NetBSD-Amiga Frequently Asked Questions

Revision 3.2
26th January 1995

Tim Walls, et al

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/OU2=P1/OU=JNB1/O=Siemens/P=SCN/A=TELKOM400/C=ZA/@x400.scn.de`

Third revision by Tim Walls

Currently maintained by Tim Walls `tjw1@doc.ic.ac.uk`

NetBSD-Amiga was initially ported by Markus Wild (mtk@irc) (mw@eunet.ch)

1 Introduction

1.1 About this document (Please Read!)

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/OU2=P1/OU=JNB1/O=Siemens/P=SCN/A=TELKOM400/C=ZA/@x400.scn.de`

Third revision (restructured) by Tim Walls

Currently maintained by Tim Walls `tjw1@doc.ic.ac.uk`

1.1.2 Copyright

1.1.3 Distribution

1.1.4 Work in progress

This document is *work in progress*. I am in the middle of 'rationalising' the TeXInfo node names etc., and it is possible I have missed some links; there are also a few sections that need tidying up. If you notice any errors, bugs etc. (missing links, grammatical errors, plain lies ;-) please mail me - `tjw1@doc.ic.ac.uk`.

1.2 Introduction

Welcome to the NetBSD-Amiga-FAQ!

This document contains a bunch of useful infomation about NetBSD on the Amiga. It was mainly created - and is regularly updated - 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 [INFORM], 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 I 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.

One last thing before we start: If you find this document useful and want to thank me for maintaining it, feel free to send me an email. If you want to send me any money or other donations, contact me under the above email-address for my 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, HP, Mac, Sun3, Sparc, PC, ... It is developed by people all around the world on the Internet. Therefore it is called NetBSD. It has all the features you would expect in a modern Unix, including true multitasking, virtual memory, shared libraries, demand loading, shared copy-on-write executables, proper memory management 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 FAQ.

See Chapter 4 [INFORM], 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. I don't want to list any names, because I am sure I would forget some brave soul. The maintainer of the "projects" file produces a list of projects being worked on. Consult this file for more information about who is working on what.

See Chapter 4 [INFORM], page 9, for information on how to get this file and how to contact the developers.

3 What do I need to run NetBSD

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)
- 68030
- 68040

Note, that it is very important that you have a **real** CPU, i.e. not an EC-version. These versions are missing the MMU-part of the chip and are therefore cheaper. It is not possible to simply add a 68851 (MMU) to these chips. You have to replace your CPU, if possible, or you have to buy a seperate CPU-board or accelerator-board equipped with one of the **real** CPUs mentioned above to be able to run NetBSD.

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. However, 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. If someone rewrites the program that loads the kernel into the memory, it would be possible to boot the kernel with 512KB CHIP-RAM. Currently 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 6MB 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.

The current NetBSD kernel can only use contiguous memory; typically, the loader program will choose the largest contiguous block of memory. This is important to note - the machine described above could have 8MB of non-CHIP memory (assuming there is 4MB of 32bit RAM), but NetBSD can only use 4MB of it (typically the 32bit RAM has a higher priority, so it will be used in preference).

Patches to allow the kernel to use non-contiguously mapped memory **do** exist, but at the time of writing are not ‘publicly' released. (Would anyone like to upload their patches to the ftp site? If so, let me know!). In the situation above, there is a tradeoff - extra memory vs. slower execution (16bit memory can degrade performance considerably) - a better solution would be to add extra 32bit store.

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. It is planned to create a NetBSD-CD it the not so far away future. Once it is available you can work with NetBSD from the CD-ROM. But then again, CD-ROMs are a lot slower then harddisks, so if you want to go for maximum speed, you will need some harddiskspace for NetBSD. How much you actually need depends heavily on what you want to do with NetBSD.

8MB of hard disk space will be sufficient to boot NetBSD and play around with it a bit. 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. Once it is available, you should definitely consider the NetBSD-CD a good source for programs. Having the seldom used binaries executed directly from the CD can save you a lot of harddiskspace.

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.

See Chapter 4 [INFORM], page 9, for information on how to get the file '`compatibility-list`'. It contains information about the supported harddisks.

3.3.1 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.

See Chapter 4 [INFORM], page 9, for information on how to get the files '`compatibility-list`' '`projects`'. The former contains information about which devices already work with NetBSD. The latter is a list of projects currently being worked on. The file '`wish-list`' documents the devices, which are currently *not* supported.

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 5 [KERNEL], page 15, for information on how to do this. See Chapter 4 [INFORM], page 9, for information on how to get in contact with other developers.

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.

A CD-ROM-drive is recommended, too, because it allows you to save a lot of harddiskspace, when you run keep most of the seldom used things on the CD.

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. Chose 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 the news-groups `comp.os.386bsd.announce`. Despite the term *386* in the name of this group, announcements for all flavors of *BSD are discussed here. The latest version of the '`NetBSD-Amiga-FAQ`', the document you are currently reading, and the BSD-FAQ, which is maintained by Dave Burgess (burgess@s069.infonet.net), are posted regularly to this group. You should also subscribe to the other `comp.os.386bsd.*` groups to take part in active discussion or simply for posting questions.

4.2 Anonymous FTP

If you don't have access to Usenet, but have the possibily 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.

Additional information is available on `ftp.uni-regensburg.de`. Look into the directory '`/pub/NetBSD-Amiga/docs`'.

4.3 World Wide Web

This document should also be available via the World Wide Web (Mosaic, Netscape etc.); at the time of going to press, exactly where isn't known :*). Keep an eye on the mailing lists mentioned below for details of new revisions of the FAQ etc.

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

- http://www.informatik.uni-erlangen.de/tree/Departments/IMMD-IV/Persons/roessler/NetBSD-Amiga/index.html
- ftp://ftp.uni-regensburg.de/pub/NetBSD-amiga/.index.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:

```
Return-Path: <Majordomo-Owner@netbsd.org>
...
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.
It's been hacked some by cgd, Revision 1.8.

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>.

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

    who <list>
        Find out who is on the named <list>.

    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
"Majordomo-Owner@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:

amiga
amiga-dev
amiga-x
core
current
current-users
m68k
macbsd-development
macbsd-general
netbsd-bugs
netbsd-help
netbsd-ports
netbsd-users
port-hp300
port-i386
port-m68k
port-pmax
port-sparc
port-sun3
portable-ppp
source-changes
sun-lamp-users
tech-install
tech-kern
tech-net
tech-ports
tech-userlevel

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

NOTE: Most of these mailing lists are archived.  Do not
send things to them if you do not wish your messages to be
```

```
archived.
>>>> end
END OF COMMANDS
```

The '`amiga`' list is for a general discussion of Amiga-related problems. You can send beginners questions, *not* answered in the FAQ to this list. The '`amiga-dev`' list is dedicated to discussions between developers. *Do not post any beginners questions here!* This will hinder the discussion about the kernel development. The X-Window-System on NetBSD-Amiga is discussed on '`amiga-x`'. Note, that there is a seperate FAQ for X. It should be posted regularly on this list. The '`NetBSD-Amiga-FAQ`' is sent regularly to the '`amiga`'-list.

It is highly recommended to get on the '`current-users`' and on the '`amiga`' lists, because this is the best way to get in contact with the developers. To do so, your email would look like this:

```
To: majordomo@netbsd.org
Subject: None
subscribe amiga
subscribe current-users
end
```

You will receive an answer-mail from majordomo, that he successfully processed your subscription and that you are on the desired mailing-lists. Note that there is some traffic on these lists, so expect a few mails more to arrive dayly in your mailbox. 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.

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 eaxmple, if you would like to write a mail to the amiga-list, send your mail to `amiga@netbsd.org`.

4.6 Private Email

If, and only if, all of the above fails, contact one of the developers or the FAQ-maintainer in a personal email. So, if you cannot get hold of a copy of the latest FAQ, feel free to contact the current maintainer of this FAQ. He will be happy to send you a recent copy.

4.7 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.

Sadly, however, I don't have the time or money to go out and buy them all myself - so if you have a book you'd like to mention (good or bad), mail me with the details.

4.7.1 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
- Publisher: The USENIX association and O'Reilly & Assosciates, Inc.
- ISBN: 1-56592-080-5

5 How do I build a kernel myself?

5.1 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 almost everybody works under NetBSD itself. It is highly recommended that you build the kernel under NetBSD, because you save a lot of HD-space on the Amiga side and, the more important reason, the possibility that other developers can help you with the problems that might occur is much higher. The only sensible reason for compiling under AmigaDOS is that you cannot run NetBSD, because NetBSD doesn't support your harddisk-controller, and you want to develop a driver for your controller.

So, if you want to compile NetBSD under AmigaDOS, get the document '`RECOMPILE`'. Although it is old, it contains some advice for this, but again, I strongly recommend that you do any kernel-development under NetBSD itself.

5.2 Compiling a new kernel under NetBSD

This section is work-in-progress, and necessarily sketchy at the moment! I'll write something more coherent once I've pushed this revision of the FAQ outta the door - Tim

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

5.2.1 Configuring the kernel

To build the new kernel under NetBSD, make sure you have the compiler package and the kernel source tree installed. Details of where you can find the kernel source will be included just as soon as I can remember where I found it (Anyone want to send me some information I could include about the mystic intricacies of sup?).

Change to the directory '`/sys/arch/amiga/conf`'. This is where the *configuration file* for the kernel lives. This file details exactly what options (eg. CD file systems) you want installed, and what hardware you have. The easiest way to configure your new kernel is to copy the file GENERIC, and name it after <your kernel 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 definately 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.new`'. Type the following command

```
config.new BABYLON
```

(substituting your kernel name for BABYLON, obviously ;-). The program '`config.new`' will create a directory '`../compile/BABYLON`' (or whatever), and place a customised kernel makefile & 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'.

5.2.2 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 short while sorting out some computery stuff it has to do (to be technical about it ;-), and now is probably as good a time as any 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
```

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 ;-).

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.

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
```

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`'.

See Section 7.2.5 [FAQ&AS.NEWSYS.DEVICE], page 22, for information on how to create device-files, if the file '`/dev/reload`' doesn't exist.

6 NetBSD - Linux - Amiga-Unix - Minix

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

The first one is Amiga-Unix. This is a commercial unix from Commodore. 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 machnine-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, the HP-300 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. 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 designed to run on Intel-based PCs. However, Hamish MacDonald did a major rewrite of the sources, so that it is now possible to run Linux on the Amiga. It is now quite useable, but it is not as stable and mature as NetBSD, but this situation might change in the future. The major-drawback with Amiga-Linux is, that it isn't incorporated in the Linux-source-tree. That means that every change, made in the Intel-PC-Unix, must be ported to the Amiga, whereas most of the changes in the NetBSD-source are made to the unique source-tree for all platforms.

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.

7 Frequently Asked Questions and their answers

This chapter contains a list of Frequently Asked Questions (FAQs) and thir answers. It is divided into three sections.

See Chapter 4 [INFORM], page 9 for information on where to find the files and documents mentioned in the answers.

7.1 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 [INFORM], page 9).

7.1.1 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 [INFORM], page 9 for details.

7.1.2 How do I install NetBSD?

There is another document you should read, called (I seem to remember) something logical like '`INSTALL`'; once again, you should be able to find it wherever you found this FAQ. See Chapter 4 [INFORM], page 9 for details.

7.1.3 Does NetBSD run on the 68040?

Thanks to Michael L. Hitch (`osymh@montana.edu`), yes it does!

7.1.4 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`.

7.1.5 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.

7.1.6 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.

7.1.7 NetBSD refuses to boot into MultiUser mode

Make sure the line which reads

```
name_server=NO
```

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

7.1.8 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 FAQ&AS.HARDWR.BINPAT 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.

7.1.9 NetBSD tells me: init FATAL error: console: Interrupted system call

You forgot to install '`libexec/getty`'.

7.1.10 Which distribution should I use?

At the time of writing, there are two principle distributions you could look at, '`NetBSD-1.0`' 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-1.0` 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 [INFORM], page 9 for information on where to find the relevant files.

7.1.11 Does NetBSD run with the PROTO-Chip?

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 mode in the kernel is a binpatch job, and involves *something like* (I'm not sure of the exact symbol name, I'll update this later) patching the symbol `_inhibit_sync` to zero.

7.2 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.

7.2.1 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.

7.2.2 Where is the `uptime` command?

Make the following link:

```
ln -s /usr/bin/w /usr/bin/uptime
```

you can now use the '`uptime`' command as expected.

7.2.3 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 [INFORM], page 9 for details of mailing lists and newsgroups to subscribe to.

7.2.3.1 What about X-Windows?

X servers for NetBSD-Amiga are available for the standard custom chips, and also available (or in development) for a number of graphics cards. See Chapter 4 [INFORM], page 9, as usual, for details of how to find the '`NetBSD-Amiga-X-FAQ`'.

7.2.3.2 ...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 some ethernet cards. See Chapter 4 [INFORM], page 9 for information on locating the excellent '`NetBSD-Amiga-Networking-FAQ`'.

7.2.4 Can I access AmigaDOS files from NetBSD?

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

7.2.5 How do I create device files?

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

```
cd /dev
MKDEV view00
```

7.2.6 How do I create accounts and change user information?

There are two utilities which allow you to edit the password database in relative safety - '`vipw`' and '`chpass`'. '`vipw`' allows you to edit the master password file directly, to add new users and manually change user information, and '`chpass`' lets you alter information such as password expiry dates, group IDs etc. for a given user. You should always use these two utilities to edit the password database because of the password shadowing system used by NetBSD.

7.2.6.1 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.

7.2.7 How can I reboot straight into NetBSD?

Do the following (as root):

```
mv /sbin/reboot /sbin/reboot.amiga
echo "cp /vmunix /dev/reload" >/sbin/reboot
```

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

7.2.8 Why doesnt 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 '/vmunix'. You may also want to make a link from this file to '/netbsd'.

7.3 Hardware, drivers and binpatching

This section details certain specific hardware problems you may run into. See Chapter 4 [INFORM], page 9 for details of how to obtain the file 'compatibility-list'; this lists the devices known to work with NetBSD. The file 'projects' details drivers which are being worked on.

7.3.1 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 19200bps 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 boot up in single-user (ie. root access) mode via that terminal without the root password. Normally the only terminal regarded as secure would be the local screen.

7.3.2 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 7.3.4 [FAQ&AS.HARDWR.BINPAT], page 24 for details of using `binpatch`.

7.3.3 Can I use IDE hard drives?

Yes; thanks to Michael L. Hitch (`osymh@montana.edu`) for making it possible.

7.3.4 How do I use binpatch?

7.3.4.1 What is binpatch, anyway?

'`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 should be able to find the `binpatch` program for AmigaDOS from the same site you downloaded the `loadbsd` program etc. - See Chapter 4 [INFORM], page 9 for details.

7.3.4.2 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 -r 0x02 netbsd
```

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

7.3.4.3 Other 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.

7.3.5 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!

7.3.5.1 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/rewindind. The solution is to power cycle the tapedrive before or after booting NetBSD, forcing it to reset.

7.3.5.2 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 '`vmunix`' to the tape, I would use the command

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

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

```
tar cvfp /dev/rst0 vmunix
```

7.3.6 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.

7.3.6.1 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.

7.3.6.2 So how do I tell the difference?

Don’t bother - just use ‘`/dev/rst8`’ for all data cartridge systems (if it isn’t a 4mm or 8mm tape, it probably is a data cartridge).

7.4 Miscellaneous questions

Some of these could probably live in the "New Systems" section, I had a hard time deciding (which probably means I shouldn’t have bothered); but in general, this is stuff which didn’t fit anywhere else.

7.4.1 Can I update header files automatically?

When new kernel sources are released, it is likely that certain kernel structures will hace changed. Keeping track of the changes by hand is a pain - an easier way is to make symbolic links into the kernel source tree.

In particular, the files under ‘`/usr/include/machine`’ and ‘`/usr/include/sys`’ are often changed. To replace these files with the ones from your kernel source tree, simply make symbolic links as shown below:

```
cd /usr/include
mv sys sys.old
mv machine machine.old
ln -s /usr/src/sys/sys sys
ln -s /usr/src/sys/arch/amiga/include machine
```

7.4.2 Is there a debugger for NetBSD?

The GNU debugger '`gdb`' is available for NetBSD (indeed, I believe it is in the standard comp* distribution package).

7.4.3 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 of the mailing lists (See Chapter 4 [INFORM], 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!).

7.4.4 Why can't NetBSD update the clock?

NetBSD is capable of reading the battery-backed clock, but at the moment it cannot update it; the principle reason is the added complexity introduced because different models of Amiga require different treatment. However, it shouldn't be too hard to add if someone has the time - keep an eye on the mailing lists for information.

7.4.5 vi complains about '`/etc/termcap`'

This is due to a bug in the termcap library; it is looking for the file in '`/usr/share/misc/termcap`', but if it can't find it, produces an error relating to '`/etc/termcap`'.

The solution is simply to copy (or leave) the termcap file in '`/etc/termcap`', and make a symbolic link from '`/usr/share/misc/termcap`' to it.

7.4.6 How do I change the console to autowrap?

Use the command `echo -n "^[[?7h"`. Alternatively, you can change the autowrap default in the file '`ite.c`' to 1, and recompile the kernel.

7.4.7 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.

7.4.8 My keymap is all wrong!

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
./din-kbdmap >din
./loadkmap din
```

x

Table of Contents