NetBSD/amiga FAQ

List of Frequently Asked Questions and Answers for NetBSD/amiga

February 12th, 1996

by Hubert Feyrer, et al.

1 Introduction

1.1 About this document

1.1.1 History

Original document by Mike Schwartz `<mykes@shell.portal.com>`.

Idea of using TeXInfo by Carsten Hammer `<chammer@dave.hrz.uni-bielfeld.de>`.

Second revision (completely rewritten) by Guenther Grau `<s_grau@ira.uka.de>`.

Third revision (restructured) by Tim Walls `<tjw1@doc.ic.ac.uk>`.

Fourth revision (updates for 1.1) by Hubert Feyrer `<hubert.feyrer@rz.uni-regensburg.de>`. Thanks to Thorsten Frueauf, Frank Neumann and Bernd Ernesti for proofreading, inspirations and contributions.

The FAQ is currently maintained by Hubert Feyrer `<hubert.feyrer@rz.uni-regensburg.de>`.

1.1.2 Copyright

1.1.3 Distribution

Whenever this FAQ is updated, notes about where to find the new version will be postet to the following mailing-lists and news-groups: `netbsd-announce@netbsd.org`, `port-amiga@netbsd.org`, `comp.unix.amiga` , `de.comp.sys.amiga.unix` and `comp.unix.bsd.netbsd.announce`. You can find it in various formats under '`/pub/NetBSD-Amiga/docs/NetBSD-Amiga-FAQ`' on ftp.uni-regensburg.de and its mirrors. The FAQ is also available via the World Wide Web at http://rfhs1012.fh.uni-regensburg.de/~feyrer/NetBSD/AFAQ/ .

1.2 Introduction to the FAQ

Welcome to the NetBSD-Amiga-FAQ!

This document contains a bunch of useful infomation about NetBSD on the Amiga. It was mainly created for people who never used NetBSD on the Amiga before, but it contains a lot of valuable information for all users, especially in the last chapter. This last chapter contains a list of Frequently Asked Questions (FAQs) and their respective answers.

In general the people on the Internet are very helpful, so if you have a problem when using NetBSD, don't be shy and ask for help. But ...

BE SURE YOU HAVE READ AN UP-TO-DATE VERSION OF THIS FAQ FIRST!!!

It is really annoying and time consuming, if a lot of people ask the same questions and we have to answer them over and over again. See Chapter 4 [Where can I find more information on NetBSD?], page 9, for information on how to get an up-to-date version of this document. Also, please redistribute this FAQ, so that it is easier for people who are interested in NetBSD to find it.

Please write all comments, suggestions, spelling corrections, etc. to the current maintainter of the FAQ. This will help to keep this FAQ up-to-date and useful. Thank you for reading the FAQ, because that's why we put so many hours of work into it! You will find the name and the email-address of the current maintainer of this document on the second page. Later on all references to the author of this FAQ will be name- and address-less, for the ease of maintainance. If you ever need to contact the author, see the second page for his name and email-address.

2 What is NetBSD?

This chapter contains general information about NetBSD.

2.1 The name of the game

NetBSD is a Unix-like operating system. It is very portable and runs on multiple architectures such as Amiga, Atari, HP300, Mac, Sun3, Sparc, PC, VAX, Mips, Alpha, ... It is developed by people all around the world on the Internet, therefore it's called *Net*BSD. It has all the features you would expect in a modern Unix, including true (preemtive) multitasking, virtual memory, shared libraries, demand loading, shared copy-on-write executables and TCP/IP networking.

For a complete description of a Unix-like operating system, please refer to some introductionary books about Unix. If you have access to Usenet, get any of the comp.unix.* Newsgroups FAQs.

See Chapter 4 [Where can I find more information on NetBSD?], page 9, for information on how to get more and detailed information on NetBSD.

2.2 Who is working on the port?

NetBSD was ported to the Amiga by Markus Wild `<mw@eunet.ch>`, who also initially ported GCC to AmigaDOS and who did the ixemul.library. Once NetBSD-Amiga was useable, a lot of people joined in and contributed a lot to the project.

Here's an attempt to list them all (in alphabetic order):

- Alan Bair <abair@amcu-tx.sps.mot.com>: Fontdumper, Rootfs.
- Stefan G. Berg <sgberg@charon.bloomington.in.us>: Writing documentation.
- Dave Blaszyk <dvb@ssd.kodak.com>: a driver for IVS Vector SCSI board
- Klaus Burkert <crest@arkon.dontpanic.sub.org>: Picasso II and Ariadne-support
- Philippe Brand <PhB@telesys-innov.fr>: X-server for custom chips, X-server for Retina.
- David Crooke <dcc@dcs.ed.ac.uk>: Writing an installation-document.
- William J. Coldwell <billc@warped.com>: Maintaining the Mailing-List in the old days, Various programming.
- Bernd Ernesti <bernd@arresum.inka.de>: Debugging WarpEngine's SCSI-driver and Ariadne ethernet driver, Xcl, Picasso II and CyberVision'64 drivers, compiling X11R6-snapshots.
- Hubert Feyrer <hubert.feyrer@rz.uni-regensburg.de>: Writing documentation, maintainer of ftp.uni-regensburg.de:/pub/NetBSD-Amiga.
- Thorsten Frueauf <frueauf@ira.uka.de>: Updating NetBSD-Amiga-X-FAQ
- Guenther Grau <s_grau@ira.uka.de>: Extensively writing documents.
- Niklas Hallqvist <niklas@appli.se>: Porting the Mach VM system, that NetBSD-Amiga uses, driver for GVP, ADos-FS, GoldenGate (ISA) support.
- Rob Healey <rhealey@aggregate.com>: a LOT of bug-fixes.
- Andy Heffernan <ahh@netcom.com>: Porting GDB, compiling X11R5.

- Michael L. Hitch <osymh@montana.edu>: Support for the 68040, drivers for various SCSI-devices, changing loadbsd, fixing bugs, 5380-Driver, AGA-modes for the console and X, A4000 IDE driver, a LOT of bug-fixes, implementing a floppy-driver, IVS-Vector-driver etc.
- Chris Hopps <chopps@emunix.emich.edu>: Support for the custom chips, reintegrating the changes back to the NetBSD-current source tree, fixing bugs and (re-)designing some of the code. Main coordinator of the source and official maintainer of the NetBSD/amiga port. ADos-Filesystem.
- Eduardo E. Horvath <eeh@btr.com>: X11R5-Mono-Server.
- Markus Illenseer <markus@tiger.teuto.de>: Writing documentation, Picasso- and Ariadne-support, Porting X-clients, Gateway!-CD
- Oliver Lahaye <lahaye_o@epita.fr>: X-server for custom chips, X-server for Retina.
- Kari Mettinen <Kari.Mettinen@helsinki.fi>: Xcl and Picasso II support.
- Brad Pepers <pepersb@cuug.ab.ca>: Implementing a floppy-driver.
- Oliver Raoul <raoul_o@epita.fr>: X-server for custom chips, X-server for Retina.
- Stephen J. Roznowski <sjr@zombie.ncsc.mil>: Building binary-snapshots from sun-lamp sources.
- Ty Sarna <tsarna@endicor.com>: Various programming, Installation.
- Ignatios Souvatzis <is@beverly.rhein.de>: A2060 (Arcnet) and A2410 (TIGA gfx card) support
- Ezra Story <ezy@panix.com>: Spektrum support
- Michael Teske <teske@wotan.desy.de>: CyberVision'64 support
- Roy Trevino <rtrevino@sedona.intel.com>: X11R5-Mono-Server.
- Lutz Vieweg <lkv@mania.RoBIN.de>: providing information and writing a tty-device-driver for the Retina Z2
- Markus Wild <mw@eunet.ch>: Initially started the port of NetBSD-current to the Amiga, Xamiga24, still working on various projects.

I hope I didn't forget anyone. If so, please tell me!

3 What do I need to run NetBSD/amiga

This chapter describes the hardware related topics of NetBSD. It will explain, what kind of hardware you need to be able to run NetBSD on your Amiga.

In general, there are two major requirements your system must fulfill to be able to run NetBSD. You must have an apropriate CPU and enough memory. To comfortably run NetBSD you should dedicate some harddiskspace to NetBSD.

3.1 Which CPU is needed?

It depends heavily on the type of CPU your system has, whether or not NetBSD runs on your Amiga. NetBSD, as any other modern Unix-derivate, too, must have a memory-management-unit (MMU) to be able to run. This is needed as all the programs that run under Unix are separated from each other, so no program can do any harm to any other.

You need to have one of the following processors (CPUs) to be able to run NetBSD:

- 68020 with 68851 (MMU) and 68881 or 68882 (FPU)
- 68030 with 68881 or 68882 (FPU)
- 68040

Note, that it is very important that you have a *real* CPU, i.e. not an EC- or LC-version. These versions are missing the MMU- and/or FPU-part of the chip and are therefore cheaper. It is not possible to simply add a 68851 (MMU) or 68881/68882 (FPU) to these chips, if your CPU-board doesn't provide a place for it. If you have an 68030EC, you can however replace it by a 68030, or you have to buy a seperate CPU-board or accelerator-board equipped with one of the CPUs mentioned above to be able to run NetBSD.

NetBSD/amiga does currently **not** support the 68060!

Also note, that you currently need a floating-point-unit (FPU) to be able to run NetBSD. The kernel itself, that is the core part of NetBSD, does not need an FPU, but some of the essential utilities do. Other ports of NetBSD on other architectures have an FPU-emulator implemented in the kernel, so they can do without an FPU. So, if someone implements an FPU-emulator for NetBSD on the 680x0-architecture, the need for a seperate FPU will eventually vanish, but for now you need an FPU.

3.2 How much memory do I need?

Theoretically, you only need 2MB of FAST-RAM and 1MB of CHIP-RAM. The 1MB CHIP-memory is only needed due to the way NetBSD is currently booted. The kernel is loaded into the CHIP-memory and then copied into the largest chunk of fast memory found. Therefore 2MB is the bare minimum and will not allow you to do anything useful, but booting.

Note that the complete GENERIC kernel (including drivers for every device supported) is over 1MB in size; you can't load this kernel with only 1MB of CHIP RAM, 2MB is required. The basic INSTALL kernel will load with 1MB CHIP fine however.

It is possible to run NetBSD in a low memory situation, because it uses a mechanism called paging. It writes currently unused parts of programs to the disk, thus freeing some memory for other parts. Writing to and reading from disk is very slow, compared to the execution of programs in memory, so the system will be too slow to be useable. So, to have an actual useable system, you should have at least 4MB of FAST-RAM, preferably more. When you want to use X-Windows, a graphical user interface on top of NetBSD, you probably need about 8MB or even more to do anything useful.

3.2.1 Non contiguous memory

With the Amiga's architecture being what it is, it is often the case that the memory map will have "holes" in it - say, 1MB of CHIP RAM, followed by a gap, 4MB of 16bit RAM, a gap, and then in (very) high memory a block of 32bit RAM; this is often the case with 68000 machines upgraded to an 030 or above.

To use more then one such chunk of memory, you need a kernel that was build with the following line in the config-file (See Chapter 6 [How do I build a kernel myself?], page 21, for more information on how to compile your own kernel):

```
options MACHINE_NONCONTIG # Non-contiguous memory support
```

Besides that, you have to tell loadbsd to use all memory chunks, which you do by passing it '`-n 2`' when booting NetBSD. It will then use all available memory segments.

3.3 How much diskspace do I need?

Theoretically, you can do without any harddiskspace. It is possible to boot NetBSD from floppy-disks, but you cannot do any useful work with such a setup. To get an actually useable system, you will need to dedicate some harddiskspace to NetBSD. How much you actually need depends heavily on what you want to do with NetBSD.

A small, but complete working system needs about 50 to 80MB. It should be clear, that the more goodies you want to add, the more space you need.

To prevent you from running out of memory, you should use about twice your RAM amount just for a swap partition on your drive (or about three times if you intend to use the X-Windows system). That is where NetBSD writes the parts of running programs which are currently not used to free up some main-memory for other programs.

Add another, say, 30M it you want to recompile your own kernels from source. (This can be stripped down to less then 20M if you delete the source for ports you don't want, but you'd have to unpack the full source first!).

The X window system binaries take another 30M of harddisk space.

3.4 What other hardware requirements are there?

There are no other hardware requirements to run NetBSD, but you are free to use additional devices, provided they are supported.

There is a general scheme on how NetBSD deals with devices. To get a device working with NetBSD you have to provide a so called 'driver' for this device. This is a hardwaredependent part of the kernel, which knows how to access the particular device. There are already device-drivers for the most commonly used devices, but there are still some device-drivers to be written.

It is a difficult task to add support for a new device. So you need to have good programming skills and need to know a lot about how the kernel deals with devices. If you think that you are an experienced programmer and want to add support for a new device, you need to be able to recompile the kernel yourself. See Chapter 6 [How do I build a kernel myself?], page 21, for information on how to do this. See Chapter 4 [Where can I find more information on NetBSD?], page 9, for information on how to get in contact with other developers.

If you use a device that's not supported, there is one danger: the device may generate (hardware-) interrupts for which there are no handlers and which will therefor crash the machine! So, if you use some card which isn't supported or you aren't sure if it is, and your machine locks up *early while booting*, try removing that card.

A tape drive is recommended for NetBSD, as for any other operating system, too. You should do regularly backups in order to be able to restore the data you might have accidentially destroyed. Especially when you are doing kernel-development, chances are, that you may trash your harddisk. The system is normally quite stable, so doing backups is not a must but a sensible option.

3.5 What hardware is supported by NetBSD/amiga V1.1, anyways?

Here's what the '`INSTALL`'-file says. Supported devices include:

- A4000/A1200 IDE controller.
- SCSI host adapters:
  - 33c93 based boards: A2091, A3000 and GVP series II.
  - 53c80 based boards: 12 Gauge, IVS and Wordsync/Bytesync.
  - 53c710 based boards: A4091, Magnum, Warp Engine and Zeus.
  - FAS216 based boards: FastLane Z3, Blizzard.
- Video controllers:
  - ECS, AGA and A2024 built in on various amigas.
  - Retina Z2 and Retina Z3.
  - Picasso II.
  - GVP Spectrum.
  - Piccolo.
  - A2410.
  - CyberVision 64.

- Ethernet controllers:
  - A2065 Ethernet
  - Hydra Ethernet
  - ASDG Ethernet
  - A4066 Ethernet
  - Ariadne Ethernet
  - Quicknet Ethernet
- Arcnet controllers:
  - A2060 Arcnet
- Tape drives: Most SCSI tape drives, including Archive Viper, Cipher SCSI-2 ST150.
- CD-ROM drives: Most SCSI CD-ROM drives
- Serial cards:
  - MultiFaceCard II and III
  - A2232
- Amiga floppy drives.
- Amiga parallel port.
- Amiga serial port.
- Amiga mouse.

Experimental versions of new drivers can sometimes be found in on ftp.uni-regensburg.de and its mirrors in '`/pub/NetBSD-Amiga/experimental`'.

3.6 Things currently *not* supported by NetBSD/amiga

People have asked for the following things to be supported in NetBSD/amiga, but so far, you're out of luck:

- 68060 CPU, either on the Blizzard A2060, the CyberStorm or some other 060 board.
- Octagon SCSI controller
- Golem SCSI controller
- Evolution SCSI controller
- Cobra Ferret SCSI controller
- Blizzard III and IV SCSI controller
- ALF 2/3 SCSI controller
- Rapidfire SCSI controller
- the A1200's PCMCIA-slot and any card using it
- Networking via AmigaLink & PLIP

Please don't bother people about support for those devices. As soon as they are available, they'll be removed from the above list.

4 Where can I find more information on NetBSD?

This chapter contains information on how to get more information on NetBSD. There are several ways to get the desired information. Choose the one which uses the least resources. To do this, simply use the first method described here which is appropriate for you.

4.1 UseNet News

If you have access to Usenet, subscribe to (some of) the following news-groups:

- `comp.unix.amiga` : Discussions, Questions and anwers about any on Amiga (NetBSD, Linux, Amix, AmiNIX, Minix, etc.).
- `de.comp.sys.amiga.unix`: Basically the same as above, but in German language.
- `comp.unix.bsd.netbsd.announce`: NetBSD-related announcements, all ports. Dave Burgess (`burgess@s069.infonet.net`) posts his excellent 386BSD-FAQ here, also available as http://cynjut.neonramp.com/FAQ.html via WWW. Have a look at it!
- `comp.unix.questions`: For general questions on Unix. They got an excellent FAQ there!
- `comp.unix.bsd.netbsd.misc`: Discussion on NetBSD on all ports, not just amiga, questions & answers.

4.2 Anonymous FTP

If you don't have access to Usenet, but have the possibility to get files from the Internet with the File Transfer Protocol (FTP), then get the latest FAQ from `rtfm.mit.edu`. This is the place where all FAQs from all news-groups are collected and archived.

To find general information on NetBSD, try the following sites:

- NetBSD.rmit.edu.au:`/pub/NetBSD/NetBSD-1.1`
- ftp.ms.mff.cuni.cz:`/MIRRORS/sup.netbsd.org/pub/NetBSD/NetBSD-1.1`
- ftp.uni-paderborn.de:`/pub/NetBSD/NetBSD-1.1`
- ftp.uni-regensburg.de:`/pub/NetBSD-1.1`
- ftp.iastate.edu:`/pub/netbsd/NetBSD-1.1`
- wuarchive.wustl.edu:`/systems/unix/NetBSD/NetBSD-1.1`
- ftp.luth.se:`/pub/NetBSD/packages/binaries/NetBSD-1.1`
- ftp.demon.co.uk:`/pub/unix/NetBSD/NetBSD-1.1`
- sunsite.doc.ic.ac.uk:`/packages/NetBSD-1.1`

Additional information on NetBSD/amiga is available on `ftp.uni-regensburg.de` and its mirrors, look into the directory '`/pub/NetBSD-Amiga/docs`' . This site carries lots of things related to NetBSD/amiga. Here's a full list of all known mirrors:

- ftp.luth.se:`/pub/NetBSD/NetBSD-Amiga`

- ftp.hrz.uni-kassel.de:`/pub/machines/amiga/NetBSD`
- ftp.uni-stuttgart.de:`/pub/systems/amiga/netbsd`
- ftp.uni-erlangen.de:`/pub/amiga/unix/NetBSD`
- ftp.uni-trier.de:`/pub/unix/systems/NetBSD-Amiga`
- ftp.germany.eu.net:`/pub/comp/amiga/NetBSD-Amiga`
- ftp.uni-augsburg.de:`/pub/amiga/NetBSD`
- ftp.funet.fi:`/index/NetBSD-Amiga`
- ftp.cs.umn.edu:`/pub/NetBSD-Amiga`
- ftp.warped.com:`/pub/NetBSD-Amiga`
- ftp.unina.it:`/pub/amiga/NetBSD-Amiga`
- ftp.nvg.unit.no:`/pub/NetBSD-Amiga`

4.3 World Wide Web

This document is available via the World Wide Web (Mosaic, Netrape, lynx etc.) at http://rfhs1012.fh.uni-regensburg.de/~feyrer/NetBSD/AFAQ/ .

The following URLs point to sites that hold information or files relating to NetBSD, including this FAQ. (Note that `ftp.uni-regensburg.de` is the home site for all NetBSD-Amiga specific files).

- http://www.netbsd.org/
- ftp://ftp.uni-regensburg.de/pub/NetBSD-Amiga/.index.html
- http://www4.informatik.uni-erlangen.de/~roessler/NetBSD-Amiga/
- http://rfhs1012.fh.uni-regensburg.de/~feyrer/NetBSD/
- http://cynjut.neonramp.com/FAQ.html

4.4 Internet Relay Chat

If you have access to IRC (Internet Relay Chat), try to join the channels '`#amiga`' or '`#amigager`'. Discussions on the former one are in English, on the latter one in German. These channels are not dedicated to NetBSD, but you are likely to find some of the developers and users of NetBSD there.

A dedicated NetBSD channel also exists, suprisingly called '`#NetBSD`'.

4.5 Electronic Mailing Lists

There are several mailing-lists available for NetBSD . To get information on these mailing-lists, use your favourite mailer and send the following message:

```
To: majordomo@netbsd.org
Subject: Don't care, isn't processed.
help
```

```
end
```

You will receive an answer that might look like this:

```
To: s_grau@ira.uka.de   <– This will be replaced with your email-address.
From: majordomo@netbsd.org
Subject: Majordomo results
Reply-To: majordomo@netbsd.org

--

>>>> help
This is Brent Chapman's "Majordomo" mailing list manager, version 1.92.

In the description below items contained in []'s are optional. When
providing the item, do not include the []'s around it.

It understands the following commands:

    subscribe <list> [<address>]
        Subscribe yourself (or <address> if specified) to the named <list>.

    unsubscribe <list> [<address>]
        Unsubscribe yourself (or <address> if specified) from the named <list>.

    get <list> <filename>
        Get a file related to <list>.

    index <list>
        Return an index of files you can "get" for <list>.

    which [<address>]
        Find out which lists you (or <address> if specified) are on.

    info <list>
        Retrieve the general introductory information for the named <list>.

    lists
        Show the lists served by this Majordomo server.

    help
        Retrieve this message.

    end
        Stop processing commands (useful if your mailer adds a signature).

Commands should be sent in the body of an email message to
"majordomo@NetBSD.ORG".

Commands in the "Subject:" line NOT processed.
```

```
If you have any questions or problems, please contact
"owner-majordomo@NetBSD.ORG".
```

If you send the command `lists` to majordomo, you will currently receive a mail like this:

```
...
>>>> lists
majordomo@NetBSD.ORG serves the following lists:

  Old
  current-users       General discussion regarding NetBSD-current
  m68k                Technical discussion regarding all NetBSD/m68k ports
  macbsd-development  Technical discussion regarding NetBSD/mac
  macbsd-general      General discussion regarding NetBSD/mac
  netbsd-announce     Moderated list for announcements only
  netbsd-bugs         Bug reports
  netbsd-help         Questions and answers
  netbsd-ports
  netbsd-users        General discussion
  port-alpha          Technical discussion regarding NetBSD/alpha
  port-amiga          Technical discussion regarding NetBSD/amiga
  port-atari          Technical discussion regarding NetBSD/atari
  port-hp300          Technical discussion regarding NetBSD/hp300
  port-i386           Technical discussion regarding NetBSD/i386
  port-m68k           Technical discussion regarding all NetBSD/m68k ports
  port-mac68k         Technical discussion regarding NetBSD/mac68k
  port-pc532          Technical discussion regarding NetBSD/pc532
  port-pmax           Technical discussion regarding NetBSD/pmax
  port-sparc          Technical discussion regarding NetBSD/sparc
  port-sun3           Technical discussion regarding NetBSD/sun3
  port-vax            Technical discussion regarding NetBSD/vax
  portable-ppp        Technical discussion regarding user-space PPP
  source-changes      Source tree changes
  tech-install        Technical discussion regarding installation tools
  tech-kern           Technical discussion regarding all aspects of the kernel
  tech-misc           Technical discussion not appropriate for the other tech-* lists
  tech-net            Technical discussion regarding networking software
  tech-ports          Technical discussion regarding all ports
  tech-userlevel      Technical discussion regarding user-level programs

Use the 'info <list>' command to get more information
about a specific list.
```

The '`port-amiga`' is for all discussion of Amiga-related problems. There used to be three lists ('`amiga`', '`amiga-dev`' and '`amiga-x`'), but they aren't used any more.

Besides '`port-amiga`', it is highly recommended to get on the '`netbsd-announce`' lists, because you'll be informed about official releases, etc. there.

To do so, your email would look like this:

```
To: majordomo@netbsd.org
Subject: None
subscribe port-amiga
subscribe netbsd-announce
end
```

You will receive an answer-mail from majordomo, that it successfully processed your subscription and that you are on the desired mailing-lists.

If you're interrested in development or want to be up to date with what's going on, join 'current-users', too.

Note that there is some traffic on all of these lists, so expect a few mails more to arrive in your mailbox each day. It is recommended, but not required, that you get on a list before sending any mail to it, because most of the answers are simply sent to the list and not to the sender. Please note also that most of these mailing lists are archived , so do not send things to them if you do not wish your messages to be archived.

Be sure to have read the latest version of this document, before sending any mail to these lists!

To send any mail to the lists, simply replace majordomo in the address with the list you want to address. For example, if you would like to write a mail to the amiga-list, send your mail to port-amiga@netbsd.org.

4.6 Useful books

There aren't many books which can help you with (say) possible kernel bugs or installation problems, but administering any Unix system is not a trivial task. Luckily, there is plenty of paper out there about the assorted jobs that befall a Unix sysadmin.

4.6.1 Basic System Administration

The books listed below deal with Unix and Unix system administration in general.

- Unix Unleashed

  'The Only UNIX Book You'll Ever Need!' is written on the cover page of this 1600-pages-book, and indeed it covers about every topic one could think of. The book consists of many small introductory texts to various topics, a reference card for common keystrokes in vi and emacs, pictures of some of the authors (ever wondered what the author of NetBSD's UUCP package looks like? :-), plus a CD that doesn't have a rockridge extension. The CD includes sources of the more common Unix tools described and mentioned in the book itself. All in all, this book's worth having a look at if you don't want to buy several books.

  Contents: What is an operating system, the Unix philosophy, getting started, the Unix filesystem, working with files & directories, listing files, some popular tools (dc, cal, su, man, grep, compress, tee, touch), editors (vi, emacs, sed), using network (rlogin, rcp, ftp, uucp), communication with other users (email, news, talk, irc), in depth introductions to various shells

(Bourne, Korn, C), introduction do several programming languages (awk, perl, C), process control (concept, ps, kill, cron), text formatting (nroff, troff, macro packages, tbl, eqn, pic, grap), RCS, SCCS, tar, shar, backups (dump, restore), system startup & shutdown, file system administration, user administration, networking, accounting, performance monitoring, kernel tuning, administration of mail, news & uucp, running INN, a discussion on system security, unix flavours, graphical user interfaces and even some Motif-programming.

Now, if this doesn't at least *mention* something you're interrested in, you're in the wrong place here. :)

- Name: Unix Unleashed
- Author(s): many
- Publisher: Sams Publishing
- ISBN: 0-672-30402-3
- Pages: 1620

- AEleen Frisch : Essential System Administration

  A very nice description of all the basic and not so basic tasks of system administration, maybe a bit too much emphasized on System V, but also covers BSD.

  Contents: the system administrator's job, communicating with users, menu interfaces, the unix way: files, processes, devices, filesystem layout, tools, finding files, deleting files, startup & shutdown, boot process, managing user accounts & groups, security, managing system resources (cpu, memory, IO, disk space), automating tasks with scripts and such, filesystems & disks, fsck, adding disks, backup & restore, terminals & modems, printers and the spooling subsystem, TCP/IP network management, electronic mail, configuring & building kernels (SCO Unix, Linux, DEC & SUN, HP, Solaris, AIX), accounting.

  - Name: Essential System Administration
  - Author(s): AEleen Frisch
  - Publisher: O'Reilly & Assosciates, Inc.
  - ISBN: 1-56592-127-5
  - Pages: 788

- 4.4BSD System Managers Manual

  This is a nicely produced reference book (which we've come to expect from O'Reilly - complete with the magic lay-flat binding, and tab marks at the edges of the pages), which consists of two broad sections. In the first section are reproductions of the complete BSD4.4 man(8) pages (ie. all those relating to system administration). Before you shout "I've already got those on disk", this is useful for two reasons: Firstly, some of us like to have a book by the keyboard and an uncluttered screen, and secondly a complete permuted index of the pages is included.

  The second section is a (disparate) collection of papers detailing a whole host of topics, from setting up name-servers to the inimitable '`sendmail`'. The papers are all somewhat terse, but usually contain a pointer to the information you want somewhere.

  This is certainly not the kind of book you sit down to read in a spare five minutes; but I have referred to it enough in emergencies to recommend it. Its probably best to have a look through it in the shop, though, to see if the style suits you.

- Name: 4.4 System Managers Manual
- Author(s): Computer Systems Research Group, UC Berkeley
- Publisher: The USENIX association and O'Reilly & Assosciates, Inc.
- ISBN: 1-56592-080-5
- Pages: 804

4.6.2 Advanced System Administration

- Craig Hunt : TCP/IP Network Administration

  The book if you want to know about TCP/IP networking. Starting with the very basics of routing, it describes how to configure your machine, routers and every crucial service. With this book in your hand, you don't need some of the following books if you just want to get it going, without tackling with every single option. I love it!

  Contents: Overview of TCP/IP (layers: concept Internet, transport, application), addressing, routing, multiplexing, subnetting, RARP, protocols, ports & sockets, name service concepts, basic system configuration of several unix systems (BSD, System V, SCO), SLIP, PPP, various routing protocols (RIP, EGP, gated), configuring DNS, rlogin, FTP, NFS, sendmail, hints on troubleshooting, network security.

  - Name: TCP/IP Network Administration
  - Author(s): Craig Hunt
  - Publisher: O'Reilly & Assosciates, Inc.
  - ISBN: 0-937175-82-X
  - Pages: 502

- Bryan Costales, etc. : Sendmail

  This book is dedicated to sendmail entirely. Beginning from where to get the source and how to install it, it describes how to set up a simple (client) mail system and documents the transition to a more complex one. The appendix describes every option in detail, as well as specialities of of IDA & V8 sendmail and rule sets.

  Contents: MUA vs MTA, parts of sendmail, running sendmail by hand, header, body & envelope, local delivery, network forwarding, how to run sendmail (command line options), the sendmail.cf file (comments, macros, rules, rule sets, class macros, options, priority), mail delivery agents, macros, addresses and rules, rule sets, the workspace, description of several rule sets, classes, options in the config file, header, priority, MX records, compiling and installing sendmail, obtaining the source, decisions in '`conf.h`', decisions in '`conf.c`', decisions in '`pathnames.h`', decisions in Makefile, pitfalls, DNS and sendmail, set up MX records, prepare for disasster, security, permissions, forged mail, the queue, parts of a queued file, printing the queue, aliases, mailing lists and '`~/.forward`', logging and statistics.

  - Name: sendmail
  - Author(s): Bryan Costales with Eric Allman & Neil Rickert
  - Publisher: O'Reilly & Assosciates, Inc.
  - ISBN: 1-56592-056-2
  - Pages: 792

- Cricket Liu, etc. : DNS and BIND

  This book discusses the Domain Name System and it's most popular implementation, the Berkeley Internet Name Domain software, which is responsible for managing the Internet's distributed database of host information and routing mail to the right destination.

  Contents: Background, How does DNS work (domain names, domains, records, delegation, resolvers, caching), how to start, setting it all up (SOA, NS, PTR records, loopback address, cache data), tools, primary & secondary nameserver, DNS and electronic mail (MX), configuring the resolver, vendor specific options, tuning the installation, parenting, handling nslookup, troubleshooting, programming with the resolver library.

  - Name: DNS and BIND
  - Author(s): Paul Albitz, Cricket Liu
  - Publisher: O'Reilly & Assosciates, Inc.
  - ISBN: 1-56592-010-4
  - Pages: 380

- Robert King Ables: The Keys to Successful Unix System Management

  This is not the 'usual' sysadmin handbook that describes how to create some user accounts or doing your backups. Much rather, it discusses the philosophy of being a system administrator, how to deal with your users, your customers, and employees, how to manage stress and disaster situations as well as plans for backing up your system, how to prevent disaster and what to do if destiny struck anyway.

  Contents: Who is the System Administrator, What does the System Administrator do, Why is the System Administrator needed, where is the SA needed, Software: Backups, User Accounts, the general health of your system, third-party software, device management, file system management, operating system modifications, performance tuning, dealing with heterogenity, staying ahead. Hardware: equipment components, equipment installation, equipment maintenance; physical environment: your computer room, aif conditioning, fire suppression, capacity management, cooling, disaster recovery, emergency procedures. Networking: standards, protocols, applications; Security: why be secure, types of risk, security planning, administrative controls, software controls, physical controls, network security, handling a breach; Administrative Skills: information management, planning, financial management; Personal skells: user service, communcation with users, management, collegues, vendors; about yourself, stress management, continuing education; the career: hiring a System Administrator, job description, keeping a system administrator, the career path.

  - Name: Keys to Successful UNIX Systems Management
  - Author(s): Robert King Ables
  - Publisher: Prentice-Hall, Inc.
  - ISBN: 0139483810
  - Pages: 316

4.6.3 Programming under Unix

- Larry Wall, Randal Schwartz : Programming perl

  This book, also known as the 'Camel book', describes the 'Practical Extraction and Report Language' in a tight fashion, showing up all the features and giving numerous small examples as well as a full reference for the grammar and all the functions provided by perl. Describing only perl version 4, this book is a nice reference for all the things you need every day, but not up-to-date if you want to use the latest features.

Contents: overview: filehandles, variables, operators, lists, pattern matching, associative arrays, lines and paragraphs, command-line switches, generating reports, subroutines, recursion, exact discussion of data types & objects, operators, statements, subroutines, regular expressions, formats, special variables and packages, all functions, common tasks with perl, real perl programs (database manipulation, grep programs, programming aids, system administration stuff, filename manipulation, text manipulation tools, processes, interprocess communications), debugging, efficiency, setuid scripts.

- Name: Programming perl
- Author(s): Larry Wall & Randal L. Schwartz
- Publisher: O'Reilly & Assosciates, Inc.
- ISBN: 0-937175-64-1
- Pages: 482

- Brian W. Kernighan, Rob Pike: The Unix Programming Environment

  This book is good for absolute Unix beginners which want to learn how to do some programming with all those tools that come ready to use with any Unix system.

  Topics covered are first steps from logging in, file handling (cat, mv, cp, rm), directory handling (cd, mkdir, ...) shell-basics (variables, wildcards), filters (grep, sed, awk), shell programming (loops, signals, arguments, stdio), an introduction to system calls (read, write, open, creat, ...), an introduction to programming with lex, yacc & make and documentation using troff, tbl & eqn.

  Despite the seemingly broad range this book aims at, it's mainly a *very* good introduction to shell programming.

  - Name: The Unix Programming Environment
  - Author(s): Brian W. Kernighan , Rob Pike
  - Publisher: Prentice-Hall, Inc.
  - ISBN: 3-446-14273-8 (German language edition!!!)
  - Pages: ~300

- Brian W. Kernighan , Dennis Ritchie: The C Programming Language

  This is *the* C-bible, written by the people who created the C programming language. It describes the full language, starting from simple function calls, variables and loops to functions, structures, arrays and pointers. The appendix contains a list of all functions from the ANSI-C-library together with their prototype and a shoft description. Be sure to get the second edition which describes ANSI-C.

  Contents: [Damn, whom did i borrow my K&R?! Anyone who can give me a rough summary here? - HF]

  - Name: The C Programming Language
  - Author(s): Brian W. Kernighan , Dennis Ritchie:
  - Publisher: Prentice-Hall, Inc. (english) - Hanser (german)

- ISBN: 0-13-110330-X (3) (english) - 3-446-15497-3 (german)
- Pages: 228

- Richard Stevens: Advanced Programming in the Unix Environment

  A book for those that want to use the basic services the Unix operating system offers. They will find hints on the layer between the bare kernel and some application program here, that's usually hidden in some library.

  Contents: The book starts with an introduction to basic concepts like files and directories, input and output, as well as various Unix-related standards and their limits and limitations. Then file IO is explained with file descriptors, fcntl, ioctl, file sharing, stat, permissions and links. Then the stdio-lib is examined with concerns to efficiency, binary and formatted IO, temporary files and alternatices to stdio. System data files like passwd, shadow passwords, groups, supplementary groups, logging and accounting data are described next, followed by a description of the environment a unix process runs in (main-function, command line arguments and environment, setjmp & longjmp), process control through fork, vfork, various wait- and exec-functions, process relations (regarding terminal and network logins, process groups, sessions, job control), signals (signal, kill, sig*), terminal IO (sttyy, termcap, terminfo, curses, advanced io (nonblocking io, record locking, streams, IO multiplexing, asynchronous IO, memory mapped IO), some introduction to process communication using pipes, FIFOs, message queues, semaphores and shared memory, passing file descriptors under various unix implementations (SVR4, 4.3BSD, >4.3BSD), client server connection functions. The book closes with three big examples, a database library, communicating with a PostScript printer and a modem dialer.

  - Name: Advanced Programming in the UNIX Environment
  - Author(s): W. Richard Stevens
  - Publisher: Addison-Wesley Publishing Company, Inc.
  - ISBN: 0-201-56317-7
  - Pages: 744

4.6.4 Introductions to Unix Internals

- Mike Loukides : System Performance Tuning

  Improving the overall performance of your system is the goal of this book. To do so, it describes the verious subsystems Unix consists of and how to monitor them for problems and bottlenecks as well as all the internals that keep the system running and the screws and buttons you can play with and what they do. Very interresting if you want to know why your box is swapping like hell again and what you can do against it.

  Contents: System performance issues, system resources (CPU, memory, IO), user communities, monitoring system activity (system load average, ps, cron, accounting, sar, benchmarks), managing the workload, some tricks for users, reducing the workload, scheduling priority, off-peak job submission, shell time limits, cpu capacity, memory performance, paging and swapping, how to tell if your system is paging, conserving memory, tuning the paging algorithm, managing the swap area, computing memory requirements, disk performance issues, IO subsystem configuration, partitions and filesystems, balancing IO workload, filesystem buffers, striped filesystems, conserving disk space, network performance issues, networks and cpu load, RFS, STREMAS, terminal performance, kernel configuration, why build a custom kernel, configuring a BSD kernel, configuring a System V kernel.

- Name: System Performance Tuning
- Author(s): Mike Loukides
- Publisher: O'Reilly & Assosciates, Inc.
- ISBN: 0-937175-60-9
- Pages: 336

- Samuel J. Leffer, et al: Design and Implementation of the 4.3BSD Operating System

  This book is to BSD-Unix what 'Programming Perl' is to the perl programming language and Kernighan/Ritchie's 'C Programming language' is to C. Also based on 4.3BSD, it's still the most comlete description of all the internals of a BSD-favoured unix, from device drivers to interrupt handling and from memory management to the filesystem. This might not be the book you look into when you're about to write 'just' a device driver, but it's certainly of value if you want to understand the relationship between the verious subsystems of the operating system. An 'upate' for 4.4BSD is in the works.

  Contents: [Heh, i don't own that book! Anyone care to give me a summary of the contents? -HF]

  - Name: Design and Implementation of the 4.3BSD Operating System
  - Author(s): Samuel J. Leffler, Marshall Kirk McKusick, Michael J. Karels, John S. Quarterman
  - Publisher: Addison-Wesley Publishing Company
  - ISBN: 0-201-06196-1
  - Pages: 471

5 Where to get sources?

5.1 Release

Sources for the current release of NetBSD (V1.1) are available from ftp.netbsd.org under `/pub/NetBSD/NetBSD-1.1/source` and its mirrors. See Section 4.2 [Anonymous FTP], page 9 for a list of some of them.

The sources are contained in several directories:

`src11`: sources for all the binaries from `/bin`, `/sbin`, `/usr/bin`, `/usr/sbin`, `/usr/libexec`, `/usr/lib` and `/usr/games`.

`ksrc11`: kernel source for all architectures: alpha, amiga, atari, hp300, i386, mac68k, mvme68k, pc532, pmax, sparc, sun3 and vax

`gsrc11`: source tainted with the GNU copyleft, e.g. gcc, gcc2, libg++, UUCP, bc, diff, cpio, gas, gawk, gdb, grep, groff, ld, rcs, tar, gzip, sort.

`dsrc11`: Domestic source that's likely to underly certain export restrictions. Mainly Kerberos, DES and crypt-related things.

`ssrc11`: source for things under `/usr/share`, mainly some dictionaries, documentation, man-pages, termcap-source, timezone-codes, etc.

To unpack, do something like the following:

```
cat .../source/ksrc11/*.?? | tar zvxCf / -
```

5.2 NetBSD-current with tar-balls

If you want to track NetBSD-current and you're really sure you know what you do, you can get tar-archives (usually called tar-balls for some odd reason) of the NetBSD-current source from ftp.netbsd.org in `/pub/NetBSD/NetBSD-current/tar_files/src` directory (and mirrors).

5.3 NetBSD-current with SUP

Sup stands for Software Update Protocol and provides a facility to update your source-tree by transmitting only those files that have changed since you've checked the last time.

Get `/pub/NetBSD/sup/supkit/README.sup` from ftp.netbsd.org and follow the instructions there, there's enough text that should help you get it going.

6 How do I build a kernel myself?

6.1 Introduction

Theoretically you have two options here. You can build the kernel under AmigaDOS or under NetBSD itself. Initially NetBSD was compiled under AmigaDOS, of course, but now everybody works under NetBSD itself. Recent attempts to compile the kernel under AmigaOS showed that the compiler, assembler, linker etc. have developped differently, and that you can't produce a working kernel under AmigaOS anymore.

For more general information on compiling the NetBSD kernel, refer to the 386BSD-FAQ; there is a complete chapter related to the subject.

6.2 Getting the kernel source

To build the new kernel under NetBSD, make sure you have the compiler package (comp11) and the kernel source tree (ksrc11/*) installed. See Chapter 5 [Where to get sources?], page 20, for ways to get the kernel source.

6.3 Configuring the kernel

Change to the directory '`/usr/src/sys/arch/amiga/conf`'. This is where the *configuration file* for the kernel lives. This file details exactly what drivers (eg. CD file systems) you want in your kernel, and what hardware you have. The easiest way to configure your new kernel is to copy the file GENERIC, and name it *YOUR-MACHINE-NAME*. (My machine is called 'Babylon', so my kernel config file is named '`BABYLON`' too). Edit this file, commenting in/out options you do/don't want. It should be pretty clear from the comments what you can safely remove and what you definitely shouldn't.

Once you have tailored the config file to your system, you need to ask the system to build a '`Makefile`' for you. This is done with the program '`config`'. Type the following command

```
config BABYLON
```

(substituting your kernel name for BABYLON, obviously ;-). The program '`config`' will create a directory '`../compile/BABYLON`' (or whatever), and place a customised kernel Makefile and a bunch of header files in there. If all goes to plan, you should get a message along the lines of 'Don't forget to run make depend'.

6.4 Compiling your kernel

You are now about ready to compile! Change directory to the compile directory for your kernel (eg. '`../compile/BABYLON`'), and type

```
make depend
```

The computer will take a while sorting out some computery stuff it has to do (to be technical about it ;-), and now is probably a good a time to start making the buckets of coffee you will inevitably need for the full kernel recompile. When the make finishes, you can set the system off building your kernel with

```
make
```

As an alternative, you can type: '`make depend all`' and have one really large coffee instead of two medium sized ones. If this is your first kernel compile, the estimated time is "Forever". Particularly if you don't have too much memory to spare; however, it is feasible to do a kernel compile in only 4MB - I have ;-). An A3000 with 16MB RAM takes approximately 2 hours, my A2000 (030@25, 4MB 16bit RAM, 4MB 32bit RAM) takes about 4 hours.

Several cups of coffee later...

Assuming you only had coffee (and nothing more, say, alcoholic...), you should be able to use your new kernel once the compilation has finished. The kernel will be named '`netbsd`' in your compile directory.

6.5 Installing your kernel

Do not delete your old kernel etc. until you have fully tested the new one; bear in mind that once you reboot into the new kernel, commands like `ps` wont work unless you copy the new kernel image over the one in '`/netbsd`', so:

```
mv /netbsd /netbsd.old
cp netbsd /netbsd
```

For booting NetBSD via `loadbsd`, you'll also have to copy your new kernel over to the AmigaOS-side of your machine! See Section 8.3.4 [Can I access NetBSD files from AmigaOS?], page 32, for more information on how to do this.

When you successfully build and installed a new kernel, you can reboot your computer into the new kernel with the following command:

```
cp netbsd /dev/reload
```

Note that this may hang if the new kernel differs substancially from the old one. Just reset your machine (three-finger-salut) and reboot NetBSD via `loadbsd` instead.

See Section 8.3.5 [How do I create device files?], page 32, for information on how to create device-files, if the file '`/dev/reload`' doesn't exist.

7 NetBSD - Linux - Amix - Minix

Currently there are four possibilies to run Unix on the Amiga.

The first one is Amiga-Unix, also called Amix. This is a commercial unix from Commodore, actually, it's one of the very first (and most complete!) implementations of System V Release 4 (SVR4). It is neither sold nor supported any more.

The second unix on the Amiga was Minix from A. S. Tanenbaum, a very famous operating system professor. It runs on every Amiga and does not support memory protection and some other features needed to get the real unix feeling. It is commercial, too. It's major aim is to be an operating system to play around with. It was developed for the computer science students Tanenbaum held lectures for.

The third available unix on the Amiga was NetBSD. It gives you almost anything you might want to expect from a free unix clone. It is being developed on several platforms, and therefore has a reasonable amount of developers supporting it. It was designed to be as portable as possible, as all machine-dependent code is separated. Currently there are ports to the Amiga, Intel-based PCs, HP-300 (680x0-based), Macs(680x0-based), Sun3, Sun-Sparcs and some other platforms. NetBSD-Amiga has a binary-compatibility-mode for Sun3-SunOS-binaries. Of course, this means only binary-compatible with static linked binaries, unless you have the original SunOS shared-libs available. The far end aim is, that all NetBSD-implementations will be binary-compatible on the same processor. That means, that there will be NetBSD-m68k-binaries, that run on the Amiga, the Macs and the Sun3. This shows some of the possibilities of NetBSD. NetBSD-Amiga is part of the NetBSD-current sources, so any platform-independent improvement will be an improvement for NetBSD-Amiga as well. The new features from 4.4BSD are already incorporated into NetBSD. NetBSD contains a lot of other fancy features, other unix-based operating systems are missing, e.g. cpu-time- or disk-quotas, mounting CD-images, swapping to files and concatenating several disks to one large disk. NetBSD is copyrighted software, but you are free to use, modify and distribute it. Note that it is *NOT* under the GPL (General Public License, the Gnu Copyright (-left)) and the developers of NetBSD want it to keep this state. Therefore it is not possible to include any software which is under the GPL into the kernel. NetBSD is available in source, but anybody is free to take the current sources and provide them together with his own binaries, i.e. you don't have to provide the sources of your own work. This offers the opportunity for software developers to keep the source of the programs they sell. This is not possible under GPL, where you must provide the source. However, it is appreciated if you also release the source of your derived work to the public. If nobody releases his sources, the free software will soon disappear.

The fourth available Unix on the Amiga is Linux . It was first designed to run on Intel-based PCs. However, HamishMcDonald and others started a major rewrite of the sources so that it is now possible to run Linux on the Amiga (and Atari, by the way). The first public kernel release happened around October 1993, a few months after the first NetBSD kernels came out. Nowadays Linux has become quite usable, although some people still report problems on the '040 then and now. Attempts to make the kernel run on '060 machines are in the works. The major drawback with Linux/m68k is that it is not yet incorporated into the Linux mainstream source tree. This means that every change made in the Intel-PC source tree has to be adapted manually to the m68k source tree. In NetBSD, on the other hand, changes are always made to the unique source tree for all platforms. Also, there are not yet as much device drivers for Linux/m68k as for NetBSD/Amiga (esp. in the graphics section), and there is as of now no such 'binary distribution' as known from PCs (Slackware , SuSE etc.) and whole NetBSD. All of these problems are the main goals of development for the next months, some of the results so far look pretty promising.

For interested people: The Linux/m68k FAQ can be found at: http://www-agrw.informatik.uni-kl.de/~jmayer/linux68k/linux68k-faq .

Everybody has to decide by themselves, which unix is best for them. For now I can say, that using NetBSD is the best way to go, as Linux still has some more nasty bugs in it and is missing a lot of the functionality you might want to use, but this can change in the future.

8 Frequently Asked Questions and their answers

This chapter contains a list of Frequently Asked Questions (FAQs) and their answers. See Chapter 4 [Where can I find more information on NetBSD?], page 9 for information on where to find the files and documents mentioned in the answers.

8.1 Installing a new system

Installing the system is probably the most painful part of running NetBSD-Amiga; this section should hopefully answer most of the questions which will arise, in conjunction with the installation document (See Chapter 4 [Where can I find more information on NetBSD?], page 9).

8.1.1 How do I get NetBSD?

The first thing you should do is find the file '`getting-NetBSD`', which details exactly where to find all the files you'll need. You can probably find it the same place you found this FAQ. See Chapter 4 [Where can I find more information on NetBSD?], page 9 for details.

8.1.2 How do I install NetBSD?

There's a file called '`INSTALL`' that comes for each architecture that's supported by a NetBSD release, and which describes the installation procedure in great detail. For the 1.1-release, this document can be found in the following places:

- ftp.netbsd.org : /pub/NetBSD/NetBSD-1.1/amiga/INSTALL and mirrors
- ftp.uni-regensburg.de : /pub/NetBSD-Amiga/release/INSTALL and mirrors

Please consult this file for further instructions! See Section 4.2 [Anonymous FTP], page 9, for a list of mirrors of the above server.

8.1.3 How can I autoboot into MultiUser mode

The '`loadbsd`' program has a number of options to control the boot process - type '`loadbsd -?`' to see all the options. The one you need in particular (for the current version of LoadBSD, anyway) is '`-a`'.

8.1.4 NetBSD always asks me for a root device

Make sure you are using the latest kernel and version of `LoadBSD`. Note that `LoadBSD` has an option, `-b`, to explicitly force it to ask for the root device on boot - make sure you aren't running `LoadBSD` with that option enabled!

Also, ensure that your hard disk (in particular DOSType IDs etc.) is configured properly so that NetBSD can locate the root partition correctly.

8.1.5 Where can I find ixemul.library version >= 39.46

Well, you don't strictly speaking need it; If LoadBSD pops up a requester asking you for version 39.47, click *OK* and `LoadBSD` will continue using version 39.45.

If you don't have any version of '`ixemul.library`' (where have you been living? :-), have a look on an Aminet site or get it from ftp.uni-regensburg.de:/pub/NetBSD-Amiga/tools/ixemul.library.gz .

8.1.6 NetBSD hangs while booting into MultiUser mode

Make sure the lines which read

```
name_server=NO
timed_flags=NO
```

in the file '`/etc/netstart`' do indeed read `NO` twice; '`named`' and '`timed`' will hang the machine if there is no network attached when it starts.

Also, delete the directory '`/var/yp`' if you don't use YP (NIS), because the upcoming '`ypbind`' would search for a NIS server forever.

8.1.7 I keep getting cksum errors when I try to install!

This (and other bizarre symptoms) could be the result of a bug in your SCSI host adapter's DMA controller or ROM - I have seen this with an A2091. (It also seems to depend on the phase of the moon and other sundry factors). The solution is to switch DMA transfers off in the kernel. To do this using '`binpatch`' (See Section 8.2.3 [How do I use binpatch?], page 28 for details) type

```
binpatch -s _sbic_no_dma=0x1 kernelfile
```

(not forgetting to replace '`kernelfile`' with the name of your kernel image!). You should now be able to boot & install using programmed-IO (PIO) instead of DMA.

8.1.8 Which distribution should I use?

At the time of writing, there are two principle distributions you could look at, '`NetBSD-1.1`' and '`NetBSD-current`'.

Which you choose really depends on what sort of system you are looking to run. If you want a stable, out-of-the-box, install-and-forget type system, go for the `NetBSD-1.1` distribution. The

`NetBSD-current` distribution is for the OS hackers amongst you; it is in a state of constant change, but as a result it'll be the first place to find neat new features etc. The choice is yours!

See Chapter 4 [Where can I find more information on NetBSD?], page 9 for information on where to find the relevant files.

8.1.9 Does NetBSD run with the PROTO-Chip?

The first A3000 and A2091 were equipped with a prototype version of the SCSI controller chip (exact name: 33c93-4), the later ones had a full-blown chip (33c93-8).

The answer to the above question is *yes*. The only problem which may occur is that some hard drives do not like to be enabled in sync mode; it depends on your kernel patches, the hard drive, controller etc. etc. if you'll find this problem. To disable sync negotiation, pass '`-I ff`' to loadbsd.

8.1.10 My A2000 is recognized as A1200

On some machines, the detection of which machine one has doesn't work and so you have to tell it by hand giving the '`-c`' switch to loadbsd:

```
loadbsd -c 2000 netbsd
```

8.2 Hardware, drivers and binpatching

This section details certain specific hardware problems you may run into.

8.2.1 How do I get a serial terminal to work correctly?

The file '`/etc/ttys`' tells `init` which ports to run a `getty` (ie. login) program on. Edit the file, and (supposing you had a 19200bd vt100 terminal on the serial port) add or edit a line to read

```
tty00 "/usr/libexec/getty std.19200" vt100 on insecure
```

'`tty00`' indicates the internal serial port, the '`std.19200`' tells `getty` what serial parameters to use - look in '`/etc/gettytab`' for details of other speeds etc. - and the '`vt100`' indicates what terminal emulation to use; look in '`/usr/share/misc/termcap`' for details of other terminal emulation names.

Note the '`insecure`' entry - this means that you (or an imposter, more importantly) cannot log in as '`root`' on that terminal, even if he has the root password. You've to log in as 'normal' user then use '`su`' to switch to root. Be sure to be in the right group (wheel, See Section 8.3.1 [Why canI su to root?], page 30) for this, especially if you've marked the console as secure.

Marking the console as insecure will result in the root password being requested when booting into single-user. Normally the only terminal regarded as secure would be the console.

8.2.2 Does NetBSD work with the Retina video card?

A qualified yes! With a Retina card, NetBSD tries to open a screen with a resolution of approximately 800 by 600, with a display rate of 75KHz. Not all monitors are capable of handling such a high display rate - indeed, it could cause some damage - so it comes highly reccomended that you use '`binpatch`' to set the symbol `_retina_default_mon` to one of the following values depending on the resolution/refresh rate your monitor can handle:

- `0x01` – 640x512 at 31.5kHz
- `0x02` – 768x600 at 38kHz

See Section 8.2.3 [How do I use binpatch?], page 28 for details of using `binpatch`.

8.2.3 How do I use binpatch?

8.2.3.1 Introduction

'`binpatch`' is a very useful little utility which allows you to change the value of variables in a compiled binary (in particular, the NetBSD kernel) without needing to recompile; it uses the symbol table information included in the binary to locate where the variable's value is stored, and directly modifies it. This is very useful for setting certain flags in the kernel necessary to make it boot on your system (until you are in a position to compile a kernel to your own requirements!).

You will be able to find a `binpatch` for AmigaOS in '`binpatch-array-1.1.tar.gz`' located in '`.../tools`' on ftp.uni-regensburg.de and its mirrors .

8.2.3.2 Patching the kernel

In order to patch the kernel, you need to know basically two things - the name of the symbol (eg. `_retina_default_mon`) and the value to patch it to. Armed with this information, it is basically just a case of running the binpatch with the name of the kernel file. For example, assuming a kernel image named '`netbsd`',

```
binpatch -s _retina_default_mon=0x02 netbsd
```

would patch the kernel to boot into a Retina screen of resolution 768 by 600 at 38kHz.

8.2.3.3 Other binpatch options

There are a few more 'advanced' options you can call binpatch with; these are detailed below.

- `-s` – Specify the symbol of the variable (as above).
- `-a <addr>` – Explicitly specify the address of the location to patch.
- `-b` – Patch only a byte value.

- `-w` – Patch a word value (2 bytes).
- `-l` – Patch a longword value (4 bytes).
- `-r <val>` – The value to replace the old value.

8.2.3.4 List of all binpatchable symbols

[Such a list would be very nice, but I didn't come across one yet. Please contact the maintainer of this FAQ if you want to say something here! - HF]

8.2.4 Tapedrive problems

This FAQ deals with a number of problems tapedrive owners may experience with NetBSD - and more importantly suggests a few workarounds! As far as I know, at the time of writing no tape drive has been found that cannot be made to work with NetBSD.

The first thing to point out is that for some tape drives NetBSD reports that the tape drive is not supported on boot - ignore this message it is lying!

[Note of the editor: i don't know if either of the followint two problem is still valid. Besides that, what about those Tandberg streamers that didn't work with 1.0? Someone tell me, please! -HF]

8.2.4.1 Problem 1

You may find that if you boot NetBSD immediately after using `btn` in AmigaDOS to write a file to the tape, that `tar` reports some error when trying to read from the tape.

The reason is that NetBSD doesn't send SCSI IDs 4 and 5 a reset command, since they are assumed to be tape drives, and sending a reset can cause some drives to spend a lot of time resetting/rewinding. The solution is to power cycle the tapedrive before or after booting NetBSD, forcing it to reset.

8.2.4.2 Problem 2

If you write small files to the tape, there may appear to be nothing on the tape when you attempt to read them back - this can occur with tapes written from either NetBSD or AmigaDOS. The solution is simply to append a fairly large file to the end of the tape. For example, if I wanted to write the file '`netbsd`' to the tape, I would use the command

```
tar cvfp /dev/rst0 netbsd rootfs.gz
```

(assuming '`rootfs.gz`' is fairly large) instead of

```
tar cvfp /dev/rst0 netbsd
```

8.2.5 Reading Sun tape cartridges

Reading tapes created by Sun machines is another source of some confusion. The original Sun Archive tape drive is capable of reading/writing two tape formats, QIC-11 and QIC-24. To maximise your chances of being able to read the tape, use QIC-24 format; the command

```
tar -cvf /dev/rst8
```

will create a QIC-24 formatted tape.

8.2.5.1 But the Sun has an Archive Viper drive

Although the Archive Viper hardware can read/write a variety of formats, the Sun will only let you use one – QIC-150. All the tape drive devices ('`/dev/rst0`', '`/dev/rst8`', '`/dev/rst16`' etc.) will therefore produce a QIC-150 tape. Just use '`/dev/rst8`' for all data cartridge systems (if it isn't a 4mm or 8mm tape, it probably is a data cartridge).

8.2.6 How to deal with silo-overflows?

Those silo overflows usually occur when you work with a high speed modem connection, and indicated the kernel's buffer for serial i/o is full. To fix this problem, compile a new kernel with the following lines in your config-file:

```
options         LEV6_DEFER      # serial is more important than clock
options         "SERIBUF_SIZE=8192" # buffer size for serial stuff
```

8.2.7 How can I run X on the A2410 TIGA board?

You can't run X on the A2410 TIGA board, as there is currently no X server available for that card.

8.3 Teething troubles

If you think its all over once you've finished the installation, you're mistaken! There are numerous little things that need tweaking and configuring before you can sit down and let the system run itself. Happily, once these things are set up, very little further work is necessary, bar day to day adding of users, adding new services etc.

8.3.1 Why can't I su to root?

For a user to switch user ('`su`') to root, they must be a member of the group `wheel`. Make sure that in the '`/etc/group`' file there is a line like

```
wheel:*:0:root,tjw1
```

The comma-separated list indicates which users are members of the `wheel` group - add any users you wish to be able to '`su`' to root here; note that although the user still requires root's password, the wheel group should really be restricted to system administrators only. Please note also that if you remove 'root' from this list, unpredictable things may happen.

8.3.2 What packages can I install to enhance NetBSD?

Since NetBSD is basically a flavour of Unix, more or less anything written for Unix can probably be ported (or already has been) to NetBSD; the list is enormous! See Chapter 4 [Where can I find more information on NetBSD?], page 9 for details of mailing lists and newsgroups to subscribe to.

If you're looking for precompiled binaries that you just have to install, have a look at '`/pub/NetBSD-Amiga/contrib`' on ftp.uni-regensburg.de and its mirrors .

8.3.2.1 What about X-Windows?

First of all, there ain't no such thing as 'X windows'. The thing you think of is either called, X, X Window System or (in it's current release) X11R6. Read X(1) if you want to find out more.

X servers for NetBSD-Amiga are available for the standard custom chips, and also available (or in development) for a number of graphics cards:

- OCS, ECS, AGA: Xdaniver
- Picasso II, Piccolo, Spektrum: Xcl
- Retina Z2 & Z3, CyberVision'64: Xamiga24

A binary distribution of the X Window System for NetBSD/amiga V1.1 can be found under '`/pub/NetBSD-Amiga/contrib/X11/X11R6/bin11`' on ftp.uni-regensburg.de and its mirrors . It will take up about 30MB of diskspace if fully installed. Please read the files '`README`' and '`README.2`' there, as well as the '`NetBSD-Amiga-X-FAQ`' for full instruction on installation.

8.3.2.2 And networking?

Networking is an integral part of the BSD system, and already a part of your kernel; support is currently available for serial IP protocols (SLIP, PPP) over the built-in serial port, and also several ethernet cards. See Section 3.5 [What hardware is supported by NetBSD/amiga V1.1, anyways?], page 7, for a full list of all supported ethernet cards.

See Chapter 4 [Where can I find more information on NetBSD?], page 9 for information on locating the slightly outdated '`NetBSD-Amiga-Networking-FAQ`' .

8.3.3 Can I access AmigaOS files from NetBSD?

The ADOS file system driver for NetBSD-Amiga can currently mount AmigaOS file systems read-only; support for writing to AmigaOS partitions may be added at a later date. For more information on how to mount AmigaOS filesystems, see the '`mount`' and '`mount_ados`' man pages.

In any case, don't forget to add '`-o ro`' as the options to the mount command, as NetBSD's AmigaDOS-filesystem can't write to AmigaOS-files (yes, this is necessary since 1.0! :-).

8.3.4 Can I access NetBSD files from AmigaOS?

Yes, by using the BFFS-Filesystem, but be sure to get version 1.4b0, which does *not* provide code for writing files to NetBSD-disks but can only read them. There's a V1.31b that can write to *old* (4.3) filesystems (e.g. from before NetBSD/amiga V1.0 or ones that were created with option '`-O`' to `newfs`), but that will nuke your 4.4-filesystems you `newfs`'ed under V1.1 or V1.0 without mercy. You've been warned!

The file's available as '`bffs-1.4b0.gz`' from '`/pub/NetBSD-Amiga/tools`' on ftp.uni-regensburg.de and its mirrors , some documentation on how to set up the filesystem is in the '`bffs-1.3.lha`' archive.

8.3.5 How do I create device files?

There is a shell script (in the '`/dev`' directory) called '`MAKEDEV`' which creates device files for any of the standard device drivers. To create, say, the '`view00`' device, type (from a root shell)

```
cd /dev
./MAKEDEV ./view00
```

8.3.6 How do I create accounts and change user information?

8.3.6.1 Introduction

From the adduser(8) manpage:

A new user must choose a login name, which must not already appear in '`/etc/passwd`' or '`/etc/aliases`'. It must also not begin with the hyphen '`-`' character. It is strongly recommended that it is all lowercase, and not contain the dot "`.`" character, as that tends to confuse mailers. An account can be added by editing a line into the passwd file; this must be done with the password file locked e.g. by using chpass(1) or vipw(8).

A new user is given a group and user id. Login and user id's should be unique across the system, and often across a group of systems, since they are used to control file access. Typically, users working on similar projects will be put in the same groups. At the University of California, Berkeley, we have groups for system staff, faculty, graduate students, and special groups for large projects.

A skeletal account for a new user ernie might look like:

```
ernie::25:30::0:0:Ernie Kovacs,,,:/home/ernie:/bin/csh
```

For a description of each of these fields, see passwd(5).

It is useful to give new users some help in getting started, supplying them with a few skeletal files such as '`.profile`' if they use '`/bin/sh`' or '`/usr/local/bin/bash`' (which does *not* come with the NetBSD distribution!), or '`.cshrc`' and '`.login`' if they use '`/bin/csh`' or '`/usr/local/bin/tcsh`' (which also doesn't come with the NetBSD distribution). The directory '`/usr/share/skel`' contains skeletal definitions of such files. New users should be given copies of these files which, for instance, use tset(1) automatically at each login.

8.3.6.2 Adding a new user

The easiest way to add a new user is to run the '`vipw`' command to edit the master password file, and add a new entry with the user's username and userid (which must be unique), but all other fields left blank. Now run the '`chpass`' command, specifying the new user's name, eg.

```
chpass tjw1
```

You can now edit the user's password, finger information, group etc. etc. in comfort.

For more information on the '`vipw`' and '`chpass`' commands, see the relevant manual pages.

8.3.6.3 But - isn't there an adduser-command?

The 1.1-release of NetBSD doesn't include an `adduser` command. You can get one as '`.../contrib/adduser-1.2.tar.gz`' from ftp.uni-regensburg.de and its mirrors , you need `perl5` for it, though (to be found in the same directory :-).

8.3.7 How can I reboot straight into NetBSD?

The '`reboot`' command that's distributed with NetBSD/amiga performs a 'full' reboot through AmigaOS or whatever, but that may hang occasionally. NetBSD/amiga also provides the facility to just 'reload' (and restart) the kernel itself. To enable this, do the following (as root):

```
mv /sbin/reboot /sbin/reboot.amiga
echo "cp /netbsd /dev/reload" >/sbin/reboot
chmod 755j /sbin/reboot
```

You can now use the '`reboot`' command to reboot into NetBSD, and '`reboot.amiga`' to reboot into AmigaDOS.

Also, ‘`shutdown -r`’ will use the ‘`/sbin/reboot`’ that goes directly back to NetBSD. In case your system doesn’t have a ‘`/dev/reload`’ for some odd reasons, you can generate one with the following options:

```
cd /dev
./MAKEDEV std
```

8.3.8 Why doesn’t the ps command work?

For programs that use the kernel symbol table to retrieve information (like ‘`ps`’), you must make sure that there is a copy of your kernel image named ‘`/netbsd`’. You may also want to make a link from this file to ‘`/vmunix`’ for *very* old programs.

8.3.9 Is there a debugger for NetBSD?

The GNU debugger ‘`gdb`’ is available for NetBSD (it is in the standard comp* distribution package).

8.3.10 How do I report bugs?

If you think you have discovered a bug, and you are sure its not your mistake, the best thing to do is to mail on one of the mailing lists (See Chapter 4 [Where can I find more information on NetBSD?], page 9 for details). Try to include as much information as possible - the configuration of your machine, the output you saw, the exact circumstances etc.; if possible try to isolate the bug as far as is possible (ie. don’t just say ‘my machine crashes’, try to explain what causes it to crash!).

8.3.11 How do I get a Meta-Key for Emacs?

Executing the commands ‘`stty cs8 -istrip -parenb`’ will allow you to use any of the Amiga keys as a meta-key.

8.3.12 My keymap is all wrong!

8.3.12.1 Fixing the console’s keymap

There is a little program called ‘`loadkmap`’ which allows you to change the keyboard map. The maps are edited in the form of a simple C file - a german keymap ‘`din-kbdmap.c`’ is supplied. It should be relatively trivial to edit this to your requirements. To install the German keyboard, for example, you would type the following commands

```
cd /sys/arch/amiga/stand/loadkmap
gcc din-kbdmap.c -o din-kbdmap
gcc loadkmap.c -o loadkmap
./din-kbdmap >din
./loadkmap din
```

To make this permanent, add the '`loadkmap`' command to '`/etc/rc.local`'. There's also a readily compiled version of those files in '`.../contrib/kbdmaps-1.0-bin11.tar.gz`' on ftp.uni-regensburg.de and it's mirrors .

8.3.12.2 Fixing the keymap under X

Under X, you change your keymap-settings with the '`xmodmap`' command, which reads its configuration from the file '`$HOME/.xmodmaprc`' when invoked (e.g. from '`$HOME/.xinitrc`'). Such configuration files for French , German , British and Swedish keyboards exist under '`.../contrib/X11`' on ftp.uni-regensburg.de and its mirrors .

8.3.13 Xcl reports "Cannot open screen"

Add the following line to '`/etc/X0.hosts`' (assuming your hostname is `anarchy.tue.schwaben.de`):

```
INET:anarchy.tue.schwaben.de
```

You have to set up networking (See Section 8.3.2.2 [And networking?], page 31), then you can get your hostname with the '`hostname`' command.

8.3.14 Where do I get Motif? XView? OpenGL?

8.3.14.1 Motif

Motif is a commercial product and currently *not* available for NetBSD/amiga. The only option you have here is to get a binary licence for Sun3 libs, and link those with NetBSD/amiga.

[Someone who tried this and can get me apointer on where to get Motif for Sun3? - HF]

8.3.14.2 XView

XView (also known as the Open Look toolkit) is currently only available as a binary that was built in the times of NetBSD v0.9, this binary does *not* work with NetBSD V1.1.

So, if you run a recent version of NetBSD, your only choice os to get the sources and compile it yourself. The sources for version 3.2 are are available from ftp.x.org in /R5contrib/xview3.2 .

8.3.14.3 OpenGL

There's a free implementation of the OpenGL API called " Mesa 3-D graphics library ", or simply Mesa. A precompiled version is available from ftp.uni-regensburg.de and its mirrors as .../contrib/X11/Mesa-1.0b-bin10.tar.gz , sources are available in /pub/Mesa from iris.ssec.wisc.edu.

8.3.15 What's all this stuff about binary emulation?

Binary emulation means two things:

1. to read different executable file formats: About every second operating system uses a different file format in which it saves its binaries, using different headers, magic cookies, hunks, whatever. The ones NetBSD supports natively is a.out, and through some of the above emulations, ELF's supported, too.

2. (the more important thing!) Almost every operating system offers also a different range of services to the system programmer, with different ways to access them. This is called the system-call-interface. Under AmigaOS, this is done with an index into a table containing pointers to functions pointed to by some address register and the data in some data register, NetBSD uses some trap-command, I think. Other operating systems use other methods, traps, tables, registers, etc. and this is handled by the binary emulation.

Of course, for a program to generate such system calls (in a controlled manner :-), it has to be compiled for the CPU it's running on. You won't be able to execute Ultrix or SunOS-SPARC binaries on a Motorola CPU. But what you can do is use SunOS-Sun3-binaries, which just use different ways to reach the kernel (=the operating system itself) to use its services.

On NetBSD/amiga, you can use binaries for the following architectures (as long as they don't access any special hardware or drivers):

- NetBSD/amiga, mac68k, atari and sun3. The hp300-port uses a different pagesize to remain binary compatible with HP-UX, and therefore won't work by default.
- Sun3, SunOS till 4.1.1. You can run programs compiled for Suns with Motorola CPUs, *not* SPARC CPUs. So you can *not* just download Netscape for SunOS 4.1.3, as it is still compiled for the wrong CPU. Sorry.

8.3.16 How can I boot NetBSD without keeping an AmigaOS partition?

There are two ways to boot NetBSD without having a small AmigaOS partition which holds `loadbsd` and the kernel:

1. Setup the BFFS filesystem (See Section 8.3.4 [Can I access NetBSD files from AmigaOS?], page 32) for your root partition with `hdtoolbox`, so it doesn't need a '`Mountlist`' entry to be mounted, but rather gets mounted automatically on system bootup. Next, create a directory '`/s`' that holds the AmigaOS `loadbsd` binary and a '`startup-sequence`'. Then add the following line to '`/s/startup-sequence`':

   ```
   s:loadbsd -a :netbsd
   ```

You may need some additional assigns vor '`ENV:`', but I think you get the basic idea.

2. Use Michael Hitch's Bootloader which is in an experimental state as of this writing. This Bootloader can be found in '`.../experimental/bootblock.tar.gz`' on ftp.uni-regensburg.de and its mirrors . Follow the instructions given in the README-file to set up the bootblock.

8.3.17 Where can I find an archive of the mailing lists?

All the official NetBSD mailing lists are archived on ftp.netbsd.org in /pub/NetBSD/mailing-lists . An archive of the discussions during the very early days of NetBSD/amiga can be found in /pub/NetBSD-Amiga/docs/Mailinglist-Archive on ftp.uni-regensburg.de and its mirrors .

Table of Contents