FreeBSD 5.X 6.X

FreeBSD

FreeBSD 5.X 6.X

FreeBSD

FreeBSD 5.X 6.X FreeBSD 5.0
(<XXX>) /
e-mail FreeBSD documentation project
(http://lists.FreeBSD.org/mailman/listinfo/freebsd-doc) FreeBSD
(http://www.FreeBSD.org/doc/en_US.ISO8859-1/books/faq/index.html)
HTTP HTML (book.html) FreeBSD FTP
(ftp://ftp.FreeBSD.org/pub/FreeBSD/doc/) PostScript® PDF
(http://www.FreeBSD.org/search/search.html)

1.

FreeBSD 4.X-6.X FAQ!

Usenet FAQ FreeBSD ( ) FAQ

FAQ
FreeBSD documentation project (http://lists.FreeBSD.org/mailman/listinfo/freebsd-doc)

1. FreeBSD

FreeBSD Alpha/AXP, AMD64 Intel® EM64T, i386™ IA-64, PC-98, UltraSPARC® UN*X-like U.C. Berkeley 4.4BSD-Lite 4.4BSD-Lite2 U.C. Berkeley William Jolitz i386 Net/2 386BSD 386BSD FreeBSD FreeBSD (http://www.FreeBSD.org/index.html) FreeBSD

FreeBSD ISP

FreeBSD FreeBSD (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/index.html)

2. FreeBSD

FreeBSD ( )

source tree GPL (http://www.FreeBSD.org/copyright/COPYING) LGPL (http://www.FreeBSD.org/copyright/COPYING.LIB) GPL FreeBSD (http://www.FreeBSD.org/copyright/freebsd-license.html) GPL

3. FreeBSD

FreeBSD (http://www.FreeBSD.org/copyright/freebsd-license.html)

4. FreeBSD

FreeBSD

Ports Collection (http://www.freebsd.org/ports/)

FreeBSD

FreeBSD

UNIX® FreeBSD

UNIX FreeBSD

Windows® Mac OS®

UNIX FAQ FreeBSD

(http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/index.html)

5. FreeBSD

FreeBSD

( )

source tree (

)

free

FreeBSD FreeBSD

6. FreeBSD NetBSD, OpenBSD open source BSD

James Howard DaemonNews (http://www.daemonnews.org/) The BSD Family Tree (http://ezine.daemonnews.org/200104/bsd_family.html) *BSD

7. FreeBSD

FreeBSD *5-STABLE* (release) 5.X *6-STABLE* (release) 6.X

5.3 release 4.X *-STABLE* 5.3 5.X *-STABLE* 4.X ( ) "extended support" 5-STABLE legacy 6-STABLE

2006 5 9.1 (ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/i386/9.1-RELEASE) *6-STABLE* 2006 5 8.3 (ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/i386/8.3-RELEASE/) *5-STABLE*

-STABLE *-CURRENT* snapshot
ISP branch release
-CURRENT *-STABLE* *-CURRENT*

Release FreeBSD
FreeBSD-CURRENT FreeBSD-STABLE)

FreeBSD FreeBSD Release Engineering
(http://www.FreeBSD.org/releng/index.html)

8. FreeBSD-CURRENT

FreeBSD-CURRENT (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/cutting-edge.html#CURRENT)
FreeBSD-STABLE
-CURRENT
(http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/index.html)
(http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/cutting-edge.html#CURRENT)

FreeBSD-CURRENT
FreeBSD-CURRENT
-CURRENT mailing list
make world groups

-CURRENT -STABLE snapshot
(http://www.FreeBSD.org/snapshots/) snapshot

-CURRENT -STABLE

(CVS
:)

-CURRENT snapshot
release -STABLE snapshots

snapshots (http://www.FreeBSD.org/snapshots/) -CURRENT snapshot release

snapshots

9. FreeBSD-STABLE

FreeBSD 2.0.5 FreeBSD -STABLE
(http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/current-stable.html#STABLE)
(ISP)
-CURRENT (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/current-stable.html#CURRENT)
2.0 6.2-RELEASE ( )

6-STABLE 6.0-RELEASE (5-STABLE 5.3-RELEASE)
FreeBSD-CURRENT 7-CURRENT

2.2-STABLE 2.2.8 3-STABLE 3.5.1
3.X 5-STABLE

6-STABLE -STABLE 6-STABLE 2006 5
9.1-RELEASE

7-CURRENT FreeBSD -CURRENT
FreeBSD-CURRENT

10. FreeBSD

FreeBSD

1) 4.4.2-RELEASE 4.4-BETA A.B.C A (Major) B
(Minor) C (Patch)
(ALPHA BETA GAMMA) (RELEASE)

2) 6.0-STABLE 7.0-CURRENT FreeBSD -CURRENT
CURRENT 7.0 FreeBSD -STABLE
STABLE 6.0 -STABLE _CURRENT
7.0-CURRENT 7.0-STABLE
8.0-CURRENT -CURRENT -STABLE

-STABLE -RELEASE -STABLE -CURRENT
-STABLE -CURRENT
4.4-STABLE 2001/10/08

3) 5.0-011025-SNAP 4.2-010816-RELENG SNAP RELENG
3.0-970625-SNAP 2001 10 25 5.0 (SNAPshot) CURRENT
4.2-010816-RELENG 2001 8 16 4.2 STABLE
FreeBSD SNAP SNAP
ALPHA->BETA->GAMMA RELEASE RELENG
RELENG RELEASE

SNAP (011025) Build
Number ALPHA BETA RELEASE(RELENG)

11. FreeBSD

Release Engineering Team <`re@FreeBSD.org`> release

-STABLE
FreeBSD

(release) FreeBSD
(http://www.FreeBSD.org/releng/index.html)

( ) (-CURRENT)
snapshots

12. FreeBSD

FreeBSD source tree
9 core team
(http://www.FreeBSD.org/doc/zh_TW.Big5/articles/contributors/article.html#STAFF-CORE)
300 commiters
(http://www.FreeBSD.org/doc/zh_TW.Big5/articles/contributors/article.html#STAFF-COMMITTERS)
FreeBSD source tree

mailing lists

13. FreeBSD

FreeBSD ftp FreeBSD FTP (ftp://ftp.FreeBSD.org/pub/FreeBSD/)

6-STABLE 9.1-RELEASE 9.1-RELEASE
(ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/i386/9.1-RELEASE/)

7-CURRENT Snapshot (ftp://current.FreeBSD.org/pub/FreeBSD/)
-CURRENT

5-STABLE 8.3-RELEASE 8.3-RELEASE
(ftp://ftp.FreeBSD.org/pub/FreeBSD/releases/i386/8.3-RELEASE/)

5.X 6X snapshots (ftp://current.FreeBSD.org/pub/FreeBSD/snapshots/)

FreeBSD CD DVD
(http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/mirrors.html)

14. (Problem Report PR)

PR
(http://www.FreeBSD.org/cgi/query-pr-summary.cgi?query) ( )

send-pr(1) PR
(http://www.FreeBSD.org/send-pr.html)

FreeBSD
(http://www.FreeBSD.org/doc/zh_TW.Big5/articles/problem-reports/article.html)

15. FreeBSD

FreeBSD (http://www.FreeBSD.org) (http://www.FreeBSD.org/docs.html)

2.

1. FreeBSD

FreeBSD http://www.FreeBSD.org/docs.html
FreeBSD manual(man) doc
FAQ (Bibliography) FreeBSD

2. (ASCII) PostScript

FTP FreeBSD FTP
/pub/FreeBSD/doc/ (ftp://ftp.FreeBSD.org/pub/FreeBSD/doc/)

`faq` ( ) `handbook` (FreeBSD)

locale (FreeBSD
`/usr/share/locale`) ( )

Locale	( )
`en_US.ISO8859-1`	(US English)
`de_DE.ISO8859-1`	(German)
`es_ES.ISO8859-1`	(Spanish)
`fr_FR.ISO8859-1`	(French)
`it_IT.ISO8859-15`	(Italian)
`ja_JP.eucJP`	(Japanese EUC)
`ru_RU.KOI8-R`	(Russian KOI8-R)
`zh_CN.GB2312`	(Simplified Chinese GB2312)
`zh_TW.Big5`	(Traditional Chinese Big5)

:

:

`html-split`	
`html`	
`pdb`	Palm Pilot iSilo (http://www.iSilo.com/)

`pdf`	Adobe's PDF
`ps`	PostScript
`rtf`	Microsoft's RTF [a]
`txt`	(ASCII)

:

a. MS Word RTF (**CTRL+A**, **CTRL+END**, **F9**)

1. (`html-split`) tar(1) `.tar` tar
2. `book.` (`book.pdb` `book.html` .. .).

`zip`	Zip FreeBSD zip `chinese/unzip` `archivers/unzip`
`bz2`	BZip2 zip bz2 `archivers/bzip2`

Handbook PostScript BZip2 `handbook/book.ps.bz2`

FreeBSD *(package)*

package FreeBSD pkg_add(1) pkg_delete(1)

package (package) `packages` . . . `.tgz`

FAQ (PDF) package `faq.en_US.ISO8859-1.pdf.tgz`

FAQ (PDF) package `faq.zh_TW.Big5.pdf.tgz`

FAQ

```
# pkg_add ftp://ftp.FreeBSD.org/pub/FreeBSD/doc/packages/faq.zh_TW.Big5.pdf.tgz
```

pkg_info(1)

```
# pkg_info -f faq.zh_TW.Big5.pdf
Information for faq.zh_TW.Big5.pdf:

Packing list:
```

```
        Package name: faq.zh_TW.Big5.pdf
        CWD to /usr/share/doc/zh_TW.Big5/books/faq
File: book.pdf
        CWD to .
File: +COMMENT (ignored)
File: +DESC (ignored)
```

book.pdf /usr/share/doc/zh_TW.Big5/books/faq

package

(split HTML) FAQ (bzip2(1))
doc/en_US.ISO8859-1/books/faq/book.html-split.tar.bz2

```
# fetch ftp://ftp.FreeBSD.org/pub/FreeBSD/doc/en_US.ISO8859-1/books/faq/book.html-split.tar.bz2
# bzip2 -d book.html-split.tar.bz2
# tar xvf book.html-split.tar
```

.html index.html (
)

3. FreeBSD (mailing lists)

FreeBSD (mailing-lists) (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/eresources.html#ERESOURCES-MAIL)

4. FreeBSD (news groups) ?

FreeBSD (newsgroups) (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/eresources-news.html)

5. "pf(4)" "pfctl(8)"

man

1->generl commands

2->system calls and error numbers

3->C libraries

4->devices and device drivers

5->file formats

6->games

7->miscellaneous information pages

8->system maintenance and operation commands

pf(4) `man 4 pf`

6. FreeBSD IRC (Internet Relay Chat)

IRC FreeBSD

EFNet (http://www.efnet.org/index.php) #FreeBSD FreeBSD

irc.chat.org

EFNet (http://www.efnet.org/index.php) #FreeBSDhelp FreeBSD #FreeBSD

DALNET (http://www.dal.net/) #FreeBSD irc.dal.net( ) irc.eu.dal.net( )

DALNET (http://www.dal.net/) #FreeBSDHelp irc.dal.net( ) irc.eu.dal.net( )

UNDERNET (http://www.undernet.org/) #FreeBSD us.undernet.org( ) eu.undernet.org( )

RUSNET (http://www.rusnet.org.ru/) #FreeBSD FreeBSD

freebsd-gnome (http://freenode.net/irc_servers.shtml) #FreeBSD irc.freenode.net Gnome FreeBSD

freenode (http://freenode.net/irc_servers.shtml) #bsdchat irc.freenode.net FreeBSD (UTF-8)

IRC /

7. FreeBSD

DaemonNews FreeBSD BSD Mall (http://www.bsdmall.com/)

FreeBSD Mall BSD FreeBSD Mall (http://www.freebsdmall.com/)

FreeBSD documentation project (http://lists.FreeBSD.org/mailman/listinfo/freebsd-doc)

3.

1. FreeBSD

FreeBSD 4.X image `floppies/kernel.flp` `floppies/mfsroot.flp` image `fdimage` dd(1) FreeBSD 5.3 ( ) `floppies/boot.flp` `floppies/kernX` (X 1 2 `floppies/boot.flp` 3)

distributions (MS-DOS®) distributions

base/ (4.X bin/)

manpages/

compat*/

doc/

src/ssys.*

FreeBSD FreeBSD (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/install.html)

2. image

3.5 (1.44MB) 1474560 bytes image 1474560 bytes

FTP *binary*

FTP client *ascii* client newline(UNIX) Windows client CR-LF(MS-DOS) image image FTP binary

FTP : FTP *binary* binary image

MS-DOS `copy` (GUI) image

`copy` image image (`fdimage` `rawrite`) raw (FreeBSD FreeBSD (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/install.html))

3. FreeBSD

FreeBSD FreeBSD (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/install.html)

4. FreeBSD

FreeBSD 4.X 386 PC (RAM) 5 MB
60 MB FreeBSD (RAM) 16 MB

FreeBSD 5.X 486 PC (RAM) 24 MB
150 MB

FreeBSD MDA ... X11R6 VGA

µÚ 4

5. RAM 4MB FreeBSD

FreeBSD 4.X 5 MB FreeBSD 5.X ( ) 8 MB

5.X FreeBSD 4 MB
4 MB
16 MB FreeBSD

4 MB kernel( )
2 MB FreeBSD (..)

6.

release( )

(release) Release Engineering
(http://www.FreeBSD.org/doc/zh_TW.Big5/articles/releng/article.html)

7. Windows FreeBSD

Windows FreeBSD FreeBSD (boot manager)
Windows FreeBSD FreeBSD Windows Windows
FreeBSD (boot manager)

8. .. Windows (boot manager)

FreeBSD (boot manager)

FreeBSD FTP /pub/FreeBSD/tools/ (ftp://ftp.FreeBSD.org/pub/FreeBSD/doc/)
`bootinst.exe` `boot.bin` binary DOS

```
> bootinst.exe boot.bin
```

(boot manager)

FreeBSD Custom installation( ) Partition
(boot manager) ( ) partition editor
W Write Boot Manager

Boot Manager (boot manager)

FreeBSD Fixit Fixit (live filesystem) fixit shell

```
Fixit# fdisk -B -b /boot/boot0 bootdevice
```

bootdevice `ad0` (IDE) `ad4` (first IDE disk on auxiliary controller), `da0` (SCSI)

9. IBM Thinkpad A T X FreeBSD

(2000 ~ 2001) IBM BIOS (bug) FreeBSD FAT BIOS FreeBSD

IBM [1] /BIOS

	BIOS
T20	IYET49WW( )
T21	KZET22WW( )
A20p	IVET62WW( )
A20m	IWET54WW( )
A21p	KYET27WW( )
A21m	KXET24WW( )
A21e	KUET30WW

IBM BIOS Jacques Vidrine FreeBSD laptop computer (http://lists.FreeBSD.org/mailman/listinfo/freebsd-mobile) (http://docs.FreeBSD.org/cgi/mid.cgi?20010427133759.A71732) IBM FreeBSD (BIOS)

BIOS BIOS FreeBSD ID

First, you will need to restore the machine to a state where it can get through its self-test screen. Doing this requires powering up the machine without letting it find a FreeBSD partition on its primary disk. One way is to remove the hard disk and temporarily move it to an older ThinkPad (such as a ThinkPad 600) or a desktop PC with an appropriate conversion cable. Once it is there, you can delete the FreeBSD partition and move the hard disk back. The ThinkPad should now be in a bootable state again.

With the machine functional again, you can use the workaround procedure described here to get a working FreeBSD installation.

1. http://people.FreeBSD.org/~bmah/ThinkPad/ `boot1` `boot2`

2. FreeBSD ThinkPad `Dangerously Dedicated`

3. shell (**ALT+F4**) Emergency Holographic Shell fixit shell

4. fdisk(8) FreeBSD ID `165` `166`(OpenBSD ID)
5. `boot1` `boot2`
6. disklabel(8) `boot1` `boot2` FreeBSD slice

   ```
   # disklabel -B -b boot1 -s boot2 ad0sn
   ```

   n FreeBSD slice *n* slice
7. boot prompt `OpenBSD` FreeBSD

OpenBSD FreeBSD IBM ThinkPad ...
:p

10. FreeBSD

( ) :(

IDE (
IDE remapping remapping
)

SCSI

11.

1. (
   ...=_="")
2. binary(image) image (
   ASCII binary)
3. Windows 95/98/ME/NT/2000/XP/2003 DOS
   `fdimage` `rawrite`
   GUI DOS shell

   Netscape® image
   FTP client (binary)

12. ...

IDE
Secondary Slave Secondary Master ATAPI
Windows BIOS
BIOS FreeBSD

IDE Master IDE
Slave IDE Master

13. PLIP(Parallel Line IP) FreeBSD

Laplink FreeBSD PLIP (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/network-plip.html)

14. (geometry)

: (geometry) cylinders heads sectors/track C/H/S PC BIOS

SCSI *physical* geometry FreeBSD disk blocks physical geometry physical geometry (geometry) IDE FreeBSD C/H/S block

logical geometry BIOS FreeBSD BIOS BIOS (geometry)

SCSI (geometry) SCSI extended translation(support >1GB (1GB) `N` cylinders 64 heads 32 sectors/track (geometry) `N` (MB) 2GB 2048 cylinders 64 heads 32 sectors/track

SCSI extended translation (MS-DOS) 1GB (geometry) M cylinders 255 heads 63 sectors(64) `M` (MB) 7.844238 2GB 261 cylinders 255 heads 63 sectors

FreeBSD (geometry) DOS (partition) DOS partition editor

`pfdisk.exe` FreeBSD FTP `tools` partition editor

15.

root 1024 cylinders BIOS (PC BIOS FreeBSD)

SCSI `root` (/) 1024MB (extended translation 4096MB) IDE 504MB

16. (disk managers)

FreeBSD Ontrack Disk Manager disk manager

FreeBSD disk manager BIOS
FreeBSD MFM
FreeBSD cylinders

FreeBSD disk manager FreeBSD
slice 1024 cylinders 20MB

17. FreeBSD `Missing Operating System`

FreeBSD DOS (geometry)

18. (boot manager) `F?`

BIOS FreeBSD
(geometry) BIOS cylinder translation (support >1GB(
1 GB) FreeBSD

19. FreeBSD sources

source `base`( )
`sys` (kernel source) source config(8) kernel
src kernel source NFS
kernel-source mount `/usr/src` symbolic link(
ln(1)) `/usr/src`

source FreeBSD

source `sysinstalll` (FreeBSD 5.2 `/stand/sysinstall`)
Configure Distributions src

20. build kernel

build kernel FreeBSD
kernel FreeBSD 4.X( ) FreeBSD
(boot:) `"-c"` flag kernel ISA FreeBSD
5.X( ) "hints"

RAM kernel
kernel

21. DES Blowfish MD5

FreeBSD *MD5* MD5 UNIX *DES*
( ) DES
(sysinstall crypto source) crypto crypto libraries
Blowfish

`/etc/login.conf` passwd_format (crypto) des blf md5 login.conf(5)

22. `Probing Devices...`

IDE Iomega Zip® Jaz®

release FreeBSD

23. `panic: can't mount root`

kernel IDE Jumper Master IDE FreeBSD Secondary IDE kernel IDE ad2 ad0(BIOS) kernel (ad0) mount ad2

1. `Booting kernel in 10 seconds; hit [Enter] to interrupt` **Enter** boot loader

   `set root_disk_unit="disk_number"` *`disk_number`*

   FreeBSD Primary IDE Master 0 Primary IDE Slave 1 Secondary IDE Master 2 Secondary IDE Slave 3

   `boot`

   `/boot/loader.conf.local`

   `root_disk_unit="disk_number"` (*`disk_number`*)

2. FreeBSD Primary IDE

24.

i386 4 GB(gigabytes) FreeBSD 4.9 5.1 pae(4) kernel kernel PAE

```
options PAE
```

FreeBSD pc98 4 GB PAE Alpha Alpha FreeBSD

25.

8TB(2G blocks) 8K block 16TB 1 TB 4TB ( )

block 4K 1G blocks(4TB)

3-1.

block	works	should work
4K	4T-1	>4T
8K	>32G	32T-1
16K	>128G	32T-1
32K	>512G	64T-1
64K	>2048G	128T-1

When the fs block size is 4K, triple indirect blocks work and everything should be limited by the maximum fs block number that can be represented using triple indirect blocks (approx. 1K^3 + 1K^2 + 1K), but everything is limited by a (wrong) limit of 1G-1 on fs block numbers. The limit on fs block numbers should be 2G-1. There are some bugs for fs block numbers near 2G-1, but such block numbers are unreachable when the fs block size is 4K.

block 8K block 2G-1 1G-1 2G-1 block

26. kernel `archsw.readin.failed`

world kernel kernel 4.11 world 4.8 `make buildworld` `make buildkernel` kernel

loader "|" kernel

27. security profiles

A security profile is a set of configuration options that attempts to achieve the desired ratio of security to convenience by enabling and disabling certain programs and other settings. For full details, see the Security Profile (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/install-post.html#SECURITYPROFILE) section of the Handbook's post-install chapter (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/install-post.html).

28. ACPI

ACPI support bootloader

`OK`

`unset acpi_load`

`boot`

1. Keith Frechette e-mail <kfrechet@us.ibm.com>

4.

4.1.

1. FreeBSD

FreeBSD

FreeBSD 9.1 (http://www.FreeBSD.org/releases/9.1R/hardware.html) 8.3 (http://www.FreeBSD.org/releases/8.3R/hardware.html) (http://www.FreeBSD.org/search/#mailinglists)

FreeBSD-mobile FreeBSD-questions (pc98, Alpha)

4.2. CPU

1. FreeBSD x86

FreeBSD Intel x86 and DEC (HP-Compaq) Alpha FreeBSD 5.0 AMD64 Intel EM64T, IA-64 Sparc64® MIPS® PowerPC® FreeBSD MIPS porting (http://lists.FreeBSD.org/mailman/listinfo/freebsd-mips) FreeBSD PowerPC porting (http://lists.FreeBSD.org/mailman/listinfo/freebsd-ppc) FreeBSD non-Intel platforms porting (http://lists.FreeBSD.org/mailman/listinfo/freebsd-platforms)

BSD NetBSD (http://www.netbsd.org/) OpenBSD (http://www.openbsd.org/)

2. FreeBSD CPU (SMP, Symmetric Multiprocessing)

SMP FreeBSD 5.2 kernel(*GENERIC*)

FreeBSD 5.3 release SMP (emt64) SMP FreeBSD 5.4

FreeBSD 4.X kernel SMP options SMP kernel kernel `/sys/i386/conf/LINT`

4.3. DVD

1. FreeBSD

FreeBSD EIDE SCSI (SCSI) Western Digital (MFM RLL ESDI IDE) ESDI (WD1002/3/6/7)

2. SCSI

FreeBSD (9.1 (http://www.FreeBSD.org/releases/9.1R/hardware.html) 8.3 (http://www.FreeBSD.org/releases/8.3R/hardware.html))

3.

FreeBSD SCSI QIC-36 (QIC-02) 8-mm (Exabyte) DAT

8-mm SCSI-2 FreeBSD

4. FreeBSD

FreeBSD ch(4) chio(1) SCSI chio(1)

If you are not using **AMANDA** or some other product that already understands changers, remember that they only know how to move a tape from one point to another, so you need to keep track of which slot a tape is in, and which slot the tape currently in the drive needs to go back to.

5. FreeBSD

SCSI SCSI

- Mitsumi LU002 (8bit), LU005 (16bit) FX001D (16bit 2x Speed)
- Sony CDU 31/33A
- Sound Blaster SCSI
- Matsushita/Panasonic
  - ATAPI IDE CDROMs
  - SCSI SCSI ATAPI

Daemon News FreeBSD Mall FreeBSD (ISO)

6. FreeBSD

FreeBSD ATAPI IDE CD-R CD-RW burncd(8)

FreeBSD SCSI CD-R CD-RW port packag `cdrecord` kernel `pass` (kernel.GENERIC device pass)

7. FreeBSD Iomega Zip ?

FreeBSD SCSI ATAPI(IDE) Iomega Zip SCSI ZIP SCSI ID 5 6 SCSI BIOS ID 0 1 SCSI ID

FreeBSD Parallel Port Zip kernel `scbus0` `da0` `ppbus0` `vp0` (GENERIC kernel `vp0`) Parallel Port Zip `/dev/da0s4`

`mount /dev/da0s4 /mnt` (DOS)`mount_msdos /dev/da0s4 /mnt`

8. FreeBSD Jaz EZ

IDE EZ drive SCSI FreeBSD SCSI

FreeBSD

mount(8) umount(8) (SCSI)camcontrol(8) atacontrol(8) FAQ

4.4.

1. FreeBSD USB

FreeBSD (USB keyboards Enable USB support in `/etc/rc.conf`.

USB AT USB AT `/dev/kbd0` USB `/dev/kbd1` USB `/dev/ukbd0`

console USB console USB

```
# kbdcontrol -k /dev/kbd1 < /dev/ttyv0 > /dev/null
```

USB `/dev/ukbd0`

```
# kbdcontrol -k /dev/ukbd0 < /dev/ttyv0 > /dev/null
```

`/etc/rc.i386`

USB X
USB (Hot-plugging and unplugging) FreeBSD

ukbd(4)

2. bus

FreeBSD (Microsoft Logitech ATI) bus InPort bus
kernel(GENERIC) kernel

```
device mse0 at isa? port 0x23c irq5
```

Bus port address IRQ
mse(4)

3. PS/2 (mouse port keyboard)

PS/2 `psm` kernel(GENERIC)
kernel psm kernel

```
device psm0 at atkbdc? irq 12
```

kernel `psm0` `/dev` `psm0` `root`

```
# cd /dev; sh MAKEDEV psm0
```

: FreeBSD 5.0-RELEASE( ) devfs(5) `/dev`

4. X Window

console (syscons(4)) console
moused(8)

```
# moused -p /dev/xxxx -t yyyy
# vidcontrol -m on
```

`xxxx` *`yyyy`* protocol moused
(serial) protocol protocol
`auto` moused(8) type

PS/2 `moused_enable="YES"` `/etc/rc.conf`
virtual terminals console
`allscreens_flags="-m on"` `/etc/rc.conf`

moused moused X
FAQ X

5. console

moused ( )

moused(8)

6. console

...

X X

7.

PS/2

4.5. serial

1. FreeBSD

FreeBSD

2. FreeBSD PnP(Plug & Play) modem?

modem PnP ID serial PnP ID

1. kernel `controller pnp0` kernel PnP
2. kernel PnP ID `/usr/src/sys/isa/sio.c`(752) `SUP1310` (`sio_ids[]`) kernel modem PnP ID
3. kernel modem

`pnp` PnP kernel

```
pnp 1 0 enable os irq0 3 drq0 0 port0 0x2f8
```

3. FreeBSD Winmodems modem

FreeBSD modem `comms/ltmdm` Lucent LT `comms/mwavem` IBM Thinkpad 600 700 modem

modem FreeBSD

4. Broadcom 43xx (Native driver)

Broadcom

(FCC)

5. FreeBSD multi-port serial

(http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/install.html#INSTALL-MISC)

(AST)

sio(4)

6. serial console boot:

1. kernel `options COMCONSOLE`
2. /boot.config `-P`
3.

`/usr/src/sys/i386/boot/biosboot/README.serial`

4.6.

1. FreeBSD

FreeBSD SoundBlaster® SoundBlaster Pro SoundBlaster 16 Pro Audio Spectrum 16 AdLib Gravis UltraSound sound cards (FreeBSD (http://www.FreeBSD.org/releases/) snd(4)) MPU-401 MIDI Microsoft® Sound System

: SoundBlaster
, SCSI SoundBlaster SCSI SCSI

2. pcm(4)

es1370
/etc/rc.local

```
# mixer pcm 100 vol 100 cd 100
```

4.7. (ACPI ..)

1. FreeBSD

(http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/install.html#INSTALL-MISC)

2. FreeBSD

FreeBSD 4.X( ) APM apm(4)

FreeBSD 5.X( ) ACPI acpi(4)
APM ACPI

3. ACPI

```
hint.acpi.0.disabled="1"
```

/boot/device.hints

4. Micron FreeBSD

Micron BIOS PCI PCI BIOS
FreeBSD

BIOS Plug and Play Operating System

5. 3Com® PCI Micron

BIOS OS PnP

BIOS Plug and Play Operating System

6. (ASUS) K7V

BIOS boot virus protection

7. PCMCIA cbb0: unsupported card type detected.

kernel

```
device cbb
device pccard
device cardbus
```

```
device pcic
device card 1
```

Handbook FreeBSD Kernel
(http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/kernelconfig.html)
kernel

8. Dell^{TM} PowerEdge^{TM} 2850 FreeBSD

(cdsheen) Dell PowerEdge 2850 FreeBSD 6.0
console FreeBSD 5.3 FreeBSD 5.4
Dell Remote Access Controller (DRAC) USB
Keyboard PS/2

1. Single User Mode
2.

   ```
   # fsck -y /
   # mount -u /
   ```

3. `/etc/devd.conf` USB Keyboard #

   ```
   # When a USB keyboard arrives, attach it as the console keyboard.
   #attach 100 {
   #       device-name "ukbd0";
   #       action "kbdcontrol -k /dev/ukbd0 < /dev/console && /etc/rc.d/syscons restart";
   #};
   #detach 100 {
   #       device-name "ukbd0";
   #       action "kbdcontrol -k /dev/kbd0 < /dev/console";
   #};
   ```

4. `exit` Single User Mode

Dell DRAC/BMC ...

5.

1. FreeBSD

The convention for most PC hardware is to use the memory area between 3.5G and 4G for a special purpose (usually for PCI). This address space is used to access PCI hardware. As a result real, physical memory can not appear in that address space.

What happens to the memory that should appear in that location is dependent on your hardware. Unfortunately, some hardware does nothing and the ability to use that last 500M of RAM is entirely lost.

Luckily, most hardware remaps the memory to a higher location so that it can still be used. However, this can cause some confusion if you watch the boot messages.

On a 32 bit version of FreeBSD, the memory appears lost, since it will be remapped above 4G, which a 32 bit kernel is unable to access. In this case, the solution is to build a PAE enabled kernel. See this FAQ entry for more information.

On a 64 bit version of FreeBSD, or when running a PAE-enabled kernel, FreeBSD will correctly detect and remap the memory so it is usable. During boot, however, it may seem as if FreeBSD is detecting more memory than the system really has. This is normal and the available memory will be corrected as the boot process completes.

2. ?

SCSI re-mapping
...
page first device page mode FreeBSD
`root`

```
# scsi -f /dev/rsd0c -m 1 -e -P 3
```

AWRE ARRE 0 1:-

```
AWRE  Auto Write Reallocation Enbld     1
ARRE  Auto Read Reallocation Enbld     1
```

Ted Mittelstaedt `<tedm@toybox.placo.com>`

IDE IDE remapping
IDE

IDE IDE

ESDI RLL MFM PC
BIOS BIOS DOS
FreeBSD BIOS bad144
bad144 wd FreeBSD 4.0 SCSI
bad144

bad144 -
/kernel
BIOS kernel bad144
1024 cylinder 16 head 63 sector bad144 500 MB

bad144 FreeBSD fdisk Bad Block ON
FreeBSD 2.2.7 cylinder 1024

1024 cylinder ESDI ESDI
DOS fdisk ❁ set geometry geometry wd
dangerously dedicated FreeBSD
geometry fdisk geometry
FreeBSD geometry
block

ESDI ESDI DOS format
DOS FreeBSD fdisk DOS blocksize block
number geometry DOS DOS
blocksize cooperative FreeBSD
bad144 Alt-F2
geometry -
DOS format
remapping

3. FreeBSD HP Netserver SCSI

HP Netserver on-board EISA SCSI 11
EISA EISA EISA >= 10 PCI
FreeBSD auto-configuration

kernel ❁ `EISA_SLOTS` 12 :)
Handbook kernel (../../handbook/kernelconfig.html) ❁
kernel

FreeBSD *UserConfig*
visual

```
eisa 12
quit
```

kernel

: HP Netserver `dangerously dedicated`

4. `ed1: timeout`

(interrupt conflict) (IRQ) FreeBSD 2.0.5 IRQ 2.0.5 IRQ -c ed0/de0/..

BNC (device) timeout ( )

NE2000 UTP

5. 3Com 3C509

DOS `3c5x9.exe`

6.

(../handbook/printing-intro-setup.html#PRINTING-PARALLEL-PORT-MODE) (../handbook/printing-intro-setup.html)

7. `Signal 11`

Signal 11 process

1.
2. FreeBSD FAQ -current

FreeBSD

make buildworld compiler `ls.c` `ls.o` make buildworld compiler compiler

debugger gdb

1.

2.

3. SIMM/DIMM

4. BIOS jumper timing RAM wait states BIOS RAM Speed: Turbo BIOS

5. I/O CDROM 250 300

SIG11 FAQ Linux

FreeBSD

FAQ the SIG11 problem FAQ (http://www.bitwizard.nl/sig11/)

8. `Fatal trap 12: page fault in kernel mode` `panic:`

FreeBSD FAQ kernel panics kernel backtrace

9.

ATI Mach 64 `2e8` serial port sio(4) bug feature sio3

1. `-c` kernel
2. `sio0` `sio1` `sio2` `sio3` sio ->
3. exit

`/usr/src/sys/i386/isa/sio.c` `0x2e8` kernel

X Window
XFree86 XFree86 3.3.3 Mach 64
X Server

10. 128 MB RAM FreeBSD 64MB

FreeBSD BIOS 16 bits KByte
65535 KBytes = 64MB .. BIOS 16MB 64MB
RAM FreeBSD

kernel BIOS

BIOS ... kernel

```
options "MAXMEM=n"
```

n KB 128MB RAM `131072`

11. RAM 1GB kmem_map too small panic

FreeBSD kernel RAM
1GB RAM( )
.. kernel
kernel panic

`VM_KMEM_SIZE_MAX` kernel kernel

```
options VM_KMEM_SIZE_MAX=419430400
```

400 MB kernel 400 MB 6GB RAM

12. RAM 1GB `kmem_map too small!` panic

panic network buffer virtual memory (mbuf clusters)
mbuf clusters virtual memory FreeBSD
(http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/configtuning-kernel-limits.html#NMBCLUSTERS)

13. `/kernel: proc: table is full`

FreeBSD kernel process kernel
`MAXUSERS` `MAXUSERS` kernel network buffer (
) (load) `MAXUSERS`
process

`MAXUSERS` FreeBSD /Process
(http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/configtuning-kernel-limits.html#KERN-MAXFILES)
(process)

FreeBSD 4.4 `MAXUSERS` `/boot/loader.conf` `kern.maxusers`
FreeBSD kernel

process sysctl `kern.maxproc` process `kern.maxprocperuid` `kern.maxproc` 1 (1 init(8))

sysctl FreeBSD `/etc/sysctl.conf` `/etc/rc.local`

14. kernel `CMAP busy`

`/var/db/kvm_*.db` (mismatch) panic

single

```
# rm /var/db/kvm_*.db
```

15. `ahc0: brkadrint, Illegal Host Access at seqaddr 0x0`

Ultrastor SCSI conflict

kernel `uha0`

16. `ahc0: illegal cable configuration`

SCSI BIOS AIC7XXX ( ) (external logic) EEPROM "automatic termination" SCSI

17. Sendmail `mail loops back to myself`

sendmail FAQ :-

* "Local configuration error"

553 relay.domain.net config error: mail loops back to myself
554 <user@domain.net>... Local configuration error

MX domain : domain.net relay.domain.net domain.net domain.net /etc/sendmail.cw FEATURE(use_cw_file)) sendmail.cf "Cw domain.net" sendmail FAQ (ftp://rtfm.mit.edu/pub/usenet/news.answers/mail/sendmail-faq) sendmail comp.mail.sendmail (news:comp.mail.sendmail) comp.mail.misc (news:comp.mail.misc) comp.mail.smail (news:comp.mail.smail) comp.answers (news:comp.answers) news.answers (news:news.answers). Email <mail-server@rtfm.mit.edu> `send usenet/news.answers/mail/sendmail-faq` FAQ

18. remote machine

FreeBSD console `cons25`

logging shell TERM `ansi` `sco`

VT100 FreeBSD console **screen** **screen** terminal ❀ session **screen** VT100 TERM `vt100`

terminal database `cons25`

FreeBSD X X Window `xterm` `rxvt` TERM `xterm` `vt100`

19. `calcru: negative time...`

interrupt / bug MTU TCP/IP

process SIGXCPU exceeded cpu time limit

FreeBSD 3.0 1998 11 29 sysctl

```
# sysctl -w kern.timecounter.method=1
```

sysctl 1 kernel `NTIMECOUNTER` `NTIMECOUNTER` 20

20. `pcm0 not found` `pcm1` kernel ❀ `device pcm0`

FreeBSD 3.x PCI `pcm0` device ISA PCI `pcm1`

: kernel device pcm1 pcm1
ISA PCI pcm2 pcm1 not found

PCI make snd1 device snd0

```
# cd /dev
# ./MAKEDEV snd1
```

FreeBSD 4.x *PnP* pcm0 device
ISA

21. FreeBSD 4.X PnP unknown

FreeBSD 4.X *PnP* FreeBSD 3.X PnP

Peter Wemm freebsd-questions mailing list
FreeBSD 4.x []

> The PNP bios preconfigured it [the modem] and left it laying around in port space so [in 3.x] the old-style ISA probes found it there.
>
> Under 4.0 the ISA code is much more PnP-centric. It was possible [in 3.x] for an ISA probe to find a stray device and then for the PNP device id to match and then fail due to resource conflicts. So it disables the programmable cards first so this double probing cannot happen. It also means that it needs to know the PnP id's for supported PnP hardware. Making this more user tweakable is on the TODO list.

PnP id ISA
pnpinfo(8) pnpinfo(8)

```
# pnpinfo
Checking for Plug-n-Play devices...

Card assigned CSN #1
Vendor ID PMC2430 (0x3024a341)  Serial Number 0xffffffff
PnP Version 1.0  Vendor Version 0
Device Description: Pace 56 Voice Internal Plug & Play Modem

Logical Device ID: PMC2430 0x3024a341 #0
        Device supports I/O Range Check
TAG Start DF
    I/O Range 0x3f8 .. 0x3f8  alignment 0x8  len 0x8
        [16-bit addr]
    IRQ: 4  - only one type (true/edge)
```

[more TAG lines elided]

```
TAG End DF
End Tag

Successfully got 31 resources  1 logical fdevs
```

```
-- card select # 0x0001

CSN PMC2430 (0x3024a341)  Serial Number 0xffffffff

Logical device #0
IO:  0x03e8 0x03e8 0x03e8 0x03e8 0x03e8 0x03e8 0x03e8 0x03e8
IRQ 5 0
DMA 4 0
IO range check 0x00 activate 0x01
```

Vendor ID 0x3024a341
PnP id PMC2430 ASCII id
/usr/src/sys/isa/sio.c

sio.c submit PR
patch submit PR .. .. sio.c

```
static struct isa_pnp_id sio_ids[] = {
```

ASCII Vender ID pnpinfo(8)

```
{0x0f804f3f  NULL}      /* OZO800f - Zoom 2812 (56k Modem) */
{0x39804f3f  NULL}      /* OZO8039 - Zoom 56k flex */
{0x3024a341  NULL}      /* PMC2430 - Pace 56 Voice Internal Modem */
{0x1000eb49  NULL}      /* ROK0010 - Rockwell ? */
{0x5002734a  NULL}      /* RSS0250 - 5614Jx3(G) Internal Modem */
```

Vender ID kernel
FreeBSD 3.X sio

22. top systat nlist failed
kernel symbol

kernel userland kernel
installworld symbol table
/usr/src/UPDATING

/boot/loader kernel boot2 boot(8)
/boot/loader kernel symbols

23. ssh telnet

TCP telnet(1) login

server software client IP
FreeBSD Telnet SSH

DNS

Internet ISP

IP hosts(5) named(8) Internet

www.yahoo.com

24. `stray IRQ`

Stray IRQs IRQ

irq

`isa_strayintr()` 5 0

irq 7 PPP ide
irq 15

25. `file: table is full` dmesg

file descriptors Tuning Kernel Limits
(../handbook/configtuning-kernel-limits.html) kern.maxfiles
(../handbook/configtuning-kernel-limits.html#KERN-MAXFILES)

26.

FreeBSD

dmesg(8) `Timecounter` FreeBSD `TSC`

```
# dmesg | grep Timecounter
Timecounter "i8254"  frequency 1193182 Hz
Timecounter "TSC"  frequency 595573479 Hz
```

sysctl(3) `kern.timecounter.hardware`

```
# sysctl kern.timecounter.hardware
kern.timecounter.hardware: TSC
```

BIOS TSC
FreeBSD

`i8254` sysctl(3)
`kern.timecounter.hardware`

```
# sysctl -w kern.timecounter.hardware=i8254
kern.timecounter.hardware: TSC -> i8254
```

/etc/sysctl.conf

```
kern.timecounter.hardware=i8254
```

27. PC card

BSD PC card
inconsistent state pccardd
"(null)""(null)"

PC card

PC card

28. BIOS FreeBSD boot loader Read error

FreeBSD boot loader geometry fdisk
FreeBSD slice

geometry BIOS cylinders heads sectors

sysinstall(8) fdisk **G** geometry

cylinders heads sectors BIOS
/

5000 cylinders 250 sectors 60 sectors **5000/250/60**

enter **W**

29. Boot Manager

sysinstall(8) Configure Fdisk Boot Manager **W**
boot loader Boot Manager

30. swap_pager: indefinite wait buffer:

page memory 20

/var/log/messages dmesg

31. buildworld/installworld `touch: not found` ?

touch(1)
CMOS-clock (GMT +08:00 CST
) single user `adjkerntz -i` kernel clock
CMOS-clock

6.

: :) FreeBSD (FreeBSD FreeBSD) Vendors page (../../../../commercial/index.html)

1. FreeBSD Office

OpenOffice (http://www.openoffice.org) open-source office FreeBSD Linux® StarOffice (http://www.sun.com/staroffice/), closed-source OpenOffice FreeBSD

FreeBSD (Spreadsheet) Ports Collection

2. FreeBSD Motif

Open Group Motif 2.1.30 `open-motif` package ports handbook ports (../handbook/ports.html)

: Open Motif open source (http://www.opensource.org/)

Motif Motif closed-source Apps2go FreeBSD (i386 alpha) ELF Motif 2.1.20

runtime

OSF/Motif manager, xmbind, panner, wsm.

Development kit with uil, mrm, xm, xmcxx, include and Imake files.

Static and dynamic ELF libraries (for use with FreeBSD 3.0 and above).

Demonstration applets.

Apps2go NetBSD OpenBSD FreeBSD Motif FTP

Apps2go WWW page (http://www.apps2go.com/)

`<sales@apps2go.com>` `<support@apps2go.com>`

phone (817) 431 8775 or +1 817 431-8775

Xi Graphics FreeBSD a.out Motif 2.0

OSF/Motif manager, xmbind, panner, wsm.

Development kit with uil, mrm, xm, xmcxx, include and Imake files.

Static and dynamic libraries (for use with FreeBSD 2.2.8 and earlier).

Demonstration applets.

Preformatted man pages.

Motif FreeBSD *Xi Graphics* BSDI Linux Motif CDE

3. FreeBSD CDE

Xi Graphics FreeBSD CDE

KDE (http://www.kde.org/) open source CDE xfce (http://www.xfce.org/) KDE xfce ports (http://www.FreeBSD.org/ports/index.html)

4. X servers

Xi Graphics (http://www.xig.com/) FreeBSD Intel X

Xi Graphics X Server Binary FreeBSD Linux Xi Graphics X Server

5.0 demo

Xi Graphics FreeBSD Motif CDE

Xi Graphics WWW page (http://www.xig.com/)

`<sales@xig.com>` `<support@xig.com>`

phone (800) 946 7433 or +1 303 298-7478.

5. FreeBSD

! FreeBSD (../../../../commercial/software_bycat.html#CATEGORY_DATABASE)

ports Databases (../../../../ports/databases.html)

6. FreeBSD Oracle

FreeBSD Linux Oracle

http://www.scc.nl/~marcel/howto-oracle.html (http://www.scc.nl/~marcel/howto-oracle.html)

http://www.lf.net/lf/pi/oracle/install-linux-oracle-on-freebsd (http://www.lf.net/lf/pi/oracle/install-linux-oracle-on-freebsd)

7.

1. .. ?

ports (../../../../ports/index.html) port FreeBSD 24,000 port FreeBSD
freebsd-announce mailing list

ports 4.X 5.X 6.X FreeBSD release
ports tree release cd ports/

package gzip binary
package
‘ /

sysinstall(FreeBSD 5.2 /stand/sysinstall) post-configuration
package package pkg_add(1) Package
.tgz .tbz FreeBSD CDROM packages/All
FreeBSD

4.X-RELEASE/4-STABLE

ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/i386/packages-4-stable/
(ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/i386/packages-4-stable/)

5.X-RELEASE/5-STABLE

ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/i386/packages-5-stable
(ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/i386/packages-5-stable/)

6.X-RELEASE/6-STABLE

ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/i386/packages-6-stable
(ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/i386/packages-6-stable/)

7-CURRENT

ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/i386/packages-7-current
(ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/i386/packages-7-current/)

mirror

port port package
ftp.FreeBSD.org (ftp://ftp.FreeBSD.org/pub/FreeBSD/) package

2. INN(Internet News) news server

package port news/inn Dave Barr’s INN Page (http://www.visi.com/~barr/INN.html)
INN INN FAQ

3. FreeBSD Java^{TM}

http://www.FreeBSD.org/java/ (http://www.FreeBSD.org/java/index.html)

4. libc.so.3.0?

2.1.x 2.2/3.x/4.0
port/package

5. ?`Error: can't find libc.so.4.0`?

4.X 5.X package 2.X 3.X
package

6. ghostscript 386/486SX

kernel
kernel

```
options GPL_MATH_EMULATE
```

: `MATH_EMULATE`

7. SCO/iBCS2 `socksys` (FreeBSD 3.0
)

`/etc/sysconfig` (`/etc/rc.conf`, rc.conf(5))
`YES`

```
# Set to YES if you want ibcs2 (SCO) emulation
          loaded at startup ibcs2=NO
```

ibcs2 kernel

/compat/ibcs2/dev

```
lrwxr-xr-x  1 root  wheel         9 Oct 15 22:20 X0R@ -> /dev/null
lrwxr-xr-x  1 root  wheel         7 Oct 15 22:20 nfsd@ -> socksys
-rw-rw-r--  1 root  wheel         0 Oct 28 12:02 null
lrwxr-xr-x  1 root  wheel         9 Oct 15 22:20 socksys@ -> /dev/null
crw-rw-rw-  1 root  wheel   41,   1 Oct 15 22:14 spx
```

socksys `/dev/null` (null(4)) open & close -current ibcs2
`spx`
kernel `SPX_HACK`

8. Microsoft FrontPage

Use the Port, Luke ports tree FrontPage Apache

9. 3.X-STABLE port?

FreeBSD -CURRENT -STABLE ports
http://www.FreeBSD.org/ports/ (../../../../ports/index.html)
-CURRENT -STABLE ports -CURRENT
-STABLE send-pr(1)

10. ld.so

a.out a.out Netscape Navigator ELF
FreeBSD a.out `/usr/libexec/ld.so`
a.out (compat22) sysinstall(8)
FreeBSD

```
# cd /usr/src/lib/compat/compat22# make install clean
```

`make world` compat22 `/etc/make.conf`
`COMPAT22=YES`

3.1-RELEASE 3.2-RELEASE (ERRATA)

11. ports ?

FreeBSD ports

pkg_version(1) ports tree script

```
# pkg_version -c > /tmp/myscript
```

script pkg_version(1) script exit(1)

script ports

ports

`portupgrade` `portupgrade`
package `sysutils/portupgrade` Ruby
FreeBSD

periodic(8)
`/etc/periodic.conf` `weekly_status_pkg_enable="YES"`

12. `/bin/sh` FreeBSD `bash` shell

POSIX shell

shell script POSIX shell
script Bourne shell
(make(1) , system(3), popen(3), Perl Tcl) Bourne shell
Bourne Shell

/bin/sh
shell ports shell bash, scsh, tcsh zsh (
shell ps -u VSZ RSS
)

13. Netscape Opera

DNS Netscape Opera DNS DNS

8. kernel

1. kernel

kernel (../../handbook/kernelconfig.html)

: `kernel.YYMMDD` `/module` `/modules.YYMMDD` `kernel.GENERIC` GENERIC kernel

2. `_hw_float`

`npx0` (npx(4)) kernel :-) `npx0` `npx0`

3. kernel (10MB)

kernel kernel kernel FreeBSD 3.0 panic kernel kernel

kernel

```
makeoptions DEBUGS=-g
```

config(8) `-g`

kernel kernel kernel kernel 1.5MB 2MB

4. multi-port serial code

multi-port serial code

FreeBSD kernel port IRQ

```
#
# Multiport high-speed serial line - 16550 UARTS
#
device sio2 at isa? port 0x2a0 tty irq 5 flags 0x501 vector siointr
device sio3 at isa? port 0x2a8 tty flags 0x501 vector siointr
device sio4 at isa? port 0x2b0 tty flags 0x501 vector siointr
device sio5 at isa? port 0x2b8 tty flags 0x501 vector siointr
```

5. kernel GENERIC

make buildkernel make installkernel
(4.0-RELEASE 4.3-RELEASE)
/usr/src/UPDATING COMMON
ITEMS

make buildkernel make installkernel make buildworld
make buildkernel make buildworld

FreeBSD-STABLE
FreeBSD-STABLE RELEASE

9. Boot Loader

1.

This is highly recommended if you have been tracking -STABLE for more than one release, or have updated a release instead of installing a new one. You can install booteasy on both disks with boot0cfg(8), and dual boot them until you are happy with the new configuration. Skip the next paragraph to find out how to move the data after doing this.

Should you decide not to do a fresh install, you need to partition and label the new disk with either `sysinstall`(FreeBSD 5.2 `/stand/sysinstall`), or fdisk(8) and disklabel(8). You should also install booteasy on both disks with boot0cfg(8), so that you can dual boot to the old or new system after the copying is done.

Now you have the new disk set up, and are ready to move the data. Unfortunately, you cannot just blindly copy the data. Things like device files (in `/dev`), flags, and links tend to screw that up. You need to use tools that understand these things, which means dump(8). Although it is suggested that you move the data in single user mode, it is not required.

You should never use anything but dump(8) and restore(8) to move the root filesystem. The tar(1) command may work - then again, it may not. You should also use dump(8) and restore(8) if you are moving a single partition to another empty partition. The sequence of steps to use dump to move a partitions data to a new partition is:

1. newfs the new partition.
2. mount it on a temporary mount point.
3. cd to that directory.
4. dump the old partition, piping output to the new one.

For example, if you are going to move root to `/dev/ad1s1a`, with `/mnt` as the temporary mount point, it is:

```
# newfs /dev/ad1s1a
# mount /dev/ad1s1a /mnt
# cd /mnt
# dump 0af - / | restore xf -
```

Rearranging your partitions with dump takes a bit more work. To merge a partition like `/var` into its parent, create the new partition large enough for both, move the parent partition as described above, then move the child partition into the empty directory that the first move created:

```
# newfs /dev/ad1s1a
# mount /dev/ad1s1a /mnt
# cd /mnt
# dump 0af - / | restore xf -
# cd var
# dump 0af - /var | restore xf -
```

To split a directory from its parent, say putting `/var` on its own partition when it was not before, create both partitions, then mount the child partition on the appropriate directory in the temporary mount point, then move the old single partition:

```
# newfs /dev/ad1s1a
# newfs /dev/ad1s1d
# mount /dev/ad1s1a /mnt
# mkdir /mnt/var
# mount /dev/ad1s1d /mnt/var
# cd /mnt
# dump 0af - / | restore xf -
```

You might prefer cpio(1), pax(1), tar(1) to dump(8) for user data. At the time of this writing, these are known to lose file flag information, so use them with caution.

2. Will a dangerously dedicated disk endanger my health?

The installation procedure allows you to chose two different methods in partitioning your hard disk(s). The default way makes it compatible with other operating systems on the same machine, by using fdisk table entries (called slices in FreeBSD), with a FreeBSD slice that employs partitions of its own. Optionally, one can chose to install a boot-selector to switch between the possible operating systems on the disk(s). The alternative uses the entire disk for FreeBSD, and makes no attempt to be compatible with other operating systems.

So why it is called dangerous ? A disk in this mode does not contain what normal PC utilities would consider a valid fdisk table. Depending on how well they have been designed, they might complain at you once they are getting in contact with such a disk, or even worse, they might damage the BSD bootstrap without even asking or notifying you. In addition, the dangerously dedicated disk's layout is known to confuse many BIOSes, including those from AWARD (e.g. as found in HP Netserver and Micronics systems as well as many others) and Symbios/NCR (for the popular 53C8xx range of SCSI controllers). This is not a complete list, there are more. Symptoms of this confusion include the `read error` message printed by the FreeBSD bootstrap when it cannot find itself, as well as system lockups when booting.

Why have this mode at all then? It only saves a few kbytes of disk space, and it can cause real problems for a new installation. Dangerously dedicated mode's origins lie in a desire to avoid one of the most common problems plaguing new FreeBSD installers - matching the BIOS geometry numbers for a disk to the disk itself.

Geometry is an outdated concept, but one still at the heart of the PC's BIOS and its interaction with disks. When the FreeBSD installer creates slices, it has to record the location of these slices on the disk in a fashion that corresponds with the way the BIOS expects to find them. If it gets it wrong, you will not be able to boot.

Dangerously dedicated mode tries to work around this by making the problem simpler. In some cases, it gets it right. But it is meant to be used as a last-ditch alternative - there are better ways to solve the problem 99 times out of 100.

So, how do you avoid the need for DD mode when you are installing? Start by making a note of the geometry that your BIOS claims to be using for your disks. You can arrange to have the kernel print this as it boots by specifying `-v` at the `boot:` prompt, or using `boot -v` in the loader. Just before the installer starts, the kernel will print a list of BIOS geometries. Do not panic - wait for the installer to start and then use scrollback to read the numbers. Typically the BIOS disk units will be in the same order that FreeBSD lists your disks, first IDE, then SCSI.

When you are slicing up your disk, check that the disk geometry displayed in the FDISK screen is correct (ie. it matches the BIOS numbers); if it is wrong, use the **g** key to fix it. You may have to do this if there is absolutely nothing on the disk, or if the disk has been moved from another system. Note that this is only an issue with the disk that you are going to boot from; FreeBSD will sort itself out just fine with any other disks you may have.

Once you have got the BIOS and FreeBSD agreeing about the geometry of the disk, your problems are almost guaranteed to be over, and with no need for DD mode at all. If, however, you are still greeted with the dreaded `read error` message when you try to boot, it is time to cross your fingers and go for it - there is nothing left to lose.

To return a dangerously dedicated disk for normal PC use, there are basically two options. The first is, you write enough NULL bytes over the MBR to make any subsequent installation believe this to be a blank disk. You can do this for example with

```
# dd if=/dev/zero of=/dev/rda0 count=15
```

Alternatively, the undocumented DOS feature

```
C:\> fdisk /mbr
```

will to install a new master boot record as well, thus clobbering the BSD bootstrap.

3. Which partitions can safely use Soft Updates? I have heard that Soft Updates on / can cause problems.

Short answer: you can usually use Soft Updates safely on all partitions.

Long answer: There used to be some concern over using Soft Updates on the root partition. Soft Updates has two characteristics that caused this. First, a Soft Updates partition has a small chance of losing data during a system crash. (The partition will not be corrupted; the data will simply be lost.) Also, Soft Updates can cause temporary space shortages.

When using Soft Updates, the kernel can take up to thirty seconds to actually write changes to the physical disk. If you delete a large file, the file still resides on disk until the kernel actually performs the deletion. This can cause a very simple race condition. Suppose you delete one large file and immediately create another large file. The first large file is not yet actually removed from the physical disk, so the disk might not have enough room for the second large file. You get an error that the partition does not have enough space, although you know perfectly well that you just released a large chunk of space! When you try again mere seconds later, the file creation works as you expect. This has left more than one user scratching his head and doubting his sanity, the FreeBSD filesystem, or both.

If a system should crash after the kernel accepts a chunk of data for writing to disk, but before that data is actually written out, data could be lost or corrupted. This risk is extremely small, but generally manageable. Use of IDE write caching greatly increases this risk; it is strongly recommended that you disable IDE write caching when using Soft Updates.

These issues affect all partitions using Soft Updates. So, what does this mean for the root partition?

Vital information on the root partition changes very rarely. Files such as `/kernel` and the contents of `/etc` only change during system maintenance, or when users change their passwords. If the system crashed during the thirty-second window after such a change is made, it is possible that data could be lost. This risk is negligible for most applications, but you should be aware that it exists. If your system cannot tolerate this much risk, do not use Soft Updates on the root filesystem!

/ is traditionally one of the smallest partitions. By default, FreeBSD puts the `/tmp` directory on /. If you have a busy `/tmp`, you might see intermittent space problems. Symlinking `/tmp` to `/var/tmp` will solve this problem.

4. What is inappropriate about my ccd?

The symptom of this is:

```
# ccdconfig -C
ccdconfig: ioctl (CCDIOCSET): /dev/ccd0c: Inappropriate file type or format
```

This usually happens when you are trying to concatenate the `c` partitions, which default to type `unused`. The ccd driver requires the underlying partition type to be FS_BSDFFS. Edit the disklabel of the disks you are trying to concatenate and change the types of partitions to `4.2BSD`.

5. Why can I not edit the disklabel on my ccd?

The symptom of this is:

```
# disklabel ccd0
(it prints something sensible here, so let us try to edit it)
# disklabel -e ccd0
(edit, save, quit)
disklabel: ioctl DIOCWDINFO: No disk label on disk;
use "disklabel -r" to install initial label
```

This is because the disklabel returned by ccd is actually a fake one that is not really on the disk. You can solve this problem by writing it back explicitly, as in:

```
# disklabel ccd0 > /tmp/disklabel.tmp
# disklabel -Rr ccd0 /tmp/disklabel.tmp
# disklabel -e ccd0
(this will work now)
```

6. Can I mount other foreign filesystems under FreeBSD?

FreeBSD supports a variety of other filesystems.

Digital UNIX

UFS CDROMs can be mounted directly on FreeBSD. Mounting disk partitions from Digital UNIX and other systems that support UFS may be more complex, depending on the details of the disk partitioning for the operating system in question.

Linux

FreeBSD supports `ext2fs` partitions. See mount_ext2fs(8) for more information.

Windows NT®

FreeBSD includes a read-only NTFS driver. For more information, see mount_ntfs(8).

FAT

FreeBSD includes a read-write FAT driver. For more information, see mount_msdosfs(8).

FreeBSD also supports network filesystems such as NFS (see mount_nfs(8)), NetWare (see mount_nwfs(8)), and Microsoft-style SMB filesystems (see mount_smbfs(8)).

7. How do I mount a secondary DOS partition?

The secondary DOS partitions are found after ALL the primary partitions. For example, if you have an E partition as the second DOS partition on the second SCSI drive, you need to create the special files for slice 5 in `/dev`, then mount `/dev/da1s5`:

```
# cd /dev
# sh MAKEDEV da1s5
# mount -t msdos /dev/da1s5 /dos/e
```

: You can omit this step if you are running FreeBSD 5.0-RELEASE or newer with devfs(5) enabled.

8. FreeBSD

FreeBSD 5.0 gbde(8) FreeBSD 6.0 geli(8) `security/cfs` port

9. How can I use the Windows NT loader to boot FreeBSD?

The general idea is that you copy the first sector of your native root FreeBSD partition into a file in the DOS/Windows NT partition. Assuming you name that file something like `c:\bootsect.bsd` (inspired by `c:\bootsect.dos`), you can then edit the `c:\boot.ini` file to come up with something like this:

```
[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(1)\WINDOWS
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Windows NT"
C:\BOOTSECT.BSD="FreeBSD"
C:\="DOS"
```

If FreeBSD is installed on the same disk as the Windows NT boot partition simply copy `/boot/boot1` to `C:\BOOTSECT.BSD`. However, if FreeBSD is installed on a different disk `/boot/boot1` will not work, `/boot/boot0` is needed.

`/boot/boot0` needs to be installed using sysinstall(FreeBSD 5.2 `/stand/sysinstall`) by selecting the FreeBSD boot manager on the screen which asks if you wish to use a boot manager. This is because `/boot/boot0` has the partition table area filled with NULL characters but sysinstall copies the partition table before copying `/boot/boot0` to the MBR.

Do not simply copy `/boot/boot0` instead of `/boot/boot1`; you will overwrite your partition table and render your computer un-bootable!

When the FreeBSD boot manager runs it records the last OS booted by setting the active flag on the partition table entry for that OS and then writes the whole 512-bytes of itself back to the MBR so if you just copy `/boot/boot0` to `C:\BOOTSECT.BSD` then it writes an empty partition table, with the active flag set on one entry, to the MBR.

10. How do I boot FreeBSD and Linux from LILO?

If you have FreeBSD and Linux on the same disk, just follow LILO's installation instructions for booting a non-Linux operating system. Very briefly, these are:

Boot Linux, and add the following lines to `/etc/lilo.conf`:

```
other=/dev/hda2
        table=/dev/hda
        label=FreeBSD
```

(the above assumes that your FreeBSD slice is known to Linux as `/dev/hda2`; tailor to suit your setup). Then, run `lilo` as `root` and you should be done.

If FreeBSD resides on another disk, you need to add `loader=/boot/chain.b` to the LILO entry. For example:

```
other=/dev/dab4
        table=/dev/dab
        loader=/boot/chain.b
        label=FreeBSD
```

In some cases you may need to specify the BIOS drive number to the FreeBSD boot loader to successfully boot off the second disk. For example, if your FreeBSD SCSI disk is probed by BIOS as BIOS disk 1, at the FreeBSD boot loader prompt you need to specify:

```
Boot: 1:da(0,a)/kernel
```

You can configure boot(8) to automatically do this for you at boot time.

The Linux+FreeBSD mini-HOWTO (http://sunsite.unc.edu/LDP/HOWTO/mini/Linux+FreeBSD.html) is a good reference for FreeBSD and Linux interoperability issues.

11. How do I boot FreeBSD and Linux using BootEasy?

Install LILO at the start of your Linux boot partition instead of in the Master Boot Record. You can then boot LILO from BootEasy.

If you are running Windows 95 and Linux this is recommended anyway, to make it simpler to get Linux booting again if you should need to reinstall Windows 95 (which is a Jealous Operating System, and will bear no other Operating Systems in the Master Boot Record).

12. How do I change the boot prompt from `???` to something more meaningful?

You can not do that with the standard boot manager without rewriting it. There are a number of other boot managers in the `sysutils` ports category that provide this functionality.

13. I have a new removable drive, how do I use it?

Whether it is a removable drive like a Zip or an EZ drive (or even a floppy, if you want to use it that way), or a new hard disk, once it is installed and recognized by the system, and you have your cartridge/floppy/whatever slotted in, things are pretty much the same for all devices.

(this section is based on Mark Mayo's ZIP FAQ (http://www.vmunix.com/mark/FreeBSD/ZIP-FAQ.html))

If it is a ZIP drive or a floppy, you have already got a DOS filesystem on it, you can use a command like this:

```
# mount -t msdos /dev/fd0c /floppy
```

if it is a floppy, or this:

```
# mount -t msdos /dev/da2s4 /zip
```

for a ZIP disk with the factory configuration.

For other disks, see how they are laid out using fdisk(8) or sysinstall(8).

The rest of the examples will be for a ZIP drive on da2, the third SCSI disk.

Unless it is a floppy, or a removable you plan on sharing with other people, it is probably a better idea to stick a BSD filesystem on it. You will get long filename support, at least a 2X improvement in performance, and a lot more stability. First, you need to redo the DOS-level partitions/filesystems. You can either use fdisk(8) or `sysinstall`(FreeBSD 5.2 `/stand/sysinstall`), or for a small drive that you do not want to bother with multiple operating system support on, just blow away the whole FAT partition table (slices) and just use the BSD partitioning:

```
# dd if=/dev/zero of=/dev/rda2 count=2
# disklabel -Brw da2 auto
```

You can use disklabel or `sysinstall` to create multiple BSD partitions. You will certainly want to do this if you are adding swap space on a fixed disk, but it is probably irrelevant on a removable drive like a ZIP.

Finally, create a new filesystem, this one is on our ZIP drive using the whole disk:

```
# newfs /dev/rda2c
```

and mount it:

```
# mount /dev/da2c /zip
```

and it is probably a good idea to add a line like this to `/etc/fstab` (see fstab(5)) so you can just type `mount /zip` in the future:

```
/dev/da2c /zip ffs rw,noauto 0 0
```

14. Why do I get `Incorrect super block` when mounting a CDROM?

You have to tell mount(8) the type of the device that you want to mount. This is described in the Handbook section on optical media (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/creating-cds.html), specifically the section Using Data CDs (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/creating-cds.html#MOUNTING-CD).

15. Why do I get `Device not configured` when mounting a CDROM?

This generally means that there is no CDROM in the CDROM drive, or the drive is not visible on the bus. Please see the Using Data CDs (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/creating-cds.html#MOUNTING-CD) section of the Handbook for a detailed discussion of this issue.

16. Why do all non-English characters in filenames show up as ? on my CDs when mounted in FreeBSD?

Your CDROM probably uses the Joliet extension for storing information about files and directories. This is discussed in the Handbook chapter on creating and using CDROMs (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/creating-cds.html), specifically the section on Using Data CDROMs (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/creating-cds.html#MOUNTING-CD).

17. I burned a CD under FreeBSD and now I can not read it under any other operating system. Why?

You most likely burned a raw file to your CD, rather than creating an ISO 9660 filesystem. Take a look at the Handbook chapter on creating CDROMs (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/creating-cds.html), particularly the section on burning raw data CDs (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/creating-cds.html#RAWDATA-CD).

18. How can I create an image of a data CD?

This is discussed in the Handbook section on duplicating data CDs (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/creating-cds.html#IMAGING-CD). For more on working with CDROMs, see the Creating CDs Section (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/creating-cds.html) in the Storage chapter in the Handbook.

19. Why can I not `mount` an audio CD?

If you try to mount an audio CD, you will get an error like `cd9660: /dev/acd0c: Invalid argument`. This is because `mount` only works on filesystems. Audio CDs do not have filesystems; they just have data. You need a program that reads audio CDs, such as the `audio/xmcd` port.

20. How do I `mount` a multi-session CD?

By default, mount(8) will attempt to mount the last data track (session) of a CD. If you would like to load an earlier session, you must use the `-s` command line argument. Please see mount_cd9660(8) for specific examples.

21. How do I let ordinary users mount floppies, CDROMs and other removable media?

Ordinary users can be permitted to mount devices. Here is how:

1. As `root` set the sysctl variable `vfs.usermount` to `1`.

   ```
   # sysctl -w vfs.usermount=1
   ```

2. As `root` assign the appropriate permissions to the block device associated with the removable media.

   For example, to allow users to mount the first floppy drive, use:

   ```
   # chmod 666 /dev/fd0
   ```

   To allow users in the group `operator` to mount the CDROM drive, use:

   ```
   # chgrp operator /dev/acd0c
   # chmod 640 /dev/acd0c
   ```

3. If you are running FreeBSD 5.X or later, you will need to alter `/etc/devfs.conf` to make these changes permanent across reboots.

   As `root`, add the necessary lines to `/etc/devfs.conf`. For example, to allow users to mount the first floppy drive add:

```
# Allow all users to mount the floppy disk.
own       /dev/fd0       root:operator
perm     /dev/fd0        0666
```

To allow users in the group `operator` to mount the CD-ROM drive add:

```
# Allow members of the group operator to mount CD-ROMs.
own       /dev/acd0      root:operator
perm     /dev/acd0       0660
```

4. Finally, add the line `vfs.usermount=1` to the file `/etc/sysctl.conf` so that it is reset at system boot time.

All users can now mount the floppy `/dev/fd0` onto a directory that they own:

```
% mkdir ~/my-mount-point
% mount -t msdos /dev/fd0 ~/my-mount-point
```

Users in group `operator` can now mount the CDROM `/dev/acd0c` onto a directory that they own:

```
% mkdir ~/my-mount-point
% mount -t cd9660 /dev/acd0c ~/my-mount-point
```

Unmounting the device is simple:

```
% umount ~/my-mount-point
```

Enabling `vfs.usermount`, however, has negative security implications. A better way to access MS-DOS formatted media is to use the `emulators/mtools` package in the ports collection.

: The device name used in the previous examples must be changed according to your configuration.

22. The `du` and `df` commands show different amounts of disk space available. What is going on?

You need to understand what `du` and `df` really do. `du` goes through the directory tree, measures how large each file is, and presents the totals. `df` just asks the filesystem how much space it has left. They seem to be the same thing, but a file without a directory entry will affect `df` but not `du`.

When a program is using a file, and you delete the file, the file is not really removed from the filesystem until the program stops using it. The file is immediately deleted from the directory listing, however. You can see this easily enough with a program such as `more`. Assume you have a file large enough that its presence affects the output of `du` and `df`. (Since disks can be so large today, this might be a *very* large file!) If you delete this file while using `more` on it, `more` does not immediately choke and complain that it cannot view the file. The entry is simply removed from the directory so no other program or user can access it. `du` shows that it is gone   it has walked the directory tree and the file is not listed. `df` shows that it is still there, as the filesystem knows that `more` is still using that space. Once you end the `more` session, `du` and `df` will agree.

Note that Soft Updates can delay the freeing of disk space; you might need to wait up to 30 seconds for the change to be visible!

This situation is common on web servers. Many people set up a FreeBSD web server and forget to rotate the log files. The access log fills up `/var`. The new administrator deletes the file, but the system still complains that the partition is full. Stopping and restarting the web server program would free the file, allowing the system to release the disk space. To prevent this from happening, set up newsyslog(8).

23. How can I add more swap space?

In the Configuration and Tuning (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/config-tuning.html) section of the Handbook, you will find a section (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/adding-swap-space.html) describing how to do this.

24. Why does FreeBSD see my disk as smaller than the manufacturer says it is?

Disk manufacturers calculate gigabytes as a billion bytes each, whereas FreeBSD calculates them as 1,073,741,824 bytes each. This explains why, for example, FreeBSD's boot messages will report a disk that supposedly has 80GB as holding 76319MB.

Also note that FreeBSD will (by default) reserve 8% of the disk space.

25. How is it possible for a partition to be more than 100% full?

A portion of each UFS partition (8%, by default) is reserved for use by the operating system and the `root` user. df(1) does not count that space when calculating the `Capacity` column, so it can exceed 100%. Also, you will notice that the `Blocks` column is always greater than the sum of the `Used` and `Avail` columns, usually by a factor of 8%.

For more details, look up the `-m` option in tunefs(8).

10.

1.

2.0.5R 2.2.1R `/etc/sysconfig` `/etc/rc` (rc(8)) `/etc/netstart`

`/etc/sysconfig`

post-2.2.1 3.0 `/etc/sysconfig` rc.conf(5) `/etc/netstart` `/etc/rc.network` `cp /usr/src/etc/rc* /etc`

3.1 `/etc/rc.conf` `/etc/defaults/rc.conf` `/etc/defaults/rc.conf` `/etc/rc.conf`

FreeBSD 3.1 DNS named

```
# echo named_enable="YES" >> /etc/rc.conf
```

FreeBSD 3.1 shell script `/usr/local/etc/rc.d` shell script .sh FreeBSD 3.0 `/etc/rc.local`

`/etc/rc.serial`

`/etc/rc.i386` Intel iBCS2 PC

2.

adduser(8) pw(8)

rmuser(8) pw(8)

3. crontab `root: not found`

crontab (`/etc/crontab`) crontab(1)

```
# crontab /etc/crontab
```

crontab crontab(1) crontab (crontab(5))

crontab `/etc/crontab`

```
# crontab -r
```

`/etc/crontab` cron(8)

shell script
/usr/local/etc/periodic cron periodic(8)

crontab
FreeBSD crontab root crontab root crontab (
crontab) cron(8) root

4. su root you are not in the correct group to su root

su root (superuser) wheel
root superuser
wheel su(1)

su root wheel

5. rc.conf

shell ENTER mount /
mount -a -t ufs

ed(1)

vi(1) emacs(1) export TERM=cons25
termcap(5)

/etc/rc.conf
(kernel)

6.

Handbook Handbook
(../handbook/printing.html)

(host-based driver) FreeBSD
WinPrinters DOS Windows NT 4.0
WinPrinter print/pnm2ppa

7. (keyboard mapping)

Handbook using localization (../handbook/using-localization.html) console setup
(../handbook/using-localization.html#SETTING-CONSOLE)

8. unknown: <PNP0303> can’t assign resources

FreeBSD-CURRENT

(can’t assign resources) ISA PNP

Garrett Wollman <wollman@FreeBSD.org>, 2001 24

9. user quotas

1. kernel quotas ( ) kernel

```
options QUOTA
```

Handbook quotas (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/quotas.html)

2. / quotas
3. quotas

	Quota
/usr	/usr/admin/quotas
/home	/home/admin/quotas

10. FreeBSD System V IPC

FreeBSD System V-style IPC kernel

```
options    SYSVSHM          # enable shared memory
options    SYSVSEM          # enable for semaphores
options    SYSVMSG          # enable for messaging
```

: FreeBSD 3.2 *GENERIC*

11. sendmail UUCP

FreeBSD sendmail UUCP sendmail

/etc/sendmail.cf 8 sendmail m4(1)

/usr/src/usr.sbin/sendmail/cf

full sources sendmail
mount

```
# cd /cdrom/src
# cat scontrib.?? | tar xzf - -C /usr/src contrib/sendmail
```

`cf` `README` m4
UUCP `mailertable` sendmail

`.mc` `/usr/src/usr.sbin/sendmail/cf/cf`
`foo.mc`
`sendmail.cf`

```
# cd /usr/src/usr.sbin/sendmail/cf/cf
# make foo.cf
# cp foo.cf /etc/mail/sendmail.cf
```

`.mc`

```
VERSIONID(`Your version number')
OSTYPE(bsd4.4)

FEATURE(accept_unresolvable_domains)
FEATURE(nocanonify)
FEATURE(mailertable, `hash -o /etc/mail/mailertable')

define(`UUCP_RELAY', your.uucp.relay)
define(`UUCP_MAX_SIZE', 200000)
define(`confDONT_PROBE_INTERFACES')

MAILER(local)
MAILER(smtp)
MAILER(uucp)

Cw    your.alias.host.name
Cw    youruucpnodename.UUCP
```

`accept_unresolvable_domains` `nocanonify` `confDONT_PROBE_INTERFACES`
DNS `UUCP_RELAY`
.UUCP ISP (mail replay)

`/etc/mail/mailertable`

```
#
# makemap hash /etc/mail/mailertable.db < /etc/mail/mailertable
```

`uucp-dom:your.uucp.relay`

```
#
```

```
# makemap hash /etc/mail/mailertable.db < /etc/mail/mailertable
#
horus.interface-business.de   uucp-dom:horus
interface-business.de        uucp-dom:if-bus
interface-business.de         uucp-dom:if-bus
heep.sax.de                  smtp8:%1
horus.UUCP                    uucp-dom:horus
if-bus.UUCP                   uucp-dom:if-bus
                             uucp-dom:
```

UUCP (UUCP neighbor)
SMTP UUCP .UUCP
`uucp-neighbor !recipient` UUCP
UUCP `uucp-dom:` UUCP
`uuname`

DBM mailertable
mailertable

`-bt` sendmail sendmail
address test mode `0`
( ) Control-D

```
% sendmail -bt
ADDRESS TEST MODE (ruleset 3 NOT automatically invoked)
Enter <ruleset> <address>
> 3,0 foo@example.com
canonify           input: foo @ example . com
..
parse            returns: $# uucp-dom $@ your.uucp.relay $: foo < @ example . com . >
> ^D
```

12.

IP sendmail

IP ppp ISP
ISP `example.net` `user`
`bsd.home` ISP `relay.example.net`

Fetchmail
ISP POP3 user-ppp
`/etc/ppp/ppp.linkup`

```
MYADDR:
  !bg su user -c fetchmail
```

sendmail ( )

```
  !bg su user -c "sendmail -q"
```

sendmail mailqueue

bsd.home user bsd.home user
.fetchmailrc

```
poll example.net protocol pop3 fetchall pass MySecret
```

user MySecret

from: **sendmail** user@example.net user@bsd.home
sendmail relay.example.net

.mc

```
VERSIONID('bsd.home.mc version 1.0')
OSTYPE(bsd4.4)dnl
FEATURE(nouucp)dnl
MAILER(local)dnl
MAILER(smtp)dnl
Cwlocalhost
Cwbsd.home
MASQUERADE_AS('example.net')dnl
FEATURE(allmasquerade)dnl
FEATURE(masquerade_envelope)dnl
FEATURE(nocanonify)dnl
FEATURE(nodns)dnl
define('SMART_HOST', 'relay.example.net')
Dmbsd.home
define('confDOMAIN_NAME','bsd.home')dnl
define('confDELIVERY_MODE','deferred')dnl
```

.mc sendmail.cf sendmail.cf
sendmail

13. Sendmail

Sendmail (http://www.sendmail.org/) FreeBSD
(port)

port mail/exim mail/postfix mail/qmail mail/zmailer

Sendmail Qmail archive

14. root

Boot: **boot -s** (3.2-RELEASE
-s) shell ENTER # mount -u /
(mount) / passwd root root exit(1)

15. Control-Alt-Delete

FreeBSD 2.2.7-RELEASE syscons( )
kernel

```
options SC_DISABLE_REBOOT
```

FreeBSD 2.2.5-RELEASE PCVT

```
options PCVT_CTRL_ALT_DEL
```

FreeBSD `boot` `nop`
`/usr/share/syscons/keymaps/us.iso.kbd` `/etc/rc.conf`

16. DOS UNIX

perl

```
% perl -i.bak -npe 's/\r\n/\n/g' file ...
```

file .bak

tr(1)

```
% tr -d '\r' < dos-text-file > unix-file
```

dos-text-file DOS *unix-text-file* perl

17. process

killall(1)

18. su `root` ACL

Kerberos -K
su Kerberos

19. Kerberos

Kerberos release bin distribution CDROM
mount cd(/cdrom)

```
# cd /cdrom/bin
# ./install.sh
```

`/etc/make.conf` "MAKE_KERBEROS" build world.

20.

telnet ssh X screen

1. kernel

```
pseudo-device pty 256
```

2.

```
# cd /dev
# sh MAKEDEV pty{1,2,3,4,5,6,7}
```

256

3. `/etc/ttys` 256

```
ttyqc none network
```

`tty[pqrsPQRS][0-9a-v]`

4. kernel

21. /dev/snd0

`snd` FreeBSD
`mixer` `sequencer` `dsp`

```
# cd /dev
# sh MAKEDEV snd0
```

22. `/etc/rc.conf` `/etc/rc` ?

```
# shutdown now
(Note: without -r or -h)
```

```
# return
# exit
```

23. -STABLE -RC -PRERELEASE

RC Release Candiate
FreeBSD -PRERELEASE (-BETA
-PRERELEASE)

FreeBSD release (3.0-RELEASE 4.0-RELEASE) -CURRENT (3.1-RELEASE 4.2-RELEASE) -STABLE 4.3-RELEASE ( )

4.5-STABLE 4.6-PRERELEASE 4.6-RC RC (4.6-RELEASE) 4.6-STABLE

CVS Release Engineering (../../articles/releng/article.html)

24. chflags

securelevel

FreeBSD securelevel (system flags) securelevel

```
# sysctl kern.securelevel
```

securelevel /etc/rc.conf securelevel init(8) securelevel /etc/defaults/rc.conf rc.conf(5) rc.conf

25.

securelevel 1

securelevel 1 FreeBSD securelevel

```
# sysctl kern.securelevel
```

securelevel /etc/rc.conf init(8) securelevel /etc/defaults/rc.conf rc.conf(5) rc.conf

26. `rpc.statd` 256 megabytes

memory leak 256 Mbyte ( )

top(1) ps(1)

rpc.statd(8) (/var) mmap(2) 0x10000000 IA32 256MB

27. schg

securelevel (0) securelevel FAQ securelevel init(8)

28. FreeBSD `.shosts` SSH

FreeBSD `.shosts` ssh(1) suid `root`

`/etc/make.conf` `ENABLE_SUID_SSH` `true` build ssh (`make world`)

`root` `chmod 4755 /usr/bin/ssh` `/usr/bin/ssh` `4555` `ENABLE_SUID_SSH= true` `/etc/make.conf` `make world`

29. `vnlru`?

`kern.maxvnodes` `vnlru` vnode

11. X Window System Virtual Consoles

1. X

xorgconfig(1) Xorg
xorgcfg(1)
Xaccel server Xi Graphics

2. X, `startx` `KDENABIO failed (Operation not permitted)`

securelevel securelevel X
init(8)

securelevel (`/etc/rc.conf`
) xdm(1) (securelevel)
8. xdm(1)

3. X

syscons (console) FreeBSD virtual console X syscons `/dev/sysmouse`
event moused sysmouse virtual console
X 4. moused

`/etc/XF86Config`

```
Section         Pointer
Protocol        "SysMouse"
Device          "/dev/sysmouse"
....
```

XFree86 3.3.2 *Protocol* *MouseSystems*

X `/dev/mouse`
`/dev/mouse` `/dev/sysmouse` (sysmouse(4)):

```
# cd /dev
# rm -f mouse
# ln -s sysmouse mouse
```

4. X

X client Colas Nahaboo
(http://www.inria.fr/koala/colas/mouse-wheel-scroll/) (http://www.inria.fr/koala/colas/mouse-wheel-scroll/).

imwheel

1. event Translate the Wheel Events

imwheel event event moused(8) X event

a. moused(8) Event

moused(8) event moused(8) `-z 4` `moused -p /dev/psm0` moused(8) `moused -p /dev/psm0 -z 4` `/etc/rc.conf` moused(8) `/etc/rc.conf` `moused_flags` `-z 4`

X `/etc/XF86Config` Pointer `Buttons 5` `/etc/XF86Config` Pointer

11-1. XFree86 3.3.x XF86Config Pointer moused

```
Section "Pointer"
   Protocol        "SysMouse"
   Device          "/dev/sysmouse"
   Buttons         5
EndSection
```

11-2. XFree86 4.x XF86Config InputDevice X Server

```
Section "InputDevice"
   Identifier      "Mouse1"
   Driver          "mouse"
   Option          "Protocol" "auto"
   Option          "Device" "/dev/sysmouse"
   Option          "Buttons" "5"
EndSection
```

11-3. .emacs

```
;; wheel mouse
(global-set-key [mouse-4] 'scroll-down)
(global-set-key [mouse-5] 'scroll-up)
```

b. X Server Event

moused(8) moused(8) event X server event `/etc/XF86Config` IntelliMouse XFree86 MouseMan+ MouseManPlusPS/2 Pointer `Protocol`

X server ZAxisMapping

moused(8) IntelliMouse PS/2
/etc/XF86Config

11-4. XF86Config Pointer X Server

```
Section "Pointer"
   Protocol        "IntelliMouse"
   Device          "/dev/psm0"
   ZAxisMapping    4 5
EndSection
```

11-5. XFree86 4.x XF86Config InputDevice X Server

```
Section "InputDevice"
   Identifier      "Mouse1"
   Driver          "mouse"
   Option          "Protocol" "auto"
   Option          "Device" "/dev/psm0"
   Option          "ZAxisMapping" "4 5"
EndSection
```

11-6. .emacs

```
;; wheel mouse
(global-set-key [mouse-4] 'scroll-down)
(global-set-key [mouse-5] 'scroll-up)
```

2. **imwheel**

   Ports **imwheel** x11 event event **Page Up** **Imwheel** event event **imwheel** /usr/X11R6/etc/imwheelrc ~/.imwheelrc imwheel(1)

3. **Emacs** **Imwheel** ( )

   emacs **Xemacs** ~/.emacs **emacs**

11-7. Imwheel Emacs

```
;;; For imwheel
(setq imwheel-scroll-interval 3)
(defun imwheel-scroll-down-some-lines ()
  (interactive)
```

```
  (scroll-down imwheel-scroll-interval))
(defun imwheel-scroll-up-some-lines ()
  (interactive)
  (scroll-up imwheel-scroll-interval))
(global-set-key [?\M-\C-\)] 'imwheel-scroll-up-some-lines)
(global-set-key [?\M-\C-\(] 'imwheel-scroll-down-some-lines)
;;; end imwheel section
```

Xemacs `~/.emacs`

11-8. Imwheel Xemacs

```
;;; For imwheel
(setq imwheel-scroll-interval 3)
(defun imwheel-scroll-down-some-lines ()
  (interactive)
  (scroll-down imwheel-scroll-interval))
(defun imwheel-scroll-up-some-lines ()
  (interactive)
  (scroll-up imwheel-scroll-interval))
(define-key global-map [(control meta \))] 'imwheel-scroll-up-some-lines)
(define-key global-map [(control meta \()] 'imwheel-scroll-down-some-lines)
;;; end imwheel section
```

4. **Imwheel**

xterm `imwheel`
imwheel `.xinitrc` `.xsession`
imwheel PID Linux **imwheel**

5. X Window

Num Lock

Num Lock `XF86Config`
`Keyboard`

```
# Let the server do the NumLock processing.  This should only be
# required when using pre-R6 clients
    ServerNumLock
```

6. virtual console

virtual console X

login
login name password virtual console ( )

FTP
Alt-F2 (Alt F2)
virtual console login Alt-F1

FreeBSD virtual console (3.3-RELEASE) Alt-F1 Alt-F2
Alt-F3

virtual console `/etc/ttys` (ttys(5)) Virtual terminals `ttyv4` `ttyvc`

```
# Edit the existing entry for ttyv3 in /etc/ttys and change
# "off" to "on".
ttyv3   "/usr/libexec/getty Pc"         cons25  on secure
ttyv4   "/usr/libexec/getty Pc"         cons25  on secure
ttyv5   "/usr/libexec/getty Pc"         cons25  on secure
ttyv6   "/usr/libexec/getty Pc"         cons25  on secure
ttyv7   "/usr/libexec/getty Pc"         cons25  on secure
ttyv8   "/usr/libexec/getty Pc"         cons25  on secure
ttyv9   "/usr/libexec/getty Pc"         cons25  on secure
ttyva   "/usr/libexec/getty Pc"         cons25  on secure
ttyvb   "/usr/libexec/getty Pc"         cons25  on secure
```

virtual terminal 8MB
`secure` `insecure`

: X ( ) virtual terminal
Alt login X -

console 12
terminal X virtual terminal 12

```
ttyvb   "/usr/libexec/getty Pc"         cons25  on  secure
```

```
ttyvb   "/usr/libexec/getty Pc"         cons25  off secure
```

10

```
ttyv9   "/usr/libexec/getty Pc"         cons25  off secure
ttyva   "/usr/libexec/getty Pc"         cons25  off secure
ttyvb   "/usr/libexec/getty Pc"         cons25  off secure
```

( )
`/etc/ttys` virtual terminal

```
# cd /dev
# sh MAKEDEV vty12
```

virtual console ( )
X Window `root`

```
# kill -HUP 1
```

X Window
kill

7. X virtual console

Ctrl+Alt+F*n* virtual console **Ctrl+Alt+F1** virtual console

console **Alt+F*n*** console

X X virtual console X (`startx`) X virtual console console virtual terminal X **Alt+F9** X

8. XDM

xdm (http://www.FreeBSD.org/cgi/man.cgi?manpath=xfree86&query=xdm) `/etc/ttys` ttys(5) `rc.local` (rc(8)) xdm `/usr/local/etc/rc.d` `X.sh` X login:

ttys X vty logout X server init rc.local X xdm

rc.local `xdm` daemon getty xdm getty xdm console script sleep 10 xdm

`/etc/ttys` `xdm` `xdm` getty(8) `/usr/X11R6/lib/X11/xdm/Xservers` `vt`

```
:0 local /usr/X11R6/bin/X vt4
```

X server `/dev/ttyv3` X server vty FreeBSD vty

9. xconsole `Couldn't open console`

`startx` `startx` `/dev/console` `xterm -C xconsole`

console
console VTY login fbtab(5)

`/etc/fbtab` (fbtab(5))

```
/dev/ttyv0 0600 /dev/console
```

`/dev/ttyv0` console

10. XFree86 `root`

X server `root` XFree86 (<= 3.3.6) server `root` (setuid `root`) X server XFree86 server setuid `root`

X server `root` X `xdm` display manager (`kdm`) `Xwrapper`

`xdm` daemon getty(8) login(1) `xdm` XFree86 (http://www.xfree86.org/support.html) FAQ

`Xwrapper` X server X server X server display manager ports `/usr/ports/x11/wrapper`

11. PS/2 X

2.2.5 X virtual terminal kernel

```
options PSM_CHECKSYNC
```

kernel

PS/2 ALPS GlidePoint tap

2.2.6 PS/2 GlidePoint (PSM_CHECKSYNC)

```
psmintr: out of sync (xxxx != yyyy)
```

PS/2 flag 0x100 `-c` *UserConfig*

```
boot: -c
```

UserConfig

```
UserConfig> flags psm0 0x100
UserConfig> quit
```

12. PS/2 MouseSystem

PS/2

2.0.X 2.1.X 2.2 2.2.5 `/sys/i386/isa/psm.c` patch kernel

```
@@ -766,6 +766,8 @@
     if (verbose >= 2)
         log(LOG_DEBUG, "psm%d: SET_DEFAULTS return code:%04x\n",
             unit, i);
+    set_mouse_resolution(sc->kbdc, PSMD_RES_HIGH);
+
 #if 0
     set_mouse_scaling(sc->kbdc);        /* 1:1 scaling */
     set_mouse_mode(sc->kbdc);                /* stream mode */
```

2.2.6 PS/2 0x04 flag `-c` *UserConfig*

```
boot: -c
```

UserConfig

```
UserConfig> flags psm0 0x04
UserConfig> quit
```

13. X `imake` `Imake.tmpl`

`Imake.tmpl` Imake Imake X `Imake.tmpl` header file X X prog distribution sysinstall(FreeBSD 5.2 `/stand/sysinstall`) X distribution

14. build X XFree86 3.3.x XFree86 4.x

port XFree86 4.x `/etc/make.conf` ( )

```
XFREE86_VERSION=        4
```

15.

`.xinitrc` `.xsession` `xmodmap -e "pointer = 3 2 1"`

16. splash

FreeBSD 3.1 splash
256 (*.BMP) ZSoft PCX(*.PCX) 320x200
VGA kernel VESA
1024x768 VESA VM86 kernel VESA kernel
VESA VESA kld module

FreeBSD splash FreeBSD 3.2
splash

FreeBSD 3.1

3.1 Windows
/boot/splash.bmp /boot/loader.rc

```
load kernel
load -t splash_image_data /boot/splash.bmp
load splash_bmp
autoboot
```

FreeBSD 3.2+

PCX FreeBSD 3.2
FreeBSD 3.1 splash_bmp splash_pcx PCX
/boot/loader.rc

```
include /boot/loader.4th
start
```

/boot/loader.conf

```
splash_bmp_load="YES"
bitmap_load="YES"
```

/boot/splash.bmp splash PCX
/boot/splash.pcx /boot/loader.rc /boot/loader.conf

```
splash_pcx_load="YES"
bitmap_load="YES"
bitmap_name="/boot/splash.pcx"
```

splash http://www.baldwin.cx/splash/

17. X Windows ®

xmodmap(1)

Windows keycode

115 - Windows Ctrl Alt

116 - Windows Alt-Gr

117 - Ctrl

Windows

```
# xmodmap -e "keycode 115 = comma"
```

windows manager

Windows X ~/.xinitrc xmodmap ~/.xmodmaprc xmodmap ~/.xinitrc

```
xmodmap $HOME/.xmodmaprc
```

F13 F14 F15 window manager

~/.xmodmaprc

```
keycode 115 = F13
keycode 116 = F14
keycode 117 = F15
```

fvwm2 F13 ( ) F14 ( ) F15 Workplace (application) ( )

~/.fvwmrc

```
Key F13        FTIWS    A        Iconify
Key F14        FTIWS    A        RaiseLower
Key F15        A        A        Menu Workplace Nop
```

18. OpenGL 3D

3D XFree86 NVIDIA FreeBSD NVIDIA Driver Initiative (http://nvidia.netexplorer.org/) XFree86-4 NVIDIA 3D XFree86-4 Matrox G200/G400, ATI Rage 128/Radeon, 3dfx Voodoo 3, 4, 5, Banshee XFree86-4 Direct Rendering on FreeBSD (http://gladstone.uoregon.edu/~eanholt/dri/) XFree 3.3 Utah-GLX port graphics/utah-glx Matrox Gx00, ATI Rage Pro, SiS 6326, i810, Savage, NVIDIA OpenGL

12. Networking

1. diskless booting

Diskless booting FreeBSD server FreeBSD (../handbook/diskless.html)

2. FreeBSD (router)

FreeBSD advanced networking (../handbook/routing.html) routing and gateways (../handbook/routing.html#DEDICATED-ROUTER)

3. FreeBSD Win95 Internet

FreeBSD Win95 FreeBSD Internet FreeBSD Win95

... FreeBSD 3.x (user-mode) ppp(8) `-nat` `/etc/rc.conf` `-nat` `gateway_enable` *YES* ppp(8) Windows

Steve Sims Pedantic PPP Primer (../ppp-primer/index.html)

(kernel-mode) PPP (Ethernet connection) Internet natd(8) FAQ natd

4. FreeBSD SLIP PPP

man pages slattach(8) sliplogin(8) ppp(8) pppd(8) ppp(8) pppd(8) sliplogin(8) slattach(8)

, FreeBSD PPP SLIP (../handbook/ppp-and-slip.html)

shell account Internet `net/slirp` ( ) ftp http

5. FreeBSD NAT Masquerading

( ) Internet provider IP number (IP number) natd(8) natd(8) IP number internet

ppp(8) `-nat` alias library (libalias(3))

6. FreeBSD (parallel line) PLIP

PLIP section (../handbook/plip.html)

7. `/dev/ed0` device

Berkeley kernel `/etc/rc.network` manual pages BSD FreeBSD SunOS 4.0 Ultrix

8. Ethernet aliases

alias `netmask 0xffffffff` ifconfig(8) command-line

```
# ifconfig ed0 alias 192.0.2.2 netmask 0xffffffff
```

```
# ifconfig ed0 alias 172.16.141.5 netmask 0xffffff00
```

9. 3C503 network port

port ifconfig(8) port `link0` AUI port BNC port `link2` flags ifconfig_* `/etc/rc.conf` (rc.conf(5))

10. FreeBSD NFS

PC NFS

the Handbook entry on NFS (../handbook/nfs.html)

11. NFS-mount Linux

Linux NFS privileged port mount request

```
# mount -o -P linuxbox:/blah /mnt
```

12. NFS-mount Sun

SunOS 4.X Sun privileged port mount request

```
# mount -o -P sunbox:/blah /mnt
```

13. mountd can't change attributes bad exports list
FreeBSD NFS

/etc/exports exports(5) NFS (../handbook/nfs.html) configuring NFS (../handbook/nfs.html#CONFIGURING-NFS)

14. PPP NeXTStep

TCP extensions /etc/rc.conf (rc.conf(5)) NO

```
tcp_extensions=NO
```

Xylogic Annex

15. IP multicast support

FreeBSD 2.0 Multicast host multicast router compile kernel MROUTING mrouted(8) FreeBSD 2.2 mrouted(8) /etc/rc.conf mrouted_enable "YES"

MBONE ports mbone (http://www.FreeBSD.org/ports/mbone.html) vic vat

16. DEC PCI chipset

Glen Foster <gfoster@driver.nsta.org>

12-1. Network cards based on the DEC PCI chipset

Vendor	Model
ASUS	PCI-L101-TB
Accton	ENI1203
Cogent	EM960PCI
Compex	ENET32-PCI
D-Link	DE-530
Dayna	DP1203, DP2100
DEC	DE435, DE450
Danpex	EN-9400P3
JCIS	Condor JC1260
Linksys	EtherPCI
Mylex	LNP101
SMC	EtherPower 10/100 (Model 9332)
SMC	EtherPower (Model 8432)
TopWare	TE-3500P
Znyx (2.2.x)	ZX312, ZX314, ZX342, ZX345, ZX346, ZX348

Vendor	Model
Znyx (3.x)	ZX345Q, ZX346Q, ZX348Q, ZX412Q, ZX414, ZX442, ZX444, ZX474, ZX478, ZX212, ZX214 (10mbps/hd)

17. FQDN

foo.bar.edu

mumble example.org Fully-Qualified Domain Name mumble.example.org mumble

BSD BIND resolver FreeBSD **bind** (see named(8)) FQDN mumble mumble.foo.example.org

mumble.example.org mumble.edu RFC 1535

,

```
search foo.example.org example.org
```

instead of the previous

```
domain foo.example.org
```

/etc/resolv.conf (resolv.conf(5)) RFC 1535 boundary between local and public administration

18. Permission denied

kernel IPFIREWALL 2.1.7R (2.1-STABLE)

I firewall root

```
# ipfw add 65534 allow all from any to any
```

/etc/rc.conf firewall_type="open"

FreeBSD firewall (../handbook/firewalls.html)

19. IPFW

Firewalls (../handbook/firewalls.html) IPFW Overhead & Optimization (../handbook/firewalls.html#IPFW-OVERHEAD)

20. ipfw fwd redirect

(network address translation, NAT) fwd

```
01000 fwd 10.0.0.1 from any to foo 21
```

foo *10.0.0.1* *foo* *10.0.0.1* fwd

natd(8) ports collection (../../../../ports/index.html)

21. (service request)

ports sysutils `socket` FTP socket

```
ftp stream tcp nowait nobody /usr/local/bin/socket socket ftp.example.com ftp
```

ftp.example.com *ftp* port

22.

FreeBSD dummynet(4) FreeBSD (, ipfw(4)); ALTQ (http://www.csl.sony.co.jp/person/kjc/programs.html) Emerging Technologies (http://www.etinc.com/) Bandwidth Manager

23. `/dev/bpf0: device not configured`

(Berkeley Packet Filter) (bpf(4)) kernel kernel kernel

```
pseudo-device bpf          # Berkeley Packet Filter
```

device node `/dev`

```
# sh MAKEDEV bpf0
```

device node Handbook (../handbook/kernelconfig-nodes.html)

24. Windows , Linux smbmount

SMBFS kernel (userland programs) ports `net/smbfs` 4.5-RELEASE

25. icmp-response bandwidth limit 300/200 pps

ICMP TCP (RST) ICMP UDP TCP TCP

(DoS) ( )

( )

kernel kernel sysctl `net.inet.icmp.icmplim` , `300` packets per second

```
# sysctl -w net.inet.icmp.icmplim=300
```

sysctl `net.inet.icmp.icmplim_output`

```
# sysctl -w net.inet.icmp.icmplim_output=0
```

`net.inet.icmp.icmplim` ( ) `0`

26. `arp: unknown hardware address format`

FreeBSD MAC cable modem FreeBSD

27. CVSup

```
/usr/libexec/ld-elf.so.1: Shared object "libXaw.so.6" not found
```

`net/cvsup` **XFree86** **CVSup** GUI **XFree86** **CVSup** `net/cvsup-without-gui` FreeBSD CVSup (http://www.freebsd.org/handbook/cvsup.html)

13.

1. sandbox

Sandbox

userid security named

ntalk /etc/inetd.conf userid root
tty tty sandbox ntalk
userid

chroot

mount
-

sandbox sandbox

UNIX sandbox userid

UNIX
Windows

UNIX userid userid root
Userid

2. securelevel?

securelevel securelevel
superuser (root) securelevel

schg (, the system immutable flag)

/dev/mem /dev/kmem

ipfirewall(4)

securelevel

```
# sysctl kern.securelevel
```

sysctl(8) (kern.securelevel)
securelevel (0) securelevel

securelevel
securelevel (installworld)
/etc/rc.conf securelevel (kern_securelevel kern_securelevel_enable)

securelevel init(8)

securelevel

securelevel
securelevel (
securelevel) securelevel

freebsd-security
(../../../../search/index.html) securelevel

3. BIND (named) 53 (high-numbered port) (Listen)

FreeBSD 3.0 BIND
53
/etc/namedb/named.conf

```
options {
        query-source address * port 53;
};
```

* IP address
sockstat(1)

4. Sendmail 25 587
Sendmail mail submission 587

5. UID 0 toor

toor alternative (toor root) bash(1)
shell
root shell shell (ports shell
package) /usr/local/bin root shell

`/usr/local/bin` `/usr` (`/usr/local/bin`) mount `root` (shell)

`toor` `root` shell `root` shell (single user mode) `toor` `root` `toor`

6. `suidperl`

`suidperl` suid bit suid

```
# chmod u+s /usr/bin/suidperl
```

source `suidperl` suid `/etc/make.conf` `ENABLE_SUIDPERL=true` `make buildworld`

14. PPP

1. I cannot make ppp(8) work. What am I doing wrong?

You should first read the ppp(8) man page and the PPP section of the handbook (../handbook/ppp-and-slip.html#USERPPP). Enable logging with the command

```
set log Phase Chat Connect Carrier lcp ipcp ccp command
```

This command may be typed at the ppp(8) command prompt or it may be entered in the `/etc/ppp/ppp.conf` configuration file (the start of the `default` section is the best place to put it). Make sure that `/etc/syslog.conf` (see syslog.conf(5)) contains the lines

```
!ppp
*.*        /var/log/ppp.log
```

and that the file `/var/log/ppp.log` exists. You can now find out a lot about what is going on from the log file. Do not worry if it does not all make sense. If you need to get help from someone, it may make sense to them.

If your version of ppp(8) does not understand the `set log` command, you should download the latest version (http://people.FreeBSD.org/~brian/). It will build on FreeBSD version 2.1.5 and higher.

2. Why does ppp(8) hang when I run it?

This is usually because your hostname will not resolve. The best way to fix this is to make sure that `/etc/hosts` is consulted by your resolver first by editing `/etc/host.conf` and putting the `hosts` line first. Then, simply put an entry in `/etc/hosts` for your local machine. If you have no local network, change your `localhost` line:

```
127.0.0.1        foo.bar.com foo localhost
```

Otherwise, simply add another entry for your host. Consult the relevant man pages for more details.

You should be able to successfully `ping -c1 'hostname'` when you are done.

3. Why will ppp(8) not dial in `-auto` mode?

First, check that you have got a default route. By running `netstat -rn` (see netstat(1)), you should see two entries like this:

```
Destination        Gateway            Flags     Refs      Use     Netif Expire
default            10.0.0.2           UGSc        0         0      tun0
10.0.0.2           10.0.0.1           UH          0         0      tun0
```

This is assuming that you have used the addresses from the handbook, the man page or from the ppp.conf.sample file. If you do not have a default route, it may be because you are running an old version of ppp(8) that does not understand the word `HISADDR` in the ppp.conf file. If your version of ppp(8) is from before FreeBSD 2.2.5, change the

```
add 0 0 HISADDR
```

line to one saying

```
add 0 0 10.0.0.2
```

Another reason for the default route line being missing is that you have mistakenly set up a default router in your `/etc/rc.conf` (see rc.conf(5)) file (this file was called `/etc/sysconfig` prior to release 2.2.2), and you have omitted the line saying

```
delete ALL
```

from `ppp.conf`. If this is the case, go back to the Final system configuration (../handbook/ppp-and-slip.html#USERPPP-FINAL) section of the handbook.

4. What does `No route to host` mean?

This error is usually due to a missing

```
MYADDR:
  delete ALL
  add 0 0 HISADDR
```

section in your `/etc/ppp/ppp.linkup` file. This is only necessary if you have a dynamic IP address or do not know the address of your gateway. If you are using interactive mode, you can type the following after entering `packet mode` (packet mode is indicated by the capitalized PPP in the prompt):

```
delete ALL
add 0 0 HISADDR
```

Refer to the PPP and Dynamic IP addresses (../handbook/ppp-and-slip.html#USERPPP-DYNAMICIP) section of the handbook for further details.

5. Why does my connection drop after about 3 minutes?

The default PPP timeout is 3 minutes. This can be adjusted with the line

```
set timeout NNN
```

where *NNN* is the number of seconds of inactivity before the connection is closed. If *NNN* is zero, the connection is never closed due to a timeout. It is possible to put this command in the `ppp.conf` file, or to type it at the prompt in interactive mode. It is also possible to adjust it on the fly while the line is active by connecting to **ppp**s server socket using telnet(1) or pppctl(8). Refer to the ppp(8) man page for further details.

6. Why does my connection drop under heavy load?

If you have Link Quality Reporting (LQR) configured, it is possible that too many LQR packets are lost between your machine and the peer. Ppp deduces that the line must therefore be bad, and disconnects. Prior to FreeBSD version 2.2.5, LQR was enabled by default. It is now disabled by default. LQR can be disabled with the line

```
disable lqr
```

7. Why does my connection drop after a random amount of time?

Sometimes, on a noisy phone line or even on a line with call waiting enabled, your modem may hang up because it thinks (incorrectly) that it lost carrier.

There is a setting on most modems for determining how tolerant it should be to temporary losses of carrier. On a USR Sportster for example, this is measured by the S10 register in tenths of a second. To make your modem more forgiving, you could add the following send-expect sequence to your dial string:

```
set dial "...... ATS10=10 OK ......"
```

Refer to your modem manual for details.

8. Why does my connection hang after a random amount of time?

Many people experience hung connections with no apparent explanation. The first thing to establish is which side of the link is hung.

If you are using an external modem, you can simply try using ping(8) to see if the TD light is flashing when you transmit data. If it flashes (and the RD light does not), the problem is with the remote end. If TD does not flash, the problem is local. With an internal modem, you will need to use the `set server` command in your `ppp.conf` file. When the hang occurs, connect to ppp(8) using pppctl(8). If your network connection suddenly revives (PPP was revived due to the activity on the diagnostic socket) or if you cannot connect (assuming the `set socket` command succeeded at startup time), the problem is local. If you can connect and things are still hung, enable local async logging with `set log local async` and use ping(8) from another window or terminal to make use of the link. The async logging will show you the data being transmitted and received on the link. If data is going out and not coming back, the problem is remote.

Having established whether the problem is local or remote, you now have two possibilities:

9. The remote end is not responding. What can I do?

There is very little you can do about this. Most ISPs will refuse to help if you are not running a Microsoft OS. You can `enable lqr` in your `ppp.conf` file, allowing ppp(8) to detect the remote failure and hang up, but this detection is relatively slow and therefore not that useful. You may want to avoid telling your ISP that you are running user-PPP...

First, try disabling all local compression by adding the following to your configuration:

```
disable pred1 deflate deflate24 protocomp acfcomp shortseq vj
deny pred1 deflate deflate24 protocomp acfcomp shortseq vj
```

Then reconnect to ensure that this makes no difference. If things improve or if the problem is solved completely, determine which setting makes the difference through trial and error. This will provide good ammunition when you contact your ISP (although it may make it apparent that you are not running a Microsoft product).

Before contacting your ISP, enable async logging locally and wait until the connection hangs again. This may use up quite a bit of disk space. The last data read from the port may be of interest. It is usually ascii data, and may even describe the problem (Memory fault, core dumped ?).

If your ISP is helpful, they should be able to enable logging on their end, then when the next link drop occurs, they may be able to tell you why their side is having a problem. Feel free to send the details to Brian Somers <`brian@FreeBSD.org`>, or even to ask your ISP to contact me directly.

10. ppp(8) has hung. What can I do?

Your best bet here is to rebuild ppp(8) by adding `CFLAGS+=-g` and `STRIP=` to the end of the Makefile, then doing a `make clean && make && make install`. When ppp(8) hangs, find the ppp(8) process id with `ps ajxww | fgrep ppp` and run `gdb ppp PID`. From the gdb prompt, you can then use `bt` to get a stack trace.

Send the results to <`brian@Awfulhak.org`>.

11. Why does nothing happen after the Login OK! message?

Prior to FreeBSD version 2.2.5, once the link was established, ppp(8) would wait for the peer to initiate the Line Control Protocol (LCP). Many ISPs will not initiate negotiations and expect the client to do so. To force ppp(8) to initiate the LCP, use the following line:

```
set openmode active
```

: It usually does no harm if both sides initiate negotiation, so openmode is now active by default. However, the next section explains when it *does* do some harm.

12. I keep seeing errors about magic being the same. What does it mean?

Occasionally, just after connecting, you may see messages in the log that say magic is the same . Sometimes, these messages are harmless, and sometimes one side or the other exits. Most PPP implementations cannot survive this problem, and even if the link seems to come up, you will see repeated configure requests and configure acknowledgments in the log file until ppp(8) eventually gives up and closes the connection.

This normally happens on server machines with slow disks that are spawning a getty on the port, and executing ppp(8) from a login script or program after login. I have also heard reports of it happening consistently when using slirp. The reason is that in the time taken between getty(8) exiting and ppp(8) starting, the client-side ppp(8) starts sending Line Control Protocol (LCP) packets. Because ECHO is still switched on for the port on the server, the client ppp(8) sees these packets reflect back.

One part of the LCP negotiation is to establish a magic number for each side of the link so that reflections can be detected. The protocol says that when the peer tries to negotiate the same magic number, a NAK should be sent and a new magic number should be chosen. During the period that the server port has ECHO turned on, the client ppp(8) sends LCP packets, sees the same magic in the reflected packet and NAKs it. It also sees the NAK reflect (which also means ppp(8) must change its magic). This produces a potentially enormous number of magic number changes, all of which are happily piling into the server's tty buffer. As soon as ppp(8) starts on the server, it is flooded with magic number changes and almost immediately decides it has tried enough to negotiate LCP and gives up. Meanwhile, the client, who no longer sees the reflections, becomes happy just in time to see a hangup from the server.

This can be avoided by allowing the peer to start negotiating with the following line in your ppp.conf file:

```
set openmode passive
```

This tells ppp(8) to wait for the server to initiate LCP negotiations. Some servers however may never initiate negotiations. If this is the case, you can do something like:

```
set openmode active 3
```

This tells ppp(8) to be passive for 3 seconds, and then to start sending LCP requests. If the peer starts sending requests during this period, ppp(8) will immediately respond rather than waiting for the full 3 second period.

13. LCP negotiations continue until the connection is closed. What is wrong?

There is currently an implementation mis-feature in ppp(8) where it does not associate LCP, CCP & IPCP responses with their original requests. As a result, if one PPP implementation is more than 6 seconds slower than the other side, the other side will send two additional LCP configuration requests. This is fatal.

Consider two implementations, `A` and `B`. `A` starts sending LCP requests immediately after connecting and `B` takes 7 seconds to start. When `B` starts, `A` has sent 3 LCP REQs. We are assuming the line has ECHO switched off, otherwise we would see magic number problems as described in the previous section. `B` sends a REQ, then an ACK to the first of `A`'s REQs. This results in `A` entering the OPENED state and sending and ACK (the first) back to `B`. In the meantime, `B` sends back two more ACKs in response to the two additional REQs sent by `A` before `B` started up. `B` then receives the first ACK from `A` and enters the OPENED state. `A` receives the second ACK from `B` and goes back to the REQ-SENT state, sending another (forth) REQ as per the RFC. It then receives the third ACK and enters the OPENED state. In the meantime, `B` receives the forth REQ from `A`, resulting in it reverting to the ACK-SENT state and sending another (second) REQ and (forth) ACK as per the RFC. `A` gets the REQ, goes into REQ-SENT and sends another REQ. It immediately receives the following ACK and enters OPENED.

This goes on until one side figures out that they are getting nowhere and gives up.

The best way to avoid this is to configure one side to be `passive` - that is, make one side wait for the other to start negotiating. This can be done with the

```
set openmode passive
```

command. Care should be taken with this option. You should also use the

```
set stopped N
```

command to limit the amount of time that ppp(8) waits for the peer to begin negotiations. Alternatively, the

```
set openmode active N
```

command (where *`N`* is the number of seconds to wait before starting negotiations) can be used. Check the manual page for details.

14. Why does ppp(8) lock up shortly after connection?

Prior to version 2.2.5 of FreeBSD, it was possible that your link was disabled shortly after connection due to ppp(8) mis-handling Predictor1 compression negotiation. This would only happen if both sides tried to negotiate different Compression Control Protocols (CCP). This problem is now corrected, but if you are still running an old version of ppp(8) the problem can be circumvented with the line

```
disable pred1
```

15. Why does ppp(8) lock up when I shell out to test it?

When you execute the `shell` or `!` command, ppp(8) executes a shell (or if you have passed any arguments, ppp(8) will execute those arguments). Ppp will wait for the command to complete before continuing. If you attempt to use the PPP link while running the command, the link will appear to have frozen. This is because ppp(8) is waiting for the command to complete.

If you wish to execute commands like this, use the `!bg` command instead. This will execute the given command in the background, and ppp(8) can continue to service the link.

16. Why does ppp(8) over a null-modem cable never exit?

There is no way for ppp(8) to automatically determine that a direct connection has been dropped. This is due to the lines that are used in a null-modem serial cable. When using this sort of connection, LQR should always be enabled with the line

```
enable lqr
```

LQR is accepted by default if negotiated by the peer.

17. Why does ppp(8) dial for no reason in -auto mode?

If ppp(8) is dialing unexpectedly, you must determine the cause, and set up Dial filters (dfilters) to prevent such dialing.

To determine the cause, use the following line:

```
set log +tcp/ip
```

This will log all traffic through the connection. The next time the line comes up unexpectedly, you will see the reason logged with a convenient timestamp next to it.

You can now disable dialing under these circumstances. Usually, this sort of problem arises due to DNS lookups. To prevent DNS lookups from establishing a connection (this will *not* prevent ppp(8) from passing the packets through an established connection), use the following:

```
set dfilter 1 deny udp src eq 53
set dfilter 2 deny udp dst eq 53
set dfilter 3 permit 0/0 0/0
```

This is not always suitable, as it will effectively break your demand-dial capabilities - most programs will need a DNS lookup before doing any other network related things.

In the DNS case, you should try to determine what is actually trying to resolve a host name. A lot of the time, sendmail(8) is the culprit. You should make sure that you tell sendmail not to do any DNS lookups in its configuration file. See the section on Mail Configuration for details on how to create your own configuration file and what should go into it. You may also want to add the following line to your `.mc` file:

```
define(`confDELIVERY_MODE', `d')dnl
```

This will make sendmail queue everything until the queue is run (usually, sendmail is invoked with `-bd -q30m`, telling it to run the queue every 30 minutes) or until a `sendmail -q` is done (perhaps from your ppp.linkup file).

18. What do these CCP errors mean?

I keep seeing the following errors in my log file:

```
CCP: CcpSendConfigReq
CCP: Received Terminate Ack (1) state = Req-Sent (6)
```

This is because ppp(8) is trying to negotiate Predictor1 compression, and the peer does not want to negotiate any compression at all. The messages are harmless, but if you wish to remove them, you can disable Predictor1 compression locally too:

```
disable pred1
```

19. Why does ppp(8) lock up during file transfers with IO errors?

Under FreeBSD 2.2.2 and before, there was a bug in the tun driver that prevents incoming packets of a size larger than the tun interface's MTU size. Receipt of a packet greater than the MTU size results in an IO error being logged via syslogd.

The PPP specification says that an MRU of 1500 should *always* be accepted as a minimum, despite any LCP negotiations, therefore it is possible that should you decrease the MTU to less than 1500, your ISP will transmit packets of 1500 regardless, and you will tickle this non-feature - locking up your link.

The problem can be circumvented by never setting an MTU of less than 1500 under FreeBSD 2.2.2 or before.

20. Why does ppp(8) not log my connection speed?

In order to log all lines of your modem conversation, you must enable the following:

```
set log +connect
```

This will make ppp(8) log everything up until the last requested expect string.

If you wish to see your connect speed and are using PAP or CHAP (and therefore do not have anything to chat after the CONNECT in the dial script - no `set login` script), you must make sure that you instruct ppp(8) to expect the whole CONNECT line, something like this:

```
set dial "ABORT BUSY ABORT NO\\sCARRIER TIMEOUT 4 \
  \"\" ATZ OK-ATZ-OK ATDT\\T TIMEOUT 60 CONNECT \\c \\n"
```

Here, we get our CONNECT, send nothing, then expect a line-feed, forcing ppp(8) to read the whole CONNECT response.

21. Why does ppp(8) ignore the `\` character in my chat script?

Ppp parses each line in your config files so that it can interpret strings such as `set phone "123 456 789"` correctly (and realize that the number is actually only *one* argument. In order to specify a `"` character, you must escape it using a backslash (`\`).

When the chat interpreter parses each argument, it re-interprets the argument in order to find any special escape sequences such as `\P` or `\T` (see the man page). As a result of this double-parsing, you must remember to use the correct number of escapes.

If you wish to actually send a `\` character to (say) your modem, you would need something like:

```
set dial "\"\" ATZ OK-ATZ-OK AT\\\\X OK"
```

resulting in the following sequence:

```
ATZ
OK
AT\X
OK
```

or

```
set phone 1234567
set dial "\"\" ATZ OK ATDT\\T"
```

resulting in the following sequence:

```
ATZ
OK
ATDT1234567
```

22. Why does ppp(8) get a seg-fault, but I see no `ppp.core` file?

Ppp (or any other program for that matter) should never dump core. Because ppp(8) runs with an effective user id of 0, the operating system will not write ppp(8)'s core image to disk before terminating it. If, however ppp(8) is actually terminating due to a segmentation violation or some other signal that normally causes core to be dumped, *and* you are sure you are using the latest version (see the start of this section), then you should do the following:

```
% tar xfz ppp-*.src.tar.gz
% cd ppp*/ppp
% echo STRIP= >>Makefile
% echo CFLAGS+=-g >>Makefile
% make clean all
% su
# make install
# chmod 555 /usr/sbin/ppp
```

You will now have a debuggable version of ppp(8) installed. You will have to be `root` to run ppp(8) as all of its privileges have been revoked. When you start ppp(8), take a careful note of what your current directory was at the time.

Now, if and when ppp(8) receives the segmentation violation, it will dump a core file called `ppp.core`. You should then do the following:

```
% su
# gdb /usr/sbin/ppp ppp.core
(gdb) bt
.....
(gdb) f 0
....
(gdb) i args
....
(gdb) l
.....
```

All of this information should be given alongside your question, making it possible to diagnose the problem.

If you are familiar with gdb, you may wish to find out some other bits and pieces such as what actually caused the dump and the addresses & values of the relevant variables.

23. Why does the process that forces a dial in auto mode never connect?

This was a known problem with ppp(8) set up to negotiate a dynamic local IP number with the peer in auto mode. It is fixed in the latest version - search the man page for `iface`.

The problem was that when that initial program calls connect(2), the IP number of the tun interface is assigned to the socket endpoint. The kernel creates the first outgoing packet and writes it to the tun device. ppp(8) then reads the packet and establishes a connection. If, as a result of ppp(8)'s dynamic IP assignment, the interface address is changed, the original socket endpoint will be invalid. Any subsequent packets sent to the peer will usually be dropped. Even if they are not, any responses will not route back to the originating machine as the IP number is no longer owned by that machine.

There are several theoretical ways to approach this problem. It would be nicest if the peer would re-assign the same IP number if possible `:-)` The current version of ppp(8) does this, but most other implementations do not.

The easiest method from our side would be to never change the tun interface IP number, but instead to change all outgoing packets so that the source IP number is changed from the interface IP to the negotiated IP on the fly. This is essentially what the `iface-alias` option in the latest version of ppp(8) is doing (with the help of libalias(3) and ppp(8)'s `-nat` switch) - it is maintaining all previous interface addresses and NATing them to the last negotiated address.

Another alternative (and probably the most reliable) would be to implement a system call that changes all bound sockets from one IP to another. ppp(8) would use this call to modify the sockets of all existing programs when a new IP number is negotiated. The same system call could be used by dhcp clients when they are forced to re-bind() their sockets.

Yet another possibility is to allow an interface to be brought up without an IP number. Outgoing packets would be given an IP number of 255.255.255.255 up until the first SIOCAIFADDR ioctl is done. This would result in fully binding the socket. It would be up to ppp(8) to change the source IP number, but only if it is set to 255.255.255.255, and only the IP number and IP checksum would need to change. This, however is a bit of a hack as the kernel would be sending bad packets to an improperly configured interface, on the assumption that some other mechanism is capable of fixing things retrospectively.

24. Why do most games not work with the -nat switch?

The reason games and the like do not work when libalias is in use is that the machine on the outside will try to open a connection or send (unsolicited) UDP packets to the machine on the inside. The NAT software does not know that it should send these packets to the interior machine.

To make things work, make sure that the only thing running is the software that you are having problems with, then either run tcpdump on the tun interface of the gateway or enable ppp(8) tcp/ip logging (`set log +tcp/ip`) on the gateway.

When you start the offending software, you should see packets passing through the gateway machine. When something comes back from the outside, it will be dropped (that is the problem). Note the port number of these packets then shut down the offending software. Do this a few times to see if the port numbers are consistent. If they are, then the following line in the relevant section of `/etc/ppp/ppp.conf` will make the software functional:

```
nat port proto internalmachine:port port
```

where *`proto`* is either `tcp` or `udp`, *`internalmachine`* is the machine that you want the packets to be sent to and *`port`* is the destination port number of the packets.

You will not be able to use the software on other machines without changing the above command, and running the software on two internal machines at the same time is out of the question - after all, the outside world is seeing your entire internal network as being just a single machine.

If the port numbers are not consistent, there are three more options:

1. Submit support in libalias. Examples of special cases can be found in `/usr/src/lib/libalias/alias_*.c` (`alias_ftp.c` is a good prototype). This usually involves reading certain recognised outgoing packets, identifying the instruction that tells the outside machine to initiate a connection back to the internal machine on a specific (random) port and setting up a route in the alias table so that the subsequent packets know where to go.

   This is the most difficult solution, but it is the best and will make the software work with multiple machines.
2. Use a proxy. The application may support socks5 for example, or (as in the cvsup case) may have a passive option that avoids ever requesting that the peer open connections back to the local machine.
3. Redirect everything to the internal machine using `nat addr`. This is the sledge-hammer approach.

25. Has anybody made a list of useful port numbers?

Not yet, but this is intended to grow into such a list (if any interest is shown). In each example, *`internal`* should be replaced with the IP number of the machine playing the game.

Asheron's Call

```
nat port udp internal :65000 65000
```

Manually change the port number within the game to 65000. If you have got a number of machines that you wish to play on assign a unique port number for each (i.e. 65001, 65002, etc) and add a `nat port` line for each one.

Half Life

```
nat port udp internal:27005 27015
```

PCAnywhere 8.0

`nat port udp internal:5632 5632`

`nat port tcp internal:5631 5631`

Quake

`nat port udp internal:6112 6112`

Alternatively, you may want to take a look at www.battle.net (http://www.battle.net/support/proxy/) for Quake proxy support.

Quake 2

`nat port udp internal:27901 27910`

`nat port udp internal:60021 60021`

`nat port udp internal:60040 60040`

Red Alert

`nat port udp internal:8675 8675`

`nat port udp internal:5009 5009`

26. What are FCS errors?

FCS stands for `F`rame `C`heck `S`equence. Each PPP packet has a checksum attached to ensure that the data being received is the data being sent. If the FCS of an incoming packet is incorrect, the packet is dropped and the HDLC FCS count is increased. The HDLC error values can be displayed using the `show hdlc` command.

If your link is bad (or if your serial driver is dropping packets), you will see the occasional FCS error. This is not usually worth worrying about although it does slow down the compression protocols substantially. If you have an external modem, make sure your cable is properly shielded from interference - this may eradicate the problem.

If your link freezes as soon as you have connected and you see a large number of FCS errors, this may be because your link is not 8 bit clean. Make sure your modem is not using software flow control (XON/XOFF). If your datalink *must* use software flow control, use the command `set accmap 0x000a0000` to tell ppp(8) to escape the `^Q` and `^S` characters.

Another reason for seeing too many FCS errors may be that the remote end has stopped talking PPP. You may want to enable `async` logging at this point to determine if the incoming data is actually a login or shell prompt. If you have a shell prompt at the remote end, it is possible to terminate ppp(8) without dropping the line by using the `close lcp` command (a following `term` command will reconnect you to the shell on the remote machine.

If nothing in your log file indicates why the link might have been terminated, you should ask the remote administrator (your ISP?) why the session was terminated.

27. Why do MacOS and Windows 98 connections freeze when running PPPoE on the gateway?

Thanks to Michael Wozniak <`mwozniak@netcom.ca`> for figuring this out and Dan Flemming <`danflemming@mac.com`> for the Mac solution:

This is due to what is called a Black Hole router. MacOS and Windows 98 (and maybe other Microsoft OSs) send TCP packets with a requested segment size too big to fit into a PPPoE frame (MTU is 1500 by default for Ethernet) *and* have the do not fragment bit set (default of TCP) and the Telco router is not sending ICMP must fragment back to the www site you are trying to load. (Alternatively, the router is sending the ICMP packet correctly, but the firewall at the www site is dropping it.) When the www server is sending you frames that do not fit into the PPPoE pipe the Telco router drops them on the floor and your page does not load (some pages/graphics do as they are smaller than a MSS.) This seems to be the default of most Telco PPPoE configurations (if only they knew how to program a router... sigh...)

One fix is to use regedit on your 95/98 boxes to add the following registry entry...

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Class\NetTrans\0000\MaxMTU
```

It should be a string with a value 1436 , as some ADSL routers are reported to be unable to deal with packets larger than this. This registry key has been changed to `Tcpip\Parameters\Interfaces\`*`ID for adapter`*`\MTU` in Windows 2000 and becomes a DWORD.

Refer to the Microsoft Knowledge Base documents Q158474 - Windows TCPIP Registry Entries (http://support.microsoft.com/support/kb/articles/Q158/4/74.asp) and Q120642 - TCPIP & NBT Configuration Parameters for Windows NT (http://support.microsoft.com/support/kb/articles/Q120/6/42.asp) for more information on changing Windows MTU to work with a NAT router.

Another regedit possibility under Windows 2000 is to set the `Tcpip\Parameters\Interfaces\`*`ID for adapter`*`\EnablePMTUBHDetect` DWORD to 1 as mentioned in the Microsoft document 120642 mentioned above.

Unfortunately, MacOS does not provide an interface for changing TCP/IP settings. However, there is commercial software available, such as OTAdvancedTuner (OT for OpenTransport, the MacOS TCP/IP stack) by Sustainable Softworks (http://www.softworks.com/), that will allow users to customize TCP/IP settings. MacOS NAT users should select `ip_interface_MTU` from the drop-down menu, enter `1450` instead of `1500` in the box, click the box next to `Save as Auto Configure`, and click `Make Active`.

The latest version of ppp(8) (2.3 or greater) has an `enable tcpmssfixup` command that will automatically adjust the MSS to an appropriate value. This facility is enabled by default. If you are stuck with an older version of ppp(8), you may want to look at the **tcpmssd** port.

28. None of this helps - I am desperate! What can I do?

If all else fails, send as much information as you can, including your config files, how you are starting ppp(8), the relevant parts of your log file and the output of the `netstat -rn` command (before and after connecting) to the FreeBSD general questions (http://lists.FreeBSD.org/mailman/listinfo/freebsd-questions) or the comp.unix.bsd.freebsd.misc (news:comp.unix.bsd.freebsd.misc) news group, and someone should point you in the right direction.

15. Serial Communications

This section answers common questions about serial communications with FreeBSD. PPP and SLIP are covered in the µÚ 12 section.

1. How do I tell if FreeBSD found my serial ports?

As the FreeBSD kernel boots, it will probe for the serial ports in your system for which the kernel was configured. You can either watch your system closely for the messages it prints or run the command

```
% dmesg | grep sio
```

after your system is up and running.

Here is some example output from the above command:

```
sio0 at 0x3f8-0x3ff irq 4 on isa
sio0: type 16550A
sio1 at 0x2f8-0x2ff irq 3 on isa
sio1: type 16550A
```

This shows two serial ports. The first is on irq 4, is using port address `0x3f8`, and has a 16550A-type UART chip. The second uses the same kind of chip but is on irq 3 and is at port address `0x2f8`. Internal modem cards are treated just like serial ports---except that they always have a modem attached to the port.

The `GENERIC` kernel includes support for two serial ports using the same irq and port address settings in the above example. If these settings are not right for your system, or if you have added modem cards or have more serial ports than your kernel is configured for, just reconfigure your kernel. See section about building a kernel for more details.

2. How do I tell if FreeBSD found my modem cards?

Refer to the answer to the previous question.

3. I just upgraded to 2.0.5 and my `tty0X` are missing! How do I solve this problem?

Do not worry, they have been merged with the `ttydX` devices. You will have to change any old configuration files you have, though.

4. How do I access the serial ports on FreeBSD?

The third serial port, `sio2` (see sio(4), known as COM3 in DOS), is on `/dev/cuaa2` for dial-out devices, and on `/dev/ttyd2` for dial-in devices. What is the difference between these two classes of devices?

You use `ttydX` for dial-ins. When opening `/dev/ttydX` in blocking mode, a process will wait for the corresponding `cuaaX` device to become inactive, and then wait for the carrier detect line to go active. When you open the `cuaaX` device, it makes sure the serial port is not already in use by the `ttydX` device. If the port is available, it steals it from the `ttydX` device. Also, the `cuaaX` device does not care about carrier detect. With this scheme and an auto-answer modem, you can have remote users log in and you can still dial out with the same modem and the system will take care of all the conflicts.

5. How do I enable support for a multiport serial card?

Again, the section on kernel configuration provides information about configuring your kernel. For a multiport serial card, place an sio(4) line for each serial port on the card in the kernel configuration file. But place the irq and vector specifiers on only one of the entries. All of the ports on the card should share one irq. For consistency, use the last serial port to specify the irq. Also, specify the `COM_MULTIPORT` option.

The following example is for an AST 4-port serial card on irq 7:

```
options "COM_MULTIPORT"
device sio4 at isa? port 0x2a0 tty flags 0x781
device sio5 at isa? port 0x2a8 tty flags 0x781
device sio6 at isa? port 0x2b0 tty flags 0x781
device sio7 at isa? port 0x2b8 tty flags 0x781 irq 7 vector siointr
```

The flags indicate that the master port has minor number 7 (`0x700`), diagnostics enabled during probe (`0x080`), and all the ports share an irq (`0x001`).

6. Can FreeBSD handle multiport serial cards sharing irqs?

Not yet. You will have to use a different irq for each card.

7. Can I set the default serial parameters for a port?

The `ttydX` (or `cuaaX`) device is the regular device you will want to open for your applications. When a process opens the device, it will have a default set of terminal I/O settings. You can see these settings with the command

```
# stty -a -f /dev/ttyd1
```

When you change the settings to this device, the settings are in effect until the device is closed. When it is reopened, it goes back to the default set. To make changes to the default set, you can open and adjust the settings of the initial state device. For example, to turn on CLOCAL mode, 8 bits, and XON/XOFF flow control by default for ttyd5, do:

```
# stty -f /dev/ttyid5 clocal cs8 ixon ixoff
```

A good place to do this is in `/etc/rc.serial`. Now, an application will have these settings by default when it opens `ttyd5`. It can still change these settings to its liking, though.

You can also prevent certain settings from being changed by an application by making adjustments to the lock state device. For example, to lock the speed of `ttyd5` to 57600 bps, do

```
# stty -f /dev/ttyld5 57600
```

Now, an application that opens `ttyd5` and tries to change the speed of the port will be stuck with 57600 bps.

Naturally, you should make the initial state and lock state devices writable only by `root`. The MAKEDEV(8) script does *NOT* do this when it creates the device entries.

8. How can I enable dialup logins on my modem?

So you want to become an Internet service provider, eh? First, you will need one or more modems that can auto-answer. Your modem will need to assert carrier-detect when it detects a carrier and not assert it all the time. It will need to hang up the phone and reset itself when the data terminal ready (DTR) line goes from on to off. It should probably use `RTS/CTS` flow control or no local flow control at all. Finally, it must use a constant speed between the computer and itself, but (to be nice to your callers) it should negotiate a speed between itself and the remote modem.

For many Hayes command-set--compatible modems, this command will make these settings and store them in nonvolatile memory:

```
AT &C1 &D3 &K3 &Q6 S0=1 &W
```

See the section on sending AT commands below for information on how to make these settings without resorting to an MS-DOS terminal program.

Next, make an entry in `/etc/ttys` (see ttys(5)) for the modem. This file lists all the ports on which the operating system will await logins. Add a line that looks something like this:

```
ttyd1 "/usr/libexec/getty std.57600" dialup on insecure
```

This line indicates that the second serial port (`/dev/ttyd1`) has a modem connected running at 57600 bps and no parity (`std.57600`, which comes from the file `/etc/gettytab`, see gettytab(5)). The terminal type for this port is `dialup`. The port is `on` and is `insecure`---meaning `root` logins on the port are not allowed. For dialin ports like this one, use the `ttydX` entry.

It is common practice to use `dialup` as the terminal type. Many users set up in their `.profile` or `.login` files a prompt for the actual terminal type if the starting type is dialup. The example shows the port as insecure. To become `root` on this port, you have to login as a regular user, then su(1) to become `root`. If you use `secure` then `root` can login in directly.

After making modifications to `/etc/ttys`, you need to send a hangup or HUP signal to the init(8) process:

```
# kill -HUP 1
```

This forces the init(8) process to reread `/etc/ttys`. The init process will then start getty processes on all `on` ports. You can find out if logins are available for your port by typing

```
% ps -ax | grep '[t]tyd1'
```

You should see something like:

```
747 ??  I      0:00.04 /usr/libexec/getty std.57600 ttyd1
```

9. How can I connect a dumb terminal to my FreeBSD box?

If you are using another computer as a terminal into your FreeBSD system, get a null modem cable to go between the two serial ports. If you are using an actual terminal, see its accompanying instructions.

Then, modify `/etc/ttys` (see ttys(5)), like above. For example, if you are hooking up a WYSE-50 terminal to the fifth serial port, use an entry like this:

```
ttyd4 "/usr/libexec/getty std.38400" wyse50 on secure
```

This example shows that the port on `/dev/ttyd4` has a wyse50 terminal connected at 38400 bps with no parity (`std.38400` from `/etc/gettytab`, see gettytab(5)) and `root` logins are allowed (`secure`).

10. Why can I not run `tip` or `cu`?

On your system, the programs tip(1) and cu(1) are probably executable only by `uucp` and group `dialer`. You can use the group `dialer` to control who has access to your modem or remote systems. Just add yourself to group dialer.

Alternatively, you can let everyone on your system run tip(1) and cu(1) by typing:

```
# chmod 4511 /usr/bin/cu
# chmod 4511 /usr/bin/tip
```

11. My stock Hayes modem is not supported---what can I do?

Actually, the man page for tip(1) is out of date. There is a generic Hayes dialer already built in. Just use `at=hayes` in your `/etc/remote` (see remote(5)) file.

The Hayes driver is not smart enough to recognize some of the advanced features of newer modems---messages like `BUSY`, `NO DIALTONE`, or `CONNECT 115200` will just confuse it. You should turn those messages off when you use tip(1) (using `ATX0&W`).

Also, the dial timeout for tip(1) is 60 seconds. Your modem should use something less, or else tip will think there is a communication problem. Try `ATS7=45&W`.

Actually, as shipped tip(1) does not yet support it fully. The solution is to edit the file `tipconf.h` in the directory `/usr/src/usr.bin/tip/tip`. Obviously you need the source distribution to do this.

Edit the line `#define HAYES 0` to `#define HAYES 1`. Then `make` and `make install`. Everything works nicely after that.

12. How am I expected to enter these AT commands?

Make what is called a direct entry in your `/etc/remote` file (see remote(5)). For example, if your modem is hooked up to the first serial port, `/dev/cuaa0`, then put in the following line:

```
cuaa0:dv=/dev/cuaa0:br#19200:pa=none
```

Use the highest bps rate your modem supports in the br capability. Then, type `tip cuaa0` (see tip(1)) and you will be connected to your modem.

If there is no `/dev/cuaa0` on your system, do this:

```
# cd /dev
# sh MAKEDEV cuaa0
```

Or use cu as `root` with the following command:

```
# cu -lline -sspeed
```

with *`line`* being the serial port (e.g. `/dev/cuaa0`) and *`speed`* being the speed (e.g.`57600`). When you are done entering the AT commands hit `~.` to exit.

13. Why does the `<@>` sign for the pn capability not work?

The `<@>` sign in the phone number capability tells tip to look in `/etc/phones` for a phone number. But the `<@>` sign is also a special character in capability files like `/etc/remote`. Escape it with a backslash:

```
pn=\@
```

14. How can I dial a phone number on the command line?

Put what is called a "generic" entry in your `/etc/remote` file (see remote(5)). For example:

```
tip115200|Dial any phone number at 115200 bps:\
        :dv=/dev/cuaa0:br#115200:at=hayes:pa=none:du:
tip57600|Dial any phone number at 57600 bps:\
        :dv=/dev/cuaa0:br#57600:at=hayes:pa=none:du:
```

Then you can do something like `tip -115200 5551234`. If you prefer cu(1) over tip(1), use a generic cu entry:

```
cu115200|Use cu to dial any number at 115200bps:\
        :dv=/dev/cuaa1:br#57600:at=hayes:pa=none:du:
```

and type `cu 5551234 -s 115200`.

15. Do I have to type in the bps rate every time I do that?

Put in an entry for `tip1200` or `cu1200`, but go ahead and use whatever bps rate is appropriate with the br capability. tip(1) thinks a good default is 1200 bps which is why it looks for a `tip1200` entry. You do not have to use 1200 bps, though.

16. How can I more easily access a number of hosts through a terminal server?

Rather than waiting until you are connected and typing `CONNECT` *`host`* each time, use tip's `cm` capability. For example, these entries in `/etc/remote` (see remote(5)):

```
pain|pain.deep13.com|Forrester's machine:\
        :cm=CONNECT pain\n:tc=deep13:
muffin|muffin.deep13.com|Frank's machine:\
        :cm=CONNECT muffin\n:tc=deep13:
deep13:Gizmonics Institute terminal server:\
        :dv=/dev/cuaa2:br#38400:at=hayes:du:pa=none:pn=5551234:
```

will let you type `tip pain` or `tip muffin` to connect to the hosts `pain` or `muffin`; and `tip deep13` to get to the terminal server.

17. Can tip try more than one line for each site?

This is often a problem where a university has several modem lines and several thousand students trying to use them...

Make an entry for your university in `/etc/remote` (see remote(5)) and use `<\@>` for the `pn` capability:

```
big-university:\
        :pn=\@:tc=dialout
dialout:\
        :dv=/dev/cuaa3:br#9600:at=courier:du:pa=none:
```

Then, list the phone numbers for the university in `/etc/phones` (see phones(5)):

```
big-university 5551111
big-university 5551112
big-university 5551113
big-university 5551114
```

tip(1) will try each one in the listed order, then give up. If you want to keep retrying, run tip(1) in a while loop.

18. Why do I have to hit CTRL+P twice to send CTRL+P once?

CTRL+P is the default force character, used to tell tip(1) that the next character is literal data. You can set the force character to any other character with the `~s` escape, which means set a variable .

Type `~sforce=`*`single-char`* followed by a newline. *`single-char`* is any single character. If you leave out *`single-char`*, then the force character is the nul character, which you can get by typing CTRL+2 or CTRL+SPACE. A pretty good value for *`single-char`* is SHIFT+CTRL+6, which I have seen only used on some terminal servers.

You can have the force character be whatever you want by specifying the following in your `$HOME/.tiprc` file:

```
force=single-char
```

19. Why is everything I type suddenly in UPPER CASE?

You must have pressed CTRL+A, tip(1) raise character , specially designed for people with broken caps-lock keys. Use `~s` as above and set the variable raisechar to something reasonable. In fact, you can set it to the same as the force character, if you never expect to use either of these features.

Here is a sample .tiprc file perfect for Emacs users who need to type CTRL+2 and CTRL+A a lot:

```
force=^^
raisechar=^^
```

The ^^ is SHIFT+CTRL+6.

20. How can I do file transfers with `tip`?

If you are talking to another Unix system, you can send and receive files with `~p` (put) and `~t` (take). These commands run cat(1) and echo(1) on the remote system to accept and send files. The syntax is:

```
~p <local-file> [<remote-file>]
~t <remote-file> [<local-file>]
```

There is no error checking, so you probably should use another protocol, like zmodem.

21. How can I run zmodem with **tip**?

First, install one of the zmodem programs from the ports collection (such as one of the two from the comms category, **lrzsz** or **rzsz**.

To receive files, start the sending program on the remote end. Then, press enter and type `~C rz` (or `~C lrz` if you installed **lrzsz**) to begin receiving them locally.

To send files, start the receiving program on the remote end. Then, press enter and type `~C sz` *`files`* (or `~C lsz` *`files`*) to send them to the remote system.

22. Why does FreeBSD not find my serial ports, even when the settings are correct?

Motherboards and cards with Acer UARTs do not probe properly under the FreeBSD sio probe. Obtain a patch from www.lemis.com (http://www.lemis.com/serial-port-patch.html) to fix your problem.

16.

1. FreeBSD (swap) Linux

FreeBSD (swap) Linux
FreeBSD (swap)
Linux (swap)
(swap)

FreeBSD (swap) FreeBSD
(swap)
(swap)

2. `top`

(kernel) (disk cache) top(1) `(Inact)` `(Cache)`
`(Buf)` (cached)
top(1) `(Free)`

3. ( ) a.out ELF

Freebsd `ELF` Unix

: FreeBSD 3.x FreeBSD a.out

a.out(5)

Unix
(a.out(5)) .text .data
.bss

COFF

SVR3 .text .data .bss

ELF

ELF COFF 32-bits 64-bits
ELF ABI
SYSV SVR4 Solaris SCO ABI

ABI(Application Binary Interface) ABI
CPU
(Binary Code) Sun Solaris ABI CPU Solaris
Windows Intel x86 Windows 9x/me Windows
NT/2k/XP

FreeBSD ABI brandelf(1)
i

FreeBSD FreeBSD a.out(5) BSD
FreeBSD ELF ( ) ,
FreeBSD ELF Linux ELF
`a.out` Linux (shared libraries)
(jump-table)
ELF (shared libraries)

FreeBSD (shared libraries) **SunOS**
3.0 FreeBSD ELF `a.out`
GNU `a.out`
compiler linker GNU
ISO-C++ (constructors) (destructors) ELF

4. , , ?

a.out
(PDP-11) Unix
a.out Motorola 68k VAXen

CPU
(RISC) `a.out`
`a.out`
COFF ECOFF
ELF
(shared libraries)
(VM System) `a.out`

main `a.out`
`a.out`
ELF
ELF `a.out`

FreeBSD build tools ( ) FreeBSD
(shared libraries) GNU
(building cross compilers) (plugging in different formats) FreeBSD
FreeBSD as ld GNU (Cross Compiler) ELF
(shared libraries) C++ ... ELF
FreeBSD ELF `a.out`

ELF a.out ELF
ELF a.out
...
GENERIC (kernel) `a.out`
`a.out` (kernel)

5. chmod (symlink)

Symlinks chmod(1) symlinks
`foo` symlink `bar`

```
% chmod g-w bar
```

`foo`

`-H` `-L` `-R` chmod(1) symlink(7)

`-R` chmod(1) (RECURSIVE) chmod(1)
chmod(1)
symlink (/) `foo` `bar` `foo` (`bar`)

```
% chmod 555 foo/
```

chmod(1) `foo` `bar`

6. FreeBSD 2.2.x (login names)

`UT_NAMESIZE`
( ) (`8` `9`
`15` `20`) ( )
Sun NIS Client Unix

FreeBSD 3.0 16
3.0

`/usr/include/utmp.h` UT_NAMESIZE `/usr/include/sys/param.h`
MAXLOGNAME UT_NAMESIZE , /usr/include
/usr/src/..

7. FreeBSD DOS

3.0 BSDI **doscmd** DOS
FreeBSD-emulation
(http://lists.FreeBSD.org/mailman/listinfo/freebsd-emulation)

3.0 ports 8088 BIOS DOS **pcemu** X Windows(XFree86)

8. FreeBSD

FreeBSD (../fdp-primer/translations.html)

9. FreeBSD.org

FreeBSD.org

IP

FreeBSD

HTML

FreeBSD IP IP

DNS DNS (DSL cable dialup)

SMTP EHLO/HELO hostname IP

SMTP

DNS

message ID localhost

message ID

message ID

10. FreeBSD

FreeBSD Unix

Arbornet, Inc (http://www.arbornet.org/) M-Net 1983 Unix Altos System III 1991 BSD/OS 2000 FreeBSD M-Net SSH telnet FreeBSD M-Net M-Net BBS

Grex (http://www.grex.org/) M-Net BBS Sun 4M SunOS

11. `sup`

SUP (http://www.FreeBSD.org/cgi/ports.cgi?^sup) Software Update Protocol CMU

SUP
CVSup (../handbook/synching.html#CVSUP)

12.

BSD
(beastie) beastie BSD
BSD (http://www.mckusick.com/beastie/index.html)

13. BSD

BSD Marshall Kirk McKusick
BSD (http://www.mckusick.com/beastie/mainpage/copyright.html)

Kirk McKusick
BSD (http://www.mckusick.com/beastie/index.html)

14. BSD

`/usr/share/examples/BSD_daemon/` Xfig eps

15.

FreeBSD (http://www.FreeBSD.org/doc/zh_TW.Big5/books/handbook/freebsd-glossary.html)

16.

sleep(1) Poul-Henning Kamp
`<phk@FreeBSD.org>` ( )...
(http://www.FreeBSD.org/cgi/getmsg.cgi?fetch=506636+517178+/usr/local/www/db/text/1999/freebsd-hackers/19991003.freebsd-hackers)

?

C· · (C.
Northcote Parkinson) 1960 Parkinson’s Law(
)

[ ]

( )

Richard P. Feynmann

Los Alamos

(setting your fingerprint)

!

Poul-Henning Kamp <`phk@FreeBSD.org`> on freebsd-hackers, October 2, 1999

17. FreeBSD

1. How cool is FreeBSD

FreeBSD Linux DOS FreeBSD

250

LSD-25 35% FreeBSD Linux

Apple Drag and Drop Scratch and
Sniff It's a funny old business we're in

FreeBSD Linux HLT (halt)
APM(Automatic Power Management) FreeBSD CPU

2.

FreeBSD (
)

BSD ---

DOS `fdisk /mbr`
Bill Gates BSD
DOS Windows

3. FreeBSD hacker

-current

email -questions

submit PR doc

commit buildworld

flame war PR patch

buildworld

buildworld cvsup

patch -hackers

patch -current PR

committer Community
-core

patch style(9)

GPL

flame war GPL BSD MIT NPL license FSF

-chat -advocacy

commit

commit FreeBSD

-core

FreeBSD

-hackers -chat SNR unsubscribe

unsubscribe unsubscribe list
majordomo footer

TenDRA 0.364%
FreeBSD TenDRA EGCS

PR MFC

Nik Clayton <*`nik@FreeBSD.org`*>

list

:-)

4. `/dev/null`

CPU / CPU
`/dev/null`
CPU `/dev/null` (CPU) CPU

`/dev/random` CPU
/ ISP

Paul Robinson

FAQ

There are other methods. As every good sysadmin knows, it is part of standard practise to send data to the screen of interesting variety to keep all the pixies that make up your picture happy. Screen pixies (commonly mis-typed or re-named as 'pixels') are categorised by the type of hat they wear (red, green or blue) and will hide or appear (thereby showing the colour of their hat) whenever they receive a little piece of food. Video cards turn data into pixie-food, and then send them to the pixies - the more expensive the card, the better the food, so the better behaved the pixies are. They also need constant simulation - this is why screen savers exist.

To take your suggestions further, you could just throw the random data to console, thereby letting the pixies consume it. This causes no heat to be produced at all, keeps the pixies happy and gets rid of your data quite quickly, even if it does make things look a bit messy on your screen.

Incidentally, as an ex-admin of a large ISP who experienced many problems attempting to maintain a stable temperature in a server room, I would strongly discourage people sending the data they do not want out to the network. The fairies who do the packet switching and routing get annoyed by it as well.

5. 1993 make world

jkh commit (http://www.freebsd.org/cgi/cvsweb.cgi/src/Makefile.diff?r1=1.5;r2=1.6;f=h) Aug 13 22:47:28 1994 UTC make world

pttlz.bbs@ptt.cc ( ) Apr 7 17:09:47 2008 CST make world (15) freebsd 1993 pttlz make world 15

18.

1. FreeBSD

FreeBSD UNIX
FreeBSD BSD

Handbook (../handbook/bibliography-osinternals.html)

2. FreeBSD

Contributing to FreeBSD (../../articles/contributing/article.html)

3. SNAP RELEASE

/ FreeBSD CVS Repository (http://www.FreeBSD.org/cgi/cvsweb.cgi) (
)

`RELENG_5` *5-STABLE*

`RELENG_6` *6-STABLE*

`HEAD` *-CURRENT* *7.X-CURRENT*

`HEAD` branch tag symbolic constant *current*
( ) -CURRENT

-CURRENT 7.X `5-STABLE` (RELENG_5) 2004 10
-CURRENT fork `6-STABLE` (RELENG_6) 2005 11 -CURRENT fork

4. release

Release (../../articles/releng/article.html)

5. `make world` binary

`make world` binary
( )

`make world` `make install` `DESTDIR` binary
`${DESTDIR}` shared library binary
`make world`

6. Why isn't cvsup.FreeBSD.org a round robin DNS entry to share the load amongst the various CVSup servers?

While CVSup mirrors update from the master CVSup server hourly, this update might happen at any time during the hour. This means that some servers have newer code than others, even though all servers have code that is less than an hour old. If `cvsup.FreeBSD.org` was a round robin DNS entry that simply redirected users to a random CVSup server, running CVSup twice in a row could download code older than the code already on the system.

7. (bus speed defaulted)

Adaptec 1542 SCSI 1542
kernel TUNE_1542

8. current

CTM (../handbook/synching.html#CTM)

9. 240k

BSD `-b` byte

`/usr/src/Makefile`

```
bin-tarball:
(cd ${DISTDIR}; \
tar cf - . \
gzip --no-name -9 -c | \
split -b 240640 - \
${RELEASEDIR}/tarballs/bindist/bin_tgz.)
```

10. kernel

Contributing to FreeBSD (../../articles/contributing/article.html)

11. ISA

Frank Durda IV <uhclem@nemesis.lonestar.org>

PnP PnP I/O port
PnP PnP PnP port #
wired-OR yes bit
(Microsoft/Intel) X 0
X X
X-(limit/4) semi-binary search
PnP 2^64

ID 32-bit(2^64) + 8bit 32 bits
ID 32 bits

32 bits # 32 bits
32 bit 64 bits

32 bit binary search wired-OR

ID I/O port

ECU MLB BIOS ECU MLB BIOS PnP PnP BIOS ECU PnP

PnP I/O DMA IRQ

Microsoft port PnP I/O cycles PnP IBM printer board port MS port + `0x800` `0x200` `0x3ff` port

12. major number

`files.i386` kernel MAKEDEV(8) character major number 32 block major number 8 FreeBSD technical discussions (http://lists.FreeBSD.org/mailman/listinfo/freebsd-hackers)

13.

1983 FFS cylinder group find depth first search(aka ftw) archive inode cylinder group depth first search / cylinder group (/cylinder greoup) 10 restore (archive) ( ) (10) 10 cylinder group

Kirk McKusick, September 1998

14. kernel panics

[Bill Paul <`wpaul@FreeBSD.org`> FreeBSD-CURRENT mailing list Dag-Erling Smørgrav <`des@FreeBSD.org`>]

```
From: Bill Paul <wpaul@skynet.ctr.columbia.edu>
Subject: Re: the fs fun never stops
To: Ben Rosengart
Date: Sun, 20 Sep 1998 15:22:50 -0400 (EDT)
Cc: current@FreeBSD.org
```

[Ben panic]

```
> Fatal trap 12: page fault while in kernel mode
```

```
> fault virtual address   = 0x40
> fault code              = supervisor read, page not present
> instruction pointer     = 0x8:0xf014a7e5
                                ^^^^^^^^^^
> stack pointer           = 0x10:0xf4ed6f24
> frame pointer           = 0x10:0xf4ed6f28
> code segment            = base 0x0, limit 0xfffff, type 0x1b
>                         = DPL 0, pres 1, def32 1, gran 1
> processor eflags        = interrupt enabled, resume, IOPL = 0
> current process         = 80 (mount)
> interrupt mask          =
> trap number             = 12
> panic: page fault
```

instruction pointer
kernel image
snapshot GENERIC kernel kernel

1. instruction pointer `0x8:` `0xf0xxxxxx`
2.

```
% nm -n /( panic kernel ) | grep f0xxxxxx
```

`f0xxxxxx` instruction pointer
kernel symbol table symbol instruction pointer
instruction pointer

```
% nm -n /( panic kernel ) | grep f0xxxxx
```

panic

panic instruction pointer kernel symbol table

panic crash dump gdb(1) stack trace

1. kernel kernel debugger `options DDB` ( )
2. `config -g` *`KERNELCONFIG`*
3. `cd /sys/compile/` *`KERNELCONFIG`*`; make`
4. kernel
5. `make install`
6.

make(1) kernel `kernel` `kernel.debug` `kernel` `/kernel`
`kernel.debug` gdb(1) debugging symbols

crash dump `/etc/rc.conf` `dumpdev` swap rc(8) dumpon(8)
crash dump dumpon(8) panic crash dump savecore(8)
`/etc/rc.conf` `dumpdev` rc(8) savecore(8) crash dump
`/var/crash`

: FreeBSD crash dump 64MB crash dump
64MB `/var/crash` savecore(8) crash dump
crash dump kernel `options MAXMEM=(foo)`
kernel 128MB kernel
16MB crash dump 16MB 128MB

crash dump gdb(1) stack trace

```
% gdb -k /sys/compile/KERNELCONFIG/kernel.debug /var/crash/vmcore.0
(gdb) where
```

script(1) debug
symbol kernel panic stack strace
crash gdb(1)
crash

gdb(1)
gdb(1) kernel kernel

[Bill " DDB kernel DDB
'panic' panic (crash dump) panic
'continue' crash dump " -ed]

15. dlsym() ELF

ELF dynamic linker symbol `dlsym()`
`dlopen(NULL, flags)` handle symbol

`dlsym()` process symbol link ELF linker (ld(1))
`-export-dynamic`

16. kernel

FreeBSD 3.x kernel 256 MB FreeBSD 4.x 1 GB
(FTP HTTP) 256 MB
kernel
kernel

`src/sys/i386/include/pmap.h` `NKPDE` 1 GB

```
#ifndef NKPDE
```

```
#ifdef SMP
#define NKPDE                    254     /* addressable number of page tables/pde's */
#else
#define NKPDE                    255     /* addressable number of page tables/pde's */
#endif  /* SMP */
#endif
```

`NKPDE` (megabyte) 4 CPU 1 CPU 2

kernel 0x100100000 (byte) 1 GB 0xc0100000 `src/sys/i386/conf/Makefile.i386` LOAD_ADDRESS `src/sys/i386/conf/kernel.script` section location counter

```
OUTPUT_FORMAT("elf32-i386", "elf32-i386", "elf32-i386")
OUTPUT_ARCH(i386)
ENTRY(btext)
SEARCH_DIR(/usr/lib); SEARCH_DIR(/usr/obj/elf/home/src/tmp/usr/i386-unknown-freebsdelf/lib);
SECTIONS
{
  /* Read-only sections, merged into text segment: */
  . = 0xc0100000 + SIZEOF_HEADERS;
  .interp     : { *(.interp)    }
```

kernel ps(1) top(1) `make world` (`pmap.h` `/usr/include/vm/` `libkvm` ps(1) top(1))

kernel 4 megabytes

[David Greenman <`dg@FreeBSD.org`> *kernel* *2* *high order address bits* *256 MB* *]*

19.

FAQ FAQ Maintainer <faq@FreeBSD.org>

FreeBSD Core Team

Jordan K. Hubbard <jkh@FreeBSD.org>

FAQ

Doug White <dwhite@FreeBSD.org>

freebsd-questions

Jörg Wunsch <joerg@FreeBSD.org>

Usenet

Garrett Wollman <wollman@FreeBSD.org>

Networking and formatting

Jim Lowe

Multicast information

Peter da Silva <pds@FreeBSD.org>

FreeBSD FAQ

The FreeBSD Team

Kvetching, moaning, submitting data

FAQ

20. Ports and Packages

1. tarball

tarball

```
# make fetch
```

port editors/joe

```
# cd /usr/ports/editors/joe
# make fetch
```

editors/joe tarball /usr/ports/distfiles

ports ports tarball systuils/portupgrade

```
# cd /usr/ports/systuils/portupgrade
# make fetch-recursive
```

port port tarball /usr/ports/distfiles

ports tarball

```
# cd /usr/ports
# make fetch
```

ports tarball /usr/ports/distfiles

ftp ports tarball

```
# cd /usr/ports/ftp
# make fetch-recursive
```

ftp ports tarball /usr/ports/distfiles

2. tarball

patch editors/joe

```
# cd /usr/ports/editors/joe
# make extract
```

tarball /usr/ports/editors/joe/work

3. tarball patch

make extract patch patch
patch editors/joe

```
# cd /usr/ports/editors/joe
```

```
# make patch
```

tarball /usr/ports/editors/joe/work patch

4. port

port editors/joe

```
# cd /usr/ports/editors/joe
# make install
```

joe

make clean

```
# cd /usr/ports/editors/joe
# make clean
```

ports ports

```
# cd /usr/ports
# make clean
```

5. port (package)

package editors/joe

```
# cd /usr/ports/editors/joe
# make package
```

joe (package) package /usr/ports/editors/joe

```
# mkdir -p /usr/ports/packages
```

packages

make clean

```
# cd /usr/ports/editors/joe
# make package clean
```

6. port ports

make package ports port ports ports packages sysutils/portupgrade

```
# cd /usr/ports/sysutils/portupgrade
# make DEPENDS_TARGET=package package
```

portupgrade ports

7. port

editors/joe

```
# cd /var/db/pkg
# pkg_create -b joe-{ }
```

port /var/db/pkg

8. ports

port (work) editors/joe

```
# cd /usr/ports/editors/joe
# make clean
```

ports

```
# cd /usr/ports
# make clean
```

ftp

```
# cd /usr/ports/ftp
# make clean
```

9. ports tarball

make clean (work) ports tarball (tarball /usr/ports/distfiles) tarball editors/joe

```
# cd /usr/ports/editors/joe
```

```
# make distclean
```

make distclean **make clean** tarball
work

ports work tarball

```
# cd /usr/ports
# make distclean
```

ftp work tarball

```
# cd /usr/ports/ftp
# make distclean
```

10. ports

ports `mail/p5-Mail-SpamAssassin`

```
# cd /usr/ports/mail/p5-Mail-SpamAssassin
# make all-depends-list
```

make all-depends-list

```
# cd /usr/ports/mail/p5-Mail-SpamAssassin
# make pretty-print-build-depends-list
```

make all-depends-list

```
# cd /usr/ports/mail/p5-Mail-SpamAssassin
# make pretty-print-run-depends-list
```

make all-depends-list

11. ports

`editors/joe`

```
# cd /usr/ports/editors/joe
# make deinstall
```

pkg_delete

```
# pkg_delete joe-{version}
```

```
# pkg_delete -f joe-{version}
```

make deinstall pkg_delete

12. ports

sysutils/portupgrade

```
# cd /usr/ports/sysutils/portupgrade
# make deinstall-depends
```

make-depends-list

pkg_delete

-f

```
# pkg_delete -r portupgrade-{version}
```

make deinstall pkg_delete

13. ports

editors/joe

```
# cd /usr/ports/editors/joe
# make deinstall clean install
```

```
# cd /usr/ports/editors/joe
# make reinstall
```

14. ports

ports collection "ldap" ports

```
# cd /usr/ports
# make search key=ldap
```

ftp ports "ldap" ports

```
# cd /usr/ports/ftp
```

```
# make search key=ldap
```

key name ports
"ldap"

```
# cd /usr/ports
# make search name=ldap
```

15. ports

port `editors/joe`

```
# cd /usr/ports/editors/joe
# make clean reinstall
```

16.

```
# pkg_info
```

17.

joe

```
# pkg_info | grep joe
```

18.

`/usr/local/bin/joe`

```
# pkg_info -W /usr/local/bin/joe
```

FreeBSD

19.

`joe`

```
# pkg_info -L /var/db/pkg/joe-{version}
```

20. ports

CVS

FreeBSD ports CVS repository
Freshports (http://www.freshports.org/) FreeBSD Mailing FreeBSD cvs
(http://lists.freebsd.org/pipermail/cvs-all/) FreeBSD ports cvsweb
(http://www.freebsd.org/cgi/cvsweb.cgi/ports/)

CVS tag ports CVS tag
`/usr/share/examples/cvsup/ports-supfile` 2002/10/05

```
# vi /usr/share/examples/cvsup/ports-supfile
```

```
*default date=2002.10.05.00.00.00  #   date
```

CVSup csup CVSup csup (make update) ports collections
ports collections

ports `lang/perl5.8`
lang

```
# vi /usr/share/examples/cvsup/ports-supfile
```

```
#ports-all #   ports-all
```

```
ports-lang #
```

CVSup csup port

21. ports Mk

Mk (`/usr/ports/Mk`) ports ports collections Mk
Mk

```
# cd /usr/src
```

```
# make update
```

```
# cd /usr/src/share/mk
```

```
# make install
```

22. ports sed -i

BSD style sed BSD sed ports sed
sed_inplace (`textproc/sed_inplace`) ports

```
# make -DUSE_REINPLACE install
```

23. ports

ports collection ports collections

```
# pkg_version -c
```

24. ports

ports collections Makefile `sysutils/portupgrade` `/usr/ports/sysutils/portupgrade/Makefile`

```
# cd /usr/ports/sysutils/portupgrade
# make -DNOPORTDOCS install
```

ports NOPORTDOCS

ports ports

```
# cd /usr/ports/sysutils/portupgrade
# grep defined Makefile
```

25. ports

ports collections

ports collections

ftp/pure-ftpd inetd ports collections

```
# cd /usr/ports/ftp/pure-ftpd
# make CONFIGURE_ARGS+="--without-inetd" install
```

26. ports

ports collecions (`/usr/local/`)

`editors/joe`

```
# cd /usr/ports/editors/joe
# make PREFIX=/usr install
```

joe /usr /usr/local

27. ports FORCE_PKG_REGISTER

Ohaha FAQ (http://ohaha.ks.edu.tw/faq-0003.htm)

28. ports Shared object libintl.so.X not found

Ohaha FAQ (http://ohaha.ks.edu.tw/faq-0004.htm)

29. packages

FreeBSD packages .tgz packages pkg_add joe tgz

```
# pkg_add joe-{version}.tgz
```

30. packages

packages packages packages

packages packages

packages joe tgz

```
# pkg_add -f joe-{version}.tgz
```

31. packages packages

packages packages packages

packages packages portupgrade packages

```
# pkg_info -r portupgrade-{version}.tgz
```

32. packages

PACKAGEROOT

```
# setenv PACKAGEROOT ftp://ftp.tw.freebsd.org
```

PACKAGESITE packages

ftp.tw.freebsd.org i386 current pakcages

```
# setenv PACKAGESITE ftp://ftp.tw.freebsd.org/pub/FreeBSD/ports/i386/packages-current/Latest/
```

packages portupgrade

```
# pkg_add -r portupgrade
```

ftp.tw.freebsd.org portupgrade packages

33. INDEX

4.x INDEX `/usr/ports/INDEX` 5.x `/usr/ports/INDEX-5`

INDEX ports ports INDEX

```
# cd /usr/ports

# make index
```

INDEX

```
# cd /usr/ports

# make fetchindex
```

34. INDEX HTML

FreeBSD ports collection lynx, w3m, links ports collection

```
# cd /usr/ports

# make readmes
```

/usr/ports/ftp

```
# cd /usr/ports/ftp

# make readme
```

/usr/ports/ftp
README.html

35. ports ports update

ports ports update
`/usr/sup/refuse`

```
# ports/french

# ports/german

# ports/lang/perl5.8
```

36. make deinstall pkg_delete

make deinstall port

pkg_delete port pkg_delete wild card p ports

```
# cd /var/db/pkg
# pkg_delete p*
```

make deinstall ports collection pkg_delete

ports collections make deinstall pkg_delete pkg_delete

Bibliography

FreeBSD Unleashed, Michael Urban Brian Tiemann, Sams, 1st edition, 992 pages, October 2001, ISBN 0-67232-206-4.

4.4BSD System Manager's Manual, Computer Systems Research Group, University of California, Berkeley, O'Reilly and Associates, 1st edition, June 1994, 804 pages, ISBN 1-56592-080-5.

4.4BSD User's Reference Manual, Computer Systems Research Group, University of California, Berkeley, O'Reilly and Associates, 1st edition, June 1994, 905 pages, ISBN 1-56592-075-9.

4.4BSD User's Supplementary Documents, Computer Systems Research Group, University of California, Berkeley, O'Reilly and Associates, 1st edition, June 1994, 712 pages, ISBN 1-56592-076-7.

4.4BSD Programmer's Reference Manual, Computer Systems Research Group, University of California, Berkeley, O'Reilly and Associates, 1st edition, June 1994, 866 pages, ISBN 1-56592-078-3.

4.4BSD Programmer's Supplementary Documents, Computer Systems Research Group, University of California, Berkeley, O'Reilly and Associates, 1st edition, June 1994, 596 pages, ISBN 1-56592-079-1.

The Design and Implementation of the 4.4BSD Operating System, M. K. McKusick, Kirk Marshall, Keith Bostic, Michael J Karels, John Quarterman, Addison-Wesley, Reading, 1996, ISBN 0-201-54979-4.

The Design and Implementation of the FreeBSD Operating System, M. K. McKusick George V. Neville-Neil, Addison-Wesley, Boston, 2004, ISBN 0-201-70245-2.

Unix System Administration Handbook, Evi Nemeth, Garth Snyder, Scott Seebass, Trent R. Hein, John Quarterman, Prentice-Hall, 3rd edition, 2000, ISBN 0-13-020601-6.

The Complete FreeBSD, Greg Lehey, Walnut Creek, 3rd edition, June 1999, 773 pages, ISBN 1-57176-246-9.

[McKusick et al, 1994] *Berkeley Software Architecture Manual, 4.4BSD Edition*, M. K. McKusick, M. J. Karels, S. J. Leffler, W. N. Joy, R. S. Faber, 5:1-42.

FreeBSD for PC 98'ers (in Japanese), SHUWA System Co, LTD., ISBN 4-87966-468-5 C3055 P2900E.

FreeBSD (in Japanese), CUTT, ISBN 4-906391-22-2.

Complete Introduction to FreeBSD (in Japanese), Shoeisha Co., Ltd, ISBN 4-88135-473-6 P3600E.

Personal UNIX Starter Kit FreeBSD (in Japanese), ASCII, ISBN 4-7561-1733-3 P3000E.

FreeBSD Handbook (Japanese translation), ASCII, ISBN 4-7561-1580-2 P3800E.

FreeBSD mit Methode (in German), Computer und Literature Verlag/Vertrieb Hanser, 1998, ISBN 3-932311-31-0.

FreeBSD install and Utilization Manual (in Japanese), Mainichi Communications Inc..

Building Internet Server with FreeBSD (in Indonesia Language), Elex Media Komputindo, Onno W Purbo, Dodi Maryanto, Syahrial Hubbany, Widjil Widodo.

The FreeBSD Corporate Networker's Guide, Addison-Wesley.

UNIX in a Nutshell, O'Reilly & Associates, Inc., 1990, ISBN 093717520X.

What You Need To Know When You Can't Find Your Unix System Administrator, O'Reilly & Associates, Inc., 1995, Linda Mui, ISBN 1-56592-104-6.

FreeBSD User's Reference Manual (Japanese translation), Mainichi Communications Inc., Jpman Project, Japan FreeBSD Users Group, 1998, ISBN 4-8399-0088-4 P3800E.

Online Guide for newcomers to the UNIX environment (http://unixhelp.ed.ac.uk/), Edinburgh University (http://www.ed.ac.uk/).

DNS and BIND, O'Reilly & Associates, Inc, ISBN 1-56592-512-2, Paul Albitz Albitz Cricket Liu, 1998, 3rd edition.

Sendmail, O'Reilly & Associates, Inc, 1997, 2nd edition, Brian Costales, ISBN 1-56592-222-0.

Essential System Administration, Æleen Frisch, 2nd edition, O'Reilly & Associates, 1995, ISBN 1-56592-127-5.

TCP/IP Network Administration, Craig Hunt, 2nd edition, O'Reilly & Associates, Inc, 1997, ISBN 1-56592-322-7.

Managing NFS and NIS, Hal Stern, O'Reilly & Associates, Inc, 1991, ISBN 0-937175-75-7.

FreeBSD System Administration's Manual (http://www.pc.mycom.co.jp/FreeBSD/sam.html), Jpman Project, Japan FreeBSD Users Group (http://www.jp.FreeBSD.org), Mainichi Communications Inc. (http://www.pc.mycom.co.jp/), 1998, ISBN 4-8399-0109-0 P3300E.

X Window System Toolkit, Digital Press, Paul Asente, ISBN 1-55558-051-3.

C: A Reference Manual, Prentice Hall, 1995, 4th edition, Samuel P. Harbison Guy L. Jr. Steele, ISBN 0-13-326224-3.

The C Programming Language, Prentice Hall, 1998, Brian Kernighan Dennis Ritchie, ISBN 0-13-110362-9.

Porting UNIX Software, Greg Lehey, O'Reilly & Associates, Inc., 1995, ISBN 1-56592-126-7.

The Standard C Library, Prentice Hall, 1992, P. J. Plauger, ISBN 0-13-131509-9.

Advanced Programming in the UNIX Environment, Addison-Wesley, 1992, W. Richard Stevens, ISBN 0-201-56317-7.

UNIX Network Programming, W. Richard Stevens, Prentice Hall, 1998, 2nd edition, ISBN 0-13-490012-X.

Writing Serial Drivers for UNIX, Bill Wells, December 1994, Dr. Dobb's Journal, pp68-71, pp97-99.

UNIX System Architecture, Prentice-Hall, Inc, 1990, Prabhat K. Andleigh, ISBN 0-13-949843-5.

Porting UNIX to the 386, William Jolitz, Dr. Dobb's Journal, January 1991-July 1992.

TCP/IP Illustrated, Volume 1: The Protocols, W. Richard Stevens, Addison-Wesley, 1996, ISBN 0-201-63346-9.

Unix Systems for Modern Architectures, Addison-Wesley, Curt Schimmel, 1994, ISBN 0-201-63338-8.

TCP/IP Illustrated, Volume 3: TCP for Transactions, HTTP, NNTP and the UNIX Domain Protocols, Addison-Wesley, 1996, W. Richard Stevens, ISBN 0-201-63495-3.

UNIX Internals -- The New Frontiers, Uresh Vahalia, Prentice Hall, 1996, ISBN 0-13-101908-2.

TCP/IP Illustrated, Volume 2: The Implementation, Gary R. Wright W. Richard Stevens, 1995, Addison-Wesley, ISBN 0-201-63354-X.

Firewalls and Internet Security: Repelling the Wily Hacker, William R. CHeswick Steven M. Bellovin, Addison-Wesley, 1995, ISBN 0-201-63357-4.

Practical UNIX Security, Simson Garfinkel Gene Spafford, 1996, 2nd edition, O'Reilly & Associates, Inc, ISBN 1-56592-148-8.

PGP Pretty Good Privacy, Simson Garfinkel, O'Reilly & Associates, Inc, 1995, ISBN 1-56592-098-8.

Pentium Processor System Architecture, Don Anderson Tom Shanley, Addison-Wesley, 1995, 2nd edition, ISBN 0-201-40992-5.

Programmer's Guide to the EGA, VGA, and Super VGA Cards, Richard F. Ferraro, 3rd edition, Addison-Wesley, 1995, ISBN 0-201-62490-7.

80486 System Architecture, Tom Shanley, Addison-Wesley, 1995, 3rd edition, ISBN 0-201-40994-1.

ISA System Architecture, Tom Shanley, Addison-Wesley, 3rd edition, 1995, ISBN 0-201-40996-8.

PCI System Architecture, Tom Shanley, Addison-Wesley, 1995, 3rd edition, ISBN 0-201-40993-3.

The Undocumented PC, Frank Van Gilluwe, Addison-Wesley, 1994, ISBN 0-201-62277-7.

Bell System Technical Journal, Unix Time-Sharing System, American Telephone & Telegraph Company, July-August 1978, Vol 57, No 6, Part 2, ISSN0005-8580.

Lion's Commentary on UNIX, John Lion, ITP Media Group, 1996, 6th edition, ISBN 1573980137.

The New Hacker's Dictionary, Eric S. Raymond, MIT Press, 1996, 3rd edition, ISBN 0-262-68092-0.

A quarter century of UNIX, Peter H. Salus, Addison-Wesley, 1994, ISBN 0-201-54777-5.

The UNIX-HATERS Handbook, Steven Strassman, Daniel Weise, Simon Garfinkel, IDG Books Worldwide, Inc, 1994, ISBN 1-56884-203-1.

Life with UNIX special edition, Don Libes Sandy Ressler, Prentice-Hall, 1989, ISBN 0-13-536657-7.

The BSD Family Tree (http://svnweb.FreeBSD.org/base/head/share/misc/bsd-family-tree?view=co), 1997.

Absolute BSD, Michael Lucas, No Starch Press, June 2002, ISBN 1-886411-74-3.

The C/C++ Users Journal, R&D Publications Inc., ISSN 1075-2838.

Sys Admin The Journal for UNIX System Administrators, Miller Freeman, Inc, ISSN 1061-2688.