

```

*****
17051 Sun Oct 28 17:19:47 2012
new/usr/src/man/man1m/Makefile
815 Need ipadm(1M) manual page
*****
1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet
9 # at http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright 2011, Richard Lowe
14 # Copyright 2012 Nexenta Systems, Inc. All rights reserved.
15 #
16 #
17 include ../../Makefile.master
18 #
19 MANSECT =      1m
20 #
21 COMMON_MANFILES =
22     6to4relay.1m      \
23     Intro.1m         \
24     Uutry.1m         \
25     accept.1m        \
26     acct.1m          \
27     acctadm.1m       \
28     acctcms.1m       \
29     acctcon.1m       \
30     acctmerg.1m      \
31     acctprc.1m       \
32     acctsh.1m        \
33     adbgen.1m        \
34     add_allocatable.1m \
35     add_drv.1m       \
36     addbadsec.1m     \
37     arp.1m           \
38     atohexlabel.1m   \
39     audit.1m         \
40     audit_warn.1m    \
41     auditconfig.1m   \
42     auditd.1m        \
43     auditrecord.1m   \
44     auditreduce.1m   \
45     auditstat.1m     \
46     automount.1m     \
47     automountd.1m    \
48     autopush.1m      \
49     bart.1m          \
50     beadm.1m         \
51     boot.1m          \
52     bootadm.1m       \
53     bootconfchk.1m   \
54     busstat.1m       \
55     cachefsd.1m      \
56     cachefslog.1m    \
57     cachefspack.1m   \
58     cachefsstat.1m   \
59     cachefswsize.1m  \
60     captainfo.1m     \
61     catman.1m        \
62     cfgadm.1m        \

```

```

62     cfgadm_ac.1m     \
63     cfgadm_cardbus.1m \
64     cfgadm_fp.1m     \
65     cfgadm_ib.1m     \
66     cfgadm_pci.1m    \
67     cfgadm_sata.1m   \
68     cfgadm_sbd.1m    \
69     cfgadm_scsi.1m   \
70     cfgadm_sdcard.1m \
71     cfgadm_sysctrl.1m \
72     cfgadm_usb.1m    \
73     cfsadmin.1m      \
74     chat.1m          \
75     check-hostname.1m \
76     check-permissions.1m \
77     chroot.1m        \
78     clear_locks.1m   \
79     clinfo.1m        \
80     clri.1m          \
81     consadm.1m       \
82     conv_lp.1m       \
83     conv_lpd.1m      \
84     coreadm.1m       \
85     cpustat.1m       \
86     cron.1m          \
87     cryptoadm.1m     \
88     datadm.1m        \
89     dd.1m            \
90     devattr.1m       \
91     devfree.1m       \
92     devfsadm.1m      \
93     device_remap.1m  \
94     devinfo.1m       \
95     devlinks.1m      \
96     devnm.1m         \
97     devprop.1m       \
98     devreserv.1m     \
99     df.1m            \
100    df_ufs.1m         \
101    dfmounts.1m       \
102    dfmounts_nfs.1m   \
103    dfshares.1m       \
104    dfshares_nfs.1m   \
105    dhcpagent.1m      \
106    dhcpconfig.1m     \
107    dhcpgmgr.1m       \
108    dhtadm.1m         \
109    disks.1m          \
110    diskscan.1m       \
111    dispadmin.1m      \
112    dladm.1m          \
113    dlmgmt.1m         \
114    dmesg.1m          \
115    dminfo.1m         \
116    dns-sd.1m         \
117    domainname.1m     \
118    drvconfig.1m      \
119    dsbitmap.1m       \
120    dscfg.1m          \
121    dscfgadm.1m       \
122    dscfglockd.1m     \
123    dsstat.1m         \
124    dsvclockd.1m     \
125    dtrace.1m         \
126    dumpadm.1m        \
127    editmap.1m        \

```

```

128          edquota.lm          \/
129          eeprom.lm           \/
130          embedded_su.lm      \/
131          etrn.lm              \/
132          fcinfo.lm           \/
133          fdetach.lm          \/
134          fdisk.lm             \/
135          ff.lm                \/
136          ff_ufs.lm           \/
137          fiocompress.lm      \/
138          flowadm.lm          \/
139          fmadm.lm            \/
140          fmd.lm               \/
141          fmdump.lm           \/
142          fmstat.lm           \/
143          fmthard.lm          \/
144          format.lm           \/
145          fruadm.lm           \/
146          fsck.lm             \/
147          fsck_cacheufs.lm    \/
148          fsck_pcfs.lm        \/
149          fsck_udfs.lm        \/
150          fsck_ufs.lm         \/
151          fsdb.lm              \/
152          fsdb_udfs.lm        \/
153          fsdb_ufs.lm         \/
154          fsirand.lm          \/
155          fssnap.lm           \/
156          fssnap_ufs.lm       \/
157          fsstat.lm           \/
158          fstyp.lm             \/
159          ftpaddhost.lm       \/
160          ftpconfig.lm        \/
161          ftprestart.lm       \/
162          ftpshut.lm          \/
163          fuser.lm            \/
164          fwflash.lm          \/
165          fwtmp.lm            \/
166          getdev.lm           \/
167          getdevpolicy.lm     \/
168          getdgrp.lm          \/
169          getent.lm           \/
170          gettable.lm         \/
171          getty.lm            \/
172          getvol.lm           \/
173          groupadd.lm         \/
174          groupdel.lm         \/
175          groupmod.lm         \/
176          growfs.lm           \/
177          gsscred.lm          \/
178          gssd.lm             \/
179          hal-device.lm       \/
180          hal-fdi-validate.lm \/
181          hal-find.lm         \/
182          hal-get-property.lm \/
183          hald.lm             \/
184          halt.lm             \/
185          hextoalabel.lm      \/
186          hostconfig.lm       \/
187          htable.lm           \/
188          ickey.lm            \/
189          id.lm                \/
190          idmap.lm            \/
191          idmapd.lm           \/
192          idsconfig.lm        \/
193          if_mpadm.lm         \/

```

```

194          ifconfig.lm        \/
195          ifparse.lm          \/
196          iiadm.lm           \/
197          iicpbmp.lm         \/
198          iicpsd.lm          \/
199          ikeadm.lm          \/
200          ikecert.lm         \/
201          in.chargend.lm      \/
202          in.comsat.lm        \/
203          in.daytimed.lm     \/
204          in.dhccpd.lm       \/
205          in.discardd.lm     \/
206          in.echod.lm        \/
207          in.fingerd.lm      \/
208          in.ftpd.lm          \/
209          in.iked.lm         \/
210          in.lpd.lm          \/
211          in.mpathd.lm       \/
212          in.ndpd.lm         \/
213          in.rarpd.lm        \/
214          in.rdisc.lm        \/
215          in.rexecd.lm       \/
216          in.ripngd.lm       \/
217          in.rlogind.lm      \/
218          in.routed.lm       \/
219          in.rshd.lm         \/
220          in.rwhod.lm        \/
221          in.talkd.lm        \/
222          in.telnetd.lm      \/
223          in.tftpd.lm        \/
224          in.timed.lm        \/
225          in.uucpd.lm        \/
226          inetadm.lm         \/
227          inetconv.lm        \/
228          inetd.lm           \/
229          infocmp.lm         \/
230          init.lm            \/
231          inityp21.lm         \/
232          install.lm         \/
233          installf.lm        \/
234          installgrub.lm     \/
235          intrd.lm           \/
236          intrstat.lm        \/
237          iostat.lm          \/
238          ipaddrsel.lm       \/
239          ipadm.lm           \/
240          ipf.lm              \/
241          ipfs.lm             \/
242          ipfstat.lm         \/
243          ipmon.lm           \/
244          ippstat.lm         \/
245          ipnat.lm           \/
246          ippool.lm          \/
247          ipqosconf.lm       \/
248          ipsecalgs.lm       \/
249          ipsecconf.lm       \/
250          ipseckey.lm        \/
251          iscsiadm.lm        \/
252          isns.lm            \/
253          isnsadm.lm         \/
254          itadm.lm           \/
255          k5srvutil.lm       \/
256          kadb.lm            \/
257          kadmin.lm          \/
258          kadmind.lm         \/
259          kclient.lm         \/

```

```

260 kdb5_ldap_util.1m //
261 kdb5_util.1m //
262 kdcmgr.1m //
263 kernel.1m //
264 keyser.1m //
265 killall.1m //
266 kprop.1m //
267 kpropd.1m //
268 kproplog.1m //
269 krb5kdc.1m //
270 ksslcfg.1m //
271 kstat.1m //
272 ktkk_warnd.1m //
273 labelit.1m //
274 labelit_hfs.1m //
275 labelit_udfs.1m //
276 labelit_ufs.1m //
277 latencytop.1m //
278 ldap_cachemgr.1m //
279 ldapaddent.1m //
280 ldapclient.1m //
281 link.1m //
282 listdgrp.1m //
283 listen.1m //
284 locator.1m //
285 lockfs.1m //
286 lockstat.1m //
287 lofiadm.1m //
288 logadm.1m //
289 logins.1m //
290 lpadmin.1m //
291 lpfilter.1m //
292 lpforms.1m //
293 lpget.1m //
294 lpmove.1m //
295 lpsched.1m //
296 lpset.1m //
297 lpshut.1m //
298 lpsystem.1m //
299 lpusers.1m //
300 luxadm.1m //
301 mail.local.1m //
302 makedbm.1m //
303 makemap.1m //
304 mdmonitor.1m //
305 medstat.1m //
306 metaclear.1m //
307 metadb.1m //
308 metadevadm.1m //
309 metahs.1m //
310 metaimport.1m //
311 metainit.1m //
312 metaoffline.1m //
313 metaparam.1m //
314 metarecover.1m //
315 metarename.1m //
316 metareplace.1m //
317 metaroot.1m //
318 metaset.1m //
319 metassist.1m //
320 metastat.1m //
321 metasync.1m //
322 metattach.1m //
323 mkdevalloc.1m //
324 mkdevmaps.1m //
325 mkfifo.1m //

```

```

326 mkfile.1m //
327 mkfs.1m //
328 mkfs_pcfs.1m //
329 mkfs_udfs.1m //
330 mkfs_ufs.1m //
331 mknod.1m //
332 mkpwdict.1m //
333 modinfo.1m //
334 modload.1m //
335 modunload.1m //
336 mount.1m //
337 mount_cachefs.1m //
338 mount_hfs.1m //
339 mount_nfs.1m //
340 mount_pcfs.1m //
341 mount_smbfs.1m //
342 mount_tmpfs.1m //
343 mount_udfs.1m //
344 mount_ufs.1m //
345 mountall.1m //
346 mountd.1m //
347 mpathadm.1m //
348 mpstat.1m //
349 msgid.1m //
350 mvdir.1m //
351 ncaconfd.1m //
352 ncheck.1m //
353 ncheck_ufs.1m //
354 ndd.1m //
355 ndmpadm.1m //
356 ndmpd.1m //
357 ndmpstat.1m //
358 netstat.1m //
359 netstrategy.1m //
360 newaliases.1m //
361 newfs.1m //
362 newkey.1m //
363 nfs4cbd.1m //
364 nfsd.1m //
365 nfslogd.1m //
366 nfsmapid.1m //
367 nfsstat.1m //
368 nlsadmin.1m //
369 nscadm.1m //
370 nscd.1m //
371 nwamd.1m //
372 passmgmt.1m //
373 pbind.1m //
374 picld.1m //
375 ping.1m //
376 pkgadd.1m //
377 pkgadm.1m //
378 pkgask.1m //
379 pkgchk.1m //
380 pkggrm.1m //
381 plockstat.1m //
382 pmadm.1m //
383 pmconfig.1m //
384 pntadm.1m //
385 polkit-is-privileged.1m //
386 pooladm.1m //
387 poolbind.1m //
388 poolcfg.1m //
389 pool.1m //
390 poolstat.1m //
391 ports.1m //

```

```

392 powerd.lm //
393 powertop.lm //
394 pppmgr.lm //
395 pppd.lm //
396 pppdump.lm //
397 pppoec.lm //
398 pppoed.lm //
399 pppstats.lm //
400 praudit.lm //
401 print-service.lm //
402 printmgr.lm //
403 projadd.lm //
404 projdel.lm //
405 projmod.lm //
406 prstat.lm //
407 prtconf.lm //
408 prtdiag.lm //
409 prtfru.lm //
410 prtpicl.lm //
411 prtvtoc.lm //
412 psradm.lm //
413 psrinfo.lm //
414 psrset.lm //
415 putdev.lm //
416 putdgrp.lm //
417 pwck.lm //
418 pwconv.lm //
419 quot.lm //
420 quota.lm //
421 quotacheck.lm //
422 quotaon.lm //
423 raidctl.lm //
424 ramdiskadm.lm //
425 rcapadm.lm //
426 rcapd.lm //
427 rctladm.lm //
428 rdate.lm //
429 reboot.lm //
430 rem_drv.lm //
431 remove_allocatable.lm //
432 removef.lm //
433 repquota.lm //
434 rmmount.lm //
435 rmt.lm //
436 rmvolmgr.lm //
437 roleadd.lm //
438 roledel.lm //
439 rolemod.lm //
440 root_archive.lm //
441 route.lm //
442 routeadm.lm //
443 rpc.bootparamd.lm //
444 rpc.mdcommd.lm //
445 rpc.metad.lm //
446 rpc.metamedd.lm //
447 rpc.metamhd.lm //
448 rpc.rexd.lm //
449 rpc.rstatd.lm //
450 rpc.rusersd.lm //
451 rpc.rwalld.lm //
452 rpc.smserverd.lm //
453 rpc.sprayd.lm //
454 rpc.yppasswdd.lm //
455 rpc.yppdated.lm //
456 rpcbind.lm //
457 rpcinfo.lm //

```

```

458 rquotad.lm //
459 rsh.lm //
460 rtc.lm //
461 rtquery.lm //
462 runacct.lm //
463 rwall.lm //
464 sac.lm //
465 sacadm.lm //
466 saf.lm //
467 sar.lm //
468 sasinfo.lm //
469 savecore.lm //
470 sbdadm.lm //
471 scadm.lm //
472 scmadm.lm //
473 sdpadm.lm //
474 sendmail.lm //
475 setuname.lm //
476 sftp-server.lm //
477 share.lm //
478 share_nfs.lm //
479 shareall.lm //
480 sharectl.lm //
481 sharemgr.lm //
482 showmount.lm //
483 shutdown.lm //
484 slpd.lm //
485 smbadm.lm //
486 smbd.lm //
487 smbiod.lm //
488 smbios.lm //
489 smbstat.lm //
490 smrsh.lm //
491 smtnrhdb.lm //
492 smtnrhtp.lm //
493 smtnzonecfg.lm //
494 sndradm.lm //
495 sndrd.lm //
496 sndrsyncd.lm //
497 snoop.lm //
498 soconfig.lm //
499 sppptun.lm //
500 spray.lm //
501 ssh-keysign.lm //
502 sshd.lm //
503 statd.lm //
504 stmfadm.lm //
505 stmsboot.lm //
506 strace.lm //
507 strclean.lm //
508 strerr.lm //
509 sttydefs.lm //
510 su.lm //
511 sulogin.lm //
512 svadm.lm //
513 svc.configd.lm //
514 svc.ipfd.lm //
515 svc.startd.lm //
516 svcadm.lm //
517 svccfg.lm //
518 swap.lm //
519 sync.lm //
520 syncinit.lm //
521 syncloop.lm //
522 syncstat.lm //
523 sysdef.lm //

```

```

524 syseventadm.lm //
525 syseventconfd.lm //
526 syseventd.lm //
527 syslogd.lm //
528 tapes.lm //
529 tcpd.lm //
530 tcpdchk.lm //
531 tcpdmatch.lm //
532 th_define.lm //
533 th_manage.lm //
534 tic.lm //
535 tnchkdb.lm //
536 tnctl.lm //
537 tnd.lm //
538 tninfo.lm //
539 tpmadm.lm //
540 traceroute.lm //
541 trapstat.lm //
542 ttyadm.lm //
543 ttymon.lm //
544 tuneufs.lm //
545 txzonemgr.lm //
546 tzselect.lm //
547 uadmin.lm //
548 ucodeadm.lm //
549 ufsdump.lm //
550 ufsrestore.lm //
551 unshare.lm //
552 unshare_nfs.lm //
553 update_drv.lm //
554 updatehome.lm //
555 useradd.lm //
556 userdel.lm //
557 usermod.lm //
558 utmpd.lm //
559 uucheck.lm //
560 uucico.lm //
561 uucleanup.lm //
562 uusched.lm //
563 uuxqt.lm //
564 vmstat.lm //
565 volcopy.lm //
566 volcopy_ufs.lm //
567 vscanadm.lm //
568 vscand.lm //
569 wall.lm //
570 wanboot_keygen.lm //
571 wanboot_keymgmt.lm //
572 wanboot_pl2split.lm //
573 wanbootutil.lm //
574 whodo.lm //
575 wificonfig.lm //
576 wpad.lm //
577 wracct.lm //
578 wusbadm.lm //
579 ypbind.lm //
580 ypinit.lm //
581 ypmake.lm //
582 ypmap2src.lm //
583 yppoll.lm //
584 yppush.lm //
585 ypserv.lm //
586 ypset.lm //
587 ypstart.lm //
588 ypxfr.lm //
589 zdb.lm //

```

```

590 zdump.lm //
591 zfs.lm //
592 zic.lm //
593 zoneadm.lm //
594 zoneadmd.lm //
595 zonecfg.lm //
596 zpool.lm //
597 zstreamdump.lm //

599 i386_MANFILES = lms.lm //
600 parted.lm //
601 mkntfs.lm //
602 ntfsclat.lm //
603 ntfsclone.lm //
604 ntfscluster.lm //
605 ntfscomp.lm //
606 ntfsfcp.lm //
607 ntfsfix.lm //
608 ntfsinfo.lm //
609 ntfslabel.lm //
610 ntfsls.lm //
611 ntfsprogs.lm //
612 ntfsresize.lm //
613 ntfsundelete.lm //

615 sparc_MANFILES = cvcd.lm //
616 dcs.lm //
617 drd.lm //
618 efdaemon.lm //
619 ldmad.lm //
620 monitor.lm //
621 obpsym.lm //
622 oplhd.lm //
623 prtdscp.lm //
624 sckmd.lm //
625 sf880drd.lm //
626 vntsd.lm //

628 MANSOFILES = acctcon1.lm //
629 acctcon2.lm //
630 acctdisk.lm //
631 acctdusg.lm //
632 accton.lm //
633 acctprcl.lm //
634 acctprc2.lm //
635 acctwtmp.lm //
636 bootparamd.lm //
637 chargefee.lm //
638 ckpacct.lm //
639 closewtmp.lm //
640 comsat.lm //
641 dcopy.lm //
642 devfsadmd.lm //
643 dodisk.lm //
644 fcadm.lm //
645 fingerd.lm //
646 ftpd.lm //
647 grpck.lm //
648 hal-find-by-capability.lm //
649 hal-find-by-property.lm //
650 hal-set-property.lm //
651 intro.lm //
652 kadmin.local.lm //
653 lastlogin.lm //
654 metadetch.lm //
655 metaonline.lm //

```

```

656 monacct.lm          \|
657 nulladm.lm         \|
658 poweroff.lm        \|
659 prctmp.lm          \|
660 prdaily.lm         \|
661 prtacct.lm         \|
662 quotaoff.lm        \|
663 rarpd.lm           \|
664 rdisc.lm           \|
665 reject.lm          \|
666 restricted_shell.lm \|
667 rexd.lm            \|
668 rexecd.lm          \|
669 rlogind.lm         \|
670 routed.lm         \|
671 rshd.lm            \|
672 rstatd.lm          \|
673 rusersd.lm         \|
674 rwalld.lm         \|
675 rwhod.lm           \|
676 sal.lm             \|
677 sa2.lm             \|
678 sadc.lm            \|
679 shutacct.lm       \|
680 sprayd.lm          \|
681 startup.lm         \|
682 talkd.lm           \|
683 telinit.lm         \|
684 telnetd.lm        \|
685 tftpd.lm           \|
686 turnacct.lm       \|
687 umount.lm          \|
688 umount_smbfs.lm   \|
689 umountall.lm      \|
690 unlink.lm          \|
691 unshareall.lm     \|
692 utmp2wtmp.lm      \|
693 uucpd.lm           \|
694 uutry.lm           \|
695 wtmpfix.lm        \|
696 yppasswd.lm       \|
697 ypstop.lm         \|
698 yppupdated.lm     \|
699 ypxfr_1perday.lm  \|
700 ypxfr_1perhour.lm \|
701 ypxfr_2perday.lm \|
702 ypxfrd.lm         \|

704 MANFILES =      $(COMMON_MANFILES)  ${$(MACH)_MANFILES}  $(MANSOFILES)

706 intro.lm         := SOSRC = man1m/Intro.lm

708 uutry.lm         := SOSRC = man1m/Uutry.lm

710 reject.lm        := SOSRC = man1m/accept.lm

712 acctdisk.lm      := SOSRC = man1m/acct.lm
713 acctdusg.lm      := SOSRC = man1m/acct.lm
714 accton.lm        := SOSRC = man1m/acct.lm
715 acctwtmp.lm      := SOSRC = man1m/acct.lm
716 closewtmp.lm    := SOSRC = man1m/acct.lm
717 utmp2wtmp.lm     := SOSRC = man1m/acct.lm

719 acctcon1.lm      := SOSRC = man1m/acctcon.lm
720 acctcon2.lm      := SOSRC = man1m/acctcon.lm

```

```

722 acctprc1.lm      := SOSRC = man1m/acctprc.lm
723 acctprc2.lm      := SOSRC = man1m/acctprc.lm

725 chargefee.lm    := SOSRC = man1m/acctsh.lm
726 ckpacct.lm      := SOSRC = man1m/acctsh.lm
727 dodisk.lm        := SOSRC = man1m/acctsh.lm
728 lastlogin.lm     := SOSRC = man1m/acctsh.lm
729 monacct.lm       := SOSRC = man1m/acctsh.lm
730 nulladm.lm       := SOSRC = man1m/acctsh.lm
731 prctmp.lm        := SOSRC = man1m/acctsh.lm
732 prdaily.lm       := SOSRC = man1m/acctsh.lm
733 prtacct.lm       := SOSRC = man1m/acctsh.lm
734 shutacct.lm     := SOSRC = man1m/acctsh.lm
735 startup.lm       := SOSRC = man1m/acctsh.lm
736 turnacct.lm      := SOSRC = man1m/acctsh.lm

738 dcopy.lm         := SOSRC = man1m/clri.lm

740 devfsadmd.lm     := SOSRC = man1m/devfsadm.lm

742 fcadm.lm         := SOSRC = man1m/fcinfo.lm

744 wtmpfix.lm       := SOSRC = man1m/fwtmp.lm

746 hal-find-by-capability.lm := SOSRC = man1m/hal-find.lm
747 hal-find-by-property.lm   := SOSRC = man1m/hal-find.lm

749 hal-set-property.lm      := SOSRC = man1m/hal-get-property.lm

751 poweroff.lm              := SOSRC = man1m/halt.lm

753 comsat.lm                := SOSRC = man1m/in.comsat.lm
754 fingerd.lm               := SOSRC = man1m/in.fingerd.lm
755 ftpd.lm                   := SOSRC = man1m/in.ftpd.lm
756 rarpd.lm                  := SOSRC = man1m/in.rarpd.lm
757 rdisc.lm                   := SOSRC = man1m/in.rdisc.lm
758 rexecd.lm                  := SOSRC = man1m/in.rexecd.lm
759 rlogind.lm                 := SOSRC = man1m/in.rlogind.lm
760 routed.lm                  := SOSRC = man1m/in.routed.lm
761 rshd.lm                     := SOSRC = man1m/in.rshd.lm
762 rwhod.lm                    := SOSRC = man1m/in.rwhod.lm
763 talkd.lm                    := SOSRC = man1m/in.talkd.lm
764 telnetd.lm                  := SOSRC = man1m/in.telnetd.lm
765 tftpd.lm                     := SOSRC = man1m/in.tftpd.lm
766 uucpd.lm                     := SOSRC = man1m/in.uucpd.lm

768 telinit.lm                 := SOSRC = man1m/init.lm

770 kadmin.local.lm            := SOSRC = man1m/kadmin.lm

772 unlink.lm                   := SOSRC = man1m/link.lm

774 metaonline.lm               := SOSRC = man1m/metaoffline.lm

776 metadetach.lm               := SOSRC = man1m/metattach.lm

778 umount.lm                    := SOSRC = man1m/mount.lm

780 umount_smbfs.lm             := SOSRC = man1m/mount_smbfs.lm

782 umountall.lm                := SOSRC = man1m/mountall.lm

784 grpck.lm                     := SOSRC = man1m/pwck.lm

786 quotaoff.lm                  := SOSRC = man1m/quotaoon.lm

```

new/usr/src/man/man1m/Makefile

13

```
788 bootparamd.lm      := SOSRC = man1m/rpc.bootparamd.lm
789 rexd.lm             := SOSRC = man1m/rpc.rexd.lm
790 rstatd.lm           := SOSRC = man1m/rpc.rstatd.lm
791 rusersd.lm          := SOSRC = man1m/rpc.rusersd.lm
792 rwalld.lm           := SOSRC = man1m/rpc.rwalld.lm
793 sprayd.lm           := SOSRC = man1m/rpc.sprayd.lm
794 yppasswdd.lm        := SOSRC = man1m/rpc.yppasswdd.lm
795 yppupdated.lm       := SOSRC = man1m/rpc.yppupdated.lm

797 restricted_shell.lm := SOSRC = man1m/rsh.lm

799 sa1.lm              := SOSRC = man1m/sar.lm
800 sa2.lm              := SOSRC = man1m/sar.lm
801 sadc.lm             := SOSRC = man1m/sar.lm

803 unshareall.lm      := SOSRC = man1m/shareall.lm

805 ypxfrd.lm          := SOSRC = man1m/ypserv.lm

807 ypstop.lm          := SOSRC = man1m/ypstart.lm

809 ypxfr_1perday.lm    := SOSRC = man1m/ypxfr.lm
810 ypxfr_1perhour.lm  := SOSRC = man1m/ypxfr.lm
811 ypxfr_2perday.lm   := SOSRC = man1m/ypxfr.lm

814 .KEEP_STATE:

816 include ../Makefile.man

818 install: $(ROOTMANFILES)
```

110562 Sun Oct 28 17:19:47 2012

new/usr/src/man/man1m/dladm.1m

815 Need ipadm(1M) manual page

```

1 \" te
2.\" Copyright (c) 2008, Sun Microsystems, Inc. All Rights Reserved
3.\" Sun Microsystems, Inc. gratefully acknowledges The Open Group for permission
4.\" The Institute of Electrical and Electronics Engineers and The Open Group, ha
5.\" are reprinted and reproduced in electronic form in the Sun OS Reference Manu
6.\" and Electronics Engineers, Inc and The Open Group. In the event of any discr
7.\" This notice shall appear on any product containing this material.
8.\" The contents of this file are subject to the terms of the Common Development
9.\" See the License for the specific language governing permissions and limitati
10.\" fields enclosed by brackets \"[]\" replaced with your own identifying informat
11.TH DLADM 1M \"Sep 23, 2009\"
12.SH NAME
13.dladm - administer data links
14.SH SYNOPSIS
15.LP
16.nf
17.\fBdladm show-link\fR [\fB-P\fR] [\fB-s\fR [\fB-i\fR \fIinterval\fR]] [[\fB-p\fR
18.\fBdladm rename-link\fR [\fB-R\fR \fIroot-dir\fR] \fIlink\fR \fInew-link\fR
19.fi
21.LP
22.nf
23.\fBdladm delete-phys\fR \fIphys-link\fR
24.\fBdladm show-phys\fR [\fB-P\fR] [\fB-m\fR] [[\fB-p\fR] \fB-o\fR \fIfield\fR[,...]]
25.fi
27.LP
28.nf
29.\fBdladm create-aggr\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] [\fB-P\fR \fIpolicy
30. [\fB-T\fR \fItime\fR] [\fB-u\fR \fIaddress\fR] \fB-l\fR \fIether-link1\fR [
31.\fBdladm modify-aggr\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] [\fB-P\fR \fIpolicy
32. [\fB-T\fR \fItime\fR] [\fB-u\fR \fIaddress\fR] \fIaggr-link\fR
33.\fBdladm delete-aggr\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] \fIaggr-link\fR
34.\fBdladm add-aggr\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] \fB-l\fR \fIether-link
35. \fIaggr-link\fR
36.\fBdladm remove-aggr\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] \fB-l\fR \fIether-1
37. \fIaggr-link\fR
38.\fBdladm show-aggr\fR [\fB-PLx\fR] [\fB-s\fR [\fB-i\fR \fIinterval\fR]] [[\fB-p\fR
39. [\fIaggr-link\fR]
40.fi
42.LP
43.nf
44.\fBdladm create-bridge\fR [\fB-P\fR \fIprotect\fR] [[\fB-R\fR \fIroot-dir\fR] [[\f
45. [\fB-m\fR \fIimax-age\fR] [[\fB-h\fR \fIhello-time\fR] [\fB-d\fR \fIforward-d
46. [\fB-l\fR \fIlink\fR...] \fIbridge-name\fR
47.fi
49.LP
50.nf
51.\fBdladm modify-bridge\fR [\fB-P\fR \fIprotect\fR] [[\fB-R\fR \fIroot-dir\fR] [[\f
52. [\fB-m\fR \fIimax-age\fR] [[\fB-h\fR \fIhello-time\fR] [\fB-d\fR \fIforward-d
53. \fIbridge-name\fR
54.fi
56.LP
57.nf
58.\fBdladm delete-bridge\fR [\fB-R\fR \fIroot-dir\fR] \fIbridge-name\fR
59.fi
61.LP

```

```

62.nf
63.\fBdladm add-bridge\fR [\fB-R\fR \fIroot-dir\fR] \fB-l\fR \fIlink\fR [\fB-l\fR \f
64.fi
66.LP
67.nf
68.\fBdladm remove-bridge\fR [\fB-R\fR \fIroot-dir\fR] \fB-l\fR \fIlink\fR [\fB-l\f
69.fi
71.LP
72.nf
73.\fBdladm show-bridge\fR [\fB-flt\fR] [[\fB-s\fR [\fB-i\fR \fIinterval\fR]] [[\fB-
74. [\fIbridge-name\fR]
75.fi
77.LP
78.nf
79.\fBdladm create-vlan\fR [\fB-ft\fR] [\fB-R\fR \fIroot-dir\fR] \fB-l\fR \fIether-
80.\fBdladm delete-vlan\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] \fIvlan-link\fR
81.\fBdladm show-vlan\fR [\fB-P\fR] [[\fB-p\fR] \fB-o\fR \fIfield\fR[,...]] [\fIvla
82.fi
84.LP
85.nf
86.\fBdladm scan-wifi\fR [[\fB-p\fR] \fB-o\fR \fIfield\fR[,...]] [\fIwifi-link\fR]
87.\fBdladm connect-wifi\fR [\fB-e\fR \fIessid\fR] [\fB-i\fR \fIbssid\fR] [\fB-k\fR
88. [\fB-s\fR none | wep | wpa ] [\fB-a\fR open | shared] [\fB-b\fR bss | ibss]
89. [\fB-m\fR a | b | g] [\fB-T\fR \fItime\fR] [\fIwifi-link\fR]
90.\fBdladm disconnect-wifi\fR [\fB-a\fR] [\fIwifi-link\fR]
91.\fBdladm show-wifi\fR [[\fB-p\fR] \fB-o\fR \fIfield\fR[,...]] [\fIwifi-link\fR]
92.fi
94.LP
95.nf
96.\fBdladm show-ether\fR [\fB-x\fR] [[\fB-p\fR] \fB-o\fR \fIfield\fR[,...]] [\fIet
97.fi
99.LP
100.nf
101.\fBdladm set-linkprop\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] \fB-p\fR \fIprop\f
102.\fBdladm reset-linkprop\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] [\fB-p\fR \fIpro
103.\fBdladm show-linkprop\fR [\fB-P\fR] [[\fB-c\fR] \fB-o\fR \fIfield\fR[,...]] [\f
104.fi
106.LP
107.nf
108.\fBdladm create-secobj\fR [\fB-t\fR] [[\fB-R\fR \fIroot-dir\fR] [\fB-f\fR \fIfile
109.\fBdladm delete-secobj\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] \fIsecobj\fR[,...
110.\fBdladm show-secobj\fR [\fB-P\fR] [[\fB-p\fR] \fB-o\fR \fIfield\fR[,...]] [\fIs
111.fi
113.LP
114.nf
115.\fBdladm create-vnic\fR [\fB-t\fR] \fB-l\fR \fIlink\fR [\fB-R\fR \fIroot-dir\fR]
116. {factory \fB-n\fR \fIslot-identifier\fR} | {random [\fB-r\fR \fIprefix\fR]
117. [\fB-v\fR \fIvlan-id\fR] [\fB-p\fR \fIprop\fR=\fIvalue\fR[,...]] \fIvnic-li
118.\fBdladm delete-vnic\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] \fIvnic-link\fR
119.\fBdladm show-vnic\fR [\fB-pP\fR] [\fB-s\fR [\fB-i\fR \fIinterval\fR]] [[\fB-o\fR
120. [\fB-l\fR \fIlink\fR] \fIvnic-link\fR]
121.fi
123.LP
124.nf
125.\fBdladm create-etherstub\fR [\fB-t\fR] [[\fB-R\fR \fIroot-dir\fR] \fIetherstub\f
126.\fBdladm delete-etherstub\fR [\fB-t\fR] [[\fB-R\fR \fIroot-dir\fR] \fIetherstub\f
127.\fBdladm show-etherstub\fR [\fIetherstub\fR]

```



```

128 .fi
129 .LP
130 .nf
131 .nf
132 \fBdladm create-iptun\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] \fB-T\fR \fItype\f
133 \fIiptun-link\fR
134 \fBdladm modify-iptun\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] [\fB-s\fR \fIsrc\f
135 \fBdladm delete-iptun\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] \fIiptun-link\fR
136 \fBdladm show-iptun\fR [\fB-P\fR] [[\fB-p\fR] \fB-o\fR \fIfield\fR[,...]] [\fIip
137 .fi

139 .LP
140 .nf
141 \fBdladm show-usage\fR [\fB-a\fR] \fB-f\fR \fIfilename\fR [\fB-p\fR \fIplotfile\f
142 [\fB-e\fR \fItime\fR] [\fIlink\fR]
143 .fi

145 .SH DESCRIPTION
146 .sp
147 .LP
148 The \fBdladm\fR command is used to administer data-links. A data-link is
149 represented in the system as a \fBSTREAMS DLPI\fR (v2) interface which can be
150 plumbed under protocol stacks such as \fBTCP/IP\fR. Each data-link relies on
151 either a single network device or an aggregation of devices to send packets to
152 or receive packets from a network.
153 .sp
154 .LP
155 Each \fBdladm\fR subcommand operates on one of the following objects:
156 .sp
157 .ne 2
158 .na
159 \fB\blink\fR
160 .ad
161 .sp .6
162 .RS 4n
163 A datalink, identified by a name. In general, the name can use any alphanumeric
164 characters (or the underscore, \fB_\fR), but must start with an alphabetic
165 character and end with a number. A datalink name can be at most 31 characters,
166 and the ending number must be between 0 and 4294967294 (inclusive). The ending
167 number must not begin with a zero. Datalink names between 3 and 8 characters
168 are recommended.
169 .sp
170 Some subcommands operate only on certain types or classes of datalinks. For
171 those cases, the following object names are used:
172 .sp
173 .ne 2
174 .na
175 \fB\bphys-link\fR
176 .ad
177 .sp .6
178 .RS 4n
179 A physical datalink.
180 .RE

182 .sp
183 .ne 2
184 .na
185 \fB\bvlan-link\fR
186 .ad
187 .sp .6
188 .RS 4n
189 A VLAN datalink.
190 .RE

192 .sp
193 .ne 2

```

```

194 .na
195 \fB\baggr-link\fR
196 .ad
197 .sp .6
198 .RS 4n
199 An aggregation datalink (or a key; see NOTES).
200 .RE

202 .sp
203 .ne 2
204 .na
205 \fB\bether-link\fR
206 .ad
207 .sp .6
208 .RS 4n
209 A physical Ethernet datalink.
210 .RE

212 .sp
213 .ne 2
214 .na
215 \fB\bwifi-link\fR
216 .ad
217 .sp .6
218 .RS 4n
219 A WiFi datalink.
220 .RE

222 .sp
223 .ne 2
224 .na
225 \fB\bvnic-link\fR
226 .ad
227 .sp .6
228 .RS 4n
229 A virtual network interface created on a link or an \fB\betherstub\fR. It is a
230 pseudo device that can be treated as if it were a network interface card on a
231 machine.
232 .RE

234 .sp
235 .ne 2
236 .na
237 \fB\bip-tun-link\fR
238 .ad
239 .sp .6
240 .RS 4n
241 An IP tunnel link.
242 .RE

244 .RE

246 .sp
247 .ne 2
248 .na
249 \fB\bdev\fR
250 .ad
251 .sp .6
252 .RS 4n
253 A network device, identified by concatenation of a driver name and an instance
254 number.
255 .RE

257 .sp
258 .ne 2
259 .na

```

```

260 \fB\fBetherstub\fR\fR
261 .ad
262 .sp .6
263 .RS 4n
264 An Ethernet stub can be used instead of a physical NIC to create VNICs. VNICs
265 created on an \fBetherstub\fR will appear to be connected through a virtual
266 switch, allowing complete virtual networks to be built without physical
267 hardware.
268 .RE

270 .sp
271 .ne 2
272 .na
273 \fB\fBbridge\fR\fR
274 .ad
275 .sp .6
276 .RS 4n
277 A bridge instance, identified by an administratively-chosen name. The name may
278 use any alphanumeric characters or the underscore, \fB_\fR, but must start and
279 end with an alphabetic character. A bridge name can be at most 31 characters.
280 The name \fBdefault\fR is reserved, as are all names starting with \fBSUNW\fR.
281 .sp
282 Note that appending a zero (\fB0\fR) to a bridge name produces a valid link
283 name, used for observability.
284 .RE

286 .sp
287 .ne 2
288 .na
289 \fB\fBsecobj\fR\fR
290 .ad
291 .sp .6
292 .RS 4n
293 A secure object, identified by an administratively-chosen name. The name can
294 use any alphanumeric characters, as well as underscore (\fB_\fR), period
295 (\fB&.\fR), and hyphen (\fB-\fR). A secure object name can be at most 32
296 characters.
297 .RE

299 .SS "Options"
300 .sp
301 .LP
302 Each \fBdladm\fR subcommand has its own set of options. However, many of the
303 subcommands have the following as a common option:
304 .sp
305 .ne 2
306 .na
307 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
308 .ad
309 .sp .6
310 .RS 4n
311 Specifies an alternate root directory where the operation-such as creation,
312 deletion, or renaming-should apply.
313 .RE

315 .SS "SUBCOMMANDS"
316 .sp
317 .LP
318 The following subcommands are supported:
319 .sp
320 .ne 2
321 .na
322 \fB\fBdladm show-link\fR [\fB-P\fR] [\fB-s\fR [\fB-i\fR \fIinterval\fR]]
323 [[\fB-p\fR] \fB-o\fR \fIfield\fR[...]][\fB-l\fR \fIlink\fR]\fR
324 .ad
325 .sp .6

```

```

326 .RS 4n
327 Show link configuration information (the default) or statistics, either for all
328 datalinks or for the specified link \fB-l\fR. By default, the system is
329 configured with one datalink for each known network device.
330 .sp
331 .ne 2
332 .na
333 \fB\fB-o\fR \fIfield\fR[...], \fB--output\fR=\fIfield\fR[...]\fR
334 .ad
335 .sp .6
336 .RS 4n
337 A case-insensitive, comma-separated list of output fields to display. When not
338 modified by the \fB-s\fR option (described below), the field name must be one
339 of the fields listed below, or the special value \fBall\fR to display all
340 fields. By default (without \fB-o\fR), \fBshow-link\fR displays all fields.
341 .sp
342 .ne 2
343 .na
344 \fB\fBBLINK\fR\fR
345 .ad
346 .sp .6
347 .RS 4n
348 The name of the datalink.
349 .RE

351 .sp
352 .ne 2
353 .na
354 \fB\fBCLASS\fR\fR
355 .ad
356 .sp .6
357 .RS 4n
358 The class of the datalink. \fBdladm\fR distinguishes between the following
359 classes:
360 .sp
361 .ne 2
362 .na
363 \fB\fBphys\fR\fR
364 .ad
365 .sp .6
366 .RS 4n
367 A physical datalink. The \fBshow-phys\fR subcommand displays more detail for
368 this class of datalink.
369 .RE

371 .sp
372 .ne 2
373 .na
374 \fB\fBBaggr\fR\fR
375 .ad
376 .sp .6
377 .RS 4n
378 An IEEE 802.3ad link aggregation. The \fBshow-aggr\fR subcommand displays more
379 detail for this class of datalink.
380 .RE

382 .sp
383 .ne 2
384 .na
385 \fB\fBvlan\fR\fR
386 .ad
387 .sp .6
388 .RS 4n
389 A VLAN datalink. The \fBshow-vlan\fR subcommand displays more detail for this
390 class of datalink.
391 .RE

```

```

393 .sp
394 .ne 2
395 .na
396 \fB\fBvnic\fR\fR
397 .ad
398 .sp .6
399 .RS 4n
400 A virtual network interface. The \fBshow-vnic\fR subcommand displays more
401 detail for this class of datalink.
402 .RE

404 .RE

406 .sp
407 .ne 2
408 .na
409 \fB\fBMTU\fR\fR
410 .ad
411 .sp .6
412 .RS 4n
413 The maximum transmission unit size for the datalink being displayed.
414 .RE

416 .sp
417 .ne 2
418 .na
419 \fB\fBSTATE\fR\fR
420 .ad
421 .sp .6
422 .RS 4n
423 The link state of the datalink. The state can be \fBup\fR, \fBdown\fR, or
424 \fBunknown\fR.
425 .RE

427 .sp
428 .ne 2
429 .na
430 \fB\fBBRIDGE\fR\fR
431 .ad
432 .sp .6
433 .RS 4n
434 The name of the bridge to which this link is assigned, if any.
435 .RE

437 .sp
438 .ne 2
439 .na
440 \fB\fBBOVER\fR\fR
441 .ad
442 .sp .6
443 .RS 4n
444 The physical datalink(s) over which the datalink is operating. This applies to
445 \fBagggr\fR, \fBbridge\fR, and \fBvlan\fR classes of datalinks. A VLAN is
446 created over a single physical datalink, a bridge has multiple attached links,
447 and an aggregation is comprised of one or more physical datalinks.
448 .RE

450 When the \fB-o\fR option is used in conjunction with the \fB-s\fR option, used
451 to display link statistics, the field name must be one of the fields listed
452 below, or the special value \fBall\fR to display all fields
453 .sp
454 .ne 2
455 .na
456 \fB\fBBLINK\fR\fR
457 .ad

```

```

458 .sp .6
459 .RS 4n
460 The name of the datalink.
461 .RE

463 .sp
464 .ne 2
465 .na
466 \fB\fBIPACKETS\fR\fR
467 .ad
468 .sp .6
469 .RS 4n
470 Number of packets received on this link.
471 .RE

473 .sp
474 .ne 2
475 .na
476 \fB\fBRYTES\fR\fR
477 .ad
478 .sp .6
479 .RS 4n
480 Number of bytes received on this link.
481 .RE

483 .sp
484 .ne 2
485 .na
486 \fB\fBIERRORS\fR\fR
487 .ad
488 .sp .6
489 .RS 4n
490 Number of input errors.
491 .RE

493 .sp
494 .ne 2
495 .na
496 \fB\fBOPACKETS\fR\fR
497 .ad
498 .sp .6
499 .RS 4n
500 Number of packets sent on this link.
501 .RE

503 .sp
504 .ne 2
505 .na
506 \fB\fBBYTES\fR\fR
507 .ad
508 .sp .6
509 .RS 4n
510 Number of bytes received on this link.
511 .RE

513 .sp
514 .ne 2
515 .na
516 \fB\fBOERRORS\fR\fR
517 .ad
518 .sp .6
519 .RS 4n
520 Number of output errors.
521 .RE

523 .RE

```

```

525 .sp
526 .ne 2
527 .na
528 \fB\fB-p\fR, \fB--parseable\fR\fR
529 .ad
530 .sp .6
531 .RS 4n
532 Display using a stable machine-parseable format. The \fB-o\fR option is
533 required with \fB-p\fR. See "Parseable Output Format", below.
534 .RE

536 .sp
537 .ne 2
538 .na
539 \fB\fB-P\fR, \fB--persistent\fR\fR
540 .ad
541 .sp .6
542 .RS 4n
543 Display the persistent link configuration.
544 .RE

546 .sp
547 .ne 2
548 .na
549 \fB\fB-s\fR, \fB--statistics\fR\fR
550 .ad
551 .sp .6
552 .RS 4n
553 Display link statistics.
554 .RE

556 .sp
557 .ne 2
558 .na
559 \fB\fB-i\fR \fIinterval\fR, \fB--interval\fR=\fIinterval\fR\fR
560 .ad
561 .sp .6
562 .RS 4n
563 Used with the \fB-s\fR option to specify an interval, in seconds, at which
564 statistics should be displayed. If this option is not specified, statistics
565 will be displayed only once.
566 .RE

568 .RE

570 .sp
571 .ne 2
572 .na
573 \fB\fBdladm rename-link\fR [\fB-R\fR \fIroot-dir\fR] \fIlink\fR
574 \fInew-link\fR\fR
575 .ad
576 .sp .6
577 .RS 4n
578 Rename \fIlink\fR to \fInew-link\fR. This is used to give a link a meaningful
579 name, or to associate existing link configuration such as link properties of a
580 removed device with a new device. See the \fBEXAMPLES\fR section for specific
581 examples of how this subcommand is used.
582 .sp
583 .ne 2
584 .na
585 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
586 .ad
587 .sp .6
588 .RS 4n
589 See "Options," above.

```

```

590 .RE

592 .RE

594 .sp
595 .ne 2
596 .na
597 \fB\fBdladm delete-phys\fR \fIphys-link\fR\fR
598 .ad
599 .sp .6
600 .RS 4n
601 This command is used to delete the persistent configuration of a link
602 associated with physical hardware which has been removed from the system. See
603 the \fBEXAMPLES\fR section.
604 .RE

606 .sp
607 .ne 2
608 .na
609 \fB\fBdladm show-phys\fR [\fB-P\fR] [[\fB-p\fR] \fB-o\fR \fIfield\fR[,...]]
610 [\fB-H\fR] [\fIphys-link\fR]\fR
611 .ad
612 .sp .6
613 .RS 4n
614 Show the physical device and attributes of all physical links, or of the named
615 physical link. Without \fB-P\fR, only physical links that are available on the
616 running system are displayed.
617 .sp
618 .ne 2
619 .na
620 \fB\fB-H\fR\fR
621 .ad
622 .sp .6
623 .RS 4n
624 Show hardware resource usage, as returned by the NIC driver. Output from
625 \fB-H\fR displays the following elements:
626 .sp
627 .ne 2
628 .na
629 \fB\fBBLINK\fR\fR
630 .ad
631 .sp .6
632 .RS 4n
633 A physical device corresponding to a NIC driver.
634 .RE

636 .sp
637 .ne 2
638 .na
639 \fB\fBGROUP\fR\fR
640 .ad
641 .sp .6
642 .RS 4n
643 A collection of rings.
644 .RE

646 .sp
647 .ne 2
648 .na
649 \fB\fBGROUPTYPE\fR\fR
650 .ad
651 .sp .6
652 .RS 4n
653 RX or TX. All rings in a group are of the same group type.
654 .RE

```

```

656 .sp
657 .ne 2
658 .na
659 \fB\fBRINGS\fR\fR
660 .ad
661 .sp .6
662 .RS 4n
663 A hardware resource used by a data link, subject to assignment by a driver to
664 different groups.
665 .RE

667 .RE

677 .RE

679 .sp
680 .ne 2
681 .na
682 \fB\fB-o\fR \fIfield\fR, \fB--output\fR=\fIfield\fR\fR
683 .ad
684 .sp .6
685 .RS 4n
686 A case-insensitive, comma-separated list of output fields to display. The field
687 name must be one of the fields listed below, or the special value \fBall\fR, to
688 display all fields. For each link, the following fields can be displayed:
689 .sp
690 .ne 2
691 .na
692 \fB\fBLINK\fR\fR
693 .ad
694 .sp .6
695 .RS 4n
696 The name of the datalink.
697 .RE

699 .sp
700 .ne 2
701 .na
702 \fB\fBMEDIA\fR\fR
703 .ad
704 .sp .6
705 .RS 4n
706 The media type provided by the physical datalink.
707 .RE

709 .sp
710 .ne 2
711 .na
712 \fB\fBSTATE\fR\fR
713 .ad
714 .sp .6
715 .RS 4n
716 The state of the link. This can be \fBup\fR, \fBdown\fR, or \fBunknown\fR.
717 .RE

719 .sp
720 .ne 2
721 .na

```

```

722 \fB\fBSPEED\fR\fR
723 .ad
724 .sp .6
725 .RS 4n
726 The current speed of the link, in megabits per second.
727 .RE

729 .sp
730 .ne 2
731 .na
732 \fB\fBDUPLEX\fR\fR
733 .ad
734 .sp .6
735 .RS 4n
736 For Ethernet links, the full/half duplex status of the link is displayed if the
737 link state is \fBup\fR. The duplex is displayed as \fBunknown\fR in all other
738 cases.
739 .RE

741 .sp
742 .ne 2
743 .na
744 \fB\fBDEVICE\fR\fR
745 .ad
746 .sp .6
747 .RS 4n
748 The name of the physical device under this link.
749 .RE

751 .RE

753 .sp
754 .ne 2
755 .na
756 \fB\fB-p\fR, \fB--parseable\fR\fR
757 .ad
758 .sp .6
759 .RS 4n
760 Display using a stable machine-parseable format. The \fB-o\fR option is
761 required with \fB-p\fR. See "Parseable Output Format", below.
762 .RE

764 .sp
765 .ne 2
766 .na
767 \fB\fB-P\fR, \fB--persistent\fR\fR
768 .ad
769 .sp .6
770 .RS 4n
771 This option displays persistent configuration for all links, including those
772 that have been removed from the system. The output provides a \fBFLAGS\fR
773 column in which the \fB-r\fR flag indicates that the physical device associated
774 with a physical link has been removed. For such links, \fBdelete-phys\fR can be
775 used to purge the link's configuration from the system.
776 .RE

778 .RE

780 .sp
781 .ne 2
782 .na
783 \fB\fBdladm create-aggr\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] [\fB-P\fR
784 \fIpolicy\fR] [\fB-L\fR \fImode\fR] [\fB-T\fR \fItime\fR] [\fB-u\fR
785 \fIaddress\fR] \fB-l\fR \fIether-link1\fR [\fB-l\fR \fIether-link2\fR...]
786 \fIaggr-link\fR\fR
787 .ad

```

```

788 .sp .6
789 .RS 4n
790 Combine a set of links into a single IEEE 802.3ad link aggregation named
791 \fIaggr-link\fR. The use of an integer \fIkey\fR to generate a link name for
792 the aggregation is also supported for backward compatibility. Many of the
793 \fB*\fR\fB-aggr\fR subcommands below also support the use of a \fIkey\fR to
794 refer to a given aggregation, but use of the aggregation link name is
795 preferred. See the \fBNOTES\fR section for more information on keys.
796 .sp
797 \fBdladm\fR supports a number of port selection policies for an aggregation of
798 ports. (See the description of the \fB-P\fR option, below.) If you do not
799 specify a policy, \fBcreate-aggr\fR uses the default, the L4 policy, described
800 under the \fB-P\fR option.
801 .sp
802 .ne 2
803 .na
804 \fB\fB-l\fR \fIether-link\fR, \fB--link\fR=\fIether-link\fR\fR
805 .ad
806 .sp .6
807 .RS 4n
808 Each Ethernet link (or port) in the aggregation is specified using an \fB-l\fR
809 option followed by the name of the link to be included in the aggregation.
810 Multiple links are included in the aggregation by specifying multiple \fB-l\fR
811 options. For backward compatibility with previous versions of Solaris, the
812 \fBdladm\fR command also supports the using the \fB-d\fR option (or
813 \fB--dev\fR) with a device name to specify links by their underlying device
814 name. The other \fB*\fR\fB-aggr\fR subcommands that take \fB-l\fR options also
815 accept \fB-d\fR.
816 .RE

818 .sp
819 .ne 2
820 .na
821 \fB\fB-t\fR, \fB--temporary\fR\fR
822 .ad
823 .sp .6
824 .RS 4n
825 Specifies that the aggregation is temporary. Temporary aggregations last until
826 the next reboot.
827 .RE

829 .sp
830 .ne 2
831 .na
832 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
833 .ad
834 .sp .6
835 .RS 4n
836 See "Options," above.
837 .RE

839 .sp
840 .ne 2
841 .na
842 \fB\fB-P\fR \fIpolicy\fR, \fB--policy\fR=\fIpolicy\fR\fR
843 .ad
844 .br
845 .na
846 \fB\fR
847 .ad
848 .sp .6
849 .RS 4n
850 Specifies the port selection policy to use for load spreading of outbound
851 traffic. The policy specifies which \fIdev\fR object is used to send packets. A
852 policy is a list of one or more layers specifiers separated by commas. A layer
853 specifier is one of the following:

```

```

854 .sp
855 .ne 2
856 .na
857 \fB\fBL2\fR\fR
858 .ad
859 .sp .6
860 .RS 4n
861 Select outbound device according to source and destination \fBMAC\fR addresses
862 of the packet.
863 .RE

865 .sp
866 .ne 2
867 .na
868 \fB\fBL3\fR\fR
869 .ad
870 .sp .6
871 .RS 4n
872 Select outbound device according to source and destination \fBIP\fR addresses
873 of the packet.
874 .RE

876 .sp
877 .ne 2
878 .na
879 \fB\fBL4\fR\fR
880 .ad
881 .sp .6
882 .RS 4n
883 Select outbound device according to the upper layer protocol information
884 contained in the packet. For \fBTCp\fR and \fBUDP\fR, this includes source and
885 destination ports. For IPsec, this includes the \fBSPi\fR (Security Parameters
886 Index).
887 .RE

889 For example, to use upper layer protocol information, the following policy can
890 be used:
891 .sp
892 .in +2
893 .nf
894 -P L4
895 .fi
896 .in -2
897 .sp

899 Note that policy L4 is the default.
900 .sp
901 To use the source and destination \fBMAC\fR addresses as well as the source and
902 destination \fBIP\fR addresses, the following policy can be used:
903 .sp
904 .in +2
905 .nf
906 -P L2,L3
907 .fi
908 .in -2
909 .sp

911 .RE

913 .sp
914 .ne 2
915 .na
916 \fB\fB-L\fR \fImode\fR, \fB--laccp-mode\fR=\fImode\fR\fR
917 .ad
918 .sp .6
919 .RS 4n

```

```

920 Specifies whether \fBLACP\fR should be used and, if used, the mode in which it
921 should operate. Supported values are \fBoff\fR, \fBactive\fR or \fBpassive\fR.
922 .RE

924 .sp
925 .ne 2
926 .na
927 \fB\fB-T\fR \fItime\fR, \fB--lacp-timer\fR=\fItime\fR\fR
928 .ad
929 .br
930 .na
931 \fB\fR
932 .ad
933 .sp .6
934 .RS 4n
935 Specifies the \fBLACP\fR timer value. The supported values are \fBshort\fR or
936 \fBlong\fRj.
937 .RE

939 .sp
940 .ne 2
941 .na
942 \fB\fB-u\fR \fIaddress\fR, \fB--unicast\fR=\fIaddress\fR\fR
943 .ad
944 .sp .6
945 .RS 4n
946 Specifies a fixed unicast hardware address to be used for the aggregation. If
947 this option is not specified, then an address is automatically chosen from the
948 set of addresses of the component devices.
949 .RE

951 .RE

953 .sp
954 .ne 2
955 .na
956 \fB\fBdladm modify-aggr\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] [\fB-P\fR
957 \fIpolicy\fR] [\fB-L\fR \fImode\fR] [\fB-T\fR \fItime\fR] [\fB-u\fR
958 \fIaddress\fR] \fIaggr-link\fR\fR
959 .ad
960 .sp .6
961 .RS 4n
962 Modify the parameters of the specified aggregation.
963 .sp
964 .ne 2
965 .na
966 \fB\fB-t\fR, \fB--temporary\fR\fR
967 .ad
968 .sp .6
969 .RS 4n
970 Specifies that the modification is temporary. Temporary aggregations last until
971 the next reboot.
972 .RE

974 .sp
975 .ne 2
976 .na
977 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
978 .ad
979 .sp .6
980 .RS 4n
981 See "Options," above.
982 .RE

984 .sp
985 .ne 2

```

```

986 .na
987 \fB\fB-P\fR \fIpolicy\fR, \fB--policy\fR=\fIpolicy\fR\fR
988 .ad
989 .sp .6
990 .RS 4n
991 Specifies the port selection policy to use for load spreading of outbound
992 traffic. See \fBdladm create-aggr\fR for a description of valid policy values.
993 .RE

995 .sp
996 .ne 2
997 .na
998 \fB\fB-L\fR \fImode\fR, \fB--lacp-mode\fR=\fImode\fR\fR
999 .ad
1000 .sp .6
1001 .RS 4n
1002 Specifies whether \fBLACP\fR should be used and, if used, the mode in which it
1003 should operate. Supported values are \fBoff\fR, \fBactive\fR, or \fBpassive\fR.
1004 .RE

1006 .sp
1007 .ne 2
1008 .na
1009 \fB\fB-T\fR \fItime\fR, \fB--lacp-timer\fR=\fItime\fR\fR
1010 .ad
1011 .br
1012 .na
1013 \fB\fR
1014 .ad
1015 .sp .6
1016 .RS 4n
1017 Specifies the \fBLACP\fR timer value. The supported values are \fBshort\fR or
1018 \fBlong\fR.
1019 .RE

1021 .sp
1022 .ne 2
1023 .na
1024 \fB\fB-u\fR \fIaddress\fR, \fB--unicast\fR=\fIaddress\fR\fR
1025 .ad
1026 .sp .6
1027 .RS 4n
1028 Specifies a fixed unicast hardware address to be used for the aggregation. If
1029 this option is not specified, then an address is automatically chosen from the
1030 set of addresses of the component devices.
1031 .RE

1033 .RE

1035 .sp
1036 .ne 2
1037 .na
1038 \fB\fBdladm delete-aggr\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR]
1039 \fIaggr-link\fR\fR
1040 .ad
1041 .sp .6
1042 .RS 4n
1043 Deletes the specified aggregation.
1044 .sp
1045 .ne 2
1046 .na
1047 \fB\fB-t\fR, \fB--temporary\fR\fR
1048 .ad
1049 .sp .6
1050 .RS 4n
1051 Specifies that the deletion is temporary. Temporary deletions last until the

```

```

1052 next reboot.
1053 .RE

1055 .sp
1056 .ne 2
1057 .na
1058 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
1059 .ad
1060 .sp .6
1061 .RS 4n
1062 See "Options," above.
1063 .RE

1065 .RE

1067 .sp
1068 .ne 2
1069 .na
1070 \fB\fBdladm add-aggr\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] \fB-l\fR
1071 \fIether-link1\fR [\fB--link\fR=\fIether-link2\fR...] \fIaggr-link\fR\fR
1072 .ad
1073 .sp .6
1074 .RS 4n
1075 Adds links to the specified aggregation.
1076 .sp
1077 .ne 2
1078 .na
1079 \fB\fB-l\fR \fIether-link\fR, \fB--link\fR=\fIether-link\fR\fR
1080 .ad
1081 .sp .6
1082 .RS 4n
1083 Specifies an Ethernet link to add to the aggregation. Multiple links can be
1084 added by supplying multiple \fB-l\fR options.
1085 .RE

1087 .sp
1088 .ne 2
1089 .na
1090 \fB\fB-t\fR, \fB--temporary\fR\fR
1091 .ad
1092 .sp .6
1093 .RS 4n
1094 Specifies that the additions are temporary. Temporary additions last until the
1095 next reboot.
1096 .RE

1098 .sp
1099 .ne 2
1100 .na
1101 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
1102 .ad
1103 .sp .6
1104 .RS 4n
1105 See "Options," above.
1106 .RE

1108 .RE

1110 .sp
1111 .ne 2
1112 .na
1113 \fB\fBdladm remove-aggr\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] \fB-l\fR
1114 \fIether-link1\fR [\fB--l\fR=\fIether-link2\fR...] \fIaggr-link\fR\fR
1115 .ad
1116 .sp .6
1117 .RS 4n

```

```

1118 Removes links from the specified aggregation.
1119 .sp
1120 .ne 2
1121 .na
1122 \fB\fB-l\fR \fIether-link\fR, \fB--link\fR=\fIether-link\fR\fR
1123 .ad
1124 .sp .6
1125 .RS 4n
1126 Specifies an Ethernet link to remove from the aggregation. Multiple links can
1127 be added by supplying multiple \fB-l\fR options.
1128 .RE

1130 .sp
1131 .ne 2
1132 .na
1133 \fB\fB-t\fR, \fB--temporary\fR\fR
1134 .ad
1135 .sp .6
1136 .RS 4n
1137 Specifies that the removals are temporary. Temporary removal last until the
1138 next reboot.
1139 .RE

1141 .sp
1142 .ne 2
1143 .na
1144 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
1145 .ad
1146 .sp .6
1147 .RS 4n
1148 See "Options," above.
1149 .RE

1151 .RE

1153 .sp
1154 .ne 2
1155 .na
1156 \fB\fBdladm show-aggr\fR [\fB-PLx\fR] [\fB-s\fR [\fB-i\fR \fIinterval\fR]]
1157 [\fB-p\fR] \fB-o\fR \fIfield\fR[,...] [\fIaggr-link\fR]\fR
1158 .ad
1159 .sp .6
1160 .RS 4n
1161 Show aggregation configuration (the default), \fBBLACP\fR information, or
1162 statistics, either for all aggregations or for the specified aggregation.
1163 .sp
1164 By default (with no options), the following fields can be displayed:
1165 .sp
1166 .ne 2
1167 .na
1168 \fB\fBBLINK\fR\fR
1169 .ad
1170 .sp .6
1171 .RS 4n
1172 The name of the aggregation link.
1173 .RE

1175 .sp
1176 .ne 2
1177 .na
1178 \fB\fBPOLICY\fR\fR
1179 .ad
1180 .sp .6
1181 .RS 4n
1182 The LACP policy of the aggregation. See the \fBcreate-aggr\fR \fB-P\fR option
1183 for a description of the possible values.

```



```

1184 .RE

1186 .sp
1187 .ne 2
1188 .na
1189 \fB\fBADDRPOLICY\fR\fR
1190 .ad
1191 .sp .6
1192 .RS 4n
1193 Either \fBauto\fR, if the aggregation is configured to automatically configure
1194 its unicast MAC address (the default if the \fB-u\fR option was not used to
1195 create or modify the aggregation), or \fBfixed\fR, if \fB-u\fR was used to set
1196 a fixed MAC address.
1197 .RE

1199 .sp
1200 .ne 2
1201 .na
1202 \fB\fBLACPACTIVITY\fR\fR
1203 .ad
1204 .sp .6
1205 .RS 4n
1206 The LACP mode of the aggregation. Possible values are \fBBoff\fR, \fBActive\fR,
1207 or \fBPassive\fR, as set by the \fB-l\fR option to \fBcreate-aggr\fR or
1208 \fBmodify-aggr\fR.
1209 .RE

1211 .sp
1212 .ne 2
1213 .na
1214 \fB\fBLACPTIMER\fR\fR
1215 .ad
1216 .sp .6
1217 .RS 4n
1218 The LACP timer value of the aggregation as set by the \fB-T\fR option of
1219 \fBcreate-aggr\fR or \fBmodify-aggr\fR.
1220 .RE

1222 .sp
1223 .ne 2
1224 .na
1225 \fB\fBFLAGS\fR\fR
1226 .ad
1227 .sp .6
1228 .RS 4n
1229 A set of state flags associated with the aggregation. The only possible flag is
1230 \fBf\fR, which is displayed if the administrator forced the creation the
1231 aggregation using the \fB-f\fR option to \fBcreate-aggr\fR. Other flags might
1232 be defined in the future.
1233 .RE

1235 The \fBshow-aggr\fR command accepts the following options:
1236 .sp
1237 .ne 2
1238 .na
1239 \fB\fB-L\fR, \fB--lacp\fR\fR
1240 .ad
1241 .sp .6
1242 .RS 4n
1243 Displays detailed \fBLACP\fR information for the aggregation link and each
1244 underlying port. Most of the state information displayed by this option is
1245 defined by IEEE 802.3. With this option, the following fields can be displayed:
1246 .sp
1247 .ne 2
1248 .na
1249 \fB\fBLINK\fR\fR

```

```

1250 .ad
1251 .sp .6
1252 .RS 4n
1253 The name of the aggregation link.
1254 .RE

1256 .sp
1257 .ne 2
1258 .na
1259 \fB\fBPORT\fR\fR
1260 .ad
1261 .sp .6
1262 .RS 4n
1263 The name of one of the underlying aggregation ports.
1264 .RE

1266 .sp
1267 .ne 2
1268 .na
1269 \fB\fBAGGREGATABLE\fR\fR
1270 .ad
1271 .sp .6
1272 .RS 4n
1273 Whether the port can be added to the aggregation.
1274 .RE

1276 .sp
1277 .ne 2
1278 .na
1279 \fB\fBSYNC\fR\fR
1280 .ad
1281 .sp .6
1282 .RS 4n
1283 If \fBByes\fR, the system considers the port to be synchronized and part of the
1284 aggregation.
1285 .RE

1287 .sp
1288 .ne 2
1289 .na
1290 \fB\fBCOLL\fR\fR
1291 .ad
1292 .sp .6
1293 .RS 4n
1294 If \fBByes\fR, collection of incoming frames is enabled on the associated port.
1295 .RE

1297 .sp
1298 .ne 2
1299 .na
1300 \fB\fBDIST\fR\fR
1301 .ad
1302 .sp .6
1303 .RS 4n
1304 If \fBByes\fR, distribution of outgoing frames is enabled on the associated
1305 port.
1306 .RE

1308 .sp
1309 .ne 2
1310 .na
1311 \fB\fBDEFAULTED\fR\fR
1312 .ad
1313 .sp .6
1314 .RS 4n
1315 If \fBByes\fR, the port is using defaulted partner information (that is, has not

```

```

1316 received LACP data from the LACP partner).
1317 .RE

1319 .sp
1320 .ne 2
1321 .na
1322 \fB\FBEXPIRED\fR\fR
1323 .ad
1324 .sp .6
1325 .RS 4n
1326 If \fBByes\fR, the receive state of the port is in the \fBEXPIRED\fR state.
1327 .RE

1329 .RE

1331 .sp
1332 .ne 2
1333 .na
1334 \fB\FB-x\fR, \fB--extended\fR\fR
1335 .ad
1336 .sp .6
1337 .RS 4n
1338 Display additional aggregation information including detailed information on
1339 each underlying port. With \fB-x\fR, the following fields can be displayed:
1340 .sp
1341 .ne 2
1342 .na
1343 \fB\FBLINK\fR\fR
1344 .ad
1345 .sp .6
1346 .RS 4n
1347 The name of the aggregation link.
1348 .RE

1350 .sp
1351 .ne 2
1352 .na
1353 \fB\FBPORT\fR\fR
1354 .ad
1355 .sp .6
1356 .RS 4n
1357 The name of one of the underlying aggregation ports.
1358 .RE

1360 .sp
1361 .ne 2
1362 .na
1363 \fB\FBSPEED\fR\fR
1364 .ad
1365 .sp .6
1366 .RS 4n
1367 The speed of the link or port in megabits per second.
1368 .RE

1370 .sp
1371 .ne 2
1372 .na
1373 \fB\FBDUPLEX\fR\fR
1374 .ad
1375 .sp .6
1376 .RS 4n
1377 The full/half duplex status of the link or port is displayed if the link state
1378 is \fBUp\fR. The duplex status is displayed as \fBUnknown\fR in all other
1379 cases.
1380 .RE

```

```

1382 .sp
1383 .ne 2
1384 .na
1385 \fB\FBSTATE\fR\fR
1386 .ad
1387 .sp .6
1388 .RS 4n
1389 The link state. This can be \fBUp\fR, \fBDown\fR, or \fBUnknown\fR.
1390 .RE

1392 .sp
1393 .ne 2
1394 .na
1395 \fB\FBADDRESS\fR\fR
1396 .ad
1397 .sp .6
1398 .RS 4n
1399 The MAC address of the link or port.
1400 .RE

1402 .sp
1403 .ne 2
1404 .na
1405 \fB\FBPORTSTATE\fR\fR
1406 .ad
1407 .sp .6
1408 .RS 4n
1409 This indicates whether the individual aggregation port is in the \fBStandby\fR
1410 or \fBAttached\fR state.
1411 .RE

1413 .RE

1415 .sp
1416 .ne 2
1417 .na
1418 \fB\FB-o\fR \fIfield\fR[,...], \fB--output\fR=\fIfield\fR[,...]\fR
1419 .ad
1420 .sp .6
1421 .RS 4n
1422 A case-insensitive, comma-separated list of output fields to display. The field
1423 name must be one of the fields listed above, or the special value \fBall\fR, to
1424 display all fields. The fields applicable to the \fB-o\fR option are limited to
1425 those listed under each output mode. For example, if using \fB-L\fR, only the
1426 fields listed under \fB-L\fR, above, can be used with \fB-o\fR.
1427 .RE

1429 .sp
1430 .ne 2
1431 .na
1432 \fB\FB-p\fR, \fB--parseable\fR\fR
1433 .ad
1434 .sp .6
1435 .RS 4n
1436 Display using a stable machine-parseable format. The \fB-o\fR option is
1437 required with \fB-p\fR. See "Parseable Output Format", below.
1438 .RE

1440 .sp
1441 .ne 2
1442 .na
1443 \fB\FB-P\fR, \fB--persistent\fR\fR
1444 .ad
1445 .sp .6
1446 .RS 4n
1447 Display the persistent aggregation configuration rather than the state of the

```

```

1448 running system.
1449 .RE

1451 .sp
1452 .ne 2
1453 .na
1454 \fB\fB-s\fR, \fB--statistics\fR\fR
1455 .ad
1456 .sp .6
1457 .RS 4n
1458 Displays aggregation statistics.
1459 .RE

1461 .sp
1462 .ne 2
1463 .na
1464 \fB\fB-i\fR \fIinterval\fR, \fB--interval\fR=\fIinterval\fR\fR
1465 .ad
1466 .sp .6
1467 .RS 4n
1468 Used with the \fB-s\fR option to specify an interval, in seconds, at which
1469 statistics should be displayed. If this option is not specified, statistics
1470 will be displayed only once.
1471 .RE

1473 .RE

1475 .sp
1476 .ne 2
1477 .na
1478 \fB\fBdladm create-bridge\fR [ \fB-P\fR \fIprotect\fR] [\fB-R\fR
1479 \fIroot-dir\fR] [ \fB-p\fR \fIpriority\fR] [ \fB-m\fR \fIimax-age\fR] [ \fB-h\fR
1480 \fIhello-time\fR] [ \fB-d\fR \fIforward-delay\fR] [ \fB-f\fR
1481 \fIforce-protocol\fR] [\fB-l\fR \fIlink\fR...] \fIbridge-name\fR\fR
1482 .ad
1483 .sp .6
1484 .RS 4n
1485 Create an 802.1D bridge instance and optionally assign one or more network
1486 links to the new bridge. By default, no bridge instances are present on the
1487 system.
1488 .sp
1489 In order to bridge between links, you must create at least one bridge instance.
1490 Each bridge instance is separate, and there is no forwarding connection between
1491 bridges.
1492 .sp
1493 .ne 2
1494 .na
1495 \fB\fB-P\fR \fIprotect\fR, \fB--protect\fR=\fIprotect\fR\fR
1496 .ad
1497 .sp .6
1498 .RS 4n
1499 Specifies a protection method. The defined protection methods are \fBstp\fR for
1500 the Spanning Tree Protocol and trill for \fBTRILL\fR, which is used on
1501 Rbridges. The default value is \fBstp\fR.
1502 .RE

1504 .sp
1505 .ne 2
1506 .na
1507 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
1508 .ad
1509 .sp .6
1510 .RS 4n
1511 See "Options," above.
1512 .RE

```

```

1514 .sp
1515 .ne 2
1516 .na
1517 \fB\fB-p\fR \fIpriority\fR, \fB--priority\fR=\fIpriority\fR\fR
1518 .ad
1519 .sp .6
1520 .RS 4n
1521 Specifies the Bridge Priority. This sets the IEEE STP priority value for
1522 determining the root bridge node in the network. The default value is
1523 \fB32768\fR. Valid values are \fB0\fR (highest priority) to \fB61440\fR (lowest
1524 priority), in increments of 4096.
1525 .sp
1526 If a value not evenly divisible by 4096 is used, the system silently rounds
1527 downward to the next lower value that is divisible by 4096.
1528 .RE

1530 .sp
1531 .ne 2
1532 .na
1533 \fB\fB-m\fR \fIimax-age\fR, \fB--max-age\fR=\fIimax-age\fR\fR
1534 .ad
1535 .sp .6
1536 .RS 4n
1537 Specifies the maximum age for configuration information in seconds. This sets
1538 the STP Bridge Max Age parameter. This value is used for all nodes in the
1539 network if this node is the root bridge. Bridge link information older than
1540 this time is discarded. It defaults to 20 seconds. Valid values are from 6 to
1541 40 seconds. See the \fB-d\fR \fIforward-delay\fR parameter for additional
1542 constraints.
1543 .RE

1545 .sp
1546 .ne 2
1547 .na
1548 \fB\fB-h\fR \fIhello-time\fR, \fB--hello-time\fR=\fIhello-time\fR\fR
1549 .ad
1550 .sp .6
1551 .RS 4n
1552 Specifies the STP Bridge Hello Time parameter. When this node is the root node,
1553 it sends Configuration BPDUs at this interval throughout the network. The
1554 default value is 2 seconds. Valid values are from 1 to 10 seconds. See the
1555 \fB-d\fR \fIforward-delay\fR parameter for additional constraints.
1556 .RE

1558 .sp
1559 .ne 2
1560 .na
1561 \fB\fB-d\fR \fIforward-delay\fR, \fB--forward-delay\fR=\fIforward-delay\fR\fR
1562 .ad
1563 .sp .6
1564 .RS 4n
1565 Specifies the STP Bridge Forward Delay parameter. When this node is the root
1566 node, then all bridges in the network use this timer to sequence the link
1567 states when a port is enabled. The default value is 15 seconds. Valid values
1568 are from 4 to 30 seconds.
1569 .sp
1570 Bridges must obey the following two constraints:
1571 .sp
1572 .in +2
1573 .nf
1574 2 * (\fIforward-delay\fR - 1.0) >= \fIimax-age\fR

1576 \fIimax-age\fR >= 2 * (\fIhello-time\fR + 1.0)
1577 .fi
1578 .in -2
1579 .sp

```

1581 Any parameter setting that would violate those constraints is treated as an
 1582 error and causes the command to fail with a diagnostic message. The message
 1583 provides valid alternatives to the supplied values.
 1584 .RE

1586 .sp
 1587 .ne 2
 1588 .na
 1589 \fB\fB-f\fR \fIforce-protocol\fR,
 1590 \fB--force-protocol\fR=\fIforce-protocol\fR\fR
 1591 .ad
 1592 .sp .6
 1593 .RS 4n
 1594 Specifies the MSTP forced maximum supported protocol. The default value is 3.
 1595 Valid values are non-negative integers. The current implementation does not
 1596 support RSTP or MSTP, so this currently has no effect. However, to prevent MSTP
 1597 from being used in the future, the parameter may be set to \fB0\fR for STP only
 1598 or \fB2\fR for STP and RSTP.
 1599 .RE

1601 .sp
 1602 .ne 2
 1603 .na
 1604 \fB\fB-l\fR \fIlink\fR, \fB--link\fR=\fIlink\fR\fR
 1605 .ad
 1606 .sp .6
 1607 .RS 4n
 1608 Specifies one or more links to add to the newly-created bridge. This is similar
 1609 to creating the bridge and then adding one or more links, as with the
 1610 \fBadd-bridge\fR subcommand. However, if any of the links cannot be added, the
 1611 entire command fails, and the new bridge itself is not created. To add multiple
 1612 links on the same command line, repeat this option for each link. You are
 1613 permitted to create bridges without links. For more information about link
 1614 assignments, see the \fBadd-bridge\fR subcommand.
 1615 .RE

1617 Bridge creation and link assignment require the \fBPRIV_SYS_DL_CONFIG\fR
 1618 privilege. Bridge creation might fail if the optional bridging feature is not
 1619 installed on the system.
 1620 .RE

1622 .sp
 1623 .ne 2
 1624 .na
 1625 \fB\fBdladm modify-bridge\fR [\fB-P\fR \fIprotect\fR] [\fB-R\fR
 1626 \fIroot-dir\fR] [\fB-p\fR \fIpriority\fR] [\fB-m\fR \fImax-age\fR] [\fB-h\fR
 1627 \fIhello-time\fR] [\fB-d\fR \fIforward-delay\fR] [\fB-f\fR
 1628 \fIforce-protocol\fR] [\fB-l\fR \fIlink\fR...] \fIbridge-name\fR\fR
 1629 .ad
 1630 .sp .6
 1631 .RS 4n
 1632 Modify the operational parameters of an existing bridge. The options are the
 1633 same as for the \fBcreate-bridge\fR subcommand, except that the \fB-l\fR option
 1634 is not permitted. To add links to an existing bridge, use the \fBadd-bridge\fR
 1635 subcommand.
 1636 .sp
 1637 Bridge parameter modification requires the \fBPRIV_SYS_DL_CONFIG\fR privilege.
 1638 .RE

1640 .sp
 1641 .ne 2
 1642 .na
 1643 \fB\fBdladm delete-bridge\fR [\fB-R\fR \fIroot-dir\fR] \fIbridge-name\fR\fR
 1644 .ad
 1645 .sp .6

1646 .RS 4n
 1647 Delete a bridge instance. The bridge being deleted must not have any attached
 1648 links. Use the \fBremove-bridge\fR subcommand to deactivate links before
 1649 deleting a bridge.
 1650 .sp
 1651 Bridge deletion requires the \fBPRIV_SYS_DL_CONFIG\fR privilege.
 1652 .sp
 1653 The \fB-R\fR (\fB--root-dir\fR) option is the same as for the
 1654 \fBcreate-bridge\fR subcommand.
 1655 .RE

1657 .sp
 1658 .ne 2
 1659 .na
 1660 \fB\fBdladm add-bridge\fR [\fB-R\fR \fIroot-dir\fR] \fB-l\fR \fIlink\fR
 1661 [\fB-l\fR \fIlink\fR...] \fIbridge-name\fR\fR
 1662 .ad
 1663 .sp .6
 1664 .RS 4n
 1665 Add one or more links to an existing bridge. If multiple links are specified,
 1666 and adding any one of them results in an error, the command fails and no
 1667 changes are made to the system.
 1668 .sp
 1669 Link addition to a bridge requires the \fBPRIV_SYS_DL_CONFIG\fR privilege.
 1670 .sp
 1671 A link may be a member of at most one bridge. An error occurs when you attempt
 1672 to add a link that already belongs to another bridge. To move a link from one
 1673 bridge instance to another, remove it from the current bridge before adding it
 1674 to a new one.
 1675 .sp
 1676 The links assigned to a bridge must not also be VLANs, VNICs, or tunnels. Only
 1677 physical Ethernet datalinks, aggregation datalinks, wireless links, and
 1678 Ethernet stubs are permitted to be assigned to a bridge.
 1679 .sp
 1680 Links assigned to a bridge must all have the same MTU. This is checked when the
 1681 link is assigned. The link is added to the bridge in a deactivated form if it
 1682 is not the first link on the bridge and it has a differing MTU.
 1683 .sp
 1684 Note that systems using bridging should not set the \fBBeeprom\fR(LM)
 1685 \fBlocal-mac-address\fR variable to false.
 1686 .sp
 1687 The options are the same as for the \fBcreate-bridge\fR subcommand.
 1688 .RE

1690 .sp
 1691 .ne 2
 1692 .na
 1693 \fB\fBdladm remove-bridge\fR [\fB-R\fR \fIroot-dir\fR] \fB-l\fR \fIlink\fR
 1694 [\fB-l\fR \fIlink\fR...] \fIbridge-name\fR\fR
 1695 .ad
 1696 .sp .6
 1697 .RS 4n
 1698 Remove one or more links from a bridge instance. If multiple links are
 1699 specified, and removing any one of them would result in an error, the command
 1700 fails and none are removed.
 1701 .sp
 1702 Link removal from a bridge requires the \fBPRIV_SYS_DL_CONFIG\fR privilege.
 1703 .sp
 1704 The options are the same as for the \fBcreate-bridge\fR subcommand.
 1705 .RE

1707 .sp
 1708 .ne 2
 1709 .na
 1710 \fB\fBdladm show-bridge\fR [\fB-flt\fR] [\fB-s\fR [\fB-i\fR \fIinterval\fR]]
 1711 [\fB-p\fR] \fB-o\fR \fIifield\fR, ...] [\fIbridge-name\fR]\fR

```

1712 .ad
1713 .sp .6
1714 .RS 4n
1715 Show the running status and configuration of bridges, their attached links,
1716 learned forwarding entries, and \fBTRILL\fR nickname databases. When showing
1717 overall bridge status and configuration, the bridge name can be omitted to show
1718 all bridges. The other forms require a specified bridge.
1719 .sp
1720 The show-bridge subcommand accepts the following options:
1721 .sp
1722 .ne 2
1723 .na
1724 \fB\fB-i\fR \fIinterval\fR, \fB--interval\fR=\fIinterval\fR\fR
1725 .ad
1726 .sp .6
1727 .RS 4n
1728 Used with the \fB-s\fR option to specify an interval, in seconds, at which
1729 statistics should be displayed. If this option is not specified, statistics
1730 will be displayed only once.
1731 .RE

1733 .sp
1734 .ne 2
1735 .na
1736 \fB\fB-s\fR, \fB--statistics\fR\fR
1737 .ad
1738 .sp .6
1739 .RS 4n
1740 Display statistics for the specified bridges or for a given bridge's attached
1741 links. This option cannot be used with the \fB-f\fR and \fB-t\fR options.
1742 .RE

1744 .sp
1745 .ne 2
1746 .na
1747 \fB\fB-p\fR, \fB--parseable\fR\fR
1748 .ad
1749 .sp .6
1750 .RS 4n
1751 Display using a stable machine-parsable format. See "Parsable Output Format,"
1752 below.
1753 .RE

1755 .sp
1756 .ne 2
1757 .na
1758 \fB\fB-o\fR \fIifield\fR[...], \fB--output\fR=\fIifield\fR[...]\fR
1759 .ad
1760 .sp .6
1761 .RS 4n
1762 A case-insensitive, comma-separated list of output fields to display. The field
1763 names are described below. The special value all displays all fields. Each set
1764 of fields has its own default set to display when \fB-o\fR is not specified.
1765 .RE

1767 By default, the \fBshow-bridge\fR subcommand shows bridge configuration. The
1768 following fields can be shown:
1769 .sp
1770 .ne 2
1771 .na
1772 \fB\fBBRIDGE\fR\fR
1773 .ad
1774 .sp .6
1775 .RS 4n
1776 The name of the bridge.
1777 .RE

```

```

1779 .sp
1780 .ne 2
1781 .na
1782 \fB\fBADDRESS\fR\fR
1783 .ad
1784 .sp .6
1785 .RS 4n
1786 The Bridge Unique Identifier value (MAC address).
1787 .RE

1789 .sp
1790 .ne 2
1791 .na
1792 \fB\fBPRIORITY\fR\fR
1793 .ad
1794 .sp .6
1795 .RS 4n
1796 Configured priority value; set by \fB-p\fR with \fBcreate-bridge\fR and
1797 \fBmodify-bridge\fR.
1798 .RE

1800 .sp
1801 .ne 2
1802 .na
1803 \fB\fBBMAXAGE\fR\fR
1804 .ad
1805 .sp .6
1806 .RS 4n
1807 Configured bridge maximum age; set by \fB-m\fR with \fBcreate-bridge\fR and
1808 \fBmodify-bridge\fR.
1809 .RE

1811 .sp
1812 .ne 2
1813 .na
1814 \fB\fBBHELLOTIME\fR\fR
1815 .ad
1816 .sp .6
1817 .RS 4n
1818 Configured bridge hello time; set by \fB-h\fR with \fBcreate-bridge\fR and
1819 \fBmodify-bridge\fR.
1820 .RE

1822 .sp
1823 .ne 2
1824 .na
1825 \fB\fBBFWDDelay\fR\fR
1826 .ad
1827 .sp .6
1828 .RS 4n
1829 Configured forwarding delay; set by \fB-d\fR with \fBcreate-bridge\fR and
1830 \fBmodify-bridge\fR.
1831 .RE

1833 .sp
1834 .ne 2
1835 .na
1836 \fB\fBFORCEPROTO\fR\fR
1837 .ad
1838 .sp .6
1839 .RS 4n
1840 Configured forced maximum protocol; set by \fB-f\fR with \fBcreate-bridge\fR
1841 and \fBmodify-bridge\fR.
1842 .RE

```

```

1844 .sp
1845 .ne 2
1846 .na
1847 \fb\fbTCTIME\fr\fr
1848 .ad
1849 .sp .6
1850 .RS 4n
1851 Time, in seconds, since last topology change.
1852 .RE

1854 .sp
1855 .ne 2
1856 .na
1857 \fb\fbTCCOUNT\fr\fr
1858 .ad
1859 .sp .6
1860 .RS 4n
1861 Count of the number of topology changes.
1862 .RE

1864 .sp
1865 .ne 2
1866 .na
1867 \fb\fbTCHANGE\fr\fr
1868 .ad
1869 .sp .6
1870 .RS 4n
1871 This indicates that a topology change was detected.
1872 .RE

1874 .sp
1875 .ne 2
1876 .na
1877 \fb\fbDESROOT\fr\fr
1878 .ad
1879 .sp .6
1880 .RS 4n
1881 Bridge Identifier of the root node.
1882 .RE

1884 .sp
1885 .ne 2
1886 .na
1887 \fb\fbROOTCOST\fr\fr
1888 .ad
1889 .sp .6
1890 .RS 4n
1891 Cost of the path to the root node.
1892 .RE

1894 .sp
1895 .ne 2
1896 .na
1897 \fb\fbROOTPORT\fr\fr
1898 .ad
1899 .sp .6
1900 .RS 4n
1901 Port number used to reach the root node.
1902 .RE

1904 .sp
1905 .ne 2
1906 .na
1907 \fb\fbMAXAGE\fr\fr
1908 .ad
1909 .sp .6

```

```

1910 .RS 4n
1911 Maximum age value from the root node.
1912 .RE

1914 .sp
1915 .ne 2
1916 .na
1917 \fb\fbHELLOTIME\fr\fr
1918 .ad
1919 .sp .6
1920 .RS 4n
1921 Hello time value from the root node.
1922 .RE

1924 .sp
1925 .ne 2
1926 .na
1927 \fb\fbFWDELAY\fr\fr
1928 .ad
1929 .sp .6
1930 .RS 4n
1931 Forward delay value from the root node.
1932 .RE

1934 .sp
1935 .ne 2
1936 .na
1937 \fb\fbHOLDTIME\fr\fr
1938 .ad
1939 .sp .6
1940 .RS 4n
1941 Minimum BPDU interval.
1942 .RE

1944 By default, when the \fb-o\fr option is not specified, only the \fbBRIDGE\fr,
1945 \fbADDRESS\fr, \fbPRIORITY\fr, and \fbDESROOT\fr fields are shown.
1946 .sp
1947 When the \fb-s\fr option is specified, the \fbshow-bridge\fr subcommand shows
1948 bridge statistics. The following fields can be shown:
1949 .sp
1950 .ne 2
1951 .na
1952 \fb\fbBRIDGE\fr\fr
1953 .ad
1954 .sp .6
1955 .RS 4n
1956 Bridge name.
1957 .RE

1959 .sp
1960 .ne 2
1961 .na
1962 \fb\fbDROPS\fr\fr
1963 .ad
1964 .sp .6
1965 .RS 4n
1966 Number of packets dropped due to resource problems.
1967 .RE

1969 .sp
1970 .ne 2
1971 .na
1972 \fb\fbFORWARDS\fr\fr
1973 .ad
1974 .sp .6
1975 .RS 4n

```

1976 Number of packets forwarded from one link to another.
1977 .RE

1979 .sp
1980 .ne 2
1981 .na
1982 \fB\FBMBCAST\fR\fR
1983 .ad
1984 .sp .6
1985 .RS 4n
1986 Number of multicast and broadcast packets handled by the bridge.
1987 .RE

1989 .sp
1990 .ne 2
1991 .na
1992 \fB\FBRECV\fR\fR
1993 .ad
1994 .sp .6
1995 .RS 4n
1996 Number of packets received on all attached links.
1997 .RE

1999 .sp
2000 .ne 2
2001 .na
2002 \fB\FBSENT\fR\fR
2003 .ad
2004 .sp .6
2005 .RS 4n
2006 Number of packets sent on all attached links.
2007 .RE

2009 .sp
2010 .ne 2
2011 .na
2012 \fB\FBUNKNOWN\fR\fR
2013 .ad
2014 .sp .6
2015 .RS 4n
2016 Number of packets handled that have an unknown destination. Such packets are
2017 sent to all links.
2018 .RE

2020 By default, when the \fB-o\fR option is not specified, only the \fBBRIDGE\fR,
2021 \fBDROPS\fR, and \fBFORWARDS\fR fields are shown.
2022 .sp
2023 The \fBshow-bridge\fR subcommand also accepts the following options:
2024 .sp
2025 .ne 2
2026 .na
2027 \fB-l\fR, \fB--link\fR\fR
2028 .ad
2029 .sp .6
2030 .RS 4n
2031 Displays link-related status and statistics information for all links attached
2032 to a single bridge instance. By using this option and without the \fB-s\fR
2033 option, the following fields can be displayed for each link:
2034 .sp
2035 .ne 2
2036 .na
2037 \fB\FBLINK\fR\fR
2038 .ad
2039 .sp .6
2040 .RS 4n
2041 The link name.

2042 .RE

2044 .sp
2045 .ne 2
2046 .na
2047 \fB\FBINDEX\fR\fR
2048 .ad
2049 .sp .6
2050 .RS 4n
2051 Port (link) index number on the bridge.
2052 .RE

2054 .sp
2055 .ne 2
2056 .na
2057 \fB\FBSTATE\fR\fR
2058 .ad
2059 .sp .6
2060 .RS 4n
2061 State of the link. The state can be \fBdisabled\fR, \fBdiscarding\fR,
2062 \fBlearning\fR, \fBforwarding\fR, \fBnon-stp\fR, or \fBbad-mtu\fR.
2063 .RE

2065 .sp
2066 .ne 2
2067 .na
2068 \fB\FBUPTIME\fR\fR
2069 .ad
2070 .sp .6
2071 .RS 4n
2072 Number of seconds since the last reset or initialization.
2073 .RE

2075 .sp
2076 .ne 2
2077 .na
2078 \fB\FBOPERCOST\fR\fR
2079 .ad
2080 .sp .6
2081 .RS 4n
2082 Actual cost in use (1-65535).
2083 .RE

2085 .sp
2086 .ne 2
2087 .na
2088 \fB\FBOPERP2P\fR\fR
2089 .ad
2090 .sp .6
2091 .RS 4n
2092 This indicates whether point-to-point (\fBFP2P\fR) mode been detected.
2093 .RE

2095 .sp
2096 .ne 2
2097 .na
2098 \fB\FBOPEREDGE\fR\fR
2099 .ad
2100 .sp .6
2101 .RS 4n
2102 This indicates whether edge mode has been detected.
2103 .RE

2105 .sp
2106 .ne 2
2107 .na

```

2108 \fb\fbDESROOT\fr\fr
2109 .ad
2110 .sp .6
2111 .RS 4n
2112 The Root Bridge Identifier that has been seen on this port.
2113 .RE

2115 .sp
2116 .ne 2
2117 .na
2118 \fb\fbDESCOST\fr\fr
2119 .ad
2120 .sp .6
2121 .RS 4n
2122 Path cost to the network root node through the designated port.
2123 .RE

2125 .sp
2126 .ne 2
2127 .na
2128 \fb\fbDESRIDGE\fr\fr
2129 .ad
2130 .sp .6
2131 .RS 4n
2132 Bridge Identifier for this port.
2133 .RE

2135 .sp
2136 .ne 2
2137 .na
2138 \fb\fbDESSPORT\fr\fr
2139 .ad
2140 .sp .6
2141 .RS 4n
2142 The ID and priority of the port used to transmit configuration messages for
2143 this port.
2144 .RE

2146 .sp
2147 .ne 2
2148 .na
2149 \fb\fbTACK\fr\fr
2150 .ad
2151 .sp .6
2152 .RS 4n
2153 This indicates whether Topology Change Acknowledge has been seen.
2154 .RE

2156 When the \fb-l\fr option is specified without the \fb-o\fr option, only the
2157 \fbLINK\fr, \fbSTATE\fr, \fbUPTIME\fr, and \fbDESROOT\fr fields are shown.
2158 .sp
2159 When the \fb-l\fr option is specified, the \fb-s\fr option can be used to
2160 display the following fields for each link:
2161 .sp
2162 .ne 2
2163 .na
2164 \fb\fbLINK\fr\fr
2165 .ad
2166 .sp .6
2167 .RS 4n
2168 Link name.
2169 .RE

2171 .sp
2172 .ne 2
2173 .na

```

```

2174 \fb\fbCFGBPDU\fr\fr
2175 .ad
2176 .sp .6
2177 .RS 4n
2178 Number of configuration BPDUs received.
2179 .RE

2181 .sp
2182 .ne 2
2183 .na
2184 \fb\fbTCNPDU\fr\fr
2185 .ad
2186 .sp .6
2187 .RS 4n
2188 Number of topology change BPDUs received.
2189 .RE

2191 .sp
2192 .ne 2
2193 .na
2194 \fb\fbRSTPBPDU\fr\fr
2195 .ad
2196 .sp .6
2197 .RS 4n
2198 Number of Rapid Spanning Tree BPDUs received.
2199 .RE

2201 .sp
2202 .ne 2
2203 .na
2204 \fb\fbTXBPDU\fr\fr
2205 .ad
2206 .sp .6
2207 .RS 4n
2208 Number of BPDUs transmitted.
2209 .RE

2211 .sp
2212 .ne 2
2213 .na
2214 \fb\fbDROPS\fr\fr
2215 .ad
2216 .sp .6
2217 .RS 4n
2218 Number of packets dropped due to resource problems.
2219 .RE

2221 .sp
2222 .ne 2
2223 .na
2224 \fb\fbRECV\fr\fr
2225 .ad
2226 .sp .6
2227 .RS 4n
2228 Number of packets received by the bridge.
2229 .RE

2231 .sp
2232 .ne 2
2233 .na
2234 \fb\fbXMIT\fr\fr
2235 .ad
2236 .sp .6
2237 .RS 4n
2238 Number of packets sent by the bridge.
2239 .RE

```



```

2241 When the \fB-o\fR option is not specified, only the \fBBLINK\fR, \fBBDROPS\fR,
2242 \fBRECV\fR, and \fBXMIT\fR fields are shown.
2243 .RE

2245 .sp
2246 .ne 2
2247 .na
2248 \fB\fB-f\fR, \fB--forwarding\fR\fR
2249 .ad
2250 .sp .6
2251 .RS 4n
2252 Displays forwarding entries for a single bridge instance. With this option, the
2253 following fields can be shown for each forwarding entry:
2254 .sp
2255 .ne 2
2256 .na
2257 \fB\fBDEST\fR\fR
2258 .ad
2259 .sp .6
2260 .RS 4n
2261 Destination MAC address.
2262 .RE

2264 .sp
2265 .ne 2
2266 .na
2267 \fB\fBAGE\fR\fR
2268 .ad
2269 .sp .6
2270 .RS 4n
2271 Age of entry in seconds and milliseconds. Omitted for local entries.
2272 .RE

2274 .sp
2275 .ne 2
2276 .na
2277 \fB\fBFLAGS\fR\fR
2278 .ad
2279 .sp .6
2280 .RS 4n
2281 The \fBBL\fR (local) flag is shown if the MAC address belongs to an attached
2282 link or to a VNIC on one of the attached links.
2283 .RE

2285 .sp
2286 .ne 2
2287 .na
2288 \fB\fBOUTPUT\fR\fR
2289 .ad
2290 .sp .6
2291 .RS 4n
2292 For local entries, this is the name of the attached link that has the MAC
2293 address. Otherwise, for bridges that use Spanning Tree Protocol, this is the
2294 output interface name. For Rbridges, this is the output \fBTRILL\fR nickname.
2295 .RE

2297 When the \fB-o\fR option is not specified, the \fBDEST\fR, \fBAGE\fR,
2298 \fBFLAGS\fR, and \fBOUTPUT\fR fields are shown.
2299 .RE

2301 .sp
2302 .ne 2
2303 .na
2304 \fB\fB-t\fR, \fB--trill\fR\fR
2305 .ad

```

```

2306 .sp .6
2307 .RS 4n
2308 Displays \fBTRILL\fR nickname entries for a single bridge instance. With this
2309 option, the following fields can be shown for each \fBTRILL\fR nickname entry:
2310 .sp
2311 .ne 2
2312 .na
2313 \fB\fBNICK\fR\fR
2314 .ad
2315 .sp .6
2316 .RS 4n
2317 \fBTRILL\fR nickname for this RBridge, which is a number from 1 to 65535.
2318 .RE

2320 .sp
2321 .ne 2
2322 .na
2323 \fB\fBFLAGS\fR\fR
2324 .ad
2325 .sp .6
2326 .RS 4n
2327 The \fBBL\fR flag is shown if the nickname identifies the local system.
2328 .RE

2330 .sp
2331 .ne 2
2332 .na
2333 \fB\fBBLINK\fR\fR
2334 .ad
2335 .sp .6
2336 .RS 4n
2337 Link name for output when sending messages to this RBridge.
2338 .RE

2340 .sp
2341 .ne 2
2342 .na
2343 \fB\fBNEXTHOP\fR\fR
2344 .ad
2345 .sp .6
2346 .RS 4n
2347 MAC address of the next hop RBridge that is used to reach the RBridge with this
2348 nickname.
2349 .RE

2351 When the \fB-o\fR option is not specified, the \fBBLINK\fR, \fBFLAGS\fR,
2352 \fBBLINK\fR, and \fBEXTHOP\fR fields are shown.
2353 .RE

2355 .RE

2357 .sp
2358 .ne 2
2359 .na
2360 \fB\fBdladm create-vlan\fR [\fB-ft\fR] [\fB-R\fR \fIroot-dir\fR] \fB-l\fR
2361 \fIether-link\fR \fB-v\fR \fIvid\fR [\fIvlan-link\fR]\fR
2362 .ad
2363 .sp .6
2364 .RS 4n
2365 Create a tagged VLAN link with an ID of \fIvid\fR over Ethernet link
2366 \fIether-link\fR. The name of the VLAN link can be specified as
2367 \fIvlan\fR-\fIlink\fR. If the name is not specified, a name will be
2368 automatically generated (assuming that \fIether-link\fR is \fIname\fR\fIppa\fR)
2369 as:
2370 .sp
2371 .in +2

```

```

2372 .nf
2373 <\fIname\fR><1000 * \fIvlan-tag\fR + \fIPPA\fR>
2374 .fi
2375 .in -2
2376 .sp

2378 For example, if \fIether-link\fR is \fBbge\fR and \fIvid\fR is 2, the name
2379 generated is \fBbge2001\fR.
2380 .sp
2381 .ne 2
2382 .na
2383 \fB\fB-f\fR, \fB--force\fR\fR
2384 .ad
2385 .sp .6
2386 .RS 4n
2387 Force the creation of the VLAN link. Some devices do not allow frame sizes
2388 large enough to include a VLAN header. When creating a VLAN link over such a
2389 device, the \fB-f\fR option is needed, and the MTU of the IP interfaces on the
2390 resulting VLAN must be set to 1496 instead of 1500.
2391 .RE

2393 .sp
2394 .ne 2
2395 .na
2396 \fB\fB-l\fR \fIether-link\fR\fR
2397 .ad
2398 .sp .6
2399 .RS 4n
2400 Specifies Ethernet link over which VLAN is created.
2401 .RE

2403 .sp
2404 .ne 2
2405 .na
2406 \fB\fB-t\fR, \fB--temporary\fR\fR
2407 .ad
2408 .sp .6
2409 .RS 4n
2410 Specifies that the VLAN link is temporary. Temporary VLAN links last until the
2411 next reboot.
2412 .RE

2414 .sp
2415 .ne 2
2416 .na
2417 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
2418 .ad
2419 .sp .6
2420 .RS 4n
2421 See "Options," above.
2422 .RE

2424 .RE

2426 .sp
2427 .ne 2
2428 .na
2429 \fB\fBdladm delete-vlan\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR]
2430 \fIvlan-link\fR\fR
2431 .ad
2432 .sp .6
2433 .RS 4n
2434 Delete the VLAN link specified.
2435 .sp
2436 The \fBdelete-vlan\fRsubcommand accepts the following options:
2437 .sp

```

```

2438 .ne 2
2439 .na
2440 \fB\fB-t\fR, \fB--temporary\fR\fR
2441 .ad
2442 .sp .6
2443 .RS 4n
2444 Specifies that the deletion is temporary. Temporary deletions last until the
2445 next reboot.
2446 .RE

2448 .sp
2449 .ne 2
2450 .na
2451 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
2452 .ad
2453 .sp .6
2454 .RS 4n
2455 See "Options," above.
2456 .RE

2458 .RE

2460 .sp
2461 .ne 2
2462 .na
2463 \fB\fBdladm show-vlan\fR [\fB-P\fR] [[\fB-p\fR] \fB-o\fR \fIfield\fR[,...]]
2464 [\fIvlan-link\fR]\fR
2465 .ad
2466 .sp .6
2467 .RS 4n
2468 Display VLAN configuration for all VLAN links or for the specified VLAN link.
2469 .sp
2470 The \fBshow-vlan\fRsubcommand accepts the following options:
2471 .sp
2472 .ne 2
2473 .na
2474 \fB\fB-o\fR \fIfield\fR[,...], \fB--output\fR=\fIfield\fR[,...]\fR
2475 .ad
2476 .sp .6
2477 .RS 4n
2478 A case-insensitive, comma-separated list of output fields to display. The field
2479 name must be one of the fields listed below, or the special value \fBAll\fR, to
2480 display all fields. For each VLAN link, the following fields can be displayed:
2481 .sp
2482 .ne 2
2483 .na
2484 \fB\fBBLINK\fR\fR
2485 .ad
2486 .sp .6
2487 .RS 4n
2488 The name of the VLAN link.
2489 .RE

2491 .sp
2492 .ne 2
2493 .na
2494 \fB\fBVID\fR\fR
2495 .ad
2496 .sp .6
2497 .RS 4n
2498 The ID associated with the VLAN.
2499 .RE

2501 .sp
2502 .ne 2
2503 .na

```

```

2504 \fb\fBOVER\fr\fr
2505 .ad
2506 .sp .6
2507 .RS 4n
2508 The name of the physical link over which this VLAN is configured.
2509 .RE

2511 .sp
2512 .ne 2
2513 .na
2514 \fb\fbFLAGS\fr\fr
2515 .ad
2516 .sp .6
2517 .RS 4n
2518 A set of flags associated with the VLAN link. Possible flags are:
2519 .sp
2520 .ne 2
2521 .na
2522 \fb\fbf\fr\fr
2523 .ad
2524 .sp .6
2525 .RS 4n
2526 The VLAN was created using the \fb-f\fr option to \fbcreate-vlan\fr.
2527 .RE

2529 .sp
2530 .ne 2
2531 .na
2532 \fb\fbf\fr\fr
2533 .ad
2534 .sp .6
2535 .RS 4n
2536 The VLAN was implicitly created when the DLPI link was opened. These VLAN links
2537 are automatically deleted on last close of the DLPI link (for example, when the
2538 IP interface associated with the VLAN link is unplumbed).
2539 .RE

2541 Additional flags might be defined in the future.
2542 .RE

2544 .RE

2546 .sp
2547 .ne 2
2548 .na
2549 \fb\fb-p\fr, \fb--parseable\fr\fr
2550 .ad
2551 .sp .6
2552 .RS 4n
2553 Display using a stable machine-parseable format. The \fb-o\fr option is
2554 required with \fb-p\fr. See "Parseable Output Format", below.
2555 .RE

2557 .sp
2558 .ne 2
2559 .na
2560 \fb\fb-P\fr, \fb--persistent\fr\fr
2561 .ad
2562 .sp .6
2563 .RS 4n
2564 Display the persistent VLAN configuration rather than the state of the running
2565 system.
2566 .RE

2568 .RE

```

```

2570 .sp
2571 .ne 2
2572 .na
2573 \fb\fbdladm scan-wifi\fr [[\fb-p\fr] \fb-o\fr \fIfield\fr[...]]
2574 [\fIwifi-link\fr]\fr
2575 .ad
2576 .sp .6
2577 .RS 4n
2578 Scans for \fbWiFi\fr networks, either on all \fbWiFi\fr links, or just on the
2579 specified \fIwifi-link\fr.
2580 .sp
2581 By default, currently all fields but \fbBSTYPE\fr are displayed.
2582 .sp
2583 .ne 2
2584 .na
2585 \fb\fb-o\fr \fIfield\fr[...], \fb--output\fr=\fIfield\fr[...]\fr
2586 .ad
2587 .sp .6
2588 .RS 4n
2589 A case-insensitive, comma-separated list of output fields to display. The field
2590 name must be one of the fields listed below, or the special value \fbBall\fr to
2591 display all fields. For each \fbWiFi\fr network found, the following fields can
2592 be displayed:
2593 .sp
2594 .ne 2
2595 .na
2596 \fb\fbLINK\fr\fr
2597 .ad
2598 .sp .6
2599 .RS 4n
2600 The name of the link the \fbWiFi\fr network is on.
2601 .RE

2603 .sp
2604 .ne 2
2605 .na
2606 \fb\fbESSID\fr\fr
2607 .ad
2608 .sp .6
2609 .RS 4n
2610 The \fbESSID\fr (name) of the \fbWiFi\fr network.
2611 .RE

2613 .sp
2614 .ne 2
2615 .na
2616 \fb\fbBSSID\fr\fr
2617 .ad
2618 .sp .6
2619 .RS 4n
2620 Either the hardware address of the \fbWiFi\fr network's Access Point (for
2621 \fbBSS\fr networks), or the \fbWiFi\fr network's randomly generated unique
2622 token (for \fbIBSS\fr networks).
2623 .RE

2625 .sp
2626 .ne 2
2627 .na
2628 \fb\fbSEC\fr\fr
2629 .ad
2630 .sp .6
2631 .RS 4n
2632 Either \fbNone\fr for a \fbWiFi\fr network that uses no security, \fbWep\fr for
2633 a \fbWiFi\fr network that requires WEP (Wired Equivalent Privacy), or \fbWpa\fr
2634 for a WiFi network that requires WPA (Wi-Fi Protected Access).
2635 .RE

```

```

2637 .sp
2638 .ne 2
2639 .na
2640 \fB\fBMODE\fR\fR
2641 .ad
2642 .sp .6
2643 .RS 4n
2644 The supported connection modes: one or more of \fBa\fR, \fBb\fR, or \fBg\fR.
2645 .RE

2647 .sp
2648 .ne 2
2649 .na
2650 \fB\fBSTRENGTH\fR\fR
2651 .ad
2652 .sp .6
2653 .RS 4n
2654 The strength of the signal: one of \fBexcellent\fR, \fBvery good\fR,
2655 \fBgood\fR, \fBweak\fR, or \fBvery weak\fR.
2656 .RE

2658 .sp
2659 .ne 2
2660 .na
2661 \fB\fBSPEED\fR\fR
2662 .ad
2663 .sp .6
2664 .RS 4n
2665 The maximum speed of the \fBWi-Fi\fR network, in megabits per second.
2666 .RE

2668 .sp
2669 .ne 2
2670 .na
2671 \fB\fBSTYPE\fR\fR
2672 .ad
2673 .sp .6
2674 .RS 4n
2675 Either \fBbss\fR for \fBBSS\fR (infrastructure) networks, or \fBibss\fR for
2676 \fBIBSS\fR (ad-hoc) networks.
2677 .RE

2679 .RE

2681 .sp
2682 .ne 2
2683 .na
2684 \fB\fB-p\fR, \fB--parseable\fR\fR
2685 .ad
2686 .sp .6
2687 .RS 4n
2688 Display using a stable machine-parseable format. The \fB-o\fR option is
2689 required with \fB-p\fR. See "Parseable Output Format", below.
2690 .RE

2692 .RE

2694 .sp
2695 .ne 2
2696 .na
2697 \fB\fBdladm connect-wifi\fR [\fB-e\fR \fIessid\fR] [\fB-i\fR \fIbssid\fR]
2698 [\fB-k\fR \fIkey\fR, ...] [\fB-s\fR \fInone\fR | \fBwep\fR | \fBwpa\fR]
2699 [\fB-a\fR \fBopen\fR|\fBshared\fR] [\fB-b\fR \fBbss\fR|\fBibss\fR] [\fB-c\fR]
2700 [\fB-m\fR \fBa\fR|\fBb\fR|\fBg\fR] [\fB-T\fR \fItime\fR] [\fB-iwif-link\fR]\fR
2701 .ad

```

```

2702 .sp .6
2703 .RS 4n
2704 Connects to a \fBWi-Fi\fR network. This consists of four steps: \fIdiscovery\fR,
2705 \fIfiltration\fR, \fIprioritization\fR, and \fIassociation\fR. However, to
2706 enable connections to non-broadcast \fBWi-Fi\fR networks and to improve
2707 performance, if a \fBBSSID\fR or \fBESSID\fR is specified using the \fB-e\fR or
2708 \fB-i\fR options, then the first three steps are skipped and \fBconnect-wifi\fR
2709 immediately attempts to associate with a \fBBSSID\fR or \fBESSID\fR that
2710 matches the rest of the provided parameters. If this association fails, but
2711 there is a possibility that other networks matching the specified criteria
2712 exist, then the traditional discovery process begins as specified below.
2713 .sp
2714 The discovery step finds all available \fBWi-Fi\fR networks on the specified
2715 Wi-Fi link, which must not yet be connected. For administrative convenience, if
2716 there is only one \fBWi-Fi\fR link on the system, \fIiwifi-link\fR can be
2717 omitted.
2718 .sp
2719 Once discovery is complete, the list of networks is filtered according to the
2720 value of the following options:
2721 .sp
2722 .ne 2
2723 .na
2724 \fB\fB-e\fR \fIessid\fR \fB--essid\fR=\fIessid\fR\fR
2725 .ad
2726 .sp .6
2727 .RS 4n
2728 Networks that do not have the same \fIessid\fR are filtered out.
2729 .RE

2731 .sp
2732 .ne 2
2733 .na
2734 \fB\fB-b\fR \fBbss\fR|\fBibss\fR, \fB--bsstype\fR=\fBbss\fR|\fBibss\fR\fR
2735 .ad
2736 .sp .6
2737 .RS 4n
2738 Networks that do not have the same \fBbsstype\fR are filtered out.
2739 .RE

2741 .sp
2742 .ne 2
2743 .na
2744 \fB\fB-m\fR \fBa\fR|\fBb\fR|\fBg\fR, \fB--mode\fR=\fBa\fR|\fBb\fR|\fBg\fR\fR
2745 .ad
2746 .sp .6
2747 .RS 4n
2748 Networks not appropriate for the specified 802.11 mode are filtered out.
2749 .RE

2751 .sp
2752 .ne 2
2753 .na
2754 \fB\fB-k\fR \fIkey, ... \fR, \fB--key\fR=\fIkey, ... \fR\fR
2755 .ad
2756 .sp .6
2757 .RS 4n
2758 Use the specified \fBsecobj\fR named by the key to connect to the network.
2759 Networks not appropriate for the specified keys are filtered out.
2760 .RE

2762 .sp
2763 .ne 2
2764 .na
2765 \fB\fB-s\fR \fInone\fR|\fBwep\fR|\fBwpa\fR,
2766 \fB--sec\fR=\fInone\fR|\fBwep\fR|\fBwpa\fR\fR
2767 .ad

```

```

2768 .sp .6
2769 .RS 4n
2770 Networks not appropriate for the specified security mode are filtered out.
2771 .RE

2773 Next, the remaining networks are prioritized, first by signal strength, and
2774 then by maximum speed. Finally, an attempt is made to associate with each
2775 network in the list, in order, until one succeeds or no networks remain.
2776 .sp
2777 In addition to the options described above, the following options also control
2778 the behavior of \fBconnect-wifi\fR:
2779 .sp
2780 .ne 2
2781 .na
2782 \fB\fB-a\fR \fBopen\fR|\fBshared\fR, \fB--auth\fR=\fBopen\fR|\fBshared\fR\fR
2783 .ad
2784 .sp .6
2785 .RS 4n
2786 Connect using the specified authentication mode. By default, \fBopen\fR and
2787 \fBshared\fR are tried in order.
2788 .RE

2790 .sp
2791 .ne 2
2792 .na
2793 \fB\fB-c\fR, \fB--create-ibss\fR\fR
2794 .ad
2795 .sp .6
2796 .RS 4n
2797 Used with \fB-b ibss\fR to create a new ad-hoc network if one matching the
2798 specified \fBESSID\fR cannot be found. If no \fBESSID\fR is specified, then
2799 \fB-c -b ibss\fR always triggers the creation of a new ad-hoc network.
2800 .RE

2802 .sp
2803 .ne 2
2804 .na
2805 \fB\fB-T\fR \fBtime\fR, \fB--timeout\fR=\fBtime\fR\fR
2806 .ad
2807 .sp .6
2808 .RS 4n
2809 Specifies the number of seconds to wait for association to succeed. If
2810 \fBtime\fR is \fBforever\fR, then the associate will wait indefinitely. The
2811 current default is ten seconds, but this might change in the future. Timeouts
2812 shorter than the default might not succeed reliably.
2813 .RE

2815 .sp
2816 .ne 2
2817 .na
2818 \fB\fB-k\fR \fBkey,...\fR, \fB--key\fR=\fBkey,...\fR\fR
2819 .ad
2820 .sp .6
2821 .RS 4n
2822 In addition to the filtering previously described, the specified keys will be
2823 used to secure the association. The security mode to use will be based on the
2824 key class; if a security mode was explicitly specified, it must be compatible
2825 with the key class. All keys must be of the same class.
2826 .sp
2827 For security modes that support multiple key slots, the slot to place the key
2828 will be specified by a colon followed by an index. Therefore, \fB-k mykey:3\fR
2829 places \fBmykey\fR in slot 3. By default, slot 1 is assumed. For security modes
2830 that support multiple keys, a comma-separated list can be specified, with the
2831 first key being the active key.
2832 .RE

```

```

2834 .RE

2836 .sp
2837 .ne 2
2838 .na
2839 \fB\fBdladm disconnect-wifi\fR [\fB-a\fR] [\fIwifi-link\fR]\fR
2840 .ad
2841 .sp .6
2842 .RS 4n
2843 Disconnect from one or more \fBWi-Fi\fR networks. If \fIwifi-link\fR specifies a
2844 connected \fBWi-Fi\fR link, then it is disconnected. For administrative
2845 convenience, if only one \fBWi-Fi\fR link is connected, \fIwifi-link\fR can be
2846 omitted.
2847 .sp
2848 .ne 2
2849 .na
2850 \fB\fB-a\fR, \fB--all-links\fR\fR
2851 .ad
2852 .sp .6
2853 .RS 4n
2854 Disconnects from all connected links. This is primarily intended for use by
2855 scripts.
2856 .RE

2858 .RE

2860 .sp
2861 .ne 2
2862 .na
2863 \fB\fBdladm show-wifi\fR [[\fB-p\fR] \fB-o\fR \fIfield\fR,...]
2864 [\fIwifi-link\fR]\fR
2865 .ad
2866 .sp .6
2867 .RS 4n
2868 Shows \fBWi-Fi\fR configuration information either for all \fBWi-Fi\fR links or
2869 for the specified link \fIwifi-link\fR.
2870 .sp
2871 .ne 2
2872 .na
2873 \fB\fB-o\fR \fIfield,...\fR, \fB--output\fR=\fIfield\fR\fR
2874 .ad
2875 .sp .6
2876 .RS 4n
2877 A case-insensitive, comma-separated list of output fields to display. The field
2878 name must be one of the fields listed below, or the special value \fBall\fR, to
2879 display all fields. For each \fBWi-Fi\fR link, the following fields can be
2880 displayed:
2881 .sp
2882 .ne 2
2883 .na
2884 \fB\fBBLINK\fR\fR
2885 .ad
2886 .sp .6
2887 .RS 4n
2888 The name of the link being displayed.
2889 .RE

2891 .sp
2892 .ne 2
2893 .na
2894 \fB\fBSTATUS\fR\fR
2895 .ad
2896 .sp .6
2897 .RS 4n
2898 Either \fBconnected\fR if the link is connected, or \fBdisconnected\fR if it is
2899 not connected. If the link is disconnected, all remaining fields have the value

```

```

2900 \fB--\fR.
2901 .RE

2903 .sp
2904 .ne 2
2905 .na
2906 \fB\fBESSID\fR\fR
2907 .ad
2908 .sp .6
2909 .RS 4n
2910 The \fBESSID\fR (name) of the connected \fBWiFi\fR network.
2911 .RE

2913 .sp
2914 .ne 2
2915 .na
2916 \fB\fBSSID\fR\fR
2917 .ad
2918 .sp .6
2919 .RS 4n
2920 Either the hardware address of the \fBWiFi\fR network's Access Point (for
2921 \fBSS\fR networks), or the \fBWiFi\fR network's randomly generated unique
2922 token (for \fBIBSS\fR networks).
2923 .RE

2925 .sp
2926 .ne 2
2927 .na
2928 \fB\fBSEC\fR\fR
2929 .ad
2930 .sp .6
2931 .RS 4n
2932 Either \fBnone\fR for a \fBWiFi\fR network that uses no security, \fBwep\fR for
2933 a \fBWiFi\fR network that requires WEP, or \fBwpa\fR for a WiFi network that
2934 requires WPA.
2935 .RE

2937 .sp
2938 .ne 2
2939 .na
2940 \fB\fBMODE\fR\fR
2941 .ad
2942 .sp .6
2943 .RS 4n
2944 The supported connection modes: one or more of \fBa\fR, \fBb\fR, or \fBg\fR.
2945 .RE

2947 .sp
2948 .ne 2
2949 .na
2950 \fB\fBSTRENGTH\fR\fR
2951 .ad
2952 .sp .6
2953 .RS 4n
2954 The connection strength: one of \fBexcellent\fR, \fBvery good\fR, \fBgood\fR,
2955 \fBweak\fR, or \fBvery weak\fR.
2956 .RE

2958 .sp
2959 .ne 2
2960 .na
2961 \fB\fBSPEED\fR\fR
2962 .ad
2963 .sp .6
2964 .RS 4n
2965 The connection speed, in megabits per second.

```

```

2966 .RE

2968 .sp
2969 .ne 2
2970 .na
2971 \fB\fBAUTH\fR\fR
2972 .ad
2973 .sp .6
2974 .RS 4n
2975 Either \fBopen\fR or \fBshared\fR (see \fBconnect-wifi\fR).
2976 .RE

2978 .sp
2979 .ne 2
2980 .na
2981 \fB\fBSTYPE\fR\fR
2982 .ad
2983 .sp .6
2984 .RS 4n
2985 Either \fBbss\fR for \fBSS\fR (infrastructure) networks, or \fBibss\fR for
2986 \fBIBSS\fR (ad-hoc) networks.
2987 .RE

2989 By default, currently all fields but \fBAUTH\fR, \fBSSID\fR, \fBSTYPE\fR are
2990 displayed.
2991 .RE

2993 .sp
2994 .ne 2
2995 .na
2996 \fB\fB-p\fR, \fB--parseable\fR\fR
2997 .ad
2998 .sp .6
2999 .RS 4n
3000 Displays using a stable machine-parseable format. The \fB-o\fR option is
3001 required with \fB-p\fR. See "Parseable Output Format", below.
3002 .RE

3004 .RE

3006 .sp
3007 .ne 2
3008 .na
3009 \fB\fBdladm show-ether\fR [\fB-x\fR] [[\fB-p\fR] \fB-o\fR \fIfield\fR,...]
3010 [\fIfether-link\fR]\fR
3011 .ad
3012 .sp .6
3013 .RS 4n
3014 Shows state information either for all physical Ethernet links or for a
3015 specified physical Ethernet link.
3016 .sp
3017 The \fBshow-ether\fR subcommand accepts the following options:
3018 .sp
3019 .ne 2
3020 .na
3021 \fB\fB-o\fR \fIfield\fR,..., \fB--output\fR=\fIfield\fR\fR
3022 .ad
3023 .sp .6
3024 .RS 4n
3025 A case-insensitive, comma-separated list of output fields to display. The field
3026 name must be one of the fields listed below, or the special value \fBall\fR to
3027 display all fields. For each link, the following fields can be displayed:
3028 .sp
3029 .ne 2
3030 .na
3031 \fB\fBLINK\fR\fR

```

```

3032 .ad
3033 .sp .6
3034 .RS 4n
3035 The name of the link being displayed.
3036 .RE

3038 .sp
3039 .ne 2
3040 .na
3041 \fB\fBPTYPE\fR\fR
3042 .ad
3043 .sp .6
3044 .RS 4n
3045 Parameter type, where \fBcurrent\fR indicates the negotiated state of the link,
3046 \fBcapable\fR indicates capabilities supported by the device, \fBadv\fR
3047 indicates the advertised capabilities, and \fBpeeradv\fR indicates the
3048 capabilities advertised by the link-partner.
3049 .RE

3051 .sp
3052 .ne 2
3053 .na
3054 \fB\fBSTATE\fR\fR
3055 .ad
3056 .sp .6
3057 .RS 4n
3058 The state of the link.
3059 .RE

3061 .sp
3062 .ne 2
3063 .na
3064 \fB\fBAUTO\fR\fR
3065 .ad
3066 .sp .6
3067 .RS 4n
3068 A \fBByes\fR/\fBno\fR value indicating whether auto-negotiation is advertised.
3069 .RE

3071 .sp
3072 .ne 2
3073 .na
3074 \fB\fBSPEED-DUPLEX\fR\fR
3075 .ad
3076 .sp .6
3077 .RS 4n
3078 Combinations of speed and duplex values available. The units of speed are
3079 encoded with a trailing suffix of \fBG\fR (Gigabits/s) or \fBM\fR (Mb/s).
3080 Duplex values are encoded as \fBf\fR (full-duplex) or \fBh\fR (half-duplex).
3081 .RE

3083 .sp
3084 .ne 2
3085 .na
3086 \fB\fBPAUSE\fR\fR
3087 .ad
3088 .sp .6
3089 .RS 4n
3090 Flow control information. Can be \fBno\fR, indicating no flow control is
3091 available; \fBtx\fR, indicating that the end-point can transmit pause frames,
3092 but ignores any received pause frames; \fBrx\fR, indicating that the end-point
3093 receives and acts upon received pause frames; or \fBbi\fR, indicating
3094 bi-directional flow-control.
3095 .RE

3097 .sp

```

```

3098 .ne 2
3099 .na
3100 \fB\fBREM_FAULT\fR\fR
3101 .ad
3102 .sp .6
3103 .RS 4n
3104 Fault detection information. Valid values are \fBnone\fR or \fBfault\fR.
3105 .RE

3107 By default, all fields except \fBREM_FAULT\fR are displayed for the "current"
3108 \fBPTYPE\fR.
3109 .RE

3111 .sp
3112 .ne 2
3113 .na
3114 \fB\fB-p\fR, \fB--parseable\fR\fR
3115 .ad
3116 .sp .6
3117 .RS 4n
3118 Displays using a stable machine-parseable format. The \fB-o\fR option is
3119 required with \fB-p\fR. See "Parseable Output Format", below.
3120 .RE

3122 .sp
3123 .ne 2
3124 .na
3125 \fB\fB-x\fR, \fB--extended\fR\fR
3126 .ad
3127 .sp .6
3128 .RS 4n
3129 Extended output is displayed for \fBPTYPE\fR values of \fBcurrent\fR,
3130 \fBcapable\fR, \fBadv\fR and \fBpeeradv\fR.
3131 .RE

3133 .RE

3135 .sp
3136 .ne 2
3137 .na
3138 \fB\fBdladm set-linkprop\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] \fB-p\fR
3139 \fIprop\fR=\fIvalue\fR[,...] \fIlink\fR\fR
3140 .ad
3141 .sp .6
3142 .RS 4n
3143 Sets the values of one or more properties on the link specified. The list of
3144 properties and their possible values depend on the link type, the network
3145 device driver, and networking hardware. These properties can be retrieved using
3146 \fBshow-linkprop\fR.
3147 .sp
3148 .ne 2
3149 .na
3150 \fB\fB-t\fR, \fB--temporary\fR\fR
3151 .ad
3152 .sp .6
3153 .RS 4n
3154 Specifies that the changes are temporary. Temporary changes last until the next
3155 reboot.
3156 .RE

3158 .sp
3159 .ne 2
3160 .na
3161 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
3162 .ad
3163 .sp .6

```

```

3164 .RS 4n
3165 See "Options," above.
3166 .RE

3168 .sp
3169 .ne 2
3170 .na
3171 \fB\fB-p\fR \fIprop\fR=\fIvalue\fR[,...], \fB--prop\fR
3172 \fIprop\fR=\fIvalue\fR[,...]\fR
3173 .ad
3174 .br
3175 .na
3176 \fB\fR
3177 .ad
3178 .sp .6
3179 .RS 4n
3180 A comma-separated list of properties to set to the specified values.
3181 .RE

3183 Note that when the persistent value is set, the temporary value changes to the
3184 same value.
3185 .RE

3187 .sp
3188 .ne 2
3189 .na
3190 \fB\fBdladm reset-linkprop\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] [\fB-p\fR
3191 \fIprop\fR,...] \fIlink\fR\fR
3192 .ad
3193 .sp .6
3194 .RS 4n
3195 Resets one or more properties to their values on the link specified. Properties
3196 are reset to the values they had at startup. If no properties are specified,
3197 all properties are reset. See \fBshow-linkprop\fR for a description of
3198 properties.
3199 .sp
3200 .ne 2
3201 .na
3202 \fB\fB-t\fR, \fB--temporary\fR\fR
3203 .ad
3204 .sp .6
3205 .RS 4n
3206 Specifies that the resets are temporary. Values are reset to default values.
3207 Temporary resets last until the next reboot.
3208 .RE

3210 .sp
3211 .ne 2
3212 .na
3213 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
3214 .ad
3215 .sp .6
3216 .RS 4n
3217 See "Options," above.
3218 .RE

3220 .sp
3221 .ne 2
3222 .na
3223 \fB\fB-p\fR \fIprop, ... \fR, \fB--prop\fR=\fIprop, ... \fR\fR
3224 .ad
3225 .sp .6
3226 .RS 4n
3227 A comma-separated list of properties to reset.
3228 .RE

```

```

3230 Note that when the persistent value is reset, the temporary value changes to
3231 the same value.
3232 .RE

3234 .sp
3235 .ne 2
3236 .na
3237 \fB\fBdladm show-linkprop\fR [\fB-P\fR] [[\fB-c\fR] \fB-o\fR
3238 \fIfield\fR[,...]][\fB-p\fR \fIprop\fR[,...]] [\fIlink\fR]\fR
3239 .ad
3240 .sp .6
3241 .RS 4n
3242 Show the current or persistent values of one or more properties, either for all
3243 datalinks or for the specified link. By default, current values are shown. If
3244 no properties are specified, all available link properties are displayed. For
3245 each property, the following fields are displayed:
3246 .sp
3247 .ne 2
3248 .na
3249 \fB\fB-o\fR \fIfield\fR[,...], \fB--output\fR=\fIfield\fR\fR
3250 .ad
3251 .sp .6
3252 .RS 4n
3253 A case-insensitive, comma-separated list of output fields to display. The field
3254 name must be one of the fields listed below, or the special value \fBall\fR to
3255 display all fields. For each link, the following fields can be displayed:
3256 .sp
3257 .ne 2
3258 .na
3259 \fB\fBLINK\fR\fR
3260 .ad
3261 .sp .6
3262 .RS 4n
3263 The name of the datalink.
3264 .RE

3266 .sp
3267 .ne 2
3268 .na
3269 \fB\fBPROPERTY\fR\fR
3270 .ad
3271 .sp .6
3272 .RS 4n
3273 The name of the property.
3274 .RE

3276 .sp
3277 .ne 2
3278 .na
3279 \fB\fBPERM\fR\fR
3280 .ad
3281 .sp .6
3282 .RS 4n
3283 The read/write permissions of the property. The value shown is one of \fBro\fR
3284 or \fBrw\fR.
3285 .RE

3287 .sp
3288 .ne 2
3289 .na
3290 \fB\fBVALUE\fR\fR
3291 .ad
3292 .sp .6
3293 .RS 4n
3294 The current (or persistent) property value. If the value is not set, it is
3295 shown as \fB-\fR. If it is unknown, the value is shown as \fB?\fR. Persistent

```


3296 values that are not set or have been reset will be shown as \fB--\fR and will
 3297 use the system \fBDEFAULT\fR value (if any).
 3298 .RE

3300 .sp
 3301 .ne 2
 3302 .na
 3303 \fB\fBDEFAULT\fR
 3304 .ad
 3305 .sp .6
 3306 .RS 4n
 3307 The default value of the property. If the property has no default value,
 3308 \fB--\fR is shown.
 3309 .RE

3311 .sp
 3312 .ne 2
 3313 .na
 3314 \fB\fBPOSSIBLE\fR
 3315 .ad
 3316 .sp .6
 3317 .RS 4n
 3318 A comma-separated list of the values the property can have. If the values span
 3319 a numeric range, \fBimin\fR - \fBimax\fR might be shown as shorthand. If the
 3320 possible values are unknown or unbounded, \fB--\fR is shown.
 3321 .RE

3323 The list of properties depends on the link type and network device driver, and
 3324 the available values for a given property further depends on the underlying
 3325 network hardware and its state. General link properties are documented in the
 3326 \fBBLINK PROPERTIES\fR section. However, link properties that begin with
 3327 "\fB_\fR" (underbar) are specific to a given link or its underlying network
 3328 device and subject to change or removal. See the appropriate network device
 3329 driver man page for details.
 3330 .RE

3332 .sp
 3333 .ne 2
 3334 .na
 3335 \fB\fB-c\fR, \fB--parseable\fR
 3336 .ad
 3337 .sp .6
 3338 .RS 4n
 3339 Display using a stable machine-parseable format. The \fB-o\fR option is
 3340 required with this option. See "Parseable Output Format", below.
 3341 .RE

3343 .sp
 3344 .ne 2
 3345 .na
 3346 \fB\fB-P\fR, \fB--persistent\fR
 3347 .ad
 3348 .sp .6
 3349 .RS 4n
 3350 Display persistent link property information
 3351 .RE

3353 .sp
 3354 .ne 2
 3355 .na
 3356 \fB\fB-p\fR \fBiprop, ... \fR, \fB--prop\fR=\fBiprop, ... \fR
 3357 .ad
 3358 .sp .6
 3359 .RS 4n
 3360 A comma-separated list of properties to show. See the sections on link
 3361 properties following subcommand descriptions.

3362 .RE

3364 .RE

3366 .sp
 3367 .ne 2
 3368 .na
 3369 \fB\fBdladm create-secobj\fR [\fB-t\fR] [\fB-R\fR \fBroot-dir\fR] [\fB-f\fR
 3370 \fBfile\fR] \fB-c\fR \fBclass\fR \fBsecobj\fR
 3371 .ad
 3372 .sp .6
 3373 .RS 4n
 3374 Create a secure object named \fBsecobj\fR in the specified \fBclass\fR to be
 3375 later used as a WEP or WPA key in connecting to an encrypted network. The value
 3376 of the secure object can either be provided interactively or read from a file.
 3377 The sequence of interactive prompts and the file format depends on the class of
 3378 the secure object.
 3379 .sp
 3380 Currently, the classes \fBwep\fR and \fBwpa\fR are supported. The \fBWEP\fR
 3381 (Wired Equivalent Privacy) key can be either 5 or 13 bytes long. It can be
 3382 provided either as an \fBASCII\fR or hexadecimal string -- thus, \fB12345\fR
 3383 and \fB0x3132333435\fR are equivalent 5-byte keys (the \fB0x\fR prefix can be
 3384 omitted). A file containing a \fBWEP\fR key must consist of a single line using
 3385 either \fBWEP\fR key format. The WPA (Wi-Fi Protected Access) key must be
 3386 provided as an ASCII string with a length between 8 and 63 bytes.
 3387 .sp
 3388 This subcommand is only usable by users or roles that belong to the "Network
 3389 Link Security" \fBFBAC\fR profile.
 3390 .sp
 3391 .ne 2
 3392 .na
 3393 \fB\fB-c\fR \fBclass\fR, \fB--class\fR=\fBclass\fR
 3394 .ad
 3395 .sp .6
 3396 .RS 4n
 3397 \fBclass\fR can be \fBwep\fR or \fBwpa\fR. See preceding discussion.
 3398 .RE

3400 .sp
 3401 .ne 2
 3402 .na
 3403 \fB\fB-t\fR, \fB--temporary\fR
 3404 .ad
 3405 .sp .6
 3406 .RS 4n
 3407 Specifies that the creation is temporary. Temporary creation last until the
 3408 next reboot.
 3409 .RE

3411 .sp
 3412 .ne 2
 3413 .na
 3414 \fB\fB-R\fR \fBroot-dir\fR, \fB--root-dir\fR=\fBroot-dir\fR
 3415 .ad
 3416 .sp .6
 3417 .RS 4n
 3418 See "Options," above.
 3419 .RE

3421 .sp
 3422 .ne 2
 3423 .na
 3424 \fB\fB-f\fR \fBfile\fR, \fB--file\fR=\fBfile\fR
 3425 .ad
 3426 .sp .6
 3427 .RS 4n

```

3428 Specifies a file that should be used to obtain the secure object's value. The
3429 format of this file depends on the secure object class. See the \fBEXAMPLES\fR
3430 section for an example of using this option to set a \fBWEP\fR key.
3431 .RE

3433 .RE

3435 .sp
3436 .ne 2
3437 .na
3438 \fB\fBdladm delete-secobj\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR]
3439 \fIsecobj\fR[,...]\fR
3440 .ad
3441 .sp .6
3442 .RS 4n
3443 Delete one or more specified secure objects. This subcommand is only usable by
3444 users or roles that belong to the "Network Link Security" \fBRBAC\fR profile.
3445 .sp
3446 .ne 2
3447 .na
3448 \fB\fB-t\fR, \fB--temporary\fR\fR
3449 .ad
3450 .sp .6
3451 .RS 4n
3452 Specifies that the deletions are temporary. Temporary deletions last until the
3453 next reboot.
3454 .RE

3456 .sp
3457 .ne 2
3458 .na
3459 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
3460 .ad
3461 .sp .6
3462 .RS 4n
3463 See "Options," above.
3464 .RE

3466 .RE

3468 .sp
3469 .ne 2
3470 .na
3471 \fB\fBdladm show-secobj\fR [\fB-P\fR] [[\fB-p\fR] \fB-o\fR \fIfield\fR[,...]]
3472 [\fIsecobj\fR,...]\fR
3473 .ad
3474 .sp .6
3475 .RS 4n
3476 Show current or persistent secure object information. If one or more secure
3477 objects are specified, then information for each is displayed. Otherwise, all
3478 current or persistent secure objects are displayed.
3479 .sp
3480 By default, current secure objects are displayed, which are all secure objects
3481 that have either been persistently created and not temporarily deleted, or
3482 temporarily created.
3483 .sp
3484 For security reasons, it is not possible to show the value of a secure object.
3485 .sp
3486 .ne 2
3487 .na
3488 \fB\fB-o\fR \fIfield\fR[,...] , \fB--output\fR=\fIfield\fR[,...]\fR
3489 .ad
3490 .sp .6
3491 .RS 4n
3492 A case-insensitive, comma-separated list of output fields to display. The field
3493 name must be one of the fields listed below. For displayed secure object, the

```

```

3494 following fields can be shown:
3495 .sp
3496 .ne 2
3497 .na
3498 \fB\fBOBJECT\fR\fR
3499 .ad
3500 .sp .6
3501 .RS 4n
3502 The name of the secure object.
3503 .RE

3505 .sp
3506 .ne 2
3507 .na
3508 \fB\fBCLASS\fR\fR
3509 .ad
3510 .sp .6
3511 .RS 4n
3512 The class of the secure object.
3513 .RE

3515 .RE

3517 .sp
3518 .ne 2
3519 .na
3520 \fB\fB-p\fR, \fB--parseable\fR\fR
3521 .ad
3522 .sp .6
3523 .RS 4n
3524 Display using a stable machine-parseable format. The \fB-o\fR option is
3525 required with \fB-p\fR. See "Parseable Output Format", below.
3526 .RE

3528 .sp
3529 .ne 2
3530 .na
3531 \fB\fB-P\fR, \fB--persistent\fR\fR
3532 .ad
3533 .sp .6
3534 .RS 4n
3535 Display persistent secure object information
3536 .RE

3538 .RE

3540 .sp
3541 .ne 2
3542 .na
3543 \fB\fBdladm create-vmnic\fR [\fB-t\fR] \fB-l\fR \fIlink\fR [\fB-R\fR
3544 \fIroot-dir\fR] [\fB-m\fR \fIvalue\fR | auto | {factory [\fB-n\fR
3545 \fIslot-identifier\fR]} | {random [\fB-r\fR \fIprefix\fR}]}] [\fB-v\fR
3546 \fIvlan-id\fR] [\fB-p\fR \fIprop\fR=\fIvalue\fR[,...]] \fIvmnic-link\fR\fR
3547 .ad
3548 .sp .6
3549 .RS 4n
3550 Create a VMNIC with name \fIvmnic-link\fR over the specified link.
3551 .sp
3552 .ne 2
3553 .na
3554 \fB\fB-t\fR, \fB--temporary\fR\fR
3555 .ad
3556 .sp .6
3557 .RS 4n
3558 Specifies that the VMNIC is temporary. Temporary VMNICs last until the next
3559 reboot.

```

```

3560 .RE

3562 .sp
3563 .ne 2
3564 .na
3565 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
3566 .ad
3567 .sp .6
3568 .RS 4n
3569 See "Options," above.
3570 .RE

3572 .sp
3573 .ne 2
3574 .na
3575 \fB\fB-l\fR \fIlink\fR, \fB--link\fR=\fIlink\fR\fR
3576 .ad
3577 .sp .6
3578 .RS 4n
3579 \fIlink\fR can be a physical link or an \fBetherstub\fR.
3580 .RE

3582 .sp
3583 .ne 2
3584 .na
3585 \fB\fB-m\fR \fIvalue\fR | \fIkeyword\fR, \fB--mac-address\fR=\fIvalue\fR |
3586 \fIkeyword\fR\fR
3587 .ad
3588 .sp .6
3589 .RS 4n
3590 Sets the VNIC's MAC address based on the specified value or keyword. If
3591 \fIvalue\fR is not a keyword, it is interpreted as a unicast MAC address, which
3592 must be valid for the underlying NIC. The following special keywords can be
3593 used:
3594 .sp
3595 .ne 2
3596 .na
3597 \fBfactory [\fB-n\fR \fIslot-identifier\fR],\fR
3598 .ad
3599 .br
3600 .na
3601 \fBfactory [\fB--slot\fR=\fIslot-identifier\fR]\fR
3602 .ad
3603 .sp .6
3604 .RS 4n
3605 Assign a factory MAC address to the VNIC. When a factory MAC address is
3606 requested, \fB-m\fR can be combined with the \fB-n\fR option to specify a MAC
3607 address slot to be used. If \fB-n\fR is not specified, the system will choose
3608 the next available factory MAC address. The \fB-m\fR option of the
3609 \fBshow-phys\fR subcommand can be used to display the list of factory MAC
3610 addresses, their slot identifiers, and their availability.
3611 .RE

3613 .sp
3614 .ne 2
3615 .na
3616 \fB\fR
3617 .ad
3618 .br
3619 .na
3620 \fBrandom [\fB-r\fR \fIprefix\fR],\fR
3621 .ad
3622 .br
3623 .na
3624 \fBrandom [\fB--mac-prefix\fR=\fIprefix\fR]\fR
3625 .ad

```

```

3626 .sp .6
3627 .RS 4n
3628 Assign a random MAC address to the VNIC. A default prefix consisting of a valid
3629 IEEE OUI with the local bit set will be used. That prefix can be overridden
3630 with the \fB-r\fR option.
3631 .RE

3633 .sp
3634 .ne 2
3635 .na
3636 \fBauto\fR
3637 .ad
3638 .sp .6
3639 .RS 4n
3640 Try and use a factory MAC address first. If none is available, assign a random
3641 MAC address. \fBauto\fR is the default action if the \fB-m\fR option is not
3642 specified.
3643 .RE

3645 .sp
3646 .ne 2
3647 .na
3648 \fB\fB-v\fR \fIvlan-id\fR\fR
3649 .ad
3650 .sp .6
3651 .RS 4n
3652 Enable VLAN tagging for this VNIC. The VLAN tag will have id \fIvlan-id\fR.
3653 .RE

3655 .RE

3657 .sp
3658 .ne 2
3659 .na
3660 \fB\fB-p\fR \fIprop\fR=\fIvalue\fR,..., \fB--prop\fR
3661 \fIprop\fR=\fIvalue\fR,...\fR
3662 .ad
3663 .sp .6
3664 .RS 4n
3665 A comma-separated list of properties to set to the specified values.
3666 .RE

3668 .RE

3670 .sp
3671 .ne 2
3672 .na
3673 \fB\fB-dladm delete-vnic\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR]
3674 \fIvnic-link\fR\fR
3675 .ad
3676 .sp .6
3677 .RS 4n
3678 Deletes the specified VNIC.
3679 .sp
3680 .ne 2
3681 .na
3682 \fB\fB-t\fR, \fB--temporary\fR\fR
3683 .ad
3684 .sp .6
3685 .RS 4n
3686 Specifies that the deletion is temporary. Temporary deletions last until the
3687 next reboot.
3688 .RE

3690 .sp
3691 .ne 2

```

```

3692 .na
3693 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
3694 .ad
3695 .sp .6
3696 .RS 4n
3697 See "Options," above.
3698 .RE

3700 .RE

3702 .sp
3703 .ne 2
3704 .na
3705 \fB\fBdladm show-vnic\fR [\fB-p\fR] [\fB-s\fR [\fB-i\fR \fIinterval\fR]]
3706 [\fB-o\fR \fIfield\fR[,...]] [\fB-l\fR \fIlink\fR] [\fB-vnic-link\fR]\fR
3707 .ad
3708 .sp .6
3709 .RS 4n
3710 Show VNIC configuration information (the default) or statistics, for all VNICs,
3711 all VNICs on a link, or only the specified \fIvnic-link\fR.
3712 .sp
3713 .ne 2
3714 .na
3715 \fB\fB-o\fR \fIfield\fR[,...] , \fB--output\fR=\fIfield\fR[,...]\fR
3716 .ad
3717 .sp .6
3718 .RS 4n
3719 A case-insensitive, comma-separated list of output fields to display. The field
3720 name must be one of the fields listed below. The field name must be one of the
3721 fields listed below, or the special value \fBAll\fR to display all fields. By
3722 default (without \fB-o\fR), \fBshow-vnic\fR displays all fields.
3723 .sp
3724 .ne 2
3725 .na
3726 \fB\fBLINK\fR\fR
3727 .ad
3728 .sp .6
3729 .RS 4n
3730 The name of the VNIC.
3731 .RE

3733 .sp
3734 .ne 2
3735 .na
3736 \fB\fBBOVER\fR\fR
3737 .ad
3738 .sp .6
3739 .RS 4n
3740 The name of the physical link over which this VNIC is configured.
3741 .RE

3743 .sp
3744 .ne 2
3745 .na
3746 \fB\fBSPPEED\fR\fR
3747 .ad
3748 .sp .6
3749 .RS 4n
3750 The maximum speed of the VNIC, in megabits per second.
3751 .RE

3753 .sp
3754 .ne 2
3755 .na
3756 \fB\fBMACADDRESS\fR\fR
3757 .ad

```

```

3758 .sp .6
3759 .RS 4n
3760 MAC address of the VNIC.
3761 .RE

3763 .sp
3764 .ne 2
3765 .na
3766 \fB\fBMACADDRTYPE\fR\fR
3767 .ad
3768 .sp .6
3769 .RS 4n
3770 MAC address type of the VNIC. \fBdladm\fR distinguishes among the following MAC
3771 address types:
3772 .sp
3773 .ne 2
3774 .na
3775 \fB\fBRandom\fR\fR
3776 .ad
3777 .sp .6
3778 .RS 4n
3779 A random address assigned to the VNIC.
3780 .RE

3782 .sp
3783 .ne 2
3784 .na
3785 \fB\fBfactory\fR\fR
3786 .ad
3787 .sp .6
3788 .RS 4n
3789 A factory MAC address used by the VNIC.
3790 .RE

3792 .RE

3794 .RE

3796 .sp
3797 .ne 2
3798 .na
3799 \fB\fB-p\fR, \fB--parseable\fR\fR
3800 .ad
3801 .sp .6
3802 .RS 4n
3803 Display using a stable machine-parseable format. The \fB-o\fR option is
3804 required with \fB-p\fR. See "Parseable Output Format", below.
3805 .RE

3807 .sp
3808 .ne 2
3809 .na
3810 \fB\fB-P\fR, \fB--persistent\fR\fR
3811 .ad
3812 .sp .6
3813 .RS 4n
3814 Display the persistent VNIC configuration.
3815 .RE

3817 .sp
3818 .ne 2
3819 .na
3820 \fB\fB-s\fR, \fB--statistics\fR\fR
3821 .ad
3822 .sp .6
3823 .RS 4n

```

```

3824 Displays VNIC statistics.
3825 .RE

3827 .sp
3828 .ne 2
3829 .na
3830 \fB\fB-i\fR \fIinterval\fR, \fB--interval\fR=\fIinterval\fR\fR
3831 .ad
3832 .sp .6
3833 .RS 4n
3834 Used with the \fB-s\fR option to specify an interval, in seconds, at which
3835 statistics should be displayed. If this option is not specified, statistics
3836 will be displayed only once.
3837 .RE

3839 .sp
3840 .ne 2
3841 .na
3842 \fB\fB-l\fR \fIlink\fR, \fB--link\fR=\fIlink\fR\fR
3843 .ad
3844 .sp .6
3845 .RS 4n
3846 Display information for all VNICs on the named link.
3847 .RE

3849 .RE

3851 .sp
3852 .ne 2
3853 .na
3854 \fB\fR
3855 .ad
3856 .br
3857 .na
3858 \fB\fBdladm create-etherstub\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR]
3859 \fIetherstub\fR\fR
3860 .ad
3861 .sp .6
3862 .RS 4n
3863 Create an etherstub with the specified name.
3864 .sp
3865 .ne 2
3866 .na
3867 \fB\fB-t\fR, \fB--temporary\fR\fR
3868 .ad
3869 .sp .6
3870 .RS 4n
3871 Specifies that the etherstub is temporary. Temporary etherstubs do not persist
3872 across reboots.
3873 .RE

3875 .sp
3876 .ne 2
3877 .na
3878 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
3879 .ad
3880 .sp .6
3881 .RS 4n
3882 See "Options," above.
3883 .RE

3885 VNICs can be created on top of etherstubs instead of physical NICs. As with
3886 physical NICs, such a creation causes the stack to implicitly create a virtual
3887 switch between the VNICs created on top of the same etherstub.
3888 .RE

```

```

3890 .sp
3891 .ne 2
3892 .na
3893 \fB\fR
3894 .ad
3895 .br
3896 .na
3897 \fB\fBdladm delete-etherstub\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR]
3898 \fIetherstub\fR\fR
3899 .ad
3900 .sp .6
3901 .RS 4n
3902 Delete the specified etherstub.
3903 .sp
3904 .ne 2
3905 .na
3906 \fB\fB-t\fR, \fB--temporary\fR\fR
3907 .ad
3908 .sp .6
3909 .RS 4n
3910 Specifies that the deletion is temporary. Temporary deletions last until the
3911 next reboot.
3912 .RE

3914 .sp
3915 .ne 2
3916 .na
3917 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
3918 .ad
3919 .sp .6
3920 .RS 4n
3921 See "Options," above.
3922 .RE

3924 .RE

3926 .sp
3927 .ne 2
3928 .na
3929 \fB\fBdladm show-etherstub\fR [\fIetherstub\fR]\fR
3930 .ad
3931 .sp .6
3932 .RS 4n
3933 Show all configured etherstubs by default, or the specified etherstub if
3934 \fIetherstub\fR is specified.
3935 .RE

3937 .sp
3938 .ne 2
3939 .na
3940 \fB\fBdladm create-iptun\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] \fB-T\fR
3941 \fItype\fR [\fB-s\fR \fIsrc\fR] [\fB-d\fR \fIdst\fR] \fIiptun-link\fR\fR
3942 .ad
3943 .sp .6
3944 .RS 4n
3945 Create an IP tunnel link named \fIiptun-link\fR. Such links can additionally be
3946 protected with IPsec using \fBipsecconf\fR(1M).
3947 .sp
3948 An IP tunnel is conceptually comprised of two parts: a virtual link between two
3949 or more IP nodes, and an IP interface above this link that allows the system to
3950 transmit and receive IP packets encapsulated by the underlying link. This
3951 subcommand creates a virtual link. The \fBifconfig\fR(1M) command is used to
3952 configure IP interfaces above the link.
3953 .sp
3954 .ne 2
3955 .na

```

```

3956 \fB\fB-t\fR, \fB--temporary\fR\fR
3957 .ad
3958 .sp .6
3959 .RS 4n
3960 Specifies that the IP tunnel link is temporary. Temporary tunnels last until
3961 the next reboot.
3962 .RE

3964 .sp
3965 .ne 2
3966 .na
3967 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
3968 .ad
3969 .sp .6
3970 .RS 4n
3971 See "Options," above.
3972 .RE

3974 .sp
3975 .ne 2
3976 .na
3977 \fB\fB-T\fR \fItype\fR, \fB--tunnel-type\fR=\fItype\fR\fR
3978 .ad
3979 .sp .6
3980 .RS 4n
3981 Specifies the type of tunnel to be created. The type must be one of the
3982 following:
3983 .sp
3984 .ne 2
3985 .na
3986 \fB\fBip4\fR\fR
3987 .ad
3988 .sp .6
3989 .RS 4n
3990 A point-to-point, IP-over-IP tunnel between two IPv4 nodes. This type of tunnel
3991 requires IPv4 source and destination addresses to function. IPv4 and IPv6
3992 interfaces can be plumbed above such a tunnel to create IPv4-over-IPv4 and
3993 IPv6-over-IPv4 tunneling configurations.
3994 .RE

3996 .sp
3997 .ne 2
3998 .na
3999 \fB\fBip6\fR\fR
4000 .ad
4001 .sp .6
4002 .RS 4n
4003 A point-to-point, IP-over-IP tunnel between two IPv6 nodes as defined in IETF
4004 RFC 2473. This type of tunnel requires IPv6 source and destination addresses to
4005 function. IPv4 and IPv6 interfaces can be plumbed above such a tunnel to create
4006 IPv4-over-IPv6 and IPv6-over-IPv6 tunneling configurations.
4007 .RE

4009 .sp
4010 .ne 2
4011 .na
4012 \fB\fB6to4\fR\fR
4013 .ad
4014 .sp .6
4015 .RS 4n
4016 A 6to4, point-to-multipoint tunnel as defined in IETF RFC 3056. This type of
4017 tunnel requires an IPv4 source address to function. An IPv6 interface is
4018 plumbed on such a tunnel link to configure a 6to4 router.
4019 .RE

4021 .RE

```

```

4023 .sp
4024 .ne 2
4025 .na
4026 \fB\fB-s\fR \fIsrc\fR, \fB--tunnel-src\fR=\fIsrc\fR\fR
4027 .ad
4028 .sp .6
4029 .RS 4n
4030 Literal IP address or hostname corresponding to the tunnel source. If a
4031 hostname is specified, it will be resolved to IP addresses, and one of those IP
4032 addresses will be used as the tunnel source. Because IP tunnels are created
4033 before naming services have been brought online during the boot process, it is
4034 important that any hostname used be included in \fB/etc/hosts\fR.
4035 .RE

4037 .sp
4038 .ne 2
4039 .na
4040 \fB\fB-d\fR \fIdst\fR, \fB--tunnel-dst\fR=\fIdst\fR\fR
4041 .ad
4042 .sp .6
4043 .RS 4n
4044 Literal IP address or hostname corresponding to the tunnel destination.
4045 .RE

4047 .RE

4049 .sp
4050 .ne 2
4051 .na
4052 \fB\fBdladm modify-iptun\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR] [\fB-s\fR
4053 \fIsrc\fR] [\fB-d\fR \fIdst\fR] \fIlink\fR
4054 .ad
4055 .sp .6
4056 .RS 4n
4057 Modify the parameters of the specified IP tunnel.
4058 .sp
4059 .ne 2
4060 .na
4061 \fB\fB-t\fR, \fB--temporary\fR\fR
4062 .ad
4063 .sp .6
4064 .RS 4n
4065 Specifies that the modification is temporary. Temporary modifications last
4066 until the next reboot.
4067 .RE

4069 .sp
4070 .ne 2
4071 .na
4072 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
4073 .ad
4074 .sp .6
4075 .RS 4n
4076 See "Options," above.
4077 .RE

4079 .sp
4080 .ne 2
4081 .na
4082 \fB\fB-s\fR \fIsrc\fR, \fB--tunnel-src\fR=\fIsrc\fR\fR
4083 .ad
4084 .sp .6
4085 .RS 4n
4086 Specifies a new tunnel source address. See \fBcreate-iptun\fR for a
4087 description.

```

```

4088 .RE
4090 .sp
4091 .ne 2
4092 .na
4093 \fB\fB-d\fR \fIitdst\fR, \fB--tunnel-dst\fR=\fIitdst\fR\fR
4094 .ad
4095 .sp .6
4096 .RS 4n
4097 Specifies a new tunnel destination address. See \fBcreate-iptun\fR for a
4098 description.
4099 .RE
4101 .RE
4103 .sp
4104 .ne 2
4105 .na
4106 \fB\fBdladm delete-iptun\fR [\fB-t\fR] [\fB-R\fR \fIroot-dir\fR]
4107 \fIiptun-link\fR\fR
4108 .ad
4109 .sp .6
4110 .RS 4n
4111 Delete the specified IP tunnel link.
4112 .sp
4113 .ne 2
4114 .na
4115 \fB\fB-t\fR, \fB--temporary\fR\fR
4116 .ad
4117 .sp .6
4118 .RS 4n
4119 Specifies that the deletion is temporary. Temporary deletions last until the
4120 next reboot.
4121 .RE
4123 .sp
4124 .ne 2
4125 .na
4126 \fB\fB-R\fR \fIroot-dir\fR, \fB--root-dir\fR=\fIroot-dir\fR\fR
4127 .ad
4128 .sp .6
4129 .RS 4n
4130 See "Options," above.
4131 .RE
4133 .RE
4135 .sp
4136 .ne 2
4137 .na
4138 \fB\fBdladm show-iptun\fR [\fB-P\fR] [[\fB-p\fR] \fB-o\fR \fIfield\fR[...]]
4139 [\fIiptun-link\fR]\fR
4140 .ad
4141 .sp .6
4142 .RS 4n
4143 Show IP tunnel link configuration for a single IP tunnel or all IP tunnels.
4144 .sp
4145 .ne 2
4146 .na
4147 \fB\fB-P\fR, \fB--persistent\fR\fR
4148 .ad
4149 .sp .6
4150 .RS 4n
4151 Display the persistent IP tunnel configuration.
4152 .RE

```

```

4154 .sp
4155 .ne 2
4156 .na
4157 \fB\fB-p\fR, \fB--parseable\fR\fR
4158 .ad
4159 .sp .6
4160 .RS 4n
4161 Display using a stable machine-parseable format. The -o option is required with
4162 -p. See "Parseable Output Format", below.
4163 .RE
4165 .sp
4166 .ne 2
4167 .na
4168 \fB\fB-o\fR \fIfield\fR[...], \fB--output\fR=\fIfield\fR[...]\fR
4169 .ad
4170 .sp .6
4171 .RS 4n
4172 A case-insensitive, comma-separated list of output fields to display. The field
4173 name must be one of the fields listed below, or the special value \fBall\fR, to
4174 display all fields. By default (without \fB-o\fR), \fBshow-iptun\fR displays
4175 all fields.
4176 .sp
4177 .ne 2
4178 .na
4179 \fB\fBLINK\fR\fR
4180 .ad
4181 .sp .6
4182 .RS 4n
4183 The name of the IP tunnel link.
4184 .RE
4186 .sp
4187 .ne 2
4188 .na
4189 \fB\fBTYPE\fR\fR
4190 .ad
4191 .sp .6
4192 .RS 4n
4193 Type of tunnel as specified by the \fB-T\fR option of \fBcreate-iptun\fR.
4194 .RE
4196 .sp
4197 .ne 2
4198 .na
4199 \fB\fBFLAGS\fR\fR
4200 .ad
4201 .sp .6
4202 .RS 4n
4203 A set of flags associated with the IP tunnel link. Possible flags are:
4204 .sp
4205 .ne 2
4206 .na
4207 \fB\fBs\fR\fR
4208 .ad
4209 .sp .6
4210 .RS 4n
4211 The IP tunnel link is protected by IPsec policy. To display the IPsec policy
4212 associated with the tunnel link, enter:
4213 .sp
4214 .in +2
4215 .nf
4216 # \fBipseconf -ln -i \fItunnel-link\fR\fR
4217 .fi
4218 .in -2
4219 .sp

```

4221 See \fBipsecconf\fR(1M) for more details on how to configure IPsec policy.
 4222 .RE

4224 .sp
 4225 .ne 2
 4226 .na
 4227 \fB\fBi\fR\
 4228 .ad
 4229 .sp .6
 4230 .RS 4n
 4231 The IP tunnel link was implicitly created with \fBifconfig\fR(1M), and will be
 4232 automatically deleted when it is no longer referenced (that is, when the last
 4233 IP interface over the tunnel is unplumbed). See \fBifconfig\fR(1M) for details
 4234 on implicit tunnel creation.
 4235 .RE

4237 .RE

4239 .sp
 4240 .ne 2
 4241 .na
 4242 \fB\fBSOURCE\fR\
 4243 .ad
 4244 .sp .6
 4245 .RS 4n
 4246 The tunnel source address.
 4247 .RE

4249 .sp
 4250 .ne 2
 4251 .na
 4252 \fB\fBDESTINATION\fR\
 4253 .ad
 4254 .sp .6
 4255 .RS 4n
 4256 The tunnel destination address.
 4257 .RE

4259 .RE

4261 .RE

4263 .sp
 4264 .ne 2
 4265 .na
 4266 \fB\fBdladm show-usage\fR [\fB-a\fR] \fB-f\fR \fIfilename\fR [\fB-p\fR
 4267 \fIplotfile\fR \fB-F\fR \fIformat\fR] [\fB-s\fR \fItime\fR] [\fB-e\fR
 4268 \fItime\fR] [\fIlink\fR]\fR
 4269 .ad
 4270 .sp .6
 4271 .RS 4n
 4272 Show the historical network usage from a stored extended accounting file.
 4273 Configuration and enabling of network accounting through \fBacctadm\fR(1M) is
 4274 required. The default output will be the summary of network usage for the
 4275 entire period of time in which extended accounting was enabled.
 4276 .sp
 4277 .ne 2
 4278 .na
 4279 \fB\fB-a\fR\
 4280 .ad
 4281 .sp .6
 4282 .RS 4n
 4283 Display all historical network usage for the specified period of time during
 4284 which extended accounting is enabled. This includes the usage information for
 4285 the links that have already been deleted.

4286 .RE

4288 .sp
 4289 .ne 2
 4290 .na
 4291 \fB\fB-f\fR \fIfilename\fR, \fB--file\fR=\fIfilename\fR\
 4292 .ad
 4293 .sp .6
 4294 .RS 4n
 4295 Read extended accounting records of network usage from \fIfilename\fR.
 4296 .RE

4298 .sp
 4299 .ne 2
 4300 .na
 4301 \fB\fB-F\fR \fIformat\fR, \fB--format\fR=\fIformat\fR\
 4302 .ad
 4303 .sp .6
 4304 .RS 4n
 4305 Specifies the format of \fIplotfile\fR that is specified by the \fB-p\fR
 4306 option. As of this release, \fBgplot\fR is the only supported format.
 4307 .RE

4309 .sp
 4310 .ne 2
 4311 .na
 4312 \fB\fB-p\fR \fIplotfile\fR, \fB--plot\fR=\fIplotfile\fR\
 4313 .ad
 4314 .sp .6
 4315 .RS 4n
 4316 Write network usage data to a file of the format specified by the \fB-F\fR
 4317 option, which is required.
 4318 .RE

4320 .sp
 4321 .ne 2
 4322 .na
 4323 \fB\fB-s\fR \fItime\fR, \fB--start\fR=\fItime\fR\
 4324 .ad
 4325 .br
 4326 .na
 4327 \fB\fB-e\fR \fItime\fR, \fB--stop\fR=\fItime\fR\
 4328 .ad
 4329 .sp .6
 4330 .RS 4n
 4331 Start and stop times for data display. Time is in the format
 4332 \fIMM\fR/\fIDD\fR/\fIYYYY\fR,\fIhh\fR:\fImm\fR:\fIss\fR.
 4333 .RE

4335 .sp
 4336 .ne 2
 4337 .na
 4338 \fB\fBIlink\fR\
 4339 .ad
 4340 .sp .6
 4341 .RS 4n
 4342 If specified, display the network usage only for the named link. Otherwise,
 4343 display network usage for all links.
 4344 .RE

4346 .RE

4348 .SS "Parseable Output Format"
 4349 .sp
 4350 .LP
 4351 Many \fBdladm\fR subcommands have an option that displays output in a

4352 machine-parseable format. The output format is one or more lines of colon
 4353 (\fB:\fR) delimited fields. The fields displayed are specific to the subcommand
 4354 used and are listed under the entry for the \fB-o\fR option for a given
 4355 subcommand. Output includes only those fields requested by means of the
 4356 \fB-o\fR option, in the order requested.
 4357 .sp
 4358 .LP
 4359 When you request multiple fields, any literal colon characters are escaped by a
 4360 backslash (\fB\e\fR) before being output. Similarly, literal backslash
 4361 characters will also be escaped (\fB\e\fR). This escape format is parseable
 4362 by using shell \fBread\fR(1) functions with the environment variable
 4363 \fBIFS=\fR (see \fBEXAMPLES\fR, below). Note that escaping is not done when
 4364 you request only a single field.
 4365 .SS "General Link Properties"
 4366 .sp
 4367 .LP
 4368 The following general link properties are supported:
 4369 .sp
 4370 .ne 2
 4371 .na
 4372 \fB\fBautopush\fR\fR
 4373 .ad
 4374 .sp .6
 4375 .RS 4n
 4376 Specifies the set of STREAMS modules to push on the stream associated with a
 4377 link when its DLPI device is opened. It is a space-delimited list of modules.
 4378 .sp
 4379 The optional special character sequence \fB[anchor]\fR indicates that a STREAMS
 4380 anchor should be placed on the stream at the module previously specified in the
 4381 list. It is an error to specify more than one anchor or to have an anchor first
 4382 in the list.
 4383 .sp
 4384 The \fBautopush\fR property is preferred over the more general
 4385 \fBautopush\fR(1M) command.
 4386 .RE
 4388 .sp
 4389 .ne 2
 4390 .na
 4391 \fB\fBcpus\fR\fR
 4392 .ad
 4393 .sp .6
 4394 .RS 4n
 4395 Bind the processing of packets for a given data link to a processor or a set of
 4396 processors. The value can be a comma-separated list of one or more processor
 4397 ids. If the list consists of more than one processor, the processing will
 4398 spread out to all the processors. Connection to processor affinity and packet
 4399 ordering for any individual connection will be maintained.
 4400 .sp
 4401 The processor or set of processors are not exclusively reserved for the link.
 4402 Only the kernel threads and interrupts associated with processing of the link
 4403 are bound to the processor or the set of processors specified. In case it is
 4404 desired that processors be dedicated to the link, \fBbpsrset\fR(1M) can be used
 4405 to create a processor set and then specifying the processors from the processor
 4406 set to bind the link to.
 4407 .sp
 4408 If the link was already bound to processor or set of processors due to a
 4409 previous operation, the binding will be removed and the new set of processors
 4410 will be used instead.
 4411 .sp
 4412 The default is no CPU binding, which is to say that the processing of packets
 4413 is not bound to any specific processor or processor set.
 4414 .RE
 4416 .sp
 4417 .ne 2

4418 .na
 4419 \fB\fBlearn_limit\fR\fR
 4420 .ad
 4421 .sp .6
 4422 .RS 4n
 4423 Limits the number of new or changed MAC sources to be learned over a bridge
 4424 link. When the number exceeds this value, learning on that link is temporarily
 4425 disabled. Only non-VLAN, non-VNIC type links have this property.
 4426 .sp
 4427 The default value is \fB1000\fR. Valid values are greater or equal to 0.
 4428 .RE
 4430 .sp
 4431 .ne 2
 4432 .na
 4433 \fB\fBlearn_decay\fR\fR
 4434 .ad
 4435 .sp .6
 4436 .RS 4n
 4437 Specifies the decay rate for source changes limited by \fBlearn_limit\fR. This
 4438 number is subtracted from the counter for a bridge link every 5 seconds. Only
 4439 non-VLAN, non-VNIC type links have this property.
 4440 .sp
 4441 The default value is \fB200\fR. Valid values are greater or equal to 0.
 4442 .RE
 4444 .sp
 4445 .ne 2
 4446 .na
 4447 \fB\fBmaxbw\fR\fR
 4448 .ad
 4449 .sp .6
 4450 .RS 4n
 4451 Sets the full duplex bandwidth for the link. The bandwidth is specified as an
 4452 integer with one of the scale suffixes (\fBK\fR, \fBM\fR, or \fBG\fR for Kbps,
 4453 Mbps, and Gbps). If no units are specified, the input value will be read as
 4454 Mbps. The default is no bandwidth limit.
 4455 .RE
 4457 .sp
 4458 .ne 2
 4459 .na
 4460 \fB\fBpriority\fR\fR
 4461 .ad
 4462 .sp .6
 4463 .RS 4n
 4464 Sets the relative priority for the link. The value can be given as one of the
 4465 tokens \fBhigh\fR, \fBmedium\fR, or \fBlow\fR. The default is \fBhigh\fR.
 4466 .RE
 4468 .sp
 4469 .ne 2
 4470 .na
 4471 \fB\fBstp\fR\fR
 4472 .ad
 4473 .sp .6
 4474 .RS 4n
 4475 Enables or disables Spanning Tree Protocol on a bridge link. Setting this value
 4476 to \fB0\fR disables Spanning Tree, and puts the link into forwarding mode with
 4477 BPDU guarding enabled. This mode is appropriate for point-to-point links
 4478 connected only to end nodes. Only non-VLAN, non-VNIC type links have this
 4479 property. The default value is \fB1\fR, to enable STP.
 4480 .RE
 4482 .sp
 4483 .ne 2

```

4484 .na
4485 \fB\fBforward\fR\fR
4486 .ad
4487 .sp .6
4488 .RS 4n
4489 Enables or disables forwarding for a VLAN. Setting this value to \fB0\fR
4490 disables bridge forwarding for a VLAN link. Disabling bridge forwarding removes
4491 that VLAN from the "allowed set" for the bridge. The default value is \fB1\fR,
4492 to enable bridge forwarding for configured VLANs.
4493 .RE

4495 .sp
4496 .ne 2
4497 .na
4498 \fB\fBdefault_tag\fR\fR
4499 .ad
4500 .sp .6
4501 .RS 4n
4502 Sets the default VLAN ID that is assumed for untagged packets sent to and
4503 received from this link. Only non-VLAN, non-VNIC type links have this property.
4504 Setting this value to \fB0\fR disables the bridge forwarding of untagged
4505 packets to and from the port. The default value is \fBVLAN ID 1\fR. Valid
4506 values are from 0 to 4094.
4507 .RE

4509 .sp
4510 .ne 2
4511 .na
4512 \fB\fBstp_priority\fR\fR
4513 .ad
4514 .sp .6
4515 .RS 4n
4516 Sets the STP and RSTP Port Priority value, which is used to determine the
4517 preferred root port on a bridge. Lower numerical values are higher priority.
4518 The default value is \fB128\fR. Valid values range from 0 to 255.
4519 .RE

4521 .sp
4522 .ne 2
4523 .na
4524 \fB\fBstp_cost\fR\fR
4525 .ad
4526 .sp .6
4527 .RS 4n
4528 Sets the STP and RSTP cost for using the link. The default value is \fBauto\fR,
4529 which sets the cost based on link speed, using \fB100\fR for 10Mbps, \fB19\fR
4530 for 100Mbps, \fB4\fR for 1Gbps, and \fB2\fR for 10Gbps. Valid values range from
4531 1 to 65535.
4532 .RE

4534 .sp
4535 .ne 2
4536 .na
4537 \fB\fBstp_edge\fR\fR
4538 .ad
4539 .sp .6
4540 .RS 4n
4541 Enables or disables bridge edge port detection. If set to \fB0\fR (false), the
4542 system assumes that the port is connected to other bridges even if no bridge
4543 PDUs of any type are seen. The default value is \fB1\fR, which detects edge
4544 ports automatically.
4545 .RE

4547 .sp
4548 .ne 2
4549 .na

```

```

4550 \fB\fBstp_p2p\fR\fR
4551 .ad
4552 .sp .6
4553 .RS 4n
4554 Sets bridge point-to-point operation mode. Possible values are \fBtrue\fR,
4555 \fBfalse\fR, and \fBauto\fR. When set to \fBauto\fR, point-to-point connections
4556 are automatically discovered. When set to \fBtrue\fR, the port mode is forced
4557 to use point-to-point. When set to \fBfalse\fR, the port mode is forced to use
4558 normal multipoint mode. The default value is \fBauto\fR.
4559 .RE

4561 .sp
4562 .ne 2
4563 .na
4564 \fB\fBstp_mcheck\fR\fR
4565 .ad
4566 .sp .6
4567 .RS 4n
4568 Triggers the system to run the RSTP \fBForce BPDU Migration Check\fR procedure
4569 on this link. The procedure is triggered by setting the property value to
4570 \fB1\fR. The property is automatically reset back to \fB0\fR. This value cannot
4571 be set unless the following are true:
4572 .RS +4
4573 .TP
4574 .ie t \((bu
4575 .el o
4576 The link is bridged
4577 .RE
4578 .RS +4
4579 .TP
4580 .ie t \((bu
4581 .el o
4582 The bridge is protected by Spanning Tree
4583 .RE
4584 .RS +4
4585 .TP
4586 .ie t \((bu
4587 .el o
4588 The bridge \fBforce-protocol\fR value is at least 2 (RSTP)
4589 .RE
4590 The default value is 0.
4591 .RE

4593 .sp
4594 .ne 2
4595 .na
4596 \fB\fBzone\fR\fR
4597 .ad
4598 .sp .6
4599 .RS 4n
4600 Specifies the zone to which the link belongs. This property can be modified
4601 only temporarily through \fBdladm\fR, and thus the \fB-t\fR option must be
4602 specified. To modify the zone assignment such that it persists across reboots,
4603 please use \fBzonecfg\fR(1M). Possible values consist of any exclusive-IP zone
4604 currently running on the system. By default, the zone binding is as per
4605 \fBzonecfg\fR(1M).
4606 .RE

4608 .SS "Wifi Link Properties"
4609 .sp
4610 .LP
4611 The following \fBWiFi\fR link properties are supported. Note that the ability
4612 to set a given property to a given value depends on the driver and hardware.
4613 .sp
4614 .ne 2
4615 .na

```

```

4616 \fB\fBchannel\fR\fR
4617 .ad
4618 .sp .6
4619 .RS 4n
4620 Specifies the channel to use. This property can be modified only by certain
4621 \fBWi-Fi\fR links when in \fBIBSS\fR mode. The default value and allowed range
4622 of values varies by regulatory domain.
4623 .RE

4625 .sp
4626 .ne 2
4627 .na
4628 \fB\fBpowermode\fR\fR
4629 .ad
4630 .sp .6
4631 .RS 4n
4632 Specifies the power management mode of the \fBWi-Fi\fR link. Possible values are
4633 \fBBoff\fR (disable power management), \fBmax\fR (maximum power savings), and
4634 \fBfast\fR (performance-sensitive power management). Default is \fBBoff\fR.
4635 .RE

4637 .sp
4638 .ne 2
4639 .na
4640 \fB\fBradio\fR\fR
4641 .ad
4642 .sp .6
4643 .RS 4n
4644 Specifies the radio mode of the \fBWi-Fi\fR link. Possible values are \fBon\fR
4645 or \fBoff\fR. Default is \fBon\fR.
4646 .RE

4648 .sp
4649 .ne 2
4650 .na
4651 \fB\fBspeed\fR\fR
4652 .ad
4653 .sp .6
4654 .RS 4n
4655 Specifies a fixed speed for the \fBWi-Fi\fR link, in megabits per second. The
4656 set of possible values depends on the driver and hardware (but is shown by
4657 \fBshow-linkprop\fR); common speeds include 1, 2, 11, and 54. By default, there
4658 is no fixed speed.
4659 .RE

4661 .SS "Ethernet Link Properties"
4662 .sp
4663 .LP
4664 The following MII Properties, as documented in \fBIEEE802.3\fR(5), are
4665 supported in read-only mode:
4666 .RS +4
4667 .TP
4668 .ie t \(\bu
4669 .el o
4670 \fBduplex\fR
4671 .RE
4672 .RS +4
4673 .TP
4674 .ie t \(\bu
4675 .el o
4676 \fBstate\fR
4677 .RE
4678 .RS +4
4679 .TP
4680 .ie t \(\bu
4681 .el o

```

```

4682 \fBadv_autoneg_cap\fR
4683 .RE
4684 .RS +4
4685 .TP
4686 .ie t \(\bu
4687 .el o
4688 \fBadv_10gfdx_cap\fR
4689 .RE
4690 .RS +4
4691 .TP
4692 .ie t \(\bu
4693 .el o
4694 \fBadv_1000fdx_cap\fR
4695 .RE
4696 .RS +4
4697 .TP
4698 .ie t \(\bu
4699 .el o
4700 \fBadv_1000hdx_cap\fR
4701 .RE
4702 .RS +4
4703 .TP
4704 .ie t \(\bu
4705 .el o
4706 \fBadv_100fdx_cap\fR
4707 .RE
4708 .RS +4
4709 .TP
4710 .ie t \(\bu
4711 .el o
4712 \fBadv_100hdx_cap\fR
4713 .RE
4714 .RS +4
4715 .TP
4716 .ie t \(\bu
4717 .el o
4718 \fBadv_10fdx_cap\fR
4719 .RE
4720 .RS +4
4721 .TP
4722 .ie t \(\bu
4723 .el o
4724 \fBadv_10hdx_cap\fR
4725 .RE
4726 .sp
4727 .LP
4728 Each \fBadv_\fR property (for example, \fBadv_10fdx_cap\fR) also has a
4729 read/write counterpart \fBen_\fR property (for example, \fBen_10fdx_cap\fR)
4730 controlling parameters used at auto-negotiation. In the absence of Power
4731 Management, the \fBadv_\fR* speed/duplex parameters provide the values that are
4732 both negotiated and currently effective in hardware. However, with Power
4733 Management enabled, the speed/duplex capabilities currently exposed in hardware
4734 might be a subset of the set of bits that were used in initial link parameter
4735 negotiation. Thus the MII \fBadv_\fR* parameters are marked read-only, with an
4736 additional set of \fBen_\fR* parameters for configuring speed and duplex
4737 properties at initial negotiation.
4738 .sp
4739 .LP
4740 Note that the \fBadv_autoneg_cap\fR does not have an \fBen_autoneg_cap\fR
4741 counterpart: the \fBadv_autoneg_cap\fR is a 0/1 switch that turns off/on
4742 autonegotiation itself, and therefore cannot be impacted by Power Management.
4743 .sp
4744 .LP
4745 In addition, the following Ethernet properties are reported:
4746 .sp
4747 .ne 2

```

```

4748 .na
4749 \fB\fBspeed\fR\fR
4750 .ad
4751 .sp .6
4752 .RS 4n
4753 (read-only) The operating speed of the device, in Mbps.
4754 .RE

4756 .sp
4757 .ne 2
4758 .na
4759 \fB\fBmtu\fR\fR
4760 .ad
4761 .sp .6
4762 .RS 4n
4763 The maximum client SDU (Send Data Unit) supported by the device. Valid range is
4764 68-65536.
4765 .RE

4767 .sp
4768 .ne 2
4769 .na
4770 \fB\fBflowctrl\fR\fR
4771 .ad
4772 .sp .6
4773 .RS 4n
4774 Establishes flow-control modes that will be advertised by the device. Valid
4775 input is one of:
4776 .sp
4777 .ne 2
4778 .na
4779 \fB\fBno\fR\fR
4780 .ad
4781 .sp .6
4782 .RS 4n
4783 No flow control enabled.
4784 .RE

4786 .sp
4787 .ne 2
4788 .na
4789 \fB\fBbrx\fR\fR
4790 .ad
4791 .sp .6
4792 .RS 4n
4793 Receive, and act upon incoming pause frames.
4794 .RE

4796 .sp
4797 .ne 2
4798 .na
4799 \fB\fBtx\fR\fR
4800 .ad
4801 .sp .6
4802 .RS 4n
4803 Transmit pause frames to the peer when congestion occurs, but ignore received
4804 pause frames.
4805 .RE

4807 .sp
4808 .ne 2
4809 .na
4810 \fB\fBbi\fR\fR
4811 .ad
4812 .sp .6
4813 .RS 4n

```

```

4814 Bidirectional flow control.
4815 .RE

4817 Note that the actual settings for this value are constrained by the
4818 capabilities allowed by the device and the link partner.
4819 .RE

4821 .sp
4822 .ne 2
4823 .na
4824 \fB\fBtagmode\fR\fR
4825 .ad
4826 .sp .6
4827 .RS 4n
4828 This link property controls the conditions in which 802.1Q VLAN tags will be
4829 inserted in packets being transmitted on the link. Two mode values can be
4830 assigned to this property:
4831 .sp
4832 .ne 2
4833 .na
4834 \fB\fBnormal\fR\fR
4835 .ad
4836 .RS 12n
4837 Insert a VLAN tag in outgoing packets under the following conditions:
4838 .RS +4
4839 .TP
4840 .ie t \ (bu
4841 .el o
4842 The packet belongs to a VLAN.
4843 .RE
4844 .RS +4
4845 .TP
4846 .ie t \ (bu
4847 .el o
4848 The user requested priority tagging.
4849 .RE
4850 .RE

4852 .sp
4853 .ne 2
4854 .na
4855 \fB\fBvlanonly\fR\fR
4856 .ad
4857 .RS 12n
4858 Insert a VLAN tag only when the outgoing packet belongs to a VLAN. If a tag is
4859 being inserted in this mode and the user has also requested a non-zero
4860 priority, the priority is honored and included in the VLAN tag.
4861 .RE

4863 The default value is \fBvlanonly\fR.
4864 .RE

4866 .SS "IP Tunnel Link Properties"
4867 .sp
4868 .LP
4869 The following IP tunnel link properties are supported.
4870 .sp
4871 .ne 2
4872 .na
4873 \fB\fBhoplimit\fR\fR
4874 .ad
4875 .sp .6
4876 .RS 4n
4877 Specifies the IPv4 TTL or IPv6 hop limit for the encapsulating outer IP header
4878 of a tunnel link. This property exists for all tunnel types. The default value
4879 is 64.

```

```

4880 .RE
4882 .sp
4883 .ne 2
4884 .na
4885 \fB\fBencaplimit\fR\fR
4886 .ad
4887 .sp .6
4888 .RS 4n
4889 Specifies the IPv6 encapsulation limit for an IPv6 tunnel as defined in RFC
4890 2473. This value is the tunnel nesting limit for a given tunneled packet. The
4891 default value is 4. A value of 0 disables the encapsulation limit.
4892 .RE
4894 .SH EXAMPLES
4895 .LP
4896 \fBExample 1\fR \fRConfiguring an Aggregation
4897 .sp
4898 .LP
4899 To configure a data-link over an aggregation of devices \fBbge0\fR and
4900 \fBbge1\fR with key 1, enter the following command:
4902 .sp
4903 .in +2
4904 .nf
4905 # \fBdladm create-aggr -d bge0 -d bge1 1\fR
4906 .fi
4907 .in -2
4908 .sp
4910 .LP
4911 \fBExample 2\fR \fRConnecting to a WiFi Link
4912 .sp
4913 .LP
4914 To connect to the most optimal available unsecured network on a system with a
4915 single \fBWiFi\fR link (as per the prioritization rules specified for
4916 \fBconnect-wifi\fR), enter the following command:
4918 .sp
4919 .in +2
4920 .nf
4921 # \fBdladm connect-wifi\fR
4922 .fi
4923 .in -2
4924 .sp
4926 .LP
4927 \fBExample 3\fR \fRCreating a WiFi Key
4928 .sp
4929 .LP
4930 To interactively create the \fBWEP\fR key \fBmykey\fR, enter the following
4931 command:
4933 .sp
4934 .in +2
4935 .nf
4936 # \fBdladm create-secobj -c wep mykey\fR
4937 .fi
4938 .in -2
4939 .sp
4941 .sp
4942 .LP
4943 Alternatively, to non-interactively create the \fBWEP\fR key \fBmykey\fR using
4944 the contents of a file:

```

```

4946 .sp
4947 .in +2
4948 .nf
4949 # \fBumask 077\fR
4950 # \fBcat >/tmp/mykey.$$ <<EOF\fR
4951 \fB12345\fR
4952 \fBEOF\fR
4953 # \fBdladm create-secobj -c wep -f /tmp/mykey.$$ mykey\fR
4954 # \fBrm /tmp/mykey.$$ \fR
4955 .fi
4956 .in -2
4957 .sp
4959 .LP
4960 \fBExample 4\fR \fRConnecting to a Specified Encrypted WiFi Link
4961 .sp
4962 .LP
4963 To use key \fBmykey\fR to connect to \fBSSID\fR \fBwlan\fR on link \fBath0\fR,
4964 enter the following command:
4966 .sp
4967 .in +2
4968 .nf
4969 # \fBdladm connect-wifi -k mykey -e wlan ath0\fR
4970 .fi
4971 .in -2
4972 .sp
4974 .LP
4975 \fBExample 5\fR \fRChanging a Link Property
4976 .sp
4977 .LP
4978 To set \fBpowermode\fR to the value \fBfast\fR on link \fBpcwl0\fR, enter the
4979 following command:
4981 .sp
4982 .in +2
4983 .nf
4984 # \fBdladm set-linkprop -p powermode=fast pcwl0\fR
4985 .fi
4986 .in -2
4987 .sp
4989 .LP
4990 \fBExample 6\fR \fRConnecting to a WPA-Protected WiFi Link
4991 .sp
4992 .LP
4993 Create a WPA key \fBpsk\fR and enter the following command:
4995 .sp
4996 .in +2
4997 .nf
4998 # \fBdladm create-secobj -c wpa psk\fR
4999 .fi
5000 .in -2
5001 .sp
5003 .sp
5004 .LP
5005 To then use key \fBpsk\fR to connect to ESSID \fBwlan\fR on link \fBath0\fR,
5006 enter the following command:
5008 .sp
5009 .in +2
5010 .nf
5011 # \fBdladm connect-wifi -k psk -e wlan ath0\fR

```

```

5012 .fi
5013 .in -2
5014 .sp

5016 .LP
5017 \fBExample 7 \fRRenaming a Link
5018 .sp
5019 .LP
5020 To rename the \fBbge0\fR link to \fBmgmt0\fR, enter the following command:

5022 .sp
5023 .in +2
5024 .nf
5025 # \fBdladm rename-link bge0 mgmt0\fR
5026 .fi
5027 .in -2
5028 .sp

5030 .LP
5031 \fBExample 8 \fRReplacing a Network Card
5032 .sp
5033 .LP
5034 Consider that the \fBbge0\fR device, whose link was named \fBmgmt0\fR as shown
5035 in the previous example, needs to be replaced with a \fBce0\fR device because
5036 of a hardware failure. The \fBbge0\fR NIC is physically removed, and replaced
5037 with a new \fBce0\fR NIC. To associate the newly added \fBce0\fR device with
5038 the \fBmgmt0\fR configuration previously associated with \fBbge0\fR, enter the
5039 following command:

5041 .sp
5042 .in +2
5043 .nf
5044 # \fBdladm rename-link ce0 mgmt0\fR
5045 .fi
5046 .in -2
5047 .sp

5049 .LP
5050 \fBExample 9 \fRRemoving a Network Card
5051 .sp
5052 .LP
5053 Suppose that in the previous example, the intent is not to replace the
5054 \fBbge0\fR NIC with another NIC, but rather to remove and not replace the
5055 hardware. In that case, the \fBmgmt0\fR datalink configuration is not slated to
5056 be associated with a different physical device as shown in the previous
5057 example, but needs to be deleted. Enter the following command to delete the
5058 datalink configuration associated with the \fBmgmt0\fR datalink, whose physical
5059 hardware (\fBbge0\fR in this case) has been removed:

5061 .sp
5062 .in +2
5063 .nf
5064 # \fBdladm delete-phys mgmt0\fR
5065 .fi
5066 .in -2
5067 .sp

5069 .LP
5070 \fBExample 10 \fRUsing Parseable Output to Capture a Single Field
5071 .sp
5072 .LP
5073 The following assignment saves the MTU of link \fBnet0\fR to a variable named
5074 \fBmtu\fR.

5076 .sp
5077 .in +2

```

```

5078 .nf
5079 # \fBmtu=\`dladm show-link -p -o mtu net0`\fR
5080 .fi
5081 .in -2
5082 .sp

5084 .LP
5085 \fBExample 11 \fRUsing Parseable Output to Iterate over Links
5086 .sp
5087 .LP
5088 The following script displays the state of each link on the system.

5090 .sp
5091 .in +2
5092 .nf
5093 # \fBdladm show-link -p -o link,state | while IFS=: read link state; do
5094     print "Link $link is in state $state"
5095     done\fR
5096 .fi
5097 .in -2
5098 .sp

5100 .LP
5101 \fBExample 12 \fRConfiguring VNICs
5102 .sp
5103 .LP
5104 Create two VNICs with names \fBhello0\fR and \fBtest1\fR over a single physical
5105 link \fBbge0\fR:

5107 .sp
5108 .in +2
5109 .nf
5110 # \fBdladm create-vnic -l bge0 hello0\fR
5111 # \fBdladm create-vnic -l bge0 test1\fR
5112 .fi
5113 .in -2
5114 .sp

5116 .LP
5117 \fBExample 13 \fRConfiguring VNICs and Allocating Bandwidth and Priority
5118 .sp
5119 .LP
5120 Create two VNICs with names \fBhello0\fR and \fBtest1\fR over a single physical
5121 link \fBbge0\fR and make \fBhello0\fR a high priority VNIC with a
5122 factory-assigned MAC address with a maximum bandwidth of 50 Mbps. Make
5123 \fBtest1\fR a low priority VNIC with a random MAC address and a maximum
5124 bandwidth of 100Mbps.

5126 .sp
5127 .in +2
5128 .nf
5129 # \fBdladm create-vnic -l bge0 -m factory -p maxbw=50,priority=high hello0\fR
5130 # \fBdladm create-vnic -l bge0 -m random -p maxbw=100M,priority=low test1\fR
5131 .fi
5132 .in -2
5133 .sp

5135 .LP
5136 \fBExample 14 \fRConfiguring a VNIC with a Factory MAC Address
5137 .sp
5138 .LP
5139 First, list the available factory MAC addresses and choose one of them:

5141 .sp
5142 .in +2
5143 .nf

```

```

5144 # \fBdladm show-phys -m bge0\fR
5145 LINK          SLOT      ADDRESS          INUSE    CLIENT
5146 bge0          primary   0:e0:81:27:d4:47 yes      bge0
5147 bge0          1         8:0:20:fe:4e:a5 no
5148 bge0          2         8:0:20:fe:4e:a6 no
5149 bge0          3         8:0:20:fe:4e:a7 no
5150 .fi
5151 .in -2
5152 .sp

5154 .sp
5155 .LP
5156 Create a VNIC named \fBhello0\fR and use slot 1's address:

5158 .sp
5159 .in +2
5160 .nf
5161 # \fBdladm create-vnic -l bge0 -m factory -n 1 hello0\fR
5162 # \fBdladm show-phys -m bge0\fR
5163 LINK          SLOT      ADDRESS          INUSE    CLIENT
5164 bge0          primary   0:e0:81:27:d4:47 yes      bge0
5165 bge0          1         8:0:20:fe:4e:a5 yes      hello0
5166 bge0          2         8:0:20:fe:4e:a6 no
5167 bge0          3         8:0:20:fe:4e:a7 no
5168 .fi
5169 .in -2
5170 .sp

5172 .LP
5173 \fBExample 15 \fRCreating a VNIC with User-Specified MAC Address, Binding it to
5174 Set of Processors
5175 .sp
5176 .LP
5177 Create a VNIC with name \fBhello0\fR, with a user specified MAC address, and a
5178 processor binding \fB0, 1, 2, 3\fR.

5180 .sp
5181 .in +2
5182 .nf
5183 # \fBdladm create-vnic -l bge0 -m 8:0:20:fe:4e:b8 -p cpus=0,1,2,3 hello0\fR
5184 .fi
5185 .in -2
5186 .sp

5188 .LP
5189 \fBExample 16 \fRCreating a Virtual Network Without a Physical NIC
5190 .sp
5191 .LP
5192 First, create an etherstub with name \fBstub1\fR:

5194 .sp
5195 .in +2
5196 .nf
5197 # \fBdladm create-etherstub stub1\fR
5198 .fi
5199 .in -2
5200 .sp

5202 .sp
5203 .LP
5204 Create two VNICs with names \fBhello0\fR and \fBtest1\fR on the etherstub. This
5205 operation implicitly creates a virtual switch connecting \fBhello0\fR and
5206 \fBtest1\fR.

5208 .sp
5209 .in +2

```

```

5210 .nf
5211 # \fBdladm create-vnic -l stub1 hello0\fR
5212 # \fBdladm create-vnic -l stub1 test1\fR
5213 .fi
5214 .in -2
5215 .sp

5217 .LP
5218 \fBExample 17 \fRShowing Network Usage
5219 .sp
5220 .LP
5221 Network usage statistics can be stored using the extended accounting facility,
5222 \fBacctadm\fR(1M).

5224 .sp
5225 .in +2
5226 .nf
5227 # \fBacctadm -e basic -f /var/log/net.log net\fR
5228 # \fBacctadm net\fR
5229     Network accounting: active
5230     Network accounting file: /var/log/net.log
5231     Tracked Network resources: basic
5232     Untracked Network resources: src_ip,dst_ip,src_port,dst_port,protocol,
5233                                   dsfield
5234 .fi
5235 .in -2
5236 .sp

5238 .sp
5239 .LP
5240 The saved historical data can be retrieved in summary form using the
5241 \fBshow-usage\fR subcommand:

5243 .sp
5244 .in +2
5245 .nf
5246 # \fBdladm show-usage -f /var/log/net.log\fR
5247 LINK          DURATION  IPACKETS  RBYTES   OPACKETS  OBYTES   BANDWIDTH
5248 e1000g0       80         1031      546908   0          0         2.44 Kbps
5249 .fi
5250 .in -2
5251 .sp

5253 .LP
5254 \fBExample 18 \fRDisplaying Bridge Information
5255 .sp
5256 .LP
5257 The following commands use the \fBshow-bridge\fR subcommand with no and various
5258 options.

5260 .sp
5261 .in +2
5262 .nf
5263 # \fBdladm show-bridge\fR
5264 BRIDGE        PROTECT ADDRESS          PRIORITY DESROOT
5265 foo           stp    32768/8:0:20:bf:f 32768     8192/0:d0:0:76:14:38
5266 bar           stp    32768/8:0:20:e5:8 32768     8192/0:d0:0:76:14:38

5268 # \fBdladm show-bridge -l foo\fR
5269 LINK          STATE      UPTIME    DESROOT
5270 hme0          forwarding 117       8192/0:d0:0:76:14:38
5271 qfel          forwarding 117       8192/0:d0:0:76:14:38

5273 # \fBdladm show-bridge -s foo\fR
5274 BRIDGE        DROPS     FORWARDS
5275 foo           0         302

```

```

5277 # \fBdladm show-bridge -ls foo\fr
5278 LINK          DROPS      RECV      XMIT
5279 hme0           0          360832    31797
5280 qfel           0          322311    356852

5282 # \fBdladm show-bridge -f foo\fr
5283 DEST          AGE      FLAGS     OUTPUT
5284 8:0:20:bc:a7:dc 10.860  --      hme0
5285 8:0:20:bf:f9:69  --      L       hme0
5286 8:0:20:c0:20:26 17.420  --      hme0
5287 8:0:20:e5:86:11 --      L       qfel
5288 .fi
5289 .in -2
5290 .sp

5292 .LP
5293 \fBExample 19 \frCreating an IPv4 Tunnel
5294 .sp
5295 .LP
5296 The following sequence of commands creates and then displays a persistent IPv4
5297 tunnel link named \fBmytunnel0\fr between 66.1.2.3 and 192.4.5.6:

5299 .sp
5300 .in +2
5301 .nf
5302 # \fBdladm create-iptun -T ipv4 -s 66.1.2.3 -d 192.4.5.6 mytunnel0\fr
5303 # \fBdladm show-iptun mytunnel0\fr
5304 LINK          TYPE     FLAGS     SOURCE          DESTINATION
5305 mytunnel0     ipv4    --        66.1.2.3        192.4.5.6
5306 .fi
5307 .in -2
5308 .sp

5310 .sp
5311 .LP
5312 A point-to-point IP interface can then be created over this tunnel link:

5314 .sp
5315 .in +2
5316 .nf
5317 # \fBbifconfig mytunnel0 plumb 10.1.0.1 10.1.0.2 up\fr
5318 .fi
5319 .in -2
5320 .sp

5322 .sp
5323 .LP
5324 As with any other IP interface, configuration persistence for this IP interface
5325 is achieved by placing the desired \fBbifconfig\fr commands (in this case, the
5326 command for "\fB10.1.0.1 10.1.0.2\fr") into \fB/etc/hostname.mytunnel0\fr.

5328 .LP
5329 \fBExample 20 \frCreating a 6to4 Tunnel
5330 .sp
5331 .LP
5332 The following command creates a 6to4 tunnel link. The IPv4 address of the 6to4
5333 router is 75.10.11.12.

5335 .sp
5336 .in +2
5337 .nf
5338 # \fBdladm create-iptun -T 6to4 -s 75.10.11.12 sitetunnel0\fr
5339 # \fBdladm show-iptun sitetunnel0\fr
5340 LINK          TYPE     FLAGS     SOURCE          DESTINATION
5341 sitetunnel0   6to4    --        75.10.11.12    --

```

```

5342 .fi
5343 .in -2
5344 .sp

5346 .sp
5347 .LP
5348 The following command plumbs an IPv6 interface on this tunnel:

5350 .sp
5351 .in +2
5352 .nf
5353 # \fBbifconfig sitetunnel0 inet6 plumb up\fr
5354 # \fBbifconfig sitetunnel0 inet6\fr
5355 sitetunnel0: flags=2200041 <UP,RUNNING,NUD,IPv6> mtu 65515 index 3
5356     inet tunnel src 75.10.11.12
5357     tunnel hop limit 64
5358     inet6 2002:4b0a:b0c::1/16
5359 .fi
5360 .in -2
5361 .sp

5363 .sp
5364 .LP
5365 Note that the system automatically configures the IPv6 address on the 6to4 IP
5366 interface. See \fBbifconfig\fr(1M) for a description of how IPv6 addresses are
5367 configured on 6to4 tunnel links.

5369 .SH ATTRIBUTES
5370 .sp
5371 .LP
5372 See \fBattributes\fr(5) for descriptions of the following attributes:
5373 .sp
5374 .LP
5375 \fB/usr/sbin\fr
5376 .sp

5378 .sp
5379 .TS
5380 box:
5381 c | c
5382 l | l .
5383 ATTRIBUTE TYPE     ATTRIBUTE VALUE
5384 _
5385 Interface Stability     Committed
5386 .TE

5388 .sp
5389 .LP
5390 \fB/usr/sbin\fr
5391 .sp

5393 .sp
5394 .TS
5395 box:
5396 c | c
5397 l | l .
5398 ATTRIBUTE TYPE     ATTRIBUTE VALUE
5399 _
5400 Interface Stability     Committed
5401 .TE

5403 .SH SEE ALSO
5404 .sp
5405 .LP
5406 \fBacctadm\fr(1M), \fBautopush\fr(1M), \fBbifconfig\fr(1M), \fBbipadm\fr(1M),
5407 \fBbipsecconf\fr(1M), \fBbndd\fr(1M), \fBbprset\fr(1M), \fBwpad\fr(1M),

```



```
5408 \fBzonecfg\fR(1M), \fBattributes\fR(5), \fBIEEE802.3\fR(5), \fBdlpi\fR(7P)
5406 \fBacctadm\fR(1M), \fBautopush\fR(1M), \fBifconfig\fR(1M), \fBipsecconf\fR(1M),
5407 \fBndd\fR(1M), \fBprset\fR(1M), \fBwpad\fR(1M), \fBzonecfg\fR(1M),
5408 \fBattributes\fR(5), \fBIEEE802.3\fR(5), \fBdlpi\fR(7P)
5409 .SH NOTES
5410 .sp
5411 .LP
5412 The preferred method of referring to an aggregation in the aggregation
5413 subcommands is by its link name. Referring to an aggregation by its integer
5414 \fIkey\fR is supported for backward compatibility, but is not necessary. When
5415 creating an aggregation, if a \fIkey\fR is specified instead of a link name,
5416 the aggregation's link name will be automatically generated by \fBdladm\fR as
5417 \fBaggr\fR\fIkey\fR.
```

```

*****
66454 Sun Oct 28 17:19:47 2012
new/usr/src/man/man1m/ifconfig.1m
815 Need ipadm(1M) manual page
*****
1 '\" te
2 .\" Copyright (C) 2012, Darren Reed. All rights reserved
3 .\" Copyright (C) 2009, Sun Microsystems, Inc. All Rights Reserved
4 .\" Copyright 1989 AT&T
5 .\" Copyright (c) 1983 Regents of the University of California. All rights reser
6 .TH IFCONFIG 1M "July 23, 2012"
7 .SH NAME
8 ifconfig \- configure network interface parameters
9 .SH SYNOPSIS
10 .LP
11 .nf
12 \fBifconfig\fR \fIinterface\fR [\fIaddress_family\fR] [\fIaddress\fR [\fI/prefix
13 [\fIidest_address\fR]] [\fBaddif\fR \fIaddress\fR [\fI/prefix_length\fR]]
14 [\fBremoveif\fR \fIaddress\fR [\fI/prefix_length\fR]] [\fBarp\fR | \fB-arp\fR]
15 [\fBauth_algs\fR \fIauthentication_algorithm\fR] [\fBencr_algs\fR \fIencryption
16 [\fBencr_auth_algs\fR \fIauthentication_algorithm\fR] [\fBauto-revarp\fR]
17 [\fBbroadcast\fR \fIaddress\fR] [\fBdeprecated\fR | \fB-deprecated\fR]
18 [\fBpreferred\fR | \fB-preferred\fR] [\fBdestination\fR \fIidest_address\fR]
19 [ether [\fIaddress\fR]] [\fBfailover\fR | \fB-failover\fR] [\fBgroup\fR
20 [\fIname\fR | ""] \fBfR] [\fBindex\fR \fIif_index\fR] [ipmp] [\fBmetric\fR \fIn
21 [modinsert \fImod_name@pos\fR] [modremove \fImod_name@pos\fR]
22 [\fBmtu\fR \fIn\fR] [\fBnetmask\fR \fImask\fR] [\fBplumb\fR] [\fBunplumb\fR] [\fB
23 | \fB-private\fR] [\fBnud\fR | \fB-nud\fR] [\fBset\fR [\fIaddress\fR] [\fI/netm
24 [\fBstandby\fR | \fB-standby\fR] [\fBsubnet\fR \fIsubnet_address\fR] [\fBtdst\f
25 \fItunnel_dest_address\fR] [\fBtoken\fR \fIaddress\fR/\fIprefix_length\fR]
26 [\fBtsrc\fR \fItunnel_src_address\fR] [\fBtrailers\fR | \fB-trailers\fR]
27 [\fBup\fR] [\fBdown\fR] [\fBusesrc\fR [\fIname\fR | none]] [\fBxmit\fR | \fB-xm
28 [\fBencaplimit\fR \fIn\fR | \fB-encaplimit\fR] [\fBthoplimit\fR \fIn\fR] [\fBBro
29 | \fB-router\fR] [zone \fIzonename\fR | \fB-zone\fR | \fB-all-zones\fR]
30 .fi

32 .LP
33 .nf
34 \fBifconfig\fR [\fIaddress_family\fR] \fIinterface\fR {\fBauto-dhcp\fR | \fBdhcp
35 [\fBwait\fR \fIseconds\fR] \fBdrop\fR | \fBxtend\fR | \fBinform\fR | \fBping\f
36 | \fBrelease\fR | \fBstart\fR | \fBstatus\fR
37 .fi

39 .SH DESCRIPTION
40 .sp
41 .LP
42 The command \fBifconfig\fR is used to assign an address to a network interface
43 and to configure network interface parameters. The \fBifconfig\fR command must
44 be used at boot time to define the network address of each interface present on
45 a machine; it may also be used at a later time to redefine an interface's
46 address or other operating parameters. If no option is specified,
47 \fBifconfig\fR displays the current configuration for a network interface. If
48 an address family is specified, \fBifconfig\fR reports only the details
49 specific to that address family. Only privileged users may modify the
50 configuration of a network interface. Options appearing within braces
51 (\fB{\}\fR) indicate that one of the options must be specified.
52 .SS Network Interface Observability
53 .sp
54 .LP
55 Network interface observability with \fBifconfig\fR is limited to those
56 network interfaces that have been prepared for use with the IP
57 protocol suite. The preferred method for configuring a network
58 interface for use with TCP/IP is with \fBipadm\fR and alternatively
59 with the use of the \fBplumb\fR option as documented below. Network
60 interfaces that have not been configured for use with the IP
61 protocol suite can only be observed by using the \fBdladm\fR command.

```

```

62 .SS DHCP Configuration
63 .sp
64 .LP
65 The forms of \fBifconfig\fR that use the \fBauto-dhcp\fR or \fBdhcp\fR
66 arguments are used to control the Dynamic Host Configuration Protocol
67 ("fBDHCP") configuration of the interface. In this mode, \fBifconfig\fR is
68 used to control operation of \fBdhcpagent\fR(1M), the \fBDHCP\fR client daemon.
69 Once an interface is placed under \fBDHCP\fR control by using the \fBstart\fR
70 operand, \fBifconfig\fR should not, in normal operation, be used to modify the
71 address or characteristics of the interface. If the address of an interface
72 under \fBDHCP\fR is changed, \fBdhcpagent\fR will remove the interface from its
73 control.
74 .SH OPTIONS
75 .sp
76 .LP
77 When the \fBifconfig\fR command is executed without any options
78 its behavior is the same as when the \fB-a\fR option is supplied
79 with no other options or arguments.
80 .LP
81 The following options are supported:
82 .sp
83 .ne 2
84 .na
85 \fB\fBaddif\fR \fIaddress\fR\fR
86 .ad
87 .sp .6
88 .RS 4n
89 Create the next unused logical interface on the specified physical interface.
90 .RE

92 .sp
93 .ne 2
94 .na
95 \fB\fBall-zones\fR\fR
96 .ad
97 .sp .6
98 .RS 4n
99 Make the interface available to every shared-IP zone on the system. The
100 appropriate zone to which to deliver data is determined using the
101 \fBtzonecfg\fR database. This option is available only if the system is
102 configured with the Solaris Trusted Extensions feature.
103 .sp
104 The \fBtzonecfg\fR database is described in the \fBtzonecfg(4)\fR man page,
105 which is part of the \fISolaris Trusted Extensions Reference Manual\fR.
106 .RE

108 .sp
109 .ne 2
110 .na
111 \fB\fBanycast\fR\fR
112 .ad
113 .sp .6
114 .RS 4n
115 Marks the logical interface as an anycast address by setting the \fBANICAST\fR
116 flag. See "INTERFACE FLAGS," below, for more information on anycast.
117 .RE

119 .sp
120 .ne 2
121 .na
122 \fB\fB-anycast\fR\fR
123 .ad
124 .sp .6
125 .RS 4n
126 Marks the logical interface as not an anycast address by clearing the
127 \fBANICAST\fR flag.

```

```

128 .RE
130 .sp
131 .ne 2
132 .na
133 \fB\fBarp\fR\fR
134 .ad
135 .sp .6
136 .RS 4n
137 Enable the use of the Address Resolution Protocol ("

```

```

194 an interface as the primary one will not have much significance once the client
195 work station has booted, as many applications will already have started and
196 been configured with data read from the previous primary interface.
197 .RE
199 .sp
200 .ne 2
201 .na
202 \fB\fBwait \fIseconds\fR\fR
203 .ad
204 .sp .6
205 .RS 4n
206 The \fBifconfig\fR command will wait until the operation either completes or
207 for the interval specified, whichever is the sooner. If no wait interval is
208 given, and the operation is one that cannot complete immediately,
209 \fBifconfig\fR will wait 30 seconds for the requested operation to complete.
210 The symbolic value \fBforever\fR may be used as well, with obvious meaning.
211 .RE
213 .sp
214 .ne 2
215 .na
216 \fB\fBdrop\fR\fR
217 .ad
218 .sp .6
219 .RS 4n
220 Remove the specified interface from \fBDHCP\fR control without notifying the
221 DHCP server, and record the current lease for later use. Additionally, for
222 IPv4, set the IP address to zero. For IPv6, unplumb all logical interfaces
223 plumbed by \fBdhcpageant\fR.
224 .RE
226 .sp
227 .ne 2
228 .na
229 \fB\fBextend\fR\fR
230 .ad
231 .sp .6
232 .RS 4n
233 Attempt to extend the lease on the interface's IP address. This is not
234 required, as the agent will automatically extend the lease well before it
235 expires.
236 .RE
238 .sp
239 .ne 2
240 .na
241 \fB\fBinform\fR\fR
242 .ad
243 .sp .6
244 .RS 4n
245 Obtain network configuration parameters from \fBDHCP\fR without obtaining a
246 lease on \fBIP\fR addresses. This is useful in situations where an \fBIP\fR
247 address is obtained through mechanisms other than \fBDHCP\fR.
248 .RE
250 .sp
251 .ne 2
252 .na
253 \fB\fBping\fR\fR
254 .ad
255 .sp .6
256 .RS 4n
257 Check whether the interface given is under \fBDHCP\fR control, which means that
258 the interface is managed by the \fBDHCP\fR agent and is working properly. An
259 exit status of \fB0\fR means success.

```

```

260 .RE
262 .sp
263 .ne 2
264 .na
265 \fBrelease\fR
266 .ad
267 .sp .6
268 .RS 4n
269 Relinquish the IP addresses on the interface by notifying the server and
270 discard the current lease. For IPv4, set the IP address to zero. For IPv6, all
271 logical interfaces plumbed by \fBdhcpageant\fR are unplumbed.
272 .RE
274 .sp
275 .ne 2
276 .na
277 \fBstart\fR
278 .ad
279 .sp .6
280 .RS 4n
281 Start \fBDHCP\fR on the interface.
282 .RE
284 .sp
285 .ne 2
286 .na
287 \fBstatus\fR
288 .ad
289 .sp .6
290 .RS 4n
291 Display the \fBDHCP\fR configuration status of the interface.
292 .RE
294 .RE
296 .sp
297 .ne 2
298 .na
299 \fB\bauto-revarp\fR\fR
300 .ad
301 .sp .6
302 .RS 4n
303 Use the Reverse Address Resolution Protocol (RARP) to automatically acquire an
304 address for this interface. This will fail if the interface does not support
305 RARP; for example, IPoIB (IP over InfiniBand), and on IPv6 interfaces.
306 .RE
308 .sp
309 .ne 2
310 .na
311 \fB\broadcast\fR \fIaddress\fR\fR
312 .ad
313 .sp .6
314 .RS 4n
315 For IPv4 only. Specify the address to use to represent broadcasts to the
316 network. The default broadcast address is the address with a host part of all
317 \fB1\fR's. A "\fB+\fR" (plus sign) given for the broadcast value causes the
318 broadcast address to be reset to a default appropriate for the (possibly new)
319 address and netmask. The arguments of \fBifconfig\fR are interpreted left to
320 right. Therefore
321 .sp
322 .in +2
323 .nf
324 example% ifconfig -a netmask + broadcast +
325 .fi

```

```

326 .in -2
327 .sp
329 and
330 .sp
331 .in +2
332 .nf
333 example% ifconfig -a broadcast + netmask +
334 .fi
335 .in -2
336 .sp
338 may result in different values being assigned for the broadcast addresses of
339 the interfaces.
340 .RE
342 .sp
343 .ne 2
344 .na
345 \fB\bdeprecated\fR\fR
346 .ad
347 .sp .6
348 .RS 4n
349 Marks the logical interface as deprecated. An address associated with a
350 deprecated interface will not be used as source address for outbound packets
351 unless either there are no other addresses available on the interface or the
352 application has bound to this address explicitly. The status display shows
353 \fB\bDEPRECATED\fR as part of flags. See for information on the flags supported
354 by \fBifconfig\fR.
355 .RE
357 .sp
358 .ne 2
359 .na
360 \fB\b-deprecated\fR\fR
361 .ad
362 .sp .6
363 .RS 4n
364 Marks a logical interface as not deprecated. An address associated with such an
365 interface could be used as a source address for outbound packets.
366 .RE
368 .sp
369 .ne 2
370 .na
371 \fB\bpreferred\fR\fR
372 .ad
373 .sp .6
374 .RS 4n
375 Marks the logical interface as preferred. This option is only valid for IPv6
376 addresses. Addresses assigned to preferred logical interfaces are preferred as
377 source addresses over all other addresses configured on the system, unless the
378 address is of an inappropriate scope relative to the destination address.
379 Preferred addresses are used as source addresses regardless of which physical
380 interface they are assigned to. For example, you can configure a preferred
381 source address on the loopback interface and advertise reachability of this
382 address by using a routing protocol.
383 .RE
385 .sp
386 .ne 2
387 .na
388 \fB\b-preferred\fR\fR
389 .ad
390 .sp .6
391 .RS 4n

```

```

392 Marks the logical interface as not preferred.
393 .RE

395 .sp
396 .ne 2
397 .na
398 \fB\fBdestination\fR \fIdest_address\fR\fR
399 .ad
400 .sp .6
401 .RS 4n
402 Set the destination address for a point-to point interface.
403 .RE

405 .sp
406 .ne 2
407 .na
408 \fB\fBdhcp\fR\fR
409 .ad
410 .sp .6
411 .RS 4n
412 This option is an alias for option \fBauto-dhcp\fR
413 .RE

415 .sp
416 .ne 2
417 .na
418 \fB\fBdown\fR\fR
419 .ad
420 .sp .6
421 .RS 4n
422 Mark a logical interface as "down". (That is, turn off the \fBIFUP\fR bit.)
423 When a logical interface is marked "down," the system does not attempt to use
424 the address assigned to that interface as a source address for outbound packets
425 and will not recognize inbound packets destined to that address as being
426 addressed to this host. Additionally, when all logical interfaces on a given
427 physical interface are "down," the physical interface itself is disabled.
428 .sp
429 When a logical interface is down, all routes that specify that interface as the
430 output (using the \fB-ifp\fR option in the \fBroute\fR(1M) command or
431 \fBRTA_IFP\fR in a \fBroute\fR(7P) socket) are removed from the forwarding
432 table. Routes marked with \fBRTF_STATIC\fR are returned to the table if the
433 interface is brought back up, while routes not marked with \fBRTF_STATIC\fR are
434 simply deleted.
435 .sp
436 When all logical interfaces that could possibly be used to reach a particular
437 gateway address are brought down (specified without the interface option as in
438 the previous paragraph), the affected gateway routes are treated as though they
439 had the \fBRTF_BLACKHOLE\fR flag set. All matching packets are discarded
440 because the gateway is unreachable.
441 .RE

443 .sp
444 .ne 2
445 .na
446 \fB\fBencaplimit\fR \fIn\fR\fR
447 .ad
448 .sp .6
449 .RS 4n
450 Set the tunnel encapsulation limit for the interface to n. This option applies
451 to IPv4-in-IPv6 and IPv6-in-IPv6 tunnels only, and it simply modifies the
452 \fBencaplimit\fR link property of the underlying IPv6 tunnel link (see
453 \fBdladm\fR(1M)). The tunnel encapsulation limit controls how many more tunnels
454 a packet can enter before it leaves any tunnel, that is, the tunnel nesting
455 level.
456 .sp
457 This option is obsolete, superseded by the \fBdladm\fR(1M) \fBencaplimit\fR

```

```

458 link property.
459 .RE

461 .sp
462 .ne 2
463 .na
464 \fB\fBencaplimit\fR\fR
465 .ad
466 .sp .6
467 .RS 4n
468 Disable generation of the tunnel encapsulation limit. This option applies only
469 to IPv4-in-IPv6 and IPv6-in-IPv6 tunnels. This simply sets the \fBencaplimit\fR
470 link property of the underlying IPv6 tunnel link to 0 (see \fBdladm\fR(1M)
471 \fBencaplimit\fR).
472 .sp
473 This option is obsolete, superseded by the \fBdladm\fR(1M) \fBencaplimit\fR
474 link property.
475 .RE

477 .sp
478 .ne 2
479 .na
480 \fB\fBencr_auth_algs\fR \fIauthentication_algorithm\fR\fR
481 .ad
482 .sp .6
483 .RS 4n
484 For a tunnel, enable IPsec \fBESP\fR with the authentication algorithm
485 specified. It can be either a number or an algorithm name, including \fBany\fR
486 or \fBnone\fR, to indicate no algorithm preference. If an \fBESP\fR encryption
487 algorithm is specified but the authentication algorithm is not, the default
488 value for the \fBESP\fR authentication algorithm will be \fBany\fR.
489 .sp
490 It is now preferable to use the \fBbipsecconf\fR(1M) command when configuring a
491 tunnel's security properties. If \fBbipsecconf\fR was used to set a tunnel's
492 security properties, this keyword will not affect the tunnel.
493 .RE

495 .sp
496 .ne 2
497 .na
498 \fB\fBencr_algs\fR \fIencryption_algorithm\fR\fR
499 .ad
500 .sp .6
501 .RS 4n
502 For a tunnel, enable IPsec \fBESP\fR with the encryption algorithm specified.
503 It can be either a number or an algorithm name. Note that all IPsec tunnel
504 properties must be specified on the same command line. To disable tunnel
505 security, specify the value of \fBencr_alg\fR as \fBnone\fR. If an \fBESP\fR
506 authentication algorithm is specified, but the encryption algorithm is not, the
507 default value for the \fBESP\fR encryption will be \fBnull\fR.
508 .sp
509 It is now preferable to use the \fBbipsecconf\fR(1M) command when configuring a
510 tunnel's security properties. If \fBbipsecconf\fR was used to set a tunnel's
511 security properties, this keyword will not affect the tunnel.
512 .RE

514 .sp
515 .ne 2
516 .na
517 \fB\fBether\fR [ \fIaddress\fR ]\fR
518 .ad
519 .sp .6
520 .RS 4n
521 If no address is given and the user is root or has sufficient privileges to
522 open the underlying datalink, then display the current Ethernet address
523 information.

```

```

524 .sp
525 Otherwise, if the user is root or has sufficient privileges, set the Ethernet
526 address of the interfaces to \fiaddress\fR. The address is an Ethernet address
527 represented as \fIx:x:x:x:x\fR where \fIx\fR is a hexadecimal number between
528 0 and FF. Similarly, for the IPoIB (IP over InfiniBand) interfaces, the address
529 will be 20 bytes of colon-separated hex numbers between \fB0\fR and \fBFF\fR.
530 .sp
531 Some, though not all, Ethernet interface cards have their own addresses. To use
532 cards that do not have their own addresses, refer to section 3.2.3(4) of the
533 IEEE 802.3 specification for a definition of the locally administered address
534 space. Note that all IP interfaces in an IPMP group must have unique hardware
535 addresses; see \fBin.mpathd\fR(1M).
536 .RE

538 .sp
539 .ne 2
540 .na
541 \fB\fB-failover\fR\fR
542 .ad
543 .sp .6
544 .RS 4n
545 Set \fBNOFAILOVER\fR on the logical interface. This makes the associated
546 address available for use by \fBin.mpathd\fR to perform probe-based failure
547 detection for the associated physical IP interface. As a side effect,
548 \fBDEPRECATED\fR will also be set on the logical interface. This operation is
549 not permitted on an IPMP IP interface.
550 .RE

552 .sp
553 .ne 2
554 .na
555 \fB\fBfailover\fR\fR
556 .ad
557 .sp .6
558 .RS 4n
559 Clear \fBNOFAILOVER\fR on the logical interface. This is the default. These
560 logical interfaces are subject to migration when brought up (see \fBIP
561 MULTIPATHING GROUPS\fR).
562 .RE

564 .sp
565 .ne 2
566 .na
567 \fB\fBgroup\fR [ \fIname\fR | \fB"\fR]\fR
568 .ad
569 .sp .6
570 .RS 4n
571 When applied to a physical interface, it places the interface into the named
572 group. If the group does not exist, it will be created, along with one or more
573 IPMP IP interfaces (for IPv4, IPv6, or both). Any \fBUP\fR addresses that are
574 not also marked \fBNOFAILOVER\fR are subject to migration to the IPMP IP
575 interface (see \fBIP MULTIPATHING GROUPS\fR). Specifying a group name of
576 \fB"\fR removes the physical IP interface from the group.
577 .sp
578 When applied to a physical IPMP IP interface, it renames the IPMP group to have
579 the new name. If the name already exists, or a name of \fB"\fR is specified,
580 it fails. Renaming IPMP groups is discouraged. Instead, the IPMP IP interface
581 should be given a meaningful name when it is created by means of the \fBipmp\fR
582 subcommand, which the system will also use as the IPMP group name.
583 .RE

585 .sp
586 .ne 2
587 .na
588 \fB\fBindex\fR \fIn\fR\fR
589 .ad

```

```

590 .sp .6
591 .RS 4n
592 Change the interface index for the interface. The value of \fIn\fR must be an
593 interface index (\fIif_index\fR) that is not used on another interface.
594 \fIif_index\fR will be a non-zero positive number that uniquely identifies the
595 network interface on the system.
596 .RE

598 .sp
599 .ne 2
600 .na
601 \fB\fBipmp\fR\fR
602 .ad
603 .sp .6
604 .RS 4n
605 Create an IPMP IP interface with the specified name. An interface must be
606 separately created for use by IPv4 and IPv6. The \fIaddress_family\fR parameter
607 controls whether the command applies to IPv4 or IPv6 (IPv4 if unspecified). All
608 IPMP IP interfaces have the \fBIPMP\fR flag set.
609 .RE

611 .sp
612 .ne 2
613 .na
614 \fB\fBmetric\fR \fIn\fR\fR
615 .ad
616 .sp .6
617 .RS 4n
618 Set the routing metric of the interface to \fIn\fR; if no value is specified,
619 the default is \fB0\fR. The routing metric is used by the routing protocol.
620 Higher metrics have the effect of making a route less favorable. Metrics are
621 counted as addition hops to the destination network or host.
622 .RE

624 .sp
625 .ne 2
626 .na
627 \fB\fBmodinsert\fR \fImod_name@pos\fR\fR
628 .ad
629 .sp .6
630 .RS 4n
631 Insert a module with name \fImod_name\fR to the stream of the device at
632 position \fIpos\fR. The position is relative to the stream head. Position
633 \fB0\fR means directly under stream head.
634 .sp
635 Based upon the example in the \fBmodlist\fR option, use the following command
636 to insert a module with name \fBipqos\fR under the \fBip\fR module and above
637 the firewall module:
638 .sp
639 .in +2
640 .nf
641 example% ifconfig eri0 modinsert ipqos@2
642 .fi
643 .in -2
644 .sp

646 A subsequent listing of all the modules in the stream of the device follows:
647 .sp
648 .in +2
649 .nf
650 example% ifconfig eri0 modlist
651 0 arp
652 1 ip
653 2 ipqos
654 3 firewall
655 4 eri

```

```
656 .fi
657 .in -2
658 .sp

660 .RE

662 .sp
663 .ne 2
664 .na
665 \fB\fBmodlist\fR\fR
666 .ad
667 .sp .6
668 .RS 4n
669 List all the modules in the stream of the device.
670 .sp
671 The following example lists all the modules in the stream of the device:
672 .sp
673 .in +2
674 .nf
675 example% ifconfig eri0 modlist
676 0 arp
677 1 ip
678 2 firewall
679 4 eri
680 .fi
681 .in -2
682 .sp

684 .RE

686 .sp
687 .ne 2
688 .na
689 \fB\fBmodremove\fR\fR \fImod_name@pos\fR\fR
690 .ad
691 .sp .6
692 .RS 4n
693 Remove a module with name \fImod_name\fR from the stream of the device at
694 position \fIpos\fR. The position is relative to the stream head.
695 .sp
696 Based upon the example in the \fBmodinsert\fR option, use the following command
697 to remove the firewall module from the stream after inserting the \fBipqos\fR
698 module:
699 .sp
700 .in +2
701 .nf
702 example% ifconfig eri0 modremove firewall@3
703 .fi
704 .in -2
705 .sp

707 A subsequent listing of all the modules in the stream of the device follows:
708 .sp
709 .in +2
710 .nf
711 example% ifconfig eri0 modlist
712 0 arp
713 1 ip
714 2 ipqos
715 3 eri
716 .fi
717 .in -2
718 .sp

720 Note that the core IP stack modules, for example, \fBip\fR and \fBtun\fR
721 modules, cannot be removed.
```

```
722 .RE

724 .sp
725 .ne 2
726 .na
727 \fB\fBmtu\fR\fR \fIn\fR\fR
728 .ad
729 .sp .6
730 .RS 4n
731 Set the maximum transmission unit of the interface to \fIn\fR. For many types
732 of networks, the \fBmtu\fR has an upper limit, for example, \fB1500\fR for
733 Ethernet. This option sets the \fBFIXEDMTU\fR flag on the affected interface.
734 .RE

736 .sp
737 .ne 2
738 .na
739 \fB\fBnetmask\fR\fR \fImask\fR\fR
740 .ad
741 .sp .6
742 .RS 4n
743 For IPv4 only. Specify how much of the address to reserve for subdividing
744 networks into subnetworks. The mask includes the network part of the local
745 address and the subnet part, which is taken from the host field of the address.
746 The mask contains 1's for the bit positions in the 32-bit address which are to
747 be used for the network and subnet parts, and 0's for the host part. The mask
748 should contain at least the standard network portion, and the subnet field
749 should be contiguous with the network portion. The mask can be specified in one
750 of four ways:
751 .RS +4
752 .TP
753 1.
754 with a single hexadecimal number with a leading 0x,
755 .RE
756 .RS +4
757 .TP
758 2.
759 with a dot-notation address,
760 .RE
761 .RS +4
762 .TP
763 3.
764 with a "\fB+\fR" (plus sign) address, or
765 .RE
766 .RS +4
767 .TP
768 4.
769 with a pseudo host name/pseudo network name found in the network database
770 \fBnetworks\fR(4).
771 .RE
772 If a "\fB+\fR" (plus sign) is given for the netmask value, the mask is looked
773 up in the \fBnetmasks\fR(4) database. This lookup finds the longest matching
774 netmask in the database by starting with the interface's IPv4 address as the
775 key and iteratively masking off more and more low order bits of the address.
776 This iterative lookup ensures that the \fBnetmasks\fR(4) database can be used
777 to specify the netmasks when variable length subnetmasks are used within a
778 network number.
779 .sp
780 If a pseudo host name/pseudo network name is supplied as the netmask value,
781 netmask data may be located in the \fBhosts\fR or \fBnetworks\fR database.
782 Names are looked up by first using \fBgethostbyname\fR(3NSL). If not found
783 there, the names are looked up in \fBgetnetbyname\fR(3SOCKET). These interfaces
784 may in turn use \fBnsswitch.conf\fR(4) to determine what data store(s) to use
785 to fetch the actual value.
786 .sp
787 For both \fBinet\fR and \fBinet6\fR, the same information conveyed by
```

```

788 \fImask\fR can be specified as a \fIprefix_length\fR attached to the
789 \fIaddress\fR parameter.
790 .RE

792 .sp
793 .ne 2
794 .na
795 \fB\fBnud\fR\fR
796 .ad
797 .sp .6
798 .RS 4n
799 Enables the neighbor unreachability detection mechanism on a point-to-point
800 physical interface.
801 .RE

803 .sp
804 .ne 2
805 .na
806 \fB\fBnud\fR\fR
807 .ad
808 .sp .6
809 .RS 4n
810 Disables the neighbor unreachability detection mechanism on a point-to-point
811 physical interface.
812 .RE

814 .sp
815 .ne 2
816 .na
817 \fB\fBplumb\fR\fR
818 .ad
819 .sp .6
820 .RS 4n
821 For a physical IP interface, open the datalink associated with the physical
822 interface name and set up the plumbing needed for IP to use the datalink. When
823 used with a logical interface name, this command is used to create a specific
824 named logical interface on an existing physical IP interface.
825 .sp
826 An interface must be separately plumbed for IPv4 and IPv6 according to the
827 \fIaddress_family\fR parameter (IPv4 if unspecified). Before an interface has
828 been plumbed, it will not be shown by \fBifconfig\fR \fB-a\fR.
829 .sp
830 Note that IPMP IP interfaces are not tied to a specific datalink and are
831 instead created with the \fBipmp\fR subcommand.
832 .RE

834 .sp
835 .ne 2
836 .na
837 \fB\fBprivate\fR\fR
838 .ad
839 .sp .6
840 .RS 4n
841 Tells the \fBin.routed\fR routing daemon that a specified logical interface
842 should not be advertised.
843 .RE

845 .sp
846 .ne 2
847 .na
848 \fB\fBprivate\fR\fR
849 .ad
850 .sp .6
851 .RS 4n
852 Specify unadvertised interfaces.
853 .RE

```

```

855 .sp
856 .ne 2
857 .na
858 \fB\fBremoveif\fR \fIaddress\fR\fR
859 .ad
860 .sp .6
861 .RS 4n
862 Remove the logical interface on the physical interface specified that matches
863 the \fIaddress\fR specified.
864 .RE

866 .sp
867 .ne 2
868 .na
869 \fB\fBrouter\fR\fR
870 .ad
871 .sp .6
872 .RS 4n
873 Enable IP forwarding on the interface. When enabled, the interface is marked
874 \fBROUTER\fR, and IP packets can be forwarded to and from the interface.
875 Enabling \fBROUTER\fR on any IP interface in an IPMP group enables it on all IP
876 interfaces in that IPMP group.
877 .RE

879 .sp
880 .ne 2
881 .na
882 \fB\fB-router\fR\fR
883 .ad
884 .sp .6
885 .RS 4n
886 Disable IP forwarding on the interface. IP packets are not forwarded to and
887 from the interface. Disabling \fBROUTER\fR on any IP interface in an IPMP group
888 disables it on all IP interfaces in that IPMP group.
889 .RE

891 .sp
892 .ne 2
893 .na
894 \fB\fBset\fR\fR
895 .ad
896 .sp .6
897 .RS 4n
898 Set the \fIaddress\fR, \fIprefix_length\fR or both, for a logical interface.
899 .RE

901 .sp
902 .ne 2
903 .na
904 \fB\fBstandby\fR\fR
905 .ad
906 .sp .6
907 .RS 4n
908 Mark the physical IP interface as a \fBSTANDBY\fR interface. If an interface is
909 marked \fBSTANDBY\fR and is part of an IPMP group, the interface will not be
910 used for data traffic unless another interface in the IPMP group becomes
911 unusable. When a \fBSTANDBY\fR interface is functional but not being used for
912 data traffic, it will also be marked \fBINACTIVE\fR. This operation is not
913 permitted on an IPMP IP interface.
914 .RE

916 .sp
917 .ne 2
918 .na
919 \fB\fB-standby\fR\fR

```



```

920 .ad
921 .sp .6
922 .RS 4n
923 Clear \fBSTANDBY\fR on the interface. This is the default.
924 .RE

926 .sp
927 .ne 2
928 .na
929 \fB\fBsubnet\fR\fR
930 .ad
931 .sp .6
932 .RS 4n
933 Set the subnet \fIaddress\fR for an interface.
934 .RE

936 .sp
937 .ne 2
938 .na
939 \fB\fBtdest\fR \fItunnel_dest_address\fR\fR
940 .ad
941 .sp .6
942 .RS 4n
943 Set the destination address of a tunnel. The address should not be the same as
944 the \fBdest_address\fR of the tunnel, because no packets leave the system over
945 such a tunnel.
946 .sp
947 This option is obsolete, superseded by the \fBdldm\fR(1M) \fBcreate-iptun\fR
948 and \fBmodify-iptun\fR subcommands.
949 .RE

951 .sp
952 .ne 2
953 .na
954 \fB\fBthoplimit\fR \fIn\fR\fR
955 .ad
956 .sp .6
957 .RS 4n
958 Set the hop limit for a tunnel interface. The hop limit value is used as the
959 \fBTTTL\fR in the IPv4 header for the IPv6-in-IPv4 and IPv4-in-IPv4 tunnels. For
960 IPv6-in-IPv6 and IPv4-in-IPv6 tunnels, the hop limit value is used as the hop
961 limit in the IPv6 header. This option simply modifies the \fBhoplimit\fR link
962 property of the underlying IP tunnel link (see \fBdldm\fR(1M)).
963 .sp
964 This option is obsolete, superseded by the \fBdldm\fR(1M) \fBhoplimit\fR link
965 property.
966 .RE

968 .sp
969 .ne 2
970 .na
971 \fB\fBtoken\fR \fIaddress\fR/\fIprefix_length\fR\fR
972 .ad
973 .sp .6
974 .RS 4n
975 Set the IPv6 token of an interface to be used for address autoconfiguration.
976 .sp
977 .in +2
978 .nf
979 example% \fBifconfig eri0 inet6 token ::1/64\fR
980 .fi
981 .in -2
982 .sp

984 .RE

```

```

986 .sp
987 .ne 2
988 .na
989 \fB\fBtrailers\fR\fR
990 .ad
991 .sp .6
992 .RS 4n
993 This flag previously caused a nonstandard encapsulation of IPv4 packets on
994 certain link levels. Drivers supplied with this release no longer use this
995 flag. It is provided for compatibility, but is ignored.
996 .RE

998 .sp
999 .ne 2
1000 .na
1001 \fB\fB-trailers\fR\fR
1002 .ad
1003 .sp .6
1004 .RS 4n
1005 Disable the use of a "trailer" link level encapsulation.
1006 .RE

1008 .sp
1009 .ne 2
1010 .na
1011 \fB\fBtsrc\fR \fItunnel_src_address\fR\fR
1012 .ad
1013 .sp .6
1014 .RS 4n
1015 Set the source address of a tunnel. This is the source address on an outer
1016 encapsulating \fBIP\fR header. It must be an address of another interface
1017 already configured using \fBifconfig\fR.
1018 .sp
1019 This option is obsolete, superseded by the \fBdldm\fR(1M) \fBcreate-iptun\fR
1020 and \fBmodify-iptun\fR subcommands.
1021 .RE

1023 .sp
1024 .ne 2
1025 .na
1026 \fB\fBunplumb\fR\fR
1027 .ad
1028 .sp .6
1029 .RS 4n
1030 For a physical or IPMP interface, remove all associated logical IP interfaces
1031 and tear down any plumbing needed for IP to use the interface. For an IPMP IP
1032 interface, this command will fail if the group is not empty. For a logical
1033 interface, the logical interface is removed.
1034 .sp
1035 An interface must be separately unplumbed for IPv4 and IPv6 according to the
1036 \fIaddress_family\fR parameter (IPv4 if unspecified). Upon success, the
1037 interface name will no longer appear in the output of \fBifconfig\fR \fB-a\fR.
1038 .RE

1040 .sp
1041 .ne 2
1042 .na
1043 \fB\fBup\fR\fR
1044 .ad
1045 .sp .6
1046 .RS 4n
1047 Mark a logical interface \fBUP\fR. As a result, the IP module will accept
1048 packets destined to the associated address (unless the address is zero), along
1049 with any associated multicast and broadcast IP addresses. Similarly, the IP
1050 module will allow packets to be sent with the associated address as a source
1051 address. At least one logical interface must be \fBUP\fR for the associated

```

```

1052 physical interface to send or receive packets
1053 .RE

1055 .sp
1056 .ne 2
1057 .na
1058 \fB\fBusesrc\fR [ \fIname\fR | \fBnone\fR ]\fR
1059 .ad
1060 .sp .6
1061 .RS 4n
1062 Specify a physical interface to be used for source address selection. If the
1063 keyword \fBnone\fR is used, then any previous selection is cleared.
1064 .sp
1065 When an application does not choose a non-zero source address using
1066 \fBbind\fR(3SOCKET), the system will select an appropriate source address based
1067 on the outbound interface and the address selection rules (see
1068 \fBipaddrsel\fR(1M)).
1069 .sp
1070 When \fBusesrc\fR is specified and the specified interface is selected in the
1071 forwarding table for output, the system looks first to the specified physical
1072 interface and its associated logical interfaces when selecting a source
1073 address. If no usable address is listed in the forwarding table, the ordinary
1074 selection rules apply. For example, if you enter:
1075 .sp
1076 .in +2
1077 .nf
1078 # \fBifconfig eri0 usesrc vni0\fR
1079 .fi
1080 .in -2
1081 .sp

1083 \&...and \fBvni0\fR has address 10.0.0.1 assigned to it, the system will prefer
1084 10.0.0.1 as the source address for any packets originated by local connections
1085 that are sent through \fBerio0\fR. Further examples are provided in the
1086 \fBEXAMPLES\fR section.
1087 .sp
1088 While you can specify any physical interface (or even loopback), be aware that
1089 you can also specify the virtual IP interface (see \fBvni\fR(7D)). The virtual
1090 IP interface is not associated with any physical hardware and is thus immune to
1091 hardware failures. You can specify any number of physical interfaces to use the
1092 source address hosted on a single virtual interface. This simplifies the
1093 configuration of routing-based multipathing. If one of the physical interfaces
1094 were to fail, communication would continue through one of the remaining,
1095 functioning physical interfaces. This scenario assumes that the reachability of
1096 the address hosted on the virtual interface is advertised in some manner, for
1097 example, through a routing protocol.
1098 .sp
1099 Because the \fBifconfig\fR \fBpreferred\fR option is applied to all interfaces,
1100 it is coarser-grained than the \fBusesrc\fR option. It will be overridden by
1101 \fBusesrc\fR and \fBsetsrc\fR (route subcommand), in that order.
1102 .sp
1103 IPMP and the \fBusesrc\fR option are mutually exclusive. That is, if an
1104 interface is part of an IPMP group or marked \fBSTANDBY\fR, then it cannot be
1105 specified by means of \fBusesrc\fR, and vice-versa.
1106 .RE

1108 .sp
1109 .ne 2
1110 .na
1111 \fB\fBxmit\fR\fR
1112 .ad
1113 .sp .6
1114 .RS 4n
1115 Enable a logical interface to transmit packets. This is the default behavior
1116 when the logical interface is up.
1117 .RE

```

```

1119 .sp
1120 .ne 2
1121 .na
1122 \fB\fB-xmit\fR\fR
1123 .ad
1124 .sp .6
1125 .RS 4n
1126 Disable transmission of packets on an interface. The interface will continue to
1127 receive packets.
1128 .RE

1130 .sp
1131 .ne 2
1132 .na
1133 \fB\fBzone\fR \fIzone\fR\fR
1134 .ad
1135 .sp .6
1136 .RS 4n
1137 Place the logical interface in zone \fIzone\fR. The named zone must be
1138 active in the kernel in the ready or running state. The interface is unplumbed
1139 when the zone is halted or rebooted. The zone must be configured to be an
1140 shared-IP zone. \fBzonecfg\fR(1M) is used to assign network interface names to
1141 exclusive-IP zones.
1142 .RE

1144 .sp
1145 .ne 2
1146 .na
1147 \fB\fB-zone\fR\fR
1148 .ad
1149 .sp .6
1150 .RS 4n
1151 Place IP interface in the global zone. This is the default.
1152 .RE

1154 .SH OPERANDS
1155 .sp
1156 .LP
1157 The \fIinterface\fR operand, as well as address parameters that affect it, are
1158 described below.
1159 .sp
1160 .ne 2
1161 .na
1162 \fB\fIinterface\fR\fR
1163 .ad
1164 .sp .6
1165 .RS 4n
1166 A string of one of the following forms:
1167 .RS +4
1168 .TP
1169 .ie t \(\bu
1170 .el o
1171 \fIname physical-unit\fR, for example, \fBerio0\fR or \fBcel0\fR
1172 .RE
1173 .RS +4
1174 .TP
1175 .ie t \(\bu
1176 .el o
1177 \fIname physical-unit\fR:\fB:\fR\fIlogical-unit\fR, for example, \fBerio:1\fR
1178 .RE
1179 .RS +4
1180 .TP
1181 .ie t \(\bu
1182 .el o
1183 \fBip.tun\fR\fIN\fR, \fBip6.tun\fR\fIN\fR, or \fBip6to4.tun\fR\fIN\fR for

```

```

1184 implicit IP tunnel links
1185 .RE
1186 If the interface name starts with a dash (-), it is interpreted as a set of
1187 options which specify a set of interfaces. In such a case, \fB-a\fR must be
1188 part of the options and any of the additional options below can be added in any
1189 order. If one of these interface names is given, the commands following it are
1190 applied to all of the interfaces that match.
1191 .sp
1192 .ne 2
1193 .na
1194 \fB\fB-a\fR\fR
1195 .ad
1196 .sp .6
1197 .RS 4n
1198 Apply the command to all interfaces of the specified address family. If no
1199 address family is supplied, either on the command line or by means of
1200 \fB/etc/default/inet_type\fR, then all address families will be selected.
1201 .RE

1203 .sp
1204 .ne 2
1205 .na
1206 \fB\fB-d\fR\fR
1207 .ad
1208 .sp .6
1209 .RS 4n
1210 Apply the commands to all "down" interfaces in the system.
1211 .RE

1213 .sp
1214 .ne 2
1215 .na
1216 \fB\fB-D\fR\fR
1217 .ad
1218 .sp .6
1219 .RS 4n
1220 Apply the commands to all interfaces not under \fBDHCP\fR (Dynamic Host
1221 Configuration Protocol) control.
1222 .RE

1224 .sp
1225 .ne 2
1226 .na
1227 \fB\fB-u\fR\fR
1228 .ad
1229 .sp .6
1230 .RS 4n
1231 Apply the commands to all "up" interfaces in the system.
1232 .RE

1234 .sp
1235 .ne 2
1236 .na
1237 \fB\fB-Z\fR\fR
1238 .ad
1239 .sp .6
1240 .RS 4n
1241 Apply the commands to all interfaces in the user's zone.
1242 .RE

1244 .sp
1245 .ne 2
1246 .na
1247 \fB\fB-4\fR\fR
1248 .ad
1249 .sp .6

```

```

1250 .RS 4n
1251 Apply the commands to all IPv4 interfaces.
1252 .RE

1254 .sp
1255 .ne 2
1256 .na
1257 \fB\fB-6\fR\fR
1258 .ad
1259 .sp .6
1260 .RS 4n
1261 Apply the commands to all IPv6 interfaces.
1262 .RE

1264 .RE

1266 .sp
1267 .ne 2
1268 .na
1269 \fB\fIaddress_family\fR\fR
1270 .ad
1271 .sp .6
1272 .RS 4n
1273 The address family is specified by the \fIaddress_family\fR parameter. The
1274 \fBifconfig\fR command currently supports the following families: \fBinet\fR
1275 and \fBinet6\fR. If no address family is specified, the default is \fBinet\fR.
1276 .sp
1277 \fBifconfig\fR honors the \fBDEFAULT_IP\fR setting in the
1278 \fB/etc/default/inet_type\fR file when it displays interface information . If
1279 \fBDEFAULT_IP\fR is set to \fBIP_VERSION4\fR, then \fBifconfig\fR will omit
1280 information that relates to IPv6 interfaces. However, when you explicitly
1281 specify an address family (\fBinet\fR or \fBinet6\fR) on the \fBifconfig\fR
1282 command line, the command line overrides the \fBDEFAULT_IP\fR settings.
1283 .RE

1285 .sp
1286 .ne 2
1287 .na
1288 \fB\fIaddress\fR\fR
1289 .ad
1290 .sp .6
1291 .RS 4n
1292 For the IPv4 family (\fBinet\fR), the \fIaddress\fR is either a host name
1293 present in the host name data base (see \fBhosts\fR(4)) or in the Network
1294 Information Service (NIS) map \fBhosts\fR, or an IPv4 address expressed in the
1295 Internet standard "dot notation".
1296 .sp
1297 For the IPv6 family (\fBinet6\fR), the \fIaddress\fR is either a host name
1298 present in the host name data base (see \fBhosts\fR(4)) or in the Network
1299 Information Service (\fBNIS\fR) map \fBipnode\fR, or an IPv6 address expressed
1300 in the Internet standard colon-separated hexadecimal format represented as
1301 \fIix:x:x:x:x:x:x:x\fR where \fIix\fR is a hexadecimal number between \fB0\fR and
1302 \fBFFFF\fR.
1303 .RE

1305 .sp
1306 .ne 2
1307 .na
1308 \fB\fIprefix_length\fR\fR
1309 .ad
1310 .sp .6
1311 .RS 4n
1312 For the IPv4 and IPv6 families (\fBinet\fR and \fBinet6\fR), the
1313 \fIprefix_length\fR is a number between 0 and the number of bits in the
1314 address. For \fBinet\fR, the number of bits in the address is 32; for
1315 \fBinet6\fR, the number of bits in the address is 128. The \fIprefix_length\fR

```

1316 denotes the number of leading set bits in the netmask.
 1317 .RE

1319 .sp
 1320 .ne 2
 1321 .na
 1322 \fB\fIdest_address\fR\fR
 1323 .ad
 1324 .sp .6
 1325 .RS 4n
 1326 If the \fIdest_address\fR parameter is supplied in addition to the
 1327 \fIaddress\fR parameter, it specifies the address of the correspondent on the
 1328 other end of a point-to-point link.
 1329 .RE

1331 .sp
 1332 .ne 2
 1333 .na
 1334 \fB\fItunnel_dest_address\fR\fR
 1335 .ad
 1336 .sp .6
 1337 .RS 4n
 1338 An address that is or will be reachable through an interface other than the
 1339 tunnel being configured. This tells the tunnel where to send the tunneled
 1340 packets. This address must not be the same as the interface destination address
 1341 being configured.
 1342 .RE

1344 .sp
 1345 .ne 2
 1346 .na
 1347 \fB\fItunnel_src_address\fR\fR
 1348 .ad
 1349 .sp .6
 1350 .RS 4n
 1351 An address that is attached to an already configured interface that has been
 1352 configured "up" with \fBifconfig\fR.
 1353 .RE

1355 .SH INTERFACE FLAGS
 1356 .sp
 1357 .LP
 1358 The \fBifconfig\fR command supports the following interface flags. The term
 1359 "address" in this context refers to a logical interface, for example,
 1360 \fBer0:0\fR, while "interface" refers to the physical interface, for example,
 1361 \fBer0\fR.
 1362 .sp
 1363 .ne 2
 1364 .na
 1365 \fB\fBADDRCONF\fR\fR
 1366 .ad
 1367 .sp .6
 1368 .RS 4n
 1369 The address is from stateless \fBaddrconf\fR. The stateless mechanism allows a
 1370 host to generate its own address using a combination of information advertised
 1371 by routers and locally available information. Routers advertise prefixes that
 1372 identify the subnet associated with the link, while the host generates an
 1373 "interface identifier" that uniquely identifies an interface in a subnet. In
 1374 the absence of information from routers, a host can generate link-local
 1375 addresses. This flag is specific to IPv6.
 1376 .RE

1378 .sp
 1379 .ne 2
 1380 .na
 1381 \fB\fBANYCAST\fR\fR

1382 .ad
 1383 .sp .6
 1384 .RS 4n
 1385 Indicates an \fBanycast\fR address. An \fBanycast\fR address identifies the
 1386 nearest member of a group of systems that provides a particular type of
 1387 service. An \fBanycast\fR address is assigned to a group of systems. Packets
 1388 are delivered to the nearest group member identified by the \fBanycast\fR
 1389 address instead of being delivered to all members of the group.
 1390 .RE

1392 .sp
 1393 .ne 2
 1394 .na
 1395 \fB\fBBROADCAST\fR\fR
 1396 .ad
 1397 .sp .6
 1398 .RS 4n
 1399 This \fBbroadcast\fR address is valid. This flag and \fBPOINTTOPOINT\fR are
 1400 mutually exclusive
 1401 .RE

1403 .sp
 1404 .ne 2
 1405 .na
 1406 \fB\fBCoS\fR\fR
 1407 .ad
 1408 .sp .6
 1409 .RS 4n
 1410 This interface supports some form of Class of Service (CoS) marking. An example
 1411 is the 802.1D user priority marking supported on \fBVLAN\fR interfaces. For
 1412 IPMP IP interfaces, this will only be set if all interfaces in the group have
 1413 CoS set.
 1414 .sp
 1415 Note that this flag is only set on interfaces over VLAN links and over Ethernet
 1416 links that have their \fBdldm\fR(1M) \fBtagmode\fR link property set to
 1417 \fBnormal\fR.
 1418 .RE

1420 .sp
 1421 .ne 2
 1422 .na
 1423 \fB\fBDEPRECATED\fR\fR
 1424 .ad
 1425 .sp .6
 1426 .RS 4n
 1427 This address is deprecated. This address will not be used as a source address
 1428 for outbound packets unless there are no other addresses on this interface or
 1429 an application has explicitly bound to this address. An IPv6 deprecated address
 1430 is part of the standard mechanism for renumbering in IPv6 and will eventually
 1431 be deleted when not used. For both IPv4 and IPv6, \fBDEPRECATED\fR is also set
 1432 on all \fBNOFAILOVER\fR addresses, though this may change in a future release.
 1433 .RE

1435 .sp
 1436 .ne 2
 1437 .na
 1438 \fB\fBDHCPRUNNING\fR\fR
 1439 .ad
 1440 .sp .6
 1441 .RS 4n
 1442 The logical interface is managed by \fBdhcpagent\fR(1M).
 1443 .RE

1445 .sp
 1446 .ne 2
 1447 .na

```

1448 \fB\fBDUPLICATE\fR\fR
1449 .ad
1450 .sp .6
1451 .RS 4n
1452 The logical interface has been disabled because the IP address configured on
1453 the interface is a duplicate. Some other node on the network is using this
1454 address. If the address was configured by DHCP or is temporary, the system will
1455 choose another automatically, if possible. Otherwise, the system will attempt
1456 to recover this address periodically and the interface will recover when the
1457 conflict has been removed from the network. Changing the address or netmask, or
1458 setting the logical interface to \fBup\fR will restart duplicate detection.
1459 Setting the interface to \fBdown\fR terminates recovery and removes the
1460 \fBDUPLICATE\fR flag.
1461 .RE

1463 .sp
1464 .ne 2
1465 .na
1466 \fB\fBFAILED\fR\fR
1467 .ad
1468 .sp .6
1469 .RS 4n
1470 The \fBin.mpathd\fR daemon has determined that the interface has failed.
1471 \fBFAILED\fR interfaces will not be used to send or receive IP data traffic. If
1472 this is set on a physical IP interface in an IPMP group, IP data traffic will
1473 continue to flow over other usable IP interfaces in the IPMP group. If this is
1474 set on an IPMP IP interface, the entire group has failed and no data traffic
1475 can be sent or received over any interfaces in that group.
1476 .RE

1478 .sp
1479 .ne 2
1480 .na
1481 \fB\fBFIXEDMTU\fR\fR
1482 .ad
1483 .sp .6
1484 .RS 4n
1485 The MTU has been set using the \fB-mtu\fR option. This flag is read-only.
1486 Interfaces that have this flag set have a fixed MTU value that is unaffected by
1487 dynamic MTU changes that can occur when drivers notify IP of link MTU changes.
1488 .RE

1490 .sp
1491 .ne 2
1492 .na
1493 \fB\fBINACTIVE\fR\fR
1494 .ad
1495 .sp .6
1496 .RS 4n
1497 The physical interface is functioning but is not used to send or receive data
1498 traffic according to administrative policy. This flag is initially set by the
1499 \fBstandby\fR subcommand and is subsequently controlled by \fBin.mpathd\fR. It
1500 also set when \fBFAILBACK=no\fR mode is enabled (see \fBin.mpathd\fR(1M)) to
1501 indicate that the IP interface has repaired but is not being used.
1502 .RE

1504 .sp
1505 .ne 2
1506 .na
1507 \fB\fBIPMP\fR\fR
1508 .ad
1509 .sp .6
1510 .RS 4n
1511 Indicates that this is an IPMP IP interface.
1512 .RE

```

```

1514 .sp
1515 .ne 2
1516 .na
1517 \fB\fBLOOPBACK\fR\fR
1518 .ad
1519 .sp .6
1520 .RS 4n
1521 Indicates that this is the loopback interface.
1522 .RE

1524 .sp
1525 .ne 2
1526 .na
1527 \fB\fBMULTI_BCAST\fR\fR
1528 .ad
1529 .sp .6
1530 .RS 4n
1531 Indicates that the broadcast address is used for multicast on this interface.
1532 .RE

1534 .sp
1535 .ne 2
1536 .na
1537 \fB\fBMULTICAST\fR\fR
1538 .ad
1539 .sp .6
1540 .RS 4n
1541 The interface supports multicast. \fBIP\fR assumes that any interface that
1542 supports hardware broadcast, or that is a point-to-point link, will support
1543 multicast.
1544 .RE

1546 .sp
1547 .ne 2
1548 .na
1549 \fB\fBNOARP\fR\fR
1550 .ad
1551 .sp .6
1552 .RS 4n
1553 There is no address resolution protocol (\fBARP\fR) for this interface that
1554 corresponds to all interfaces for a device without a broadcast address. This
1555 flag is specific to IPv4.
1556 .RE

1558 .sp
1559 .ne 2
1560 .na
1561 \fB\fBNOFAILOVER\fR\fR
1562 .ad
1563 .sp .6
1564 .RS 4n
1565 The address associated with this logical interface is available to
1566 \fBin.mpathd\fR for probe-based failure detection of the associated physical IP
1567 interface.
1568 .RE

1570 .sp
1571 .ne 2
1572 .na
1573 \fB\fBNOLOCAL\fR\fR
1574 .ad
1575 .sp .6
1576 .RS 4n
1577 The interface has no address , just an on-link subnet.
1578 .RE

```

```

1580 .sp
1581 .ne 2
1582 .na
1583 \fb\fbNONUD\fr\fr
1584 .ad
1585 .sp .6
1586 .RS 4n
1587 \fbNUD\fr is disabled on this interface. \fbNUD\fr (neighbor unreachability
1588 detection) is used by a node to track the reachability state of its neighbors,
1589 to which the node actively sends packets, and to perform any recovery if a
1590 neighbor is detected to be unreachable. This flag is specific to IPv6.
1591 .RE

1593 .sp
1594 .ne 2
1595 .na
1596 \fb\fbNORTEXCH\fr\fr
1597 .ad
1598 .sp .6
1599 .RS 4n
1600 The interface does not exchange routing information. For RIP-2, routing packets
1601 are not sent over this interface. Additionally, messages that appear to come
1602 over this interface receive no response. The subnet or address of this
1603 interface is not included in advertisements over other interfaces to other
1604 routers.
1605 .RE

1607 .sp
1608 .ne 2
1609 .na
1610 \fb\fbNOXMIT\fr\fr
1611 .ad
1612 .sp .6
1613 .RS 4n
1614 Indicates that the address does not transmit packets. RIP-2 also does not
1615 advertise this address.
1616 .RE

1618 .sp
1619 .ne 2
1620 .na
1621 \fb\fbOFFLINE\fr\fr
1622 .ad
1623 .sp .6
1624 .RS 4n
1625 The interface is offline and thus cannot send or receive IP data traffic. This
1626 is only set on IP interfaces in an IPMP group. See \fb\fbmpadm\fr(1M) and
1627 \fb\fbcfgadm\fr(1M).
1628 .RE

1630 .sp
1631 .ne 2
1632 .na
1633 \fb\fbPOINTOPOINT\fr\fr
1634 .ad
1635 .sp .6
1636 .RS 4n
1637 Indicates that the address is a point-to-point link. This flag and
1638 \fb\fbBROADCAST\fr are mutually exclusive
1639 .RE

1641 .sp
1642 .ne 2
1643 .na
1644 \fb\fbPREFERRED\fr\fr
1645 .ad

```

```

1646 .sp .6
1647 .RS 4n
1648 This address is a preferred IPv6 source address. This address will be used as a
1649 source address for IPv6 communication with all IPv6 destinations, unless
1650 another address on the system is of more appropriate scope. The
1651 \fb\fbDEPRECATED\fr flag takes precedence over the \fb\fbPREFERRED\fr flag.
1652 .RE

1654 .sp
1655 .ne 2
1656 .na
1657 \fb\fbPRIVATE\fr\fr
1658 .ad
1659 .sp .6
1660 .RS 4n
1661 Indicates that this address is not advertised. For RIP-2, this interface is
1662 used to send advertisements. However, neither the subnet nor this address are
1663 included in advertisements to other routers.
1664 .RE

1666 .sp
1667 .ne 2
1668 .na
1669 \fb\fbPROMISC\fr\fr
1670 .ad
1671 .sp .6
1672 .RS 4n
1673 A read-only flag indicating that an interface is in promiscuous mode. All
1674 addresses associated with an interface in promiscuous mode will display (in
1675 response to \fb\fbifconfig\fr \fb-a\fr, for example) the \fb\fbPROMISC\fr flag.
1676 .RE

1678 .sp
1679 .ne 2
1680 .na
1681 \fb\fbROUTER\fr\fr
1682 .ad
1683 .sp .6
1684 .RS 4n
1685 Indicates that IP packets can be forwarded to and from the interface.
1686 .RE

1688 .sp
1689 .ne 2
1690 .na
1691 \fb\fbRUNNING\fr\fr
1692 .ad
1693 .sp .6
1694 .RS 4n
1695 Indicates that the required resources for an interface are allocated. For some
1696 interfaces this also indicates that the link is up. For IPMP IP interfaces,
1697 \fb\fbRUNNING\fr is set as long as one IP interface in the group is active.
1698 .RE

1700 .sp
1701 .ne 2
1702 .na
1703 \fb\fbSTANDBY\fr\fr
1704 .ad
1705 .sp .6
1706 .RS 4n
1707 Indicates that this physical interface will not be used for data traffic unless
1708 another interface in the IPMP group becomes unusable. The \fb\fbINACTIVE\fr and
1709 \fb\fbFAILED\fr flags indicate whether it is actively being used.
1710 .RE

```

```

1712 .sp
1713 .ne 2
1714 .na
1715 \fB\fbTEMPORARY\fR\fR
1716 .ad
1717 .sp .6
1718 .RS 4n
1719 Indicates that this is a temporary IPv6 address as defined in RFC 3041.
1720 .RE

1722 .sp
1723 .ne 2
1724 .na
1725 \fB\fbUNNUMBERED\fR\fR
1726 .ad
1727 .sp .6
1728 .RS 4n
1729 This flag is set when the local IP address on the link matches the local
1730 address of some other link in the system
1731 .RE

1733 .sp
1734 .ne 2
1735 .na
1736 \fB\fbUP\fR\fR
1737 .ad
1738 .sp .6
1739 .RS 4n
1740 Indicates that the logical interface (and the associated physical interface) is
1741 up. The IP module will accept packets destined to \fBUP\fR addresses (unless
1742 the address is zero), along with any associated multicast and broadcast IP
1743 addresses. Similarly, the IP module will allow packets to be sent with an
1744 \fBUP\fR address as a source address.
1745 .RE

1747 .sp
1748 .ne 2
1749 .na
1750 \fB\fbVIRTUAL\fR\fR
1751 .ad
1752 .sp .6
1753 .RS 4n
1754 Indicates that the physical interface has no underlying hardware. It is not
1755 possible to transmit or receive packets through a virtual interface. These
1756 interfaces are useful for configuring local addresses that can be used on
1757 multiple interfaces. (See also the \fBusesrc\fR option.)
1758 .RE

1760 .sp
1761 .ne 2
1762 .na
1763 \fB\fbXRESOLV\fR\fR
1764 .ad
1765 .sp .6
1766 .RS 4n
1767 Indicates that the interface uses an IPv6 external resolver.
1768 .RE

1770 .SH LOGICAL INTERFACES
1771 .sp
1772 .LP
1773 Solaris \fBfbTCP/IP\fR allows multiple logical interfaces to be associated with a
1774 physical network interface. This allows a single machine to be assigned
1775 multiple \fBfbIP\fR addresses, even though it may have only one network
1776 interface. Physical network interfaces have names of the form \fBdriver-name
1777 physical-unit-number\fR, while logical interfaces have names of the form

```

```

1778 \fBdriver-name physical-unit-number\fR:\fBdriver-logical-unit-number\fR. A
1779 physical interface is configured into the system using the \fBplumb\fR command.
1780 For example:
1781 .sp
1782 .in +2
1783 .nf
1784 example% \fBfbifconfig eri0 plumb\fR
1785 .fi
1786 .in -2
1787 .sp

1789 .sp
1790 .LP
1791 Once a physical interface has been "plumbed", logical interfaces associated
1792 with the physical interface can be configured by separate \fBplumb\fR or
1793 \fBfbaddif\fR options to the \fBfbifconfig\fR command.
1794 .sp
1795 .in +2
1796 .nf
1797 example% \fBfbifconfig eri0:1 plumb\fR
1798 .fi
1799 .in -2
1800 .sp

1802 .sp
1803 .LP
1804 allocates a specific logical interface associated with the physical interface
1805 \fBfberi0\fR. The command
1806 .sp
1807 .in +2
1808 .nf
1809 example% \fBfbifconfig eri0 addif 192.168.200.1/24 up\fR
1810 .fi
1811 .in -2
1812 .sp

1814 .sp
1815 .LP
1816 allocates the next available logical unit number on the \fBfberi0\fR physical
1817 interface and assigns an \fBfbaddress\fR and \fBfbiprefix_length\fR.
1818 .sp
1819 .LP
1820 A logical interface can be configured with parameters (
1821 \fBfbaddress\fR, \fBfbiprefix_length\fR, and so on) different from the physical
1822 interface with which it is associated. Logical interfaces that are associated
1823 with the same physical interface can be given different parameters as well.
1824 Each logical interface must be associated with an existing and "up" physical
1825 interface. So, for example, the logical interface \fBfberi0:1\fR can only be
1826 configured after the physical interface \fBfberi0\fR has been plumbed.
1827 .sp
1828 .LP
1829 To delete a logical interface, use the \fBfbunplumb\fR or \fBfbremoveif\fR options.
1830 For example,
1831 .sp
1832 .in +2
1833 .nf
1834 example% \fBfbifconfig eri0:1 down unplumb\fR
1835 .fi
1836 .in -2
1837 .sp

1839 .sp
1840 .LP
1841 will delete the logical interface \fBfberi0:1\fR.
1842 .SH IP MULTIPATHING GROUPS
1843 .sp

```

```

1844 .LP
1845 Physical interfaces that share the same link-layer broadcast domain \fBmust\fR
1846 be collected into a single IP Multipathing (IPMP) group using the \fBgroup\fR
1847 subcommand. Each IPMP group has an associated IPMP IP interface, which can
1848 either be explicitly created (the preferred method) by using the \fBbipmp\fR
1849 subcommand or implicitly created by \fBifconfig\fR in response to placing an IP
1850 interface into a new IPMP group. Implicitly-created IPMP interfaces will be
1851 named \fBbipmp\fR\fIN\fR where \fIN\fR is the lowest integer that does not
1852 conflict with an existing IP interface name or IPMP group name.
1853 .sp
1854 .LP
1855 Each IPMP IP interface is created with a matching IPMP group name, though it
1856 can be changed using the \fBgroup\fR subcommand. Each IPMP IP interface hosts a
1857 set of highly-available IP addresses. These addresses will remain reachable so
1858 long as at least one interface in the group is active, where "active" is
1859 defined as having at least one \fBUP\fR address and having \fBINACTIVE\fR,
1860 \fBFAILED\fR, and \fBOffline\fR clear. IP addresses hosted on the IPMP IP
1861 interface may either be configured statically or configured through DHCP by
1862 means of the \fBdhcp\fR subcommand.
1863 .sp
1864 .LP
1865 Interfaces assigned to the same IPMP group are treated as equivalent and
1866 monitored for failure by \fBin.mpathd\fR. Provided that active interfaces in
1867 the group remain, IP interface failures (and any subsequent repairs) are
1868 handled transparently to sockets-based applications. IPMP is also integrated
1869 with the Dynamic Reconfiguration framework (see \fBcfgadm\fR(1M)), which
1870 enables network adapters to be replaced in a way that is invisible to
1871 sockets-based applications.
1872 .sp
1873 .LP
1874 The IP module automatically load-spreads all outbound traffic across all active
1875 interfaces in an IPMP group. Similarly, all \fBUP\fR addresses hosted on the
1876 IPMP IP interface will be distributed across the active interfaces to promote
1877 inbound load-spreading. The \fBbipmpstat\fR(1M) utility allows many aspects of
1878 the IPMP subsystem to be observed, including the current binding of IP data
1879 addresses to IP interfaces.
1880 .sp
1881 .LP
1882 When an interface is placed into an IPMP group, any \fBUP\fR logical interfaces
1883 are "migrated" to the IPMP IP interface for use by the group, unless:
1884 .RS +4
1885 .TP
1886 .ie t \(\bu
1887 .el o
1888 the logical interface is marked \fBNOFAILOVER\fR;
1889 .RE
1890 .RS +4
1891 .TP
1892 .ie t \(\bu
1893 .el o
1894 the logical interface hosts an IPv6 link-local address;
1895 .RE
1896 .RS +4
1897 .TP
1898 .ie t \(\bu
1899 .el o
1900 the logical interface hosts an IPv4 0.0.0.0 address.
1901 .RE
1902 .sp
1903 .LP
1904 Likewise, once an interface is in a group, if changes are made to a logical
1905 interface such that it is \fBUP\fR and not exempted by one of the conditions
1906 above, it will also migrate to the associated IPMP IP interface. Logical
1907 interfaces never migrate back, even if the physical interface that contributed
1908 the address is removed from the group.
1909 .sp

```

```

1910 .LP
1911 Each interface placed into an IPMP group may be optionally configured with a
1912 "test" address that \fBin.mpathd\fR will use for probe-based failure detection;
1913 see \fBin.mpathd\fR(1M). These addresses must be marked \fBNOFAILOVER\fR (using
1914 the \fB-failover\fR subcommand) prior to being marked \fBUP\fR. Test addresses
1915 may also be acquired through DHCP by means of the \fBdhcp\fR subcommand.
1916 .sp
1917 .LP
1918 For more background on IPMP, please see the IPMP-related chapters of the
1919 \fISystem Administration Guide: Network Interfaces and Network
1920 Virtualization\fR.
1921 .SH CONFIGURING IPV6 INTERFACES
1922 .sp
1923 .LP
1924 When an IPv6 physical interface is plumbed and configured "up" with
1925 \fBifconfig\fR, it is automatically assigned an IPv6 link-local address for
1926 which the last 64 bits are calculated from the \fBMAC\fR address of the
1927 interface.
1928 .sp
1929 .in +2
1930 .nf
1931 example% \fBifconfig eri0 inet6 plumb up\fR
1932 .fi
1933 .in -2
1934 .sp

1936 .sp
1937 .LP
1938 The following example shows that the link-local address has a prefix of
1939 \fBfe80::/10\fR.
1940 .sp
1941 .in +2
1942 .nf
1943 example% \fBifconfig eri0 inet6\fR
1944 ce0: flags=2000841<UP,RUNNING,MULTICAST,IPv6>
1945         mtu 1500 index 2 \
1946         inet6 fe80::a00:20ff:fe8e:f3ad/10
1947 .fi
1948 .in -2
1949 .sp

1951 .sp
1952 .LP
1953 Link-local addresses are only used for communication on the local subnet and
1954 are not visible to other subnets.
1955 .sp
1956 .LP
1957 If an advertising IPv6 router exists on the link advertising prefixes, then the
1958 newly plumbed IPv6 interface will autoconfigure logical interface(s) depending
1959 on the prefix advertisements. For example, for the prefix advertisement
1960 \fB2001:0db8:3c4d:0:55::/64\fR, the autoconfigured interface will look like:
1961 .sp
1962 .in +2
1963 .nf
1964 eri0:2: flags=2080841<UP,RUNNING,MULTICAST,ADDRCONF,IPv6>
1965         mtu 1500 index 2
1966         inet6 2001:0db8:3c4d:55:a00:20ff:fe8e:f3ad/64
1967 .fi
1968 .in -2
1969 .sp

1971 .sp
1972 .LP
1973 Even if there are no prefix advertisements on the link, you can still assign
1974 global addresses manually, for example:
1975 .sp

```



```

1976 .in +2
1977 .nf
1978 example% \fBifconfig eri0 inet6 addif \e
1979 2001:0db8:3c4d:55:a00:20ff:fe8e:f3ad/64 up\fR
1980 .fi
1981 .in -2
1982 .sp

1984 .sp
1985 .LP
1986 To configure boot-time defaults for the interface \fBeri0\fR, place the
1987 following entry in the \fB/etc/hostname6.eri0\fR file:
1988 .sp
1989 .in +2
1990 .nf
1991 addif 2001:0db8:3c4d:55:a00:20ff:fe8e:f3ad/64 up
1992 .fi
1993 .in -2

1995 .SS "Configuring IP-over-IP Tunnel Interfaces"
1996 .sp
1997 .LP
1998 An IP tunnel is conceptually comprised of two parts: a virtual link between two
1999 or more IP nodes, and an IP interface above this link which allows the system
2000 to transmit and receive IP packets encapsulated by the underlying link.
2001 .sp
2002 .LP
2003 The \fBdldadm\fR(1M) command is used to configure tunnel links, and
2004 \fBifconfig\fR is used to configure IP interfaces over those tunnel links. An
2005 IPv4-over-IPv4 tunnel is created by plumbing an IPv4 interface over an IPv4
2006 tunnel link. An IPv6-over-IPv4 tunnel is created by plumbing an IPv6 interface
2007 over an IPv6 tunnel link, and so forth.
2008 .sp
2009 .LP
2010 When IPv6 interfaces are plumbed over IP tunnel links, their IPv6 addresses are
2011 automatically set. For IPv4 and IPv6 tunnels, source and destination link-local
2012 addresses of the form \fBfe80::\fR\fIinterface-id\fR are configured. For IPv4
2013 tunnels, the \fIinterface-id\fR is the IPv4 tunnel source or destination
2014 address. For IPv6 tunnels, the \fIinterface-id\fR is the last 64 bits of the
2015 IPv6 tunnel source or destination address. For example, for an IPv4 tunnel
2016 between 10.1.2.3 and 10.4.5.6, the IPv6 link-local source and destination
2017 addresses of the IPv6 interface would be \fBfe80::a01:203\fR and
2018 \fBfe80::a04:506\fR. For an IPv6 tunnel between \fB2000::1234:abcd\fR and
2019 \fB3000::5678:abcd\fR, the IPv6 link-local source and destination addresses of
2020 the interface would be \fBfe80::1234:abcd\fR and \fBfe80::5678:abcd\fR. These
2021 default link-local addresses can be overridden by specifying the addresses
2022 explicitly, as with any other point-to-point interface.
2023 .sp
2024 .LP
2025 For 6to4 tunnels, a 6to4 global address of the form \fB2002:\fR\fIsrc\fR::1/16\fR
2026 is configured. The \fIsrc\fR portion is the tunnel source IPv4 address. The
2027 prefix length of the 6to4 interface is automatically set to 16, as all 6to4
2028 packets (destinations in the \fB2002::/16\fR range) are forwarded to the 6to4
2029 tunnel interface. For example, for a 6to4 link with a tunnel source of
2030 75.1.2.3, the IPv6 interface would have an address of
2031 \fB2002:4b01:203::1/16\fR.
2032 .sp
2033 .LP
2034 Additional IPv6 addresses can be added using the \fBaddif\fR option or by
2035 plumbing additional logical interfaces.
2036 .sp
2037 .LP
2038 For backward compatibility, the plumbing of tunnel IP interfaces with special
2039 names will implicitly result in the creation of tunnel links without invoking
2040 \fBdldadm create-iptun\fR. These tunnel names are:
2041 .sp

```

```

2042 .ne 2
2043 .na
2044 \fB\bip.tun\fR\fR\fR
2045 .ad
2046 .RS 15n
2047 An IPv4 tunnel
2048 .RE

2050 .sp
2051 .ne 2
2052 .na
2053 \fB\bip6.tun\fR\fR\fR
2054 .ad
2055 .RS 15n
2056 An IPv6 tunnel
2057 .RE

2059 .sp
2060 .ne 2
2061 .na
2062 \fB\bip.6to4tun\fR\fR\fR
2063 .ad
2064 .RS 15n
2065 A 6to4 tunnel
2066 .RE

2068 .sp
2069 .LP
2070 These tunnels are "implicit tunnels", denoted with the \fBif\fR flag in \fBdldadm
2071 show-iptun\fR output. The tunnel links over which these special IP interfaces
2072 are plumbed are automatically created, and they are automatically deleted when
2073 the last reference is released (that is, when the last IP interface is
2074 unplumbed).
2075 .sp
2076 .LP
2077 The \fBtsrc\fR, \fBtdst\fR, \fBencaplim\fR, and \fBhoplimit\fR options to
2078 \fBifconfig\fR are obsolete and maintained only for backward compatibility.
2079 They are equivalent to their \fBdldadm\fR(1M) counterparts.
2080 .SS "Display of Tunnel Security Settings"
2081 .sp
2082 .LP
2083 The \fBifconfig\fR output for IP tunnel interfaces indicates whether IPsec
2084 policy is configured for the underlying IP tunnel link. For example, a line of
2085 the following form will be displayed if IPsec policy is present:
2086 .sp
2087 .in +2
2088 .nf
2089 tunnel security settings --> use 'ipsecconf -ln -i ip.tun1'
2090 .fi
2091 .in -2
2092 .sp

2094 .sp
2095 .LP
2096 If you do net set security policy, using either \fBifconfig\fR or
2097 \fBbipsecconf\fR(1M), there is no tunnel security setting displayed.
2098 .SH EXAMPLES
2099 .LP
2100 \fBExample 1\fR Using the \fBifconfig\fR Command
2101 .sp
2102 .LP
2103 If your workstation is not attached to an Ethernet, the network interface, for
2104 example, \fBeri0\fR, should be marked "down" as follows:

2106 .sp
2107 .in +2

```

```

2108 .nf
2109 example% \fBifconfig eri0 down\fR
2110 .fi
2111 .in -2
2112 .sp

2114 .LP
2115 \fBExample 2 \fRPrinting Addressing Information
2116 .sp
2117 .LP
2118 To print out the addressing information for each interface, use the following
2119 command:

2121 .sp
2122 .in +2
2123 .nf
2124 example% \fBifconfig -a\fR
2125 .fi
2126 .in -2
2127 .sp

2129 .LP
2130 \fBExample 3 \fRResetting the Broadcast Address
2131 .sp
2132 .LP
2133 To reset each interface's broadcast address after the netmasks have been
2134 correctly set, use the next command:

2136 .sp
2137 .in +2
2138 .nf
2139 example% \fBifconfig -a broadcast +\fR
2140 .fi
2141 .in -2
2142 .sp

2144 .LP
2145 \fBExample 4 \fRChanging the Ethernet Address
2146 .sp
2147 .LP
2148 To change the Ethernet address for interface \fBce0\fR, use the following
2149 command:

2151 .sp
2152 .in +2
2153 .nf
2154 example% \fBifconfig ce0 ether aa:1:2:3:4:5\fR
2155 .fi
2156 .in -2
2157 .sp

2159 .LP
2160 \fBExample 5 \fRConfiguring an IP-in-IP Tunnel
2161 .sp
2162 .LP
2163 To configure an IP-in-IP tunnel, first create an IP tunnel link (\fBtunsrc\fR
2164 and \fBtundst\fR are hostnames with corresponding IPv4 entries in
2165 \fB/etc/hosts\fR):

2167 .sp
2168 .in +2
2169 .nf
2170 example% \fBdladm create-iptun -T ipv4 -s tunsrc -d tundst tun0\fR
2171 .fi
2172 .in -2
2173 .sp

```

```

2175 .sp
2176 .LP
2177 Then plumb a point-to-point interface, supplying the source and destination
2178 addresses (\fBmysrc\fR and \fBthedst\fR are hostnames with corresponding IPv4
2179 entries in \fB/etc/hosts\fR):

2181 .sp
2182 .in +2
2183 .nf
2184 example% \fBifconfig tun0 plumb mysrc thedst up\fR
2185 .fi
2186 .in -2
2187 .sp

2189 .sp
2190 .LP
2191 Use \fBipsecconf\fR(1M), as described above, to configure tunnel security
2192 properties.

2194 .sp
2195 .LP
2196 Configuring IPv6 tunnels is done by using a tunnel type of \fBbip6\fR with
2197 \fBbcreate-iptun\fR. IPv6 interfaces can also be plumbed over either type of
2198 tunnel.

2200 .LP
2201 \fBExample 6 \fRConfiguring 6to4 Tunnels
2202 .sp
2203 .LP
2204 To configure 6to4 tunnels, first create a 6to4 tunnel link (\fBmyv4addr\fR is a
2205 hostname with a corresponding IPv4 entry in \fB/etc/hosts\fR):

2207 .sp
2208 .in +2
2209 .nf
2210 example% \fBdladm create-iptun -T 6to4 -s myv4addr my6to4tun0\fR
2211 .fi
2212 .in -2
2213 .sp

2215 .sp
2216 .LP
2217 Then an IPv6 interface is plumbed over this link:

2219 .sp
2220 .in +2
2221 .nf
2222 example% \fBifconfig my6to4tun0 inet6 plumb up\fR
2223 .fi
2224 .in -2
2225 .sp

2227 .sp
2228 .LP
2229 The IPv6 address of the interface is automatically set as described above.

2231 .LP
2232 \fBExample 7 \fRConfiguring IP Forwarding on an Interface
2233 .sp
2234 .LP
2235 To enable IP forwarding on a single interface, use the following command:

2237 .sp
2238 .in +2
2239 .nf

```

```

2240 example% \fBifconfig eri0 router\fR
2241 .fi
2242 .in -2
2243 .sp

2245 .sp
2246 .LP
2247 To disable IP forwarding on a single interface, use the following command:

2249 .sp
2250 .in +2
2251 .nf
2252 example% \fBifconfig eri0 -router\fR
2253 .fi
2254 .in -2
2255 .sp

2257 .LP
2258 \fBExample 8 \fRConfiguring Source Address Selection Using a Virtual Interface
2259 .sp
2260 .LP
2261 The following command configures source address selection such that every
2262 packet that is locally generated with no bound source address and going out on
2263 \fBqfe2\fR prefers a source address hosted on \fBvni0\fR.

2265 .sp
2266 .in +2
2267 .nf
2268 example% \fBifconfig qfe2 usesrc vni0\fR
2269 .fi
2270 .in -2
2271 .sp

2273 .sp
2274 .LP
2275 The \fBifconfig\fR \fB-a\fR output for the \fBqfe2\fR and \fBvni0\fR interfaces
2276 displays as follows:

2278 .sp
2279 .in +2
2280 .nf
2281 qfe2: flags=1100843<UP,BROADCAST,RUNNING,MULTICAST,ROUTER,IPv4> mtu
2282 1500 index 4
2283 usesrc vni0
2284 inet 1.2.3.4 netmask ffffffff broadcast 1.2.3.255
2285 ether 0:3:ba:17:4b:e1
2286 vni0: flags=20011100c1<UP,RUNNING,NOARP,NOXMIT,ROUTER,IPv4,VIRTUAL>
2287 mtu 0 index 5
2288 srcof qfe2
2289 inet 3.4.5.6 netmask ffffffff
2290 .fi
2291 .in -2

2293 .sp
2294 .LP
2295 Observe, above, the \fBusesrc\fR and \fBsrcof\fR keywords in the \fBifconfig\fR
2296 output. These keywords also appear on the logical instances of the physical
2297 interface, even though this is a per-physical interface parameter. There is no
2298 \fBsrcof\fR keyword in \fBifconfig\fR for configuring interfaces. This
2299 information is determined automatically from the set of interfaces that have
2300 \fBusesrc\fR set on them.

2302 .sp
2303 .LP
2304 The following command, using the \fBnone\fR keyword, undoes the effect of the
2305 preceding \fBifconfig\fR \fBusesrc\fR command.

```

```

2307 .sp
2308 .in +2
2309 .nf
2310 example% \fBifconfig qfe2 usesrc none\fR
2311 .fi
2312 .in -2
2313 .sp

2315 .sp
2316 .LP
2317 Following this command, \fBifconfig\fR \fB-a\fR output displays as follows:

2319 .sp
2320 .in +2
2321 .nf
2322 qfe2: flags=1100843<UP,BROADCAST,RUNNING,MULTICAST,ROUTER,IPv4> mtu
2323 1500 index 4
2324 inet 1.2.3.4 netmask ffffffff broadcast 1.2.3.255
2325 ether 0:3:ba:17:4b:e1
2326 vni0: flags=20011100c1<UP,RUNNING,NOARP,NOXMIT,ROUTER,IPv4,VIRTUAL>
2327 mtu 0 index 5
2328 inet 3.4.5.6 netmask ffffffff
2329 .fi
2330 .in -2

2332 .sp
2333 .LP
2334 Note the absence of the \fBusesrc\fR and \fBsrcof\fR keywords in the output
2335 above.

2337 .LP
2338 \fBExample 9 \fRConfiguring Source Address Selection for an IPv6 Address
2339 .sp
2340 .LP
2341 The following command configures source address selection for an IPv6 address,
2342 selecting a source address hosted on \fBvni0\fR.

2344 .sp
2345 .in +2
2346 .nf
2347 example% \fBifconfig qfe1 inet6 usesrc vni0\fR
2348 .fi
2349 .in -2
2350 .sp

2352 .sp
2353 .LP
2354 Following this command, \fBifconfig\fR \fB-a\fR output displays as follows:

2356 .sp
2357 .in +2
2358 .nf
2359 qfe1: flags=2000841<UP,RUNNING,MULTICAST,IPv6> mtu 1500 index 3
2360 usesrc vni0
2361 inet6 fe80::203:baff:fe17:4be0/10
2362 ether 0:3:ba:17:4b:e0
2363 vni0: flags=2002210041<UP,RUNNING,NOXMIT,NOAUD,IPv6,VIRTUAL> mtu 0
2364 index 5
2365 srcof qfe1
2366 inet6 fe80::203:baff:fe17:4444/128
2367 vni0:1: flags=2002210040<RUNNING,NOXMIT,NOAUD,IPv6,VIRTUAL> mtu 0
2368 index 5
2369 srcof qfe1
2370 inet6 fec0::203:baff:fe17:4444/128
2371 vni0:2: flags=2002210040<RUNNING,NOXMIT,NOAUD,IPv6,VIRTUAL> mtu 0

```

```

2372 index 5
2373 srcof qfel
2374 inet6 2000::203:baff:fe17:4444/128
2375 .fi
2376 .in -2

2378 .sp
2379 .LP
2380 Depending on the scope of the destination of the packet going out on
2381 \fBqfel\fR, the appropriately scoped source address is selected from \fBvni0\fR
2382 and its aliases.

2384 .LP
2385 \fBExample 10 \fRUsing Source Address Selection with Shared-IP Zones
2386 .sp
2387 .LP
2388 The following is an example of how the \fBusesrc\fR feature can be used with
2389 the \fBzones\fR(5) facility in Solaris. The following commands are invoked in
2390 the global zone:

2392 .sp
2393 .in +2
2394 .nf
2395 example% \fBifconfig hme0 usesrc vni0\fR
2396 example% \fBifconfig eri0 usesrc vni0\fR
2397 example% i\fBifconfig qfe0 usesrc vni0\fR
2398 .fi
2399 .in -2
2400 .sp

2402 .sp
2403 .LP
2404 Following the preceding commands, the \fBifconfig\fR \fB-a\fR output for the
2405 virtual interfaces would display as:

2407 .sp
2408 .in +2
2409 .nf
2410 vni0: flags=20011100c1<UP,RUNNING,NOARP,NOXMIT,ROUTER,IPv4,VIRTUAL>
2411 mtu 0 index 23
2412 srcof hme0 eri0 qfe0
2413 inet 10.0.0.1 netmask ffffffff
2414 vni0:1:
2415 flags=20011100c1<UP,RUNNING,NOARP,NOXMIT,ROUTER,IPv4,VIRTUAL> mtu 0
2416 index 23
2417 zone test1
2418 srcof hme0 eri0 qfe0
2419 inet 10.0.0.2 netmask ffffffff
2420 vni0:2:
2421 flags=20011100c1<UP,RUNNING,NOARP,NOXMIT,ROUTER,IPv4,VIRTUAL> mtu 0
2422 index 23
2423 zone test2
2424 srcof hme0 eri0 qfe0
2425 inet 10.0.0.3 netmask ffffffff
2426 vni0:3:
2427 flags=20011100c1<UP,RUNNING,NOARP,NOXMIT,ROUTER,IPv4,VIRTUAL> mtu 0
2428 index 23
2429 zone test3
2430 srcof hme0 eri0 qfe0
2431 inet 10.0.0.4 netmask ffffffff
2432 .fi
2433 .in -2

2435 .sp
2436 .LP
2437 There is one virtual interface alias per zone (\fBtest1\fR, \fBtest2\fR, and

```

```

2438 \fBtest3\fR). A source address from the virtual interface alias in the same
2439 zone is selected. The virtual interface aliases were created using
2440 \fBzonecfg\fR(1M) as follows:

2442 .sp
2443 .in +2
2444 .nf
2445 example% \fBzonecfg -z test1\fR
2446 zonecfg:test1> \fBadd net\fR
2447 zonecfg:test1:net> \fBset physical=vni0\fR
2448 zonecfg:test1:net> \fBset address=10.0.0.2\fR
2449 .fi
2450 .in -2
2451 .sp

2453 .sp
2454 .LP
2455 The \fBtest2\fR and \fBtest3\fR zone interfaces and addresses are created in
2456 the same way.

2458 .LP
2459 \fBExample 11 \fRTurning Off DHCPv6
2460 .sp
2461 .LP
2462 The following example shows how to disable automatic use of DHCPv6 on all
2463 interfaces, and immediately shut down DHCPv6 on the interface named \fBhme0\fR.
2464 See \fBin.ndpd\fR(1M) and \fBndpd.conf\fR(4) for more information on the
2465 automatic DHCPv6 configuration mechanism.

2467 .sp
2468 .in +2
2469 .nf
2470 example% \fBecho ifdefault StatefulAddrConf false >> /etc/inet/ndpd.conf\fR
2471 example% \fBpkill -HUP -x in.ndpd\fR
2472 example% \fBifconfig hme0 dhcp release\fR
2473 .fi
2474 .in -2
2475 .sp

2477 .SH FILES
2478 .sp
2479 .ne 2
2480 .na
2481 \fB\FB/etc/netmasks\fR
2482 .ad
2483 .sp .6
2484 .RS 4n
2485 Netmask data.
2486 .RE

2488 .sp
2489 .ne 2
2490 .na
2491 \fB\FB/etc/default/inet_type\fR
2492 .ad
2493 .sp .6
2494 .RS 4n
2495 Default Internet protocol type.
2496 .RE

2498 .SH ATTRIBUTES
2499 .sp
2500 .LP
2501 See \fBattributes\fR(5) for descriptions of the following attributes:
2502 .sp

```

```
2504 .sp
2505 .TS
2506 box;
2507 c | c
2508 l | l .
2509 ATTRIBUTE TYPE ATTRIBUTE VALUE
2510 ─
2511 T{
2512 Interface Stability for command-line options
2513 T} Committed
2514 ─
2515 Interface Stability for command output Uncommitted
2516 .TE

2518 .SH SEE ALSO
2519 .sp
2520 .LP
2521 \fBdhcpcinfo\fR(1), \fBcfdm\fR(1M), \fBdhcpcagent\fR(1M), \fBdldm\fR(1M),
2522 \fBif_mpadm\fR(1M), \fBbin.mpathd\fR(1M), \fBbin.ndpd\fR(1M), \fBbin.routed\fR(1M),
2523 \fBbipadm\fR(1M), \fBbimpstat\fR(1M), \fBbipseconf\fR(1M), \fBbndd\fR(1M),
2524 \fBbif_mpadm\fR(1M), \fBbin.mpathd\fR(1M), \fBbin.ndpd\fR(1M),
2525 \fBbin.routed\fR(1M), \fBbimpstat\fR(1M), \fBbipseconf\fR(1M), \fBbndd\fR(1M),
2526 \fBbnetstat\fR(1M), \fBzoneadm\fR(1M), \fBzonecfg\fR(1M), \fBbethers\fR(3SOCKET),
2527 \fBgethostbyname\fR(3NSL), \fBgetnetbyname\fR(3SOCKET), \fBhosts\fR(4),
2528 \fBinet_type\fR(4), \fBndpd.conf\fR(4), \fBnetmasks\fR(4), \fBnetworks\fR(4),
2529 \fBnsswitch.conf\fR(4), \fBattributes\fR(5), \fBprivileges\fR(5),
2530 \fBzones\fR(5), \fBarp\fR(7P), \fBbipsec\fR(7P), \fBbipsecesp\fR(7P)
2531 .sp
2532 .LP
2533 \fBifconfig\fR sends messages that indicate if:
2534 .RS +4
2535 .ie t \(\bu
2536 .el o
2537 the specified interface does not exist
2538 .RE
2539 .RS +4
2540 .ie t \(\bu
2541 .el o
2542 the requested address is unknown
2543 .RE
2544 .RS +4
2545 .ie t \(\bu
2546 .el o
2547 the user is not privileged and tried to alter an interface's configuration
2548 .RE
2549 .SH NOTES
2550 .sp
2551 .LP
2552 Do not select the names \fBbroadcast\fR, \fBdown\fR, \fBprivate\fR,
2553 \fBtrailers\fR, \fBup\fR or other possible option names when you choose host
2554 names. If you choose any one of these names as host names, it can cause unusual
2555 problems that are extremely difficult to diagnose.
```

21834 Sun Oct 28 17:19:47 2012

new/usr/src/man/man1m/ipadm.1m

815 Need ipadm(1M) manual page

```

1  \" te
2  .\" Copyright (c) 2012, Joyent, Inc. All Rights Reserved
3  .\" The contents of this file are subject to the terms of the Common Development
4  .\" You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE or http:
5  .\" When distributing Covered Code, include this CDDL HEADER in each file and in
6  .TH IPADM 1M \"May 14, 2012\"
7  .SH NAME
8  ipadm \- configure IP network interfaces and protocol properties.
9  .SH SYNOPSIS
10 .LP
11 .nf
12 \fBipadm\fR create-if [\fB-t\fR] \fIinterface\fR
13 .fi
14
15 .LP
16 .nf
17 \fBipadm\fR disable-if \fB-t\fR \fIinterface\fR
18 .fi
19
20 .LP
21 .nf
22 \fBipadm\fR enable-if \fB-t\fR \fIinterface\fR
23 .fi
24
25 .LP
26 .nf
27 \fBipadm\fR delete-if \fIinterface\fR
28 .fi
29
30 .LP
31 .nf
32 \fBipadm\fR show-if [[\fB-p\fR] \fB-o\fR \fIfield\fR[,...]] [\fIinterface\fR]
33 .fi
34
35 .LP
36 .nf
37 \fBipadm\fR set-ifprop [\fB-t\fR] \fB-p\fR \fIprop\fR=<\fIvalue\fR[,...]> \fB-m\
38 .fi
39
40 .LP
41 .nf
42 \fBipadm\fR reset-ifprop [\fB-t\fR] \fB-p\fR \fIprop\fR \fB-m\fR \fIprotocol\fR
43 .fi
44
45 .LP
46 .nf
47 \fBipadm\fR show-ifprop [[\fB-c\fR] \fB-o\fR \fIfield\fR[,...]] [\fB-p\fR \fIprop
48 [\fIinterface\fR]
49 .fi
50
51 .LP
52 .nf
53 \fBipadm\fR create-addr [\fB-t\fR] \fB-T\fR static [\fB-d\fR]
54 \fB-a\fR {local|remote}=\fIaddr\fR[/\fIprefixlen\fR],... \fIaddrobj\fR
55 .fi
56
57 .LP
58 .nf
59 \fBipadm\fR create-addr [\fB-t\fR] \fB-T\fR dhcp [\fB-w\fR \fIseconds\fR | forev
60 .fi

```

```

62 .LP
63 .nf
64 \fBipadm\fR create-addr [\fB-t\fR] \fB-T\fR addrconf [\fB-i\fR \fIinterface-id\f
65 [\fB-p\fR {stateful|stateless}={yes|no},...] \fIaddrobj\fR
66 .fi
67
68 .LP
69 .nf
70 \fBipadm\fR down-addr [\fB-t\fR] \fIaddrobj\fR
71 .fi
72
73 .LP
74 .nf
75 \fBipadm\fR up-addr [\fB-t\fR] \fIaddrobj\fR
76 .fi
77
78 .LP
79 .nf
80 \fBipadm\fR disable-addr \fB-t\fR \fIaddrobj\fR
81 .fi
82
83 .LP
84 .nf
85 \fBipadm\fR enable-addr \fB-t\fR \fIaddrobj\fR
86 .fi
87
88 .LP
89 .nf
90 \fBipadm\fR refresh-addr [\fB-i\fR] \fIaddrobj\fR
91 .fi
92
93 .LP
94 .nf
95 \fBipadm\fR delete-addr [\fB-r\fR] \fIaddrobj\fR
96 .fi
97
98 .LP
99 .nf
100 \fBipadm\fR show-addr [[\fB-p\fR] \fB-o\fR \fIfield\fR[,...]] [\fIaddrobj\fR]
101 .fi
102
103 .LP
104 .nf
105 \fBipadm\fR set-addrprop [\fB-t\fR] \fB-p\fR \fIprop\fR=<\fIvalue\fR[,...]> \fIa
106 .fi
107
108 .LP
109 .nf
110 \fBipadm\fR reset-addrprop [\fB-t\fR] \fB-p\fR \fIprop\fR=<\fIvalue\fR[,...]> \f
111 .fi
112
113 .LP
114 .nf
115 \fBipadm\fR show-addrprop [[\fB-c\fR] \fB-o\fR \fIfield\fR[,...]] [\fB-p\fR \fIp
116 .fi
117
118 .LP
119 .nf
120 \fBipadm\fR set-prop [\fB-t\fR] \fB-p\fR \fIprop\fR[+|-]=<\fIvalue\fR[,...]> \fI
121 .fi
122
123 .LP
124 .nf
125 \fBipadm\fR reset-prop [\fB-t\fR] \fB-p\fR \fIprop\fR \fIprotocol\fR
126 .fi

```

```

128 .LP
129 .nf
130 \fBipadm\fR show-prop [[\fB-c\fR] \fB-o\fR \fIfield\fR[,...]] [\fB-p\fR \fIprop\
131 .fi

133 .SH DESCRIPTION
134 .sp
135 .LP

137 The \fBipadm\fR command is a stable replacement for the \fBifconfig\fR(1M) and
138 \fBndd\fR(1M) commands. It is used to create IP interfaces and to configure IP
139 addresses on those interfaces. It is also used to get, set or reset properties
140 on interfaces, addresses and protocols.
141 .LP
142 For subcommands that take an \fIaddrobj\fR, the \fIaddrobj\fR specifies a
143 unique address on the system. It is made up of two parts, delimited by a '/'.
144 The first part is the name of the interface and the second part is a string up
145 to 32 characters long. For example, "lo0/v4" is a loopback interface
146 addrobj name.
147 .LP
148 For subcommands that take a \fIprotocol\fR, this can be one of
149 the following values: ip, ipv4, ipv6, icmp, tcp, sctp or udp.

151 .SH SUBCOMMANDS
152 .sp
153 .LP
154 The following subcommands are supported:
155 .sp
156 .ne 2
157 .na
158 \fBfbcreate-if\fR [\fB-t\fR] \fIinterface\fR\fR
159 .ad
160 .sp .6
161 .RS 4n
162 The \fBfbcreate-if\fR subcommand is used to create an IP interface that will
163 handle both IPv4 and IPv6 packets. The interface will be enabled as part of
164 the creation process. The IPv4 interface will have the address 0.0.0.0.
165 The IPv6 interface will have the address ::.
166 .sp
167 The \fB-t\fR option (also \fB--temporary\fR) means
168 that the creation is temporary and will not be persistent across reboots.
169 .sp

171 .RE

173 .sp
174 .ne 2
175 .na
176 \fBfbdisable-if\fR \fB-t\fR \fIinterface\fR\fR
177 .ad
178 .sp .6
179 .RS 4n
180 The \fBfbdisable-if\fR subcommand is used to disable an IP interface.
181 .sp
182 The \fB-t\fR option (also \fB--temporary\fR) means
183 that the disable is temporary and will not be persistent across reboots.
184 .sp

186 .RE

188 .sp
189 .ne 2
190 .na
191 \fBfbenable-if\fR \fB-t\fR \fIinterface\fR\fR
192 .ad
193 .sp .6

```

```

194 .RS 4n
195 The \fBfbenable-if\fR subcommand is used to enable an IP interface.
196 .sp
197 The \fB-t\fR option (also \fB--temporary\fR) means
198 that the enable is temporary and will not be persistent across reboots.
199 .sp

201 .RE

203 .sp
204 .ne 2
205 .na
206 \fBfbdelete-if\fR \fIinterface\fR\fR
207 .ad
208 .sp .6
209 .RS 4n
210 The \fBdelete-if\fR subcommand is used to permanently delete an IP interface.
211 .sp

213 .RE

215 .sp
216 .ne 2
217 .na
218 \fBfbshow-if\fR [[\fB-p\fR] \fB-o\fR \fIfield\fR[,...]] [\fIinterface\fR]\fR
219 .ad
220 .sp .6
221 .RS 4n
222 The \fBfbshow-if\fR subcommand is used to show the current IP interface
223 configuration.
224 .sp
225 The \fB-p\fR option (also \fB--parsable\fR) prints
226 the output in a parsable format.
227 .sp
228 The \fB-o\fR option (also \fB--output\fR) is used
229 to select which fields will be shown. The field value can be one of the
230 following names:
231 .sp
232 .ne 2
233 .na
234 .RS 4n
235 \fBFBALL\fR
236 .ad
237 .RS 4n
238 Display all fields
239 .RE

241 .sp
242 .ne 2
243 .na
244 \fBFBIFNAME\fR
245 .ad
246 .RS 4n
247 The name of the interface
248 .RE

250 .sp
251 .ne 2
252 .na
253 \fBFBSTATE\fR
254 .ad
255 .RS 4n
256 The state can be one of the following values:
257 .sp
258 .ne 2
259 .na

```

```

260 .RS 4n
261 ok - resources for the interface have been allocated
262 .sp
263 offline - the interface is offline
264 .sp
265 failed - the interface's datalink is down
266 .sp
267 down - the interface is down
268 .sp
269 disabled - the interface is disabled
270 .RE
271 .RE

273 .sp
274 .ne 2
275 .na
276 \fBCURRENT\fR
277 .ad
278 .RS 4n
279 A set of single character flags indicating the following:
280 .sp
281 .ne 2
282 .na
283 .RS 4n
284 b - broadcast (mutually exclusive with 'p')
285 .br
286 m - multicast
287 .br
288 p - point-to-point (mutually exclusive with 'b')
289 .br
290 v - virtual interface
291 .br
292 I - IPMP
293 .br
294 s - IPMP standby
295 .br
296 i - IPMP inactive
297 .br
298 V - VRRP
299 .br
300 a - VRRP accept mode
301 .br
302 4 - IPv4
303 .br
304 6 - IPv6
305 .RE
306 .RE

308 .sp
309 .ne 2
310 .na
311 \fBPERSISTENT\fR
312 .ad
313 .RS 4n
314 A set of single character flags showing what configuration will be used the
315 next time the interface is enabled:
316 .sp
317 .ne 2
318 .na
319 .RS 4n
320 s - IPMP standby
321 .br
322 4 - IPv4
323 .br
324 6 - IPv6
325 .RE

```

```

326 .RE
327 .RE

329 .RE

331 .sp
332 .ne 2
333 .na
334 \fB\fBset-ifprop\fR [\fB-t\fR] \fB-p\fR \fIprop\fR=<\fIvalue\fR[,...]> \fB-m\fR
335 .ad
336 .sp .6
337 .RS 4n
338 The \fBset-ifprop\fR subcommand is used to set a property's value(s) on the IP
339 interface.
340 .sp
341 The \fB-t\fR option (also \fB--temporary\fR) means
342 that the setting is temporary and will not be persistent across reboots.
343 .sp
344 The \fB-p\fR option (also \fB--prop\fR) specifies the property name and
345 value(s). The property name can be one of the following:
346 .sp
347 .ne 2
348 .na

350 .RS 4n

352 \fBarp\fR
353 .ad
354 .RS 4n
355 Enables ("on") or disables ("off") ARP.
356 .RE

358 .sp
359 .ne 2
360 .na
361 \fBexchange_routes\fR
362 .ad
363 .RS 4n
364 Enables ("on") or disables ("off") the exchange of routing data.
365 .RE

367 .sp
368 .ne 2
369 .na
370 \fBforwarding\fR
371 .ad
372 .RS 4n
373 Enables ("on") or disables ("off") IP forwarding.
374 .RE

376 .sp
377 .ne 2
378 .na
379 \fBmetric\fR
380 .ad
381 .RS 4n
382 Set the routing metric to the numeric value. The value is treated as extra
383 hops to the destination.
384 .RE

386 .sp
387 .ne 2
388 .na
389 \fBmtu\fR
390 .ad
391 .RS 4n

```



```

392 Set the maximum transmission unit to the numeric value.
393 .RE

395 .sp
396 .ne 2
397 .na
398 \fBnud\fR
399 .ad
400 .RS 4n
401 Enables ("on") or disables ("off") neighbor unreachability detection.
402 .RE

404 .sp
405 .ne 2
406 .na
407 \fBusesrc\fR
408 .ad
409 .RS 4n
410 Indicates which interface to use for source address selection. A value
411 "none" may also be used.
412 .RE
413 .RE

415 .sp
416 The \fB-m\fR option (also \fB--module\fR) specifies which protocol
417 the setting applies to.
418 .sp

420 .RE
421 .RE

423 .sp
424 .ne 2
425 .na
426 \fB\breset-ifprop\fR [\fB-t\fR] \fB-p\fR \fIprop\fR \fB-m\fR \fIprotocol\fR \fI
427 .ad
428 .sp .6
429 .RS 4n
430 The \fBreset-ifprop\fR subcommand is used to reset an IP interface's property
431 value to the default.
432 .sp
433 The \fB-t\fR option (also \fB--temporary\fR) means
434 that the disable is temporary and will not be persistent across reboots.
435 .sp
436 The \fB-p\fR option (also \fB--prop\fR) specifies the property name.
437 See the \fBset-ifprop\fR subcommand for the list of property names.
438 .sp
439 The \fB-m\fR option (also \fB--module\fR) specifies which protocol
440 the setting applies to.
441 .sp

443 .RE

445 .sp
446 .ne 2
447 .na
448 \fB\bshow-ifprop\fR [[\fB-c\fR]\fB-o\fR \fIfield\fR[,...]] [\fB-p\fR \fIprop\fR
449 [\fIinterface\fR]\fR
450 .ad
451 .sp .6
452 .RS 4n
453 The \fBshow-ifprop\fR subcommand is used to display the property values
454 for one or all of the IP interfaces.
455 .sp
456 The \fB-c\fR option (also \fB--parsable\fR) prints
457 the output in a parsable format.

```

```

458 .sp
459 The \fB-o\fR option (also \fB--output\fR) is used
460 to select which fields will be shown. The field value can be one of the
461 following names:
462 .sp
463 .ne 2
464 .na
465 .RS 4n
466 \fBALL\fR
467 .ad
468 .RS 4n
469 Display all fields
470 .RE

472 .sp
473 .ne 2
474 .na
475 \fBIFNAME\fR
476 .ad
477 .RS 4n
478 The name of the interface
479 .RE

481 .sp
482 .ne 2
483 .na
484 \fBPROPERTY\fR
485 .ad
486 .RS 4n
487 The name of the property
488 .RE

490 .sp
491 .ne 2
492 .na
493 \fBPROTO\fR
494 .ad
495 .RS 4n
496 The name of the protocol
497 .RE

499 .sp
500 .ne 2
501 .na
502 \fBPERM\fR
503 .ad
504 .RS 4n
505 If the property is readable ("r") and/or writable ("w").
506 .RE

508 .sp
509 .ne 2
510 .na
511 \fBCURRENT\fR
512 .ad
513 .RS 4n
514 The value of the property
515 .RE

517 .sp
518 .ne 2
519 .na
520 \fBPERSISTENT\fR
521 .ad
522 .RS 4n
523 The persistent value of the property

```

```

524 .RE

526 .sp
527 .ne 2
528 .na
529 \fBDEFAULT\fR
530 .ad
531 .RS 4n
532 The default value of the property
533 .RE

535 .sp
536 .ne 2
537 .na
538 \fBPOSSIBLE\fR
539 .ad
540 .RS 4n
541 The possible values for the property
542 .RE
543 .RE

545 .sp
546 The \fB-p\fR option (also \fB--prop\fR) is used
547 to specify which properties to display. See the \fBset-ifprop\fR
548 subcommand for the list of property names.
549 .sp
550 The \fB-m\fR option (also \fB--module\fR) specifies which protocol
551 to display.
552 .sp

554 .RE

556 .sp
557 .ne 2
558 .na
559 \fB\fBcreate-addr\fR [\fB-t\fR] \fB-T\fR static [\fB-d\fR] \\  

560 \fB-a\fR {local|remote}=\fIaddr\fR[/\fIprefixlen\fR],... \fIaddrobj\fR\fR
561 .br
562 \fB\fBcreate-addr\fR [\fB-t\fR] \fB-T\fR dhcp [\fB-w\fR \fIseconds\fR | forever
563 .br
564 \fB\fBcreate-addr\fR [\fB-t\fR] \fB-T\fR addrconf [\fB-i\fR \fIinterface-id\fR]
565 [\fB-p\fR {stateful|stateless}={yes|no},...] \fIaddrobj\fR\fR
566 .ad
567 .sp .6
568 .RS 4n
569 The \fBcreate-addr\fR subcommand is used to set an address on an IP interface.
570 The address will be enabled but can disabled using the \fBdisable-addr\fR
571 subcommand. This subcommand has three different forms, depending on the
572 value of the \fB-T\fR option.
573 .sp
574 The \fB-t\fR option (also \fB--temporary\fR) means
575 that the address is temporary and will not be persistent across reboots.
576 .sp
577 The \fB-T\fR static option creates a static addrobj. This takes the following
578 options:
579 .RS 4n

581 The \fB-d\fR option (also \fB--down\fR) means the address is down.
582 .sp
583 The \fB-a\fR option (also \fB--address\fR) specifies the address.
584 The "local" or "remote" prefix can be used for a point-to-point interface.
585 In this case, both addresses must be given.
586 Otherwise, the equal sign ("=") should be omitted and the address should be
587 provided by itself and with no second address.
588 .sp

```

```

590 .RE

592 The \fB-T\fR dhcp option causes the address to be obtained via DHCP.
593 This takes the following options:
594 .RS 4n

596 The \fB-w\fR option (also \fB--wait\fR) gives the time, in seconds,
597 that the command should wait to obtain an address.
598 .sp

600 .RE

602 The \fB-T\fR addrconf option creates an auto-configured address.
603 This takes the following options:
604 .RS 4n

606 The \fB-i\fR option (also \fB--interface-id\fR) gives the interface ID to
607 be used.
608 .sp
609 The \fB-p\fR option (also \fB--prop\fR) indicates which method of
610 auto-configuration should be used.
611 .sp

613 .RE
614 .RE

616 .sp
617 .ne 2
618 .na
619 \fB\fBdown-addr\fR [\fB-t\fR] \fIaddrobj\fR\fR
620 .ad
621 .sp .6
622 .RS 4n
623 The \fBdown-addr\fR subcommand is used to down the address. This will
624 stop packets from being sent or received.
625 .sp
626 The \fB-t\fR option (also \fB--temporary\fR) means
627 that the down is temporary and will not be persistent across reboots.
628 .sp

630 .RE

632 .sp
633 .ne 2
634 .na
635 \fB\fBup-addr\fR [\fB-t\fR] \fIaddrobj\fR\fR
636 .ad
637 .sp .6
638 .RS 4n
639 The \fBup-addr\fR subcommand is used to up the address. This will
640 enable packets to be sent and received.
641 .sp
642 The \fB-t\fR option (also \fB--temporary\fR) means
643 that the up is temporary and will not be persistent across reboots.
644 .sp

646 .RE

648 .sp
649 .ne 2
650 .na
651 \fB\fBdisable-addr\fR \fB-t\fR \fIaddrobj\fR\fR
652 .ad
653 .sp .6
654 .RS 4n
655 The \fBdisable-addr\fR subcommand is used to disable the address.

```

```

656 .sp
657 The \fB-t\fR option (also \fB--temporary\fR) means
658 that the disable is temporary and will not be persistent across reboots.
659 .sp

661 .RE

663 .sp
664 .ne 2
665 .na
666 \fB\fBenable-addr\fR \fB-t\fR \fIiaddr\bj\fR\fR
667 .ad
668 .sp .6
669 .RS 4n
670 The \fBenable-addr\fR subcommand is used to enable the address.
671 .sp
672 The \fB-t\fR option (also \fB--temporary\fR) means
673 that the disable is temporary and will not be persistent across reboots.
674 .sp

676 .RE

678 .sp
679 .ne 2
680 .na
681 \fB\fBrefresh-addr\fR [\fB-i\fR] \fIiaddr\bj\fR\fR
682 .ad
683 .sp .6
684 .RS 4n
685 The \fBrefresh-addr\fR subcommand is used to extend the lease for DHCP
686 addresses. It also restarts duplicate address detection for Static addresses.
687 .sp
688 The \fB-i\fR option (also \fB--inform\fR) means
689 that the network configuration will be obtained from DHCP without taking
690 a lease on the address.
691 .sp

693 .RE

695 .sp
696 .ne 2
697 .na
698 \fB\fBdelete-addr\fR [\fB-r\fR] \fIiaddr\bj\fR\fR
699 .ad
700 .sp .6
701 .RS 4n
702 The \fBdelete-addr\fR subcommand deletes the given address.
703 .sp
704 The \fB-r\fR option (also \fB--release\fR) is used for DHCP-assigned
705 addresses to indicate that the address should be released.
706 .sp

708 .RE

710 .sp
711 .ne 2
712 .na
713 \fB\fBshow-addr\fR [[\fB-p\fR] \fB-o\fR \fIfield\fR[,...]] [\fIiaddr\bj\fR]\fR
714 .ad
715 .sp .6
716 .RS 4n
717 The \fBshow-addr\fR subcommand is used to show the current address properties.
718 .sp
719 The \fB-p\fR option (also \fB--parsable\fR) prints
720 the output in a parsable format.
721 .sp

```

```

722 The \fB-o\fR option (also \fB--output\fR) is used
723 to select which fields will be shown. The field value can be one of the
724 following names:
725 .sp
726 .ne 2
727 .na
728 .RS 4n
729 \fBALL\fR
730 .ad
731 .RS 4n
732 Display all fields
733 .RE

735 .sp
736 .ne 2
737 .na
738 \fBADDROBJ\fR
739 .ad
740 .RS 4n
741 The name of the address
742 .RE

744 .sp
745 .ne 2
746 .na
747 \fBTYPE\fR
748 .ad
749 .RS 4n
750 The type of the address. It can be "static", "dhcp" or "addrconf".
751 .RE

753 .sp
754 .ne 2
755 .na
756 \fBSTATE\fR
757 .ad
758 .RS 4n
759 The state of the address. It can be one of the following values:
760 .sp
761 .ne 2
762 .na
763 .RS 4n
764 disabled s see the \fBdisable-addr\fR subcommand
765 .sp
766 down - see the \fBdown-addr\fR subcommand
767 .sp
768 duplicate - the address is a duplicate
769 .sp
770 inaccessible - the interface for this address has failed
771 .sp
772 ok - the address is up
773 .sp
774 tentative - duplicate address detection in progress
775 .RE
776 .RE

778 .sp
779 .ne 2
780 .na
781 \fBCURRENT\fR
782 .ad
783 .RS 4n
784 A set of single character flags indicating the following:
785 .sp
786 .ne 2
787 .na

```

```

788 .RS 4n
789 U - up
790 .br
791 u - unnumbered (matches another local address)
792 .br
793 p - private, not advertised to routing
794 .br
795 t - temporary IPv6 address
796 .br
797 d - deprecated (not used for outgoing packets)
798 .RE
799 .RE

801 .sp
802 .ne 2
803 .na
804 \fBPERSISTENT\fR
805 .ad
806 .RS 4n
807 A set of single character flags showing the configuration which will be used
808 when the address is enabled.
809 .sp
810 .ne 2
811 .na
812 .RS 4n
813 U - up
814 .br
815 p - private, not advertised to routing
816 .br
817 d - deprecated (not used for outgoing packets)
818 .RE
819 .RE

821 .sp
822 .ne 2
823 .na
824 \fBADDR\fR
825 .ad
826 .RS 4n
827 The address
828 .RE
829 .RE

831 .RE

833 .sp
834 .ne 2
835 .na
836 \fB\bset-addrprop\fR [\fB-t\fR] \fB-p\fR \fIprop\fR=<\fIvalue\fR[,...]> \fIaddr
837 .ad
838 .sp .6
839 .RS 4n
840 The \fBset-addrprop\fR subcommand is used to set a property's value(s) on the
841 addrobj.
842 .sp
843 The \fB-t\fR option (also \fB--temporary\fR) means
844 that the setting is temporary and will not be persistent across reboots.
845 .sp
846 The \fB-p\fR option (also \fB--prop\fR) specifies the property name and
847 value(s). The property name can be one of the following:
848 .sp
849 .ne 2
850 .na

852 .RS 4n

```

```

854 \fBbroadcast\fR
855 .ad
856 .RS 4n
857 The broadcast address (read-only)
858 .RE

860 .sp
861 .ne 2
862 .na
863 \fBdeprecated\fR
864 .ad
865 .RS 4n
866 The address should not be used to send packets but can still receive packets.
867 Can be "on" or "off".
868 .RE

870 .sp
871 .ne 2
872 .na
873 \fBprefixlen\fR
874 .ad
875 .RS 4n
876 The number of bits in the IPv4 netmask or IPv6 prefix.
877 .RE
878 .br
879 .sp
880 .ne 2
881 .na
882 \fBprivate\fR
883 .ad
884 .RS 4n
885 The address is not advertised to routing.
886 Can be "on" or "off".
887 .RE
888 .br
889 .sp
890 .ne 2
891 .na
892 \fBtransmit\fR
893 .ad
894 .RS 4n
895 Packets can be transmitted.
896 Can be "on" or "off".
897 .RE
898 .br
899 .sp
900 .ne 2
901 .na
902 \fBzone\fR
903 .ad
904 .RS 4n
905 The zone the addrobj is in.
906 .RE
907 .br
908 .RE
909 .RE

911 .sp
912 .ne 2
913 .na
914 \fB\breset-addrprop\fR [\fB-t\fR] \fB-p\fR \fIprop\fR \fIaddrobj\fR\fR
915 .ad
916 .sp .6
917 .RS 4n
918 The \fBreset-addrprop\fR subcommand is used to reset an addrobj's property
919 value to the default.

```

```

920 .sp
921 The \fB-t\fR option (also \fB--temporary\fR) means
922 that the disable is temporary and will not be persistent across reboots.
923 .sp
924 The \fB-p\fR option (also \fB--prop\fR) specifies the property name.
925 See the \fBset-addrprop\fR subcommand for the list of property names.
926 .sp

928 .RE

930 .sp
931 .ne 2
932 .na
933 \fB\fBshow-addrprop\fR [[\fB-c\fR]\fB-o\fR \fIfield\fR[,...]] [\fB-p\fR \fIprop\
934 .ad
935 .sp .6
936 .RS 4n
937 The \fBshow-addrprop\fR subcommand is used to display the property values
938 for one or all of the addrobjects.
939 .sp
940 The \fB-c\fR option (also \fB--parsable\fR) prints
941 the output in a parsable format.
942 .sp
943 The \fB-o\fR option (also \fB--output\fR) is used
944 to select which fields will be shown. The field value can be one of the
945 following names:
946 .sp
947 .ne 2
948 .na
949 .RS 4n
950 \fB\fball\fR
951 .ad
952 .RS 4n
953 Display all fields
954 .RE

956 .sp
957 .ne 2
958 .na
959 \fB\fbaddrobject\fR
960 .ad
961 .RS 4n
962 The name of the addrobject
963 .RE

965 .sp
966 .ne 2
967 .na
968 \fB\fbproperty\fR
969 .ad
970 .RS 4n
971 The name of the property
972 .RE

974 .sp
975 .ne 2
976 .na
977 \fB\fbperm\fR
978 .ad
979 .RS 4n
980 If the property is readable ("r") and/or writable ("w").
981 .RE

983 .sp
984 .ne 2
985 .na

```

```

986 \fB\fbcurrent\fR
987 .ad
988 .RS 4n
989 The value of the property
990 .RE

992 .sp
993 .ne 2
994 .na
995 \fB\fbpersistent\fR
996 .ad
997 .RS 4n
998 The persistent value of the property
999 .RE

1001 .sp
1002 .ne 2
1003 .na
1004 \fB\fbdefault\fR
1005 .ad
1006 .RS 4n
1007 The default value of the property
1008 .RE

1010 .sp
1011 .ne 2
1012 .na
1013 \fB\fbpossible\fR
1014 .ad
1015 .RS 4n
1016 The possible values for the property
1017 .RE
1018 .RE

1020 .sp
1021 The \fB-p\fR option (also \fB--prop\fR) is used
1022 to specify which properties to display. See the \fBset-addrprop\fR
1023 subcommand for the list of property names.
1024 .sp

1026 .RE

1028 .sp
1029 .ne 2
1030 .na
1031 \fB\fbset-prop\fR [\fB-t\fR] \fB-p\fR \fIprop\fR[+|-]=<\fIvalue\fR[,...]> \fIpro
1032 .ad
1033 .sp .6
1034 .RS 4n
1035 The \fBset-prop\fR subcommand is used to set a property's value(s) on the
1036 protocol.
1037 .sp
1038 The \fB-t\fR option (also \fB--temporary\fR) means
1039 that the setting is temporary and will not be persistent across reboots.
1040 .sp
1041 The \fB-p\fR option (also \fB--prop\fR) specifies the property name and
1042 value(s). The optional [+|-] syntax can be used to add/remove values from the
1043 current list of values on the property.
1044 The property name can be one of the following:
1045 .sp
1046 .ne 2
1047 .na

1049 .RS 4n

1051 \fB\fbecn\fR

```

```

1052 .ad
1053 .RS 4n
1054 Explicit congestion control (TCP-only)
1055 Can be "never", "passive" or "active".
1056 .RE

1058 \fBextra_priv_ports\fR
1059 .ad
1060 .RS 4n
1061 Additional privileged ports (SCTP, TCP or UDP)
1062 .RE

1064 \fBforwarding\fR
1065 .ad
1066 .RS 4n
1067 Packet forwarding is enabled.
1068 Can be "on" or "off".
1069 .RE

1071 \fBhoplimit\fR
1072 .ad
1073 .RS 4n
1074 The IPv6 hoplimit.
1075 .RE

1077 \fBlargest_anon_port\fR
1078 .ad
1079 .RS 4n
1080 Largest ephemeral port (SCTP, TCP or UDP)
1081 .RE

1083 \fBrecv_maxbuf\fR
1084 .ad
1085 .RS 4n
1086 Receive buffer size (ICMP, SCTP, TCP or UDP)
1087 .RE

1089 \fBsack\fR
1090 .ad
1091 .RS 4n
1092 Selective acknowledgement (TCP).
1093 Can be "active", "passive" or "never".
1094 .RE

1096 \fBsend_maxbuf\fR
1097 .ad
1098 .RS 4n
1099 Send buffer size (ICMP, SCTP, TCP or UDP)
1100 .RE

1102 \fBsmallest_anon_port\fR
1103 .ad
1104 .RS 4n
1105 Smallest ephemeral port (SCTP, TCP or UDP)
1106 .RE

1108 \fBsmallest_nonpriv_port\fR
1109 .ad
1110 .RS 4n
1111 Smallest non-privileged port (SCTP, TCP or UDP)
1112 .RE

1114 \fBttl\fR
1115 .ad
1116 .RS 4n
1117 The IPv4 time-to-live.

```

```

1118 .RE

1120 .RE
1121 .RE

1123 .sp
1124 .ne 2
1125 .na
1126 \fB\fBreset-prop\fR [\fB-t\fR] \fB-p\fR \fIprop\fR \fIprotocol\fR\fR
1127 .ad
1128 .sp .6
1129 .RS 4n
1130 The \fBreset-prop\fR subcommand is used to reset a protocol's property
1131 value to the default.
1132 .sp
1133 The \fB-t\fR option (also \fB--temporary\fR) means
1134 that the disable is temporary and will not be persistent across reboots.
1135 .sp
1136 The \fB-p\fR option (also \fB--prop\fR) specifies the property name.
1137 See the \fBset-prop\fR subcommand for the list of property names.
1138 .sp

1140 .RE

1142 .sp
1143 .ne 2
1144 .na
1145 \fB\fBshow-prop\fR [[\fB-c\fR]\fB-o\fR \fIfield\fR[,...]] [\fB-p\fR \fIprop\fR,
1146 .ad
1147 .sp .6
1148 .RS 4n
1149 The \fBshow-prop\fR subcommand is used to display the property values
1150 for one or all of the protocols.
1151 .sp
1152 The \fB-c\fR option (also \fB--parsable\fR) prints
1153 the output in a parsable format.
1154 .sp
1155 The \fB-o\fR option (also \fB--output\fR) is used
1156 to select which fields will be shown. The field value can be one of the
1157 following names:
1158 .sp
1159 .ne 2
1160 .na
1161 .RS 4n
1162 \fBALL\fR
1163 .ad
1164 .RS 4n
1165 Display all fields
1166 .RE

1168 .sp
1169 .ne 2
1170 .na
1171 \fBPROTO\fR
1172 .ad
1173 .RS 4n
1174 The name of the protocol
1175 .RE

1177 .sp
1178 .ne 2
1179 .na
1180 \fBPROPERTY\fR
1181 .ad
1182 .RS 4n
1183 The name of the property

```

```
1184 .RE
1186 .sp
1187 .ne 2
1188 .na
1189 \fBPERM\fR
1190 .ad
1191 .RS 4n
1192 If the property is readable ("r") and/or writable ("w").
1193 .RE
1195 .sp
1196 .ne 2
1197 .na
1198 \fBCURRENT\fR
1199 .ad
1200 .RS 4n
1201 The value of the property
1202 .RE
1204 .sp
1205 .ne 2
1206 .na
1207 \fBPERSISTENT\fR
1208 .ad
1209 .RS 4n
1210 The persistent value of the property
1211 .RE
1213 .sp
1214 .ne 2
1215 .na
1216 \fBDEFAULT\fR
1217 .ad
1218 .RS 4n
1219 The default value of the property
1220 .RE
1222 .sp
1223 .ne 2
1224 .na
1225 \fBPOSSIBLE\fR
1226 .ad
1227 .RS 4n
1228 The possible values for the property
1229 .RE
1230 .RE
1232 .sp
1233 The \fB-p\fR option (also \fB--prop\fR) is used
1234 to specify which properties to display. See the \fBset-prop\fR
1235 subcommand for the list of property names.
1236 .sp
1238 .RE
1240 .SH SEE ALSO
1241 .sp
1242 .LP
1243 \fBifconfig\fR(1M), \fBdladm\fR(1M), \fBndd\fR(1M), \fBzonecfg\fR(1M),
1244 \fBbarp\fR(1M), \fBcfgadm\fR(1M), \fBif_mpadm\fR(1M), \fBnsswitch.conf\fR(4),
1245 and \fBdhcp\fR(5).
```

```

*****
4418 Sun Oct 28 17:19:48 2012
new/usr/src/pkg/manifests/system-network.mf
815 Need ipadm(1M) manual page
*****
1 #
2 # CDDL HEADER START
3 #
4 # The contents of this file are subject to the terms of the
5 # Common Development and Distribution License (the "License").
6 # You may not use this file except in compliance with the License.
7 #
8 # You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
9 # or http://www.opensolaris.org/os/licensing.
10 # See the License for the specific language governing permissions
11 # and limitations under the License.
12 #
13 # When distributing Covered Code, include this CDDL HEADER in each
14 # file and include the License file at usr/src/OPENSOLARIS.LICENSE.
15 # If applicable, add the following below this CDDL HEADER, with the
16 # fields enclosed by brackets "[]" replaced with your own identifying
17 # information: Portions Copyright [yyyy] [name of copyright owner]
18 #
19 # CDDL HEADER END
20 #

22 #
23 # Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
24 #

26 set name=pkg.fmri value=pkg:/system/network@$(PKGVERS)
27 set name=pkg.description \
28   value="core software for network infrastructure configuration"
29 set name=pkg.summary value="Core Solaris Network Infrastructure"
30 set name=info.classification value=org.opensolaris.category.2008:System/Core
31 set name=variant.arch value=$(ARCH)
32 dir path=etc group=sys
33 dir path=etc/default group=sys
34 dir path=etc/dladm group=netadm owner=dladm
35 dir path=etc/inet group=sys
36 dir path=etc/inet/ike group=sys
37 dir path=etc/inet/ike/crls group=sys
38 dir path=etc/inet/ike/publickeys group=sys
39 dir path=etc/inet/secret group=sys mode=0700
40 dir path=etc/inet/secret/ike.privatekeys group=sys mode=0700
41 dir path=etc/ipadm group=netadm owner=netadm
42 dir path=etc/nwam group=netadm owner=netadm
43 dir path=etc/nwam/loc group=netadm owner=netadm
44 dir path=etc/nwam/loc/NoNet group=netadm owner=netadm
45 dir path=sbin group=sys
46 dir path=usr/share/man
47 dir path=usr/share/man/man1m
48 file path=etc/default/dhccpagent group=sys \
49   original_name=SUNWcnetr:etc/default/dhccpagent preserve=true
50 file path=etc/default/inetinit group=sys \
51   original_name=SUNWcnetr:etc/default/inetinit preserve=true
52 file path=etc/default/ipsec group=sys \
53   original_name=SUNWcnetr:etc/default/ipsec preserve=true
54 file path=etc/default/mpathd group=sys \
55   original_name=SUNWcnetr:etc/default/mpathd preserve=true
56 file path=etc/dladm/datalink.conf group=netadm \
57   original_name=SUNWcnetr:etc/dladm/datalink.conf owner=dladm preserve=true
58 file path=etc/dladm/flowadm.conf group=netadm \
59   original_name=SUNWcnetr:etc/dladm/flowadm.conf owner=dladm preserve=true
60 file path=etc/dladm/flowprop.conf group=netadm \
61   original_name=SUNWcnetr:etc/dladm/flowprop.conf owner=dladm preserve=true

```

```

62 file path=etc/dladm/secobj.conf group=netadm mode=0660 \
63   original_name=SUNWcnetr:etc/dladm/secobj.conf owner=dladm preserve=true
64 file path=etc/inet/datemsr.ndpd group=sys mode=0444
65 file path=etc/inet/ike/config.sample group=sys mode=0444
66 file path=etc/inet/ipsecalg group=sys \
67   original_name=SUNWcnetr:etc/inet/ipsecalg preserve=true
68 file path=etc/inet/ipsecinit.sample group=sys mode=0444
69 file path=etc/inet/secret/ike.preshared group=sys mode=0600 \
70   original_name=SUNWcnetr:etc/inet/secret/ike.preshared preserve=true
71 file path=etc/inet/secret/ipseckeys.sample group=sys mode=0600
72 file path=etc/ipadm/ipadm.conf group=netadm owner=netadm preserve=true
73 file path=etc/nwam/loc/NoNet/ipf.conf.dfl group=netadm owner=netadm \
74   preserve=true
75 file path=etc/nwam/loc/NoNet/ipf6.conf.dfl group=netadm owner=netadm \
76   preserve=true
77 file path=etc/nwam/loc/create_loc_auto group=netadm owner=netadm preserve=true
78 file path=etc/nwam/loc/create_loc_nonet group=netadm owner=netadm \
79   preserve=true
80 file path=sbin/dladm mode=0555
81 file path=sbin/dlstat mode=0555
82 file path=sbin/flowadm mode=0555
83 file path=sbin/flowstat mode=0555
84 file path=sbin/ipadm mode=0555
85 file path=usr/share/man/man1m/dladm.1m
86 file path=usr/share/man/man1m/flowadm.1m
87 file path=usr/share/man/man1m/ipadm.1m
88 group groupname=netadm gid=65
89 legacy pkg=SUNWcnetr \
90   desc="core software for network infrastructure configuration" \
91   name="Core Solaris Network Infrastructure (Root)"
92 license cr_Sun license=cr_Sun
93 license lic_CDDL license=lic_CDDL
94 user username=dladm ftpuser=false gcos-field="Datalink Admin" group=netadm \
95   uid=15
96 user username=netadm ftpuser=false gcos-field="Network Admin" group=netadm \
97   uid=16
98 user username=netcfg ftpuser=false gcos-field="Network Configuration Admin" \
99   group=netadm uid=17

```