The command line is generated by incorporating the user-specified options and other information derived from /etc/fstab. This option allows the user to verify the command line.

EXTERNAL INFLUENCES
Environment Variables
`LC_MESSAGES` determines the language in which messages are displayed.

If `LC_MESSAGES` is not specified in the environment or is set to the empty string, the value of LANG is used as a default for each unspecified or empty variable. If LANG is not specified or is set to the empty string, a default of "C" (see lang(5)) is used instead of LANG.

If any internationalization variable contains an invalid setting, df behaves as if all internationalization variables are set to "C". See environ(5).

International Code Set Support
Single-byte and multi-byte character code sets are supported.

EXAMPLES
Report the number of free disk blocks for all mounted file systems:
\texttt{df}

Report the number of free disk blocks for all mounted HFS file systems:
\texttt{df -F hfs}

Report the number of free files for all mounted NFS file systems:
\texttt{df -F nfs -e}

Report the total allocated block figures and the number of free blocks, for all mounted file systems:
\texttt{df -t}

Report the total allocated block figures and the number of free blocks, for the file system mounted as /usr:
\texttt{df -t /usr}

FILES
/\texttt{dev/dsk/*} File system devices
/\texttt{etc/fstab} Static information about the file systems
/\texttt{etc/mnttab} Mounted file system table

SEE ALSO
du(1), df_hfs(1M), df_vxfs(1M), fsck(1M), fstab(4), fsyp(1M), statvfs(2), mnttab(4).

STANDARDS CONFORMANCE
\texttt{df}: SVID2, SVID3, XPG2, XPG3, XPG4
NAME
df_hfs: df - report number of free CDFS, HFS, or NFS file system disk blocks

SYNOPSIS
/usr/bin/df [-F FStype] [-befgikntv] [-B] [-o specific_options] [-V] [special | directory]...

DESCRIPTION
The df command displays the number of free 512-byte blocks and free inodes available for file systems by examining the counts kept in the superblock or superblocks. If a special or a directory is not specified, the free space on all mounted file systems is displayed. If the arguments to df are path names, df reports on the file systems containing the named files. If the argument to df is a special of an unmounted file system, the free space in the unmounted file system is displayed.

Options
df recognizes the following options:
-\(b\) Report only the number of kilobytes (KB) free.
-\(B\) Report the total number of blocks allocated for swapping to the file system as well as the number of blocks free for swapping to the file system. This option is supported on HFS file systems only.
-\(e\) Report the number of files free.
-\(f\) Report only the actual count of the blocks in the free list (free inodes are not reported). When this option is specified, df reports on raw devices.
-\(F\) FStype Report only on the FStype file system type (see fstyp(1M)). For the purposes of this manpage, FStype can be one of cdfs, hfs, and nfs, for the CDFS, HFS, and NFS file systems, respectively.
-\(g\) Report the entire structure described in statvfs(2).
-\(i\) Report the total number of inodes, the number of free inodes, number of used inodes, and the percentage of inodes in use.
-\(k\) Report the allocation in kilobytes (KB).
-\(l\) Report on local file systems only.
-\(n\) Report the file system name. If used with no other options, display a list of mounted file system types.
-\(o\) specific_options Specify options specific to the HFS file system type. specific_options is a comma-separated list of suboptions.
The available suboption is:
-\(i\) Report the number of used and free inodes.
-\(t\) Report the total allocated block figures and the number of free blocks.
-\(v\) Report the percentage of blocks used, the number of blocks used, and the number of blocks free. This option cannot be used with other options.
-\(V\) Echo the completed command line, but perform no other action. The command line is generated by incorporating the user-specified options and other information derived from /etc/fstab. This option allows the user to verify the command line.

When df is used on an HFS file system, the file space reported is the space available to the ordinary user, and does not include the reserved file space specified by fs_minfree.

Unreported reserved blocks are available only to users who have appropriate privileges. See tunefs(1M) for information about fs_minfree.

When df is used on NFS file systems, the number of inodes is displayed as -1. This is due to superuser access restrictions over NFS.

EXAMPLES
Report the number of free disk blocks for all mounted file systems:
**df_hfs(1M)**

**df**
Report the number of free disk blocks for all mounted HFS file systems:

```
    df -F hfs
```

Report the number of free files for all mounted NFS file systems:

```
    df -F nfs -e
```

Report the total allocated block figures and the number of free blocks, for all mounted file systems:

```
    df -t
```

Report the total allocated block figures and the number of free blocks, for the file system mounted as `/usr`:

```
    df -t /usr
```

**WARNINGS**

**df** does not account for:

- Disk space reserved for swap space,
- Space used for the HFS boot block (8K bytes, 1 per file system),
- HFS superblocks (8K bytes each, 1 per disk cylinder),
- HFS cylinder group blocks (1K-8K bytes each, 1 per cylinder group),
- Inodes (currently 128 bytes reserved for each inode).

Non-HFS file systems may have other items that this command does not account for.

The `-b` option, from prior releases, has been replaced by the `-B` option.

**FILES**

- `/dev/dsk/*` File system devices.
- `/etc/fstab` Static information about the file systems
- `/etc/mnttab` Mounted file system table

**SEE ALSO**

du(1), df(1M), fsck(1M), fstab(4), fstyp(1M), statvfs(2), mnttab(4).

**STANDARDS CONFORMANCE**

df: SVID2, XPG2, XPG3
**NAME**

df_vxfs: df - report number of free disk blocks on a VxFS file system

**SYNOPSIS**

df [-F vxfs] [-V] [-befgiklnvt] [-o s] [special | directory]...

**DESCRIPTION**

df prints the number of free blocks and free inodes in VxFS file systems or directories based on the counts kept in the super-blocks.

VxFS dynamically allocates inodes from a pool of free blocks. The number of free inodes and blocks reported by df is an estimate based on the number of free 8K or larger extents and the current ratio of allocated inodes to allocated blocks. (Extents smaller than 8K may not be usable for all types of allocation, so df does not count free blocks in extents smaller than 8K.) Allocating additional blocks may therefore decrease the count of free inodes and vice versa.

If the operand to df is a special device name, the file system can be an unmounted or mounted file system (for example, /dev/dsk/c0t1d0). If you specify a directory name, df displays information for the file system at that mount point. If neither special nor directory is specified, the free space on all of the mounted file systems is printed.

**Options**

df recognizes the following options:

- **-b** Report only the number of kilobytes free.
- **-e** Report the number of files free.
- **-f** Report only an actual count of the blocks in the free list (free inodes are not reported). When this option is specified, df reports on raw devices.
- **-F vxfs** Specify the file system type (vxfs).
- **-g** Report the entire statvfs(2) structure.
- **-i** Report the total number of inodes, the number of free inodes, number of used inodes and the percentage of inodes in use.
- **-k** Report the allocation in kilobytes.
- **-l** Report on local file systems only.
- **-n** Report the file system name. If invoked with no other options this option prints a list of mounted file system types.
- **-o s** Print the number of free extents of each size. Free extents are always an integral power of two in length, ranging from a minimum of one block to the maximum extent size supported by the file system.
- **-t** Report the total allocated block figures and the number of free blocks.
- **-v** Report the percentage of blocks used, the number of blocks used and the number of blocks free. This option cannot be used with other options.
- **-V** Echo the completed command line, but performs no other action. The command line is generated by incorporating the user-specified options and other information derived from /etc/fstab. This option allows the user to verify the command line.

There are a number of options that specify output formats, some combinations of which are incompatible. If an incompatible combination is specified, one of the options will override the other(s).

**Operands**

df recognizes the following operands:

- **directory** Name of the mount point from which the VxFS file system is accessed.
- **special** Device name, which contains a mounted or unmounted VxFS file system.

**EXAMPLES**

Report the number of free disk blocks for all mounted file systems:
Report the number of free extents of each size, for all mounted VxFS file systems:

```
    df -F vxfs -o s
```

Report the number of free files for all mounted VxFS file systems:

```
    df -F vxfs -e
```

Report the total allocated block figures and the number of free blocks, for all mounted file systems:

```
    df -t
```

Report the total allocated block figures and the number of free blocks, for the file system mounted as `/usr`:

```
    df -t /usr
```

**FILES**

- `/dev/vg00/*` - File-system devices.
- `/dev/dsk/*` - File-system devices.
- `/etc/fstab` - Static information about the file systems.
- `/etc/mnttab` - mounted-file-system table.

**SEE ALSO**

du(1), df(1M), fsck_vxfs(1M), statvfs(2), fs_vxfs(4), mnttab(4).

**STANDARDS CONFORMANCE**

df: SVID2, XPG2, XPG3
NAME
dhcpclient - Client for Dynamic Host Configuration Protocol Server

SYNOPSIS

[-p] [-s server_ipaddress] [-t log_type] [-x send_attempts] [-z time_to_wait]
dhcpclient -d interface [-s server_ipaddress]
[-t log_type] [-z time_to_wait]
dhcpclient -v

DESCRIPTION
dhcpclient is used to obtain the configuration parameters from the Dynamic Host Configuration Protocol (DHCP) server (bootpd(1M)) to configure the host. Currently, dhcpclient supports 10bt (10BaseT) and 100bt (100BaseT) Ethernet networks only. It does not support FDDI and Token-Ring networks.
dhcpclient is invoked by the auto_parms script (see auto_parms(1M)) at boot up. The /etc/rc.config.d/netconf file is used for the configuration of the interfaces. To invoke dhcpclient using the auto_parms script, set the DHCP_ENABLE[index] variable to a non-zero value in the /etc/rc.config.d/netconf file. dhcpclient can also be invoked at the command prompt, when the interface is not configured.

Once the dhcpclient obtains the configuration parameters from the server, it is stored in a configuration file named /etc/dhcpclient.data. The configuration parameters are valid for a fixed time, lease time.

The configuration information in /etc/dhcpclient.data file will be of the following format.

  <code> <length> <data>
  00  <length> IFname
  01  <length> hostname
  02  <length> sname
  03  <length> bootfile
  04  <length> NIS_domain
  05  <length> domain_name
  06  <length> lease_duration
  07  <length> lease_expiration
  08  <length> T1
  09  <length> T2
  10  <length> htype
  11  <length> chaddr
  12  <length> IP_addr
  13  <length> subnet_mask
  14  <length> broadcast_addr
  15  <length> default_gateway
  16  <length> server_addr
  17  <length> boot_server
  18  <length> dest_gateway
  19  <length> DNS_server
  20  <length> LPR_server
  21  <length> swap_server
  22  <length> NIS_server
  23  <length> NTP_server
  24  <length> raw_options

The length field denotes the length of the data field following it.

All these lines collectively indicate one record. The configuration file will have series of records.

Since this file contains all the configuration information, it can be used to identify the server from which the configuration parameters were obtained. The configuration file can also be used to obtain the same configuration parameters from the server. So, once the dhcpclient is invoked, it will try to read the
configuration file, /etc/dhcpclient.data.

While trying to read the /etc/dhcpclient.data file, the following conditions are possible:

1. It may be possible that the configuration file had been deleted or it may not be available for reading.
2. The file may have got corrupted and contains irrelevant data for the particular interface.
3. There may be no configuration information for the particular interface.
4. User may have specified -n option meaning that a new lease is needed.

For all of the above cases, dhcpclient has to first find the server which is ready for the service. For this purpose, it will send a DHCPDISCOVER message. If the user has specified server ip-address using the -s option, then this message will be sent to that ip-address, else, this message is broadcasted. The server in turn will reply with the configuration parameters that it can offer. This message is called DHCPOFFER. If the client has sent the DHCPDISCOVER to broadcast address, then it will receive DHCPOFFER from more than one server. The dhcpclient selects one of the DHCPOFFER obtained. Then, it will send DHCPREQUEST to the server, requesting the configuration parameters. The server will reply with a positive acknowledgement called DHCPACK, if it is willing to send the configuration parameters to the host and with a negative acknowledgement, if it is not willing to do so. If the dhcpclient has received DHCPACK, it will copy all the configuration parameters to the configuration file, else, it will send another DHCPDISCOVER message and repeat the whole process. This process is called "Obtaining a new lease".

If all the above 4 cases are not true, then the dhcpclient can directly send the DHCPREQUEST message and obtain the configuration parameters as mentioned above. This process is called 'Renewing the lease'.

The various options supported by dhcpclient are as described below.

Options
-G When this option is used, the client will accept the reply from the DHCP server only if the class-id in the request and reply packet matches.
-N hostname This option is used to send a hostname. The hostname should be specified as an ASCII string.
-b interface This option is used during bootup time to obtain a new lease. If the client is trying to obtain the configuration parameters from the server for the first time, then /etc/dhcpclient.data does not exist. So, using this option, the client obtains a new lease for the interface specified.
-d interface This option is used for dumping the internal data of an interface. A sample dump is shown below.

<table>
<thead>
<tr>
<th>Interface</th>
<th>PPA</th>
<th>MAC Type</th>
<th>Physical Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>lan0</td>
<td>0</td>
<td>ETHER</td>
<td>0x080009709631</td>
</tr>
</tbody>
</table>
-f tracefile This option is used to specify the file name for the debugging output.
-g class-id This option is used for specifying the class-id of the device group. class-id should be specified as an ASCII string.
-l debug_level This option is used for logging of debug information. debug_level specifies the amount of debug information logged. The debugging messages are sent to STDERR.
-m interface This option is used for maintaining the lease. When invoked with this option, the dhcpclient runs as a daemon. The dhcpclient calculates how much time it needs to sleep before the lease needs to be renewed and then sleeps for that time. When it wakes up, it extends the lease by sending the request to the DHCP server.
-n This option is used for ignoring the current lease. When this option is used, the current configuration information is discarded and request for a new lease is sent.
-p This option is used for tracing the packets. It displays the packet contents in a readable format.
-r interface This option is used for releasing a lease. The dhcpclient releases all the assigned configuration parameters. This is done by sending the DHCPRELEASE message to the server. The dhcpclient then empties the record corresponding to that interface in the configuration file, /etc/dhcpclient.data.
**-s server_ipaddress**
This option is used to specify the ip address of the **DHCP** server to which the request should be sent. **server_ipaddress** should be specified in dotted decimal form.

**-t log_type**
If the **log_type** is a non zero value and if the **debug_level** is low enough for the process to detach from the tty (see **syslog(3C)**), then, the logging goes to **syslog**, else it goes to **STDERR**. This option is used when the **dhcpcd** is running as a daemon.

**-v**
This option prints the various version strings of the **dhcpcd** to stdout.

**-x send_attempts**
This option is used to specify the number of attempts the packet should be sent before obtaining a reply. **send_attempts** must be specified as an unsigned number.

**-z time_to_wait**
This option is used to specify the time interval between successive packet retransmission. **time_to_wait** will be in seconds.

**EXAMPLES**
To maintain the lease for the interface lan0 and invoke logging in **syslog**, the following set of commands is invoked at the command prompt:
```
dhcpcd -b lan0 -N "dhcpl"
dhcpcd -m lan0 -N "dhcpl" -l 3 -t 1
```
To trace the packet and get the debugging messages, use:
```
dhcpcd -b lan0 -N "dhcpl" -p -l 8
```
This is an example of a /etc/dhcpcd.data file.
```
00 4 lan0
01 5 dhcp
02 0
03 0
04 0
05 0
06 4 4294967295
07 4 4294967295
08 4 0
09 4 0
10 4 1
11 6 8 0 9 25 a5 b1
12 4 192.11.22.107
13 4 255.255.255.0
14 4 0.0.0.0
15 0
16 4 192.11.22.3
17 4 192.11.22.3
18 0
19 0
20 0
21 4 0.0.0.0
22 0
23 0
24 39 63 82 53 63 35 1 5 33 4 ff ff ff ff 36 4 c0 b8
16 3 1 4 ff ff ff 0 c 5 64 68 63 70 68 2 4 ff ff b2 a8 ff
```

**AUTHOR**

**dhcpcd** was developed by HP.

**FILES**

/etc/dhcpcd.data configuration file.
SEE ALSO
auto_parms(1M), bootpd(1M), bootpquery(1M), dhcptools(1M), syslog(3C).
DARPA Internet Request For Comments: RFC1541, RFC1542, RFC1533, RFC1534, Assigned Numbers
RFC.
NAME
dhcpdb2conf - convert DHCP client database to config file parameters

SYNOPSIS
dhcpdb2conf [-pac] [-hdirt] [-s index] [lan-interfaces]

DESCRIPTION
The dhcpdb2conf command translates a client system’s DHCP database parameters (from
 dhcpclient(1M)) into a set of standard configuration file parameters. This is an essential step for
 configuring a DHCP client system, and is invoked by auto_parms(1M) upon every reboot when DHCP
 is active.

A DHCP client database can contain settings for such items as: hostname, IP address, and default gateway
(a subset of the “initial identity parameters” managed by set_parms(1M) and geocustoms(1M)). You can list
the contents of the database to the screen, create a set of configuration staging files, or execute direct edits
on existing configuration files using the values contained in the client database.

Arguments: You can provide a list of lan-interfaces on which to operate, such as "lan0 lan1". If you specify
no LAN interfaces, dhcpdb2conf processes all entries referenced in the client’s DHCP database. The
entries are each defined as a unique LAN interface and a corresponding list of attributes.

Options
The following options determine the results of DHCP parameter processing.

-p Print results to the screen (standard output). This is the default action if neither -a nor -c is
 specified.

-a Apply directly: Using the results of the specified filters (see below), directly apply the parame-
ter definitions to the existing configuration files, for example, to
/etc/rc.config.d/netconf. (This is how dhcpdb2conf is called from
 auto_parms.)

-c Create copies: Create a set of staging files using the results of the selected filters (see below). Each
parameter processed is applied to its corresponding copy of one configuration file
[re]created by dhcpdb2conf.

For example, /etc/rc.config.d/netconf is copied to
/etc/rc.config.d/netconf.dhcp. (If the file already exists and can be written, it is
overwritten; otherwise the command fails.) Once this staging file is created, the parameter
being processed is applied to the newly created staging file rather than the real configuration
file.

The following "filter" options control which parameters are processed. The options can be combined in any
manner. The default with no options is to process all attributes for each LAN interface.

-h Hostname: Process the HOSTNAME parameter.

-d DNS: Process the DNS parameter set in /etc/resolv.conf: domain, nameserver.

-i Interface: Process the INTERFACE parameter set: INTERFACE_NAME[],
 IP_ADDRESS[], SUBNET_MASK[], BROADCAST_MASK[], LANCONFIG_ARGS[]

-r Route: Process the ROUTE parameter set: ROUTE_DESTINATION[],
 ROUTE_GATEWAY[], ROUTE_COUNT[]

-n NIS: Process the NIS parameter set: NISDOMAIN, YPSET_ADDR

-t Time: Process the NTPDATE_SERVER parameter.

-s index Set array index for the set of related parameters for one LAN interface in the configuration
files. For example, with -s0 (the default) the output would include
INTERFACE_NAME[0]=lan0.

The dhcpdb2conf command can be run only by the user with appropriate privilege.

Configuration Files and Parameter Names
The following files, and parameters in each file, can be processed by dhcpdb2conf:

/etc/resolv.conf
EXAMPLES
List the entire contents of the DHCP client database:
   dhcpdb2conf
List only the INTERFACE parameter set for lan0:
   dhcpdb2conf -i lan0
List the INTERFACE and ROUTE parameter sets for lan0 and lan1:
   dhcpdb2conf -ir lan0 lan1
Apply the INTERFACE and ROUTE parameter sets for lan0 to the existing configuration files:
   dhcpdb2conf -ira lan0
Apply all parameter sets to the existing configuration files using lan0, and set index = 1:
   dhcpdb2conf -a -s 1 lan0

WARNINGS
Using the -a option overrides any existing values which are currently set in the system’s configuration files (while the -c option merely revises staging files, if any).

FILES
   /usr/lbin/dhcpdb2conf
   The command itself.
   /etc/dhcpclient.data
   Current DHCP parameters received from a DHCP server by dhcpclient.
   /etc/resolv.conf.[dhcp]
   /etc/rc.config.d/netconf.[dhcp]
   /etc/rc.config.d/namesvrs.[dhcp]
   /etc/rc.config.d/netdaemons.[dhcp]
   System configuration files that can be modified by dhcpdb2conf.

SEE ALSO
   auto_parms(1M), set_parms(1M), geocustoms(1M), dhcpclient(1M).
NAME
dhcptools - command line tool for DHCP elements of bootpd

SYNOPSIS
dhcptools -d
   dhcptools -h fip=first_IP_address no=number_of_entries_to_generate sn=subnet_mask
           hn=hostname_template [dn=domain_name]
   dhcptools -p ht=hardware_type ha=hardware_address sn=subnet_identifier [lt=lease_time]
    [rip=requested_IP_address]
   dhcptools -P ci=client_identifier sn=subnet_identifier [lt=lease_time]
    [rip=requested_IP_address]
   dhcptools -C cl=class_identifier sn=subnet_identifier [lt=lease_time]
    [rip=requested_IP_address]
   dhcptools -r ip=IP_address ht=hardware_type ha=hardware_address
   dhcptools -R ip=IP_address ci=client_identifier
   dhcptools -t [ct=count]
   dhcptools -v [bt=bootptabfile] [dt=dhcptabfile]

DESCRIPTION
   dhcptools is a command line tool that provides access to DHCP-related options for the bootpd server. The options provide control for dumping internal data structures, generating a hosts file, previewing client address assignment, reclaiming unused addresses, tracing packets, and validating configuration files.

Options
   dhcptools supports the following options:
      -d Dump internal bootpd data to output files. The dump output files are /tmp/dhcp.dump.bootptab, /tmp/dhcp.dump.dhcptab, and /tmp/dhcp.dump.other. The first file reports fixed address clients known to the currently active bootpd server. The second file reports bootpd global and group configuration. The third file reports miscellaneous bootpd internal data.
      -h Generate a hosts file in /etc/hosts format; see hosts(4). The output file is /tmp/dhcphosts. The file can be incorporated into a name database in advance of bootpd server activation so that the server can automatically allocate a host name along with an IP address to a DHCP client. For IP address allocation to DHCP clients, the bootpd server uses gethostbyaddr(3N) to find the host name associated with a particular IP address. Each host entry in dhcphosts contains an IP address followed by a host-name. The IP address of the first entry is first_IP_address. The hostname of the first entry is derived from the hostname_template. Each subsequent host entry contains a unique IP address and hostname derived from the first_IP_address, subnet_mask, and hostname_template. The wildcards permitted in the hostname_template are *#?. A * means to use a character selected sequentially from the range [a-z,0-9]. A # means to use a digit selected sequentially from the range [0-9]. A ? means to use a letter selected sequentially from the range [a-z]. A maximum of 3 wildcards can be specified. If a domain_name is specified, it will be appended to the hostname. The maximum number_of_entries_to_generate is 1000.
      -p Preview a client’s address assignment based on current conditions for the bootpd server. The output is written to stdout. The subnet-identifier tells bootpd the subnet for which the client is requesting an IP address. Optionally, the user may request a specific IP address and lease duration using the parameters lease-time and requested-IP-address. Use Internet address dot notation (see inet(3N) for the IP address and an integer number of seconds for the lease-time.
      -P Preview a client’s address assignment based on current conditions for the bootpd server. This option is the same as -p except that the client is identified by a unique client-identifier. See bootpd(1M).
      -C Preview a client’s address assignment based on current conditions of the bootpd server. This option is the same as -p except that the class identifier is used to identify the device
group from which the client is requesting an IP address. See bootpd(1M).

-\textbf{r}\ Reclaim a client's IP address for re-use by the bootpd server. This option is intended for limited use by the bootpd administrator to return an allocated but unused IP address to a DHCP allocation pool. The option may be useful to clear the bootpd database of old entries (e.g. for clients retired from service while holding an unexpired IP address lease). Do not reclaim an address that belongs to an active client. See bootpd(1M). The IP_address, hardware_address, and hardware_type can be obtained from the bootpd database file.

-\textbf{R}\ Reclaim a client's IP address for re-use by the bootpd server. This option is the same as -\textbf{r} except that the client is identified by its unique client_identifier. See bootpd(1M). The IP_address and matching client_identifier can be obtained from the bootpd database file.

-\textbf{t}\ Establish packet tracing for bootpd. This will trace the inbound and outbound BOOTP/DHCP packets for the local bootpd server. The output file is /tmp/dhcptrace. The packet trace count can be a value from 0 to 100. To query the current count, use dhcptools -t. To turn off packet tracing use dhcptools -t ct=0.

-\textbf{v}\ Validate bootpd configuration files. The default configuration files that will be validated are /etc/bootptab and /etc/dhcptab. When a bootptabfile or dhcptabfile is specified, the full pathname is required. The output file for validate is /tmp/dhcpvalidate.

Only one of the -\textbf{d}, -\textbf{h}, -\textbf{t}, -\textbf{p}, -\textbf{P}, -\textbf{r}, -\textbf{R}, or -\textbf{v} options is allowed per dhcptools command.

\textbf{RETURN VALUE}  

dhcptools returns zero upon successful completion or non-zero if the command failed, in which case an explanation is written to standard error.

\textbf{EXAMPLES}  

Dump the active bootpd server's internal data to the dump output files:

dhcptools -d

Generate a /tmp/dhcphosts file with 10 entries:

dhcptools -h fip=192.11.22.0 no=10 sm=255.255.255.0 hn=workstation#?

Query the active bootpd daemon for the the current packet trace count:

dhcptools -t

Set the count to 10 packets:

dhcptools -t ct=10

Preview two clients' address assignments by hardware address:

dhcptools -p ht=1 ha=080009000001 sm=192.11.22.0 lt=infinite

dhcptools -p ht=1 ha=080009000002 sm=192.11.22.0 lt=600 rip=192.11.22.105

To preview a client's address assignment by client identifier, a unique client identifier value is needed. This information can be obtained for actual DHCP clients (provided they support a client identifier) from the manufacturer's documentation. See bootpd(1M) for more information about the client identifier. Assuming that \texttt{serial_number_12345678} is a valid client identifier, the preview command is:

\begin{verbatim}
dhcptools -P ci="serial_number_12345678" sn=192.11.22.0
\end{verbatim}

To reclaim an IP address by hardware address:

\begin{verbatim}
dhcptools -r ip=192.11.22.149 ht=1 ha=080009000006
\end{verbatim}

The parameter values were obtained from this sample entry in the dhcpdb file:

C 192.11.22.0: 192.11.22.149 00 1 080009000006 FFFFFFFF 00

To reclaim an IP address by client identifier (see earlier example of preview by client identifier):

\begin{verbatim}
dhcptools -R ip=192.11.22.110 ci="serial_number_12345678"
\end{verbatim}

To validate a bootptab and dhcptab file:
dhcptools -v bt=/home/mydir/bootptab dt=/home/mydir/dhcptab

WARNINGS
The dhcptools operations of dump, packet trace, preview, and reclaim depend on communication with the local bootpd server. If the server is not running, you may encounter an error.

AUTHOR
dhcptools was developed by HP.

FILES
/tmp/dhcphosts hostgen output file in /etc/hosts format
/tmp/dhcptrace packet trace output file
/tmp/dhcpcvalidate validate output file
/tmp/libdhcp.s1 library file
/tmp/dhcp.dump.bootptab dump output file
/tmp/dhcp.dump.dhcptab dump output file
/tmp/dhcp.dump.other dump output file
/etc/bootptab default bootptab file for validate
/etc/dhcptab default dhcptab file for validate
/tmp/dhcpfifo.root FIFO file for dhcptools to bootpd(1M) communication
/tmp/dhcpfifo.any FIFO file for dhcptools to bootpd(1M) communication
/tmp/dhcpfifo FIFO file for bootpd(1M) to dhcptools communication

SEE ALSO
bootpd(1M), bootpquery(1M);
DARPA Internet Request For Comments RFC1541, RFC1542, RFC1533, RFC1534, Assigned Numbers
NAME
dhcpv6clientd - DHCPv6 client daemon

SYNOPSIS
/usr/sbin/dhcpv6clientd [-d config_options] [-l]

DESCRIPTION
dhcpv6clientd obtains the configuration parameters from the Dynamic Host Configuration Protocol (DHCPv6) server to configure the host. See dhcpd(1M).

Options
dhcpv6clientd supports the following options:
- d config_options
  Specify the list of parameters that the client daemon must request from the server daemon. The list of configuration parameters that the client can request are listed below:
  - dns_sa Obtain the DNS server address
  - dns_sx Obtain the DNS suffix
  - ntp_sa Obtain the NTP server address
  - nis_dn Obtain the NIS domain name
  - nis_sa Obtain the NIS server address
  - nispcl_dn Obtain the NIS+ client domain address
  - nisp_sa Obtain the NIS+ server address
  - slp_da Obtain the SLP Directory Agent (DA) address and it's scope
  - slp_ss Obtain the SLP service scope
  - tz Obtain the timezone information

- l Enable the logging of all messages by dhcpv6clientd to the /var/adm/syslog/syslog.log file, including the information and warning types.

Configuration
The /etc/rc.config.d/netconf-ipv6 file is used for the configuration of the interfaces. To invoke dhcpv6clientd using the dhcpv6config script, set the DHCPV6_ENABLE [index] variable to a non-zero value in the /etc/rc.config.d/netconf-ipv6 file. The dhcpv6config script invokes dhcpv6clientd and dhcpv6client_ui during bootup to obtain IP addresses. dhcpv6clientd can also be invoked at the command prompt.

Once dhcpv6clientd has obtained the configuration parameters from the server, it stores them in a configuration file named /etc/dhcpv6client.data. The configuration parameters are valid for a fixed time, the lease time.

The format of the configuration information in /etc/dhcpv6client.data file is shown below:

<table>
<thead>
<tr>
<th>code</th>
<th>length</th>
<th>data</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>&lt;length&gt;</td>
<td>End of the record (EOR)</td>
</tr>
<tr>
<td>01</td>
<td>&lt;length&gt;</td>
<td>Interface name</td>
</tr>
<tr>
<td>02</td>
<td>&lt;length&gt;</td>
<td>Address of the DHCP server</td>
</tr>
<tr>
<td>03</td>
<td>&lt;length&gt;</td>
<td>UUID</td>
</tr>
<tr>
<td>04</td>
<td>&lt;length&gt;</td>
<td>IA</td>
</tr>
<tr>
<td>05</td>
<td>&lt;length&gt;</td>
<td>T1</td>
</tr>
<tr>
<td>06</td>
<td>&lt;length&gt;</td>
<td>T2</td>
</tr>
<tr>
<td>07</td>
<td>&lt;length&gt;</td>
<td>Group Multicast Address</td>
</tr>
<tr>
<td>08</td>
<td>&lt;length&gt;</td>
<td>DNS Server Address</td>
</tr>
<tr>
<td>09</td>
<td>&lt;length&gt;</td>
<td>DNS Suffix</td>
</tr>
<tr>
<td>10</td>
<td>&lt;length&gt;</td>
<td>NTP Server Address</td>
</tr>
<tr>
<td>11</td>
<td>&lt;length&gt;</td>
<td>NIS Domain Name</td>
</tr>
<tr>
<td>12</td>
<td>&lt;length&gt;</td>
<td>NIS Server Address</td>
</tr>
<tr>
<td>13</td>
<td>&lt;length&gt;</td>
<td>NIS+ Client Domain Name</td>
</tr>
<tr>
<td>14</td>
<td>&lt;length&gt;</td>
<td>NIS+ Server Address</td>
</tr>
</tbody>
</table>
dhcv6clientd(1M)

15 <length> SLP DA address and its scope
16 <length> SLP Service Scope
17 <length> Time Zone
18 <length> Retransmission Variables

The <length> field denotes the length of the data field following it.

All these lines collectively indicate one record. The configuration file will have a series of record. See the EXAMPLE section below for a sample configuration file.

Since this file contains all the configuration information, it can be used to identify the server from which the configuration parameters were obtained.

When dhcv6clientd is invoked, it will first try to read the configuration file, /etc/dhcv6client.data. However, the configuration file
1. may have been deleted or is not available for reading;
2. may be corrupted or contain irrelevant data for a particular interface;
3. may not contain the configuration information for a particular interface.

In those cases, dhcv6clientd needs to obtain a new set of addresses from the server. Thus the client will form a SOLICIT message and send it to the pre-determined ALL DHCP Agents address. In response to this, the client will receive the ADVERTISE message from DHCP servers. The client will select a DHCP server based on the DHCP server preference value.

EXAMPLES

dhcv6clientd treats the NIS server and DNS server addresses as default configuration parameters.

dhcv6clientd -d nis_sa dns_sa

The following is an example of a /etc/dhcv6client.data file:

01 4 lan4
02 16 3ffe:1234
03 8 22:c2:00:10:83:b9:53:3c
04 29 4 3ffe::428d 64 1005134643 1005134663
05 4 1005134548
06 4 1005134551
07 0
08 32 3ffe::12 3ffe::19
09 6 hp.com
10 16 3ffe::19
11 12 india.hp.com
12 32 3ffe::13 3ffe::19
13 12 india.hp.com
14 32 3ffe::15 3ffe::19
15 22 16 3ffe::15 (proxy=rio) 0
15 0
16 0
17 0
18 0
00 0

AUTHOR

dhcv6clientd was developed by Hewlett-Packard.

FILES

/etc/dhcv6tab
/etc/rc.config.d/netconf-ipv6
/etc/dhcv6client.data Client configuration database file.

SEE ALSO

dhcv6d(1M), syslog(3C).
NAME
dhcpv6d - Dynamic Host Configuration Protocol Server daemon for IPv6

SYNOPSIS
/usr/sbin/dhcpv6d [-d] [-c config_file]
/usr/sbin/dhcpv6d -k
/usr/sbin/dhcpv6d -r

DESCRIPTION
The dhcpv6d server daemon is the IPv6 version of the bootpd daemon. This version supports the following features:

· Dynamic renumbering
· Relay preconfiguration with server addresses, or use of multicast addresses
· Multiple IP addresses for an interface

The dhcpv6d daemon is run as a stand-alone daemon and not spawned by inetd.

Options
-d Toggle debugging on or off.
-c config_file Specify an alternate configuration file for the server. The default configuration file is /etc/dhcpv6tab.
-r Re-read the configuration file and trigger the server to send a reconfig-init message to the clients when the configuration file has any new or updated information.
-k Kill the server gracefully.

Configuration
When an unconfigured host needs to be configured, it checks for router advertisements. From the router advertisement, depending on the prefix obtained, the host will generate either a site-local or a global address.

If there are no routers on the link on which the host is located, it uses dhcpv6client to configure itself. The client forms a SOLICIT message and sends it to the pre-determined ALL DHCP agents address. In response to this, the client will receive ADVERTISE messages from a few DHCP servers. The client will select the server based on the preference values in advertise messages. Refer to the "Configuraion File" section for information on how to set the DHCP's server preference value.

In response to the ADVERTISE message, the client will send a REQUEST message to the server either directly if the server is on the same link as that of the client or through the on-link relay. The server responds with REPLY message containing the requested configuration parameters.

The DHCP server assigns addresses to the clients using "Identity Association (IA)", which is a collection of addresses assigned to a client. The client uses the addresses in the IA to configure its interfaces.

If the client determines that the address allocated to it by the server is already in use, it will send a RELEASE message to the server.

To extend the lifetime of the addresses assigned to an IA, the client sends a RENEW message to the server, which contains the IA and the associated addresses. The server will determine the new lifetimes for the addresses on the IA based on the server’s administrative configuration. The server will respond to this message with a REPLY message. The server controls the time at which the client contacts the server to extend the lifetimes on assigned addresses through the T1 and T2 parameters that are assigned to the IA. The client initiates a RENEW message at time T1.

If the server to which the RENEW message is sent does not respond, then the client will increase the timeout and wait for a response. The client continues sending the message until the expiration of T2. But at time T2 for the IA, the client initiates a REBIND message. This message is sent to all the servers in the domain. The server will respond to this message and will renew the lifetime of the assigned addresses. If the client does not receive a response to its REBIND message, it tries resending the message until the expiration of lifetimes of all the addresses in IA. The client will choose one of the following alternatives:

· Some addresses in the IA may have lifetimes that extend beyond the lease of the IA. The client may choose to continue to use those addresses.
The client may have other addresses in other IAs, so the client can discard the expired IA and use the addresses in the other IAs.

The client sends a RELEASE message to release an IA. This message is sent to the server that assigned the addresses to the client initially and the client will wait until the timeout value expires. If the server cannot be reached after 5 attempts, the client will abandon this release attempt. These addresses will be claimed by the servers when the lifetimes of the addresses expire.

The administrator will trigger `dhcv6d` to unicast a RECONFIGURE-INIT message to all those clients who have their addresses assigned by this server when the server configuration changes are vital. This message is a trigger which will cause the clients to initiate a standard REQUEST message to the server to obtain the new or updated configuration parameters.

**Configuration File**

Upon startup, `dhcv6d` reads the configuration file, `/etc/dhcpv6tab`, or a file specified with the `-c` option to build its internal database and then listens for SOLICIT/REQUEST messages from the client. If configuration parameters are added, deleted, or modified, then their entries in the internal database are updated when `dhcv6d` is invoked with the `-r` option.

The configuration file contains the following sections:

1. DHCPv6 client Default settings
2. DHCPv6 pool group settings
3. DHCPv6 relay settings

Blank lines and lines beginning with "#" in the configuration file are ignored. Entries are separated from one another by a semicolon. Multiple entries of a group can be extended over multiple lines if the lines end with a backslash \. A final semicolon ; followed by a new line indicates the end of a group.

IP addresses are specified in the standard IPv6 notation (colon notation) and can use hexadecimal numbers.

The tags for the DHCP_CLIENT_DEFAULT_SETTINGS are as listed below. These tags are applicable to all the addresses that the server assigns to the DHCPv6 client.

```
client-settings-name=client-settings-name-in-string-format;
POSIX-time-zone=POSIX-time-zone-string;
DNS-server-address=List-of-IP-addresses-in-IPv6-Address-Format;
DNS-suffix=string-format;
NIS-domain-name=string-format;
NIS-server-address=List-of-IP-addresses-in-IPv6-Address-Format;
NIS+client-domain-name=string-format;
NIS+server-address=List-of-IP-addresses-in-IPv6-Address-Format;
NTP-server-address=List-of-IP-addresses-in-IPv6-Address-Format;
SLP-DA-address=list-of-DA-addresses-followed-by-their-scope-list;
```

This tag specifies the name of the group default settings.

This tag specifies the POSIX time zone of the client. This should be as per the IEEE 1003.1 format.

This tag specifies the list of DNS nameservers available to the client. Multiple nameserver addresses are separated by white spaces.

This tag specifies the default domain name suffix that the client should use while resolving hostnames via DNS.

This tag specifies the name of the client’s NIS domain. The NIS-domain-name is formatted as a character string consisting of characters from the NVT-ASCII character set.

This tag specifies the list of NIS servers available to the client. Servers SHOULD be listed in the order of their preference.

This tag specifies the name of the client’s NIS+ domain. The domain is formatted as a character string consisting of characters from the NVT-ASCII character set.

This tag specifies the list of NIS+ servers available to the client.

This tag specifies the list of NTP servers available to the client.

This tag specifies the list of SLP Directory agents available to the client. The
scope list should be as per the format specified in RFC 2165.

**SLP-scope-list**=scope-list;

This tag specifies the scope list. The scope list should be as per the format specified in RFC 2165.

The message transaction configuration values are as listed below:

**min-sol-delay**=seconds;

This tag specifies the minimum allowable delay (in seconds) before the first SOLICIT message is sent.

**max-sol-delay**=seconds;

This tag specifies the maximum allowable delay (in seconds) before the first SOLICIT message is sent.

**adv-msg-timeout**=milliseconds;

This tag specifies the time (in milliseconds), for which the client will wait for the ADVERTISE message.

**adv-msg-max**=seconds;

This tag specifies the maximum advertise message timeout value in seconds.

**sol-max-attempts**=value;

This tag specifies the maximum number of times the SOLICIT message will be (re)sent to the server.

**rep-msg-timeout**=milliseconds;

This tag specifies the time by which the REQUEST message times out.

**req-msg-attempts**=value;

This tag specifies the maximum number of attempts the client makes to deliver the REQUEST message to the server to get a response.

**rel-msg-attempts**=value;

This tag specifies the maximum number of attempts the client makes to deliver the RELEASE message to the server to get a response.

**rec-rep-min**=value;

This tag specifies the minimum pause interval (in seconds) before the REQUEST message is sent in response to the RECONFIGURE-INIT message.

**rec-rep-max**=value;

This tag specifies the maximum pause interval (in seconds) before the REQUEST message is sent in response to the RECONFIGURE-INIT message.

Note: Currently, the mechanism of transferring these parameters to the client is not defined in RFCs. This may be made possible in the future releases.

The tags applicable for the individual DHCPv6 pool groups are as listed below:

**DHCP_POOL_GROUP**;

This tag specifies the individual pool groups.

**pool-group-name**=name-of-the-pool-group;

This tag specifies the name of the pool group as an ASCII string.

**default-settings**=default-settings-name;

This tag specifies the default settings for the pool group.

**subnet-prefix**=IPv6-hex-address-format;

This tag specifies the subnet-prefix of the network that the server manages. Note that you need to specify the complete address of the subnet.

**prefix-length**=unsigned-integer;

This tag specifies the length of the subnet-prefix.

**address-pool**=start_addr1 - end_addr1 start_addr2 - end_addr2...;

This tag specifies the list of start and end address, which constitute the address pool.

**reserved-address-list**=start_addr1 - end_addr1 start_addr2 - end_addr2...;

This tag specifies the start and end addresses of the list of addresses, which constitute the reserved addresses of the address pool.
reserved-addresses=lsit-of-IPv6-addresses;
This tag specifies the list of reserved IPv6 addresses separated by a white space.

preference=unsigned-integer;
This tag specifies the server preference to manage the subnet. This value can range between 0 and 255.

T1=time-value-in-seconds;
This variable is used by the server to control the time at which the client contacts the server to extend the lifetimes on assigned addresses. On expiration of T1, the client sends a RENEW message to the server. Default value is 250 seconds.

T2=time-value-in-seconds;
This variable is used by the server to control the time at which the client contacts the server to extend the lifetimes on assigned addresses. On expiration of T2, the client sends a REBIND message to the server. Default value is 437 seconds.

preferred-life-time=time-value-in-seconds;
This tag specifies the preferred life time for the IP addresses in seconds. Default is 500 seconds.

valid-life-time=time-value-in-seconds;
This tag specifies the valid life time for the addresses in seconds. Default is 600 seconds.

reconf-grace-period=value;
This tag specifies the grace period (in seconds) until when the old configuration parameters remain valid. Default is 300 seconds.

Note: Values for the above listed tags must satisfy the following condition:
reconf-grace-period < T1 < T2 < preferred-life-time < valid-life-time
If the above condition fails, the server takes the default values.

The DHCPv6 relay-specific tags are as listed below:

DHCP_RELAY_SETTINGS;
This tag indicates the start of the DHCP relay settings.

pool-group-name=name-of-the-pool-group;
This tag specifies the name of the pool group as an ASCII string.

subnet-prefix=IPv6-hex-address-format;
This tag specifies the IPv6 subnet prefix in hex. This should be a complete address of the subnet.

prefix-length=unsigned-integer;
This tag specifies the length of the subnet prefix.

dest-dhcp-server-address=IPv6-hex-address-format;
This tag specifies the address of the DHCPv6 server to which the relay forwards the client messages.

EXAMPLES
A sample /etc/dhcpv6tab file is shown below:

DHCP_CLIENT_DEFAULT_SETTINGS;
client-settings-name=TESTING-GROUP;\nPOSIX-time-zone=EST5EDT4,116/02:00:00,298/02:00:00;\nDNS-server-address=3ffe::200:fe23:b580 3ffe::200:fe23:b540;\nDNS-suffix=india.hp.com;\nNIS-domain-name=india.hp.com;\nNIS-server-address=3ffe::200:fe23:b123 3ffe::200:fe23:b034;\nNIS+client-domain-name=cup.hp.com;\nNIS+server-address=3ffe:200:fe23::2341;\nNTP-server-address=3ffe::200:fe23::2341;\nSLP-DA-address=3ffe::200:fe23:b123 (netman=mgmt),(proxystuff=labs);\n
DHCP_POOL_GROUP;\npool-group-name=TESTING-POOL;\n
default-settings=TESTING-GROUP;
subnet-prefix=3ffe:305:1002:1::;
prefix-length=64;
T1=56;
T2=90;
preferred-life-time=300;
valid-life-time=500;
address-pool = 3ffe:305:1002:1:200:c0ff:fe23:b560 -
3ffe:305:1002:1:200:c0ff:fe23:b590;
preference=255;
NTP-server-address=3ffe::200:fe23:b102;

DHCP_RELAY_SETTINGS;
pool-group-name=RED-POOL;
subnet-prefix=5ffe:305:1002:1::;
prefix-len=64;
dest-dhcp-server-address=5ffe:305:1002:1:2345:203:1ff3:3048;

AUTHOR
dhcpv6d was developed by Hewlett-Packard.

FILES
/etc/dhcpv6tab This is the dhcpv6 server configuration file.

SEE ALSO
dhcpv6client_ui(1), dhcpv6clientd(1M), dhcpv6db2conf(1M).
NAME
dhcpsv6db2conf - DHCPv6 client database converter

SYNOPSIS
/usr/bin/dhcpsv6db2conf [-a | -c | -p| [-d] | [-i] | [-n] | [-t] [lan_interfaces]

DESCRIPTION
dhcpsv6db2conf provides a means of translating the DHCPv6 client database,
/etc/dhcpsv6client.data, into a set of standard configuration file variables. The client database
consists of a series of records. Each record represents a unique lan interface and a list of attributes, which
correspond to that interface. The attributes are configuration parameters like: host’s IP address, NTP
server address, address of the DHCPv6 server, NIS domain name etc for that interface. With
dhcpsv6db2conf, you can list the contents of the database to the screen, create a set of configuration
staging files, or execute direct edits on existing configuration files using the values contained in the client
database.

Options
dhcpsv6db2conf allows you to specify a list of lan interfaces on the command line (e.g. lan0 lan1...)
whose data needs to be translated to a readable format. If no lan interface is specified, dhcpsv6db2conf
will process the entries of all interfaces which are UP. The attributes can be selected for processing by
specifying a combination of one or more filter flags on the command line. If no filter flag is specified, all the
attributes for that lan interface will be processed.

The following options are supported:

- a Using the results of the specified filter, directly apply the variable definitions to the existing
configuration files (for example, /etc/rc.config.d/netconf-ipv6).
- c Create a set of staging files using the results of the selected filter(s). Each variable pro-
cessed will be applied to its corresponding staging configuration file. Specifically,
dhcpsv6db2conf will generate a copy of the existing configuration file. For example,
/etc/rc.config.d/netconf-ipv6 will be copied to
/etc/rc.config.d/netconf-ipv6.dhcpv6. Once this staging file has been
created, the variable that is being processed will be applied to this newly created file.
- d Process the DNS variable set: [domain, nameserver ].
- i Process the INTERFACE variable set: [IPV6_SECONDARY_INTERFACE_NAME, 
IPV6_ADDRESS, IPV6_PREFIXLEN, DHCPV6_ENABLE].
- n Process the NIS variable set: [NIS_DOMAIN, YPSET_ADDR].
- p Print results to the screen (stdout), this is the default action if neither -c nor -a are specified.
- t Process the NTPDATE_SERVER variable.

lan_interfaces lan interfaces such as lan0, lan1, etc. whose data needs to be translated to a readable
format. If not specified, dhcpsv6db2conf will process the entries of all interfaces which are UP.

Configuration Files and Variable Names
The following files and variables can be processed:

/etc/rc.config.d/netconf-ipv6
  IPV6_SECONDARY_INTERFACE_NAME
  IPV6_ADDRESS
  IPV6_PREFIXLEN
  DHCPV6_ENABLE

/etc/rc.config.d/namesvrs
  NIS_DOMAIN
  YPSET_ADDR

/etc/rc.config.d/netdaemons
  NTPDATE_SERVER
/etc/resolv.conf
  domain
  nameserver

EXAMPLES
  To list the entire contents of the DHCP client database type:
    dhcpv6db2conf
  To list only the INTERFACE variable set for lan0 type:
    dhcpv6db2conf -i lan0

WARNINGS
  The NIS and NTP parameters are not IPv6-compatible. Hence do not use dhcpv6db2conf to update them.

AUTHOR
  dhcpv6db2conf was developed by Hewlett-Packard.

FILES
  /etc/dhcpv6client.data  Client configuration database

SEE ALSO
  dhcpv6clientd(1M), dhcpv6d(1M).
NAME
dig - domain information groper

SYNOPSIS

dig [@server] [options] domain [query-type] [query-class] [query-options]
dig [global-server] [global-d-options] domain [@server] [options] [q-options] [q-type] [q-class]

DESCRIPTION
dig (domain information groper) is a flexible tool for interrogating Domain Name System (DNS) servers. It performs DNS lookups and displays the answers that are returned from the name server(s) that were queried. Most DNS administrators use dig to troubleshoot DNS problems because of its flexibility, ease of use, and clarity of output. The dig command has two modes: simple command-line mode for single or multiple queries and batch mode for reading lookup requests from a file.

Arguments
dig accepts the following arguments:

- @server Specifies the DNS server that is queried in each query. If a specific name server is not provided, dig will try each of the servers listed in /etc/resolv.conf.
- @global-server Specifies the name of the server that is used in multiple queries to provide a single server for all the queries.
- domain Specifies the domain name to look up.
- query-type Specifies the resource record types for DNS queries and responses. The textual representation is used in master files. The binary representation is used in DNS queries and responses. The resource record types are:
  - a a host address (dotted quad). This is the default value for query-type.
  - AAAA resource record type for IPv6 queries.
  - any request data of any type for a name.
  - axfr a request for a transfer of an entire zone.
  - hinfo host information.
  - mx a mail exchange.
  - ns an authoritative name server.
  - soa marks the start of a zone of authority.
  - txt text strings.
  Ensure that you use ixfr=version for type ixfr. ixfr transfers only the incremental/changed data to the slave servers when the data in the master server changes.
- query-class Classes are the fields that appear in resource records. The values defined for a class are: IN (Internet), CS (CSNET), CH (CHAOS), and HS (Hesiod). The default value for query-class is IN.
- query-option Query options affect the way in which lookups are made and how the results are displayed. Each query option is identified by a keyword preceded by a +[no] See the "Query Options" subsection below for details.
- global-d-opt Global domain query options control the lookup and display of results for multiple queries and affect all queries. Note that query options set globally can be overwritten by query options set for each individual query.
- options
  - -b This option is used to set the source IP address of the query to address. This must be a valid address on one of the host's network interfaces.
-f This option is used to perform batch processing. It allows to group queries into one file and to pass this file to dig for processing. Example: dig -f /home/bind/some-file, where some-file contains all the queries that need to be processed as a group.

-k This option is used to sign the DNS queries sent by dig and their responses using transaction signatures (TSIG).

-p This option can be used when you want to specify a different port for dig to contact the name-server for its queries.

-x This option allows queries using an IP address instead of a domain name. This option cannot be used with IPv6 addresses.

-y This option is used to specify the TSIG key on the command line.

-t & -c
The -t (type) and -c (class) option. Equivalent to query-type and query-class.

-h Displays usage information of the dig command.

A typical dig command is:

dig @server domain query-type

where @server is the name or IP address of the name server, which is to be queried. An IPv4 address can be provided in a dotted-decimal notation, xxx.xxx.xxx. dig resolves the host name before querying that name server. If no argument is provided, dig consults /etc/resolv.conf and queries the name servers listed there. The reply from the name server that responds to the query is displayed.

domain is the name of the resource record, which is to be looked up.

query-type indicates the required query type ie., ANY, A, MX, SIG etc. It can be any valid query type. The dig command will perform a lookup for an A record if no query-type argument is specified.

Query Options
dig uses a number of query options to affect lookups and to affect the results that are displayed. Some options set or reset flag bits in the query header, some options determine which sections of the answer get displayed, and other options determine the timeout and retry strategies.

Each query option is identified by a keyword preceded by +[no] which causes an option to be set or reset or to negate the meaning of that keyword. Other keywords assign values to options like the timeout interval. They have the form +keyword=value. The query options are:

+tcp
Use [or do not use] TCP when querying name servers. The default behavior is to use UDP unless an AXFR or IXFR query is requested, in which case a TCP connection is used.

+vc Use [or do not use] virtual circuit when querying name servers. This alternate syntax to +tcp is provided for backwards compatibility.

+ignore Ignore [or do not ignore] truncation in UDP responses instead of retrying with TCP. By default, TCP retries are performed.

+domain=somename Set the default domain to somename as it is specified in a directive in the /etc/resolv.conf file.

+search Use [or do not use] the search list in /etc/resolv.conf (if any). The search list is not used by default.

+defname Use [or do not use] the default domain name, if any, in the /etc/resolv.conf file while making queries. By default, this name is not appended to name while making queries.

+a
Authenticate [or do not authenticate] the client when it queries a server. If this option is set, whenever a client tries querying a server, the client will be subjected to an authentication check to make sure that the client has sufficient permissions to query the server.
**dig(1M)**

---

**+[no]adflag**
Set [or do not set] the AD (authenticate data) bit in the query. The AD bit currently has a standard meaning only in responses and not in queries. The ability to set the bit in the query is provided for completeness.

**+[no]cdflag**
Set [or do not set] the CD (checking disabled) bit in the query. This requests the server not to perform DNSSEC validation of responses.

**+[no]recursive**
Set [or do not set] the RD (recursion desired) bit in the query. This bit is set by default, which means `dig` normally sends recursive queries. Recursion is automatically disabled when the `+nssearch` or `+trace` query options are used.

**+[no]nssearch**
Attempt [or do not attempt] to find the authoritative name servers for the zone containing the name being looked up and display the SOA record that each name server has for the zone.

**+[no]trace**
Trace [or do not trace] the delegation path from the root name servers for the name being looked up. Tracing is disabled by default. When tracing is enabled, `dig` makes iterative queries to resolve the name that is being looked up. It will follow referrals from the root servers, showing the answer from each server that was used to resolve the lookup.

**+[no]cmd**
+[no]comment `dig` and the query options that have been applied. The comment is included in the answer section by default.

**+[no]short**
Display [or do not display] a short answer. The query results can be displayed in two forms: Complete and Short answers. In the short form, only the result will be displayed whereas in the complete form, additional information (like info about other servers that might answer your query) is also included. By default, the answer is printed in a verbose form.

**+[no]identify**
Show [or do not show] the IP address and port number that supplied the answer when the `+short` option is enabled. If short form answers are requested, source address and port number of the server that provided the answer are not shown by default.

**+[no]comments**
Display [or do not display] comment lines in the output. By default, the comments are printed.

**+[no]stats**
Print [or do not print] statistics such as the size of the reply when the query was made. By default, the query characteristics are printed.

**+[no]qr**
Print [or do not print] the query before actually sending the query. By default, the query is not printed.

**+[no]question**
Print [or do not print] the question section of a query when an answer is returned. By default, the question section is printed as a comment.

**+[no]answer**
Display [or do not display] the answer section of a reply. By default, the answer section is printed.

**+[no]authority**
Display [or do not display] the authority section of a reply. By default, the authority section is displayed.

**+[no]additional**
Display [or do not display] the additional section of a reply. By default, the additional section is displayed.

**+[no]multiline**
Print records like the SOA records in a verbose multi-line format with human-readable comments. The default is to print each record on a single line, thereby facilitating machine parsing of the `dig` output.
dig(1M)

+|no|all
Set or clear all display flags.

+time=\T
Set the timeout for a query to \T seconds. The default timeout is 5 seconds, and 1 second is the minimum value to which \T can be set. Even if you try setting \T to a value less than 1, \T will be set to 1 second.

+tries=\A
This option sets the number of times to retry UDP queries to server to \A instead of the default, 3. If \A is less than or equal to zero, then the number of retries is set to 1.

+ndots=\D
Set the number of dots that appear in hostname to \D. The default value is to use either the ndots statement in /etc/resolv.conf or 1 if no ndots statement is present. Names with fewer dots are interpreted as relative names and will be searched for, in the domains listed in the search or the domain directive in the /etc/resolv.conf file.

+bufsize=\B
Set the UDP message buffer size advertised using EDNS0 to \B bytes. The maximum and minimum sizes of this buffer are 65535 and 0 respectively. If the \B size is specified outside of this range, then the size is rounded up or down appropriately.

Multiple Queries
dig allows multiple queries on the command line (in addition to supporting the -f batch file option). Each of those queries can be supplied with its own set of options, query class, query type and query options.

A global set of query options, which should be applied to all queries can also be supplied via global-d-options.

These global query options must precede the first set of domain, class, type, options, and query options supplied on the command line. Any global query options can be overridden by a query-specific set of query options for each individual query.

For example:

```
dig +qr www.bind.org any -x 127.0.0.1 bind.org ns +noqr
```

shows how dig can be used from the command line to make three lookups:

```
www.bind.org any
An ANY query for domain name www.bind.org.

-x 127.0.0.1
A reverse lookup of 127.0.0.1

bind.org ns +noqr
A name server lookup for domain bind.org, suppressing the query display for this query only (+noqr).
```

EXAMPLES
1. To look up information about domain a.example.com using DNS-Server 10.53.0.2 asking for host address a records:

```
/usr/bin/dig +tcp +noadd +nosea +nostat +noquest +nocomd -p 5300 a.example.com @10.53.0.2 a
```

2. To query a.example.com using DNS-Server 10.53.0.2 without authentication, asking for a records:

```
/usr/bin/dig +tcp +noadd +nosea +nostat +noauth +nocomd -p 5300 a.example.com @10.53.0.2 a
```

3. To request a transfer:

```
/usr/bin/dig +tcp +noadd +nosea +nostat +noquest +nocomm +nocomd example.com @10.53.0.2 axfr -p 5300
```

4. To request a transfer with Transaction Signature (TSIG):

```
/usr/bin/dig +tcp +noadd +nosea +nostat +noquest +nocomm +nocomd tsigzone.com @10.53.0.3 axfr -y tsigzone.com:1234abcd8765 -p 5300
```
where 1234abcd8765 is the key.

To secure server-to-server communication BINDv9 primarily uses TSIG for zone transfer, notify, and recursive query messages. TSIG is very useful for dynamic updates.

SEE ALSO
dnsec-keygen(1), dnsec-makekeyset(1), dnsec-signkey(1), dnsec-signzone(1), host(1), nsupdate(1), hosts_to_named(1M), lwresd(1M), named(1M), gethostent(3N), hostname(5).
NAME
diskinfo - describe characteristics of a disk device

SYNOPSIS
/usr/sbin/diskinfo [-b | -v] character_devicefile

DESCRIPTION
The diskinfo command determines whether the character special file named by character_devicefile is associated with a SCSI or floppy disk drive. If so, diskinfo summarizes the disk's characteristics.

The diskinfo command displays information about the following characteristics of disk drives:

- **Vendor name**
- **Manufacturer of the drive (SCSI only)**
- **Product ID**
- **Product identification number or ASCII name**
- **Type**
- **floppy or SCSI classification for the device**
- **Disk**
- **Size of disk specified in bytes**
- **Sector**
- **Specified as bytes per sector**

Both the size of disk and bytes per sector represent formatted media.

Options
The diskinfo command recognizes the following options:

- **-b**
  - Return the size of the disk in 1024-byte sectors.

- **-v**
  - Display a verbose summary of all of the information available from the device. For floppy drives, this option has no effect.

  SCSI disk devices return the following:
  - Vendor and product ID
  - Device type
  - Size (in bytes and in logical blocks)
  - Bytes per sector
  - Revision level
  - SCSI conformance level data

WARNINGS
As of release 10.20 of HP-UX, certain IDE devices, CD-ROMs in particular, will respond to diskinfo inquiries as if they were SCSI devices. Therefore, the text "SCSI describe" in the output of the diskinfo command does not definitively mean that the disk is in fact a SCSI drive (especially in the case of CD-ROMs). Use ioscan(1M), ioscan -fn, and check which type of INTERFACE node, SCSI or IDE, the device's hardware path lies beneath, in order to definitively determine a drive's interface.

DEPENDENCIES
General
The diskinfo command supports floppy and HP SCSI disk devices.

SCSI Devices
The SCSI specification provides for a wide variety of device-dependent formats. For non-HP devices, diskinfo may be unable to interpret all of the data returned by the device. Refer to the drive operating manual accompanying the unit for more information.

AUTHOR
diskinfo was developed by HP.

SEE ALSO
lsdev(1M), ioscan(1M), disktab(4), disk(7).
NAME
disksecn - calculate default disk section sizes

SYNOPSIS
disksecn [-p | -d] [-b block_size] [-n disk_name]

DESCRIPTION
disksecn is used to calculate the disk section sizes based on the Berkeley disk partitioning method.
disksecn recognizes the following options:
   -p Produce tables suitable for inclusion in the device driver.
   -d Produce tables suitable for generating the disk description file /etc/disktab.
   -b block_size When generating the above tables, use a sector size of block_size bytes, where block_size can be 256, 512, 1024, or 2048. Defaults to DEV_BSIZE (defined in <sys/param.h>) if not specified.
   -n disk_name Specifies the disk name to be used in calculating sector sizes; for example, hp7912 or hp7945. If an unknown disk name is specified, disksecn prompts the user for the necessary disk information.

If neither -p nor -d table selection switches are specified a default table of the section sizes and range of cylinders used is output.

Disk section sizes are based on the total amount of space on the disk as given in the table below (all values are supplied in units of 256-byte sectors). If the disk is smaller than approximately 44 Mbytes, disksecn aborts and returns the message disk too small, calculate by hand.

<table>
<thead>
<tr>
<th>Section</th>
<th>44-56MB</th>
<th>57-106MB</th>
<th>107-332MB</th>
<th>333+MB</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>97120</td>
<td>97120</td>
<td>97120</td>
<td>97120</td>
</tr>
<tr>
<td>1</td>
<td>39064</td>
<td>39064</td>
<td>143808</td>
<td>194240</td>
</tr>
<tr>
<td>3</td>
<td>39064</td>
<td>39064</td>
<td>78128</td>
<td>117182</td>
</tr>
<tr>
<td>4</td>
<td>unused</td>
<td>48560</td>
<td>110096</td>
<td>428704</td>
</tr>
<tr>
<td>6</td>
<td>7992</td>
<td>7992</td>
<td>7992</td>
<td>7992</td>
</tr>
<tr>
<td>10</td>
<td>unused</td>
<td>unused</td>
<td>unused</td>
<td>516096</td>
</tr>
</tbody>
</table>

Note
It is important to note the difference between the block size passed into disksecn via the -b switch argument and the sector size the user is asked to input when an unknown disk name is passed to disksecn via the -n switch argument.

The block size is the sector size that disksecn assumes the disk to have when it prints the requested tables. All information printed in the tables is adjusted to reflect this assumed sector size (block size) passed in by the user. The sector size requested by disksecn when an unknown disk name is passed does not necessarily have to be the same as the assumed sector size (block size) passed in by the -b switch argument.

For example, a user wants to see the device driver tables for the disk named hp7945 with an assumed sector size (block size) of 256 bytes. The user has the following information about the hp7945 disk:

- Disk type = winchester
- Sector size = 512
- Number of sectors per track (512 byte sectors) = 16
- Number of tracks = 7
- Number of cylinders = 968
- Revolutions per minute = 3600

The user invokes disksecn by typing the following command:

disksecn -p -b 256 -n hp7945

Assuming that hp7945 is an unknown disk name, disksecn prompts the user for the necessary disk information. The user should input the information as shown above, reflecting a sector size of 512 bytes. All the information will be adjusted within disksecn to reflect the assumed sector size (block size) of 256 bytes, passed as the argument of the -b switch, before the requested device driver table is output.
This adjustment also takes place when the disk name is known and an assumed sector size (block size) is passed in as the argument of the -b switch which is not DEV_BSIZE bytes, the assumed sector size (block size) used to create the /etc/disktab file.

RETURN VALUE

disksecn returns the following values:

0 Successful completion.
1 Usage error.
2 User did not input parameters for an unknown disk.
3 Disk too small or an invalid block size.

disksecn aborts and prints an error message under the following conditions:

- disksecn was invoked without specifying a disk name.
- Requested both -p and -d switch.
- Illegal block size requested.
- Unknown disk name was specified and user did not supply disk information.
- Disk's maximum storage space is less than approximately 44 MB.

WARNINGS

Alternate names are not included in the output when the -d switch is used.
Blanks are required in the command line between each of the switches when invoking disksecn.
A blank is required between the -n switch and the disk name argument to that switch. For example:

```
disksecn -p -b 1024 -n hp9712
```

disksecn does not save the block size used to generate the /etc/disktab disk description file. The system assumes that the block size used was DEV_BSIZE when it reads the information stored in the /etc/disktab file.

AUTHOR

disksecn was developed by the University of California, Berkeley.

FILES

/etc/disktab

SEE ALSO

disktab(4).
NAME
diskusg - generate disk accounting data by user ID

SYNOPSIS
/usr/sbin/acct/diskusg [options] [files]

DESCRIPTION
diskusg generates intermediate disk accounting information from data in files, or the standard input if omitted. diskusg outputs lines on the standard output, one per user, in the following format:

uid login #blocks

where:
uid User's numerical userID,
login User's login name, and
#blocks Total number of disk blocks allocated to this user.

diskusg normally reads only the inodes of file systems for disk accounting. In this case, files are the special filenames of these devices.

Options
diskusg recognizes the following options:
-s Input data is already in diskusg output format. diskusg combines all lines for a single user into a single line.
-v verbose. Print a list on standard error of all files that are charged to no one.
-i /fnmlist Ignore the data on those file systems whose file system name is in /fnmlist. /fnmlist is a list of file system names, separated by commas or enclosed within quotes. diskusg compares each name in this list with the file system name stored in the volume ID if it exists.
-p file Use file as the name of the password file to generate login names. /etc/passwd is used by default.
-u file Write records to file of files that are charged to no one. Records consist of the special file name, the inode number, and the user ID.

The output of diskusg is normally the input to acctdisk (see acct(1M)) which generates total accounting records that can be merged with other accounting records. diskusg is normally run in dodisk (see acctsh(1M)).

EXAMPLES
The following generates daily disk accounting information:

for i in /dev/rp00 /dev/rp01 /dev/rp10 /dev/rp11; do
diskusg $i > dtmp.‘basename $i’ &
done
wait
diskusg -s dtmp.* | sort +0n +1 | acctdisk > disktacct

FILES
/etc/passwd used for user-ID-to-login-name conversions

SEE ALSO
acct(1M), acctsh(1M), volcopy(1M), vxdiskusg(1M), acct(4).

STANDARDS CONFORMANCE
diskusg: SVID2, SVID3
NAME
dmesg - collect system diagnostic messages to form error log

SYNOPSIS
/usr/sbin/dmesg [-] [core] [system]

DESCRIPTION

dmesg looks in a system buffer for recently printed diagnostic messages and prints them on the standard output. The messages are those printed by the system when unusual events occur (such as when system tables overflow or the system crashes). If the - argument is specified, dmesg computes (incrementally) the new messages since the last time it was run and places these on the standard output. This is typically used with cron (see cron(1)) to produce the error log /var/adm/messages by running the command:

   /usr/sbin/dmesg - >> /var/adm/messages

   every 10 minutes.

   The arguments core and system allow substitution for the defaults /dev/kmem and /stand/vmunix respectively, where core should be a file containing the image of the kernel virtual memory saved by the savecrash(1M) command and system should be the corresponding kernel. If the system is booted with a kernel other than /stand/vmunix say /stand/vmunix_new, dmesg must be passed this name, the command must be,

   /usr/sbin/dmesg [-] /dev/kmem /stand/vmunix_new

WARNINGS

   The system error message buffer is of small, finite size. dmesg is run only every few minutes, so there is no guarantee that all error messages will be logged.

AUTHOR

dmesg was developed by the University of California, Berkeley.

FILES

   /var/adm/messages error log (conventional location)
   /var/adm/msgbuf memory scratch file for - option
   /dev/kmem special file containing the image of kernel virtual memory
   /stand/vmunix the kernel, system name list

SEE ALSO

   savecrash(1M).
NAME
dpp - dedicated ports parser used by DDFA software

SYNOPSIS
dpp dp_file [\[-c\]] [\[-k\]] [\[-l\] log_file] [\[-p\] ocd_program]

DESCRIPTION
The Dedicated Ports Parser command (dpp) is part of the Data Communications and Terminal Controller (DTC) Device File Access (DDFA) software. It parses the Dedicated Ports file (dp) and spawns an Outbound Connection Daemon (ocd) for each valid entry in the dp file.

dpp can be run from the shell or it can be included in a system initialization script to automatically run the DDFA software each time the system is booted.

See ddfa(7) for more information on how to configure the DDFA software and for an explanation of how it works.

Options and Arguments
dpp recognizes the following options and arguments:

dp_file  It must be the first argument. The dp file (dp_file) defines the link between a terminal server port and the device file used by applications to access the port. Its contents must meet the specifications given in dp(4). If it is modified, dpp must be run again to activate the changes.

-c  Specify that the dp file should be parsed and that all incorrect entries should be logged without invoking any ocd processes. This option is useful for debugging the dp file before running it properly. The -p option is ignored if the -c option is used.

-k  Specify that the device file corresponding to each valid entry in the dp file should be removed before launching ocd for each valid entry. Removing the device file eventually causes an ocd process (if any is running) to shutdown. If this option is omitted, no device files will be removed and, therefore, only newly added valid entries in the dp file will have ocd launched.

ocd normally creates and removes devices files. However, if the process is killed incorrectly, such as with kill -9, the device file may remain. If the system is rebooted, the -k option can be specified to restart all dp file entries correctly.

If a corresponding ocd no longer exists, the device file is removed by any following invocation of ocd that requires the same device file.

In order to shutdown every ocd running without restarting them, the following command can be executed:

kill -15 `ps -e | grep ocd | awk '{print $1}``

-l log_file  Specify where to log error messages. If this option is omitted, all error messages are logged to standard output.

If the specified file does not already exist, it is created. The file must be nonexecutable and readable by dpp.

-p ocd_program  Specify the path for an outbound connection daemon. The default path for is /usr/sbin/ocd. The daemon must be executable.

DIAGNOSTICS
Error messages are logged for bad arguments, bad file entries, and ocd creation errors. By default, they are logged to standard output. If the -l option is used, they are appended to the specified log file.

(0) ERROR: dp file is mandatory
(1) ERROR: dp file must be the first argument
(2) ERROR: Cannot read dp file (filename)

The dp file either does not exist or cannot be accessed with the current access privileges.

(3) ERROR: No log file defined (-l option)
(4) ERROR: Cannot create log file (-l filename)
The log file cannot be created, either because of an invalid path or because of insufficient access privileges.

(5) **ERROR: Cannot access log file** (-l \*filename\*)

The log file cannot be accessed, either because of an invalid path or because of insufficient access privileges. The log file must be readable by everyone.

(6) **ERROR: No ocd file defined in program option**

(7) **ERROR: Cannot execute ocd program** (-p \*pathname\*)

The ocd program specified in the -p option either does not exist or is not an executable file with the current access privileges.

(8) **ERROR: Cannot purge device file** (/dev/\*filename\*)

The -k option has been specified and the device file exists, but it cannot be purged because of insufficient access privileges.

(9) **ERROR: Cannot execute default program** (/usr/sbin/ocd)

The default ocd cannot be executed, either because of insufficient access privileges or because it has not been correctly installed.

(10) **ERROR: Entry ignored** (Bad IP address)

The dp file entry specified does not have a valid IP address.

(11) **ERROR: Entry ignored** (no port/board info)

(12) **ERROR: Entry ignored** (Bad port number)

The port specified is either not a decimal value or a string composed of \*x\* or \*X\* characters.

(13) **ERROR: Entry ignored** (Bad board number)

The board specified is either not a decimal value or a string composed of \*x\* or \*X\* characters.

(14) **ERROR: No more processes available on system**

The ocd program specified cannot be started because there are no processes available on the system.

(15) **ERROR: Entry ignored** (no device_name)

(16) **ERROR: Entry ignored** (Bad device_name)

The device file specified cannot be created, either because of an invalid path or because of insufficient access privileges.

(17) **ERROR: Entry ignored** (Bad config name)

The specified configuration file cannot be read, either because of an invalid path or because of insufficient access privileges.

(18) **ERROR: Entry ignored** (Invalid log level)

The specified logging level is not in the range 0 to 3.

(19) **ERROR: Entry ignored** (Bad node name)

The specified node name does not exist or does not have an entry in a name database.

**WARNINGS**

To ensure that commands (such as ps) display the correct device file name (that is, the pseudonym), all pseudonyms should be placed into the directory /dev/telnet. If pseudonyms are not specified for placement in this directory, the correct display of device file names with many commands is not guaranteed.

In addition, to ensure that commands (such as w, passwd, finger, and wall) work correctly, each pseudonym must be unique in its first 17 characters (including the directory prefix /dev/telnet/). If pseudonyms are not unique in their first 17 characters, the correct functioning of many commands is not guaranteed.

Also, in order to reliably handle timing mark negotiations (and ensure that files printing on a printer attached to a terminal server have been completely flushed to that printer), the following line must be added near the end of each printer interface script for printers attached to a terminal server:

```
stty exta <&1 2>/dev/null
```
The printer interface scripts reside in the directory `/etc/lp/interface`. The line must be added just prior to the final `exit` command in each printer interface script.

If this line is not added as specified, the printing reliability of printers attached to a terminal server is not guaranteed.

Finally, `ocd` should be killed using `kill -15`. Do not use `kill -9` for this purpose as it does not remove the device file. `ocd` verifies the validity of an existing pseudonym before trying to use it. `dpp` and `ocd` use data stored in the file `/var/adm/utmp.dfa` to verify whether a process still owns a pseudonym before taking it over. If `ocd` finds an unowned pseudonym, it uses it.

**FILES**

- `/usr/examples/ddfa/dp`
- `/usr/examples/ddfa/pcf`
- `/usr/sbin/dpp`
- `/usr/sbin/ocd`
- `/usr/sbin/ocdebug`
- `/var/adm/dpp_login.bin`
- `/var/adm/utmp.dfa`

**SEE ALSO**

- `ocd(1M)`, `ocdebug(1M)`, `dp(4)`, `pcf(4)`, `ddf(7)`.
NAME
drd - manage an inactive system image

SYNOPSIS
drd [-?] [-x -?]

Command Modes
drd activate [-?] [-p] [-v] [-x option=value] [-x -?] [-X option_file]
drd clone [-?] [-p] [-v] [-t target_device_file] [-x option=value] [-x -?] [-X option_file]
drd deactivate [-?] [-p] [-v] [-x option=value] [-x -?] [-X option_file]
drd mount [-?] [-p] [-v] [-x option=value] [-x -?] [-X option_file]
drd runcmd [-?] [-v] [-x option=value] [-x -?] [-X option_file] cmd [args]
drd umount [-?] [-p] [-v] [-x option=value] [-x -?] [-X option_file]

DESCRIPTION
The drd command provides a command line interface to Dynamic Root Disk (DRD) tools. The drd command has five major modes of operation:

clone
Clones a booted system to an inactive system image. The drd clone mode copies the LVM volume group or VxVM disk group containing the volume on which the root file system (/) is mounted. See drd-clone(1M).

mount
Mounts all file systems in an inactive system image. See drd-mount(1M). The mount point of the root file system is:

/var/opt/drd/mnts/sysimage_000
If the inactive system image was created by the most recent drd clone command.

/var/opt/drd/mnts/sysimage_001
If the inactive system image was the booted system when the most recent drd clone command was run.

umount
Unmounts all file systems in the inactive system image previously mounted by a drd mount command. See drd-umount(1M).

runcmd
Runs a command on an inactive system image. See drd-runcmd(1M). Only a select group of commands may be run by the runcmd mode. These are commands that have been verified to have no affect on the booted system when executed by drd runcmd. Such commands are referred to as DRD-Safe. The commands kctune, swinstall, swjob, swlist, swmodify, swremove, swverify, and view are currently certified DRD-Safe. An attempt to execute any other command will result in a runcmd error. In addition, not every software package may safely be processed by sw* commands. The DRD-Safe SW-DIST commands are aware of running in a DRD session and will reject any unsafe packages. For more information about DRD-Safe packages, see drd-runcmd(1M).

activate
Sets the inactive system image to be the primary boot disk the next time the system is booted. See drd-activate(1M).

deactivate
Sets the active system image (that is, the booted image) to be the primary boot disk the next time the system is booted. See drd-deactivate(1M).

Options
drd without a command mode recognizes the following options:

-? Displays the usage message.
-x -? Displays the list of possible -x (extended) options.

In command mode, the -X option_file can be used to get the extended options from a file. See the EXAMPLES section for an example option_file. The option_file can have one or more lines as follows:

extended_option=value

For example: log verbosity=3

Refer to the Extended -x Options section of each DRD man-page.
**major_mode . extended_option=value**

For example: `clone.verbosity=5`

Refer to the **DESCRIPTION** section for an explanation of major modes.

---

# comment

Comment lines begin with # and are ignored.

---

**RETURN VALUE**

The **drd** command modes return one of the following values:

0 Success.
1 Error.
2 Warning.

---

**EXAMPLES**

To display **drd** usage information:

```
  drd -?
```

To display **drd** extended option usage:

```
  drd -x -?
```

To display usage for the **drd clone** command:

```
  drd clone -?
```

To clone the root LVM volume group or VxVM disk group to a physical device:

For 11i v2:
```
  drd clone -t /dev/dsk/c1t1d0
```

For 11i v3:
```
  drd clone -t /dev/disk/disk1
```

To preview the clone of the root LVM volume group or VxVM disk group to a physical device:

For 11i v2:
```
  drd clone -p -t /dev/dsk/c1t15d0
```

For 11i v3:
```
  drd clone -p -t /dev/disk/disk7
```

To display all **drd clone** extended options:
```
  drd clone -x -?
```

To mount the inactive system image:
```
  drd mount
```

If the system image mounted was created by the most recent **drd clone** command, the root file system will be mounted at `/var/opt/drd/mnts/sysimage_001`.

If the system image was booted when the most recent **drd clone** command was run, the root file system will be mounted at `/var/opt/drd/mnts/sysimage_000`.

To display all **drd mount** extended options:
```
  drd mount -x -?
```

To unmount the inactive system image:
```
  drd umount
```

To display all **drd umount** extended options:
```
  drd umount -x -?
```

To see the software that is installed on the inactive system image (without any need to mount the image first):

---

**HP-UX 11i Version 2: December 2007 Update**
To install PHCO_0001 from the depot /var/opt/patches, located on the system patchsvr:

```sh
drd runcmd swinstall -s patchsvr:/var/opt/patches PHCO_0001
```

To run a preview installation of PHCO_0001 from the depot /var/opt/patches, located on the system patchsvr:

```sh
drd runcmd swinstall -p -s patchsvr:/var/opt/patches PHCO_0001
```

To verify all software on the inactive system image:

```sh
drd runcmd swverify *
```

To remove PHKL_9999 from the inactive system image:

```sh
drd runcmd swremove PHKL_9999 *
```

To view the swagent log on the inactive system image:

```sh
drd runcmd view /var/adm/sw/swagent.log
```

To display all `drd runcmd` extended options:

```sh
drd runcmd -x -?
```

To set the inactive system image as the primary boot disk:

```sh
drd activate
```

To set the inactive system image as the primary boot disk and a different disk as the alternate boot disk:

For 11i v2:

```sh
drd activate -x alternate_bootdisk=/dev/dsk/c1t1d0
```

For 11i v3:

```sh
drd activate -x alternate_bootdisk=/dev/disk/disk1
```

To boot to the inactive system image immediately:

```sh
drd activate -x reboot=true
```

To display all `drd activate` extended options:

```sh
drd activate -x -?
```

To restore the active (booted) system image as the primary boot disk:

```sh
drd deactivate
```

To restore the active (booted) system image as the primary boot disk and set a different disk as the alternate boot disk:

For 11i v2:

```sh
drd deactivate -x alternate_bootdisk=/dev/dsk/c1t1d0
```

For 11i v3:

```sh
drd deactivate -x alternate_bootdisk=/dev/disk/disk1
```

To display all `drd deactivate` extended options:

```sh
drd deactivate -x -?
```

To define the `log verbosity` extended option for all `drd` command modes, and the `verbosity` extended option for the `clone`, `mount`, and `runcmd` modes in an option file named `/tmp/drd_defaults`:

```sh
# more /tmp/drd_defaults
log verbosity=3
clone verbosity=5
mount verbosity=4
runcmd verbosity=4
```

To run the `drd clone` command using the `/tmp/drd_defaults` option file:
drd(1M)

# drd clone -p -t /dev/dsk/c1t3d0 -x overwrite=true -X /tmp/drd_defaults

AUTHOR

drd was developed by HP.

FILES

/var/opt/drd/drd.log Log file.

SEE ALSO

drd-activate(1M), drd-clone(1M), drd-deactivate(1M), drd-mount(1M), drd-runcmd(1M), drd-umount(1M)

http://docs.hp.com/en/DRD.
NAME
drd-activate: drd - activate mode sets the inactive system image to be the primary boot disk the next time
the system is booted

SYNOPSIS
drd activate [-?] [-p] [-q]... [-v]... [-x option=value]... [-x -?] [-X option_file]

DESCRIPTION
The drd activate command invokes the setboot command (see setboot(1M)) to set the primary boot
disk to the inactive system image. If a clone has just been created, the inactive system image is the target
of the clone operation. If the clone has been booted, the inactive system image is the original system that
was cloned.

Note: Running drd activate multiple times with no intervening boot has the same effect as running it
once. To 'undo' drd activate, use drd deactivate. See drd-deactivate(1M).

Options
drd activate recognizes the following options:
-? Displays the usage message.
-p Sets preview mode. See also -x preview, in the Extended -x Options section below.
-q Decreases the verbosity level by one each time it is specified. For the interaction with -x
verbosity, see the Extended -x Options section below.
-v Increases the verbosity level by one each time it is specified. For the interaction with -x
verbosity, see the Extended -x Options section below.
-x -? Displays the list of possible -x (extended) options.
-x option=value Sets the extended option to a value. See the Extended -x Options section below.
-x option_file Gets the extended options from a file. See drd(1M) for an explanation of the format and an
example of an option file.

Extended -x Options
drd activate recognizes the following -x (extended) options. If a default value exists, it is shown in
bold.
-x alternate_bootdisk=diskpath
Usage: Basic
This is the alternate boot disk. The diskpath must be specified as a block device file.
-x HA_alternate_bootdisk=diskpath
Usage: Basic
This is the High Availability alternate boot disk. The diskpath must be specified as a block
device file. The High Availability alternate boot disk is supported only on Itanium-based
architecture and on PA-RISC systems that support hardware partitions.
-x logfile=/var/opt/drd/drd.log
Usage: Basic
This is the path to the log file for this command. Each time DRD is run, this file will grow
larger. This can be changed, for example, to a month-specific location for easier archiving,
off-host backup, and rotation.
-x log_verbosity=4
Usage: Basic
Specifies the level of message verbosity in the log file. (See also -x verbosity.) Legal
values are:
0 Saves only ERROR messages and the starting/ending BANNER messages.
1 Adds WARNING messages.
 Adds NOTE messages.
3  Adds INFO messages (informational messages preceded by the * character).
4  Adds verbose INFO messages.
5  Adds very-verbose INFO messages.

-x preview=false
Usage: Basic
If true, run this command in preview mode only (that is, complete the analysis phase and
exit; do not commit changes to disk). Setting this option to true has the same effect as
specifying -p on the command line.

-x reboot=false
Usage: Basic
If true, the system is rebooted at the successful completion of a drd activate opera-
tion.

-x verbosity=3
Usage: Basic
 Specifies the level of standard error verboseness. Legal values are:
0  Displays only ERROR messages and the starting/ending BANNER messages.
1  Adds WARNING messages.
2  Adds NOTE messages.
3  Adds INFO messages (informational messages preceded by the * character).
4  Adds verbose INFO messages.
5  Adds very-verbose INFO messages.

See also the -v and -q options in the Options section above. The -v option increases the
verboseness by 1 (for example, from 3 to 4) and the -q option decreases the verboseness by 1.
The -v and -q options can be given more than once. Multiples can be merged, as in -qq.
For example, if both -x verbosity=5 and -qqq are included on the command line, the
effective verboseness is 2. The minimum combined verboseness level is 0.
For example, if both -x verbosity=1 and -vv are included on the command line, the
effective verboseness is 3. The maximum combined verboseness level is 5.

RETURN VALUE
drd activate returns one of the following values:
0   Success.
1   Error.
2   Warning.

EXAMPLES
To set the inactive system image as the primary boot disk:
drd activate
To set the inactive system image as the primary boot disk and a different disk as the alternate boot disk:
For 11i v2:
drd activate -x alternate_bootdisk=/dev/dsk/c1t1d0
For 11i v3:
drd activate -x alternate_bootdisk=/dev/disk/disk1
To boot to the inactive system image immediately:
drd activate -x reboot=true
To display all `drd activate` extended options:

```
   drd activate -x -?
```

**AUTHOR**

`drd` was developed by HP.

**FILES**

/var/opt/drd/drd.log Log file.

**SEE ALSO**

`drd(1M), drd-clone(1M), drd-deactivate(1M), drd-mount(1M), drd-runcmd(1M), drd-umount(1M), setboot(1M)`

NAME

drd-clone: drd - clone mode clones the root volume group

SYNOPSIS

drd clone [-?] [-p] [-q]... [-v]...
   -t target_device_file [-x option=value] [-x -?] [-X option_file]

DESCRIPTION

The `drd clone` command creates a copy of the LVM volume group or VxVM disk group containing the root file system (/). In particular, it creates a new group, creates a volume in the new group for each volume in the root group, configures swap and dump volumes, and copies the contents of each file system in the root group to the corresponding file system in the new group.

Options

`drd clone` recognizes the following options:

-? Displays the usage message.

-p Sets preview mode. See also `-x preview`, in the Extended -x Options section below.

-q Decreases the verbosity level by one each time it is specified. For the interaction with `-x verbosity`, see the Extended -x Options section below.

-t device_special_file
   Specifies the block device special file of a single physical disk on which the cloned system image is to be written. The block device special file must exist on the system and be writeable. All data previously on the disk will be unavailable after a clone operation. See also `-x overwrite`, in the Extended -x Options section below.

-v Increases the verbosity level by one each time it is specified. For the interaction with `-x verbosity`, see the Extended -x Options section below.

-x -?
   Displays the list of possible `-x` (extended) options.

-x option=value
   Sets the extended option to a value. See the Extended -x Options section below.

-X option_file
   Gets the extended options from a file. See `drd(1M)` for an explanation of the format and an example of an option file.

Extended -x Options

`drd clone` recognizes the following `-x` (extended) options. If a default value exists, it is shown in bold.

-x ignore_unmounted_fs=false
   Usage: Basic
   Controls whether a clone fails when an unmounted file system in the root volume group is detected.

-x logfile=/var/opt/drd/drd.log
   Usage: Basic
   This is the path to the log file for this command. Each time DRD is run, this file will grow larger. This can be changed, for example, to a month-specific location for easier archiving, off-host backup, and rotation.

-x log verbosity=4
   Usage: Basic
   Specifies the level of message verbosity in the log file. (See also `-x verbosity`. Legal values are:

<table>
<thead>
<tr>
<th>Value</th>
<th>Message Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Saves only ERROR messages and the starting/ending BANNER messages.</td>
</tr>
<tr>
<td>1</td>
<td>Adds WARNING messages.</td>
</tr>
<tr>
<td>2</td>
<td>Adds NOTE messages.</td>
</tr>
<tr>
<td>3</td>
<td>Adds INFO messages (informational messages preceded by the * character).</td>
</tr>
</tbody>
</table>

Hewlett-Packard Company 201
4 Adds verbose INFO messages.
5 Adds very-verbose INFO messages.

-x mirror_disk=mirrordisk
Usage: Basic
The block device special file of the mirror disk of the target. The device special file mirror-
disk, should refer to an entire disk, not to a partition. This option requires that LVM mir-
roring be installed. The block device file specified will be used to mirror each logical volume
in the target of the clone operation.

-x overwrite=false
Usage: Basic
Controls whether or not a disk containing boot, LVM, or VxVM records can be overwritten.
Note that DRD will not overwrite a disk associated with an active LVM volume group or
VxVM disk group, regardless of the setting of the overwrite option. If a previously
created clone is mounted, use the drd umount command to unmount it before attempt-
ning to create a new clone on the disk.
true Allow a disk to be overwritten, even if it contains boot, LVM, or VxVM records.
false Prevent a disk that contains boot, LVM, or VxVM records from being overwrit-
ten.

-x preview=false
Usage: Basic
If true, run this command in preview mode only (that is, complete the analysis phase and
exit; do not commit changes to disk). Setting this option to true has the same effect as
specifying -p on the command line.

-x verbosity=3
Usage: Basic
Specifies the level of standard error verboseness. Legal values are:
0 Display only ERROR messages and the starting/ending BANNER messages.
1 Adds WARNING messages.
2 Adds NOTE messages.
3 Adds INFO messages (informational messages preceded by the * character).
4 Adds verbose INFO messages.
5 Adds very-verbose INFO messages.

See also the -v and -q options in the Options section above. The -v option increases the
verbosity by 1 (for example, from 3 to 4) and the -q option decreases the verbosity by 1.
The -v and -q options can be given more than once. Multiples can be merged, as in -qq.
For example, if both -x verbosity=5 and -qqq are included on the command line,
the effective verbosity is 2. The minimum combined verbosity level is 0.
For example, if both -x verbosity=1 and -vv are included on the command line, the
effective verbosity is 3. The maximum combined verbosity level is 5.

The Source
The source for the drd clone command — the group that is copied — is the group containing the root
file system (/).

Choosing a Target Disk
The target disk must be specified as a block device file. An appropriate target disk should be writeable by
the system, not currently in use by other applications, and large enough to hold a copy of each logical
volume in the root group.
The physical disk need not be as large as the disk allocated for the root group as long as there is sufficient
space for a copy of each volume in the root group. However, the disk will need to be bigger than the used
space in each volume, since each volume will be created with the number of physical extents currently allo-
cated to the corresponding root volume.
The HP System Management Homepage (see hpsmh(1M)) or System Administration Manager (see sam(1M)) can be used to investigate the disks on the system and their current usage.

Alternatively, the following command-line utilities may be useful in determining an appropriate target disk:

- The command `ioscan -fknC disk` can be used to determine the physical disks on the system. See ioscan(1M).
- Commands from various volume managers and applications can be used to determine the current usage of disks on the system. For example, `vgdisplay -v` displays the disks currently in use by the LVM volume manager, usually labeled with the identifier PV Name. See vgdisplay(1M). Similarly, the command `vxdisk -o alldgs list` can be used to display information about all disks managed by VxVM. See vxdisk(1M).
- The command `swapinfo` can be used to display information about the disks that are currently used for swap. See swapinfo(1M).

Depending on the applications in use on the system, further checks may be needed to ensure that disks are not in use. For example, any "raw" disks in use by databases may need to be identified.

**Note:** It is the administrator's responsibility to determine which disks are not currently in use and may therefore be used for a clone of the root group.

The `drd clone` command itself will perform the following checks:

- If the disk is currently in use by the LVM volume manager, it will be rejected by `drd clone`.
- If the disk is currently in use by the VxVM volume manager, it will only be accepted as a `drd clone` target if the disk is an inactive image managed by DRD and the extended option `-x overwrite=true` is specified.
- If the disk is not currently in use by LVM or VxVM, but contains LVM, VxVM, or boot records, it will only be accepted as a `drd clone` target if `-x overwrite=true` is specified.

For further information on choosing a target disk for a clone operation, see the Dynamic Root Disk Administrator's Guide (see the SEE ALSO section).

### The Target Volume Manager

The target volume manager must be the same as the source volume manager.

- If the source group is an LVM volume group of the form `vg nn`, the clone is imported with the volume group name `drd nn` and booted with the original volume name `vg nn`. For example, if `vg00` is cloned, the clone is imported as `drd00` and booted as `vg00`.
- If the source group is an LVM volume group not of the form `vg nn`, the clone is imported with a volume group name formed by prefixing the source group with `drd_`.
- If the source group is a VxVM volume group not beginning with `drd_`, the clone is imported and booted with a volume group name formed by prefixing the source group with `drd_`.
- If the source group is a VxVM volume group beginning with `drd_`, the clone is imported and booted with a volume group name formed by removing the prefix `drd_`.

### The DRD Log

The DRD log resides at `/var/opt/drd/drd.log`. During any DRD operation, the DRD log is written to the booted system. In addition, since the log is part of the `/var` file system, it is copied by the `drd clone` command to the target of the `drd clone` operation. Since the file systems on the clone are unmounted before the log has been completely written, the DRD log file on the target of a `drd clone` operation will be truncated. The messages following the truncated clone log will be those from the first `drd` operation run on the clone after it is booted.

### RETURN VALUE

`drd clone` returns one of the following values:

- **0** Success.
- **1** Error.
- **2** Warning.
drd-clone(1M)

**EXAMPLES**
To display usage for the `drd clone` command:
```
    drd clone -?
```
To clone the root LVM volume group or VxVM disk group to a physical device:
For 11i v2:
```
    drd clone -t /dev/dsk/c1t1d0
```
For 11i v3:
```
    drd clone -t /dev/disk/disk1
```
To preview the clone of the root LVM volume group or VxVM disk group to a physical device:
For 11i v2:
```
    drd clone -p -t /dev/dsk/c1t15d0
```
For 11i v3:
```
    drd clone -p -t /dev/disk/disk7
```
To display all `drd clone` extended options:
```
    drd clone -x -?
```

**AUTHOR**
drd was developed by HP.

**FILES**
```
/var/opt/drd/drd.log
```
Log file.

**SEE ALSO**
drd(1M), drd-activate(1M), drd-deactivate(1M), drd-mount(1M), drd-runcmd(1M), drd-umount(1M),
ioscan(1M), swapinfo(1M), vgdisplay(1M), vxdisk(1M)

*Dynamic Root Disk Administrator's Guide*, available on the HP Technical Documentation web site at
```
http://docs/hp.com/en/DRD.
```
NAME
drd-deactivate: drd - deactivate mode sets the active system image to be the primary boot disk the next
time the system is booted

SYNOPSIS
drd deactivate [-?] [-p] [-q]... [-v]... [-x option=value] [-x -?] [-X option_file]

DESCRIPTION
The drd deactivate command invokes the setboot command (see setboot(1M)) to set the primary
boot disk to the active (booted) system image.

Note: drd deactivate is a mechanism for "undoing" a drd activate command.

Options
drd deactivate recognizes the following options:
  -?   Displays the usage message.
  -p   Sets preview mode. See also -x preview, in the Extended -x Options section below.
  -q   Decreases the verbosity level by one each time it is specified. For the interaction with -x
       verbosity, see the Extended -x Options section below.
  -v   Increases the verbosity level by one each time it is specified. For the interaction with -x
       verbosity, see the Extended -x Options section below.
  -x -? Displays the list of possible -x (extended) options.
  -x option=value    Sets the list of possible -x (extended) options.
  -x option_file     Gets the extended options from a file. See drd(1M) for an explanation of the format and an
                     example of an option file.

Extended -x Options
drd deactivate recognizes the following -x (extended) options. If a default value exists, it is shown
in bold.

  -x alternate_bootdisk=diskpath
     Usage: Basic
     This is the alternate boot disk. The diskpath must be specified as a block device file.

  -x HA_alternate_bootdisk=diskpath
     Usage: Basic
     This is the High Availability alternate boot disk. The diskpath must be specified as a block
     device file. High Availability alternate boot disk is supported only on Itanium-based archi-
     tecture and on PA-RISC systems that support hardware partitions.

  -x logfile=/var/opt/drd/drd.log
     Usage: Basic
     This is the path to the log file for this command. Each time DRD is run, this file will grow
     larger. This can be changed, for example, to a month-specific location for easier archiving,
     off-host backup, and rotation.

  -x log_verbosity=4
     Usage: Basic
     Specifies the level of message verbosity in the log file. (See also -x verbosity.) Legal
     values are:
     0   Saves only ERROR messages and the starting/ending BANNER messages.
     1   Adds WARNING messages.
     2   Adds NOTE messages.
     3   Adds INFO messages (informational messages preceded by the * character).
drd-deactivate(1M)

4 Adds verbose INFO messages.
5 Adds very-verbose INFO messages.

-x preview=false
Usage: Basic
If true, run this command in preview mode only (that is, complete the analysis phase and exit; do not commit changes to disk). Setting this option to true has the same effect as specifying -p on the command line.

-x verbosity=3
Usage: Basic
Specifies the level of standard error verboseness. Legal values are:
0 Displays only ERROR messages and the starting/ending BANNER messages.
1 Adds WARNING messages.
2 Adds NOTE messages.
3 Adds INFO messages (informational messages preceded by the * character).
4 Adds verbose INFO messages.
5 Adds very-verbose INFO messages.
See also the -v and -q options in the Options section above. The -v option increases the verbosity by 1 (for example, from 3 to 4) and the -q option decreases the verbosity by 1. The -v and -q options can be given more than once. Multiples can be merged, as in -qq.
For example, if both -x verbosity=5 and -qq are included on the command line, the effective verbosity is 2. The minimum combined verbosity level is 0.
For example, if both -x verbosity=1 and -vv are included on the command line, the effective verbosity is 3. The maximum combined verbosity level is 5.

RETURN VALUE
drd deactivate returns one of the following values:
0 Success.
1 Error.
2 Warning.

EXAMPLES
To restore the active (booted) system image as the primary boot disk:
  drd deactivate
To restore the active (booted) system image as the primary boot disk and set a different disk as the alternate boot disk:
  For 11i v2:
  drd deactivate -x alternate_bootdisk=/dev/dsk/c1t1d0
  For 11i v3:
  drd deactivate -x alternate_bootdisk=/dev/disk/disk1
To display all drd deactivate extended options:
  drd deactivate -x -?

AUTHOR
drd was developed by HP.

FILES
/var/opt/drd/drd.log Log file.
SEE ALSO

drd(1M), drd-activate(1M), drd-clone(1M), drd-mount(1M), drd-runcmd(1M), drd-umount(1M)

http://docs/hp.com/en/DRD.
NAME
drd-mount: drd - mount mode mounts the inactive system image

SYNOPSIS
drd mount [-p] [-q]... [-v]... [-x option=value]... [-x -?] [-x option_file]

DESCRIPTION
The drd mount command mounts all the file systems in the inactive system image.
The DRD commands manage two system images:
  • The image sysimage_000 consists of the LVM volume group or VxVM disk group that was booted
    when the most recent drd clone command was run, and resides on the physical disk that contains
    that group.
  • The image sysimage_001 consists of the LVM volume group or VxVM disk group that was created
    by the drd clone command, and resides on the target disk of the drd clone command.
Since the system may have been booted from the clone between running drd clone and drd mount,
either sysimage_001 or sysimage_000 may be active (booted) when the drd mount command is run.
The root file system of the inactive system image is mounted as /var/opt/drd/mnts/name, where
name is sysimage_000, or sysimage_001, as described above. All other file systems are mounted
under the root file system.

Options
drd mount recognizes the following options:
-? Displays the usage message.
-p Sets preview mode. See also -x preview, in the Extended -x Options section below.
-q Decreases the verbosity level by one each time it is specified. For the interaction with -x
  verbosity, see the Extended -x Options section below.
-v Increases the verbosity level by one each time it is specified. For the interaction with -x
  verbosity, see the Extended -x Options section below.
-x -? Displays the list of possible -x (extended) options.
-x option=value Sets the extended option to a value. See the Extended -x Options section below.
-X option_file Gets the extended options from a file. See drd(1M) for an explanation of the format and an
  example of an option file.

Extended -x Options
drd mount recognizes the following -x (extended) options. If a default value exists, it is shown in bold.
-x logfile=/var/opt/drd/drd.log
  Usage: Basic
  This is the path to the log file for this command. Each time DRD is run, this file will grow
  larger. This can be changed, for example, to a month-specific location for easier archiving,
  off-host backup, and rotation.
-x log_verbosity=4
  Usage: Basic
  Specifies the level of message verbosity in the log file. (See also -x verbosity.) Legal
  values are:
  0 Saves only ERROR messages and the starting/ending BANNER messages.
  1 Adds WARNING messages.
  2 Adds NOTE messages.
  3 Adds INFO messages (informational messages preceded by the * character).
drd-mount(1M)

4    Adds verbose INFO messages.
5    Adds very-verbose INFO messages.

-x preview=false
Usage: Basic
    If true, run this command in preview mode only (that is, complete the analysis phase and
    exit; do not commit changes to disk). Setting this option to true has the same effect as
    specifying -p on the command line.

-x verbosity=3
Usage: Basic
    Specifies the level of standard error verboseness:
    0    Displays only ERROR messages and the starting/ending BANNER messages.
    1    Adds WARNING messages.
    2    Adds NOTE messages.
    3    Adds INFO messages (informational messages preceded by the * character).
    4    Adds verbose INFO messages.
    5    Adds very-verbose INFO messages.

See also the -v and -q options in the Options section above. The -v option increases the
verbosity by 1 (for example, from 3 to 4) and the -q option decreases the verbosity by 1.
The -v and -q options can be given more than once. Multiples can be merged, as in -qq.

For example, if both -x verbosity=5 and -qqq are included on the command line, the
effective verbosity is 2. The minimum combined verbosity level is 0.

For example, if both -x verbosity=1 and -vv are included on the command line, the
effective verbosity is 3. The maximum combined verbosity level is 5.

RETURN VALUE
drd mount returns one of the following values:

0    Success.
1    Error.
2    Warning.

EXAMPLES
To mount the inactive system image:

    drd mount

    If the system image mounted was created by the most recent drd clone command, the root file
    system will be mounted at /var/opt/drd/mnts/sysimage_001.

    If the system image was booted when the most recent drd clone command was run, the root file
    system will be mounted at /var/opt/drd/mnts/sysimage_000.

To display all drd mount extended options:

    drd mount -x -?

AUTHOR
drd was developed by HP.

FILES
/var/opt/drd/drd.log    Log file.
/var/opt/drd/mnts/sysimage_001
    Mount point for the root file system of an inactive system image created by a drd clone command.
/var/opt/drd/mnts/sysimage_000
    Mount point for the root file system of an inactive system image that

was the booted system when \texttt{drd~clone} was run.

\textbf{SEE ALSO}

drd(1M), drd-activate(1M), drd-clone(1M), drd-deactivate(1M), drd-runcmd(1M), drd-umount(1M)

NAME
drd-runcmd: drd - runcmd mode runs a command on the inactive system image that will not make any changes to the booted system, the kernel, or the process space

SYNOPSIS
drd runcmd [-?] [-q]... [-v]... [-x option=value]... [-x -?] [-X option_file] cmd [args]

DESCRIPTION
The drd runcmd command runs a command on the inactive system image that will not make any changes to the booted system, the kernel, or the process space.

The drd runcmd command enables an administrator to make changes to the inactive system image without incurring any disruption to the currently booted system. This means that not all commands can safely be executed by the drd runcmd operation. For example, commands that start or stop daemons or change dynamic kernel tunables are disruptive to current processes and must be prevented by the drd runcmd operation.

Commands which are not disruptive to the booted system and which perform appropriate actions on the inactive system are known as DRD-Safe. A short list of commands is recognized by the drd runcmd operation to be DRD-Safe. An attempt to use DRD to run commands that are not DRD-Safe will terminate with an ERROR return code without executing the command. The DRD-Safe commands are listed in the Operands section below in the cmd column.

Options
drd runcmd recognizes the following options:
- ? Displays the usage message.
- p Sets preview mode. See also -x preview, in the Extended -x Options section below.
- q Decreases the verbosity level by one each time it is specified. For the interaction with -x verbosity, see the Extended -x Options section below.
- v Increases the verbosity level by one each time it is specified. For the interaction with -x verbosity, see the Extended -x Options section below.
- x - ? Displays the list of possible -x (extended) options.
- x option=value Sets the extended option to a value. See the Extended -x Options section below.
- X option_file Gets the extended options from a file. See drd(1M) for an explanation of the format and an example of an option file.

Operands
drd runcmd recognizes the following operands:
cmd [args] Specifies the DRD-Safe command to be run. The rest of the command line (args) is passed as arguments of cmd. The value of cmd must be one of the following:

<table>
<thead>
<tr>
<th>cmd</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>kctune</td>
<td>Change or display kernel tunables.</td>
</tr>
<tr>
<td>swinstall</td>
<td>Install software products and patches that are DRD safe.</td>
</tr>
<tr>
<td>swjob</td>
<td>Display job information</td>
</tr>
<tr>
<td>swlist</td>
<td>Display information about software products and patches.</td>
</tr>
<tr>
<td>swmodify</td>
<td>Modify software products and patches.</td>
</tr>
<tr>
<td>swremove</td>
<td>Remove software products and patches.</td>
</tr>
<tr>
<td>swverify</td>
<td>Verify software products and patches.</td>
</tr>
<tr>
<td>view</td>
<td>Read files on the inactive image.</td>
</tr>
</tbody>
</table>

Extended -x Options
drd runcmd recognizes the following -x (extended) options. If a default value exists, it is shown in bold.

HP-UX 11i Version 2: December 2007 Update
-x logfile=/var/opt/drd/drd.log
  Usage: Basic
  This is the path to the log file for this command. Each time DRD is run, this file will grow larger. This can be changed, for example, to a month-specific location for easier archiving, off-host backup, and rotation.

- x log_verbosity=4
  Usage: Basic
  Specifies the level of message verbosity in the log file. (See also -x verbosity.) Legal values are:
  0  Saves only ERROR messages and the starting/ending BANNER messages.
  1  Adds WARNING messages.
  2  Adds NOTE messages.
  3  Adds INFO messages (informational messages preceded by the * character).
  4  Adds verbose INFO messages.
  5  Adds very-verbose INFO messages.

- x preview=false
  Usage: Basic
  If true, run this command in preview mode only (that is, complete the analysis phase and exit; do not commit changes to disk). Setting this option to true has the same effect as specifying -p on the command line.

- x verbosity=3
  Usage: Basic
  Specifies the level of standard error verboseness:
  0  Displays only ERROR messages and the starting/ending BANNER messages.
  1  Adds WARNING messages.
  2  Adds NOTE messages.
  3  Adds INFO messages (informational messages preceded by the * character).
  4  Adds verbose INFO messages.
  5  Adds very-verbose INFO messages.

See also the -v and -q options in the Options section above. The -v option increases the verbosity by 1 (for example, from 3 to 4) and the -q option decreases the verbosity by 1. The -v and -q options can be given more than once. Multiples can be merged, as in -qq.

For example, if both -x verbosity=5 and -qqq are included on the command line, the effective verbosity is 2. The minimum combined verbosity level is 0.

For example, if both -x verbosity=1 and -vv are included on the command line, the effective verbosity is 3. The maximum combined verbosity level is 5.

Restrictions on Commands Executed by drd runcmd
DRD supports read access of registered directory depots and directory depots on the booted system. It does not support serial depots and it does not support depots on the inactive system image.

Depots cannot be listed or used as targets by drd runcmd. Thus, the commands swlist, swremove, swmodify, and swverify can be run by drd runcmd to view or modify the inactive system image, but not to view or modify depot contents.

Syntax Restrictions for sw* Commands
Since depot manipulation is not supported under drd runcmd, any option that indicates a depot target is not supported. This includes the following:

- The -d option on swlist, swmodify, swverify, and swremove.
- The @ option on swinstall, swlist, and swverify.
Any double quotation marks (" or wild cards (* or ?) in the command line of the command to be run must be escaped to prevent inappropriate shell expansion or truncation of the command.

Any file referenced by the command line, that is, the argument of the view command and any argument of any of the sw* command options, -c, -f, -P, -S, -X, -x logfile, or -x patch_filter must reside on the inactive system image and must be designated by the path it would have if the inactive image were booted. This is the same as its path relative to the mount point of a mounted inactive image. For example, the location that will be /var/opt/list when the image is booted is either /var/opt/drd/mnts/sysimage_001/var/opt/list or /var/opt/drd/mnts/sysimage_000/var/opt/list when the inactive image is mounted. If this file is an argument of one of the options listed above, it must be specified simply as /var/opt/list.

This restriction also applies to the arguments of -x logfile and -x patch_filter that appear in a file that is the argument of the sw* -X option, and to paths that appear in a file that is the argument of an sw* -S option.

Note that this restriction does NOT apply to local directory depots, which must reside on the booted system, and be designated by their location on the booted system.

Note that you can have -x and -X options for both the drd runcmd (which would contain the runcmd's options) command and for the command you are running (which would contain that command's options).

The following (rare) options of sw* commands are not supported by drd runcmd:

<table>
<thead>
<tr>
<th>Option</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>-c</td>
<td>Specification of a catalog is not supported.</td>
</tr>
<tr>
<td>-d</td>
<td>Depot targets are not supported by drd runcmd.</td>
</tr>
<tr>
<td>-F</td>
<td>Execution of fix scripts is not supported by drd runcmd.</td>
</tr>
<tr>
<td>-J</td>
<td>Scheduling of jobs is not supported by drd runcmd.</td>
</tr>
<tr>
<td>-Q</td>
<td>Scheduling of jobs is not supported by drd runcmd.</td>
</tr>
<tr>
<td>-r</td>
<td>Installation to SD-style alternate roots is not supported by drd runcmd.</td>
</tr>
<tr>
<td>-t</td>
<td>The inactive system image is always the target in a drd runcmd operation.</td>
</tr>
<tr>
<td>-x admin_dir</td>
<td>Nonstandard IPD locations are not supported by drd runcmd.</td>
</tr>
<tr>
<td>-x agent_auto_exit</td>
<td>Nonstandard IPD locations are not supported by drd runcmd.</td>
</tr>
<tr>
<td>-x ask</td>
<td>Use of response files is not supported by drd runcmd.</td>
</tr>
<tr>
<td>-x auto_kernel_build</td>
<td>Regardless of the value of this option, a swremove executed by drd runcmd builds the kernel for the next boot.</td>
</tr>
<tr>
<td>-x autoreboot</td>
<td>Regardless of the value of this option, drd runcmd suppresses all reboots.</td>
</tr>
<tr>
<td>-x controller_source</td>
<td>This option is not supported by drd runcmd.</td>
</tr>
<tr>
<td>-x distribution_source_directory</td>
<td>This option is not supported by drd runcmd.</td>
</tr>
<tr>
<td>-x distribution_target_directory</td>
<td>This option is not supported by drd runcmd.</td>
</tr>
<tr>
<td>-x force_single_target</td>
<td>This option is obsolete and is not supported by drd runcmd.</td>
</tr>
<tr>
<td>-x mount_all_filesystems</td>
<td>This option defaults to false in a drd runcmd operation. If it is set to true on the command line, the operation will probably not succeed.</td>
</tr>
<tr>
<td>-x rpc_binding_info_target</td>
<td>This option is not supported by drd runcmd.</td>
</tr>
<tr>
<td>-x remove_empty_depot</td>
<td>Operations targeted at depots are not supported by drd runcmd.</td>
</tr>
<tr>
<td>-x un_as_superuser</td>
<td>This option is not meaningful for drd runcmd, since it must be run as root.</td>
</tr>
<tr>
<td>-x select_local</td>
<td>This option is not supported by drd runcmd.</td>
</tr>
<tr>
<td>-x source_tape</td>
<td>Tape depots are not supported by drd runcmd.</td>
</tr>
</tbody>
</table>
drd-runcmd(1M)

-x target_shared_root  This option is obsolete and is not supported by drd runcmd.
-x targets           This option is not supported by drd runcmd. The target is always the inactive system image.
-x use_alternate_source  This option is obsolete and is not supported by drd runcmd.

RETURN VALUE

   drd runcmd returns the following values:
   0  Success.
   1  Error.
   2  Warning.

EXAMPLES

To see the software that is installed on the inactive system image (without any need to mount the image first):

    drd runcmd swlist

To install PHCO_0001 from the depot /var/opt/patches, located on the system patchsvr:

    drd runcmd swinstall -s patchsvr:/var/opt/patches PHCO_0001

To run a preview installation of PHCO_0001 from the depot /var/opt/patches, located on the system patchsvr:

    drd runcmd swinstall -p -s patchsvr:/var/opt/patches PHCO_0001

To verify all software on the inactive system image:

    drd runcmd swverify \*

To remove PHKL_9999 from the inactive system image:

    drd runcmd swremove PHKL_9999 \*

To view the swagent log on the inactive system image:

    drd runcmd view /var/adm/sw/swagent.log

To display all drd runcmd extended options:

    drd runcmd -x -?

AUTHOR

   drd was developed by HP.

FILES

   /var/opt/drd/drd.log  Log file.

SEE ALSO

   view(1), drd(1M), drd-activate(1M), drd-clone(1M), drd-deactivate(1M), drd-mount(1M), drd-umount(1M),
   kctune(1M), swinstall(1M), swlist(1M), swmodify(1M), swremove(1M), swverify(1M)

   http://docs.hp.com/en/DRD.
NAME
drd-umount: drd - umount mode unmounts the inactive system image

SYNOPSIS
drd umount [-?] [-p] [-q]... [-v]... [-x option=value]... [-x -?] [-X option_file]

DESCRIPTION
The drd umount command unmounts all the file systems in the inactive system image.
The root file system of the inactive system image is expected to be mounted at either
/var/opt/drd/mnts/sysimage_000 or /var/opt/drd/mnts/sysimage_001.
If the inactive image contains an LVM volume group, the drd umount command then deactivates and
exports the volume group from the system.
If the inactive image contains a VxVM disk group, the drd umount command then issues vxvol stopall
but does not deport the disk group. This ensures that the inactive image remains bootable.

Note: Before running drd umount, all processes (including shells and editing sessions) must change
directories to a file system outside of the inactive system image or be terminated. Otherwise, drd
umount will fail with a Device busy error.

Options
drd umount recognizes the following options:
-? Displays the usage message.
-p Sets preview mode. See also -x preview, in the Extended -x Options section below.
-q Decreases the verbosity level by one each time it is specified. For the interaction with -x verbosity, see the Extended -x Options section below.
-v Increases the verbosity level by one each time it is specified. For the interaction with -x verbosity, see the Extended -x Options section below.
-x -? Displays the list of possible -x (extended) options.
-x option=value Sets the extended option to a value. See the Extended -x Options section below.
-X option_file Gets the extended options from a file. See drd(1M) for an explanation of the format and an example of an option file.

Extended -x Options
drd umount recognizes the following -x (extended) options. If a default value exists, it is shown in
bold.
-x logfile=/var/opt/drd/drd.log
Usage: Basic
This is the path to the log file for this command. Each time DRD is run, this file will grow
larger. This can be changed, for example, to a month-specific location for easier archiving,
off-host backup, and rotation.

-x log_verbosity=4
Usage: Basic
Specifies the level of message verbosity in the log file. (See also -x verbosity.) Legal
values are:
0 Saves only ERROR messages and the starting/ending BANNER messages.
1 Adds WARNING messages.
2 Adds NOTE messages.
3 Adds INFO messages (informational messages preceded by the * character).
4 Adds verbose INFO messages.
5 Adds very-verbose INFO messages.
-x preview=false
  Usage: Basic
  If true, run this command in preview mode only (that is, complete the analysis phase and
  exit; do not commit changes to disk). Setting this option to true has the same effect as
  specifying -p on the command line.

-x verbosity=3
  Usage: Basic
  Specifies the level of standard error verboseness:
  0  Displays only ERROR messages and the starting/ending BANNER messages.
  1  Adds WARNING messages.
  2  Adds NOTE messages.
  3  Adds INFO messages (informational messages preceded by the * character).
  4  Adds verbose INFO messages.
  5  Adds very-verbose INFO messages.
  See also the -v and -q options in the Options section above. The -v option increases the
  verbosity by 1 (for example, from 3 to 4) and the -q option decreases the verbosity by 1.
  The -v and -q options can be given more than once. Multiples can be merged, as in -qq.
  For example, if both -x verbosity=5 and -qq are included on the command line,
  the effective verbosity is 2. The minimum combined verbosity level is 0.
  For example, if both -x verbosity=1 and -vv are included on the command line, the
  effective verbosity is 3. The maximum combined verbosity level is 5.

RETURN VALUE
  drd umount returns one of the following values:
  0  Success.
  1  Error.
  2  Warning.

EXAMPLES
  To unmount the inactive system image:
  drd umount
  To display all drd umount extended options:
  drd umount -x -?

AUTHOR
  drd was developed by HP.

FILES
  /var/opt/drd/drd.log  Log file.
  /var/opt/drd/mnts/sysimage_000
    Mount point for the root file system of an inactive system image that
    was the booted system when drd clone was run.
  /var/opt/drd/mnts/sysimage_001
    Mount point for the root file system of an inactive system image
    created by a drd clone command.

SEE ALSO
  drd(1M), drd-activate(1M), drd-clone(1M), drd-deactivate(1M), drd-mount(1M), drd-runcmd(1M)
  http://docs/hp.com/en/DRD.
NAME
    drd_register_mirror - notify DRD that a system image has been manually mirrored using LVM or VxVM commands

SYNOPSIS
    drd_register_mirror [-?] mirror_block_device_file

DESCRIPTION
    The drd_register_mirror command notifies DRD that either the active system image or the inactive system image has been mirrored to the mirror_block_device_file specified.
    The drd_register_mirror command must be run after the mirror has been created.

Options
    drd_register_mirror recognizes the following option:
    -? Displays the usage message.

Operands
    drd_register_mirror recognizes the following operand:
    mirror_block_device_file
    The block device special file of the mirror disk. The file should refer to the entire disk, not to a partition.

RETURN VALUE
    drd_register_mirror returns one of the following values:
    0  Success.
    1  Error.
    2  Warning.

EXAMPLES
    To display drd_register_mirror usage information:
        drd_register_mirror -?
    To notify DRD that all logical volumes in the root group have been mirrored to disk /dev/dsk/c1t2d0:
        drd_register_mirror /dev/dsk/c1t2d0
    To notify DRD that all logical volumes in the inactive system image have been mirrored to disk /dev/dsk/c2t3d0:
        drd_register_mirror /dev/dsk/c2t3d0

AUTHOR
    drd_register_mirror was developed by HP.

SEE ALSO
    drd-activate(1M), drd-clone(1M), drd-deactivate(1M), drd-mount(1M), drd-runcmd(1M), drd-umount(1M)

NAME

drd_unregister_mirror - notify DRD that an LVM or VxVM mirror of a system image is about to be manually removed

SYNOPSIS

    drd_unregister_mirror [-?] mirror_block_device_file

DESCRIPTION

The drd_unregister_mirror command notifies DRD that a mirror of either the active system image or the inactive system image has been removed or is about to be removed.

The drd_unregister_mirror command can be run before or after the mirror is actually removed, but there are some benefits to running drd_unregister_mirror first: You CANNOT remove the disk that DRD identifies as the boot disk of a system image. The drd_unregister_mirror command will exit with a failure return code if you attempt to do so. If you are not sure which mirrored copy of a system image is identified by DRD as the boot disk, run drd_unregister_mirror to attempt to remove the disk. If you have chosen the boot disk, the command will fail. The other disk can then be unregistered and removed.

Options

drd_unregister_mirror recognizes the following option:

-? Displays the usage message.

Operands

drd_unregister_mirror recognizes the following operand:

    mirror_block_device_file

The block device special file of the mirror disk. The file should refer to the entire disk, not to a partition.

RETURN VALUE

   drd_unregister_mirror returns one of the following values:

   0       Success.
   1       Error.
   2       Warning.

EXAMPLES

To display drd_unregister_mirror usage information:

    drd_unregister_mirror -?

To notify drd that all mirrors of volumes in the root group on the disk /dev/dsk/c1t2d0 have been removed or are about to be removed.

    drd_unregister_mirror /dev/dsk/c1t2d0

To notify drd that all mirrors of volumes in the inactive system image on the disk /dev/dsk/c2t3d0 have been removed or are about to be removed.

    drd_unregister_mirror /dev/dsk/c2t3d0

AUTHOR

    drd_unregister_mirror was developed by HP.

SEE ALSO

drd-activate(1M), drd-clone(1M), drd-deactivate(1M), drd-mount(1M), drd-runcmd(1M), drd-umount(1M)

dump(1M)

NAME
dump, rdump - incremental file system dump, local or across network

SYNOPSIS
/usr/sbin/dump [option [argument ...] filesystem]
/usr/sbin/rdump [option [argument ...] filesystem]

DESCRIPTION
The dump and rdump commands copy to magnetic tape all files in the filesystem that have been changed after a certain date. This information is derived from the files /var/adm/dumpdates and /etc/fstab. option specifies the date and other options about the dump. option consists of characters from the set 0123456789bdfnsuWw. The dump and rdump commands work only on file systems of type hfs. If the given file system is not of type hfs, dump and rdump will abort after printing an error message.

Options

0-9 This number is the "dump level". All files modified since the last date stored in file /var/adm/dumpdates for the same file system at lesser levels will be dumped. If no date is determined by the level, the beginning of time is assumed. Thus, the option 0 causes the entire file system to be dumped.

b The blocking factor is taken from the next argument (default is 10 if not specified). Block size is defined as the logical record size times the blocking factor. dump writes logical records of 1024 bytes. When dumping to tapes with densities of 6250 BPI or greater without using the b option, the default blocking factor is 32.

d The density of the tape (expressed in BPIs) is taken from the next argument. This is used in calculating the amount of tape used per reel. The default value of 1600 assumes a reel tape.

f Place the dump on the next argument file instead of the tape. If the name of the file is -, dump writes to the standard output. When using rdump, this option should be specified, and the next argument supplied should be of the form machine:device.

n Whenever dump and rdump require operator attention, notify all users in group operator by means similar to that described by wall(1).

s The size of the dump tape is specified in feet. The number of feet is taken from the next argument. When the specified size is reached, dump and rdump wait for reels to be changed. The default tape size value of 2300 feet assumes a reel tape.

u If the dump completes successfully, write on file /var/adm/dumpdates the date when the dump started. This file records a separate date for each file system and each dump level. The format of /var/adm/dumpdates is user-readable and consists of one free-format record per line: file system name, increment level, and dump date in ctime(3C) format. The file /var/adm/dumpdates can be edited to change any of the fields if necessary.

W For each file system in /var/adm/dumpdates, print the most recent dump date and level, indicating which file systems should be dumped. If the W option is set, all other options are ignored and dump exits immediately.

w Operates like W, but prints only file systems that need to be dumped.

If no arguments are given, option is assumed to be 9u and a default file system is dumped to the default tape.

Sizes are based on 1600-BPI blocked tape; the raw magnetic tape device must be used to approach these densities. Up to 32 read errors on the file system are ignored. Each reel requires a new process; thus parent processes for reels already written remain until the entire tape is written.

The rdump command creates a server, /usr/sbin/rmt or /etc/rmt, on the remote machine to access the tape device.
dump and rdump require operator intervention for any of the following conditions:

- end of tape,
- end of dump,
- tape-write error,
- tape-open error, or
- disk-read error (if errors exceed threshold of 32).

In addition to alerting all operators implied by the n option, dump and rdump interact with the control terminal operator by posing questions requiring yes or no answers when it can no longer proceed or if something is grossly wrong.

Since making a full dump involves considerable time and effort, dump and rdump each establish a checkpoint at the start of each tape volume. If, for any reason, writing that volume fails, dump and rdump will, with operator permission, restart from the checkpoint after the old tape has been rewound and removed and a new tape has been mounted.

dump and rdump periodically report information to the operator, including typically low estimates of the number of blocks to write, the number of tapes it will require, the time needed for completion, and the time remaining until tape change. The output is verbose to inform other users that the terminal controlling dump and rdump is busy and will be for some time.

Access Control Lists (ACLs)

The optional entries of a file's access control list (ACL) are not backed up with dump and rdump. Instead, the file's permission bits are backed up and any information contained in its optional ACL entries is lost (see acl(5)).

EXAMPLES

In the following example, assume that the file system /mnt is to be attached to the file tree at the root directory, (/). This example causes the entire file system (/mnt) to be dumped on /dev/rmt/c0t0d0BEST and specifies that the density of the tape is 6250 BPI.

```
/usr/sbin/dump 0df 6250 /dev/rmt/c0t0d0BEST /mnt
```

WARNINGS

dump will not backup a file system containing large files.

Tapes created from file systems containing files with UID/GIDs greater than 60,000 will have a new magic number in the header to prevent older versions of restore(1M) from incorrectly restoring ownerships for these files.

AUTHOR

dump and rdump were developed by the University of California, Berkeley.

FILES

- /dev/rdsk/c0d0s0 Default file system to dump from.
- /dev/rmt/0m Default tape unit to dump to.
- /var/adm/dumpdates New format-dump-date record.
- /etc/fstab Dump table: file systems and frequency.
- /etc/group Used to find group operator.

SEE ALSO

restore(1M), rmt(1M), fstab(4), acl(5).
dumpfs(1M)

NAME
dumpfs - dump file system information

SYNOPSIS
/usr/sbin/dumpfs rootdir | special

DESCRIPTION
The dumpfs command prints the super block and cylinder group information for an HFS file system to the standard output. The file system may be specified by its root directory or the name of the device special file on which it resides. The information is very long and detailed. This command can be used to find file system information such as the file system block size or the minimum free space percentage.

DEPENDENCIES
The dumpfs command can only be used on HFS file systems.

AUTHOR
dumpfs was developed by the University of California, Berkeley.

SEE ALSO
fsck(1M), mkfs(1M), newfs(1M), tunefs(1M), disktab(4).
NAME
edquota - edit user disk quotas

SYNOPSIS
/usr/sbin/edquota [-p proto-user] username ...
/usr/sbin/edquota -t

DESCRIPTION
The edquota command is the quota editor. One or more user names can be specified on the command line. For each username, a temporary file is created with a textual representation of the current disk quotas for that user, and an editor is invoked on the file. The quotas can then be modified, new quotas added, and so forth. Upon leaving the editor, edquota reads the temporary file and modifies the binary quota files to reflect the changes made.

The editor invoked is specified by the EDITOR environment variable. It defaults to vi (see vi(1)).

In order for quotas to be established on a file system, the root directory of the file system must contain a file named quotas. See quota(5) for details.

Quotas can be established only for users whose user ID is less than 67,000,000. Attempts to establish quotas for other users result in an error message. This restriction will be removed in a future version of HP-UX.

If you have the HP DiskQuota-Enh product installed, the above restriction for user IDs does not exist for VxFS filesystems that are mounted with largefiles enabled (see mkfs_vxfs(1M)).

Only users who have appropriate privileges can edit quotas.

Options
-p proto_user Duplicate the quotas of the user name proto_user for each username. This is the normal mechanism used to initialize quotas for groups of users.
-t Edit the time limits for each file system. Time limits are set for file systems, not users. When a user exceeds the soft limit for blocks or inodes on a file system, a countdown timer is started and the user has an amount of time equal to the time limit in which to reduce usage to below the soft limit (the required action is given by the quota command). If the time limit expires before corrective action is taken, the quota system enforces policy as if the hard limit had been exceeded. The default time limit of 0 is interpreted to mean the value in <sys/quota.h>, or one week (7 days). Time units of sec(onds), min(utes), hour(s), day(s), week(s), and month(s) are understood. Time limits are printed in the greatest possible time unit such that the value is greater than or equal to one.

Temporary File Formats
Here is an example of the temporary file created for editing user block and inode quotas:

```
fs /mnt blocks (soft = 100, hard = 120) inodes (soft = 0, hard = 0)
fs / blocks (soft = 1000, hard = 1200) inodes (soft = 200, hard = 200)
```

Here is the format for editing quota time limits:

```
fs /mnt blocks time limit = 10.00 days, files time limit = 20.00 days
fs / blocks time limit = 0 (default), files time limit = 0 (default)
```

When editing (default) values, it is not necessary to remove the (default) string. For example, to change the blocks time limit for /, changing the 0 to 4 days is sufficient.

WARNINGS
When establishing quotas for a user who has had none before, (for either blocks or inodes), the quota statistics for that user do not include any currently occupied file system resources. Therefore, it is necessary to run quotacheck (see quotacheck(1M)) to collect statistics for that user's current usage of that file system. See quota(5) for a detailed discussion of this topic.

edquota only edits quotas on local file systems.

AUTHOR
edquota was developed by the University of California, Berkeley, and by Sun Microsystems, Inc.
edquota(1M)

FILES

/etc/fstab
/etc/mnttab
directory /quotas

Static information about the file systems.
Mounted file system table.
Quota statistics static storage for a file system, where directory is the root of the
file system as specified to the mount command (see mount(1M)).

SEE ALSO

quota(1), vi(1), quotacheck(1M), quotacheck_hfs(1M), quotas(5).
NAME
efi_cp - copy to or from EFI file

SYNOPSIS
efi_cp [-d devicefile] [-u] file1 file2
efi_cp [-d devicefile] file1 [file2 ...] dest-directory
efi_cp [-d devicefile] -r file_or_dir1 [file_or_dir2 ...] dest-directory

DESCRIPTION
efi_cp copies files between HP-UX and EFI file systems.
The EFI file system is based on the FAT file system and used by the Itanium-based system BIOS to locate
an HP-UX bootloader. See efi(4).
Without the -u option, efi_cp copies from an HP-UX file system to the EFI volume specified by
devicefile; in this case, the destination should be relative to the root of the EFI volume specified by
devicefile. Use the -u option to copy from an EFI file system to an HP-UX file system; with the -u
option, only a single regular file may be copied at a time.
With two arguments:
• If file2 does not exist, efi_cp creates it and copies the contents of file1 into file2.
• If file2 exists and is a regular file, efi_cp deletes the the contents of file2 and copies the contents of
  file1 into file2.
• If file2 exists and is a directory, efi_cp creates a copy of file1 in that directory.
With more than two arguments, the final argument must be an existing directory. If any of the other argu-
ments is a directory, you must use the -r option, which specifies that the source directory and the subtree
rooted there are to be copied to the destination directory.

Options
Options can be used singly or combined in any order before the file names. The space between option and
argument is optional.
-d devicefile  devicefile is the device special file of the EFI file system to be copied. Without the -d
option, efi_cp uses the device file specified in the environment variable
EFI_PARTITION if it is defined. The -d option overrides the EFI_PARTITION
environment variable.
-r  Recursively copy the subtree rooted at each source directory to the destination directory.
-u  Copy file1 from an EFI volume to file2 in HP-UX file system. file1 should be relative to the
    root of the EFI volume specified by devicefile, not to an HP-UX file system. With the -u
    option, only a single regular file may be copied at a time.

RETURN VALUE
efi_cp returns exit code 0 if the copy is successful. Otherwise it prints a diagnostic message and returns
nonzero.
  0  Successful completion.
  >0  Error condition occurred.

EXAMPLES
All the following examples copy to or from the EFI file system on device /dev/rdsk/c1t4d0s1.
Copy bootprogram into directory /abc/def in the EFI file system on the specified device:
  efi_cp -d /dev/rdsk/c1t4d0s1 bootprogram /abc/def/
Copy all files in current directory into directory /abc/def/ in the EFI file system on the specified device
  efi_cp -d /dev/rdsk/c1t4d0s1 * /abc/def/
Copy bootprogram into directory /abc/def/ in the EFI file system on the specified device and
rename it boot1:
  efi_cp -d /dev/rdsk/c1t4d0s1 bootprogram /abc/def/boot1

224  Hewlett-Packard Company  - 1 -  HP-UX 11i Version 2: December 2007 Update
Copy `bootloader` into the EFI boot directory of the EFI file system on the specified device:

```
efi_cp -d /dev/rdsk/c1t4d0s1 bootloader /EFI/HPUX
```

Copy `bootloader` into `/EFI/HPUX/` of the EFI file system on the device specified in the `EFI_PARTITION` environment variable and rename it to `hpux.efi`:

```
efi_cp bootloader /EFI/HPUX/hpux.efi
```

Copy `/EFI/HPUX/AUTO` from the EFI file system on the specified device into the current HP-UX directory:

```
efi_cp -d /dev/rdsk/c1t4d0s1 -u /EFI/HPUX/AUTO
```

**AUTHOR**

`efi_cp` was developed by HP.

**FILES**

None

**SEE ALSO**

`efi_fsinit(1M), efi_ls(1M), efi_mkdir(1M), efi_rm(1M), efi_rmdir(1M), efi(4)`. 
NAME
  efi_fsinit - write an EFI file system header on a device file

SYNOPSIS
  efi_fsinit [-d devicefile]

DESCRIPTION
  efi_fsinit writes an EFI file system header on a device file.
  The EFI file system is based on the FAT file system and used by the Itanium(R)-based system BIOS to
  locate an HP-UX bootloader. See efi(4).
  Before any other EFI commands can be run, efi_fsinit must be run to initialize the file system on a
  device file.

Options
  efi_fsinit recognizes the following option:
  -d devicefile Write the file system header on device file. Without the -d option, efi_fsinit
    uses the device file specified in the environment variable EFI_PARTITION if it is
    defined. The -d option overrides the EFI_PARTITION environment variable.

RETURN VALUE
  efi_fsinit returns exit code 0 if the EFI file system is initialized successfully. Otherwise it prints a
  diagnostic message and returns nonzero.
  0  Successful completion.
  >0  Error condition occurred.

EXAMPLES
  Write an EFI file system header on device file /dev/rdsk/c1t4d0s1:
    efi_fsinit -d /dev/rdsk/c1t4d0s1

AUTHOR
  efi_fsinit was developed by HP.

FILES
  None

SEE ALSO
  efi_cp(1M), efi_ls(1M), efi_mkdir(1M), efi_rm(1M), efi_rmdir(1M), efi(4).
NAME
  efi_ls - list EFI file information or contents of an EFI directory

SYNOPSIS
  efi_ls [-d devicefile] [path]

DESCRIPTION
  efi_ls lists file information or contents of an EFI directory.
  The EFI file system is based on the FAT file system and used by the Itanium(R)-based system BIOS to
  locate an HP-UX bootloader. See efi(4).

  efi_ls lists filenames, last modified dates, and file sizes. devicefile should be a device file for an EFI
  volume. Given a path, efi_ls determines whether the path is a file or a directory. For a file, efi_ls
  lists the file by itself. For a directory, efi_ls lists the contents of the directory. If no path is given,
  efi_ls lists the root directory for EFI volume. A path should be relative to the root of the EFI volume
  specified by devicefile, not to an HP-UX file system.

  Options
  efi_ls recognizes the following option:

    -d devicefile     devicefile is the device special file of the EFI file system to be listed. Without the -d
                       option, efi_ls uses the device file specified in the environment variable
                       EFI_PARTITION if it is defined. The -d option overrides the EFI_PARTITION
                       environment variable.

RETURN VALUE
  efi_ls returns exit code 0 on success. Otherwise it prints a diagnostic message and returns nonzero.

   0    Successful completion.
   >0   Error condition occurred.

EXAMPLES
  List a root directory of device file /dev/rdsk/c1t4d0s1, which contains an EFI file system:
    efi_ls -d /dev/rdsk/c1t4d0s1

  List the contents of directory /HPUX/EFI of device file /dev/rdsk/c1t4d0s1, which contains an EFI
  file system:
    efi_ls -d /dev/rdsk/c1t4d0s1 /HPUX/EFI

  List information about the file /HPUX/EFI/bootfile of device file /dev/rdsk/c1t4d0s1, which
  contains an EFI file system:
    efi_ls -d /dev/rdsk/c1t4d0s1 /HPUX/EFI/bootfile

AUTHOR
  efi_ls was developed by HP.

FILES
  None

SEE ALSO
  efi_cp(1M), efi_csvin(1M), efi_mkdir(1M), efi_rmdir(1M), efi_rm(1M), efi(4).
NAME
  efi_mkdir - make an EFI directory

SYNOPSIS
  efi_mkdir [-d devicefile] [-p] dirname

DESCRIPTION
  efi_mkdir makes an EFI directory.

  The EFI file system is based on the FAT file system and used by the Itanium(R)-based system BIOS to
  locate an HP-UX bootloader. See efi(4).

  efi_mkdir creates directory dirname in the EFI volume specified by devicefile. If dirname already
  exists, efi_mkdir exits with a diagnostic message, and the directory is not changed. If more than one
  subdirectory needs to be created to make dirname, efi_mkdir exits with a diagnostic message and the
  directory is not changed, unless the -p option is specified. dirname should be relative to the root of the
  EFI volume specified by devicefile, not to an HP-UX file system.

Options
  Options can be used singly or combined in any order before the file names. The space between option and
  argument is optional.

  -d devicefile  devicefile is the device special file of the EFI file system where the directory is to be made.
  Without the -d option, efi_mkdir uses the device file specified in the environment variable EFI_PARTITION if it is defined. The -d option overrides the
  EFI_PARTITION environment variable.

  -p
  Create intermediate directories as necessary.

RETURN VALUE
  efi_mkdir returns exit code 0 if the directory is made successfully. Otherwise it prints a diagnostic mes-
  sage and returns nonzero.

  0   Successful completion.
  >0  Error condition occurred.

EXAMPLES
  Create directory hp-ux in device file /dev/rdsk/c1t4d0s1, which contains an EFI file system:

      efi_mkdir -d /dev/rdsk/c1t4d0s1 hp-ux

  Create a directory /abc/def/ in device file /dev/rdsk/c1t4d0s1, which contains an EFI file sys-
  tem, when /abc does not exist yet:

      efi_mkdir -d /dev/rdsk/c1t4d0s1 -p /abc/def/

AUTHOR
  efi_mkdir was developed by HP.

SEE ALSO
  efi_fsinit(1M), efi_cp(1M), efi_ls(1M), efi_rm(1M), efi_rmdir(1M), efi(4).
efi_rm(1M)

(Itanium(R)-Based Processor Family Only)

NAME
efi_rm - remove an EFI file

SYNOPSIS
efi_rm [-d devicefile] file

DESCRIPTION
efi_rm removes an EFI file.

The EFI file system is based on the FAT file system and used by the Itanium(R)-based system BIOS to locate an HP-UX bootloader. See efi(4).

devicefile should be the device file for the EFI volume.

If file is a directory, efi_rm prints a diagnostic message and returns without removing the directory; use efi_rmdir(1M) to remove EFI directories. file should be relative to the root of the EFI volume specified by devicefile, not to an HP-UX file system.

Options
efi_rm recognizes the following option:

-d devicefile devicefile is the device special file of the EFI file system from which the file is to be removed. Without the -d option, efi_rm uses the device file specified in the environment variable EFI_PARTITION if it is defined. The -d option overrides the EFI_PARTITION environment variable.

RETURN VALUE
efi_rm returns exit code 0 if the file is removed successfully. Otherwise it prints a diagnostic message and returns nonzero.

0 Successful completion.
>0 Error condition occurred.

EXAMPLES
Remove bootfile from /hp-ux/efi/ of device file /dev/rdsk/c1t4d0s1, which contains an EFI file system:

efi_rm -d /dev/rdsk/c1t4d0s1 /hp-ux/efi/bootfile

AUTHOR
efi_rm was developed by HP.

SEE ALSO
efi_fsinit(1M), efi_ls(1M), efi_cp(1M), efi_mkdir(1M), efi_rmdir(1M), efi(4).
NAME
efi_rmdir - remove an EFI directory

SYNOPSIS
efi_rmdir [-d devicefile] directory

DESCRIPTION
efi_rmdir removes an EFI directory

The EFI file system is based on the FAT file system and used by the Itanium(R)-based system BIOS to locate an HP-UX bootloader. See efi(4).

devicefile should be the device file for the EFI volume.

If directory is not a directory or the directory is not empty, efi_rmdir prints a diagnostic message and returns without removing the file; use efi_rm(1M) to remove EFI files. directory should be relative to root of the EFI volume specified by devicefile, not to an HP-UX file system.

Options
efi_rmdir recognizes the following option:

-d devicefile devicefile is the device special file of the EFI file system from which the directory is to be removed. Without the -d option, efi_rmdir uses the device file specified in the environment variable EFI_PARTITION if it is defined. The -d option overrides the EFI_PARTITION environment variable.

RETURN VALUE
efi_rmdir returns exit code 0 if the directory is removed successfully. Otherwise it prints a diagnostic message and returns nonzero.

0 Successful completion.
>0 Error condition occurred.

EXAMPLES
Remove efi from /hp-ux/ of device file /dev/rdsk/c1t4d0s1 with EFI file system:

efi_rmdir -d /dev/rdsk/c1t4d0s1 /hp-ux/efi

AUTHOR
efi_rmdir was developed by HP.

SEE ALSO
efi_fsinit(1M), efi_ls(1M), efi_cp(1M), efi_mkdir(1M), efi_rm(1M), efi(4).
NAME
envd - system physical environment daemon

SYNOPSIS
/usr/sbin/envd [-f configfile]

DESCRIPTION
The envd daemon provides a means for the system to respond to environmental conditions detected by
hardware. Such responses are typically designed to maintain file system integrity and prevent data loss.
The environmental conditions currently recognized by envd are over-temperature and chassis fan failure.

envd logs messages and then executes actions when a supported environmental event is detected.
Whether to do message logging and what actions to perform for a given environmental event are deter-
mined by configfile (default is /etc/envd.conf). If no -f option was specified and the default
configfile /etc/envd.conf does not exist, envd fails. A recommended default configfile is available in
/usr/newconfig/etc/envd.conf. The configfile (or /etc/envd.conf) is only examined when
the daemon is started or when it receives a SIGHUP signal to restart and re-initialize the daemon itself.

envd uses the syslog message logging facility to log warning messages. If configfile specifies messages
to be logged, the destination of the warning messages is determined by the configuration of the
LOG_DAEMON facility of the syslogd daemon (see syslogd(1M) and syslog(3C) for details) and various
syslog priorities defined below for the corresponding environmental events. Warning messages are writ-
ten to the console if envd is unable to send to syslogd.

The configfile is composed of event lines, each of which followed by zero or more action lines. Comment
lines can be interspersed at any point. No more than one event line can be specified for a given event.

Event Event lines consist of an event keyword and a message indicator, separated by a colon
(:). Valid event keywords are OVERTEMP_CRIT, OVERTEMP_EMERG,
FANFAIL_CRIT, and FANFAIL_EMERG. Valid message indicators are y and n.
An example is OVERTEMP_EMERG:y, indicating that warning messages are to be
sent for the OVERTEMP_EMERG event.

Event keywords must start in the first column, and only one event and one message
indicator are allowed on a given line.

Action Action lines can consist of a sequence of any valid /usr/bin/sh commands or pipe-
lines. Lines from one event line to the next event line, or to the end of the file, are
part of the action lines for the preceding event, and are passed intact to the shell to
execute upon detecting the event. The action for an event can span across several
lines, but the syntax of every line must be understood by /usr/bin/sh. There are
no default actions for any events if no action lines are specified.

No parsing or syntax checking is performed on the action lines; system administrators
are responsible for verifying the correctness of the action syntax.

Comments Lines beginning with the # character in the first column are comment lines, and all
characters up to the subsequent new-line character are ignored.
Blank lines are ignored as comment lines.

Here is an example /etc/envd.conf file:

# The example below configures envd to log the warning message and
# to rcp critical applications to a remote machine at OVERTEMP_CRIT
# or FANFAIL_CRIT. It configures envd to log emergency messages
# and to perform system shutdown at OVERTEMP_EMERG or FANFAIL_CRIT,
# in order to reserve data integrity.

OVERTEMP_CRIT:y
/usr/bin/rcp critical_appl_files \
remote_machine:/backup

OVERTEMP_EMERG:y
/usr/sbin/reboot -qh

FANFAIL_CRIT:y
/usr/bin/rcp critical_appl_files \
remote_machine:/backup
FANFAIL_EMERG:
/usr/sbin/reboot -qh

Only users with appropriate privileges can invoke envd.

Over-temperature and Fan Failure Handling
Over-temperature and fan failure handling is supported only on systems equipped with appropriate sensing hardware. Over-temperature and fan failure limits vary, depending on the hardware. Each system processor defines its own thresholds for supported equipment combinations. The table below shows temperature and fan failure states. For the temperature ranges and fan states specific to your system configuration, refer to any of the following documents for your system: Site Planning and Preparation Guide, Installation and Configuration Guide, or Operator Handbook.

<table>
<thead>
<tr>
<th>State</th>
<th>State Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NORMAL</td>
<td>Within normal operating temperature range</td>
</tr>
<tr>
<td>OVERTEMP_CRIT</td>
<td>Temperature has exceeded the normal operating range of the system, but is still within the operating limit of the hardware media.</td>
</tr>
<tr>
<td>OVERTEMP_EMERG</td>
<td>Temperature has exceeded the maximum specified operating limit of hardware media; power loss is imminent. A minimum of about 60 seconds is guaranteed between the OVERTEMP_MID state and the OVERTEMP_POWERLOSS (power loss) state.</td>
</tr>
<tr>
<td>OVERTEMP_POWERLOSS</td>
<td>Hardware will disconnect all power from all cards in the system chassis.</td>
</tr>
<tr>
<td>FAN_NORMAL</td>
<td>All chassis fans are operating normally.</td>
</tr>
<tr>
<td>FANFAIL_CRIT</td>
<td>One or more chassis fans have failed, but the system has enough redundant fans to allow continued operation while the failed fans are replaced.</td>
</tr>
<tr>
<td>FANFAIL_EMERG</td>
<td>Chassis fan failures prevent continued operation of the system; power loss is imminent.</td>
</tr>
<tr>
<td>FANFAIL_POWERLOSS</td>
<td>Hardware will disconnect all power from all cards in the system chassis.</td>
</tr>
</tbody>
</table>

The syslog priorities mapped to the environmental events are: LOG_EMERG (for OVERTEMP_EMERG and FANFAIL_EMERG) and LOG_CRIT (for OVERTEMP_CRIT and FANFAIL_CRIT).

Any non-shutdown activities (e.g. file transfer) should be performed at OVERTEMP_CRIT and FANFAIL_CRIT. It is important to configure only critical activities for OVERTEMP_CRIT because the over-temperature might rise dramatically fast to OVERTEMP_EMERG. It is recommended to perform a quick shutdown using /usr/sbin/reboot -qh at OVERTEMP_EMERG and FANFAIL_EMERG to preserve file system data integrity. If the hardware enters the OVERTEMP_POWERLOSS or FANFAIL_POWERLOSS state and the system has not been shut down, the sudden loss of power could result in data loss. Note that power-fail recovery functionality is not available in this case. When the hardware powers down, no warning messages are produced, and no action is taken by the system.

Whenever an environmental state changes from one level to another (such as from NORMAL to OVERTEMP_CRIT or from FANFAIL_CRIT to FANFAIL_EMERG), the warning message, if specified, is logged, and the corresponding action is executed once, and only once, per state change.

AUTHOR
envd was developed by HP.

FILES
/usr/sbin/envd envd executable file
/etc/envd.conf envd configuration file
/etc/syslog.conf default syslog configuration file
envd(1M) envd(1M)

/var/tmp/envd.action[123]  envd work files

SEE ALSO
reboot(1M), shutdown(1M), syslog(1M), syslog(3C).
HP-UX System Administration manuals.
NAME
exportfs - export and unexport directories to NFS clients

SYNOPSIS
/usr/sbin/exportfs [-auv]
/usr/sbin/exportfs [-uv] [dir ...]
/usr/sbin/exportfs -i [-o options] [-v] [dir ...]

DESCRIPTION
The exportfs command makes a local directory or file available to NFS clients for mounting over the network. Directories and files cannot be NFS-mounted unless they are first exported by exportfs.

exportfs is normally invoked at boot time by the /sbin/init.d/nfs.server script, and uses information contained in the /etc/exports file to export the file or file system named by each dir, which must be specified as a full path name.

If no options or arguments are specified in the command line, exportfs displays a list of the currently exported directories and files on standard output.

A superuser can run exportfs at any time to alter the list or characteristics of exported directories and files.

Options
exportfs recognizes the following options:

-a Export all directories listed in /etc/exports. If -u is also specified, unexport all of the currently exported directories.

-i Ignore the options in /etc/exports. Normally, exportfs consults /etc/exports for the options associated with the exported directory.

-u Unexport the indicated directories.

-v Verbose. Print each directory or file name as it is exported or unexported.

-o options
Specify a comma-separated list of optional characteristics for the directory being exported. The list of options can include any of the following:

async
All NFS Protocol Version 2 mounts will be asynchronous. This option is ignored for NFS PV3. Refer to exports(4) for warnings when using this option.

ro Export the directory read-only. If not specified, the directory is exported read-write. The ro and rw options cannot be used on the same exportfs command line.

rw=hostname[:hostname]...
Export the directory read-mostly. Read-mostly means read-only to most machines, but read-write to those specified. If neither ro nor rw is specified, the directory is exported read-write to all. The ro and rw options cannot be used on the same exportfs command line. Up to 256 hostnames can be specified. With a server configured for DNS naming in the nslookup "hosts" entry, any hostname must be represented as a fully qualified DNS name. Currently HP-UX will attempt to match a non-fully qualified hostname; this HP-only feature will be obsoleted in a later release of HP-UX.

anon=uid
If a request comes from an unknown user, use uid as the effective user ID.

Root users (user ID 0) are always treated as user unknown by the NFS server unless they are included in the root option below.

If the client is a UNIX system, only root users are considered unknown. All other users are recognized even if they are not in /etc/passwd.

The default value for uid is the user ID of user nobody. If user nobody does not exist, the value -2 is used. Setting the value of anon to -1 disables anonymous access.

root=hostname[:hostname]...
Give root access only to the root users from a specified hostname. The default is for no
hosts to be granted root access. Up to 256 *hostnames* can be specified. *hostnames* on this list are not guaranteed to successfully mount the specified file system. If a non-empty *access* list is specified, the *hostname* must also meet one of the *access_list* criteria for *access* or be on the *rw* list. With a server configured for DNS naming in the nsswitch "hosts" entry, any hostname must be represented as a fully qualified DNS name. Currently HP-UX will attempt to match a non-fully qualified hostname; this HP-only feature will be obsoleted in a later release of HP-UX.

```
access=[access_list]|:[access_list]...
```

Give mount access to each *access_list* listed. See the "*access_list*" subsection below. An empty *access* list allows all machines to mount the specified mount point. *hostnames* on the *rw* list do not have to exist on the *access_list* in order to successfully mount the exported file system. *hostnames* on the *root* list must either appear on the *rw* list or *access* list in order to successfully mount the file system.

**access_list**

The *access_list* argument is a colon-separated list whose components may be one or more of the following:

- **hostname**
  The name of a host. With a server configured for DNS naming in the nsswitch "hosts" entry, any hostname must be represented as a fully qualified DNS name. Currently HP-UX will allow a match for a non-fully qualified hostname; this HP-only feature will be obsoleted in a later release of HP-UX.

- **netgroup**
  A netgroup contains a number of hostnames. With a server configured for DNS naming in the nsswitch "hosts" entry, any hostname in a netgroup must be represented as a fully qualified DNS name.

- **DNS suffix**
  To use domain membership, the server must use DNS to resolve hostnames to IP addresses. That is, the "hosts" entry in the */etc/nsswitch.conf* file must specify "dns" ahead of "nis" or "nisplus", since only DNS returns the full domain name of the host. Other name services like NIS or NIS+ cannot be used to resolve hostnames on the server, because when mapping an IP address to a hostname, they do not return domain information. For example, NIS or NIS+

  129.144.45.9 --> "myhost"

  DNS

  129.144.45.9 --> "myhost.myd.myc.com"

  The DNS suffix is distinguished from hostnames and netgroups by a prefixed dot. A dot by itself will match "myhost" but not "myhost.myd.myc.com". This single dot feature can be used to match hosts resolved from NIS and NIS+ rather than DNS.

- **network**
  The network or subnet component is preceded by an at-sign (@). It can be either a name or a dotted address. If a name, it will be converted to a dotted address by *getnetbyname* (see *getnetent*(3N)). Entries in */etc/networks* must contain all four octets in order to be valid.

  The network prefix assumes an octet aligned netmask determined from the zero octets in the low order part of the address. In the case where network prefixes are not byte-aligned, the syntax will allow a mask length to be specified explicitly following a slash (/) delimiter. The mask is the number of leftmost contiguous significant bits in the corresponding IP address.

  - A prefixed minus sign (-) denies access to that component of *access_list*. The list is searched sequentially until a match is found that either grants or denies access, or until the end of the list is reached. This option is valid only in conjunction with hostname, network and DNS Suffix. If prefixing a hostname and you are configured for DNS naming, you must fully qualify the hostname.
exportfs(1M)

DIAGNOSTICS
If an NFS-mounted directory is unexported by exportfs, any access by the client to the directory causes an NFS stale file handle error. However, if exportfs is used to remove a client from the access list of an exported directory, an NFS stale file handle error does not result from any access by the client to the directory.

EXAMPLES
The following invocation of exportfs lists currently exported directories and files:

```
exportfs
```
Export entries in /etc/exports:
```
exportfs -a
```
Unexport all exported files and directories:
```
exportfs -a
```
Unexport all exported files and directories and print each directory or file name as it is unexported:
```
exportfs -uav
```
Export /usr to the world, ignoring options in /etc/exports:
```
exportfs -i /usr
```
or
```
exportfs -i -o access= /usr
```
Export /usr/bin and /var/adm read-only to the world:
```
exportfs -i -o ro /usr/bin /var/adm
```
Export /usr/bin read-write only to systems, polk and vanness, when using DNS as the name service:
```
exportfs -i -o rw=polk.myd.myc.com:vanness.myd.myc.com /usr/bin
```
Export /usr/bin read-write only to systems, polk and vanness, when using NIS or NIS+ as the name services:
```
exportfs -i -o rw=polk:vanness /usr/bin
```
Export root access on /var/adm only to the system named pine, and mount access to both pine and geary when using DNS as the name services:
```
exportfs -i -o root=pine.myd.myc.com,access=pine.myd.myc.com:geary.myd.myc.com
```
Export access to /var/adm for all hosts in the myd.myc.com domain.
```
exportfs -i -o access=.myd.myc.com /var/adm
```
Export access to /var/adm for all hosts in the same NIS domain, but deny access to all hosts in the DNS name space:
```
exportfs -i -o access=. /var/adm
```
Export access to /var/adm using a network submask that is a dotted address:
```
exportfs -i -o access=@192.144 /var/adm
```
or
```
exportfs -i -o access=@192.144.0.0 /var/adm
```
or using a name where mynetwork is defined in /etc/networks as follows:
```
mynetwork 192.144.0.0 mount_144 #allow mounts using this mask
exportfs -i -o access=mynetwork /var/adm
```
Export access to /var/adm where the network prefixes are not byte aligned:


exportfs(1M)

exportfs -i -o access=@192.144.132/17 /var/adm
or
exportfs -i -o access=@mynetwork/17 /var/adm
Export access to /var/adm where the hostname, terra, in the netgroup engineering is denied access:
exportfs -i -o access=-terra:engineering /var/adm
Export access to /var/adm where the hostname, terra, is granted access because it is part of the netgroup engineering.
exportfs -i -o access=engineering:-terra /var/adm

WARNINGS
You cannot export a directory that resides within the same file system and is either a parent or sub-directory of a directory that is currently exported. For example, /usr and /usr/local cannot both be exported if they reside in the same disk partition.
If you unexport a directory, remove a client from the access list, then export again, the client still has access to the directory until the client unmounts the directory. Removing a client from the root or rw list takes effect immediately.

/etc/xtab is a system file that contains a list of currently exported directories and files. This file is maintained by exportfs. To ensure that this file is always synchronous with current system data structures, do not attempt to edit /etc/xtab by hand.

FILES
/etc/exports Static export information
/etc/hosts List of hostnames
/etc/netgroup List of network groups
/etc/xtab Current state of exported directories
/etc/networks Network information

SEE ALSO
showmount(1M), exports(4), hosts(4), netgroup(4), networks(4).
**NAME**
extendfs - extend a file system size (generic)

**SYNOPSIS**
/usr/sbin/extendfs [-F FStype] [-q] [-v] [-s size] special

**DESCRIPTION**
If the original file system image created on special does not make use of all of the available space, **extendfs** can be used to increase the capacity of a file system by updating the file system structure to include the extra space.

The command-line parameter special specifies the device special file of either a logical volume or a disk partition. The special must be un-mounted before **extendfs** can be run (see **mount**(1M)).

**Options**
**extendfs** recognizes the following options:
- **-F FStype**
  Specify the file system type on which to operate (see **fstyp**(1M) and **fs_wrapper**(5)). If this option is not included on the command line, then the file system type is determined from the file /etc/default/fs.
- **-q**
  Query the size of special. No file system extension will be done.
- **-v**
  Verbose flag.
- **-s size**
  Specifies the number of DEV_BSIZE blocks to be added to the file system. If size is not specified, the maximum possible size is used.

**EXAMPLES**
To increase the capacity of a file system created on a logical volume, enter:

```
  umount /dev/vg00/lvol1
  lvextend -L larger_size /dev/vg00/lvol1
  extendfs -F hfs /dev/vg00/rlvol1
  mount /dev/vg00/lvol1 mount_directory
```

**SEE ALSO**
fstyp(1M), lvextend(1M), mkfs(1M), mount(1M), umount(1M), fs_wrapper(5).
NAME
extendfs_hfs: extendfs - extend HFS file system size

SYNOPSIS
/usr/sbin/extendfs [-F hfs] [-q] [-v] [-s size] special

DESCRIPTION
If the original HFS file system image created on special does not make use of all of the available space, the extendfs command can be used to increase the capacity of an HFS file system by updating the file system structure to include the extra space.

The command-line parameter special specifies the character device special file of either a logical volume or a disk partition. The special must be unmounted before the extendfs command can be run (see mount(1M)).

Options
extendfs recognizes the following options:
- `-F hfs` Specify the HFS file system type.
- `-q` Query the size of special. No file system extension will be done.
- `-v` Verbese flag.
- `-s size` Specifies the number of DEV_BSIZE blocks to be added to the file system. If the number of blocks is not specified, the maximum possible size is used.

EXAMPLES
To increase the capacity of a file system created on a logical volume, enter:

```
umount /dev/vg00/lvoll
lvextend -L larger_size /dev/vg00/lvoll
extendfs -F hfs /dev/vg00/rlvoll
mount /dev/vg00/lvoll mount_directory
```

WARNINGS
The root file system cannot be extended using the extendfs command because the root file system is always mounted, and the extendfs command only works on unmounted file systems.

extendfs will fail if used on a file system, on a logical volume, where the logical block size of the logical volume is greater than the file system's fragment size. The logical block size, of a logical volume changes, when additional disks with larger sector size are added.

RETURN VALUE
extendfs returns the following values:
- `0` No errors were detected and file system was successfully extended.
- `1` Command aborted.

SEE ALSO
extendfs(1M), lvextend(1M), mkfs(1M), mount(1M), umount(1M).
NAME
extendfs_vxfs: extendfs - extend VxFS file system size

SYNOPSIS
extendfs [-F vxfs] [-q] [-v] [-s size] special

DESCRIPTION
If the VxFS file system image created on special does not use all of the available space, extendfs increases the capacity of a VxFS file system by updating the file system structure to include the extra space.

special specifies the device special file of either a logical volume or a disk partition. If special refers to a mounted file system, you must unmount special before running extendfs (see mount(1M)).

Options
extendfs recognizes the following options:
-F vxfs
Specify the VxFS file system type.
-q
Query special to determine the size. The file system is not extended.
-s size
Specify the number of DEV_BSIZE blocks to add to the file system. If size is omitted, the maximum possible size is used.
-v
Specify verbose mode, which displays the resulting size of file system along with fsck output of the newly sized file system. Without -v there is no output.

Operands
extendfs recognizes the following operand:
special
The device special file of either a logical volume or a disk partition. If special refers to a mounted file system, you must unmount special before running extendfs (see mount(1M)).

Notes
The Version 5 disk layout supports file systems up to 32 terabytes. For you to increase a file system to 32 terabytes, it must be on a 64-bit kernel operating system and must reside on a VERITAS Volume Manager volume. The size to which a Version 5 disk layout file system can be increased depends on the file system block size:

- 1024 bytes 4,294,967,039 sectors (4 TB)
- 2048 bytes 8,589,934,078 sectors (8 TB)
- 4096 bytes 17,179,868,156 sectors (16 TB)
- 8192 bytes 34,359,736,312 sectors (32 TB)

EXAMPLES
This example shows how to increase the capacity of a file system created on a logical volume.

umount /dev/vg00/lvol1
lvextend -L larger_size /dev/vg00/lvol1
extendfs -F vxfs /dev/vg00/rlvol1
mount /dev/vg00/lvol1 mount_directory

SEE ALSO
extendfs(1M), lvextend(1M), mkfs(1M), mount(1M), umount(1M), fs_vxfs(4).
fbackup(1M)

NAME
fbackup - selectively back up files

SYNOPSIS
/usr/sbin/fbackup -f device [-f device] ... [-0-9] [-nsuyvAEl] [-i path] [-e path]
[-g graph] [-d path] [-I path] [-V path] [-c config]

fbackup

DESCRIPTION
fbackup combines features of dump and ftio to provide a flexible, high-speed file system backup mechanism (see dump(1M) and ftio(1)). fbackup selectively transfers files to an output device. For each file transferred, the file's contents and all the relevant information necessary to restore it to an equivalent state are copied to the output device. The output device can be a raw magnetic tape drive (for example, a DLT tape drive), the standard output, a rewritable magneto-optical disk, or a file.

The selection of files to back up is done by explicitly specifying trees of files to be included or excluded from an fbackup session. The user can construct an arbitrary graph of files by using the -I or -e options on the command line, or by using the -g option with a graph file. For backups being done on a regular basis, the -g option provides an easier interface for controlling the backup graph. fbackup selects files in this graph and attempts to transfer them to the output device. The selectivity depends on the mode (full or incremental) in which fbackup is being used.

When doing full backups, all files in the graph are selected. When doing incremental backups, only files in the graph that have been modified since a previous backup of that graph are selected. If an incremental backup is being done at level 4 and the -g option is used, the database file is searched for the most recent previous backup at levels 0-3. If a file's modification time is before the time when the last appropriate session began and the i-node change time is before the time that same session ended, the file is not backed up. All directories lying on the path to a file that qualifies for the incremental backup will also be on the backup media, even if the directories do not qualify on their own status.

If fbackup is used for incremental backups, a database of past backups must be kept. fbackup maintains this data in the text file /var/adm/fbackupfiles/dates, by default. Note that the directory /var/adm/fbackupfiles must be created prior to the first time fbackup is used for incremental backups. The -d option can be used to specify an alternate database file. The user can specify to update this file when an fbackup session completes successfully. Entries for each session are recorded on separate pairs of lines. The following four items appear on the first line of each pair: the graph file name, backup level, starting time, and ending time (both in strftime(3C) format). The second line of each pair contains the same two times, but in strptime(3C) format. These lines contain the local equivalent of STARTED: the start time, the local equivalent of ENDED: the ending time. These second lines serve only to make the dates file more readable; fbackup does not use them. All fields are separated by white space. Graph file names are compared character-by-character when checking the previous-backup database file to ascertain when a previous session was run for that graph. Caution must be exercised to ensure that, for example, graph and ./graph are not used to specify the same graph file because fbackup treats them as two different graph files.

The general structure of an fbackup volume is the same, no matter what type of device is used. There are some small specific differences due to differing capabilities of devices. The general structure is as follows:

- reserved space for ASCII tape label (1024 bytes)
- fbackup volume header (2048 bytes)
- session index (size in field of volume header)
- data

Each file entry in the index contains the file size, the volume number and the pathname of the file. At the beginning of every volume, fbackup assumes that all files not already backed up will fit on that volume, an erroneous assumption for all but the last volume. Indices are accurate only for the previous volumes in the same set. Hence, the index on the last volume may indicate that a file resides on that volume, but it may not have actually been backed up (for example, if it was removed after the index was created, but before fbackup attempted to back it up). The only index guaranteed to be correct in all cases is the on-line index (-i option), which is produced after the last volume has been written.

Specific differences in the structure of fbackup volumes are listed below:
When using magnetic tape devices, the main blocks of information (tape label, volume header, index, data) are separated by EOF marks. `fbackup` also checkpoints the media periodically to enhance error recovery. If a write error is detected, the user normally has two options: (1) a new volume can be mounted and that volume rewritten from the beginning; or, (2) if the volume is not too severely damaged, the good data before the error can be saved, and the write error is treated as a normal end-of-media condition. The blocks of data with their checkpoint records are also separated by EOF marks. In addition, for DDS tape drives, if fast search marks are supported, these will be used to enhance selective recovery speed by placing them between blocks of files. Similarly, on DLT tape drives, faster selective recovery is achieved using the EOF marks used for checkpointing in conjunction with the file sizes given in the index.

For a magneto-optical device, a disk, a file, or standard output, there are no special marks separating the information pieces; the backup is always a single file (volume).

`fbackup` provides the ability to use UCB-mode tape drives. This makes it possible to overlap the tape rewind times if two or more tape drives are connected to the system.

Set-up

There are several things the user will want to consider when setting up `fbackup` for regular use. These include type of device and media, full versus incremental frequency, amount of logging information to keep on-line, structure of the graph file, and on-line versus off-line backup.

The type of device used for backups can affect such things as media expenses, ability to do unattended backups, and speed of the backup. Using 36-track tapes will probably result in the highest performance, but require user intervention for changing tapes. Both DLT and DDS autochangers and libraries can provide unattended backups. A magneto-optical autochanger can also provide an unattended backup for a large system and long life media, however the media cost is high. Lower cost and good performance can be achieved with a single DLT tape drive, but multi-volume backups must be attended.

It is also important to consider how often full backups should be made, and how many incremental backups to make between full backups. Time periods can be used, such as a full backup every Friday and incrementals on all other days. Media capacities can be used if incremental backups need to run unattended. The availability of personnel to change media can also be an important factor as well as the length of time needed for a full backup. Other factors may affect the need for full and incremental backup combinations such as contractual or legal requirements.

If backup information (output from the `-V` or `-I` options) is kept on-line, the required storage space must also be considered. Index file sizes are hard to predict in advance because they depend on system configuration. Each volume header file takes less than 1536 bytes. Of course the more information that is kept on-line, the faster locating a backup media for a recovery will be.

There are several ways to structure the graph file or files used in a system backup. The first decision involves whether to use one or more than one graph file for the backup. Using one file is simpler, but less flexible. Using two or more graph files simplifies splitting backups into logical sets. For example, one graph file can be used for system disks where changes tend to be less frequent, and another graph file for the users area. Thus two different policies can be implemented for full and incremental backups.

`fbackup` was designed to allow backups while the system is in use by providing the capability to retry an active file. When absolute consistency on a full backup is important, the system should probably be in single-user mode. However, incremental backups can be made while the system is in normal use, thus improving system up-time.

Options

```
-c config
```

`config` is the name of the configuration file, and can contain values for the following parameters:

- Number of 1024-byte blocks per record.
- Number of records of shared memory to allocate.
- Number of records between checkpoints. Since the EOF marks between checkpoints are also used for fast searching on DLT tape drives, changing the checkpoint frequency may also affect selective recovery speed (see WARNINGS section).
- Number of file-reader processes.
- Maximum number of times `fbackup` is to retry an active file.
- Maximum number of bytes of media to use while retrying the backup of an active file.
Maximum number of times a magnetic tape volume can be used.
- Name of a file to be executed when a volume change occurs. This file must exist and be executable.
- Name of a file to be executed when a fatal error occurs. This file must exist and be executable.
- The number of files between the **fast search marks** on DDS tapes. The cost of these marks are negligible in terms of space on the DDS tape. Not all DDS tape devices support **fast search marks**.

Each entry in the configuration file consists of one line of text in the following format: identifier, white space, argument. In the following sample configuration file, the number of blocks per record is set to 16; the number of shared memory records is set to 16; the checkpoint frequency is set to 256; the number of file reader processes is set to 2; the maximum number of retries of an active file is set to 5; the maximum retry space for active files is set to 5,000,000 bytes; the maximum number of times a magnetic tape volume can be used is set to 100; the file to be executed at volume change time is `/var/adm/fbackupfiles/chgvol`; the file to be executed when a fatal error occurs is `/var/adm/fbackupfiles/error`; and the number of files between **fast search marks** on DDS tapes is set to 200.

```
blocksperrecord 16
records 16
checkpointfreq 256
readerprocesses 2 (maximum of 6)
maxretries 5
retrylimit 5000000
maxvoluses 100
chgvol /var/adm/fbackupfiles/chgvol
error /var/adm/fbackupfiles/error
filesperfsm 200
```

Each value listed is also the default value, except `chgvol` and `error`, which default to null values.

- `-d path` This specifies a path to a database for use with incremental backups. It overrides the default database file `/var/adm/fbackupfiles/dates`.

- `-e path` *path* specifies a tree to be excluded from the backup graph. This tree must be a subtree of part of the backup graph. Otherwise, specifying it will not exclude any files from the graph. There is no limit on how many times the `-e` option can be specified.

- `-f device` *device* specifies the name of an output file. If the name of the file is `-`, `fbackup` writes to the standard output. There is no default output file; at least one must be specified. If more than one output file is specified, `fbackup` uses each one successively and then repeats in a cyclical pattern. Patterns can be used in the device name in a manner resembling file name expansion as done by the shell (see `sh(1)` and other shell manual entries). The patterns must be protected from expansion by the shell by quoting them. The expansion of the pattern results in all matching names being in the list of devices used.

There is slightly different behavior if remote devices are used. A device on the remote machine can be specified in the form `machine:device`. `fbackup` creates a server process from `/usr/sbin/rmt` on the remote machine to access the tape device. If `/usr/sbin/rmt` does not exist on the remote system, `fbackup` creates a server process from `/etc/rmt` on the remote machine to access the tape device. Only magnetic tapes can be remote devices. When remote DDS tape devices are used, the **fast search marks** capability is not used.

- `-g graph` *graph* defines the graph file. The graph file is a text file containing the list of file names of trees to be included or excluded from the backup graph. These trees are interpreted in the same manner as when they are specified with the `-i` and `-e` options. Graph file entries consist of a line beginning with either `i` or `e`, followed by white space, and then the path name of a tree. Lines not beginning with `i` or `e` are treated as an error. There is no default graph file. For example, to back up all of `/usr` except for the subtree `/usr/lib`, a file could be created with the following two records:

```
fbackup(1M)  

-i path  
path specifies a tree to be included in the backup graph. There is no limit on how many times the -i option can be specified.

-n  
Cross NFS mount points. By default, fbackup does not cross NFS mount points, regardless of paths specified by the -i or -g options.

-l  
Includes LOFS files specified by the backup graph. By default, fbackup does not cross LOFS mount points. If -l is specified, and the backup graph includes files which are also in an LOFS directory that is in the backup graph, then those files will be backed up twice.

-s  
Back up the object that a symbolic link refers to. The default behavior is to back up the symbolic link.

-u  
Update the database of past backups so that it contains the backup level, the time of the beginning and end of the session, and the graph file used for this fbackup session. For this update to take place, the following conditions must exist: Neither the -i nor the -e option can be used; the -g option must be specified exactly once (see below); the fbackup must complete successfully.

-v  
Run in verbose mode. Generates status messages that are otherwise not seen.

-y  
Automatically answer yes to any inquiries.

-A  
Do not back up optional entries of access control lists (ACLs) for files. Normally, all mode information is backed up including the optional ACL entries. With the -A option, the summary mode information (as returned by stat()) is backed up. Use this option when backing up files from a system that contains ACLs to be recovered on a system that does not understand ACLs (see acl(5)).

-E  
Do not back up extent attributes. Normally, all extent attributes that have been set are included with the file. This option only applies to file systems which support extent attributes.

-I path  
path specifies the name of the on-line index file to be generated. It consists of one line for each file backed up during the session. Each line contains the file size, the volume number on which that file resides, and the file name. If the -I option is omitted, no index file is generated.

-V path  
The volume header information is written to path at the end of a successful fbackup session. The following fields from the header are written in the format label:value with one pair per line.

Magic Field  
On a valid fbackup media it contains the value FBACKUP_LABEL (HP-UX release 10.20 and beyond). Before HP-UX release 10.20, it contained the value FBACKUP_LABEL.

Machine Identification  
This field contains the result of uname -m.

System Identification  
This field contains the result of uname -s.

Release Identification  
This field contains the result of uname -r.

Node Identification  
This field contains the result of uname -n.

User Identification  
This field contains the result of cuserid() (see cuserid(3S)).

Record Size  
This field contains the maximum length in bytes of a data record.

Time  
This field contains the clock time when fbackup was started.

Media Use  
This field contains the number of times the media has been used for backup. Since the information is actually on the media, this field will always contain the value 0.

Volume Number  
This field contains a # character followed by 3 digits, and identifies the number of volumes in the backup.

Checkpoint Frequency  
This field contains the number of data records between checkpoints.
fbackup(1M)

Index Size
This field contains the size of the index.

Backup Identification Tag
This field is composed of two items: the process ID (pid) and the
start time of that process.

Language
This field contains the language used to make the backup.

-R restart
Restart an fbackup session from where it was previously interrupted. The restart file
contains all the information necessary to restart the interrupted session. None of the -
[iego0-9] options can be used together with the restart option.

-0-9
This single-digit number is the backup level. Level 0 indicates a full backup. Higher levels
are generally used to perform incremental backups. When doing an incremental backup of
a particular graph at a particular level, the database of past backups is searched to find the
date of the most recent backup of the same graph that was done at a lower level. If no such
entry is found, the beginning of time is assumed. All files in the graph that have been
modified since this date are backed up.

Access Control Lists (ACLs)
If a file has optional ACL entries, the -A option is required to enable its recovery on a system where
the ACL capability is not present.

EXTERNAL INFLUENCES
Environment Variables
- LC_COLLATE determines the order in which files are stored on the backup device and the order of output
  by the -I option.
- LC_TIME determines the format and contents of date and time strings.
- LC_MESSAGES determines the language in which messages are displayed.

If LC_COLLATE, LC_TIME, and LC_MESSAGES are not all specified in the environment, or if either is
set to the empty string, the value of LANG is used as a default for each unspecified or empty variable. If
LANG is not specified or is set to the empty string, a default of "C" (see lang(5)) is used instead of LANG. If
any internationalization variable contains an invalid setting, fbackup behaves as if all internationaliza-
tion variables are set to "C". See environ(5).

International Code Set Support
Single- and multi-byte character code sets are supported.

RETURN VALUE
fbackup returns one of the following values:
0 upon normal completion.
1 if it is interrupted but allowed to save its state for possible restart.
2 if any error conditions prevent the session from completing.
4 if any warning conditions are encountered.

If warnings occur, the operator should check the fbackup logs to verify the sanity of the backup.

EXAMPLES
In the following two examples, assume the graph of interest specifies all of /usr except /usr/lib (as
described for the -g option above).

The first example is a simple case where a full backup is done but the database file is not updated. This can
be invoked as follows:

/usr/sbin/fbackup -0i /usr -e /usr/lib -f /dev/rmt/c0t0d0BEST

The second example is more complicated, and assumes the user wants to maintain a database of past
fbackup sessions so that incremental backups are possible.
If sufficient on-line storage is available, it may be desirable to keep several of the most recent index files on
disk. This eliminates the need to recover the index from the backup media to determine if the files to be
recovered are on that set. One method of maintaining on-line index files is outlined below. The system
administrator must do the following once before fbackup is run for the first time (creating intermediate
level directories where necessary):
**fbackup(1M)**

- Create a suitable configuration file called `config` in the directory `/var/adm/fbackupfiles`.
- Create a graph file called `usr-usrlib` in the directory `/var/adm/fbackupfiles/graphs`.
- Create a directory called `usr-usrlib` in the directory `/var/adm/fbackupfiles/indices`.

A shell script that performs the following tasks could be run for each `fbackup` session:

- Build an index file path name based on both the graph file used (passed as a parameter to the script) and the start time of the session (obtained from the system). For example:
  ```bash
  /var/adm/fbackupfiles/indices/usr-usrlib/871128.15:17
  ```
  (for Nov 28, 1987 at 3:17 PM)
- Invoke `fbackup` with this path name as its index file name. For example:
  ```bash
  cd /var/adm/fbackupfiles
  /usr/sbin/fbackup -0uc config -g graphs/usr-usrlib\n  -I indices/usr-usrlib/871128.15:17\n  -f /dev/rmt/c0t0d0BEST
  ```

When the session completes successfully, the index is automatically placed in the proper location.

**WARNINGS**

- `fbackup` consists of multiple executable objects, all of which are expected to reside in directory `/usr/sbin`.
- `fbackup` does not require special privileges. However, if the user does not have access to a given file, the file is not backed up.

For security reasons, configuration files and the `chgvol` and `error` executable files should only be writable by their owners.

With release 10.20, HP-UX supports large files (greater than 2GB) and increased UID/GIDs (greater than 60,000). Archives containing files with these attributes would cause severe problems on systems that do not support the increased sizes. For this reason, `fbackup` creates tapes with a new magic number (“FBACKUP_LABEL”). This prevents `fbackup` tape archives from being restored on pre-10.20 HP-UX systems. `frecover` still reads both tape formats so that `fbackup` tape archives created on pre-10.20 HP-UX systems can be restored.

EOF marks are used for checkpointing on all magnetic tape devices. On DLT tape devices, these EOF marks are also used for fast searching on a selective recovery; “fast searching” in this case means spacing to the nearest checkpoint before the desired file, and then reading until the file is found. With this dual purpose for checkpoints, caution should be used when changing the checkpoint frequency parameter.

Starting with HP-UX Release 8.0, `fbackup` does not back up network special files because RFA networking is obsolete. A warning message is issued if a network special file is encountered in the backup graph and the file is skipped.

The use of `fbackup` for backing up NFS mounted file systems is not guaranteed to work as expected if the backup is done as a privileged user. This is due to the manner in which NFS handles privileged-user access by mapping user `root` and uid 0 to user `nobody`, usually uid `-2`, thus disallowing root privileges on the remote system to a root user on the local system.

The utility set comprised of `fbackup` and `frecover` was originally designed for use on systems equipped with not more than one gigabyte of total file system storage. Although the utilities have no programming limitations that restrict users to this size, complete backups and recoveries of substantially larger systems can cause a large amount of system activity due to the amount of virtual memory (swap space) used to store the indices. Users who want to use these utilities, but are noticing poor system-wide performance due to the size of the backup, are encouraged to back up their systems in multiple smaller sessions, rather than attempting to back up the entire system at one time.

Due to present file-system limitations, files whose inode data, but not their contents, are modified while a backup is in progress might be omitted from the next incremental backup of the same graph. Also, `fbackup` does not reset the inode change times of files to their original values.

`fbackup` should not be used with no-rewind devices, for example, `/dev/rmt/0mn`.

`fbackup` allocates resources that are not returned to the system if it is killed in an ungraceful manner. If it is necessary to kill `fbackup`, send it a `SIGTERM`, not a `SIGKILL`.
If sparse files are backed up without using data compression, a very large amount of media can be consumed.

`fbackup` creates volumes with a format that makes duplication of volumes by `dd` impossible (see `dd(1)`). Copying an `fbackup` volume created on one media type to another media type does not produce a valid `fbackup` volume on the new media because the formats of volumes on raw magnetic tape, on a regular file, and on rewritable optical disks are not identical.

When configuring the parameter `blocksperrecord` (see `-c` option), the record size is limited by the maximum allowed for the tape drive. Common record sizes include 128 blocks for DLT and DDS tape drives, and 60 blocks for the HP 7980. Note also that the `blocksize` used in earlier releases (7.0 and before) was 512 bytes, whereas it is now 1024 bytes. This means that the same value specified in `blocksperrecord` in an earlier release creates blocks twice their earlier size in the current release; for example, a `blocksperrecord` parameter of 32 would create 16-Kbyte blocks at Release 7.0, but now creates 32-Kbyte blocks. If `blocksperrecord` exceeds the byte count allowed by the tape drive, the tape drive rejects the write, causing an error to be communicated to `fbackup` which `fbackup` interprets as a bad tape. The resulting write error message resembles the following:

```
fbackup (3013): Write error while writing backup at tape block 0.  
Diagnostic error from tape 11...... SW_PROBLEM (printed by driver on console)  
fbackup (3102): Attempting to make this volume salvageable.  
etc.
```

**DEPENDENCIES**

**NFS**
Access control lists of networked files are summarized (as returned in `st_mode` by `stat()`), but not copied to the new file (see `stat(2)`).

`fbackup` does not support QIC-120 and QIC-150 formats on QIC devices. If `fbackup` is attempted for these formats, `fbackup` fails and the following message is displayed:

```
mt lu X: Write must be a multiple of 512 bytes in QIC 120 or QIC 150
```

**AUTHOR**

`fbackup` was developed by HP.

**FILES**

```
/var/adm/fbackupfiles/dates  database of past backups
```

**SEE ALSO**

`cpio(1)`, `ftio(1)`, `dump(1M)`, `frecover(1M)`, `restore(1M)`, `rmt(1M)`, `stat(2)`, `acl(5)`, `mt(7)`.
NAME
fcmsutil - Fibre Channel Mass Storage Utility Command for Fibre Channel Host Bus Adapters

SYNOPSIS
/opt/fcms/bin/fcmsutil device_file [options]

The device_file specifies the Fibre Channel device special file associated with the Fibre Channel HBA port.

The device file has the format /dev/FC_driverX, where X is the instance number of the Fibre Channel HBA port, as reported by the ioscan output.

The [options] that follow the device file are the ones listed in the fcmsutil display usage for each type of Fibre Channel driver. The usage of fcmsutil can be seen by running the command alone. Options listed under the fcmsutil usage for different HBAs are supported for that HBA.

In case an invalid device file is specified, that is, if the device file does not belong to the fc class, the fcmsutil command will return an error and display the usage options of fcmsutil.

Note: For all options that take remote-N-Port-ID as an argument, remote-N-Port-ID can be replaced with -l loop_id (in Private Loop), or -w wwn.

SECURITY RESTRICTIONS
The usage of fcmsutil command is restricted to processes having super user privileges. Some of the options require detailed knowledge of the device specific adapter. See privileges(5) for more information about privileged access on systems that support fine-grained privileges.

DESCRIPTION
For a list of all supported Fibre Channel HBAs, please refer to the HP-UX Fibre Channel HBA Support Matrix available at:

The fcmsutil command is a common diagnostic tool used for all Fibre Channel Host Bus Adapters supported on HP-UX. This command provides the ability to perform Fibre Channel Test and Echo functionality, read the card’s registers, and so on. This command requires the use of a device file to indicate the interface over which the requested command needs to be performed.

Options
fcmsutil options are described below. Some options are HBA-specific and, therefore, are not supported by all Fibre Channel HBAs.

The list of options supported by each HBA may be obtained by running fcmsutil without specifying any arguments.

All keywords are case-insensitive and are position dependent.

device_file Can be used alone or with other options.

When used without any options it provides information such as the N_Port ID, Node World Wide Name and Port World Wide Name of the HBA and the switch (when applicable), Topology, the negotiated Link Speed, Vendor ID, Device ID, the Driver State, Hardware Path, Maximum Frame Size, and Driver Version.

As per the FC protocol, Node WNN will be numerically one more than the Port WNN.

The following topologies are defined:

UNINITIALIZED.
There is no connectivity to the host bus adapter port or the topology could not be determined.

UNKNOWN.
The Fibre Channel driver has returned a topology code that the utility does not understand.

PRIVATE_LOOP.
The host bus adapter is attached to a loop/loopback hood.

PUBLIC_LOOP.
The host adapter is attached to a loop connected to a Fibre Channel switch.
**IN_PTOPT_NPORT/PTTOPT_NPORT.**
The host bus adapter has come up in a point to point topology. This topology can
be an error if the card was expected to come up in loop topology. Not legal in most
cases.

**PTTOPT_FABRIC.**
The host bus adapter has come up in a point to point topology when connected
through a Fibre Channel switch. This topology can be an error if the card was
expected to come up in loop topology.

The following driver states are defined:

**LOOPBACK_STATE.**
The host bus adapter is in the loop back test phase.

**OFFLINE/DISABLED.**
The host bus adapter is not participating on the loop. This state is the result of
user disabling the card through the disable option in `fcmsutil` or the Fibre
Channel driver being unable to recover from an error.

**READY/ONLINE.**
The driver is up and functional.

**RESETTING.**
The host bus adapter is being reset.

**SUSPENDED.**
The driver has been suspended by the user.

**AWAITING_LINK_UP.**
The driver is waiting for the Fibre Channel link to come up. There is no connec-
tivity to the HBA port.

All other states are only transient and should not continue for long. If the transient state
persists, there might be a problem in the hardware connectivity or configuration.

The following link speeds are defined:

**UNINITIALIZED or UNKNOWN.**
The host bus adapter not could converge to a common link speed or adapter is not
connected.

1Gb. The link is online and the operating speed is 1 gigabits per second.

2Gb. The link is online and the operating speed is 2 gigabits per second.

4Gb. The link is online and the operating speed is 4 gigabits per second.

**vpd**
This option is used to display "Vital Product Data" information of the HBA. It includes
information such as the product description, part number, engineering date code, part
serial number, and so on. This option is not supported by the TACHYON TL HBAs.

**echo remote-N-Port-ID data_size [count]**
This option is used to send an ECHO ELS frame on the wire. It requires two parameters,
the remote-N-Port-ID and data-size (size of packet to send). An optional third argument
(count) can be specified for the number of echo packets to be sent. If the count option is not
specified, one packet will be sent.

Fibre Channel Echo packets of the specified size are sent to the remote node. The com-
mand completes successfully when an echo response is received from the remote node and
matches the data sent, for all packets sent. The command times out if a response is not
received in twice RA_TOV time. Echo packets cannot be sent in a PUBLIC_LOOP topol-
ygy.

Note: Packet size specified must be a multiple of 4.

**rls remote-N-Port-ID**
This option is used to send an RLS (Request Link Status) ELS frame on the wire. It
requires one parameter, the remote-N-Port-ID. The ELS is sent to this remote-N-Port-ID
and the response data is displayed.
**test** remote-N-Port-ID data_size [count]

This option is used to send a TEST ELS on the wire. It requires two parameters, the remote-N-Port-ID and data_size (size of packet to send). An optional third argument (count) can be specified for the number of echo packets to be sent. If the count option is not specified, one packet will be sent.

The command completes successfully and immediately on sending all the test packets.

Note: Packet size specified must be a multiple of 4.

**read** offset [pci]

This option is used to read from HBA's internal registers. It requires one parameter, the offset of the register to read from. The offset can be specified in either hex or in decimal format. The offset specified is an offset from the base of the Memory Map. The user of this command is therefore expected to have internal knowledge of the chip. Reading from the TACHYON frame manager status register (0x01c8) is restricted.

An optional second argument (pci) can be specified for Fibre Channel HBAs, to read from the PCI config space. If no second argument is specified, it reads from the chip register space.

**write** offset value [pci]

This option is used to write into HBA's registers. It requires two parameters, the offset of the register to write to and the value to be written.

An optional third argument (pci) can be specified for the Fibre Channel HBAs, to write into the PCI config space. If no third argument is specified, it writes into the chip register space.

**[-f] lb**

This option is used to perform loopback tests on the port.

Warning: This is a DESTRUCTIVE test, and DATA LOSS during the execution of this test may occur.

The -f option can be used to suppress the warning message displayed by the Fibre Channel driver utility.

For TL and XL2 HBAs, this option requires one parameter and an optional count:

[-f] lb {plm | crpat | cjtpat} [count]

Here plm refers to physical link module or gigabit link module, which builds the default payload for the loopback frame. If either crpat or cjtpat is used, then the card builds specific payloads based on the recommendations in *Fibre Channel - Methodologies for Jitter Specifications*. These patterns are designed to generate bit patterns which stress the transmit and receive channels of the card. The self test then involves sending a packet and receiving back the packet within the adapter and checking its integrity. Since this self test is at the adapter level, no packet goes on the fibre link.

All Fibre Channel HBAs (except TL and XL2) need to specify two parameters. Here is the syntax:

[-f] lb {ext | int} | crpat | cjtpat} [count]

The first parameter should be either ext or int to specify whether the loopback should be external or internal, respectively. The second parameter specifies the loopback pattern. Only crpat and cjtpat options are supported for these cards. Frames are looped back at the single bit interface in the Internal loopback mode. For external loopback, frames are sent out and received from the wire. External loopback mode is supported only in Loop topology.

NOTE: In the internal loopback mode, frames are also sent out on the wire even though they are internally looped back at the 1 Bit interface. The receiver, however, is turned off during this operation. Therefore, it is not safe to run Internal loopback tests when the fiber is connected to a Switch or Hub, as the transmitted loopback frames can disrupt operation on the SAN.

An optional third argument (count) can be specified for the number of loopback packets to be sent. If the count option is not specified, one packet will be sent.
get local
get fabric
get remote {all | remote-N-Port-ID}

This option is used to obtain Fibre Channel login parameters of either the local port, the fabric port, or a remote port. The Fibre Channel HBAs do not support the local option. If the all argument is specified for the remote option, login parameters and current states of all N_Ports that the initiator is aware of, are displayed.

[-f] reset

This option is used to reset the HBA (or a single FC port in case of multi-port HBAs).
WARNING: This is a DESTRUCTIVE test. The reset operation will result in aborting communication to all nodes till the process is completed.

The -f option can be used to suppress the warning message displayed by the Fibre Channel driver utility.

[-f] bdr target-device_file

This option is used to issue a Bus Device Reset to device.
WARNING: This is a DESTRUCTIVE test.

The -f option can be used to suppress the warning message displayed by the Fibre Channel driver utility.

This option resets the target, clearing all commands, without doing any checks.

read_cr

This option can be used to read all of the readable registers on the card and format the detailed information.

stat [-s]

This option is used to obtain detailed statistics maintained by the driver. An optional argument -s can be specified to obtain a shortened version of the statistics maintained by the driver. Generally, the link statistics for the HBA port is displayed.

clear_stat

This option is used to clear the statistics maintained by the driver.

nsstat

This option is used to obtain detailed nameserver statistics maintained by the driver.

clear_nsstat

This option is used to clear the nameserver statistics maintained by the driver.

devstat {all | remote-N-Port-ID}

This option is used to obtain detailed statistics associated with each N_Port that this N_Port has communicated with.

If the remote-N-Port-ID is specified, then the statistics associated with that N_Port are displayed. If the all option is specified, statistics associated with all N_Ports that the initiator has been able to communicate with are displayed. Along with the statistics for each N_Port, it also displays the loop_id (in Private Loop) and the port_id (in Fabric).

clear_devstat {all | remote-N-Port-ID}

This option is used to clear the statistics associated with a target.

If the remote-N-Port-ID is specified, then the statistics associated with that N_Port are cleared. If the all option is specified, statistics associated with all valid N_Port_IDS are cleared.

replace_dsk remote-N-Port-ID

This option is used to specify that no authentication should be performed the next time we communicate with the device. This option is to be used by system administrators for replacing a disk, with another (with the same N_Port_ID). Note that this is not required, if the new disk acquires a new N_Port_ID.

[-f] disable

This option is used to disable the card.

WARNING: This is a DESTRUCTIVE test and communication to all nodes will be terminated.

The -f option can be used to suppress the warning message displayed by the Fibre Channel driver utility.
This option is typically used when a hardware problem cannot be resolved and is interfering with system performance.

**enable**

This option is used to enable the card, typically when a previous hardware problem has been resolved.

**ns_query_ports [-k]**

This option is used to query the name server and get the list of nports for the Fibre Channel driver.

The `-k` option is used to get the list of nports cached in the driver query buffer. The name server will not be queried in this case.

**[-f] dump_current_state**

This option is used to force the driver and firmware to dump their current state information and other data structures.

**WARNING:** This is a DESTRUCTIVE operation. This might result in failure of current I/O requests.

The `-f` option can be used to suppress the warning message displayed by the Fibre Channel driver utility.

The dump data will be saved in the `/tmp` directory. The firmware dump will be stored in a file named `FC-driverfw_date-timestamp.dmp` and the driver dump will be saved in a file named `FC-driverdrv_date-timestamp.dmp`.

**dump_saved_state**

This option is used to retrieve firmware and driver dump saved in the driver memory. The driver initiates a dump when an internal error is encountered. Internal errors could be either due to firmware hang or to an irrecoverable error in the firmware or hardware. The dump files will be saved in the `/tmp` directory. The firmware dump will be stored in a file named `FC-driverfw_date-timestamp.dmp` and the driver dump will be saved in a file named `FC-driverdrv_date-timestamp.dmp`. These dumps should be sent to HP for further analysis of the problem.

**NOTE:** The driver does not save any new dumps, until the previously saved dump is retrieved with this option. The availability of a saved dump can be checked by running `fcmsutil device_file`.

**dump_nvram**

This option is used to display the contents of NVRAM on the adapter.

**rom_fw_update ROM-firmware-file**

This option is used to update the ROM firmware stored in card’s FLASH ROM.

**WARNING:** This is a DESTRUCTIVE operation. Using this option may result in failure of current I/O requests.

This option requires the name of a binary image file that contains the updated firmware. This operation should only be performed by qualified personnel. Failure to successfully complete the firmware update may result in adapter and/or system failure in case the boot disks are accessed through this card.

**efi_drv_update EFI-Driver-file**

This option is used to update the EFI driver stored in card’s FLASH ROM.

**WARNING:** This is a DESTRUCTIVE operation. Using this option may result in failure of current I/O requests.

This option requires the name of a binary image file that contains the EFI driver. This operation should only be performed by qualified personnel. Failure to successfully complete the EFI driver update may result in adapter and/or system boot up failure if the boot disks are accessed through this card.

**[-f] set_int_delay**

This option is used to set the interrupt delay mode and value, or to turn off interrupt delay. Settings made using this option are not persistent across reboots.

**WARNING:** This is a DESTRUCTIVE operation and will abort communication to all target devices until the process is completed.
The complete syntax for this command is:

```
[-f] set_int_delay {off | [-z {5 | 6}] value}
```

The `-f` option can be used to suppress the warning message.

The `off` option turns off interrupt delay.

The interrupt delay mode, also known as Zero Interrupt Operation (ZIO) mode, is set using the `-z` option. There are two interrupt delay modes available: 5 and 6. Mode 5 delays every interrupt by the interrupt delay period. Mode 6 delays an interrupt unless there are no active I/Os in the HBA port, in which case the interrupt is generated immediately. If the `-z` option is not specified, then mode 6 is used by default. The interrupt delay period is calculated from the interrupt delay value using the formula:

\[ \text{value} \times 200 \text{ microseconds} \]

`get_int_delay`

This option displays the current interrupt delay settings for the HBA port.

`sfp`

This option is used to display diagnostics information from the card's optical transceiver. It includes information from the SFF-8472 specification such as cable lengths, current temperature, voltage, transmit and receive power, TX bias, as well as other data.

NOTE: This option is only supported by 4Gb/s capable Fibre Channel cards.

**EXAMPLES**

Print the remote port parameters using the `get remote` option if the driver is idle. In this example, `/dev/td1` is the device file and `/dev/rdsk/c27t0d0` is the respective raw disk file.

```
fcmsutil /dev/td1 get remote 0x98 < /dev/rdsk/c27t0d0
```

Print a short listing of the statistics maintained by the driver, with `/dev/td1` as the device file.

```
fcmsutil /dev/td1 stat -s
```

Send 5 echo packets of 200 bytes each to a remote N_Port with `loop_id` 4, with `/dev/td1` as the device file.

```
fcmsutil /dev/td1 echo -l 4 200 5
```

Print a short listing of the statistics of the device whose `remote-N-Port-ID` is 0x02ae4 and with `/dev/td1` as the device file.

```
fcmsutil /dev/td1 devstat 0x02ae4
```

Clear the device statistics of the device whose `wwn` is 0x100000e002219f45 and with `/dev/fcd2` as the device file.

```
fcmstutil /dev/fcd2 clear_devstat -w 0x100000e002219f45
```

Perform a Internal loopback test, sending 1000 packets with `/dev/fcd2` as the device file.

```
fcmstutil /dev/fcd2 lb int crpat 1000
```

Display diagnostics information from the HBA's optical transceiver with `/dev/fcd2` as the device file.

```
fcmstutil /dev/fcd2 sfp
```

**AUTHOR**

`/opt/fcms/bin/fcmstutil` was developed by HP.

**SEE ALSO**

`ioscan(1M)`, `privileges(5)`.
NAME
fdetach - detach a STREAMS-based file descriptor from a filename

SYNOPSIS
fdetach path

DESCRIPTION
The fdetach command detaches or disassociates a file descriptor for an open STREAMS device or pipe from its filename in the file system. The path argument is the path that was previously associated with the file descriptor by the fattach() function.

Operations on path will subsequently affect the file system node, not the STREAMS device or pipe. The permissions and status of the node are returned to the state that they were in before the STREAMS device or pipe was attached. Any other paths that the STREAMS device or pipe may be attached to are not affected.

To successfully issue the fdetach command, the user must be superuser or must be the owner of the file and have write permission.

RETURN VALUE
fdetach returns 0 (zero) on success. If fdetach fails, it returns 1 and prints a message to stderr.

EXAMPLES
To detach the file descriptor for the STREAMS file /tmp/streamfile from its associated file system node, enter:

fdetach /tmp/streamfile

FILES
/usr/lib/nls/C/fdetach.cat NLS catalog for fdetach.

SEE ALSO
fattach(3C), fdetach(3C), streamio(7).
NAME
ff - list file names and statistics for a file system

SYNOPSIS
/usr/sbin/ff [-F FStype] [-o specific_options] [-V] special ...

DESCRIPTION
The ff command reads the i-list and directories of each special file, assuming it to be a file system, saving i-node data for files that match the selection criteria. Output consists of the path name for each saved i-node, plus any other file information requested with the -o option. Output fields are positional. The output is produced in i-node order; fields are separated by tabs. The default line produced by ff includes the path name and i-number fields.

Options and Arguments
ff recognizes the following options and arguments:

- F FStype Specify the file system type on which to operate (see fstyp(1M) and fs_wrapper(5)). If this option is not included on the command line, then the file system type is determined from the file /etc/fstab by matching each special with an entry in that file. If there is no entry in /etc/fstab, then the file system type is determined from the file /etc/default/fs.

- o specific_options Specify options specific to each file system type. specific_options is a list of suboptions and/or keyword/attribute pairs intended for a specific FStype-specific module of the command. See the file-system-specific man pages for a description of the specific_options supported, if any.

- V Echo the completed command line, but perform no other action. The command line is generated by incorporating the user-specified options and other information derived from /etc/fstab. This option allows the user to verify the command line.

EXAMPLES
List the path names and i-numbers of all files in the file system /dev/dsk/c1d2s0:

ff /dev/dsk/c1d2s0

Execute the ff command on HFS file system /dev/dsk/c1d2s0:

ff -F hfs /dev/dsk/c1d2s0

Display a completed command line without executing the command:

ff -V /dev/dsk/c1d2s0

FILES
/etc/default/fs File that specifies the default system type.
/etc/fstab Static information about the file systems.

SEE ALSO
find(1), ff_hfs(1M), ff_vxfs(1M), fstyp(1M), ncheck(1M), fstab(4), fs_wrapper(5).
**NAME**

ff_hfs: ff - list file names and statistics for HFS file system

**SYNOPSIS**

```
[-p prefix] [-s] [-u] [-V] special ...
```

**DESCRIPTION**

The `ff` command reads the i-list and directories of each special file `special`, assuming it to be an HFS file system, saving i-node data for files that match the selection criteria. Output consists of the path name for each saved i-node, plus any other file information requested using the print options below. Output fields are positional. The output is produced in i-node order; fields are separated by tabs. The default line produced by `ff` contains the path name and i-number fields. With all options specified, the output fields include path name, i-number, size, and user ID.

The `num` parameter in the options descriptions is a decimal number, where `+num` means more than `num`, `-num` means less than `num`, and `num` means exactly `num`. A day is defined as a 24-hour period.

`ff` lists only a single path name out of many possible ones for an i-node with more than one link, unless you specify the `-l` option. With `-l`, `ff` applies no selection criteria to the names listed. All possible names for every linked file on the file system are included in the output. On very large file systems, memory may run out before `ff` completes execution.

**Options and Arguments**

`ff` recognizes the following options and arguments:

- `-a num` Select a file if the i-node has been accessed in `num` days.
- `-c num` Select a file if the i-node has been changed in `num` days.
- `-F hfs` Specify the HFS file system type.
- `-i in-node-list` Generate names for any i-node specified in the `inode-list`.
- `-I` Do not display the i-node number after each path name.
- `-l` Generate a list of all path names for files with more than one link.
- `-m num` Select a file associated with an i-node if it has been modified in `num` days.
- `-n file` Select a file associated with an i-node if it has been modified more recently than the specified `file`.
- `-p prefix` Add the specified `prefix` to each path name. The default prefix is `. (dot)`.
- `-s` Write the file size, in bytes, after each path name.
- `-u` Write the owner’s login name after each path name.
- `-V` Echo the completed command line, but performs no other action. The command line is generated by incorporating the user-specified options and other information derived from `/etc/fstab`. This option allows the user to verify the command line.

**EXAMPLES**

List the path names and i-numbers of all files in the file system `/dev/dsk/c1d2s0`:

```
ff /dev/dsk/c1d2s0
```

Same as above, but suppress the printing of i-numbers:

```
ff -I /dev/dsk/c1d2s0
```

List files on the same file system that have been modified recently, displaying the path name, i-number, and owner’s user name (the `-u` option). List only files that have been modified within the last two days (the `-m -2` option):

```
ff -m -2 -u /dev/dsk/c1d2s0
```

List all files on the same file system, including the path name and i-number of each file, that was last accessed more than 30 days ago (`-a +30`):

```
ff -a +30 /dev/dsk/c1d2s0
```
ff_hfs(1M)

Find all path names associated with i-nodes 451 and 76 (the -l option):

```
ff -l -i 451,76 /dev/dsk/c1d2s0
```

Execute the `ff` command on an HFS file system `/dev/dsk/c1d2s0`:

```
ff -F hfs /dev/dsk/c1d2s0
```

FILES

/etc/fstab Static information about the file systems.

SEE ALSO

`find(1), ff(1M), ff_vxfs(1M), ncheck(1M), fstab(4)`. 

ff_hfs(1M)
NAME
ff_vxfs: ff - fast find: list file names and statistics for a VxFS file system

SYNOPSIS
[-o s] [-p prefix] special...

DESCRIPTION
ff reads the inode list and directories of each special file, assuming it to be a VxFS file system, and prints
inode data for files that match the selection criteria. Output consists of the pathname for each saved inode,
plus any other file information requested using the print options below. Output fields are positional. The
output is produced in inode order; fields are separated by tabs. The default line produced by ff is:

pathname inumber

The pathname is preceded by a dot (.) unless you specify the -p generic option. The maximum information
ff displays is:

pathname inumber size owner

The num parameter in the options descriptions is a decimal number, where +num means more than num
days, -num means less than num days, and num means exactly num days. A day is defined as a 24-hour
period.

ff prints summary information to standard error output in addition to the report sent to standard output.

Options
ff recognizes the following options:

-a num     Select a file if the inode has been accessed in num days.
-c num     Select a file if the inode has been changed in num days.
-F vxfs     Specify the VxFS file system type.
-i inode-list Generate names for any inodes specified in the inode-list.
-I          Do not display the inode number after each pathname.
-l          Generate a list of all pathnames for files with more than one link.
-m num     Select a file associated with the inode if it has been modified in num days.
-n file     Select a file associated with an inode if it has been modified more recently than the
specified file.
-p prefix   Add the specified prefix to each pathname. The default prefix is dot (.).
-o s        Print only special files and files with set-user-ID mode (VxFS-specific option).
-s          Write the file size, in bytes, after each pathname.
-u          Write the owner's login name after each pathname.
-V          Echo the completed command line, but performs no other action. The command line
is generated by incorporating the user specified options and other information derived
from /etc/fstab. This option allows the user to verify the command line.

Operands
ff recognizes the following operand:

special     Name of a VxFS file system.

EXAMPLES
List the pathnames and inumbers of all files in the file system /dev/vg01/rlvol1:

ff /dev/vg01/rlvol1

Same as above, but suppress the printing of inumbers:

ff -I /dev/vg01/rlvol1

List files on the same file system that have been modified in the last two days (-m -2), displaying the
pathname, inumber, and owner's user name (-u).
ff -m -2 -u /dev/vg01/rlvol1
List all files on the same file system, including the pathname and inumber of each file, that were last accessed more than 30 days ago (-a +30):

ff -a +30 /dev/vg01/rlvol1
Find all pathnames associated with inodes 451 and 76 (-l):

ff -l -i 451,76 /dev/vg01/rlvol1
Execute the ff command on a VxFS file system /dev/vg01/rlvol1:

ff -F vxfs /dev/vg01/rlvol1

FILES
/etc/fstab Static information about the file systems.

SEE ALSO
find(1), ff(1M), ncheck_vxfs(1M), fstab(4).
fingerd(1M)

NAME
fingerd - remote user information server

SYNOPSIS
/usr/lib/fingerd [-r]

DESCRIPTION
fingerd is the server for the RFC 742 Name/Finger protocol. It provides a network interface to
finger, which gives a status report of users currently logged in on the system or a detailed report about a
specific user (see finger(1)). The Internet daemon executes fingerd when it receives a service request at
the port listed in the services data base for “finger” using “tcp” protocol; see inetd(1M) and services(4).

To start fingerd from inetd, the configuration file /etc/inetd.conf must contain an entry as
follows:

    finger stream tcp nowait bin /usr/lib/fingerd

Once a remote host is connected, fingerd reads a single “command line” terminated by a carriage-return
and line-feed. It uses this command line as the arguments to an invocation of finger. fingerd sends
the output of finger to the remote host and closes the connection.

If the command line is null (contains only a carriage-return and line-feed pair), finger returns a report
that lists all users logged in on the system at that moment.

If a user name is specified on the command line (for example, user<CR><LF>), the response lists more
extended information for only that particular user, whether logged in or not. See finger(1) for the details of
this extended information.

If fingerd is run with the -r option, it allows remote user names on the command line (for example,
user@host<CR><LF>). Otherwise, if the command line contains a remote user name, fingerd prints the
error message Remote finger not allowed and closes the connection.

AUTHOR
fingerd was developed by the University of California, Berkeley and HP.

SEE ALSO
finger(1), inetd(1M), services(4),
RFC 742 for the Name/Finger protocol.
fixman(1M)

NAME
fixman - fix manpages for faster viewing with man command

SYNOPSIS
/usr/sbin/fixman [-A alt-path]

DESCRIPTION
The fixman command is a shell script that processes manpages in the cat* directories to unexpand spaces to tabs where possible, and to remove all character-backspace pairs (which usually exist to cause overstriking or underscoring for printer output). Removal of unnecessary character sequences improves the speed of man(1), and reduces disk space consumption. The fixman command should be run after using catman to create formatted, cat-able manpages from unformatted, nroff-compatible source files (see catman(1M)).

By default, fixman searches for cat* subdirectories in the following parent directories in the order indicated:

- /usr/share/man
- /usr/contrib/man
- /usr/local/man

If the MANPATH environment variable is set, the directory paths specified by MANPATH are searched instead of the default. See environ(5) for a description of the MANPATH environment variable.

The fixman command does not remove duplicate blank lines. Thus, all files remain a multiple of one page (66 lines) long and can still be passed directly to lp (see lp(1)). (Note that man(1) normally uses more -s to accomplish this removal.)

To ensure success, fixman should be run by a user who has appropriate privileges. It will take awhile to complete depending on system speed, load, memory size, etc. As a side-effect, file ownerships and permissions may be changed.

Options
-A alt-path
Perform actions based on the given alternate root. With this option, alt-path will be prepended to all directory paths, including default paths or the paths defined by MANPATH.

EXTERNAL INFLUENCES
Environment Variables
MANPATH, if set, defines the directories to be searched for cat-able manpages.

WARNINGS
If the value of MANPATH is not the same while fixman is running as it was when catman was run or when manpage files were installed, some files may be missed and not processed (see catman(1M)).

EXAMPLES
Run fixman from a server to fix the manpages on a diskless under the alternate root /export/shared_roots/OS_700:

fixman -A /export/shared_roots/OS_700

This will fix manpages in cat* directories under:

- /export/shared_roots/OS_700/usr/share/man/
- /export/shared_roots/OS_700/usr/contrib/man/
- /export/shared_roots/OS_700/usr/local/man/

FILES
/usr/share/man/cat*[.Z] Directories containing [compressed] nroff(1)-formatted versions of manpages
/usr/local/man/cat*[.Z]
/usr/contrib/man/cat*[.Z]

AUTHOR
fixman was developed by HP.
SEE ALSO

   catman(1M), chmod(1), expand(1), lp(1), man(1), mv(1), sed(1), environ(5).
NAME
format - format an HP SCSI disk array LUN

SYNOPSIS
format device_file

DESCRIPTION
format formats one LUN of the HP SCSI disk array associated with device file, device_file. The format
will usually be a soft or zeroing format, in which the controller writes zeroes to the data area and parity
area, if any, of the LUN.

NOTE: The above should always be true of a sub-LUN, but the controller might decide, based on certain
conditions, to do a full format of a regular LUN, which consists of sending a mode select and a media
initialization command to the physical drive(s) in question, followed by zeroing the data and parity
area, if any. The conditions which will cause a full format to be done are as follows:

1. The controller received a Mode Select command which requires a drive sector size change.
2. The controller received a Mode Select command which changed a parameter in the Format Dev-
   ice Page (0x03).
3. The LUN contains one or more failed drives. In this case only a certain subset of the drives con-
   taining the failed drives will be formatted.
4. Either the FmtData or the CmpLst bit in the Format Unit CDB is set.

RETURN VALUE
format returns the following values:
   0 Successful completion.
   -1 Command failed.

DIAGNOSTICS AND ERRORS
Errors can originate from problems with:

- format
- SCSI (device level) communications
- system calls

Error messages generated by format:
usage: format <special>
An error in command syntax has occurred. Enter command again with all required arguments, in the
order shown.

format: device busy
To ensure that format does not modify a disk array that is being used by another process, format
attempts to obtain exclusive access to the disk array. If the disk array is already opened by another
process (for example, LVM — the Logical Volume Manager), a "device busy" error message is
returned by the driver. To eliminate the "device busy" condition, determine what process has the
device open. In the case of LVM, it is necessary to deactivate the volume group containing the array
before formatting array LUNs (see vgchange(1M)).

format: LUN # too big
The LUN number, which is derived from the device file name, is out of range.

format: LUN does not exist
The addressed LUN is not configured, and thus is not known to the array controller.

format: Not a raw file
Utilities must be able to open the device file for raw access.

format: Not an HP SCSI disk array
The device being addressed is not an HP SCSI disk array.

SCSI (device level) communication errors:
Sense data associated with the failed operation is printed.
Error messages generated by system calls:

format uses the following system calls:

malloc(), free(), stat(), open(), close(), read(), write(), and ioctl().

Documentation for these HP-UX system calls contains information about the specific error conditions associated with each call. format does not alter the value of errno. The interpretation of errno for printing purposes is performed by the system utility strerror().

EXAMPLES

To format the HP SCSI disk array LUN /dev/rdsk/c2t0d0 on a Series 800:

format /dev/rdsk/c2t0d0

WARNING

The format command will destroy all user data on the addressed LUN.

DEPENDENCIES

The HP C2425 and HP C2427 disk arrays are only supported on Series 700 systems running HP-UX version 9.0X.

The HP C2430 disk array is supported on Series 700 and 800 systems running HP-UX versions 9.0X and 10.0X.

AUTHOR

format was developed by HP.
NAME
frecover - selectively recover files

SYNOPSIS
/usr/sbin/frecover -R path [-f device]
/usr/sbin/frecover -x [-hmosvyAFNOX] [-c config] [-e path] [-f device] [-g graph]
    [-i path] [-S skip] [-E extarg]
/usr/sbin/frecover -I path [-vy] [-f device] [-c config]
/usr/sbin/frecover -V path [-vy] [-f device] [-c config]

DESCRIPTION
frecover reads media written by the fbackup command. Its actions are controlled by the selected function -r, -R, -x, -V, or -I. The function performed by frecover is specified by one of the following options:

-r
The backup media is read and the contents are loaded into the directories from which they were backed up. This option should only be used to recover a complete backup onto a clear directory or to recover an incremental backup after a full level-zero recovery (see fbackup(1M)). This is the default behavior.

-x
The files identified by the -i, -e, and -g options (see below) are extracted or not extracted from the backup media. If a file to be extracted matches a directory whose contents have been written to the backup media, and the -h option is not specified, the directory is recursively extracted. The owner, modification time, and access control list (including optional entries, unless the -A option is specified) are recovered. If no file argument is given (including an empty graph file), all files on the backup media are extracted, unless the -h option is specified.

-I path
The index on the current volume is extracted from the backup media and is written to path.

-V path
The volume header on the current volume is extracted from the backup media and is written to path. The following fields from the header are extracted in the format label:value with one pair per line.

Magic Field
On valid fbackup media, it contains the value FBACKUP_LABEL. On pre-10.20 fbackup media, it contains FBACKUP_LABEL.

Machine Identification
This field contains the result of uname -m.

System Identification
This field contains the result of uname -s.

Release Identification
This field contains the result of uname -r.

Node Identification
This field contains the result of cuserid(3S).

User Identification
This field contains the result of cuserid(3S).

Record Size
This field contains the maximum length in bytes of a data record.

Time
This field contains the time fbackup was started.

Media Use
This field contains the number of times the media has been used for backup.

Volume Number
This field contains a # character followed by 3 digits, and identifies the current volume in the backup.

Checkpoint Frequency
This field contains the number of data records between checkpoints.

Fast Search Mark Frequency
This field contains the number of files between fast search.
**frecover(1M)**

**Index Size**
This field contains the size of the index.

**Backup Identification Tag**
This field is composed of 2 items: the process ID (pid), and the start time of that process.

**Language**
This field contains the language used to make the backup.

**-R path**
An interrupted full recovery can be continued using this option. `frecover` uses the information in file `path` to continue the recovery from where it was interrupted. The only command line option used by `frecover` with this option is `-f`. The values in `path` override all other options to `frecover`. Note also that only full recoveries are restarted with this option, because no history of include or exclude lists is stored in the restart file. If a partial recovery (i.e., using the `-x` option) is interrupted then restarted with this option, `frecover` continues recovering where the partial recovery left off, but restores all files on the backup media beyond this point.

The following options can be used in addition to the option above that selects the desired function:

**-c config**
`config` specifies the name of a configuration file to be used to alter the behavior of `frecover`. The configuration file allows the user to specify the action to be taken on all errors, the maximum number of attempts at resynchronizing on media errors (`-S` option), and the action to be taken on media errors. Each entry of a configuration file consists of an action identifier followed by a separator followed by the specified action. Valid action identifiers are `error`, `chgvol`, and `sync`. Separators can be either tabs or spaces. In the following sample configuration file, each time an error is encountered, the script `/var/adm/fbackupfiles/frecovererror` is executed. The script `/var/adm/fbackupfiles/frecoverchgvol` is executed each time the backup media is to be changed. The maximum number of resynchronization attempts is five.

```
error /var/adm/fbackupfiles/frecovererror
chgvol /var/adm/fbackupfiles/frecoverchgvol
sync 5
```

**-e path**
`path` is interpreted as a graph to be excluded from the recovery. There is no limit on how many times the `-e` option can be specified.

**-f device**
`device` identifies the backup device to be used instead of the default `/dev/rmt/0m`. If `device` is `-`, `frecover` reads from standard input. Thus `fbackup` and `frecover` can be used in a pipeline to backup and recover a file system as follows:

```
fbackup -i /usr -f - | (cd /mnt; frecover -Xrf -)
```

If more than one output file is specified, `frecover` uses each one successively and then repeats in a cyclical pattern. Patterns can be used in the device name in a way similar to file name expansion as done by `sh(1)`. The expansion of the pattern results in all matching names being in the list of devices used. A device on the remote machine can be specified in the form `machine:device`. `frecover` creates a server process from `/etc/rmt` on the remote machine to access the tape device. The pattern matching capability does not apply to remote devices. Only raw magnetic tapes can be remote devices. The fast search marks capability is not used when accessing remote DDS devices.

**-g graph**
`graph` defines a graph file. Graph files are text files and contain the list of file names (graphs) to be recovered or skipped. Files are recovered using the `-i` option; so, for example, if the user wants to recover all of `/usr`, the graph file contains one entry:

```
i /usr
```

It is also possible to skip files by using the `-e` option. For example, if a user wants to recover all of `/usr` except for the subgraph `/usr/lib`, the graph file contains two entries:

```
i /usr
e /usr/lib
```

If the graph file is missing, `frecover` exits with an error message. An empty graph file results in recovering all files on the media.
frecover(1M)

-h Extract the actual directory, rather than the files that it references. This prevents hierarchical restoration of complete subtrees from the backup media.

-i path path is interpreted as a graph to be included in the recovery. There is no limit on how many times the -i option can be specified.

-m Print a message each time a file marker is encountered. Using this option, frecover prints a message each time either a DDS fast search mark, a filemark (EOF), or a checkpoint record is read. Although useful primarily for troubleshooting, these messages can also be used to reassure the user that the backup is progressing during long, and otherwise silent, periods during the recovery.

-o Recover the file from the backup media irrespective of age. Normally frecover does not overwrite an existing file with an older version of the file.

-s Attempt to optimize disk usage by not writing null blocks of data to sparse files.

-v Normally frecover works silently. Verbose option. Displays the file type and name of each file processed.

-y Automatically answer yes to any inquiries.

-A Do not recover any optional entries in access control lists (ACLs). Normally, all access control information, including optional ACL entries, is recovered. This option drops any optional entries and sets the permissions of the recovered file to the permissions of the backed up file. Use this option when recovering files backed up from a system with ACLs on a system where ACLs are not present (see acl(5)).

-F Recover files without recovering leading directories. For example, this option would be used if a user wants to recover /usr/bin/vi, /usr/bin/sh, and /etc/passwd to a local directory without creating each of the graph structures.

-E extarg Specifies the handling of any extent attributes backed up by fbackup. The -E option takes the following keywords as arguments:

warn Issue a warning message if extent attributes cannot be restored, but restore the file anyway.

ignore Do not restore extent attributes.

force Issue an error message and do not restore the file if extent attributes cannot be restored.

Extent attributes cannot be restored if the files are being restored to a file system which does not support extent attributes or if the file system's block size is incompatible with the extent attributes. If -E is not specified, extarg defaults to warn.

-N (no recovery) Prevent frecover from actually recovering any files onto disk, but read the backup as if it was, in fact, recovering the data from the backup, producing the same output that it would on a normal recovery. This option is useful for verifying backup media contents in terms of validity (block checksum errors are reported), and contents (a listing of files can be produced by using the -N and -v options together). Note that the listing of files produced with the -N and -v options requires the reading of the entire backup, but is therefore a more accurate reflection of the backup's contents than the index stored at the beginning of the backup (which was created at the start of the backup session, and is not changed during the course of the backup).

-O Use the effective uid and gid for the owner and group of the recovered file instead of the values on the backup media.

-S skip frecover does not ask whether it should abort the recovery if it gets a media error. It tries to skip the bad block or blocks and continue. Residual or lost data is written to the file named by skip. The user can then edit this file and recover otherwise irretrievable data.

-X Recover files relative to the current working directory. Normally frecover recovers files to their absolute path name.

EXTERNAL INFLUENCES

Environment Variables

LC_COLLATE determines the order in which frecover expects files to be stored on the backup device and the order in which file names are output by the -I option.
**frecover(1M)**

**LC_MESSAGES** determines the language in which messages are displayed.

If **LC_COLLATE** and **LC_MESSAGES** are not specified in the environment or are set to the empty string, the value of **LANG** is used as a default for each unspecified or empty variable. If **LANG** is not specified or is set to the empty string, a default of "C" (see lang(5)) is used instead of **LANG**. If any internationalization variable contains an invalid setting, **frecover** behaves as if all internationalization variables are set to "C". See environ(5).

**International Code Set Support**

Single- and multi-byte character code sets are supported.

**WARNINGS**

For incremental backups created prior to installing HP-UX Release 8.0, or for recoveries that do not begin with the first volume (such as when reading tape 3 first), it is possible for the preceding directories to a recoverable file to not be on the media. This can happen, for example, if the directories did not change since the last full backup. If **frecover** encounters a file on the backup that should be recovered, but it has not recovered the file's parent directories from the backup, it prints a message stating that the recovery will continue with that file, and attempts to create the file's parent directories as needed.

Use of **frecover** does not require special privileges. However, if a user does not have access permission to a given file, the file is not recovered.

The **fbackup** index format now includes the file size in the first field; the previous format simply had the ‘#’ character in that field. The implementation provides both forward and backward compatibility between the old and new index formats. However, the file sizes are used in conjunction with the checkpoints to increase selective recovery speed on DLT devices, so recovery of an **fbackup** volume that does not have the new index format will not see that performance gain.

When using a DDS tape written with the current release of **fbackup** to do a partial recovery, **frecover** attempts to use the DDS fast-search capability to find files on the tape more quickly. In order to do this, however, **frecover** needs to create an in-memory copy of the index, and mark the files on that index which it needs to recover before actually reading through the tape to find the files. This is done when the first index is read from the tape, and accounts for a period of time just after recovery is begun where the tape is inactive while this in-memory index is constructed. The larger the index is, the longer this period lasts.

The utility set comprised of **fbackup** and **frecover** was originally designed for use on systems equipped with not more than one gigabyte of total file system storage. Although the utilities have no programming limitations that restrict users to this size, complete backups and recoveries of substantially larger systems can cause a large amount of system activity due to the amount of virtual memory (swap space) used to store the indices. Users who want to use these utilities, but are noticing poor system-wide performance due to the size of the backup, are encouraged to back up their systems in multiple smaller sessions, rather than attempting to back up the entire system at one time. However, if the entire backup must be done with a single session, the user may encounter an error in **frecover** if there is not enough virtual memory available. If this happens, the user might consider adjusting the maxdsiz parameter or the swap space; both of these require a reboot.

Note that when recovering files with access control lists, the ACL entries are stored on the backup as user login names. If a login name cannot be found in the password file, the file is recovered without its ACL, and an error is printed. In order to fully recover files backed up with ACLs, the password file (/etc/passwd) must be recovered before attempting to recover any desired ACLs.

Network special files are obsolete. Therefore, **frecover** cannot restore these files. A warning message is issued if an attempt is made to recover a network special file, and the file is skipped.

Care should be taken to match the names specified by the include and exclude options with the names in the index on the tape. Since the files are stored on the backup in lexicographic order as defined by the **LANG** or **LC_COLLATE** environment variable, **frecover** uses the exact path names to determine when a partial recovery is complete, and when an earlier tape needs to be loaded. If a user’s specification of a file to be recovered is misspelled, this may cause confusing messages, such as **frecover** asking for the previous volume, when volume one is mounted.
frecover(1M)  frecover(1M)

DEPENDENCIES
frecover does not support QIC-120 and QIC-150 formats on QIC devices. If frecover is attempted for these formats, frecover fails and the following message is displayed:

    mt lu X:Read must be a multiple of 512 bytes in QIC 120 and QIC 150

AUTHOR
frecover was developed by HP.

FILES
/dev/rmt/0m    Default backup device.

SEE ALSO
cpio(1), dump(1M), fbackup(1M), restore(1M), rmt(1M), acl(5).
frupower - turn on/off or display current status of power for cells and I/O chassis

NAME
frupower - turn on/off or display current status of power for cells and I/O chassis

SYNOPSIS
frupower -c cell [-c cell]... [-d] [-o] [-f]
    [ [ -u username : passwd ] | -h ] IPaddress | hostname ]
frupower -i I/O chassis [-i I/O chassis]... [-d] [-o] [-f]
    [ [ -u username : passwd ] | -g ] IPaddress | hostname ]
frupower -C [-I] [-d] [-l cabinet] [-l cabinet]...
    [ [ -u username : passwd ] | -g ] IPaddress | hostname ]

DESCRIPTION
The frupower command turns power on or off, or displays the current status of power for cells and I/O chassis.

The operations allowed by this command are subject to the following conditions:

- An active cell cannot be powered off. An active cell is one that is assigned to a partition and that is being used by the partition.

- The -i option can be used to display the power state of any I/O chassis, but can only control the power state of an I/O chassis in the following two situations:
  - The I/O chassis is attached to an active cell and is powered off. The I/O chassis may be powered on, but will not become active until the partition to which the cell is assigned has been rebooted.
  - The I/O chassis is attached to an active cell and is powered on, but the I/O chassis has not yet become active. The I/O chassis may be powered off.

- If the target of the command is the local partition, meaning that neither the -u nor -g option is used:
  - Any user can run the command to display the power state of cells and I/O chassis.
  - Root permission is required to change the power state of a component (cell or chassis).
  - Cell power operations are limited to free cells and inactive or powered-off cells assigned to the local partition. If the nPartition Configuration Privilege is restricted, then control of free cells is not allowed.

  Note: The nPartition Configuration Privilege state can only be changed at the service processor's Command menu.

- When connecting to a remote partition using the -u and -h options:
  - Any local user may run the command. Root permission is not required, and the user's system need not be an nPartition.
  - The set of allowed operations depends on the privileges of username on the remote partition, and the state of the nPartition Configuration Privilege on the complex on which the remote partition exists. The rules and conditions are the same as for the local partition description.

- When connecting to the service processor of a complex using the -g and -h options:
  - Any user may run the command. Root permission is not required, and the user's system need not be an nPartition.
  - The user may display the power state of any cell or I/O chassis. Any powered off cell can be powered on, and any inactive cell can be powered off, whether or not the cell is assigned to a partition or is a free cell. The state of the complex's nPartition Configuration Privilege does not affect this mode of command operation.

  Note that knowledge of the -g passwd gives the user full authorization to carry out any of the above operations, regardless of her permissions on her local system.

Refer to the HP System Partitions Guide for a description of the partition management terms used in this man page.
Options and Arguments

frupower recognizes the following command line options and arguments:

- **-d**
  Displays power status of specified cells or I/O chassis. This is the default.

- **-o**
  Powers on specified cells or I/O chassis.

- **-f**
  Powers off specified cells or I/O chassis.

The **-d**, **-o**, and **-f** options are mutually exclusive. The **-o** and **-f** options are unavailable with **-c** and **-i**.

- **-u username [password]**
  Specifies the required authorization to access a partition other than the local partition and to control components on it, subject to the conditions stated in the **DESCRIPTION** section (but can also be used as a loopback access to the local partition). The target complex is the one in which this target partition resides.

  The **-h** option is required if this option is used.

  **username** specifies a configured user name on the target partition. Any **username** can display the power state of components on the target partition. **username** must have root permission to change any power state.

  **password** specifies the password associated with the **username**. If this field is empty, the command prompts for the password.

  **Note:** This command is a Web-Based Enterprise Management (WBEM) Client Application. The **-u** option accesses the target partition using a Secure Sockets Layer (SSL) connection. If errors are reported, check that the conditions described in the **DEPENDENCIES** section are satisfied.

  **SECURITY WARNING:** Specifying the password directly on the command line may pose a security risk in your environment. The **ps** or other related commands can be invoked in such a way as to display the command line of a process. In this situation, any authenticated user on the system can potentially see the password while the process is executing. Hence, it is highly recommended not to specify the password on the command line, and instead allow the command to prompt for the password.

  **Note:** Specifying **password** on the command line will not be supported in future releases.

- **-h IPaddress|hostname**
  This option should only be used in combination with either the **-u** or **-g** option.

  **IPaddress|hostname** specifies the IP address or hostname of the target partition (**-u**) or complex (**-g**).

- **-g [password]**
  Allows access to the complex specified by the **-h** option. The accessed complex is then considered the target complex. Access is via the service processor's LAN port.

  The **-h** option is required if this option is used.

  **password** specifies the IPMI password of the service processor. If this field is omitted, the command prompts for the password.

  If an error is reported when you attempt to connect using this option, check to see that IPMI LAN access has not been disabled on the remote service processor. Access to the complex via IPMI over LAN can be enabled or disabled by logging on to the service processor and using the **SA** command from the Command Menu.

  The **-u** and **-g** options are mutually exclusive.

  **SECURITY WARNING:** Specifying the password directly on the command line may pose a security risk in your environment. The **ps** or other related commands can be invoked in such a way as to display the command line of a process. In this situation, any authenticated user on the system can potentially see the password while the process is executing. Hence, it is highly recommended not to specify the password on the command line, and instead allow the command to prompt for the password.

  **Note:** Specifying **password** on the command line will not be supported in future releases.

- **-c cell**
  The specified cell is powered on/off or the power status is displayed. Refer to the **DESCRIPTION** section for the conditions under which power control is allowed.
If a power operation on multiple cells is specified, the command processes each one, regardless of the outcome of the operation on a particular cell. It is possible for an operation to succeed on one cell and fail on another.

A cell can be specified either in the local (cabinet#/slot#) or global (cell#) format. For example, the cell located in cabinet 0, slot 4 is locally identified as 0/4 or globally as simply 4.

-`i I/Ochassis` The specified I/Ochassis is powered on/off or the power status is displayed. Refer to the DESCRIPTION section for the conditions under which power control is allowed.

If a power operation on multiple I/O chassis is specified, the command processes each one, regardless of the outcome of the operation on a particular I/O chassis. It is possible for an operation to succeed on one I/O chassis and fail on another.

An I/O chassis id must be specified in the form of cabinet#/enclosure#/chassis#. For example, the I/O chassis located in cabinet 0, enclosure 1 and I/O chassis slot 3 is identified as 0/1/3.

-`C` Displays power status of all cells. If the -u or -g option is specified, the operation applies to all component cells of the accessed complex.

-`I` Displays power status of all I/O chassis. If the -u or -g option is specified, the operation applies to all I/O chassis of the accessed complex.

The -c, -i, -C, and -I options are all mutually exclusive.

-`l cabinet` Limits the scope of the -C or -I option to the specified cabinet(s).

Mapping of Global Cell Numbers to Local Cell Numbers
The cabinets in a complex are numbered starting from 0. The cell slots in each cabinet are also numbered starting from 0. Each cabinet can have a maximum of 8 cells. For example, the cells located in cabinet 0 will have the following cell numbers in global format: 0, 1, 2, 3, 4, 5, 6, 7. The cell numbers in corresponding local format will be 0/0, 0/1, 0/2, 0/3, 0/4, 0/5, 0/6, 0/7.

Similarly the cells located in cabinet 1 will have the following cell numbers in global format: 8, 9, 10, 11, 12, 13, 14, 15. The cell numbers in corresponding local format will be 1/0, 1/1, 1/2, 1/3, 1/4, 1/5, 1/6, 1/7.

From the above convention the cell located in cabinet 0, slot 0 is identified in the local format as 0/0 or in the global format as 0/0. The parstatus command will display the above cell as “cab1,cell0”. The cell located in cabinet 0, slot 4 is identified in the local format as 0/4 or in the global format as 12. The parstatus command will display the above cell as “cab1,cell4”. See parstatus(1).

RETURN VALUE
The frupower command exits with one of the following values:

0  Successful completion.
1  Error condition occurred.

EXAMPLES
Power on the cell located in cabinet 0 slot 4 of a partition identified by the hostname GreenRiver:

    frupower -o -c 0/4 -u RemoteAdmin: -h GreenRiver

RemoteAdmin must have root permissions on GreenRiver. Since a password was not supplied on the command line, the command prompts for it.

Power off two cells located in cabinet 0 slot 4 and cabinet 0 slot 6.

    frupower -f -c 0/4 -c 0/6

WARNINGS
HP-UX 11i Version 2 is the last HP-UX release on which passing passwd for the -u and -g options is supported. Specifying passwd on the command line will not be supported in future releases. The command will prompt for the password.

DEPENDENCIES
This command uses the Web-Based Enterprise Management (WBEM) product and certain of its configuration settings. If you encounter connection errors when using the -u option, check that the following two conditions are satisfied:

Use the `cimconfig` command (see `cimconfig(1M)` in the WBEM product documentation) to verify (and correct if necessary) the setting of the following two variables:

- `enableRemotePrivilegedUserAccess=true`
- `enableHttpsConnection=true`

You must have appended the target partition's digital certificate to the local partition's Trust Store file. For the `nPartition` commands, the Trust Store file is `/var/opt/wbem/client.pem`.

*Note:* You must have appended the target partition's digital certificate to the local partition's Trust Store file. For the `nPartition` commands, the Trust Store file is `/var/opt/wbem/client.pem`. This file is used by the commands that come with WBEM installation. Hence, if the commands that come with the WBEM installation trust a target partition, then `nPartition` commands will also trust the target partition.

Refer to the WBEM documents specified in the `SEE ALSO` section below for further information.

**AUTHOR**

`frupower` was developed by the Hewlett-Packard Company.

**SEE ALSO**

`fruled(1)`, `parstatus(1)`, `cplxmodify(1M)`, `parcreate(1M)`, `parmgr(1M)`, `parmodify(1M)`, `parremove(1M)`, `parunlock(1M)`, `partition(5)`.

*HP System Partitions Guide* on [http://docs.hp.com](http://docs.hp.com),

*HP WBEM Services for HP-UX System Administrator’s Guide* on [http://docs.hp.com](http://docs.hp.com),

*HP WBEM Services for HP-UX 11i v2.0 on Integrity Servers Version A.01.05 Release Notes* on [http://docs.hp.com](http://docs.hp.com).
NAME
fsadm - a file system administration command

SYNOPSIS
/usr/sbin/fsadm [-F FStype] [-V] [-o specific_options] special

DESCRIPTION
The fsadm command is designed to perform selected administration tasks on file systems. These tasks may differ between file system types. special is a device file containing an unmounted file system. However, if the file system is of the type that provides online administration capabilities the special could be a directory. directory must be the root of a mounted file system.

Only a superuser can invoke fsadm.

Options
- F FStype Specify the file system type on which to operate (see fstyp(1M) and fs_wrapper(5)). If this option is not included on the command line, then the file system type is determined from the file /etc/fstab by matching each special with an entry in that file. If there is no entry in /etc/fstab, then the file system type is determined from the file /etc/default/fs.

-o specific_options Specify options specific to each file system type. specific_options is a list of comma separated suboptions and/or keyword/attribute pairs intended for a specific FStype-specific module of the command. See the file system specific manual entries for a description of the specific_options supported, if any.

-V Echo the completed command line, but perform no other action. The command line is generated by incorporating the user-specified options and other information derived from /etc/fstab. This option allows the user to verify the command line.

EXAMPLES
Convert a HFS file system from a nolargefiles file system to a largefiles file system:

    fsadm -F hfs -o largefiles /dev/vg02/lvol1

Display HFS relevant file system statistics:

    fsadm -F hfs /dev/vg02/lvol1

FILES
/etc/fstab Static information about the systems

SEE ALSO
    fsadm_hfs(1M), fsadm_vxfs(1M), fsck(1M), fstab(4), fs_wrapper(5).
fsadm_hfs(1M)

NAME
fsadm_hfs: fsadm - HFS file system administration command

SYNOPSIS
/usr/sbin/fsadm [-F hfs] [-V] [-o specific_options] special

DESCRIPTION
The fsadm command is designed to perform selected administration tasks on HFS file systems. special is a device file containing an unmounted file system.

Only a superuser can invoke fsadm.

Options
-F hfs Specify the HFS file system type.
-o specific_options Specify a list of comma separated suboptions and/or keyword/attribute pairs from the list below. The following specific_options are valid on HFS file systems.

largefiles Converts a nolargefiles file system to a largefiles file system. The file system should be unmounted and must be in a clean state (see fsck(1M)). A largefiles file system supports file sizes greater than 2 gigabytes.

nolargefiles Converts a largefiles file system to a nolargefiles file system. The file system should be unmounted and must be in a clean state (see fsck(1M)). All largefiles should be purged from the file system for the conversion to succeed.

-V Echo the completed command line, but perform no other action. The command line is generated by incorporating the user-specified options and other information derived from /etc/fstab. This option allows the user to verify the command line.

DIAGNOSTICS
Error and warning messages may originate from fsadm and fsck. See fsadm(1M) or fsck(1M) to interpret the error and warning messages.

EXAMPLES
Convert a nolargefiles HFS file system to a largefiles HFS file system:
fsadm -F hfs -o largefiles /dev/vg02/rlvol1

Convert a largefiles HFS file system to a nolargefiles file system:
fsadm -F hfs -o nolargefiles /dev/vg02/rlvol1

Display relevant HFS file system statistics:
fsadm -F hfs /dev/vg02/rlvol1

WARNINGS
The size of a file system will impact the performance of the fsadm command.

During conversion from largefiles file system to a nolargefiles file system fsadm scans the entire file system for a large file. This functionality degrades the performance of the fsadm command.

FILES
/etc/fstab Static information about the systems

SEE ALSO
fsadm(1M), fsadm_vxfs(1M), fsck(1M), fstab(4), fs_wrapper(5).
NAME

fsadm_vxfs: fsadm - resize or reorganize a VxFS file system

SYNOPSIS

[-k ckpt_name] [-l largesize] [-p passes] [-r rawdev] [-t time] mount_point
fsadm [-F vxfs] [-V] [-b newsize] [-r rawdev] mount_point
fsadm [-F vxfs] [-V] [-o largefiles | nolargefiles] mount_point | special
fsadm [-F vxfs] [-V] [-c] mount_point

DESCRIPTION

fsadm performs online administration functions on VxFS file systems, Storage Checkpoints, or individual files and directories. fsadm supports file-system resizing, extent reorganization, directory reorganization, and querying or changing the largefiles flag. fsadm operates on file systems mounted for read/write access, however, the -o option can also operate on a special device containing a clean, unmounted file system. Only a privileged user can change the largefiles flag on a mounted file system, or resize or reorganize a file system. You can invoke only one instance of fsadm per file system at a time.

If mount_point is a Storage Checkpoint, fsadm performs the specified operation on the entire file system, including all of its Storage Checkpoints.

VxFS file systems running on HP-UX 10.20 and later contain features that are incompatible with earlier versions of HP-UX and applications. These features are large files (file sizes greater than two gigabytes), and hierarchical storage management via the DMAPI (Data Management Applications Programming Interface).

The online reorganization and online resize features of fsadm are available only with the HP OnLineJFS product.

Options

fsadm recognizes the following options:

-a days  Consider files not accessed within the specified number of days as aged files. The default is 14 days. The -a days option applies only with -d.
-b newsize  Resize the file system to newsize sectors. -b cannot be used with the -c options.
-c  Convert the inode format of files upgraded from the Version 2 disk layout so that they can grow beyond a two-gigabyte offset. This option is required because some file systems upgraded from the Version 2 disk layout could not be extended past two gigabytes or contain more than 8 million inodes.
-d  Reorganize directories. Directory entries are reordered to place subdirectory entries first, then all other entries in decreasing order of time of last access. The directory is also compacted to remove free space.
-D  Report on directory fragmentation. If specified in conjunction with the -d option, the fragmentation report is produced both before and after the directory reorganization.
-e  Reorganize extents. Minimize file system fragmentation. Files are reorganized to have the minimum number of extents.
-E  Report on extent fragmentation. If specified in conjunction with the -e option, the fragmentation report is produced both before and after the extent reorganization.
-f filename  Performs and reports extent reorganization or directory reorganization on an individual file or directory. For example, when used with the -d option, a reorganization is performed on the specified directory. When used with the -e option, an extent reorganization is performed on a specified file. If you specify - (dash), the command reads input from the standard input device instead of using a file name or directory name.
-F vxfs  Specify the VxFS file system type.
-k ckpt_name
Reorganizes or reports directories or extent fragmentation on the specified Storage Check-
point.

-1 largesize
Large extent size in file system blocks. Indicates the size of extents to be considered as
large extents, that is, extents that are immovable while performing an extent defragmenta-
tion. The value must be between 8 and 2048 blocks. The -1 largesize option applies only with -E or -e.

-o specific_options
Specify options specific to the vxfs file system type.
The following specific_options are valid on a VxFS file system:

largefiles
Set the largefiles flag for the file system. When this flag is set, large files
(greater than two gigabytes) can be created on the file system.

nolargefiles
Clear the largefiles flag for the file system. When this flag is not set, large files
cannot be created on the file system. Any attempt to clear the flag fails if a large file
exists on the file system.

You can set or clear the flag on a mounted file system by specifying mount_point, or on an
unmounted file system on the device special.

When invoked without arguments, fsadm prints the current state of the largefiles flag.
The -o option cannot be used with the -bcdDeE options.

Note: Large files are supported on HP-UX 10.20 systems and above. Be careful when
implementing large file system capability. System administration utilities such as backup
may not operate correctly if they are not large file aware.

-p passes
Maximum number of passes to run. The default is 5 passes. Reorganizations are processed
until reorganization is complete, or the specified number of passes are run.

-r rawdev
Pathname of raw device to read to determine file layout and fragmentation. This option
can be used when fsadm cannot determine the raw device.

-s
Print a summary of activity at the end of each pass.

-t time
Maximum time to run. Reorganizations are processed until reorganization is complete, or
the time limit has expired. time is specified in seconds.

-v
Echo the completed command line, but do not execute the command. The command line is
generated by incorporating the user-specified options. This option allows the user to verify
the command line.

Specify verbose mode. Report reorganization activity.

If no options are specified, fsadm prints the current largefiles flag setting, then exits. The -b, -o
largefiles, and -o nolargefiles options cannot be specified if any other options are given. If
both -e and -d are specified, fsadm first completes the directory reorganization, then does the extent
reorganization.

Operands
fsadm recognizes the following operands:

mount_point
Name of the mount_point for a mounted VxFS file.

special
Name of a special device containing a clean, unmounted file system.

Largefiles Flag
Files larger than two gigabytes are called large files. The -o largefiles and -o nolargefiles
options change the largefiles flag, allowing or disallowing large files in the file system.
Large files can be created only on file systems with disk layout Version 3 or above. A file system with large files cannot be mounted on an HP-UX system older than HP-UX 10.20. Many existing applications cannot operate on large files.

Setting the flag with the `-o largefiles` option succeeds only if the file system has the Version 3 disk layout or above. See `vxupgrade(1M)` for information on how to upgrade a file system from an older disk layout to the current version. Clearing the flag with the `-o nolargefiles` option succeeds only if the flag is set and there are no large files present on the file system. See `mkfs_vxfs(1M)` and `mount_vxfs(1M)` for information on creating and mounting file systems with large files.

The `-o largefiles` and `-o nolargefiles` options are the only `fsadm` options that can be used on an unmounted file system. An unmounted file system can be specified by invoking `fsadm` with a special device rather than a mount point. If an unmounted file system is specified, it must be clean.

Changing the `largefiles` flag may require changes to `/etc/fstab`. For example, if `fsadm` is used to set the `largefiles` flag, but `nolargefiles` is specified as a mount option in `/etc/fstab`, the files system is not mountable.

**Defragmentation**

For optimal performance, the kernel-extent allocator must be able to find large extents when it wants them. To maintain file-system performance, run `fsadm` periodically against all VxFS file systems to reduce fragmentation. The frequency depends on file system usage and activity patterns, and the importance of performance; typically between once a day and once a month against each file system. The `-v` option can be used to examine the amount of work performed by `fsadm`. You can adjust the frequency of reorganization based on the rate of file system fragmentation.

There are two options that are available to control the amount of work done by `fsadm`. The `-t` option specifies a maximum length of time to run. The `-p` option specifies a maximum number of passes to run. If both are specified, `fsadm` exits if either of the terminating conditions is reached. By default, `fsadm` runs 5 passes. If both the `-e` and `-d` options are specified, `fsadm` runs all the directory reorganization passes before any extent reorganization passes.

`fsadm` uses the file `.fsadm` in the `lost+found` directory as a lock file. When `fsadm` is invoked, it opens the file `lost+found/.fsadm` in the root of the file system specified by `mount_point`. If the file does not exist, it is created. The `fcntl(2)` system call obtains a write lock on the file. If the write lock fails, `fsadm` assumes that another instance of `fsadm` is running and fails. `fsadm` reports the process ID of the process holding the write lock on the `.fsadm` file.

**File System Resizing**

If the `-b` option is specified, `fsadm` resizes the file system whose mount point is `mount_point`. If `newsize` is larger than the current size of the file system, the file system is expanded to `newsize` sectors. Similarly, if `newsize` is smaller than the current size of the file system, `fsadm` shrinks the file system to `newsize` sectors.

Increasing the size of a file system requires that the file system contain enough free space, prior to the expansion, for the growth of the structural files. In the case where a file system has no free blocks available, the attempt to increase the size of the file system will fail (see `extendfs(1M)` for an alternate method to increase file system size).

In a Version 3 or above disk layout, if there are file system resources in use in the sectors being removed, `fsadm` relocates those resources to sectors staying within the resized file system. The time needed for relocation depends on the number of blocks being moved.

In older disk layouts, file system structural components are fixed, so reducing the size of a file system fails if there are file system resources in use in the sectors being removed. In that case, a reorganization (using `fsadm -e`) can free busy resources and allow shrinking the file system. If there are still file system structural components within the area to be removed, you must upgrade the file system to a Version 3 or above disk layout to do a resize (see `vxupgrade(1M)`).

**Reporting on Directory Fragmentation**

As files are allocated and freed, directories tend to grow and become sparse. In general, a directory is as large as the largest number of files it ever contained, even if some files were subsequently removed.

To obtain a directory fragmentation report, use the command syntax:

```
fsadm -D [-r rawdev] mount_point
```
The following is some example output from the `fsadm -D` command:

```
# fsadm -F vxfs -D /lhome

Directory Fragmentation Report

<table>
<thead>
<tr>
<th></th>
<th>Total</th>
<th>Immed</th>
<th>Immeds</th>
<th>Dirs to Blocks to Searched</th>
<th>Dirs to Add</th>
<th>Reduce</th>
<th>Reduce</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>15</td>
<td>3</td>
<td>12</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
```

The column labeled "Dirs Searched" contains the total number of directories. A directory is associated with the extent-allocation unit containing the extent in which the directory's inode is located. The column labeled "Total Blocks" contains the total number of blocks used by directory extents.

The column labeled "Immed Dirs" contains the number of directories that are immediate, meaning that the directory data is in the inode itself, as opposed to being in an extent. Immediate directories save space and speed up pathname resolution. The column labeled "Immeds to Add" contains the number of directories that currently have a data extent, but that could be reduced in size and contained entirely in the inode.

The column labeled 'Dirs to Reduce' contains the number of directories for which one or more blocks could be freed if the entries in the directory are compressed to make the free space in the directory contiguous. Since directory entries vary in length, it is possible that some large directories may contain a block or more of total free space, but with the entries arranged in such a way that the space cannot be made contiguous. As a result, it is possible to have a non-zero "Dirs to Reduce" calculation immediately after running a directory reorganization. The `-v` (verbose) option of directory reorganization reports occurrences of failure to compress free space.

The column labeled "Blocks to Reduce" contains the number of blocks that could be freed if the entries in the directory are compressed.

### Measuring Directory Fragmentation

If the totals in the columns labeled "Dirs to Reduce" are substantial, a directory reorganization can improve performance of pathname resolution. The directories that fragment tend to be the directories with the most activity. A small number of fragmented directories may account for a large percentage of name lookups in the file system.

### Directory Reorganization

If the `-d` option is specified, `fsadm` reorganizes the directories on the file system whose mount point is `mount_point`. Directories are reorganized in two ways: compression and sorting.

For compression, `fsadm` moves valid entries to the front of the directory and groups the free space at the end of the directory. If there are no entries in the last block of the directory, the block is released and the directory size is reduced.

If the total space used by all directory entries is small enough, `fsadm` puts the directory in the inode immediate data area.

`fsadm` also sorts directory entries to improve pathname lookup performance. Entries are sorted based on the last access time of the entry. The `-a` option specifies a time interval; 14 days is the default if `-a` is not specified. The time interval is broken up into 128 buckets, and all times within the same bucket are considered equal. All access times older than the time interval are considered equal, and those entries are placed last. Subdirectory entries are placed at the front of the directory and symbolic links are placed after subdirectories, followed by the most-recently-accessed files.

The command syntax for reorganizing directories in a file system is:

```
```

The following example shows the output of the `fsadm -d -D` command:

```
#fsadm -F vxfs -d -D -s /opt

Directory Fragmentation Report

<table>
<thead>
<tr>
<th></th>
<th>Total</th>
<th>Immed</th>
<th>Immeds</th>
<th>Dirs to Blocks to Searched</th>
<th>Dirs to Add</th>
<th>Reduce</th>
<th>Reduce</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>34663</td>
<td>8800</td>
<td>2569</td>
<td>2716</td>
<td>0</td>
<td>4428</td>
<td>2569</td>
</tr>
</tbody>
</table>

Directory Reorganization Statistics (pass 1 of 2)

<table>
<thead>
<tr>
<th></th>
<th>Total</th>
<th>Failed</th>
<th>Blocks to Ioctls</th>
<th>Reduced</th>
<th>Blocks to Added</th>
<th>Immeds</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>999</td>
<td>8008</td>
<td>3121</td>
<td>5017</td>
<td>0</td>
<td>279</td>
</tr>
</tbody>
</table>
The column labeled "Dirs Searched" contains the number of directories searched. Only directories with data extents are reorganized. Immediate directories are skipped. The column labeled "Dirs Changed" contains the number of directories for which a change was made.

The column labeled "Total Ioctls" contains the total number of VX_DIRSORT ioctls performed. Reorganization of directory extents is performed using this ioctl.

The column labeled "Failed Ioctls" contains the number of requests that failed for some reason. The reason for failure is usually that the directory being reorganized is active. A few failures should be no cause for alarm. If the -v option is used, all ioctl calls and status returns are recorded.

The column labeled "Blocks Reduced" contains the total number of directory blocks freed by compressing entries. The column labeled "Blocks Changed" contains the total number of directory blocks updated while sorting and compressing entries.

The column labeled "Immeds Added" contains the total number of directories with data extents that were compressed into immediate directories.

**Reporting on Extent Fragmentation**

As files are created and removed over time, the free extent map for an allocation unit changes from having one large free area to having many smaller free areas. This process is known as fragmentation. Also, when files increase in size (particularly when growth occurs in small increments) small files can be allocated in multiple extents. In the best case, each file that is not sparse would have exactly one extent (containing the entire file), and the free-extent map is one continuous range of free blocks.

Conversely, in a case of extreme fragmentation, there can be free space in the file system, none of which can be allocated. For example, on Version 2 disk layouts, the indirect-address extent size is always 8K long. This means that to allocate an indirect-address extent to a file, an 8K extent must be available. If no extent of 8K byes or larger is available, even though more than 8K of free space is available, an attempt to allocate a file into indirect extents fails and returns ENOSPC.

**Determining Fragmentation**

To determine whether a file system is fragmented, the free extents for that file system must be examined. If a large number of small extents are free, then there is fragmentation. If more than half of the amount of free space is taken up by small extents (smaller than 64 blocks), or there is less than 5 percent of total file system space available in large extents, then there is serious fragmentation.

**Running the Extent-Fragmentation Report**

The extent-fragmentation report provides detailed information about the degree of fragmentation in a given file system.

The command syntax for an extent-fragmentation report is:

```
fsadm -E [-l largesize] [-r rawdev] mount_point
```

The extent reorganization facility considers some extents to be immovable: that is, if reallocating and consolidating extents does not improve performance, those extents are considered immovable. For example, if a file already contains large extents, reallocating and consolidating these extents does not improve performance. The -l option controls when fsadm considers an extent as immovable. By default, largesize is 64 blocks, meaning that any extent larger than 64 blocks is considered to be immovable. For the extent-fragmentation report, the value for largesize affects which extents are reported as being immovable extents.

The following is an example of the output generated by the `fsadm -E` command:

```
# fsadm -F vxfs -E /home
```

Hewlett-Packard Company
### Extent Fragmentation Report

<table>
<thead>
<tr>
<th>Total</th>
<th>Average</th>
<th>Average</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Files</td>
<td>File Blks</td>
<td># Extents</td>
<td>Free Blks</td>
</tr>
<tr>
<td>9293</td>
<td>115</td>
<td>1</td>
<td>149352</td>
</tr>
</tbody>
</table>

Blocks used for indirects: 48

- Free blocks in extents smaller than 64 blks: 10.40%
- Free blocks in extents smaller than 8 blks: 0.56%
- Blocks allocated to extents 64 blks or larger: 91.67%

#### Free Extents By Size

<table>
<thead>
<tr>
<th>Size</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>156</td>
</tr>
<tr>
<td>2</td>
<td>140</td>
</tr>
<tr>
<td>4</td>
<td>101</td>
</tr>
<tr>
<td>8</td>
<td>292</td>
</tr>
<tr>
<td>16</td>
<td>290</td>
</tr>
<tr>
<td>32</td>
<td>241</td>
</tr>
<tr>
<td>64</td>
<td>94</td>
</tr>
<tr>
<td>128</td>
<td>256</td>
</tr>
<tr>
<td>256</td>
<td>43</td>
</tr>
<tr>
<td>512</td>
<td>1024</td>
</tr>
<tr>
<td>1024</td>
<td>20</td>
</tr>
<tr>
<td>2048</td>
<td>1</td>
</tr>
<tr>
<td>4096</td>
<td>8</td>
</tr>
<tr>
<td>8192</td>
<td>1</td>
</tr>
<tr>
<td>16384</td>
<td>0</td>
</tr>
<tr>
<td>32768</td>
<td>0</td>
</tr>
<tr>
<td>65536</td>
<td>0</td>
</tr>
<tr>
<td>131072</td>
<td>0</td>
</tr>
<tr>
<td>262144</td>
<td>0</td>
</tr>
<tr>
<td>524288</td>
<td>0</td>
</tr>
<tr>
<td>1048576</td>
<td>0</td>
</tr>
<tr>
<td>2097152</td>
<td>0</td>
</tr>
<tr>
<td>4194304</td>
<td>0</td>
</tr>
<tr>
<td>8388608</td>
<td>0</td>
</tr>
<tr>
<td>16777216</td>
<td>0</td>
</tr>
<tr>
<td>33554432</td>
<td>0</td>
</tr>
<tr>
<td>67108864</td>
<td>0</td>
</tr>
<tr>
<td>134217728</td>
<td>0</td>
</tr>
<tr>
<td>268435456</td>
<td>0</td>
</tr>
<tr>
<td>536870912</td>
<td>0</td>
</tr>
<tr>
<td>1073741824</td>
<td>0</td>
</tr>
</tbody>
</table>

The numbers in the column "Total Files" indicate the total number of files that have data extents. The column "Average File Blks" contains the average number of blocks belonging to all files. The column "Average # Extents" contains the average number of extents used by files in the file system. The column "Total Free Blks" contains the total number of free blocks in the file system. The total number of blocks used for indirect address extent are reported as "blocks used for indirects".

The general shape of free extent map is also reported. There are two percentages reported: % free extents smaller than 64 blocks and % free extents smaller than 8 blocks. These numbers are typically near zero on an unfragmented file system.

Another metric reported is the percentage of blocks that are part of extents 64 blocks or larger. Files with a small extent are not included in this calculation. This number is generally large on file systems that contain many large files, and is small on file systems that contain many small files.

The figures under the heading 'Free Extents By Size' indicate the totals for free extents of each size. The totals are for free extents of size 1, 2, 4, 8, 16, ... up to a maximum of the number of data blocks in an allocation unit. The totals are similar to the output of the `df -o` command unless there was recent allocation or deallocation activity (because `fsadm` acts on mounted file systems). These figures provide an indication of fragmentation and extent availability on a file system.

### Extent Reorganization

If the `-e` option is specified, `fsadm` reorganizes the data extents on the file system whose mount point is `mount_point`. The primary goal of extent reorganization is to defragment the file system.

To reduce fragmentation, extent reorganization tries to place all small files in one contiguous extent. The `-l` option specifies the size of a file that is considered large. The default is 64 blocks. Extent reorganization also tries to group large files into large extents of at least 64 blocks. Extent reorganization can improve performance. Small files can be read or written in one I/O operation. Large files can approach raw-disk performance for sequential I/O operations.

`fsadm` performs extent reorganization on all inodes on the file system. Each pass through the inodes will move the file system closer to optimal organization.

`fsadm` reduces both file fragmentation and free extent fragmentation in each pass. In older versions of VxFS, considerable effort was made to obtain an optimal file system layout. In current versions, `fsadm` relies on VxFS kernel allocation mechanisms to reallocate files in a more favorable extent geometry. At the same time, the kernel allocation mechanism is prevented from using blocks in areas of the free list that `fsadm` tries to make more contiguous.

The command syntax to perform extent reorganization is

```
```

The following example shows the output from the `fsadm -F vxfs -e -s -E` command:

```
# fsadm -F vxfs -e -s -E /home
```
### Extent Fragmentation Report

<table>
<thead>
<tr>
<th>Total</th>
<th>Average</th>
<th>Average</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Files</td>
<td>File Blks</td>
<td># Extents</td>
<td>Free Blks</td>
</tr>
<tr>
<td>9293</td>
<td>115</td>
<td>1</td>
<td>149352</td>
</tr>
</tbody>
</table>

- blocks used for indirects: 48
- Free blocks in extents smaller than 64 blks: 10.40
- Free blocks in extents smaller than 8 blks: 0.56
- blks allocated to extents 64 blks or larger: 91.67

### Free Extents By Size

<table>
<thead>
<tr>
<th>Size</th>
<th>Extents</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>156</td>
</tr>
<tr>
<td>2</td>
<td>140</td>
</tr>
<tr>
<td>4</td>
<td>101</td>
</tr>
<tr>
<td>8</td>
<td>292</td>
</tr>
<tr>
<td>16</td>
<td>290</td>
</tr>
<tr>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td>64</td>
<td>94</td>
</tr>
<tr>
<td>128</td>
<td>256</td>
</tr>
<tr>
<td>512</td>
<td>33</td>
</tr>
<tr>
<td>1024</td>
<td>20</td>
</tr>
<tr>
<td>2048</td>
<td>1</td>
</tr>
<tr>
<td>4096</td>
<td>1</td>
</tr>
<tr>
<td>8192</td>
<td>1</td>
</tr>
<tr>
<td>16384</td>
<td>1</td>
</tr>
<tr>
<td>32768</td>
<td>0</td>
</tr>
<tr>
<td>65536</td>
<td>0</td>
</tr>
<tr>
<td>131072</td>
<td>0</td>
</tr>
<tr>
<td>262144</td>
<td>0</td>
</tr>
<tr>
<td>524288</td>
<td>0</td>
</tr>
<tr>
<td>1048576</td>
<td>0</td>
</tr>
<tr>
<td>2097152</td>
<td>0</td>
</tr>
<tr>
<td>4194304</td>
<td>0</td>
</tr>
<tr>
<td>8388608</td>
<td>0</td>
</tr>
<tr>
<td>16777216</td>
<td>0</td>
</tr>
<tr>
<td>33554432</td>
<td>0</td>
</tr>
<tr>
<td>67108864</td>
<td>0</td>
</tr>
<tr>
<td>134217728</td>
<td>0</td>
</tr>
<tr>
<td>268435456</td>
<td>0</td>
</tr>
<tr>
<td>536870912</td>
<td>0</td>
</tr>
<tr>
<td>1073741824</td>
<td>0</td>
</tr>
</tbody>
</table>

### Pass 1 Statistics

<table>
<thead>
<tr>
<th>Extents</th>
<th>Reallocations</th>
<th>Ioctls</th>
<th>Errors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Searched</td>
<td>Attempted</td>
<td>Issued</td>
<td>FileBusy</td>
</tr>
<tr>
<td>Total</td>
<td>12547</td>
<td>287</td>
<td>158</td>
</tr>
</tbody>
</table>

### Pass 2 Statistics

<table>
<thead>
<tr>
<th>Extents</th>
<th>Reallocations</th>
<th>Ioctls</th>
<th>Errors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Searched</td>
<td>Attempted</td>
<td>Issued</td>
<td>FileBusy</td>
</tr>
<tr>
<td>Total</td>
<td>13157</td>
<td>148</td>
<td>72</td>
</tr>
</tbody>
</table>

Note that the default five passes were scheduled, but the reorganization finished in two passes.

This file system had a significant amount of free space although there were several free small extents. The situation was corrected by reallocating one or more of the extents on many of the files. The files selected for reallocation in this case are those with extents in the heavily fragmented section of the allocation units. The time it takes to complete extent reorganization varies, depending on the degree of fragmentation, disk speed, and the number of inodes in the file system. In general, extent reorganization takes approximately one minute for every 100 megabytes of disk space.

In the preceding example, the column "Extents Searched" contains the total number of extents examined. The column "Reallocations Attempted" contains the total number of consolidations or merging of extents performed. The column "Ioctls Issued" contains the total number of reorganization request calls made during the pass. This corresponds closely to the number of files that are being operated on in that pass as most
files can be reorganized with a single ioctl. (More than one extent may be consolidated in one operation.)

The column "FileBusy" (located under the heading "Errors") shows the total number of reorganization requests that failed because the file was active during reorganization. The column "NoSpace" (located under the heading "Errors") contains the total number of reorganization requests that failed because an extent presumed free was allocated during the reorganization. The column "Total" (located under the heading "Errors") is the total number of errors encountered during the reorganization and may include errors that were not included with "FileBusy" or "NoSpace."

The following command performs an extent reorganization on all individual files under the mount point /home.

    find /home -print | fsadm -F vxfs -e -f -

FILES

lost+found/.fsadm  lock file
/dev/rdsk/  *  file system devices
/etc/fstab  Contains static information about file systems.

SEE ALSO

    df_vxfs(1M) fsadm(1M), mkfs_vxfs(1M), mount_vxfs(1M), vxupgrade(1M), fcntl(2), fstab(4), vxfsio(7).
fscat_vxfs(1M)

NAME
fscat_vxfs: fscat - cat a VxFS file system

SYNOPSIS

DESCRIPTION
fscat provides an interface to a VxFS snapshot file system, similar to that provided by dd, invoked on the
block or character special file of regular VxFS file systems. fscat works when executed on the special
device of any VxFS file system.

On most VxFS file systems, the block or character special file for the file system provides access to a raw
image of the file system to back up the file system to tape. On a snapshot file system, access to the
 corresponding block or character special file provides little useful information. fscat provides a stream
of bytes representing the file system snapshot. This data stream is written by default to standard output,
although the -f output-file option can be used to specify another destination. The data stream on standard
output can be processed several ways, such as in a pipeline, or written to a tape.

By default, the output is a stream of bytes that starts at the beginning of the file system and continues to
the last byte. On a snapshot file system, data is read from the file system using special ioctls on the mount
point. On other VxFS file systems, data is read from the specified
special file. Unless otherwise specified,
data is written to standard output.

All numbers entered as option arguments can have 0 as a prefix to indicate octal, or 0x as a prefix to indi-
cate hexadecimal. A b or B can be appended to indicate the value is in 512-byte blocks, a k or K to indicate
the value is in kilobytes, an m or M to indicate the value is in megabytes, or a g or G to indicate the value is
in gigabytes. An appended letter can be separated from the number by a space, in which case the letter
and number should be enclosed in a set of quotes. For example:

"512 b"

All numbers entered as options must be in multiples of 512 bytes. For example, a value of 5713 as an offset
is rejected.

Options
fscat recognizes the following options:

- b block_size Specify the output block size, in bytes. block_size must be less than or equal to 1 megabyte.
- F vxfs Specify the VxFS file system type.
- f output-file Specify an output file in which to write the data stream.
- l length Specify the transfer length, in bytes. A length of 0 includes the remainder of the file sys-
tem after the specified offset.
- o offset Specify the starting offset in bytes.

Operands
fscat recognizes the following operand:
special Name of the special device from which the VxFS file system is accessed.

Notes
Snapshot file systems are only available with the HP OnLineJFS product.

A snapshot file system cannot be written to.

A snapshot file system exists only as long as it is mounted; once unmounted, the special file no longer con-
tains a snapshot file system.

When fscat is run on a mounted VxFS snapshot, the content of free blocks (that is, blocks not allocated to
any file or metadata) is undefined. The content of free blocks can change after additional data is written to
the primary (snapped) file system, but blocks associated with files always display the content they had
when the snapshot was created.

fscat does not work with Storage Checkpoints.

284 Hewlett-Packard Company – 1 – HP-UX 11i Version 2: December 2007 Update
SEE ALSO
   dd(1), fs_vxfs(4), vxfsio(7).
NAME
fsck - file system consistency check and interactive repair

SYNOPSIS
/usr/sbin/fsck [-F FSType] [-m] [-s] [-V] [special ...]
/usr/sbin/fsck [-F FSType] [-o FSpecific-options] [-s] [-V] [special ...]

DESCRIPTION
The fsck command audits and interactively repairs inconsistent conditions for HP-UX file systems on
mass storage device files identified by special. If the file system is consistent, the number of files on that
file system and the number of used and free blocks are reported. If the file system is inconsistent, fsck
provides a mechanism to fix these inconsistencies, depending on which form of the fsck command is used.
special represents a special device (e.g., /dev/rdsk/c1d0s8).

Options
fsck recognizes the following options:
  -F FSType Specify the file system type on which to operate (see ftyp(1M) and fs_wrapper(5)). If
this option is not included on the command line, then the file system type is deter-
mined from the file /etc/fstab by matching special with an entry in that file. If
there is no entry in /etc/fstab, then the file system type is determined from the
file /etc/default/fs.
  -m Perform a sanity check only. fsck will return 0 if the file system is suitable for
mounting. If the file system needs additional checking, the return code is 32. If the
file system is mounted, the return code is 33. Error codes larger than 33 indicate that
the file system is badly damaged.
  -o FSpecific-options Specify options specific to each file system type. FSpecific-options is a list of subop-
tions and/or keyword/attribute pairs intended for a file-system-specific version of
the command. See the file-system-specific manual entries for a description of the
specific_options supported, if any.
  -s Safe performance mode. To improve performance, a system wide sync(2) will not be
issued. This feature is underlying file system dependent.
  -V Echo the completed command line, but perform no other action. The command line is
generated by incorporating the user-specified options and other information derived
from /etc/fstab. This option allows the user to verify the command line.

RETURN VALUE
The following values are returned by the -m option to fsck:
  0 Either no errors were detected or all errors were corrected.
  32 The file system needs additional checking.
  33 The file system is mounted.
Return values greater that 33 indicate that file system is badly corrupted. File system specific versions of
fsck will have their own additional return values (see fsck_cachefs(1M), fsck_hfs(1M), or fsck_vxfs(1M)).

WARNINGS
This command may not be supported for all file system types.

FILES
/etc/default/fs Specifies the default file system type
/etc/fstab Default list of file systems to check

SEE ALSO
fsck_cachefs(1M), fsck_hfs(1M), fsck_vxfs(1M), mkfs(1M), newfs(1M), fstab(4), fs_wrapper(5).

STANDARDS CONFORMANCE
fsck: SVID3

286  Hewlett-Packard Company
NAME
fsck_cachefs: fsck - check integrity of data cached with CacheFS

SYNOPSIS
fsck -F cachefs [ -m | -o noclean ] cache_directory

DESCRIPTION
The CacheFS version of the fsck command checks the integrity of a cache directory. By default it corrects any CacheFS problems it finds. There is no interactive mode. The most likely invocation of fsck for CacheFS file systems is at boot time from an entry in the /etc/fstab file.

Options
Two command line options are available:

- Check, but do not repair.
- Force a check on the cache even if there is no reason to suspect there is a problem.

EXAMPLES
The following example forces a check on the cache directory /cache3:

fsck -F cachefs -o noclean /cache3

AUTHOR
fsck_cachefs was developed by Sun Microsystems, Inc.

SEE ALSO
cfsadmin(1M), fsck(1M), mount_cachefs(1M).
NAME
fsck_hfs: fsck - HFS file system consistency check and interactive repair

SYNOPSIS
/usr/sbin/fsck [-F hfs] [-m] [-s] [-V] [-b blocknum] [special]...
[-q] [-s] [-V] [special]...

DESCRIPTION
The fsck command audits and repairs inconsistent conditions for HFS file systems on mass storage device files identified by special. If the file system is consistent, the number of files on that file system and the number of used and free blocks are reported. If the file system is inconsistent, fsck provides a mechanism to fix these inconsistencies, depending on which form of the fsck command is used. special represents a special device (for example, /dev/rdsk/c1d0s8).

If the target device is a swap device, fsck does not continue to process. fsck also checks the target device to ensure a mounted file system is not being checked. If a mounted device is specified but the -f option is omitted, fsck prompts the user for a response.

If the -p|P option is used and special is not specified, fsck reads the pass numbers in /etc/fstab to determine which groups of disks to inspect in parallel, taking maximum advantage of I/O overlap to process the file systems as quickly as possible. The -p|P option is normally used in the script /sbin/bcheckrc during automatic reboot.

Normally, the root file system is checked on pass 1, and other "root" (section 0) file systems on pass 2. Other small file systems are checked on separate passes (such as the section 4 file systems on pass 3 and the section 7 file systems on pass 4), and finally the large user file systems are checked on the last pass (for example, pass 5). A pass number of 0 in /etc/fstab causes a file system not to be checked. If the optional fields are not present on a line in /etc/fstab, fsck processes the file system on such lines sequentially after all eligible file systems with positive pass numbers have been processed.

The inconsistencies that fsck with the -p|P option corrects are shown below. These are inconsistencies that are correctable without data loss. If it encounters other inconsistencies, it exits with an abnormal return status. For each corrected inconsistency, one or more lines are printed identifying the file system on which the correction will take place and the nature of the correction. Correctable inconsistencies are limited to the following:

- Unreferenced inodes
- Unreferenced continuation inodes
- Unreferenced pipes and FIFOs
- Link counts in inodes too large
- Missing blocks in the free list
- Blocks in the free list also in files
- Counts in the superblock wrong.

The -P option operates in the same manner as the -p option except that cleanly unmounted file systems are not checked (see fsclean(1M)). This can greatly decrease the amount of time required to reboot a system that was brought down cleanly.

If the -p|P option is not specified, the pass numbers are ignored and the file systems are checked interactively in the order they are listed in /etc/fstab.

Without the -p|P option, fsck prompts for concurrence before each correction is attempted when the file system is inconsistent. It should be noted that some corrective actions result in a loss of data. The amount and severity of data loss can be determined from the diagnostic output. The default action for each consistency correction is to wait for the operator to respond yes or no. If the operator does not have write permission, fsck defaults to a -n action.

Options
fsck recognizes the following options:

- -F hfs Specify the HFS file system.
- -c size Set the size of the buffer cache which fsck uses to cache disk blocks. size is the number of cache blocks, and is between 0 and 100 inclusive. The most common use of this option is...
-c 0 to disable all caches, thus reducing memory usage.

-b blocknum
Use the specified blocknum as the superblock for the file system. An alternate superblock can usually be found at block \(((\text{SBSIZE}+\text{BBSIZE})/\text{DEV_BSIZE})\), typically block 16. DEV_BSIZE is defined in `<sys/param.h>`. You can also find a list of alternate superblocks in `/var/adm/sbtab` (see `mkfs(1M)`).

-f Force `fsck` to check a mounted file system.

-m Perform a sanity check only. Verify whether `special` is mounted, or needs additional checking. Refer to the RETURN VALUE section for more information.

-n | -N Assume a no response to all questions asked by `fsck` about repairing a file system. Do not open the file system for writing.

-p "Preen" the file system. Proceed to process and repair file systems without user interaction, as described above. Exit immediately if there is a problem requiring intervention.

-P Same as -p except that cleanly unmounted file systems are not checked.

-q Quiet. Do not print size-check messages in Phase 1. Unreferenced fifos are silently removed. If `fsck` requires it, counts in the superblock and cylinder groups are automatically fixed.

-s Safe performance mode. To improve performance, a system wide `sync(2)` will not be issued.

-V Echo the completed command line, but perform no other actions. The command line is generated by incorporating the user-specified options and other information derived from `/etc/fstab`. This option allows the user to verify the command line.

-y | -Y Assume a yes response to all questions asked by `fsck` about repairing a file system. This should be used with great caution, because this is a free license to continue after essentially unlimited trouble has been encountered.

In all cases, `fsck` checks the following inconsistencies:

- Blocks claimed by more than one inode or the free list.
- Blocks claimed by an inode or the free list outside the range of the file system.
- Incorrect link counts.
- Size checks:
  - Directory size not of proper format.
  - Bad inode format.
  - Blocks not accounted for anywhere.
- Directory checks:
  - File pointing to unallocated inode.
  - Inode number out of range.
- Superblock checks:
  - More blocks for inodes than there are in the file system.
  - Bad free block list format.
  - Total free block and/or free inode count incorrect.
  - Invalid continuation inode number in a primary inode.

Orphaned files and directories (allocated but unreferenced) are, with the operator's concurrence, reconciled by placing them in the `lost+found` directory. The name assigned is the inode number. The only restriction is that the directory `lost+found` must have empty slots in which entries can be made. This is accomplished by copying a number of files to the directory, then removing them before `fsck` is executed.

Unreferenced continuation inodes are removed with the -p option, since they do not refer back to the primary inode. When a primary inode contains an invalid continuation inode number, the continuation inode number should be cleared (that is, set to 0). This is not done automatically (with the -p option), because access control list information may have been lost and should be corrected.

After `fsck` has checked and fixed the file system, it stores the correct `fs_clean` flag in the superblock if it is not already there. For a nonroot file system, `FS_CLEAN` is stored there. For the root file system, which is mounted at the time of the `fsck`, no changes are required to the superblock if no problems were found and `FS_OK` was already set.

Checking the raw device is almost always faster.
RETURN VALUE

fsck returns the following values:

- **0**: Either no errors were detected or all errors were corrected.
- **1**: A syntax error or other operational error occurred when invoked with the -V option.
- **4**: Root file system errors were corrected. The system must be rebooted.
- **8**: Some uncorrected errors exist on one or more of the file systems checked, there was a syntax error, or some other operational error occurred.
- **12**: A signal was caught during processing.
- **32**: The file system is unmounted and needs additional checking.
- **33**: The file system is mounted.
- **34**: The file system is damaged.

WARNINGS

fsck should not be run on mounted file systems or on the root device. If you do run on mounted file systems, be sure the system is in single-user state (see shutdown(1M)).

The special case of the -c option, -c 0, will disable all internal caches, which will reduce memory usage but may impact performance.

The -F option, from prior releases, has been replaced by the -f option.

AUTHOR

fsck was developed by HP, AT&T, the University of California, Berkeley.

FILES

/etc/fstab: Default list of file systems to check.
/var/adm/sbtab: List of locations of the superblocks for file systems. The mkfs command appends entries to this file.

SEE ALSO

dumpfs(1M), fsck(1M), fsck_vxfs(1M), fsck(1M), mkfs(1M), newfs(1M), shutdown(1M), fstab(4), acl(5),
fs_wrapper(5).

STANDARDS CONFORMANCE

fsck: SVID3
fsck_vxfs(1M)  fsck_vxfs(1M)

NAME
fsck_vxfs: fsck - check and repair a VxFS file system

SYNOPSIS
fsck [-F vxfs] [-V] [-mNpPsY] [-pP] [-o p] [special ... ]
fsck [-F vxfs] [-V] [-mNpPsY] [-o full,mounted,nolog] [special ... ]

DESCRIPTION
fsck checks VxFS file systems for consistency. Because VxFS records pending file system updates in an
intent log, fsck typically replays the intent log instead of doing a full structural file system check. You
can use options (-o full or -y) to force a full structural file system check.

special specifies one or more special character devices, for example, /dev/rdsk/c1t0d0.

If multiple devices are specified, each device is checked in turn unless the -P option or the -o p suboption
is also specified, in which case the devices are checked in parallel. If special is not specified, fsck prompts
you with each file system listed in /etc/fstab to determine which file system to check, unless you
specify -y or -Y, to automatically answer yes to the prompts.

Options
fsck recognizes the following options:

-F vxfs       Specify the VxFS file system type.
-m            Check whether or not the file system is marked clean. This option does not validate the file
              system. The file system could have been corrupted since it was marked clean (for example,
              by a system crash), and if so, a mount could fail. In that case, a full fsck would be
              required to clean it. Use fsck -n to test for file system corruption.
-n | -N         Assume a "no" response to all prompts by fsck; do not open the file system for writing, do
              not replay the intent log. A full file system check is performed.
-o specific_options
              Specify VxFS file system specific options. See the subsection The -o Specific Options
              below.
-p            Produce messages that identify the device being checked.
-P            With VxFS, -P is used by fsck by default; it does not provide any functionality.
-s            Safe performance mode. To improve performance, a system wide sync() will not be
              issued (see sync(2)).
-V            Echo the completed command line, but do not execute the command. The command line is
              generated by incorporating the user specified options and other information derived from
              /etc/fstab. This option allows the user to verify the command line.
-y | -Y        Assume a "yes" response to all prompts by fsck. Additionally, if the file system requires a
              full file system check after the log replay, or if the nolog suboption causes the log replay
to be skipped and the file system is not clean, then a full file system check is performed.

Because VxFS maintains an intent log, a complete check is generally not required; the default is to replay
the intent log only. If fsck_vxfs detects file system damage or the log replay operation detects damage,
an indication that a complete check is required is placed in the super-block. In this case, if the -y option
was specified, the full check will be run after the log replay. If the -y option was not used, fsck must be
run again, with the -o full option to perform the full structural check.

Operands
fsck recognizes the following operand:

special       Name of one or more special character devices which contain VxFS file systems.

The -o Specific Options
The -o option specifies VxFS file system specific options. These options can be a combination of the follow-
ing in a comma-separated list:

full           Perform a full file system check.
mounted        Allows a full check of a mounted file system. -o mounted is only used internally as part
                of the primary cluster node recovery process after the primary fails. Never enter this
option from the command line as it can destroy a file system if not used correctly.

**nolog**
Do not perform log replay. This option may be used if the log area was physically damaged.

**Note:** Use the -n option to verify whether there are file system inconsistencies. Use `fsck -o full,nolog` to fix a corrupted file system and avoid a log replay. If you run `fsck -o full` without nolog on a clean file system, it replays the intent log and performs a full file system check.

**p**
Allows parallel log replay for several VxFS file systems. Each message from `fsck` is prefixed with the device name to identify the device. This suboption does not perform a full file system check in parallel; that is still done sequentially on each device, even when multiple devices are specified. This option is compatible only with the `-y` or `-Y` option (that is, non-interactive full file system check), in which case a log replay is done in parallel on all specified devices. A sequential full file system check is performed on devices where needed. The number of devices that can be checked in parallel is determined by the amount of physical memory in the system. One instance of `fsck` on a single device can consume up to a maximum of 32 megabytes of memory.

### Check a File System
A full check looks for the following inconsistencies:

- Blocks claimed by more than one inode or the free list.
- Blocks claimed by an inode outside the range of the file system.
- Incorrect link counts.
- Size checks:
  - Incorrect number of blocks.
  - Directory entry format.
- Bad inode format.
- Blocks not accounted for anywhere.
- Directory checks:
  - File pointing to unallocated inode.
  - Inode number out of range.
  - Linkage to parent directory.
  - Hash chain linkage.
  - Free space count.
- Super-block checks:
  - Checksum mismatch.
  - More blocks for inodes than there are in the file system.
- Structural Files:
  - Fileset headers.
  - Object Location Table (OLT).
  - Inode list files.
  - Inode allocation summary files.
  - Attribute files (including Access Control Lists).
  - Attribute link counts.
- Bad free block list format.
- Total free block and/or free inode count incorrect.

### Lost and Found Directory
Orphaned files and directories (allocated but unreferenced) are, with the user’s agreement, reconnected by placing them in the `lost+found` directory. The name assigned is the inode number. The only restriction is that the directory `lost+found` must already exist in the file system’s root directory.

### Notes
Checking the raw device is almost always faster.

Unlike 2.x and earlier releases of VxFS, a full file system check does not always perform pending extended inode operations. Some extended operations can only be processed when the file system is mounted. A file system that has been marked `CLEAN` can still contain extended operations.

If a structural flaw is detected during the intent log replay, the full `fsck` flag is set on the file system without operator interaction.

If `fsck` encounters a large file on an older OS version, the command stops without completing the file system check.
RETURN VALUES

Structural errors discovered during a full check are displayed on standard output. Responses required during a full check are read from standard input.

The following return codes are used for the -m option for all devices other than the one used by the root file system:

0  The file system is unmounted and clean.
32  The file system is unmounted and needs checking.
33  The file system is mounted.
34  The stat of the device failed.

Other

The state could not be determined because of an error.

The following return codes are used for the -m option for the device used by the root file system:

0  The root file system is mounted read-only and is clean, or the root file system is mounted read/write and therefore is clean.
32  The root file system is mounted read-only and needs checking.
34  The stat of the device failed.

Other

The state could not be determined because of an error.

In most cases, fsck prints the following messages:

log replay in progress
replay complete - marking super-block as CLEAN

If the file system is already clean, fsck prints the following message instead:

file system is clean - log replay is not required

If fsck prints any other messages, a full structural check is needed. If the -y option is specified, fsck performs (if necessary) a full check after running the intent log replay. If the -y option is not used, fsck must be invoked with the -o full option to perform a full structural check.

If -o p or -P is specified, fsck prints the following messages for a device, for example /dev/rdsk/c0t0d0:

/dev/rdsk/c0t0d0:log replay in progress
/dev/rdsk/c0t0d0:replay complete - marking super-block as CLEAN

DIAGNOSTICS

All error messages that relate to the contents of a file system produced during a log replay are displayed on standard output. All I/O failures and exit messages are displayed on standard error.

WARNINGS

-o mounted allows a full check of a mounted file system. -o mounted is only used internally as part of the primary cluster node recovery process after the primary fails. Never enter this option from the command line as it can destroy a file system if not used correctly.

The -s (safe performance mode) option will be obsoleted in future releases.

SEE ALSO

fsck(1M), fsck_hfs(1M), mkfs(1M), mkfs_vxfs(1M), ncheck_vxfs(1M), sync(2), fs_vxfs(4).
NAME
fsclean - determine the shutdown status of HFS file systems

SYNOPSIS
/sbin/fsclean [-q] [-v] [special ...]

DESCRIPTION
The fsclean command determines the shutdown status of the HFS file system specified by special or, in the absence of special, the file systems listed in /etc/fstab of type hfs with the rw, default, or ro options set. All optional fields in /etc/fstab must be present for fsclean to be able to check each file system.

fsclean reads the superblock to determine whether the file system's last shutdown was done correctly, and returns one of the following values:

0 All of the checked file systems were shut down correctly.
1 One or more checked file systems were not shutdown correctly, implying that fsck should be run (see fsck(1M)).
2 Other error (such as cannot open the specified device file).

The fsclean command is usually silent.

Options:
-q Check quotas. Instead of checking the file system shutdown status, fsclean checks the validity of disk quota statistics. This option is useful for determining whether quotacheck should be run (see quotacheck(1M)). If special is not provided, then all file systems in /etc/fstab of type hfs with the rw (or default) and quota options are checked.
-v Be verbose. Prints the status of each file system checked.

DEPENDENCIES
fsclean only operates on HFS file systems.

AUTHOR
fsclean was developed by HP.

FILES
/etc/fstab Default list of file systems to check

SEE ALSO
dumpfs(1M), fsck(1M), fsck_hfs(1M), mount(1M), quotacheck(1M), quotacheck_hfs(1M), reboot(1M), fstab(4).
NAME
fsdb - file system debugger (generic)

SYNOPSIS
/usr/sbin/fsdb [-F FStype] [-o specific_options] [-V] special

Remarks
Always execute the fsck command (see fsck(1M)) after running fsdb.

DESCRIPTION
The fsdb command can be used to patch up a damaged file system after a crash. It is intended for experienced users only. The file system type to be debugged is specified as FStype. Each file system type has a unique structure requiring different debugging capabilities. The manual entries for the file-system-specific fsdb should be consulted before attempting any debugging or modifications.

Options and Arguments
fsdb recognizes the following options and arguments:

special         The file name of the special file containing the file system.

-F FStype       Specify the file system type on which to operate (see fstyp(1M) and fs_wrapper(5)). If this option is not included on the command line, then the file system type is determined from the file /etc/fstab by matching special with an entry in that file. If there is no entry in /etc/fstab, then the file system type is determined from the file /etc/default/fs.

-o specific_options
Specify suboptions specific to each file system type. specific_options is a comma-separated list of suboptions and/or keyword/attribute pairs supported by the specific FStype.

-V               Echo the completed command line, but perform no other action. The command line is generated by incorporating the user-specified options and other information derived from the /etc/fstab file. This option allows the user to verify the command line.

EXAMPLES
Invoke the file system debugger on HFS file system /dev/dsk/c1d2s0:

fsdb -F hfs /dev/dsk/c1d2s0

Display a completed command line without executing the debugger:

fsdb -V /dev/dsk/c1d2s0

The previous command might display:

fsdb -F hfs /dev/dsk/c1d2s0

WARNINGS
Only experienced users should use fsdb. The failure to fully understand the usage of fsdb and the file system’s internal organization can lead to complete destruction of the file system and total loss of data.

AUTHORS
fsdb was developed by HP and AT&T.

FILES
/etc/default/fs    Specifies the default file system type
/etc/fstab        Static information about the file systems

SEE ALSO
fsck(1M), fsdb_hfs(1M), fsdb_vxfs(1M), fstyp(1M), stat(2), fs_wrapper(5).

STANDARDS CONFORMANCE
fsdb: SVID3
NAME
fsdb_hfs: fsdb - HFS file system debugger

SYNOPSIS
/usr/sbin/fsdb [-F hfs] [-V] special [-b blocknum] [-]

Remarks
Always execute the fsck command (see fsck(1M)) after running fsdb.

DESCRIPTION
The fsdb command can be used to patch up a damaged file system after a crash.

Options and Arguments
fsdb recognizes the following options and arguments.
   special
      The file name of the special file containing the file system.
   -    Initially disable the error-checking routines that are used to verify the inode and fragment addresses. See the O symbol. If used, this option must follow special on the command line.
   -b blocknum
      Use blocknum as the superblock for the file system. If used, this option must follow special on the command line.
   -F hfs
      Specify the HFS file system type.
   -V
      Echo the completed command line, but perform no other action. The command line is generated by incorporating the user-specified options and other information derived from the /etc/fstab file. This option allows the user to verify the command line.

Operation
fsdb normally uses the first superblock for the file system, located at the beginning of the disk section, as the effective superblock. An alternate superblock can always be found at block \((\text{BSIZE}+\text{BSIZE})/\text{DEV_BSIZE}\), typically block 16. The -b option can be used to specify the superblock location.

fsdb deals with the file system in terms of block fragments, which are the unit of addressing in the file system and the minimum unit of space allocation. To avoid possible confusion, fragment is used to mean that, and block is reserved for the larger true block. fsdb has conversions to translate fragment numbers and i-numbers into their corresponding disk addresses. Also included are mnemonic offsets to access different parts of an inode. These greatly simplify the process of correcting control block entries or descending the file system tree.

fsdb contains several error-checking routines to verify inode and fragment addresses. These can be disabled if necessary by invoking fsdb with the optional - argument, or by using the O symbol.

Numbers are considered decimal by default. Octal numbers must be prefixed with a zero. Hexadecimal numbers must be prefixed with 0x. During any assignment operation, numbers are checked for a possible truncation error due to a size mismatch between source and destination.

fsdb reads a fragment at a time. A buffer management routine is used to retain commonly used fragments of data in order to reduce the number of read system calls. All assignment operations result in an immediate write-through of the corresponding fragment.

Symbols
The following symbols are recognized by fsdb:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>!</td>
<td>Escape to shell</td>
</tr>
<tr>
<td>#</td>
<td>Absolute address</td>
</tr>
<tr>
<td>+</td>
<td>Address arithmetic</td>
</tr>
<tr>
<td>=</td>
<td>Numerical assignment</td>
</tr>
<tr>
<td>=+</td>
<td>Incremental assignment</td>
</tr>
<tr>
<td>-=</td>
<td>Decremental assignment</td>
</tr>
</tbody>
</table>

296   Hewlett-Packard Company
<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>f</code></td>
<td>File print facility</td>
</tr>
<tr>
<td><code>i</code></td>
<td>Convert from i-number to inode address; for continuation inodes as well as primary inodes</td>
</tr>
<tr>
<td><code>q</code></td>
<td>Quit</td>
</tr>
<tr>
<td><code>B</code></td>
<td>Byte mode</td>
</tr>
<tr>
<td><code>D</code></td>
<td>Double-word mode</td>
</tr>
<tr>
<td><code>O</code></td>
<td>Error checking flip-flop</td>
</tr>
<tr>
<td><code>W</code></td>
<td>Word mode</td>
</tr>
<tr>
<td><code>X</code></td>
<td>Hexadecimal flip-flop</td>
</tr>
</tbody>
</table>

Dots, tabs, and spaces can be used as function delimiters, but are not necessary. A line with just a newline character increments the current address by the size of the data type last printed. That is, the address is set to the next byte, word, double word, directory entry, or inode, allowing the user to step through a region of a file system.

Information is printed in a format appropriate to the data type. If the `X` toggle is off, bytes, words, and double words are printed in the form:

```
  octal-address: octal-value  (decimal-value)
```

If the `X` toggle is on, bytes, words, and double words are printed in the form:

```
  hex-address: hex-value
```

If the `B` (byte) or `D` (double-word) mode is in effect, the colon (:) shown above is preceded by `.B` or `.D`, respectively.

Directories are printed as a directory slot offset followed by the decimal i-number and the character representation of the entry name.

Inodes are printed with labeled fields describing each element.

Print Facilities

The print facilities generate a formatted output in various styles. Octal numbers are prefixed with a zero. Hexadecimal numbers are prefixed with `0x`. The current address is normalized to an appropriate boundary before printing begins. It advances with the printing and is left at the address of the last item printed. The output can be terminated at any time by typing the interrupt character. If a number follows the `p` symbol, that many entries are printed. A check is made to detect fragment boundary overflows since logically sequential blocks are generally not physically sequential. If a count of zero is used, all entries to the end of the current fragment are printed. The print options available are:

```
b  Print as octal bytes
  Print as characters
  Print as directories
  Print as decimal words
  Print as inodes (primary or continuation)
o  Print as octal words
  Print as hexadecimal words
```

The `f` symbol prints data fragments associated with the current inode. If followed by a number, that fragment of the file is printed. (Fragments are numbered from zero). The desired print option letter follows the fragment number, if present, or the `f` symbol. This print facility works for small as well as large files except for special files such as FIFOs, and device special files.

Inode and Directory Mnemonics

The following mnemonics are used for inode examination and refer to the current working inode:

```
anum  Data block numbers (num is in the range 0 – 14)
at    Time last accessed
cli   Continuation inode number
tct   Last time inode changed
gid   Group ID number
ln    Link count
maj   Major device number
```

The following mnemonics are used for directory examination:

<table>
<thead>
<tr>
<th>Mnemonic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>di</td>
<td>I-number of the associated directory entry</td>
</tr>
<tr>
<td>nm</td>
<td>Name of the associated directory entry</td>
</tr>
</tbody>
</table>

**EXAMPLES**

- **386i**: Print i-number 386 in an inode format. This now becomes the current working inode.
- **ln=4**: Change the link count for the working inode to 4.
- **ln+=1**: Increment the link count by 1.
- **fc**: Print in ASCII fragment zero of the file associated with the working inode.
- **2i.fd**: Print the first fragment-size piece of directory entries for the root inode of this file system.
- **d5i.fc**: Change the current inode to that associated with the fifth directory entry (numbered from zero) found from the above command. The first fragment's worth of bytes of the file are then printed in ASCII.
- **1b.px**: Print the first fragment of the superblock of this file system in hexadecimal.
- **2i.a0b.d7=3**: Change the i-number for the seventh directory slot in the root directory to 3. This example also shows how several operations can be combined on one command line.
- **d7.nm="newname"**: Change the name field in the directory slot to the given string. Quotes are optional if the first character of the name field is alphabetic.
- **a2b.p0d**: Print the third fragment of the current inode as directory entries.

**WARNINGS**

Only experienced users should use `fsdb`. The failure to fully understand the usage of `fsdb` and the file system's internal organization can lead to complete destruction of the file system and total loss of data.

**AUTHOR**

`fsdb` was developed by HP and AT&T.

**FILES**

`/etc/fstab` Static information about the file systems

**SEE ALSO**

dumpfs(1M), fsck(1M), fsdb(1M), stat(2), dir(4).

**STANDARDS CONFORMANCE**

`fsdb`: SVID3
NAME
fsdb_vxfs: fsdb - VxFS file system debugger

SYNOPSIS
fsdb [-F vxfs] [-z inumber] special

DESCRIPTION
fsdb can analyze VxFS file system problems or repair a damaged VxFS file system after a crash. A special
device special specifies the file system to debug. The fsdb command is intended for experienced users
only.

fsdb can convert block and inumbers into their corresponding disk addresses. In addition, mnemonic
offsets allow access to different parts of an inode. These greatly simplify the process of correcting control
block entries or descending the file system tree.

By default, numbers are considered decimal. You must prefix octal numbers with 0, and prefix hexade-
cimal numbers with 0x. When using hexadecimal numbers, it is preferable to follow the number with a
space, because several commands are letters that are also hexadecimal digits. In this document a pound
sign (#) indicates that a number is to be specified.

fsdb reads a block at a time and works with raw and block I/O. All I/O is unbuffered, so changes made to
the file system are immediate and changes made by other processes or by the kernel are immediately seen
by fsdb.

Options
fsdb recognizes the following options:

generic_options
   Supported by the generic fsdb command. See fsdb(1M).

-F vxfs
   Specify the VxFS file-system type.

-z inumber
   Clear the inode identified by inumber (non-interactive). Multiple -z options accumu-
late.

Operands
fsdb recognizes the following operand:

special
   Name of the special device that contains the VxFS file system to debug.

Using Commands and Symbols
It is best to separate each token on a command line with a space. Although the command parser does not
require element separation, there is no ambiguity in the command language if each token is separated with
a space. For example, the command 0x23b b sets the current position to block 0x23b hexadecimal. The
command 0x23bb is invalid, because the command is parsed as simply a hexadecimal number. The com-
mand 23b positions to block 23 decimal, since the command is not ambiguous.

You can separate commands with new lines. You can put multiple commands on one line and separate
them with a dot (.) or a semicolon (;) If multiple commands are placed on one line, generally only the last
command displays results. This allows positioning commands to be followed by printing commands or
change commands without intermediate printing.

Symbols for the fsdb Command
The following symbols are recognized by the fsdb command:

h|mod|print
   Print summary of commands that display [modify|format] the file system.
?
   Print summary of commands that display [modify|format] the file system.
help|mod|print
   Print summary of commands that display [modify|format] the file system.
!
   Escape to shell.
|  Pipe output of fsdb to a shell command.
q
   Quit.
string

A character string. Inside a character string, a NULL character can be specified with "\0"; a double quote can be specified with "\"; and a backslash can be specified with "\".

+ - * / % Add, subtract, multiply, divide, and modulus.

= Assignment

i An inode in the primary inode list.

ai An inode in the attribute inode list.

au An allocation unit.

b A block.

im The immediate data area of an inode. Small directories and symbolic link files (96 bytes or less) are stored directly in the inode itself, in the area normally occupied by data block numbers and extent sizes.

attr An attribute inode.

cdb Current directory block.

d A directory entry.

a An inode address entry.

B A byte.

H A half-word (2 bytes)

W A word (4 bytes)

D A double-word (8 bytes)

p General print facility

calc Simple calculator and base converter

find Find a matching pattern in the file system

fset A fileset.

iau An inode allocation unit in the primary inode list.

aiiu An inode allocation unit in the attribute inode list.

cut The current usage table.

olt The object location table.

mapi Map logical file offset to an inode extent.

reset Reset device.

Print Facility Print Formats

The print facility recognizes the following print formats:

S Print as a super-block.

A Print as an allocation-unit header.

AS Print as an auxiliary super-block.

L Print as intent-log records.

I Print as inodes.

ST List state table blocks.

T Print as typed extent descriptors.

dent Print as directory entries.

db Print as a directory block.

dh Print as a directory header.
Print as octal words.

Print as octal bytes, half-words, words, or double-words.

Print as hexadecimal words.

Print as hexadecimal bytes, half-words, words, or double-words.

Print as decimal words.

Print as decimal bytes, half-words, words, or double-words.

Print as hexadecimal bytes, half-words, words, or double-words.

Print as decimal bytes, half-words, words, or double-words.

Print as unsigned words.

Print as unsigned bytes, half-words, words, or double-words.

Print as characters.

Print as fileset headers.

Print as current usage table entries.

Print as an inode allocation unit header.

Print as an object location table extent.

Prints as a BSD quota record.

Print as a device record.

Symbols in Inode Fields
Changes to inode fields can be made symbolically. The following symbols represent inode fields:

Inode mode field
Inode link count field
Inode user ID Number field
Inode group ID Number field
Low-order word of inode file size field
High-order word of inode file size field
Inode file size field
Inode direct extent data block numbers (0 - 9)
Inode direct extent sizes (0 - 9)
Inode indirect extent data block numbers (0 - 1)
Inode indirect extent size
Inode access time field (seconds)
Inode access time field (microseconds).
Inode change time field (seconds).
Inode change time field (microseconds).
Inode modification time field (seconds).
Inode modification time field (microseconds).
Inode allocation flags field.
Inode generation count field.
Inode mapping type field.
Inode fixed extent size field.
Symbols for Directory Block Fields
Changes to directory block fields can be made symbolically. The following symbols represent directory
block fields:

- tfree: Total free space (only if in a data block).
- hash#: Hash chain start (0 through 31, only if in a data block).
- d#: Directory entry (variable number of entries).
- nhash: Number of hash chains.

Symbols for Directory Entry Fields
Changes to directory entry fields can be made symbolically. The following symbols represent directory
entry fields:

- ino: Inode number
- nm: Entry name
- nmlen: Name length
- reclen: Record length (only if in a data block)
- hnext: Name hash next (only if in a data block)

Positions in File Systems and Positioning Commands
fsdb maintains several positions in the file system:

- current position
- current primary-inode position (i)
- current attribute-inode position (ai)
- current inode type (i or ai)
- current fileset-header position (fset)
- current allocation-unit position (au)
- current primary-inode allocation-unit (iau) position
- current inode allocation-unit type (iau or aiau)
- current attribute-inode allocation-unit (aiu) position

These are used by various fsdb commands. (The au positions are only supported on the Version 2 disk
layout.)

The following commands are supported:
fsdb_vxfs(1M)

Set the current position in the file system to the specified offset in bytes, half-words, words, or double-words. If it is the last command on a line, print the byte, half-word, word, or double-words in hexadecimal.

Set the current position to the specified relative offset in bytes, half-words, words, or double-words. If it is the last command on a line, print the byte, half-word, word, or double-words in hexadecimal.

Set the current position in the file system to the specified allocation unit (au) position. Set the current allocation unit position to the resulting offset. If it is the last command on a line, print the allocation unit header.

Set the current position in the file system to the specified position relative to the current allocation unit (au) position. Set the current allocation unit position to the resulting offset. If it is the last command on a line, print the allocation unit header.

Set the current position in the file system to the specified allocation unit (au) position. If it is the last command on a line, print the allocation unit header.

Set the current position in the file system to the specified offset in blocks. Set the current block position to the resulting offset. The block size is the block size of the file system. If it is the last command on a line, print the first word in the block in hexadecimal.

Set the current position to specified relative offset in blocks. Set the current block position to the resulting offset. If it is the last command on a line, print the first word in the block in hexadecimal.

Set the current position to current block position (the block specified by the last [+|- # b] operation). If it is the last command on a line, print the first word in the block in hexadecimal.

Set the current position to the current usage table (cut). If it is the last command on a line, print the first current usage table entry.

Set the current position to the primary device's configuration record. If it is the last command on a line, print the device-configuration record.

Set the current position in the file system to the fileset header entry for the specified fileset index. Set the current fileset position to the resulting offset. If it is the last command on a line, print the specified fileset header.

Set the current position in the file system to the fileset header entry for the specified position relative to the current fileset position. Set the current fileset position to the resulting offset. If it is the last command on a line, print the specified fileset header.

Set the current position in the file system to the fileset header entry for the specified fileset index. Set the current fileset position to the resulting offset. If it is the last command on a line, print the fileset header for the current fileset.

Set the current position in the file system to the specified attribute inode allocation unit (aiau) in a fileset. Set the current attribute inode allocation unit position to the resulting offset. If it is the last command on a line, print the attribute inode allocation unit header.

Set the current position in the file system to the specified position relative to the current attribute inode allocation unit (aiau) position. Set the current attribute inode allocation unit position to the resulting offset. If it is the last command on a line, print the attribute inode allocation unit header.

Set the current position in the file system to the current attribute inode allocation unit (aiau) position. If it is the last command on a line, print the attribute inode allocation unit header.

Set the current position in the file system to the specified inode allocation unit (iau) in a fileset. Set the current inode allocation unit position to the resulting offset. If it is the last command on a line, print the inode allocation unit header.

Set the current position in the file system to the specified position relative to the current inode allocation unit (iau) position. Set the current inode allocation unit position to the resulting offset. If it is the last command on a line, print the inode allocation unit header.

HP-UX 11i Version 2: December 2007 Update

Hewlett-Packard Company
Set the current position in the file system to the current inode allocation unit (iau) position. If it is the last command on a line, print the inode allocation unit header.

Set the current position in the current fileset to the ilist entry for the specified attribute inode. Set the current attribute inode position to the resulting offset. If it is the last command on a line, print the ilist entry for the inode.

Set the current position in the current fileset to the ilist entry for the specified relative attribute inode. Set the current attribute inode position to the resulting offset. If it is the last command on a line, print the ilist entry for the inode.

Set the current position in the current fileset to the current attribute inode position. If it is the last command on a line, print the ilist entry for the inode.

Set the current position in the current fileset to the ilist entry for the specified inode. Set the current inode position to the resulting offset. If it is the last command on a line, print the ilist entry for the inode.

Set the current position to specified offset in blocks specified by the inode address #. Addresses 0 through 9 are for direct extents (de). Addresses 10-11 are for indirect extents (ie). The addresses are displayed when printing an ilist entry. Set the current block position to the resulting offset. If it is the last command on a line, print the first word in the block in hexadecimal.

Set the current position to immediate data area of the current inode. Set the current block position to the resulting offset. If it is the last command on a line, print the first word of the area in hexadecimal.

Set the current position to attribute data area of the current inode. Set the current block position to the resulting offset. If it is the last command on a line, print the first word in the block in hexadecimal.

Set the current position and change the number at the specified offset to the given number. If a double-word offset is specified, then two numbers separated by a space are required. The resulting value is printed in hexadecimal.

Set the current position and change the number at the specified relative offset to the given number. If a double-word offset is specified, then two numbers separated by a space are required. The resulting value is printed in hexadecimal.

Set the current position and change the characters at the specified offset to the given string. The resulting value is printed as a character string.

Set the current position and change the characters at the specified relative offset to the given string. The resulting value is printed as a character string.

Set the current position to the object location table (olt). If it is the last command on a line, print the object location table.

Print the contents of the file system at the current offset as the specified number of entries of a given format. The allowable print formats are specified above. If a number of entries to print is not specified, one entry is printed.

Set the contents of the given inode field to the specified number. The current inode specifies the inode list entry to be modified. The symbols representing inode fields are listed above.
directory_block_field = #
Set the contents of the given directory block field to the specified number. The
current block is treated as a directory block and the offset in that block which is
represented by the given field is changed. The symbols representing directory block
fields are listed above.

d#
Set the current directory entry to the specified number. The current block is treated
as a directory block. If the current block is an immediate data area for an inode, then
the block is treated as containing immediate directory entries. If it is the last com-
mand on a line, the directory entry at the resulting offset is printed.

directory_entry_field = #
Set the contents of the given directory entry field to the specified number. The current
directory entry specifies where the directory entry is located. The resulting value is
printed in hexadecimal.

nm = string
Set the directory name field of the current directory entry to the specified string. The
resulting value is printed as a character string.

calc # [+|-*]|/
Take a number or the sum, difference, product or dividend of two numbers and print
in decimal, octal, hexadecimal and character format.

find # B|H|W|D [#]
Search for the given numeric pattern in the file system. The size of the object to
match is specified. If a double-word is specified, then two numbers must be given.
The search is performed forward from the current offset. A maximum number of
blocks to search can be specified. If found, the location and value are printed in hexa-
decimal.

find string [#]
Search for the given character string in the file system. The search is performed for-
ward from the current offset. A maximum number of blocks to search can be speci-
fied. If found, the location and string are printed.

fmtlog
Format all intent log entries. A completely formatted intent log can be quite lengthy.
It is a good idea use the fsdb command as a filter and redirect the output to a file or
pager to look at a complete log format.

listfset
List all filesets by their indexes and names.

mapi #
Treat the number as a logical offset in the file described by the current inode, and
print the extent that it maps to.

reset
Does the equivalent of exiting fsdb and restarting on same device.

Help Commands
The following help commands are supported:

h | help
Display primary help screen.

h mod
Display modification-commands help screen.

h print
Display print-commands help screen.

EXAMPLES
386i
Prints inode number 386 in an inode format. This now becomes the current working inode.

ln=4
Changes the link count for the working inode to 4.

8192B.p S
Prints the super-block of this file system symbolically.

d7.nm = foo
Changes the name field in the directory slot to foo.

2i.a0.d7.ino = 3
Changes the inumber for the seventh directory slot in the root directory to three (also
shows how to combine several operations on one command line)

23i.im.d5
Prints the sixth directory entry in the immediate area of inode 23.
WARNINGS
Always run a file system check (see fsck(1M)) after using the fsdb command to modify a file system. (Use fsck -o full,nolog.)
Some aspects of fsdb apply to a specific VxFS disk layout version.

SEE ALSO
fsck_vxfs(1M), fsdb(1M).
NAME
fsirand - install random inode generation numbers

SYNOPSIS
/usr/sbin/fsirand [-p] special

DESCRIPTION
fsirand installs random inode generation numbers on all the inodes on device special, and also installs a filesystem ID in the superblock. This process increases the security of filesystems exported by NFS.

Use fsirand only on an unmounted filesystem that was checked with fsck (see fsck(1M)). The only exception is that it can be used on the root filesystem in single-user mode if the system is immediately rebooted afterwards using reboot -n.

The -p option prints the generation numbers for all inodes.

WARNINGS
fsirand is only supported on HFS filesystems. fsirand should not be run on mounted filesystems. If executing fsirand on the root filesystem, the system should be in single-user mode and should be rebooted immediately afterwards, using reboot -n.

AUTHOR
fsirand was developed by Sun Microsystems, Inc.

SEE ALSO
statfs(2).
NAME
fstyp - determine file system type

SYNOPSIS
/usr/sbin/fstyp [-v] special

DESCRIPTION
The _fstyp_ command allows the user to determine the file system type of a mounted or unmounted file
system. _special_ represents a device special file (for example: /dev/dsk/c1t6d0).

The file system type is determined by reading the superblock of the supplied _special_ file. If the superblock is
read successfully, the command prints the file system type identifier on the standard output and exits with
an exit status of 0. If the type of the file system cannot be identified, the error message
unknown_fstyp (no matches) is printed and the exit status is 1. Exit status 2 is not currently returned,
but is reserved for the situation where the file system matches more than one file system type. Any other
error will cause exit status 3 to be returned.

The file system type is determined by reading the superblock of the supplied _special_ file.

Options
-v Produce verbose output. The output contains information about the file system's superblock.

RETURN VALUE
 fstyp returns the following values:
0 Successful completion.
1 Unknown file system type.
2 File system matches more than one type.
3 Usage error or access problem.

EXAMPLES
Find the type of the file system on a disk, /dev/dsk/c1t6d0:
 fstyp /dev/dsk/c1t6d0

Find the type of the file system on a logical volume, /dev/vg00/lvol6:
 fstyp /dev/vg00/lvol6

Find the file system type for a particular device file and also information about its super block:
 fstyp -v /dev/dsk/c1t6d0

SEE ALSO
stat(2), statvfsdev(3C).
NAME
ftpd - DARPA Internet File Transfer Protocol server

SYNOPSIS

DESCRIPTION
ftpd is the DARPA Internet File Transfer Protocol server. It expects to be run by the Internet daemon
(see inetd(1M) and inetd.conf(4)). inetd runs ftpd when a service request is received at the port indicated in the ftp service specification in /etc/services (see services(4)).

Options
ftpd recognizes the following options and command-line arguments.
-a Enables the use of the configuration file /etc/ftpd/ftpaccess. (See ftpaccess(4)).
-A Disables the use of the configuration file /etc/ftpd/ftpaccess. (See ftpaccess(4)).
-B size Sets the buffer size of the data socket to size blocks of 1024 bytes. The valid range for size is from 1 to 2097151 (default is 56). Note: A large buffer size will improve the performance of ftpd on fast links (e.g. FDDI), but may cause long connection times on slow links (e.g. X.25).
Note: If the buffer size needs to be set to any value other than multiples of 1024 bytes, use ‘B’ immediately after size without any space. The size value will be taken in terms of bytes. For example, to set the buffer size to a value equal to “1500”, use -B 1500B.
-c ctrlport
-C dataport Overrides the control and the data port numbers respectively that is used by the dae-
mon. Normally, the daemon determines the port numbers by looking in /etc/services (see services(4)) for "ftp" and "ftp-data". If there is no /etc/services entry for "ftp-data" and the -C option is not specified, the daemon uses the port just prior to the control connection port. The -c and -C options are both available if running as a standalone daemon. Otherwise, only the -C option can be used.
-i Logs all the files received by ftpd server to xferlog (see xferlog(5)). This option is overridden by the /etc/ftpd/ftpaccess file. (See ftpaccess(4)).
-I Enables the use of RFC931 (AUTH/ident) to attempt to determine the username on the client.
-K Applicable only in a secure environment based on Kerberos V5. Causes access to be denied if network authentication fails. See sis(5).
-l Causes each FTP session to be logged in the syslog file.
-L Logs all commands sent to the ftpd server to be logged to the syslog. The -L option is overridden by the /etc/ftpd/ftpaccess file (see ftpaccess(4)). If the -L option is used, commands will be logged to syslog by default.
-m number_of_tries Specifies the number of tries for a bind() socket call.
-n nice_value Sets the nice value for an ftpd process. When using this option, make sure that the nice clause in /etc/ftpd/ftpaccess file (see ftpaccess(4)) is not set.
-o Logs all files transmitted by ftpd to xferlog (see xferlog(5)). It logs outgoing files from the ftpd server. This option is overridden by the /etc/ftpd/ftpaccess file (see ftpaccess(4)).
-p The default action of ftpd does not allow usage of reserved ports as the originating port on the client’s system i.e., the PORT command cannot specify a reserved port. This option allows the client to specify a reserved port. Note, allowing usage of reserved ports can result in the misuse of ftpd. The security ramifications should be
understood before the option is turned on.

-\texttt{P} Enables third party transfer.

-\texttt{q} | -\texttt{Q} Determines whether the daemon uses the PID files. These files are required by the limit directive to determine the number of current users in each access class. Disabling the use of the PID files disables user limits. The default (-\texttt{q}) specifies to use the PID files. Specify -\texttt{Q} to disable using the PID files. The -\texttt{Q} option can be used when testing the server as a normal user when access permissions prevent the use of the PID files. Large, busy sites which do not wish to impose limits on the number of concurrent users may also consider disabling the PID files.

-\texttt{r rootdir} Instructs the daemon to chroot (see chroot(2)) to the specified rootdir immediately upon loading. This can improve system security by limiting the files which may be damaged should a break-in occur through the daemon. This option is like anonymous FTP. For this option to work properly additional files may be needed under the specified rootdir, which can vary from system to system.

-\texttt{s} | -\texttt{S} Runs the daemon in standalone operation mode. The -\texttt{S} option runs the daemon in the background and is useful in startup scripts during system initialization (ie., in rc.local). The -\texttt{s} option leaves the daemon in foreground and is useful when running from init (see init(1M)).

-\texttt{t timeout} Causes ftpd to timeout inactive sessions after timeout seconds. By default, ftpd terminates an inactive session after 15 minutes.

-\texttt{T maxtimeout} A client can also request a different timeout period. The -\texttt{T} option sets to maxtimeout the maximum timeout that client can request, in seconds. By default, the maximum timeout is 2 hours.

-\texttt{u umask} Change default ftpd umask from 027 to umask.

-\texttt{U} Disables the use of sendfile() and uses send() for sending data. Use this option if the link cannot handle more than one buffer per packet (e.g. Gigabit Ethernet).

-\texttt{v} The debugging information is written to the syslog file.

-\texttt{V} Causes the program to display copyright and version information, then terminate.

-\texttt{w} | -\texttt{W} Determines whether the user logins are to be recorded in the wtmps and btmps files. If the -\texttt{W} option is specified, user logins are not recorded in the wtmps or btmps file. The default (-\texttt{w}) is used to record every login, logout, and bad login attempts.

-\texttt{X} Specifies that the output created by the -\texttt{i} and -\texttt{o} options is not saved to the xferlog file but saved via syslog so that the output can be collected from several hosts on one central loghost.

\texttt{ftpd} currently supports the following commands (uppercase and lowercase are interpreted as equivalent):

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABOR</td>
<td>Abort previous command</td>
</tr>
<tr>
<td>ACCT</td>
<td>Specify account (ignored)</td>
</tr>
<tr>
<td>ALLO</td>
<td>Allocate storage (vacuously)</td>
</tr>
<tr>
<td>APPE</td>
<td>Append to a file</td>
</tr>
<tr>
<td>CDUP</td>
<td>Change to parent of current working directory</td>
</tr>
<tr>
<td>CWD</td>
<td>Change working directory</td>
</tr>
<tr>
<td>DELE</td>
<td>Delete a file</td>
</tr>
<tr>
<td>EPSV</td>
<td>Sets the server to listen on a data port and wait for a connection</td>
</tr>
<tr>
<td>EPRT</td>
<td>Use extended address for data connection</td>
</tr>
<tr>
<td>HELP</td>
<td>Give help information</td>
</tr>
<tr>
<td>LIST</td>
<td>Give list files in a directory (\texttt{ls -l})</td>
</tr>
<tr>
<td>LPRT</td>
<td>Use long address for data connection</td>
</tr>
<tr>
<td>LPSV</td>
<td>Sets the server to listen on a data port and wait for a connection</td>
</tr>
<tr>
<td>MKD</td>
<td>Make a directory</td>
</tr>
<tr>
<td>MDTM</td>
<td>Show last modification time of file</td>
</tr>
<tr>
<td>MODE</td>
<td>Specify data transfer mode</td>
</tr>
</tbody>
</table>
ftp (1M)

NLST      Give name list of files in directory
NOOP      Do nothing
PASS      Specify password
PASV      Prepare for server-to-server transfer
PORT      Specify data connection port
PWD       Print the current working directory
QUIT      Terminate session
REST      Restart incomplete transfer
RETR      Retrieve a file
RMD       Remove a directory
RNFR      Specify rename-from file name
RNTO      Specify rename-to file name
SITE      Non-standard commands (see next section)
SIZE      Return size of file
STAT      Return status of server
STOR      Store a file
STOU      Store a file with a unique name
STRU      Specify data transfer structure
SYST      Show operating system type of server system
TYPE      Specify data transfer type
USER      Specify user name
XCUP      Change to parent of current working directory
XCWD      Change working directory
XMKD      Make a directory
XPWD      Print the current working directory
XRMD      Remove a directory

The following commands are supported when ftpd is operating in a secure environment which is based on Kerberos V5 (see sis(5)).

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTH</td>
<td>Authentication/security mechanism</td>
</tr>
<tr>
<td>ADAT</td>
<td>Authentication/security data</td>
</tr>
<tr>
<td>CCC</td>
<td>Clear command channel</td>
</tr>
<tr>
<td>ENC</td>
<td>Privacy protected command</td>
</tr>
<tr>
<td>MIC</td>
<td>Integrity protected command</td>
</tr>
<tr>
<td>PROT</td>
<td>Data channel protection level (level 'C' only)</td>
</tr>
<tr>
<td>PBSZ</td>
<td>Protection buffer size (has no effect)</td>
</tr>
</tbody>
</table>

These commands are described in draft 8 of the FTP security extensions.

The following non-standard or HP-UX specific commands are supported by the SITE command:

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UMASK</td>
<td>Change umask. (e.g., SITE UMASK 002)</td>
</tr>
<tr>
<td>IDLE</td>
<td>Set idle-timer. (e.g., SITE IDLE 60)</td>
</tr>
<tr>
<td>CHMOD</td>
<td>Change mode of a file. (e.g., SITE CHMOD 755 filename)</td>
</tr>
<tr>
<td>HELP</td>
<td>Give help information. (e.g., SITE HELP)</td>
</tr>
<tr>
<td>NEWER</td>
<td>List files newer than a particular date.</td>
</tr>
<tr>
<td>MINFO</td>
<td>Works like SITE NEWER, but gives extra information.</td>
</tr>
<tr>
<td>GROUP</td>
<td>Request for special group access. (e.g., SITE GROUP foo)</td>
</tr>
<tr>
<td>GPASS</td>
<td>Give special group access password. (e.g., SITE GPASS bar)</td>
</tr>
<tr>
<td>EXEC</td>
<td>Execute a program. (e.g., SITE EXEC program params)</td>
</tr>
</tbody>
</table>

For the SITE EXEC command, in order to execute a program it has to be placed in the /etc/ftpd/ftp-exec directory. The program to be executed must be either a binary program file or a valid shell. For example for the following program:

```
cat /etc/ftpd/ftp-exec/hi.sh
#!/usr/bin/sh
echo hello
```

When we give the following SITE EXEC command:

```
ftp> site exec hi.sh
```

The output will be as follows:

```
200-hi.sh
200-hello
200 (end of 'hi.sh')
```

**Note:** The security of the system will entirely be dependent on what binaries or shell programs that the administrator has placed in the directory `/etc/ftpd/ftp-exec`. Making this functionality available to real users who have shell access does not have any major security ramifications, but for anonymous and guest users who do not have shell access, it does.

The remaining FTP requests specified in Internet RFC 959 are recognized, but not implemented. `MDTM` and `SIZE` are not specified in RFC 959, but are expected in the next updated FTP RFC.

The FTP server aborts an active file transfer only when the `ABOR` command is preceded by a Telnet "Inter-rupt Process" (IP) signal and a Telnet "Synch" signal in the command Telnet stream, as described in Internet RFC 959. If `ftpd` receives a `STAT` command during a data transfer, preceded by a Telnet IP and Synch, it returns the status of the transfer.

`ftpd` interprets file names according to the "globbing" conventions used by `csh(1)`. This allows users to utilize the metacharacters `*`, `..`, `[`, `]`, `~`, and `?`.

`ftpd` authenticates users according to three rules:

- The user name must be in the password data base, `/etc/passwd`, and not have a null password. The client must provide the correct password for the user before any file operations can be performed.
- The user name must not appear in the file `/etc/ftpd/ftpusers` (see `ftpusers(4)`).
- The user must have a standard shell returned by `getusershell()`.

Optionally, a system administrator can permit public access or "anonymous FTP." If this has been set up, users can access the anonymous FTP account with the user name `anonymous` or `ftp` and any non-null password (by convention, the client host's name). `ftpd` does a `chroot()` to the home directory of user `ftp`, thus limiting anonymous FTP users' access to the system. If the user name is `anonymous` or `ftp`, an anonymous FTP account must be present in the password file (user `ftp`). In this case the user is allowed to log in by specifying any password (by convention this is given as the user's e-mail address).

In order to permit anonymous FTP, there must be an entry in the `passwd(4)` database for an account named `ftp`. The password field should be *, the group membership should be `guest`, and the login shell should be `/usr/bin/false`. For example (assuming the `guest` group ID is 10):

```
ftp:*:500:10:anonymous ftp:/home/ftp:/usr/bin/false
```

The anonymous FTP directory should be set up as follows:

```
~ftp
```

The home directory of the FTP account should be owned by user `root` and mode 555 (not writable). Since `ftpd` does a `chroot()` to this directory, it must have the following subdirectories and files:

```
~ftp/usr/bin
```

This directory must be owned by root and mode 555 (not writable). The file `/sbin/ls` should be copied to `~ftp/usr/bin`. This is needed to support directory listing by `ftpd`. The command should be mode 111 (executable only). If the FTP account is on the same file system as `/sbin`, `~ftp/usr/bin/ls` can be hard link, but it may not be a symbolic link, because of the `chroot()` function. The command must be replaced when the system is updated.

**Note:** The file `/usr/bin/ls` can also be copied to the directory `~ftp/usr/bin` in place of `/sbin/ls`. However, if this is done, a set of relevant libraries must also be copied under the directory `~ftp/usr/lib`. See the HP-UX Remote Access Services Administrator's Guide for details of required libraries. The directory `~ftp/usr/lib` must be owned by root and mode 555 (not writable). All the libraries copied under this directory must be mode 555 (not writable).

```
~ftp/etc
```

This directory must be owned by root and mode 555 (not writable). It should contain versions of the files `passwd` and `group`. See `passwd(4)` and `group(4)`. These files must be owned by root and mode 444 (readable only). These files must be present for the
**LIST** command to be able to produce owner names rather than numbers.

- **ftp/etc/passwd**
  This file should contain entries for the *ftp* user and any other users who own files under the anonymous *ftp* directory. Such entries should have * for passwords. Group IDs must be listed in the anonymous FTP group file, *ftp/etc/group*. The path names of home directories in *ftp/etc/passwd* must be with respect to the anonymous FTP home directory.

- **ftp/etc/group**
  This file should contain the group names associated with any group IDs in file *ftp/etc/passwd* and any group IDs of files in the anonymous FTP subdirectories.

- **ftp/pub** (optional)
  This directory is used by anonymous FTP users to deposit files on the system. It should be owned by user *ftp* and should be mode 777 (readable and writable by all).

- **ftp/dist** (optional)
  Directories used to make files available to anonymous ftp users should be mode 555 (not writable), and any files to be distributed should be owned by root and mode 444 (readable only) so that they cannot be modified or removed by anonymous FTP users.

Note: The steps that are followed to create an anonymous account are used to create a guest account also.

**DIAGNOSTICS**

`ftpd` replies to FTP commands to ensure synchronization of requests and actions during file transfers, and to indicate the status of `ftpd`. Every command produces at least one reply, although there may be more than one. A reply consists of a three-digit number, a space, some text, and an end of line. The number is useful for programs; the text is useful for users. The number must conform to this standard, but the text can vary.

The first digit of the message indicates whether the reply is good, bad, or incomplete. Five values exist for the first digit. The values and the interpretations of the values are:

1. The requested action is being initiated; expect another reply before proceeding with a new command.
2. The requested action is complete. The server is ready for a new request.
3. The command has been accepted, but the requested action requires more information.
4. The command was not accepted, the requested action failed, but the error condition is temporary and the action can be requested again.
5. The command was not accepted, the requested action failed, and the error condition would most likely occur again if the same command sequence is repeated.

The second digit indicates the functional area that the message addresses. The values of the second digit and the interpretations of these values are:

0. Syntax. A message with a 0 for the second digit indicates that a syntax error occurred.
1. Information. A message with a 1 as the second digit indicates that the message is in reply to a request for information.
2. Connections. A message with a 2 as the second digit indicates that the message is a reply to a request for control and data connection information.
3. Authentication and accounting. A message with a 3 as the second digit indicates that the message is a reply to a login or accounting procedure.
4. Not currently specified.
5. File system. A message with a 5 as the second digit indicates that the text following the number contains information concerning the status of the server file system.

The third digit provides a further clarification of the information supplied by the second digit. Following are several examples of messages. Note that `ftpd`'s replies match the number but not the text.

110. Restart marker reply. `MARK yyyy=mmmm` where `yyyy` is a user process data stream marker, and `mmmm` is `ftpd`'s equivalent marker.
GENERAL FTP EXTENSIONS

There are some extensions to the FTP server such that if the user specifies a filename (when using a RETRIEVE command), the following actions will occur:

<table>
<thead>
<tr>
<th>True Filename</th>
<th>Specified Filename</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>filename.Z</td>
<td>filename</td>
<td>Decompress (uncompress) file before transmitting</td>
</tr>
<tr>
<td>filename</td>
<td>filename.tar.Z</td>
<td>Tar and compress filename before transmitting</td>
</tr>
<tr>
<td>filename.tar</td>
<td>filename.tar.Z</td>
<td>Tar and compress filename before transmitting</td>
</tr>
<tr>
<td>filename.tar</td>
<td>filename.tar.Z</td>
<td>Tar and compress filename before transmitting</td>
</tr>
</tbody>
</table>

Also, the FTP server will attempt to check for valid e-mail addresses and notify the user if invalid e-mail addresses are found. For users whose FTP client will hang on “long replies” (i.e. multiline responses), using a dash as the first character of the password will disable this "long replies" feature.

**Note:** Users whose password starts with a dash, have to use an extra dash in the beginning of the password for login to succeed. However, the "long replies" feature will be disabled in this case.

The FTP server can also log all file transmission and reception, keeping the following information for each file transmission that takes place.

```
%.24s %d %s %d %s %c %s %c %s %s %d %s
```

1. current time in the form DDD MMM dd hh:mm:ss YYYY
2. transfer time in seconds
3. remote host name
4. file size in bytes
5. name of file
6. transfer type (a>scii, b>inary)
7. special action flags (concatenated as needed):
   - C file was compressed
   - U file was uncompressed
   - T file was tar'ed
     - no action taken
8. file was sent to user (o>utgoing) or received from user (i>ncoming)
9. accessed anonymously (r>eal, a>nonymous, g>uest)
10. local username or, if guest, ID string given (anonymous FTP password)
11. service name (‘ftp’, other)
12. authentication method (bitmask)
   - 0 none
   - 1 RFC931 Authentication
13. authenticated user id (if available, '*' otherwise)

**WARNINGS**

The password is sent unencrypted through the socket connection.
Anonymous FTP is inherently dangerous to system security.

DEPENDENCIES
Pluggable Authentication Modules (PAM)
PAM is an Open Group standard for user authentication, password modification, and validation of accounts. In particular, `pam_authenticate()` is invoked to perform all functions related to login. This includes retrieving the password, validating the account, and displaying error messages.

AUTHOR
`ftpd` was developed by the University of California, Berkeley and the Washington University, St. Louis, Missouri.

SEE ALSO
`ftp(1), inetd(1M), chroot(2), send(2), sendfile(2), getusershell(3C), pam_authenticate(3), ftpaccess(4), ftpusers(4), group(4), inetd.conf(4), passwd(4), sis(5), xferlog(5)`
NAME
fuser - list processes using a file or file structure

SYNOPSIS
/usr/sbin/fuser [-c | -f] [-ku] file ... [-c | -f] [-ku] file ...

DESCRIPTION
The fuser command lists the process IDs of processes that have each specified file open. For block special devices, all processes using any file on that device are listed. The process ID may be followed by a letter, identifying how the file is being used, as follows:
- c file is current directory of the process.
- r file is the root directory of the process, as set up by the chroot command (see chroot(1M)).
- o The process has file open.
- m The process has file memory mapped.
- t file is the text file of the process.

The process IDs associated with each file are printed to standard output as a single line separated by spaces and terminated with a single newline. All other output — the file name, the letter, and the user name — is written to standard error.

Options
fuser has the following options:
- c Display the use of a mount point and any file beneath that mount point. Each file must be a file system mount point.
- f Display the use of the named file only, not the files beneath it if it is a mounted file system. This is the default.
- u Display the login user name in parentheses following each process ID.
- k Send the SIGKILL signal to each process using each file. You must have appropriate privileges to kill processes that you do not own.

You can respecify options between groups of files. The new set of options replaces the old set. A dash (-) by itself cancels all options currently in force.

Operands
fuser has the following operand:
- file One of the following values:
  - With the -f option, the name of a file.
  - With the -f option, the name of a mounted file system or special file.
  - With the -c option, the name of a file system mount point.

NETWORKING FEATURES
You can use fuser with NFS file systems or files. If the file name is in the format used in /etc/mnttab to identify an NFS file system, fuser treats the NFS file system as a block special device and identifies any process using that file system.

If contact with an NFS file system is lost, fuser fails, since contact is required to obtain the file system identification. Once the NFS file system is recontacted, stale file handles from the previous contact can be identified, provided that the NFS file system has the same file system identification.

EXAMPLES
Terminate all processes that are preventing disk drive 1 from being unmounted, listing the process ID and login name of each process being killed.

    fuser -ku /dev/dsk/c201d1s?
List process IDs and login names of processes that have the password file open.

```
fuser -u /etc/passwd
```

Combine both the above examples into a single command line.

```
fuser -ku /dev/dsk/c201d1s? - -u /etc/passwd
```

If the device `/dev/dsk/c201d1s7` is mounted on directory `/home`, list the process IDs and login names of processes using the device. Alternately, if `/home` is the mount point for an NFS file system, list process IDs and login names of processes using that NFS file system.

```
fuser -cu /home
```

If `machine1:/filesystem/2mount` is an NFS file system, list all processes using any file on that file system. If it is not an NFS file system, treat it as a regular file.

```
fuser machine1:/filesystem/2mount
```

SEE ALSO

`ps(1), mount(1M), kill(2), signal(2)`.

STANDARDS CONFORMANCE

`fuser`: SVID2, SVID3
fwtmp(1M)

NAME
fwtmp, wtmpfix - manipulate connect accounting records

SYNOPSIS
/usr/sbin/acct/fwtmp [-icX]
/usr/sbin/acct/wtmpfix [files]

DESCRIPTION
fwtmp
fwtmp reads from the standard input and writes to the standard output, converting binary records of the type found in wtmps to formatted ASCII records. The ASCII version is useful to enable editing, via ed(1), bad records or for general purpose maintenance of the file.

The argument -ic is used to denote that input is in ASCII form, and output is to be written in binary form. The arguments i and c are independent, respectively specifying ASCII input and binary output. Therefore, -i is an ASCII to ASCII copy and -c is a binary to binary copy. -X should be used for reading wtmps-like records. If -X is not used, utmp-like structure is read.

wtmpfix
wtmpfix examines the standard input or named files in wtmps format, corrects the time/date stamps to make the entries consistent, and writes to the standard output. A - can be used in place of files to indicate the standard input. If time/date corrections are not performed, acctcon1 will fault when it encounters certain date-change records.

Each time the date is set, a pair of date change records is written to WTMPS_FILE. The first record is the old date denoted by the string old_time placed in the line field and the flag OLD_TIME placed in the type field of the utmps structure. The second record specifies the new date, and is denoted by the string new_time placed in the line field and the flag NEW_TIME placed in the type field. wtmpfix uses these records to synchronize all time stamps in the file. wtmpfix nullifies date change records when writing to the standard output by setting the time field of the utmps structure in the old date change record equal to the time field in the new date change record. This prevents wtmpfix and acctcon1 from factoring in a date change record pair more than once.

In addition to correcting time/date stamps, wtmpfix checks the validity of the name field to ensure that it consists solely of alphanumeric characters or spaces. If it encounters a name that is considered invalid, it changes the login name to INVALID and writes a diagnostic to the standard error. This minimizes the risk that acctcon1 will fail when processing connect accounting records.

DIAGNOSTICS
wtmpfix generates the following diagnostic messages:

Cannot make temporary: xxx failed to make temp file
Input truncated at offset: xxx missing half of date pair
New date expected at offset: xxx missing half of date pair
Cannot read from temp: xxx some error reading
Bad file at offset: xxx ut_line entry not digit, alpha, nor | or { (First character only checked.)
Out of core: malloc fails. ( Saves table of date changes.)
No dtab: software error (Rarely seen, if ever.)

WARNINGS
fwtmp generates no errors, even on garbage input.

FILES
/usr/include/utmp.h
/var/adm/wtmp
/var/adm/wtmps

SEE ALSO
ed(1), acct(1M), acctcms(1M), acctcom(1M), acctcon(1M), acctmerg(1M), acctprc(1M), acctsh(1M), runacct(1M), acct(2), acct(4), utmp(4), wtmps(4).
fwtmp(1M)

STANDARDS CONFORMANCE
fwtmp: SVID2, SVID3
wtmpfix: SVID2, SVID3
NAME
gated - gateway routing daemon

SYNOPSIS
gated [-b buffer_size] [-c] [-C] [-n] [-N] [-r] [-t trace_options] [-f config_file] [trace_file]

DESCRIPTION
gated is a routing daemon that handles multiple routing protocols and replaces routed, egup, and any routing daemon that speaks the HELLO routing protocol. gated currently handles the RIP, BGP, EGP, HELLO, and OSPF routing protocols. The gated process can be configured to perform all routing protocols or any subset of them (see WARNINGS below).

Options
The command-line options are:

-b buffer_size  Specifies a buffer size for the socket read/write buffer. The buffer size should not be less than one MB (megabyte) and should not exceed the available system memory. If this option is not specified, the buffer size will be set to the available system memory.

-c  Specifies that the configuration file will be parsed for syntax errors and then gated will exit. gated will leave a dump file in /var/tmp/gated_dump if there were no errors. gated does not need to be run as the superuser to use the -c option but it may not be possible to read the kernel forwarding table and interface configuration if not run as superuser. The -c option implies -tgeneral. All trace_option clauses in the configuration file will be ignored.

-C  Specifies that the configuration file will just be parsed for syntax errors. gated will exit with a status 1 if there were any errors and 0 (zero) if there were not. gated does not need to be run as the superuser to use the -C option but it may not be possible to read the kernel forwarding table and interface configuration if not run as the superuser.

-n  Specifies that gated will not modify the kernel forwarding table. This is used for testing gated configurations with actual routing data.

-N  Specifies that gated will not daemonize. Normally, if tracing to stderr is not specified gated will daemonize if the parent process ID is not 1. This allows the use of an /etc/inittab-like method of invoking gated that does not have a PID of 1.

-r  Specifies that gated will add routes to the kernel in the decreasing order of IP addresses of gateways when there are multiple routes to a destination.

-t trace_options  Specifies a comma separated list of trace options to be enabled on startup. If no flags are specified, general is assumed. No space is allowed between this option and it's arguments.

Signal Processing
The following signals may be used to control gated:

SIGHUP  Re-read configuration. A SIGHUP causes gated to reread the configuration file. gated first performs a clean-up of all allocated policy structures. All BGP and EGP peers are flagged for deletion and the configuration file is re-parsed.
If the re-parse is successful, any BGP and EGP peers that are no longer in the configuration are shut down, and new peers are started. `gated` attempts to determine if changes to existing peers require a shutdown and restart. OSPF is not capable of reconfiguring; it is shutdown and restarted during a reconfiguration. This may have an adverse impact on the routing system.

It should also be possible to enable/disable any protocol without restarting `gated`.

**SIGINT**

Snap-shot of current state.

The current state of all `gated` tasks, timers, protocols and tables are written to `/var/tmp/gated_dump`

On systems supporting `fork()`, this is done by forking a subprocess to dump the table information so as not to impact `gated`’s routing functions. On systems where memory management does not support copy-on-write, this will cause the `gated` address space to be duplicated; this may cause a noticeable impact on the system. On system not supporting `fork()`, the main process immediately processes the dump, which may impact `gated`’s routing functions.

**SIGTERM**

Graceful shutdown.

On receipt of a `SIGTERM`, `gated` attempts a graceful shutdown. All tasks and protocols are asked to shutdown. Most will terminate immediately, the exception being EGP peers which wait for confirmation. It may be necessary to repeat the `SIGTERM` once or twice if it this process takes too long.

All protocol routes are removed from the kernel’s routing table on receipt of a `SIGTERM`. Interface routes, routes with RTF_STATIC set (from the route command where supported) and static routes specifying `retain` will remain. To terminate `gated` with the exterior routes intact, use `SIGKILL`.

**SIGUSR1**

Toggle tracing.

On receipt of a `SIGUSR1`, `gated` will close the trace file. A subsequent `SIGUSR1` will cause it to be reopened. This will allow the file to be moved regularly.

It is not possible to use `SIGUSR1` if a trace file has not been specified, or tracing is being performed to stderr.

**SIGUSR2**

Check for interface changes.

On receipt of a `SIGUSR2`, `gated` will rescan the kernel interface list looking for changes.

**WARNINGS**

`gated` contains provisions for BGP protocol, but it is not officially supported by HP at the present time. Some RIP version 2 features (RFC1388) are not currently supported: MIB and route tag. The optional OSPF version 2 (RFC1247) feature of TOS (type of service) based routing is not supported. The route aggregation, generating a more general route from compressing the specific routes through the explicit configuration, is not supported in this release.

**AUTHOR**

`gated` was primarily developed by Cornell University which includes code from the Regents of the University of California and the University of Maryland.

This software and associated documentation is Copyright 1990, 1991, 1992 by Cornell University.

**SEE ALSO**

`netstat(1)`, `arp(1M)`, `gdc(1M)`, `ifconfig(1M)`, `ospf_monitor(1M)`, `ripquery(1M)`, `fork(2)`, `gated.conf(4)`, `GateD Documentation`, `GateD Configuration Guide`.

RFC 891 DCN Local-Network Protocols (HELLO)
RFC 904 Exterior Gateway Protocol Formal Specification
RFC 1058 Routing Information Protocol
RFC 1163 A Border Gateway Protocol (BGP)
RFC 1164 Application of the Border Gateway Protocol in the Internet
NAME
gdc - operational user interface for gated

SYNOPSIS

DESCRIPTION

gdc provides a user-oriented interface for the operation of the gated(1M) routing daemon. It provides support for starting and stopping the daemon, for the delivery of signals to manipulate the daemon when it is operating, for the maintenance and syntax checking of configuration files, and for the production and removal of state dumps and core dumps.

gdc can reliably determine gated's running state and produces a reliable exit status when errors occur, making it advantageous for use in shell scripts which manipulate gated. Commands executed using gdc and, optionally, error messages produced by the execution of those commands, are logged via the same syslogd(1M) facility which gated itself uses, providing an audit trail of operations performed on the daemon.

If installed as a setuid root program gdc will allow non-root users who are members of a trusted group (by default the gdmaint group) to manipulate the routing daemon while denying access to others. The name of the user is logged along via syslogd(1M) along with an indication of each command executed, for audit purposes.

The command-line options are:

- Run without changing the kernel forwarding table. Useful for testing, and when operating as a route server which does no forwarding.
-q Run quietly. With this option informational messages which are normally printed to the standard output are suppressed and error messages are logged via syslogd(1M) instead of being printed to the standard error output. This is often convenient when running gdc from a shell script.
-t seconds Specifies the time in seconds which gdc will spend waiting for gated to complete certain operations, in particular at termination and startup. By default this value is set to 10 seconds.

These additional command-line options may be present, depending on the options used to compile gdc:

- Sets the maximum size of a core dump a gated started with gdc will produce. Useful on systems where the default maximum core dump size is too small for gated to produce a full core dump on errors.
- Sets the maximum file size a gated started with gdc will produce. Useful on systems where the default maximum file dump size is too small for gated to produce a full state dump when requested.
- Sets the maximum size of the data segment of a gated started with gdc. Useful on systems where the default data segment size is too small for gated to run.
- Sets the maximum size of stack of a gated started with gdc. Useful on systems where the default maximum stack size is too small for gated to run.

The following commands cause signals to be delivered to gated for various purpose:

COREDUMP Sends an abort signal to gated, causing it to terminate with a core dump.
dump Signal gated to dump its current state into the file /usr/tmp/gated_dump.
interface Signal gated to recheck the interface configuration. gated normally does this periodically in any event, but the facility can be used to force the daemon to check interface status immediately when changes are known to have occurred.
KILL Cause gated to terminate ungracefully. Normally useful when the daemon has hung.
reconfig Signal gated to reread its configuration file, reconfiguring its current state as appropriate.
term Signal gated to terminate after shutting down all operating routing protocols gracefully. Executing this command a second time should cause gated to terminate even if some protocols have not yet fully shut down.
gdc(1M)

toggletrace

If `gated` is currently tracing to a file, cause tracing to be suspended and the trace file to be closed. If `gated` tracing is currently suspended, cause the trace file to be reopened and tracing initiated. This is useful for moving trace files.

By default `gated` obtains its configuration from a file normally named `/etc/gated.config`. The `gdc` program also maintains several other versions of the configuration file, in particular named:

- `/etc/gated.conf+` The *new* configuration file. When `gdc` is requested to install a new configuration file, this file is renamed `/etc/gated.conf`.
- `/etc/gated.conf-` The *old* configuration file. When `gdc` is requested to install a new configuration file, the previous `/etc/gated.conf` is renamed to this name.
- `/etc/gated.conf--` The *really old* configuration file. `gdc` retains the previous *old* configuration file under this name.

The following commands perform operations related to configuration files:

- **checkconf** Check `/etc/gated.conf` for syntax errors. This is usefully done after changes to the configuration file but before sending a `reconfig` signal to the currently running `gated`, to ensure that there are no errors in the configuration which would cause the running `gated` to terminate on reconfiguration. When this command is used, `gdc` issues an informational message indicating whether there were parse errors or not, and if so saves the error output in a file for inspection.
- **checknew** Like `checkconf` except that the *new* configuration file, `/etc/gated.conf+`, is checked instead.
- **newconf** Move the `/etc/gated.conf+` file into place as `/etc/gated.conf`, retaining the older versions of the file as described above. `gdc` will decline to do anything when given this command if the *new* configuration file doesn't exist or otherwise looks suspect.
- **backout** Rotate the configuration files in the *newer* direction, in effect moving the *old* configuration file to `/etc/gated.conf`. The command will decline to perform the operation if `/etc/gated.conf-` doesn't exist or is zero length, or if the operation would delete an existing, non-zero length `/etc/gated.conf+` file.
- **BACKOUT** Perform a `backout` operation even if `/etc/gated.conf+` exists and is of non-zero length.
- **modeconf** Set all configuration files to mode 664, owner root, group gdmaint. This allows a trusted non-root user to modify the configuration files.
- **createconf** If `/etc/gated.conf+` does not exist, create a zero length file with the file mode set to 664, owner root, group gdmaint. This allows a trusted non-root user to install a new configuration file.

The following commands provide support for starting and stopping `gated`, and for determining its running state:

- **running** Determine if `gated` is currently running. This is done by checking to see if `gated` has a lock on the file containing its pid, if the pid in the file is sensible and if there is a running process with that pid. Exits with zero status if `gated` is running, non-zero otherwise.
- **start** Start `gated`. The command returns an error if `gated` is already running. Otherwise it executes the `gated` binary and waits for up to the delay interval (10 seconds by default, as set with the `-t` option otherwise) until the newly started process obtains a lock on the pid file. A non-zero exit status is returned if an error is detected while executing the binary, or if a lock is not obtained on the pid file within the specified wait time.
- **stop** Stop `gated`, gracefully if possible, ungracefully if not. The command returns an error (with non-zero exit status) if `gated` is not currently running. Otherwise it sends a terminate signal to `gated` and waits for up to the delay interval (10 seconds by default, as specified with the `-t` option otherwise) for the process to exit. Should `gated` fail to exit within the delay interval it is then signaled again with a second terminate signal. Should it fail to exit by the end of the second delay interval it is signaled for a third time with a kill signal. This should force immediate termination unless something is very broken. The command terminates with zero exit status when it detects that `gated` has terminated, non-zero otherwise.
restart

If `gated` is running it is terminated via the same procedure as is used for the `stop` command above. When the previous `gated` terminates, or if it was not running prior to command execution, a new `gated` process is executed using the procedures described for the `start` command above. A non-zero exit status is returned if any step in this procedure appears to have failed.

The following commands allow the removal of files created by the execution of some of the commands above:

- `rmcore` Removes any existing `gated` core dump file.
- `rmdump` Removes any existing `gated` state dump file.
- `rmparse` Removes the parse error file generated when a `checkconf` or `checknew` command is executed and syntax errors are encountered in the configuration file being checked.

**AUTHOR**

gdc was developed by Dennis Ferguson and Cornell University.

**FILES**

Many of default filenames listed below contain the string `%s`, which is replaced by the name with which `gated` is invoked. Normally this is `gated`, but if invoked as `gated-test`, `gated` will by default look for `/etc/gated-test.conf`. These paths may all be changed at compilation time.

- `/usr/sbin/gated` The `gated` binary.
- `/etc/gated.conf` Current `gated` configuration file.
- `/etc/gated.conf+` Newer configuration file.
- `/etc/gated.conf-` Older configuration file.
- `/etc/gated.conf--` Much older configuration file.
- `/var/run/gated.pid` Where `gated` stores its pid.
- `/var/tmp/gated_dump` `gated`’s state dump file.
- `/var/tmp/gated_parse` Where config file parse errors go.
- `/var/tmp` Where `gated` drops its core file.

**SEE ALSO**
gated(1M), ospf_monitor(1M), ripquery(1M), syslogd(1M), gated.conf(4), GateD Documentation, GateD Configuration Guide.

**BUGS**

Many commands only work when `gated` is installed in the system directory it was configured with.

There is not yet any way to tell gdc about systems which name their core dump other than `core` (`core.gated` is a less common possibility).
NAME
geocustoms - configure system language on multi-language systems

SYNOPSIS
geocustoms [-l locale]

DESCRIPTION
The geocustoms command manages default selection and retention/removal of multiple languages
installed on "ignited" (Instant Ignition) systems. The geocustoms command is executed at first-boot (see
auto_parms(1M)) on ignited systems with multiple languages available. In subsequent sessions, invoking
the command /usr/sbin/geocustoms starts geocustoms. When invoked with no options, geocustoms
runs interactively, either with a graphical display if the DISPLAY environment parameter is set
and usable, otherwise with a terminal interface.

Options
-1 locale  Non-interactively set the LANG parameter, and all other appropriate dependencies, based
on the value of locale. (To view all locales available on the system, run locale -a.) If
the locale argument is not a valid option for the system, an error message is issued and
the graphical or terminal user interface (UI) appears, as if the -l option were not used.

If the argument to the -l option is SET_NULL_LOCALE, locale parameters are set to NULL by default. A
null locale causes programs to execute without using localized message catalogs, which can increase system
performance. All HP-UX messages appear in English if the locale is set to NULL.

The geocustoms command must be invoked by a user with appropriate privilege.

If geocustoms is invoked directly by the user rather than automatically at first-boot, it might be neces-
sary to log out and log in again for language changes to take effect.

EXTERNAL INFLUENCES
Environment Parameters
The geocustoms command writes default values to system configuration files for the following environ-
ment parameters: LANG, LC_ALL, LC_CTYPE, LC_COLLATE, LC_MONETARY, LC_NUMERIC,
LC_TIME, and LC_MESSAGES.

International Code Set Support
Native Language Support (NLS)
If present, standard message catalogs reside in /usr/lib/nls and the geocustoms command uses
them. If standard message catalogs are not installed on the system, all messages appear in English, which
is standard NLS behavior.
All European languages for CDE (Common Desktop Environment) are supported. For HP-UX 11i Version 1
this includes English, French, German, Italian, Spanish and Swedish. All prompts and logging messages
are localized.
Locale (Language Variant) names are always localized in accordance with standard NLS behavior.
Note: NLS is extended to allow multiple "fonts" at the same time on the initial geocustoms graphical
screen through use of bitmapped images.

RETURN VALUES
0  Successful completion and/or clean exit from command.
1  Command unable to complete all objectives.

DIAGNOSTICS
Standard Output
The non-interactive geocustoms command does not write to stdout.

Standard Error
The geocustoms command writes to stderr and to /var/adm/sw/lang.log. It only writes to
stderr in case of command line error or a request for syntax (the conventional -? option). Any UI error
messages appear via an error window.
Logging
Both interactive and non-interactive sessions log summary events in `/var/adm/sw/lang.log`.

EXAMPLES
To set the default system language non-interactively to German:

```
/usr/sbin/geocustoms -l de_DE.iso88591
```

DEPENDENCIES
Compatibility
This product is designed for compatibility with releases including and after HP-UX 11.00 running a Common Desktop Environment (CDE). No attempt was made to support the Visual User Environment (VUE).

Note: If language bundles are marked for removal it can cause a delay of several minutes in `swagentd(1M)` at the next system boot.

Limitations
The `geocustoms` command does not do the following:

- Manage languages at the codeset level.
- Provide a user interface for Asian languages.
- Manage keyboard selection.
- Create or remove locale definitions.
- Provide a special interface for restoring or adding languages to the system from depots or media.

FILES
```
/usr/sbin/geocustoms
The command itself.

/usr/lib/nls/
Directory containing NLS language files, as discussed in Native Language Support above.

/etc/rc.config.d/LANG
/etc/dt/config/Xconfig
System configuration files modified by `geocustoms`.

/var/adm/sw/lang.log
Logfile created by `geocustoms`.
```

AUTHOR
The `geocustoms` command was developed by HP.

SEE ALSO
locale(1), auto_parms(1M), swinstall(1M), swlist(1M), swremove(1M), setlocale(3C).

STANDARDS CONFORMANCE
POSIX.2, UNIX95 (SPEC1170 and XPG4).
NAME
gext - get VxFS extent attributes

SYNOPSIS
  getext [-F vxfs] [-V] [-f] [-s] file...

DESCRIPTION
getext displays extent attribute information associated with a set of files.

Options
getext recognizes the following options:
  -f  Do not print the filenames for which extent attributes are displayed.
  -F vxfs  Specify the VxFS file system type.
  -s  Do not print output for files that do not have fixed extent sizes or reservations.
  -V  Echo the completed command line, but do not execute the command. The command
       line is generated by incorporating the user-specified options. This option allows the
       user to verify the command line.

Operands
getext recognizes the following operand:
  file  Name of file in a VxFS file system.

Notes
Only the align and noextend allocation flags (set through setext(1M) or the VX_SETTEXT ioctl) are per-
sistent attributes of the file and therefore visible via getext or the VX_GETEXT ioctl. trim is also visi-
ble, although it is cleared and the reservation is reduced on the final close of the file.

EXAMPLES
The following example shows a file with a block size of 1024 bytes, 36 blocks reserved, a fixed extent size of
3 blocks, and all extents aligned to 3 block boundaries:
  file1:  Bsize 1024 Reserve 36 Extent Size 3 align noextend

The file size cannot be extended once the current reservation is exhausted. Reservations and fixed extent
sizes are allocated in units of the file system block size.

SEE ALSO
setext(1M), vxfsio(7).
NAME
getfilexsec - display security attributes of binary executables

SYNOPSIS
getfilexsec [-r] [-R] [-p] [-P] [-f] [-c] filename ...

DESCRIPTION
The getfilexsec command displays various extended security attributes associated with binary executable files. These attributes include retained privileges, permitted privileges, and compartment and security attribute flags. See privileges(5) and exec(2)

Options
The getfilexsec command recognizes the following options:

The following options are supported:
- -c Displays the compartment name of the files.
- -f Displays security attribute flags. The only currently defined flag is the privilege start flag.
- -p Displays the minimum permitted privileges.
- -P Displays the maximum permitted privileges.
- -r Displays the minimum retained privileges.
- -R Displays the maximum retained privileges.

If none of the above options are specified, all extended security attributes of the binary files are displayed.

Operands
getfilexsec supports the following operand:

filename Binary executable file. All file names given as arguments must be binary executables.
Files of other types (for example, script executables, text files, and so on) are not permitted.

Security Restrictions
The user invoking this command must be able to open the directory in which the binary executable files are present.

RETURN VALUE
getfilexsec returns the following values:

0 Successful completion. The attributes are displayed.
>0 An error occurs. An error can be caused by an invalid option or inadequate permissions to perform the operation.

EXAMPLES
Example 1: Display the maximum permitted privileges and privilege-aware flag of binary executable file /web/java:

# getfilexsec -P -f /web/java

Sample output:

/web/java:
Flag: start_nil
PermittedMaxPrivileges: CMPTREAD, CMPTWRITE

SEE ALSO
setfilexsec(1M), exec(2), compartments(5), privileges(5).
NAME
getmemwindow - extracts window ids of user processes from /etc/services.window

SYNOPSIS
getmemwindow string

DESCRIPTION
getmemwindow is the command used to extract window ids of user processes from the
/etc/services.window file. User applications are encouraged to place a unique string defining an
application and its associated window id in the /etc/services.window file and then extract that win-
dow id using the getmemwindow command.

This allows for changing the application window id in one central location, rather than changing scripts
with hard-coded values, in the event of a memory window collision between two applications.

EXAMPLES
# Extract the window id for "HP" from the /etc/services.window file
# and start the program "HP_startup_script" with arguments arg1 and
# arg2.
#
WinId=$(getmemwindow HP)
setmemwindow -i $WinId HP_startup_script arg1 arg2

AUTHOR
getmemwindow was developed by HP.

FILES
/etc/services.window File containing applications' associated window id.

SEE ALSO
setmemwindow(1M), services.window(4), 11.0 Memory Windows White Paper.
NAME
getprocxsec - display security attributes of a process

SYNOPSIS
getprocxsec [-c] [-e] [-f] [-p] [-r] [pid]

DESCRIPTION
The getprocxsec command displays various security attributes associated with a running process. These attributes include the permitted privilege set, effective privilege set, retained privilege set, euid, and the compartment name. See privileges(5) and compartments(5).

Each process has a permitted privilege set, effective privilege set, and retained privilege set. If the compartmentalization feature is enabled, it also has a compartment. When a process is created, the child process inherits these attributes from the parent. When a process executes a binary, these attributes can be changed. See setfilexsec(1M) and getfilexsec(1M) for information on how these extended attributes can be manipulated at execution time.

For compatibility, the kernel handles processes with effective uid of zero in special ways. If the compartmentalization feature is disabled, these processes are treated as though they have all root replacement privileges. If, on the other hand, the compartmentalization feature is enabled, these processes are treated as though they have all the root replacement privileges except those configured as disallowed privileges for the compartment.

Options
getprocxsec recognizes the following options:
- c Displays the compartment name of the process. If compartments are not enabled, nothing is reported for this option.
- e Displays the implementation effective privilege set.
- f Displays the full form of the lists.
- p Displays the implementation permitted privilege set.
- r Display the implementation retained privilege set.

If none of the above options are specified, the default is -perc -1.

Operands
getprocxsec recognizes the following operand:
pid The process ID of the process whose attributes are being displayed. If pid is -1, getprocxsec displays attributes of this process. If pid is -2, it displays attributes of the process’ parent. If pid is not specified, it defaults to this process (equivalent to -1).

Security Restrictions
The specified process must be visible to the user invoking this command or the user must have the COM-MALLOWED privilege.

RETURN VALUE
getprocxsec returns the following values:
0 Successful completion. The attributes are displayed.
>0 An error occurred. An error can be caused by an invalid option or because the specified process is not visible to the user.

EXAMPLES
Example 1
Display the privilege sets and compartment of the current process:

# getprocxsec

Sample output:
effective= BASIC
permitted= BASIC
retained= BASIC
cmpt= init
Example 2
Display the privilege sets and compartment of the parent process:

```
# getprocxsec -2
```

Sample output:
```
effective= BASIC
permitted= BASIC
retained= BASIC
cmpt= init
euid= zero
```

Example 3
Display the full privilege sets and compartment of an arbitrary process:

```
# getprocxsec -f 801
```

Sample output:
```
effective= FORK EXEC SESSION LINKANY
permitted= FORK EXEC SESSION LINKANY
retained= FORK EXEC SESSION LINKANY
cmpt= web
euid= non-zero
```

SEE ALSO
getfilexsec(1M), setfilexsec(1M), compartments(5), privileges(5).
getprpw(1M)

NAME
getprpw - display protected password database

SYNOPSIS
getprpw [-l] [-n [domain]] [-r] [-m parm[,parm]] username

DESCRIPTION
getprpw displays the user's protected password database settings. This command is available only to the
superuser in a trusted system. Normally it is only used via SAM, see sam(1M).

The database contains information for both local and NIS+ users. However, some NIS+ information is kept
on the master. Since a user may be both local and NIS+, getprpw uses the nsswitch.conf(4) default if nei-
ther -l nor -n are specified.

Options
getprpw recognizes the following options...

-l Specifies to get information from the local user.

-n Can be specified with or without domain name; i.e., -n [domain]. If -n [domain] is specified,
dispalyes data for the NIS+ user. The domain name must be fully qualified, with a terminating period.
If domain name is not specified, the local domain will be used.

-r Displays the arguments supplied to -m in raw format

-m Displays the database value for the argument passed.

An "invalid-opt" is printed if a list of options passed to -m "contains an invalid option. The rest of the
options will be processed. If getprpw is specified without -m, all parameters are displayed in the
order given below.

Boolean values are returned as YES, NO, or DFT (for system default values in
/tcb/files/auth/system/default).

Numeric values are specified as positive numbers, 0, or -1. A value of -1 indicates that the field has
not been assigned a value in the database.

Units of time are returned in number of days (>=0), although the database keeps them in seconds.
This and other minor differences between the command parameters and the database fields are con-
sistent with modprpw(1M).

The following parameters for the user can be displayed using the -m option.
They are listed below in the order shown in prot.h. The database fields are fully explained in
prpwd(4).

uid user uid
bootpw boot authorization flag
audid audit id
audflg audit flag
mintm minimum time between password changes
maxpwln maximum password length
exptm password expiration time
lftm password lifetime
spwchg last successful password change time
upwchg last unsuccessful password change time
acctexp account expiration time
llog last login time interval
expwarn password expiration warning time
usrpick whether user picks password, YES/NO/DFT
getprpw(1M)

sysnpw   whether system generates pronounceable passwords, **YES/NO/DFT**
restpw   whether password is restricted, i.e. checked for triviality, **YES/NO/DFT**
nulppw   NULL passwords are allowed, **YES/NO/DFT**. **Not recommended!**
syschpw  whether system generates passwords having characters only, **YES/NO/DFT**
sysltpw  whether system generates passwords having letters only, **YES/NO/DFT**
timeod   time of day allowed for login
slogint  time of last successful login
uologint time of last unsuccessful login
sloginy  tty of last successful login
uologiny tty of last unsuccessful login
umaxlntr maximum unsuccessful login tries
alock    administrator lock, **YES** if on, **NO** if off, **DFT** if not set.
lockout  returns the reason for a lockout in a "bit" valued string, where 0 = condition not present, 1 is present. The position, left to right represents:
  1 past password lifetime
  2 past last login time (inactive account)
  3 past absolute account lifetime
  4 exceeded unsuccessful login attempts
  5 password required and a null password
  6 admin lock
  7 password is a *

RETURN VALUE

0   success
1   user not privileged
2   incorrect usage
3   cannot find the password file
4   system is not trusted

EXAMPLES
Displays the database aging fields for user "someusr".

    getprpw -m minm,exptm,expwarn,lftm someusr

The command displays:

    minm=1, exptm=2, expwarn=-1, lftm=3

WARNINGS

This command is intended for SAM use only. It may change with each release and can not be guaranteed to be backward compatible.

Several database fields interact with others. The side effects of an individual change may not cause a problem till much later.

Special meanings may apply in the following cases:

- an absent field
- a field without a value
- a field with a zero value

AUTHOR

getprpw was developed by HP.

FILES

/etc/passwd   System Password file
getprpw(1M)

/\tcb/files/auth/*/" Protected Password Database
/\tcb/files/auth/system/default System Defaults Database

SEE ALSO
modprpw(1M), prpwd(4), nsswitch.conf(4).
getrules(1M)

NAME
getrules - display compartment rules

SYNOPSIS
getrules [-f] [-i] [-n] [-p] [-P] [compartment_name]...
getrules -l network_interface_name[...]

DESCRIPTION
getrules displays rules defined for compartment(s) or network interface(s). This command can only be
used when compartmentalization is enabled (see cmpt_tune(1M)).

If no options are specified, all subsystem rules for the given compartment are displayed. If no
compartment_name is specified, information on all compartments is displayed.

Options
getrules recognizes the following options:
-f Displays the file system rules for the compartment(s).
-i Displays the IPC system rules for the compartment(s).
-l Displays the compartment names associated with the network interface(s).
-n Displays the network system rules for the compartment(s).
-p Displays the disallowed privileges list in short form for compartment(s). The short form includes
compound privileges in the privilege list.
-P Displays the disallowed privileges list in literal form for compartment(s). The literal form
expands compound privileges in the privilege list.

If the -l option is used, you must specify a network_interface_name.

Operands
getrules recognizes the following operands:
compartment_name Name of the compartment for which information is displayed.
network_interface_name Name of the network interface for which information is displayed.

For compartment_name or network_interface_name, you can specify multiple instances separated by spaces.

Notes
The getrules command is provided for diagnostic purposes, and as such the output of the command may change.

Some rules can be expressed in multiple forms. For instance, compartment A specifying that it can
send a signal to compartment B is the same as compartment B specifying that it can receive signals
from compartment A. As this command displays the rules only once, it can be misleading to interpret
the output.

RETURN VALUE
getrules returns the following values:
0 Successful completion. The rules are displayed.
>0 An error occurred. An error can be caused by an invalid option or because the user does not have permissions to perform the operation.

EXAMPLES
Example: Display all file system rules for the compartment named web:

# getrules -f web
Sample output:

Compartment Name: web : sealed
Disallowed Privileges: POLICY

File System Rules:
------------------
PERMISSION PATHNAME
read, write, create, unlink /

SEE ALSO

cmpt_tune(1M), setrules(1M), compartments(4), compartments(5), privileges(5).
NAME
  getty - set terminal type, modes, speed, and line discipline

SYNOPSIS
/usr/sbin/getty [-f] [-h] [-t timeout] line [speed [type [linedesc]]]
/usr/sbin/getty -c file

DESCRIPTION
getty is a program that is invoked by init (see init(1M)). It is the second process in the series,
init>getty>login>shell, that ultimately connects a user with the HP-UX system. Initially, if the
/etc/issue file exists, getty prints its contents to the user's terminal, followed by the login message
field for the entry it is using from the /etc/gettydefs file. getty reads the user's login name and
invokes the login command with the user's name as login arguments (see login(1)). While reading the
name, getty attempts to adapt the system to the speed and type of terminal being used. See Operation
below for more detail.

Options and Operands
getty recognizes the following options and operands:

  line
  The name of a tty line in /dev to which getty is to attach itself. getty uses this
  string as the name of a file in the /dev directory to open for reading and writing. By
default, getty forces a hangup on the line by setting the speed to zero before setting
the speed to the default or specified speed. However, when getty is run on a direct
port, getty does not force a hangup on the line since the driver ignores changes to
zero speed on ports open in direct mode (see modem(7)).

  -f
  Tells getty to get the default settings for special control characters from the
/dev/ttyconf file.

  Use the stty command to configure the settings by changing the control characters
in /dev/ttyconf (see stty(1)). This flexibility allows the user to login by means of getty and type a #, @,
and so on, as part of the login name or arguments, if these are not the special control characters
defined in /dev/ttyconf.

  -h
  Tells getty not to force a hangup on the line before setting the speed to the default
or specified speed.

  -t timeout
  Tells getty to exit if the open on the line succeeds and no one types anything within
timeout seconds.

  speed
  The label of a speed and tty definition in the file /etc/gettydefs. This definition
tells getty at what speed to run initially, what the login message should look like,
what the initial tty settings are, and what speed to try next should the user indicate
that the speed is inappropriate by typing a break character. The default speed is 300
baud.

  type
  A character string describing what type of terminal is connected to the line in ques-
tion. getty understands the following types:
    none  default
    vt01  DEC vt100
    vt100 DEC vt100
    hp45  Hewlett-Packard HP2645
    c100  Concept 100

The default terminal is none; that is, any CRT or normal terminal unknown to the
system. Also, for terminal type to have any meaning, the virtual terminal handlers
must be compiled into the operating system. They are available, but not compiled, in
the default condition.

  linedesc
  A character string describing which line discipline to use when communicating with
the terminal. Hooks for line disciplines are available in the operating system, but
there is only one presently available — the default line discipline, LDISCO.
**Operation**

When given no optional arguments, `getty` sets the speed of the interface to 300 baud, specifies that raw mode is to be used (awaken on every character), that echo is to be suppressed, that either parity is to be allowed, that newline characters are to be converted to carriage-return-linefeed, and that tab expansion is to be performed on standard output. It types the login message before reading the user's name a character at a time. If a null character (or framing error) is received, it is assumed to be the result of the user pushing the *break* key. This causes `getty` to attempt the next speed in the series. The series that `getty` tries is determined by what it finds in `/etc/gettydefs`.

The user's name is terminated by a newline or carriage-return character. The latter results in the system being set to treat carriage returns appropriately (see `ioctl(2)`).

The user's name is scanned to see if it contains any lowercase alphabetic characters; if not, and if the name is nonempty, the system is told to map any future uppercase characters into the corresponding lowercase characters.

`getty` also understands the "standard" ESS2 protocols for erasing, killing and aborting a line, and terminating a line. If `getty` sees the ESS erase character (_), or kill character ($), or abort character (&), or the ESS line terminators (/ or !), it arranges for this set of characters to be used for these functions.

With the `-f` option, `getty` obtains the special control characters for the terminal from the `/dev/ttyconf` file. This terminal setting will be inherited by `login`.

Finally, `login` is called with the user's name as an argument. Additional arguments can be typed after the login name. These are passed to `login`, which places them in the environment (see `login(1)`).

**Check Option**

A check option is provided. When `getty` is invoked with the `-c` file option, it scans file as if scanning `/etc/gettydefs` and prints the results on standard output. If there are any unrecognized modes or improperly constructed entries, `getty` reports these. If the entries are correct, `getty` prints out the values of the various flags. See `ioctl(2)` for an interpretation of values. Note that some values are added to the flags automatically.

**WARNINGS**

While `getty` does understand simple single character quoting conventions, it is not possible to quote the special control characters that `getty` uses to determine when the end of the line has been reached, which protocol is being used, and what the erase character is.

**DEPENDENCIES**

**HP2334 MultiMux**

The modem control parameter MRTS must be present in the `/etc/gettydefs` file when using `getty` in conjunction with an HP2334 or HP2335 MultiMux to ensure that the RTS modem control signal is asserted correctly.

Example:

```
9600# B9600 HUPCL PARENB MRTS # B9600 SANE PARENB ISTRIP IXANY #login: #19200
```

MRTS is not intended for use with devices other than the HP2334 or HP2335 MultiMux.

**FILES**

```
/dev/ttyconf
/etc/gettydefs
/etc/issue
```

**SEE ALSO**

`ct(1)`, `login(1)`, `stty(1)`, `init(1M)`, `ioctl(2)`, `gettydefs(4)`, `inittab(4)`, `modem(7)`, `termio(7)`.
NAME
getx25 - get x25 line

SYNOPSIS
/usr/sbin/getx25 line speed pad-type

DESCRIPTION
The uucp commands, including getx25, are targeted for removal from HP-UX; see the WARNINGS below.

getx25 is functionally very similar to getty (see getty(1M)) but is used only for incoming lines that are connected to an X.25 PAD. It performs special functions such as setting up an initial PAD configuration. It also logs the number of the caller in /var/uucp/.Log/LOGX25. The third parameter is the name of the PAD being used. HP 2334A is the only one supported at this time. A typical invocation would be:

/usr/sbin/getx25 x25.1 2 HP2334A

WARNINGS
Use of uucp commands, including getx25, is discouraged because they are targeted for removal from HP-UX. Use ftp(1) or rcp(1) instead.

AUTHOR
getx25 was developed by HP.

SEE ALSO
login(1), uucp(1), getty(1M).
NAME
groupadd - add a new group to the system

SYNOPSIS
   groupadd [-g gid [-o] ] group

DESCRIPTION
   The groupadd command creates a new group on the system by adding the appropriate entry to the
   /etc/group file. The groupadd command expects the group argument, which is the name of the new
   group. The name consists of a string of printable characters that may not include a colon (:) or newline
   (\n).

Options
   The groupadd command may be used with the following options:
   -g gid   Specifies the group ID for the new group. gid must be a non-negative decimal integer less
             than MAXUID as defined in the <param.h> header file. By default the next available
             unique group ID in the valid range is allocated. Group IDs in the range 0-99 are reserved.
   -o       Allow the gid to be non-unique (i.e., a duplicate).

NETWORKING FEATURES
   The groupadd command is aware of NIS user entries. Only local groups may be added with this com-
   mand. Attempts to add an NIS group will result in an error. NIS groups must be administered from the
   NIS server. If groupadd is used on a system where NIS is installed, it may fail with the error
   group x is not unique
   (return value 9) if the group specified is not present in the local /etc/group file, but is an NIS group
   (see group(4)). NIS groups are also checked when verifying uniqueness of the new gid, which may result in
   the error
   GID # is not unique
   (return value 4).

RETURN VALUE
   The groupadd command exits with one of the following values:
   0   No error.
   2   Invalid command syntax.
   3   Invalid argument supplied to an option.
   4   gid is not unique (when -o is not used).
   9   group is not unique.
   10  Cannot modify the /etc/group file.
   11  /etc/passwd file or /etc/ptmp file busy. Another command may be modifying the
        /etc/passwd file.
   12  Unable to open /etc/ptmp file or /etc/passwd file is non-existent.

EXAMPLES
   Add the group project1 to the /etc/group file.
   groupadd project1
   Add the group project12 to the /etc/group file with the group ID 111 as long as no group currently
   exists with a group ID of 111.
   groupadd -g 111 project12

WARNINGS
   As many users may try to write the /etc/passwd file simultaneously, a passwd locking mechanism was
   devised. If this locking fails after subsequent retrying, groupadd terminates.

FILES
   /etc/group
   /etc/ptmp
SEE ALSO

   users(1), groupdel(1M), groupmod(1M), login(1M), useradd(1M), userdel(1M), usermod(1M), group(4).

STANDARDS CONFORMANCE

   groupadd: SVID3
NAME
groupdel - delete a group from the system

SYNOPSIS
groupdel group

DESCRIPTION
The groupdel command deletes a group from the system by removing the appropriate entry from the /etc/group file.
The groupdel command must be used with the group argument. group is the name of the group to be deleted, consisting of a string of printable characters.

NETWORKING FEATURES
This command is aware of NIS user entries. Only local groups may be deleted with groupdel. Attempts to delete an NIS group will result in an error. NIS groups must be administered from the NIS server. If groupdel is used on a system where NIS is installed, it may fail with the error
group x does not exist
(return value 6), if the group specified is an NIS group (see group(4)).

RETURN VALUE
groupdel exits with one of the following values:
  0  No error.
  2  Invalid command syntax.
  3  Invalid argument supplied to an option.
  6  group does not exist.
  10 Cannot modify the /etc/group file.
  11 /etc/passwd file or /etc/ptmp file busy. Another command may be modifying the /etc/passwd file.
  12 Unable to open /etc/ptmp or /etc/passwd file is non-existent.

EXAMPLES
Delete the group project1 from the /etc/group file if it exists:
groupdel project1

WARNINGS
As many users may try to write the /etc/passwd file simultaneously, a passwd locking mechanism was devised. If this locking fails after subsequent retrying, groupdel terminates.

FILES
/etc/group
/etc/ptmp

SEE ALSO
users(1), groupadd(1M), groupmod(1M), logins(1M), useradd(1M), userdel(1M), usermod(1M), group(4).

STANDARDS CONFORMANCE
  groupdel: SVID3
NAME
groupmod - modify a group on the system

SYNOPSIS
groupmod [-g gid [-o] [-n name]] group

DESCRIPTION
The groupmod command modifies a group on the system by altering the appropriate entry in the
/etc/group file.
The groupmod command must be used with the group argument, which is the name of the group to be
modified.

Options
The groupmod command may be used with the following options:
- g gid Change the value of the group ID to gid. gid must be a non-negative decimal integer less
than MAXUID as defined in the <param.h> header file.
- o Allow the gid to be non-unique (i.e., a duplicate).
- n name Change the name of the group to name. name consists of a string of printable characters
that may not include a colon (:) or newline (\n).

NETWORKING FEATURES
This command is aware of NIS user entries. Only local groups may be modified with groupmod. Attempts
to modify an NIS group will result in an error. NIS groups must be administered from the NIS server. If
groupmod is used on a system where NIS is installed, it may fail with the error

  group x does not exist

  (return value 6) if the group specified is an NIS group (see group(4)). However, NIS groups are checked
when verifying uniqueness of the new gid or new group name, which may result in the above error, or the
error

  GID # is not unique

  (return value 4).

RETURN VALUES
groupmod exits with one of the following values:
0 No error.
2 Invalid command syntax.
3 Invalid argument supplied to an option.
4 gid is not unique (when -o is not used).
6 group does not exist.
9 group is not unique.
10 Cannot modify the /etc/group file.
11 /etc/passwd file or /etc/ptmp file busy. Another command may be modifying the
   /etc/passwd file.
12 Unable to open /etc/ptmp file or the /etc/passwd file is non-existent.

EXAMPLES
Change the group ID of the group project2 to 111 in the file /etc/group if the group project2
exists. This is done even if the group ID 111 is already in use.
groupmod -g 111 -o project2
Change the name of project2 to project22 in the file /etc/group if the group project22 does
not already exist.
groupmod -n project22 project2

WARNINGS
As many users may try to write the /etc/passwd file simultaneously, a passwd locking mechanism was
deviced. If this locking fails after subsequent retrying, groupmod terminates.
groupmod(1M)

FILES
 /etc/group
 /etc/ptmp

SEE ALSO
 users(1), groupadd(1M), groupdel(1M), login(1M), useradd(1M), userdel(1M), usermod(1M), group(4).

STANDARDS CONFORMANCE
 groupmod: SVID3
hosts_to_named(1M)  

NAME  hosts_to_named - translate host table to name server file format

SYNOPSIS  
hosts_to_named -d domain -n network-number [options]

DESCRIPTION  
hosts_to_named translates the host table, /etc/hosts, into files that are usable by the name server named(1M). The format of these files is defined in RFC 1035. The files are created in the current directory. Once the host table is translated, the name server files can be maintained directly, or the translation can be repeated after each change to the host table.

If a line in the host table contains no domain names, all names on the line are assumed to be in the default domain. The first domain listed is the "default domain". If data is being created for more than one domain or if certain options are used, there must be domain names in the host table to determine which names belong in which domain.

The name server data is referred to as 'resource records'.

Options are:

- a network-number
  Add the information about hosts in the local domain from network network-number. This is the same as the -n option except that no pointer (PTR) data is created. This is useful when there are multiple domains on a network and a different server is handling the address-to-name mapping for network-number.

- b bootfile
  Name the boot file bootfile. The default is named.boot (if named is 4.x) or named.conf (if named is 8.x or 9.x) in the current directory.

- c subdomain
  Create alias (CNAME) records for hosts in subdomain of the default domain. When a subdomain is delegated, it is useful to create aliases for the old names in the default domain that point to the new names in the subdomain. After creating the alias (CNAME) records, ignore lines in the host table that contain names in the subdomain. This option can be used more than once on the command line. This option requires domain names in the host table. When the old names in this domain are no longer used, they can be ignored with the -e option. If the subdomain name does not have dots, the default domain is appended to subdomain.

- d domain
  Create data for domain. This option can be used more than once on the command line if data is being created for more than one domain. The first domain listed is the "default domain". This option requires domain names in the host table for all hosts in domains except the default domain.

- e subdomain
  Eliminate lines from the host table that contain names in the subdomain before translating. If the subdomain name does not have dots, the default domain is appended. This option may be used more than once on the command line. This option requires domain names in the host table.

- f file
  Read command line options from file. The -f option is not allowed within a file.

- h host
  Declare host to be the host in the start of authority (SOA) record that the name server data was created on. Also use host for the electronic mail address of the responsible user in the SOA record. The default is the host this command is run on. This option applies to all the domains generated by hosts_to_named. If this option is specified more than once, the last value will be considered.

- m weight:mailhub
  For each canonical hostname from the host table, create mail exchanger (MX) records with the specified weight and mail hub. The weight is a positive integer. The mail hub is a hostname. If the mail hub name has no dots, the default domain is appended. This option can be used more than once on the command line.

- n network-number[=mask]
  Create data for network-number. See below for description of network-number. If only one domain is listed with -d, all data for network-number is assumed to be in domain. The optional subnet mask mask can be used instead of supplying each network-number for a subnet using multiple -n options. mask must be in dot
-o refresh:retry:expire:min
Set the values in the start-of-authority (SOA) record to those specified. See below for description of the start-of-authority (SOA) record.

-p domain
Create only pointer (PTR) data for hosts in domain. This is useful when there are multiple domains on a network and a different server is responsible for domain, but this server is responsible for the address-to-name mapping. This option can be used more than once on the command line. This option requires domain names in the host table.

-q
Run quietly. No messages are printed.

-r
Create name server data indicating that the name server is authoritative for . (the root of the domain tree). The file created is db.root. Use this only when your network is isolated from the Internet. If other root servers exist for the isolated network, they must be added manually.

-s server
Create name server (NS) records that declare server is an authoritative name server for all of the domains created. If more than one server is authoritative, each needs to be declared. If the server name does not have any dots in it, the default domain is appended. If you do not specify the -s option, the default name server is the host on which you execute the script. You can use this option more than once on the command line to specify multiple name servers.

-t
Create text (TXT) records from the comments that appear with host data. The comments will all be in lower case because the host table is translated to lower case. If [no smtp] appears in a comment, it is omitted. The [no smtp] is used to control mail exchanger (MX) data.

-u user
Declare user to be the electronic mail address of the person responsible for this domain. This is used in the start of authority (SOA) record. The format required in the name server data is user,host (host must be a domain name). If given as user, the host on which this script is run is appended. If given as user@host, the @ is replaced with a dot (.). The default user is root. This option applies to all the domains generated by hosts_to_named. If you specify this option more than once, the last value will be used.

-w
Create well known services (WKS) data declaring that the host provides the SMTP service. This is done only when mail exchanger (MX) data is also being created and only for hosts without [no smtp] in a comment.

-z internet-address
Create a secondary boot file, boot.sec.save (if named is 4.x) or conf.sec.save (if named is 8.x or 9.x), from the primary boot file listing internet-address as the server to load the data from. The boot file has the server back up the data on disk. The internet-address defaults to the value used with -z. You can specify this option more than once to declare multiple servers from which data can be loaded.

Note: All the servers declared using this option will be listed for all the domains.

-A
Do not create name server data for aliases in the host table.

-C file
Create resource records from strings in the comment field of the host table. Each string in the comment field (except [no smtp]) is searched for in file. The format of file is a string, a colon, and a resource record. If the string in the comment field matches the string before the colon in file, a resource record is added consisting of the name of the host followed by everything after the colon from the matching line in file. For example, host information (HINFO) records can be created by adding 360:IN HINFO hp9000s360 hp-ux to file and adding 360 to comments in the host table.

-D
Do not create name server data for domain names in the host table.

-F
By default, the serial number is incremented for a domain only if the data has changed (pointer (PTR) data only). This option forces the serial number to be incremented, even if the data has not changed.

hosts_to_named(1M) hosts_to_named(1M)

notation. This option is not applicable for creating IPv6 addresses.
hosts_to_named(1M)

- **H host-file** Use host-file instead of /etc/hosts.

- **-M** Do not create mail exchanger (MX) records for hosts in the host table.

- **-N mask** Apply the default subnet mask `mask` to each `network-number` specified with `-n` except for ones with their subnet masks already provided. `mask` must be in dot notation. This is the same as supplying each `network-number` for a subnet using multiple `-n` options.

- **-S server** This option is the same as the `-s` option, but it only applies to the last `domain` specified with `-d` or the last `network-number` specified with `-n`. This option is for when `server` is backing up some, but not all, of the domains.

**Note:** At least one name server (either default or explicitly declared using the `-s` option) will be included in all the database files, in addition to those declared using the `-S` option.

- **-Z internet-address** Create a secondary boot file, `boot.sec` (if `named` is 4.x) or `conf.sec` (if `named` is 8.x or 9.x), from the primary boot file listing `internet-address` as the server to load the data from. The boot file does not have the server back up the data on disk. The `internet-address` defaults to value used with `-z`. You can use this option more than once to declare multiple servers from which data can be loaded.

**Note:** All the servers declared using this option will be listed for all domains.

- **-1** This option is obsolete.

`hosts_to_named` translates the host table to lower case to help eliminate duplicate data. Since the name server treats uppercase and lowercase as equivalent, names that differ only in case are considered the same.

The `hosts_to_named` configuration file migration script does not add the `listen-on-v6` option to the `named.conf` file on dual stack machine. In order to enable DNS services to listen on IPv6 interfaces, this entry needs to be added manually.

Alias (CNAME) records are created for `subdomains` delegated with `-c`. Lines from the host table that contain names in `subdomains` from `-c` and `-e` are removed from the lowercase copy of the host table.

The host table is then used to create the name server data for each `network-number` declared on the command line. Do not include the trailing 0's in the network number. No distinction is made between class A, B, or C addresses nor is there any understanding of subnets unless a subnet mask is supplied. Example network numbers are: 10 (for all addresses of the form 10.*.*.*), 10.1 (for addresses of the form 10.1.*.*), or 10.2.2 (for addresses of the form 10.2.2.*).

Address (A) records are created for mapping hostnames to IP addresses. Alias (CNAME) records are created for aliases of hosts that are not multi-homed. The data are placed in a file named `db.DOMAIN` where `DOMAIN` is the first part of the domain from the command line. For the domain `div.inc.com`, the file is named `db.div`. All other name server data goes in this file except the pointer (PTR) records described below.

Pointer (PTR) records are created for mapping IP addresses to host names. PTR records are placed in a file named `db.NET` where `NET` is the network number from the command line. Network 10 data is placed in `db.10`. Network 10.1 data are placed in “db.10.1”.

Address (AAAA) records and A6 records are created for mapping hostnames to IPv6 addresses.

Pointer (PTR) records are created for mapping IPv6 addresses to host names. The IPv6 PTR records are placed in a file named `db.IP6.INT`.

Mail exchanger (MX) records are created unless the `-M` option is used. The default MX record has a weight of 10 with the host itself as its mail exchanger. No default MX record is created for a host if `[no smtp]` is in the comment section of that line in the host table. MX records for each mail hub declared with the `-m` option are added for each host even if `[no smtp]` is in the comment section.

Well known services (WKS) records are created for each host that handles SMTP mail (does not have `[no smtp]`) if `-w` is used. The only service listed is SMTP.

Text (TXT) records are created for comments associated with hosts in the host table if `-t` is used. The comments do not include `[no smtp]`.

HP-UX 11i Version 2: December 2007 Update

Hewlett-Packard Company

347
For each domain, a start of authority (SOA) record is created. The SOA record requires two domain names: the host that the data is created on and the electronic mail address of the person responsible. The `-h` and `-u` options influence the names. In addition, the SOA record requires 5 values: a serial number, a refresh time, a retry time, an expire time, and a minimum ttl (time to live). The first time the data is created, the serial number is set to 1, the refresh time is set to 3 hours, the retry time is set to 1 hour, the expire time is set to 1 week, and the minimum ttl is set to 1 day. The `-o` option changes these values except for the serial number. Each subsequent time `hosts_to_named` is run, the serial number is incremented. If any of the other fields in the SOA record are modified, the changed values are retained.

If there are files named `spcl.DOMAIN` or `spcl.NET` in the current directory, `$INCLUDE` directives are added to the corresponding `db.DOMAIN` or `db.NET` file for the `spcl` file. In this way, special data can be added to the data generated by `hosts_to_named`.

The first time `hosts_to_named` is run, it creates a default boot file for a primary name server. Each subsequent time `hosts_to_named` is run, the boot file is updated if necessary. New entries are made in the boot file for any additional networks or domains not already in the boot file. No entries are deleted from the boot file.

The boot file for a caching-only server, `boot.cacheonly` (if `bind` is 4.x) or `conf.cacheonly` (if `bind` is 8.x or 9.x), is created if it does not exist. The boot files for secondary servers, `boot.sec.save` or `conf.sec.save`) and (`boot.sec` or `conf.sec`), are created if the `-z` or `-Z` options are used. The boot files for secondary servers are created new each time from the primary server boot file so that they are equivalent.

### EXAMPLES

Create name server data for networks 15.19.8 and 15.19.9 in `div.inc.com`.

```bash
hosts_to_named -d div.inc.com -n 15.19.8 -n 15.19.9
```


```bash
hosts_to_named -f option_file
```

Option file contains the following lines:

- `d div.inc.com`
- `n 15.19.8` `-n 15.19.9`
- `m 20:aaa`
- `m 30:bbb.mkt.inc.com`
- `A`

Network 15.19.15 has hosts in the `xx.inc.com` domain and the `div.inc.com` domain. Create name server data for `xx.inc.com`. Create only pointer (PTR) data for hosts in `div.inc.com` on network 15.19.15 (this requires the hosts in `div.inc.com` to have the canonical name or an alias of the form `x.div.inc.com`).

```bash
hosts_to_named -d xx.inc.com -n 15.19.15 -p div.inc.com
```

Create name server data for network 15.19.8 in `div.inc.com`. Include `div.inc.com` data from network 15.19.15 but do not create pointer (PTR) data for 15.19.15 since that is being handled by the `xx.inc.com` server.

```bash
hosts_to_named -d div.inc.com -n 15.19.8 -a 15.19.15
```

Create name server data for multiple domains `div1.inc.com`, `div2.inc.com` and `div3.inc.com`. You must declare 15.18.1.1 as the host in the start of authority (SOA) record for all the domains. Also, create a secondary boot file from the primary boot file listing 15.18.1.1 and 15.18.2.1 as the servers to load the data from, to all the domains. Declare 15.18.2.1 as an authoritative name server for the domain `div2.inc.com` in addition to the default server.

```bash
hosts_to_named -f option_file
```

Option file contains the following lines:

- `d div1.inc.com`
- `n 15.18.1`
- `d div2.inc.com`
- `n 15.18.2`

Hewlett-Packard Company - 4 - HP-UX 11i Version 2: December 2007 Update
hosts_to_named(1M)

-S 15.18.2.1
-d div3.inc.com
-n 15.18.3
-h 15.18.1.1
-z 15.18.1.1
-z 15.18.2.1

AUTHOR
hosts_to_named was developed by HP.

FILES
/etc/hosts          The host table
named.boot          Primary server boot file (4.x)
named.conf          Primary server boot file (8.x or 9.x)
boot.cacheonly      Caching only server boot file (4.x)
conf.cacheonly      Caching only server boot file (8.x or 9.x)
boot.sec.save       Secondary server boot file (4.x)
conf.sec.save       Secondary server boot file (8.x or 9.x)
boot.sec            Secondary server boot file (4.x)
conf.sec            Secondary server boot file (8.x or 9.x)
db.127.0.0          Pointer information for 127.0.0.1
db.cache            Stub cache file for root server addresses
db.root             Data for servers for the root domain
db.DOMAIN           Address and other data for a domain
db.DOMAIN.in-addr   Pointer data for all network-numbers
db.NET              Pointer data for a network-number
db.IP6.INT          Pointer data for a IPv6 network-number

SEE ALSO
named(1M).
RFC 1034, RFC 1035.
NAME
hotplugd - PCI I/O hotplug (attention button) events daemon

SYNOPSIS
/usr/sbin/hotplugd logfile openmode

DESCRIPTION
The hotplugd daemon handles PCI I/O hotplug (also known as attention button, AB, or doorbell) events that are generated by pressing the attention button corresponding to a PCI I/O slot.

Only one attention button event is processed by the system at any point in time. If more than one attention button is pressed, the events are put in a queue within the kernel.

This daemon invokes /usr/bin/olrad to perform the various online addition or replacement operations (OLAR, OL*) on the slot on which the attention button has been pressed. The daemon does not use the -f (override CRA results) option of olrad.

Normally, this daemon is started by the /sbin/init.d/pci_olar startup script at boot time. See the Daemon Startup subsection for details.

If an attention button is pressed before the startup of the daemon, the event is dropped and no messages are logged. Only users with superuser privileges may use this command to perform OL* functions.

Currently, only the online addition and replacement functions are supported. Online deletion is not supported.

The blinking of the power LED is not supported on all hardware platforms.

Operands
The following operands are required. See also the Daemon Startup subsection.

logfile Log file where the daemon will log its messages. The standard output and standard error of the /usr/bin/olrad command are also captured in this log file.

openmode Mode in which to open the logfile. It can be one of the following:

append Open the file in append mode. New log information is appended at the end of logfile.

trunc Open the file and truncate it. If the logfile exists, its length is truncated to 0 and the mode and owner are unchanged.

Daemon Startup
In the normal (and recommended) operation hotplugd is invoked at boot time through the startup script, /sbin/init.d/pci_olar. The startup script reads the configuration file, /etc/rc.config.d/pci_olar for the configuration file variables, HOTPLUGD_LOGFILE and HOTPLUGD_MODE, which are assigned to the logfile and openmode parameters, respectively. The installed default values are:

HOTPLUGD_LOGFILE=/var/adm/hotplugd.log
HOTPLUGD_MODE=trunc

To start a new instance of hotplugd with new values for logfile and mode, you can do so by stopping the running instance of hotplugd and manually starting it with the new values.

To have hotplugd always invoked on startup with the new values, change the values of the HOTPLUGD_LOGFILE and HOTPLUGD_MODE variables in the configuration file, /etc/rc.config.d/pci_olar. At the next boot, hotplugd will be invoked with these new values.

DIAGNOSTICS
Messages in the System Log File
The following messages are logged in the system log file, /var/adm/syslog/syslog.log. If an attention button event occurs after the daemon has been terminated following a successful start of the daemon, the messages are logged in the system log file.
Could not allocate memory for PCI I/O Attention Button event
A chunk of memory, used for storing the attention button event information, could not be allocated.
The event is dropped after the power LED is set to PWR_RAIL mode. That is, the power LED will be set to ON if the power to the slot is ON, or the power LED will be set to OFF if the power to the slot is OFF.

Could not add the event to the queue, dropping AB event
The queue has not been initialized (daemon not running) or the queue is under flow control. The queue can go into flow control if the queue already has maximum number of events. Currently the maximum is 128.
The event is dropped after setting the power LED to PWR_RAIL mode (see previous message). Wait till the attention button events are processed by the daemon.

Dropping AB event, slotId not found in the registered slotIds
The attention button event was received on a slot ID that has not been registered (during the system bootup) with the kernel OLAR module.
The event is dropped after setting the power LED to PWR_RAIL mode (see previous message). Contact your HP response center.

Could not get the slot information for PCI I/O Attention Button event :
slotId , olar_error_no = olarErrMsg
The call to get the slot information failed. olarErrMsg gives the reason for the error.
This results in an invalid PCI OL* event to the daemon.

hotplugd daemon (pid= pid) is shutting down... dropping AB event : slotId
The system is clearing the attention button events in the queue due to the abnormal termination of the daemon (for example, with the SIGKILL signal (see kill(1)) and there are pending attention button events in the queue.
While clearing the events from the queue, the system sets the power LED corresponding to the slotId, to PWR_RAIL mode (see previous message).

Setting the power led to PWR_RAIL mode failed : slotId , olar_error_no = olarErrMsg
Setting the power LED to PWR_RAIL mode failed. The system sets the power LED to PWR_RAIL mode while clearing attention button events from the queue, triggered by the abnormal termination of the hotplugd daemon (for example, with the SIGKILL signal (see kill(1)).

Messages in the hotplugd Daemon Log File
The following messages are logged in the hotplugd daemon log file, defined by logfile.

ioct1(DEV_OLAR_GET_ABEVENT/WAIT) error : errno = errMsg
An error has occurred on the ioctl() call to get attention button events from the kernel in the WAIT mode. errMsg gives the reason for the error.

ioct1(DEV_OLAR_GET_ABEVENT/NOWAIT) error : errno = errMsg
An error has occurred on the ioctl() call to get attention button events from the kernel in the NOWAIT mode. errMsg gives the reason for the error.

ioct1(DEV_OLAR_GET_ABEVENT/WAIT) error : errno = errMsg sleeping for n seconds
An error has occurred on the ioctl() call to get attention button events from the kernel in the WAIT mode. errMsg gives the reason for the error. This error occurred within 1 second of the previous error. The daemon will sleep for n seconds before making another ioctl() call.

#oprnNum oprn - slot( slotId ) - Failed - retval( ret )
The specified oprn PCI OL* operation failed. oprn may be one of the following: ADD, POST_REPLACE, POWER_OFF, PRE_REPLACE.

Error opening /dev/olar in read-write mode : errno = errMsg
Could not open the device in the read-write mode.  

**ErrorMsg** gives the reason for the error.

**Error setting close-on-exec flag on /dev/olar file = errMsg**

Setting the close-on-exec flag on /dev/olar using `fcntl()` failed.  
ErrorMsg gives the reason for the error.  
At times, terminating the daemon with `kill -9` may not clear all kernel data structures.  
If you try to restart the daemon, it may report an error,  

**ioctl(DEV_OLAR_INIT_ABEVENT) Failed.**

In such circumstances, the system must be rebooted to start the daemon successfully.

**ioctl(DEV_OLAR_INIT_ABEVENT) Failed, olar_error_no = olarErrMsg**

The initialization of hotplugd for getting attention button events failed.  
olarErrMsg gives the reason for the error.

**ioctl(DEV_OLAR_INIT_ABEVENT) Failed, errno = errMsg**

The initialization of hotplugd for getting attention button events failed.  
ErrorMsg gives the reason for the error.

**Shutdown of /dev/olar for Attention Button events Failed : olar_error_no = olarErrMsg**

The shutdown of hotplugd failed.  
olarErrMsg gives the reason for the error.

**Shutdown of /dev/olar for Attention Button events Failed : errno = errMsg**

The shutdown of hotplugd failed.  
ErrorMsg gives the reason for the error.

**#oprnNum oprn - slot(slotId) - Dropping AB Event - rcvd-time(time)**

hotplugd is dropping oprn PCI OL* operation on the slotId slot because it is shutting down.  
*time* gives the time at which the attention button was pressed.

**#oprnNum oprn - slot(slotId) - time(time) - Invalid PCI OL* operation**

hotplugd has received an invalid PCI OL* operation on slot slotId.  
oprn can be one of the following:  
**INVALID, INVALID_INSERTION, INVALID_REMOVAL**  
*time* is the time at which the attention button was pressed.

This could happen if **INSERTION** or **REMOVAL** is received and the slot is not in the right state.  
For example, if **INSERTION** is received, and the slot is not suspended, and the driver is attached, then it is an invalid PCI OL* operation.  
Or, it is invalid if **REMOVAL** is received, and the slot is suspended.

This could also happen if the call to get the slot status information fails.

**#oprnNum - slot(slotId) - time(time) - Invalid PCI OL* operation**

(oprnCode)

hotplugd received a PCI OL* operation which it does not understand on slot slotId.  
oprnCode is the integer representation of the PCI OL* operation.  
*time* is the time at which the attention button was pressed.

Contact your HP response center.

**#oprnNum olarIoEvent - slot(slotId) - time(time) - Dropping AB Event,**

**received, while processing another event on the slot**

The attention button event has been dropped because it was received while another event was being processed or pending for the same slotId.  
olarIoEvent is the event that was received.  It can be one of the following:  
**BUS_MODE_MISMATCH, FREQ_MISMATCH, INSERTION POWERED,**  
**INSERTION UNPOWERED, NORMAL REMOVAL, POWER_FAULT, SURPRISE REMOVAL**  
*time* gives the time at which the event was received.

Wait for the attention button processing on a slot to complete before pressing the attention button again on that slot.

**#oprnNum - slot(slotId) - Setting power led to PWR_RAIL mode failed,**

**errno = errMsg**

The power LED could not be set to **PWR_RAIL** mode for the given slotId.  
ErrorMsg gives the reason for the error.

**#oprnNum - slot(slotId) - Setting power led to PWR_RAIL mode failed,**

**olar_error_no = olarErrMsg**
The power LED could not be set to PWR_RAIL mode for the given slotId. olarErrMsg gives the reason for the error.

Usage: hotplugd <logFileName> <append|trunc>

The hotplugd program was invoked with an invalid set of parameters.

hotplugd: fork() failed, errno = errMsg

hotplugd could not fork itself. errMsg gives the reason for the error.

hotplugd: Could not open log file: logFile, errno = errMsg

hotplugd could not open the given logFile in write mode. errMsg gives the reason for the error.

hotplugd: Could not get lock on file: logFile, errno = errMsg

hotplugd could not get the lock on the logFile. errMsg gives the reason for the error.

It could be because another instance of hotplugd is running.

FILES
/etc/rc.config.d/pci_olar
Configuration file for PCI I/O OLAR operations.

/sbin/init.d/pci_olar
Startup script, used at boot time to start the daemon.

/var/adm/hotplugd.log
Installed default name of file where the daemon logs messages. See the DIAGNOSTICS section and the Daemon Startup subsection for details.

/var/adm/syslog/syslog.log
System log file. See the DIAGNOSTICS section for details.

SEE ALSO
olrad(1M), pdweb(1M), syslogd(1M), errno(2), fcntl(2), ioctl(2)
NAME
hpx - HP-UX bootstrap

SYNOPSIS
[=i] [boot] [devicefile]
hpx ll [devicefile] (same as hpx ls -aFln)
hpx ls [-afiln] [devicefile]
hpx set autofile devicefile string
hpx show autofile [devicefile]
hpx -v
hpx restore devicefile (Workstations only; see DEPENDENCIES.)

DESCRIPTION
hpx is the HP-UX specific secondary system loader (SSL) utility for bootstrap (see isl(1M) for the
initial system loader). It supports the operations summarized below, as shown in the SYNOPSIS and
detailed later in this DESCRIPTION.

This hpx(1M) manpage only documents features on PA-RISC systems. For bootstrap information on
Itanium(R)-based systems, see hpx.efi(1M).

Operations
The following operations are supported on PA-RISC systems:

boot Loads an object file from an HP-UX file system or raw device and transfers con-
trol to the loaded image. (Note, the boot operation is position dependent).
ll Lists the contents of HP-UX directories in a format similar to ls -aFln. (See
ls(1); ls only works on a local disk with a HFS file system).
ls Lists the contents of HP-UX directories. (See ls(1); ls only works on a local disk
with a HFS file system).
show autofile Displays the contents of the autoexecute file.
set autofile Changes the contents of the autoexecute file to that specified by string.
-v Displays the release and version numbers of the hpx utility.
restore Recovers the system from a properly formatted bootable tape. (Workstation
specific; see DEPENDENCIES.)

hpx commands can be given interactively from the keyboard, or provided in an isl autoexecute file.
hpx is limited to operations on the interface initialized by pdc(1M). In most cases, operations are limited
to the boot device interface.

Notation
hpx accepts numbers (numeric constants) in many of its options. Numbers follow the C language nota-
tion for decimal, octal, and hexadecimal constants. A leading 0 (zero) implies octal and a leading 0x or 0X
implies hexadecimal. For example, 037, 0x1F, 0X1f, and 31 all represent the same number, decimal 31.

hpx boot, ll, ls, set autofile, show autofile, and restore operations accept devicefile
specifications, which have the following format:
manager (w/x/y/z;n) filename

The devicefile specification is comprised of a device name and a file name. The device name
(manager (w/x/y/z;n)), consists of a generic name of an I/O system manager (device or interface driver)
such as disc, a hardware path to the device, and minor number. The manager name can be omitted
entirely if the default is used. w/x/y/z is the physical hardware path to the device, identifying bus con-
verters, slot numbers, and hardware addresses. For workstations, there are a set of mnemonics that can be
used instead of the hardware paths. The n is the minor number that controls manager-dependent func-
tionality. The file name part, filename, is a standard HP-UX path name. Some hpx operations have
defaults for particular components. A devicefile specification containing a device part only specifies a raw
device. A devicefile specification containing a file name implies that the device contains an HP-UX file sys-
tem, and that the filename resides in that file system.
A typical boot `devicefile` specification is

```
disc(2/4.0.0;0)/stand/vmunix
```

The manager is `disc`, the hardware path to the disk device is `2/4.0.0`, the minor number shown as `0` by default, and the `/stand/vmunix` is the filename for the boot device.

`hpux` now supports a consolidated list of managers: `disc`, `tape`, and `lan`. The manager `disc` manages all disks connected via SCSI, (formerly `disc3`), and all autochanger disk devices (formerly `disc30`). The manager `lan` manages remote boot through the HP 28652A NIO based LAN interface (formerly `lan1`). Remote boot is currently supported on this card only and not on any CIO-based LAN card. The manager `tape` manages tape drives via SCSI (formerly `tape2`).

The hardware path in a `devicefile` specification is a string of numbers, each suffixed by slash, (`/`), followed by a string of numbers separated by dots (`.`), each number identifying a hardware component notated sequentially from the bus address to the device address. A hardware component suffixed by a slash indicates a bus converter and may not be necessary on your machine. For example, in `w/x/y/z` `w` is the address of the bus converter, `x` is the address of the MID-BUS module, `y` is the CIO slot number, and `z` is the HP 27111 bus address.

The minor number, `n`, in a `devicefile` specification controls driver-dependent functionality. (See the manual, Configuring HP-UX for Peripherals, for minor-number bit assignments of specific drivers).

File names are standard HP-UX path names. No preceding slash (`/`) is necessary and specifying one will not cause problems.

### Defaults

Default values chosen by `hpux` to complete a command are obtained through a sequence of steps. First, any components of the command specified explicitly are used. If the command is not complete, `hpux` attempts to construct defaults from information maintained by `pdc` (see `pdc(1M)`). If sufficient information to complete the command is unavailable, the `autoexecute` file is searched. If the search fails, any remaining unresolved components of the command are satisfied by hard-coded defaults.

There is no hard-coded default choice for a `manager`; if none can be chosen, `hpux` reports an error.

When the hardware path to the boot device is not specified, `hpux` defaults to information maintained by `pdc`. The hardware path element has no hard-coded default.

If the minor number element is not supplied, `hpux` takes its default from the `autoexecute` file. Failing that, the hard-coded default of `0` is used.

For the `boot` command, a `devicefile` specification without a file name indicates that the boot device does not contain an HP-UX file system. `hpux` interprets this as a NULL (instead of missing) file name and does not search for a default. If the entire `devicefile` specification is missing, `hpux` searches for a default; either the `autoexecute` file contents or the hard-coded default is chosen.

There are two possible hard-coded default `devicefile` specifications. One hard-coded default `devicefile` specification is `/vmunix`. The other hard-coded default `devicefile` specification is `/stand/vmunix`. If you have a LVM or VxVM system where the boot volume and the root volume are on different logical volumes, the kernel would be `/vmunix`. This is because the boot volume will be mounted under `/stand` when the system is up.

For all other configurations, the kernel would be `/stand/vmunix`. The search order for the hard-coded defaults is `/stand/vmunix` and then `/vmunix`.

### boot Operation

The `boot` operation loads an object file from an HP-UX file system or raw device as specified by the optional `devicefile`. It then transfers control to the loaded image.

Any missing components in a specified `devicefile` are supplied with a default. For example, a `devicefile` of `vmunix` would actually yield:

```
disc(8.0.0;0)vmunix
```

and a `devicefile` of `(8/0/19/0.14.0)/stand/vmunix`, for booting from the disk at Ultra Wide SCSI address 14, would yield

```
disc(8/0/19/0.14.0;0)/stand/vmunix
```
To boot a saved kernel configuration, specify a devicefile of /stand/configname/vmunix, where configname is the name of the saved configuration to boot. For more details on saved kernel configurations, see kconfig(5).

Regardless of how incomplete the specified devicefile may be, boot announces the complete devicefile specification used to find the object file. Along with this information, boot gives the sizes of the TEXT, DATA, and BSS, segments and the entry offset of the loaded image, before transferring control to it.

The boot operation accepts several options. Note that boot options must be specified positionally as shown in the syntax statement in the SYNOPSIS. Options for the boot operations are as follows:

- `a[C|R|S|D] devicefile` Accept a new location (as specified by devicefile) and pass it to the loaded image. If that image is an HP-UX kernel, the kernel will erase its predefined I/O configuration, and configure in the specified devicefile. If the C, R, S, or D option is specified, the kernel configures the devicefile as the console, root, swap, or dump device, respectively. Note that -a can be repeated multiple times.

- `fnumber` Use the number and pass it as the flags word to the loaded image.

- `i]string` Set the initial run-level for init (see init(1M)) when booting the system. The run-level specified will override any run-level specified in an initdefault entry in /etc/inittab (see inittab(4)).

- `lm` Boot the system in LVM maintenance mode, configure only the root volume, and then initiate single user mode.

- `vm` Boot the system in VxVM maintenance mode, configure only the root volume, and then initiate single user mode.

- `tm` Boot the system in tunable maintenance mode, also known as "failsafe boot" mode. This option will disregard the tunable settings and module settings in the kernel configuration, and boot with known good settings instead. Note: some systems that have been updated from earlier versions of HP-UX have boot loaders that do not support this flag. On those systems, the flag -fo00000 can be used instead.

- `lq` Boot the system with quorum override option. This option is used in a scenario where a disk is removed from the system or is otherwise unavailable, but the corresponding entry for the physical volume has not yet been removed from the volume group using vgreduce.

- `F` Used with SwitchOver/UX software. However, SwitchOver/UX is not supported on HP-UX 10.30 or later systems. The -F option is used to ignore any locks on the boot disk. The -F option should be used only when it is known that the processor holding the lock is no longer running. (If this option is not specified and a disk is locked by another processor, the kernel will not boot from it, to avoid the corruption that would result if the other processor were still using the disk).

boot places some restrictions on object files it can load. It accepts only the HP-UX magic numbers EXEC-MAGIC (0407), SHAREMAGIC (0410), and DEMANDMAGIC (0413). See magic(4). The object file must contain an Auxiliary Header of the HPUX_AUX_ID type and it must be the first Auxiliary Header (see a.out(4)).

ll and ls Operations
The ll and ls operations list the contents of the HP-UX directory specified by the optional devicefile. The output is similar to that of ls -aFl command, except the date information is not printed.

The default devicefile is generated just as for boot, defaulting to the current directory.

set autofile Operation
The set autofile operation overwrites the contents of the autoexecute file, autofile, with the string specified (see autoexecute in the EXAMPLES section).

show autofile Operation
The show autofile operation displays the contents of the autoexecute file, autofile (see autoexecute in the EXAMPLES section).
**hpux(1M)**

**(PA-RISC Systems Only)**

**DIAGNOSTICS**

If an error is encountered, **hpux** prints diagnostic messages to indicate the cause of the error. These messages fall into the General, Boot, Copy, Configuration, and System Call categories. System Call error messages are described in **errno(2)**. The remaining messages are listed below.

**General**

- **bad minor number in devicefile spec**
  The minor number in the **devicefile** specification is not recognized.

- **bad path in devicefile spec**
  The hardware path in the **devicefile** specification is not recognized.

- **command too complex for parsing**
  The command line contains too many arguments.

- **no path in devicefile spec**
  The **devicefile** specification requires (but does not contain) a hardware path component.

- **panic (in hpuxboot): (display==number, flags==number) string**
  A severe internal **hpux** error has occurred. Report to your nearest HP Field Representative.

**Boot**

- **bad magic**
  The specified object file does not have a recognizable magic number.

- **bad number in flags spec**
  The flags specification in the **-f** option is not recognized.

- **Exec failed: Cannot find /stand/vmunix or /vmunix.**
  Neither /stand/vmunix or /vmunix could be found.

- **booting from raw character device**
  In booting from a raw device, the **manager** specified only has a character interface, which might cause problems if the block size is incorrect.

- **isl not present, please hit system RESET button to continue**
  An unsuccessful **boot** operation has overlaid **isl** in memory. It is impossible to return control to **isl**.

- **short read**
  The specified object file is internally inconsistent; it is not long enough.

- **would overlay**
  Loading the specified object file would overlay **hpux**.

**Configuration**

- **cannot add path, error number**
  An unknown error has occurred in adding the hardware path to the I/O tree. The internal error number is given. Contact your HP Field Representative.

- **driver does not exist**
  The manager specified is not configured into **hpux**.

- **driver is not a logical device manager**
  The **manager** named is not that of a logical device manager and cannot be used for direct I/O operations.

- **error rewinding device**
  An error was encountered attempting to rewind a device.

- **error skipping file**
  An error was encountered attempting to forward-space a tape device.

- **negative skip count**
  The skip count, if specified, must be greater than or equal to zero.

- **no major number**
  The specified **manager** has no entry in the block or character device switch tables.

- **path incompatible with another path**
  Multiple incompatible hardware paths have been specified.
path long
The hardware path specified contains too many components for the specified manager.

path short
The hardware path specified contains too few components for the specified manager.

table full
Too many devices have been specified to hpux.

EXAMPLES
As a preface to the examples which follow, here is a brief overview of HP-UX system boot-up sequences.

Automatic Boot
Automatic boot processes on various HP-UX systems follow similar general sequences. When power is applied to the HP-UX system processor, or the system Reset button is pressed, processor-dependent code (firmware) is executed to verify hardware and general system integrity (see pdc(1M)). After checking the hardware, pdc gives the user the option to override the autoboot sequence by pressing the Esc key. At that point, a message resembling the following usually appears on the console.

(c) Copyright. Hewlett-Packard Company. 1994.
All rights reserved.
PDC ROM rev. 130.0
32 MB of memory configured and tested.
Selecting a system to boot.
To stop selection process, press and hold the ESCAPE key...

If no keyboard activity is detected, pdc commences the autoboot sequence by loading isl (see isl(1M)) and transferring control to it. Since an autoboot sequence is occurring, isl finds and executes the autoexecute file which, on an HP-UX system, requests that hpux be run with appropriate arguments. Messages similar to the following are displayed by isl on the console:

Booting from: scsi.6 HP 2213A
Hard booted.
ISL Revision A.00.09 March 27, 1990
ISL booting hpux bootdisk(;0)/stand/vmunix

hpux, the secondary system loader, then announces the operation it is performing, in this case boot, the devicefile from which the load image comes, and the TEXT size, DATA size, BSS size, and start address of the load image, as shown below, before control is passed to the image.

Booting disk(scsi.6;0)/stand/vmunix
966616+397312+409688 start 0x6c50

The loaded image then displays numerous configuration and status messages.

Interactive Boot
To use hpux interactively, isl must be brought up in interactive mode by pressing the Esc key during the interval allowed by pdc. pdc then searches for and displays all bootable devices and presents a set of boot options. If the appropriate option is chosen, pdc loads isl and isl interactively prompts for commands. Information similar to the following is displayed:

Selection process stopped.
Searching for Potential Boot Devices.
To terminate search, press and hold the ESCAPE key.

Device Selection Device Path Device Type
-----------------------------------------------
P0 scsi.6.0 QUANTUM PD210S
P1 scsi.1.0 HP 2213A
p2 lan.ffffff-ffffff.f.f hpfoobar

b) Boot from specified device
s) Search for bootable devices
a) Enter Boot Administration mode
x) Exit and continue boot sequence
Although all of the operations and options of `hpux` can be used from `isl` interactively, they can also be executed from an `autoexecute` file. In the examples below, user input is the remainder of the line after each `ISL>` prompt shown. The remainder of each example is text displayed by the system. Before going over specific examples of the various options and operations of `hpux`, here is an outline of the steps taken in the automatic boot process. Although the hardware configuration and boot paths shown are for a single server machine, the user interfaces are consistent across all models. When the system `Reset` button is depressed, `pdc` executes self-test, and assuming the hardware tests pass, `pdc` announces itself, sends a BELL character to the controlling terminal, and gives the user 10 seconds to override the `autoboot` sequence by entering any character. Text resembling the following is displayed on the console:

```
Processor Dependent Code (PDC) revision 1.2
Duplex Console IO Dependent Code (IODC) revision 3
Console path = 56.0.0.0.0.0.0 (dec)
38.0.0.0.0.0.0 (hex)
Primary boot path = 44.3.0.0.0.0.0 (dec)
2c.00000003.0.0.0.0.0 (hex)
Alternate boot path = 52.0.0.0.0.0.0 (dec)
34.0.0.0.0.0.0 (hex)
32 MB of memory configured and tested.
Autosearch for boot path enabled
To override, press any key within 10 seconds.
```

If no keyboard character is pressed within 10 seconds, `pdc` commences the `autoboot` sequence by loading `isl` and transferring control to it. Because an `autoboot` sequence is occurring, `isl` merely announces itself, finds and executes the `autoexecute` file which, on an HP-UX system, requests that `hpux` be run with appropriate arguments. The following is displayed on the console.

```
10 seconds expired.
Proceeding with autoboot.
```

```
Trying Primary Boot Path
------------------------
Booting...
```

```
Boot IO Dependent Code (IODC) revision 2
HARD Booted.
```

```
ISL Revision A.00.2G Mar 20, 1994
ISL booting hpux
```

`hpux` then announces the operation it is performing, in this case `boot`, the `devicefile` from which the load image comes, and the `TEXT` size, `DATA` size, `BSS` size, and start address of the load image. The following is displayed before control is passed to the image.

```
Boot:
: disc3(44.3.0;0)/stand/vmunix
3288076 + 323584 + 405312 start 0x11f3e8
```

Finally, the loaded image displays numerous configuration and status messages, then proceeds to `init run-level 2` for multiuser mode of operation.

`isl` must be brought up in interactive mode to use the operations and options of `hpux`. To do this, simply enter a character during the 10 second interval allowed by `pdc`. `pdc` then asks if the primary boot path is acceptable. Answering yes (Y) is usually appropriate. `pdc` then loads `isl` and `isl` interactively prompts for commands. The following lines show the boot prompt, the Y response, subsequent boot messages, and
finally the Initial System Loader (ISL) prompt that are sent to the display terminal:

```
Boot from primary boot path (Y or N)?> y
Interact with IPL (Y or N)?> y
Booting...
Boot IO Dependent Code (IODC) revision 2
HARD Booted.
ISL Revision A.00.2G Mar 20, 1994
ISL>
```

Although all of the operations and options of hpux can be used from isl interactively, they can also be executed from an autoexecute file. In the examples below, all user input follows the ISL> prompt on the same line. Subsequent text is resultant messages from the ISL.

**Default Boot**

Entering hpux initiates the default boot sequence. The boot path read from pdc is 8.0.0, the manager associated with the device at that path is disc, the minor number, in this case derived from the autoexecute file, is 4 specifying section 4 of the disk, and the object file name is /stand/vmunix.

```
ISL> hpux
Boot : disc3(44.3.0;0)/stand/vmunix
3288076 + 323584 + 405312 start 0x11f3e8
```

**Booting Another Kernel Configuration**

In this example, hpux initiates a boot operation for the saved kernel configuration myconfig.

```
ISL> hpux myconfig/vmunix
Boot : disc3(44.3.0;0)/stand/myconfig/vmunix
3288076 + 323584 + 405312 start 0x11f3e8
```

**Booting From Another Section**

In this example (shown for backward compatibility), a kernel is booted from another section of the root disk. For example, suppose kernel development takes place under /mnt/azure/root.port which happens to reside in its own section, section 3 of the root disk. By specifying a minor number of 3 in the above example, the object file sys.azure/S800/vmunix is loaded from /mnt/azure/root.port.

```
ISL> hpux (;3)sys.azure/S800/vmunix
Boot : disc(8.0.0;0x3)sys.azure/S800/vmunix
966616+397312+409688 start 0x6c50
```

**Booting From Another Disk**

Only the hardware path and file name are specified in this example. All other values are boot defaults. The object file comes from the file system on another disk.

```
ISL> hpux (52.5.0.0)/stand/vmunix
Boot : disc(52.5.0.0)/stand/vmunix
966616+397312+409688 start 0x6c50
```

**Booting From LAN**

This example shows how to boot a cluster client from the LAN. Though this example specifies a devicefile, you can also use default boot, as shown in a previous example. For a boot operation other than default boot, the file name must be specified and can be no longer than 11 characters. Booting to isl from a local disk then requesting an image to be loaded from the LAN is not supported.

```
ISL> hpux lan(32)/stand/vmunix
Boot : lan(32;0x0)/stand/vmunix
```
Booting To Single User Mode
In this example, the `-i` option is used to make the system come up in run-level `s`, for single user mode of operation.

```
ISL> hpux -is
Boot
: disc(8.0.0;0x0)/stand/vmunix
966616+397312+409688 start 0x6c50

(INIT: Overriding default level with level 's'
INIT: SINGLE USER MODE
WARNING: YOU ARE SUPERUSER !!)
```

Booting With A Modified I/O Configuration
Here, a disc driver is configured in at CIO slot 5, SCSI address 5 as a dump device. Regardless of what was present in the kernel's original I/O configuration, the driver `disc` is now configured at that hardware path. Similarly, `asio0` is configured in at CIO slot 63 which is to be the console. The only other devices configured are the console and root device, which `boot` derived from `pdc`.

```
ISL> hpux -aC asio0(8/0/63) -aD disc(8/16/5.5)
Boot
: disk(8/0/19/0.14.0.0.0.0.0;0)/stand/vmunix
: Adding console (8.0.63;0)...
: Adding dump (8.16.5.5;0)...
6463488 + 1101824 + 939616 start 0x39168

(Additional Kernel Startup Messages Omitted)
```

Displaying The Autoexecute File
In this example, `show autofile` is used to print the contents of the `autoexecute` file residing in the boot LIF, on the device from which `hpux` was booted. Optionally, a `devicefile` can be specified in order to read the `autoexecute` file from the boot LIF of another boot device.

```
ISL> hpux show autofile
Show autofile
: AUTO file contains (hpux)
```

Changing The Autoexecute File
This example shows how to change the contents of the `autoexecute` file. Once done, the system can be reset, and the new command will be used during any unattended boot.

```
ISL> hpux set autofile "hpux /stand/myconfig/vmunix"
Set autofile
: AUTO file now contains "(hpux /stand/myconfig/vmunix)"
```

Listing Directory Contents
The contents of the directory `/stand` on the root disk are listed. The format shows the file protections, number of links, user id, group id, and size in bytes for each file in the directory. There are three available kernel configurations to boot: the default configuration (`vmunix`), the automatic backup configuration (`backup`), and one other saved configuration (`good`). Listing the files over the LAN is not supported.

```
ISL> hpux ll /stand
Ls
: disk(2/0/1.3.0.0.0.0.0.0;0)/stand
dr-xr-xr-x 9 2 2 1024 ./
dr-xr-xr-x 14 2 2 1024 ../
drwxr-xr-x 5 0 3 1024 backup/
-rw-r--r-- 1 0 3 19 bootconf
```

HP-UX 11i Version 2: December 2007 Update
Getting The Version

The -v option is used to get the version numbers of hpux.

```
ISL> hpux -v
Release: 10.00
```

DEPENDENCIES

Workstations Only

The restore operation is provided as a recovery mechanism in the event that a disk becomes totally corrupted. It copies data from a properly formatted bootable tape to disk. When this tape contains a backup image of the disk, the entire disk is restored. To create a properly formatted tape (DDS ONLY), the following commands should be executed:

```
dd if=/usr/lib/uxbootlf of=/dev/rmt/0mn bs=2k
dd if=/dev/rdsk/1ss of=/dev/rmt/0m bs=64k
```

The first dd puts a boot area on the tape, making it a bootable image (see dd(1)). Once the boot image is on tape, the tape is not rewound. The next dd appends an image of the disk to the tape. The entire process takes about one hour for a 660 MB HP 2213 disk. To avoid later problems with fsck after the disk is restored, bring the system to single user mode and type sync a few times before doing the second dd (see fsck(1M)). Once created, the tape can be used to completely restore the disk:

1. Insert the tape into the tape drive.
2. Instruct the machine to boot to ISL from the tape. This is usually done by specifying scsi.3 as the boot path.
3. Enter the following in response to the ISL prompt:

```
ISL> hpux restore disk(scsi.1;0)
```

This restores the disk image from the tape to the actual disk at scsi.1. Any existing data on the disk will be lost. This command destroys the contents of the device specified by devicefile. The restoration process takes about one hour for a 660 MB drive.

NOTE: There is a 2 GB limit on the amount of data that can be restored. The tape and disk must be on the boot device interface.

Also, this command may be replaced in the future by superior installation and recovery mechanisms. At that time, this command will be removed.

SEE ALSO

boot(1M), fsck(1M), hpux.efi(1M), init(1M), isl(1M), pdc(1M), errno(2), a.out(4), inittab(4), magic(4).
NAME
hpux.efi - HP-UX bootstrap for Itanium-based systems

SYNOPSIS
hpux.efi [-V] [-tm] [-vm]

DESCRIPTION
hpux.efi is the HP-UX-specific operating system loader utility for bootstrap. It is a native efi(4) application that can be run on the Itanium-based platform.

hpux.efi supports the following options:

- **-V**  Display the release and version numbers of the hpux utility.
- **-tm**  Boot the system in tunable maintenance mode, also known as "failsafe boot" mode. This option will disregard the tunable settings and module settings in the kernel configuration, and boot with known good settings instead.
- **-vm**  Boot the system in VxVM maintenance mode, configure only the root volume, and then enter single user mode.

The following commands can be given interactively from the keyboard or provided in an efi(4) AUTO file.

**boot [file]**  Load a kernel object file from an HP-UX file system and transfer control to the loaded image. (Note, the boot operation is position-dependent). Currently, hpux.efi is limited to disk for booting. file can be the name of a kernel object file or the name of a kernel configuration directory.

The following commands can be given interactively from the keyboard only after hpux is launched.

**ll**  List the contents of HP-UX directories in detail.

**ls**  List the contents of HP-UX directories.

**showauto**  Display the contents of the AUTO file.

**setauto string**  Change the contents of the AUTO file to string.

**boot Operation**
The boot operation loads an ELF kernel object file from supported HP-UX file system (HFS, VxFS). It then transfers control to the loaded image.

In addition to the kernel object file, hpux.efi also loads persistent system-specific files from the boot.sys directory and kernel configuration-specific files from the configname/bootfs directory during boot operation.

hpux.efi always assumes /stand (the boot directory) as the current directory. For example, myconfig/vmunix would actually yield:

/stand/myconfig/vmunix

Along with this information, boot gives the sizes of the TEXT, DATA, and BSS segments before transferring control to it.

The boot operation accepts several options. Note that boot options must be specified positionally as shown in the syntax statement in the SYNOPSIS. Options for the boot operations are as follows:

**-irun-level**  Set the initial run-level for init when booting the system. This run-level overrides any run-level specified in an initdefault entry in /etc/inittab. See init(1M) and inittab(4).

boot can load only ELF object files.

**ll and ls Operations**
The ll and ls operations list the contents of the HP-UX directory. In addition to filenames, ll displays size and date information when used on supported HP-UX filesystems.

For VxVM disks, only boot volume (/stand) information can be displayed. To display boot volume information, either a relative path to /stand or an absolute path can be used.
setauto Operation
The setauto operation overwrites the contents of the autoexec file, AUTO, with the string specified.

showauto Operation
The showauto operation displays the contents of the autoexec file, AUTO. See autoexecute in the EXAMPLES section.

EXAMPLES
Automatic Boot
This is an overview of HP-UX automatic boot processes on Itanium-based systems. When power is applied to the HP-UX system processor, or the system Reset button is pressed, Itanium-based system firmware code is executed to verify hardware and general system integrity (see boot(1M)). After initializing the hardware, the EFI boot_manager is launched. The EFI boot_manager gives the user the option to automatically start any EFI application. After hpux.efi is launched by the EFI boot_manager, it looks for an AUTO file and uses its content for the boot string. Then it gives the user the option to override the autoboot sequence by pressing any key:

*** HP-UX Boot Loader for IA-64 ***
*** Version #.# ***
*** Type 'help' for help ***
Press Any Key to interrupt Autoboot
AUTO ==> boot vmunix -is
Seconds left till autoboot - 10

If hpux.efi does not detect a key press, it initiates the autoboot sequence using the AUTO string. hpux.efi displays messages similar to the following on the console:

AUTOBOOTING...
AUTO BOOT> boot vmunix -is

If hpux.efi successfully locates the kernel object file, it displays the progress of loading each sections and the symbol table of the object file.

loading section 0
....................
loading section 1
.....
Loading symbol table. Num of Sec Header(79)

Then it displays the progress of loading system- and kernel configuration-specific directories.

Loading System Directory(boot.sys) to MFS
....
Loading Kernel Boot Directory(bootfs) to MFS
....

Finally, boot displays the complete path to the kernel object file loaded and the TEXT size, DATA size, and BSS size of the load image before control is passed to the image.

Launching /stand/vmunix...

The loaded image then displays numerous configuration and status messages.

Interactive Boot
To use hpux.efi interactively, the user must interrupt the autoboot sequence by pressing a key or removing the AUTO file. Then hpux.efi interactively prompts for commands.

HPUX>

SEE ALSO
boot(1M), hpux(1M), init(1M), inittab(4), efi(4).
identd(1M)

NAME
identd - TCP/IP IDENT protocol server

SYNOPSIS
/usr/lbin/identd [-i | -w | -b] [-tseconds] [-u|uid] [-g|gid] [-p|port] [-a|address]

DESCRIPTION
identd is a server which implements the TCP/IP proposed standard IDENT user identification protocol as
specified in the RFC 1413 document.

identd operates by looking up specific TCP/IP connections and returning the user name of the process
owning the connection.

Options
The identd protocol server recognizes the following options:

- `a|address` Specify the local address to bind the socket to if using the `-b` mode of operation. Can only
  be specified by the IP address and not by the domain name. The default value in IPv4 is
  INADDR_ANY, and in IPv6 is in6addr_any, which normally represents all the local
  addresses.

- `b` Run in standalone mode without assistance from inetd. This mode is the least preferred
  mode and not supported by HP. A bug or any other fatal condition in the server will make
  the server terminate, and it will then have to be restarted manually.

- `c|charset` Add the optional (according to the IDENT protocol) character set designator to the reply
  generated. charset should be a valid character set as described in the MIME RFC in upper-
  case characters.

- `d` Enable some debugging code that normally should NOT be enabled since it breaks the pro-
  tocol and may reveal information that should not be available to outsiders.

- `e` Always return UNKNOWN-ERROR instead of the NO-USER or INVALID-PORT errors.

- `g|gid` Specify a group id number which the ident server should switch to after binding itself to
  the TCP/IP port if using the `-b` mode of operation.

- `i` Default mode. This mode should be used when starting the daemon from inetd with the
  “nowait” option in the /etc/inetd.conf file. Use of this mode will make inetd start
  one identd daemon for each connection request.

- `l` Use the system logging daemon syslogd for logging purposes.

- `m` Use a mode of operation that allows multiple requests to be processed per session. Each
  request is specified one per line and the responses will be returned one per line. The con-
  nection will not be closed until the connecting part closes its end of the line. Please note
  that this mode violates the protocol specification as it currently stands.

- `n` Always return user numbers instead of user names in order to keep the user names a
  secret.

- `N` Check for a file .noident in each home directory for a user which the daemon is about to
  return the user name for. If that file exists then the daemon will give the error HIDDEN-
  USER instead of the normal USERID response.

- `o` Do not reveal the operating system type it is run on and always return OTHER instead.

- `p|port` Specify an alternative port number to bind to if using the `-b` mode of operation. The port
  can be specified by name or by number. It defaults to the IDENT port (113).

- `tseconds` Used to specify the timeout limit. This is the number of seconds a server started with the
  `-w` flag will wait for new connections before terminating. The server is automatically res-
  tartaed by inetd whenever a new connection is requested if it has terminated. A suitable
  value for seconds is 120 (two minutes), if used. It defaults to no timeout. That is, it will
  wait forever, or until a fatal condition occurs in the server.
identd(1M)

-seconds
Specifies the maximum number of seconds a server will wait for the client's input before closing the connection on timeout. By default, the timeout limit is 0 (zero); that is, the server waits for an indefinite amount of time. This option is ignored for invalid timeout intervals.

-usid
Specify a user id number which the ident server should switch to after binding itself to the TCP/IP port if using the -b mode of operation.

-v
Display the version number and exit.

-w
Use this option when starting the daemon from inetd with the "wait" option in the /etc/inetd.conf file. The identd daemon will either run forever or until a timeout, as specified by the -t flag, occurs.

INSTALLATION
identd is invoked either by the internet server (see inetd(1M)) for requests to connect to the IDENT port as indicated by the /etc/services file (see services(4)) when using the -w or -i modes of operation or started manually by using the -b mode of operation.

WARNINGS
The -w and -t options are currently not supported on HP-UX.

EXAMPLES
Assume the server is located in /usr/lbin/identd. Add either of the following two lines into the /etc/inetd.conf file:

ident stream tcp6 wait bin /usr/lbin/identd identd -w -t20

or:

ident stream tcp6 nowait bin /usr/lbin/identd identd -i

To start identd using the unsupported -b mode of operation, add the following line to the /sbin/init.d/sendmail file under the start section:

/usr/lbin/identd -b -u2 -g2

This will cause identd to be started as daemon whenever sendmail is running. It will run in the background as user 2, group 2 (user bin, group bin).

SEE ALSO
inetd.conf(4).
NAME
idisk - create partitions for Itanium(R)-based system's disks

SYNOPSIS
idisk [-p | -a | -l | -b | -q | -v | -r | -R | -w | [-f { - | partition_description_file}]] device

DESCRIPTION
idisk creates operating system partitions on a disk that is to be used for Itanium-based systems. It reads
in the partition information from a data file that may be specified in the command string or redirected from
stdin. By default, idisk operates in read-only mode and displays the partition information that is
currently on the disk. To write new partition information on the disk the user must specify the -w option.

Options
idisk recognizes the following options:

- p Print the primary EFI partition header and partition tables.
- a Print the alternate EFI partition header and partition tables.
- l Print the legacy partition table that resides in the master boot record. idisk writes partition
  information for the first four partition in the partition table in used by legacy DOS and Windows.
  This information is used as a backup in the event all the EFI information is corrupted.
- b Print the first usable and last usable block numbers that are available to create partitions. First
  usable is the first block a partition can start on. Last usable is the last block that can be con-
tained in a partition. These numbers are relative to the whole disk and do not take into account
any partitions that may exist. They represent the total disk space that can be partitioned. Use
the -q option to only print the values without headings.
- q Work silently. No user prompts or warnings. For use in shell scripts.
- v Validate EFI partition information. Does the same checks the driver does verifying that both the
  primary and alternate EFI partition headers and tables are correct. Returns two if either is bad
  and zero if both are correct.
- r Restore the EFI partition headers and tables. This option checks both the primary header and
  tables and the alternate header and tables. If one is found bad it is restored from the other good
  version. One of either the primary or alternate header and tables must be good for this option to
  succeed. The -w option must be specified for information to be written to the disk.
- R Remove all EFI partition headers and tables from the disk. This option also destroys the informa-
tion contained in the MBR (master boot record). The -w option must be specified for infor-
mation to be written to the disk.
- f { - | partition_description_file}
The partition_description_file contains the number of partitions to be created and the type and
requested size of each partition. The filename may be specified here or redirected from stdin
when the "-" is used.
- w Enable write mode. By default idisk operates in read-only mode. To create and write parti-
tion information to the disk you must specify the -w option.

Partition Description File
The first entry in the partition description file is the number of partitions to create. This is followed by a
line containing the type and size for each of the partitions. Recognized partition types are: EFI, HPUX,
and HPDUMP. Size may be specified in megabytes or as a percentage of the whole disk. Internally,
idisk creates the partitions whose size is specified in MB first then creates those whose size was specified
as a percentage. Those partitions specified as a percent are assigned space from what is available after the
MB partitions are created. If the size of a partition is specified as 100% then all space remaining is
assigned to that partition.

An example partition description file is shown below:

2
  EFI 100MB
  HPUX 100%
idisk(1M)  idisk(1M)

(Itanium(R)-based Processor Family Only)

The first entry specifies the number of partitions to create. The second specifies an EFI partition of 100 megabytes. The last entry specifies a HPUX partition consisting of all the remaining space on the disk after the EFI partition has been created.

When creating partitions, the device file name must be that of the whole disk and must not have any partition number bits set in the minor number. For disk devices, the last eight bits of the minor number represent the option bits. For Itanium-based system's disks, the last four option bits are used to indicate the partition number. Since there are only four bits for partition number, only one to fifteen partitions are supported. For example, a device node with a minor number of 0x008001 would indicate a disk at target eight, partition number one. A minor number of 0x00500F would indicate a disk at target 5 partition fifteen. A minor number with no partition bits set would indicate the whole disk (ie 0x008000 would be the same disk as above but represent the whole disk and not a partition).

Note
idisk has been ported to Windows NT 4.0 and 2000.

RETURN VALUE
Exit values are:

  0  Successful completion.
 >0  Error condition occurred.

EXAMPLES
Create the partitions specified in the above description file, printing only the primary partition information:

  idisk -w -p -f ia64dsk.dat /dev/rdsk/c1t4d0

Create the partitions specified in the above description file, printing all available information (default), redirecting input from stdin:

  idisk -w -f - /dev/rdsk/c1t4d0 < ia64dsk.dat

Only read the disk, printing all tables (default) on the disk:

  idisk /dev/rdsk/c1t4d0

Get the first and last usable block for partitioning on the disk:

  idisk -b /dev/rdsk/c1t4d0

Destroy all partition information on the disk:

  idisk -R /dev/rdsk/c1t4d0

Restore partition information from either a good primary or alternate header or table to the header or table that is bad:

  idisk -w -r /dev/rdsk/c1t4d0

AUTHOR
idisk was developed by HP.

SEE ALSO
efi(4).
NAME
ifconfig - configure network interface parameters

SYNOPSIS
ifconfig [-m mod1[, mod2][, ...]] interface [address_family] [address[dest_address]] [parameters]
ifconfig interface [address_family]

DESCRIPTION
The first form of the ifconfig command assigns an address to a network interface and/or configures network interface parameters. ifconfig must be used at boot time to define the network address of each interface present on a machine. It can also be used at other times to redefine an interface's address or other operating parameters. If the address_family is not specified, the address family defaults to IPv4.

The second form of the command, without address_family, displays the current configuration for interface.

If address_family is not specified, ifconfig reports the details on all supported address families. An exception is when the user has not configured any interface with an IPv6 address, ifconfig does not display the IPv6 loopback interface.

Only a user with appropriate privileges can modify the configuration of a network interface. All users can run the second form of the command.

Arguments
ifconfig recognizes the following arguments:

-m mod1[, mod2][, ...]
A list of modules that can be pushed on a stream associated with an interface. The -m option can be used to configure an interface manually with the specified module names. There is no space between the module names and only a comma is used to separate the module names. If the -m option is specified, ifconfig pushes all modules on the stream associated with the interface in the specified order. For example, module mod2 is pushed on top of module mod1. The modules are pushed between IP and network drivers. If the -m option is not specified, the modules (if any) specified in the DEFAULT_INTERFACE_MODULES variable are used for configuring the interface (see netconf and netconf-ipv6 in /etc/rc.config.d/).

address
Either a host name present in the host name database (see hosts(4)), or a DARPA Internet address expressed in Internet standard dot notation (see inet(3N)) for an IPv4 address and in colon notation (see inet6(3N)) for an IPv6 address.

address_family
Name of protocol on which naming scheme is based. An interface can receive transmissions in differing protocols, each of which may require separate naming schemes. The address_family, affects the interpretation of the remaining parameters on the command line. The only address families currently supported are inet (DARPA-Internet family) for IPv4 addresses, and inet6 for IPv6 addresses.

dest_address
Address of destination system. Consists of either a host name present in the host name database (see hosts(4)), or a DARPA Internet address expressed in Internet standard dot notation (see inet(3N)) for an IPv4 address, and in colon notation (see inet6(3N)) for an IPv6 address.

interface
A string of the form name unit, such as lan0. (See the Interface Naming subsection given below.)

parameters
One or more of the following operating parameters:

up
Mark an interface "up". Enables interface after an ifconfig down. Occurs automatically when setting the address on an interface. Setting this flag has no effect if the hardware is "down". A secondary interface (see the Interface Naming subsection given below) can be marked up only if the primary interface is already up.

down
Mark an interface "down". When an interface is marked "down", the system will not attempt to transmit messages through that interface. A primary interface (see the Interface Naming subsection given below) can be marked down only if all the secondary interfaces on the same physical device are already down.
broadcast  
(inet only) Specify the address that represents broadcasts to the network. The default broadcast address is the address with a host part of all 1's.

encaplimit  
Specify the tunnel encapsulation limit value n. The tunnel encapsulation limit is the maximum number of additional encapsulations permitted for the packets. The tunnel encapsulation limit option is defined in RFC 2473. This option is valid only for tunnel types ipinip6 and ip6inip6. The default is 4.

forward  
(inet6 only) Enable forwarding of packets by this interface. This is the default behavior.

-forward  
(inet6 only) Disable forwarding of packets by this interface.

metric  
Set the routing metric of the interface to n. The default is 0. The routing metric is used by the routing protocol (see gated(1M)). Higher metrics have the effect of making a route less favorable; metrics are counted as additional hops to the destination network or host.

netmask  
(inet only) Specify how much of the address to reserve for subdividing networks into sub-networks or aggregating networks into supernets. mask can be specified as a single hexadecimal number with a leading 0x, with a dot-notation Internet address, or with a pseudo-network name listed in the network table (see networks(4)). For subdividing networks into sub-networks, mask must include the network part of the local address, and the subnet part which is taken from the host field of the address. mask must contain 1's in the bit positions in the 32-bit address that are to be used for the network and subnet parts, and 0's in the host part. The 1's in the mask must be contiguous starting from the leftmost bit position in the 32-bit field. mask must contain at least the standard network portion, and the subnet field must be contiguous with the network portion. The subnet field must contain at least 1 bit. For aggregating networks into supernets, mask must only include a portion of the network part. mask must contain contiguous 1's in the bit positions starting from the leftmost bit of the 32-bit field.

prefix  
(inet6 only) n indicates the length of the network prefix associated with this interface. The primary interface (see Interface Naming subsection given below) prefix length is always 10, and is not configurable. The prefix option can be used only with the address option, and only for secondary interfaces. Default: 64. Range: 1 to 128.

private  
(inet6 only) Enable processing of received router advertisements. Secondary address(es) will be autoconfigured on the interface using the prefix(es) received in router advertisement(s). This is the default behavior.

-private  
(inet6 only) Disable processing of received router advertisements.

tunnel type  
Specify the type of tunnel. The tunnel types can be ip6inip, ipinip6, ip6inip6, and 6to4.

tdst  
Specify the destination addr of the tunnel. This is the destination address in the encapsulating (outer) header. It should be an address configured on an interface on the tunnel exit-point node. For tunnel types ipinip6 and ip6inip6, the addr should be an IPv6 address. For tunnel type ip6inip, the addr should be an IPv4 address; and for tunnel type 6to4, the tdst parameter should not be specified.

tsaddr  
Specify the source addr of the tunnel. This is the source address in the encapsulating (outer) header. It should be an address configured on an interface in the tunnel entry-point node. For tunnel types ipinip6 and ip6inip6, the addr should be an IPv6 address. For
tunnel types `ip6inip` and `6to4`, the `addr` should be an IPv4 address.

`arp` (inet only) Enable the user of the Address Resolution Protocol in mapping between network level addresses and link level addresses (default). If an interface already had the Address Resolution Protocol disabled, the user must "unplumb" the interface before it can be enabled for Address Resolution Protocol.

`-arp` (inet only) Disable the use of the Address Resolution Protocol. If an interface already had the Address Resolution Protocol enabled, the user must "unplumb" the interface before it can be disabled for Address Resolution Protocol.

`plumb` Setup the Streams plumbing needed for TCP/IP for a primary interface name. (See the Interface Naming subsection given below.) By default, the `plumb` operation is done automatically when an IP address is specified for an interface.

`unplumb` Tear down the Streams plumbing for a primary interface name. (See the Interface Naming subsection given below.) Secondary interface does not require "plumbing". A secondary IPv4 interface can be removed by assigning an IP address of 0.0.0.0 to it. Remove a secondary IPv6 interface by assigning an IP address of :: to it.

### Interface Naming

The interface name associated with a network card is composed of the name of the interface (e.g. `lan` or `snap`), the `ppa` number which identifies the card instance for this interface, and an optional IP index number which allows the configuration of multiple IP addresses for an interface. For LAN cards, the interface name `lan` will be used to designate Ethernet encapsulation and `snap` for IEEE 802.3 encapsulation. The `lanscan` command can be used to display the `interface name` and `ppa number` of each interface that is associated with a network card (see `lanscan(1M)`).

IPv4 and IPv6 interfaces can coexist over the same physical network interface device using the same naming scheme. IPv6 interfaces are configured using the "inet6" `ifconfig` subcommand. (See the IPv6 subsection given below.)

The tunnel interface names should be `iptu*` for "IP6-in-IP" tunnels and `6to4` tunnels. Example: `iptu0`, `iptu1`. The tunnel interface names should be `ip6tu*` for "IP-in-IP6" tunnels and "IP6-in-IP6" tunnels. Example: `ip6tu0`, `ip6tu1`.

### IP Index Number

Multiple IP addresses assigned to the same `interface name` may be in different subnets. An example of an interface name without an IP index number is `lan0`. An example of an interface name with a IP index number is `lan0:1`. Note: specifying `lan0:0` is equivalent to `lan0`.

A primary interface is an interface whose IP index number is zero. A secondary interface is an interface whose IP index number is non-zero.

### Loopback Interface

The loopback interface (`lo0`) is automatically configured when the system boots with the TCP/IP software. The IP address and netmask of the primary IPv4 loopback interface are 127.0.0.1 and 255.0.0.0, respectively. The IP address and prefix of the primary IPv6 loopback interface are ::1 and 128 respectively. The user is not permitted to change the address of the primary loopback interface (`lo0`). It is permissible to assign other IP addresses to lo0 with non-zero IP index numbers (`lo0:1`, `lo0:2`, etc). This allows a system to have a "system IP" address that is available as long as one interface remains usable.

### Supernets

(inet only) A supernet is a collection of smaller networks. Supernetting is a technique of using the netmask to aggregate a collection of smaller networks into a supernet.

This technique is particularly useful when the limit of 254 hosts per class C network is too restrictive. In those situations a netmask containing only a portion of the network part may be applied to the hosts in these networks to form a supernet. This supernet netmask should be applied to those interfaces that connect to the supernet using the `ifconfig` command. For example, a host can configure its interface to connect to a class C supernet, 192.6, by configuring an IP address of 192.6.1.1 and a netmask of 255.255.0.0.
IPv6 Interfaces

*inet6* must be specified when an IPv6 interface is configured. The address for an IPv6 interface can either be a hostname present in the host name database (see *hosts*(4)), or an address in the IPv6 colon notation.

Stateless Address Auto-configuration

Unlike IPv4 interfaces, IPv6 interfaces can be configured without an address and/or a prefix. Stateless address autoconfiguration requires no manual configuration of hosts, minimal (if any) configuration of routers, and no additional servers. A primary interface (*lanX:*0) is automatically assigned a link-local address by the system when the interface is configured. A link-local address comprises the well-known link-local prefix FE80::/10 and the interface identifier, which is typically 64 bits long and is based on EUI-64 identifiers. The link-local address allows automatic discovery of other hosts and routers on the same link, using the Neighbor Discovery Protocol (see *NDP*(7P)). The link-local address can be used as the source address to communicate with other nodes when no routers are present. If a router on the local link advertises prefixes in router advertisements, the host autoconfigures its secondary interfaces and its default gateway. The address of an autoconfigured secondary interface is formed by prepending the prefix received from the router to the interface identifier, the same interface identifier that is used in forming the primary interface.

Manual Address Configuration

IPv6 interfaces can also be configured with manually assigned addresses and/or prefixes. A primary interface must be configured with a link-local address and the prefix must not be specified. The prefix is always 10. The universal/local bit, the U bit, of the interface identifier must be 0, per section 2.5.1 of RFC 2373. Accordingly, a manually assigned address for a primary interface must have the following pattern: FE80::Xxx:xxxx:xxxx:xxxx where x is any hexadecimal digit, and M must be 0, 1, 4, 5, 8, 9, C, or D.

When a primary interface is configured with a manually assigned address, secondary interfaces will be autoconfigured if the host receives prefixes from router advertisements. The addresses on the secondary interfaces will be derived from the interface identifier portion of manually configured address in the primary interface.

When a secondary interface is configured with a manually assigned address, and if the user chooses an IP index number that has been used for an autoconfigured secondary interface, the manual configuration overwrites the autoconfiguration. When this happens, network connectivity through the overwritten autoconfigured IP address is temporarily lost. At a later time, when the host receives the next router advertisement, the host will bring up another secondary interface with a different IP index number, but with the same IP address, and network connectivity through that IP address is restored. Normally, a user can avoid this by checking used IP index numbers. However, there is always a possibility that address autoconfiguration due to router advertisement is happening concurrently while the user manually configures secondary interfaces.

To disable communication through a specific IP address on an autoconfigured secondary interface, that secondary interface should be marked down, not removed or overwritten with a different IP address. If that interface is removed or overwritten, the host will reconfigure another secondary interface with the same IP address when it receives the next router advertisement. Alternatively, the router can be configured to stop advertising the prefix that corresponds to the offending IP address.

IPv6 interface flags displayed:

An IPv6 interface may have three new flags that are not present in an IPv4 interface: TUNNEL, AUTO, and ONLINK. The TUNNEL flag is set for the tunnel interfaces. The AUTO flag is set for autoconfigured secondary interfaces. The ONLINK flag is set for interfaces with IP addresses that can be reached without going through a router.

Examples:

Stateless address autoconfiguration with link-local address

```
ifconfig lan0 inet6 up
```

Manual configuration for a primary interface with link-local address

```
ifconfig lan0 inet6 fe80::1 up
```

Manual configuration for a secondary interface with link-local address

```
ifconfig lan0:1 inet6 fe80::3 up
```

Manual configuration for a secondary interface with global address

ifconfig lan0:3 inet6 2222::4 up

Tunnel interface configuration:

HP-UX supports "IP6-in-IP" configured tunnels as specified in RFC 2893, "IP-in-IP6" and "IP6-in-IP6" configured tunnels as specified in RFC 2473, and 6to4 automatic tunnel as specified in RFC 3056.

IP6-in-IP tunnel interface configuration

"IP6-in-IP" configured tunnel allows dual stack IPv6/IPv4 nodes to communicate over an IPv4 infrastructure, by encapsulating the IPv6 packet inside an IPv4 header. The tunnel configuration must be done on both the local (tunnel entry-point) system and the remote (tunnel exit-point) system. "IP6-in-IP" tunnels can be configured as shown below:

```
ifconfig iptu0 inet6 tunnel ip6inip [source_link-local_address] \ 
[destination_link-local_address] tsrc tunnel_local_IPv4_address \ 
tdst tunnel_remote_IPv4_address up
```

The source and destination link-local IPv6 addresses of the tunnel interface are optional; if they are not specified, they will be autoconfigured based on the tunnel_local_IPv4_address and tunnel_remote_IPv4_address, respectively.

The tunnel_local_IPv4_address should be an address configured on the local system, and tunnel_remote_IPv4_address should be an address configured on the remote system.

Example. On the local system:

```
# ifconfig iptu0 inet6 tunnel ip6inip tsrc 192.168.1.1 \ 
tdst 192.168.2.2 up
```

Example. On the remote system:

```
# ifconfig iptu0 inet6 tunnel ip6inip tsrc 192.168.2.2 \ 
tdst 192.168.1.1 up
```

If multiple tunnels are configured with the same tunnel_local_IPv4_address, autoconfiguration of only the first tunnel will succeed. Other tunnels should be manually configured with link-local addresses.

Manual link-local addresses can be assigned to the tunnel interface as shown below:

```
# ifconfig iptu0 inet6 tunnel ip6inip fe80::1 fe80::2 \ 
    tsrc 10.10.1.1 tdst 10.10.2.2 up
```

Secondary addresses to the tunnel interfaces can be assigned as shown below:

```
# ifconfig iptu0:1 inet6 2ffe::1 3ffe::1 up
```

IP-in-IP6 tunnel interface configuration

"IP-in-IP6" tunnel configuration allows transmission of IPv4 packets encapsulated in an IPv6 header. "IP-in-IP6" tunnels can be configured as shown below:

```
ifconfig ip6tu0 inet tunnel ipinip6 source_IPv4_address \ 
destination_IPv4_address tsrc tunnel_local_IPv4_address \ 
tdst tunnel_remote_IPv6_address up
```

The tunnel_local_IPv4_address should be an address configured on the local system, and tunnel_remote_IPv6_address should be an address configured on the remote system. The tunnel configuration should be done on both the local and the remote systems.

Example. On the local system:

```
# ifconfig ip6tu0 inet tunnel ipinip6 10.10.1.1 10.10.2.2 \ 
    tsrc 2ffe::1 tdst 3ffe::1 up
```

Example. On the remote system:

```
# ifconfig ip6tu0 inet tunnel ipinip6 10.10.2.2 10.10.1.1 \ 
    tsrc 3ffe::1 tdst 2ffe::1 up
```

IP6-in-IP6 tunnel interface configuration

"IP6-in-IP6" tunnel configuration allows transmission of IPv6 packets encapsulated in an IPv6 header. "IP6-in-IP6" tunnels can be configured as shown below:
The `tunnel_remote_IPv6_address` should be an address configured on the remote system. The tunnel configuration should be done on both the local and the remote systems.

Example. On the local system:

```
# ifconfig ip6tu0 inet6 tunnel ip6inip6 fe80::1 fe80::2 \
    tsrc 2ffe::1 tdst 3ffe::1 up
```

Example. On the remote system:

```
# ifconfig ip6tu1 inet6 tunnel ip6inip6 fe80::2 fe80::1 \
    tsrc 3ffe::1 tdst 2ffe::1 up
```

### 6to4 tunnel interface configuration

6to4 tunnel interface configuration allows automatic tunneling of IPv6 packets encapsulated in an IPv4 header over an IPv4 infrastructure. 6to4 tunnel interface can be configured as shown below:

```
ifconfig iptu1 inet6 tunnel 6to4 [6to4_primary_address] \
    tsrc tunnel_local_IPv4_address up
```

The `tunnel_local_IPv4_address` should be a global IPv4 address. The primary address of the 6to4 interface should be a 6to4 address and not a link-local address. The 6to4 primary address is optional, if it is not specified, a 6to4 address will be autoconfigured based on `tunnel_local_IPv4_address`.

Example:

```
# ifconfig iptu1 inet6 tunnel 6to4 tsrc 15.13.136.204 up
```

A 6to4 address can be manually assigned as shown below:

```
# ifconfig iptu1 inet6 tunnel 6to4 2002:f0d:88cc::1 \
    tsrc 15.13.136.204 up
```

To advertise a 6to4 prefix, see `rtradvd(1M)`.

### DIAGNOSTICS

Messages indicate if the specified interface does not exist, the requested address is unknown, or the user is not privileged and tried to alter an interface’s configuration.

### AUTHOR

`ifconfig` was developed by HP and the University of California, Berkeley.

### SEE ALSO

`netstat(1), lanscan(1M), route(1M), rtradvd(1M), inet(3N), inet6(3N), hosts(4), NDP(7P), routing(7).`

IP Version 6 Addressing Architecture, RFC2373, Hinden, Derring.


Connection of IPv6 Domains via IPv4 Clouds, RFC 3056, Carpenter, Moore.
SYNOPSIS

inetd(1M) inetd(1M)

NAME
inetd - Internet services daemon

SYNOPSIS

/usr/sbin/inetd [-a] [-p proc_limit] [-r count [interval]] [-l -s]
/usr/sbin/inetd [-c]
/usr/sbin/inetd [-k]

DESCRIPTION

The inetd daemon is the Internet superserver, which invokes Internet server processes as needed. It must be running before other hosts can connect to the local host through ftp, rcp, remsh, rlogin, and telnet. The inetd daemon also supports services based on the Remote Procedure Call (RPC) protocol (NFS), such as rwald and rusersd. If RPC servers are started by inetd, the portmap server (see portmap(1M)) must be started before inetd.

The inetd daemon is designed to invoke all the Internet servers as needed, thus reducing load on the system. It is normally started at system boot time. Only one inetd can run at any given time.

The inetd daemon starts servers for both stream and datagram type services. For stream services, inetd listens for connection requests on Internet stream sockets. When a connection is requested for one of its sockets, inetd decides which service the socket will support, forks a process, invokes an appropriate server for the connection, and passes the connected socket to the server as stdin and stdout. Then inetd returns to listening for connection requests.

For datagram services, inetd waits for activity on Internet datagram sockets. When an incoming datagram is detected, inetd forks a process, invokes an appropriate server, and passes the socket to the server as stdin and stdout. Then inetd waits, ignoring activity on that datagram socket, until the server exits.

The inetd daemon is normally started by the /sbin/init.d/inetd script, which is invoked during the boot-time initialization. Otherwise, inetd can be started only by the superuser.

The Internet daemon and the servers it starts inherit the LANG and TZ environment variables and the umask of the process that started inetd. If inetd is started by the superuser, it inherits the superuser’s umask, and passes that umask to the servers it starts.

Note: Services currently supported by inetd will work in an IPv6 environment with a few changes to the configuration file /etc/inetd.conf (see inetd.conf(4)). When invoked, inetd reads /etc/inetd.conf and configures itself to support whatever services are included in that file (see inetd.conf(4)). The inetd daemon also performs a security check if the file /var/adm/inetd.sec exists (see inetd.sec(4)). If the Internet daemon refuses a connection for security reasons, the connection is shut down. Most RPC-based services, if their first connection is refused, attempt to connect four more times at 5-second intervals before timing out. In such cases, inetd refuses the connection from the same service invocation five times. This is visible in the system log if the inetd connection logging and syslogd logging for the daemon facility are both enabled (see syslogd(1M)).

The inetd daemon provides several “trivial” services internally by use of routines within itself. The services are echo, discard, chargen (character generator), daytime (human readable time), and time (machine readable time in the form of the number of seconds since midnight, January 1, 1900). The inetd daemon provides both TCP- and UDP-based servers for each of these services. See inetd.conf(4) for instructions on configuring internal servers.

Options

inetd recognizes the following options. These options can be used only by a superuser.

-a Enable user level auditing. Services started by inetd will be audited based on the user’s audit specification (see audit(1M), userdbset(1M), and the user field in inetd.conf(4)).

The HP-UX Standard Mode Security Extensions package (StdModSecExt) must be installed on the system in order to obtain the userdbset command.

If a service is audit unaware (see audit(4)), it will not be audited if the user’s audit specification is disabled.

Example: If auditing for root is disabled, audit unaware services, such as remshd invoked by inetd for the specified username will not be audited even though auditing for the username is enabled.
inetd(1M)

- c  Reconfigure the Internet daemon; in other words, force the current inetd to reread /etc/inetd.conf. This option sends the signal SIGHUP to the Internet daemon that is currently running. Any configuration errors that occur during the reconfiguration are logged to the syslogd daemon facility.

- k  Kill the current inetd. This option sends the signal SIGTERM to the Internet daemon that is currently running, causing it to exit gracefully. This option is the preferred method of killing inetd.

- l  By default, inetd starts with connection logging disabled. If no inetd is running, the -l option causes the inetd to start with connection logging enabled. Otherwise the -l option causes inetd to send the signal SIGQUIT to the inetd that is already running, which causes it to toggle the state of connection logging.

- p  proc_limit
By default, inetd spawns any number of child processes to serve incoming connections. When inetd is started with the -p option, inetd does not spawn a child process if the number of child processes already running in the system has reached the proc_limit value. inetd spawns a new child process only when the number of running child processes is less than the proc_limit value. If an invalid value or zero is specified for the -p option, inetd spawns any number of child processes to serve incoming connections. This option cannot be used when an inetd process is already running.

- r  count [interval]
inetd identifies a UDP service as broken or in an infinite loop when it receives count number of connections in interval seconds of time. When inetd finds any such broken service, it discards the packet requesting the socket connection, and refuses access to that service. inetd tries enabling that service after 10 minutes and accepts connections for that service. This is applicable to all UDP services other than tftp, bootp and rpc. Using the -r option, you can specify the values for count and interval, which need to be decimal numbers. If you invoke inetd without this option or specify invalid values for this option, the default values 40 and 60 are taken for count and interval, respectively. This option cannot be used when an inetd process is already running.

- s  This option is similar to the -l option, but it suppresses the hostname while logging into the syslog file. If inetd is not running, the -s option causes inetd to start with suppressed hostname logging enabled. If inetd is running, the -s option causes inetd to send SIGFPE signal to inetd that is already running. This causes inetd to toggle the state of suppressed hostname logging.

When inetd is running with either of -l or of -s logging enabled, the Internet daemon logs attempted connections to services. It also logs connection attempts which fail the security check. This information can be useful when trying to determine if someone is repeatedly trying to access your system from a particular remote system (in other words, trying to break into your system). Successful connection attempts are logged to the syslogd daemon facility at the info log level. Connection attempts failing the security check are logged at the notice log level. inetd also logs whether the connection logging has been enabled or disabled at the info log level.

DIAGNOSTICS
The following diagnostics are returned by the Internet daemon before it disconnects from the terminal.

An inetd is already running
An attempt was made to start an Internet daemon when one was already running. It is incorrect to call the Internet daemon a second time without the -c, -k, -l or -s option.

There is no inetd running
An attempt was made to reconfigure an Internet daemon when none was running.

Inetd not found
This message occurs if inetd is called with -c and another Internet daemon is running but cannot be reconfigured. This occurs if the original Internet daemon died without removing its semaphore.

Next step: Use the inetd -k command to remove the semaphore left by the previous Internet daemon; then restart the daemon.
The following diagnostics are logged to the syslogd daemon facility. Unless otherwise indicated, messages are logged at the error log level.

/etc/inetd.conf: Unusable configuration file

The Internet daemon is unable to access the configuration file /etc/inetd.conf. The error message preceding this one specifies the reason for the failure.

/etc/inetd.conf: line number: error

There is an error on the specified line in /etc/inetd.conf. The line in the configuration file is skipped. This error does not stop the Internet daemon from reading the rest of the file and configuring itself accordingly.

Next step: Fix the line with the error and reconfigure the Internet daemon by executing the inetd -c command.

system_call: message

system_call failed. See the corresponding manual entry for a description of system_call. The reason for the failure is explained in message.

Cannot configure inetd

None of the services/servers listed in the configuration file could be set up properly, due to configuration file errors.

Too many services (max n)

The number of active services listed in the configuration file exceeds the "hard" limit that can be supported by the system (see setrlimit(2)).

Next step: Reduce the number of services listed in the configuration file, then reconfigure the Internet daemon by running the command inetd -c.

file: \ found before end of line line

file can be either inetd.conf or inetd.sec. If a backslash is not immediately followed by an end of line, it is ignored and the information up to the end of line is accepted. In this case, the next line of the file is not appended to the end of the current line. Unless all the information required is present on a single line, configuration file error messages are also output. This message is logged at the warning log level.

service/protocol: Unknown service

The call to the library routine getservbyname (see getservent(3N)) failed. The service is not listed in /etc/services.

Next step: Include that service in /etc/services or eliminate the entry for the service in /etc/inetd.conf.

service/protocol: Server failing (looping), service terminated.

When inetd tries to start 40 servers within 60 seconds for a datagram service, other than bootp, rpc, or tftp, it assumes that the server is failing to handle the connection. To avoid entering a potentially infinite loop, inetd issues this message, discards the packet requesting the socket connection, and refuses further connections for this service. After 10 minutes, inetd tries to reinstate the service, and once again accepts connections for the service. inetd provides -r command-line option to modify the default values 40 and 60.

service/protocol: socket: message

service/protocol: listen: message

service/protocol: getsockname: message

Any one of the three errors above makes the service unusable. For another host to communicate with the server host through this service, the Internet daemon needs to be reconfigured after any of these error messages.

service/protocol: bind: message

If this error occurs, the service is temporarily unusable. After 10 minutes, inetd tries again to make the service usable by binding to the Internet socket for the service.
service/protocol: Access denied to remote_host (address)

The remote host failed to pass the security test for the indicated service. This information can be useful when trying to determine if someone is repeatedly trying to access your system from a particular remote system (in other words, trying to break into your system). This message is logged at the warning log level.

service/protocol: Connection from remote_host (address)

When connection logging is enabled, this message indicates a successful connection attempt to the specified service. This message is logged at the notice log level.

service/protocol: Added service, server executable

Keeps track of the services added when reconfiguring the Internet daemon. This message is logged at the info log level.

service/protocol: New list

Lists the new user IDs, servers or executables used for the service when reconfiguring the Internet daemon. This message is logged at the info log level.

service/protocol: Deleted service

Keeps track of the services deleted when reconfiguring the Internet daemon. This message is logged at the info log level.

Security File (inetd.sec) Errors

The following errors, prefixed by /var/adm/inetd.sec:, are related to the security file inetd.sec:

Field contains other characters in addition to * for service

For example, field 2 of the Internet address 10.5*.8.7 is incorrect.

Missing low value in range for service

For example, field 2 of the Internet address 10.-5.8.7 is incorrect.

Missing high value in range for service

For example, field 2 of the Internet address 10.5-.8.7 is incorrect.

High value in range is lower than low value for service

For example, field 2 of the Internet address 10.5-3.8.7 is incorrect.

allow/deny field does not have a valid entry for service

The entry in the allow/deny field is not one of the keywords allow or deny. No security for this service is implemented by inetd since the line in the security file is ignored. This message is logged at the warning log level.

RPC Related Errors for NFS Users

These errors are specific to RPC-based servers:

/etc/inetd.conf: line number: Missing program number
/etc/inetd.conf: line number: Missing version number

Error on the specified line of /etc/inetd.conf. The program or version number for an RPC service is missing. This error does not stop the Internet daemon from reading the rest of the file and configuring itself accordingly. However, the service corresponding to the error message will not be configured correctly.

Next step: Fix the line with the error, then reconfigure the Internet daemon by executing the inetd -c command.

/etc/inetd.conf: line number: Invalid program number

Error on the specified line of /etc/inetd.conf. The program number for an RPC service is not a number. This error does not stop the Internet daemon from reading the rest of the file and configuring itself accordingly. However, the service corresponding to the error message will not be correctly configured.

Next step: Fix the line with the error, then reconfigure the Internet daemon by executing the inetd -c command.
AUTHOR

`inetd` was developed by HP and the University of California, Berkeley.

NFS was developed by Sun Microsystems, Inc.

FILES

`/etc/inetd.conf` List of Internet server processes.

`/var/adm/inetd.sec` Optional security file.

SEE ALSO

`umask(1), portmap(1M), syslogd(1M), getservent(3N), inetd.conf(4), inetd.sec(4), protocols(4), services(4), environ(5)`.
NAME
inetsvcs_sec - enable/disable secure internet services

SYNOPSIS
inetsvcs_sec [enable | disable | status]

DESCRIPTION
/usr/sbin/inetsvcs_sec is used to enable or disable secure internet services (SIS) by updating inetsvcs.conf(4) with the appropriate entry. SIS provide network authentication when used in conjunction with HP DCE security services, the HP Praesidium/Security Server, or other software products that provide a Kerberos V5 Network Authentication Services environment.

Options
inetsvcs_sec recognizes the following options:

- enable The secure internet services are enabled. The services now provide network authentication through Kerberos V5.
- disable The secure internet services are disabled. The services now follow the traditional behavior of prompting for passwords.
- status This option displays the current authentication mechanism used (i.e., whether Kerberos authentication is enabled or not).

SEE ALSO
sis(5), inetsvcs.conf(4).
NAME
infocmp - compare or print out terminfo descriptions

SYNOPSIS
[-w width] [-A directory] [-B directory] [termname]...

DESCRIPTION
infocmp can be used to compare a binary terminfo entry with other terminfo entries, rewrite a
terminfo description to take advantage of the use= terminfo field, or print out a terminfo description
from the binary file (term) in a variety of formats. In all cases, the boolean fields will be printed first,
followed by the numeric fields, followed by the string fields.

Default Options
If no options are specified and zero or one termnames are specified, the -I option will be assumed. If more
than one termname is specified, the -d option will be assumed.

Comparison Options [-d] [-c] [-n]
infocmp compares the terminfo description of the first terminal termname with each of the descriptions
given by the entries for the other terminal's termnames. If a capability is defined for only one of the
terminals, the value returned will depend on the type of the capability: F for boolean variables, -1 for
integer variables, and NULL for string variables.
-d produces a list of each capability that is different between two entries. This option is useful to show
the difference between two entries, created by different people, for the same or similar terminals.
-c produces a list of each capability that is common between two entries. Capabilities that are not set
are ignored. This option can be used as a quick check to see if the -u option is worth using.
-n produces a list of each capability that is in neither entry. If no termnames are given, the environment
variable TERM will be used for both of the termnames. This can be used as a quick check to see if any-
things was left out of a description.

Source Listing Options [-I] [-L] [-C] [-r]
The -I, -L, and -C options will produce a source listing for each terminal named.
-I use the terminfo names
-L use the long C variable name listed in <term.h>
-C use the termcap names
-r when using -C, put out all capabilities in termcap form
If no termnames are given, the environment variable TERM will be used for the terminal name.
The source produced by the -C option may be used directly as a termcap entry, but not all of the
parameterized strings may be changed to th termcap format. infocmp will attempt to convert most of
the parameterized information, but anything not converted will be plainly marked in the output and com-
mented out. These should be edited by hand.
All padding information for strings will be collected together and placed at the beginning of the string
where termcap expects it. Mandatory padding (padding information with a trailing '/') will become
optional.
All termcap variables no longer supported by terminfo, but which are derivable from other ter-
minfo variables, will be output. Not all terminfo capabilities will be translated; only those variables
which were part of termcap will normally be output. Specifying the -r option will take off this restric-
tion, allowing all capabilities to be output in termcap form.
Note that because padding is collected to the beginning of the capability, not all capabilities are output.
Mandatory padding is not supported. Because termcap strings are not as flexible, it is not always possi-
ble to convert a terminfo string capability into an equivalent termcap format. A subsequent conver-
sion of the termcap file back into terminfo format will not necessarily reproduce the original ter-
minfo source.
Some common terminfo parameter sequences, their termcap equivalents, and some terminal types
which commonly have such sequences, are:

HP-UX 11i Version 2: December 2007 Update – 1 – Hewlett-Packard Company 381
Use= Option [-u]

-\texttt{u} produces a \textit{terminfo} source description of the first terminal \textit{termname} which is relative to the sum of the descriptions given by the entries for the other terminals \textit{termnames}. It does this by analyzing the differences between the first \textit{termname} and the other \textit{termnames} and producing a description with \texttt{use=} fields for the other terminals. In this manner, it is possible to retrofit generic \textit{terminfo} entries into a terminal's description. Or, if two similar terminals exist, but were coded at different times or by different people so that each description is a full description, using \texttt{infocmp} will show what can be done to change one description to be relative to the other.

A capability will get printed with an at-sign (@) if it no longer exists in the first \textit{termname}, but one of the other \textit{termname} entries contains a value for it. A capability's value gets printed if the value in the first \textit{termname} is not found in any of the other \textit{termname} entries, or if the first of the other \textit{termname} entries that has this capability gives a different value for the capability than that in the first \textit{termname}.

The order of the other \textit{termname} entries is significant. Since the \textit{terminfo} compiler \texttt{tic} does a left-to-right scan of the capabilities, specifying two \texttt{use=} entries that contain differing entries for the same capabilities will produce different results depending on the order that the entries are given in. \texttt{infocmp} will flag any such inconsistencies between the other \textit{termname} entries as they are found.

Alternatively, specifying a capability after a \texttt{use=} entry that contains that capability will cause the second specification to be ignored. Using \texttt{infocmp} to recreate a description can be a useful check to make sure that everything was specified correctly in the original source description.

Another error that does not cause incorrect compiled files, but will slow down the compilation time, is specifying extra \texttt{use=} fields that are superfluous. \texttt{infocmp} will flag any other \textit{termname} \texttt{use=} fields that were not needed.

Other Options [\texttt{-s d|i|l|c|\texttt{-v}\ |\texttt{-V}\ |\texttt{-1}\ |\texttt{-w width}]

\texttt{-s} sorts the fields within each type according to the argument below:

\begin{itemize}
  \item \texttt{d} leave fields in the order that they are stored in the \textit{terminfo} database.
  \item \texttt{i} sort by \textit{terminfo} name.
  \item \texttt{l} sort by the long C variable name.
  \item \texttt{c} sort by the \textit{termcap} name.
\end{itemize}

If the \texttt{-s} option is not given, the fields printed out will be sorted alphabetically by the \textit{terminfo} name within each type, except in the case of the \texttt{-C} or the \texttt{-L} options, which cause the sorting to be done by the \textit{termcap} name or the long C variable name, respectively.

\texttt{-v} prints out tracing information on standard error as the program runs.
\texttt{-V} prints out the version of the program in use on standard error and exit.
\texttt{-1} causes the fields to be printed out one to a line. Otherwise, the fields will be printed several to a line to a maximum width of 60 characters.
\texttt{-w} changes the output to \textit{width} characters.

Changing Databases [\texttt{-A directory}\ |\texttt{-B directory}]

The location of the compiled \textit{terminfo} database is taken from the environment variable \texttt{TERMINFO}. If the variable is not defined, or the terminal is not found in that location, the system \textit{terminfo} database, usually in \texttt{/usr/lib/terminfo}, will be used. The options \texttt{-A} and \texttt{-B} may be used to override this location. The \texttt{-A} option will set \texttt{TERMINFO} for the first \textit{termname} and the \texttt{-B} option will set \texttt{TERMINFO} for the other \textit{termnames}. With this, it is possible to compare descriptions for a terminal with the same name located in two different databases. This is useful for comparing descriptions for the same terminal created by different people.
infocmp(1M)

FILES
/usr/lib/terminfo/*/ Compiled terminal description database.

SEE ALSO
captoinfo(1M), tic(1M), curses_intro(3X), terminfo(4).
init(1M) init(1M)

NAME
init - process control initialization

SYNOPSIS
/sbin/init 0|1|2|3|4|5|6|S|s|q|a|b|c

DESCRIPTION
The init daemon and command is a general process spawner. Its primary role is to create processes from a script stored in the file /etc/inittab (see inittab(4)). This file usually has init spawn a getty on each line where users can log in. It also controls autonomous processes required by any particular system.

At boot time, init is started as a system daemon.

While the system is running, a user-spawned init directs the actions of the boot init. It accepts a one-character argument and signals the boot init with the kill() system call to perform the appropriate action.

The arguments have the following effect:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-6</td>
<td>Place the system in one of the run levels 0 through 6.</td>
</tr>
<tr>
<td>a</td>
<td>b</td>
</tr>
<tr>
<td>Q</td>
<td>q</td>
</tr>
<tr>
<td>S</td>
<td>s</td>
</tr>
</tbody>
</table>

Boot init considers the system to be in a run level at any given time. A run level can be viewed as a software configuration of the system, where each configuration allows only a selected group of processes to exist. The processes spawned by boot init for each of these run levels are defined in the inittab file.

Boot init can be in one of eight run levels, 0-6, and S or s. The run level is changed by having a privileged user run the init command. This user-spawned init sends appropriate signals to the boot init.

Boot init is invoked inside the HP-UX system as the last step in the boot procedure. Boot init first performs any required machine-dependent initialization, such as setting the system context. Next, boot init looks for the inittab file to see if there is an entry of the type initdefault (see inittab(4)). If an initdefault entry is found, boot init uses the run level specified in that entry as the initial run level to enter. If this entry is not in inittab, or inittab is not found, boot init requests that the user enter a run level from the logical system console, /dev/syscon. If S or s is entered, boot init goes into the single-user level. This is the only run level that does not require the existence of a properly formatted inittab file. If initdefault does not exist, then by default the only legal run level that boot init can enter is the single-user level.

In the single-user level, the logical system console terminal /dev/syscon is opened for reading and writing, and the command /usr/bin/su, /usr/bin/sh, or /sbin/sh is invoked immediately. To exit from the single-user run level, one of two options can be selected:

- If the shell is terminated with an end-of-file, boot init reprompts for a new run level.
- User init can signal boot init and force it to change the current system run level.

When attempting to boot the system, some processes spawned by boot init may send display messages to the system console (depending on the contents of inittab). If messages are expected but do not appear during booting, it may be caused by the logical system console (/dev/syscon) being linked to a device that is not the physical system console (/dev/systty). If this occurs, you can force boot init to relink /dev/syscon to /dev/systty by pressing the DEL (delete) key (ASCII 127) on the physical system console.

When boot init prompts for the new run level, you can only enter one of the digits 0 through 6 or the letter S or s. If you enter S, boot init operates as previously described in single-user mode with the additional result that /dev/syscon is linked to the user's terminal line, thus making it the logical system console. A message is generated on the physical system console, /dev/systty, identifying the new logical system console.

When boot init comes up initially, and whenever it switches out of single-user state to normal run states, it sets the states (see ioctl(2)) of the logical system console, /dev/syscon, to those modes saved in the file /etc/ioctl.syscon. This file is written by boot init whenever single-user mode is entered. If
this file does not exist when boot init wants to read it, a warning is printed and default settings are
assumed.

If 0 through 6 is entered, boot init enters the corresponding run level. Any other input is rejected and a
new prompt is issued. If this is the first time boot init has entered a run level other than single-user,
boot init first scans inittab for special entries of the type boot and bootwait. These entries are
performed — provided that the run level entered matches that of the entry — before any normal processing
of inittab takes place. In this way, any special initialization of the operating system, such as mounting
file systems, can take place before users are allowed onto the system. The inittab file is scanned to find
all entries that are to be processed for that run level.

Run levels in HP-UX are defined as follows:

0 Shut down HP-UX.
1 Start a subset of essential system processes. This state can also be used to perform system
administration tasks.
2 Start most system daemons and login processes. This state is often called the "multi-user
state". Login processes either at local terminals or over the network are possible.
3 Export filesystems and start other system processes. In this state NFS filesystems are often
exported, as may be required for an NFS server.
4 Activate graphical presentation managers and start other system processes.
5–6 These states are available for user-defined operations.

The default run level is usually run level 3 or 4, depending on the system configuration.

When init transitions into a new run level 0–6, the master sequencer script rc is invoked. rc in turn
invokes each of the start or kill scripts for each installed subsystem for each intervening run level. When
transitioning to a higher run level start scripts are invoked, and when transitioning to a lower run level kill
scripts are invoked. See rc(1M).

In a multiuser environment, the inittab file is usually set up so that boot init creates a process for
each terminal on the system.

For terminal processes, ultimately the shell terminates because of an end-of-file either typed explicitly or
generated as the result of hanging up. When boot init receives a child death signal telling it that a pro-
cess it spawned has died, it records the fact and the reason it died in the utmps database, /etc/utmp,
/var/adm/wtmps, and /var/adm/wtmp if it exist (see who(1)). A history of the processes spawned is
kept in /var/adm/wtmps and /var/adm/wtmp if it exists.

To spawn each process in the inittab file, boot init reads each entry and, for each entry that should be
respawned, it forks a child process. After it has respawned all of the processes specified by the inittab
file, boot init waits for one of its descendant processes to die, a powerfail signal, or until it is signaled by
a user init to change the system's run level. When one of the above three conditions occurs, boot init
re-examines the inittab file. New entries can be added to the inittab file at any time. However, boot init
still waits for one of the above three conditions to occur. For an instantaneous response, use the
init Q (or init q) command to wake up boot init to re-examine the inittab file without changing
the run level.

If boot init receives a powerfail signal (SIGPWR) and is not in single-user mode, it scans inittab for
special powerfail entries. These entries are invoked (if the run levels permit) before any other processing
takes place by boot init. In this way, boot init can perform various cleanup and recording functions
whenever the operating system experiences a power failure. Note, however, that although boot init
receives SIGPWR immediately after a power failure, boot init cannot handle the signal until it resumes
execution. Since execution order is based on scheduling priority, any eligible process with a higher priority
executes before boot init can scan inittab and perform the specified functions.
When boot init is requested to change run levels via a user init, it sends the warning signal SIGTERM to all processes that are undefined in the target run level. Boot init waits 20 seconds before forcibly terminating these processes with the kill signal SIGKILL. Note that boot init assumes that all these processes (and their descendants) remain in the same process group that boot init originally created for them. If any process changes its process group affiliation with either setpgrp() or setpgrp2() (see setsid(2) and setpgid(2)), it will not receive these signals. (Common examples of such processes are the shells csh and ksh (see csh(1) and ksh(1)).) Such processes need to be terminated separately.

A user init can be invoked only by users with appropriate privileges.

SECURITY FEATURES

Boot Authentication

The system administrator can enable the boot authentication feature. If enabled, the system cannot be booted into single user mode until the password of a user authorized to boot the system in single user mode is provided. Refer to the /etc/default/security file in the security(4) manual page for detailed information on configurable parameters that affect the behavior of this command. The currently supported parameters for boot authentication are:

BOOT_AUTH and BOOT_USERS

On systems that have been converted to trusted mode, use the System Administration Manager (SAM) program (see sam(1M)).

DIAGNOSTICS

If boot init finds that it is continuously respawning an entry from inittab more than 10 times in 2 minutes, it will assume that there is an error in the command string, generate an error message on the system console, and refuse to respawn this entry until either 5 minutes have elapsed or it receives a signal from a user init. This prevents boot init from using up system resources if there is a typographical error in the inittab file or a program is removed that is referenced in inittab.

WARNINGS

Boot init assumes that processes and descendants of processes spawned by boot init remain in the same process group that boot init originally created for them. When changing init states, special care should be taken with processes that change their process group affiliation, such as csh and ksh.

One particular scenario that often causes confusing behavior can occur when a child csh or ksh is started by a login shell. When boot init is asked to change to a run level that would cause the original login shell to be killed, the shell's descendant csh or ksh process does not receive a hangup signal since it has changed its process group affiliation and is no longer affiliated with the process group of the original shell. Boot init cannot kill this csh or ksh process (or any of its children).

If a getty process is later started on the same tty as this previous shell, the result may be two processes (the getty and the job control shell) competing for input on the tty.

To avoid problems such as this, always be sure to manually kill any job control shells that should not be running after changing init states. Also, always be sure that user init is invoked from the lowest level (login) shell when changing to an init state that may cause your login shell to be killed.

If init is unable to write to /etc/ioctl.syscon, a message is logged to the console. This may lead to the corruption of console settings.

FILES

/dev/syscon
/dev/systty
/etc/default/security
/etc/inittab
/etc/ioctl.syscon
/etc/utmp
/var/adm/wtmp
/var/adm/wtmps

SEE ALSO

csh(1), ksh(1), login(1), sh(1), who(1), getty(1M), rc(1M), utmpd(1M), ioctl(2), kill(2), setpgid(2), setsid(2), getutsent(3C), updatebwdb(3C), inittab(4), security(4), utmp(4).

STANDARDS CONFORMANCE
init: SVID2, SVID3
insf(1M)

NAME
insf - install special (device) files

SYNOPSIS
/sbin/insf


DESCRIPTION
The insf command installs special files in the devices directory, normally /dev. If required, insf
creates any subdirectories that are defined for the resulting special file.

If no options are specified, special files are created for all new devices in the system. New devices are those
devices for which no special files have been previously created. A subset of the new devices can be selected
with the -C, -d, and -H options.

With the -e option, insf reinstalls the special files for pseudo-drivers and existing devices. This is useful
for restoring special files when one or more have been removed.

Normally, insf displays a message as the special files are installed for each driver. The -q (quiet) option
suppresses the installation message. The -v (verbose) option displays the installation message and the
name of each special file as it is created.

Options
insf recognizes the following options.

-C class  Match devices that belong to a given device class, class. Device classes can be listed
          with the lsdev command (see lsdev(1M)). They are defined in the files in the directory
          /usr/conf/master.d. The special class pseudo includes all pseudo-drivers. This option cannot be used with -d.

-d driver  Match devices that are controlled by the specified device driver, driver. Device
drivers can be listed with the lsdev command (see lsdev(1M)). They are defined in
the files in the directory /usr/conf/master.d. This option cannot be used with -C.

-D directory Override the default device installation directory /dev and install the special files in
directory instead. directory must exist; otherwise, insf displays an error message
and exits. See WARNINGS.

-e Reinstall the special files for pseudo-drivers and existing devices. This is useful for
restoring special files if one or more have been removed.

-H hw-path Match devices at a given hardware path, hw-path. Hardware paths can be listed with the
ioscan command (see ioscann(1M)). A hardware path specifies the addresses of
the hardware components leading to a device. It consists of a string of numbers
separated by periods (.), such as 52 (a card), 52.3 (a target address), and 52.3.0
(a device). If a hardware component is a bus converter, the following period, if any, is
replaced by a slash (/) as in 2, 2/3, and 2/3.0.

If the specified path contains fewer numbers than are necessary to reach a device, spe-
cial files are made for all devices at addresses that extend the given path. If the
specified path is 56, then special files are made for the devices at addresses 56.0,
56.1, 56.2, etc.

-I instance Match a device with the specified instance number. Instances can be listed with the
-f option of the ioscann command (see ioscann(1M)).
This option is effective only if the -e option is specified or if an appropriate device
class or driver is specified with a -C or -d option.

-n npty Install npty special files for each specified ptym and pty driver. The pty driver
specifies both the ptym and pty drivers. npty is a decimal number.

This option is effective only if the -e option is specified or if an appropriate device
class or driver is specified with a -C or -d option.

If this option is omitted, npty defaults to 60 for the pty and pty drivers.
-p first-optical-disk: last-optical-disk
Install the special files for those optical disks located in slots in the range first-optical-disk to last-optical-disk. The two variables can have values from the set 1a, 1b, ..., 32a, 32b. This option only applies to the autox0 and schgr drivers. If it is omitted, the 64 special files for both sides of 32 optical disks (1a through 32b) will be installed.

-q
Quiet option. Normally, insf displays a message as each driver is processed. This option suppresses the driver message, but not error messages. See the -v option.

-s nstrpty
Install nstrpty slave-side stream special files for the pts driver. nstrpty is a decimal number. This option only applies to the pts special file installation.
This option is effective only if the -e option is specified or if an appropriate device class or driver is specified with a -C or -d option.
If this option is omitted, nstrpty defaults to 60.

-v
Verbose option. In addition to the normal processing message, display the name of each special file as it is created. See the -q option.

Naming Conventions
Many special files are named using the ccardtargetdevice naming convention. These variables have the following meaning wherever they are used.
card The unique interface card identification number from ioscan (see ioscan(1M)). It is represented as a decimal number with a typical range of 0 to 255.
target The device target number, for example the address on a HP-FL or SCSI bus. It is represented as a decimal number with a typical range of 0 to 15.
device A address unit within a device, for example, the unit in a HP-FL device or the LUN in a SCSI device. It is represented as a decimal number with a typical range of 0 to 15.

Special Files
This subsection shows which special files are created and the permissions for each device driver.
The special file names are relative to the installation directory, normally /dev. This directory may be overridden with the -D option.

insf sets the file permissions and the owner and group IDs. They are shown here in a format similar to that of the ll command:
special-file permissions owner group

For example:
tty rw-rw-rw- bin bin

device Driver Special Files and Description

arp The following special file is installed:

arp rw-rw-rw- root sys

asio0 For the built-in serial port, the following special files are installed for each card instance:
ttycard/p0 rw---w---w- bin bin Direct connect

asio0 For the SAS console ports, the following special files are installed for each card instance:
ttycard/p0 rw---w---w- bin bin local console port (direct connect)
ttycard/p1 rw---w---w- bin bin remote session port (direct connect)
sassy rw------- root sys internal console port (direct connect)
### insf(1M)

**tty**

**asyncdsk**

The following special files are installed:

- **asyncdsk**: rw-rw-rw- bin bin
- **async**: rw-rw-rw- bin bin

**audio**

The following special files are installed. Note the underscore (_) before *card* in each special file name. For *card* 0, the device files are linked to files without the trailing _0 in their names.

- **audio_card**: rw-rw-rw- bin bin
- **audioCtl_card**: rw-rw-rw- bin bin
- **audioBA_card**: rw-rw-rw- bin bin
- **audioBL_card**: rw-rw-rw- bin bin
- **audioBU_card**: rw-rw-rw- bin bin
- **audioEA_card**: rw-rw-rw- bin bin
- **audioEL_card**: rw-rw-rw- bin bin
- **audioEU_card**: rw-rw-rw- bin bin
- **audioIA_card**: rw-rw-rw- bin bin
- **audioIL_card**: rw-rw-rw- bin bin
- **audioIU_card**: rw-rw-rw- bin bin
- **audioLA_card**: rw-rw-rw- bin bin
- **audioLL_card**: rw-rw-rw- bin bin
- **audioLU_card**: rw-rw-rw- bin bin
- **audioNA_card**: rw-rw-rw- bin bin
- **audioNL_card**: rw-rw-rw- bin bin
- **audioNU_card**: rw-rw-rw- bin bin

**autox0**

Special file names for *autox0* and *schgr* use the format:

```
card target device surface
```
For each autochanger device, the following special files are installed:

```
ac/card/target/device_surface rw-r----- bin sys
```
Block entry

```
rac/card/target/device_surface rw-r----- bin sys
```
Character entry

```
rac/card/target/device rw-------- bin sys
```
Character entry

### Beep

The following special file is installed:

```
beep rw-rw-rw- bin bin
```

### CentIf

For each card instance, the following special file is installed.

```
ccard/target/device_lp rw-rw-rw- lp bin
```
Handshake mode 2, character entry

### Consp1

For each card instance, the following special files are installed:

```
tty/card/p0 rw--w--w- bin bin
```
Direct connect

### CN

The following special files are installed:

```
syscon rw--w--w- bin bin
systty rw--w--w- bin bin
console rw--w--w- root sys
ttyconf rw-------- root sys
```

### Devconfig

The following special file is installed:

```
config rw-r----- root sys
```

### Diag0

The following special file is installed:

```
diag/diag0 rw-------- bin bin
```

### Diag1

The following special file is installed:

```
diag/diag1 rw-------- bin bin
```

### Diag2

The following special files are installed:

```
diag2 rw-------- bin bin
diag/diag2 rw-------- bin bin
```

### Disc3 Sdisk

For each disk device, the following special files are installed:

```
dsk/card/target/device rw-r----- bin sys
```
Block entry

```
rdsk/card/target/device rw-r----- bin sys
```
Character entry

For disc3 instances, the following additional special files are installed:

```
floppy/card/target/device rw-r----- bin sys
```
For disk instances that are partitioned, the following additional special files are installed:

```
dsk/card/target/device partition  rw-r----- bin sys  Block entry
rdsk/card/target/device partition  rw-r----- bin sys  Character entry
```

dlp

The following special files are installed:

```
dlp  rw-rw-rw- root sys
dlp0  rw-rw-rw- root sys
dlp1  rw-rw-rw- root sys
dlp2  rw-rw-rw- root sys
dlp3  rw-rw-rw- root sys
dlp4  rw-rw-rw- root sys
```

dmem

The following special file is installed:

```
dmem  rw----- bin bin
```

echo

The following special file is installed:

```
echo  rw-rw-rw- root sys
```

fddi

The following special file is installed:

```
lancard  rw-rw-rw- bin bin
```

framebuf

For each graphics device claimed by 'graph3' driver, the following special files are installed.

```
crt/device_number  rw-rw-rw- bin bin
ocrt/device_number  rw-rw-rw- bin bin
```

device_number is 0 indexed and is assigned in the order in which the devices appear in `ioscan(1M)` output.

If the console device is a graphics device, the files `crt` and `ocrt` are created as the console device. If the console is not a graphics device, `crt` and `ocrt` are identical to `crt0` and `ocrt0`.

hil

For each device, the following special files are installed. Note the underscore (_) before `card` in each special file name.

For `card 0`, the device files are linked to files named `hiladdr` for the link addresses 1 to 7; `hilkbd` for the cooked keyboard device; and `rhil` for the `hil` controller device.

```
hil_card.addr  rw-rw-rw- bin bin
addr: link addresses 1 to 7
hilkbd_card  rw-rw-rw- bin bin
rhil_card  rw-rw-rw- bin bin
```

inet_clts

The following special file is installed:

```
inet_clts  rw-rw-rw- root sys
```

inet_cots

The following special file is installed:
The following special file is installed:

**inet_cots**

rw-rw-rw- root sys

**ip**
The following special file is installed:

**ip**

rw-rw-rw- root sys

**kepd**
The following special file is installed:

**kepd**

rw-r--r-- root other

**klog**
The following special file is installed:

**klog**

rw------- bin bin

**lan0** lan1 lan2 lan3**

For each card instance, the following special files are installed:

**lancard**

rw-rw-rw- bin bin

**ethercard**

rw-rw-rw- bin bin

**diag/lancard**

rw------- bin bin

**lantty0**

For each card instance, the following special files are installed:

**lanttycard**

rw-rw-rw- bin bin

Normal access

**diag/lanttycard**

rw-rw-rw- bin bin

Exclusive access

**lpr2** lpr3**

For each card instance, the following special files are installed:

**ccard**targetddevice lp

rw------- lp bin

**diag/ccard**targetddevice lp

rw------- bin bin

**mm**
The following special files are installed:

**mem**

rw-r----- bin sys

Minor 0

**kmem**

rw-r----- bin sys

Minor 1

**null**

rw-rw-rw- bin bin

Minor 2

**mux0**

For each instance of a 6-channel card, the following special files are installed:

**ttycard**pport

rw-r--w-- bin bin

port: 0 to 5, direct connect

**muxcard**

rw------- bin bin

**diag/muxcard**

rw------- bin bin

For each instance of a 16-channel card, the following special files are installed:

**ttycard**pport

rw-r--w-- bin bin

port: 0 to 15, direct connect

**muxcard**

rw------- bin bin

**diag/muxcard**

rw------- bin bin

**mux2**

For each instance of an 16-channel card, the following special files are installed:

**ttycard**pport

rw-r--w-- bin bin

port: 0 to 15, direct connect
For each card instance of an 8-channel card, the following special files are installed:

- **tty/card**
  - **port**: 0 to 7, direct connect

- **mux/card**

For each card instance of an 3-channel card, the following special files are installed:

- **tty/card**
  - **port**: 0, 1, and 7, direct connect

- **mux/card**

For each card instance, the following special files are installed:

- **tty/card**
  - **port**: 0 and 1, direct connect

- **netqa**
  - The following special file is installed:
    - **netqa**
      - **rw-rw-rw-** root sys

- **nuls**
  - The following special file is installed:
    - **nuls**
      - **rw-rw-rw-** root sys

- **pci_mux0**  **pci_mux1**
  - For each instance of a PCI mux card, the following "Direct Connect" special files are created. The term **card** below refers to the instance number of the mux card.
    - **tty/card/port_module/port**
      - port: 1 to 16, port number
    - **mux/card**
    - **diag/mux/card**
    - **diag/mux/card_1**
    - **diag/mux/card_2**

- **pflop**  **sflop**
  - For each card instance, the following special files are installed:
    - **floppy/card/target/device**
      - Block entry
    - **rfloppy/card/target/device**
      - Character entry

- **ps2**
  - The following special files are installed:
    - **ps2kbd**
      - Autosearch for first ps2 keyboard
    - **ps2mouse**
      - Autosearch for first ps2 mouse
    - **ps2_0**
      - ps2 port 0
ps2_1  rw-rw-rw-  bin  bin
        ps2 port 1

ptm  The following special file is installed:

ptmx  rw-rw-rw-  root  sys

pts  The following special files are installed:

pts/number  rw-rw-rw-  root  sys
            number: 0 to 59

pty  Specifying this driver tells insf to install the special files for both the master and slave pty drivers, ptym and ptys. The command insf -d pty is equivalent to the two commands insf -d ptym and insf -d ptys.

ptym  The following special files are installed:

ptym/clone  rw-r--r--  root  other

ptym/ptyindex number  rw-rw-rw-  bin  bin
            index: p to z, a to c, e to o; number: 0 to f (hexadecimal)
            The first 48 special files ptym/pty* are linked to pty*.

ptym/ptyindex number  rw-rw-rw-  bin  bin
            index: p to z, a to c, e to o; number: 00 to 99

ptym/ptyindex number  rw-rw-rw-  bin  bin
            index: p to z, a to c, e to o; number: 000 to 999

ptys  The following special files are installed:

ptys/ptyindex number  rw-rw-rw-  bin  bin
            index: p to z, a to c, e to o; number: 000 to 999

rawip  The following special file is installed:

rawip  rw-rw-rw-  root  sys

root  The following special files are installed:

root  rw-r-----  bin  sys

rroot  rw-r-----  bin  sys

sad  The following special file is installed:

sad  rw-rw-rw-  root  sys

sastty  For each card instance, the following special files are installed:

ttycardpport  rw--w--w-  bin  bin
            port: 0 to 1, direct connect

schgr  See autox0.

sdisk  See disc3.

sflop  See pflop.
For each driver instance, different special files are installed depending on the number of characters allowed in the target directory. There are two lists below, one for long file name directories and one for short file name directories (14 characters maximum). Short file names are used for files installed on an NFS file system.

Note that the first four special files in each list for tape driver instances 0-9 are also linked to `rmt/instance`, `rmt/instanceb`, `rmt/instanceM`, and `rmt/instanceMN`, respectively.

For installation in a long file name directory:

- `rmt/ctargetddevicenew BEST rw-rw-rw- bin bin` AT&T-style, best available density, character entry
- `rmt/ctargetddevicenew Bestb rw-rw-rw- bin bin` Berkeley-style, best available density, character entry
- `rmt/ctargetddevicenew Bestn rw-rw-rw- bin bin` AT&T-style, no rewind, best available density, character entry
- `rmt/ctargetddevicenew Bestnb rw-rw-rw- bin bin` Berkeley-style, no rewind, best available density, character entry

For installation in a short file name directory:

- `rmt/ctargetddevicenew f0 rw-rw-rw- bin bin` AT&T-style, best available density, character entry
- `rmt/ctargetddevicenew f0b rw-rw-rw- bin bin` Berkeley-style, best available density, character entry
- `rmt/ctargetddevicenew f0n rw-rw-rw- bin bin` AT&T-style, no rewind, best available density, character entry
- `rmt/ctargetddevicenew f0nb rw-rw-rw- bin bin` Berkeley-style, no rewind, best available density, character entry

For both long and short file name directories, the following additional files are created.

- `rmt/driver_name_config rw--r-- bin bin` Tape configuration, character entry
- `diag/rmt/ctargetddevicenew rw------- bin bin` For `tape2` only, diagnostic access, character entry

**stcpmap**

The following special file is installed:

- `stcpmap rw-rw-rw- root sys`

**strlog**

The following special file is installed:

- `strlog rw-rw-rw- root sys`

**tty**

The following special file is installed:

- `tty rw-rw-rw- bin bin`

**tape2**

See `stape`.

**tcp**

The following special file is installed:

- `tcp rw-rw-rw- root sys`

**telm**

The following special file is installed:

- `telnetm rw-rw-rw- root sys`

**tels**

The following special file is installed:

- `pts/tnumber rw-rw-rw- root sys`
The following special file is installed:

tlclts

tlclts rw-rw-rw- root sys

tlcots

The following special file is installed:

tlcots rw-rw-rw- root sys

tlcotsod

The following special file is installed:

tlcotsod rw-rw-rw- root sys

token2

The following special file is installed:

token2 rw-rw-rw- root sys

The following special file is installed:

tlclts rw-rw-rw- root sys

tlcots

The following special file is installed:

tlcots rw-rw-rw- root sys

tlcotsod

The following special file is installed:

tlcotsod rw-rw-rw- root sys

RETURN VALUE

insf exits with one of the following values:

0  Successful completion, including warning diagnostics.
1  Failure.

DIAGNOSTICS

Most diagnostic messages from insf are self-explanatory. Listed below are some messages deserving further clarification.

Warnings

Device driver name is not in the kernel

Device class name is not in the kernel

The indicated device driver or device class is not present in the kernel. A device driver and/or device class can be added to the kernel using kcmd(1M).

No instance number available for device class name

All of the instance numbers available for the device class are already assigned. Use the rmsf command to remove any unneeded devices from the system (see rmsf(1M)).

Don’t know how to handle driver name - no special files created for path

insf does not know how to create special files for the specified device driver. Use mknod to create special files for the device (see mknod(1M)).

EXAMPLES

Install special files for all new devices belonging to the tty device class:

insf -C tty

Install special files to the new device added at hardware path 2/4.0.0:

insf -H 2/4.0.0
WARNINGS

`insf` can change the mode, owner, or group of an existing special file, or unlink and recreate one; special files that are currently open may be left in an indeterminate state. Hence `insf` is recommended to be run in single-user mode.

Many commands and subsystems assume their device files are in `/dev`, therefore the use of the `-D` option is discouraged.

AUTHOR

`insf` was developed by HP.

FILES

`/dev/config` I/O system special file
`/etc/ioconfig` I/O system configuration database

SEE ALSO

`kcmodule(1M), ioscan(1M), lsdev(1M), lssf(1M), mknod(1M), mksf(1M), rmsf(1M)`. 
install(1M)

NAME
install - install commands

SYNOPSIS
    file [dirx ...]

DESCRIPTION
install is a command most commonly used in "makefiles" (see make(1)) to install a file (updated target
file) in a specific place within a file system. Each file is installed by copying it into the appropriate directory,
thereby retaining the mode and owner of the original command. The program prints messages telling the
user exactly what files it is replacing or creating and where they are going.

install is useful for installing new commands, or new versions of existing commands, in the standard
directories (i.e. /usr/bin, /usr/sbin, etc.).

If no options or directories (dirx...) are given, install searches a set of default directories (/usr/bin,
/usr/sbin, /sbin, and /usr/lib, in that order) for a file with the same name as file. When the first occurrence is found, install issues a message saying that it is overwriting that file with file (the
new version), and proceeds to do so. If the file is not found, the program states this and exits without
further action.

If one or more directories (dirx ...) are specified after file, those directories are searched before the directo-
tories specified in the default list.

Options
Options are interpreted as follows:

- c dira  Installs a new command (file) in the directory specified by dira, only if it is not found.
If it is found, install issues a message saying that the file already exists, and exits without overwriting it.
Can be used alone or with the -s option.

- f dirb  Forces file to be installed in given directory, whether or not one already exists. If the
file being installed does not already exist, the mode and owner of the new file will be
set to 755 and bin, respectively. If the file already exists, the mode and owner will
be that of the already existing file. Can be used alone or with the -o or -s options.

- i       Ignores default directory list, searching only through the given directories (dirx ...).
Can be used alone or with any other options other than -c and -f.

- n dirc  If file is not found in any of the searched directories, it is put in the directory specified
in dirc. The mode and owner of the new file will be set to 755 and bin, respec-
tively. Can be used alone or with any other options other than -c and -f.

- o       If file is found, this option saves the "found" file by copying it to OLDfile in the directory
in which it was found. This option is useful when installing a normally busy text
file such as /usr/bin/sh or /usr/sbin/getty, where the existing file cannot
be removed. Can be used alone or with any other options other than -c.

- g group Causes file to be owned by group group. This option is available only to users who
have appropriate privileges. Can be used alone or with any other option.

- u user  Causes file to be owned by user user. This option is available only to users who have
appropriate privileges. Can be used alone or with any other option.

- s       Suppresses printing of messages other than error messages. Can be used alone or
with any other options.

When no directories are specified (dirx ...), or when file cannot be placed in one of the directories specified,
install checks for the existence of the file /etc/syslist. If /etc/syslist exists, it is used to
determine the final destination of file. If /etc/syslist does not exist, the default directory list is
further scanned to determine where file is to be located.

The file /etc/syslist contains a list of absolute pathnames, one per line. The pathname is the
"official" destination (for example /usr/bin/echo) of the file as it appears on a file system. The file
/etc/syslist serves as a master list for system command destinations. If there is no entry for file in
the file /etc/syslist the default directory list is further scanned to determine where file is to be
located.
Cross Generation

The environment variable **ROOT** is used to locate the locations file (in the form $ROOT/etc/syslist). This is necessary in cases where cross generation is being done on a production system. Furthermore, each pathname in $ROOT/etc/syslist is appended to $ROOT (for example, $ROOT/usr/bin/echo), and used as the destination for file. Also, the default directories are appended to $ROOT so that the default directories are actually $ROOT/usr/bin, $ROOT/usr/sbin, $ROOT/sbin, and $ROOT/usr/lbin.

The file /etc/syslist ($ROOT/etc/syslist) does not exist on a distribution tape; it is created and used by local sites.

**WARNINGS**

**install** cannot create alias links for a command (for example, vi(1) is an alias link for ex(1)).

**SEE ALSO**

make(1), cpset(1M).
NAME
intctl - manage the interrupt configuration of the system

SYNOPSIS
/usr/contrib/bin/intctl [-h | -F | -p | -c cpu_id]
/usr/contrib/bin/intctl [-C class] [-H hw_path]
/usr/contrib/bin/intctl [-M -H hw_path -I intr_id -c cpu_id]
/usr/contrib/bin/intctl [-r file | -s file]
/usr/contrib/bin/intctl [-b [-w] [-a algorithm]
  [-i io:hw_path:intr_id:...]
  [-i cpu:hw_path:...]
  [-i drv:driver_name:...]
  [-o drv:driver_name:weight:...]
  [-o trig:balance_on_cpu:distribute_to_cpu]

DESCRIPTION
A processor receives an interrupt when either the processor’s interrupt pin is asserted (for line based interrupts) or if a processor detects an interrupt message bus transaction on the system bus (for transaction based interrupts).

Interrupts from the interface cards can be line or transaction based. Interrupts are routed to different processors during boot time.

The intctl command is a tool that allows a performance expert to display and modify these interrupt assignments. The tool only supports migration of external device interrupts. The performance analyst can also save and restore the interrupt configuration. If interrupt migration process completes successfully, a message is logged to the console and/or to the /var/adm/syslog/syslog.log file.

intctl resides in /usr/contrib/bin, and the command can be executed only by the superuser. The intctl command is not a general system administration command. It should be used only by performance tuning experts with a high level of system knowledge. The performance specialist can use the intctl command to view the interrupt configuration of the system and modify the interrupt assignments of the CPUs to re-distribute the system load across the CPUs.

intctl is synchronized with other High Availability (HA) events happening simultaneously on the system. An HA event can be a PCI OLA/R or Processor allocation/de-allocation. If any of these events are happening when intctl is trying to display interrupt information or is trying to migrate an interrupt to a CPU, intctl will exit with the error message below, and the user should retry the intctl command:

“Another HA event is in progress, try again!”

Interrupt migration is not enabled on workstations. Also non-MP safe drivers do not support interrupt migration. The tool will display an error message if the user tries to move the interrupts of a non-MP safe driver to a different CPU.

On a system with virtual partitions (vPars), intctl will only display CPUs in the current partition.

Using the -b option, the command can be used to balance interrupt distribution on a system. Interrupt assignments in a given system may not be distributed in a balanced manner. Most of the time, imbalance in distribution is caused by CPU migrations.

These migrations may cause the interrupts to get assigned to CPUs available in the system in a non-optimal fashion and they will not be distributed when more CPUs become available.

By default, HP-UX will distribute interrupts at boot time based on the round robin method. Each interrupt owned gets assigned to the available CPUs in a round robin fashion. However, CPU migrations, which may happen because of Work Load Manager (WLM) configuration, vPar administration activity, and Instant Capacity (iCAP/TiCAP) administration activity, can cause the interrupts to be assigned to a smaller set of CPUs causing an imbalance and thus a non-optimally configured system.

Using the -b option allows the user to manually balance the interrupt distribution of the system. Users can choose one of these two balancing algorithm of their choice to balance interrupts:

- **driver_weight**

  The default balancing algorithm used by intctl is **driver_weight**, which associates weights to each driver based on its interrupt frequency and balances the system such that each CPU is loaded with
a similar average weight from the interrupt load perspective.

- **round_robin**
  
  Another supported algorithm is **round_robin**, which assigns interrupts to the available CPUs in a round robin fashion. This is similar to the HP-UX default boot time interrupt distribution method, but the interrupt assignments may differ because of the difference in the way I/O cards and CPUs are discovered.

**driver_weight** is a better choice of algorithm for systems having I/O cards that demand largely varying range of interrupt processing needs. Hence **driver_weight** is chosen as the default algorithm. In systems where all I/O cards demand similar interrupt processing capacity or when it becomes difficult to determine interrupt load generated by each driver, then the **round_robin** algorithm can be used.

Administrators can also configure automatic balancing of interrupts at periodic intervals. Balancing will be performed only if there is an interrupt distribution imbalance. This is desirable in a dynamic CPU migration environment such as WLM (Work load Manager). Refer to **intrbald(1M)** for more details.

Several settings are provided for managing balancing of interrupts. This information can be provided using the command line options or can be persistently configured in the configuration file `/etc/intctl.conf`, see the **Interrupt Configuration File** section below.

**Options**

By default, the command displays interrupt information about all the interface cards on the system.

**intctl** recognizes the following options:

- **-a algorithm** Balancing of interrupts may be performed any time during system up time to reduce CPU overload because of interrupt handling. Users have several options to control balancing of interrupts.

  There are two algorithms supported to balance interrupt distribution.

  - **Driver weight (\-a driver_weight)** based approach.
    
    This is the default algorithm used to balance interrupts when no algorithm is specified. The default can be set by changing **INTCTL_DEFAULT_ALGO** in the configuration file `/etc/intctl.conf`.

    Each driver is given a weight based on the number of interrupts it may generate. Balancing operations ensure that each CPU is loaded (from interrupt load perspective) with a comparable total driver weight. These weights can be between 0 and **INT_MAX** (see **limits(5)**). Most of the HP-UX drivers are already defined in the configuration file section **INTCTL_DRIVER_WEIGHTS**. Users can modify or override these driver weights, but they should make sure not to set unrealistic driver weights without knowing the amount of interrupt load the driver could generate.

  - **Round robin (\-a round_robin)** approach.

    Each interrupt in the system will be assigned an available CPU in round robin fashion. This balancing approach can be used when it becomes difficult to differentiate drivers based on their interrupt load. Compared to the driver weight based approach, round robin could result in more interrupt migrations while balancing interrupts.

- **-b** Balance the interrupt distribution of the system by performing the least number of migrations that will distribute interrupt load across a specified set of CPUs.

- **-c cpu_id** By itself (without any other options), display interrupt information about the specified CPU.

  When used with the **-M** option, **-c cpu_id** specifies the CPU ID of the CPU to which the interrupt is to be moved.

- **-C class** Display interrupt information about all the interface cards belonging to the specified class.

  Can be used with the **-H hw_path** option to display interrupt information about the interface card under the **hw_path** that belongs to the specified **class**.

- **-F** Produce a compact listing of fields separated by colons.

- **-h** Display the usage of the command.
-H hw_path
Display interrupt information about all interface cards connected at the specified hardware path. For hardware paths / and 0, intctl prints the interrupt information about all the interface cards on the system.

When used with the -C class option, -H displays information about all interface cards connected to the path and which belong to the specified class.

When used with the -M option, -H hw_path specifies the hardware path of the interrupt that needs to be moved to a different CPU.

-i
While balancing interrupts, it may be desirable to avoid some interrupt assignments based on the I/O cards, CPUs, and so on. Multiple parameters can be specified by using the -i option multiple times.

NOTE: Specifying many parameters may reduce the scope of balancing interrupts and may cause interrupt distribution imbalance.

- I: Ignore I/O Card Interrupt (-i io:hw_path:intr_id)
While balancing interrupts, interrupt intr_id associated with this I/O card at the hardware path hw_path will not be migrated. intr_id represents a specific interrupt id or all interrupt ids of the specified I/O card if -l is used. These values can also be specified in the configuration file section INTCTL_HW_IGNORE.

- I: Ignore CPU (-i cpu:hw_path)
None of the interrupts assigned to the CPU with the specified hw_path will be migrated while balancing interrupts. Also, no new interrupts will be assigned to these CPUs while balancing interrupts. The same information can be specified in the INTCTL_CPU_IGNORE section of the configuration file.

- I: Ignore Driver (-i drv:driver_name)
None of the I/O cards claimed by the driver driver_name will be selected for interrupt migrations while balancing interrupts. Care needs to be taken while specifying a driver_name as one driver could possibly claim multiple I/O cards and specifying such a driver will reduce the scope of balancing of interrupts. The same information can be specified in the INTCTL_DRV_IGNORE section of the configuration file.

NOTE: Please refer to the INTCTL_DRV_IGNORE section of the configuration file for drivers that are currently not supported while balancing interrupts.

- L: intr_id
Used with -M option to specify the interrupt ID of the interrupt to be moved.

-M
Migrate an interrupt to a specified CPU. This option must be specified with the -H, -I, -c options.

The -M option should be specified first in the command line followed by the remaining options which can be specified in any order.

-o
This option can be used for overriding some existing parameters from the configuration file or for specifying new parameters.

- O: Override driver weight information (-o drv:driver_name:weight)
Specify a new driver weight or override an existing driver weight specified in the configuration file. driver_name is the driver name of an existing driver in the configuration file section INTCTL_DRIVER_WEIGHTS or a new driver name. weight is the interrupt load that the driver may generate.

NOTE: All I/O card drivers present on the system but not specified in the INTCTL_DRIVER_WEIGHTS section of the configuration file will be assigned a default weight of 10.

- O: Override the trigger for balancing of interrupts (-o trig:balance_on_cpu:distribute_to_cpu)
Using this option, the user can override the configuration file setting INTCTL_BALANCE_TRIGGER. These values determine when the interrupts will be balanced and the scope of balancing of interrupts.

balance_on_cpu is the minimum percentage of available number of CPUs that should be handling interrupts. Balancing of interrupts will start only if the number of CPUs
handling the interrupts is less than this percentage. A value 100 will always trigger balancing of interrupts, but if the system is optimally balanced with respect to interrupt distribution then it may not result in any interrupt migrations. The default value is 50.

distribute_to_cpu is the percentage of available number of CPUs that should be handling interrupts. Balancing of interrupts would distribute interrupts across this percentage of available number of CPUs. The default value is 75.

**NOTE:** If WLM (Work Load Manager) is configured to load balance across partitions by migrating CPUs, it is desirable not to set this value to more than 75.

**-p**
Display interrupt information about all the CPUs on the system in a long format with spacing in between the fields.

**-r file**
Restore the system interrupt configuration from the specified file, file. The interrupt configuration is restored only if all the interface cards and CPUs referenced in the saved configuration file are still present on the system and the CPUs are in the same state as in the saved configuration. If new cards and new CPUs are added to the system, `intctl` will continue to restore the interrupt configuration as long as the old configuration has not changed. `intctl` will fail to restore the interrupt configuration if the file permission is not 0600.

In restoring the system configuration, the command will assign interrupts from the interface cards to the CPUs as specified in the file.

**-s file**
Save the system interrupt configuration to the specified file, `file`, with file permission 0600. If the file exists, the content of the file will be overwritten and the file permissions will be changed to 0600. The command will store the interrupt information of all the CPUs on the system. This file can be used to restore the interrupt configuration of the system later using the `-r` option.

**-w**
Force migrations without asking for user input. Without this option, a user confirmation is requested while balancing interrupts (option `-b`). When used with `-b`, `-w` allows the user to balance interrupts without asking for migrations that should be skipped.

### Interrupt Configuration Display

The interrupt configuration can be displayed sorted by CPU ID (`intctl -p`) or sorted by interface card hardware path (`intctl -H hw_path`).

By default, the command displays interrupt information about all the interface cards on the system. Here is a sample interrupt configuration display, and the fields are explained below.

<table>
<thead>
<tr>
<th>hw path</th>
<th>class</th>
<th>drv name</th>
<th>card</th>
<th>cpu</th>
<th>cpu ID</th>
<th>intr card</th>
<th>intr</th>
<th>card</th>
<th>card cell</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0/0/0/0</td>
<td>lan</td>
<td>btlan</td>
<td>N/A</td>
<td>0</td>
<td>N/A</td>
<td>0</td>
<td>N/A</td>
<td>L</td>
<td>5</td>
<td>HP PCI</td>
</tr>
<tr>
<td>10/100Base-TX Core</td>
<td>ext_bus</td>
<td>c720</td>
<td>N/A</td>
<td>0</td>
<td>N/A</td>
<td>L</td>
<td>0</td>
<td>SCSI C895 Fast</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wide LVD</td>
<td>ext_bus</td>
<td>c720</td>
<td>N/A</td>
<td>0</td>
<td>N/A</td>
<td>L</td>
<td>1</td>
<td>SCSI C87x Ultra</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wide Single-Ended</td>
<td>ext_bus</td>
<td>c720</td>
<td>N/A</td>
<td>0</td>
<td>N/A</td>
<td>L</td>
<td>2</td>
<td>SCSI C87x Ultra</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**hw path**
A numerical string of hardware components separated by slash (/), to represent a bus converter. The first component in the hardware path is the cell (for a cell based system) or the system bus adapter (for a non-cell based system). The system bus adapter is followed by the address of the local bus adapter and the interface card. Subsequent numbers are separated by periods (.). Each number represents the location of a hardware component on the path of the device.

**class**
The class of the interface card, such as: `lan`, `tty`, `ext_bus`.

**drv name**
The driver associated with the card.

**card cell**
The cell number of the cell to which the card is connected.

**cpu ID**
An integer value representing the identity of the CPU to which the card’s interrupt is assigned.
The following is the output of `intctl -b` (balancing interrupts):

The following interrupt migrations will be performed for balancing interrupts (algorithm selected: `driver_weight`)

<table>
<thead>
<tr>
<th>Sl. Num.</th>
<th>Card H/W Path</th>
<th>Driver Name</th>
<th>Intr ID</th>
<th>From CPU ID (CPU H/W path)</th>
<th>To CPU ID (CPU H/W path)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1/0/8/1/1</td>
<td>fcd</td>
<td>1</td>
<td>0 (1/120)</td>
<td>7 (3/123)</td>
</tr>
<tr>
<td>2</td>
<td>1/0/8/1/0</td>
<td>fcd</td>
<td>1</td>
<td>0 (1/120)</td>
<td>6 (3/122)</td>
</tr>
<tr>
<td>3</td>
<td>1/0/0/3/0</td>
<td>igelan</td>
<td>1</td>
<td>0 (1/120)</td>
<td>5 (3/121)</td>
</tr>
<tr>
<td>4</td>
<td>1/0/0/3/0</td>
<td>c8xx</td>
<td>2</td>
<td>0 (1/120)</td>
<td>4 (3/120)</td>
</tr>
<tr>
<td>5</td>
<td>1/0/0/3/0</td>
<td>c8xx</td>
<td>1</td>
<td>0 (1/120)</td>
<td>4 (3/120)</td>
</tr>
<tr>
<td>6</td>
<td>1/0/0/2/1</td>
<td>c8xx</td>
<td>2</td>
<td>0 (1/120)</td>
<td>4 (3/120)</td>
</tr>
<tr>
<td>7</td>
<td>1/0/0/2/1</td>
<td>c8xx</td>
<td>1</td>
<td>0 (1/120)</td>
<td>4 (3/120)</td>
</tr>
</tbody>
</table>

Please select the migrations you want to skip.
Comma separated serial numbers or 'all' or 'none' : 1,2,7

The following migrations (serial number(s)) will be skipped.

Do you wish to skip these migrations (y/n): y

`intctl`: Moved the interrupt: 1, of card 1/0/0/1/0, driver igelan, from CPU:0 to CPU:5
`intctl`: Moved the interrupt: 2, of card 1/0/0/3/0, driver c8xx, from CPU:0 to CPU:4
`intctl`: Moved the interrupt: 1, of card 1/0/0/3/0, driver c8xx, from CPU:0 to CPU:4
`intctl`: Moved the interrupt: 2, of card 1/0/0/2/1, driver c8xx, from CPU:0 to CPU:4

Balancing of interrupts done.

The following is a description of each of the columns in the table:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sl. Num.</td>
<td>Serial number (starting from 1) of all the migrations that will be performed for balancing of interrupts. These serial numbers can be used in selecting migrations that have to be skipped.</td>
</tr>
<tr>
<td>Card H/W Path</td>
<td>The hardware path of I/O cards whose interrupt is getting migrated.</td>
</tr>
<tr>
<td>Driver Name</td>
<td>The driver associated with the card.</td>
</tr>
<tr>
<td>Intr ID</td>
<td>The identity of the interrupt to be moved.</td>
</tr>
<tr>
<td>From CPU ID (CPU H/W path)</td>
<td>The CPU id and CPU hardware path to which the interrupt is currently bound and the interrupt will be migrated off this CPU.</td>
</tr>
</tbody>
</table>
intctl(1M)

To CPU ID (CPU H/W path)
The CPU id and CPU hardware path to which the interrupt will get migrated.

Redirection
The intctl command allows the performance specialist to modify the interrupt assignment of an interface card. The user must specify the hardware path of interface card, the interrupt ID that needs to be moved, and the new CPU ID that the interrupt will be routed to.

When an interrupt is moved from one CPU to another, if the interrupt shares a line with other interrupts, all the interrupts on that line will be moved to the specified CPU. The kernel will add a message to the /var/adm/syslog/syslog.log file which will contain the hardware path and interrupt IDs of the interrupts being moved and the CPU ID of the CPU to which these interrupts were moved.

When migrating an interrupt from one CPU to another, if the card that the interrupt belongs to is in an erroneous or timed out state, the interrupt will not be moved. If an interrupt shares a line with other interrupts, and if any of the cards is in an erroneous state, then none of the interrupts on the line will be moved to the specified CPU.

Saving and Restoring System Interrupt Configurations
The intctl command can save and restore the system interrupt configuration in a user specified file. Before restoring the configuration, the intctl command checks to see if the system setup has changed by checking that all the interface cards and CPUs from the saved configuration are still present in the system and that the CPUs are in the same state as in the saved configuration. The command will continue to restore the configuration if new cards or CPUs have been added to the system since the interrupt configuration was saved.

Interrupt Configuration File
/etc/intctl.conf is the interrupt configuration file.

intctl parameters can be saved in this configuration file, which makes them persistent across reboot. These parameters can be changed or overridden by the command line options -i, -o and -a.

The different sections in the /etc/intctl.conf configuration file described below.

1. INTCTL_DRIVER_WEIGHTS
   Each line after the above string is expected to be of the form driver_name weight. driver_name is a string corresponding to the driver and weight is an integer corresponding to the driver's weight. These weights will be used while balancing interrupts using the driver_weight based algorithm. If a driver is not specified in this section and is present on the system, then a default weight of 10 is assumed. Weight can range from 0 to INT_MAX (see limits(5)). A 0 weight is considered as no interrupt load. A positive integer is considered as the relative interrupt load on the CPU with respect to different driver weights. More weight corresponds to more interrupt load on the CPU.

   The option -o drv:driver_name:weight can be used to override an existing driver weight or specify new driver weights temporarily.

   Example:
   
   INTCTL_DRIVER_WEIGHTS
   graf 300

2. INTCTL_HW_IGNORE
   Each line after the above string is expected to be of the form hw.path intr_id. hw.path is a string corresponding to the hardware path of the I/O card and intr_id is an integer corresponding to the interrupt ID. The specified I/O card and the interrupt ID combination is ignored (that is, will not be migrated) while balancing interrupts. If interrupt ID is -1, then all the interrupt IDs associated with that I/O card are ignored.

   The option -i io:hw.path:intr_id can be used to specify more I/O cards to be ignored temporarily.

   NOTE: If an I/O card shares the interrupt line with another I/O card whose driver is non MP-safe, then the interrupt of this I/O card cannot be migrated. intctl will display the following message if this happens (the actual hardware path will be different):

   "NOTE: I/O card <hardware_path> is sharing the interrupt line with another I/O card which is claimed by a non MP-safe driver. Add the line 'hardware_path -1' (hardware_path intr_id) to INTCTL_HW_IGNORE section of the configuration file /etc/intctl.conf to avoid getting this message."

This entry needs to be made in `INTCTL_HW_IGNORE` section of the `/etc/intctl.conf` configuration file. This will avoid selecting this card for further migrations while balancing interrupts. There is no impact on the system or balancing of interrupts if this activity is not performed. The only impact will be an interrupt migration failure message in the syslog and the above `intctl` message.

Example:
```
INTCTL_HW_IGNORE
1/2/1/0 2
1/3/1/0 -1
```

3. **INTCTL_CPU_IGNORE**

Each line after the above string is expected to be of the form `hw_path`. `hw_path` is a string corresponding to the hardware path of the CPU that should be ignored while balancing interrupts. These CPUs will not be affected by balancing of interrupts.

The option `-i cpu:hw_path` can be used to ignore more CPUs temporarily.

Example:
```
INTCTL_CPU_IGNORE
122/1
```

4. **INTCTL_DRV_IGNORE**

Each line after the above string is expected to be of the form `driver_name`. `driver_name` is a string corresponding to the driver name of the driver that should be ignored while balancing interrupts. Any I/O card claimed by this driver will not be selected for interrupt migrations while balancing interrupts (irrespective of the balancing algorithm chosen).

The option `-i drv:driver_name` can be used to ignore more drivers temporarily.

Example:
```
INTCTL_DRV_IGNORE
ipmi
```

NOTE: Please refer to this section of the configuration file for drivers that are currently not supported while balancing interrupts.

5. **INTCTL_DEFAULT_ALGO**

The following line should specify the default algorithm to be used while balancing interrupts (when `-a` is not used). The only supported options are `driver_weight` and `round_robin`.

The option `-a algorithm` can be used to change the algorithm temporarily.

Example:
```
INTCTL_DEFAULT_ALGO
driver_weight
```

6. **INTCTL_BALANCE_TRIGGER**

The following line should specify the trigger for balancing of interrupts and is expected to be of the form `balance_on_cpu distribute_to_cpu`. `balance_on_cpu` is an integer that specifies the minimum percentage of number of available CPUs that should be handling the interrupts. `distribute_to_cpu` is an integer that specifies the percentage of number of available CPUs that can handle interrupts after balancing interrupts.

If the percentage of number of available CPUs handling the interrupts is less than the `balance_on_cpu`, then balancing of interrupts is performed and interrupts are distributed across `distribute_to_cpu` percentage of CPUs. Both values should be in the range 0-100 and `balance_on_cpu` should be smaller than `distribute_to_cpu`.

NOTE: If WLM (Work Load Manager) is configured to load balance across partitions by migrating CPUs, it is desirable not to set this value to more than 75.

The option `-o trig:balance_on_cpu:distribute_to_cpu` can be used to override this setting temporarily.

Example:
```
INTCTL_BALANCE_TRIGGER
60 80
```
RETURN VALUE
Exit values are:

  0  Successful completion.
 >0  An error condition occurred.

EXAMPLES
Display information about all interface cards which belong to the class lan:

    intctl -C lan

Display the interrupt information of the card with hardware path 0/4/0/0/5/0:

    intctl -H 0/4/0/0/5/0

Display interrupt information of all the interface cards under the path, 0/4:

    intctl -H 0/4

Display interrupt information of all interface cards under the hardware path 0/4 and which belong to class lan:

    intctl -C lan -H 0/4

Display interrupt information about the CPU with CPU ID 3:

    intctl -c 3

Migrate the interrupt with ID 1, coming from the card whose hardware path is 0/4/0/0/5/0 to CPU 3:

    intctl -M -H 0/4/0/0/5/0 -I 1 -c 3

Store the system interrupt configuration to myconfig; if myconfig already exists, its contents are overwritten:

    intctl -s myconfig

Restore the system interrupt configuration from myconfig:

    intctl -r myconfig

Balance interrupts using driver_weight algorithm and without user confirmation:

    intctl -b -w -a driver_weight

Balance interrupts only if less than 40% of the available CPUs are handling interrupts, and distribute the interrupts across 75% of the CPUs available:

    intctl -b -w -o trig:40:75

Balance interrupts ignoring all interrupts of the I/O card with hardware path 1/2/1/0, ignore the CPU with hardware path 133/2, and ignore the driver graf1; also, ask for confirmation before performing interrupt migrations:

    intctl -b -i io:1/2/1/0:-1 -i cpu:133/2 -i drv:graf1

Balance interrupts according to the configuration file, but add a new driver graf with weight 300 and change the weight of existing btlan driver from 10 (specified in the configuration file) to 15:

    intctl -b -o drv:graf:300 -o drv:btlan:15

Balance interrupts if the current percentage of number of available CPUs handling interrupts is below 60% and distribute the interrupts across 80% of number of available CPUs the system:

    intctl -b -o trig:60:80

WARNINGS
The intctl command can be executed only by the superuser. The intctl command should be used only by performance analysts for performance tuning purposes. If care is not taken to redistribute the interrupts properly, it could lead to a decrease in the overall system performance by overloading some processors and by not optimally utilizing the remaining processors.
FILES

/etc/intctl.conf  intctl configuration file. See the Interrupt Configuration File section above.

SEE ALSO

intrbd(1M), ioscan(1M), limits(5).
intrbald(1M)

NAME
intrbald - daemon for balancing interrupts

SYNOPSIS
/usr/sbin/intrbald [-i interval  |  -k  |  -c]

DESCRIPTION
Balancing interrupts is a process which distributes interrupts across the available CPUs in an optimal fashion. This action is performed by intrctl(1M).

Performing frequent CPU migrations between system partitions may require you to perform periodic balancing of interrupts. This is especially true with systems that use WLM (Workload Manager), iCAP (Instant Capacity), and vPARs (Virtual Partitions) products. If the frequency of CPU migration is high then balancing of interrupts should be performed more frequently.

The intrbald daemon monitors the system for imbalance in interrupt distribution. If an imbalance is detected, after the specified interval (INTRBALD_INTERVAL), intrbald executes intrctl -b to balance the interrupt distribution. intrctl uses the parameters from the /etc/intrctl.conf configuration file for balancing the interrupts. For more information about the intrctl command, see intrctl(1M) and the /etc/intrctl.conf configuration file.

The intrbald daemon is started from the general purpose sequencer rc script (see rc(1M)), /sbin/init.d/intrbald_init, at run level 2 and is killed at run level 1. By default, it is not enabled. The daemon configuration file /etc/rc.config.d/intrbaldconf must be edited and the parameter INTRBALD_STATE set to enabled in order to enable running the daemon. After setting INTRBALD_STATE to enabled for the first time, you can start the daemon manually by executing the rc script (/sbin/init.d/intrbald_init start). Otherwise, the daemon will automatically start the next time the system is restarted.

Any actions performed by the daemon are logged in syslog and details of interrupt migrations performed will be logged in the file /var/adm/intrbald.log.

Options

-c
Instruct the running daemon to reexamine the configuration file /etc/rc.config.d/intrbaldconf. If the INTRBALD_INTERVAL parameter has been updated, then the running daemon uses the new value. If the INTRBALD_STATE parameter is set to disabled, then the running daemon stops executing.

-i interval
The daemon checks if the system requires any balancing of interrupts after interval seconds; valid range is 5 seconds to INT_MAX seconds (see limits(5)).

This value can also be specified persistently in the configuration file /etc/rc.config.d/intrbaldconf by setting the INTRBALD_INTERVAL parameter.

The default value is 60 seconds.

Setting this value too small can cause unnecessary interrupt migrations.

-k
Gracefully kills the running daemon.

intrbald rc Configuration File

/etc/rc.config.d/intrbaldconf is the intrbald rc configuration file.

The following parameters can be persistently specified in this configuration file:

INTRBALD_STATE
Specifies if the daemon should be started automatically during the system bootup process by the rc script. The daemon will only be started automatically if this parameter is set to enabled.

NOTE: This value is disabled by default. Before using the daemon, the value needs to be changed to enabled in the /etc/rc.config.d/intrbaldconf file. Then the daemon needs to be started, either manually by running intrbald, or it will be started automatically at next boot via the rc script.

INTRBALD_INTERVAL
The daemon will check if the system requires any interrupt balancing after this interval. Interval is specified in seconds; valid range is 5 seconds to INT_MAX seconds (see limits(5)). The default is 60 seconds.
intrbald(1M)

Setting this value too small could cause unnecessary interrupt migrations.

RETURN VALUE

Exit values are:

- 0  Successful completion.
- >0  An error condition occurred.

WARNINGS

The intrbald command can be executed only by the superuser. This command should be used only by
performance analysts for performance tuning purposes. If care is not taken to redistribute the interrupts
properly, it could lead to a decrease in the overall system performance by overloading some processors and
by not optimally utilizing the remaining processors.

FILES

/etc/rc.config.d/intrbaldconf intrbald rc configuration file. See the intrbald rc
Configuration File section above.
/sbin/init.d/intrbald_init intrbald rc script.
/sbin/rc2.d/S900intrbald_init rc start symbolic link.
/sbin/rc1.d/K100intrbald_init rc kill symbolic link.
/var/adm/intrbald.log daemon log file.

SEE ALSO

intctl(1M), ioscan(1M), rc(1M), limits(5).
ioinit(1M)

NAME
ioinit - test and maintain consistency between the kernel I/O data structures and /etc/ioconfig

SYNOPSIS
/sbin/ioinit -i [-r]
/sbin/ioinit -c
/sbin/ioinit -f infile [-r]

DESCRIPTION
The ioinit command is invoked by the init process when the system is booted, based on the ioin entry in /etc/inittab:

    ioin::sysinit:/sbin/ioinitrc > /dev/console 2>&1

where ioinitrc is a script to invoke ioinit with the -i and -r options. Given the -i option, ioinit checks consistency between the kernel I/O data structures (initialized with /stand/ioconfig, which is accessible for NFS-diskless support when the system boots up) and information read from /etc/ioconfig. If these are consistent, ioinit invokes insf to install special files for all new devices. If the kernel is inconsistent with /etc/ioconfig, ioinit updates /stand/ioconfig from /etc/ioconfig, and, if the -r option is given, reboots the system.

If /etc/ioconfig is corrupted or missing when the system reboots, ioinitrc brings the system up in single-user mode. The user should then restore /etc/ioconfig from backup or invoke the ioinit with the -c option to recreate /etc/ioconfig from the kernel.

If the -f option is given, ioinit reassigns instance numbers to existing devices within a given class based on infile. Reassignment takes effect when the system reboots. If ioinit finds no errors associated with the reassignment, and the -r option is given, the system is rebooted. (See the WARNINGS section.)

If the -c option is given, ioinit recreates /etc/ioconfig from the existing kernel I/O data structures.

Options
ioinit recognizes the following options:

- i          Invoke insf to install special files for new devices after checking consistency between the kernel and /etc/ioconfig.
- f infile  Use the file infile to reassign instance numbers to devices within a specified class. infile may have multiple entries, each to appear on a separate line, each field in the entry separated by 1 or more blanks. Entries should conform to the following format:

    h/w_path  class_name  instance_#

ioinit preprocesses the contents of infile, looking for invalid entries, and prints out explanatory messages. An entry is considered to be invalid if the specified hardware path or class name does not already exist in the system, or if the specified instance number already exists for the given class. For ext_bus class of devices, specified instance numbers should not exceed 255.

- r          Reboot the system when it is required to correct the inconsistent state between the kernel and /etc/ioconfig, as used with the -i option. When used with the -f option, if there are no errors associated with the instance reassignment, -r reboots the system.
- c          Recreate /etc/ioconfig, if the file is corrupted or missing and cannot be restored from backup. If -c is invoked, any previous binding of hardware path to device class and instance number is lost.

RETURN VALUE
0  No errors occurred, although warnings might be issued.
1  ioinit encountered an error.

DIAGNOSTICS
Most of the diagnostic messages from ioinit are self-explanatory. Listed below are some messages deserving further clarification. Errors cause ioinit to halt immediately.
Errors
/etc/ioconfig is missing.
/etc/ioconfig is corrupted.
   Either restore /etc/ioconfig from backup and then reboot, or recreate /etc/ioconfig using ioinit -c.
Permission to access /etc/ioconfig is denied.
   Change permissions to /etc/ioconfig to allow access by ioinit.
ext_bus instance value exceeds one byte limit
   Change specified instance number for ext_bus class of devices in infile. Note that the value of instance number should not exceed 255.
exec of insf failed.
ioinit completed successfully, but insf failed.
Instance number is already in kernel.
   Instance number already exists for a given class. Use rmsf to remove the existing instance number, then retry.
Hardware path is not in the kernel.
   The given hardware path is not in the kernel. Use ioscan -k to get the correct hardware path, then retry.
Device class name is not in the kernel.
   The given class name is not in the kernel. Use ioscan -k to get the correct class name, then retry.

EXAMPLES
To reassign an instance number to a device and class (specified in infile) and reboot the system:
/sbin/ioinit -f infile -r
where infile contains the following:

56.52    scsi    2
56.52    is the h/w_path, scsi is the class_name, and 2 is the instance_#.

WARNINGS
Running rmsf or insf overwrites the effect of reassignment by ioinit before the system is rebooted.

AUTHOR
ioinit was developed by HP.

FILES
/etc/ioconfig

SEE ALSO
init(1M), insf(1M), ioscan(1M), rmsf(1M), inittab(4), ioconfig(4).
NAME
ioscan - scan I/O system

SYNOPSIS
/usr/sbin/ioscan [-k | -u] [-e] [-d driver | -C class] [-I instance] [-H hw_path]
[-f[-n] | -F[-n]] [devfile]
/usr/sbin/ioscan -M driver -H hw_path [-I instance]
/usr/sbin/ioscan -t

DESCRIPTION
ioscan scans system hardware, usable I/O system devices, or kernel I/O system data structures as appropriate, and lists the results. For each hardware module on the system, ioscan displays by default the hardware path to the hardware module, the class of the hardware module, and a brief description.

By default, ioscan scans the system and lists all reportable hardware found. The types of hardware reported include processors, memory, interface cards and I/O devices. Scanning the hardware may cause drivers to be unbound and others bound in their place in order to match actual system hardware. Entities that cannot be scanned are not listed.

In the second form shown, ioscan forces the specified software driver into the kernel I/O system at the given hardware path and forces software driver to be bound. This can be used to make the system recognize a device that cannot be recognized automatically; for example, because it has not yet been connected to the system, does not support autoconfiguration, or because diagnostics need to be run on a faulty device.

In the third form, ioscan displays the date and time at which system hardware was last scanned.

Note: The -t option cannot be used with any other options available for this command.

Generally, ioscan requires superuser privileges. A non root user may use the -k option, only to display the kernel hardware tree. Driver binding and actual hardware scanning is restricted to root.

Options
ioscan recognizes the following options:
- -C class
  Restrict the output listing to those devices belonging to the specified class. Cannot be used with -d.
- -d driver
  Restrict the output listing to those devices controlled by the specified driver. Cannot be used with -C.
- -e
  Display EFI (Extensible Firmware Interface) device paths when available.
- -f
  Generate a full listing, displaying the module’s class, instance number, hardware path, driver, software state, hardware type, and a brief description.
- -F
  Produce a compact listing of fields (described below), separated by colons. This option overrides the -f option.
- -H hw_path
  Restrict the scan and output listing to those devices connected at the specified hardware path. The hardware path must be a bus path. Scanning below the bus level will not probe the hardware and may produce incorrect results. For example, specifying the path at the target level will always change the state of the device attached to it as NO_HW. The state of the device may be restored by retrying ioscan from a bus node above the NO_HW node. When used with -M, this option specifies the full hardware path at which to bind the software modules.
- -I instance
  Restrict the scan and output listing to the specified instance, when used with either -d or -C. When used with -M, specifies the desired instance number for binding.
- -k
  Scan kernel I/O system data structures instead of the actual hardware and list the results. No binding or unbinding of drivers is performed. The -d, -C, -I, and -H options can be used to restrict listings. Cannot be used with -u. This option does not require superuser privileges.
- -M driver
  Specify the software driver to bind at the hardware path given by the -H option. Must be used with the -H option.
ioscan(1M)

- **n**
  List device file names in the output. Only special files in the /dev directory and its subdirectories are listed.

- **t**
  Display the date and time at which the system hardware was last scanned. The output of the ioscans command used with this option is as follows: **Fri Nov 22 11:22:21 2002**

- **u**
  Scan and list usable I/O system devices instead of the actual hardware. Usable I/O devices are those having a driver in the kernel and an assigned instance number. The -d, -C, -I, and -H options can be used to restrict listings. The -u option cannot be used with -k.

The -d and -C options can be used to obtain listings of subsets of the I/O system, although the entire system is still scanned. Specifying -d or -C along with -I, or specifying -H or a devfile causes ioscans to restrict both the scan and the listing to the hardware subset indicated.

**Fields**

The -F option can be used to generate a compact listing of fields separated by colons (:), useful for producing custom listings with awk. Fields include the module's bus type, cdio, is_block, is_char, is_pseudo, block major number, character major number, minor number, class, driver, hardware path, identify bytes, instance number, module path, module name, software state, hardware type, a brief description, and card instance. If a field does not exist, consecutive colons hold the field's position. Fields are defined as follows:

- **class**
  A device category, defined in the files located in the directory /usr/conf/master.d and consistent with the listings output by lsdev (see lsdev(1M)). Examples are disk, printer, and tape.

- **instance**
  The instance number associated with the device or card. It is a unique number assigned to a card or device within a class. If no driver is available for the hardware component or an error occurs binding the driver, the kernel will not assign an instance number and a (-1) is listed.

- **hw path**
  A numerical string of hardware components, notated sequentially from the bus address to the device address. Typically, the initial number is appended by slash (/), to represent a bus converter (if required by your machine), and subsequent numbers are separated by periods (.). Each number represents the location of a hardware component on the path to the device.

- **driver**
  The name of the driver that controls the hardware component. If no driver is available to control the hardware component, a question mark (?) is displayed in the output.

- **software state**
  The result of software binding.

  **CLAIMED** software bound successfully
  **UNCLAIMED** no associated software found
  **SUSPENDED** associated software and hardware is in suspended state
  **DIFF_HW** software found does not match the associated software
  **NO_HW** the hardware at this address is no longer responding
  **ERROR** the hardware at this address is responding but is in an error state
  **SCAN** node locked, try again later

- **hardware type**
  Entity identifier for the hardware component. It is one of the following strings:

  **UNKNOWN** There is no hardware associated or the type of hardware is unknown
  **PROCESSOR** Hardware component is a processor
  **MEMORY** Hardware component is memory
  **BUS_NEXUS** Hardware component is bus converter or bus adapter
  **INTERFACE** Hardware component is an interface card
  **DEVICE** Hardware component is a device

- **bus type**
  Bus type associated with the node.
cdio The name associated with the Context-Dependent I/O module.

is_block A boolean value indicating whether a device block major number exists. A T or F is generated in this field.

is_char A boolean value indicating whether a device character major number exists. A T or F is generated in this field.

is_pseudo A boolean value indicating a pseudo driver. A T or F is generated in this field.

block major The device block major number. A -1 indicates that a device block major number does not exist.

character major The device character major number. A -1 indicates that a device character major number does not exist.

minor The device minor number.

identify bytes The identify bytes returned from a module or device.

module path The software components separated by periods ( ).

module name The module name of the software component controlling the node.

description A description of the device.

card instance The instance number of the hardware interface card.

RETURN VALUE
ioscan returns 0 upon normal completion and 1 if an error occurred.

EXAMPLES
Scan the system hardware and list all the devices belonging to the disk device class.

ioscan -C disk

Forcibly bind driver tape2 at the hardware path 8.4.1.

ioscan -M tape2 -H 8.4.1

AUTHOR
ioscan was developed by HP.

FILES
/dev/config
/dev/*

SEE ALSO
kcmodule(1M), lsdev(1M), olrad(1M), ioconfig(4).
NAME
isl - initial system loader

DESCRIPTION
isl implements the operating system independent portion of the bootstrap process. It is loaded and executed after self-test and initialization have completed successfully.

The processor contains special purpose memory for maintaining critical configuration related parameters (e.g. Primary Boot, Alternate Boot, and Console Paths). Two forms of memory are supported: Stable Storage and Non-Volatile Memory (NVM).

Typically, when control is transferred to isl, an autoboot sequence takes place. An autoboot sequence allows a complete bootstrap operation to occur with no intervention from an operator. isl executes commands from the autoexecute file in a script-like fashion. autoboot is enabled by a flag in Stable Storage.

autosearch is a mechanism that automatically locates the boot and console devices. For further information, see pdc(1M).

During an autoboot sequence, isl displays its revision and the name of any utility it executes. However, if autoboot is disabled, after isl displays its revision, it then prompts for input from the console device. Acceptable input is any isl command name or the name of any utility available on the system. If a non-fatal error occurs or the executed utility returns, isl again prompts for input.

Commands
There are several commands available in isl. The following is a list with a short description. Parameters may be entered on the command line following the command name. They must be separated by spaces. isl prompts for any necessary parameters that are not entered on the command line.

? , help  Help - List commands and available utilities
listf, ls  List available utilities
autoboot  Enable or disable the autoboot sequence
         Parameter - on or off
autosearch  Enable or disable the autosearch sequence
         Parameter - on or off
primpath  Modify the Primary Boot Path
         Parameter - Primary Boot Path in decimal
altpath  Modify the Alternate Boot Path
         Parameter - Alternate Boot Path in decimal
conspath  Modify the Console Path
         Parameter - Console Path in decimal
lsautofl, listautofl  List contents of the autoexecute file
display  Display the Primary Boot, Alternate Boot, and Console Paths
readnvm  Display the contents of one word of NVM in hexadecimal
         Parameter - NVM address in decimal or standard hexadecimal notation
readss  Display the contents of one word of Stable Storage in hexadecimal
         Parameter - Stable Storage address in decimal or standard hexadecimal notation

DIAGNOSTICS
isl displays diagnostic information through error messages written on the console and display codes on the LED display.

For the display codes, CE0x are informative only. CE1x and CE2x indicate errors, some of which are fatal and cause the system to halt. Other errors merely cause isl to display a message.
Non-fatal errors during an autoboot sequence cause the autoboot sequence to be aborted and isl to prompt for input. After non-fatal errors during an interactive isl session, isl merely prompts for input. Fatal errors cause the system to halt. The problem must be corrected and the system RESET to recover.

CE00 isl is executing.
CE01 isl is autobooting from the autoexecute file.
CE02 Cannot find an autoexecute file. autoboot aborted.
CE03 No console found, isl can only autoboot.
CE05 Directory of utilities is too big, isl reads only 2K bytes.
CE06 autoexecute file is inconsistent. autoboot aborted.
CE07 Utility file header inconsistent: SOM values invalid.
CE08 autoexecute file input string exceeds 2048 characters. utoboot aborted.
CE09 isl command or utility name exceeds 10 characters.
CE0F isl has transferred control to the utility.
CE10 Internal inconsistency: Volume label - FATAL
CE11 Internal inconsistency: Directory - FATAL
CE12 Error reading autoexecute file.
CE13 Error reading from console - FATAL.
CE14 Error writing to console - FATAL.
CE15 Not an isl command or utility.
CE16 Utility file header inconsistent: Invalid System ID.
CE17 Error reading utility file header.
CE18 Utility file header inconsistent: Bad magic number.
CE19 Utility would overlay isl in memory.
CE1A Utility requires more memory than is configured.
CE1B Error reading utility into memory.
CE1C Incorrect checksum: Reading utility into memory.
CE1D Console needed - FATAL.
CE1E Internal inconsistency: Boot device class - FATAL.
CE21 Destination memory address of utility is invalid.
CE22 Utility file header inconsistent: pdc_cache entry.
CE23 Internal inconsistency: iodc_entry_init - FATAL.
CE24 Internal inconsistency: iodc_entry_init - console - FATAL.
CE25 Internal inconsistency: iodc_entry_init - boot device - FATAL.
CE26 Utility file header inconsistent: Bad aux_id.
CE27 Bad utility file type.

SEE ALSO
boot(1M), pdc(1M).
NAME
  itemap - load an ITE (Internal Terminal Emulator) keyboard mapping.

SYNOPSIS
  itemap [options]

DESCRIPTION
  The itemap command loads a keyboard mapping into the ITE (the graphics console driver), or displays
ITE keyboard mappings. itemap is run by /etc/bcheckrc automatically. It is not usually explicitly
invoked by the user.

Options
  -d name
  -d keyboard_ID
  Dump a keymap to standard output in hexadecimal notation.
  -h
  Load the specified keymap into the kernel mapping table used for HP-HIL keyboards.
  -i
  Interactively prompt for a PS2 DIN keyboard mapping. itemap scans the keymap
database file for all mapping names beginning with a PS2_DIN prefix. Each of these
names is displayed, and one must be selected.
  -k database_file_name
  The name of the keymap database file to be used for input. The default is
/etc/X11/XHPKeymaps.
  -L
  Load the appropriate keymap. itemap scans the hardware for a keyboard, determines
the language of that keyboard, and loads the keymap corresponding to that keyboard.
  Because itemap cannot determine the language of PS2 DIN keyboards, use the -i
option when using -L with PS2 DIN keyboards.
  -l name
  -l keyboard_ID
  Load a specified keyboard map. Once loaded, ITE uses the specified mapping.
  When loading a keyboard mapping with the -l option, itemap matches the suffix of the
name of the specified keyboard mapping with those found in /etc/X11/XHPKeymaps
to determine the keyboard language. This information is used by the ITE to perform ISO
7-to-8 bit conversion. Keymap names added by users, via
/usr/contrib/bin/X11/keymap_ed
should use the same suffixes as those already used in /etc/X11/XHPKeymaps. For
example, a French keyboard mapping can be named New_French, for consistency with existing ITF_French and PS2_French mappings. A mapping called NewStuff
would not match any suffix patterns found by itemap, and would result in incorrect ISO
7-to-8 bit conversion.
  -p
  Load the specified keymap into the kernel mapping table used for PS2 DIN keyboards.
  -v
  Perform actions verbosely.
  -w file_name
  If a keymap for a PS2 DIN keyboard is loaded, write its name to file_name.

EXAMPLES
To automatically install the correct mapping for an HP-HIL keyboard:
  itemap -L
To explicitly load the ITF_French mapping for an HP-HIL keyboard:
  itemap -h -l ITF_French
To explicitly load the PS2_DIN_French mapping for a PS2 DIN keyboard:
  itemap -p -l PS2_DIN_French
To interactively choose a PS2 DIN keyboard mapping:

```bash
itemap -Li
```

To generate a list of the available keyboard mappings:

```bash
/usr/contrib/bin/X11/keymap_ed -l
```

### FILES

- `/usr/contrib/bin/X11/keymap_ed`  
  Keymap database editor
- `/etc/X11/XHPKeymaps`  
  System keymap database
- `/etc/kbdlang`  
  Contains mapping name configured for PS2 DIN keyboards

### SEE ALSO

ps2(7), termio(7), keymap_ed(1X111).
NAME
kclog - manage kernel configuration log file

SYNOPSIS
kclog [-a] [-c config] [-f string] [-t type] [-n name] [count]

kclog -C comment

DESCRIPTION
kclog is the administrative command for the HP-UX kernel configuration log file. The log file is automatically maintained by all of the kernel configuration commands (kconfig(1M), kmodule(1M), and kctune(1M)). Any change to any kernel configuration gets logged to this log file, which is located at /var/adm/kc.log. Note that this file is a plain text file which can be viewed and manipulated using standard Unix file management commands; kclog exists simply for convenience in finding particular log file entries.

Under normal usage, kclog prints the last count entries in the log file. When one of the options is specified, kclog prints the last count entries that match the specified criteria. If count is not specified, it defaults to 1.

Options
- a kclog will print all entries matching criteria. If this option is not specified, kclog will only print the last count entries that match the specified criteria.
- c config kclog will print only log file entries describing changes to the saved kernel configuration named config. If this option is not specified, kclog will print log file entries describing changes to any saved or running kernel configuration.
- C comment kclog will not print any entries. Instead, kclog will create a new entry, as if a kernel configuration change had been made, containing the specified comment. Super-user permissions are required for this option.
- f string kclog will print only log file entries that contain the given string.
- n name kclog will print only log file entries that refer to a configuration object (module or tunable) of the given name.
- t type kclog will print only log file entries that refer to configuration objects of the specified type: module or tunable.

RETURN VALUE
kclog returns zero for success. It returns non-zero and prints a diagnostic message if an error occurs.

EXAMPLES
To see the last three entries in the log:
$ kclog 3
To see the last entry for the tunable file_sys_max:
$ kclog -t tunable -n file_sys_max
To see the last five entries for module changes:
$ kclog -t module 5
To see all entries for module changes:
$ kclog -a -t module 5
To see the last entry mentioning Aberdeen:
$ kclog -f Aberdeen

WARNINGS
The format of the log file may be changed without notice.

Some configuration changes can be made without using the kernel configuration commands. No log file entries are made for such changes.

The log file should not be manually edited. Doing so may cause kclog to behave unpredictably.
SEE ALSO
kconfig(5), kmodule(1M), kconfig(1M), ktune(1M).
NAME
kmodule - manage kernel modules and subsystems

SYNOPSIS
kmodule [-adhvBDKS] [-c config] [-C comment] [-P fields]
[module=[unused | static | loaded | auto | best | uninstall]]

DESCRIPTION
kmodule is the administrative command for HP-UX kernel modules. It gives information about kernel
modules and their usage, and makes changes to their usage.

This command can work with any saved kernel configuration, or with the currently running kernel
configuration, depending on the use of the -c flag (see below). By default, changes to the currently run-
ning kernel configuration are applied immediately. Some changes cannot be applied without a reboot; if
any such changes are requested, or the -h flag is given, all changes on the kmodule command line will
be held until next boot.

Super-user permissions are required when making changes to module usage.

Options
-a Include all modules in the output listing. Normally only "interesting" modules are listed:
required modules and container (library) modules are omitted, as are multiple versions of a
module. Not valid in combination with -D or -S.

-b Forces a backup of the currently running configuration before making the requested
change. For more information on backup configurations, see kconfig(5). Not valid in combi-
nation with -c.

-c config Tells kmodule to view or change the saved kernel configuration named config. If this
option is not specified, kmodule views or changes the currently running kernel
configuration.

See kconfig(5) for more information on saved kernel configurations.

-C comment The specified comment will be included in the kernel configuration log file entry made for
this invocation of kmodule. For more details on the kernel configuration log file, see
kclog(1M). Note that it will usually be necessary to quote the comment in order to avoid
interpretation by the shell.

-d Adds the description of each module to the output.

-D Restricts the output to only those modules that have a state change being held for next
reboot. kmodule will return 1 if there are any such modules; see RETURN VALUE
below. Not valid in combination with -a, -c, or -S.

-h Changes will be held until next boot, even if they could be applied immediately. Not valid
in combination with -c.

-K Prevents a backup of the currently running configuration before making the requested
change. For more information on backup configurations, see kconfig(5). Not valid in combi-
nation with -c.

-P fields Tells kmodule to include only the specified fields in its output, and to print them in the
machine-readable form described in kconfig(5). See the Developers Note, below. Not valid in
combination with -d or -v.

-S Only modules in non-default states are included in the output. In other words, the listing
includes only optional modules that are in use by explicit request. It does not include
unused modules, required modules, or modules that were automatically selected to resolve
dependencies. Not valid in combination with -a or -D.

-v Print verbose information about each module. The information includes the name, version
and state of the module, its allowed states and its dependencies on other modules and inter-
faces. Not valid in combination with -d or -P.
Arguments
The arguments to kcmodule may be any mixture of module state queries and assignments. The arguments must each take one of the forms listed below. No spaces are permitted within each argument. If no arguments are given, kcmodule performs a query on all modules (subject to the constraints of the -a, -D, or -S flags).

module The state of the module will be reported. No change is made.
module= The module will be put into its best state.
module=state The module will be put into the specified state. The possible states are:
- unused The module is not used in any way.
- static The module is statically bound into the kernel executable.
- auto The module will be dynamically loaded into the kernel when something tries to use it.
- loaded The module is dynamically loaded into the kernel.
- best The module will be put into the state identified by the kernel module developer as its "best" state. Typically this will be auto, if supported by the module, otherwise loaded, if supported by the module, otherwise static. Note that a module in best state will inherit any changes that HP makes to the "best" state for a module, in a patch or a future release of HP-UX.
- uninstall The module will be put into the unused state. In addition, all of the module's tunable settings and other associated configuration data will be purged from the configuration. This state should be specified only when a module is being physically removed from the system.

Some modules do not support all of the possible states. To see which states a module supports, run kcmodule -v modulename.

Moving modules into or out of the static state requires a kernel relink, so such changes cannot be applied without a system reboot. Other module state changes may also require a system reboot, depending on the nature of the specified module.

Moving a module from loaded to auto has no effect on the currently running system; the module remains loaded. It will be autoloaded on first use after future reboots.

Developer's Note
The layout and content of kcmodule's output may change without notice, except when -P fields is specified. Scripts or applications that need to parse the output of kcmodule are expected to use the -P fields option. See kconfig(5) for details.

The fields supported in a kcmodule request are:
- name The name of the module.
- alias This field will produce a line in the output for each alternate name for the module. (There may be zero such lines.)
- desc A short description of the module.
- version The version number of the module, if it has one; otherwise, this field will be omitted from the output.
- state The state of the module. The states are listed in the table under -s, above.
- cause This field indicates how the module got into its current state. It will have one of the following values:
  - explicit The module was explicitly put into its current state by the administrator.
  - auto The module was put in auto state by the administrator. An attempt was made to use the module, so it has been automatically loaded.
The module inherited its state from another module that depends on it.

required The module is in use because it is marked required.

best The module is in this state because it is the “best” state for this module as specified by the module developer.

next_state The state of the module at next boot. This field is present only if -c is not specified.

next_cause This field indicates how the module was given its state for next boot. It has the same values as cause, above. This field is present only if -c is not specified.

before_state The state of the module before the current change. This field is present only for modules for which an immediate value change has been made during the current invocation of kcmodule.

before_cause The cause of the module state before the current change. This field is present only for modules for which an immediate value change has been made during the current invocation of kcmodule.

capable This field will contain a space-separated list of the states that this module can support. The states are listed in the table under “Arguments”, above.

depend This field will produce a line in the output for each dependency this module has on another module or interface. (There may be zero such lines.) Each line has the form:

depend type name:version

where type is either interface or module, indicating the type of object; name is the name of the interface or module; and version is the version number of the interface or module on which this module depends.

exports This field will produce a line in the output for each interface exported by this module. (There may be zero such lines.) Each line will contain the interfacename:interfaceversion of an interface exported by this module.

The special field name ALL may be specified to indicate that all defined fields should be included in the output. The output may include fields not listed in this man page. The fields will be listed in unspecified order.

Additional fields may be added in future releases or patches.

Default Output
When kcmodule is called with no options, it shows the optional kernel modules on the system, their current state, the cause for including it in the configuration and special capabilities if any. If there are changes that are being held for nextboot, they will be shown as well. The cause field will be empty for all modules that are not included in the configuration. The special capabilities of kernel modules would be one of:

loadable The module can be dynamically changed to the state loaded.

unloadable The module can be dynamically changed to the state unused.

auto-loadable The module supports the state auto.

The layout and content of the default output may change in future releases or patches of HP-UX. Scripts or applications which need to parse the output of kcmodule must use the -P option for parsable output.

RETURN VALUE
kcmodule returns one of the following values:

0 kcmodule was successful. If -D was specified, this return value indicates that there are no module state changes being held for next boot.

1 kcmodule was successful. However, there were changes requested to the currently running system which cannot be applied until the system reboots. Therefore, all of the requested changes are being held until next boot.
If -D was specified, this return value indicates that there are module state changes being held for next boot.

2  kcmdmodule was not successful.

EXAMPLES
To see all optional modules and their current states:
$ kcmdmodule
To see all modules, including required modules, and their current states:
$ kcmdmodule -a
To see verbose information about a module:
$ kcmdmodule -v module
To load a dynamic module:
$ kcmdmodule module=loaded
To unload a dynamic module immediately:
$ kcmdmodule module=unused
To stop using a module when the system reboots:
$ kcmdmodule -h module=unused
To bind a module into the static kernel:
$ kcmdmodule module=static

SEE ALSO
kconfig(1M)

NAME
   kconfig - manage kernel configurations

SYNOPSIS
   kconfig
       -DSw
   kconfig [-av] [-p fields] [config...]    
   kconfig -c [-C comment] src dest       
   kconfig -d [-C comment] config         
   kconfig -e [-C comment] [config] filename  
   kconfig -H [-C comment]                 
   kconfig -i [-C comment] [-f BKV] [config] filename  
   kconfig -l [-C comment] [-f BK] config  
   kconfig -r [-C comment] old new       
   kconfig -s [-C comment] [-f] config filename  
   kconfig -t [-C comment] config title

DESCRIPTION
   kconfig is the administrative command for HP-UX kernel configurations. In the first form, kconfig will give summary information about the currently running kernel configuration. In the second form, kconfig will give information about saved kernel configurations. (If configuration names are listed on the command line, the output is restricted to those configurations.) In the other forms, kconfig supports various administrative tasks on kernel configurations. See kconfig(5) for more information about kernel configurations.

Super-user permissions are required when specifying the -c, -d, -H, -i, -l, -n, -N, -r, -s or -t options.

Options
-     Includes detailed information about each kernel configuration in the output listing (see -v, below). In addition, the information about each configuration will be followed by the outputs of:
   kcmodule -a -v
   kctune -v
   for the configuration, so as to include all configuration data. (Note: if additional configuration data are available in future releases of HP-UX, they will be included as well.)

-B     Forces a backup of the currently running configuration before making the requested change. For more information on backup configurations, see kconfig(5).

-c src dest
   kconfig will make a copy of the saved kernel configuration named src. The copy will be named dest. dest may not already exist.

-C comment
   The specified comment will be included in the kernel configuration log file entry made for this invocation of kconfig. For more details on the kernel configuration log file, see kclog(1M). Note that it will usually be necessary to quote the comment in order to avoid interpretation by the shell.

-d config
   kconfig will delete the saved kernel configuration named config, and any files associated with it. (Some files are shared between saved configurations; these files are not deleted until all of the saved configurations using them have been deleted.)

-D
   kconfig will display all changes to the current kernel configuration that are being held for next boot. In other words, it will execute
   kcmodule -D
   kctune -D
   (If any additional kernel configuration data become available in future HP-UX releases, changes to them will be included as well.) The return value of kconfig will be 1 if there are any such changes; see RETURN VALUE, below. Not valid in combination with any other flags.

-e [config] filename
   kconfig will export the saved kernel configuration named config to a system file named filename, replacing the file if it already exists. If no config is specified, the currently
running kernel configuration will be exported, including any changes to it that are being held for next boot. The exported file can be later imported using `kconfig -i` on this system or any other system with compatible HP-UX software. See `kconfig(5)` and `system(4)` for more information about system files.

```
-kconfig(1M) kconfig(1M)
```

- `f` 
  Tells `kconfig` to proceed with the requested operation despite its potentially dangerous nature. (When `kconfig` is asked to make a change that could result in unintended data loss, it will ask for confirmation if being run interactively, or print an error message and stop otherwise. The change can be made, and the interactive confirmation bypassed, by running the `kconfig` command with the `-f` option.)

- `h` 
  Tells `kconfig` to hold the requested changes until the next boot, even if they could be applied immediately.

- `H` 
  Discards all changes being held pending for next boot. The currently running configuration will be used at next boot.

```
-i [config] filename
```

`kconfig` will import a kernel configuration from the system file named `filename`, which may have been created by an invocation of `kconfig -e` (on any system). If `config` is specified, the resulting kernel configuration will be saved under that name. Otherwise, the resulting kernel configuration will be applied to the currently running system, overwriting any changes that were being held for next boot. (The new configuration changes will be held until next boot if they cannot be applied immediately, or if the `-h` option is specified.)

A kernel configuration may not be imported successfully unless the running system has all of the necessary kernel software installed. If the importing system is missing kernel software components that were in use on the exporting system, the import will fail. See the `-V` option, below.

```
-K
```

Prevents a backup of the currently running configuration before making the requested change. For more information on backup configurations, see `kconfig(5)`.

```
-l config
```

`kconfig` will load the saved kernel configuration named `config`, overwriting any changes that were being held for next boot. The state of the currently running system will be changed to match the saved configuration. (If the changes cannot be applied without a reboot, they will be marked to take effect at next boot.)

```
-n config
```

The saved kernel configuration named `config` will be marked for use at next boot. Any changes to the currently running configuration that are being held for next boot are discarded.

```
-P fields
```

Tells `kconfig` to include only the specified `fields` in its output, and to print them in the machine-readable form described in `kconfig(5)`. See the `Developers Note`, below. Not valid in combination with `-v`.

```
-r old new
```

The saved kernel configuration named `old` will be renamed `new`. The name `new` must not already be in use.

```
-s config
```

`kconfig` will save the running kernel configuration under the name `config`.

```
-S
```

`kconfig` will display all settings of the currently running configuration that are not at their default value. In other words, it will execute

```
kcmodule -S
kctune -S
```

If any additional kernel configuration data become available in future HP-UX releases, settings of them will be included as well.) Not valid in combination with any other flags.

```
-t config title
```

`kconfig` will change the title of the saved kernel configuration named `config` to the given `title`. `title` will usually need to be quoted to avoid interpretation by the shell.

```
-v
```

Includes verbose information about each saved kernel configuration in the output listing. The information includes the name and title of the configuration, the path name of the associated kernel, the creation date and time, etc.

```
-V
```

Requests strict version checking. With this option, an import operation will only succeed if the kernel components installed on the importing system are the exact same versions as those installed on the exporting system.
Prints information about the original source of the currently running configuration. Specifically, it describes the source of the most recent complete kernel configuration operation (load, boot, or import). Also, if a saved configuration is marked for use at next boot, this option will identify it.

Note that this source information does not imply that the currently running configuration matches its original source. `kconfig -w` may say that the currently running configuration was loaded from the saved configuration `foo`, but that does not mean that the two are identical. Either one may have been changed since the load occurred.

Developer's Note

The layout and content of `kconfig`'s output may change without notice, except when `-P fields` is specified. Scripts or applications that need to parse the output of `kconfig` are expected to use the `-P fields` option. See `kconfig(5)` for details.

The fields supported in a `kconfig` request are:

- `name` The name of the saved kernel configuration.
- `title` The title for the saved kernel configuration. If the configuration does not have a title, this field will be empty.
- `savetime` The date and time of the last save (`kconfig -s`) of the configuration, in the format returned by `ctime(3)`.
- `modifytime` The date and time of the last change to the saved configuration, in the format returned by `ctime(3)`.
- `kernel` The pathname of the corresponding kernel directory.

The special field name `ALL` may be specified to indicate that all defined fields should be included in the output. The output may include fields not listed in this man page. The fields will be listed in unspecified order.

Additional fields may be added in future releases or in patches.

RETURN VALUE

`kconfig` returns one of the following values:

- `0` The requested operation was successful. If `-D` was specified, this return value indicates that there are no configuration changes being held for next boot.
- `1` The requested operation could not be performed immediately as requested, and is being held pending until the next boot. If `-D` was specified, this return value indicates that there are configuration changes being held for next boot.
- `2` The requested operation failed. A diagnostic message is printed.

EXAMPLES

To save the current kernel configuration to "myconfig":

```bash
$ kconfig -s myconfig
```

To delete the saved configuration "useless":

```bash
$ kconfig -d useless
```

To export the current kernel configuration:

```bash
$ kconfig -e /tmp/myconfig.system
```

To import a kernel configuration:

```bash
$ kconfig -i /tmp/myconfig.system
```

To rename a configuration:

```bash
$ kconfig -r myconfig savedconfig
```

To load the "nighttime" configuration and apply it immediately:

```bash
$ kconfig -l nighttime
```

To use the "approved" configuration at next boot:

```bash
$ kconfig -n approved
```

To discard changes being held pending for next boot:

```bash
$ kconfig -H
```
To add a description to a configuration:

$ kconfig -t approved "Changes approved by management"

To get a list of all saved configurations:

$ kconfig

To get details of a saved configuration:

$ kconfig -v myconfig

SEE ALSO

kcllog(1M), kcmodule(1M), kcpath(1M), kctune(1M), mk_kernel(1M), system(4), kconfig(5).
NAME
kcpath - print kernel configuration pathnames

SYNOPSIS
kcpath [-b]
kcpath -d [config]
kcpath -x [config]

DESCRIPTION
kcpath prints the pathnames associated with a kernel configuration. If config is specified, kcpath gives
information about the saved kernel configuration named config. Otherwise, kcpath gives information
about the currently running configuration. See kconfig(5) for information on saved kernel configurations.

Options
-b kcpath will print the basename of the currently running kernel.
-d [config] kcpath will print the full pathname of the directory containing the configuration.
-x [config] kcpath will print the full pathname of the kernel executable for the configuration.
If no options are specified, kcpath acts as if -b was specified.

RETURN VALUE
kcpath returns zero for success. It returns non-zero and prints a diagnostic message if an error occurs.

EXAMPLES
Print the basename of the currently running kernel:
$ kcpath -b
vmunix

Print the directory containing the currently running kernel:
$ kcpath -d
/stand/current

Print the pathname of the currently running kernel executable:
$ kcpath -x
/stand/current/vmunix

Print the directory containing the kernel for "str1013":
$ kcpath -d str1013
/stand/str1013

Print the pathname of the kernel executable for "str1013":
$ kcpath -x str1013
/stand/str1013/vmunix

SEE ALSO
kconfig(5).
NAME
kctune - manage kernel tunable parameters

SYNOPSIS
kctune [-dghuvBDKS] [-c config] [-C comment] [-P fields] [args...]

args are:
tunable (query tunable)
tunable= (set tunable to default)
tunable=Default (set tunable to default)
tunable=expr (set tunable to expression)
tunable+=value (increase tunable by value)
tunable>=value (set tunable to value if currently less)

DESCRIPTION
kctune is the administrative command for HP-UX kernel tunable parameters. It gives information about
 tunable parameters and their values, and makes changes to tunable values.

This command can work with any saved kernel configuration, or with the currently running kernel
 configuration, depending on the use of the -c flag (see below). By default, changes to the currently run-
 ning kernel configuration are applied immediately. Some changes cannot be applied without a reboot; if
 any such changes are requested, or the -h flag is given, all changes on the kctune command line will be
 held until next boot.

Super-user permissions are required when making changes to tunable values.

Options
-B Forces a backup of the currently running configuration before making the requested
 change. For more information on backup configurations, see kconfig(5). Not valid in combi-
 nation with -c.

-c config kctune will view or change tunables in the saved kernel configuration named config. If
 this option is not specified, kctune will view or change tunables in the currently running
 kernel configuration.

See kconfig(5) for more information on saved kernel configurations.

-C comment The specified comment will be included in the kernel configuration log file entry made for
 this invocation of kctune. For more details on the kernel configuration log file, see
 kclog(1M). Note that it will usually be necessary to quote the comment in order to avoid
 interpretation by the shell.

-d Adds the description of each tunable to the output.

-D Restricts output to only those parameters which have changes being held until next boot.
 kctune will return 1 if there are any such parameters; see RETURN VALUE below. Not valid
 in combination with -c.

-g Group related tunables. The tunables in the output will be grouped and sorted by the ker-
 nel modules that define them. (Note: the set of tunables defined by each kernel module
 may change in future releases of HP-UX.)

-h Changes will be held until next boot, even if they could be applied immediately. Not valid
 in combination with -c.

-K Prevents a backup of the currently running configuration before making the requested
 change. For more information on backup configurations, see kconfig(5). Not valid in combina-
 tion with -c.

-P fields Tells kctune to include only the specified fields in its output, and to print them in the
 machine-readable form described in kconfig(5). See the Developers Note, below. Not valid
 in combination with -d, -g, or -v.

-S Only tunables with non-default values will be included in the output. Not valid in combina-
 tion with -D.

-u Allow the creation of user-defined tunables. Normally, kctune will not accept any tunable
 name that does not identify an existing tunable. When the -u option is given, kctune
 will accept an assignment to an unrecognized tunable name and use it to define a new

432 Hewlett-Packard Company
user-defined tunable. This tunable will not directly affect the kernel, but may be used in expressions defining the values of other tunables. The \texttt{-u} option is not needed when changing the value of a user-defined tunable; it is needed only to create one.

To remove a user-defined tunable, set it to \texttt{default}. A user-defined tunable may not be removed if it is used in the expressions defining any other tunable values.

\texttt{-v}

Includes verbose information about the tunable parameters in the output listing. The information includes the name and value of the tunable, a short description, its allowed values, its dependencies on other tunables and restrictions on when the tunable values can be changed. Not valid in combination with \texttt{-d}, \texttt{-g}, or \texttt{-P}.

**Arguments**

The arguments to \texttt{kctune} may be any mixture of tunable queries and assignments. The arguments must each take one of the forms listed below. No spaces are permitted within each argument. If no arguments are given, \texttt{kctune} performs a query on all tunables (subject to the constraints of the \texttt{-D} or \texttt{-S} flags).

- \texttt{tunable} The value of the tunable will be reported. No change is made.
- \texttt{tunable=} The tunable will be set to its default state.
- \texttt{tunable=Default} The tunable will be set to its default state.
- \texttt{tunable=expr} The tunable will be set to the specified expression. \texttt{expr} must be an integer expression following the expression syntax of the C programming language. Like in the C programming language, expression evaluation is subject to rollover, overflow and underflow. Setting unsigned tunables to an expression that evaluates to a negative quantity will have unpredictable results.

The expression may make use of the names of other tunables. (Some tunables cannot be used in expressions.) Tunable names may be specified in all upper case letters, for backward compatibility, but this usage is deprecated and will be removed in a future release. Note that the argument may need to be quoted to avoid interpretation by the shell.

Expressions are evaluated only when one of the KC commands is running. If a tunable's value changes under other circumstances, these expressions are not re-evaluated and the tunable values dependent on those expressions are not updated. For example, this can happen when tunables are changed using a direct call to \texttt{settune(2)} or \texttt{settune_txn(2)}, or when a tunable is reset to its default value during boot because of a validation failure.

- \texttt{tunable+=value} The tunable's value will be increased by \texttt{value}. \texttt{value} must be an integer constant (not an expression). C syntax for octal and hexadecimal constants is supported.
- \texttt{tunable>=value} The tunable's value will be set to \texttt{value}, unless it is already greater. \texttt{value} must be an integer constant (not an expression). C syntax for octal and hexadecimal constants is supported. Note that the argument will probably need to be quoted to avoid interpretation by the shell.

**Default State for Tunables**

The default value for a tunable is not necessarily fixed. Default values can change between HP-UX releases, or in patches. Some tunables have default values that are re-computed at boot time, or when there is a change to the hardware configuration of the system. Some tunables change their default values in response to changing system workloads. (These tunables are called “Automatic” tunables and are marked in \texttt{kctune} output.)

When a tunable is set to \texttt{Default}, its value is controlled by the HP-UX kernel, and will be changed whenever the default value for the tunable is re-computed. (Specific behavior of each tunable is described in each tunable's man page.) HP recommends that all tunables be set to \texttt{Default} unless the default value is known to be unsatisfactory.

Setting a tunable to its default value is not the same as setting it to \texttt{Default}. If the current default value of a tunable \texttt{example} is 4000,

\begin{verbatim}
  kctune example=4000
\end{verbatim}

will set the tunable's value to 4000 and prevent it from changing when the default value is re-computed.
kctune example=Default
will set the tunable's value to 4000, and automatically change it whenever the default value is re-computed.

Developer's Note
The layout and content of kctune's output may change without notice, except when -P fields is specified.
Scripts or applications that need to parse the output of kctune are expected to use the -P fields option.
See kconfig(5) for details.
The fields supported in a kctune request are:

name  The name of the tunable.
module The name of the module supplying the tunable, if any; otherwise, this field will be
omitted from the output.
desc  A short description of the tunable.
defvalue The default value of the tunable, if known. (For saved configurations, the default
values of some tunables may not be known until the configuration is in use.)
bootvalue The current value of the tunable at the time the system last booted.
current The value of the tunable at the time the system last booted.
next_boot The value that will be used for the tunable at next boot, if known. (The next boot
value of some tunables may not be known until the boot completes.) If the value
was specified using an expression, this field contains the result of evaluating
that expression. If the tunable is being automatically tuned, this field contains the
value the kernel is currently using.
before The value that was in use for the tunable before the change that was just made, if
known. (For saved configurations, the previous value of some tunables may not be
known.) If the value was specified using an expression, this field contains the result of
evaluating that expression. This field is present only for tunables for which an
immediate value change has been made during the current invocation of kctune.
expr  The expression used to set the value of the tunable, if any. If the tunable is set to
default, this field contains the word Default.
next_expr The expression for the tunable's value at next boot, if any. If the tunable is set to
default, this field contains the word Default. If -c is specified, this field is omitted
from the output.
before_expr The expression for the tunable's value before the change that was just made, if any. If the
tunable was set to default, this field contains the word Default. This field is
present only for tunables for which an immediate value change has been made during
the current invocation of kctune.
min  The minimum value of the tunable. This is an absolute minimum; the currently run-
ning system may not be able to support values this low. If there is no minimum
value, this field will be omitted from the output.
max  The maximum value of the tunable. This is an absolute maximum; the currently run-
ning system may not be able to support values this high. If there is no maximum
value, this field will be omitted from the output.
dynamic This field contains a 'y' if the tunable can be changed without a reboot, or an 'n' other-
wise.
canauto This field contains a 'y' if the tunable is capable of being automatically tuned, or an 'n'
otherwise.
auto_default This field contains a 'y' if the tunable's default value is automatically computed by the
system (and can therefore change over time), or an 'n' otherwise.
default This field contains a 'y' if the tunable is set to its default value, or an 'n' otherwise. If
all three of default, auto_default, and canauto are 'y', the tunable is being
automatically tuned.
**kctune(1M)**

This field contains a ‘y’ if the tunable is set to its default value at next boot, or an ‘n’ otherwise. This field is not printed for saved configurations.

**before_default**

This field contains a ‘y’ if the tunable was set to its default value before this invocation of kctune, or an ‘n’ otherwise. This field is present only for tunables for which an immediate value change has been made during the current invocation of kctune.

**signed**

This field contains a ‘y’ if the tunable values should be treated as signed integers, or an ‘n’ otherwise.

**flags**

This field contains a ‘y’ if the tunable values should be treated as signed integers, or an ‘n’ otherwise.

The special field name **ALL** may be specified to indicate that all defined fields should be included in the output. The output may include fields not listed in this man page. The fields will be listed in unspecified order.

Additional fields may be added in future releases or patches.

### Default Output

When kctune is called with no options, it shows all tunables associated with the kernel modules (as well as the user-defined tunables), their current values, expressions used to compute those values, and when changes can be made to these tunables. If there are changes that are being held for nextboot, they will be shown as well.

On a typical system, the expression for most tunables are marked **Default** meaning that the administrator is allowing the system to choose the tunable value. The changes column shows the restrictions on when the tunable value can be changes. Tunables whose value can be changed immediately are marked **Immed**. Tunables whose values are being automatically tuned by the system are marked **Auto**. If the administrator has disabled the automatic tuning by the system the tunable is marked **Imm (auto disabled)**. The tunables which have nothing in the changes column can only be changed with a reboot.

The layout and content of the default output may change in future releases or patches of HP-UX. Scripts or applications which need to parse the output of kctune must use the **-P** option for parsable output.

### RETURN VALUE

kctune returns one of the following values:

- **0**: kctune was successful. If **-D** was specified, this return value indicates that there are no tunable changes being held for next boot.

- **1**: kctune was successful. However, there were changes requested to the currently running system which cannot be applied until the system reboots. Therefore, all of the requested changes are being held until next boot.

  - If **-D** was specified, this return value indicates that there are tunable changes being held for next boot.

- **2**: kctune was not successful.

### WARNINGS

kctune always checks the validity of tunable values before applying them to the running system. When tunable value changes are held for next boot, or made to a saved configuration, some of the validity checks are not performed until the changed configuration is booted. If any tunable values are found to be invalid, messages will be printed to the system console during the boot process, and the default values for any affected tunables will be used instead.

### EXAMPLES

To see all tunables and their current values:

```
$ kctune
```

To see which tunables have new values being held until next boot:

```
$ kctune -D
```

To see verbose information about a tunable:

```
$ kctune -v tunablename
```

---

**HP-UX 11i Version 2: December 2007 Update**

Hewlett-Packard Company 435
To set a tunable value on the running system:
$ kctune tunable=12
To set a tunable value to be used when the system reboots:
$ kctune -h tunable=12
To increase a tunable's value by 100:
$ kctune tunable+=100

SEE ALSO
kcllog(1M), gettune(2), settune(2), settune_txn(2), tuneinfo2(2), kconfig(5).
Managing Kernel Configurations White Paper available on:
keyenvoy(1M)

NAME
keyenvoy - talk to keyserver

SYNOPSIS
keyenvoy

Remarks
The Network Information Service (NIS) was formerly known as Yellow Pages (yp). Although the name has changed, the functionality of the service remains the same.

DESCRIPTION
keyenvoy is a setuid root process that is used by some RPC programs to intermediate between a user process and the keyserv process, keyserv(1M), which will not talk to anything but a root process.

This program cannot be run interactively.

AUTHOR
keyenvoy was developed by Sun Microsystems, Inc.

SEE ALSO
keyserv(1M).

k
NAME
keyserv - server for storing private encryption keys

SYNOPSIS
keyserv [-d] [-D] [-n]

DESCRIPTION
keyserv is a daemon that is used for storing the private encryption keys of each user logged into the system. These encryption keys are used for accessing secure network services such as NIS+.

Normally, root's key is read from the file /etc/.rootkey when the daemon is started. This is useful during power-fail reboots when no one is around to type a password.

Options
-`d` Disable the use of default keys for nobody.
-`D` Run in debugging mode and log all requests to keyserv.
-`n` Root's secret key is not read from /etc/.rootkey. Instead, keyserv prompts the user for the password to decrypt root's key stored in the publickey database and then stores the decrypted key in /etc/.rootkey for future use. This option is useful if the /etc/.rootkey file ever gets out of date or corrupted.

AUTHOR
keyserv was developed by Sun Microsystems, Inc.

FILES
/etc/.rootkey

SEE ALSO
keylogin(1), keylogout(1), publickey(4).
NAME
  killall - kill all active processes

SYNOPSIS
/usr/sbin/killall [signal]

DESCRIPTION
  killall is a procedure used by /usr/sbin/shutdown to kill all active processes not directly related to the shutdown procedure.
  killall is chiefly used to terminate all processes with open files so that the mounted file systems are no longer busy and can be unmounted. killall sends the specified signal to all user processes in the system, with the following exceptions:

  the init process;
  all processes (including background processes) associated with the terminal from which killall was invoked;
  any ps -ef process, if owned by root;
  any sed -e process, if owned by root;
  any shutdown process;
  any killall process;
  any /sbin/rc process.

  killall obtains its process information from ps, and therefore may not be able to perfectly identify which processes to signal (see ps(1)).
  If no signal is specified, a default of 9 (kill) is used.
  killall is invoked automatically by shutdown. The use of shutdown is recommended over using killall by itself (see shutdown(1M)).

FILES
/usr/sbin/shutdown

SEE ALSO
  kill(1), ps(1), fuser(1M), shutdown(1M), signal(5).

STANDARDS CONFORMANCE
  killall: SVID2, SVID3
kllsmd(1M)

NAME
kllsm - kill the sendmail daemon

SYNOPSIS
/usr/sbin/kllsm

DESCRIPTION
kllsm reads the pid file to get the pid number of the currently running sendmail daemon, and then kills that daemon. The pid file is specified using the PidFile option in the /etc/mail/sendmail.cf file. If this option is not set, sendmail uses /etc/mail/sendmail.pid file as the default pid file.

If the sendmail daemon is not killed instantaneously, kllsm will not exit until the sendmail daemon is terminated or after a time interval of 10 seconds.

The “/sbin/init.d/sendmail stop” command does the same thing as kllsm. HP recommends that system administrators use “/sbin/init.d/sendmail start” and “/sbin/init.d/sendmail stop” to start and stop sendmail; these startup scripts are used when the system is booting to start sendmail. Advanced system administrators can put /usr/sbin into their search path and just reference “sendmail -bd -q30m” to start sendmail, and kllsm to stop it.

SEE ALSO
sendmail(1M).
NAME
kl - control kernel logging

SYNOPSIS
/usr/sbin/kl -e [-q qsize] [-s ssize] [-w [on|off]] [-l [d|e|w|i] [subsys_name...] | all]
/usr/sbin/kl -d
/usr/sbin/kl -l [d|e|w|i] [subsys_name...] | all]
/usr/sbin/kl -w [on | -s ssize | | off]
/usr/sbin/kl -s ssize
/usr/sbin/kl -i
/usr/sbin/kl -p filename [-w on [-s ssize]]
/usr/sbin/kl -q qsize

DESCRIPTION
The kl command controls the operation of the Kernel Logging facility. Kernel Logging is a high-
availability feature that gives system administrators the ability to collect the information necessary to diag-
nose problems with the HP-UX kernel while the system is running. kl is used to specify the levels of
events to be logged and the kernel subsystems that will write messages to memory or disk. kl also pro-
vides for managing the contents of the log file in memory and on disk.

At startup, Kernel Logging determines its default configuration by reading the file
/etc/nettlgen.conf. See nettlgen.conf(4) for an explanation of the file format. The kl command
permits only temporary changes to the default Kernel Logging configuration without having to stop and
restart Kernel Logging facility. Note that any values you specify on the kl command line do not modify
the contents of the /etc/nettlgen.conf file. To make permanent changes to the values in the
/etc/nettlgen.conf file, run nettlconf(1M).

Only users with appropriate privileges (root) can invoke the kl command to control the Kernel Logging
facility.

Options
kl recognizes the following options, which can be used only in the combinations indicated in the
SYNOPSIS section. All options and keywords are case-sensitive.
-e Enable the Kernel Logging facility and start up default logging as defined in the file
/etc/nettlgen.conf.
If the -l option is used in conjunction with the -e option, the -l option must be specified
as the last option on the command line.
-d Disable the Kernel Logging facility. Once this option is issued, Kernel Logging stops
accepting logging calls from the kernel subsystems.
-w [on|off] Turn on/off write-to-disk logging. If write-to-disk logging is enabled, log messages residing
on the log queue in memory are written to disk and removed from the queue.
The name of the log file on disk is formed by adding the suffix .KLOG0 to the log file name
specified in the /etc/nettlgen.conf file. If the log file (including suffix) already
exists, one of the following events takes place:
- If the existing file contains messages logged during the system run when a panic
occurred, the file will be preserved (in a new location). See the USAGE section for more
information on how this situation is handled.
- Otherwise, the existing file and whatever information it contains will be lost; that is, the
contents of the file are overwritten with new kernel logging data.
When write-to-disk logging is turned off, messages are not written out to disk, but continue
to be collected in memory (circular buffer).
-l [d | e | w | i] [subsys_name...] | all ]
Modify the level of log messages to be captured for the specified subsystem(s).
The subsys_name argument is a subsystem name specified in the file
/etc/nettlgen.conf. The list of available subsystem names can be obtained using

the command `nettconf -KL -status`. The keyword `all` changes the logging level for all subsystems specified in the file `/etc/nettlgen.conf`.

The following table describes the classes of messages that can be logged:

<table>
<thead>
<tr>
<th>Message Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disaster</td>
<td>Signals an event or condition which affected the operation of an entire subsystem, or the entire kernel, causing several programs to fail or the entire machine to shut down.</td>
</tr>
<tr>
<td>Error</td>
<td>Signals an event or condition which did not affect the overall operation of an entire subsystem, or the entire kernel, but may have caused an application program to fail.</td>
</tr>
<tr>
<td>Warning</td>
<td>Indicates abnormal events, possibly caused by problems in an individual subsystem.</td>
</tr>
<tr>
<td>Informative</td>
<td>Describes routine operations and current system values.</td>
</tr>
</tbody>
</table>

The following table identifies the classes of messages that are captured at each log level:

<table>
<thead>
<tr>
<th>Log Level</th>
<th>Classes of Messages</th>
</tr>
</thead>
<tbody>
<tr>
<td>d</td>
<td>Disaster</td>
</tr>
<tr>
<td>e</td>
<td>Disaster and Error</td>
</tr>
<tr>
<td>w</td>
<td>Disaster, Error and Warning</td>
</tr>
<tr>
<td>i</td>
<td>Disaster, Error, Warning and Informative</td>
</tr>
</tbody>
</table>

Note that, although the log level is specified as a single keyword, messages are logged according to the following rule: if level $x$ is specified, then all messages whose severity is greater than or equal to the severity of class $x$ will be logged. The order of severity is as follows: Disaster (the most severe), Error, Warning, Informative (the least severe).

- `-s fsize`

Set the size of the two log files used to store logged messages when write-to-disk is enabled. The maximum $fsize$ is 1 gigabyte; the minimum $fsize$ is 16 kilobytes.

When write-to-disk is started, the default log file size is taken from the file `/etc/nettlgen.conf`. The `-s` option allows you to modify the size of the log file without stopping write-to-disk operations.

Note that, when setting the file size, you can specify a numeric value followed by the character `M` or `K`, which indicates that $fsize$ is being defined in units of megabytes or kilobytes, respectively. This suffix character is case-insensitive.

- `-p filename`

Take a picture (snapshot) of the current contents in memory of the kernel logging buffer. This option causes all messages residing in the log queue to be dumped to `filename` and removed from the queue.

Note that taking a picture (snapshot) and write-to-disk are mutually exclusive operations, because there is no reason to take a picture of Kernel Logging messages if write-to-disk component already writes them to disk.

If the `-w on` option is specified in conjunction with the `-p` option, write-to-disk logging will be started immediately after the picture is taken.

- `-q qsize`

Set the size of the Kernel Logging queue. $qsize$ indicates the number of buffers of size 8 kbytes each. The maximum $qsize$ is 1024 buffers; the minimum $qsize$ is 8 buffers. At startup, this value is taken from the file `/etc/nettlgen.conf`.

- `-i`

Report information about the status of the Kernel Logging facility.

Information returned by the `-i` option includes:
- kernel logging facility is on or off.
- write-to-disk logging is on or off.
- picture (snapshot) tool is on or off.
- current size of the Kernel Logging queue.
- number of messages currently held in the queue.
- name of the log file used by write-to-disk.
maximum file size of the log file used by write-to-disk, together with the minimum file size available for the current session of write-to-disk.

- number of messages not written to file (could be due to the lack of memory or small size of the circular buffer).

- list of all subsystems currently specified in the /etc/nettlgen.conf file and the associated message class(es) logged for each subsystem.

**USAGE**

When to Use Kernel Logging

Mission critical systems should have KL always enabled. Failing to enable Kernel Logging causes diagnostic information about any suspicious events that might occur on the system to be lost. The recommended classes to capture are Disaster, Error and Warning. Use `kl -l w all` command to do so.

To minimize Kernel Logging's impact on a running system, use the `kl -l e all` command to set all kernel subsystems to capture error-level log messages only.

Log File Management

The write-to-disk facility uses two files to hold logging information. The base name of the log files is specified in /etc/nettlgen.conf. The default base log file name is /var/adm/kl; see nettlgen.conf(4). The most current data is always in the file with suffix .KLOG0. If the size of the .KLOG0 file reaches a user-defined maximum, Kernel Logging renames the .KLOG0 file to .KLOG1, overwriting the previous contents of the .KLOG1 file, then continues writing messages to the .KLOG0 file. (Specify maximum log file size in /etc/nettlgen.conf or use the -s option.)

The Kernel Logging facility has a protection feature for saving old log files. When write-to-disk starts and encounters old log files that contain messages collected during a prior run of the system when a panic occurred (thus, the log files may contain important information about the panic), then log files are not overwritten. Instead, write-to-disk first attempts to move the old log files to the default crash directory (typically /var/adm/crash/crash.ID, where ID is a numeric counter). If the move fails, then write-to-disk tries to rename the log files with the prefix OLD, allowing the files to remain in the same directory. For example, if write-to-disk finds an old log file named /var/adm/kl.KLOG0, it first tries to move the file into the default crash directory. If this move is not successful, write-to-disk then tries to move the file to /var/adm/OLDkl.KLOG0.

If both attempts to save old log files fail, the write-to-disk component is not started. To preserve the messages in the old log files, do one of the following:

- If possible, eliminate the conditions that caused both attempts to move the old log files to fail.
- Manually move old log files.
- Using the nettlconf command, modify the file /etc/nettlgen.conf to specify a different log file name for write-to-disk logging.

Once the problem is corrected, start write-to-disk again. If successful, the Kernel Logging facility will notify you about the move of the old log files.

**RETURN VALUE**

`kl` exits with one of the following values:

- 0 Operation was successful
- 1 kl command aborted due to error(s)

**EXAMPLES**

1. Enable the default Kernel Logging facility.
   ```
   kl -e
   ```

2. Display the information about the Kernel Logging facility.
   ```
   kl -i
   ```

3. Change level to log disaster, error and warning messages for all subsystems.
   ```
   kl -l w all
   ```

4. Request a picture (snapshot) of the collected log messages.
5. Turn on default write-to-disk logging.
   `kl -w on`

6. Change the maximum size of the current log file to 512KB.
   `kl -s 512K`

7. Turn off write-to-disk logging.
   `kl -w off`

8. Disable the Kernel Logging facility.
   `kl -d`

9. Enable on the Kernel Logging facility with the following parameters: queue size is 1000, write-to-disk component is on, maximum file size is 4 megabytes, and log level for the KL_PM subsystem is Warning.
   `kl -e -q 1000 -w on -s 4M -l w KL_PM`
   Note that `-l w KL_PM` changes the initial level of the KL_PM subsystem only. This means that the initial levels of the rest of the subsystems specified in the `/etc/nettlgen.conf` file remain unchanged.

10. Take a picture (snapshot) and start write-to-disk logging with a maximum file size of 128K.
    `kl -p -w on 128K`

**AUTHOR**

`kl` was developed by HP in partnership with NEC.

**FILES**

- `/etc/nettlgen.conf`: NetTL and KL subsystem configuration file.
- `/var/adm/kl.KLOG0`: Default log files as specified in `/etc/nettlgen.conf`
- `/var/adm/kl.KLOG1`

**SEE ALSO**

`netfmt(1M), nettl(1M), nettlconf(1M), nettlgen.conf(4)`
NAME
kmpath - retrieve kernel name and associated kernel configuration information

SYNOPSIS
/usr/sbin/kmpath [-k]
/usr/sbin/kmpath -c [kernel_name]
/usr/sbin/kmpath -i [kernel_name]

DESCRIPTION
The kmpath command is obsolete and superseded by the kcpath(1M) command. Users are encouraged to migrate to kcpath(1M). The kmpath command will be removed in a future release of HP-UX. The behavior of some of kmpath’s options have changed and they are noted in the appropriate sections below.

kmpath retrieves the main kernel file name and path information of the currently running kernel. kmpath also retrieves the configuration directory of the current or specified kernel.

Each kernel configuration consists of a kernel executable and a number of other files including dynamically loadable kernel modules. These files are all stored in a directory which represents the kernel configuration.

If no options are specified, kmpath returns the full path of the executable file for the kernel that was most recently booted. Note that the executable file may no longer be at that path, if changes have been made since boot.

Options
The kmpath options have the following meanings:
- k Return the base name of the executable file of the currently running kernel.
- c [kernel_name] Return the pathname of a kernel configuration directory. If kernel_name is not specified, returns the pathname of the directory for the current kernel configuration. If kernel_name is specified, returns the pathname of the directory that contains the specified kernel.
- i [kernel_name] This option is retired and will not return any output.

SEE ALSO
kcpath(1M), kconfig(5).
kmtune(1M)

NAME
kmtune - query, set, or reset system parameters

SYNOPSIS
/usr/sbin/kmtune [-d] [-l] [[-q name]...] [-S system_file]
/usr/sbin/kmtune [-u] [[-s name[=]value]...] [[-r name]...] [-c comment] [-S system_file]
/usr/sbin/kmtune [[-e value]...]

DESCRIPTION
kmtune is an obsolete command that is replaced by kctune(1M). Users are encouraged to migrate to kctune(1M). kmtune will be removed in a future release of HP-UX. The behavior of some of kmtune's options have changed and they are noted in the appropriate sections below.

kmtune is used to query, set, or reset system parameters. kmtune displays the planned values of all system parameters when used without any options or with the -l option. kmtune gathers information on the running kernel using the tuneinfo2() system call.

By default, kmtune's changes affect the currently running kernel configuration, marked to take effect at next boot. If the -u option is specified, kmtune's changes affect the current configuration, and take effect immediately. See kconfig(5) for more information on kernel configurations.

Options
The following options are recognized by kmtune:

- c text
Specify an optional comment string. All changes to the kernel parameters are logged with the date, old value, new value, user id and this comment. This option is used with -r and -s. The comment text will generally need to be quoted so that it is interpreted as a single parameter by the shell. The comment text may contain newlines. The log is written to the file /var/adm/kc.log. See kconfig(5) and kclog(1M) for more information about the kernel configuration log file.

- d
Print a brief difference report. Only parameters where the planned and current values are different are listed. If the planned value is a formula, it is NOT considered different from the current value. This option is useful to insure that only desired changes will be made on a subsequent invocation of kmtune -u. The -d option cannot be used with the -r, -s, or -u options.

- e value
Evaluate the expression given by "value". The expression is anything allowed in the -s option. The output will always be in decimal.

- l
Print a detailed report. The -l option cannot be used with the -r, -s or -u options.

- q name
Query the value of the specified system parameter.

- r name
Reset the value of the specified system parameter to the default.

- s name[=]value
Set the value of the specified system parameter. If the separator is an equal sign (=), the parameter is set to the value specified. If the separator is a plus sign (+), the parameter is incremented by the value specified. Negative values cannot be used with plus sign (+). The name[=]value format must not include spaces or tabs.

- S system_file
This option is obsolete. It is accepted for compatibility purposes but has no effect.

- u
Update the current kernel configuration. This modifies the -s and -r options so that their effects take effect immediately rather than at next boot. The whole command will fail if any non dynamic parameters are being set. The -s and -r options are processed in the order they are listed. At least one -s or -r option must be specified.

The expressions used with the -s and -e options are built of literal values, the upper case names of other kernel parameters, and the following operators: ( and ), the unary operators !, !, -, !, -, <, >, <, <=, >, >=, =, =, &&, and ||, and the ternary operator ?:

Using -s to create a circular dependency in the values of the kernel parameters is not allowed.
kmtune (1M)

Notes
Changing kernel parameters dynamically is a fairly new feature. Programs using pstat(2), getrlimit(2), or sysemon(2) may have been written with the previously correct assumption that the returned values do not vary while the system is running. The use of kmtune to modify the running kernel may cause such programs to produce erroneous results or even abort.

Some dynamic kernel parameters can be raised dynamically, but cannot be lowered without rebuilding the kernel and rebooting. See the manual pages for those parameters for details.

When dynamically lowering per-process limits, processes that exceed the new limit will be "grandfathered". Such processes will retain the old limit. The old limit will also apply to any child processes they create after the change. Some parameters have exceptions to this general policy; see the manual pages for those parameters for details.

It is possible to put multiple -s and -r options on a single kmtune command. kmtune will try to make all of the changes, in the order listed. If an error occurs with one of the changes, the state of the other changes is not defined.

RETURN VALUE
Upon completion, kmtune returns with one of the following exit values:

0 Successful.
1 Changes have been made and will take effect at next boot.
2 An error occurred.

Results of query requests are sent to stdout. Error and warning messages are sent to stderr.

FILES
/var/adm/kc.log Log of all kernel configuration changes

SEE ALSO
kclg(1M), kconfig(5), kctune(1M), settune(2), tuneinfo2(2), and the individual tunable parameter manual pages in section 5.
NAME
krs_flush - flush kernel registry services data to disk

SYNOPSIS
krs_flush

DESCRIPTION
krs_flush causes persistent, in core, kernel KRS data to be flushed to files on disk.
This command signals the KRS daemon, krsd(1M), causing it to save the KRS data to disk. The data will only be saved if it has changed from the time of the last save.

AUTHOR
krs_flush was developed by Hewlett-Packard Company.

SEE ALSO
krsd(1M), krs(5).
NAME
 krsd - kernel registry services daemon

SYNOPSIS
 krsd -l
 krsd [-i] [-d seconds]

DESCRIPTION
 krsd saves data, flagged as persistent in the core kernel KRS tree, to files on disk. These files are read
when the system is booted, in order to restore the aforementioned persistent data.

When running in daemon mode, krsd sleeps, waking periodically to check if any persistent data have
changed since the last save. If persistent data have changed, the new state of the data is saved to disk.
Otherwise, krsd sleeps for another period of time and then repeats the process.

When running in one time (non-daemon) mode, krsd unconditionally saves persistent data to disk and
then exits.

Normally, krsd is executed in daemon mode by init(1M). The /etc/inittab entry for krsd will
cause krsd to be respawned automatically if it is terminated.

krsd uses the syslog message logging facility to log all krsd activity.

Persistent KRS data are maintained in files based on the class of the data in question. Currently, the fol-
lowing classes are recognized:

  system specific
     Data that apply to the system in general, regardless of what configuration is booted. This data is
     saved in the file system.krs.

  configuration specific
     Data specific to a given bootable configuration. This data is saved in the file config.krs.

Options
 krsd recognizes the following options:

  -l  One time mode, unconditionally save persistent data to disk, then exit.

  -i  Run from inittab, parent does not spawn a child nor exit. Required if respawn from
       inittab is to work properly.

  -d seconds  Set the delay time to seconds seconds. This is the time interval between attempted saves of
              persistent data. The default interval is 300 seconds.

AUTHOR
 krsd was developed by Hewlett-Packard Company.

SEE ALSO
 krs_flush(1M), krs(5).
NAME
lanadmin - local area network administration program

SYNOPSIS
/usr/sbin/lanadmin [-e] [-t]
    
/usr/sbin/lanadmin [-a] [-A station_addr] [-b] [-B on|off] [-c] [-g [get_options]] [-m]
    [-x options] [-X options] PPA

DESCRIPTION
The lanadmin program administers and tests the Local Area Network (LAN). For each interface card, it
allows you to:

- Display and change the station address.
- Display and change the 802.5 Source Routing options (RIF).
- Display and change the maximum transmission unit (MTU).
- Display and change the speed setting.
- Clear the network statistics registers to zero.
- Display the interface statistics.
- Display the interface usage information.
- Reset the interface card, thus executing its self-test.
- Configure VLANs on the cards that support VLAN.

For operations other than display, you must have superuser privileges.

lanadmin reads commands from standard input, writes prompts and error messages to standard error,
and writes status information to standard output. When the program is run from a terminal, the interrupt
key (usually \^C) interrupts a currently executing command; the eof key (usually \^D) terminates the pro-
gram.

lanadmin operates in two modes: Menu Mode (see the first SYNOPSIS line) and Immediate Mode (see
the second SYNOPSIS line). If at least one of aAbBcgmMprRsSVxX options is supplied, lanadmin exe-
cutes in Immediate Mode. Otherwise, it executes in Menu Mode.

NOTE: lanadmin replaces the now obsolete landiag command beginning at 10.0.

Options
lanadmin recognizes the following Immediate Mode options. At least one aAbBcgmMprRsSVxX option
and the PPA argument must be supplied.

-a Display the current station address of the interface corresponding to PPA.
-A station_addr Set the new station address of the interface corresponding to PPA. The
    station_addr must be entered in hex format with a '0x' prefix. You must have
    superuser privileges. When station_addr is DEFAULT, the factory default physical
    address will be restored.

WARNING: To ensure the interface and the system work correctly, the interface
MUST be brought down before setting the new station address. After the new sta-
tion address is set, the interface should be brought up in order to be functional. See
ifconfig(1M) for bringing down and bringing up the interface.

-b Display the current 802.5 source routing option for the interface corresponding to
    PPA.
-B on|off Turn the 802.5 source routing option on or off for the interface corresponding to
    PPA. The default value for HP devices is on. You must have superuser privileges.

-c Clear the LAN interface network statistics registers to zero for the interface
    corresponding to PPA. You must have superuser privileges.

-g [get_options] Get and display settings corresponding to options specified by get_options of the
    interface corresponding to PPA. get_options is optional and is case insensitive. The
    allowed values are mibstats and mibstats_ext.

mibstats Display the MIB-II statistics (RFC 1213) and interface-specific
    statistics.
mibstats_ext  Display extended 64-bit MIB statistics (RFC 2863) and interface-specific statistics.

-m  Display the current MTU size of the interface corresponding to PPA.

-M mtu_size  Set the new MTU size of the interface corresponding to PPA. The mtu_size value must be within the link specific range. You must have superuser privileges.

-p [card|driver]  Display the usage information for the interface corresponding to PPA. Displays the upper level protocols and applications attached to the interface corresponding to PPA. If none is attached, it prints nothing. If card or driver keywords are used, the information about the aggregated ports, APA, and VLAN that are associated with the card or the driver corresponding to PPA is also displayed.

-r  Reset the local LAN interface card corresponding to PPA, causing it to execute its self-test. Local access to the network is interrupted during execution of reset. You must have superuser privileges.

-R  Reset the MTU size of the interface corresponding to PPA to the default for that link type. You must have superuser privileges.

-s  Display the current link speed setting of the interface corresponding to PPA.

-S speed  Set the new link speed setting of the interface corresponding to PPA. You must have superuser privileges.

-V vlan_command  Executes VLAN-specific commands. See vlan(7) for complete syntax and usage of -V option. The supported VLAN commands are:

create  Create a VLAN.
modify  Modify the properties of a VLAN.
delete  Delete a VLAN.
scan  Get information about all VLANs on a system.
info  Get information about a particular VLAN on a system.
basevppa  Get the minimum acceptable virtual PPA (VPPA) value.

-x options  Get and display driver specific options of the interface corresponding to PPA.

-X options  Set driver specific options of the interface corresponding to PPA. You must have superuser privileges.

lanadmin recognizes the following Menu Mode options. They are ignored if they are given with an Immediate Mode option.

-e  Echo the input commands on the output device.

-t  Suppress the display of the command menu before each command prompt. This is equivalent to the Test Selection Mode terse command. The default is verbose.

Immediate Mode
In Immediate Mode, you can display the station address, source routing option, MTU size, link speed of LAN interface, PPA/VPPA and information about VLANs on the system (see vlan(7)). For certain interfaces, if you have superuser privileges you can also modify the station address, source routing option, MTU size, and link speed. See Options and Arguments above.

Menu Mode
In Menu Mode, you can select an interface card, display statistics for the selected card, reset the card, and clear the statistics registers.

Menu Mode accepts either complete command words or unique abbreviations, and no distinction is made between uppercase and lowercase letters in commands. Multiple commands can be entered on one line if they are separated by spaces, tabs, or commas.

Test Selection Mode Menu
This menu is entered when Menu Mode is first selected. The available Test Selection Mode commands are:

lan  Select the LAN Interface Test Mode menu.
lanadmin(1M)

menu  Display the Test Selection Mode command menu.
quit  Terminate the lanadmin program.
terse Suppress the display of command menus.
verbose Restore the display of command menus.

LAN Interface Test Mode Menu

The following commands are available:
clear Clear the LAN interface network statistics registers to zero. You must have superuser privileges.
display Display the RFC 1213 MIB II statistics. Depending on the link, the type-specific MIB statistics may also be displayed. For instance, for Ethernet links, the RFC 1398 Ethernet-like statistics are displayed.
end Return lanadmin to Test Selection Mode.
menu Display the LAN Interface Test Mode command menu.
ppa Prompt for a PPA that corresponds to a LAN interface card. It defaults to the first LAN interface encountered in an internal list. Appropriate values can be displayed with the lanscan command (see lanscan(1M)).
quit Terminate the lanadmin program.
reset Reset the local LAN interface card, causing it to execute its self-test. Local access to the network is interrupted during execution of reset. You must have superuser privileges.
specific Display and execute commands from a driver specific menu.

Arguments

lanadmin recognizes the following Immediate Mode argument.

PPA The Physical Point of Attachment (PPA) number of the LAN interface. This argument is ignored if none of the aAbBcgMprRsSVxx options are used (Menu Mode). Any options specified after PPA are ignored. Appropriate values can be displayed with the lanscan command (see lanscan(1M)).

WARNINGS

Changes made to an interface's station address or mtu with the lanadmin command will not be preserved between system reboots. A user must modify the initialization configuration files for this feature, either manually editing configuration files or through the SAM interface.

AUTHOR

lanadmin was developed by HP.

SEE ALSO

netstat(1), lanscan(1M), linkloop(1M), ping(1M), lan(7), vlan(7).

NAME
lanscan - display LAN device configuration and status

SYNOPSIS
lanscan [-ailmnpqv]

DESCRIPTION
lanscan displays the following information about each LAN device and VLAN interface that have software support on the system:

- Hardware Path.
- Active Station Address (also known as Physical Address).
- Card Instance Number.
- Hardware State.
- Network Interface "NamePPA". The Network Interface "Name" and the "PPA" (Physical Point of Attachment) number are concatenated together. A single hardware device may have multiple "NamePPA" identifiers, which indicates multiple encapsulation methods may be supported on the device. For Ethernet/IEEE 802.3 links, the "Name" lan is used to designate Ethernet encapsulation, and snap for IEEE 802.3 encapsulation. For other links (FDDI, Token Ring), only the lan encapsulation designation is used.
- Network Management ID.
- MAC Type.
- HP DLPI Supported. Indicates whether or not the lan device driver will work with HP's Common Data Link Provider Interface.
- DLPI Major Number.
- Extended Station Address for those interfaces which require more than 48 bits. This is displayed only when the -v option is selected.
- Encapsulation Methods that the Network Interface supports. This is displayed only when the -v option is selected.

If the interface corresponds to VLAN, then lanscan displays VLAN# as its hardware path. Here, # is a number that uniquely identifies a VLAN interface.

Options
lanscan recognizes the following command-line options:

- -a Display station addresses only. No headings.
- -i Display interface names only. No headings.
- -l Display information about PPAs that are acquired by APA. No headings.
- -m Display MAC types only. No headings.
- -n Display Network Managements IDs only. No headings.
- -p Display PPA and/or VPPA (PPA associated with a VLAN) numbers only. No headings.
- -q Same as -p, except link aggregate PPA's will be followed by a list of LAN interface PPA's that are configured in the corresponding link aggregate. No headings.
- -v Verbose output. Multiple lines per interface. Includes displaying of extended station address and supported encapsulation methods. If the interface corresponds to VLAN, it also displays VLAN-specific information: VLAN id, PPA of physical interface on which VLAN is created, Priority, ToS, Priority Override, ToS Override, and VLAN name (see vlan(7)).

WARNINGS
lanscan does not display information about LAN devices that do not have software support such as LAN interface cards that fail to bind properly at boot-up time.

AUTHOR
lanscan was developed by HP.
SEE ALSO
    ifconfig(1M), ioscan(1M), lanadmin(1M), linkloop(1M), lan(7), vlan(7).
NAME
ldapcfinfo - programmatically provides LDAP-UX information to non-interactive applications

SYNOPSIS
ldapcfinfo -h
ldapcfinfo [-t type]
ldapcfinfo [-t type] [-A | -P | -D | -L | -b | -s | -f | -m atobName[,...]]
ldapcfinfo [-t type] -L
ldapcfinfo [-t type] -T template_file -R
ldapcfinfo [-t type] -a DN

DESCRIPTION
ldapcfinfo allows non-interactive applications to programmatically discover information about LDAP-UX configuration, including:

- If LDAP-UX is properly configured and active.
- LDAP-UX configuration profile location.
- LDAP-UX configuration profile information.
- Required attributes when creating new users or groups.

The following is a summary of ldapcfinfo commands:

ldapcfinfo [-t type]
Discover if LDAP-UX is properly configured for the specified service type.

ldapcfinfo [-t type] [-A | -P | -D | -L | -b | -s | -f | -m atobName[,...]]
Display information about the active LDAP-UX configuration profile.

ldapcfinfo [-t type] -L
Display the list of default and user defined template files.

ldapcfinfo [-t type] -T template_file -R
Discover the list of required attributes in the specified (or default) template file.

ldapcfinfo [-t type] -a DN
Discover a suggested list of modifiable attributes for the specified entry.

Options
-t type
Specifies the service name for which to retrieve configuration information.
Possible service names are: passwd, group, netgroup, services, rpc, hosts, networks, automount, automountmap, publickey, protocols, and pam.

If the -t argument is not specified, ldapcfinfo assumes the passwd name service (if applicable to the command specified). If the -t option is the only argument specified on the command line, ldapcfinfo will report if LDAP-UX is properly configured and active for the specified service.

-A
Reports if the user running the ldapcfinfo command has the ability to use the LDAP administrator's credential, if configured.

ldapcfinfo returns zero exit status if the user has rights to access the LDAP administrator's credential. ldapcfinfo returns a non-zero exit status if not.

Please refer to the section titled Configure LDAP-UX Client Services with Publickey Support in the LDAP-UX Client Services Administrator’s Guide for additional details about the LDAP-UX administrator credential. This document can be found at http://docs.hp.com/en/internet.html.

-P
Displays the distinguished name of the LDAP-UX configuration profile and LDAP server which hosts that profile. Format will be:

dn : distinguishedName
host : hostname/ip:port

If SSL or TLS is required to download the profile, host : will be replaced with hostssl :.
-R Displays the required attributes as defined in the default template file or the template file specified with the -T option.

If the -T option is not specified, then -t passwd or -t group must be specified to indicate which default template file should be examined.

Each attribute required by the requested template file will appear on separate lines, one per line. Since the RFC2307 POSIX attributes are a static known list and required, only non-posix attributes will be displayed.

-T template_file Specifies the LDIF template file used to create new user or group entries. The template_file parameter may either be a full or relative path name or a "short" name.

The -T option is ignored unless the -R option is also specified.

Refer to the ldapugadd(1M) manpage for a description about template file naming and specification of the template_file option.

-L Displays the list of available template files for the service specified with the -t option. The full path name of the template files will be displayed, each on a separate line.

-D Displays the default configuration values for the ldapugadd command. When -t passwd is specified, the uid range, default gid, default home and default shell values are displayed. When the -t group is specified, the gid range is displayed.

-b Displays the primary (first) configured search base for a particular service as defined with the -t option. If the -t option is not specified, the LDAP-UX default search base will be displayed.

Output format for the -b option will follow the format defined in RFC4514, Lightweight Directory Access Protocol (LDAP): String Representation of Distinguished Names.

-s Displays the primary (first) configured search scope for a particular service as defined with the -t option. If the -t option is not specified, the LDAP-UX default search base for passwd will be displayed.

Output format for the -s option will be either base, one, or sub, which represents the search scopes as defined in RFC4516, Lightweight Directory Access Protocol (LDAP): Uniform Resource Locator.

-f Displays the primary (first) configured search filter for the particular service defined with the -t option.

If the -t option is not specified, the passwd service will be assumed.

Output format will be an LDAP filter following the format defined by RFC4515, Lightweight Directory Access Protocol (LDAP):

-h Display brief help text.

-m atobName[, ...] Displays attribute or objectclass mapping for the requested attribute or objectclass name. atobName is either one of the RFC2307 attributes or the objectclass defined for the specific service requested.

If the requested attribute is mapped to more than one target attribute, each target attribute will be displayed on the same line, separated by white space. See example usage and output below.

Note that attribute and objectclass names are considered case-insensitive. atobName may be specified multiple times in a comma separated list. No white space should appear in the list.

-a DN Displays the recommended list of attributes that an interactive management tool should consider making available for modification for the specified entry.

Note that specification of the -t option is required in order for this operation to function properly.
Note
Since each -a, -D, -A, -P, -R, -L, -b, -f, -h, and -m options all generate varying output formats, only one of these options may be used per invocation of the ldapcfinfo command. Use of multiple of the above options in a single command line may prevent distinguishing which output applies to which option, and will result in an error.

The -T option is ignored unless the -R option is specified.

EXAMPLES
To display the attribute mapping for the gecos attribute (assuming it has been mapped to cn, l, and telephoneNumber) use:

```bash
# ldapcfinfo -t passwd -m gecos
gecos=cn l telephoneNumber
```

To display the default search base as configured by the LDAP-UX configuration profile use:

```bash
# ldapcfinfo -b
ou=example org,dc=example,dc=com
```

To display the default search base for the group name service (assuming ou=Groups has been configured as the search base for the groups name service) use:

```bash
# ldapcfinfo -t group -b
ou=Groups,ou=example org,dc=example,dc=com
```

To display the non-POSIX attributes required by ldapugadd command for the passwd name service (assuming the default file /etc/opt/ldapux/ug_templates/ug_passwd_default.tmpl) use:

```bash
# ldapcfinfo -t passwd -R
sn
```

To display the location of the LDAP-UX configuration profile use:

```bash
# ldapcfinfo -P
dn: cn=ldapux-profile,ou=example org,dc=example,dc=com
host: 10.42.222.15:389
```

To display attribute mapping for the passwd service, and assuming the uidNumber attribute has been mapped to employeeNumber and the gecos has been mapped to the three attributes, cn, l, and telephoneNumber, use:

```bash
# ldapcfinfo -t passwd -m uid,uidNumber,gecos
uid=uid
uidNumber=employeeNumber
gecos=cn l telephoneNumber
```

To display the mapped objectclass and related attributes for the publickey service, and assuming that objectclass has been mapped to pkiUser, use:

```bash
# ldapcfinfo -t publickey -m niskeyobject,nispublickey,nissetsecretkey
niskeyobject=pkiUser
nispublickey=userCertificate
nissetsecretkey="NULL"
```

Note: The above example is for demonstration only and does not imply the ability of LDAP-UX to be able to translate an X.509 userCertificate into an NIS public key.

RETURN VALUE
Upon exit, ldapcfinfo returns the following:

0 Success. ldapcfinfo exits with no errors or with one or more warnings.

<>0 ldapcfinfo returns with a non-zero exit status if it encounters an error, and messages will be logged to stderr.

Messages will follow the below format:

```
ERROR:  code message
```

or

WARNING: code message

Leading extra white space may be inserted to improve readability and follow 80 column screen formatting.

code will be a programmatically parsable error key-string, while
message will be human-readable. Refer to the LDAP-UX Client Services Administrator's Guide for a list of possible error codes generated by the LDAP user and group management tools.

SEE ALSO
ldapugadd(1M), ldapugdel(1M), ldapuglist(1M), ldapugmod(1M), ldapux(5).
NAME
            ldapclientd - LDAP client daemon process

SYNOPSIS
            Startup
            /opt/ldapux/bin/ldapclientd [-d level] [-o { stdout | syslog | file[size] }] [-z]

            Control
            /opt/ldapux/bin/ldapclientd [-d level] [-o { stdout | syslog | file[size] }]
            /opt/ldapux/bin/ldapclientd [ -f | -k | -L | -h | -r ]

DESCRIPTION
            ldapclientd is a daemon process that enables LDAP-UX. It enables LDAP-UX clients to work with
            LDAP directory servers. ldapclientd caches entries, supports multiple domains in the Windows
            2000/2003/2003 R2 Active Directory Server (ADS), supports X.500 group membership, maintains connec-
            tions to the LDAP Directory Server, and manages remote LP printers.

            In addition to handling communication and maintaining the connection between the LDAP-UX client and
            LDAP directory server, ldapclientd provides the following:

            • Services:
              1. Enables LDAP-UX to use LDAP directory servers to support these services:

                 automount
                 group (including dynamic groups)
                 hosts
                 netgroup
                 networks
                 passwd
                 printers
                 protocols
                 rpc
                 services
                 X.500 group membership

                 Note: LDAP-UX does not support netgroup with the Microsoft Windows 2000/2003/2003 R2
                 Active Directory Server.

            • Performance:
              1. Caches entries to reduce LDAP-UX client response time retrieving:

                 automount
                 group (including dynamic groups)
                 netgroup
                 passwd
                 X.500 group membership

                 Note: Since pwgrd still caches passwd and group entries that did not come from the directory
                 server (ldapclientd only caches entries from directory servers), pwgrd is still useful to maintain
                 high performance.

              2. Reuses and maintains connections to the directory server, reducing binding and disconnection which
                 significantly reduces the load on the server and network traffic.

            • Capability:
              1. Multiple domain Active Directory Server (ADS).

                 Enables LDAP-UX to use multiple domains for directory servers like Windows 2000/2003/2003 R2
                 Active Directory Server (ADS). It allows PAM_Kerberos to authenticate POSIX users stored in
                 remote domains.

              2. Automatic profile downloading.

                 Updates the LDAP client configuration profile by downloading a newer copy from the directory server
                 as the current one’s profileTTL (Time To Live) expires.
3. Management of remote LP printer configuration.
   Searches printer objects configured in LDAP server. Accordingly, add/modify/remove printers for the local system. By default, the printer configurator is enabled.

   By default, `ldapclientd` starts at system boot time.

   The `ldapclientd` command can be used to launch the daemon or control it when the daemon is already running.

Options

The following `ldapclientd` options are supported. These options can be used only by a superuser.

`cache` Name of any NSS backend services that the `ldapclientd` daemon caches.

Valid cache names are:

- automount
- automountmap
- domaingrp
- domain_pwd
- dynamic_group
- group
- netgroup
- passwd
- uiddn

`-c` Not supported in this version.

Earlier versions (before B.02.00) used this to reread all LDAP-UX client settings from `/etc/opt/ldapux/ldapux_client.conf` just to force a refresh of the LDAP-UX profile. Refreshing only the profile is now done with the `-r` option. If rereading all LDAP-UX client settings is necessary, kill `ldapclientd` using `-k` then restart `ldapclientd`.

`-d level` Set log level. Initializes log level during startup or sets the log level of the running daemon process.

Accepted range of logging level is from 0 to 511.

- 0 = no logging output
- 1 = critical errors
- 2 = key function hits
- 4 = key areas
- 8 = looping function hits
- 16 = mutexes
- 32 = JUDY caching
- 64 = configuration file parsing
- 128 = statistic functions
- 256 = dumps

(dumps produce a large amount of logging which significantly degrades the performance of `ldapclientd`).

Example:

```
511 = 1 + 2 + 4 + 8 + 16 + 32 + 64 + 128 + 256
```

= log everything

Default log level is set at 1.

`-D [cache]` Disable cache. Without the cache name, all valid cache names are shown.

`-E [cache]` Enable cache. Without the cache name, all valid cache names are shown.

`-f` Flush all caches.

`-h` Invoke help.

`-k` Kill the LDAP client daemon.

This option sends a signal to the currently running `ldapclientd` daemon, causing it to exit gracefully.

This option is equivalent to:
ldapclientd(1M)

kill 'head -1 /etc/opt/ldapux/ldapclientd.pid'

-L

List available cache names.

-o target

Set log output to a target stream. Initializes log output during startup or sets the log output of the running daemon process. target can be one of the following:

stdout Direct logging to the standard output.
syslog Direct logging to syslogd. syslog usage also depends on syslog settings of LDAP-UX in the configuration file, /etc/opt/ldapux/ldapux_client.conf, which must be enabled, or ldapclientd will still ignore syslog.

file[=size] Specify a file for log output; rotation will append a 1 or 0 to this file name.

size defines the maximum file size before rotation. If size is not specified, it defaults to 1048576 bytes.

Example:
-o /tmp/log=50000

This will log output into /tmp/log0 until it is 50000 bytes, then swap logging into /tmp/log1, then back to /tmp/log0 when /tmp/log1 is 50000 bytes, and so on.

By default, log output goes to syslog.

-r

Immediately refresh (download) all profiles currently listed in the LDAP-UX client configuration file in /etc/opt/ldapux/ldapux_client.conf, if their current TTL expires. ldapclientd checks TTL expiration status every minute. With -r, ldapclientd immediately checks TTL status and refresh all profiles, if needed.

-S [cache]

Show statistics for one cache. When cache is not specified, a general cache statistic summary is shown.

-z

Disable daemonize (force ldapclientd to run in the foreground instead of the background). Prevent the ldapclientd process to fork into a daemon process during startup. Helpful for debugging.

DIAGNOSTICS
By default, errors are logged into syslog if system log is enabled in the LDAP-UX client configuration file at /etc/opt/ldapux/ldapux_client.conf. Errors occurring before ldapclientd forks into a daemon process will be displayed directly to stdout.

The following diagnostics may be issued:

Already running.
An attempt was made to start an LDAP client daemon when one was already running.

Cache daemon is not running (or running but not ready)
Can mean several things:
1. Attempted to use control option features of ldapclientd when no ldapclientd daemon process is running to control.
2. Attempted to start or control ldapclientd without superuser's privilege.
3. The ldapclientd daemon process is too busy with other requests to respond at this time. Try again later.

problem reading configuration file
The /etc/opt/ldapux/ldapclientd.conf file is missing or has a syntax error. If it is syntax, the error message is accompanied by a line showing exactly where ldapclientd could not recognize the syntax or found that the setting is out of range.

WARNINGS
Whenever the system is rebooted, ldapclientd launches if [StartOnBoot] has enable=yes in the ldapclientd configuration file, /etc/opt/ldapux/ldapclientd.conf.
Downloading profiles may take time, depending on server response time and the number of profiles listed in the LDAP-UX configuration file, /etc/opt/ldapux/ldapux_client.conf.

AUTHOR
ldapclientd was developed by Hewlett-Packard Company

FILES
Configuration
/etc/opt/ldapux/ldapclientd.conf
Configures caching, threading and boot options for ldapclientd.

/etc/opt/ldapux/ldapux_client.conf
LDAP-UX client services configuration file. Contains settings like logging and domains.

/etc/opt/ldapux/domain_profiles/ldapux_profile.ldif.gc
/etc/opt/ldapux/domain_profiles/ldapux_profile.ldif.bin.gc
/etc/opt/ldapux/domain_profiles/ldapux_profile.bin.gc
/etc/opt/ldapux/domain_profiles/ldapux_profile.bin.domain
Automatically downloaded local copies of directory profiles for each domain listed in /etc/opt/ldapux/ldapux_client.conf. These files should not be modified manually.

Operations
/etc/opt/ldapux/ldapclientd.pid
Lock file to restrict ldapclientd to one running instance

/var/spool/ldapcltd/status
Shared virtual memory used to quickly inform clients that the ldapclientd daemon is running and ready to service client requests.

/var/spool/ldapcltd/daemon
The ldapclientd daemon uses this pipe socket to receive client requests.

/etc/opt/ldapux/daemon_auth
Authentication file restricting execution of ldapclientd control options to the superuser.

/sbin/init.d/ldapclientd.rc
startup and shutdown script during system reboot.

/var/spool/sockets/ldapcltd/client*
Client pipe sockets used to receive responses from the ldapclientd daemon. Cleanup of this directory is left to clients. It is safe to delete all files left by dirty clients (and then the directory) only when the ldapclientd daemon is not running. While the ldapclientd daemon is running, it will attempt to automatically clean up after dirty clients periodically.

Other
/etc/opt/ldapux/ldapux_profile.ldif
The LDAP client configuration profile downloaded from the LDAP directory, in LDIF format.

/etc/opt/ldapux/ldapux_profile.bin
The LDAP client configuration profile translated from ldapux_profile.ldif, in binary format.

/opt/ldapux/config/setup
LDAP-UX client setup tool to configure LDAP-UX client services.

SEE ALSO
rc(1M), syslog(3C), ldapclientd.conf(4), rc.config(4), ldapux(5).
ldapugadd(1M)

NAME
ldapugadd - add new accounts or groups to an LDAP directory server

SYNOPSIS
ldapugadd [-t password] [options] [-h hostname] [-p port] [-b base] [-u uid_number]
[-g group/gid] [-f full_name] [-x domain] [-G group/gid[, ...] [-s login_shell]
[-d home_directory] [-I gecos] [-c comment] [-m [-k skel_dir]] [-T template_file] uid_name
[attr=value][...]

ldapugadd -t group [options] [-h hostname] [-p port] [-b base] [-u gid_number]
[-x domain] [-M member[, ...] [-c comment] [-T template_file] group_name
[attr=value][...]

ldapugadd -D [-d default_home] [-s default_shell] [-g default_gid] [-u min uid:max uid]
[-g min gid:max gid]

DESCRIPTION
ldapugadd allows HP-UX administrators to add new POSIX accounts or groups to an LDAP directory
server (see first and second syntaxes in SYNOPSIS above). Furthermore, ldapugadd can be used to
modify the /etc/opt/ldapux/ldapug.conf file to set defaults for creation of new users or groups
(see the third syntax in the SYNOPSIS above).

ldapugadd makes use of user and group template files that allow ldapugadd to conform to the infor-
mation model used for the types of entries being created. Users of ldapugadd are required to provide
LDAP administrator credentials that have sufficient privilege to perform the user or group add operation in
the LDAP directory server.

Options
-P Prompt for the administrators bind identity (typically LDAP DN or kerberos principal) and bind
password.

Without -P, ldapugadd will discover the bind identity and password from the environment vari-
ables LDAP_BINDDN and LDAP_BINDCRED. If either the LDAP_BINDDN or
LDAP_BINDCRED environment variable has not been specified, ldapugadd will follow the bind
configuration specified in the ldapux configuration profile (see ldapux(5)).

If ldapux has specified "proxy" bind, the bind credential will be read from either the
/etc/opt/ldapux/acred or /etc/opt/ldapux/pcred file. The acred file will only be
used by users that have sufficient administrative privileges to read that file. Refer to Binding to
the Directory Server below for additional details.

-PP Prompt for the password of the user or group being created. Also, if LDAP-UX attributed mapping
for the userPassword attribute has not been defined or set to *NULL*, ldapugadd will create
new passwords in the userPassword attribute. To assure accuracy, the user will be prompted twice
for the password. ldapugadd relies on the directory server for setting of password policy, such as
user-must-change-password-at-first-login.

-PW Set the user or group password attribute. Also, if LDAP-UX attributed mapping for the userPass-
word attribute has not been defined or set to *NULL*, ldapugadd will create new passwords in the
userPassword attribute. If -PW is specified, either the LDAP_UGCRED environment variable
or the -PP option must be specified.

-Z Requires an SSL connection to the directory server, even if the LDAP-UX configuration does not
require the use of SSL. Use of -Z requires either a valid server or CA certificate be defined in the
/etc/opt/ldapux/cert8.db file. An error will occur if the SSL connection could not be
established. Refer to Binding to the Directory Server below for additional details.

-ZZ Attempt a TLS connection to the directory server, even if the LDAP-UX configuration does not
require the use of TLS. If a TLS connection is unable to be established a non-TLS and non-SSL
connection will be established. Use of -ZZ is not recommended unless alternative methods are
used to protect from network eavesdropping. Use of -ZZ requires either a valid server or CA
certificate be defined in the /etc/opt/ldapux/cert8.db file. Refer to Binding to the Direc-
tory Server below for additional details.

-ZZZ Requires a TLS connection to the directory server, even if the LDAP-UX configuration does not
require the use of TLS. Use of -ZZZ requires either a valid server or CA certificate be defined in the
/etc/opt/ldapux/cert8.db file. An error will occur if the TLS connection could not be
established. Refer to Binding to the Directory Server below for additional details.
ldapugadd(1M) ldapugadd(1M)

-Ф Force creation of new user or group entries even if particular error conditions occur. These are:
  • The user name or group name already exists in the directory server.
  • The user id or group id number already exists in the directory server
  • The shell specified with the -s option does not exist on the local system or is not an executable.
  • Adding a member to a group when that member is not defined in the LDAP directory.
  Note that some directory servers perform their own attribute uniqueness checks. In this case, even if the -F option is specified; ldapugadd will be unable to add the new entry.

-Ś Display the DN of the newly created entry.

Arguments
-Ś hostname Specifies the host name and optional port number (hostname:port) of the directory server.
This option overrides the server list configured by LDAP-UX. The hostname field also supports specification of IPv4 and IPv6 addresses.
Note that when a port is specified for an IPv6 address, the IPv6 address must be specified in square-bracketed form. If the optional port is unspecified, the port number is assumed to be 389 or 636 for SSL connections (-Z). Refer to Binding to the Directory Server below for additional details.

-p port Specifies the port number of the directory server to contact. This option is ignored if the port number is specified in the hostname as part of the -h option. Refer to Binding to the Directory Server below for additional details.

-b base This option overrides value of the $({basedn}) substitution construct used in the respective template file. Instead of discovering the $({basedn}) value from the LDAP-UX configuration profile, the value defined in base will be used. Please refer to Template Files below for additional information. base is expected to be an LDAP distinguished name.

-t type Specifies on which service type ldapugadd will operate. The service type can be either passwd or group, where
  passwd implies posixAccount-type entries, and
  group implies posixGroup-type entries.
  The command-line arguments that are applicable will depend on the service specified. If unspecified, ldapugadd defaults to passwd.
  Note: to be consistent with the Name Service Switch (see switch(4)), the term passwd is used to represent LDAP user entries which contain POSIX account-related information.

Arguments Applicable to -D
-D Used to permanently alter local host defaults which are used by ldapugadd when creating new user or group entries in the LDAP directory. Configuration changes made using the -D options will appear in the /etc/opt/ldapux/ldapug.conf file. Please refer to the LDAP-UX Client Services Administrator’s Guide for an example of the ldapug.conf file.

-d default_home Specifies the parent directory that will be used when creating new user home directories.

-s default_shell Specifies the default login shell that will be used when creating user entries.

-g default_gid Specifies the default group ID number used when creating new user entries. To avoid warning messages displayed by ldapugadd, this group ID should represent a POSIX-style group stored in the LDAP directory.
  If this group ID is not defined in the LDAP directory, ldapugadd will display a warning message every time a new user is added using this default group, since ldapugadd will be unable to add the user as a member of that group.

-g min_gid:max_gid Sets new default minimum and maximum ranges that ldapugadd will use when provisioning a group ID number for newly created group entries. The gid range is inclusive of
ldapugadd(1M)

ldapugadd(1M)

the specified end values. The colon character (:) will be used to indicate a range has been
specified, instead of the default_gid specified above.

-u min_uid:max_uid
Sets new default minimum and maximum ranges that ldapugadd will use when provi-
sioning an uid number for newly created user entries. The uid range is inclusive of the
specified end values.

Arguments Applicable to `-t passwd`

-u uid_number
Specifies the user's numeric ID number. If the specified uidNumber already exists in the
directory server, ldapugadd will not add the new entry and return an error exit status,
enless the -F option is specified.

If this argument is not specified, a new user ID number will be provisioned by randomly
selecting a value from the uidNumber range specified by ldapugadd -D -u. If
ldapugadd randomly selects a uidNumber that is already in use on the directory server,
ldapugadd will randomly select another uidNumber and try again until it finds an
unused uidNumber or exhausts retry attempts. Retry attempts will be limited to 90% of
the range of available uidNumbers (specified with -D -u min_uid:max_uid and described
above).

-g group/gid
Specifies the user's primary login group name or id number. After creating the user entry
ldapugadd will also attempt to add the user as a member of the specified group using the
ldapugmod -t group command.

Note: to support numeric group names, ldapugadd will always attempt to resolve the
specified argument as a group name (even if it is a numeric string). If the specified argu-
ment is not found as a group name, ldapugadd will check to see if the argument is a
numeric string and if so, use that as the group ID number. If that numeric group can not
be found in any active name service repository, ldapugadd will issue an ERROR mes-
 sage. If the specific argument is not numeric and can not be found in an active name ser-
vice repository, ldapugadd will exit with an ERROR and not create the new entry.

If this argument is not specified, the user will become a member of default login group, as
specified by the ldapugadd -D -g default_gid command.

-f full_name
This option is only required for the passwd service and is used to specify the user's full
name. If undefined, the user's full name will default to the account name.

-x domain
Specifies the user's domain name. This variable is used to specify the ${domain} value
that can be used in the template file. If this value is not specified, the domain name will be
created by using the first "dc" component of the new user's distinguished name. If the dis-
tinguished name does not contain any "dc" components, and the ${domain} variable is
specified in the template file, ldapugadd will generate an error.

-G group/gid,]
Specifies the user's alternate group memberships. group/gid is expected to be the POSIX
textual name of the group or the group ID number. That group must exist in the directory
server (not the /etc/group file).

If the specified group name is invalid or does not exist in the directory server, ldapugadd
will issue a warning message for each invalid group. To support numeric group names,
ldapugadd will always attempt to resolve the specified argument as a group name (even
if it is a numeric string). If the specified argument is not found as a group name,
ldapugadd will check to see if the argument is a numeric string and if so, use that as the
group ID number.

After the user's entry is successfully created (and only if), ldapugadd will call ldapug-
mod -t group (see ldapugmod(1M)) for each group specified, to add the user to listed
groups.

If more than one group is specified, each group name must be separated by a comma. No
whitespace is allowed between or within group names. If ldapugadd fails to add the user
as a member of a particular group, ldapugadd will issue a warning message and continue
to add the user to the other remaining groups specified.

ldapugadd(1M)

If this argument is not specified, the user will not be added to alternate groups.

-s login_shell

Specifies the full path name to the executable that will be used to handle login sessions for this user. If this argument is not specified, the default, as configured by ldapugadd -D -s default_shell, will be used.

-d home_directory

Specifies the full path name (including the user name) of the user's home directory. If this argument is not specified, the combination of the default base directory, as configured by ldapugadd -D -d default_home, and the user's account name, will be used. If you wish to also create the home directory on this system, the -m option must specified.

-I gecos

Specifies GECOS fields for the user. Typically the GECOS contains four fields which represent (in order):

- The user's full name
- The user's work location
- The user's work telephone number
- The user's home telephone number (often omitted)

Each field in the gecos should be separated by a comma. If the data within the gecos field contains any white space or other characters that might be parsed by the shell, the entire string must be protected by enclosing quotes. White space should not be used between each field and the separating commas.

Note that LDAP-UX supports mapping of the gecos field to multiple attributes. If attribute mapping has been specified in the LDAP-UX configuration profile, each field will be mapped to its representative attribute, in the order specified.

If -I is not specified, the gecos attribute(s) will not be added to the user's entry.

WARNING: If the -I option is specified and attribute mapping has been defined for the gecos attribute, be careful not to specify the same attributes in the command line that are also used in the gecos map. For example, suppose the gecos has been mapped to cn, l and telephoneNumber. Because -f below represents the cn attribute when creating new user account entries, the following command might produce unpredictable results since cn is specified by both -f and by the gecos mapping:

ldapugadd -f "Jim Smith" \
-I "Jim Smith,Boston,555-12354" \
 jsmith "sn=Smith" "telephoneNumber=555-1234"

In the above example, because of the gecos attribute mapping, the cn and telephoneNumbers are specified twice and will result an error when the same attribute and value are added to the directory server ldapcfinfo can be used to determine gecos attribute mapping configuration.

NOTE: Since the gecos attribute may be mapped to one or several attributes, the number of values specified with -I (between the commas) should, but is not required to, match the number of mapped attributes. If there are more mapped attributes than specified values in -I, then trailing mapped attributes will not be added to the directory server. If there are more values than mapped attributes, extra values will be combined in the last mapped attribute.

-c comment

Specifies a comment that will be stored in the description attribute, as defined by RFC2307. Attribute mapping is not defined for the description attribute. If unspecified, the description attribute will not be added to the user's entry. Since the comment field often contains white-space, be sure to protect it from shell parsing with enclosing quote characters.

-T template_file

Specifies the LDIF template file that will be used to create new user entries. The template_file parameter may either be a full or relative path name or a "short" name. Refer to Template File Naming below for additional information.

uid_name Required Argument. Contains the POSIX-style textual login name for the new user entry. This user name should conform to HP-UX login name requirements. Please refer to passwd(4) for login name requirements.

uid_name is a required parameter, and it must follow all command-line options and must precede the attr=value parameters (if provided).
ldapugadd(1M)

-\texttt{m}\hspace{1em}Create a new home directory for the defined user. User and group ownership of the newly
created directory will be assigned to the user and his/her primary login group.

If \texttt{-k} is specified, the files and sub-directories found in \texttt{skel\_dir} will be copied to the user's
home directory, and user and group ownership permissions altered as specified above. If
\texttt{-k} is not specified, skeleton files will be copied from \texttt{/etc/skel}.

The \texttt{-m} option requires the user has sufficient privilege to create the new home directory,
copy skeleton files and change ownership of those files and directories. \texttt{ldapugadd} will
create a user's home directory only after successfully adding the user's entry in the directory
server.

If \texttt{ldapugadd} is unable to properly create the user's home directory, per the above pro-
cess, the newly created changes in the directory server will not be removed. See Security
Consideration below for more information.

\texttt{-k skel\_dir} \hspace{1em} \texttt{-k} is ignored unless the \texttt{-m} option is specified. \texttt{skel\_dir} specifies a directory which contains
skeleton files and directories that should be copied into newly created user home directories. See \texttt{-m} above.

\texttt{attr=value} \hspace{1em} Allows specification of arbitrary LDAP attributes and values. Because of potential
objectclass requirements, additional information beyond the basic POSIX account and group
data may be need to be specified in order to create new entries in the directory server.

For example, if the "InetOrgPerson" objectclass is used as a structural class for \texttt{posixAccounts}, then the \texttt{sn} (surname) attribute must be specified in order to properly create a new
entry. This value would need to be defined in the template file (see Template Files), and
would need to be specified at the end of the \texttt{ldapugadd} command line.

The \texttt{attr=value} parameter is generally used to specify attributes required by the template
file. However, if an attribute is specified which is not defined in the defined template file,
that attribute/value pair will be considered as an optional attribute/value which will be
added to the entry exactly as specified.

\texttt{attr=value} parameters are optional, but must be specified as the last parameters on the
command line.

Arguments Applicable to \texttt{-t group}'

\texttt{-g gid\_number}\hspace{1em} Specifies the group's numeric id number. If the specified \texttt{gidNumber} already exists in the
directory server, \texttt{ldapugadd} will not add the new entry and return an error exit status,
unless the \texttt{-F} option is specified.

If this argument is not specified, a new group ID number will be provisioned by randomly
selecting a value from the \texttt{gidNumber} range specified by \texttt{ldapugadd -d -g min\_gid : max\_gid}. If \texttt{ldapugadd} randomly selects a \texttt{gidNumber} that is already in use on
the directory server, \texttt{ldapugadd} will randomly select another \texttt{gidNumber} and try again
until it finds an unused \texttt{gidNumber} or exhausts retry attempts. Retry attempts will be limited
to 90% of the range of available \texttt{gidNumbers} (specified with \texttt{-D -g min\_gid : max\_gid}
described above).

\texttt{-x domain}\hspace{1em} Specifies the group's domain name. This variable is used to specify the \texttt{${domain}} value
that can be used in the template file. If this value is not specified, the domain name will be
created by using the first "dc" component of the new group's distinguished name. If the dis-
tinguished name does not contain any "dc" components, and the \texttt{${domain}} variable is
specified in the template file, \texttt{ldapugadd} will generate an error.

\texttt{-M member[, ...]} \hspace{1em} Defines initial group membership by adding the specified user accounts as members. The
members must be defined as a comma-separated list of account names, similar to the \texttt{-G}
requirements defined above. Use of \texttt{-M} requires that the specified user's account already be
defined in the directory server, unless the \texttt{-F} option is specified.

When the \texttt{-F} option is used, the users group membership will be defined using the \texttt{memberUid} attribute, regardless of the attribute mapping configuration defined by LDAP-UX.
Use of \texttt{-F} is not recommended, and will not succeed if the directory server does not support
the \texttt{memberUid} attribute.

HP-UX 11i Version 2: December 2007 Update

Hewlett-Packard Company
ldapugadd(1M) ldapugadd(1M)

ldapugadd will follow the same membership syntax as defined by LDAP-UX attribute
mapping. Specifically, if LDAP-UX has mapped the RFC2307 group membership attribute
(memberUid) to a DN-based membership attribute such as member or uniqueMember, then
ldapugadd will define membership using the DN of the specified user. If memberUid has
been mapped to more than one attribute type, ldapugadd will use the first attribute
defined by the mapping.

Note that ldapugadd can only add members to a group that follow a static membership
syntax (like memberUid, member, and uniqueMember). ldapugadd will fail if the only
mapping defined uses a dynamic group membership syntax (like memberUrl).

-c comment Specifies a comment that will be stored in the description attribute, as defined by RFC2307.
Attribute mapping is not defined for the description attribute. If unspecified, the description attribute will not be added to the user's entry.

-T template_file Specifies the LDIF template file that will be used to create new group entries. The template_file parameter may either be a full or relative path name or a "short" name. Refer to Template File Naming below for additional information.

Required Argument. Contains the POSIX-style textual group name for the new group
entry. This name should conform to HP-UX group name requirements. Please refer to group_name for group name requirements. gid_name is a required parameter, must follow all command-line options and must precede the attr=value parameters (if provided).

attr=value Allows specification of arbitrary LDAP attributes and values. Refer to attr=value in the section above for additional information. attr=value parameters are optional, but must be specified as the last parameters on the command line.

Template Files
One of the benefits of LDAP directory servers is the flexibility to support customized data models to meet
organizational requirements. This flexibility allows each directory deployment to define unique data models
for users and groups. Because of this, it’s not possible for ldapugadd to be able to create new user or
group entries in the directory server and also follow the desired data model, without some description of
that data model.

Template files for user and group entries allow ldapugadd to discover the required data models for new
user and group entries. Template files define what data is required to create new user and group entries
and allow ldapugadd to discover required attributes and data elements before creating the entries.

To explain this concept, the below examples show the default templates for a standard directory server for
a passwd and group entry. Samples, such as the one below, are delivered with LDAP-UX, including sam-
ple files for ADS.

Below is a sample default template for standard directory server:

dn: uid=${uid},ou=people,${basedn}
objectclass: InetOrgPerson
objectclass: posixAccount
sn: ${Surname}
$ {posixProfile}

dn: cn=${cn},ou=groups,${basedn}
objectclass: groupOfNames
objectclass: posixGroup
{posixProfile}

Below is a sample default template for Windows ADS:

dn: cn=${cn},cn=users,${basedn}
objectclass: user
{posixProfile}
SAMAccountName: ${uid}
msSFUONisDomain: ${domain}
#By default, ldapugadd creates disabled accounts.
#Change below to 544 to enable accounts by default.
userAccountControl: 546
Each template file must follow the LDIF data format and also allow for substitution of values from the 
ldapugadd command. Two default template files (for user and group entries) for standard directory 
servers, along with two default template files for ADS are provided under 
/etc/opt/ldapux/ug_templates. The following guidelines can be used when creating template 
files.

- Each template file is used for defining only one entry in the directory server.
- Each template file may contain comment lines. Each comment line must begin with the pound (#) char-
acter.
- Each template file can be built using custom attributes and values. Customized attribute values are 
defined using the \$\{name\} construct. However, for each non-RFC2307 attribute used, when 
ldapugadd creates a new entry, each one of those attributes must be specified on the command line as an attr=value pair.
- ldapugadd supports several pre-defined substitution constructs, where name is represented by:

  - posixProfile: Represents all RFC2307-type attributes and values for the particular name service 
    (either passwd or group). If LDAP-UX has defined attribute mapping for particular 
    attributes, the mapped attributes will be substituted in its place. When used for 
    posixAccount-type entries, the following attributes and values will be added to the entry:
    - cn
    - uid
    - userPassword
    - uidNumber
    - gidNumber
    - gecos
    - homeDirectory
    - loginShell

    When used with posixGroup-type entries, the following attributes and values will be 
    added to the entry:
    - cn
    - userPassword
    - gidNumber
    - memberUid

    Note: Since use of posixProfile supports attribute mapping, if the above attributes have 
been mapped as configured in the LDAP-UX configuration profile, the mapped attributes 
and values will be added to the entry instead of the RFC2307 defined attributes. For 
example, if the posixAccount attribute gecos has been mapped to cn l telephoneNumber 
then cn, l and telephoneNumber will be added to the entry instead of gecos. And for 
another example with posixGroups, if memberUid has been mapped to uniqueMember, 
then uniqueMember will be added (using the DN syntax) to the entry instead of mem-
berUid.

  - basedn: Represents the distinguished name of the default search base (defaultSearchBase) as 
    obtained from the LDAP-UX configuration profile.

  - uid: Represents the user's account name when used in a passwd template file.

  - uidNumber: Represents the user's account ID number when used in a passwd template file.

  - cn: Represents the user's full name when used in a passwd template file. Represents the 
    group name when used in a group template file.

  - gidNumber: Represents the group ID number when used in a group template file.

- The first line of the template file is used to define the distinguished name of the new entry. Since each 
DN is unique, the first component of the DN (the Relative Distinguished Name or RDN) must be able to 
construct a unique value for each new entry. Thus the RDN should be constructed using a \$\{name\}
construct. Typically the `cn` or `uid` attribute would be used in the RDN for new `passwd` entries and the `cn` attribute would be used for new `group` entries.

- The `userPassword` attribute cannot be specified in the template file. See the `-PP` option for additional information about specifying an initial user or group password.

- The `memberUid` attribute cannot be specified in the template file, since the number of eventual members of a group can not be statically defined when the group is newly created. `ldapugadd` will ignore the `memberUid` attribute if specified in the template file.

As mentioned above, for each non-pre-defined substitution construct, using the `$(name)`, requires specification of the `name` attribute and value on the `ldapugadd` command line. If a non-POSIX attribute is specified in the template file (such as `sn`) and that attribute/value pair has not been specified on the command line (`attr=value`), `ldapugadd` will return an error.

To assist with programmatic discovery of the required attributes when creating new entries, the `ldapcfinfo` has provided command line options to list the these attributes. `ldapcfinfo -t passwd -R` will display required attributes when creating new `passwd` entries. `ldapcfinfo -t group -R` will display required attributes when creating new `group` entries.

**Multi-Valued Attributes in Template Files**

Template files can support multi-valued attributes. This means that the same attribute name and/or value can be specified more than once in the template file. Example:

```
 dn: uid=${uid},ou=people,$(basedn)
 objectclass: InetOrgPerson
 objectclass: myOrg
 objectclass: posixAccount
 sn: $({Surname})
 primaryTeam: $({primaryTeam})
 secondaryTeams: $({secondaryTeams})
 secondaryTeams: $({secondaryTeams})
 ${posixProfile}
```

In the above example we assume that `secondaryTeams` is a multi-valued attribute which should be specified at least twice for each new `posixAccount` entry created. In this case, `ldapugadd` will fill each attribute value in order specified in the template file based on the order that those attributes are specified on the command line. Note if not enough attribute values have been specified on the command line to fill the attribute values used in the template file, `ldapugadd` will return an error.

**Template File Naming**

The default template files for new `passwd` and `group` entries are stored in `/etc/opt/ldapux/ug_templates` and are named `ug_passwd_default.tmpl` and `ug_group_default.tmpl`.

All template files stored in the `/etc/opt/ldapux/ug_templates` directory must follow a specific naming format. To allow specification of template files by their short name (see below), the template file name must begin with `ug` followed by the service name being supported. Underbars are used to separate sections of the name. The remainder of the name may be any keystring, followed by a `.tmpl` extension.

For example `ug_passwd_vpn_user.tmpl` might be used when creating new users of the "VPN" type. Template files stored outside of the `ug_templates` directory need not follow any specific format.

When specifying the name of a template file as part of the `-T` option on the command line, either the exact file name or a short name may be used. When specifying the file name, that name may be either a full or relative path name, but must begin with either the slash (`/`) or dot (`.`) characters. That file name may exist anywhere in the file system.

When specifying the short name, the file must exist under the `/etc/opt/ldapux/ug_templates` directory and must follow the format specified above. For example `-T passwd -T vpn_user` would be used to specify the template file mentioned above: `/etc/opt/ldapux/ug_templates/ug_passwd_vpn_user.tmpl`.

A short name is defined as the distinguishing portion of the template file name. For example, for the `passwd` service, if the short name "operator" is specified, the resulting template file would be `/etc/opt/ldapux/ug_templates/ug_passwd_operator.tmpl`. 
ldapugadd(1M)

All LDAP-UX default template files will be stored in the /etc/opt/ldapux/ug_templates directory.

A full or relative path name must begin with either the slash (/) or dot (.) characters. If unspecified, either of the following default template file will be used:

- /etc/opt/ldapux/ug_templates/ug_passwd_default.tmpl,
- /etc/opt/ldapux/ug_templates/ug_group_default.tmpl.

Binding to the Directory Server

ldapugadd has been designed to take advantage of the existing LDAP-UX configuration for determining to which directory server to bind and how to perform the bind operation. ldapugadd will consult the LDAP-UX configuration profile for the following information:

- The list of LDAP directory server hosts.
- The authentication method (simple passwords, SASL Digest MD5, etc.)

If either of the environment variables LDAP_BINDDN or LDAP_BINDCRED have not been specified, ldapugadd will also consult the LDAP-UX configuration for additional information:

- The type of credential (user, proxy or anonymous) to use.
- The credential used for binding as a proxy user (either /etc/opt/ldapux/acred for administrative users or /etc/opt/ldapux/pcred for non-privileged users.)

As with LDAP-UX, ldapugadd will attempt to contact the first available directory server as defined in the LDAP-UX host list. As soon as a connection is established, further directory servers on the host list will not be contacted. Once connected, ldapugadd will first determine if the environment variables LDAP_BINDDN or LDAP_BINDCRED have been specified. If both are specified, then ldapugadd will attempt to bind to the directory server using the specified credentials and configured LDAP-UX authentication method.

If either of the above mentioned environment variables have not been specified, then ldapugadd will determine if the configured credential type is "proxy" and if so, attempt to bind to the directory server using the configured LDAP-UX proxy credential. If configured, the acred proxy credential will be used for administrative users (determined if the user running ldapugadd has enough privilege to read the /etc/opt/ldapux/acred file). Otherwise the credential configured in /etc/opt/ldapux/pcred will be used.

Note, to prevent discovery of the LDAP administrator's credentials, the LDAP user DN and password may not be specified as command-line options to the ldapugadd utility.

Security Considerations

- Use of ldapugadd requires permissions of an LDAP administrator when it performs its operations on the directory server. The rights to create new LDAP directory entries under the requested subtree, along with creation of the required attributes in that entry must be granted to the LDAP administrator identity that is specified when executing ldapugadd.
- As with any POSIX-type identity, the HP-UX operating system uses the specified user and group ID number to determine rights and capabilities in the OS as well as in the file system. For example, the root user ID 0, typically has unlimited OS administration and file access rights. Before creating a new entry, be aware of the selected user and group ID number and any policy that may be associated with that ID.
- If ldapugadd is used to randomly assign a user or group ID number, it only checks for ID collisions found in the LDAP directory server, and not other policy repositories. When setting user and group ID number ranges (-D option with either -u or -g) be sure to set a range that is not used by other user or group ID repositories, to assure collisions would not occur with existing users or groups that exist in other repositories.
- As would occur in any identity repository, modification of this repository will likely have impacts as defined by the organization's security policy. Users of ldapugadd are expected to have full knowledge of the impact to the organization's security policy when adding new identity information to that identity repository.
- In order to support non-interactive use of the ldapugadd command, specification of the LDAP administrator's credentials is required through use of the LDAP_BINDDN and LDAP_BINDCRED environment variables. To prevent exposure of these environment variables, they should be unset after use.
Note also that shells command history log may contain copies of the executed commands that show setting of these variables. Access to a shell's history file must be protected. Specification of the LDAP administrator's credentials on the command line is not allowed since information about the currently running processes can be exposed externally from the session.

Use of the -P eliminates the need to set the mentioned environment variables by interactively prompting for the required credentials.

**LDAP-UX PROFILE**

`ldapugadd` makes use of the LDAP-UX configuration profile to determine the information model used in the directory server to store POSIX attributes. Please refer to the *LDAP-UX Client Services Administrator's Guide* for additional information about the configuration profile.

**LDAP configuration file**

LDAP-UX supports a local configuration file, `/etc/opt/ldapux/ldapug.conf`. The `ldapugadd` tool uses the `ldapug.conf` file to manage the following default values when creating new user and group entries in an LDAP directory server:

- A default group ID for new users
- The valid UID number range for new users
- The valid GID number range for new groups
- The base path for a new user's home directory. By default, LDAP-UX appends the user's account name to the base path to create the full path name.
- The default login shell for new users

LDAP-UX provides the default `ldapug.conf` file as follows:

```plaintext
# This file is used by the ldapugadd tool for management
# of default values for creating new user and group entries.
# This file can not be modified directly, but instead through
# the ldapugadd -D command.
# uidNumber_range=100:20000
gidNumber_range=100:2000
default_gidNumber=20
default_homeDirectory=/home
default_loginShell=/usr/bin/sh
```

**EXTERNAL INFLUENCES**

**Environment Variables**

**LDAP_UGCREDD**

When used in combination with the -PW option, `LDAP_UGCREDD` specifies the password of a newly created user or group.

Note, use of passwords for groups is not recommended.

Also, if LDAP-UX attributed mapping for the `userPassword` attribute has not been defined or set to `*NULL*`, `ldapugadd` will create new passwords in the `userPassword` attribute.

**LDAP_BINDDN**

Specifies the DN of a user with sufficient directory server privilege to create new users and/or groups in the LDAP directory server.

While this variable is optional, if `LDAP_BINDDN` is specified, `LDAP_BINDCREDD` must also be specified.

**LDAP_BINDCRED**

A password or other type of credential used for the user specified by the `LDAP_BINDDN`.

While this variable is optional, if `LDAP_BINDCRED` is specified, `LDAP_BINDDN` must also be specified.

Refer to *Security Considerations* for important security impacts when these environment variables are used.
ldapugadd(1M)

RETURN VALUE

Upon exit, ldapugadd returns the following:

0  Success. ldapugadd exits with no errors or with one or more warnings.

<>0  ldapugadd returns with a non-zero exit status if it encounters an error, and messages will be logged to stderr.

Messages will follow the below format:

ERROR: code message

or

WARNING: code message

Leading extra white space may be inserted to improve readability and follow 80 column screen formatting.

code will be a programmatically parsable error key-string, while

message will be human-readable. Refer to the LDAP-UX Client Services Administrator’s Guide for a list of possible error codes generated by the LDAP user and group management tools.

LIMITATIONS

- Since LDAP directories require data be stored according to the UTF-8 (RFC3629) character encoding method, all characters passed into ldapugadd are assumed to UTF-8, and part of the ISO-10646 character set. ldapugadd does not perform conversion of the locale character set to/from the UTF-8 character set.

- Since ldapugadd calls functions to discover if groups exists before adding a user to a group, it is possible to encounter timing issues with cached information. For example, if an administrator wishes to see if a group exists by using grget, this group information will be cached by both ldapclientd and pwgrd.

If the group does not exist when calling grget, and the administrator shortly there after creates this group with ldapugadd, the information that the group still does not exist will still be cached. Then, when adding a new user and specifying that this user is a member of the just created group, an error will be generated to indicate the user can not be added to the group. To resolve this, the pwgrd and ldapclientd caches must be flushed.

SEE ALSO

ldapcfinfo(1M), ldapugdel(1M), ldapuglist(1M), ldapugmod(1M), ldapux(5).
NAME
ldapugdel - remove existing accounts or groups from an LDAP directory server

SYNOPSIS
ldapugdel [options] [-t type] [-h hostname] [-p port] [-O [protAttr,...]]
    (-D DN | uid_name | group_name)

DESCRIPTION
ldapugdel is used to remove POSIX related user or group entries from the directory server. With the
-O option, ldapugdel can be used to remove POSIX related attributes and objectclasses from user or
group entries, without removing the entry itself.

Options
-P Prompt for the administrators bind identity (typically LDAP DN or kerberos principal) and bind
password. Without -P ldapugdel will discover the bind identity and password from the
environment variable LDAP_BINDDN and LDAP_BINDCRED. If the LDAP_BINDDN or
LDAP_BINDCRED environment variable has not been specified, ldapugdel will follow the bind
configuration specified in the LDAP-UX configuration profile.

If LDAP-UX has specified "proxy" bind, the bind credential will be read from either the
/etc/opt/ldapux/acred or /etc/opt/ldapux/pcred file. The acred file will only be
used by users that have sufficient administrative privilege to read that file. Refer to Binding to the
Directory Server below for additional details.

-S Upon successful completion, displays the DN of the deleted or updated entry.

-x Used only with the -O option, forces ldapugdel to remove the uid, cn, and description attributes
for either a user or group entry, respectively.

Because use of -x removes common attributes typically used by other LDAP-enabled applications,
use of it is rarely recommended when removing posixAccount or posixGroup related attributes. If
removal of the uid, cn, or description would cause an objectclass violation, a warning message
would be generated. -x will try to remove as many attributes as allowed by the directory server.

-y Used only with the combined with -O and the -t passwd options, forces ldapugdel to
remove the userPassword attribute from the user entry.

Use of -y is rarely recommended when removing posixAccount related attributes.

-Z Requires an SSL connection to the directory server, even if the LDAP-UX configuration does not
require the use of SSL.

Use of -Z requires either a valid server or CA certificate be defined in the
/etc/opt/ldapux/cert8.db file. An error will occur if the SSL connection could not be
established.

-ZZ Attempt a TLS connection to the directory server, even if the LDAP-UX configuration does not
require the use of TLS. If a TLS connection is unable to be established a non-TLS and non-SSL
connection will be established.

Use of -ZZ is not recommended unless alternative methods are used to protect from network
eavesdropping. Use of -ZZ requires either a valid server or CA certificate be defined in the
/etc/opt/ldapux/cert8.db file. Refer to Binding to the Directory Server below for additional
details.

-ZZZ Requires a TLS connection to the directory server, even if the LDAP-UX configuration does not
require the use of TLS. Use of -ZZZ requires either a valid server or CA certificate be defined in the
/etc/opt/ldapux/cert8.db file. An error will occur if the TLS connection could not be
established. Refer to Binding to the Directory Server below for additional details.

Arguments
-h hostname Specifies the host name and optional port number (hostname:port) of the directory server.
This option overrides the server list configured by LDAP-UX.

The hostname field also supports specification of IPv4 and IPv6 addresses. Note that when
a port is specified for an IPv6 address, the IPv6 address must be specified in square-bracketed
form. If the optional port is unspecified, the port number is assumed to be 389 or
636 for SSL connections (-Z). Refer to Binding to the Directory Server below for additional
ldapugdel(1M)

details.

-p port  Specifies the port number of the directory server to contact. This option is ignored if the port number is specified in the hostname as part of the -h option. Refer to Binding to the Directory Server below for additional details.

-t type  Specifies the service type of entry to be deleted. The service type can be either passwd or group, where:

    passwd implies posixAccount-type entries and,
    group implies posixGroup-type entries.

If unspecified, ldapugdel defaults to passwd.

Note: to be consistent with the Name Service Switch (see switch(4)), the term passwd is used to represent LDAP user entries which contain POSIX account-related information.

-O[protAttr[,...]]  Do not delete the entire user or group entry. Instead delete only the posixAccount or posixGroup objectclass and associated attributes.

With the -t passwd option, ldapugdel will remove the posixAccount objectclass and the following attributes:

- uidNumber
- gidNumber
- homeDirectory
- loginShell
- gecos

With the -t group option, ldapugdel will remove the posixGroup objectclass and the following attributes:

- gidNumber
- memberUid
- userPassword

The protAttr list is of one or more of the above attribute names separated by commas with no white-space. If specified, ldapugdel will not remove the specified attribute(s).

Special notes for using the -O option:

- Since mapped attributes are often attributes that are shared with other LDAP-enabled applications, attribute mapping is not supported with ldapugdel.

For example, if uidNumber has been mapped to employeeNumber, ldapugdel will still attempt to remove the uidNumber attribute and not the employeeNumber attribute.

- Since the uid, cn, and description attributes, for user entries, and the cn and description attributes, for group entries, are commonly used by other objectclasses or as naming attributes, ldapugdel will not attempt to remove the uid, cn, or description attributes, unless failure to remove those attributes would cause an objectclass violation (because the remaining object classes for that entry would not be able to contain those attributes).

Use of -x will force removal of those attributes if allowed by the remaining object classes for that entry.

- Since the userPassword attribute is often used by other user-related objectclasses, ldapugdel will not attempt to remove the userPassword attribute when removing user entries.

Use of -y will override this option, if allowed by the remaining object classes in that entry.

- ldapugdel will attempt to remove the posixAccount and posixGroup objectclasses only if they are present. In some cases, when a user or group entry is built using an abstract class, the posixAccount and posixGroup entries may not be present in the entry.

- Since Active Directory schema and RFC2307 schema conflict in the shared definition of the homeDirectory attribute, ldapugdel will never remove the homeDirectory attribute if ldapugdel determines the entry being modified is stored on an Active Directory Server.
Directory server.

- Since the Microsoft Services for Unix schema does not use RFC2307 standard attributes, use of -O will not function, since attribute mapping is not allowed in ldapugdel. -O will function properly with Windows 2003 R2, since standard RFC2307 attributes are used, with the exception of the homeDirectory, described above.

-D DN  Normally ldapugdel will search for the named user or group using the search rules described by the service search descriptor in the LDAP-UX configuration profile. With -D, the exact DN of the entry being modified may be specified.

Only one of -D, uid_name or group_name may be specified on the command line.

uid_name  Specifies the name of the user entry to remove. Note that ldapugdel uses the configured LDAP search filter to discover the entry to be removed, such as:

(& (objectclass=posixAccount) (uid=name)).

If there is more than one entry that matches this search filter, only the first entry discovered entry will be removed.

Only one of -D, uid_name, or group_name may be specified on the command line.

group_name  Specifies the name of the group entry to remove. Note that ldapugdel uses the configured LDAP search filter to discover the entry to be removed, such as:

(& (objectclass=posixgroup) (cn=name)).

If there is more than one entry that matches this search filter, only the first entry discovered entry will be removed.

Only one of -D, uid_name, or group_name may be specified on the command line.

Binding to the Directory Server

ldapugdel has been designed to take advantage of the existing LDAP-UX configuration for determining to which directory server to bind and how to perform the bind operation. ldapugdel will consult the LDAP-UX configuration profile for the following information:

- The list of LDAP directory server hosts.
- The authentication method (simple passwords, SASL Digest MD5, etc.).

If either of the environment variables LDAP_BINDDN or LDAP_BINDCRED has not been specified, ldapugdel will consult the LDAP-UX configuration for additional information:

- The type of credential (user, proxy or anonymous) to use.
- The credential used for binding as a proxy user (either /etc/opt/ldapux/acred for administrative users or /etc/opt/ldapux/pcred for non-privileged users).

As with LDAP-UX, ldapugdel will attempt to contact the first available directory server as defined in the LDAP-UX host list. As soon as a connection is established, further directory servers on the host list will not be contacted.

Once connected, ldapugdel will first determine if the environment variables LDAP_BINDDN or LDAP_BINDCRED has been specified. If so, then ldapugdel will attempt to bind to the directory server using the specified credentials and configured LDAP-UX authentication method.

If the above mentioned environment variables have not been specified, then ldapugdel will determine if the configured credential type is "proxy" and if so, attempt to bind to the directory server using the configured LDAP-UX proxy credential.

If configured, the acred proxy credential will be used for administrative users (determined if the user running ldapugdel has enough privilege to read the /etc/opt/ldapux/acred file). Otherwise the credential configured in /etc/opt/ldapux/pcred will be used.

Note: to prevent discovery of the LDAP administrator's credentials, the LDAP user DN and password may not be specified as command-line options to the ldapugdel utility.

Security Considerations

- Use of ldapugdel requires permissions of an LDAP administrator when it performs its operations on the directory server. The rights to delete or modify existing LDAP directory entries under the requested subtree, along with removal of the required attributes in that entry must be granted to the
ldapugdel(1M)

administrator identity that is specified when executing ldapugdel.

- As would occur in any identity repository, modification of this repository will likely have impacts as defined by the organization’s security policy. Users of ldapugdel are expected to have full knowledge of the organizations security policy and the impact of deleting identity information from that identity repository.

- Removal of a POSIX account will not automatically remove that account’s membership in groups, unless that capability is intrinsically provided by the directory server.

Note some directory servers have a feature called "referential integrity" which does perform modification/removal of DN-type attributes if the specified DN is either changed or removed.

- Never use ldapugdel as part of a modification process on a user or group entry (deleting and re-adding the entry as a method used to modify that entry.) User and group entries in an LDAP directory will often contain information about the user or group that is outside the POSIX information model. Deleting and re-adding an entry will delete all information about the user or group. When the entry is re-added, recovery of the non-POSIX information may not be possible.

- In order to support non-interactive use of the ldapugdel command, specification of the LDAP administrator's credentials is required through use of the LDAP_BINDDN and LDAP_BINDCRED environment variables. To prevent exposure of these environment variables, they should be unset after use.

Note also that shells command history log may contain copies of the executed commands that show setting of these variables. Access to a shell's history file must be protected. Specification of the LDAP administrator's credentials on the command line is not allowed since information about the currently running processes can be exposed externally from the session.

Use of the -P eliminates the need to set the mentioned environment variables by interactively prompting for the required credentials.

LDAP-UX PROFILE

ldapugdel makes use of the LDAP-UX configuration profile to determine the information model used in the directory server to store POSIX attributes. Please refer to the LDAP-UX Client Services Administrator's Guide for additional information about the configuration profile.

RETURN VALUE

Upon exit, ldapugdel returns the following:

0  Success. ldapugdel exits with no errors or with one or more warnings.

<>0  ldapugdel returns with a non-zero exit status if it encounters an error, and messages will be logged to stderr.

Messages will follow the below format:

ERROR:  code
        message

or

WARNING:  code
          message

Leading extra white space may be inserted to improve readability and follow 80 column screen formatting.

code will be a programmatically parsable error key-string, while

message will be human-readable. Refer to the LDAP-UX Client Services Administrator's Guide for a list of possible error codes generated by the LDAP user and group management tools.

EXTERNAL INFLUENCES

Environment Variables

LDAP_BINDDN  Specified the DN of a user with sufficient directory server privilege to delete users and/or groups in the LDAP directory server. While this variable is optional, if LDAP_BINDDN is specified, LDAP_BINDCRED must also be specified.
LDAP BINDCRED A password or other type of credential used for the user specified by the LDAP BINDDN. While this variable is optional, if LDAP BINDCRED is specified, LDAP BINDDN must also be specified.

Refer to Security Considerations for important security impacts when these environment variables are used.

LIMITATIONS

- Since LDAP directories require data be stored according to the UTF-8 (RFC3629) character encoding method, all characters provided to ldapugdel are assumed to be UTF-8 and part of the ISO-10646 character set. ldapugdel will not perform conversion of the locale character set to/from the UTF-8 character set.

- Refer to limitations described under -O above in reference to interoperability with Microsoft Services for Unix.

SEE ALSO

ldapcfinfo(1M), ldapugadd(1M), ldapuglist(1M), ldapugmod(1M), ldapux(5).
NAME
ldapuglist - display and enumerate POSIX-like account and group entries in an LDAP directory server

SYNOPSIS
[-n name | [-f | -F] filter] [-N maxcount] [attr...]

DESCRIPTION
ldapuglist is a command-line tool used to display and enumerate POSIX-like account and group entries that reside in an LDAP directory server.

Although ldapuglist provides similar output as compared with the ldapsearch command, it has been provided to meet a few specific feature requirements. These features allow applications to discover and evaluate account and group information stored in an LDAP directory server, without requiring intimate knowledge of the methods used retrieve and evaluate that information in the LDAP directory server.

Except for the optional trailing attr list, all parameters specified above are not positional dependent.

- ldapuglist uses the existing LDAP-UX configuration, requiring minimal command-line options to discover where to search for account/group information, such as which directory server(s) to contact and proper search filters for finding accounts and groups. This tool provides command options that allow you to alter these configuration parameters.
- ldapuglist uses the existing LDAP-UX authentication configuration to determine how to bind to the LDAP directory server.
- ldapuglist supports attribute mapping as configured by LDAP-UX. Fields returned from ldapuglist will use a consistent format, similar to that defined by RFC2307, even when different attributes are actually used to store the information in the directory server.

Note, that although that format is similar to LDIF, it is not LDIF. Major differences include:
- Objectclasses will not be displayed.
- By default only POSIX-related attributes will be displayed by ldapuglist, unless an attribute list is specifically requested on the command line.
- Output lines will not be broken after 80 columns.

Options
-m When -m is specified, ldapuglist will expose the names of the mapped attributes when returning results. Normally ldapuglist will return results as:

fieldname: value

where:
fieldname is one of the pre-defined RFC2307 attribute names.
value is the resulting value for that field, after attribute mapping has been applied.

With -m, the actual attribute name will be exposed as follows:

fieldname[attributename]: value

For example, if the RFC2307 attribute gecos has been mapped to the cn, l (location), and telephoneNumber attributes, without the -m option, the output of the gecos field would appear as:

gecos: value-of-cn, value-of-l, value-of-telephoneNumber,

When -m is used, and assuming the same conditions as above, the output representing the gecos field would appear as:

gecos[cn]: value-of-cn
gecos[l]: value-of-l
gecos[telephoneNumber]: value-of-telephoneNumber

Note that when a field has been mapped to multiple attributes, those attributes will appear in the order as defined in the LDAP-UX configuration.

The -m option does not apply if the -L option is specified.

-L Display the password or group output in the following formats:
ldapuglist(1M)

/etc/passwd format:
uid:userPasswd:gidNumbr:memberUid,...

/etc/group format:
cn:userPasswd:gidNumber,...

The -m option is ignored when the -L option is specified. The attr parameter list is invalid when the -L option is specified.

-P

Prompt for the bind identity (typically LDAP DN or Kerberos principal) and bind password.

Without -P, ldapuglist will discover the bind identity and password either from the environment variables LDAP_BINDDN and LDAP_BINDCRED. If the LDAPBINDDN or LDAP_BINDCRED environment variables have not been specified, ldapuglist will follow the bind configuration specified in the LDAP-UX configuration profile.

If LDAP-UX has specified "proxy" bind, the bind credential will be read from either the /etc/opt/ldapux/acred or /etc/opt/ldapux/pcred file. The acred file will only be used by users that have sufficient administrative privilege to read that file. Refer to Binding to the Directory Server below for additional details.

-Z

Requires an SSL connection to the directory server, even if the LDAP-UX configuration does not require the use of SSL.

Use of -Z requires either a valid directory server or CA certificate be defined in the /etc/opt/ldapux/cert8.db file. An error will occur if the SSL connection could not be established. Refer to Binding to the Directory Server below for additional details.

-ZZ

Attempt a TLS connection to the directory server, even if the LDAP-UX configuration does not require the use of TLS. If a TLS connection is unable to be established a non-TLS and non-SSL connection will be established.

Use of -ZZ is not recommended unless alternative methods are used to protect from network eavesdropping. Use of -ZZ requires either a valid server or CA certificate be defined in the /etc/opt/ldapux/cert8.db file. Refer to Binding to the Directory Server below for additional details.

-ZZZ

Requires a TLS connection to the directory server, even if the LDAP-UX configuration does not require the use of TLS.

Use of -ZZZ requires either a valid directory server or CA certificate be defined in the /etc/opt/ldapux/cert8.db file. An error will occur if the TLS connection could not be established. Refer to Binding to the Directory Server below for additional details.

Arguments

-t type

Specifies which service type ldapuglist will use to display or enumerate entries. The service type can be either passwd or group, where:

passwd implies posixAccount-type entries and,
group implies posixGroup-type entries.

Specification of the type parameter indicates how to handle processing of search filters (-f) and attribute mapping. If the -t option is not specified, ldapuglist will assume the passwd type.

-h hostname

Specifies the host name and optional port number (hostname:port) of the directory server. This option overrides the server list configured by LDAP-UX.

This field supports specification of IPv4 and IPv6 addresses. Note that when a port is specified for an IPv6 address, the IPv6 address must be specified in square-bracketed form. If the optional port is unspecified, the port number is assumed to be 389 or 636 for SSL connections (-Z). Refer to Binding to the Directory Server below for additional details.

-p port

Specifies the port number of the directory server to contact. This option is ignored if the port number is specified in the hostname as part of the -h option. Refer to Bindinding to the Directory Server below for additional details.

-n name

Provides a simplified method for discovering a single account or group. Use of -n is the same as -f"(uid={name})" for accounts and -f"(cn={name})" for groups.
ldapuglist(1M)

-F and -f may not be specified on the command line if -n is used.

-b base

This option overrides the search base as defined in the LDAP-UX configuration. base is a
distinguished name (DN) that describes the highest location in the directory tree where to
start the search. If unspecified, ldapuglist will use the defaultSearchBase as defined
in the LDAP-UX configuration profile.

-s scope

This option overrides the search scope as defined in the LDAP-UX configuration. scope
specifies how deep in the directory tree ldapuglist should search. scope may be one of
base, one, or sub, where:

- base only performs a search on the base specified above,
- one searches all entries that are child entries of the base, and
- sub searches all entries below, including the base.

-f filter

Specifies an LDAP-style search filter, filter, used to select specific entries from the LDAP
directory. When -f is used, the filter specified by filter is assumed to apply to either
POSIX-style users or groups (depending on if the -t passwd or -t group option is
specified). This means the filter specified with -f will be amended with the default LDAP-
UX search filter for either the user or group object types.

In addition, when -f is used, if a known attribute for the particular service (see the lists
defined under OUTPUT FORMAT), has been mapped as defined by the LDAP-UX
configuration profile, then the mapped attribute name will be substituted in the search
filter.

Using an example with the following command:

```
ldapuglist -t passwd -f "(uidNumber=52345)"
```

And assuming the LDAP-UX product has been configured as follows:

- The configuration profile defines the search filter for the passwd service as
  `(objectclass=posixAccount)`
- The uidNumber attribute for the passwd service has been mapped to the employeeNumber attribute.

Then the actual search filter used by ldapuglist would be:

```
&(objectclass=posixAccount)(employeeNumber=52345)
```

The -f option also supports generation of search filters for multi-mapped attributes, gecos
and memberUid. In the case of gecos, each mapped attribute would be used in the search
filter using the LDAP and operation (&). And in the case of memberUid, each mapped
attribute would be used in the search filter using the LDAP or operation (|).

For an example using gecos: assume gecos has been mapped to cn, l, and telephoneNumber. If the argument to -f is `gecos=Jane Smith,BLD-5D,555-1212)`, then the resulting search filter presented to the LDAP directory server would be:

```
&(objectclass=posixAccount)(&(cn=Jane Smith)(l=BLD-5D)(telephoneNumber=555-1212))
```

Using an example for memberUid, assume memberUid has been mapped to member and
memberUid. If the argument to -f is `(memberUid=jsmith)`, then the resulting
search filter presented to the LDAP directory server would be:

```
&(objectclass=posixGroup)(!(member=cn=Jane Smith,ou=people,ou=myorg,dc=myco,dc=com)
(memberUid=jsmith))
```

NOTES:

- When -f is used and any of the attributes specified in the search filter have been
  mapped to *NULL*, ldapuglist will return an error.
- Attributes that are not part of the LDAP-UX configuration profile mapping will not be
  modified. Refer to RFC2307: An Approach for Using LDAP as a Network Information
  Service for the list of attributes that may be mapped.

HP-UX 11i Version 2: December 2007 Update
Specifying `-n` and `-f` on the same command line will result in an error.

`-F filter`
Similar to `-f`, except that `filter` is assumed to be immutable, and neither the LDAP-UX user nor group filter from the configuration profile will be amended to the specified filter, nor will attribute mapping apply to the `filter`.

**NOTES:**
- When `-F` is used, the specified `filter` should still apply to either user or group entries and match the `-t passwd` or `-t group` option. In other words, `ldapuglist` will produce unpredictable results if the search filter specified with `-F` discovers group entries, but the `-t passwd` option was specified.
- Specifying `-n` and `-F` on the same command line will result in an error.

`-N maxcount`
This option specifies the maximum number of entries to be returned. If this option is not specified, the maximum number of entries to be returned is 200 by default.

Some directory servers will limit the number of entries returned for a particular search request, regardless of how many entries are requested. If the `maxcount` limit is set too high, it may not be possible to determine if a search has returned complete results, since the directory server may have truncated the number of returned entries before reaching the requested maximum count.

Although some directory servers will indicate if a specified search exceeds an enumeration limit, if `maxcount` is above the directory servers internal configured limit, it is not always possible to determine if a search has returned complete results, since the directory server may have truncated the number of returned entries before reaching the requested maximum count.

When the `-m` option is specified, the output format for a value specified by an `attr` will always be in the form:

```
attributename[attributename]: value
```

**Note:** `ldapuglist` does not allow use of the `attr` parameter when `ldapuglist` binds to the directory server using the LDAP-UX proxy user. This limitation prevents regular HP-UX users from discovering LDAP data that was previously not displayed by LDAP-UX. Use of the `attr` parameter requires either that the user has permission to use the LDAP-UX Administrator Credential (`/etc/opt/ldapux/acred`) or that the user specifies an identity using the `-P` or `LDAP_BINDDN` and `LDAP_BINDCRED` environment variables when running `ldapuglist`.

**Binding to the Directory Server**

`ldapuglist` has been designed to take advantage of the existing LDAP-UX configuration for determining to which directory server to bind and how to perform the bind operation. `ldapuglist` will consult the LDAP-UX configuration profile for the following information:

- The list of LDAP directory server hosts.
- The authentication method (simple passwords, SASL Digest MD5, etc.).

If either of the environment variables `LDAP BINDDDN` and `LDAP BINDCRED` have not been specified, `ldapuglist` will also consult the LDAP-UX configuration for additional information:

- The type of credential (user, proxy or anonymous) to use.
- The credential used for binding as a proxy user (either `/etc/opt/ldapux/acred` for administrative users or `/etc/opt/ldapux/pcred` for non-privileged users.)

`ldapuglist` will display an error message if `LDAP BINDDDN` has been specified and `LDAP BINDCRED` has not, unless the `-P` option has been specified.

As with LDAP-UX, `ldapuglist` will attempt to contact the first available directory server as defined in the LDAP-UX host list. As soon as a connection is established, further directory servers on the host list will not be contacted.
Once connected, `ldapuglist` will first determine if the environment variables `LDAP_BINDDN` and `LDAP_BINDCRED` have been specified. If so, then `ldapuglist` will attempt to bind to the directory server using the specified credentials and configured LDAP-UX authentication method.

If the above mentioned environment variables have not been specified, then `ldapuglist` will determine if the configured credential type is "proxy" and if so, attempt to bind to the directory server using the configured LDAP-UX proxy credential. If configured, the acred proxy credential will be used for administrative users (determined if the user running `ldapuglist` has enough privilege to read the `/etc/opt/ldapux/accred` file). Otherwise the credential configured in `/etc/opt/ldapux/pcred` will be used.

Note: to prevent discovery of the LDAP administrator's credentials, the LDAP user DN and password may not be specified as command-line options to the `ldapuglist` utility.

Security Considerations

In order to support non-interactive use of the `ldapuglist` command, specification of the LDAP administrator's credentials is required through use of the `LDAP_BINDDN` and `LDAP_BINDCRED` environment variables. To prevent exposure of these environment variables, they should be unset after use.

Note also that `shells` command history log may contain copies of the executed commands that show setting of these variables. Access to a shell's history file must be protected. Specification of the LDAP administrator's credentials on the command line is not allowed since information about the currently running processes can be exposed externally from the session.

Use of the `-P` eliminates the need to set the mentioned environment variables by interactively prompting for the required credentials.

LDAP-UX PROFILE

`ldapuglist` makes use of the LDAP-UX configuration profile to determine the information model used in the directory server to store POSIX attributes. Please refer to the LDAP-UX Client Services Administrator's Guide for additional information about the configuration profile.

OUTPUT FORMAT

Output from `ldapuglist` will follow a consistent format, regardless of which attributes are used to define information in an LDAP directory. The output format is:

```
  dn: dn1
     field1: value1
     field2: value2
     field3:: base64-encoded-value3

  dn: dn2
     field1: value1
     field2: value2
```

Each entry will be preceded by a DN, followed by one or more field-value pairs. The DN and each field-value pair will be on a separate line, separated by a carriage-return and line-feed character. The field and value will be separated by a colon and space character. And each entry will be separated by a blank line.

In the case when an unencodable character is encountered (carriage-return or line-feed for example) in a value string, the whole value will be base64 encoded and the field-value separator will change to two colons and a space character. See Unencodable Characters below. When the `-t passwd` option is specified, the following fields will be returned:

```
en
cn
uid
userPassword
uidNumber
gidNumber
homeDirectory
loginShell
gecos
```

When the `-t group` option is specified, the following fields will be returned:

```
en
userPassword
```
Note that when the -m option is specified, the output format will change (for both users and groups) to:

dn: dn1
field1[attribute1]: value1
field2[attribute2]: value2
field3[attribute3]: base64-encoded-value3

Special Considerations for Output Format

Multi-Valued Attributes
Although some of the attributes used in LDAP directory servers are considered multi-valued attributes, the ldapuglist tool will only display the first value discovered for each RFC2307 attribute for each entry, since these fields appear only once in a POSIX account or group.

For non-RFC2307 attributes (those specified via the attr argument list), if the attribute is multi-valued, multiple values will be displayed. Also note that this rule does not apply to the memberUid field since POSIX groups may have multiple members.

Since the gecos attribute can be mapped to multiple attributes, the gecos field may appear multiple times in an entry if the -m option is used, once for each mapped attribute. Example:

gecos[cn]: Bill Smith
gecos[l]: Building 6A
gecos[telephoneNumber]: +1-555-555-4321

Non-POSIX Accounts & Groups
With the -F option, ldapuglist can be used to display users and groups that are not posixAccounts or posixGroups. Thus, these entries may not contain the required fields that store POSIX account and group information (such as the uidNumber). When displaying these entries, the specified fields will be missing from the output.

As non-POSIX accounts and groups are not required to contain POSIX attributes, use of the -L option may result in unexpected output. Data between the : characters may be empty, such as :::::

UTF-8
Since LDAP directories require data be stored according to the UTF-8 (RFC3629) character encoding method, all characters displayed by ldapuglist will be UTF-8, and assumed to be part of the ISO-10646 character set. ldapuglist will not perform conversion of the locale character set to/from the UTF-8 character set.

Unencodable Characters (Base64 Encoding)
In the output format of ldapuglist each displayed field will be delimited by a new line (carriage-return and line-feed). In order to assure that ldapuglist displays only printable and LDIF encodable characters, all characters less than 32 (ASCII space), except for 9 (ASCII horizontal tab) and the character 127 (ASCII delete) will result in the value being converted into a base-64 encoded string. Characters above 127 are assumed be from the UTF-8 character set, and assumed printable. If the output lines are long, the data will not be broken into multiple lines.

Encoding of the DN
ldapuglist will display DN strings according to the encoding rules defined in RFC4514. The escape character (\) will precede special characters, which may be the character itself or a 2 digit hex representation of the character.

Passwords
In many cases, ldapuglist will not be able to access the user or group password fields. This can occur when:

- ldapuglist has insufficient privilege to access the password field
- The passwords are not used to authenticate users (such as when X.500 certificates).
- The password is not stored in the LDAP directory server. The password might be stored in a third-party repository such as a Kerberos KDC.
ldapuglist(1M)

- The password is stored in a format un-parsable by HP-UX (such as SSHA, the Salted Secure Hash Algorithm).

If the password is not available to ldapuglist, the userPassword field will not be displayed. If the -L option is specified, the password field will contain the ‘x’ character. Existence or lack of the password field can not be used to determine if an account is active or inactive.

RETURN VALUE
Upon exit, ldapuglist returns the following:

0 Success. ldapuglist exits with no errors or with one or more warnings.

<>0 ldapuglist returns with a non-zero exit status if it encounters an error, and messages will be logged to stderr.

Messages will follow the below format:

ERROR: code message

or

WARNING: code message

Leading extra white space may be inserted to improve readability and follow 80 column screen formatting.

code will be a programmatically parsable error key-string, while

message will be human-readable. Refer to the LDAP-UX Client Services Administrator's Guide for a list of possible error codes generated by the LDAP user and group management tools.

EXTERNAL INFLUENCES
Environment Variables

LDAP_BINDDN Specified the DN of a user with sufficient directory server privilege to discover and enumerate users and/or groups in the LDAP directory server. While this variable is optional, if LDAP_BINDDN is specified, LDAP_BINDCRED must also be specified.

LDAP_BINDCRED A password or other type of credential used for the user specified by LDAP_BINDDN. While this variable is optional, if LDAP_BINDCRED is specified, LDAP_BINDDN must also be specified.

Refer to Security Considerations for important security impacts when these environment variables are used.

EXAMPLES
Specifying LDAP_BINDDN and LDAP_BINDCRED:

    # export LDAP_BINDDN="cn=Directory Manager"
    # export LDAP_BINDCRED="password"
    # ldapuglist -f "(uid=apierce)" sn
    dn: cn=Alan Pierce,ou=people,ou=IT,dc=FutureWidget,dc=com
    cn: Alan Pierce
    uid: apierce
    uidNumber: 22014
    gidNumber: 318
    homeDirectory: /home/apierce
    loginShell: /usr/bin/ksh
    gecos: Alan Pierce,San Francisco,+1 505-555-6525
    sn: Pierce

Using the -m option on the same entry, assuming that the uidNumber has been mapped to employeeNumber and gecos has been mapped to cn, l, and telephoneNumber.

    # export LDAP_BINDDN="cn=Directory Manager"
    # export LDAP_BINDCRED="password"
    # ldapuglist -u -m -f "(uid=apierce)"
    dn: cn=Alan Pierce,ou=people,ou=IT,dc=FutureWidget,dc=com
Listing all POSIX accounts held by users in San Francisco.

```
# ldapuglist -f "(l=Sann Francisco)"
```

```
dn: cn=Alan Pierce,ou=people,ou=IT,dc=FutureWidget,dc=com
    cn: Alan Pierce
    uid: apierce
    uidNumber: 22014
    gidNumber: 318
    homeDirectory: /home/apierce
    loginShell: /usr/bin/ksh
    gecos: Alan Pierce,San Francisco,+1 505-555-6525
```

```
dn: cn=Manuel Wolters,ou=people,ou=IT,dc=FutureWidget,dc=com
    cn: Manuel Wolters
    uid: mwolters
    uidNumber: 2284
    gidNumber: 212
    homeDirectory: /home/mwolters
    loginShell: /usr/bin/ksh
    gecos: Manuel Wolters,San Francisco,+1 505-555-5072
```

```
dn: cn=Joanie Lin,ou=people,ou=IT,dc=FutureWidget,dc=com
    cn: Joanie Lin
    uid: jlin
    uidNumber: 2840
    gidNumber: 229
    homeDirectory: /home/jlin
    loginShell: /usr/bin/ksh
    gecos: Joanie Lin,San Francisco,+1 505-555-1111
```

Listing an account that does not contain POSIX attributes.

```
# ldapuglist -m -F "(uid=apierce)"
```

```
dn: cn=Alan Pierce,ou=people,ou=IT,dc=FutureWidget,dc=com
    cn: Alan Pierce
    uid: apierce
    gecos: Alan Pierce
```

Listing an account that does not contain POSIX attributes using the -L option.

```
# ldapuglist -m -L -F "(uid=apierce)"
apierce:x:::Alan Pierce,San Francisco,+1 505-555-6525::
```

Listing a posixGroup which has its members defined using the X.500-style syntax. Note that ldapuglist will map DN-style membership to account names.

```
# ldapuglist -t group -f "(cn=mygroup1)"
```

```
dn: cn=mygroup1,ou=groups,ou=IT,dc=FutureWidges,dc=com
    cn: mygroup1
    gidNumber: 542
    memberUid: mdiaz
    memberUid: apierce
    memberUid: bjones
```

```
486 Hewlett-Packard Company – 8 – HP-UX 11i Version 2: December 2007 Update
```
Listing **posixGroups** which have Mike Diaz (**uid=mdiaz**) as a member defined using **memberUid** syntax.

```bash
# ldapuglist -t group -f "(memberUid=mdiaz)"
```

Listing **posixGroups** which have Mike Diaz (**uid=mdiaz**) as a member defined using the X.500-style (**member** or **uniqueMember**) syntax.

```bash
# ldapuglist -t group -m \
-f "((member=cn=Mike Diaz,ou=People,ou=IT,dc=FutureWidget,dc=com))"
```

Listing regular **posixGroups** which have Mike Diaz (**uid=mdiaz**) as a member:

```bash
# ldapuglist -t group -f "(cn=mygroup1)"
```

Listing a group that does not have the required **posixGroup** attributes.

```bash
# ldapuglist -t group \
-F "((cn=mygroup2)(objectClass=groupOfUniqueNames))"
```

**LIMITATIONS**

- **ldapuglist** will not support enumeration of members of a dynamic group, such as those defined by the attributes: **memberUrl**, **nssSearchFilter**, **msDS-AzLDAPQuery**, etc.
- **ldapuglist** will not perform conversion of the locale character set to/from the UTF-8 character set.
ldapuglist will not display attributes to which it does not have access rights in the LDAP directory server. Be sure to specify administrator credentials with sufficient privileges in the LDAP directory to view the requested attributes.

SEE ALSO
ldapcfinfo(1M), ldapugadd(1M), ldapugdel(1M), ldapugmod(1M), ldapux(5).
NAME
ldapugmod - modify existing POSIX accounts or groups in an LDAP directory server

SYNOPSIS
ldapugmod [-t passwd] [options] [-h hostname] [-p port] [-f full_name] [-n name]
[-u uidNumber] [-g group/gid] [-s login_shell] [-d home_directory] [-m] [-I gecos]
[-c comment] [[-A attrval] [...] ] [[-R attrval] [...] ] [-D DN | uid_name] [attr=value [...] ]
[
ldapugmod -t group [options] [-h hostname] [-p port] [-n new_name] [-g gidNumber]
[-c comment] [-a member[,]...] [-r member[,]...] [[-A attrval] [...] ]
[[[-R attrval][,]...]] [-D DN | group_name] [attr=value [...] ]
]

DESCRIPTION
ldapugmod allows HP-UX administrators to modify existing POSIX accounts or groups in an LDAP directory server.

When using extended options, ldapugmod can also be used to modify arbitrary attributes for user or group entries.

Users of ldapugmod are required to provide LDAP administrator credentials that have sufficient privilege to perform the user or group modify operations in the LDAP directory server.

Options
-P Prompt for the administrator's bind identity (typically LDAP DN or kerberos principal) and bind password.

Without -P ldapugmod will discover the bind identity and password from the environment variables LDAP_BINDDN and LDAP_BINDCRED. If either the LDAP_BINDDN or LDAP_BINDCRED environment variables have not been specified, ldapugmod will follow the bind configuration specified in the LDAP-UX configuration profile.

If LDAP-UX has specified "proxy" bind, the bind credential will be read from either the /etc/opt/ldapux/acred or /etc/opt/ldapux/pcred file. The acred file will only be used by users that have sufficient administrative privilege to read that file.

Refer to Binding to the Directory Server below for additional details.

-PP Prompt for the password of the user or group being modified. If the -PP option is not specified, the password for the modified user or group will be retrieved from the LDAP_UGCRED environment variable if the -PW option is specified.

Use of -PP implies the use of -PW.

-PW Change the user or group password attribute.

Also, if LDAP-UX attributed mapping for the userPassword attribute has not been defined or set to "NULL", ldapugmod will create new passwords in the userPassword attribute.

If -PW is specified, either the LDAP_UGCRED environment variable or the -PP option must be specified.

-O With ldapugmod, it is possible to extend posixAccount and posixGroup attributes to a user or group entry that does not already contain the posixAccount or posixGroup object class and respective attributes (depending on if the -t passwd or -t group option) to the entry being modified.

Note that when used with Active Directory service, if the user or group entry is built using the abstract "User" or "Group" class. ldapugmod will assume that the abstract class already includes the required MS SFU attributes, and thus will not add the posixAccount or posixGroup objectclass to the entry.

-Z Requires an SSL connection to the directory server, even if the LDAP-UX configuration does not require the use of SSL. Use of -Z requires either a valid server or CA certificate be defined in the /etc/opt/ldapux/cert8.db file.

An error will occur if the SSL connection could not be established. Refer to Binding to the Directory Server below for additional details.

HP-UX 11i Version 2: December 2007 Update - 1 - Hewlett-Packard Company 489
-ZZ Attempt a TLS connection to the directory server, even if the LDAP-UX configuration does not require the use of TLS. If a TLS connection is unable to be established a non-TLS and non-SSL connection will be established.

Use of -ZZ is not recommended unless alternative methods are used to protect from network eavesdropping. Use of -ZZ requires either a valid server or CA certificate be defined in the /etc/opt/ldapux/cert8.db file. Refer to Binding to the Directory Server below for additional details.

-ZZZ Requires a TLS connection to the directory server, even if the LDAP-UX configuration does not require the use of TLS.

Use of -ZZZ requires either a valid server or CA certificate be defined in the /etc/opt/ldapux/cert8.db file. An error will occur if the TLS connection could not be established. Refer to Binding to the Directory Server below for additional details.

-N Allows renaming of the RDN (Relative Distinguished Name) of an LDAP directory entry. In some cases, when an attribute is modified, it may be the same attribute that is used in the RDN portion of the entry's distinguished name. Changing the attribute and value that is used in the RDN requires changing the RDN.

For example, suppose an entry in the directory server is named:

cn=Robert Smith,ou=Marketing,dc=acme,dc=com

If the cn attribute is changed to cn=Bob Smith then the DN would also need to change to:

cn=Bob Smith,ou=Marketing,dc=acme,dc=com

Modification of an RDN is generally discouraged since the DN is often used as a unique way to identify the entry in the directory server. Often the DN is used to define membership in a group. So to prevent accidental changing of the DN, the -N option must be specified to allow changing of the RDN. When the DN of an entry changes, the group membership information for this entry may become inconsistent.

However, most directory servers have the inherent ability to update all entries that refer to the updated DN of a changed entry. So ldapugmod will not attempt to perform modifications to other entries in the directory server that refer to this entry by its DN.

NOTE: ldapugmod will not allow renaming of multi-valued RDNs; for example, an RDN of cn=test1+cn=test2 is not supported.

-F Force modification of the user or group entry even if particular error conditions occur. The error conditions that can be overridden are:

- The changed user name or group name already exists in the directory server.
- The changed user id or group id number already exists in the directory server.
- Adding a member to a group when that member is not defined in the LDAP directory. In this case, membership will always be defined using the memberUid attribute, regardless of attribute mapping defined for group membership.
- When modifying the group of a user with a group ID that can not be found in any name service repository. In this case, the group ID number must be specified.

Note that some directory servers perform their own attribute and RDN uniqueness checks. In this case, even if the -F option is specified, if the directory server detects a collision ldapugmod will be unable to modify the specified entry.

-S Upon successful completion, display the DN of the updated entry.

Arguments

-t type Specifies if the command-line arguments are applicable to modifying user or group. type is expected to be either passwd or group. If unspecified, ldapugmod defaults to passwd.

Note: to be consistent with the Name Service Switch (see switch(4)), the term passwd (instead of user) is used to represent LDAP user entries which contain POSIX account-related information.
-h hostname Specifies the host name and optional port number (hostname:port) of the directory server. This option overrides the server list configured by LDAP-UX.

The hostname field also supports specification of IPv4 and IPv6 addresses. Note that when a port is specified for an IPv6 address, the IPv6 address must be specified in square-bracketed form.

If the optional port is unspecified, the port number is assumed to be 389 or 636 for SSL connections (-Z). Refer to Binding to the Directory Server below for additional details.

-p port Specifies the port number of the directory server to contact. This option is ignored if the port number is specified in the hostname as part of the -h option. Refer to Binding to the Directory Server below for additional details.

-n name Specifies the new name of the user or group. This option will replace the uid attribute for user entries and the cn attribute for group entries, or the mapped attribute if attribute mapping has been specified for that attribute.

Use of -n is the same as replacing the corresponding attribute. For example, assuming no attribute mapping:

```
ldapugmod -t passwd -n newuid olduid
```

is the same as:

```
ldapugmod -t passwd olduid uid=newuid
```

-A attrval Specifies an attribute and value to be added to an entry. The format of attribute=value, where attribute is the name of the attribute to add, and value is the specific instance of that attribute.

The -A option is used when working with multi-valued attributes, to add a new value for a multi-valued attribute, without removing already existing values for that attribute.

Note that use of the -A option interacts with the optional attr=value parameters. See attr=value below. The -A option may be specified more than once per command line. The value portion of attrval may be an empty string.

-R attrval Specifies an attribute or specific values of an attribute to be removed from the entry. The format of attribute=value, where attribute is the name of the attribute to remove, and value is the specific instance of that attribute, if the attribute is multi-valued.

Note that use of the -R option interacts with the optional attr=value parameters. See attr=value below. The -R option may be specified more than once per command line.

-D DN Normally ldapugmod will search for the named user or group using the search rules described by the service search descriptor in the LDAP-UX configuration profile. With -D the exact DN of the entry being modified may be specified.

If the -D option is specified, the uid_name or group_name parameter should not be specified.

Options Applicable to '-t passwd'

- f full_name Replaces the user's full name. If full_name is an empty string (a pair of double quotes: ""), ldapugmod will remove the cn (or mapped) attribute.

Note, refer to the WARNING section below for impacts when using this option.

-u uidNumber Replaces the user's numeric id number. If uidNumber is an empty string (a pair of double quotes: ""), ldapugmod will remove the uidNumber (or mapped) attribute. If the specified uidNumber already exists in the directory server, ldapugmod will not modify the entry and return an error exit status, unless the -F option is specified.

Note, refer to the WARNING section below for impacts when using this option.

-g group/gid Replace the user's primary login group id number. If group/gid is an empty string, ldapugmod will remove the gidNumber (or mapped) attribute.

In order to support numeric group names, ldapugmod treats the argument to -g as a group name. If a numeric group name can not be found that matches the argument specified, ldapugmod checks to see if the value is numeric and then checks to see if the
ldapugmod(1M)

group ID number specified exists. If not, ldapugmod will exit with an error, unless the
-F option has been specified.

Note, ldapugmod does not modify the user's group membership when chaining the pri-
mary group ID. Adding the user as a member of the new group, and possibly removing the
member from the previous group, must be done with separate ldapudmod operations.

Refer to the WARNING section below for additional impacts when using this option.

-s login_shell Replaces the full path name to the executable that will be used to handle login sessions for
this user. If login_shell is an empty string, ldapugmod will remove the loginShell (or
mapped) attribute. ldapudmod will issue a WARNING if the specified login shell does not
exist on the local system.

Note, refer to the WARNING section below for impacts when using this option.

-d home_directory
Replaces the full path name (including the user name) of the user's home directory. If
home_directory is an empty string, ldapugmod will remove the homeDirectory (or
mapped) attribute.

Note, refer to the WARNING section below for impacts when using this option.

-m
Move the user's home directory to the location specified with the -d option. -m requires
the -d option to be specified. If the specified home_directory already exists, the user's current
home directory does not exist or the user running ldapugmod does not have sufficient
permissions to move the directory, ldapugmod will return an error.

-I gecos
Replaces the GECOS field(s) for the user. If gecos is an empty string, ldapugmod will
remove the gecos (or mapped) attribute(s). Typically the GECOS contains four fields which
represent (in order):
  - The user's full name
  - The user's work location
  - The user's work telephone number
  - The user's home telephone number (often omitted)

Each field in the gecos must be separated by a comma. Although each field value specified
within the gecos can contain white space (such as "Bill Smith,Building 6,555-1234"), white
space should not be used between the each field and the separating commas (such as "Bill
Smith, Building 6,555-1234").

Note that LDAP-UX supports mapping of the gecos field to multiple attributes. If attribute
mapping has been specified in the LDAP-UX configuration profile, each field will be mapped
to its representative attribute, in the order specified.

WARNING: If the -I option is specified and attribute mapping has been defined for the
gecos attribute, be careful not to specify the same attributes and values in the command
line that are also used in the gecos map. For example, suppose the gecos has been mapped
to cn, l and telephoneNumber. The following command might produce unpredictable
results:

ldapugmod -I "Jim Smith,Boston,55-5-1234" jsmith \
"cn=Jim Smith" "sn=Smith" \
"telephoneNumber=555=1234"

In the above example, because of the gecos attribute mapping, the cn and
telephoneNumbers are specified twice and will result an error when the same attribute and
value are added to the directory server. ldapcfinfo can be used to determine gecos
attribute mapping configuration.

If gecos is an empty string, ldapugmod will remove the gecos or implied mapped, attri-
butes. Note that this use of the -I option is discouraged, since the gecos attribute is often
mapped to required attributes.

Since the gecos attribute may be mapped to one or several attributes, the number of values
specified with -I (between the commas) should, but is not required to, match the number
of mapped attributes. If there are more mapped attributes than specified values in -I, then
trailing mapped attributes will be removed from the directory server. If there are
more values that mapped attributes, extra values will be combined in the last mapped
attribute.

Note, refer to the WARNING section below for impacts when using this option.

- **c** **comment**
  Replaces the comment that will be stored in the description attribute, as defined by RFC2307. Attribute mapping is not defined for the description attribute.

Note, refer to the WARNING section below for impacts when using this option.

**uid_name**
Contains the POSIX-style textual login name of the user entry to modify. This user name should conform to HP-UX login name requirements. Please refer to `passwd(4)` for login name requirements. The `uid_name` is a required parameter unless the `-D` option is specified.

**attr=value**
Allows modification of arbitrary LDAP attributes and values. `value` may be an empty string. However this usage will not remove attributes and their values from the directory server. Instead, use the `-R` option to remove arbitrary attributes.

Note, refer to the WARNING section below for impacts when using this option.

### Options Applicable to `-t group`

**-g** **gidNumber**
Replaces the group’s numeric id number. If the specified `gidNumber` already exists in the directory server, `ldapugmod` will not modify the entry and return an error exit status, unless the `-F` option is specified.

Note, refer to the WARNING section below for impacts when using this option.

**-a** **member[, ...]**
Adds one or more members to the specified group. `ldapugmod` will follow the same membership syntax as defined by LDAP-UX mapping. Specifically, if LDAP-UX has mapped the RFC2307 group membership attribute (memberUid) to a DN-based membership attribute such as `member` or `uniqueMember`, then `ldapugmod` will define membership using the DN of the specified user.

When specifying a list of members, the list must be comma separated with no white-space. Even though LDAP-UX supports mapping of the `memberUid` attribute to multiple attributes simultaneously, `ldapugmod` will only use the first mapped attribute when defining membership in the group. If the specified `member` does not exist in the LDAP directory, `-F` must be used to define the member, and only the `memberUid` attribute syntax will be used.

- `-a` only supports membership defined using static group membership structures, such as `memberUid`, `member`, `uniqueMember`. Dynamic group membership, such as represented by `memberUrl`, is not supported by `ldapugmod`.

**-r** **member[, ...]**
Removes one or more members from the specified group. `ldapugmod` will search for membership in the group defined using the `memberUid`, `member`, `uniqueMember`, and `msSFU30posixMember` attributes and remove all values that represent the specified user (either DN or uid name).

`ldapugmod` consults the LDAP-UX configuration profile for attribute mapping to determine which attributes should be modified to remove the user’s membership. When specifying a list of members, the list must be comma separated with no white-space.

**-c** **comment**
Replaces the comment that will be stored in the description attribute, as defined by RFC2307. Attribute mapping is not defined for the `description` attribute. If `comment` is an empty string, `ldapugmod` will remove the description (or mapped) attribute.

Note, refer to the WARNING section below for impacts when using this option.

**group_name**
Contains the POSIX-style textual group name for the group entry to modify. This name should conform to HP-UX group name requirements. Please refer to `group(4)` for group name requirements. `group_name` is a required parameter when used with the `-t group` option. The `group_name` should not be specified if the `-D` option is specified.

**attr=value**
Allows modification of arbitrary LDAP attributes and values. Refer to `attr=value` in the section above for additional information.

Note, refer to the WARNING section below for impacts when using this option.
**Binding to the Directory Server**

`ldapugmod` has been designed to take advantage of the existing LDAP-UX configuration for determining to which directory server to bind and how to perform the bind operation. `ldapugmod` will consult the LDAP-UX configuration profile for the following information:

- The list of LDAP directory server hosts.
- The authentication method (simple passwords, SASL Digest MD5, etc.)

If either of the environment variables `LDAP_BINDDN` or `LDAP_BINDCRED` have not been specified, `ldapugmod` will also consult the LDAP-UX configuration for additional information:

- The type of credential (user, proxy or anonymous) to use.
- The credential used for binding as a proxy user (either `/etc/opt/ldapux/acred` for administrative users or `/etc/opt/ldapux/pcred` for non-privileged users.)

As with LDAP-UX, `ldapugmod` will attempt to contact the first available directory server as defined in the LDAP-UX host list. As soon as a connection is established, further directory servers on the host list will not be contacted.

Once connected, `ldapugmod` will first determine if the environment variables `LDAP_BINDDN` and `LDAP_BINDCRED` have been specified. If so, then `ldapugmod` will attempt to bind to the directory server using the specified credentials and configured LDAP-UX authentication method. If the above mentioned environment variables have not been specified, then `ldapugmod` will determine if the configured credential type is "proxy" and if so, attempt to bind to the directory server using the configured LDAP-UX proxy credential.

If configured, the acred proxy credential will be used for administrative users (determined if the user running `ldapugmod` has enough privilege to read the `/etc/opt/ldapux/acred` file). Otherwise the credential configured in `/etc/opt/ldapux/pcred` will be used.

Note, to prevent discovery of the LDAP administrator's credentials, the LDAP user DN and password may not be specified as command-line options to the `ldapugmod` utility.

**Security Considerations**

- Use of `ldapugmod` requires permissions of an LDAP administrator when it performs its operations on the directory server. The rights to modify existing LDAP directory entries under the requested subtree, along with creation, modification and removal of the required attributes in that entry must be granted to the administrator identity that is specified when executing `ldapugmod`.

- Note that as with any POSIX-type identity, the user and group ID number specified is used by the HP-UX operating system to determine rights and capabilities in the OS as well as in the file system.

  For example, a the root user ID 0, typically has unlimited OS administration and file access rights. Before modifying an entry, be aware of the selected user and group ID number and any policy that may be associated with that ID.

- Modification (renaming) of a POSIX account will not automatically modify that account's membership in groups, unless that capability is intrinsically provided by the directory server.

  Note some directory servers have a feature known as "referential integrity," which does perform modification/removal of DN-type attributes if the specified DN is either changed or removed.

- As would occur in any identity repository, modification of this repository will likely have impacts as defined by the organizations security policy. Users of `ldapugmod` are expected to have full knowledge of the organizations security policy the impact of modifying identity information in that identity repository.

- As would occur in any identity repository, modification of this repository will likely have impacts as defined by the organization's security policy.

  For example, adding a new user with an user ID number shared with that of a secured application may impact the security of that application. Users of `ldapugmod` are expected to have full knowledge of the organizations security policy the impact of modifying identity information in that identity repository.

- In order to support non-interactive use of the `ldapugmod` command, specification of the LDAP administrator's credentials is required through use of the `LDAP_BINDDN` and `LDAP_BINDCRED` environment variables. To prevent exposure of these environment variables, they should be unset after use.
Note also that shells command history log may contain copies of the executed commands that show setting of these variables. Access to a shell’s history file must be protected. Specification of the LDAP administrator’s credentials on the command line is not allowed since information about the currently running processes can be exposed externally from the session.

Use of the `-P` eliminates the need to set the mentioned environment variables by interactively prompting for the required credentials.

**LDAP-UX PROFILE**

`ldapugmod` makes use of the LDAP-UX configuration profile to determine the information model used in the directory server to store POSIX attributes. Please refer to the *LDAP-UX Client Services Administrator’s Guide* for additional information about the configuration profile.

**EXTERNAL INFLUENCES**

**Environment Variables**

- **LDAP_UGCRED**: When used in combination with the `-PW` option, `LDAP_UGCRED` specifies the password of a user or group which need to be modified.
  
  Note, use of passwords for groups is not recommended.

- **LDAP_BINDDN**: Specified the DN of a user with sufficient directory server privilege to create new users and/or groups in the LDAP directory server. While this variable is optional, if `LDAP_BINDDN` is specified, `LDAP_BINDCRED` must also be specified.

- **LDAP_BINDCRED**: A password or other type of credential used for the user specified by the `LDAP_BINDDN`. While this variable is optional, if `LDAP_BINDCRED` is specified, `LDAP_BINDDN` must also be specified.

Refer to *Security Considerations* for important security impacts when these environment variables are used.

**RETURN VALUE**

Upon exit, `ldapugmod` returns the following:

- **0**: Success. `ldapugmod` exits with no errors or with one or more warnings.
- **<>0**: `ldapugmod` returns with a non-zero exit status if it encounters an error, and messages will be logged to stderr.

Messages will follow the below format:

```
ERROR: code message
```

or

```
WARNING: code message
```

Leading extra white space may be inserted to improve readability and follow 80 column screen formatting.

- `code` will be a programmatically parsable error key-string, while
- `message` will be human-readable. Refer to the *LDAP-UX Client Services Administrator’s Guide* for a list of possible error codes generated by the LDAP user and group management tools.

**WARNINGS**

Under common usage, `ldapugmod` uses the LDAP replace operation when changing values of an attribute in an entry. This feature can impact attributes that have multiple values, by removing all occurrences of an attribute value and replacing it with the one specified on the `ldapugmod` command line.

For example, if the `-n` argument is used to specify a new name for a posixGroup, all occurrences of the `cn` attribute will be replaced by the value specified for the `-n` argument. This mode of operation applies to all command argument specified values, including `-u`, `-g`, `-s`, `-d`, `-I`, and `-c`.

---

*HP-UX 11i Version 2: December 2007 Update*
When the attr=value parameter is used to modify an existing attribute, the ldapugmod command will also use the LDAP replace operation. The replace operation will remove all occurrences of the specified attribute for an entry and replace it with the value specified. If there are multiple values for a single attribute in an entry, the use of a single attr=value parameter will replace all values with the single value specified on the command line.

Note that it is possible to specify more than one occurrence of the same attribute on the command line, if that attribute is multi-valued. In which case, both values will be created in the entry.

Use of -A or -R changes this behavior (for both the above-listed command arguments and the attr=value parameters).

Any attribute specified as an argument to the -A or -R option will cause ldapugmod to perform an LDAP add operation instead of an LDAP replace operation. Example: Suppose an entry in an LDAP directory appears as follows:

dn: uid=mwolters,ou=people,ou=IT,dc=FutureWidget,dc=com
cn: Manuel Wolters
uid: mwolters
uidNumber: 2284
gidNumber: 212
homeDirectory: /home/mwolters
loginShell: /usr/bin/ksh
gecos: Manuel Wolters,San Francisco, +1 505-555-5072

Performing the following ldapugmod:

```
ldapugmod -t passwd mwolters "cn=M. Wolters"
```

replaces all instances of cn:

```
dn: uid=mwolters,ou=people,ou=IT,dc=FutureWidget,dc=com
cn: M. Wolters
uid: mwolters
uidNumber: 2284
gidNumber: 212
homeDirectory: /home/mwolters
loginShell: /usr/bin/ksh
gecos: Manuel Wolters,San Francisco, +1 505-555-5072
```

Assuming the entry as originally specified, if the following ldapugmod command is issued:

```
ldapugmod -t passwd -R "cn=Manny Wolters" mwolters "cn=M Wolters"
```

The resulting entry would be:

```
dn: uid=mwolters,ou=people,ou=IT,dc=FutureWidget,dc=com
cn: Manuel Wolters
uid: mwolters
uidNumber: 2284
gidNumber: 212
homeDirectory: /home/mwolters
loginShell: /usr/bin/ksh
gecos: Manuel Wolters, San Francisco, +1 505-555-5072
```

As a general rule, be cautious before using ldapugmod to change multi-valued attributes.

Also note, use of the same attribute and value pair more than once, either specified as part of attr=value, -R or -A, or from other command line options (for example -I gecos where gecos is mapped to some other attributes) is not allowed. ldapugmod will exit with error status before send any conflicting modification request to the directory server.

LIMITATIONS
Since LDAP directories require data be stored according to the UTF-8 (RFC3629) character encoding method, all characters displayed by ldapugmod will be UTF-8, and assumed to be part of the ISO-10646 character set. ldapugmod will not perform conversion of the locale character set to/from the UTF-8 character set.
SEE ALSO
ldapcfinfo(1M), ldapugadd(1M), ldapugdel(1M), ldapuglist(1M), ldapux(5).

ldapugmod(1M)
NAME
libcadmin - libc administration command

SYNOPSIS
/usr/sbin/libcadmin

DESCRIPTION
The libcadmin command is used to perform administrative functions for libc. Currently the only function that the command performs is to replace the shared 32-bit PA-RISC2.0 library in /usr/lib/libc.2 with the shared 32-bit PA-RISC1.1 version of that library.

There are no required arguments or options used with this command. The /usr/lib/libc.a archive library and the libc.sl symbolic link are unaffected by the libcadmin command.

During the system installation process, the shared 32-bit PA-RISC1.1 library is copied into /usr/lib/pa11_32. This library is the same as the PA-RISC2.0 library except that it was built with different compiler options. The PA-RISC2.0 library provides much higher performance than the PA-RISC1.1 library. However, if the system administrator desires to have the PA-RISC1.1 library installed instead, libcadmin allows the system administrator to accomplish this task. When libcadmin is executed, the PA-RISC2.0 library is stored in /usr/lib/pa20_32 and the PA-RISC1.1 library is copied into /usr/lib/libc.2.

It is recommended that the system be re-booted after executing the libcadmin command so that applications which are linked shared against libc will use the shared 32-bit PA-RISC1.1 version of that library.

Once the command is executed, the only way to put the PA-RISC2.0 library back in place is to re-install the operating system.

The libcadmin command can only be executed by a user with superuser privileges.

ERRORS
Multiple executions of the command will result in an error message indicating the PA-RISC1.1 library is already installed.

Executing this command on a PA-RISC1.1 machine will result in an error message indicating the PA-RISC1.1 library is already installed.

RETURN VALUES
Upon successful completion, libcadmin returns zero (0). Otherwise a one (1) is returned.

AUTHOR
libcadmin was developed by HP.
NAME
link, unlink - execute link() and unlink() system calls without error checking

SYNOPSIS
/usr/sbin/link file1 file2
/usr/sbin/unlink file

DESCRIPTION
The link and unlink commands perform their respective system calls (link() or unlink()) on
their arguments, abandoning most error checking.

These commands can be executed only by users who have appropriate privileges.

EXTERNAL INFLUENCES
Environment Variables
LC_MESSAGES determines the language in which messages are displayed.
If LC_MESSAGES is not specified in the environment or is set to the empty string, the value of LANG is
used as a default for each unspecified or empty variable. If LANG is not specified or is set to the empty
string, a default of "C" (see lang(5)) is used instead of LANG.

If any internationalization variable contains an invalid setting, link behaves as if all internationalization
variables are set to "C". See environ(5).

International Code Set Support
Single- and multi-byte character code sets are supported.

RETURN VALUE
link and unlink return the following values:
0 Operation successful.
1 Input syntax error.
2 The link() or unlink() call failed.

WARNINGS
If a directory that contains files other than . and .. is unlinked, the files become orphans, unless they
are also linked by some other directory.

Not all file systems permit linking to directories.

SEE ALSO
ln(1), rm(1), link(2), unlink(2).

STANDARDS CONFORMANCE
link: SVID2, SVID3
unlink: SVID2, SVID3
NAME
linkloop - verify LAN connectivity with link-level loopback

SYNOPSIS
linkloop [-i PPA] [-n count] [-r rif] [-s size] [-t timeout] [-v] linkaddr ...

DESCRIPTION
The linkloop command uses IEEE 802.2 link-level test frames to check connectivity within a local area network (LAN).

linkadd is the hardware station address of a remote node. Several addresses can be specified at one time.

linkloop tests the connectivity of the local node and the remote node specified by each hardware station address. The hardware station address of a remote node can be found by executing lanscan on the remote node. This hardware station address is usually represented as a hexadecimal string prefixed with 0x. It can also be represented as a octal string prefixed with 0 or as a decimal string. The hardware station address must not be a multicast or broadcast address.

Options
linkloop recognizes the following options:

- **-i PPA** Specify the PPA to use. If this option is omitted, linkloop uses the first PPA it encounters in an internal data structure.

- **-n count** Set the number of frames to transmit. If count is 0, linkloop transfers frames indefinitely until an interrupt signal (defined by the user shell) is received. The default value for count is 1.

- **-r rif** Specify the particular bridge route over which token ring packets should be delivered. rif is the routing information field used for token-ring networks. Its value is given as an even number of hexadecimal bytes separated by colons, up to a maximum of 16 bytes.

- **-s size** Set the size in bytes of the data message to send. The maximum data size is dependent on the type of LAN link being used. The default value is the maximum data byte count that can be used for the particular link.

- **-t timeout** Set the amount of time in seconds to wait for a reply from the remote node before aborting. If timeout is 0, linkloop waits indefinitely for a reply. The default value is 2 seconds.

- **-v** Set the verbose option. In addition to the regular summary of test results, this option displays more extensive error information. If there are header or length errors, appropriate messages are displayed. All verbose output is preceded by the number of replies accepted before an error occurred.

Connectivity Test Results
linkloop aborts upon receipt of an interrupt signal. If aborted, the current results are printed.

linkloop prints the result of the link-level connectivity test. If the test fails, it prints a summary of the test and indicates the type of error. The possible messages are:

**address has bad format**
An incorrect hardware station address was entered on the command line.

**address is not individual**
The station address entered on the command line is either a multicast or broadcast address.

**frames sent**
Total number of frames sent.

**frames received correctly**
Total number of frames received without errors.

**frames with length error**
Received frame length does not match transmitted frame length. If the verbose option is set, the length received is printed.
frames with data error
   Received frame does not match transmitted frame.

frames with header error
   Number of frames received containing unexpected frame header information. Either the source
   address does not match the remote address, the destination address does not match the local address,
   or the control field is not the TEST frame control field. These frames are ignored. linkloop con-
   tinues to try to receive the reply frame until the read operation times out.

reads that timed out
   Count of how many read operations timed out before the reply was received.

DIAGNOSTICS
illegal count parameter
   The count specified in the -n option is a negative integer, or the number specified is too large for the
   local computer.

illegal timeout parameter
   The timeout specified in the -t option is a negative integer, or the value specified multiplied by 1000
   is too large for the local computer.

illegal size parameter
   The size specified in the -s option is not in the range from 0 to the maximum link data size. Remem-
   ber that the maximum link data size can vary in value for different LAN connection types. The cur-
   rent MTU can be obtained with the linkloop command.

No valid interface associated with PPA
   The PPA specified in the -i option is not a valid PPA.

Unable to open device file /dev/dlpi
   Device file /dev/dlpi does not exist.

invalid rif parameter
   The rif value in the -r option is invalid.

rif parameter too long
   The number of bytes in rif in the -r option exceeded 16, which is the maximum allowed.

rif parameter length must be even
   The number of bytes in rif in the -r option is odd. The number of bytes must be even.

AUTHOR
   linkloop was developed by HP.

SEE ALSO
   lanadmin(1M), lanscan(1M), lan(7).
localedef(1M)

NAME
localedef - generate a locale environment

SYNOPSIS
localedef [-cenvw] [-C compiler_options] [-L loader_options]
[-m method_file] [-f charmap_file] [-i locale_definition] locale_name

DESCRIPTION
localedef sets up the language environment for the named locale. localedef reads a
locale definition file (see localedef(4) for a detailed description) from standard input (default) or from
locale_definition file, creates a locale file with the same name as specified for the locale_name parameter,
and optionally installs this locale in the appropriate directory. Installation of public locales (those accessible
to all users) requires appropriate privileges. Creation of locales (both private and public) requires access to
the ANSI C compiler.

Options
localedef recognizes the following options:
-c Create permanent output even if warning messages have been generated.
-e Generate 64-bit locale in addition to the 32-bit locale. This is the default on a 64-bit operat-
ing system (and hence on Itanium(R)-based systems) and is included to allow cross platform
development.
-n (noinstall) Create the locale file in the current directory.
-v (verbose) Generate as many diagnostic messages as possible.
-w Generate additional warning messages for duplicate definitions and ellipses use in the
LC_COLLATE category.
-f charmap_file
If locale definition file contains symbolic names (of the form <name>) use charmap_file.
See charmap(4) for a description of the format of a charmap_file.
-i locale_definition
Use locale_definition file as input, instead of standard input (default).
-m method_file
Use the specified method_file to overwrite use of default methods in processing the locale
definition.
-C compiler_options
Specify additional compiler options to be applied in compiling the locale. See cc_bundled(1)
for a complete list of options. Use with care on a 64-bit operating system since the additional
default option includes +DA2.0W on PA-RISC and +DD64 on Itanium-based systems.
-L loader_options
Specify additional loader options to be applied in linking the locale. See ld(1) for a com-
plete list of options.
locale_name This argument is required, and identifies the name of the language following the naming
convention of the LANG environment variable (see environ(5)):
language [_territory] [.codeset ]

The following is a brief description of the components that make up a locale. For a complete description
of the form and syntax of a locale_definition file, see localedef(4). For a complete description of the form and
effects of a charmap file, see charmap(4).

Six categories of data in the locale_name file are recognized by setlocale(3C), and make up a language
definition:

LC_COLLATE Information in this category affects behavior of regular-expressions and NLS
string-collation functions.

LC_CTYPE Information in this category affects behavior of character classification and
conversion functions.
localedef(1M)

localedef(1M)

LC_MONETARY Information in this category affects behavior of functions that handle monetary values.

LC_NUMERIC Information in this category affects handling of the radix character in formatted-input/output and string-conversion functions.

LC_TIME Information in this category affects behavior of time-conversion functions.

LC_MESSAGES This category contains information affecting interpretation of yes/no responses.

A locale definition file also consists of six categories. The beginning of each category is identified by a category tag having the form LC_category where category is one of the following: CTYPE, COLLATE, MONETARY, NUMERIC, TIME, or MESSAGES. The end of each category is identified by a tag consisting of the word END followed by a space and the category identifier; for example, END LC_COLLATE. Categories can appear in any order in the locale definition file. At least one category specifications is required. If a category is not specified, setlocale() sets up the default “C” locale for that category (see setlocale(3C) and lang(5)).

Each category is composed of one or more statements. Each statement begins with a keyword followed by one or more expressions. An expression is a set of well-formed metacharacters, strings, and constants. localedef also recognizes comments and separators.

More than one definition specified for each category constitutes a hard error (causes localedef to exit without generating a locale). Any category can be specified by the keyword copy followed by the name of a valid locale. This causes the information for the category to be identical to that in the named locale. Note that the copy keyword, if used for a category, must be the first and only keyword following the category tag.

A methods file is used to create locales for user-specific character encoding schemes.

Operating System Requirements

For cross platform development and development on a 64-bit operating system several requirements must be observed. Both the 32-bit and 64-bit method libraries must exist. In the case of the 64-bit shared library it must be in the directory ../hpux64 (or pa20_64 in case of PA-RISC systems) under the location where the 32-bit library is located. When the -e option is specified, or when executing on a 64-bit operating system, the resulting locale is placed in the directory hpux64 (pa20_64 in case of PA-RISC systems) under the current working directory unless the install option has been specified.

Note

A locale built for one system cannot be used on other systems.

Users will not be able to generate PA-RISC locales on Itanium-based systems.

EXTERNAL INFLUENCES

Environment Variables

LANG determines the locale to use when neither LC_ALL or the other category variables specify a locale.

LC_ALL determines locale to be used. It overrides any values specified by LANG or any other LC_* variables.

LC_COLLATE and LC_CTYPE have no effect on the processing of localedef, which behaves as if these two variables were set to the C locale.

LC_MESSAGES determines the language in which messages are displayed.

International Code Set Support

Single- and multi-byte character code sets are supported.

RETURN VALUE

localedef returns the following values:

0 No errors occurred and the locale was successfully created.
1 Warnings occurred and the locale was successfully created.
2 The locale specification exceeded implementation limits or the coded character set used is not supported.
>3 Warnings or errors occurred, and no output was generated.
AUTHOR

localedef was developed by OSF and HP.

FILES FOR PA-RISC SYSTEMS

/usr/lib/nls/loc/src
/usr/lib/nls/loc/charmaps
/usr/lib/nls/loc/methods
/usr/lib/nls/loc/pa20_64/methods
/usr/lib/nls/loc/locales/language_territory.codeset
/usr/lib/nls/loc/pa20_64/locales/language_territory.codeset

FILES FOR ITANIUM-BASED SYSTEMS

In addition to the PA-RISC files, the following files are present on Itanium-based systems.

/usr/lib/nls/loc/hpux32/src
/usr/lib/nls/loc/hpux64/src
/usr/lib/nls/loc/hpux32/charmaps
/usr/lib/nls/loc/hpux64/charmaps
/usr/lib/nls/loc/hpux32/methods
/usr/lib/nls/loc/hpux64/methods
/usr/lib/nls/loc/hpux32/locales/language_territory.codeset
/usr/lib/nls/loc/hpux64/locales/language_territory.codeset

SEE ALSO
locale(1), setlocale(3C), charmap(4), localedef(4), environ(5).

STANDARDS CONFORMANCE
localedef: XPG4, POSIX.2
NAME
lockd, rpc.lockd - network lock daemon

SYNOPSIS
/usr/sbin/rpc.lockd [-l log_file] [-t timeout] [-g graceperiod]

DESCRIPTION
lockd is an RPC server that processes NFS file locking requests from the local kernel or from another remote lock daemon. lockd forwards lock requests for remote data to the server site's lock daemon through the RPC/XDR package (see rpc(3N)). lockd then requests the status monitor daemon, statd for monitor service (see statd(1M)). The reply to the lock request is not sent to the kernel until the status daemon and the server site's lock daemon have replied.

If either the status monitor or server site's lock daemon is unavailable, the reply to a lock request for remote data is delayed until all daemons become available.

When a server recovers, it waits for a grace period for all NFS client-site lockds to submit reclaim requests. Client-site lockds are notified by the statd of the server recovery, and promptly resubmit previously granted lock requests. If a lockd fails to secure a previously granted lock at the server site, the lockd sends a SIGLOST to the process holding that lock.

A fixed port can be specified for lockd by using the LOCKD_PORT variable in the /etc/rc.config.d/nfsconf file, as shown:

```bash
LOCKD_PORT = port_number
```

The port number can have any value between 1 and 65535. After adding the port variable to the /etc/rc.config.d/nfsconf file, lockd must be restarted for this to take effect. This feature can be disabled by deleting or commenting out the port variable from the /etc/rc.config.d/nfsconf file and restarting lockd.

Options
lockd recognizes the following options and command-line arguments:

- `-l log_file`
  Log any errors to the named log file log_file. Errors are not logged if the `-l` option is not specified.

  Information logged to the file includes date and time of the error, host name, process ID and name of the function generating the error, and the error message.

- `-t timeout`
  lockd uses timeout (seconds) as the interval instead of the default value (10 seconds) to retransmit a lock request to the remote server. Note that changing this value also changes the value for grace period duration.

- `-g graceperiod`
  lockd uses \((1 + (\text{graceperiod} / \text{timeout})) \times \text{timeout}\) (seconds) as the grace period duration instead of the default value (5 \times \text{timeout} seconds). If both `-t` and `-g` are specified, the `-t` should appear first since the grace period duration is dependent on the value of timeout.

WARNINGS
The LOCKD_PORT variable will be obsoleted in HP-UX 11i V3, and lockd will use 4045 as a hard coded fixed port. For this reason, HP recommends using 4045 for LOCKD_PORT.

AUTHOR
lockd was developed by Sun Microsystems, Inc., and HP.

SEE ALSO
statd(1M), fcntl(2), lockf(2), signal(2).
NAME
logins - display system and user login data

SYNOPSIS
logins [-admopstux] [-g groups] [-l logins]

DESCRIPTION
logins displays data concerning system and user logins. The format and content of the output is controlled by command options and may include: system or user login, user ID number, /etc/passwd comment field value (e.g., user name, etc...), primary group name, primary group ID, supplementary group names, supplementary group IDs, home directory, login shell, user security level, user audit events, and password aging parameters. The default data is: login, user ID, primary group name, primary group ID, and /etc/passwd comment field value. Output is sort by user ID, with user logins following system logins. The default output consists of login, user ID, primary group, primary group ID and comment field formatted into columns.

The following options are available to this command:
-a Displays two account expiration fields. The fields show how long the account can be unused (in days) before it becomes inactive and the date the account will expire.
-d Display logins with duplicate UIDs.
-m Show multiple group membership data.
-o Display with alternate format of one line of colon separated fields.
-p Display logins with no passwords
-s Display all system logins
-t Sort output by login rather than UID.
-u Display all user logins.
-x Display extended information about selected users. This extended information includes home directory, login shell and password aging data, each on its own line. Password information consists of password status (PS for valid password, LK for locked and NP for no password) and, if a password is present, date of last change, required number of days between changes, and number of days allowed between changes. In the case of non-trusted systems, the date of last change will be the latest Thursday since the change.

-g groups
Display all users belonging to groups, sorted by login. A comma separated list specifies multiple groups.

-l logins
Display the requested logins. A comma separated list specifies multiple logins.

Multiple options may be used. Any login matching any of the criteria will be displayed. A login will be displayed only once, even if it meets multiple criteria.

EXAMPLES
logins List all logins in default format.
logins -p -d List all logins that have no password or have a duplicate UID in default format.
logins -s -o List all system logins in the alternate format.

FILES
/etc/passwd HP-UX password file.
/etc/group HP-UX group file.

SEE ALSO
listusers(1), passwd(1), group(4), passwd(4), prpwd(4), shadow(4).

STANDARDS CONFORMANCE
logins: SVID3

506 Hewlett-Packard Company – 1 – HP-UX 11i Version 2: December 2007 Update
NAME
lpadmin - configure the LP spooling system

SYNOPSIS
/usr/sbin/lpadmin -p printer [options]
/usr/sbin/lpadmin -x dest
/usr/sbin/lpadmin -d [dest]
/usr/sbin/lpadmin -q [s | l | c]
/usr/sbin/lpadmin -s [enable | disable]

DESCRIPTION
lpadmin configures LP spooling systems to describe printers, classes and devices. It is used to add and remove destinations, change membership in classes, change devices for printers, change printer interface programs, and to change the system default destination. lpadmin cannot be used when the LP scheduler, lpsched(1M), is running, except where noted below.

Exactly one of the -p, -x, -s, or -d options must be present for every legal invocation of lpadmin.

-<printer> Names a printer to which all of the options below refer. If printer does not exist, it will be created.
-<dest> Removes destination dest from the LP system. If dest is a printer and is the only member of a class, the class is deleted, too. No other options are allowed with -x.
-<dest> Makes existing destination dest the new system default destination. If dest is not supplied, there is no system default destination. This option can be used when lpsched(1M) is running. No other options are allowed with -d.
-<enable | disable> Enables or disables email notification. See the Enabling and Disabling Email Notification section below for more information on this option.

The following options are only useful with -p and can appear in any order. For ease of discussion, the printer is referred to below as printer P.
-<class> Inserts printer P into the specified class. class is created if it does not already exist.
-<printer> Copies an existing printer’s interface program to be the new interface program for printer P.
-<priority> Sets the default priority for printer P associated with lp(1). If omitted, the default priority is set to 0.
-<hardwired> Indicates that the device associated with printer P is hardwired. This option is assumed when creating a new printer unless the -l option is specified.
-<interface> Establishes a new interface program for printer P. interface is the pathname of the new program.
-<login terminal> Indicates that the device associated with printer P is a login terminal. The LP scheduler (see lpsched(1M)) disables all login terminals automatically each time it is started. Before re-enabling printer P, its current device should be established using lpadmin.
-<model> Selects a model interface program for printer P. model is one of the model interface names supplied with the LP software (see Models below).
-<class> Removes printer P from the specified class. If printer P is the last member of the class, the class is removed.
-<device> Associates a new device with printer P. device is the pathname of a file that is writable by the LP administrator lp. Note that there is nothing to stop an administrator from associating the same device with more than one printer. If only the -p and -v options are supplied, lpadmin can be used while the scheduler is running.
The following **options** are only useful with **-p** and can appear in any order. They are provided with systems that provide remote spooling.

- **-orm machine** The name of the remote machine is **machine**.
- **-orp printer** The name of the printer or printer class to use on the remote machine is **printer**.
- **-oci remcancel** Specifies that the local command **remcancel** is used to cancel requests to remote printers. To ensure that the correct command is used, specify the full path name.
- **-ocm remcancel** Specifies that the local model **remcancel** is used to cancel requests to remote printers.
- **-osi remstatus** Specifies that the command **remstatus** is used to obtain the status of requests to remote printers. To ensure that the correct command is used, specify the full path name.
- **-osm remstatus** Specifies that the model **remstatus** is used to obtain the status of requests to remote printers.

**Restrictions**

When creating a new printer, the **-v** option and one of the **-e**, **-i** or **-m** options must be specified. Only one of the **-e**, **-i** or **-m** options can be specified. The **-h** and **-l** key letters are mutually exclusive.

Printer and class names must not exceed 14 characters and must consist entirely of the characters A-Z, a-z, 0-9 and _ (underscore).

For a remote printer, only one of **-oci** or **-ocm** can be specified. If neither of them is specified, the default remote cancel model /usr/lib/lp/cmodel/rcmodel is used. Similarly, only one of **-osi** or **-osm** can be specified. If neither of them is specified, the default remote status model /usr/lib/lp/smodel/rsmodel is used.

All local printers use four-digit request numbers. All remote printers use three-digit request numbers for contact with BSD systems.

### Long Destination Name Support

You can increase the destination name length limitation of 14 characters to 250 characters by using **lpadmin** with the following set of options:

```
/usr/sbin/lpadmin -q[s|l|c]
```

Note that these options are specific to this release.

- **-q** Prints whether long destination name support is enabled or disabled.
- **-ql** Enables long destination name support. See also **WARNINGS**.
- **-qs** Disables long destination name support. This results in the removal of printers and classes with names greater than 14 characters. Even remote printers with a remote destination name (argument to **-orp**) greater than 14 characters are removed. Pending print requests to these destinations are also removed.
- **-qc** Lists all destinations that will be removed when long destination name support is disabled.

No print related activity should happen during the enabling or disabling of long destination name support.

### Enabling and Disabling Email Notification

Email notification is enabled by default. To modify or query the email notification status, use **lpadmin** with the following options:

```
/usr/sbin/lpadmin -s[enable|disable]
```

Following is a description of each of these options:

- **-s** Print the email notification status.
lpadmin(1M)

-senable  Enable email notification.
-sdisable  Disable email notification. This email notification setting is ignored if the print request is queued using the lp command and the -w or -m option. See lp(1).

Models

Model interface programs are supplied with the LP software. They are shell procedures, C programs, or other executable programs that interface between lpsched(1M) and devices. All printer models reside in directory /usr/lib/lp/model and can be used without modification with lpadmin -m. All cancel models reside in directory /usr/lib/lp/cmodel and can be used without modification with lpadmin -ocm. Models should have 644 permission if owned by lp and bin, or 664 permission if owned by bin and bin. Model file names must not exceed 14 characters. Alternatively, LP administrators can modify copies of models then use lpadmin -m to associate them with printers.

The LP model interface program does the actual printing on the device that is currently associated with the printer. The LP spooler sets standard input to /dev/null and standard output and standard error output to the device specified in the -v option of lpadmin. The interface program is then invoked for printer P from the directory /etc/lp as follows:

```
interface/P id user title copies options file ...
```

where arguments are as follows:

- **id**  request id returned by lp(1).
- **user**  login name of the user who made the request.
- **title**  optional title specified with the -t option of lp(1).
- **copies**  number of copies to be printed.
- **options**  blank-separated list of class-dependent or printer-dependent options specified with the -o option of lp(1). Options from a BSD system have the character sequence BSD attached to the beginning of the option (for example, BSD1).
- **file**  full pathname of the file to be printed.

Given the command line arguments and the output directed to the device, interface programs can format their output in any way they choose.

When printing is completed, it is the responsibility of the interface program to exit with a code indicative of the success of the print job. Only return values of 0 indicating that the job completed successfully, or values of positive 1 through 127 indicating that some error was encountered that does not affect future print jobs should be used. Negative values and positive values greater than 127 are reserved for system use and should not be used by interface programs. lpsched(1M) notifies users by mail when there is an error in printing the request. If problems are detected that are likely to affect future print jobs, the interface program should return a error value. This will make lpsched(1M) disable the printer so that other pending print requests are not lost.

The cancel and status model interface programs perform the actual communication with the remote system to cancel requests or get the status of requests. See rcancel(1M) and rlpsstat(1M) for command line arguments.

EXTERNAL INFLUENCES

Environment Variables

**LANG** determines the language in which messages are displayed.

If **LANG** is not specified or is set to the empty string, a default of "C" (see lang(5)) is used instead of **LANG**.

If any internationalization variable contains an invalid setting, lpadmin behaves as if all internationalization variables are set to "C" (see environ(5)).

EXAMPLES

Assuming an existing Hewlett-Packard laserjet printer named lp1, it will use the laserjet model interface through /dev/lp after the command:

```
/usr/sbin/lpadmin -plp1 -mlaserjet -v/dev/lp
```

Assuming a printer lp on a remote system system2, the command:
lpadmin(1M)

```
/usr/sbin/lpadmin -plp3 -v/dev/null -mrmodel -omrcmodel -osmrmodel -ormsystem2 -orlp
```
causes the spool system to use the local line printer `lp3' and the model `rmodel'. The spool system also uses the model `rcmodel' to cancel remote requests and `rsmodel' to get status from `system2'. In addition, the remote system name `system2' and the remote printer `lp' are used.

The following command performs the same operation as above.

```
/usr/sbin/lpadmin -plp3 -v/dev/null -mrmodel -ormsystem2 -orlp
```
Here the default remote cancel model `rcmodel' and the default remote status model `rsmodel' are used.

**WARNINGS**

When installing remote printers, use the option `-ocmrcmodel' instead of `-oci/usr/sbin/rcancel' to specify the method used to cancel remote requests. The option `-osmrmodel' should be used instead of `-osi/usr/sbin/rlpstat' to specify the method used for displaying remote status.

`classes' must not include `remote' printers. HP-UX systems do not have the ability to distribute print jobs in this way. Printing to a class of printers on a remote system (e.g., `systemB') must be accomplished by creating the class on the remote system, then identifying that class by using a command resembling the following (although you might have to change some of the specific values shown in the example):

```
lpadmin -plocal_name -ormsystemB -orpsystemB_class_name -v /dev/null -mrmodel -ocmrcmodel -osmrmodel
```

Some products dependent on the `lp' spooling system may not support printer and class names greater than 14 characters. The behavior of these products is unspecified when Long Destination Name support is enabled.

**FILES**

```
/var/spool/lp/*
/var/adm/lp/*
/etc/lp/*
/usr/lib/lp/*
```

**SEE ALSO**

`enable(1)', `lp(1)', `lpstat(1)', `nroff(1)', `accept(1M)', `lpname(1M)', `lpsched(1M)', `rcancel(1M), `rlp(1M), `rlpdaemon(1M), `rlpstat(1M).`
NAME
lpna - display LP spooler performance analysis information

SYNOPSIS
lpna [-d dest]

DESCRIPTION
lpna displays LP spooler performance information, which system administrators can use to optimize the
configuration of the entire spooler system.

Options
lpna recognizes one option:
-d dest Choose dest as the printer or the class of printers. If dest is a printer, the performance
analysis information is displayed for that specific printer. If dest is a class of
printers, the performance analysis information is displayed for the printers that are
members of the class. By default, lpna displays the performance analysis informa-
tion for all printers and/or classes.

lpna examines /var/adm/lp/lpana.log for the following items:
Wait AV Average waiting time from when job is spooled until start of printing.
Wait SD Standard Deviation for waiting time.
Print AV Average printing time from start to end of job.
Print SD Standard Deviation for printing time.
Bytes AV Average of number of bytes printed per request.
Bytes SD Standard Deviation for number of bytes.
Sum KB Sum of bytes printed for all requests (in kilobytes).
Num of Requests Total number of requests since logging started.

EXTERNAL INFLUENCES
Environment Variables
LANG determines the language in which messages are displayed.

WARNINGS
lpna performs its operation on the local system only.

AUTHOR
lpna was developed by HP.

FILES
/var/adm/lp/lpana.log

SEE ALSO
lp(1), lpstat(1), lpadmin(1M), lpsched(1M).
NAME
lpsched, lpshut, lpmove, lpfence - start the LP request scheduler, stop the LP request scheduler, move requests between LP destinations, and define the minimum priority for printing

SYNOPSIS
/usr/sbin/lpsched [-v] [-a]
/usr/sbin/lpshut
/usr/sbin/lpmove requests dest
/usr/sbin/lpmove dest1 dest2
/usr/sbin/lpfence printer fence

DESCRIPTION
lpsched Schedules requests taken by \texttt{lp(1)} for printing on printers. \texttt{lpsched(1M)} is typically invoked in \texttt{/sbin/rc}. This creates a process which runs in the background until \texttt{lpshut} is executed. The activity of the process is recorded in \texttt{/var/adm/lp/log}.
lpsched recognizes the following options:
-\texttt{-v} Write a verbose record of the lpsched process on \texttt{/var/adm/lp/log}.
-\texttt{-a} Write \texttt{lpama(1M)} logging data on \texttt{/var/adm/lp/lpana.log}.
lpshut Shuts down the printer scheduler. All printers that are printing at the time \texttt{lpshut} is invoked stop printing. Requests that were printing at the time the scheduler was shut down are reprinted in their entirety after lpsched is started again. All LP commands perform their functions even when \texttt{lpsched} is not running.
lpmove Moves requests that were queued by \texttt{lp(1)} between LP destinations. This command can be used only when \texttt{lpsched} is not running.
The first form of the command moves the named requests to the LP destination, \texttt{dest}. requests are request ids as returned by \texttt{lp(1)}. The second form moves all requests for destination \texttt{dest1} to destination \texttt{dest2}. As a side effect, \texttt{dest1} rejects requests.
Note that \texttt{lpmove} never checks the acceptance status (see accept(1M)) for the new destination when moving requests.
lpfence Defines the minimum required priority for the spooled file to be printed. \texttt{fence} must be in between 0 (lowest fence) and 7 (highest fence). Each printer has its own fence, which is initialized to 0 when it is configured by the \texttt{lpadmin(1M)} command. \texttt{lpfence} is used only when \texttt{lpsched} is not running.

EXTERNAL INFLUENCES
Environment Variables
\texttt{LC_TIME} determines the format and contents of date and time strings.
\texttt{LANG} determines the language in which messages are displayed.
If \texttt{LC_TIME} is not specified in the environment or is set to the empty string, the value of \texttt{LANG} is used as a default for each unspecified or empty variable. If \texttt{LANG} is not specified or is set to the empty string, a default of "C" (see lang(5)) is used instead of \texttt{LANG}. If any internationalization variable contains an invalid setting, the commands behave as if all internationalization variables are set to "C". See environ(5).

WARNINGS
lpsched, lpshut, lpmove, and lpfence perform their operation on the local system only.

FILES
/var/spool/lp/*
/var/adm/lp/*
/etc/lp/*
/usr/lib/lp/*

SEE ALSO
accept(1M), cancel(1), enable(1), \texttt{lp(1)}, \texttt{lpadmin(1M)}, \texttt{lpama(1M)}, \texttt{lpsstat(1)}, \texttt{rcancel(1M)}, \texttt{rlp(1M)}, \texttt{rlpdaemon(1M)}, \texttt{rlpstat(1M)}.
NAME
lsdev - list device drivers in the system

SYNOPSIS
/usr/sbin/lsdev [-h] [-d driver | -C class] [-b block_major] [-c char_major] [-e major]
[major ...]

DESCRIPTION
The lsdev command lists, one pair per line, the major device numbers and driver names of device drivers configured into the system and available for invocation via special files. A -1 in either the block or character column means that a major number does not exist for that type.

If no arguments are specified, lsdev lists all drivers configured into the system.

If the -h option is specified, lsdev will not print a heading. This option may be useful when the output of lsdev will be used by another program.

The -d, -C, -b, -c, and -e options are used to select specific device drivers for output. If more than one option is specified, all drivers that match the criteria specified by those options will be listed. These search options are divided into two types: name search keys (the -d and -C options) and major number search keys (the -b, -c, and -e options). If both types of options are present, only entries that match both types are printed. The same type of option may appear more than once on the command line with each occurrence providing an ORing effect of that search type. The -d and -C options may not be specified at the same time.

The ability to process major arguments is provided for compatibility and functions like the -e option.

Options
- C class List device drivers that match class.
- d driver List device drivers with the name driver.
- b block_major List device drivers with a block major number of block_major.
- c char_major List device drivers with a character major number of char_major.
- e major List device drivers with either a character major number or block major equal to major.

DIAGNOSTICS
Invalid combination of options
The -d and -C options may not be specified at the same time.

Invalid major number
A major number is malformed or out of range.

EXAMPLES
To output entries for all drivers in the pseudo class:
    lsdev -C pseudo
To output entries that are in the class disk that have either a block or character major number of 0:
    lsdev -C disk -e 0
To get the character major number of my_driver into a shell environment variable:
    C_MAJOR=${(lsdev -h -d my_driver | awk '{print $1}')}

WARNINGS
Some device drivers available from the system may be intended for use by other drivers. Attempting to use them directly from a special file may produce unexpected results.

A driver may be listed even when the hardware requiring the driver is not present. Attempts to access a driver without the corresponding hardware will fail.

lsdev only lists drivers that are configured into the currently executing kernel. For a complete list of available drivers, please run sam (see sam(1M)).
DEPENDENCIES
Since *lsdev* relies on the device driver information provided in a *driver_install* routine, *lsdev* may not list drivers installed by other means.

AUTHOR
*lsdev* was developed by HP.

SEE ALSO
*sam(1M).*
Section 7 entries related to specific device drivers.
NAME
  lssf - list a special file

SYNOPSIS
  /sbin/lssf special_file ...

DESCRIPTION
  lssf lists information about a special file. For each special_file name, lssf determines the major
  number of the special file and whether it is block or character (using stat(2)). It then scans the system
  for the device that is associated with the special file. When the device is found, the minor number of the
  special file is decoded. A mnemonic description of the minor number is printed on standard output along
  with the hardware path (i.e., address) of the device. Mnemonics used to describe the fields are closely related
  to the options used with mksf (see mksf(1M)).

DIAGNOSTICS
  Most diagnostic messages from lssf are self explanatory. Listed below are some messages deserving
  further clarification. Warnings allow lssf to continue.

Warnings
  No such device in the system
    There is no information about the device in the kernel. The special file is not usable. Use rmsf
    to remove the special file (see rmsf(1M)).

Character major <major> is not in the kernel
Block major <major> is not in the kernel
  The major number associated with the special file is not in the kernel. Use kcmodule to add
  the appropriate driver to the kernel (see kcmodule(1M)).

Device driver <name> is not in the kernel
Device class <name> is not in the kernel
  The indicated device driver or device class is not present in the kernel. An open() of a special
  file pointing to an unusable device fails. To make the device usable, the appropriate device
  driver and/or device class must be added to the kernel using kcmodule(1M). If the device is no
  longer needed, rmsf should be used to remove the special files and update /etc/ioconfig.

<special_file> is not a special file
  The file is not associated with an I/O device.

EXAMPLES
  Suppose a special file is created with the command mksf -d tape2 -H 8.6.1 -b 1600 -a
  rmt/c2t6d0m. The command lssf rmt/c2t6d0m then produces:

  tape2 instance 2 bpi 1600 att address 8.6.1 rmt/c2t6d0m

AUTHOR
  lssf was developed by HP.

FILES
  /dev/config I/O system special file
  /etc/ioconfig I/O system configuration database

SEE ALSO
  insf(1M), kcmodule(1M), mksf(1M), rmsf(1M).
NAME
lugadmin - long user and group name enablement and display

SYNOPSIS
lugadmin [-e] [-d num] [-l]

DESCRIPTION
The lugadmin command enables the support of long user and group names on the system, sets the default display width for long user and group names, and displays the current status of the system.

If no options are specified, the -l option is assumed.

Options
The following options are supported:

- e Enable long user and group name support on the system. See the Enabling Long User and Group Names subsection below for details. This option requires superuser privileges.

- d num Set num as the default display width for long user and group names. num can be in the range 0 to 31.

The -d option can be only be specified with the -e option or when long names have been previously enabled.

See the Setting the User and Group Name Display Width subsection below for details. This option requires superuser privileges.

- l Display the current status of the system. The command displays one of:

8 The system is restricted to short (8-byte) user and group names.

31 The system is enabled for long (31-byte) user and group name support. A name string can be up to 31 bytes plus a trailing NULL byte.

Enabling Long User and Group Names
To enable long user and group names,

1. Stop the pwgrd daemon.

   # /sbin/init.d/pwgr stop

2. Run the lugadmin command with the -e option.

   # lugadmin -e [-d num]

3. The command requests confirmation before enabling the system.

   Long user/group name once enabled cannot be disabled in future. Do you want to continue [yY]:

   If you respond with y or Y, the command proceeds; otherwise, it aborts.

4. The command creates the /etc/default/lugname file.

5. If the -d option is not specified, num defaults to 0. The specified or default value of num is written in lugname.

Setting the User and Group Name Display Width
The long user and group name default display width is set when lugadmin is executed with the -e option. It can be changed at any time by executing lugadmin with the -d option.

If -d is not specified with the -e option, the value of num defaults to 0.

The current display width is used by all system and conforming applications to display formatted output, as follows:

0 Display user and group names as you would for a short name system. If the name is longer than the command's default width, replace the last printable character of the name with a plus sign (+).

1 to 31 If the name is longer (in bytes) than the current display width, replace the last printable character of the name with a plus sign (+).
The above display format may change in future releases.

**EXTERNAL INFLUENCES**

*Environment Variables*

**UG_DISPLAY_WIDTH**

If it is set and long names are enabled, it overrides the width specified in the `lugname` file for all system and conforming applications that write formatted output.

**RETURN VALUE**

`lugadmin` exits with one of the following values:

- **0** Successful completion
- **-1** Failure

**DIAGNOSTICS**

Note: System is already enabled for long user/group name

You may have attempted to use the `-e` option for a second time.

Error: The system is not enabled for long user/group name

You tried to use the `-d` option before using the `-e` option.

Error: Can’t create file `/etc/defaults/lugname`

Error: Can’t open file `/etc/defaults/lugname`

Error: The pwgrd daemon must be killed before enabling the system. The daemon can be restarted once the system is enabled

`system()` failed

A call to the `system()` function failed.

Error: Long user/group name feature is not supported on trusted systems

**WARNINGS**

The `lugname` file is the switch that marks a long name system. If it is missing, short names are assumed and the **UG_DISPLAY_WIDTH** variable is ignored.

While many programs are unaffected by long names in a short name environment, some may behave abnormally and some may abort.

It is recommended that you do not attempt to restore the short name environment once long names have been enabled.

Do not use the system constant `LOGIN_NAME_MAX` with the `getconf` command or the `sysconf()` function to test for name length; it is not changed when long names are enabled.

The value of the current status of the system (`-l` option) may change in future releases.

Once a system is enabled for long user and group names, it cannot be made a trusted system.

**AUTHOR**

`lugadmin` was developed by HP.

**FILES**

`/etc/default/lugname` If this file is present, the system is enabled for long user and group names; otherwise, it is not. This file contains the default display width, as set with the `-d` option.

**SEE ALSO**

`getconf(1)`, `sysconf(2)`.

HP-UX 11i Version 2: December 2007 Update — 2 — Hewlett-Packard Company 517
NAME
lvchange - change LVM logical volume characteristics

SYNOPSIS
                [-t IO_timeout] lv_path

Remarks
Mirrored disk operations require the installation of the optional HP MirrorDisk/UX software, which is not
included in the standard HP-UX operating system.

lvchange cannot be performed if the volume group is activated in shared mode.

DESCRIPTION
The lvchange command changes certain characteristics of a logical volume. Other characteristics can be
changed with the lvextend and lvreduce commands (see lvextend(1M) and lvreduce(1M)).

The command-line options specify the type and extent of change. Each current characteristic for a logical
volume remains in effect until explicitly changed by the corresponding option. All options take effect
immediately, except -s, which takes effect only when new extents are allocated by the lvextend com-
mand.

If a logical volume is striped, its scheduling policy is always parallel and its allocation policy is always strict
and noncontiguous; these attributes cannot be changed with lvchange.

The lvchange command can also be used to change the timeout value for a logical volume. This can be
useful to control how long an IO request will be retried (for a transient error, like a device timeout), before
giving up and declaring a pending IO to be failed. The default behavior is for the system to continue to
retry an IO for a transient error until the IO can complete. Thus, the IO will not be returned to the caller
until the IO can complete. By setting a non-zero IO timeout value, this will set the maximum length of time
that the system will retry an IO. If the IO cannot complete before the length of time specified by the IO
timeout, then the IO will be returned to the caller with an error. The actual duration of the IO request may
exceed the logical volume's maximum IO timeout value when the underlying physical volume(s) have
timeouts which either exceed the logical volume's timeout value or are not an integer multiple of the logical
volume's timeout value (see pvchange(1M) for details on how to change the IO timeout value on a physical
volume).

Options and Arguments
The -c, -d, -M, and -s options are meaningful only if the optional HP MirrorDisk/UX software has been
installed on the system.

lvchange recognizes the following options and arguments:

lv_path

The block device path name of a logical volume.

-a availability

Set logical volume availability. availability can have one of the following values:

  y Make a logical volume available. An open of the logical volume will
      succeed.

  n Make a logical volume temporarily unavailable. An open of the logical
      volume will fail. However, all current processes that have the logical
      volume open remain open.

-A autobackup

Set automatic backup for this invocation of this command. autobackup can
have one of the following values:

  y Automatically back up configuration changes made to the logical
      volume. This is the default.

  n Do not back up configuration changes this time.

After this command executes, the vgcfgbackup command (see
vgcfgbackup(1M)) is executed for the volume group to which the logi-

518  Hewlett-Packard Company

-1 -  HP-UX 11i Version 2: December 2007 Update
-c mirror_consistency

Set mirror consistency recovery. This option is effective only when -M n is specified or previously set. mirror_consistency can have one of the following values:

- y Set mirror consistency recovery on. LVM achieves mirror consistency during volume group activation by going through all logical extents and copying data from a nonstale copy to the other mirror copies.
- n Set mirror consistency recovery off. LVM does not perform mirror consistency recovery on this logical volume when the volume group is activated following a system crash. This setting should only be used on logical volumes that do not require mirror consistency recovery or where mirror consistency recovery is performed by another subsystem; for example, swap. See the WARNINGS section for more details.

-C contiguous

Set the contiguous allocation policy. contiguous can have one of the following values:

- y Set a contiguous allocation policy. Physical extents are allocated in ascending order without any gap between adjacent extents and all extents are contained in a single physical volume.
- n Do not set a contiguous allocation policy.

A nonempty logical volume that has a noncontiguous allocation policy cannot be changed to a contiguous allocation policy unless it happens to meet all the requirements of the contiguous allocation policy. See lvcreate(1M) for more information about the contiguous allocation policy.

-d schedule

Set the scheduling policy when a logical extent with more than one mirror is written. (The scheduling policy of a striped logical volume is striped and cannot be changed.) schedule can have one of the following values:

- p Establish a parallel scheduling policy.
- s Establish a sequential scheduling policy. Use this value with care, because it leads to performance loss in most cases.

-D distributed

Change the distributed allocation policy. distributed can have one of the following values:

- y Turn on distributed allocation.
- n Turn off distributed allocation.
- f Force distributed allocation to be on.

When the distributed allocation policy is turned on, only one free extent is allocated from the first available physical volume. The next free extent is allocated from the next available physical volume. Allocation of free extents proceeds in round-robin order on the list of available physical volumes.

When the distributed allocation policy is turned off, all available free extents are allocated from each available physical volume before proceeding to the next available physical volume.

The distributed allocation policy REQUIRES the PVG-strict allocation policy ( -s g ) to ensure that mirrors of distributed extents do not overlap (for maximum availability).

The distributed allocation policy is incompatible with the striped scheduling policy ( -i stripes ) and the contiguous allocation policy ( -C y ).

See lvcreate(1M) for more information on the distributed allocation policy.

The -D y option will fail if the existing logical volume has any two consecutive logical extents on the same physical volume. To override this failure, use the -D f option.

If a logical volume with the distributed allocation policy has at least two consecutive logical extents on the same physical volume, then...
lvchange(1M) lvchange(1M)

lvdisplay(1M) will display the allocation as **partially-distributed** (vs. **distributed**).

See lvdisplay(1M) for display values.

-**M** mirror_write_cache  
Set the Mirror Write Cache flag. This option is allowed only when the logical volume is not opened. **mirror_write_cache** can have one of the following values:

  y  Set Mirror Write Cache on. Every write to a mirror copy is recorded in the Mirror Write Cache and written into the Mirror Consistency Record on the disk if a cache-miss occurs. This allows LVM to determine whether all mirror copies are identical, even across system crashes. When the volume group is activated, the Mirror Consistency Record is used to perform mirror consistency recovery.

  n  Set Mirror Write Cache off. Mirror write does not incur an additional write to the Mirror Consistency Record on the disk.

-**p** permission  
Set the access permission. **permission** can have one of the following values:

  w  Set the access permission to read-write.

  r  Set the access permission to read-only.

-**r** relocate  
Set the logical volume bad block relocation policy. This is an obsolete flag available only to provide compatibility with prior HP-UX releases. The **relocate** flag can have one of the following values:

  y/n/N  
This release does not provide the LVM bad block relocation feature, however for compatibility reasons the value is maintained as a logical volume attribute. Displaying the logical volume attributes will show the value of the flag selected. However, regardless of the selection, no new relocations will be done.

If the volume group is activated on a different HP-UX release that provides the bad block relocation feature, bad blocks may be relocated depending upon the value of this flag. Although no new relocations will be done, any bad block relocations present on logical volume that was activated on HP-UX releases that provided this feature, will be honored when the volume group is activated on this HP-UX release.

  y  is the default value of this flag.

-**s** strict  
Set the strict allocation policy. Mirror copies of a logical extent can be allocated to share or not share the same physical volume or physical volume group. This option only makes sense when the physical volumes of the volume group that owns the specified logical volume reside on different physical disks. **strict** can have one of the following values:

  y  Set a strict allocation policy. Mirrors of a logical extent cannot share the same physical volume.

  g  Set a PVG-strict allocation policy. Mirrors of a logical extent cannot share the same physical volume group.

  n  Do not set a strict or a PVG-strict allocation policy. Mirrors of a logical extent can share the same physical volume.

When a logical volume is mirrored, the following changes are not allowed:

- From nonstrict to strict
- From nonstrict to PVG-strict
- From strict to PVG-strict

-**t** IO_timeout  
Set the **IO_timeout** for the logical volume to the number of seconds indicated. This value will be used to determine how long to wait for IO requests to complete before concluding that an IO request cannot be completed. An **IO_timeout** value of zero (0) causes the system to use the default value of "forever". **NOTE:** The actual duration of the request may exceed the
lvchange(1M)

specified \textit{IO\_timeout} value when the underlying physical volume(s) have
 timeouts which either exceed this \textit{IO\_timeout} value or are not integer mul-
tiples of this value.

\textbf{EXTERNAL INFLUENCES}

\textbf{Environment Variables}

\texttt{LANG} determines the language in which messages are displayed.

If \texttt{LANG} is not specified or is null, it defaults to "C" (see \textit{lang}(5)).

If any internationalization variable contains an invalid setting, all internationalization variables default to
"C" (see \textit{environ}(5)).

\textbf{EXAMPLES}

Change the permission of a logical volume to read-only:

\texttt{lvchange -p r /dev/vg01/lvol3}

Change the allocation policy of a logical volume to nonstrict:

\texttt{lvchange -s n /dev/vg01/lvol7}

Turn the mirror write cache off on a logical volume:

\texttt{lvchange -M n /dev/vg01/lvol1}

Change the IO timeout value of a logical volume to 1 minute (60 seconds):

\texttt{lvchange -t 60 /dev/vg01/lvol1}

\textbf{WARNINGS}

For root, swap or dump logical volumes, the allocation policy is always contiguous. This attribute cannot be
changed with \texttt{lvchange}.

By setting mirror consistency recovery off, crash recovery time will be reduced. After a system crash the
mirrored logical volume will be available, but there may not be consistent data across each mirror copy.
The only types of data that can safely be put on a mirrored logical volume with mirror consistency recovery
turned off are:

\begin{itemize}
  \item data not needed after a crash, such as swap or other raw scratch data, or
  \item data that an application itself will automatically reconstruct; for example, a raw logical volume for
        which a database keeps a log of incomplete transactions.
\end{itemize}

\textbf{SEE ALSO}

\texttt{lvcreate(1M)}, \texttt{lvdisplay(1M)}, \texttt{lvextend(1M)}.
NAME
lvcreate - create logical volume in LVM volume group

SYNOPSIS
[-D distributed] [-i stripes [-I stripe_size]] [-l lv_name] [-L lv_size] [-m mirror_copies]
[-M mirror_write_cache] [-n lv_name] [-p permission] [-r relocate] [-s strict] vg_name

Remarks
Mirrored disk operations require the installation of the optional HP MirrorDisk/UX software, which is not included in the standard HP-UX operating system.

lvcreate cannot be performed if the volume group is activated in shared mode.

DESCRIPTION
The lvcreate command creates a new logical volume within the volume group specified by vg_name. Up to 255 logical volumes can be created in one volume group.

If you specify the -n lv_name option, a new logical volume is created with that name. Otherwise, a system-generated name of the form lvolN is created, where N is the decimal equivalent of the two least significant bytes of the minor number of the new logical volume, in the range 1 to 255 (see lvm(7)). Two device files are created in vg_name: a block device file named lv_name or lvolN, and a character (raw) device file named rlv_name or rlvolN.

If you omit the -l and -L options, the logical volume is created with zero length. This permits you to choose its physical volume location when you allocate logical extents with the lvextend command (see lvextend(1M)). If you specify -l or -L, the location is determined automatically.

The default settings provide the most commonly used characteristics. Use the options to tailor the logical volume to the requirements of the system. Once a logical volume is created, some of its characteristics can be changed with the lvchange, lvextend, and lvreduce commands (see lvchange(1M), lvextend(1M), and lvreduce(1M)).

Options and Arguments
The -c, -d, -m, -M, and -s options are only meaningful if the optional HP MirrorDisk/UX software has been installed on the system.

lvcreate recognizes the following options and arguments:

-vg_name
  The path name of a volume group.

-A autobackup
  Set automatic backup for this invocation of this command. autobackup can have one of the following values:
  y Automatically back up configuration changes made to the logical volume. This is the default.
  After this command executes, the vgcfgbackup command (see vgcfgbackup(1M)) is executed for the volume group to which the logical volume belongs.
  n Do not back up configuration changes this time.

-c mirror_consistency
  Set mirror consistency recovery. This option is effective only when -M n is specified. It is ignored for -M y. mirror_consistency can have one of the following values:
  y Set mirror consistency recovery on. This is the default.
  LVM achieves mirror consistency during volume group activation by going through all logical extents and copying data from a nonstale copy to the other mirror copies.
  n Set mirror consistency recovery off. LVM does not perform mirror consistency recovery on this logical volume when the volume group is activated following a system crash. This setting should only be used on logical volumes that do not require mirror consistency recovery or where mirror consistency recovery is performed by another subsystem; for example swap. See the WARNINGS section for more details.
-C contiguous

Set the contiguous allocation policy. A contiguous logical volume has three characteristics:

- Physical extents are allocated in ascending order,
- No gap is allowed between physical extents within a mirror copy,
- Physical extents of any mirror copy all reside on a single physical volume.

Use the strict (-s) and contiguous (-C) options together to form various combined allocation policies on a logical volume. For example, -s y -C y defines a logical volume such that each mirror copy is contiguous, yet mirror copies of a logical extent cannot share the same physical volume.

contiguous can have one of the following values:

y  Set a contiguous allocation policy.

n  Do not set a contiguous allocation policy. This is the default.

The enforcement of a contiguous allocation policy using the -C y option is not supported on a striped logical volume.

-d schedule

Set the scheduling policy when a logical extent with more than one mirror is written. (The scheduling policy of a striped logical volume is striped and cannot be changed.) schedule can have one of the following values:

p  Establish a parallel scheduling policy. This is the default.

s  Establish a sequential scheduling policy. Use this value with care, because it leads to performance loss in most cases.

-D distributed

Set the distributed allocation policy. distributed can have one of the following values:

y  Turn on distributed allocation.

n  Turn off distributed allocation. This is the default.

When the distributed allocation policy is turned on, only one free extent is allocated from the first available physical volume. The next free extent is allocated from the next available physical volume. Allocation of free extents proceeds in round-robin order on the list of available physical volumes.

When the distributed allocation policy is turned off, all available free extents are allocated from each available physical volume before proceeding to the next available physical volume. This is the default.

The distributed allocation policy REQUIRES the PVG-strict allocation policy (-s g) to ensure that mirrors of distributed extents do not overlap (for maximum availability).

lvcreate(1M) will obtain the list of available physical volumes from /etc/lvmpvg. See vextend(1M) for more information on physical volume groups and /etc/lvmpvg.

When a logical volume with distributed extents is mirrored, the resulting layout is commonly referred to as EXTENT-BASED MIRRORED STRIPES.

Note that EXTENT-BASED MIRRORED STRIPES can be created without the distributed allocation policy by adding one extent at a time to the desired physical volumes through lvextend(1M).

The distributed allocation policy is incompatible with the striped scheduling policy (-i stripes) and the contiguous allocation policy (-C y).

The lvchange(1M) command can be used to assign the distributed allocation policy to an existing logical volume.

See lvdisplay(1M) for display values.

See EXAMPLES.

-i stripes

Set the number of disks to stripe across. stripes must be in the range 2 to the number of disks in the current volume group. If -i is provided and -I
is not, the stripe size is set to 8 kilobytes.

-\texttt{I} \textit{stripe\_size}  
Set the size in kilobytes of the stripe. \textit{stripe\_size} should be a power of 2 in the range 4 to 32768. If -\texttt{I} is provided and -\texttt{I} is not, the command will fail and return an error. Stripe size should be a value less than or equal to physical extent size and must be specified with the -\texttt{l} option.

-\texttt{l} \textit{le\_number}  
Allocate space to the logical volume, specified in logical extents. \textit{le\_number} is a decimal value in the range 1 to 65535 (the implementation limit). The default is described above.

Either -\texttt{l} or -\texttt{L} can be specified, but not both.

-\texttt{L} \textit{lv\_size}  
Allocate space to the logical volume, specified in megabytes. \textit{lv\_size} is a decimal value in the range 1 to 16776960 (the implementation limit). \textit{lv\_size} is rounded up to the nearest multiple of the logical extent size, equivalent to the physical extent size defined for the volume group by the \texttt{vgcreate} command (see \texttt{vgcreate}(1M)). The default is described above.

Either the -\texttt{l} or the -\texttt{L} option can be specified, but not both.

-\texttt{m} \textit{mirror\_copies}  
Set the number of mirror copies allocated for each logical extent. A mirror copy contains the same data as the original. \textit{mirror\_copies} can have the value 1 or 2. The default value is 0 (no mirror copies).

-\texttt{M} \textit{mirror\_write\_cache}  
Set the Mirror Write Cache flag. \textit{mirror\_write\_cache} can have one of the following values:

\begin{itemize}
  \item \texttt{y} Set Mirror Write Cache on. This is the default. Every write to a mirror copy is recorded in the Mirror Write Cache. The Mirror Consistency Record in the Volume Group Reserved Area on the disk is updated whenever there is a write to a logical track group that is not already recorded in the cache. This allows LVM to determine whether all the mirror copies are identical, even across system crashes. When the volume group is activated, the Mirror Consistency Record is used to perform mirror consistency recovery.
  \item \texttt{n} Set Mirror Write Cache to off. Mirror write does not incur an additional write to the Mirror Consistency Record.
\end{itemize}

-\texttt{n} \textit{lv\_name}  
Set the name of the new logical volume to \textit{lv\_name}, where \textit{lv\_name} is a simple file name, not a path name. The default is described above.

-\texttt{p} \textit{permission}  
Set the access permission. \textit{permission} can have one of the following values:

\begin{itemize}
  \item \texttt{w} Set the access permission to read-write. This is the default.
  \item \texttt{r} Set the access permission to read-only.
\end{itemize}

-\texttt{r} \textit{relocate}  
Set the logical volume bad block relocation policy. This is an obsolete flag available only to provide compatibility with prior HP-UX releases. The \textit{relocate} flag can have one of the following values:

\begin{itemize}
  \item \texttt{y} The default value of this flag.
\end{itemize}

This release does not provide the LVM bad block relocation feature. However for compatibility reasons, the value is maintained as a logical volume attribute. Displaying the logical volume attributes will show the value of the flag selected. However, regardless of the selection, no new relocations will be done.

If the volume group is activated on a different HP-UX release that provides the bad block relocation feature, bad blocks may be relocated depending upon the value of this flag. Although no new relocations will be done, any bad block relocations present on a logical volume that was activated on HP-UX releases that provided this feature, will be honored when the volume group is activated on this HP-UX release.

\texttt{y} is the default value of this flag.
lvcreate(1M)

-s strict

Set the strict allocation policy. Mirror copies of a logical extent can be allocated to share or not share the same physical volume or physical volume group. strict can have one of the following values:

y Set a strict allocation policy. Mirrors of a logical extent cannot share the same physical volume. This is the default.

g Set a PVG-strict allocation policy. Mirrors of a logical extent cannot share the same physical volume group. A PVG-strict allocation policy cannot be set on a logical volume in a volume group that does not have a physical volume group defined.

n Do not set a strict or PVG-strict allocation policy. Mirrors of a logical extent can share the same physical volume.

Striped Logical Volume Considerations
Striped and mirrored logical volumes are supported. A logical volume striped across stripes physical volumes is allocated in sets of stripes logical extents. A set corresponds to stripes physical extents if the volume is not mirrored or to stripes \( \times (\text{mirrorcopies} + 1) \) physical extents if the volume is mirrored. stripes is the number of physical volumes the logical volume is striped across. It is set with the option -i stripes. mirrorcopies is the number of mirror copies allocated for each logical extent. It is set with the -m option. The user data is striped across stripes physical extents of the set, and each of these extents is mirrored on mirrorcopies other physical extents of the same set. Striped logical volumes are only allocated using the strict or PVG-strict allocation policies. Each physical extent of a given set is allocated on a different physical volume in the volume group.

The total number of physical extents of a striped logical volume is always a multiple of stripes (or stripes \( \times (\text{mirrorcopies} + 1) \) if the volume is mirrored).

A minimum of stripes (or stripes \( \times (\text{mirrorcopies} + 1) \) if the volume is mirrored) physical volumes with adequate free space and meeting the allocation policy is needed to allocate a striped logical volume.

EXTERNAL INFLUENCES
Environment Variables
LANG determines the language in which messages are displayed.

If LANG is not specified or is null, it defaults to "C" (see lang(5)).

If any internationalization variable contains an invalid setting, all internationalization variables default to "C" (see environ(5)).

EXAMPLES
Create a logical volume in volume group /dev/vg02:

lvcreate /dev/vg02

Create a logical volume in volume group /dev/vg03 with non-strict allocation policy:

lvcreate -s n /dev/vg03

Create a logical volume of size 100 MB in volume group /dev/vg03:

lvcreate -L 100 /dev/vg03

Create a logical volume of size 90 MB striped across 3 disks with a stripe size of 64 KB:

lvcreate -L 90 -i 3 -I 64 /dev/vg03

Create a logical volume of size 90 MB striped across 3 disks with one mirror copy and a stripe size of 64 KB:

lvcreate -L 90 -i 3 -I 64 -m 1 /dev/vg03

Distributed Allocation Policy
This example shows how the -D y option can be used to create EXTENT-BASED MIRRORED STRIPES.

Assume that volume group /dev/vgtest has two physical volume groups: pvg1 and pvg2.

Assume that each physical volume group has 2 physical volumes.

Assume that the first physical volume in each pvg has 3 extents free and the second physical volume in each pvg has 2 extents free.
The following command creates a logical volume in vgtest with EXTENT-BASED MIRRORED STRIPES:

```
 lvcreate -D y -s g -m 1 -l 5 /dev/vgtest
```

The distributed allocation proceeds as follows:
- A free extent is allocated from the 1st pv in pvg1.
- A free extent is allocated from the 2nd pv in pvg1.
- A free extent is allocated from the 1st pv in pvg1.
- A free extent is allocated from the 2nd pv in pvg1.
- A free extent is allocated from the 1st pv in pvg1.
- Mirrors for the five extents are then allocated from the free extents in pvg2 in a similar manner.

**WARNINGS**

The root, swap, and dump logical volumes (see `lvlnboot(1M)`) must be created with contiguous allocation policy.

By setting mirror consistency recovery off, crash recovery time will be reduced. After a system crash the mirrored logical volume will be available, but there may not be consistent data across each mirror copy. The only types of data that can safely be put on a mirrored logical volume with mirror consistency recovery turned off are:
- data not needed after a crash, such as swap or other raw scratch data, or
- data that an application itself will automatically reconstruct; for example, a raw logical volume for which a database keeps a log of incomplete transactions.

In order to create a logical volume greater than 2 terabytes, the kernel must be updated to support this feature, and the corresponding feature enabling patch must be installed.

In order to create a striped and mirrored logical volume, the kernel must be updated to support this feature, and the corresponding feature enabling patch must be installed.

The creation of striped and mirrored logical volume(s) may prevent the import and activation of the volume group on an earlier HP-UX release. See `lvcreate(1M)` on the earlier release to see if it explicitly states that striping and mirroring is supported. If the striped and mirrored logical volumes of the volume group are removed or un-mirrored, the volume group becomes again compatible with the older HP-UX releases.

**SEE ALSO**

`lvchange(1M), lvdisplay(1M), lvextend(1M), lvreduce(1M), lvremove(1M), pvchange(1M), lvm(7)`.
lvdisplay(1M)

NAME
lvdisplay - display information about LVM logical volumes

SYNOPSIS
/usr/sbin/lvdisplay [-k] [-v] lv_path...

Remarks
Mirrored disk information requires the installation of the optional HP MirrorDisk/UX software, which is not included in the standard HP-UX operating system.

DESCRIPTION
The lvdisplay command displays the characteristics and status of each logical volume specified by lv_path.

Options and Arguments
lvdisplay recognizes the following options and arguments:

lv_path The block device path name of a logical volume, for example, /dev/vg00/lvol1.
-v For each logical volume, display the physical volume distribution, and the mapping of the logical extents onto the physical extents of the physical volumes.
-k This option displays the same information as the -v option, except in the column where PV Name is displayed, the pvkey (Physical Volume Number in VG) will be displayed instead. Use this option with the -v option.

Display Without -v Option
If you omit the -v option, lvdisplay displays the following information for each logical volume:

--- Logical volumes ---
LV Name The block device path name of the logical volume.
VG Name The path name of the volume group.
LV Permission Access permission: read-only or read/write.
LV Status State of the logical volume:
available/stale Available but contains physical extents that are not current.
available/syncd Available with no stale extents.
unavailable Not available for use.
Mirror copies Number of physical extents beyond the original allocated for each logical extent; i.e., the number of mirrors: 0, 1, or 2.

Consistency Recovery
Mode of mirror consistency recovery which determines how LVM performs mirror consistency recovery during volume group activation:

MWC Recover mirror consistency by using the Mirror Write Cache and Mirror Consistency Record. Implies that Mirror Write Cache is on.
NOMWC Recover mirror consistency by going through all logical extents and copying data from a non-stale copy to the other mirror copies. Implies that Mirror Write Cache is off.
NONE No mirror consistency recovery during volume group activation on this logical volume following a system crash. The user of the logical volume is responsible for ensuring mirror consistency. Implies that Mirror Write Cache is off.

Schedule Striped, sequential or parallel scheduling policy. Striped policy is by default parallel scheduling for mirrored I/O.

LV Size (Mbytes) Size of the logical volume in megabytes (MB).
Current LE
Number of logical extents currently in the logical volume.

Allocated PE
Number of physical extents allocated to the logical volume.

Stripes
The number of stripes. If this field is 0, then the logical volume is not striped.

Stripe Size (Kbytes)
The size of each stripe in kilobytes (KB).

Bad block
Bad block relocation policy.

Allocation
Current allocation state, displayed as one of:

- **non-strict**
- **strict**
- **PVG-strict**
- **contiguous**

- **non-strict/contiguous**
- **strict/contiguous**
- **PVG-strict/contiguous**
- **PVG-strict/distributed**
- **PVG-strict/partially-distributed**

- **distributed**

- **partially-distributed**

Physical extents are allocated in an ascending order without any gap between adjacent extents. All physical extents of a given mirror are contained in a single physical volume.

- **non-strict**

Physical extents that belong to the same logical extent can be allocated on the same physical volume or physical volume group.

- **PVG-strict**

Mirror copies for a logical extent are not allocated on the same physical volume group.

- **strict**

Mirror copies for a logical extent are not allocated on the same physical volume.

IO Timeout (Seconds)
The IO timeout used by LVM for all IO to this logical volume. A value of default, indicates that the system will use the value of "forever". (Note: the actual duration of a request may exceed this timeout value when the underlying physical volume(s) have timeouts which either exceed this value or are not integer multiples thereof.)

Display With --v Option
If you specify the --v option, lvdisplay also lists the distribution of each logical volume across the physical volumes of the volume group and the mapping of each logical extent of the logical volume on the physical extents of the physical volume.

--- Distribution of logical volume ---
The distribution of logical volume lv_path across the physical volumes of the volume group, displayed in the following columns:

- **PV Name**
  The block device path name of the physical volume where the logical extents are allocated.

- **PVNUM**
  The Physical Volume Number in VG (if -k option is specified).

- **LE on PV**
  Number of logical extents allocated on the physical volume.

- **PE on PV**
  Number of physical extents allocated on the physical volume.

--- Logical extents ---
The mapping of logical extents onto physical extents, displayed in the following columns:
**lvdisplay(1M)**

**Logical extent number.**

**PV1** The block device path name of the physical volume that corresponds to the location of the first physical extent of the logical extent.

**PE1** First physical extent number allocated to the logical extent.

**Status 1** Status of the first physical extent: **stale** or **current**.

The following columns are displayed for one or two mirror copies:

**PV2** The block device path name of the physical volume that corresponds to the location of the second physical extent (first copy) of the logical extent.

**PE2** Second physical extent number allocated to the logical extent.

**Status 2** Status of the second physical extent: **stale** or **current**.

The following columns are displayed for two mirror copies:

**PV3** The block device path name of the physical volume that corresponds to the location of the third physical extent (second copy) of the logical extent.

**PE3** Third physical extent number allocated to the logical extent.

**Status 3** Status of the third physical extent: **stale** or **current**.

**EXTERNAL INFLUENCES**

**Environment Variables**

**LANG** determines the language in which messages are displayed.

If **LANG** is not specified or is null, it defaults to "C" (see **lang**(5)).

If any internationalization variable contains an invalid setting, all internationalization variables default to "C" (see **environ**(5)).

**EXAMPLES**

Display information about a logical volume:

```
 lvdisplay /dev/vg01/lvol3
```

Display all the available information about a logical volume, including the characteristics, status and distribution map:

```
 lvdisplay -v /dev/vg01/lvol3
```

Display all the available information about a logical volume, but display **pvkey** instead of **PV Name** in the status and distribution map.

```
 lvdisplay -v -k /dev/vg01/lvol3
```

**SEE ALSO**

lvchange(1M), lvcreate(1M), lvextend(1M), lvreduce(1M), pvdisplay(1M), vgdisplay(1M).
NAME
lvextend - increase space, increase mirrors for LVM logical volume

SYNOPSIS
/usr/sbin/lvextend [-A autobackup] [-l le_number | -L lv_size | -m mirror_copies [-s]]
 lv_path [pv_path ... | pvg_name ...]

Remarks
Mirrored disk operations require the installation of the optional HP MirrorDisk/UX software, which is not
included in the standard HP-UX operating system.

lvextend cannot be performed if the volume group is activated in shared mode.

DESCRIPTION
The lvextend command can increase a logical volume’s allocated extents, or increase its number of mir-
rored copies.

Other logical volume characteristics can be modified with the lvchange and lvreduce commands (see
lvchange(1M) and lvreduce(1M)).

To limit the allocation to specific physical volumes, specify the physical volume names as pv_path argu-
ments or specify the physical volume group names as pvg_name arguments. Otherwise, all of the physical
volumes in a volume group are available for allocating new physical extents. LVM always ensures that phy-
sical extent allocation can satisfy the current allocation policy or policies. If a physical volume is not suit-
able for use with a certain allocation policy, it is not used during physical extent allocation, even if it is
specified in a pv_path argument or indirectly in a pvg_name argument.

The pvg_name argument is allowed only if one of the allocation policies of the logical volume is PVG-strict.

Options and Arguments
The -m option is only meaningful if the optional HP MirrorDisk/UX software has been installed.

lvextend recognizes the following options and arguments:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>lv_path</td>
<td>The block device path name of a logical volume.</td>
</tr>
<tr>
<td>pv_path</td>
<td>The block device path name of a physical volume.</td>
</tr>
<tr>
<td>pvg_name</td>
<td>The name of a physical volume group (see lvmpvg(4)).</td>
</tr>
<tr>
<td>-A autobackup</td>
<td>Set automatic backup for this invocation of this command. autobackup can have one of the following values:</td>
</tr>
<tr>
<td></td>
<td>y Automatically back up configuration changes made to the logical volume. This is the default.</td>
</tr>
<tr>
<td></td>
<td>n Do not back up configuration changes this time.</td>
</tr>
<tr>
<td>-l le_number</td>
<td>Increase the space allocated to the logical volume, specified in logical extents. le_number is a decimal value greater than the current number of logical extents, in the range 1 to 65535 (the implementation limit). One, and only one, -l, -L, or -m option must be supplied.</td>
</tr>
<tr>
<td>-L lv_size</td>
<td>Increase the space allocated to the logical volume, specified in megabytes. lv_size is a decimal value greater than the current logical volume size, in the range 1 to 16776960 (the implementation limit). lv_size is rounded up to the nearest multiple of the logical extent size, equivalent to the physical extent size defined for the volume group by thevgcreate command (see vgcreate(1M)). One, and only one, -l, -L, or -m option must be specified.</td>
</tr>
<tr>
<td>-m mirror_copies</td>
<td>Set the number of mirror copies allocated for each logical extent. A mirror copy contains the same data as the original. mirror_copies can have the value 1 or 2. It must be greater than the current value.</td>
</tr>
</tbody>
</table>
Data in the new copies is synchronized unless the -s option is specified. The synchronization process can be time consuming depending on hardware characteristics and the amount of data.

One, and only one, -l, -L, or -m option must be specified.

- **s**
  
  Do not synchronize the new mirror copies. This may affect data high availability so use lvsync or vgsync to synchronize the mirrors.

  The -m option must be specified along with this option.

### Striped Logical Volume Considerations

Striped and mirrored logical volumes are supported.

An increase in size of a striped logical volume is done by increments of *stripes* logical extents. One increment corresponds to *stripes* physical extents if the volume is not mirrored or to *stripes* * (mirror_copies + 1) physical extents if the volume is mirrored. *stripes* is the number of disks the logical volume is striped across. It is set with the option -i *stripes* of the lvcreate command. mirror_copies is the number of mirror copies allocated for each extent. It is set with the -m option of the lvcreate and lvextend commands. LVM striped logical volumes are always allocated using the strict or PVG-strict allocation policies. Each physical extent of an increment is allocated on a different physical volume in the volume group.

A size increase of a striped volume requires at least *stripes* (or *stripes* * (mirror_copies + 1) if the volume is mirrored) physical volumes with adequate free space and meeting the allocation policy.

An increase of the number of mirror copies of a striped volume requires at least (*stripes* times the number of copies to add) physical volumes with adequate free space and meeting the allocation policy.

### EXTERNAL INFLUENCES

**Environment Variables**

- **LANG** determines the language in which messages are displayed.

  If LANG is not specified or is null, it defaults to "C" (see lang(5)).

  If any internationalization variable contains an invalid setting, all internationalization variables default to "C" (see environ(5)).

### EXAMPLES

Increase the number of the logical extents of a logical volume to 100:

```
 lvextend -l 100 /dev/vg01/lvol3
```

Increase the logical volume size to 400 MB:

```
 lvextend -L 400 /dev/vg01/lvol4
```

Allocate two mirrors (that is, two copies of the original) for each logical extent of a logical volume:

```
 lvextend -m 2 /dev/vg01/lvol5
```

Mirror a logical volume onto a particular physical volume:

```
 lvextend -m 1 /dev/vg00/lvol3 /dev/dsk/c0t3d0
```

Allocate one mirror and do not synchronize the new mirror copy:

```
 lvextend -m 1 -s /dev/vg04/lvol1
```

Increase the size of a file system existing on a logical volume:

First, increase the size of the logical volume.

```
 lvextend -L 400 /dev/vg06/lvol3
```

Unmount the file system.

```
 umount /dev/vg06/lvol3
```

Extend the file system to occupy the entire (larger) logical volume.

```
 extendfs /dev/vg06/r1vol3
```

Remount the file system.
mount /dev/vg06/lvol3 /mnt

WARNINGS
In order to create a logical volume greater than 2 terabytes, the kernel must be updated to support this feature, and the corresponding feature enabling patch must be installed.

In order to create a striped and mirrored logical volume, the kernel must be updated to support this feature, and the corresponding feature enabling patch must be installed.

The creation of striped and mirrored logical volume(s) may prevent the import and activation of the volume group on an earlier HP-UX release. See lvcreate(1M) on the earlier release to see if it explicitly states that striping and mirroring is supported. If the striped and mirrored logical volumes of the volume group are removed or un-mirrored, the volume group becomes again compatible with the older HP-UX releases.

SEE ALSO
lvchange(1M), lvcreate(1M), lvdisplay(1M), lvreduce(1M), lvsync(1M), pvchange(1M), pvdisplay(1M), vgsync(1M), lvm(7).
lvlnboot(1M)

NAME
lvlnboot - prepare LVM logical volume to be root, boot, primary swap, or dump volume

SYNOPSIS

Remarks
lvlnboot cannot be performed if the volume group is activated in shared mode.

DESCRIPTION
The lvlnboot command updates all physical volumes in the volume group so that the logical volume becomes the root, boot, primary swap, or a dump volume when the system is next booted on the volume group. If a nonexistent logical volume is specified, this command fails. If a different logical volume is already linked to the root or primary swap, the command fails.

This command should be run in recovery mode (-R option) whenever the configuration of the root volume group is affected by one of the following commands: lvextend(1M), lvmerge(1M), lvreduce(1M), lvsplit(1M), pvmove(1M), lvremove(1M), vgextend(1M), or vgreduce(1M). Starting with HP-UX Release 10.0, this is done automatically.

The boot information stored on disks assumes a certain ordering of disks listed in /etc/lvmtab. The lvlnboot command should be run in recovery mode (-R option) when there is a change to the order of the disks in /etc/lvmtab for any bootable volume group (see vgimport(1M) and vgscan(1M) for more information).

Options and Arguments
lvlnboot recognizes the following options and arguments:

-vg_name
The path name of a volume group.

-A autobackup
Set automatic backup for this invocation of this command. autobackup can have one of the following values:

y Automatically back up configuration changes made to the logical volume. This is the default.

After this command executes, the vgcfgbackup command (see vgcfgbackup(1M)) is executed for the volume group to which the logical volume belongs.

n Do not back up configuration changes this time.

-b boot_lv
Define boot_lv to be the boot volume the next time the system is booted on the volume group. boot_lv must be the first logical volume on the physical volume. boot_lv must be contiguous, and must not allow bad block relocation.

boot_lv is used to locate the boot file system during the boot process. The boot file system has the kernel which is read by the boot loader hpux(1M).

-d dump_lv
Define dump_lv to be one of the dump volumes the next time the system is booted on the volume group. dump_lv must be a contiguous logical volume and cannot have Bad Block Relocation enabled.

The command updates the Boot Data Reserved Area of each bootable physical volume in the volume group (see pvcreate(1M)).

The combined size of all the dump volumes should be at least 2048 bytes larger than the total memory of the system. The additional 2 KB is used to safeguard against a dump to the bottom of the disk.

Multiple dump devices can be configured, but each dump_lv must be entered with a separate lvlnboot command line.

-r root_lv
Define root_lv to be the root volume the next time the system is booted on this volume group. root_lv must be a contiguous logical volume and cannot have bad block relocation enabled.
If root_lv is the first logical volume on the physical volume, then it is configured as the combined root-boot volume. Otherwise, root_lv is configured as the separate root volume in which case a separate boot volume needs to be configured using the lvlnboot -b option.

Either the separate root or the separate boot volume can be configured first.

The command updates the Boot Data Reserved Area of each bootable physical volume (see pvcreate(1M)) to enable the volume group to be used to locate the root file system. root_lv is also used as the root volume during a maintenance-mode boot (see hpux(1M)).

The physical volumes containing root_lv must have been created using the pvcreate -B option (see pvcreate(1M)), indicating that that physical volume is to be used as a bootable physical volume. Also, the mkboot command (see mkboot(1M)) must have been run on the physical volume to create the LIF area at the top of the physical volume (see lif(4)).

-R
Recover any missing links to all of the logical volumes specified in the Boot Data Reserved Area and update the Boot Data Reserved Area of each bootable physical volume in the volume group (see pvcreate(1M)).

-s swap_lv
Define swap_lv to be the primary swap volume the next time the system is booted on the volume group. swap_lv must be a contiguous logical volume, and a root logical volume must have been previously defined with this command.

The command updates the Boot Data Reserved Area of each bootable physical volume in the volume group (see pvcreate(1M)). Any existing swap area previously defined must be removed via lvrmboot(1M).

-c
During normal boots (versus maintenance-mode boots, see hpux(1M)), this command is automatically executed by /sbin/ioinitrc (see inittab(4)).

Since this command is performed during boot, it does not need to be performed manually unless /stand/rootconf is missing in a separate root/boot configuration (or alternatively, performing a normal reboot will recreate this file).

This command updates the /stand/rootconf file with the location of the root volume in the currently booted volume group.

The /stand/rootconf file is used during maintenance-mode boots to locate the root volume for volume groups with separate boot and root volumes.

During maintenance-mode boots, since the root volume group is not activated, lvlnboot -c does not update /stand/rootconf. For separate root/boot configurations, maintenance-mode boot will fail if /stand/rootconf does not already exist with the correct location of the root volume. See WARNINGS.

When a new volume group with separate boot and root volumes is created, the first boot must be a normal boot (versus a maintenance-mode boot), so that /stand/rootconf gets created.

This option does not allow updating /stand/rootconf for any volume group other than the one that is booted.

-v
Print verbose messages. With no other arguments present, print information on root, boot, swap, and dump logical volumes. If a combined root-boot volume is configured, no information for the boot volume is displayed.

EXTERNAL INFLUENCES
Environment Variables
LANG determines the language in which messages are displayed.

If LANG is not specified or is null, it defaults to "C" (see lang(5)).

If any internationalization variable contains an invalid setting, all internationalization variables default to "C" (see environ(5)).
**EXAMPLES**

The following examples show configuration of a combined root-boot volume.

Create a root volume group, `vglvmroot`, containing root, swap, and dump logical volumes. Assume that an appropriate directory called `/dev/vglvmroot` and a corresponding group file already exist (see `lvm(7)`).

First, initialize the disk, say `/dev/dsk/c0t0d0`, so that it can be used as an LVM boot disk.

```bash
pvcreate -B /dev/rdsk/c0t0d0
```

Place the LIF information on the disk using the `mkboot` command.

```bash
mkboot /dev/rdsk/c0t0d0
```

Create the volume group `vglvmroot`.

```bash
vgcreate /dev/vglvmroot /dev/dsk/c0t0d0
```

Create a logical volume that is suitable for use as the root volume. This logical volume has to be the first in the volume group and should be a contiguous volume with bad block relocation turned off.

```bash
lvcreate -n root -L 120 -C y -r n /dev/vglvmroot
```

Create a logical volume that will be used as primary swap. This volume should be contiguous.

```bash
lvcreate -n swap -L 64 -C y /dev/vglvmroot
```

Create a logical volume that will be used as the dump volume. This volume should be contiguous.

```bash
lvcreate -n dump -L 64 -C y /dev/vglvmroot
```

Specify that the logical volume, `root`, will be used as the root volume.

```bash
lvlnboot -r /dev/vglvmroot/root
```

Specify that the logical volume, `swap`, will be used as the primary swap.

```bash
lvlnboot -s /dev/vglvmroot/swap
```

Specify that the logical volume, `dump`, will be used as the dump volume.

```bash
lvlnboot -d /dev/vglvmroot/dump
```

Display the results of the previous operations.

```bash
lvlnboot -v /dev/vglvmroot
```

The following examples show configuration of separate root and boot volumes.

Create a root volume group, `vglvmroot`, containing root, boot, swap, and dump logical volumes. Assume that an appropriate directory called `/dev/vglvmroot` and a corresponding group file already exist (see `lvm(7)`).

First, initialize the disk, say `/dev/dsk/c0t0d0`, so that it can be used as an LVM boot disk.

```bash
pvcreate -B /dev/rdsk/c0t0d0
```

Place the LIF information on the disk using the `mkboot` command.

```bash
mkboot /dev/rdsk/c0t0d0
```

Create the volume group `vglvmroot`.

```bash
vgcreate /dev/vglvmroot /dev/dsk/c0t0d0
```

Create a logical volume that is suitable for use as the boot volume. This logical volume has to be the first in the volume group and should be a contiguous volume with bad block relocation turned off.

```bash
lvcreate -n boot -L 24 -C y -r n /dev/vglvmroot
```

Create a logical volume that is suitable for use as the root volume. This logical volume should be a contiguous volume with bad block relocation turned off.

```bash
lvcreate -n root -L 64 -C y -r n /dev/vglvmroot
```

Create a logical volume that will be used as primary swap. This volume should be contiguous.

```bash
lvcreate -n swap -L 64 -C y /dev/vglvmroot
```

Create a logical volume that will be used as primary swap. This volume should be contiguous.

```bash
lvcreate -n swap -L 64 -C y /dev/vglvmroot
```
Create a logical volume that will be used as the dump volume. This volume should be contiguous.

```bash
lvcreate -n dump -L 64 -C y /dev/vg/lvmroot
```

Specify that the logical volume, `root`, will be used as the root volume.

```bash
lvlnboot -r /dev/vg/lvmroot/root
```

Specify that the logical volume, `boot`, will be used as the boot volume.

```bash
lvlnboot -b /dev/vg/lvmroot/boot
```

Specify that the logical volume, `swap`, will be used as the primary swap.

```bash
lvlnboot -s /dev/vg/lvmroot/swap
```

Specify that the logical volume, `dump`, will be used as the dump volume.

```bash
lvlnboot -d /dev/vg/lvmroot/dump
```

Display the results of the previous operations.

```bash
lvlnboot -v /dev/vg/lvmroot
```

The following example shows configuration of multiple dump volumes.

Specify that logical volumes `/dev/vg/00/swap1`, `/dev/vg/00/dump2`, and `/dev/vg/00/dump3` should be used as the dump logical volumes and that `/dev/vg/00/swap1` should also be used as primary swap. Assume that the volume group and the logical volumes have been created and the logical volumes are contiguous.

```bash
lvlnboot -s /dev/vg/00/swap1
lvlnboot -d /dev/vg/00/swap1
lvlnboot -d /dev/vg/00/dump2
lvlnboot -d /dev/vg/00/dump3
```

**WARNINGS**

**Dump Volume Warnings**

At the HP-UX 11.00 release and forward, the `lvlnboot` command will support any size dump device depending upon the IODC firmware addressability of the system. If the configured dump logical volume is out of the range of what the firmware can address, the `lvlnboot` command will return an error message such as "Unable to configure dump logical volume. Dump logical volume size beyond the IODC max address."

**Separate Root/Boot Warnings**

Whenever `mkboot(1M)` is used to restore the LIF area of a damaged root physical volume, the `-b boot_lv` option of `lvlnboot` must be performed afterwards to record the boot volume information inside the new LIF (see `lif(4)`). Subsequent `lvlnboot` commands such as `lvlnboot -R` are dependent on the `boot_lv` information inside the LIF.

If the `-v` option does not locate the boot volume `boot_lv`, and the `-r root_lv` has not yet been performed, then performing the `-r root_lv` option will enable the boot volume to be located. The `lvlnboot` command derives the location of boot volume from the location of the root volume.

**Separate Root/Boot Maintenance-Mode Warnings**

When creating additional root volumes with separate root/boot, a normal boot must be performed on each new root volume so that `/stand/rootconf`, which is required for maintenance-mode boots (see `hpux(1M)`), gets created for each new root volume.

Mirrored `root_lv` volumes should start at the same offset on each physical volume so that the location stored in `/stand/rootconf` works for maintenance-mode boots off of any mirror.

**FILES**

`/stand/rootconf` Contains the location of the root volume. Used during maintenance-mode boots (see `hpux(1M)`) to locate the root volume for volume groups with separate boot and root volumes.

**SEE ALSO**

`lvcreate(1M)`, `lvrmboot(1M)`, `mkboot(1M)`, `pvcreate(1M)`, `vgcreate(1M)`, `inittab(4)`, `lif(4)`, `lvm(7)`. 
NAME
lvmchk - check if disk volume is under HP Logical Volume Manager (LVM) control

SYNOPSIS
/usr/sbin/lvmchk pv_path

DESCRIPTION
The lvmchk command returns information about the disk volume specified by the pv_path parameter.

Options
lvmchk recognizes the following option:

   pv_path     The device path name of a disk volume. pv_path can be a block device, raw disk, or a logical volume.

RETURN VALUE
lvmchk returns the following values

   0       The disk specified by pv_path is under the control of LVM.
   1       The disk specified by pv_path is not under the control of LVM.
   255     There was an internal I/O error on the disk specified by pv_path, or a command line syntax or usage error was encountered.

EXAMPLES
Examine whether disks /dev/dsk/c0t6d0, /dev/rdsk/c0t6d0, and /dev/vg00/lvol1 belong to LVM.

lvmchk /dev/dsk/c0t6d0
lvmchk /dev/rdsk/c0t6d0
lvmchk /dev/vg00/lvol1

AUTHOR
lvmchk was developed by HP.

SEE ALSO
pvcreate(1M), pvdisplay(1M).
NAME
    lvmerge - merge two LVM logical volumes into one logical volume

SYNOPSIS
    /usr/sbin/lvmerge [-A autobackup] [-s] copy_lv_path master_lv_path

Remarks
    This command requires the installation of the optional HP MirrorDisk/UX software, which is not included
    in the standard HP-UX operating system.
    lvmerge cannot be performed if the volume group is activated in shared mode.

DESCRIPTION
    The lvmerge command merges two logical volumes of the same size. The copy_lv_path is added as a
    mirrored copy of master_lv_path and only the master_lv_path logical volume is retained. The number of mir-
    rored copies of the master_lv_path is increased by the number of copies in the copy_lv_path.
    Data previously contained in the copy_lv_path will be resynchronized using the data in the master_lv_path.
    All new data on the copy_lv_path will be destroyed and the copy_lv_path logical volume will be removed.
    By default, lvmerge completes after all the extents are resynchronized. When run with the -s option, if
    there is no bitmap (see below), lvmerge returns without resynchronizing the remaining logical volume.
    Using the -s option may affect data high availability as extents may remain stale. Use lvsync or
    vgsync to synchronize the mirrors.
    The normal usage for this command is to merge previously mirrored logical volumes that have been split
    using the lvsplit command. However, the two logical volumes are not required to have been the result
    of a previous lvsplit operation.

    NOTE: Whenever a mirrored logical volume is split into two logical volumes using lvsplit, a bitmap is
    stored that keeps track of all writes to either logical volume in the split pair. When the two logical volumes
    are subsequently merged using lvmerge, the bitmap is used to decide which areas of copy_lv_path need
    to be resynchronized with the master_lv_path. This bitmap remains in existence until one of the following
    conditions occurs:
    • The merge is completed.
    • One of the logical volumes is extended, reduced, or split again.
    • The volume group is deactivated, or cross-activated to shared mode (see vgchange(1M)).
    • The system is rebooted.
    If there is no bitmap available, all the physical extents from the copy_lv_path are marked stale. If there is
    a bitmap available, the -s option is ignored.

Options and Arguments
    lvmerge recognizes the following options and arguments:
    copy_lv_path    The block device path name of a logical volume that will be removed.
    master_lv_path  The block device path name of a logical volume that will remain.
    -A autobackup   Set automatic backup for this invocation of this command. autobackup can have
                     one of the following values:
                     y  Automatically back up configuration changes made to the logical volume.
                     This is the default.
                     After this command executes, the vgcfgbackup command (see vgcfgbackup(1M)) is executed for the volume group to which the logical
                     volume belongs.
                     n  Do not back up configuration changes this time.

    -s              If there is no bitmap, do not synchronize the new mirrors in the master_lv_path
                     after the merge. Using the -s option may affect data high availability as extents
                     may remain stale. Use lvsync or vgsync to synchronize the mirrors.
lvmerge(1M)

(Requires Optional HP MirrorDisk/UX Software)

EXTERNAL INFLUENCES

Environment Variables

LANG determines the language in which messages are displayed.

If LANG is not specified or is null, it defaults to "C" (see lang(5)).

If any internationalization variable contains an invalid setting, all internationalization variables default to "C" (see environ(5)).

EXAMPLES

Merge /dev/vg00/lvol1b with /dev/vg00/lvol1. Data in /dev/vg00/lvol1b will be overwritten by /dev/vg00/lvol1 and /dev/vg00/lvol1b will be removed.

lvmerge /dev/vg00/lvol1b /dev/vg00/lvol1

Merge /dev/vg01/lvol1b with /dev/vg01/lvol1. Data in /dev/vg01/lvol1b will be overwritten by /dev/vg01/lvol1 and /dev/vg01/lvol1b will be removed. If the bitmap does not exist, then mark all extents associated with /dev/vg01/lvol1b as stale and complete without resynchronizing.

lvmerge -s /dev/vg01/lvol1b /dev/vg01/lvol1

WARNINGS

lvmerge does not check to guarantee that the allocation policy of master_lv_path is preserved after the merge.

SEE ALSO

lvcreate(1M), lvextend(1M), lvsplit(1M), lvsync(1M), vgsync(1M).
NAME
lvmmigrate - prepare root file system for migration from partitions to LVM logical volumes

SYNOPSIS
/usr/sbin/lvmmigrate [-d disk_special_file] [-e file_system ...] [-f] [-i file_system ...] [-n] [-v]

DESCRIPTION
The lvmmigrate command records the configuration information of the current system in the LIF volume of the boot section for use with a subsequent cold-install process. If there is no LIF volume on the disk, lvmmigrate creates it using lifinit(1), then records the information in a LIF file named CUSTOM. A copy of the LIF file is saved as /tmp/LVMigrate.CFG. The information is also written to file /tmp/LVMigrate for reviewing. The install process looks for the LIF file CUSTOM, and if it exists, uses the information found as the configuration defaults for the root volume group and the root file systems. After the install process has completed, a copy of the CUSTOM final configuration can be found on the newly created system in the file /usr/lib/sw/hpux.install/config.local.

All file system entries in the /etc/mnttab and /etc/fstab files are read. lvmmigrate also searches for unmounted file systems and possible character data sections in unused disk areas. The file systems appropriate for the root volume group are marked for migration. The default file systems are: /, /home, /opt, /tmp, /usr, /var, and any file system with a mount path beginning with: /home/, /opt/, /tmp/, /usr/, /var/.

lvmmigrate displays the following information on the standard output: disks and file system names that are marked for migration, disk areas and file systems to be backed up by the user, and instructions for reinstallation.

After executing lvmmigrate, the user must back up the file systems and any raw device section having useful data to tape. The system is then reinstalled on logical volumes using the configuration information recorded by lvmmigrate.

Options
lvmmigrate recognizes the following options:

- **-d disk_special_file** Use the specified root disk for reinstallation. Without this option, the current root disk (where root file system / is currently located) is assumed and the configuration is recorded in the boot section.

- **-e file_system ...** Exclude each specified default file system from the root volume group. Note that the / file system cannot be excluded.

- **-f** Force the recording of configuration information. Information is recorded in a LIF file named CUSTOM in the boot section. Without this option, if there is a file system or LVM record in the boot section, no write is done and a warning message is displayed.

- **-i file_system ...** Include each specified file system in the root volume group, along with the default file systems.

- **-n** Perform a "no write" operation for preview purposes. Migration information is displayed on the terminal screen, but is not recorded in the boot section of the disk. The CUSTOM LIF file is not written, but the files /tmp/LVMigrate and /tmp/LVMigrate.CFG are still created.

- **-v** Display all disks, file systems, and possible raw sections present in the system.

EXAMPLES
Prepare a system for migration to root logical volumes. Create a file in the LIF area that the cold-install can use to read default configuration information. Specify verbose mode. Create files /tmp/LVMigrate and /tmp/LVMigrate.CFG:

```bash
lvmmigrate -v
```

Display a detailed list of the disks, file systems, and possible raw data sections present in the current system.
lvmmigrate -v -n

Include file system /mnt in the root volume group for migration and exclude file system /usr/source. Write configuration information in the boot section of disk /dev/dsk/c1t0d0:

lvmmigrate -d /dev/dsk/c1t0d0 -i /mnt -e /usr/source

WARNINGS
Use of the -f option results in overwriting the contents of the boot section. Before using the -f option be sure to back up all data on the boot section of the disk specified with the -d option.

If there is no LIF volume, lvmmigrate uses lifinit to create it (see lifinit(1)). If file CUSTOM already exists in the LIF volume, lvmmigrate rewrites it.

Caution: All data on disks being used for reinstallation must be backed up to a separate device because the install process overwrites data on all disks used in the new root volume group.

SEE ALSO
lifinit(1).
lvreduce(1M)

NAME
lvreduce - decrease space allocation or the number of mirror copies of logical volumes

SYNOPSIS
/usr/sbin/lvreduce [-A autobackup] [-f] -l le_number lv_path
/usr/sbin/lvreduce [-A autobackup] [-f] -L lv_size lv_path
/usr/sbin/lvreduce [-A autobackup] [-m mirror_copies lv_path [pv_path ...]]
/usr/sbin/lvreduce [-A autobackup] -k -m mirror_copies lv_path

Remarks
Mirrored disk operations require the installation of the optional HP MirrorDisk/UX software, which is not included in the standard HP-UX operating system.

lvreduce cannot be performed if the volume group is activated in shared mode.

DESCRIPTION
The lvreduce command reduces the number of logical extents allocated to a logical volume specified by lv_path. The excess physical extents in the logical volume and any mirror copies are deallocated.

Alternatively, it reduces the number of mirror copies in the logical volume. The physical extents that comprise the deleted mirror copy or copies are deallocated. If pv_path... is specified, the mirror or mirrors to be removed will be deallocated from those specific physical volumes.

lvreduce asks for confirmation before deallocating logical extents if the -f option is omitted.

Options and Arguments
The -m option and pv_path argument are only meaningful if the optional HP MirrorDisk/UX software has been installed on the system.

lvreduce recognizes the following options and arguments:

- lv_path The block device path name of a logical volume.
- pv_path The block device path name of a physical volume.
- -A autobackup Set automatic backup for invocation of this command. autobackup can have one of the following values:
  - y Automatically back up configuration changes made to the logical volume. This is the default.
  - n Do not back up configuration changes.

- -f Force reduction of the number of logical extents without first requesting confirmation.

This option can be dangerous when there is a file system on the lv_path that is larger than the size that the logical volume is being reduced to. If the file system is unmounted, the -f option forces the reduction of the logical volume without reducing the file system. The file system becomes corrupt and is not mountable. If the file system is mounted, lvreduce fails, preventing a mounted file system from becoming corrupted.

- -l le_number Decrease the space allocated to the logical volume, specified in logical extents. le_number is a decimal value smaller than the current number of logical extents, in the range 1 to 65535 (the implementation limit).

One, and only one, -l, -L, or -m option must be supplied.

- -L lv_size Decrease the space allocated to the logical volume, specified in megabytes. lv_size is a decimal value smaller than the current logical volume size, in the range 1 to 16776960 (the implementation limit). lv_size is rounded up to the nearest multiple of the logical extent size, equivalent to the physical extent size defined for the volume group by the vgcreate command (see vgcreate(1M)).
One, and only one, \(-l\), \(-L\), or \(-m\) option must be specified.

\(-m\) \textit{mirror\_copies}  
Reduce the number of mirror copies allocated for each logical extent. A mirror copy contains the same data as the original. \textit{mirror\_copies} can have the value \texttt{0} or \texttt{1}. It must be smaller than the current value.

If optional \textit{pv\_path} arguments are specified, the mirror copies are deallocated from the specified physical volumes.

One, and only one, \(-l\), \(-L\), or \(-m\) option must be specified.

\(-k\)  
This option should be used only in the special instance when you want to reduce a mirrored logical volume on a physical volume that is missing or has failed.

Use this option with the \(-m\) option.

**EXTERNAL INFLUENCES**

**Environment Variables**

\texttt{LANG} determines the language in which messages are displayed.

If \texttt{LANG} is not specified or is null, it defaults to "C" (see \texttt{lang(5)}).

If any internationalization variable contains an invalid setting, all internationalization variables default to "C" (see \texttt{environ(5)}).

**EXAMPLES**

Decrease the number of the logical extents of a logical volume to one hundred:

```
  lvreduce -l 100 /dev/vg01/lvol3
```

Reduce to one mirror (that is, an original and one copy) for each logical extent of a logical volume:

```
  lvreduce -m 1 /dev/vg01/lvol5
```

Remove mirror copies of logical extents of a logical volume from the physical volume /dev/dsk/c1t0d0:

```
  lvreduce -m 0 /dev/vg01/lvol4 /dev/dsk/c1t0d0
```

Remove a logical volume from a one-way mirrored set on the specified \textit{pv\_path}.

```
  lvreduce -m 0 -k /dev/vg01/lvol1 /dev/dsk/c1t0d0
```

**WARNINGS**

LVM does not store any information about which physical extents within a logical volume contain useful data; therefore, reducing the space allocated to a logical volume without doing a prior backup of the data could lead to the loss of useful data. The \texttt{lvreduce} command on a logical volume containing a file system of greater length than the size being reduced to will cause data corruption.

To reduce a logical volume being used for swap, that swap area must not be currently in use.

**SEE ALSO**

\texttt{lvcreate(1M), lvdisplay(1M), lvextend(1M), pvchange(1M), pvdisplay(1M)}.
NAME
lvremove - remove one or more logical volumes from LVM volume group

SYNOPSIS
/usr/sbin/lvremove [-A autobackup] [-f] lv_path ...

Remarks
lvremove cannot be performed if the volume group is activated in shared mode.

DESCRIPTION
The lvremove command removes each logical volume specified by lv_path ....
Logical volumes must be closed before they can be removed. For example, if the logical volume contains a
file system, unmount the file system before removing it.

Options and Arguments
lvremove recognizes the following options and arguments:
  lv_path     The block device path name of a logical volume.
  -A autobackup Set automatic backup for this invocation of this command. autobackup can have
                 one of the following values:
                 y  Automatically back up configuration changes made to the logical
                     volume. This is the default.
                 After this command executes, the vgcfgbackup command (see
                     vgcfgbackup(1M)) is executed for the volume group to which the logi-
                     cal volume belongs.
                 n  Do not back up configuration changes this time.

                  -f Specify that no user confirmation is required.

EXTERNAL INFLUENCES
Environment Variables
LANG determines the language in which messages are displayed.
If LANG is not specified or is null, it defaults to "C" (see lang(5)).
If any internationalization variable contains an invalid setting, all internationalization variables default to
"C" (see environ(5)).

EXAMPLES
Remove a logical volume without requiring user confirmation:
  lvremove -f /dev/vg01/lvol5

WARNINGS
This command destroys all data in the specified logical volumes.

SEE ALSO
lvchange(1M), umount(1M).
NAME
lvrmboot - remove LVM logical volume link to root, primary swap, or dump volume

SYNOPSIS

Remarks
lvrmboot cannot be performed if the volume group is activated in shared mode.

DESCRIPTION
The lvrmboot command updates all physical volumes contained in the volume group vg_name such that the logical volume is removed as a root, primary swap, or dump volume when the system is next booted on the volume group.

Options and Arguments
lvrmboot recognizes the following options and arguments:

vg_name
The path name of the volume group.

- A autobackup
Set automatic backup for this invocation of this command. autobackup can have one of the following values:

   y Automatically back up configuration changes made to the logical volume. This is the default.
   After this command executes, the vgcfgbackup command (see vgcfgbackup(1M)) is executed for the volume group to which the logical volume belongs.

   n Do not back up configuration changes this time.

-d dump_lv
Remove the definition of dump_lv as one of the dump volumes. Update the Boot Data Reserved Area.

-r
Remove the definitions of all of the root, primary swap, and all dump volumes from the given volume group. Update the Boot Data Reserved Area.

-s
Remove the definition of the primary swap volume from the given volume group. Update the Boot Data Reserved Area.

-v
Print verbose messages.

EXTERNAL INFLUENCES
Environment Variables
LANG determines the language in which messages are displayed.

If LANG is not specified or is null, it defaults to "C" (see lang(5)).

If any internationalization variable contains an invalid setting, all internationalization variables default to "C" (see environ(5)).

EXAMPLES
Specify that the logical volume /dev/vg00/lvol3 should be removed as one of the dump logical volumes:

   lvrmboot -v -d lvol3 /dev/vg00

Specify that volume group /dev/vg00 should no longer be a root volume group. Primary swap and dump are also removed.

   lvrmboot -r /dev/vg00

SEE ALSO
lvlnboot(1M).
NAME
lvsplit - split mirrored LVM logical volume into two logical volumes

SYNOPSIS
/usr/sbin/lvsplit [-A autobackup] [-s suffix] [-g PhysicalVolumeGroup] lv_path ...

Remarks
This command requires the installation of the optional HP MirrorDisk/UX software (not included in the
standard HP-UX operating system) before it can be used.

lvsplit cannot be performed if the volume group is activated in shared mode.

DESCRIPTION
The lvsplit command splits a single- or double-mirrored logical volume, lv_path, into two logical
volumes. A second logical volume is created containing one copy of the data. The original logical volume is
appropriately reset as unmirrored or single-mirrored.

If the -s option is specified, the new logical volume name has the form lv_path suffix. If -s is not
specified, suffix defaults to b, as in lv_pathb.

If more than one lv_path is specified on the command line, lvsplit ensures that all logical volumes are
brought offline together in one system call, ensuring predictable results among the logical volumes. Up to
127 logical volumes can be specified on the command line. All logical volumes must belong to the same
volume group, and there must be enough unused logical volumes remaining in the volume group to hold the
newly split logical volumes. A volume group can contain up to 255 logical volumes.

If PhysicalVolumeGroup is specified, the offline logical volumes are created using the mirror copies on the
physical volumes contained in the specified physical volume group.

Whenever a mirrored logical volume is split into two logical volumes, a bit map is stored that keeps track of
all writes to either logical volume in the split pair. When the two logical volumes are subsequently merged
using lvmerge, the bit map is used to decide which areas of the logical volumes need to be resynchronized
(see lvmerge(1M)). This bit map remains in existence until one of the following conditions occurs:

- The merge is completed.
- One of the logical volumes is extended, reduced, or split again.
- The volume group is cross-activated to shared mode (see vgchange(1M)).
- The system is rebooted.

The new logical volume must be checked with the fsck command before it is mounted (see fsck(1M)).
lvsplit flushes the file system to a consistent state except for pipes and unlinked but open files.

To rejoin two split copies of a logical volume, use the lvmerge command (see lvmerge(1M)).

Options and Arguments
lvsplit recognizes the following options and arguments:

lv_path The block device path name of a logical volume. Up to 127 logical volumes in the
same volume group can be specified at one time.

-A autobackup Set automatic backup for invocation of this command. autobackup can have one
of the following values:

y Automatically back up configuration changes made to the logical volume.
This is the default.

After this command executes, the vgcfgbackup command (see vgcfgbackup(1M)) is executed for the volume group to which the logical
volume belongs.

n Do not back up configuration changes this time.

-g PhysicalVolumeGroup The offline logical volumes will be created using the mirror copies on the physical
volumes in the specified PhysicalVolumeGroup.
lvsplit(1M) lvsplit(1M)

(Requires Optional HP MirrorDisk/UX Software)

- s suffix  Specify the suffix to use to identify the new logical volume. The new logical
volume name has the form lv_path suffix. If -s is omitted, suffix defaults to b,
as in lv_pathb.

EXTERNAL INFLUENCES

Environment Variables

LANG determines the language in which messages are displayed.

If LANG is not specified or is null, it defaults to "C" (see lang(5)).

If any internationalization variable contains an invalid setting, all internationalization variables default to
"C" (see environ(5)).

EXAMPLES

Split the mirrored logical volume /dev/vg00/lvol1 into two copies. Call the new logical volume
/dev/vg00/lvol1backup:

    lvsplit -s backup /dev/vg00/lvol1

Split the mirrored logical volume /dev/vg00/lvol1 into two copies. The offline logical volume will be
created using the mirror copy on the physical volumes contain in the physical volume group pvg1.

    lvsplit -g pvg1 /dev/vg00/lvol1

Split an online logical volume which is currently mounted on /usr so that a backup can take place:

    lvsplit /dev/vg00/lvol1
    fsck /dev/vg00/lvol1b
    mount /dev/vg00/lvol1b /usr.backup

Perform a backup operation, then:

    umount /usr.backup
    lvmerge /dev/vg00/lvol1b /dev/vg00/lvol1

Split two logical volumes at the same time:

    lvsplit /dev/vg01/database1 /dev/vg01/database2

Perform operation on split logical volumes, then rejoin them:

    lvmerge /dev/vg01/database1b /dev/vg01/database1
    lvmerge /dev/vg01/database2b /dev/vg01/database1

WARNINGS

After a two-way mirrored logical volume has been split once, it cannot be split again without merging the
logical volumes using the lvmerge command (see lvmerge(1M)).

SEE ALSO

lvcreate(1M), lvextend(1M), lvmerge(1M).
lvsync(1M)

(Requires Optional HP MirrorDisk/UX Software)

NAME
lvsync - synchronize stale mirrors in LVM logical volumes

SYNOPSIS
/usr/sbin/lvsync [-T] lv_path ...

Remarks
This command requires the installation of the optional HP MirrorDisk/UX software (not included in the standard HP-UX operating system) before it can be used.

DESCRIPTION
The lvsync command synchronizes the physical extents of each logical volume specified by lv_path. Synchronization occurs only on physical extents that are stale mirrors of the original logical extent. The synchronization process can be time consuming, depending on the hardware characteristics and the amount of data.

Arguments
lvsync recognizes the following argument:

lv_path     The block device path name of a mirrored logical volume.
-T          Perform mirror synchronization of logical volumes within a volume group using multiple parallel threads. Logical volumes belonging to different volume groups will be synchronized serially. It is possible that logical volumes start and/or complete their synchronization in a different order than specified on the command line.

The maximum number of threads used can be controlled using the PTHREAD_THREADS_MAX system tunable.

NOTE: This option has no effect if the volume group is activated in shared mode.

EXTERNAL INFLUENCES
Environment Variables
LANG determines the language in which messages are displayed.
If LANG is not specified or is null, it defaults to "C" (see lang(5)).
If any internationalization variable contains an invalid setting, all internationalization variables default to "C" (see environ(5)).

EXAMPLES
Synchronize the mirrors on a logical volume:

lvsync /dev/vg01/lvol5

Synchronize the mirrors on three logical volumes using multiple parallel threads:

lvsync -T /dev/vg02/lvol1 /dev/vg02/lvol2 /dev/vg02/lvol3

Add mirror copies using lvextend with the -s option for three logical volumes, and then synchronize mirrors on these three logical volumes using multiple parallel threads.

First, add mirror copies to each logical volume using lvextend with the -s option:

lvextend -m 1 -s /dev/vg03/lvol1
lvextend -m 2 -s /dev/vg03/lvol2
lvextend -m 2 -s /dev/vg03/lvol3

Now synchronize the mirrors on these three logical volumes using multiple parallel threads:

lvsync -T /dev/vg03/lvol1 /dev/vg03/lvol2 /dev/vg03/lvol3

SEE ALSO
lvdisplay(1M), lvextend(1M), vgsync(1M).
NAME
lwresd - lightweight resolver daemon

SYNOPSIS
[-t directory] [-u user-id]

DESCRIPTION
lwresd is the daemon providing name lookup services to clients that use the BIND 9 lightweight resolver
library. It is essentially a stripped-down, caching-only name server that answers queries using the BIND 9
lightweight resolver protocol rather than the DNS protocol.

lwresd listens for resolver queries on a UDP port on the IPv4 loopback interface, 127.0.0.1. This means
that lwresd can only be used by processes running on the local machine. By default, UDP port number
921 is used for lightweight resolver requests and responses.

Incoming lightweight resolver requests are decoded by lwresd which then resolves them using the DNS
protocol. When the DNS lookup completes, lwresd encodes the answers from the name servers in the
lightweight resolver format and returns them to the client that made the original request.

If /etc/resolv.conf contains any nameserver entries, lwresd sends recursive DNS queries to those
servers. This is similar to the use of forwarders in a caching name server. If no nameserver entries are
present, or if forwarding fails, lwresd resolves the queries autonomously starting at the root name
servers, using a compiled-in list of root servers hints.

Options
-C config-file
   This option is used to use config-file as the configuration file instead of the default, /etc/resolv.conf.

-d debuglevel
   This option is used to set the daemon's debug level to debuglevel. Debugging traces from
   lwresd become more verbose as the debug level increases.

-f
   This option is used to run lwresd in the foreground.

-g
   This option is used to run lwresd in the foreground and force all logging to stderr.

-i pid-file
   This option is used to write the daemon's process id to pid-file instead of the default path-
   name.

-n #cpus
   This option is used to create #cpus worker threads to take advantage of multiple CPUs. If
   no option is given, lwresd will try to determine the number of CPUs present and create one
   thread per CPU. If lwresd is unable to determine the number of CPUs, a single
   worker thread is created.

-P query-port#
   This option is used to send DNS lookups to port number query-port# when querying name
   servers. This provides a way of testing the lightweight resolver daemon with a name
   server that listens for queries on a non-standard port number.

-p port#
   This option is used to listen for lightweight resolver queries on the loopback interface using
   UDP port port# instead of the default port number, 921.

-s
   This option is used to write memory usage statistics to stdout on exit. This option is only
   of interest to BIND 9 developers and may be removed or changed in a future release.

-t directory
   This option tells lwresd to chroot() to directory immediately after reading its
   configuration file.

-u user-id
   This option is used to run lwresd as user-id, which is a user name or numeric id that
   must be present in the password file. The lightweight resolver daemon will change its
   user-id after it has carried out any privileged operations, such as writing the process-id file
   or binding a socket to a privileged port (typically any port less than 1024).
lwresd is a daemon for lightweight resolvers, not a lightweight daemon for resolvers.

FILES
/etc/lwresd.conf  Default configuration file
/var/run/lwresd.pid  Default process-id file

SEE ALSO
named(1M).
makedbm(1M)
makedbm(1M)

NAME
makedbm - make a Network Information System database

SYNOPSIS
/usr/sbin/makedbm [-b] [-l] [-s] [-i nis_input_file] [-o nis_output_name]
[-d nis_domain_name] [-m nis_master_name] infile outfile
/usr/sbin/makedbm -u database_name

Remarks
The Network Information Service (NIS) was formerly known as Yellow Pages (yp). Although the name has changed, the functionality of the service remains the same.

DESCRIPTION
makedbm generates databases (maps) for the Network Information System (NIS) from infile. A database created by makedbm consists of two files: outfile.pag and outfile.dir. A makedbm database contains records called dbm records composed of key-value pairs.

Each line of infile is converted to a single dbm record; all characters up to the first tab or space form the key, and the remainder of the line is the value. If a value read from infile ends with \, the value for that record is continued onto the next line. The NIS clients must interpret the # character (which means that makedbm does not treat the # as if it precedes a comment). If infile is a hyphen (-), makedbm reads standard input.

makedbm always generates a special dbm record with the key YP_LAST_MODIFIED, whose value is the time of last modification of infile (or the current time, if infile is -). This value is also known as the order number of a map, and yppoll prints it for a specified NIS map (see yppoll(1M)).

Another special dbm record created by makedbm has the key YP_MASTER_NAME. Its value is usually the host name retrieved by gethostname(); however, the -m option can be used to specify a different value (see gethostname(2)).

If the -b option is used, another special dbm record with the key YP_INTERDOMAIN key is created. When this key exists in the NIS host.by* maps and the NIS host name resolution fails, the ypserv process will query the Internet domain name server, named(1M), to provide the host name resolution. Before using the -b option, it is recommended that the name services switch, switch(4), be set to allow NIS host name resolution first. (Note that, since the ypserv process only checks hosts.byname and hosts.byaddr for the existence of the YP_INTERDOMAIN key, using the -b option on any other NIS map will have no effect. Also, the -b option should be used on both the hosts.byname and hosts.byaddr maps, not one exclusively.)

If the -s option is used, another special dbm record created is the YP_SECURE key. If this key exists in an NIS map, ypserv will only allow privileged processes (applications that can create reserved ports) to access the data within the map.

Options
makedbm recognizes the following options and command-line arguments.

-b Create a special dbm record with the key YP_INTERDOMAIN. This key, which is in the hosts.byname and hosts.byaddr maps, allows the ypserv process to query the Internet domain name server, (see named(1M)).

-l Convert the keys of the given map to lowercase. This command option allows host name matches to work independent of character-case distinctions.

-s Accept connections from secure NIS networks only.

-i Create a special dbm record with the key YP_INPUT_FILE and the value nis_input_file. If the -s option is used, another special dbm record created is the YP_SECURE key. If this key exists in an NIS map, ypserv will only allow privileged processes to access the data within the map. (i.e. applications that can create reserved ports.)

-o Create a special dbm record with the key YP_OUTPUT_NAME and the value nis_output_name.

-d Create a special dbm record with the key YP_DOMAIN_NAME and the value nis_domain_name.

-m Replace the value of the special dbm record whose key is YP_MASTER_NAME with nis_master_name.
-u  Undo the database_name (i.e., write the contents of database_name to the standard output), one dbm record per line. A single space separates each key from its value.

**EXAMPLES**

Shell scripts can be written to convert ASCII files such as /etc/netgroup to the key-value form used by makedbm. For example,

```
#!/usr/bin/sh
/usr/bin/awk 'BEGIN { FS = ":" } { print $1, $0 }' \
/etc/netgroup | \ 
  makedbm - netgroup
```

converts the file /etc/netgroup to a form that is read by makedbm to make the NIS map netgroup. The keys in the database are netgroup(4) names, and the values are the remainders of the lines in the /etc/netgroup file.

**AUTHOR**

makedbm was developed by Sun Microsystems, Inc.

**SEE ALSO**

domainname(1), ypinit(1M), ypmake(1M), yppoll(1M), gethostname(2), netgroup(4), ypfiles(4).
makemap(1M) makemap(1M)

NAME
makemap - creates database maps for sendmail

SYNOPSIS

DESCRIPTION
makemap creates the database maps used by the keyed map lookups in sendmail(1M). It reads input from the standard input and outputs them to the indicated mapname.

makemap handles up to three different database formats, selected using the maptype parameter. They may be:

- dbm   DBM format maps. (.pag,.dir)
- btree B-Tree format maps. (.db)
- hash  Hash format maps. (.db)

In all cases, makemap reads lines from the standard input consisting of two words separated by white space. The first is the database key, the second is the value. The value may contain %n strings to indicated parameter substitution. Literal parentheses should be doubled (%%). Blank lines and lines beginning with pound sign (#) are ignored.

Flags
-C file Use the specified sendmail configuration file to look up the TrustedUser option.
-N Include the null byte that terminates strings in the map. This must match the -N flag in the sendmail.cf K line.
-d Allow duplicate keys in the map. This is only allowed on B-Tree format maps. If two identical keys are read, they will both be inserted into the map.
-e Allow empty value (right hand side).
-f Normally all upper case letters in the key are folded to lower case. This flag disables that behaviour. This is intended to mesh with the -f flag in the K line in sendmail.cf. The value is never case folded.
-n Create NIS compatible alias database.
-o Append to an old file. This allows you to augment an existing file.
-r Allow replacement of existing keys. Normally makemap complains if you repeat a key, and does not do the insert.
-s Ignore safety checks on maps being created. This includes checking for hard or symbolic links in world writable directories.
-u Dump (unmap) the content of the database to standard output.
-v Verbosely print what it is doing.

SEE ALSO
sendmail(1M).

HISTORY
The makemap command appeared in 4.4BSD.
map-mbone(1M)

NAME
map-mbone - Multicast Router Connection Mapper

SYNOPSIS
/usr/sbin/map-mbone [-d debuglevel] [-f] [-g] [-n] [-r retries] [-t timeout]
[multicast-router]

DESCRIPTION
map-mbone requests the multicast router connection information from the multicast-router, and prints the information to the standard out. map-mbone sends out the ASK_NEIGHBORS igmp message to the multicast-router. When the multicast-router receives the request, it sends back its configuration information. multicast-router can be either an ip address or a system name.

If the multicast-router is not specified, flood mode is on by default and the igmp request message is sent to all the multicast router on the local network. With flood mode on, when map-mbone finds new neighbor routers from the replies, it will send the same igmp request to the new neighbor routers. This activity continues until no new neighbor routers are reported in the replies.

The command line options are:
- `d debuglevel` Sets the level for printing out the debug message. The default is 0, which prints only error and warning messages. Debug level three prints most the messages.
- `r retries` Sets the retry times to poll the routing daemon for information. The default is 1.
- `t timeout` It specifies the timeout value in seconds for waiting the reply. The default value is 2 seconds.
- `f` Sets the flood mode on. It is the default value when no multicast-router is given on the command line input.
- `g` Generates output in GRaphEd format.
- `n` Disable DNS lookup for the multicast router names.

The output contains the interface configuration information of the requested router(s). The format for each interface output is:

`interface_addr -> neighbor_addr (neighbor_name) [metrics/thresh/flags]`

If there are multiple neighbor routers on one interface, they will all be reported. The neighbor_name will not be printed if the -n option is specified on the command line.

The possible values for flags are:
- `tunnel` Neighbors are reached via tunnel.
- `srcrt` The tunnel uses IP source routing.
- `down` The interface is down.
- `disabled` The interface is administratively disabled for multicast routing.
- `querier` The local router is the querier of the subnet.

The format of the GRaphEd output is:

`interface_addr_in_integer ($ NP low_byte_addr high_byte_addr) node_name
[ neighbor_addr_in_integer metrics/threshold/flags ]`

If there is no neighbor router on an interface, then a * will be put next to the node_name. If there are multiple neighbor routers on one interface, all of them will be reported. The possible values for flags are:
- `E` The neighbor is reached via tunnel.
- `P` The neighbor is on the same network/subnet.
- `D` The interface is down.

Please see mrouted(1M) for metrics and thresh.

EXAMPLES
Querying `camden.cup.hp.com` for the multicast router connection information.
map-mbone(1M)

map-mbone hpntclt.cup.hp.com

127.0.0.1 (localhost) [version 3.3]:
  193.2.1.39 -> 0.0.0.0 (all-zeros-broadcast) [1/1/disabled]
  15.13.106.144 -> 15.255.176.33 (matmos.hpl.hp.com) [10/1/tunnel]
  15.13.106.144 -> 15.17.20.7 (hpspddc.vid.hp.com) [10/1/tunnel/down]

Querying hpntcbs.cup.hp.com for multicast router connectivity with -g option:

map-mbone -g hpntcbs.cup.hp.com

GRAPH "Multicast Router Connectivity: Wed Feb 1 17:34:59 1995"=UNDIRECTED
  252537488 ($ NP 1440 1060 $) "hpntclt.cup.hp.com"
    ;
  252538974 ($ NP 940 1120 $) "hpntcbs.cup.hp.com"
    252537488 "10/1E"
    252539807 "1/1P"
    ;
  252539807 ($ NP 1590 1150 $) "hpntc1h.cup.hp.com"
    ;

Note
map-mbone must be run as root.

AUTHOR
map-mbone was developed by Pavel Curtis.

SEE ALSO
mrouted (1M), mrinfo(1M).
NAME
mc - media changer manipulation utility

SYNOPSIS
mc [-p device] [-a num] [-q] [-c <src_element_type><dest_element_type>]
mc [-p device] [-b 0|1] [-l 0|1] [-e element_type]
mc [-p device] [-n drive_number] [-r element_type]
mc [-p device] -s <element_type><num> -d <element_type><num>
mc [-h | -?]

DESCRIPTION
The mc utility provides users with a command-line interface to send media manipulation commands to an
autoloader or media changer device. It takes "element types" as arguments to most of the options. The
valid element types (element_types) are:

D Specifies a Data Transfer (DT) element.
I Specifies an Import/Export (IE) element.
M Specifies a Medium Transport (MT) element.
S Specifies a Storage (ST) element.

An example of a Data Transfer element is the embedded tape drive(s) of the autoloader. An example of an
Import/Export element is the slot(s) by which an item of the media may be inserted or removed from the
autoloader. An example of a Medium Transport element is the robotic picker assembly of the autoloader.
An example of a Storage element is the media slot(s) of the autoloader.

Please see examples below for usage.

Options
mc recognizes the following options and arguments:

-a num Prints the SCSI bus address of the drive slot specified by num.
-b 0|1 Issue a Reserve (1) or Release (0) SCSI command to the device.
-c <src_element_type><dest_element_type> Determines whether a move from source to destination is valid. Uses device capabilities
    mode page and will return TRUE or FALSE. There should be no spaces in the source and
destination element type values. For example, -c DS specifies a Data Transfer element as
    the source and a Storage element as the destination.
-e element_type Prints out the number of elements of element type. See element types above. Multiple
types can be specified. For example, -e IDSN specifies all the valid element types.
-h | -? Prints out usage description.
-l 0|1 Allow (0) or prevent (1) media removal.
-n num Prints the serial number of the drive in the slot specified by num (if information is avail-
able).
-p device Specifies the pass-through device file to the library device. If the LIBRARY environment
    variable is set then the -p option may be omitted.
-q Prints out Vendor ID, Product ID and Product Rev standard inquiry information.
-r element_type Prints out the status (FULL/EMPTY/NONE) of element slots of element type(s). See ele-
    ment types above. If barcode information is supported by the device, this will be appended
to the status. Multiple types can be specified. For example, -r IDSN specifies all the
valid element types.
-s <element_type><num> Specifies the element type and slot number (<num>) for the move medium source. There
    should be no space between the element type and the slot number. For example, -sS1
    specifies a Storage element in slot number 1. This option cannot be specified more than
    5 elements.
mc(1M)       mc(1M)

 twice per invocation.

 -d <element_type><num>
  Specifies the element type and slot number for the move medium destination. There
  should be no space between the element type and the slot number. For example, -dD3
  specifies a Data Transfer element in slot number 3. This option cannot be specified more
  than twice per invocation.

RETURN VALUE
  mc returns 0 upon successful completion and -1 otherwise.

DIAGNOSTICS
  ERROR: 0x5 Illegal Request: 0x3b0d Medium Destination element full
  The above error message could be a result of the command mc -s S2 -d D1 that was used to
  move media to an embedded drive that is already full.

  ERROR: /dev/scsi/3: No such file or directory
  If the default SCSI pass-through device file does not exist and no other device file is specified, then the
  above error message will be printed.

EXAMPLES
  Using a DDS-2 autoloader with a six-cartridge magazine as an example:
  To see the status of the autoloader’s Data Transfer and Storage element types:
  mc -r DS

  The following shows an example of output from the above command. The output indicates that there is an
  item of media in slot 2 (ST_slot_2), an item of media in the embedded drive (DT_slot_1), and all the other
  slots are empty.

  DT_slot_1 FULL
  ST_slot_1 EMPTY
  ST_slot_2 FULL
  ST_slot_3 EMPTY
  ST_slot_4 EMPTY
  ST_slot_5 EMPTY
  ST_slot_6 EMPTY

  To move media from an embedded drive to slot 5 and then move media from slot 2 to an embedded drive:
  mc -s D1 -d S5 -s S2 -d D1

  To check if a move from a Data Transfer element to a Storage element is possible:
  mc -c DS

  The following shows an example of output from the above command. The output indicates that moves from
  Data Transfer element types to Storage element types are valid.

  DT->ST: TRUE

WARNINGS
  Note for all HP DDS autoloaders: After the mc command has been used for the first time, the autoloader
  will enter into random mode. Once in random mode, all front panel button features are disabled except for
  the Eject Button. To go back to stacker mode, the magazine must be ejected and then reinserted.

  Note for some autoloaders or libraries: The media must be unloaded using the mt offline
  command prior to using the mc command.

DEPENDENCIES
  The mc command supports all HP-supported devices that comply with the SCSI-2 Medium Changer com-
  mand specification.
A SCSI pass-through driver must be configured and the device file created before this command can be used to manipulate the autoloader.

For devices connected to SCSI interface cards (PCI), the `sctl` pass-through driver must be configured. See `scsi_ctl(7)`.

**AUTHOR**

`mc` was developed by Hewlett-Packard.

**FILES**

`/dev/scsi/3`  Default pass-through device file.

**SEE ALSO**

`mt(1), scsi(7), scsi_ctl(7)`.
NAME

mk_kernel - load a kernel configuration from a system file

SYNOPSIS

/usr/sbin/mk_kernel [-o path_name] [-s system_file] [-v]

DESCRIPTION

mk_kernel reads configuration information from the system file (see system(4)) and applies those changes to a kernel configuration (see kconfig(5)). mk_kernel performs a function similar to that of kconfig -i, but has some semantic differences in order to preserve compatibility with previous releases. mk_kernel will build a new kernel executable only if needed to effect the requested changes. mk_kernel may reuse existing kernel executables when possible.

Options

mk_kernel recognizes the following options.

-o path_name

Specify the target kernel path.

If the target kernel path specified is /stand/vmunix, changes are applied to the currently running kernel configuration. They will take effect immediately if possible; otherwise, they will take effect at next boot.

If the target kernel path specified is of the form /stand/config/vmunix, changes are applied to the saved kernel configuration named config. The changes will not take effect at next boot unless that saved kernel configuration was already marked for use at next boot.

If the target kernel name is a single word containing no slashes, that name is used as the name of a saved kernel configuration. The changes are applied to that saved kernel configuration. The changes will not take effect at next boot unless that saved kernel configuration was already marked for use at next boot.

If the target kernel path is not one of the forms specified above, then mk_kernel exits with an error. Other locations for kernel executable files are no longer supported.

If the -o option is not specified, the changes are applied to the kernel configuration called hpux_test. The changes will not take effect at next boot unless that kernel configuration was already marked for use at next boot.

-s system_file

Specify the HP-UX system description file. If this option is not specified, the system file /stand/system is used. For details see system(4).

-v

Verbose mode. (Has no effect; retained for backward compatibility.)

RETURN VALUE

mk_kernel returns 0 upon normal completion, and 1 if an error occurred.

DIAGNOSTICS

Messages and warnings are sent to stdout. Messages from kconfig(1M) and other commands are displayed when invoked from mk_kernel. Errors cause mk_kernel to halt immediately; warnings allow the program to continue.

EXAMPLES

mk_kernel -o /stand/vmunix

Uses the default HP-UX system description file /stand/system. The changes are applied to the currently running kernel configuration, marked to take effect at next boot.

mk_kernel -s /mnt/altsys/stand/system.new

Uses the system description file /mnt/altsys/stand/system.new.

mk_kernel -s /stand/system -o fred

Uses the system description file /stand/system. The changes are applied to the saved kernel configuration called fred.
**mk_kernel(1M)**

**WARNINGS**
Do not manually modify the contents of the kernel configuration directories.

**FILES**
/stand/system Default HP-UX system description file

**SEE ALSO**
kconfig(1M), system(4), kconfig(5).
mkboot(1M)

NAME
mkboot, rmboot - install, update or remove boot programs from disk

SYNOPSIS
/usr/sbin/mkboot [-b boot_file_path] [-c [-u] | -f | -h | -u] [-i included_lif_file]
/usr/sbin/mkboot [-a auto_file_string] [-v] device
/usr/sbin/rmboot device

DESCRIPTION
mkboot is used to install or update boot programs on the specified device file.

The position on device at which boot programs are installed depends on the disk layout of the device. mkboot examines device to discover the current layout and uses this as the default. If the disk is uniniti-alyzed, the default is LVM layout on PA-RISC and Whole Disk on Itanium(R)-based systems. The default can be overridden by the -l, -H, or -W options.

Boot programs are stored in the boot area in Logical Interchange Format (LIF), which is similar to a file system. For a device to be bootable, the LIF volume on that device must contain at least the ISL (the initial system loader) and HPUX (the HP-UX bootstrap utility) LIF files. If, in addition, the device is an LVM physical volume, the LABEL file must be present (see lvlnboot(1M)).

For the VERITAS Volume Manager (VxVM) layout on the Itanium-based system architecture, the only relevant LIF file is the LABEL file. All other LIF files are ignored. VxVM uses the LABEL file when the system boots to determine the location of the root, stand, swap, and dump volumes.

Options
mkboot recognizes the following options:

- **-a auto_file_string**
  If the -a option is specified, mkboot creates an autoexecute file AUTO on device, if none exists. mkboot deposits auto_file_string in that file. If this string contains spaces, it must be quoted so that it is a single parameter.

- **-b boot_file_path**
  If this option is given, boot programs in the pathname specified by boot_file_path are installed on the given device.

- **-c**
  If this option is specified, mkboot checks if the available space on device is sufficient for the boot programs. If the -i option is also specified, mkboot checks if each included_lif_file is present in the boot programs. If the -p option is specified, it checks if each preserved_lif_file is present on the device. If all these checks succeed, mkboot exits with a status code of 0. If any of these checks fail, mkboot exits with a status code of 1. If the verbose option is also selected, a message is also displayed on the standard output.

- **-e**
  Use Itanium-based system EFI layout. This option causes mkboot to copy EFI utilities from /usr/lib/efi to the EFI partition on the disk; see idisk(1M) and efi(4). This option is applicable only on Itanium-based machines; it may not be used on PA-RISC. (Use the -s option to specify a non-default source for EFI files.)

- **-f**
  This option forces the information contained in the boot programs to be placed on the specified device without regard to the current swapping status. Its intended use is to allow the boot area to grow without having to boot the system twice (see -h option).

  This option should only be used when the system is in the single user state.

  This could be a dangerous operation because swap space that is already allo-
  cated and possibly in use will be overwritten by the new boot program infor-
  mation. A message is also displayed to the standard output stating that the
  operator should immediately reboot the system to avoid system corruption
  and to reflect new information on the running system.

  A safer method for reapportioning space is to use the -h option.
This option is valid only if `device` has the Whole Disk layout.

`-h`  
Specifying this option shrinks the available space allocated to swap in the LIF header by the amount required to allow the installation of the new boot programs specified by `boot_file_path`.

After the LIF header has been modified, reboot the system to reflect the new swap space on the running system. At this point, the new boot programs can be installed and the system rebooted again to reflect the new boot programs on the running system. This is the safe method for accomplishing the capability of the `-f` option.

This option is valid only if `device` has the Whole Disk layout.

`-H`  
If this option is specified, `mkboot` treats `device` to be a Hard Partition layout disk. This option cannot be used along with the `-l` and `-W` options.

`-i included_lif_file`  
If the `-i` option is specified one or more times, `mkboot` copies each `included_lif_file` and ignores any other LIF files in the boot programs. The sole exceptions to this rule are the files ISL and HPUX, which are copied without regard to the `-i` options. If `included_lif_file` is also specified with the `-p` option, the `-i` option is ignored. If the `-i` option is used with `LABEL` as its argument and the file `LABEL` does not exist in the boot programs, and `device` is an LVM layout disk or the `-l` option is used, `mkboot` creates a minimal `LABEL` file on `device` which will permit the system to boot on `device`, possibly without swap or dump.

If the device is a disk with VxVM layout and the `-l` option is used, the `LABEL` file created by `mkboot` is not sufficient to permit the system to boot. To create a `LABEL` file for the VxVM layout, you must use the `vxvmboot` (1M) command after `mkboot` has been executed.

If `-l` is specified, `mkboot` treats `device` as a volume layout disk, regardless of whether or not it is currently set up as one. This option cannot be used along with the `-H` and `-W` options. Use the `-l` option for any volume manager, including the VERITAS Volume Manager (VxVM) as well as LVM.

If `LABEL` is specified as an argument to the `-p` option and `LABEL` does not exist on the `device`, and if the layout is LVM, `mkboot` creates a minimal `LABEL` file. In general, if `preserved_lif_file` is not on the `device`, `mkboot` fails. An exception to this condition is if the `preserved_lif_file` is `LABEL` and the layout is not LVM, in which case the `LABEL` file is ignored.

`-s efi_file_path`  
Fetch the EFI files to be copied from `efi_file_path` instead of from the default location `/usr/lib/efi`. The `-s` option is valid only with the `-e` option, which specifies the Itanium-based system EFI layout.

`-u`  
If `-u` is specified, `mkboot` uses the information contained in the LIF header to identify the location of the swap area, boot area, and raw I/O so that installation of the boot programs does not violate any user data.

Normally, the LIF header information is overwritten on each invocation of `mkboot`. This option is typically used with the `-W` option, to modify boot programs on a disk that is actively supporting swap and/or raw I/O.

`-v`  
If this option is specified, `mkboot` displays its actions, including the amount of swap space available on the specified device.

`-W`  
If this option is specified, `mkboot` treats `device` as a disk having the Whole Disk layout. This option cannot be used along with the `-l` and `-H` options. This option will also fail on a disk having large-file enabled HFS filesystem.

`device`  
Install the boot programs on the given device special file. The specified `device` can identify either a character-special or block-special device. However,
mkboot requires that both the block and character device special files be present. mkboot attempts to determine whether device is character or block special by examining the specified path name. For this reason, the complete path name must be supplied. If mkboot is unable to determine the corresponding device file, a message is written to the display, and mkboot exits.

rmboot removes the boot programs from the boot area.

EXAMPLES
Install default boot programs on the specified disk, treating it as an LVM disk:
```
mkboot -l /dev/dsk/c0t5d0
```
Use the existing layout, and install only SYSLIB and ODE files and preserve the EST file on the disk:
```
mkboot -i SYSLIB -i ODE -p EST /dev/rdsk/c0t5d0
```
Install only the SYSLIB file and retain the ODE file on the disk. Use the Whole Disk layout. Use the file /tmp/boottf to get the boot programs rather than the default. (The -i ODE option will be ignored):
```
mkboot -b /tmp/boottf -i SYSLIB -i ODE -p ODE -W /dev/rdsk/c0t5d0
```
Install EFI utilities to the EFI partition on an Itanium-based system, treating it as an LVM or VxVM disk:
```
mkboot -e -l /dev/dsk/c3t1d0
```
Create AUTO file with the string autofile command on a device. If the device is on an Itanium-based system, the file is created as /EFI/HPUX/AUTO in the EFI partition. If the device is on a PA-RISC system, the file is created as a LIF file in the boot area.
```
mkboot -a "autofile command" /dev/dsk/c2t0d0
```

WARNINGS
If device has a Whole Disk layout, a file system must reside on the device being modified.
When executing from a recovery system, the mkboot command (if used) must be invoked with the -f option; otherwise it will not be able to replace the boot area on your disk.
If device is, or is intended to become an LVM physical volume, device must specify the whole disk.
If device is, or is intended to become a Hard Partitioned disk, device must specify section 6.

DEPENDENCIES
mkboot and rmboot fail if file system type on device is not HFS.

LVM and Hard Partition Layouts
The -f, -h, and -u options are not supported.

AUTHOR
mkboot and rmboot were developed by HP.

FILES
/usr/lib/uxbootlf file containing default PA-RISC boot programs
/usr/lib/efi file containing default Itanium-based systePFm (EFI) boot programs
ISL initial system loader
HPUX HP-UX bootstrap and installation utility
AUTO defines default/automatic boot behavior (see hpux(1M))
LABEL used by LVM
RDB diagnostics tool
IOMAP diagnostics tool

SEE ALSO
boot(1M), hpux(1M), hpux.efi(1M), isl(1M), lvlnboot(1M), mkfs(1M), newfs(1M), vxvmboot(1M), efi(4), lif(4).
mkfs(1M)

NAME
mkfs - construct a file system (generic)

SYNOPSIS
/usr/sbin/mkfs [-F FStype] [-o specific_options] [-V] special [operands]

DESCRIPTION
The mkfs command creates a file system by writing on the special file special. operands are listed on file system specific manual pages (see SEE ALSO).

Options
mkfs recognizes the following options:

- **-F FStype** Specify the file system type on which to operate (see fstyp(1M) and fs_wrapper(5)). If this option is not included on the command line, then the file system type is determined from the file /etc/fstab by matching special with an entry in that file. If there is no entry in /etc/fstab, then the file system type is determined from the file /etc/default/fs.

- **-m** Display the command line that was used to create the file system. The file system must already exist. This option provides a means of determining the parameters used to construct the file system.

- **-o specific_options** Specify options specific to the file system type. specific_options is a list of suboptions and/or keyword/attribute pairs intended for an FStype-specific module of the command. See the file system specific manual entries for a description of the specific_options that are supported, if any.

- **-V** Echo the completed command line, but perform no other action. The command line is generated by incorporating the specified options and arguments with other information derived from /etc/fstab. This option allows the user to verify the command line.

EXAMPLES
Execute the mkfs command to create a 32MB HFS file system on /dev/dsk/c1t2d0:

```
mkfs -F hfs /dev/dsk/c1t2d0 32768
```

Execute the mkfs command on an HFS file system, /dev/dsk/c1t2d0, to recreate the command that was used to create the file system on /dev/dsk/c1t2d0:

```
mkfs -F hfs -m /dev/dsk/c1t2d0
```

AUTHOR
mkfs was developed by HP and the University of California, Berkeley.

FILES
/etc/default/fs Specifies the default file system type.
/etc/fstab Static information about the file systems.

SEE ALSO
chmod(1), bdf(1M), df(1M), fsadm(1M), fsck(1M), fstyp(1M), mkfs_hfs(1M), mkfs_vxfs(1M), newfs(1M), fstab(4), group(4), passwd(4), fs_wrapper(5).

STANDARDS CONFORMANCE
mkfs: SVID3
NAME
mkfs_hfs: mkfs - construct an HFS file system

SYNOPSIS
   [size [nsect ntrack blksize fragsize ncpg minfree rps nbpi]]
   special [proto [nsect ntrack blksize fragsize ncpg minfree rps nbpi]]
/usr/sbin/mkfs [-F hfs] [-m] [-V] special

Remarks
HFS file systems are normally created with the newfs command (see newfs_hfs(1M)).

DESCRIPTION
The mkfs command constructs an HFS file system by writing on the special file special. The mkfs command builds the file system with a root directory and a lost+found directory (see fsck_hfs(1M)). The FS_CLEAN magic number for the file system is stored in the superblock.

The mkfs command creates the file system with a rotational delay value of zero (see tunefs(1M)).

Options
mkfs recognizes the following options:

- `-F hfs`
  Specify the HFS file system type.
- `-d`
  This option allows the mkfs command to make the new file system in an ordinary file. In this case, special is the name of an existing file in which to create the file system. When this option is used, the size of the new file system cannot be defaulted. It must either be specified on the command line following special, or if a prototype file is being used, it must be the second token in the prototype file as usual.
- `-L|-S`
  There are two types of HFS file systems, distinguished mainly by directory formats that place different limits on the length of file names.
    - If `-L` is specified, build a long-file-name file system that allows directory entries (file names) to be up to MAXNAMLEN (255) bytes long.
    - If `-S` is specified, build a short-file-name file system that allows directory entries (file names) to be up to DIRSIZ (14) bytes long.
    - If neither `-L` nor `-S` is specified, build a file system of the same type as the root file system.
- `-m`
  Display the command line that was used to create the file system. The file system must already exist. This option provides a means to determine the parameters used to construct the file system.
- `-V`
  Echo the completed command line, but perform no other action. The command line is generated by incorporating the user-specified options and other information derived from /etc/fstab. This option allows the user to verify the command line.
- `-o specific_options`
  Specify a list of comma separated suboptions and/or keyword/attribute pairs from the list below.

`largefiles|nolargefiles`
Controls the largefile featurebit for the file system. The default is nolargefiles. This means the bit is not set, and files created on the file system will be limited to less than 2 gigabytes in size. If largefiles is specified, the bit is set and the maximum size for files created on the file system is not limited to 2 gigabytes (see mount_hfs(1M) and fsadm_hfs(1M)).

Arguments
mkfs recognizes the following arguments:

- `special`
  The file name of a special file.

One of the following arguments can be included after special:
size

The number of DEV_BSIZE blocks in the file system. DEV_BSIZE is defined in <sys/param.h>. The default value is the size of the entire disk or disk section minus any swap or boot space requested.

The size of HFS file systems are limited to 256GB-1 or 268,435,455 blocks.

proto

The name of a file that can be opened. The mkfs command assumes it is a prototype file and takes its directions from that file. See “Prototype File Structure” below.

The following optional arguments allow fine-tune control over file system parameters:

nsect

The number of sectors per track on the disk. The default value is 32 sectors per track.

nttrack

The number of tracks per cylinder on the disk. The default value is 16 tracks per cylinder.

blksize

The primary block size for files on the file system. Valid values are: 4096, 8192, 16384, 32768, and 65536. The default value is 8192 bytes.

fragsize

The fragment size for files on the file system. fragsize represents the smallest amount of disk space to be allocated to a file. It must be a power of two no smaller than DEV_BSIZE and no smaller than one-eighth of the file system block size. The default value is 1024 bytes.

ncpg

The number of disk cylinders per cylinder group. This number must be in the range 1 to 32. The default value is 16 cylinders per group.

minfree

The minimum percentage of free disk space allowed. The default value is 10 percent.

Once the file system capacity reaches this threshold, only users with appropriate privileges can allocate disk blocks.

rps

The number of disk revolutions per second. The default value is 60 revolutions per second.

nbpi

The density of inodes in the file system specified as the number of bytes per inode. The default value is 6144 bytes per inode.

This number should reflect the expected average size of files in the file system. If fewer inodes are desired, a larger number should be used; if more inodes are desired, a smaller number should be used.

Note: The number of inodes that will be created in each cylinder group of a file system is approximately the size of the cylinder group divided by the number of bytes per inode, up to a limit of 2048 inodes per cylinder group. If the size of the cylinder group is large enough to reach this limit, the default number of bytes per inode will be increased.

Prototype File Structure

A prototype file describes the initial file structure of a new file system. The file contains tokens separated by spaces or newline characters. It cannot contain comments.

The first token is the name of a file to be copied onto block zero as the bootstrap program (usually /etc/BOOT). If the file name is "", no bootstrap code is placed on the device. The second token is a number specifying the number of DEV_BSIZE blocks in the file system.

The next three tokens specify the mode, user ID, and group ID of the root directory of the new file system, followed by the initial contents of the root directory in the format described for a directory file below, and terminated with a $ token.

A file specification consists of four tokens giving the name, mode, user ID, and group ID, and an initial contents field. The syntax of the initial contents field depends on the mode.

A name token is a file name that is valid for the file system. The root directory does not have a name token.

A mode token is a 6-character string. The first character specifies the type of the file. It can be one of the following characters:

- Regular file
- b Block special file

The second character of a mode token is either u or - to specify set-user-ID mode or not. The third character of a mode token is either g or - to specify the set-group-ID mode or not. The rest of a mode token is a three-digit octal number giving the owner, group, and other read, write, and execute permissions (see chmod(1)).

The user-ID and group-ID tokens define the owner of the file. These values can be specified numerically or with symbolic names that appear in the current password and group databases.

**Regular file.** The initial contents field is the path name of an existing file in the current file system whose contents and size are copied to the new file.

**Block or character special file.** The initial contents field is two numeric tokens that specify the major and minor device numbers.

**Directory file.** The initial contents field is a list of file specifications for the entries in the directory. The list is terminated with a $ token. Directories can be nested. For each directory, the mkfs command automatically makes the . and .. entries.

**Symbolic link.** The initial contents field is a path name that is used as the path to which the symbolic link should point.

**Hard link.** The initial contents field is a path name that is used as the name of a file within the new file system to which the entry should be linked. The mode, user-ID and group-ID tokens of this entry are ignored; they are taken from the target of the link. The target of the link must be listed before the entry specifying the link. Hard links to directories are not permitted.

With the exception of the permissions field of the mode token (which is always an octal number), all numeric fields can be specified in hexadecimal (using a leading 0x), octal (using a leading 0), or decimal.

Here is a sample prototype specification. The indentation clarifies the directory recursion.

```
/etc/BOOT
12288
d--555 bin bin
sbin d--755 bin bin
   init ---555 bin bin /sbin/init
   savecore ---555 bin bin /sbin/savecore
   $
dev d--555 bin bin
   b0 b--640 root sys 0 0x0e0000
   c0 c--640 root sys 4 0x0e0000
   $
/etc d--755 bin bin
   init l--777 bin bin /sbin/init
   passwd ---444 bin bin /etc/passwd
   group ---444 bin bin /etc/group
   $
usr d--755 bin bin
   bin d--755 bin bin
      sh ---555 bin bin /usr/bin/sh
      rsh L--555 bin bin /usr/bin/sh
      su -u-555 root bin /usr/bin/su
      mailq l--777 bin bin /usr/sbin/sendmail
      $
      sbin d--755 bin bin
         sendmail -ug555 root mail /usr/sbin/sendmail
         $
   $
   $```
Access Control Lists
Every file with one or more optional ACL entries consumes an extra (continuation) inode. If you anticipate significant use of ACLs on a new file system, you can allocate more inodes by reducing the value of nbpi appropriately. The small default value typically causes allocation of many more inodes than are actually necessary, even with ACLs. To evaluate your need for extra inodes, run the `bdf -i` command on existing file systems. For more information on access control lists, see `acl(5)`.

EXAMPLES
Execute the `mkfs` command to create a 32MB HFS file system on the non-LVM disk `/dev/dsk/c1t2d0`:

```
mkfs -F hfs /dev/dsk/c1t2d0 32768
```

Display the command that was used to construct the file system on `/dev/dsk/c1t2d0`:

```
mkfs -F hfs -m /dev/dsk/c1t2d0
```

Create an HFS file system within a logical volume `/dev/vg01/my_lvol` of a size equal to the size of `my_lvol`:

```
mkfs -F hfs /dev/vg01/my_lvol
```

WARNINGS
The old `-F` option, from prior releases of `mkfs(1M)`, is no longer supported.

`mkfs_hfs(1M)` cannot be executed specifying creation of a file system on a whole disk if that disk was previously used as an LVM disk. If you wish to do this, use `mediainit(1)` to reinitialize the disk first.

The `-o largefile` option should be used with care, since older applications will not react correctly when confronted with large files.

AUTHOR
`mkfs` was developed by HP and the University of California, Berkeley.

FILES
`/var/adm/sbtab` List of locations of the superblocks for the created file system. The `mkfs` command appends entries to this file.

SEE ALSO
`chmod(1), bdf(1M), df(1M), fsadm_hfs(1M), fsck(1M), fsck_hfs(1M), fsck_clean(1M), mkfs(1M), mount_hfs(1M), newfs(1M), newfs_hfs(1M), dir(4), fstab(4), group(4), passwd(4), symlink(4), acl(5)`.

STANDARDS CONFORMANCE
`mkfs`: SVID3
NAME
  mkfs_vxfs: mkfs - construct a VxFS file system

SYNOPSIS
  mkfs [-F vxfs] [-V] [-m] special
  mkfs [-F vxfs] [-V] [-o N] [X] [bsize=bsize] [inosize=n] [largefiles|nolargefiles]
  [logsize=n] [ninode=n] [version=n] ] special size

DESCRIPTION
  mkfs creates a VxFS file system by writing on the special device file, unless either
  the -o N or -m option is specified. special must be the first argument on the command line after the options are given. The file
  system is created based on the options and size specified on the command line. The numeric size specifies
  the number of sectors in the file system. By default, size is specified in units of DEV_BSIZE sectors
  (currently, 1024 bytes). If size is not specified, mkfs determines the size of the special device. size can also
  be specified with a suffix to indicate a unit of measure other than sectors.

  mkfs builds a file system with a root directory and a lost+found directory (see fsck_vxfs(1M)). The file
  system can have disk layout Version 4 or 5. Version 4 layout adds support for Access Control Lists. You
  can choose the disk layout version with the version=n option (see below).

  The number of inodes allocated to a system depends on the disk layout version. Inode allocation is done
dynamically. There are a minimum number of inodes allocated to the file system by mkfs, and any other
inode allocations are done on an as-needed basis during file system use.

Options
  mkfs recognizes the following options:

  -F vxfs
  Specify the VxFS file system type.

  -m
  Display the command line which was used to create the file system. The file system must
  already exist.

  -V
  Echo the completed command line, but do not execute the command. The command line is
  generated by incorporating the user-specified options and other information derived from
  /etc/fstab. This option allows the user to verify the command line.

  -o specific_options
  Specify options specific to the VxFS file system type. specific_options is a comma separated
  list of suboptions and/or keyword/attribute pairs. The arguments aufirst, aupad, ausize, and nau, are no longer supported.

  The following specific_options are valid on a VxFS file system:

  N  Do not write the file system to the special file. This option gives all the information
      needed to create a file system but does not create it.

  X  Create a file system in a file. Used for debugging only.

  bsize=bsize
  bsize is the block size for files on the file system and represents the smallest amount of
  disk space allocated to a file. The value of bsize must be based on a power of 2
  selected from the range 1024 bytes to 8192 bytes. The default is 1024 bytes for file
  systems smaller than two terabytes. For Version 5 disk layout file systems larger
  than two terabytes, the maximum file system size is dependent on the block size. See
  the "VxFS Version 5 Disk Layout File System Sizes" topic in this section for more
  information. If bsize is not specified, the block size defaults are set to the appropri-
  ate value, based on size, when a file system is created.

  inosize=n
  n is the on-disk inode structure size for files on the file system. The valid values are
  256 and 512 bytes. The default is 256. There is usually no reason to increase the
  inode size, and not using the default value can adversely affect file system perform-
  ance.

  largefiles|nolargefiles
  Controls the largefiles flag for the file system. If largefiles is specified, the bit is
set and files two gigabytes or larger can be created. If `nolargefiles` is specified, the bit is cleared and files created on the file system are limited to less than two gigabytes. The default is `nolargefiles`. See `fsadm_vxfs(1M)` and `mkfs(1M)`.

If `largefiles` is not specified, and `inosize` is 256 the maximum number of inodes on the file system is approximately 8,380,000. If `largefiles` is not specified, and `inosize` is 512, the maximum number of inodes on the file system is approximately 4,190,000.

**Note:** Large files are supported on HP-UX 10.20 systems and above. Be careful when implementing large file system capability. System administration utilities such as backup may not operate correctly if they are not large-file aware.

`logsize=n`  

`n` is the number of file system blocks to allocate for an activity logging area. The minimum value for the Version 4 and Version 5 disk layout is the number of blocks that make the log no less than 256K. The maximum value for `n` is the number of blocks that make the log no greater than 16384K. The default logsize is 16 megabytes for file systems that are 512 megabytes or larger. For a small file system the default may be smaller to avoid wasting space.

A large log provides better performance on metadata-intensive workloads. A small log uses less space on the disk and leaves more room for file data. For example, an NFS-intensive workload performs better with a large log; a small floppy device requires a small log.

**Note:** The amount of virtual memory required by `fsck` (see `fsck_vxfs(1M)`) to check a VxFS file system is proportional to the size of the log. The maximum amount of virtual memory used is twice the size of the log. Therefore, the sum of physical memory and swap space must be at least 32 MB to ensure that a file system with a 16384K log can be cleaned. On small systems, take care not to create a file system with a log larger than half the available swap space. A maximum log size of one third the total of memory and swap space is a good rule of thumb (see `swapinfo(1M)`).

`ninode=n`  

`n` is the maximum number of inodes in the file system. The actual maximum number of inodes is `n` rounded up to an appropriate boundary. The digit 0 and the string `unlimited` both mean that the number of inodes is unlimited. The default is `unlimited`.

`version=n`  

`n` is the VxFS disk layout version number. Valid values are 4 and 5. The default is disk layout Version 5. The Version 5 disk layout supports file systems up to 32 terabytes in size.

**Operands**

`mkfs` recognizes the following operands:

- `special` Name of the special device file for the VxFS file system.
- `size` The number of sectors in the VxFS file system. See the "VxFS Version 5 Disk Layout File System Sizes" topic in this section for more information.

You can specify `size` with a suffix to indicate a unit of measure other than sectors. Append `k` or `K` to indicate the value is in kilobytes, `m` or `M` to indicate megabytes, or `g` or `G` to indicate gigabytes. An appended letter can be separated from the number by a space. In that case, enclose the letter and number in a set of quotes, for example:

"512 k"

**VxFS Version 5 Disk Layout File System Sizes**  
The Version 5 disk layout supports file systems up to 32 terabytes. The maximum size of the file system you can create depends on the block size. (The actual maximum file system size is slightly less than the maximum values in bytes, described below.)
The following table defines the file system sizes and their associated values:

<table>
<thead>
<tr>
<th>Block Size</th>
<th>In sectors</th>
<th>In bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1024 bytes</td>
<td>4,294,967,039</td>
<td>approximately 4 TB</td>
</tr>
<tr>
<td>2048 bytes</td>
<td>8,589,934,078</td>
<td>approximately 8 TB</td>
</tr>
<tr>
<td>4096 bytes</td>
<td>17,179,868,156</td>
<td>approximately 16 TB</td>
</tr>
<tr>
<td>8192 bytes</td>
<td>34,359,736,312</td>
<td>approximately 32 TB</td>
</tr>
</tbody>
</table>

Note: Sector size (in bytes) is specified by the DEV_BSIZE system parameter.

RETURN VALUE
mkfs returns the following value:

0  Successful completion.
1  Command failed due to syntax error.
32 Other error occurred.

EXAMPLES
Use mkfs to create a VxFS file system on /dev/rdsk/c0t6d0:
```
mkfs -F vxfs /dev/rdsk/c0t6d0 1024
```

Use mkfs to determine the command that was used to create the VxFS file system on /dev/rdsk/c0t6d0:
```
mkfs -F vxfs -m /dev/rdsk/c0t6d0
```

Create a VxFS file system on /dev/vgqa/lvol1, with a Version 4 disk layout and largefiles capability:
```
mkfs -F vxfs -o version=4,largefiles /dev/vgqa/lvol1
```

WARNINGS
If you want to reuse a special device that was previously used by LVM, you must first wipe out all the LVM information remaining on the disk. Use pvremove(1M) to remove the LVM information before executing mkfs_vxfs(1M). (You can also remove the LVM information by initializing the device with mediainit(1), but that is slower.)

The -o largefiles option should be used with care, since older applications do not react correctly when confronted with large files.

SEE ALSO
df_vxfs(1M), fsadm_vxfs(1M), fsck_vxfs(1M), mkfs(1M), mount_vxfs(1M), newfs_vxfs(1M), swapinfo(1M),
dir(4), fs_vxfs(4).

Veritas Volume Manager Administrator's Guide

STANDARDS CONFORMANCE
mkfs : SVID3
NAME
mklost+found - make a lost+found directory for fsck(1M)

SYNOPSIS
/usr/sbin/mklost+found

DESCRIPTION
The mklost+found command creates a directory named lost+found in the current directory. It also creates several empty files which are then removed to provide empty slots for the fsck command (see fsck(1M)).

For an HFS file system, the mklost+found command is not normally needed since the mkfs command automatically creates the lost+found directory when a new file system is created (see mkfs(1M)).

AUTHOR
mklost+found was developed by the University of California, Berkeley.

SEE ALSO
fsck(1M), mkfs(1M).
mknod(1M)

NAME
mknod - create special files

SYNOPSIS
/sbin/mknod name c major minor
/sbin/mknod name b major minor
/sbin/mknod name p

DESCRIPTION
The mknod command creates the following types of files:
- Character device special file (first SYNOPSIS form),
- Block device special file (second SYNOPSIS form),
- FIFO file, sometimes called a named pipe (third SYNOPSIS form).

name is the path name of the file to be created. The newly created file has a default mode that is readable
and writable by all users (0666), but the mode is modified by the current setting of the user's file mode crea-
tion mask (see umask(1)).

Character and Block Special Files
Character device special files are used for devices that can transfer single bytes at a time, such as nine-
track magnetic tape drives, printers, plotters, disk drives operating in "raw" mode, and terminals. To
create a character special file, use the c argument.

Block device special files are used for devices that usually transfer a block of data at a time, such as disk
drives. To create a block device special file, use the b argument.

The remaining arguments specify the device that will be accessible through the new special file:

major The major number specifies the major device type (for example, the device driver
number).
minor The minor number specifies the device location, which is typically, but not always, the
unit, drive, and/or line number.

The major and minor values can each be specified in hexadecimal, octal, or decimal, using C language con-
ventions (decimal: no leading zero; octal: leading zero; hexadecimal: leading 0x).

The assignment of major and minor device numbers is specific to each HP-UX system. Refer to the System
Administrator manuals supplied with your system for details.

Only users who have appropriate privileges can use mknod to create a character or block device special
file.

FIFO files
To create a FIFO (named pipe or buffer) file, use the p argument. You can also use the mkfifo command
for this purpose (see mkfifo(1)). All users can use mknod to create FIFO files.

WARNINGS
Access Control Lists
In HFS file systems, optional ACL entries can be added to special files and FIFOs with the chacl com-
mand (see chacl(1)). In JFS file systems, optional ACL entries can be added to special files and FIFOs with
the setacl command (see setacl(1)). However, system programs are likely to silently change or eliminate
the optional ACL entries for these files.

SEE ALSO
chacl(1), mkdir(1), mkfifo(1), setacl(1), umask(1), lsdev(1M), sam(1M), mknod(2), acl(5), aclv(5), mknod(5).

HP-UX System Administrator manuals.

STANDARDS CONFORMANCE
mknod: SVID2, SVID3, XPG2
NAME
mksf - make a special (device) file

SYNOPSIS
[driver-options] [special-file]
/sbin/mksf [-C class | -d driver] [-D directory] [-H hw-path] -m minor [-q | -v] [-r]

DESCRIPTION
The mksf command makes a special file in the devices directory, normally /dev, for an existing device, a device that has already been assigned an instance number by the system. The device is specified by supplying some combination of the -C, -d, -H, and -I options. If the options specified match a unique device in the system, mksf creates a special file for that device; otherwise, mksf prints an error message and exits. If required, mksf creates any subdirectories relative to the device installation directory that are defined for the resulting special file.

For most drivers, mksf has a set of built-in driver options, driver-options, and special-file naming conventions. By supplying some subset of the driver options, as in the first form above, the user can create a special file with a particular set of characteristics. If a special-file name is specified, mksf creates the special file with that special file name; otherwise, the default naming convention for the driver is used.

In the second form, the minor number and special-file name are explicitly specified. This form is used to make a special file for a driver without using the built-in driver options in mksf. The -r option specifies that mksf should make a character (raw) device file instead of the default block device file for drivers that support both.

Options
mksf recognizes the following options:

- **-C class**
  Match a device that belongs to a given device class, class. Device classes can be listed with the lsdev command (see lsdev(1M)). They are defined in the files in the directory /usr/conf/master.d. This option is not valid for pseudo devices. This option cannot be used with -d.

- **-d driver**
  Match a device that is controlled by the specified device driver, driver. Device drivers can be listed with the lsdev command (see lsdev(1M)). They are defined in the files in the directory /usr/conf/master.d. This option cannot be used with -C.

- **-D directory**
  Override the default device installation directory /dev and install the special files in directory instead. directory must exist; otherwise, mksf displays an error message and exits. See WARNINGS.

- **-H hw-path**
  Match a device at a given hardware path, hw-path. Hardware paths can be listed with the ioscan command (see ioscan(1M)). A hardware path specifies the addresses of the hardware components leading to a device. It consists of a string of numbers separated by periods (.), such as 52 (a card), 52.3 (a target address), and 52.3.0 (a device). If a hardware component is a bus converter, the following period, if any, is replaced by a slash (/) as in 2, 2/3, and 2/3.0. This option is not valid for pseudo devices.

- **-I instance**
  Match a device with the specified instance number. Instances can be listed with the -f option of the ioscan command (see ioscan(1M)). This option is not valid for pseudo devices.

- **-m minor**
  Create the special file with the specified minor number minor. The format of minor is the same as that given in mknod(1M) and mknod(5).

- **-q**
  Quiet option. Normally, mksf displays a message as each driver is processed. This option suppresses the driver message, but not error messages. See the -v option.

- **-r**
  Create a character (raw) special file instead of a block special file.

- **-v**
  Verbose option. In addition to the normal processing message, display the name of each special file as it is created. See the -q option.
Naming Conventions
Many special files are named using the card|target|device naming convention. These variables have the following meaning wherever they are used.

- **card** The unique interface card identification number from ioscan. It is represented as a decimal number with a typical range of 0 to 255.
- **target** The device target number, for example the address on a HP-FL or SCSI bus. It is represented as a decimal number with a typical range of 0 to 15.
- **device** A address unit within a device, for example, the unit in a HP-FL device or the LUN in a SCSI device. It is represented as a decimal number with a typical range of 0 to 15.

Special Files
The driver-specific options (driver-options) and default special file names (special-file) are listed below.

- **asio0**
  - **-a access-mode** Port access mode (0-2). The default access mode is 0 (Direct connect). The access-mode meanings are:

<table>
<thead>
<tr>
<th>access-mode</th>
<th>Port Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Direct connect</td>
</tr>
<tr>
<td>1</td>
<td>Dial out modem</td>
</tr>
<tr>
<td>2</td>
<td>Dial in modem</td>
</tr>
</tbody>
</table>

- **-c** CCITT.
- **-f** Hardware flow control (RTS/CTS).
- **-i** Modem dialer. Cannot be used with -l.
- **-l** Line printer. Cannot be used with -i.
- **-p port** Multiplexer port number (0 for built-in serial port; 0-1 for SAS console ports). The default port number is 0.
- **-r fifo-trigger** fifo-trigger should have a value between 0 and 3. The following table shows the corresponding FIFO trigger level for a given fifo-trigger value.

<table>
<thead>
<tr>
<th>fifo-trigger</th>
<th>Receive FIFO Trigger Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>14</td>
</tr>
</tbody>
</table>

- **-t** Transparent mode (normally used by diagnostics).
- **-x xmit-limit** xmit-limit should have a value between 0 and 3. The following table shows the corresponding transmit limit for a given xmit-limit value.

<table>
<thead>
<tr>
<th>xmit-limit</th>
<th>Transmit Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>12</td>
</tr>
</tbody>
</table>

**special-file** The default special file name depends on the access-mode and whether the -i and -l options are used.
audio

-\f format  Audio format (0-3). The format meanings are:

<table>
<thead>
<tr>
<th>format</th>
<th>Audio Format</th>
<th>File Name Modifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No change in audio format</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>8-bit Mu-law</td>
<td>U</td>
</tr>
<tr>
<td>2</td>
<td>8-bit A-law</td>
<td>A</td>
</tr>
<tr>
<td>3</td>
<td>16-bit linear</td>
<td>L</td>
</tr>
</tbody>
</table>

-\o output-dest  Output destination (0-4). The output-dest should have a value between 0 and 4. The following table shows the corresponding output destinations for a given output-dest value.

<table>
<thead>
<tr>
<th>output-dest</th>
<th>Output Destinations</th>
<th>File Name Modifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>All outputs</td>
<td>B</td>
</tr>
<tr>
<td>1</td>
<td>Headphone</td>
<td>E</td>
</tr>
<tr>
<td>2</td>
<td>Internal Speaker</td>
<td>I</td>
</tr>
<tr>
<td>3</td>
<td>No output</td>
<td>N</td>
</tr>
<tr>
<td>4</td>
<td>Line output</td>
<td>L</td>
</tr>
</tbody>
</table>

-\r  Raw, control access. This option cannot be used with either the -\f or -\o options.

special-file  The default special file name depends on the options specified.

<table>
<thead>
<tr>
<th>Options</th>
<th>Special File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>-\r</td>
<td>audioCtl_card</td>
</tr>
<tr>
<td>-\f 0</td>
<td>audio_card</td>
</tr>
<tr>
<td>all others</td>
<td>audiooutput-modformat-mod_card</td>
</tr>
</tbody>
</table>

The optional output-mod and format-mod values are given in the tables above. Note the underscore (_) before card in each special file name. Also note that for card 0, each file will be linked to a simpler name without the trailing _card.

autox0 schgr  Note that -\i cannot be used with either -\r or -\p.

-\i  Ioctl; create picker control special file.

-\p optical-disk[:last-optical-disk]  The optical disk number (starts with 1). If the optional :last-optical-disk is given then special files for the range of disks specified will be created.

-\r  Raw; create character, not block, special file.

The optional path name and option values are given in the tables above. Note the underscore (_) before card in each special file name. Also note that for card 0, each file will be linked to a simpler name without the trailing _card.

special-file  A special file cannot be given if a range of optical disks is given with the -\p option. If one is given for the single disk case, the name will have an a appended to the end for the A-side device and a b appended to the end for the B-side device. The default special file name depends on whether the -\r option is used.
**mksf(1M)**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-r</code></td>
<td>Special File Name</td>
</tr>
<tr>
<td>yes</td>
<td>rac/card/target/device_optical-diska</td>
</tr>
<tr>
<td>yes</td>
<td>rac/card/target/device_optical-diskb</td>
</tr>
<tr>
<td>no</td>
<td>ac/card/target/device_optical-diska</td>
</tr>
<tr>
<td>no</td>
<td>ac/card/target/device_optical-diskb</td>
</tr>
</tbody>
</table>

Note the underscore (_) between `device` and `optical-disk`.

**CentIf**

```
-h handshake-mode
```

Handshake mode. Valid values range from 1 to 6:

<table>
<thead>
<tr>
<th>handshake-mode</th>
<th>Handshake operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Automatic NACK/BUSY handshaking</td>
</tr>
<tr>
<td>2</td>
<td>Automatic BUSY only handshaking</td>
</tr>
<tr>
<td>3</td>
<td>Bidirectional read/write</td>
</tr>
<tr>
<td>4</td>
<td>Stream mode (NSTROBE only, no handshaking)</td>
</tr>
<tr>
<td>5</td>
<td>Automatic NACK/BUSY with pulsed NSTROBE</td>
</tr>
<tr>
<td>6</td>
<td>Automatic BUSY with pulsed NSTROBE</td>
</tr>
</tbody>
</table>

**special-file**

The default special file name is `cardt0d0_lp` for `handshake-mode` 2 and `cardt0d0h_handshake-mode_lp` for all others. For SCentIf, the only valid handshake value is 2 (Automatic BUSY only handshaking).

**consp1**

```
-f fifo-trigger
```

Fifo-trigger should have a value between 0 and 3. The following table shows the corresponding FIFO trigger level for a given `fifo-trigger` value:

<table>
<thead>
<tr>
<th>fifo-trigger</th>
<th>Receive FIFO Trigger Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>14</td>
</tr>
</tbody>
</table>

**-t**

Transparent mode (normally used by diagnostics).

**-x xmit-limit**

Xmit-limit should have a value between 0 and 3. The following table shows the corresponding transmit limit for a given `xmit-limit` value:

<table>
<thead>
<tr>
<th>xmit-limit</th>
<th>Transmit Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>12</td>
</tr>
</tbody>
</table>

**special-file**

The default special file name is as follows:

<table>
<thead>
<tr>
<th>Special File Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ttycard/p0</td>
<td>Floppy</td>
</tr>
</tbody>
</table>

**disc3**

```
-f floppy
```

Floppy.

```
-r raw
```

Raw; create character, not block, special file.

```
-s section
```

The section number.

**special-file**

The default special file name depends on whether the `-r` and `-s` options are used:
hil

Note that only one of -a, -k, or -r is allowed.

- a address  The link address (1-7).
- k          Cooked keyboard.
- n          The hil controller device.

special-file  The default special file name depends on the -a, -k, and -r options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Special File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>-a</td>
<td>hil_card.address</td>
</tr>
<tr>
<td>-k</td>
<td>hilkbd_card</td>
</tr>
<tr>
<td>-r</td>
<td>rhil_card</td>
</tr>
</tbody>
</table>

Note the underscore (_) before card. Also note that for card 0, each file will be linked to a simpler name without _card, either hiladdress, hilkbd, or rhil.

lan0 lan1 lan2 lan3

Note that only one of -e or -i is allowed.

- e          Ethernet protocol.
- i          IEEE 802.3 protocol.
- t          Transparent mode (normally used by diagnostics).

special-file  The default special file name depends on the -e, -i, and -t options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Special File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>-e</td>
<td>no</td>
</tr>
<tr>
<td>-e</td>
<td>yes</td>
</tr>
<tr>
<td>-i</td>
<td>no</td>
</tr>
<tr>
<td>-i</td>
<td>yes</td>
</tr>
</tbody>
</table>

lantty0

- e          Exclusive access.

special-file  The default special file name depends on whether the -e option is used:

<table>
<thead>
<tr>
<th>Option</th>
<th>Special File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>-e</td>
<td>no</td>
</tr>
<tr>
<td>-e</td>
<td>yes</td>
</tr>
</tbody>
</table>

lpr2 lpr3

- c          Capital letters. Convert all output to uppercase.
- e          Eject page after paper-out recovery.
- n          No form-feed.
- o          Old paper-out behavior (abort job).
- r          Raw.
- t          Transparent mode (normally used by diagnostics).
- w          No wait. Don’t retry errors on open.

special-file  The default special file name depends on whether the -r option is used:

<table>
<thead>
<tr>
<th>Option</th>
<th>Special File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>-e</td>
<td>no</td>
</tr>
<tr>
<td>-e</td>
<td>yes</td>
</tr>
</tbody>
</table>
**mksf(1M)**

**-r**  
Special File Name  
no  `card/target/device_lp`  
yes  `card/target/device_rlp`

**mux0 mux2 mux4 pci_mux0 pci_mux1**

**-a** access-mode  
Port access mode (0-2). The default access mode is 0 (Direct connect). The access-mode meanings are:

<table>
<thead>
<tr>
<th>access-mode</th>
<th>Port Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Direct connect</td>
</tr>
<tr>
<td>1</td>
<td>Dial out modem</td>
</tr>
<tr>
<td>2</td>
<td>Dial in modem</td>
</tr>
</tbody>
</table>

**-c**  
CCITT.

**-f**  
Hardware flow control (RTS/CTS).

**-i**  
Modem dialer. Cannot be used with `-l`.

**-l**  
Line printer. Cannot be used with `-i`.

**-p** port  
Multiplexer port number (0-15 for `mux0` and `mux2`; 0-1 for `mux4`; a1 - a16, b1 - b16, c1 - c16 & etc for the `pci_mux0` or `pci_mux1`). Some MUX cards controlled by a particular driver have fewer than the maximum supported ports.

**-t**  
Transparent mode (normally used by diagnostics).

**special-file**  
The default special file name depends on the access-mode and whether the `-i` and `-l` options are used. The term "card" below refers to the Instance number of the mux card.

<table>
<thead>
<tr>
<th>access-mode</th>
<th>-i</th>
<th>-l</th>
<th>Special File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>--</td>
<td>no</td>
<td>yes</td>
<td><code>ocard/pport_lp</code></td>
</tr>
<tr>
<td>2</td>
<td>no</td>
<td>no</td>
<td><code>ttyd/card/pport</code></td>
</tr>
<tr>
<td>1</td>
<td>no</td>
<td>no</td>
<td><code>cul/card/pport</code></td>
</tr>
<tr>
<td>0</td>
<td>yes</td>
<td>no</td>
<td><code>cua/card/pport</code></td>
</tr>
<tr>
<td>0</td>
<td>no</td>
<td>no</td>
<td><code>tty/card/pport</code></td>
</tr>
</tbody>
</table>

**plop sflop**

**-r**  
Raw; create character, not block, special file.

**special-file**  
The default special file name depends on whether the `-r` option is used:

<table>
<thead>
<tr>
<th>-r</th>
<th>Special File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>no</td>
<td><code>floppy/card/target/device</code></td>
</tr>
<tr>
<td>yes</td>
<td><code>rfloppy/card/target/device</code></td>
</tr>
</tbody>
</table>

**ps2**

Note that only one of `-a`, or `-p` is allowed.

**-a** auto_device  
Autosearch device. An auto_device value of 0 means first mouse; a value of 1 means first keyboard.

**-p** port  
PS2 port number.

**special-file**  
The default special file name depends on the `-a`, and `-p` options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Special File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>-a 0</td>
<td>ps2mouse</td>
</tr>
<tr>
<td>-a 1</td>
<td>ps2kbd</td>
</tr>
<tr>
<td>-p</td>
<td>ps2_port</td>
</tr>
</tbody>
</table>

Note the underscore (_) before port.

**SAS console ports**  
See `asio0`.

**SCentIf**  
See `CentIf`.

---

*HP-UX 11i Version 2: December 2007 Update*
sccl  
- **a** access-mode  
Port access mode (0–2). The default access mode is 0. The access-mode meanings are:

<table>
<thead>
<tr>
<th>access-mode</th>
<th>Port Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Direct connect</td>
</tr>
<tr>
<td>1</td>
<td>Dial out modem</td>
</tr>
<tr>
<td>2</td>
<td>Dial in modem</td>
</tr>
</tbody>
</table>

- **b**  
Port B.

- **c**  
CCITT.

- **i**  
Modem dialer. Cannot be used with -l.

- **l**  
Line printer. Cannot be used with -i.

**special-file**  
The default special file name depends on the access-mode and whether the -i and -l options are used.

<table>
<thead>
<tr>
<th>access-mode</th>
<th>-i</th>
<th>-l</th>
<th>Special File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>no</td>
<td>yes</td>
<td>ocardpportlp</td>
</tr>
<tr>
<td>2</td>
<td>no</td>
<td>no</td>
<td>ttydcardpport</td>
</tr>
<tr>
<td>1</td>
<td>no</td>
<td>no</td>
<td>culcardpport</td>
</tr>
<tr>
<td>0</td>
<td>yes</td>
<td>no</td>
<td>cuaardpport</td>
</tr>
<tr>
<td>0</td>
<td>no</td>
<td>no</td>
<td>ttyardpport</td>
</tr>
</tbody>
</table>

schgr  
See autox0.

sdisk  
- **r**  
Raw; create character, not block, special file.

- **s** section  
The section number.

**special-file**  
The default special file name depends on whether the -r and -s options are used:

<table>
<thead>
<tr>
<th>-r</th>
<th>-s</th>
<th>Special File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>yes</td>
<td>no</td>
<td>rdk/card/target/device</td>
</tr>
<tr>
<td>yes</td>
<td>yes</td>
<td>rdk/card/target/device&gt;ssection</td>
</tr>
<tr>
<td>no</td>
<td>no</td>
<td>dsk/card/target/device</td>
</tr>
<tr>
<td>no</td>
<td>yes</td>
<td>dsk/card/target/device&gt;ssection</td>
</tr>
</tbody>
</table>

sflop  
See pflop.

stape  
- **a**  
AT&T-style rewind/close.

- **b** bpi  
Bits per inch or tape density. The recognized values for bpi are:  
BEST, D1600, D3480, D3480C, D3590, D3590C, D6250, D6250C, D800, D8MM_8200, D8MM_8200C, D8MM_8500, D8MM_8500C, DDS1, DDS1C, DDS2, DDS2C, NOMOD, QIC_1000, QIC_11, QIC_120, QIC_1350, QIC_150, QIC_2100, QIC_24, QIC_2GB, QIC_525, QIC_5GB, DLT_42500_24, DLT_42500_56, DLT_62500_64, DLT_81633_64, DLT_62500_64C, DLT_81633_64C, U_18, U_18C, or a decimal number density code.

- **c** [code]  
Compression with optional compression code. The optional decimal code is used to select a particular compression algorithm on drives that support more than one compression algorithm. This option must be specified at the end of an option string.  
See mt(7) for more details.

- **e**  
Exhaustive mode. This option allows the driver to experiment with multiple configuration values in an attempt to access the media. The default behavior is to use only the configuration specified.

- **n**  
No rewind on close.

- **p**  
Partition one.

- **s** [block-size]  
Fixed block size mode. If a numeric block-size is given, it is used for a fixed block size. If the -s option is used alone, a device-specific default fixed block size is used. This
option must be specified at the end of an option string.

-\texttt{u}
  \texttt{UC Berkeley-style rewind/close.}

-\texttt{w}
  \texttt{Wait (disable immediate reporting).}

-\texttt{x index}
  Use the \texttt{index} value to access the tape device driver property table entry. Recognized values for \texttt{index} are decimal values in the range 0 to 30.

\texttt{special-file}
Put all tape special files in the \texttt{/dev/rmt} directory. This is required for proper maintenance of the Tape Property Table (see \texttt{mt(7)}). Device files located outside the \texttt{/dev/rmt} directory may not provide consistent behavior across system reboots. The default special file names are dependent on the tape drive being accessed and the options specified. All default special files begin with \texttt{rmt/card/target/device}. See \texttt{mt(7)} for a complete description of the default special file naming scheme for tapes.

\texttt{tape2}

-\texttt{a}
  \texttt{AT&T-style rewind/close.}

-\texttt{b bpi}
  Bits per inch or tape density. The recognized values for \texttt{bpi} are:
  \texttt{BEST, D1600, D3480, D3480C, D6250, D6250C, D800, D8MM_8200,
  D8MM_8200C, D8MM_8500, DDSI, DDSI1C, DDS2, DDS2C, NOMOD, QIC_1000, QIC_11, QIC_120, QIC_1350, QIC_150, QIC_2100, QIC_24, QIC_2GB, QIC_525, QIC_5GB, DLT_42500_24, DLT_42500_56, DLT_5200_64, DLT_81633_64, DLT_81633_64C, DLT_81633_64C, or a decimal number density code.}

-\texttt{c [code]}
  Compression with optional compression code. The optional decimal code is used to select a particular compression algorithm on drives that support more than one compression algorithm. This option must be specified at the end of an option string. See \texttt{mt(7)} for more details.

-\texttt{n}
  \texttt{No rewind on close.}

-\texttt{o}
  \texttt{Console messages disabled.}

-\texttt{t}
  \texttt{Transparent mode, normally used by diagnostics.}

-\texttt{u}
  \texttt{UC Berkeley-style rewind/close.}

-\texttt{w}
  \texttt{Wait (disable immediate reporting).}

-\texttt{x index}
  Use the \texttt{index} value to access the tape device driver property table entry. The recognized values for \texttt{index} are decimal values in the range 0 to 30.

-\texttt{z}
  \texttt{RTE compatible close.}

\texttt{special-file}
Put all tape special files in the \texttt{/dev/rmt} directory. This is required for proper maintenance of the Tape Property Table (see \texttt{mt(7)}). Device files located outside the \texttt{/dev/rmt} directory may not provide consistent behavior across system reboots. The default special file names are dependent on the tape drive being accessed and the options specified. All default special files begin with \texttt{rmt/card/target/device}. See \texttt{mt(7)} for a complete description of the default special file naming scheme for tapes.

\textbf{RETURN VALUE}
\texttt{mksf} exits with one of the following values:

\texttt{0}
Successful completion.

\texttt{1}
Failure. An error occurred.

\textbf{DIAGNOSTICS}
Most of the diagnostic messages from \texttt{mksf} are self-explanatory. Listed below are some messages deserving further clarification. Errors cause \texttt{mksf} to abort immediately.

\textbf{Errors}
\textbf{Ambiguous device specification}

Matched more than one device in the system. Use some combination of the \texttt{-d, -C, -H, and -I} options to specify a unique device.
No such device in the system
No device in the system matched the options specified. Use \texttt{ioscan} to list the devices in the system (see \texttt{ioscan(1M)}).

Device driver name is not in the kernel
Device class name is not in the kernel
The indicated device driver or device class is not present in the kernel. Add the appropriate device driver and/or device class to the kernel using \texttt{kcmodule}.

Device has no instance number
The specified device has not been assigned an instance number. Use \texttt{ioscan} to assign an instance to the device.

Directory directory doesn't exist
The directory argument of the \texttt{-D} option doesn't exist. Use \texttt{mkdir} to create the directory (see \texttt{mkdir(1)}).

EXEMPLES
Make a special file named \texttt{/dev/printer} for the line printer device associated with instance number 2.

\begin{verbatim}
 mksf -C printer -I 2 /dev/printer
\end{verbatim}

Make a special file, using the default naming convention, for the tape device at hardware path 8.4.1. The driver-specific options specify 1600 bits per inch and no rewind on close.

\begin{verbatim}
 mksf -C tape -H 8.4.1 -b D1600 -n
\end{verbatim}

WARNINGS
Many commands and subsystems assume their device files are in \texttt{/dev}; therefore, the use of the \texttt{-D} option is discouraged.

AUTHOR
mksf was developed by HP.

FILES
\begin{itemize}
\item \texttt{/dev/config} I/O system special file
\item \texttt{/etc/mtconfig} Tape driver property table database
\end{itemize}

SEE ALSO
\begin{itemize}
\item mkdir(1), insf(1M), ioscan(1M), kcmodule(1M), lsdev(1M), mknod(1M), rmsf(1M), mknod(2), ioconfig(4), mknod(5), mt(7).
\end{itemize}
modprpw(1M)

NAME
modprpw - modify protected password database

SYNOPSIS
modprpw [-E] [-V] [-l] [-n [domain]]
modprpw [-x] [-l] [-n [domain]] username
modprpw [-A] [-e] [-v] [-k] [-m field=value,...] [-l] [-n [domain]] username

DESCRIPTION
modprpw updates the user's protected password database settings. This command is available only to the superuser in a trusted system.

Usage other than via SAM, and/or modifications out of sync with /etc/passwd or NIS+ tables, may result in serious database corruption and the inability to access the system.

All updated values may be verified using getprpw(1M).

The database contains information for both local and NIS+ users. However, some NIS+ information is kept on the master. Since a user may be both local and NIS+, modprpw uses the nsswitch.conf(4) default if neither -l nor -n are specified.

Options
modprpw sets user's parameters as defined by the options specified. At least one option is required. If a field is not specified in the option then its value remains unchanged in the database.

modprpw recognizes the following options...

-A To add a new user entry and to return a random password which the new user must use to login the first time. This entry has to be created with the given username and the -m uid=value.

Error is returned if the user already exists.

May be combined with one of the -l or -n options. It also adds entries to the NIS+ tables, if -n is specified.

Unlike useradd(1M), it does not create nor populate the home directory, and it does not update /etc/passwd.

-E This option is specified WITHOUT a user name to expire all user's passwords. It goes through the protected password database and zeroes the successful change time of all users. The result is all users will need to enter a new password at their next login.

May be combined with one of -l or -n options.

-e This option is specified with a user name to expire the specified user's password. It zeroes the successful change time.

May be combined with options -l, -m, -n.

-k To unlock/enable a user's account that has become disabled, except when the lock is due to a missing password or * password.

May be combined with options -l, -m, -n.

-l This option specifies to modify data for a local user. It cannot be specified with the -n option. This option must be specified with other options.

-m Modify the database field to the specified value and/or resets locks. Valid with one of -A, -e, -v, -k options; and one of -l, -n options.

A list of database fields may be used with comma as a delimiter. An "invalid-opt" is printed, and processing terminates, if a list of database fields passed to -m contains an invalid database field.

Boolean values are specified as YES, NO, or DFT for system default values (/tcb/files/auth/system/default). Numeric values are specified as positive numbers, 0, or -1. If the value -1 is specified, the numeric value in the database is removed, allowing the system default value to be used. Time values are specified in days, although the database keeps them in seconds.

No aging is present if the following 4 database parameters are all zero: u_minchg, u_exp, u_life, u_pw_expire_warning.
Unless specified by n/a, all database fields can be set. They are listed below in the order shown in prot.h. The database fields are fully explained in prpwd(4).

<table>
<thead>
<tr>
<th>FIELD = VALUE</th>
<th>DATABASE FIELD</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>n/a</td>
<td>database u_name</td>
<td>Set the u_name of the user. No sanity checking is done on this value.</td>
</tr>
<tr>
<td>uid=value</td>
<td>database u_id</td>
<td>Set the uid of the user. No sanity checking is done on this value.</td>
</tr>
<tr>
<td>n/a</td>
<td>database u_pwd</td>
<td>Set the uid of the user. No sanity checking is done on this value.</td>
</tr>
<tr>
<td>n/a</td>
<td>database u_owner</td>
<td>Set the u_owner of the user.</td>
</tr>
<tr>
<td>bootpw=value</td>
<td>database u_bootauth</td>
<td>Set boot authorization privilege, YES/NO/DFT. NO removes it from the user file.</td>
</tr>
<tr>
<td>audid=value</td>
<td>database u_auditid</td>
<td>Set audit id. Automatically limited not to exceed the next available id.</td>
</tr>
<tr>
<td>audflg=value</td>
<td>database u_auditflag</td>
<td>Set audit flag.</td>
</tr>
<tr>
<td>minum=value</td>
<td>database u_minch(time)</td>
<td>Set the minimum time interval between password changes (days). 0 = none. Same as non-trusted mode minimum time.</td>
</tr>
<tr>
<td>maxpwln=value</td>
<td>database u_maxlen</td>
<td>Set the maximum password length for system generated passwords.</td>
</tr>
<tr>
<td>exptm=value</td>
<td>database u_exp(time)</td>
<td>Set the maximum password length for system generated passwords.</td>
</tr>
<tr>
<td>lftm=value</td>
<td>database u_life</td>
<td>Set the last login time interval (days). Used with u_succlog.</td>
</tr>
<tr>
<td>n/a</td>
<td>database u_succhg</td>
<td>Modified by options e, E, v, V, maybe k.</td>
</tr>
<tr>
<td>n/a</td>
<td>database u_unsucchg</td>
<td></td>
</tr>
<tr>
<td>acctexp=value</td>
<td>database u_acct_expire(time)</td>
<td>Set account expiration time interval (days). This interval is added to &quot;now&quot; to form the value in the database (database 0 = no expiration).</td>
</tr>
<tr>
<td>llog=value</td>
<td>database u_llogin</td>
<td>Set the last login time interval (days). Used with u_succlog.</td>
</tr>
<tr>
<td>expwarn=value</td>
<td>database u_pw_expire_warning(time)</td>
<td>Set password expiration warning time interval (days). 0 = none.</td>
</tr>
<tr>
<td>n/a</td>
<td>database u_pswduser</td>
<td>Obsoleted field.</td>
</tr>
<tr>
<td>usrpick=value</td>
<td>database u_pickpw</td>
<td>Set whether User Picks Password, YES/NO/DFT.</td>
</tr>
<tr>
<td>sysnpw=value</td>
<td>database u_genpwd</td>
<td>Set whether system generates pronounceable passwords, YES/NO/DFT.</td>
</tr>
<tr>
<td>rstrpw=value</td>
<td>database u_restrict</td>
<td>Set if generated password is restricted, YES/NO/DFT. If YES, password will be checked for triviality.</td>
</tr>
<tr>
<td>nullpw=value</td>
<td>database u_nullpw</td>
<td></td>
</tr>
</tbody>
</table>
modprpw(1M)

Set whether null passwords are allowed, YES/NO/DFT. YES is not recommended!

n/a
database u_pwchanger. Obsolescent field.

admnum=value
database u_pw_admin_num. Obsoleted field.

syschpw=value
database u_genchars.

Set whether system generates passwords having characters only, YES/NO/DFT.

sysltpw=value
database u_genletters.

Set whether system generates passwords having letters only, YES/NO/DFT.

timeod=value
database u_tod.

Set the time-of-day allowed for login.

The format is:

key0Starttime-Endtime, key1Starttime-Endtime,....
keynStarttime-Endtime

Where key has the following values:

Mo - Monday
Tu - Tuesday
We - Wednesday
Th - Thursday
Fr - Friday
Sa - Saturday
Su - Sunday
Any - everyday
Wk - Monday -> Friday

and Starttime and Endtime are in military format: HHMM, where: 00 <= HH <= 23, and 00 <= MM <= 59.

n/a
database u_suclog.

n/a
database u_unsuclog.

n/a
database u_suctty.

n/a
database u_numunsuclog.

n/a
database u_unsuctty.

umaxlntr=value
database u_maxtries.

Set Maximum Unsuccessful Login tries allowed. 0 = infinite.

n/a
database u_lock.

Set the administrator lock, YES/NO/DFT.

-n Can be specified with or without domain name; i.e., -n [domain]. If -n [domain] is specified, modifies data for the NIS+ user. The domain name must be fully qualified, with a terminating period. If domain name is not specified, the local domain will be used.

It cannot be specified with the -l option. This option must be specified with other options.

-V This option is specified WITHOUT a user name to "validate/refresh" all user's passwords. It goes through the protected password database and sets the successful change time to the current time for all users. The result is that all user's password aging restarts at the current time.

May be combined with one of -l or -n options.

-v This option is specified with a user name to "validate/refresh" the specified user's password. It sets the successful change time to the current time.

May be combined with options -l, -m, -n.

-x Delete the user's password and return a random password that the user must later supply to the login process to login and pick a new password. Not valid for root. Also resets locks.
modprpw(1M)

May be combined with one of -l or -n options.

RETURN VALUE
0  Success.
1  User not privileged.
2  Incorrect usage.
3  Can not find the entry or file.
4  Can not change the entry.
5  Not a Trusted System.
6  Not a NIS+ user.

EXAMPLES
Set the Minimum time between password changes to 12 (days), set the System generates pronounceable
password flag to NO, and set the System generates password having characters only flag to YES.

    modprpw -m mintm=12,syspnpw=NO,syschpw=YES someusr

The following example is to restrict the times that user joeblow can get on the system on Mondays and Fri-
days to 5PM-9PM, and Sundays from 5AM-9AM. Other days are not restricted.

    modprpw -m timeod=Mo1700-2100,Fr1700-2100,Su0500-0900 joeblow

WARNINGS
This command is intended for SAM use only. It may change with each release and can not be guaranteed
to be backward compatible.

Several database fields interact with others. Side effects may not be apparent until much later.

Special meanings may apply in the following cases:
• an absent field,
• a field without a value,
• a field with a zero value.

Very little, if any checking is done to see if values are valid. It is the user's responsibility to range check
values.

FILES
/etc/passwd  System Password file
/tcb/files/auth/*  Protected Password Database
/tcb/files/auth/system/default  System Defaults Database

AUTHOR
modprpw was developed by HP.

SEE ALSO
getprpw(1M), prpwd(4), nsswitch.conf(4).
mount(1M)

NAME
mount, umount - mount and unmount file systems

SYNOPSIS
/usr/sbin/mount [-l] [-p] [-v]  
/usr/sbin/mount [-a] [-F FStype] [-eQ]  
/usr/sbin/mount [-F FStype] [-eQrV] [-o specific_options] |special|directory|  
/usr/sbin/mount [-F FStype] [-eQrV] [-o specific_options] |special|directory|  
/usr/sbin/mount [-v] [-V] [-f] |special|directory|  
/usr/sbin/mount [-a] [-F FStype] [-h host] [-v] [-f]  

DESCRIPTION
The mount command mounts file systems. Only a superuser can mount file systems. Other users can use mount to list mounted file systems.

The mount command attaches special, a removable file system, to directory, a directory on the file tree. directory, which must already exist, will become the name of the root of the newly mounted file system. special and directory must be given as absolute path names. If either special or directory is omitted, mount attempts to determine the missing value from an entry in the /etc/fstab file. mount can be invoked on any removable file system, except /

If mount is invoked without any arguments, it lists all of the mounted file systems from the file system mount table, /etc/mnttab.

The umount command unmounts mounted file systems. Only a superuser can unmount file systems.

Options (mount)
The mount command recognizes the following options:

- **-a** Attempt to mount all file systems described in /etc/fstab. All optional fields in /etc/fstab must be included and supported. If the -F option is specified, all file systems in /etc/fstab with that FStype are mounted. If noauto is specified in an entry's option list, this entry is skipped. File systems are not necessarily mounted in the order listed in /etc/fstab.

- **-e** Verbose mode. Write a message to the standard output indicating which file system is being mounted.

- **-F FStype** Specify FStype, the file system type on which to operate. See fstyp(1M). If this option is not included on the command line, then it is determined from either /etc/fstab, by matching special with an entry in that file, or from file system statistics of special, obtained by statfsdev() (see statfsdev(3C)).

- **-l** Limit actions to local file systems only.

- **-o specific_options** Specify options specific to each file system type. specific_options is a list of comma separated suboptions and/or keyword/attribute pairs intended for a FStype-specific version of the command. See the FStype-specific manual entries for a description of the specific_options supported, if any.

- **-p** Report the list of mounted file systems in the /etc/fstab format.

- **-Q** Prevent the display of error messages that result from an attempt to mount already mounted file systems.

- **-r** Mount the specified file system as read-only. Physically write-protected file systems must be mounted in this way or errors occur when access times are updated, whether or not any explicit write is attempted.

- **-h** Unmount only those file systems listed in /etc/mnttab that are remote-mounted from host.

- **-v** Report the regular output with file system type and flags; however, the directory and special fields are reversed.
mount(1M)

mount(1M)

-V
Echo the completed command line, but perform no other action. The command line is generated by incorporating the user-specified options and other information derived from /etc/fstab. This option allows the user to verify the command line.

Options (umount)
The umount command recognizes the following options:

-a Attempt to unmount all file systems described in /etc/mnttab. All optional fields in /etc/mnttab must be included and supported. If FStype is specified, all file systems in /etc/mnttab with that FStype are unmounted. File systems are not necessarily unmounted in the order listed in /etc/mnttab.

-F FStype Specify FStype, the file system type on which to operate. If this option is not included on the command line, then it is determined from /etc/mnttab by matching special with an entry in that file. If no match is found, the command fails.

-v Verbose mode. Write a message to standard output indicating which file system is being unmounted.

-V Echo the completed command line, but perform no other action. The command line is generated by incorporating the user-specified options and other information derived from /etc/fstab. This option allows the user to verify the command line.

-f Forced unmount. Not supported by all file system types. See file system type specific man pages for more information.

EXAMPLES
List the file systems currently mounted:

mount
Mount the HFS file system /dev/dsk/c1t2d0 at directory /home:

mount -F hfs /dev/dsk/c1t2d0 /home

Unmount the same file system:

umount /dev/dsk/c1t2d0

AUTHOR
mount was developed by HP, AT&T, the University of California, Berkeley, and Sun Microsystems.

FILES
/etc/fstab Static information about the systems
/etc/mnttab Mounted file system table

SEE ALSO
fsadm(1M), mount_cachefs(1M), mount_cdfs(1M), mount_hfs(1M), mount_lofs(1M), mount_memfs(1M), mount_nfs(1M), mount_vxfs(1M), setmnt(1M), mount(2), fstab(4), mnttab(4), fs_wrapper(5), quota(5).

STANDARDS CONFORMANCE
mount: SVID3
umount: SVID3
NAME
mount_cachefs: mount - mount CacheFS file systems

SYNOPSIS
mount -F cachefs [ generic_options ] -o backfstype=file_system_type
[ other_cacheFS_options ] special_mount_point

DESCRIPTION
The CacheFS-specific version of the mount command mounts a cached file system; if necessary, it NFS-mounts its back file system. It also provides a number of CacheFS-specific options for controlling the caching process.

Options
To mount a CacheFS file system, use the generic mount command with the -F option followed by the argument cachefs. The following generic mount options are available:
- V Verbose mode. Write a message to the standard output indicating which file system is being mounted.
- r Mount the file system read-only.

The following arguments to the -o option are specifically for CacheFS mounts. Use commas to separate multiple options. Notice that the backfstype argument must be specified.
backfstype=file_system_type
The file system type of the back file system (for example, nfs).

backpath=path
Specifies where the back file system is already mounted. If this argument is not supplied, CacheFS determines a mount point for the back file system. The back file system must be read-only.

cachedir=directory
The name of the cache directory.

cacheid=ID ID is a string specifying a particular instance of a cache. If you do not specify a cache ID, CacheFS will construct one.

rpages If specified when mounting a CacheFS file system, a binary will be read and populated in the cache the first time it is loaded. Subsequent access to the binary will be satisfied from the cache.

write-around | non-shared
Write modes for CacheFS. The write-around mode (the default) handles writes the same as NFS does; that is, writes are made to the back file system, and the affected file is purged from the cache. You can use the non-shared mode when you are sure that no one else will be writing to the cached file system. In this mode, all writes are made to both the front and the back file system, and the file remains in the cache.

noconst Disables cache consistency checking. By default, periodic consistency checking is enabled. Specify noconst only when you know that the back file system will not be modified. Trying to perform cache consistency check using cfsadmin -s will result in error. demandconst and noconst are mutually exclusive.

demandconst Verifies cache consistency only when explicitly requested, rather than the periodic checking that is done by default. A consistency check is requested by using the -s option of the cfsadmin(1M) command. This option is useful for back file systems that change infrequently, for example, /usr/bin. demandconst and noconst are mutually exclusive.

local-access Causes the front file system to interpret the mode bits used for access checking instead of having the back file system verify access permissions. Do not use this argument with secure NFS.

purge Purge any cached information for the specified file system.

rw | ro Read-write (default) or read-only.
mount_cachefs(1M)
mount_cachefs(1M)
suid | nosuid
Allow (default) or disallow set-uid execution.

acregmin=n Specifies that cached attributes are held for at least \( n \) seconds after file modification. After \( n \) seconds, CacheFS checks to see if the file modification time on the back file system has changed. If it has, all information about the file is purged from the cache and new data is retrieved from the back file system. The default value is 30 seconds.

acregmax=n Specifies that cached attributes are held for no more than \( n \) seconds after file modification. After \( n \) seconds, all file information is purged from the cache. The default value is 30 seconds.

acdirmin=n Specifies that cached attributes are held for at least \( n \) seconds after directory update. After \( n \) seconds, CacheFS checks to see if the directory modification time on the back file system has changed. If it has, all information about the directory is purged from the cache and new data is retrieved from the back file system. The default value is 30 seconds.

acdirmax=n Specifies that cached attributes are held for no more than \( n \) seconds after directory update. After \( n \) seconds, all directory information is purged from the cache. The default value is 30 seconds.

actimeo=n Sets acregmin, acregmax, acdirmin, and acdirmax to \( n \).

EXAMPLES
The following example CacheFS-mounts the file system server1:/user2, which is already NFS-mounted on /usr/abc as /xyz.

```
example# mount -F cachefs -o backfstype=nfs,backpath=/usr/abc,
cachedir=/cache1 server1:/user2 /xyz
```

The lines similar to the following appear in the /etc/mnttab file after the mount command is executed:

```
server1:/user2 /usr/abc nfs
server1:/user2 /cache1/xyz cachefs backfstype=nfs
```

AUTHOR
mount_cachefs was developed by Sun Microsystems, Inc.

SEE ALSO
cfsadmin(1M), fsck_cachefs(1M), mount(1M).
NAME
mount_cdfs: mount, umount - mount and unmount CDFS file systems

SYNOPSIS
/usr/sbin/mount [-l] [-p] [-v]
/usr/sbin/mount -a [-F cdfs] [-eQ]
/usr/sbin/mount [-F cdfs] [-eQrV] [-o specific_options] {special | directory}
/usr/sbin/mount -F cdfs [-eQrV] [-o specific_options] special directory
/usr/sbin/mount -a [-F cdfs] [-v]
/usr/sbin/umount [-v] [-V] {special | directory}

DESCRIPTION
The mount command mounts file systems. Only a superuser can mount file systems. Other users can use mount to list mounted file systems.

The mount command attaches special, a removable file system, to directory, a directory on the file tree. directory, which must already exist, will become the name of the root of the newly mounted file system. If either special or directory is omitted, mount attempts to determine the missing value from an entry in the /etc/fstab file. mount can be invoked on any removable file system, except /.

If mount is invoked without any arguments, it lists all of the mounted file systems from the file system mount table, /etc/mnttab.

The umount command unmounts mounted file systems. Only a superuser can unmount file systems.

Options (mount)
mount recognizes the following options:

- -a Attempt to mount all file systems described in /etc/fstab. All optional fields in /etc/fstab must be included and supported. If -F cdfs is specified, all CDFS file systems in /etc/fstab are mounted. If noauto is specified in an entry's option list, this entry is skipped. File systems are not necessarily mounted in the order listed in /etc/fstab.
- -e Verbose mode. Write a message to standard output indicating which file system is being mounted.
- -F cdfs Specify the CDFS file system type (see fstyp(1M)).
- -l Limit actions to local file systems only.
- -o specific_options Specify options specific to the CDFS file system type. specific_options is a list of comma-separated suboptions and/or keyword/attribute pairs intended for the CDFS specific module of the command.

The following specific_options are valid on CDFS file systems.

cccase Suppress the display of version numbers. Show and match file names as lower case.
rr Use Rock Ridge extension to the ISO-9660 file system (default).
norr Do not use Rock Ridge extension to the ISO-9660 file system.
defaults Use all default options. When given, this must be the only option specified.
ro Mount read-only (default).
suid Allow set-user-ID execution (default).
nosuid Do not allow set-user-ID execution.
- -p Report the list of mounted file systems in the /etc/fstab format.
- -Q Prevent the display of error messages resulting from an attempt to mount already mounted file systems.
mount_cdfs(1M)

Mount the specified file system as read-only. This option is equivalent to the `-o ro` specific_option. For CDFS file systems this is a default option.

-`v`  Report the regular output with file system type and flags; however, directory and special fields are reversed.

-`V`  Echo the completed command line, but performs no other action. The command line is generated by incorporating the user-specified options and other information derived from `/etc/fstab`. This option allows the user to verify the command line.

Options (umount)

`umount` recognizes the following options:

-`a`  Attempt to unmount all file systems described in `/etc/mnttab`. All optional fields in `/etc/mnttab` must be included and supported. If `-F cdfs` is specified, all CDFS file systems in `/etc/mnttab` are unmounted. File systems are not necessarily unmounted in the order listed in `/etc/mnttab`.

-`F cdfs`  Specify the CDFS file system type (see `fstyp(1M)`).

-`v`  Verbose mode. Write a message to standard output indicating which file system is being unmounted.

-`V`  Echo the completed command line, but performs no other action. The command line is generated by incorporating the user-specified options and other information derived from `/etc/fstab`. This option allows the user to verify the command line.

DIAGNOSTICS

`umount` complains if the special file is not mounted or if it is busy. The file system is busy if it contains an open file or some logged-in user's working directory.

EXAMPLES

Mount a local CDFS disk:

```bash
mount -F cdfs /dev/dsk/c0t0d4 /cdrom
```

Unmount a local CDFS disk:

```bash
umount /dev/dsk/c0t0d4
```

WARNINGS

Some degree of validation is done on the file system, however, it is generally unwise to mount file systems that are defective, corrupt, or of unknown origin.

NOTES

Additional CD-ROM formats are supported using PFS (Portable File System) utilities. See `pfs(4)` for more details.

AUTHOR

`mount` was developed by HP, AT&T, the University of California, Berkeley, and Sun Microsystems.

FILES

/`etc/fstab`  Static information about the file systems

/`etc/mnttab`  Mounted file system table

SEE ALSO

`fsclean(1M)`, `mount(1M)`, `quotaon(1M)`, `mount(2)`, `fstab(4)`, `mnttab(4)`, `pfs(4)`, `fs_wrapper(5)`, `quota(5)`.

STANDARDS CONFORMANCE

`mount`: SVID3

`umount`: SVID3
mount_hfs(1M)

NAME
mount_hfs: mount, umount - mount and unmount HFS file systems

SYNOPSIS
/usr/sbin/mount [-l] [-p] [-v]
/usr/sbin/mount [-a] [-F hfs] [-eQ] [-f]
/usr/sbin/mount [-F hfs] [-eQrV] [-f] [-o specific_options] [special directory]
/usr/sbin/mount [-a] [-F hfs] [-v]
/usr/sbin/umount [-v] [-V] [special directory]

DESCRIPTION
The mount command mounts file systems. Only a superuser can mount file systems. Other users can use mount to list mounted file systems.

The mount command attaches special, a removable file system, to directory, a directory on the file tree. directory, which must already exist, will become the name of the root of the newly mounted file system. special and directory must be given as absolute path names. If either special or directory is omitted, mount attempts to determine the missing value from an entry in the /etc/fstab file. mount can be invoked on any removable file system, except /

If mount is invoked without any arguments, it lists all of the mounted file systems from the file system mount table, /etc/mnttab.

The umount command unmounts mounted file systems. Only a superuser can unmount file systems.

Options (mount)
mount recognizes the following options:

- a Attempt to mount all file systems described in /etc/fstab. All optional fields in /etc/fstab must be included and supported. If -F hfs is specified, all HFS file systems in /etc/fstab are mounted. If noauto is specified in an entry's option list, this entry is skipped. File systems are not necessarily mounted in the order listed in /etc/fstab.
- e Verbose mode. Write a message to standard output indicating which file system is being mounted.
- f Force the file system to be mounted, even if the file system clean flag indicates that the file system should have fsck run on it before mounting (see fsck(1M)). This option is valid only on HFS file systems.
- F hfs Specify the HFS file system type (see fsctype(1M)).
- l Limit actions to local file systems only.
- o specific_options
Specify options specific to the HFS file system type. specific_options is a list of comma separated suboptions and/or keyword/attribute pairs intended for the HFS specific module of the command.

The following specific_options are valid on HFS file systems.

defaults Use all default options. When given, this must be the only option specified.
rw Mount read-write (default).
ro Mount read-only.
suid Allow set-user-ID execution (default).
sosuid Do not allow set-user-ID execution.
behind Enable, where possible, asynchronous writes to disk. This is the default on 700 systems.
mount_hfs(1M)

delayed
Enable delayed or buffered writes to disk. This is the default on 800 systems.

fs_async
Enable relaxed posting of file system metadata.

no_fs_async
Enable rigorous posting of file system metadata. This is the default.

largefiles
Attempt to enable the creation of files greater than 2 gigabytes in size. File systems have to be created or configured to enable large files (see mkfs_hfs(1M) and fsadm_hfs(1M)).

nolargefiles
Attempt to disable the creation of files greater than 2 gigabytes in size. File systems have to be created or configured to disable large files. (see mkfs_hfs(1M) and fsadm_hfs(1M)).

quota
Enable disk quotas (valid only for rw file systems).
	noquota
Disable disk quotas (default).

Mounting with the quota option also enables quotas for the file system, unlike some other systems, which require the additional invocation of the quotaon command after the file system has been mounted (see quotaon(1M)). Running quotaon does no harm, but it is not necessary.

-p
Report the list of mounted file systems in the /etc/fstab format.

-Q
Prevent the display of error messages resulting from an attempt to mount already mounted file systems.

-r
Mount the specified file system as read-only. This option is equivalent to the -o ro specific_option. Physically write-protected file systems must be mounted in this way or errors occur when access times are updated, whether or not any explicit write is attempted.

-v
Report the regular output with file system type and flags; however, directory and special fields are reversed.

-V
Echo the completed command line, but performs no other action. The command line is generated by incorporating the user-specified options and other information derived from /etc/fstab. This option allows the user to verify the command line.

Options (umount)

umount recognizes the following options:

-a
Attempt to unmount all file systems described in /etc/mnttab. All optional fields in /etc/mnttab must be included and supported. If -F hfs is specified, all HFS file systems in /etc/mnttab are unmounted. File systems are not necessarily unmounted in the order listed in /etc/mnttab.

-F hfs
Specify the HFS file system type (see fstyp(1M)).

-v
Verbose mode. Write a message to standard output indicating which file system is being unmounted.

-V
Echo the completed command line, but performs no other action. The command line is generated by incorporating the user-specified options and other information derived from /etc/fstab. This option allows the user to verify the command line.

DIAGNOSTICS

umount complains if the special file is not mounted or if it is busy. The file system is busy if it contains an open file or some logged-in user's working directory.

EXAMPLES

Mount a local HFS disk:

    mount -F hfs /dev/dsk/c0t0d4 /usr

594 Hewlett-Packard Company

HP-UX 11i Version 2: December 2007 Update
mount_hfs(1M)

Unmount a local HFS disk:

    umount /dev/dsk/c0t0d4

WARNINGS

Some degree of validation is done on the file system, however, it is generally unwise to mount file systems that are defective, corrupt, or of unknown origin.

AUTHOR

mount was developed by HP, AT&T, the University of California, Berkeley, and Sun Microsystems.

FILES

/etc/fstab         Static information about the file systems
/etc/mnttab       Mounted file system table

SEE ALSO

fsck(1M), mount(1M), mkfs_hfs(1M), fsadm_hfs(1M), quota(1M), mount(2), fstab(4), mnttab(4), fs_wrapper(5), quota(5).

STANDARDS CONFORMANCE

mount: SVID3
umount: SVID3
mount_lofs(1M)

NAME
mount_lofs: mount, umount - mount and unmount LOFS file systems

SYNOPSIS
/usr/sbin/mount [-p] [-v]
/usr/sbin/mount -a [-F lofs] [-eQ]
/usr/sbin/mount [-F lofs] [-eQrV] [-o specific_options] {special_directory | directory}
/usr/sbin/mount [-F lofs] [-eQrV] [-o specific_options] special_directory directory
/usr/sbin/umount [v] [V] {special_directory | directory}
/usr/sbin/umount -a [-F lofs] [-v]

DESCRIPTION
The mount command mounts LOFS file systems. Only superuser can mount LOFS file systems. Other
users can use mount to list mounted file systems.

mount, attaches special_directory, a directory from one of the mounted file systems, to directory, an
other directory in one of the mounted file systems. This enables new file systems to be created, which
provide access to existing directories or file systems using alternate path names. Both special_directory
and directory should already exist. directory will become the root of the newly mounted LOFS file system,
containing the file system hierarchy under special_directory. special_directory and directory must be
specified as absolute path names. If either special_directory or directory is omitted, mount attempts to
determine the missing value from an entry in the /etc/fstab file. mount can be invoked on any
removable file system, except /.

If mount is invoked without any arguments, it lists all the mounted file systems from the file system
mount table, /etc/mnttab.

The umount command unmounts mounted file systems. Only a superuser can unmount file systems.

Options (mount)
mount recognizes the following options:

-a Attempt to mount all file systems described in /etc/fstab. All optional fields in
/etc/fstab must be included and supported. If -F lofs is specified, all LOFS
file systems in /etc/fstab are mounted. If noauto is specified in an entry's
option list, this entry is skipped. File systems are not necessarily mounted in the
order listed in /etc/fstab.

-e Verbose mode. Write a message to standard output indicating which file system is
being mounted.

-F lofs Specify the LOFS file system type (see fstyp(1M)).

-l Limit actions to local file systems only. LOFS is a local file system.

-o specific_options
Specify options specific to the LOFS file system type. specific_options is a list of
comma separated suboptions and/or keyword/attribute pairs intended for the LOFS
specific module of the command.

The following specific_options are valid on an LOFS file system:

defaults Use all default options. When used, this must be the only option
specified.

ro Read-only (see WARNINGS below).

-p Report the list of mounted file systems in the /etc/fstab format.

-Q Prevent display of error messages resulting from an attempt to mount already
mounted file systems.

-r Mount the specified file system as read-only (see WARNINGS below).

-v Report the output in a new style. The new style has the file system type and flags
displayed in addition to the old output. The directory and special_directory fields are
reversed.
mount_lofs(1M)

- V

Echo the completed command line, but perform no other action. The command line is generated by incorporating the user-specified options and other information derived from /etc/fstab. This option allows the user to verify the command line.

Options (umount)

The umount command recognizes the following options:

- a
  Attempt to unmount all file systems described in /etc/mnttab. All optional fields in /etc/mnttab must be included and supported. If -F lofs file system type is specified, all the LOFS file systems in /etc/mnttab are unmounted. File systems are not necessarily unmounted in the order listed in /etc/mnttab.

- F lofs
  Specify the LOFS file system type (see fstyp(1M)).

- v
 Verbose mode. Write a message to standard output indicating which file system is being unmounted.

- V
  Echo the completed command line, but perform no other action. The command line is generated by incorporating the user-specified options and other information derived from /etc/fstab. This option allows the user to verify the command line.

EXAMPLES

Mount an LOFS file system:

    mount /usr /tmp/usr

Mount another LOFS file system:

    mount -F lofs /usr/sbin /tmp/sbin

WARNINGS

LOFS file systems provide the user with numerous applications; however, they may be potentially confusing. LOFS file systems should generally be created by an experienced user.

For LOFS file systems which are mounted read-only, if the underlying file system is mounted writable, certain write operations on the LOFS will succeed. Thus LOFS should not be relied upon to provide a strictly write-only alternative image of a read-write file system.

AUTHOR

mount was developed by HP, AT&T, the University of California, Berkeley, and Sun Microsystems.

FILES

/etc/fstab
Static information about the file systems

/etc/mnttab
Mounted file system table

SEE ALSO

mount(1M), mount(2), fstab(4), mnttab(4).

STANDARDS CONFORMANCE

mount: SVID3
NAME

mount_memfs: mount, umount - mount and unmount MemFS file systems

SYNOPSIS

/usr/sbin/mount [-l] [-p] [-v]
/usr/sbin/mount -a [-F memfs] [-eQ]
/usr/sbin/mount [-F memfs] [-eQv] [-o specific_options] directory
/usr/sbin/umount -a [-F memfs] [-v]
/usr/sbin/umount [-v] [-V] directory

DESCRIPTION

The mount command mounts file systems. Only a superuser can mount file systems. Other users can use
mount to list mounted file systems.

The mount command attaches a removable file system, to directory, a directory on the file tree. directory,
which must already exist, will become the name of the root of the newly mounted file system. directory
must be given as an absolute path name.

mount can be invoked on any removable file system, except the / directory.

It is important to note that mount_memfs does an internal mkfs and hence a separate mkfs is not
needed for MemFS filesystems. It constructs the MemFS filesystem in the buffer cache. The MemFS
blocks will be swapped out of the buffer cache when there is buffer cache or system memory pressure. MemFS
blocks that are swapped out of the buffer cache are stored in memory allocated to a user process associated
with every MemFS mount instance. The user process is created during a mount and destroyed with a
corresponding umount.

The tunable memfs_bufcache_swapct can be used to mark the threshold for swapping out MemFS
blocks from the buffer cache to the user process (see memfs_bufcache_swapct(5)).

If mount is invoked without any arguments, it lists all of the mounted file systems from the file system
mount table, /etc/mnttab.

The umount command unmounts mounted file systems. Only a superuser can unmount file systems.

Options (mount)

mount recognizes the following options:

- **-a**  Attempt to mount all file systems described in /etc/fstab. All optional fields in
  /etc/fstab must be included and supported. If -F memfs is specified, all MemFS
  file systems in /etc/fstab are mounted. If noauto is specified in an entry's option
  list, this entry is skipped. File systems are not necessarily mounted in the order listed in
  /etc/fstab.

- **-e**  Verbose mode. Write a message to standard output indicating which file system is being
  mounted.

- **-F memfs**  Specify the MemFS file system type (see fstyp(1M)).

- **-l**  Limit actions to local file systems only.

- **-o specific_options**  Specify options specific to the MemFS file system type. specific_options is a list of comma
  separated suboptions and/or keyword/attribute pairs intended for the MemFS specific
  module of the command.

  The following specific_options are valid on MemFS file systems.

  defaults  Use all default options.

  size  The size for the MemFS filesystem. Append to size, kb or KB to indicate the
  value is in kilobytes, mb or MB to indicate megabytes, or gb or GB to indic-
  ate gigabytes.

  The default value is the percentage of maximum buffer cache size that can
  be occupied by MemFS, as specified by the memfs_bufcache_swapct
  tunable. The size of MemFS file systems is limited by UFS_MAXDEVBLK,
  defined in <sys/fs.h>, which is 256 GB - 1, or 268,435,455 blocks.
mount_memfs(1M)

suid  Allow set-user-ID execution (default).

nosuid  Do not allow set-user-ID execution.

-p  Report the list of mounted file systems in the /etc/fstab format.

-Q  Prevent the display of error messages resulting from an attempt to mount already
    mounted file systems.

-v  Report the regular output with file system type and flags; however, directory and special
    fields are reversed. The special field for memfs always contains the text memfs.

-V  Echo the completed command line, but performs no other action. The command line is
    generated by incorporating the user-specified options and other information derived from
    /etc/fstab. This option allows the user to verify the command line.

Options (umount)

umount recognizes the following options:

-a  Attempt to unmount all file systems described in /etc/mnttab. All optional fields in
    /etc/mnttab must be included and supported. If -F memfs is specified, all MemFS
    file systems in /etc/mnttab are unmounted. File systems are not necessarily
    unmounted in the order listed in /etc/mnttab.

-F memfs  Specify the MemFS file system type (see fstyp(1M)).

-v  Verbose mode. Write a message to standard output indicating which file system is being
    unmounted.

-V  Echo the completed command line, but performs no other action. The command line is
    generated by incorporating the user-specified options and other information derived from
    /etc/fstab. This option allows the user to verify the command line.

Notes

Quotas are not available for MemFS filesystems.

MemFS filesystems are largefiles-enabled.

MemFS filesystems do not support forced unmount.

MemFS filesystems cannot be mounted read only.

DIAGNOSTICS

umount complains if the directory is not mounted or if it is busy. The file system is busy if it contains an
open file or some logged-in user's working directory.

EXAMPLES

Mount a MemFS of size 100 MB on /mnt:

    mount -F memfs -o size=100MB /mnt

Unmount a local MemFS disk:

    umount /mnt

AUTHOR

mount was developed by HP, AT&T, the University of California, Berkeley, and Sun Microsystems.

FILES

/etc/fstab  Static information about the file systems

/etc/mnttab  Mounted file system table

SEE ALSO

fsclean(1M), mount(1M), quotaon(1M), mount(2), fstab(4), mnttab(4), pfs(4), fs_wrapper(5),
memfs_bufcache_metapct(5), memfs_bufcache_swappct(5), quota(5).

HP-UX 11i Version 2: December 2007 Update  -  2  -  Hewlett-Packard Company
mount_memfs(1M)

STANDARDS CONFORMANCE

mount: SVID3
umount: SVID3
mount_nfs(1M)       mount_nfs(1M)

NAME
       mount_nfs: mount, umount - mount and unmount NFS file systems

SYNOPSIS
       /usr/sbin/mount [-l] [-p] [-v]
       /usr/sbin/mount -a [-F nfs] [-eQ]
       /usr/sbin/mount [-F nfs] [-eQrV] [-o specific_options] [host: path] directory
       /usr/sbin/mount [-a] [-eQrV] [-o specific_options] [host: path] directory
       /usr/sbin/umount -a [-f] [-F nfs] [-h host] [-v]
       /usr/sbin/umount [-f] [-v] [-V] [host: path] directory

DESCRIPTION
       The mount command mounts file systems. Only a superuser can mount file systems. Other users can use
       mount to list mounted file systems.

       The mount command attaches host: path to directory. host is a remote system, path is a directory on this
       remote system and directory is a directory on the local file tree. directory must already exist, be given as
       an absolute path name and will become the name of the root of the newly mounted file system. If either
       host: path or directory is omitted, mount attempts to determine the missing value from an entry in the
       /etc/fstab file. mount can be invoked on any removable file system, except /.

       If mount is invoked without any arguments, it lists all of the mounted file systems from the file system
       mount table, /etc/mnttab. The umount command unmounts mounted file systems. Only a superuser
       can unmount file systems.

Options
       -r       Mount the specified file system read-only.
       -o specific_options

Set file system specific options according to a comma-separated list chosen from words below.
       rw | ro       resource is mounted read-write or read-only. The default is rw.
       suid | nosuid  Setuid execution allowed or disallowed. The default is suid.
       remount     If a file system is mounted read-only, remounts the file system read-write.
       bg | fg      If the first attempt fails, retry in the background, or, in the foreground. The default is
                   fg.
       quota       Enables quota to check whether the user is over quota on this file system; if the file
                   system has quotas enabled on the server, quotas will still be checked for operations on
                   this file system.
       noquota     Prevent quota from checking whether the user exceeded the quota on this file sys-
                   tem; if the file system has quotas enabled on the server, quotas will still be checked
                   for operations on this file system.
       retry=n     The number of times to retry the mount operation. The default is 1.
       vers=NFS_version_number
                   By default, the version of NFS protocol used between the client and the server is the
                   highest one available on both systems. If the NFS server does not support NFS Ver-
                   sion 3, then the NFS mount will use NFS Version 2.
       port=n      Set server UDP port number to n (the default is the port customarily used for NFS
                   servers).
       proto=transp Use the transport protocol transp for this mount. Valid values for transp are tcp
                   (connection-oriented) and udp (connectionless). The default behavior is to attempt a
                   tcp connection. If the tcp connection attempt fails when defaulting to tcp, a udp
                   connection will be attempted.
       grpid       By default, the GID associated with a newly created file will obey the System V
                   semantics; that is, the GID is set to the effective GID of the calling process. This
mount_nfs(1M)  mount_nfs(1M)

behavior may be overridden on a per-directory basis by setting the set-GID bit of the
parent directory; in this case, the GID of a newly created file is set to the GID of the
parent directory (see open(2) and mkdir(2)). Files created on file systems that are
mounted with the grpid option will obey BSD semantics independent of whether
the set-GID bit of the parent directory is set; that is, the GID is unconditionally inher-
ited from that of the parent directory.

\[ \text{rsize=\textit{n}} \]
Set the read buffer size to \textit{n} bytes. The default value is set by kernel.

\[ \text{wsize=\textit{n}} \]
Set the write buffer size to \textit{n} bytes. The default value is set by kernel.

\[ \text{timeo=\textit{n}} \]
Set the NFS timeout to \textit{n} tenths of a second. The default value is set by kernel.

\[ \text{retrans=\textit{n}} \]
Set the number of NFS retransmissions to \textit{n}. The default value is 5.

\[ \text{soft | hard} \]
Return an error if the server does not respond, or continue the retry request until the
server responds. The default value is hard.

\[ \text{intr | nointr} \]
Allow (do not allow) keyboard interrupts to kill a process that is hung while waiting
for a response on a hard-mounted file system. The default is intr.

\[ \text{noac} \]
Suppress attribute caching.

\[ \text{nocto} \]
Suppress fresh attributes when opening a file.

\[ \text{forcedirectio} \]
If forcedirectio is specified, then for the duration of the mount, forced direct I/O
is used. If the filesystem is mounted using forcedirectio, data is transferred
directly between client and server, with no buffering on the client. forcedirec-
tio is a performance option that is of benefit only in large sequential data transfers.
Running applications and mapping memory with mmap() are not supported over
forcedirectio mount points. The default behavior is no forcedirectio.

\[ \text{devs | nodevs} \]
Allow (do not allow) access to local devices. The default is devs.

\[ \text{acdirmax=\textit{n}} \]
Hold cached attributes for no more than \textit{n} seconds after directory update. The default
value is 60.

\[ \text{acdirmin=\textit{n}} \]
Hold cached attributes for at least \textit{n} seconds after directory update. The default value
is 30.

\[ \text{acregmax=\textit{n}} \]
Hold cached attributes for no more than \textit{n} seconds after file modification. The default
value is 60.

\[ \text{acregmin=\textit{n}} \]
Hold cached attributes for at least \textit{n} seconds after file modification. The default value
is 3.

\[ \text{actimeo=\textit{n}} \]
Set \textit{min} and \textit{max} times for regular files and directories to \textit{n} seconds. actimeo has
no default; it sets acregmin, acregmax, acdirmin, and acdirmax to the value
specified.

\[ \text{llock} \]
By default, lock/unlock requests are sent to the server’s rpc.lockd which enforces
the requests. With this option set, the lock/unlock requests are not sent to the remote
rpc.lockd and are enforced locally. These local locks are enforced much faster, but
if other NFS clients access the same data on the server, there is a risk of data corrup-
tion. Consequently, this option must be used when only one NFS client accesses the
data on the server.

\[ \text{readdir} \]
Disable the READDIRPLUS functionality, which is used by default on an NFS v3
mount point, and use the NFS v2 READDIR functionality instead. The performance
of applications that read huge directories over NFS will vary between v2 and v3
depending on the type of information they need. The find command will be faster
using the v3 READDIRPLUS while the ls command will be faster using the v2 READ-
DIR. This option must be used on a case by case basis. It has no effect on a v2 mount
point.

\[ \text{-O} \]
Overlay mount. Allow the file system to be mounted over an existing mount point, making the under-
lying file system inaccessible. If a mount is attempted on a pre-existing mount point without setting
this flag, the mount will fail, producing the error device busy.
mount_nfs(1M)

Options (umount)

`umount` recognizes the following options:

- `-a` Attempt to unmount all file systems described in `/etc/mnttab`. All optional fields in `/etc/mnttab` must be included and supported. If `-F nfs` option is specified, all NFS file systems in `/etc/mnttab` are unmounted. File systems are not necessarily unmounted in the order listed in `/etc/mnttab`.

- `-f` Forcibly unmount a file system. Without this option, `umount` does not allow a file system to be unmounted if a file on the file system is busy. Using this option can cause data loss for open files; programs which access files after the file system has been unmounted will get an error ([EIO]).

- `-F nfs` Specify the NFS file system type (see `fstyp(1M)`).

- `-h host` Unmount only those file systems listed in `/etc/mnttab` that are remote-mounted from `host`.

- `-v` Verbose mode. Write a message to standard output indicating which file system is being unmounted.

- `-V` Echo the completed command line, but performs no other action. The command line is generated by incorporating the user-specified options and other information derived from `/etc/fstab`. This option allows the user to verify the command line.

NFS File Systems

Background vs. Foreground

File systems mounted with the `bg` option indicate that `mount` is to retry in the background if the server’s mount daemon (`mountd`) does not respond. `mount` retries the request up to the count specified in the `retry=n` option. Once the file system is mounted, each NFS request made in the kernel waits `timeo=n` tenths of a second for a response. If no response arrives, the time-out is multiplied by 2 and the request is retransmitted. When the number of retransmissions has reached the number specified in the `retrans=n` option, a file system mounted with the `soft` option returns an error on the request; one mounted with the `hard` option prints a warning message and continues to retry the request.

Hard vs. Soft

File systems that are mounted read-write or that contain executable files should always be mounted with the `hard` option. Applications using `soft` mounted file systems may incur unexpected I/O errors.

To improve NFS read performance, files and file attributes are cached. File modification times get updated whenever a write occurs. However, file access times may be temporarily out-of-date until the cache gets refreshed. The attribute cache retains file attributes on the client. Attributes for a file are assigned a time to be flushed. If the file is modified before the flush time, then the flush time is extended by the time since the last modification (under the assumption that files that changed recently are likely to change soon). There is a minimum and maximum flush time extension for regular files and for directories. Setting `actimeo=n` sets flush time to `n` seconds for both regular files and directories.

EXAMPLES

To mount an NFS file system:

```
mount serv:/usr/src /usr/src
```

To mount an NFS file system readonly with no suid privileges:

```
mount -r -o nosuid serv:/usr/src /usr/src
```

To mount an NFS file system over Version 3:

```
mount -o vers=3 serv:/usr/src /usr/src
```

To unmount all file systems imported from a given host, enter the following command as root:

```
umount -h mysystem.home.work.com -a
```

The hostname must match what is in `/etc/mnttab` exactly (as shown by the `bdf` command). For example, if `bdf` shows:

```
HP-UX 11i Version 2: December 2007 Update
Hewlett-Packard Company
```
mount_nfs(1M)

mysystem:/projects,
the umount command would be

    umount -h mysystem -a.

FILES

/etc/mnttab       table of mounted file systems.
/etc/fstab        list of default parameters for each file system.

SEE ALSO

fsclean(1M), mount(1M), quotaon(1M), mount(2), fstab(4), mnttab(4), fs_wrapper(5), quota(5).

STANDARDS CONFORMANCE

mount: SVID3
umount: SVID3
mount_vxfs(1M)  mount_vxfs(1M)

NAME
mount_vxfs: mount, umount - mount and unmount VxFS file systems

SYNOPSIS

mount [-l] [-v|-p]
mount [-F vxfs] [-eQ] -a
mount [-F vxfs] [-eQrV] [-o specific_options] {special |directory}

umount [-V] [-v] {special |directory}

DESCRIPTION

mount attaches special, a removable file system, to directory, a directory on the file tree. (This directory is also known as the mount point). directory, which must already exist, becomes the name of the root of the newly mounted file system. If you omit either special or directory, mount attempts to determine the missing value from an entry in /etc/fstab. mount can be invoked on any removable file system, except /, special and directory must be given as absolute path names.

mount notifies the system that special, a VxFS block special device, is available to users from mount_point, which must exist before mount is invoked. mount_point becomes the name of the root of the newly mounted file system special.

Large files (over two gigabytes) are supported on HP-UX 10.20 systems and above. Unlike some file system commands, you cannot specify multiple -o options to mount; only the last option is used.

If you invoke mount with no arguments, it lists all the mounted file systems from the mounted file system table, /etc/mnttab.

The umount command unmounts mounted file systems.

The vxumount command provides a force option to unmount mounted file systems.

Only the superuser can mount and umount file systems. Other users can use mount to list mounted file systems.

Options

mount recognizes the following options:

-a Attempts to mount all file systems described in /etc/fstab. All optional fields in /etc/fstab must be included and supported. If -F vxfs is specified, all VxFS file systems in /etc/fstab are mounted. If noauto is specified in an entry's option list (in /etc/fstab), that entry is skipped (not mounted). File systems are not necessarily mounted in the order listed in /etc/fstab.

-e Verbose mode. Writes a message to the standard output indicating which file system is being mounted.

-F vxfs Specifies the file system type (vxfs).

-l Limits actions to local file systems only.

-o specific_options Specifies options specific to the VxFS file system type. See the Specific -o Options subsection in this section.

-p Reports the list of mounted file systems in /etc/fstab format.

-Q Prevents display of error messages, resulting from an attempt to mount already mounted file systems.

-r Mounts the specified file system as read-only. Physically write-protected file systems must be mounted in this way or errors occur when access times are updated, whether or not any explicit write is attempted.
mount_vxfs(1M)

-\v Reports the regular output with file system type and flags, however, directory and special fields are reversed.
-\V Echoes the completed command line, but does not execute the command. The command line is generated by incorporating the user-specified options and other information derived from /etc/fstab. This option allows the user to verify the command line.

umount recognizes the following options:
-\a Attempt to unmount all file systems described in /etc/mnttab. All optional fields in /etc/mnttab must be included and supported. If -F vxfs is specified, all VxFS file systems in /etc/mnttab are unmounted. File systems are not necessarily unmounted in the order listed in /etc/mnttab.
-\F vxfs Specify the file system type (vxfs).
-\v Verbose mode. Write a message to the standard output indicating which file system is being unmounted.
-\V Echo the completed command line, but do not execute the command. The command line is generated by incorporating the user-specified options and other information derived from /etc/fstab. This option allows the user to verify the command line.

Operands
mount recognizes the following operands:

special Name of a VxFS block special device, a removable file system.

mount_point Name of an existing directory from which the mounted file system is available.

Specific -o Options
mount supports a set of options specific to the VxFS file system type through the -o option. specific_options is a list of comma separated suboptions and/or keyword/attribute pairs intended for the VxFS-specific module of the command. Unlike some file system commands, multiple -o options do not accumulate; only the last option is used.

The following specific_options are valid on a VxFS file system:

blkclear
Clears all data extents before allocating them to a file (requires synchronous zeroing, on disk, of certain newly allocated extents). This prevents uninitialized data from being written to a file at the time of a system crash.

cckpt=ckpt_name
Mounts the Storage Checkpoint of a VxFS file system. ckpt_name is the name of a file system Storage Checkpoint previously created. mount_point is the directory on which to mount the Storage Checkpoint. special is the Storage Checkpoint pseudo device. Storage Checkpoints are mounted on pseudo devices that do not appear in the system name space. The pseudo devices are created and exist only while the Storage Checkpoint is mounted. A Storage Checkpoint pseudo device name has the following format:

device_path:ckpt_name

Storage Checkpoints are mounted read-only by default, but you can mount or remount them as writable using the rw option. A file system must be mounted before any of its Storage Checkpoints can be mounted. A file system can be unmounted only after all of its Storage Checkpoints are unmounted.

To mount a Storage Checkpoint in shared mode on a cluster file system, you must also specify the -o cluster option (see the EXAMPLES Section).

ccluster
Mounts a file system in shared mode. special must be a shared volume in a cluster Volume Manager (CVM) environment. Other nodes in the cluster can also mount special in shared mode. A local mount cannot be remounted in shared mode and shared mount cannot be remounted in local mode.
The first node to mount special is called the primary node. The primary node handles intent logging for the cluster. Other nodes are called secondary nodes. A secondary writable node (rw) is not allowed if the primary node is mounted as read-only (ro).

```
convosync=direct|dsync|unbuffered|closesync|delay
```

Alters the caching behavior of the file system for O_SYNC and O_DSYNC I/O operations.

- **direct** value handles any reads or writes with the O_SYNC or O_DSYNC flag as if the VX_DIRECT caching advisory is set.
- **dsync** value handles any writes with the O_SYNC flag as if the VX_DSYNC caching advisory is set. It does not modify behavior for writes with O_DSYNC set.
- **unbuffered** value handles any reads or writes with the O_SYNC or O_DSYNC flag as if the VX_UNBUFFERED caching advisory is set.
- **closesync** value delays O_SYNC or O_DSYNC writes so that they do not take effect immediately.

The convosync, dsync, direct, and unbuffered values all run the equivalent of an fsync(2) to be run when any file accessed with the O_SYNC or O_DSYNC flag is closed.

The delay value delays O_SYNC or O_DSYNC writes so that they do not take effect immediately. With this option, VxFS changes O_SYNC or O_DSYNC writes into delayed writes. No special action is performed when closing a file. This option effectively cancels data integrity guarantees normally provided by opening a file with O_SYNC or O_DSYNC.

**Note:** The convosync option is available only with the HP OnLineJFS product.

**crw**

The cluster read-write option allows asymmetric mounts, that is, you can mount a specified cluster file system in read-only or read-write mode independently of the other shared file system nodes. crw must be specified with the -o cluster option. Without specifying crw, the default functionality of the cluster mount is retained; the read-write capability of cluster secondaries are the same as the cluster primary. You can use the crw in conjunction with rw or ro as shown in the following mount compatibility matrix:

<table>
<thead>
<tr>
<th>Secondary</th>
<th>primary</th>
<th>ro</th>
<th>rw</th>
<th>ro,crw</th>
<th>rw,crw</th>
</tr>
</thead>
<tbody>
<tr>
<td>ro</td>
<td>yes</td>
<td>no</td>
<td>no</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>rw</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>ro,crw</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>rw,crw</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
</tbody>
</table>

If the primary is mounted with ro,crw, or rw,crw as shown in the first column, the secondary read and write capabilities can still be set independently. For a cluster mount, rw on the primary enables cluster-wide read-write capability.

The read and write capabilities can be changed from its original setting to another using the -o remount option. The read and write capabilities can be changed according to the following matrix:

<table>
<thead>
<tr>
<th>From/To</th>
<th>ro</th>
<th>rw</th>
<th>ro,crw</th>
<th>rw,crw</th>
</tr>
</thead>
<tbody>
<tr>
<td>ro</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>rw</td>
<td>no</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
</tr>
<tr>
<td>ro,crw</td>
<td>no</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
</tr>
<tr>
<td>rw,crw</td>
<td>no</td>
<td>yes</td>
<td>no</td>
<td>yes</td>
</tr>
</tbody>
</table>

If a cluster file system is mounted read-write (rw), the underlying disk group must have the activation mode attribute set to sharedwrite (sw).

If a cluster file system is mounted ro,crw, and the disk group activation mode is sharedread (sr), that cluster file system can never be a primary, and must be mounted seconly (see the seconly option in this section). See the VERITAS SANPoint Foundation Suite Installation and Configuration Guide.
Generally, VxFS does _O_SYNC_ or _O_DSYNC_ writes by logging the data and the time change to the inode (_datainlog_). If the _nodatainlog_ option is used, the logging of synchronous writes is disabled; _O_SYNC_ writes the data into the file and updates the inode synchronously before returning to the user.

**Note:** The _datainlog_ option is available only with the HP OnLineJFS product.

Sets the policy for handling I/O errors on a mounted file system. Multiple error policies were implemented in VxFS to handle evolving storage technologies for which a single approach is no longer adequate.

I/O errors can occur while reading or writing file data, or while reading or writing metadata. The file system can respond to these I/O errors either by halting or by gradually degrading.

_ioerror_ provides four policies that determine how the file system responds to the various errors. All four policies limit data corruption, either by stopping the file system or by marking a corrupted inode as bad.

The following matrix shows how the file system responds to the various errors depending on the policy set:

<table>
<thead>
<tr>
<th></th>
<th>file read</th>
<th>file write</th>
<th>metadata read</th>
<th>metadata write</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>disable</strong></td>
<td>disable</td>
<td>disable</td>
<td>disable</td>
<td>disable</td>
</tr>
<tr>
<td><strong>nodisable</strong></td>
<td>degrade</td>
<td>degrade</td>
<td>degrade</td>
<td>degrade</td>
</tr>
<tr>
<td><strong>wdisable</strong></td>
<td>degrade</td>
<td>disable</td>
<td>degrade</td>
<td>disable</td>
</tr>
<tr>
<td><strong>mwdisable</strong></td>
<td>degrade</td>
<td>degrade</td>
<td>degrade</td>
<td>disable</td>
</tr>
</tbody>
</table>

If _disable_ is selected, VxFS disables the file system after detecting any I/O error. You must then unmount the file system and correct the condition causing the I/O error. After the problem is repaired, run _fsck_ and mount the file system again. In most cases, replay _fsck_ is sufficient to repair the file system. A full _fsck_ is required only in cases of structural damage to the file system's metadata. Select _disable_ in environments where the underlying storage is redundant, such as RAID-5 or mirrored disks.

If _nodisable_ is selected, when VxFS detects an I/O error, it takes steps (sets the appropriate error flags), to contain the error, but continues running. Note that the "degraded" condition indicates possible data or metadata corruption, not the overall performance of the file system.

For file data read and write errors, VxFS sets the _VX_DATAIOERR_ flag in the superblock. For metadata read errors, VxFS sets the _VX_FULLFSCK_ flag in the superblock. For metadata write errors, VxFS sets the _VX_FULLFSCK_ and _VX_METAIOERR_ flags in the superblock and may mark associated metadata as bad on disk. VxFS then prints the appropriate error messages to the console (see the _VERITAS File System Administrator's Guide_ for information on what actions to take for specific errors).

You should stop the file system as soon as possible and repair the condition causing the I/O error. After the problem is repaired, run _fsck_ and mount the file system again.

Select _nodisable_ if you want to implement the policy that most closely resembles the previous VxFS error handling policy.

If _wdisable_ (write disable) or _mwdisable_ (metadata-write disable) is selected, the file system is disabled or degraded, as shown in the matrix, depending on the type of error encountered. Select _wdisable_ or _mwdisable_ for environments where read errors are more likely to persist than write errors, such as when using non-redundant storage. _mwdisable_ is the default _ioerror_ mount option for local mounts. _disable_ is the default _ioerror_ mount option for cluster mounts.
Note: If there is serious damage to the file system, or there is structural corruption of file system metadata, VxFS marks the file system for full fsck regardless of which I/O error policy is in effect.

Behavior on cluster file systems is somewhat different. If the policy selected is disable, the file system is disabled only on the node where the I/O error occurs. The file system is still accessible from the other nodes. If the I/O error is on the CFS primary, a new primary is elected from the remaining nodes and the original primary becomes a secondary.

disable is the recommended policy for cluster file systems. With any other policy, a metadata I/O error can mark the file system for a full file system check. If the CFS primary subsequently fails, the other nodes in the cluster cannot take over the primaryship, thereby disabling access to the file system from all nodes in the cluster.

Note: If the CVM disk detach policy (the way unusable disks in a shared disk group are detached) is local, the I/O error policy must be disable.

largefiles|nolargefiles
These options do not turn largefiles capability on and off (use mkfs_vxfs or fsadm_vxfs to set and clear the largefiles flag), but they do verify whether a file system is largefiles capable. If nolargefiles is specified and the mount succeeds, then the file system does not contain any files whose size is two gigabytes or larger, and such files cannot be created. If largefiles is specified and the mount succeeds, then the file system may contain files whose size is two gigabytes or larger, and large files can be created. For a mount to succeed, the option must match the largefiles flag as specified by mkfs_vxfs or fsadm_vxfs.

Note: Large files are supported on HP-UX 10.20 systems and above. Be careful when enabling large file system capability. System administration utilities such as backup may experience problems if they are not large-file aware.

logiosize=1024|2048|4096
Control size of intent log I/O buffers. Default value is determined dynamically at mount time. The performance of some storage devices (specifically, devices using the read-modify-write feature) improves if the writes are issued in one or more multiples of a particular size. When a file system is mounted with the logiosize option, VxFS writes the intent log in at least size bytes, or a multiple of size bytes, to obtain the maximum performance from such devices. The values for size can be 1024, 2048, or 4096.

log|delaylog|tmplog|nolog
Control intent logging. To maintain file system integrity after a system failure, logging must be enabled. In log mode, file system structural changes are logged to disk before the system call returns to the application. If the system crashes, fsck_vxfs(1M) completes logged operations that did not complete.

In delaylog mode, some system calls return before the intent log is written. The default is delaylog. This improves the performance of the system, but some changes are not guaranteed until a short time later when the intent log is written. This mode approximates traditional UNIX system guarantees for correctness in case of system failures.

In tmplog mode, the intent log is almost always delayed. This improves performance, but recent changes may disappear if the system crashes. This mode is only recommended for temporary file systems.

nolog is an alias for tmplog.

mincache=direct|dsync|unbuffered|closesync|tmpcache
Alter the caching behavior of the file system.

The direct value handles any reads without the O_SYNC flag, or any writes without the O_SYNC flag, VX_DSYNC, VX_DIRECT, and VX_UNBUFFERED caching advisories, as if the VX_DIRECT caching advisory was set.

The dsync value handles any writes without the O_SYNC flag or one of the VX_DIRECT, VX_DSYNC, or VX_UNBUFFERED caching advisories as if the VX_DSYNC caching advisory was set.

The unbuffered value handles any reads without the O_SYNC flag, or any writes without the O_SYNC flag, VX_DSYNC, VX_DIRECT, and VX_UNBUFFERED caching advisories, as if the VX_UNBUFFERED caching advisory was set.
mount_vxfs(1M)

The `closesync`, `dsync`, `unbuffered`, and `direct` values all cause the equivalent of an `fsync` to be run when the file is closed.

The `tmpcache` value disables delayed extending writes, trading off integrity for performance. When this option is chosen, VxFS does not zero out new extents allocated as files are sequentially written. Uninitialized data may appear in files being written at the time of a system crash. See `vxfsio(7)` for an explanation of `VX_DIRECT`, `VX_DSYNC`, and `VX_UNBUFFERED`.

Note: `mincache`=`direct`, `mincache`=`dsync`, `mincache`=`unbuffered`, and `mincache`=`tmpcache` are available only with the HP OnLineJFS product.

`noatime` Directs the file system to ignore file access time updates except when they coincide with updates to `ctime` or `mtime` (see the `stat(2)` manual page). By default, the file system is mounted with access time (`atime`) recording. You can use the `noatime` option to reduce disk activity on file systems where access times are not important.

`qio|noqio` Enables or disables the VERITAS Quick I/O for Databases option for the given file system. Quick I/O is available as a licensed feature of VxFS. By default, `mount` enables Quick I/O on the file system. If Quick I/O is not available, `mount` mounts the file system without Quick I/O. If `qio` is specified, but the feature is not licensed, `mount` prints an error message and terminates without mounting the file system. If `noqio` is specified, `mount` disables Quick I/O even if the license is installed.

For cluster file systems, `qio` is also the default if a Quick I/O license is present.

`qlog` Enables the file system to use VERITAS QuickLog. QuickLog threads multiple file system logs together and redirects the file system log writes to the `special` device. If `special` is not specified, the file system selects one of the QuickLog special devices. This option improves the performance of NFS servers. QuickLog is available only through other VERITAS products. See the VERITAS File System Release Notes for current product information.

`quota` Enable disk quotas (valid only for `rw` type file systems). VxFS maintains quota information in a private area of the file system. If the file system is mounted with quotas enabled, and the file system was previously mounted with quotas disabled and was modified, then the quota information is rebuilt. This may take a while.

`remount` Change the mount options for a mounted file system. In particular, `remount` changes the logging and caching policies. It also changes a file system from read-only to read/write.

`remount` cannot change a file system from read/write to read-only, nor can it set the `snapof` or `snapsize` attributes.

`remount` does not check possible changes in `/etc/fstab`. If you use any specific option, you must explicitly pass it in the command line.

`rw|ro` Read/write or read-only. The default is `rw`.

`seconly` Mounts a shared file system as a secondary only. A secondary-only file system cannot assume the primaryship for the specified shared file system. For a mount with the `seconly` option to succeed, primary must already be mounted. `seconly` must be specified with the `-o` `cluster` option. The `seconly` option overrides any policy that was set using the `fsclustadm` command. This option can be set or reset using the `-o remount` option. A remount with `seconly` fails if the file system node is already the primary for the file system.

`snapof=`Mount the file system as a snapshot of `filesystem`, where `filesystem` is either the directory on which a VxFS file system is mounted, or is the block special file containing a mounted VxFS file system. An explicit `-F vxfs` option is required to mount a snapshot file system.

**Note:** `snapof=` is available only with the HP OnLineJFS product.

`snapsize=` Used in conjunction with `snapof`. `size` is the size in sectors of the snapshot file system being
mount_vxfs(1M)  mount_vxfs(1M)

mounted. This option is required only when the device driver is incapable of determining the
dsize of special, and defaults to the entire device if not specified.

Note: snapsize=size is available only with the HP OnLineJFS product.

suid|nosuid

setuid execution allowed or setuid execution not allowed. The default is sud.

tranflush

When VxFS is the default boot file system (/stand) on HP-UX, there can be no intent log
replay during the initial stages of the boot process. To ensure data and metadata consistency
during the boot process, the tranflush option flushes all metadata updates to disk before
returning from a system call. The tranflush option therefore enables VxFS to approxi-
mate the behavior of a file system with no intent logging functionality. The tranflush
option automatically enables the blkclear and log options. It is advisable to specify the
mincache=dasync mount option with tranflush. tranflush must be explicitly
specified when remounting the file system.

The tmplog, delaylog, and datainlog mount options do not operate with tran-
flush. The tranflush option does not operate on read-only file systems or cluster file sys-
tems.

EXAMPLES

List the file systems currently mounted:

mount

Mount a VxFS file system /dev/dsk/c1t2d0 at directory /home

mount -F vxfs /dev/dsk/c1t2d0 /home

Unmount the same file system:

umount /dev/dsk/c1t2d0

To unmount a file system, unmount the Storage Checkpoint first:

umount /ckptdir

umount /fsdir

Mount a Storage Checkpoint of a cluster file system on a VERITAS Volume Manager volume:

mount -F vxfs -o cluster,ckpt=ckpt_name \
/dev/vx/dsk/dg_name/volume_name:ckpt_name /ckpt_mount_point

Automatically mount Storage Checkpoints when the system reboots. You can list them in the
/etc/fstab file as in the following example:

     /dev/vx/dsk/fsvol /fsdir vxfs defaults 0 2
     /dev/vx/dsk/fsvol:myckpt /ckptdir vxfs ckpt=myckpt 0 0

FILES

/etc/fstab    Static information about the file systems
/etc/mnttab  Mounted file system table

SEE ALSO

edquota(1M), fsadm_vxfs(1M), fscck_vxfs(1M), mkfs_vxfs(1M), mount(1M), vxumount(1M), fsync(2),
mount(2), setuid(2), stat(2), fstat(4), mnttab(4), fs_vxfs(4), quota(5), vxfsio(7).

VERITAS File System Administrator's Guide

VERITAS File System Release Notes

VERITAS SANPoint Foundation Suite Installation and Configuration Guide

STANDARDS CONFORMANCE

  mount: SVID3
  umount: SVID3
mountall(1M)

NAME
mountall, umountall - mount and unmount multiple file systems

SYNOPSIS
/sbin/mountall
[[-F FStype] [-l|r] [file_system_table | -]]
/sbin/mountall [-l|r] [-m]
/sbin/mountall [-n]
/sbin/umountall [-F FStype] [-k] [-l|r]

DESCRIPTION
mountall is used to mount file systems according to file_system_table. By default, /etc/fstab is the file_system_table. If a dash (\-) is specified, mountall reads file_system_table from the standard input; the standard input must be in the same format as the /etc/fstab.

Before each file system is mounted, a check is done using fsck (see fsck(1M)) to ensure that the file system is mountable. If the file system is not mountable, it is repaired by fsck before the mount is attempted.

umountall causes all mounted file systems except the non-removable file systems such as root to be unmounted.

Options
mountall and umountall recognize the following options:

-F FStype Specify the file system type (FStype) to be mounted or unmounted.
-l Specify action on local file systems only.
-r Specify action on remote file systems only.
-k Send a SIGKILL signal to processes that have files opened.
-m Attempt to mount all the unmounted file systems. This option will not perform the file system consistency check and repair.
-n Perform the file system consistency check and repair on all unmounted file system. This option will not mount the file systems.

DIAGNOSTICS
Error and warning messages may originate from fsck, mount, fuser, or umount. See fsck(1M), mount(1M), or fuser(1M) to interpret the error and warning messages.

EXAMPLES
Mount all unmounted file systems listed in /etc/fstab:

mountall

Mount all local file systems listed in /etc/fstab:

mountall -l

Mount all remote file systems listed in /etc/fstab:

mountall -r

Mount all local hfs file systems:

mountall -F hfs -l

Unmount all NFS file systems and kill any processes that have files opened in the file system:

umountall -F nfs -k

WARNINGS
umountall, especially with the -k option, should be used with extreme caution, because it can cause severe damage.

The -n option may not be available in future releases.

mountall may not be effective with some cases of LOFS file systems.
mountall(1M)

FILES
/etc/fstab  Static information about the file systems
/etc/mnttab  Mounted file system table

SEE ALSO
fsck(1M), fuser(1M), mount(1M), signal(2), fstab(4), mnttab(4).
mountd(1M)  mountd(1M)

NAME
  mountd, rpc.mountd - NFS mount request server

SYNOPSIS
  /usr/sbin/rpc.mountd [-l log_file] [-t n]

Obsolescent Options
  /usr/sbin/rpc.mountd [-p]

DESCRIPTION
  mountd is an RPC server that answers file system mount requests. It reads file
  /etc/xtab (described in exports(4)) to determine which directories are available to which machines. It also provides information
  on what file systems are mounted by which clients. This information can be printed using the
  showmount command (see showmount(1M)).

  rpc.mountd can be started at boot time by setting the variable NFS_SERVER to 1 in the file
  /etc/rc.config.d/nfsconf.

  A fixed port can be specified for mountd by using the MOUNTD_PORT variable in the
  /etc/rc.config.d/nfsconf file, as shown:

          MOUNTD_PORT=port_number

  The port number can have any value between 1 and 65535. After adding the port variable to the
  /etc/rc.config.d/nfsconf file, mountd must be restarted for this to take effect. This feature can
  be disabled by deleting or commenting out the port variable from the /etc/rc.config.d/nfsconf
  file and restarting mountd.

Options
  mountd recognizes the following options:

          -l log_file  Log any errors to the named log file, log_file. Errors are not logged if the -l option is
                         not specified.
                         The information logged to the file includes the date and time of the error, the host name,
                         process ID and name of the function generating the error, and the error message. Note
                         that different services can share a single log file since enough information is included to
                         uniquely identify each error.

          -p           This is an obsolete option.

          -t n         Specify tracing level n, where n can have one of the following values:
                         1  Errors only (default)
                         2  Errors, mount requests and mount failures

WARNINGS
  If a client crashes, executing showmount on the server will show that the client still has a file system
  mounted; i.e., the client’s entry is not removed from /etc/rmtab until the client reboots and executes
  umount -a (see showmount(1M)).

  Also, if a client mounts the same remote directory twice, only one entry appears in /etc/rmtab. Doing a
  umount of one of these directories removes the single entry and showmount no longer indicates that the
  remote directory is mounted.

The MOUNTD_PORT variable will still be supported in HP-UX 11i V3, but the 11i V3 configuration file will
be /etc/default/nfs and not /etc/rc.config.d/nfsconf.

AUTHOR
  mountd was developed by Sun Microsystems, Inc.

FILES
  /etc/rmtab  List of all hosts having file systems mounted from this machine

SEE ALSO
  inetd(1M), mount(1M), portmap(1M), showmount(1M), exports(4), inetd.conf(4), inetd.sec(4), rmtab(4), services(4).

614  Hewlett-Packard Company -- 1 --  HP-UX 11i Version 2: December 2007 Update
NAME
mrinfo - Multicast Routing Configuration Information Tool

SYNOPSIS
/usr/sbin/mrinfo [-d debuglevel] [-r retries] [-t timeout] [ multicast-router ]

DESCRIPTION
mrinfo requests the configuration information from the multicast-router, and prints the information to the standard out. multicast-router can be either an IP address or a system name. mrinfo sends out the ASK_NEIGHBORS igmp message to the specified multicast-router, when the router receives the request, it sends back its configuration information. If the multicast-router is not specified, the request is sent the local router.

The configuration information for each interface is printed in the following format:

```
interface_addr -> neighbor_addr (neighbor_name) [metrics/thresh/flags]
```

If there are multiple neighbor routers on one interface, they will all be reported on the output. The possible values for flag are:

- **tunnel**: Neighbors are reached via tunnel.
- **srcrt**: The tunnel uses IP source routing.
- **down**: The interface is down.
- **disabled**: The interface is administratively disabled for multicast routing.
- **querier**: The local router is the querier of the subnet.

Please see mrouted(1M) for metrics and thresh.

The command line options are:

- **-d debuglevel**: Sets the level for printing out the debug message. The default is 0, only error and warning messages will be printed. Debug level three prints most the messages.
- **-r retries**: Sets the retry times to pull the routing daemon for information. The default is 3.
- **-t timeout**: Specifies the timeout value in seconds for waiting the reply. The default value is 4.

EXAMPLE
The following is an example of querying the multicasting configuration from the local routing daemon.

```
mrinfo
127.0.0.1 (localhost) [version 3.3]:
15.13.106.144 -> 15.13.106.145 (hpntcbs.cup.hp.com) [10/1/querier]
193.2.1.39 -> 0.0.0.0 (all-zeros-broadcast) [1/1/disabled]
15.13.106.144 -> 15.255.176.33 (matmos.hpl.hp.com) [10/1/tunnel]
15.13.106.144 -> 15.17.20.7 (hpspddc.vid.hp.com) [10/1/tunnel/down]
```

Note
mrinfo must be run as root.

AUTHOR
mrinfo was developed by Van Jacobson.

SEE ALSO
mrouted(1M), map-mbone(1M).
NAME
mrouterd - IP multicast routing daemon

SYNOPSIS
/usr/sbin/mrouter [-d] [ -c config_file ] [ -d debug_level ]

DESCRIPTION
The mrouterd command is an implementation of the Distance-Vector Multicast Routing Protocol (DVMRP), an earlier version of which is specified in RFC-1075. It maintains topological knowledge via a distance-vector routing protocol (like RIP, described in RFC-1058), upon which it implements a multicast datagram-forwarding algorithm called Reverse Path Multicasting.

mrouterd forwards a multicast datagram along a shortest (reverse) path tree rooted at the subnet on which the datagram originates. The multicast delivery tree may be thought of as a broadcast delivery tree that has been pruned back so that it does not extend beyond those subnetworks that have members of the destination group. Hence, datagrams are not forwarded along those branches which have no listeners of the multicast group. The IP time-to-live of a multicast datagram can be used to limit the range of multicast datagrams.

In order to support multicasting among subnets that are separated by (unicast) routers that do not support IP multicasting, mrouterd includes support for "tunnels", which are virtual point-to-point links between pairs of mrouterd s located anywhere in an internet. IP multicast packets are encapsulated for transmission through tunnels, so that they look like normal unicast datagrams to intervening routers and subnets. The encapsulation is added on entry to a tunnel and stripped off on exit from a tunnel. By default, the packets are encapsulated using the IP-in-IP protocol (IP protocol number 4).

The tunnelling mechanism allows mrouterd to establish a virtual internet for the purpose of multicasting only, which is independent of the physical internet and which may span multiple Autonomous Systems.

mrouterd handles multicast routing only; there may or may not be unicast routing software running on the same machine as mrouterd. With the use of tunnels, it is not necessary for mrouterd to have access to more than one physical subnet in order to perform multicast forwarding.

Invocation
If the -d option is not specified or if the debug level is specified as 0, mrouterd detaches from the invoking terminal. Otherwise, it remains attached to the invoking terminal and responsive to signals from that terminal. If -d is specified with no argument, the debug level defaults to 2. Regardless of the debug level, mrouterd always writes warning and error messages to the system log demon. Non-zero debug levels have the following effects:

level 1 all syslog messages are also printed to stderr.
level 2 all level 1 messages plus notifications of "significant" events are printed to stderr.
level 3 all level 2 messages plus notifications of all packet arrivals and departures are printed to stderr.

Upon startup, mrouterd writes its pid to the file /var/tmp/mrouterd.pid.

Configuration
mrouterd automatically configures itself to forward on all multicast-capable interfaces (i.e., interfaces that have the IFF_MULTICAST flag set, excluding the loopback "interface"). mrouterd finds other mrouterd s directly reachable via those interfaces. To override the default configuration or to add tunnel links to other mrouterd s, configuration commands may be placed in /etc/mrouterd.conf (or an alternative file, specified by the -c option). There are four types of configuration commands:
The phyint command can be used to disable multicast routing on the physical interface identified by local IP address <local-addr>, or to associate a non-default metric or threshold with the specified physical interface. The local IP address <local-addr> may be replaced by the interface name (such as lan0). If phyint is attached to multiple IP subnets, describe each additional subnet with the altnet option. phyint commands must precede tunnel commands.

The tunnel command can be used to establish a tunnel link between local IP address <local-addr> and remote IP address <remote-addr>, and to associate a non-default metric or threshold with that tunnel. The local IP address <local-addr> may be replaced by the interface name (such as lan0). The remote IP address <remote-addr> may be replaced by a host name, if and only if the host name has a single IP address associated with it. The tunnel must be set up in the mrouted.conf files of both routers before it can be used.

The cache_lifetime is a value that determines the amount of time that a cached multicast route stays in kernel before timing out. The value of this entry should lie between 300 (5 min) and 86400 (1 day). It defaults to 300.

The pruning command is provided for mrouted to act as a non-pruning router. It is also possible to start mrouted in a non-pruning mode using the -p option on the command line. It is expected that a router would be configured in this manner for test purposes only. The default mode is pruning enabled.

You may assign names to boundaries to make configuration easier with the name command. The boundary option on phyint or tunnel commands can accept either a name or a boundary.

The metric option is the "cost" associated with sending a datagram on the given interface or tunnel; it may be used to influence the choice of routes. The metric defaults to 1. Metrics should be kept as small as possible because mrouted cannot route along paths with a sum of metrics greater than 31. The threshold is the minimum IP time-to-live required for a multicast datagram to be forwarded to the given interface or tunnel. It is used to control the scope of multicast datagrams. (The TTL of forwarded packets is only compared to the threshold; it is not decremented by the threshold. Every multicast router decrements the TTL by 1.) The default threshold is 1.

In general, all mrouteds connected to a particular subnet or tunnel should use the same metric and threshold for that subnet or tunnel.

The rate_limit option allows the network administrator to specify a certain bandwidth in Kbits/second which would be allocated to multicast traffic. It defaults to 500Kbps on tunnels and 0 (unlimited) on physical interfaces.

The boundary option allows an interface to be configured as an administrative boundary for the specified scoped address. Packets belonging to this address will not be forwarded on a scoped interface. The boundary option accepts either a name or a boundary spec.

mrouted will not initiate execution if it has fewer than two enabled vifs (virtual interface), where a vif is either a physical multicast-capable interface or a tunnel. It will log a warning if all of its vifs are tunnels; such an mrouted configuration would be better replaced by more direct tunnels.
Example Configuration

This is an example configuration for a multicast router at a large school.

# mrouted.conf example
#
# Name our boundaries to make it easier
name LOCAL 239.255.0.0/16
name EE 239.254.0.0/16
#
# lan1 is our gateway to compsci, don't forward our
# local groups to them
phyint lan1 boundary EE
#
# lan2 is our interface on the classroom net, it has four
# different length subnets on it.
# note that you can use either an ip address or an
# interface name
phyint 172.16.12.38 boundary EE altnet 172.16.15.0/26
    altnet 172.16.15.128/26 altnet 172.16.48.0/24
#
# atm0 is our ATM interface, which doesn't properly
# support multicasting.
phyint atm0 disable
#
# This is an internal tunnel to another EE subnet
# Remove the default tunnel rate limit, since this
# tunnel is over ethernets
tunnel 192.168.5.4 192.168.55.101 metric 1 threshold 1
    rate_limit 0
#
# This is our tunnel to the outside world.
# Careful with those boundaries, Eugene.
tunnel 192.168.5.4 10.11.12.13 metric 1 threshold 32
    boundary LOCAL boundary EE

Signals

mrouted responds to the following signals:

HUP restarts mrouted. The configuration file is reread every time this signal is evoked.
INT terminates execution gracefully (i.e., by sending good-bye messages to all neighboring routers).
TERM same as INT
USR1 dumps the internal routing tables to /usr/tmp/mrouted.dump.
USR2 dumps the internal cache tables to /usr/tmp/mrouted.cache.
QUIT dumps the internal routing tables to stderr (only if mrouted was invoked with a non-zero
dump level).

For convenience in sending signals, mrouted writes its pid to /var/tmp/mrouted.pid upon startup.
EXAMPLES

The routing tables look like this:

<table>
<thead>
<tr>
<th>Virtual Interface Table</th>
<th>Metric</th>
<th>Thresh</th>
<th>Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vif Local-Address</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0 36.2.0.8</td>
<td>subnet: 36.2</td>
<td>1</td>
<td>querier</td>
</tr>
<tr>
<td></td>
<td>groups: 224.0.2.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>224.0.0.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>pkts in: 3456</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>pkts out: 2322323</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 36.11.0.1</td>
<td>subnet: 36.11</td>
<td>1</td>
<td>querier</td>
</tr>
<tr>
<td></td>
<td>groups: 224.0.2.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>224.0.1.0</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>224.0.0.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>pkts in: 345</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>pkts out: 3456</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 36.2.0.8</td>
<td>tunnel: 36.8.0.77</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>peers: 36.8.0.77 (2.2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>boundaries: 239.0.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>239.1.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>pkts in: 34545433</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>pkts out: 234342</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 36.2.0.8</td>
<td>tunnel: 36.6.8.23</td>
<td>3</td>
<td>16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Multicast Routing Table (1136 entries)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Origin-Subnet</td>
</tr>
<tr>
<td>----------------</td>
</tr>
<tr>
<td>36.2</td>
</tr>
<tr>
<td>36.8</td>
</tr>
<tr>
<td>36.11</td>
</tr>
</tbody>
</table>

In this example, there are four vifs connecting to two subnets and two tunnels. The vif 3 tunnel is not in use (no peer address). The vif 0 and vif 1 subnets have some groups present; tunnels never have any groups. This instance of mrouted is the one responsible for sending periodic group membership queries on the vif 0 and vif 1 subnets, as indicated by the "querier" flags. The list of boundaries indicate the scoped addresses on that interface. A count of the number of incoming and outgoing packets is also shown at each interface.

Associated with each subnet from which a multicast datagram can originate is the address of the previous hop router (unless the subnet is directly connected), the metric of the path back to the origin, the amount of time since an update was received for this subnet, the incoming vif for multicasts from that origin, and a list of outgoing vifs. The asterisk ( * ) indicates that the outgoing vif is connected to a leaf of the broadcast tree rooted at the origin, and a multicast datagram from that origin will be forwarded on that outgoing vif only if there are members of the destination group on that leaf.

The mrouted command also maintains a copy of the kernel forwarding cache table. Entries are created and deleted by mrouted.
The cache tables look like this:

<table>
<thead>
<tr>
<th>Origin</th>
<th>Multicast-group</th>
<th>CTmr</th>
<th>Age</th>
<th>Ptmr</th>
<th>IVif</th>
<th>Forwvifs</th>
</tr>
</thead>
<tbody>
<tr>
<td>13.2.116/22</td>
<td>224.2.127.255</td>
<td>3m</td>
<td>2m</td>
<td>-</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>&gt;13.2.116.19</td>
<td>&gt;13.2.116.19</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>138.96.48/21</td>
<td>224.2.127.255</td>
<td>5m</td>
<td>2m</td>
<td>-</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>&gt;138.96.48.108</td>
<td>&gt;138.96.48.108</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>128.9.160/20</td>
<td>224.2.127.255</td>
<td>3m</td>
<td>2m</td>
<td>-</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>&gt;128.9.160.45</td>
<td>&gt;128.9.160.45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>198.106.194/24</td>
<td>224.2.135.190</td>
<td>9m</td>
<td>28s</td>
<td>9m</td>
<td>0P</td>
<td></td>
</tr>
<tr>
<td>&gt;198.106.194.22</td>
<td>&gt;198.106.194.22</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Each entry is characterized by the origin subnet number, mask, and the destination multicast group. The **CTmr** field indicates the lifetime of the entry. The entry is deleted from the cache table when the timer decrements to zero. The **Age** field is the time since this cache entry was originally created. Since cache entries get refreshed if traffic is flowing, routing entries can grow very old. The **Ptmr** field is simply a dash if no prune was sent upstream, or the amount of time until the upstream prune will time out. The **Ivif** field indicates the incoming **vif** for multicast packets from that origin. Each router also maintains a record of the number of prunes received from neighboring routers for a particular source and group. If there are no members of a multicast group on any downward link of the multicast tree for a subnet, a prune message is sent to the upstream router. They are indicated by a **P** after the **vif** number. The **Forwvifs** field shows the interfaces along which datagrams belonging to the source-group are forwarded. A **p** indicates that no datagrams are being forwarded along that interface. An unlisted interface is a leaf subnet with no members of the particular group on that subnet. A **b** on an interface indicates that it is a boundary interface; that is, traffic will not be forwarded on the scoped address on that interface. An additional line with a **>** as the first character is printed for each source on the subnet. Note that there can be many sources in one subnet.

**AUTHORS**
Steve Deering, Ajit Thyagarajan, Bill Fenner.

**FILES**
/etc/mrouted.conf
/var/run/mrouted.pid
/var/tmp/mrouted.dump
/var/tmp/mrouted.cache

**SEE ALSO**
mrinfo(1M), map-mbone(1M).

DVMRP is described, along with other multicast routing algorithms, in the paper "Multicast Routing in Internetworks and Extended LANs" by S. Deering, in the Proceedings of the ACM SIGCOMM '88 Conference.
NAME
mtail - tails the mail log file

SYNOPSIS
/usr/sbin/mtail [n]

DESCRIPTION
mtail displays the last part of the mail log, typically /var/adm/syslog/mail.log. By default, it displays the last 20 lines of this log.

Options
n Display last n lines of /var/adm/syslog/mail.log instead of just 20.

SEE ALSO
sendmail(1M).
NAME
mvdif - move a directory

SYNOPSIS
/usr/sbin/mvdir dir newdir

DESCRIPTION
mvdir moves one directory tree into another existing directory (within the same file system), or renames a
directory without moving it.
dir must be an existing directory.

If newdir does not exist but the directory that would contain it does, dir is moved and/or renamed to
newdir. Otherwise, newdir must be an existing directory not already containing an entry with the same
name as the last pathname component of dir. In this case, dir is moved and becomes a subdirectory of
newdir. The last pathname component of dir is used as the name for the moved directory.

mvdir refuses to move dir if the path specified by newdir would be a descendent directory of the path
specified by dir. Such cases are not allowed because cyclic sub-trees would be created as in the case, for
example, of mvdir x/y x/y/z/t which is prohibited.

mvdir does not allow directory . to be moved.
Only users who have appropriate privilege can use mvdir.

EXTERNAL INFLUENCES
International Code Set Support
Single- and multi-byte character code sets are supported.

AUTHOR
mvdir was developed by OSF and HP.

SEE ALSO
cp(1), mkdir(1), mv(1).

STANDARDS CONFORMANCE
mvdir: SVID2, SVID3