NAME

INSTALL – Installation procedure for NetBSD/amiga.

CONTENTS

About this Document....1
What is NetBSD?....2
Upgrade path to NetBSD 1.6....2
Changes Between The NetBSD 1.5 and 1.6 Releases....2
Kernel....3
Networking....3
File system....4
Security....4
System administration and user tools....4
Miscellaneous....5
amiga specific....6
The Future of NetBSD....6
Sources of NetBSD....6
NetBSD 1.6 Release Contents....7
NetBSD/amiga subdirectory structure....8
Miniroot file system....9
Binary distribution sets....9
NetBSD/amiga System Requirements and Supported Devices....10
Supported devices....11
Getting the NetBSD System on to Useful Media....12
Preparing your System for NetBSD installation....15
Preparing your hard disk with HDToolBox....15
Transferring the miniroot file system....16
Installing the NetBSD System....17
Booting....17
Once your kernel boots....18
Post installation steps....20
Upgrading a previously-installed NetBSD System....22
Once your kernel boots....23
Compatibility Issues With Previous NetBSD Releases....24
Issues affecting an upgrade from NetBSD 1.5....24
Issues affecting an upgrade from NetBSD 1.4 or prior....25
Using online NetBSD documentation....25
Administrivia....26
Thanks go to....26
We are....29
Legal Mumbo-Jumbo....34
The End....36

DESCRIPTION

About this Document

This document describes the installation procedure for NetBSD 1.6 on the *amiga* platform. It is available in four different formats titled `INSTALL.`*`ext`*, where *`.ext`* is one of `.ps`, `.html`, `.more`, or `.txt`:

`.ps` PostScript.

`.html` Standard Internet HTML.

`.more` The enhanced text format used on UNIX-like systems by the `more`(1) and `less`(1) pager utility programs. This is the format in which the on-line *man* pages are generally presented.

`.txt` Plain old ASCII.

You are reading the *PostScript* version.

What is NetBSD?

The NetBSD Operating System is a fully functional Open Source UNIX-like operating system derived from the University of California, Berkeley Networking Release 2 (Net/2), 4.4BSD-Lite, and 4.4BSD-Lite2 sources. NetBSD runs on fifty three different system architectures (ports), featuring seventeen machine architectures across eleven distinct CPU families, and is being ported to more. The NetBSD 1.6 release contains complete binary releases for thirty eight different system architectures. (The fifteen remaining are not fully supported at this time and are thus not part of the binary distribution. For information on them, please see the NetBSD web site at **http://www.netbsd.org/**.)

NetBSD is a completely integrated system. In addition to its highly portable, high performance kernel, NetBSD features a complete set of user utilities, compilers for several languages, the X Window System, firewall software and numerous other tools, all accompanied by full source code.

NetBSD is a creation of the members of the Internet community. Without the unique cooperation and coordination the net makes possible, it's likely that NetBSD wouldn't exist.

Upgrade path to NetBSD 1.6

If you are not installing your system ''from scratch'' but instead are going to upgrade an existing system already running NetBSD you need to know which versions you can upgrade with NetBSD 1.6.

NetBSD 1.6 is an upgrade of NetBSD 1.5.3 and earlier major and patch releases of NetBSD.

The intermediate development versions of code available on the main trunk in our CVS repository (also known as ''NetBSD-current'') from *after* the point where the release cycle for 1.6 was started are designated by version identifiers such as 1.6A, 1.6B, etc. These identifiers do not designate releases, but indicate major changes in internal kernel APIs. Note that the kernel from NetBSD 1.6 can *not* be used to upgrade a system running one of those intermediate development versions. Trying to use the NetBSD 1.6 kernel on such a system *will* probably result in problems.

Please also note that it is not possible to do a direct ''version'' comparison between any of the intermediate development versions mentioned above and 1.6 to determine if a given feature is present or absent in 1.6. The development of 1.6 and the subsequent ''point'' releases is done on a separate branch in the CVS repository. The branch was created when the release cycle for 1.6 was started, and during the release cycle of 1.6 and its patch releases selected fixes and enhancements have been imported from the main development trunk.

Changes Between The NetBSD 1.5 and 1.6 Releases

The NetBSD 1.6 release provides numerous significant functional enhancements, including support for many new devices, integration of hundreds of bug fixes, new and updated kernel subsystems, and many user-land enhancements. The result of these improvements is a stable operating system fit for production use that rivals most commercially available systems.

It is impossible to completely summarize over eighteen months of development that went into the NetBSD 1.6 release. Some highlights include:

Kernel

- Ports to new platforms including: algor, dreamcast, evbarm, hpcarm, hpcsh, newsmips, sandpoint, sgimips, sun2, and walnut.
- Unified Buffer Cache (UBC) removes size restriction of the file system's buffer cache to use all available RAM (if not otherwise used!) and improves overall system performance.
- Round-robin page colouring implemented for various ports for better cache utilisation, more deterministic run-time behaviour, and faster program execution.
- A rewritten SCSI middle layer to provide a cleaner interface between the different kernel layers, including a kernel thread to handle error recovery outside of the interrupt context. See `scsipi`(9).
- A new pipe implementation with significantly higher performance due to lower overheads, which uses the UVM Page Loan facility.
- New boot loader flags **`-v`** (*bootverbose*) and **`-q`** (*bootquiet*), to be used by kernel code to optionally print information during boot.
- An in-kernel boot time device configuration manager `userconf`(4), activated with the **`-c`** boot loader flag.
- A work-in-progress snapshot of ACPI support, based on the 20010831 snapshot of the Intel ACPICA reference implementation.
- USB 2.0 support, in the form of a preliminary driver for the `ehci`(4) host controller.
- Basic kernel support for IrDA in the form of the `irframe`(4) IrDA frame level driver. Serial dongles and the `oboe`(4) driver are currently supported.
- Kernel configuration files can be embedded into the kernel for later retrieval. Refer to `INCLUDE_CONFIG_FILE` in `options`(4) for more information.
- Many more kernel tunable variables added to `sysctl`(8).
- Linux binary emulation has been greatly improved, and now supports Linux kernel version 2.4.18.

Networking

- Hardware assisted IPv4 TCP and UDP checksumming and caching of the IPv6 TCP pseudo header. Support for checksum offloading on the DP83820 Gigabit Ethernet, 3Com 3c90xB, 3Com 3c90xC, and Alteon Tigon/Tigon2 Gigabit Ethernet cards.
- Zero-Copy for TCP and UDP transmit path achieved through page loaning code for **`sosend`**().
- In-kernel ISDN support, from the ISDN4BSD project.
- 802.1Q VLAN (virtual LAN) support. See `vlan`(4).
- IPFilter now supports IPv6 filtering.
- `ndbootd`(8) added; used to netboot NetBSD/sun2 machines.
- `racoon`(8) added; IKE key management daemon for IPsec key negotiation, from the KAME project.
- WEP encryption supported in `ifconfig`(8) and `awi`(4) driver.
- `wi`(4) and `wiconfig`(8) now support scanning for access points, and defaults to BSS instead of ad-hoc mode.

- Bridging support; currently only for ethernet. See `bridge`(4).
- In-kernel PPP over Ethernet (PPPoE) - RFC 2516, with much lower overhead than user-land PPPoE clients. See `pppoe`(4).
- `ifwatchd`(8) added; invokes up-script and down-script when a network interface goes up and down. Used by `pppoe`(4).

File system

- Enhanced stability of LFS version 2, the BSD log-structured file system.
- `dump`(8), `dumpfs`(8), `fsck_ffs`(8), `fsirand`(8), `newfs`(8), and `tunefs`(8) support a **`-F`** option to manipulate file system images in regular files.
- `makefs`(8) added; creates file system images from a directory tree. (Currently ffs only.)
- Enhanced **`ffs_dirpref`**( ) by Grigoriy Orlov, which noticeably improves performance on FFS file systems when creating directories, and subsequently manipulating them.
- Fixes for free block tracking and directory block allocation in FFS softdeps.
- Correctly support FFS file systems with a large number of cylinder groups.
- Fix the endian independant FFS (FFS_EI) support.
- `newfs`(8) calculates default block size from the file system size, and uses the largest possible cylinders/group (cpg) value if **`-c`** isn't given.
- `dpti`(4) driver added; an implementation of the DPT/Adaptec SCSI/I2O RAID management interface. Allows the use of the Linux versions of **`dptmgr`**, **`raidutil`**, **`dptelog`**, (etc).
- Support for Windows 2000 'NTFS' (NTFS5).
- Tagged queueing support for SCSI drivers based on the ncr53c9x controller.

Security

- Addition of a `chroot`(8) hierarchy for services including `named`(8), `ntpd`(8), and `sshd`(8).
- Additional `passwd`(5) ciphers: MD5, and DES with more encryption rounds. See `passwd.conf`(5).
- Several more code audits were performed.
- `/etc/security` performs many more checks and is far more flexible in how it monitors changes. See `security.conf`(5).

System administration and user tools

- `sushi`(8) added; a menu based system administration tool.
- `pgrep`(1) and `pkill`(1) added; find or signal processes by name or other attributes.
- System upgrades are made easier through the `etcupdate`(8) script which helps updating the `/etc` config files interactively, and the `/etc/postinstall` script which is provided to check for or fix configuration changes that have occurred in NetBSD.
- `stat`(1) added; a user interface to the information returned by the `stat`(2) system call.
- BSD `sort`(1) replaces GNU `sort`(1).
- The "stop" operation for `rc.d`(8) scripts waits until the service terminates before returning. This improves the reliability of "restart" operations as well.

- Swap devices can be removed at system shutdown by enabling `swapoff` in `rc.conf`(5).
- An optional watchdog timer which will terminate `rc.shutdown`(8) after the number of seconds provided in `rcshutdown_timeout` from `rc.conf`(5).

Miscellaneous

- Support for multibyte LC_CTYPE locales has been integrated from the Citrus project. Many Chinese, Japanese, Korean, and other encodings are now available.
- Full support for cross-compilation of the base system, even as a non-root user! `src/build.sh` is available for doing arbitrary cross-builds; see `src/BUILDING` for more information. At least 38 ports for the NetBSD 1.6 release were cross-built on a NetBSD/i386 system using this mechanism.
- Migrated the following CPU platforms to ELF: arm, and m68k (including amiga, hp300, mac68k, mvme68k, sun2, and x68k).
- Updates of most third party packages that are shipped in the base system to the following latest stable releases:
  - amd 6.0.6
  - BIND 8.3.3
  - binutils 2.11.2
  - bzip2 1.0.2
  - cvs 1.11
  - dhcp 3.0.1rc9
  - file 3.38
  - gcc 2.95.3
  - groff 1.16.1
  - Heimdal 0.4e
  - IPfilter 3.4.27
  - kerberos4 1.1
  - ksh from pdksh 5.2.14p2
  - less 374
  - nvi 1.79
  - OpenSSH 3.4
  - OpenSSL 0.9.6g
  - Postfix 1.1.11
  - ppp 2.4.0
  - routed 2.24
  - sendmail 8.11.6
  - tcpdump 3.7.1
- Many new packages in the *pkgsrc* system, including the latest open source desktop KDE3, OpenOffice, perl, Apache and many more. At the time of writing, there are over 3000 third party packages available in pkgsrc.
- Added AGP GART driver `agp`(4) for faster access to graphics boards.
- `init`(8) will create an mfs (memory based file system) `/dev` if `/dev/console` is missing.
- `vmstat`(8) displays kernel hash statistics with **`-H`** and **`-h`** *`hash`*.
- `wscons`(4) supports blanking of VGA consoles.

Kernel interfaces have continued to be refined, and more subsystems and device drivers are shared among the different ports. You can look for this trend to continue.

amiga specific

This is the eighth major release of NetBSD for the Amiga and DraCo line of computers.

New port-specific features include:

- Two stage boot loader capable of loading ELF or a.out as well as compressed kernels.
- New device driver support:
  - ZEUS ISDN-link and Individual Computers ISDN surfer
  - BSC ISDN MAster II
  - Repulse audio board
  - Toccata driver (experimental)

The Future of NetBSD

The NetBSD Foundation has been incorporated as a non-profit organization. Its purpose is to encourage, foster and promote the free exchange of computer software, namely the NetBSD Operating System. The foundation will allow for many things to be handled more smoothly than could be done with our previous informal organization. In particular, it provides the framework to deal with other parties that wish to become involved in the NetBSD Project.

The NetBSD Foundation will help improve the quality of NetBSD by:

- providing better organization to keep track of development efforts, including co-ordination with groups working in related fields.
- providing a framework to receive donations of goods and services and to own the resources necessary to run the NetBSD Project.
- providing a better position from which to undertake promotional activities.
- periodically organizing workshops for developers and other interested people to discuss ongoing work.

We intend to begin narrowing the time delay between releases. Our ambition is to provide a full release every six to eight months.

We hope to support even *more* hardware in the future, and we have a rather large number of other ideas about what can be done to improve NetBSD.

We intend to continue our current practice of making the NetBSD-current development source available on a daily basis.

We intend to integrate free, positive changes from whatever sources submit them, providing that they are well thought-out and increase the usability of the system.

Above all, we hope to create a stable and accessible system, and to be responsive to the needs and desires of NetBSD users, because it is for and because of them that NetBSD exists.

Sources of NetBSD

Refer to **http://www.netbsd.org/Sites/net.html**.

NetBSD 1.6 Release Contents

The root directory of the NetBSD 1.6 release is organized as follows:

`.../NetBSD-1.6/`

`CHANGES`	Changes since earlier NetBSD releases.
`LAST_MINUTE`	Last minute changes.
`MIRRORS`	A list of sites that mirror the NetBSD 1.6 distribution.
`README.files`	README describing the distribution's contents.
`TODO`	NetBSD 's todo list (also somewhat incomplete and out of date).
`patches/`	Post-release source code patches.
`source/`	Source distribution sets; see below.

In addition to the files and directories listed above, there is one directory per architecture, for each of the architectures for which NetBSD 1.6 has a binary distribution. There are also `README.export-control` files sprinkled liberally throughout the distribution tree, which point out that there are some portions of the distribution that may be subject to export regulations of the United States, e.g. code under `src/crypto` and `src/sys/crypto`. It is your responsibility to determine whether or not it is legal for you to export these portions and to act accordingly.

The source distribution sets can be found in subdirectories of the `source` subdirectory of the distribution tree. They contain the complete sources to the system. The source distribution sets are as follows:

gnusrc This set contains the ''gnu'' sources, including the source for the compiler, assembler, groff, and the other GNU utilities in the binary distribution sets.
55 MB gzipped, 247 MB uncompressed

pkgsrc This set contains the ''pkgsrc'' sources, which contain the infrastructure to build third-party packages.
12 MB gzipped, 94 MB uncompressed

sharesrc This set contains the ''share'' sources, which include the sources for the man pages not associated with any particular program; the sources for the typesettable document set; the dictionaries; and more.
4 MB gzipped, 16 MB uncompressed

src This set contains all of the base NetBSD 1.6 sources which are not in **gnusrc**, **sharesrc**, or **syssrc**.
27 MB gzipped, 136 MB uncompressed

syssrc This set contains the sources to the NetBSD 1.6 kernel for all architectures; `config`(8); and `dbsym`(8).
22 MB gzipped, 114 MB uncompressed

xsrc This set contains the sources to the X Window System.
78 MB gzipped, 394 MB uncompressed

All the above source sets are located in the `source/sets` subdirectory of the distribution tree.

The source sets are distributed as compressed tar files. Except for the **pkgsrc** set, which is traditionally unpacked into `/usr/pkgsrc`, all sets may be unpacked into `/usr/src` with the command:

```
# ( cd / ; tar -zxpf - ) < set_name.tgz
```

The sets/Split/ subdirectory contains split versions of the source sets for those users who need to load the source sets from floppy or otherwise need a split distribution. The split sets are named set_name.*xx* where set_name is the distribution set name, and *xx* is the sequence number of the file, starting with "aa" for the first file in the distribution set, then "ab" for the next, and so on. All of these files except the last one of each set should be exactly 240,640 bytes long. (The last file is just long enough to contain the remainder of the data for that distribution set.)

The split distributions may be reassembled and extracted with **cat** as follows:

```
# cat set_name.?? | ( cd / ; tar -zxpf - )
```

In each of the source distribution set directories, there are files which contain the checksums of the files in the directory:

BSDSUM Historic BSD checksums for the various files in that directory, in the format produced by the command:
cksum -o 1 *file*.

CKSUM POSIX checksums for the various files in that directory, in the format produced by the command:
cksum *file*.

MD5 MD5 digests for the various files in that directory, in the format produced by the command:
cksum -m *file*.

SYSVSUM Historic AT&T System V UNIX checksums for the various files in that directory, in the format produced by the command:
cksum -o 2 *file*.

The MD5 digest is the safest checksum, followed by the POSIX checksum. The other two checksums are provided only to ensure that the widest possible range of system can check the integrity of the release files.

NetBSD/amiga subdirectory structure

The amiga-specific portion of the NetBSD 1.6 release is found in the amiga subdirectory of the distribution:

```
.../NetBSD-1.6/amiga/
INSTALL.html
INSTALL.ps
INSTALL.txt
INSTALL.more Installation notes in various file formats, including this file. The .more file contains underlined text using the more(1) conventions for indicating italic and bold display.
binary/
             kernel/
                     netbsd-GENERIC.gz
                                  A gzipped NetBSD kernel containing code for everything supported in this release.
                     netbsd.INSTALL.gz
                                  A somewhat smaller kernel, which you can use to boot the system on memory-tight systems. This is the same kernel as present on the miniroot.
             sets/   amiga binary distribution sets; see below.
installation/
             floppy/ amiga boot and installation floppies; see below.
```

`miniroot/`
amiga miniroot file system image; see below.
`misc/` Miscellaneous amiga installation utilities; see installation section, below.

Miniroot file system

The Amiga now uses a single miniroot file system for both an initial installation and for an upgrade. A gzipped version is available, for easier downloading. (The gzipped version has the **.gz** extension added to their names.)

`miniroot.fs` This file contains a BSD root file system setup to help you install the rest of NetBSD or to upgrade a previous version of NetBSD. This includes formatting and mounting your `/` (root) and `/usr` partitions and getting ready to extract (and possibly first fetching) the distribution sets. There is enough on this file system to allow you to make a SLIP or PPP connection, configure an Ethernet, mount an NFS file system or ftp. You can also load distribution sets from a SCSI tape or from one of your existing AmigaDOS partitions.

Binary distribution sets

The NetBSD amiga binary distribution sets contain the binaries which comprise the NetBSD 1.6 release for the amiga. There are eight binary distribution sets. The binary distribution sets can be found in the `amiga/binary/sets` subdirectory of the NetBSD 1.6 distribution tree, and are as follows:

base The NetBSD 1.6 amiga **base** binary distribution. You *must* install this distribution set. It contains the base NetBSD utilities that are necessary for the system to run and be minimally functional. It includes shared library support, and excludes everything described below.
17 MB gzipped, 41 MB uncompressed

comp Things needed for compiling programs. This set includes the system include files (`/usr/include`) and the various system libraries (except the shared libraries, which are included as part of the **base** set). This set also includes the manual pages for all of the utilities it contains, as well as the system call and library manual pages.
14 MB gzipped, 52 MB uncompressed

etc This distribution set contains the system configuration files that reside in `/etc` and in several other places. This set *must* be installed if you are installing the system from scratch, but should *not* be used if you are upgrading.
1 MB gzipped, 1 MB uncompressed

games This set includes the games and their manual pages.
3 MB gzipped, 7 MB uncompressed

kern-GENERIC
This set contains a NetBSD/amiga 1.6 `GENERIC` kernel, named `/netbsd`. You *must* install this distribution set.
2 MB gzipped, 3 MB uncompressed

man This set includes all of the manual pages for the binaries and other software contained in the **base** set. Note that it does not include any of the manual pages that are included in the other sets.
7 MB gzipped, 27 MB uncompressed

misc This set includes the (rather large) system dictionaries, the typesettable document set, and other files from `/usr/share`.
3 MB gzipped, 8 MB uncompressed

text This set includes NetBSD's text processing tools, including `groff`(1), all related programs, and their manual pages.
2 MB gzipped, 6 MB uncompressed

NetBSD maintains its own set of sources for the X Window System in order to assure tight integration and compatibility. These sources are based on XFree86, and tightly track XFree86 releases. They are currently equivalent to XFree86 3.3.6. Binary sets for the X Window System are distributed with NetBSD. The sets are:

xbase The basic files needed for a complete X client environment. This does not include the X servers.
3 MB gzipped, 7 MB uncompressed

xcomp The extra libraries and include files needed to compile X source code.
2 MB gzipped, 8 MB uncompressed

xcontrib Programs that were contributed to X.
1 MB gzipped, 1 MB uncompressed

xfont Fonts needed by X.
6 MB gzipped, 7 MB uncompressed

xmisc Miscellaneous X programs.
1 MB gzipped, 1 MB uncompressed

xserver
2 MB gzipped, 4 MB uncompressed

The amiga binary distribution sets are distributed as gzipped tar files named with the extension **.tgz**, e.g. `base.tgz`.

The instructions given for extracting the source sets work equally well for the binary sets, but it is worth noting that if you use that method, the filenames stored in the sets are relative and therefore the files are extracted *below the current directory*. Therefore, if you want to extract the binaries into your system, i.e. replace the system binaries with them, you have to run the **`tar -xpf`** command from the root directory (`/`) of your system. This utility is used only in a Traditional method installation.

Note: Each directory in the amiga binary distribution also has its own checksum files, just as the source distribution does.

NetBSD/amiga System Requirements and Supported Devices

NetBSD 1.6 runs on any Amiga that has a 68020 or better CPU with some form of MMU, and on 68060 DraCos.

NetBSD does not, and will never, run on run on A1000, A500, A600, A1200, A2000, A4000/EC030, CDTV and CD32 systems that are not enhanced by a CPU board.

For 68020 and 68030 systems, a FPU is recommended but not required for the system utilities. 68LC040, 68040V and 68LC060 systems don't work correctly at the moment.

The minimal configuration requires 6 MB of RAM (not including CHIPMEM!) and about 75 MB of disk space. To install the entire system requires much more disk space, and to run X or compile the system, more RAM is recommended. (6 MB of RAM will actually allow you to compile, however it won't be speedy. X really isn't usable on a 6 MB system.)

Here is a table of recommended HD partition sizes for a full install:

Partition	Suggested	+ X	Needed	+ X
/ (root)	25 MB	25 MB	20 MB	20 MB
/usr	245 MB	270 MB	120 MB	145 MB
/var	20 MB	20 MB	5 MB	5 MB
swap	*2*RAM below 32 MB, then up to you*			

Anything else is up to you!

As you may note the recommended size of `/usr` is 125 MB greater than needed. This is to leave room for a kernel source and compile tree as you will probably want to compile your own kernel. `GENERIC` is large and bulky to accommodate all people. For example, most people's machines have an FPU, so you do not need the bulky FPU_EMULATE option.

Preconfigured or precompiled packages are installed below `/usr/pkg` by default. You should either make `/usr` larger (if you intend to install a lot of them), make `/usr/pkg` an additional partition, use the **`-p`** option to **`pkg_add`** to install them in a different place, or link `/usr/pkg` to a different place.

If you only have less than 8 MB of fast memory, you should make your swap partition larger, as your system will be doing much more swapping. Especially: do *not* place it onto a old small (and normally slow) disk!

Supported devices

- A4000/A1200 IDE controller, including ATAPI devices
- SCSI host adapters
  - 33c93 based boards: A2091, A3000 builtin, A3000 builtin modified for Apollo accelerator board, and GVP series II.
  - 53c80 based boards: 12 Gauge, IVS, Wordsync/Bytesync and Emplant The Emplant SCSI adapter has been reported by a party to hang after doing part of the installation without problems
  - 53c710 based boards: A4091, Magnum, Warp Engine, Zeus and DraCo builtin
  - FAS216 based SCSI boards: FastLane Z3, Blizzard I and II, Blizzard IV, Blizzard 2060, CyberSCSI Mk I and II
  - 53c770 based SCSI boards: Cyberstorm Mk III SCSI, Cyberstorm PPC SCSI
- Video controllers
  - ECS, AGA and A2024 built in on various Amigas
  - Retina Z2 (no X server available), Retina Z3 and Altais
  - Cirrus CL GD 54xx based boards: GVP Spectrum, Picasso II, II+ and IV, Piccolo and Piccolo SD64
  - Tseng ET4000 based boards: Domino and Domino16M proto, oMniBus, Merlin
  - A2410 (no X server available)
  - Cybervision 64
  - Cybervision 64/3D
- Audio I/O
  - Amiga builtin
  - Melody Mpeg-audio layer 2 board
  - Repulse audio board
- Ethernet controllers
  - A2065 Ethernet
  - Hydra Ethernet
  - ASDG Ethernet

  - A4066 Ethernet
  - Ariadne Ethernet
  - Ariadne II Ethernet
  - Quicknet Ethernet
  - X-surf Ethernet port
- ARCnet controllers
  - A2060 ARCnet
- Most SCSI tape drives, including Archive Viper, Cipher SCSI-2 ST150
- SCSI-2 scanners behaving as SCSI-2 scanner devices, HP Scanjet II, Mustek SCSI scanner. SCSI scanner support is machine independent, so it should work, but hasn't been tested yet on most Amiga configurations. There are reports that the Mustek and HP ScanJet hang if accessed from the A3000. This might apply to other 33C93-Adapters, too.
- Most SCSI CD-ROM drives
- Serial/Parallel cards
  - HyperCom 3Z, HyperCom 4, HyperCom 3+ and 4+
  - MultiFaceCard II and III
  - A2232 (normal and clockdoubled)
  - IOBlix Zorro-Bus
- Amiga floppy drives with Amiga (880/1760kB) and IBM (720/1440kB) encoding. Our floppy driver doesn't notice when mounted floppies are write-protected at the moment. Your floppy will stay unchanged, but you might not notice that you didn't write anything due to the buffer cache. Also note that HD floppy drives only get detected as such if a HD floppy is inserted at boot time.
- Amiga parallel port
- Amiga serial port
- Amiga mouse
- DraCo serial port, including serial mouse
- DraCo parallel printer port
- Real-time clocks
  - A2000, A3000, A4000 builtin (r/w)
  - DraCo builtin (r/o)

If its not on the above lists, there is no support for it in this release. Especially (but this is an incomplete list), there are no drivers for: Blizzard III SCSI option, Ferret SCSI, Oktagon SCSI.

Getting the NetBSD System on to Useful Media

Installation is supported from several media types, including:

- AmigaDOS
- CD-ROM
- MS-DOS floppy
- FTP
- Remote NFS partition
- Tape
- Existing NetBSD partitions, if performing an upgrade

The steps necessary to prepare the distribution sets for installation depend upon which installation medium you choose. The steps for the various media are outlined below.

AmigaDOS partition

To install NetBSD from an AmigaDOS partition, you need to get the NetBSD distribution sets you wish to install on your system on to an AmigaDOS partition. All of the `set_name.xx` pieces can be placed in a single directory instead of separate ones for each distribution set. This will also simplify the installation work later on.

Note where you place the files as you will need this later.

Once you have done this, you can proceed to the next step in the installation process, preparing your hard disk.

CD-ROM

To install NetBSD from a CD-ROM drive, make sure it is a SCSI CD-ROM on a SCSI bus currently supported by NetBSD (refer to the supported hardware list) or an ATAPI cd-rom connected to the A1200 or A4000 internal IDE connector. If it is a SCSI CD-ROM on a non-supported SCSI bus like Blizzard-3 SCSI or Apollo SCSI you must first copy the distribution sets to an AmigaDOS partition as described above.

If your SCSI CD-ROM is connected to a supported SCSI host adapter, or it is an ATAPI cd-rom connected to the A1200/A4000 internal IDE connector, simply put the CD into the drive before installation.

Find out where the distribution set files are on the CD-ROM.

Proceed to the instruction on installation.

MS-DOS floppy

Count the number of `set_name.`*`xx`* files that make up the distribution sets you want to install or upgrade. You will need one sixth that number of 1.44 MB floppies.

Format all of the floppies with MS-DOS. Do *not* make any of them bootable MS-DOS floppies, i.e. don't use `format /s` to format them. (If the floppies are bootable, then the MS-DOS system files that make them bootable will take up some space, and you won't be able to fit the distribution set parts on the disks.) If you're using floppies that are formatted for MS-DOS by their manufacturers, they probably aren't bootable, and you can use them out of the box.

Place all of the `set_name.`*`xx`* files on the MS-DOS disks.

Once you have the files on MS-DOS disks, you can proceed to the next step in the installation or upgrade process. If you're installing NetBSD from scratch, go to the section on preparing your hard disk, below. If you're upgrading an existing installation, go directly to the section on upgrading.

FTP

The preparations for this installation/upgrade method are easy; all you need to do is make sure that there's an FTP site from which you can retrieve the NetBSD distribution when you're about to install or upgrade. You need to know the numeric IP address of that site, and, if it's not on a network directly connected to the machine on which you're installing or upgrading NetBSD, you need to know the numeric IP address of the router closest to the NetBSD machine. Finally, you need to know the numeric IP address of the NetBSD machine itself. If you don't have access to a functioning nameserver during installation, the IP address of **ftp.netbsd.org** is `204.152.184.75` (as of June, 2002).

Once you have this information, you can proceed to the next step in the installation or upgrade process. If you're installing NetBSD from scratch, go to the section on preparing your hard disk, below. If you're upgrading an existing installation, go directly to the section on upgrading.

Note: This method of installation is recommended only for those already familiar with using BSD network configuration and management commands. If you aren't, this documentation should help, but is not intended to be all-encompassing.

NFS

Place the NetBSD distribution sets you wish to install into a directory on an NFS server, and make that directory mountable by the machine on which you are installing or upgrading NetBSD. This will probably require modifying the `/etc/exports` file on of the NFS server and resetting its mount daemon (mountd). (Both of these actions will probably require superuser privileges on the server.)

You need to know the numeric IP address of the NFS server, and, if the server is not on a network directly connected to the machine on which you're installing or upgrading NetBSD, you need to know the numeric IP address of the router closest to the NetBSD machine. Finally, you need to know the numeric IP address of the NetBSD machine itself.

Once the NFS server is set up properly and you have the information mentioned above, you can proceed to the next step in the installation or upgrade process. If you're installing NetBSD from scratch, go to the section on preparing your hard disk, below. If you're upgrading an existing installation, go directly to the section on upgrading.

Note: This method of installation is recommended only for those already familiar with using BSD network configuration and management commands. If you aren't, this documentation should help, but is not intended to be all-encompassing.

Tape

To install NetBSD from a tape, you need to make a tape that contains the distribution set files, in 'tar' format.

If you're making the tape on a UNIX-like system, the easiest way to do so is probably something like:

```
# tar -cf tape_device dist_directories
```

where *tape_device* is the name of the tape device that describes the tape drive you're using; possibly `/dev/rst0`, or something similar, but it will vary from system to system. (If you can't figure it out, ask your system administrator.) In the above example, *dist_directories* are the distribution sets' directories, for the distribution sets you wish to place on the tape. For instance, to put the **misc, base,** and **etc** distributions on tape (in order to do the absolute minimum installation to a new disk), you would do the following:

```
# cd .../NetBSD-1.6
# cd amiga/binary
# tar -cf tape_device misc etc kern
```

Note: You still need to fill in *tape_device* in the example.

Once you have the files on the tape, you can proceed to the next step in the installation or upgrade process. If you're installing NetBSD from scratch, go to the section on preparing your hard disk, below. If you're upgrading an existing installation, go directly to the section on upgrading.

Upgrade

If you are upgrading NetBSD, you also have the option of installing NetBSD by putting the new distribution sets somewhere in your existing file system, and using them from there. To do that, you must do the following:

Place the distribution sets you wish to upgrade somewhere in your current file system tree. Please note that the `/dev` on the floppy used for upgrades only knows about `wd0`, `wd1`, `sd0`, `sd1`, and `sd2`. If you have more than two IDE drives or more than three SCSI drives, you should take care not to place the sets on the high-numbered drives.

At a bare minimum, you must upgrade the **base** and **kern** binary distributions, and so must put the **base** and **kern** sets somewhere in your file system. If you wish, you can do the other sets, as well, but you should *not* upgrade the **etc** distribution; it contains contains system configuration files that you should review and update by hand.

Once you have done this, you can proceed to the next step in the upgrade process, actually upgrading your system.

Preparing your System for NetBSD installation

You will need an AmigaDOS hard drive prep tool to prepare your hard drives for use with NetBSD/amiga. HDToolBox is provided with the system software and on floppy installation disks since Release 2.0 of AmigaDOS, so we will provide instructions for its use.

Note that NetBSD can't currently be installed on disks with a sector size other than 512 bytes (e.g., ''640 MB'' 90mm M-O media). You can, however, mount ADOSFS partitions on such M-O's.

Preparing your hard disk with HDToolBox

A full explanation of HDToolBox can be found with your AmigaDOS manuals and is beyond the scope of this document.

The first time you partition a drive, you need to set its drive type so that you have working geometry parameters. To do this you enter the ''Change drive type'' menu, and either use ''read parameters from drive'' or set them manually.

Note you will be modifying your HD's. If you mess something up here you could lose everything on all the drives that you configure. It is therefore advised that you:

1. Write down your current configurations. Do this by examining each partition on the drive and the drives parameters (from Change drive type.)
2. Back up the partitions you are keeping.

What you need to do is partition your drives; creating at least root, swap and `/usr` partitions and possibly at least one more for `/usr/local` if you have the space.

This should be done as the HDToolBox manual describes. One thing to note is that if you are not using a Commodore controller you will need to specify the device your SCSI controller uses, e.g. if you have a Warp Engine you would:

From cli

```
hdtoolbox warpdrive.device
```

From wb Set the tooltype

```
SCSI_DEVICE_NAME=warpdrive.device
```

The important things you need to do above and beyond normal partitioning includes (from Partition Drive section):

1. Marking all NetBSD partitions as non-bootable, with two exceptions: the root partition (`/`), if you want to boot NetBSD directly, or the swap partition, if you want to boot the installation miniroot directly.

2. Changing the file system parameters of the partitions to NetBSD ones. This must be done from the partitioning section and ''Advanced options'' must be enabled.

To make the needed changes:

1. Click the ''Adv. Options'' button
2. Click the ''Change file system'' button
3. Choose ''Custom File System''
4. Turn off ''Automount'' if on.
5. Set the dostype to one of these three choices:

```
root partition  : 0x4e425207  (NBR\007)
swap partition  : 0x4e425301  (NBS\001)
other partitions: 0x4e425507  (NBU\007)
```

   Here '`other`' refers to other partitions you will format for reading and writing under NetBSD (e.g. `/usr`)

   Make sure you press `RETURN` to enter this value as some versions of HDToolBox will forget your entry if you don't.
6. Turn custom boot code off
7. Set Reserved Blocks start and end to 0.
8. Click Ok.

On the root (`/`) (and, for installation, swap) partition, set instead this:

1. Turn custom boot code on
2. Set Reserved Blocks start and end to 0.
3. Set Number of Custom Boot Blocks to 16.
4. Set Automount This Partition on
5. Click Ok.

   Mask and maxtransfer are not used with NetBSD.

Caveat: The swap (for installation) and the root partition (if you plan to use the bootblocks) *must be within the first* 4 gigabytes of the disk! The reason for the former is that xstreamtodev uses trackdisk.device compatible I/O-calls, the reason for the latter is that the bootblock gets a 32bit partition offset from the operating system.

Once this is done NetBSD/amiga will be able to recognize your disks and which partitions it should use.

Transferring the miniroot file system

The NetBSD/amiga installation or upgrade now uses a *miniroot* file system which is installed on the partition used by NetBSD for swapping. This removes the requirement of using a floppy disk for the file system used by the installation or upgrade process. It also allows more utilities to be present on the file system than would be available when using an 880 KB floppy disk.

Once the hard disk has been prepared for NetBSD, the miniroot file system (`miniroot.fs`) is transferred to the swap partition configured during the hard disk prep (or the existing swap partition in the case of an upgrade). The xstreamtodev utility provided in the `amiga/installation/misc` directory can be used on AmigaDOS to transfer the file system for either a new installation or an upgrade. The file system can also be transferred on an existing NetBSD system for an update by using dd. This should only be done after booting NetBSD into single-user state. It may also be possible to shutdown to single-user, providing that the single-user state processes are not using the swap partition.

On AmigaDOS, the command:

xstreamtodev --input=miniroot.fs --rdb-name=<swap partition>

where <swap partition> is the name you gave to the NetBSD partition to be used for swapping. If **xstreamtodev** is unable to determine the SCSI driver device name or the unit number of the specified partition, you may also need to include the option

--device=<driver.name>

and/or

--unit=<SCSI unit number>

To transfer the miniroot using NetBSD, you should be booted up in single user state on the current NetBSD system, or use the **shutdown now** command to shutdown to single-user state. Then copy the miniroot using **dd**:

dd if=miniroot.fs of=/dev/rsd0b

where `/dev/rsd0b` should be the device path of the swap partition your system is configured to use. Once the file is copied, reboot back to AmigaDOS to boot the upgrade kernel.

Caveat: Once you have started installation, if you abort it and want to retry you must reinstall the `miniroot.fs` on the swap partition.

Installing the NetBSD System

Installing NetBSD is a relatively complex process, but, if you have this document in hand and are careful to read and remember the information which is presented to you by the install program, it shouldn't be too much trouble.

Before you begin, you must have already prepared your hard disk as detailed in the section on preparing your system for install.

The following is a walk-through of the steps necessary to get NetBSD installed on your hard disk. If you wish to stop the installation, you may press `CONTROL-C` at any prompt, but if you do, you'll have to begin again from scratch.

Transfer the miniroot file system onto the hard disk partition used by NetBSD for swapping, as described in the "Preparing your System for NetBSD Installation" section above.

Booting

NetBSD, with boot blocks installed

[This description is for V40 (OS 3.1) ROMs. For older ROMs, there might be small differences. Check your AmigaDOS documentation to learn about the exact procedure.] Using bootblocks may not work on some systems, and may require a mountable file system on others.

Reboot your machine, holding down both mouse buttons if you have a 2-button mouse, the outer mouse buttons if you have a 3-button mouse. On the DraCo, press the left mouse button instead, when the boot screen prompts you for it.

From the boot menu, select **Boot Options**. Select the swap partition with the miniroot, and then **ok**. Select **Boot** now. The machine will boot the bootblock, which will prompt your for a command line. You have a few seconds time to change the default. Entering an empty line will accept the default.

The bootblock uses command lines of the form:

file [*options*]

where *file* is the kernel file name on the partition where the boot block is on, and [options] may contain the following:

If you have an AGA machine, and your monitor will handle the dblNTSC mode, you may include the **-A** option to enable the dblNTSC display mode.

If your machine has a fragmented physical memory space, as, e.g., DraCo machines, you should add the **-n2** option to enable the use of all memory segments.

Once your kernel boots

You should see the screen clear and some information about your system as the kernel configures the hardware. Note which hard disk device(s) are configured (`sd0`, `sd1`, etc.) Then you will be prompted for a root device. At this time type `sd0b`, where `sd0` is the device which contains the swap partition you created during the hard disk preparation. When prompted for a dump device, answer '**none**' for the install (normally, you would tell it one of the swap devices). When prompted for the root file system type, confirm '**generic**', which will auto-detect it.

If the system should hang after entering the root device, try again with

netbsd -I ff -b

This disables synchronous transfer on all SCSI devices on the first bus.

The system should continue to boot. For now ignore "WARNING" messages about bad dates in clocks, and a warning about `/etc/rc` not existing. Eventually you will be be asked to enter the pathname of the shell, just press `RETURN`. After a short while, you will be asked to select the type of your keyboard. After you have entered a valid response here, the system asks you if you want to install or upgrade your system. Since you are reading the *install* section, '**i**' would be the proper response here...

The installer starts with a nice welcome messages. Read this message carefully, it also informs you of the risks involved in continuing! If you still want to go on, type '**y**'. The installer now continues by trying to figure out your disk configuration. When it is done, you will be prompted to select a root device from the list of disks it has found.

You should know at this point that the disks are *not* numbered according to their SCSI-ID! The NetBSD kernel numbers the SCSI drives (and other devices on the SCSI bus) sequentially as it finds them. The drive with the lowest SCSI-ID will be called `sd0`, the next one `sd1`, etc. Also, any ATAPI disk drives (e.g. ZIP) will be configured as "SCSI" drives, too, and will be configured before any 'real' SCSI drives if connected to the Amiga internal port on A4000/A1200 (if any are present). Real IDE drives will be configured as `wd0`, `wd1`, etc.

The installer will offer you to look at the NetBSD disk label of the disks at this point. You should do this, to find out what partition letters the NetBSD kernel assigned to the partitions you created, and as a check whether the disk number you are going to use is right.

you are now at the point of no return. If you confirm that you want to install NetBSD, your hard drive will be modified, and perhaps its contents scrambled at the whim of the install program. Type `Control-C` *now* if you don't want this.

At this time, you will need to tell the installer which partition will be associated with the different file systems. Normally, you'll want to add a partition for `/usr`, at least.

Caveat: Do not use the `rsd`*N*`c` or `sd`*N*`c` partitions for anything! They are for access to the whole disk only and do *not* correspond to any Amiga partition!

The install program will now make the file systems you specified. There should be only one error per file system in this section of the installation. It will look like this:

```
newfs: ioctl (WDINFO): Invalid argument
newfs: /dev/rsd0a: can't rewrite disk label
```

If there are any others, restart from the beginning of the installation process. This error is ok as the Amiga does not write disklabels currently. You should expect this error whenever using newfs.

The install will now ask you want to configure any network information. It ill ask for the machine's host name, domain name, and other network configuration information.

Since the network configuration might have lead to additional (nfs) file system entries, you get another chance to modify your fstab.

You are finally at the point where some real data will be put on your freshly made file systems. Select the device type you wish to install from and off you go....

Some notes:

- If you want to install from tape, please read the section about how to create such a tape.
- Some tapes (e.g. Archive Viper 150) refuse to operate with the default tape density (`nrst0`). Try `nrst0h`, `nrst0m`, or `nrst0l` instead.
- Install at least the **base** and **etc** sets.
- If you have to specify a path relative to the mount-point and you need the mount-point itself, use '`.`'.

Next you will be asked to specify the timezone. Just select the timezone you are in. The installer will make the correct setup on your root file system (`/`). After the timezone-link is installed, the installer will proceed by creating the device nodes on your root file system under `/dev`.

Be patient, this will take a while...

Next, the installer will copy your keymap settings to the new system. After this, it will copy the kernel from the installation miniroot to the newly installed / upgraded system. If the installed system already has a kernel, it will ask you for confirmation.

kern.tgz distribution set, this is an old kernel, and you should answer "y" to install a working (although restricted) INSTALL kernel.

If you did install the kern.tgz kernel, you normally should answer "n".

Finally, the installer ask you if you want to install the bootblock code on your root disk. This is a matter of personal choice and can also be done from a running NetBSD system. See the `installboot`(8) manual page about how to do this.

Once the installer is done, halt the system with the **`halt`** command (wait for `halted` to be displayed) and reboot. Then again boot NetBSD this time selecting the root partition (`/`) from the boot menu, and tell it to boot

```
netbsd -s
```

You need to do your final tweaks now. First mount your file systems like so:

```
mount -av
```

Your system is now complete but not completely configured; you should adjust the `/etc/sendmail.cf` file as necessary to suit your site. You should also examine and adjust the settings in `/etc/rc.conf`. You can use `vi`(1) or `ed`(1) to edit the files. If you installed the man pages you can type **`man vi`** or **`man ed`** for instructions on how to use these somewhat non-intuitive editors.

Once you are done with the rest of configuration unmount your file systems and halt your system, then reboot:

```
# cd /
# umount -av
# halt
```

Finally you can now boot your system and it will be completely functional:

```
netbsd
```

When it boots off of the hard drive, you will have a complete NetBSD system! *Congratulations!* (You really deserve them!!!)

Post installation steps

Once you've got the operating system running, there are a few things you need to do in order to bring the system into a properly configured state, with the most important ones described below.

1. Configuring `/etc/rc.conf`

   If you or the installation software haven't done any configuration of `/etc/rc.conf` (**sysinst** usually will), the system will drop you into single user mode on first reboot with the message

   ```
   /etc/rc.conf is not configured. Multiuser boot aborted.
   ```

   and with the root file system (`/`) mounted read-only. When the system asks you to choose a shell, simply press `RETURN` to get to a `/bin/sh` prompt. If you are asked for a terminal type, respond with **vt220** (or whatever is appropriate for your terminal type) and press `RETURN`. You may need to type one of the following commands to get your delete key to work properly, depending on your keyboard:

   ```
   # stty erase '^h'
   # stty erase '^?'
   ```

   At this point, you need to configure at least one file in the `/etc` directory. You will need to mount your root file system read/write with:

   ```
   # /sbin/mount -u -w /
   ```

   Change to the `/etc` directory and take a look at the `/etc/rc.conf` file. Modify it to your tastes, making sure that you set `rc_configured=YES` so that your changes will be enabled and a multi-user boot can proceed. Default values for the various programs can be found in `/etc/defaults/rc.conf`, where some in-line documentation may be found. More complete documentation can be found in `rc.conf`(5).

   If your `/usr` directory is on a separate partition and you do not know how to use **ed**, you will have to mount your `/usr` partition to gain access to **ex** or **vi**. Do the following:

   ```
   # mount /usr
   # export TERM=vt220
   ```

   If you have `/var` on a separate partition, you need to repeat that step for it. After that, you can edit `/etc/rc.conf` with `vi`(1). When you have finished, type **exit** at the prompt to leave the single-user shell and continue with the multi-user boot.

   Other values that need to be set in `/etc/rc.conf` for a networked environment are *hostname* and possibly *defaultroute*, furthermore add an *ifconfig_int* for your <int> network interface, along the lines of

   ```
   ifconfig_de0="inet 123.45.67.89 netmask 255.255.255.0"
   ```

   or, if you have *myname.my.dom* in `/etc/hosts`:

   ```
   ifconfig_de0="inet myname.my.dom netmask 255.255.255.0"
   ```

To enable proper hostname resolution, you will also want to add an `/etc/resolv.conf` file or (if you are feeling a little more adventurous) run `named`(8). See `resolv.conf`(5) or `named`(8) for more information.

Other files in `/etc` that may require modification or setting up include `/etc/mailer.conf`, `/etc/nsswitch.conf`, and `/etc/wscons.conf`.

2. Logging in

   After reboot, you can log in as `root` at the login prompt. Unless you've set a password in **`sysinst`**, there is no initial password. If you're using the machine in a networked environment, you should create an account for yourself (see below) and protect it and the "root" account with good passwords. Unless you have connected an unusual terminal device as the console you can just press `RETURN` when it prompts for `Terminal type? [...]`.

3. Adding accounts

   Use the `useradd`(8) command to add accounts to your system; *do not* edit `/etc/passwd` directly. See `useradd`(8) for more information on how to add a new user to the system.

4. The X Window System

   If you have installed the X Window System, look at the files in `/usr/X11R6/lib/X11/doc` for information.

   Don't forget to add `/usr/X11R6/bin` to your path in your shell's dot file so that you have access to the X binaries.

5. Installing third party packages

   If you wish to install any of the software freely available for UNIX-like systems you are strongly advised to first check the NetBSD package system. This automatically handles any changes necessary to make the software run on NetBSD, retrieval and installation of any other packages on which the software may depend, and simplifies installation (and deinstallation), both from source and precompiled binaries.

   - More information on the package system is at
     http://www.netbsd.org/Documentation/software/packages.html
   - A list of available packages suitable for browsing is at
     ftp://ftp.netbsd.org/pub/NetBSD/packages/pkgsrc/README.html
   - Precompiled binaries can be found at
     ftp://ftp.netbsd.org/pub/NetBSD/packages,
     usually in the `1.6/amiga/All` subdir. You can install them with the following commands:

     ```
     # PKG_PATH=ftp://ftp.netbsd.org/pub/NetBSD/packages/1.6/amiga/All
     # export PKG_PATH
     # pkg_add -v tcsh
     # pkg_add -v apache
     # pkg_add -v perl
     ...
     ```

     The above commands will install the tcsh shell, the Apache web server and the perl programming language as well as all the packages they depend on.

   - Package sources for compiling packages on your own can be obtained by retrieving the file
     ftp://ftp.netbsd.org/pub/NetBSD/NetBSD-current/tar_files/pkgsrc.tar.gz
     They are typically extracted into `/usr/pkgsrc` (though other locations work fine), with the commands:

```
# mkdir /usr/pkgsrc
# ( cd /usr/pkgsrc ; tar -zxpf - ) < pkgsrc.tar.gz
```

After extracting, then see the README file in the extraction directory (e.g. /usr/pkgsrc/README) for more information.

6. Misc

   - Edit /etc/mail/aliases to forward root mail to the right place. Don't forget to run newaliases(1) afterwards.
   - The /etc/mail/sendmail.cf file will almost definitely need to be adjusted; files aiding in this can be found in /usr/share/sendmail. See the README file there for more information.
   - Edit /etc/rc.local to run any local daemons you use.
   - Many of the /etc files are documented in section 5 of the manual; so just invoking

     ```
     # man 5 filename
     ```

     is likely to give you more information on these files.

Upgrading a previously-installed NetBSD System

The upgrade to NetBSD 1.6 is a binary upgrade; it can be quite difficult to advance to a later version by recompiling from source due primarily to interdependencies in the various components.

To do the upgrade, you must have the NetBSD kernel on AmigaDOS and you must transfer the miniroot file system miniroot.fs onto the swap partition of the NetBSD hard disk. You must also have at least the **base** binary distribution set available, so that you can upgrade with it, using one of the upgrade methods described above. Finally, you must have sufficient disk space available to install the new binaries. Since the old binaries are being overwritten in place, you only need space for the new binaries, which weren't previously on the system. If you have a few megabytes free on each of your / (root) and /usr partitions, you should have enough space.

Since upgrading involves replacing the kernel, and most of the system binaries, it has the potential to cause data loss. You are strongly advised to BACK UP ANY IMPORTANT DATA ON YOUR DISK, whether on the NetBSD partition or on another operating system's partition, before beginning the upgrade process.

To upgrade your system, follow the following instructions:

Transfer the miniroot file system onto the hard disk partition used by NetBSD for swapping, as described in the "Preparing your System for NetBSD Installation" section above.

Now boot up NetBSD, with boot blocks installed

Note: This description is for V40 (OS 3.1) ROMs. For older ROMs, there might be small differences. Check your AmigaDOS documentation to learn about the exact procedure. Booting using bootblocks doesn't work at all on some systems, and may require a mountable file system on others.

Reboot your machine, holding down both mouse buttons if you have a 2-button mouse, the outer mouse buttons if you have a 3-button mouse. On the DraCo, press the left mouse button instead, when the boot screen prompts you for it.

From the boot menu, select **Boot Options**. Select the swap partition with the miniroot, and then **ok**. Select **Boot** now. The machine will boot the bootblock, which will prompt your for a command line. You have a few seconds time to change the default. Entering an empty line will accept the default.

The bootblock uses command lines of the form:

file [*options*]

where *file* is the kernel file name on the partition where the boot block is on, and *options* may contain

the same as described in the INSTALL section.

For installing, use

netbsd -b

If you machine has a split memory space, like, e.g., DraCo machines, use this instead:

netbsd -bn2

Once your kernel boots

You should see the screen clear and some information about your system as the kernel configures the hardware. Note which hard disk device is configured that contains your root (/) and swap partitions. When prompted for the root device, type sd0b (replacing '0' with the disk number that NetBSD used for your root/swap device). When prompted for a dump device, answer '**none**' for the upgrade. (For a normal boot, you would tell it one of the swap devices). When prompted for the root file system type, confirm '**generic**', which will auto-detect it.

You will be presented with some information about the upgrade process and a warning message, and will be asked if you wish to proceed with the upgrade process. If you answer negatively, the upgrade process will stop, and your disk will not be modified. If you answer affirmatively, the upgrade process will begin, and your disk will be modified. You may press CONTROL-C to stop the upgrade process at any time. However, if you press it at an inopportune moment, your system may be left in an inconsistent (and possibly unusable) state.

You will now be greeted and reminded of the fact that this is a potential dangerous procedure and that you should not upgrade the **etc** set.

When you decide to proceed, you will be prompted to enter your root disk. After you've done this, it will be checked automatically to make sure that the file system is in a sane state before making any modifications. After this is done, you will be asked if you want to configure your network.

You are now allowed to edit your fstab. Normally you don't have to. Note that the upgrade-kit uses it's own copy of the fstab. Whatever you do here *won't* affect your actual fstab. After you are satisfied with your fstab, the upgrade-kit will check all file systems mentioned in it. When they're ok, they will be mounted.

You will now be asked if your sets are stored on a normally mounted file system. You should answer '**y**' to this question if you have the sets stored on a file system that was present in the fstab. The actions you should take for the set extraction are pretty logical (we think).

After you have extracted the sets, the upgrade kit will proceed with setting the timezone and installing the kernel and bootcode. This is all exactly the same as described in the installation section.

Your system has now been upgraded to NetBSD 1.6.

After a new kernel has been copied to your hard disk, your machine is a complete NetBSD 1.6 system. However, that doesn't mean that you're finished with the upgrade process. There are several things that you should do, or might have to do, to insure that the system works properly.

You will probably want to get the **etc** distribution, extract it, and compare its contents with those in your /etc directory. You will probably want to replace some of your system configuration files, or incorporate some of the changes in the new versions into yours.

You will want to delete old binaries that were part of the version of NetBSD that you upgraded from and have since been removed from the NetBSD distribution. If upgrading from a NetBSD version older than 1.0, you might also want to recompile any locally-built binaries, to take advantage of the shared libraries. (Note that any new binaries that you build will be dynamically linked, and therefore take advantage of the shared libraries, by default. For information on how to make statically linked binaries, see the cc(1) and ld(1) manual pages.)

Compatibility Issues With Previous NetBSD Releases

Users upgrading from previous versions of NetBSD may wish to bear the following problems and compatibility issues in mind when upgrading to NetBSD 1.6.

Issues affecting an upgrade from NetBSD 1.5

The following issues can generally be resolved by extracting the **etc** set into a temporary directory and running *postinstall*:

```
mkdir /tmp/upgrade
cd /tmp/upgrade
pax -zrpe -f /path/to/etc.tgz
./etc/postinstall -s `pwd` check
./etc/postinstall -s `pwd` fix
```

Issues fixed by *postinstall*:

- Various files in `/etc` need upgrading. These include:
  - `/etc/defaults/*`
  - `/etc/mtree/*`
  - `/etc/daily`
  - `/etc/weekly`
  - `/etc/monthly`
  - `/etc/security`
  - `/etc/rc.subr`
  - `/etc/rc`
  - `/etc/rc.shutdown`
  - `/etc/rc.d/*`
- The following files are now obsolete: `/etc/rc.d/NETWORK` and `/etc/rc.d/gated`.
- The following `rc.conf`(5) entries are now obsolete: `amd_master`, `ip6forwarding`, `defcorename`, and `nfsiod_flags`. `critical_filesystems_beforenet` has been replaced by `critical_filesystems_local`. `critical_filesystems` has been replaced by `critical_filesystems_remote`.
- The users and groups 'named', 'ntpd', and 'sshd' need to be created.
- The configuration files for `ssh`(1) and `sshd`(8) were moved from `/etc` to `/etc/ssh`, including `ssh_known_hosts*` files and the host key files `ssh_host*_key*`. `/etc/ssh.conf` was renamed to `/etc/ssh/ssh_config`, and `/etc/sshd.conf` was renamed to `/etc/ssh/sshd_config`.
- The **`mux`** entries in `wscons.conf`(5) are now obsolete.

The following issues need to be resolved manually:

- `postfix`(8) configuration files require upgrading.

  ```
  cd /usr/share/examples/postfix
  cp post-install postfix-files postfix-script /etc/postfix
  postfix check
  ```

- The *de* ethernet driver was replaced with the *tlp* driver. This may require the renaming of the files `/etc/ifconfig.de*` to `/etc/ifconfig.tlp*`, renaming of `rc.conf`(5) entries `ifconfig_de*` to `ifconfig_tlp*`, and the reconfiguration of files such as `/etc/dhclient.conf` and `/etc/ipf.conf`.

Issues affecting an upgrade from NetBSD 1.4 or prior

- `/etc/rc` modified to use `/etc/rc.d/*`

  Prior to NetBSD 1.5, `/etc/rc` was a traditional BSD style monolithic file; each discrete program or subsystem from `/etc/rc` and `/etc/netstart` has been moved into separate scripts in `/etc/rc.d/`.

  At system startup, `/etc/rc` uses `rcorder`(8) to build a dependency list of the files in `/etc/rc.d` and then executes each script in turn with an argument of 'start'. Many `rc.d` scripts won't start unless the appropriate `rc.conf`(5) entry in `/etc/rc.conf` is set to 'YES.'

  At system shutdown, `/etc/rc.shutdown` uses `rcorder`(8) to build a dependency list of the files in `/etc/rc.d` that have a "KEYWORD: shutdown" line, reverses the resulting list, and then executes each script in turn with an argument of 'stop'. The following scripts support a specific shutdown method: `cron`, `inetd`, `local`, and `xdm`.

  Local and third-party scripts may be installed into `/etc/rc.d` as necessary. Refer to the other scripts in that directory and `rc`(8) for more information on implementing `rc.d` scripts.

- `named`(8) leaks version information.

  Previous releases of NetBSD disabled a feature of `named`(8) where the version number of the server could be determined by remote clients. This feature has not been disabled in NetBSD 1.5, because there is a `named.conf`(5) option to change the version string:

```
option {
        version "newstring";
};
```

- `sysctl`(8) was moved from `/usr/sbin/sysctl` to `/sbin/sysctl`. If you have hardcoded references to the full pathname (in shell scripts, for example) please be sure to update those.

- `sendmail`(8) configuration file pathname changed.

  Due to `sendmail`(8) upgrade from 8.9.x to 8.10.x, `/etc/sendmail.cf` is moved to `/etc/mail/sendmail.cf`. Also, the default `sendmail.cf`(5) refers different pathnames than before. For example, `/etc/aliases` is now located at `/etc/mail/aliases`, `/etc/sendmail.cw` is now called `/etc/mail/local-host-names`, and so forth. If you have customized `sendmail.cf`(5) and friends, you will need to move the files to the new locations. See `/usr/share/sendmail/README` for more information.

Using online NetBSD documentation

Documentation is available if you first install the manual distribution set. Traditionally, the "man pages" (documentation) are denoted by '`name(section)`'. Some examples of this are

- `intro`(1),
- `man`(1),
- `apropos`(1),
- `passwd`(1), and
- `passwd`(5).

The section numbers group the topics into several categories, but three are of primary interest: user commands are in section 1, file formats are in section 5, and administrative information is in section 8.

The *man* command is used to view the documentation on a topic, and is started by entering **man** [*section*] *topic*. The brackets [ ] around the section should not be entered, but rather indicate that the section is optional. If you don't ask for a particular section, the topic with the lowest numbered section name will be displayed. For instance, after logging in, enter

```
# man passwd
```

to read the documentation for passwd(1). To view the documentation for passwd(5), enter

```
# man 5 passwd
```

instead.

If you are unsure of what man page you are looking for, enter **apropos** *subject-word*

where *subject-word* is your topic of interest; a list of possibly related man pages will be displayed.

Administrivia

If you've got something to say, do so! We'd like your input. There are various mailing lists available via the mailing list server at majordomo@netbsd.org. To get help on using the mailing list server, send mail to that address with an empty body, and it will reply with instructions.

There are various mailing lists set up to deal with comments and questions about this release. Please send comments to: netbsd-comments@netbsd.org.

To report bugs, use the send-pr(1) command shipped with NetBSD, and fill in as much information about the problem as you can. Good bug reports include lots of details. Additionally, bug reports can be sent by mail to: netbsd-bugs@netbsd.org.

Use of send-pr(1) is encouraged, however, because bugs reported with it are entered into the NetBSD bugs database, and thus can't slip through the cracks.

There are also port-specific mailing lists, to discuss aspects of each port of NetBSD. Use majordomo to find their addresses, or visit **http://www.netbsd.org/MailingLists/**. If you're interested in doing a serious amount of work on a specific port, you probably should contact the 'owner' of that port (listed below).

If you'd like to help with this effort, and have an idea as to how you could be useful, send us mail or subscribe to: netbsd-help@netbsd.org.

As a favor, please avoid mailing huge documents or files to these mailing lists. Instead, put the material you would have sent up for FTP or WWW somewhere, then mail the appropriate list about it, or, if you'd rather not do that, mail the list saying you'll send the data to those who want it.

Thanks go to

- The former members of UCB's Computer Systems Research Group, including (but not limited to):

  Keith Bostic
  Ralph Campbell
  Mike Karels
  Marshall Kirk McKusick

  for their ongoing work on BSD systems, support, and encouragement.

- Also, our thanks go to:

  Mike Hibler
  Rick Macklem
  Jan-Simon Pendry
  Chris Torek

  for answering lots of questions, fixing bugs, and doing the various work they've done.

- UC Berkeley's Experimental Computing Facility provided a home for sun-lamp in the past, people to look after it, and a sense of humor. Rob Robertson, too, has added his unique sense of humor to things, and for a long time provided the primary FTP site for NetBSD.
- Vixie Enterprises for hosting the NetBSD FTP, SUP, and WWW servers.
- Redback Networks, Inc. for hosting the NetBSD mail and GNATS server.
- The Helsinki University of Technology in Finland for hosting the NetBSD CVS server.
- The Internet Research Institute in Japan for hosting the server which runs the CVSweb interface to the NetBSD source tree.
- The many organisations that provide NetBSD mirror sites.
- Without CVS, this project would be impossible to manage, so our hats go off to Brian Berliner, Jeff Polk, and the various other people who've had a hand in making CVS a useful tool.
- Dave Burgess `burgess@cynjut.infonet.net` has been maintaining the 386BSD/NetBSD/FreeBSD FAQ for quite some time, and deserves to be recognized for it.
- The following individuals and organizations (each in alphabetical order) have made donations or loans of hardware and/or money, to support NetBSD development, and deserve credit for it:

  Steve Allen
  Jason Birnschein
  Mason Loring Bliss
  Jason Brazile
  Mark Brinicombe
  David Brownlee
  Simon Burge
  Dave Burgess
  Ralph Campbell
  Brian Carlstrom
  James Chacon
  Bill Coldwell
  Charles Conn
  Tom Coulter
  Charles D. Cranor
  Christopher G. Demetriou
  Scott Ellis
  Hubert Feyrer
  Castor Fu
  Greg Gingerich
  William Gnadt
  Michael Graff
  Guenther Grau
  Ross Harvey
  Charles M. Hannum
  Michael L. Hitch
  Kenneth Alan Hornstein
  Jordan K. Hubbard
  Søren Jørvang
  Scott Kaplan
  Noah M. Keiserman
  Harald Koerfgen

John Kohl
Chris Legrow
Ted Lemon
Norman R. McBride
Neil J. McRae
Perry E. Metzger
Luke Mewburn
Toru Nishimura
Herb Peyerl
Mike Price
Dave Rand
Michael Richardson
Heiko W. Rupp
Brad Salai
Chuck Silvers
Thor Lancelot Simon
Bill Sommerfeld
Paul Southworth
Eric and Rosemary Spahr
Ted Spradley
Kimmo Suominen
Jason R. Thorpe
Steve Wadlow
Krister Walfridsson
Rob Windsor
Jim Wise
Reinoud Zandijk
Christos Zoulas

AboveNet Communications, Inc.
Advanced System Products, Inc.
Avalon Computer Systems
Bay Area Internet Solutions
Brains Corporation, Japan
Canada Connect Corporation
Co-operative Research Centre for Enterprise Distributed Systems Technology
Demon Internet, UK
Digital Equipment Corporation
Distributed Processing Technology
Easynet, UK
Free Hardware Foundation
Innovation Development Enterprises of America
Internet Software Consortium
MS Macro System GmbH, Germany
Numerical Aerospace Simulation Facility, NASA Ames Research Center
Piermont Information Systems Inc.
Precedence Technologies Ltd
Salient Systems Inc.
VMC Harald Frank, Germany
Warped Communications, Inc.
Whitecross Database Systems Ltd.

(If you're not on that list and should be, tell us! We probably were not able to get in touch with you, to

verify that you wanted to be listed.)

- Finally, we thank all of the people who've put sweat and tears into developing NetBSD since its inception in January, 1993. (Obviously, there are a lot more people who deserve thanks here. If you're one of them, and would like to mentioned, tell us!)

We are...

(in alphabetical order)

The NetBSD core group:

Jun-ichiro itojun Hagino	itojun@netbsd.org
Frank van der Linden	fvdl@netbsd.org
Luke Mewburn	lukem@netbsd.org
Christos Zoulas	christos@netbsd.org

The portmasters (and their ports):

Simon Burge	simonb@netbsd.org	**pmax**
Simon Burge	simonb@netbsd.org	**sbmips**
Jeremy Cooper	jeremy@netbsd.org	**sun3x**
Matt Fredette	fredette@netbsd.org	**sun2**
Chris Gilbert	chris@netbsd.org	**cats**
Ross Harvey	ross@netbsd.org	**alpha**
Jun-ichiro itojun Hagino	itojun@netbsd.org	**sh3**
Ben Harris	bjh21@netbsd.org	**acorn26**
Eduardo Horvath	eeh@netbsd.org	**sparc64**
Darrin Jewell	dbj@netbsd.org	**next68k**
Søren Jørvang	soren@netbsd.org	**cobalt**
Søren Jørvang	soren@netbsd.org	**sgimips**
Wayne Knowles	wdk@netbsd.org	**mipsco**
Paul Kranenburg	pk@netbsd.org	**sparc**
Frank van der Linden	fvdl@netbsd.org	**i386**
Anders Magnusson	ragge@netbsd.org	**vax**
Phil Nelson	phil@netbsd.org	**pc532**
NISHIMURA Takeshi	nsmrtks@netbsd.org	**x68k**
Tohru Nishimura	nisimura@netbsd.org	**luna68k**
NONAKA Kimihiro	nonaka@netbsd.org	**prep**
Scott Reynolds	scottr@netbsd.org	**mac68k**
Kazuki Sakamoto	sakamoto@netbsd.org	**bebox**
Noriyuki Soda	soda@netbsd.org	**arc**
Wolfgang Solfrank	ws@netbsd.org	**ofppc**
Ignatios Souvatzis	is@netbsd.org	**amiga**
Jonathan Stone	jonathan@netbsd.org	**pmax**
Shin Takemura	takemura@netbsd.org	**hpcmips**
Jason Thorpe	thorpej@netbsd.org	**alpha**
Jason Thorpe	thorpej@netbsd.org	**hp300**
Tsubai Masanari	tsubai@netbsd.org	**macppc**
Tsubai Masanari	tsubai@netbsd.org	**newsmips**
Izumi Tsutsui	tsutsui@netbsd.org	**news68k**

Leo Weppelman	leo@netbsd.org	**atari**
Nathan Williams	nathanw@netbsd.org	**sun3**
Steve Woodford	scw@netbsd.org	**mvme68k**
Steve Woodford	scw@netbsd.org	**mvmeppc**
Reinoud Zandijk	reinoud@netbsd.org	**acorn32**

The NetBSD 1.6 Release Engineering team:

Erik Berls	cyber@netbsd.org
Håvard Eidnes	he@netbsd.org
Perry Metzger	perry@netbsd.org
Luke Mewburn	lukem@netbsd.org
Jason Thorpe	thorpej@netbsd.org
Todd Vierling	tv@netbsd.org

NetBSD Developers:

Nathan Ahlstrom	nra@netbsd.org
Steve Allen	wormey@netbsd.org
Julian Assange	proff@netbsd.org
Lennart Augustsson	augustss@netbsd.org
Christoph Badura	bad@netbsd.org
Bang Jun-Young	junyoung@netbsd.org
Dieter Baron	dillo@netbsd.org
Robert V. Baron	rvb@netbsd.org
Jason Beegan	jtb@netbsd.org
Erik Berls	cyber@netbsd.org
Hiroyuki Bessho	bsh@netbsd.org
John Birrell	jb@netbsd.org
Mason Loring Bliss	mason@netbsd.org
Rafal Boni	rafal@netbsd.org
Manuel Bouyer	bouyer@netbsd.org
John Brezak	brezak@netbsd.org
Allen Briggs	briggs@netbsd.org
Mark Brinicombe	mark@netbsd.org
Aaron Brown	abrown@netbsd.org
Andrew Brown	atatat@netbsd.org
David Brownlee	abs@netbsd.org
Frederick Bruckman	fredb@netbsd.org
Jon Buller	jonb@netbsd.org
Simon Burge	simonb@netbsd.org
Robert Byrnes	byrnes@netbsd.org
D'Arcy J.M. Cain	darcy@netbsd.org
Dave Carrel	carrel@netbsd.org
James Chacon	jmc@netbsd.org
Bill Coldwell	billc@netbsd.org
Julian Coleman	jdc@netbsd.org
Jeremy Cooper	jeremy@netbsd.org
Chuck Cranor	chuck@netbsd.org

Alistair Crooks	agc@netbsd.org
Aidan Cully	aidan@netbsd.org
Johan Danielsson	joda@netbsd.org
Matt DeBergalis	deberg@netbsd.org
Rob Deker	deker@netbsd.org
Chris G. Demetriou	cgd@netbsd.org
Tracy Di Marco White	gendalia@netbsd.org
Jaromír Dolecek	jdolecek@netbsd.org
Andy Doran	ad@netbsd.org
Roland Dowdeswell	elric@netbsd.org
Emmanuel Dreyfus	manu@netbsd.org
Matthias Drochner	drochner@netbsd.org
Jun Ebihara	jun@netbsd.org
Håvard Eidnes	he@netbsd.org
Stoned Elipot	seb@netbsd.org
Enami Tsugutomo	enami@netbsd.org
Bernd Ernesti	veego@netbsd.org
Erik Fair	fair@netbsd.org
Gavan Fantom	gavan@netbsd.org
Hubert Feyrer	hubertf@netbsd.org
Jason R. Fink	jrf@netbsd.org
Matt Fredette	fredette@netbsd.org
Thorsten Frueauf	frueauf@netbsd.org
Castor Fu	castor@netbsd.org
Ichiro Fukuhara	ichiro@netbsd.org
Thomas Gerner	thomas@netbsd.org
Simon J. Gerraty	sjg@netbsd.org
Justin Gibbs	gibbs@netbsd.org
Chris Gilbert	chris@netbsd.org
Adam Glass	glass@netbsd.org
Michael Graff	explorer@netbsd.org
Brian C. Grayson	bgrayson@netbsd.org
Matthew Green	mrg@netbsd.org
Andreas Gustafsson	gson@netbsd.org
Jun-ichiro itojun Hagino	itojun@netbsd.org
Juergen Hannken-Illjes	hannken@netbsd.org
Charles M. Hannum	mycroft@netbsd.org
Ben Harris	bjh21@netbsd.org
Ross Harvey	ross@netbsd.org
Eric Haszlakiewicz	erh@netbsd.org
John Hawkinson	jhawk@netbsd.org
HAYAKAWA Koichi	haya@netbsd.org
René Hexel	rh@netbsd.org
Michael L. Hitch	mhitch@netbsd.org
Christian E. Hopps	chopps@netbsd.org
Ken Hornstein	kenh@netbsd.org
Marc Horowitz	marc@netbsd.org
Eduardo Horvath	eeh@netbsd.org
Nick Hudson	skrll@netbsd.org

Shell Hung	shell@netbsd.org
Martin Husemann	martin@netbsd.org
Dean Huxley	dean@netbsd.org
Bernardo Innocenti	bernie@netbsd.org
Tetsuya Isaki	isaki@netbsd.org
ITOH Yasufumi	itohy@netbsd.org
IWAMOTO Toshihiro	toshii@netbsd.org
Matthew Jacob	mjacob@netbsd.org
Lonhyn T. Jasinskyj	lonhyn@netbsd.org
Darrin Jewell	dbj@netbsd.org
Chris Jones	cjones@netbsd.org
Søren Jørvang	soren@netbsd.org
Takahiro Kambe	taca@netbsd.org
Antti Kantee	pooka@netbsd.org
Masanori Kanaoka	kanaoka@netbsd.org
Mattias Karlsson	keihan@netbsd.org
KAWAMOTO Yosihisa	kawamoto@netbsd.org
Mario Kemper	magick@netbsd.org
Thomas Klausner	wiz@netbsd.org
Klaus Klein	kleink@netbsd.org
Wayne Knowles	wdk@netbsd.org
John Kohl	jtk@netbsd.org
Paul Kranenburg	pk@netbsd.org
Martti Kuparinen	martti@netbsd.org
Kevin Lahey	kml@netbsd.org
Johnny C. Lam	jlam@netbsd.org
Martin J. Laubach	mjl@netbsd.org
Ted Lemon	mellon@netbsd.org
Frank van der Linden	fvdl@netbsd.org
Joel Lindholm	joel@netbsd.org
Mike Long	mikel@netbsd.org
Warner Losh	imp@netbsd.org
Tomasz Luchowski	zuntum@netbsd.org
Federico Lupi	federico@netbsd.org
Brett Lymn	blymn@netbsd.org
Paul Mackerras	paulus@netbsd.org
Anders Magnusson	ragge@netbsd.org
MAEKAWA Masahide	gehenna@netbsd.org
David Maxwell	david@netbsd.org
Dan McMahill	dmcmahill@netbsd.org
Gregory McGarry	gmcgarry@netbsd.org
Jared D. McNeill	jmcneill@netbsd.org
Neil J. McRae	neil@netbsd.org
Perry Metzger	perry@netbsd.org
Minoura Makoto	minoura@netbsd.org
Luke Mewburn	lukem@netbsd.org
der Mouse	mouse@netbsd.org
Joseph Myers	jsm@netbsd.org
Ken Nakata	kenn@netbsd.org

Phil Nelson	phil@netbsd.org
Bob Nestor	rnestor@netbsd.org
NISHIMURA Takeshi	nsmrtks@netbsd.org
Tohru Nishimura	nisimura@netbsd.org
NONAKA Kimihiro	nonaka@netbsd.org
Jesse Off	joff@netbsd.org
Tatoku Ogaito	tacha@netbsd.org
Masaru Oki	oki@netbsd.org
Atsushi Onoe	onoe@netbsd.org
Greg Oster	oster@netbsd.org
Herb Peyerl	hpeyerl@netbsd.org
Matthias Pfaller	matthias@netbsd.org
Chris Pinnock	cjep@netbsd.org
Dante Profeta	dante@netbsd.org
Chris Provenzano	proven@netbsd.org
Michael Rauch	mrauch@netbsd.org
Darren Reed	darrenr@netbsd.org
Scott Reynolds	scottr@netbsd.org
Michael Richardson	mcr@netbsd.org
Tim Rightnour	garbled@netbsd.org
Gordon Ross	gwr@netbsd.org
Heiko W. Rupp	hwr@netbsd.org
David Sainty	dsainty@netbsd.org
SAITOH Masanobu	msaitoh@netbsd.org
Kazuki Sakamoto	sakamoto@netbsd.org
Curt Sampson	cjs@netbsd.org
Wilfredo Sanchez	wsanchez@netbsd.org
Ty Sarna	tsarna@netbsd.org
SATO Kazumi	sato@netbsd.org
Jan Schaumann	jschauma@netbsd.org
Matthias Scheler	tron@netbsd.org
Karl Schilke (rAT)	rat@netbsd.org
Amitai Schlair	schmonz@netbsd.org
Konrad Schroder	perseant@netbsd.org
Reed Shadgett	dent@netbsd.org
Tim Shepard	shep@netbsd.org
Takeshi Shibagaki	shiba@netbsd.org
Takao Shinohara	shin@netbsd.org
Takuya SHIOZAKI	tshiozak@netbsd.org
Chuck Silvers	chs@netbsd.org
Thor Lancelot Simon	tls@netbsd.org
Jeff Smith	jeffs@netbsd.org
Noriyuki Soda	soda@netbsd.org
Wolfgang Solfrank	ws@netbsd.org
SOMEYA Yoshihiko	someya@netbsd.org
Bill Sommerfeld	sommerfeld@netbsd.org
Ignatios Souvatzis	is@netbsd.org
Bill Squier	groo@netbsd.org
Jonathan Stone	jonathan@netbsd.org

Bill Studenmund	wrstuden@netbsd.org
Kevin Sullivan	sullivan@netbsd.org
SUNAGAWA Keiki	kei@netbsd.org
Kimmo Suominen	kim@netbsd.org
Shin Takemura	takemura@netbsd.org
TAMURA Kent	kent@netbsd.org
Shin'ichiro TAYA	taya@netbsd.org
Matt Thomas	matt@netbsd.org
Jason Thorpe	thorpej@netbsd.org
Christoph Toshok	toshok@netbsd.org
Tsubai Masanari	tsubai@netbsd.org
Izumi Tsutsui	tsutsui@netbsd.org
UCHIYAMA Yasushi	uch@netbsd.org
Masao Uebayashi	uebayasi@netbsd.org
Shuichiro URATA	ur@netbsd.org
Todd Vierling	tv@netbsd.org
Aymeric Vincent	aymeric@netbsd.org
Paul Vixie	vixie@netbsd.org
Krister Walfridsson	kristerw@netbsd.org
Lex Wennmacher	wennmach@netbsd.org
Leo Weppelman	leo@netbsd.org
Assar Westerlund	assar@netbsd.org
Todd Whitesel	toddpw@netbsd.org
Nathan Williams	nathanw@netbsd.org
Rob Windsor	windsor@netbsd.org
Dan Winship	danw@netbsd.org
Jim Wise	jwise@netbsd.org
Michael Wolfson	mbw@netbsd.org
Steve Woodford	scw@netbsd.org
Colin Wood	ender@netbsd.org
YAMAMOTO Takashi	yamt@netbsd.org
Yuji Yamano	yyamano@netbsd.org
Reinoud Zandijk	reinoud@netbsd.org
Maria Zevenhoven	maria7@netbsd.org
Christos Zoulas	christos@netbsd.org

Other contributors:

Dave Burgess	burgess@cynjut.infonet.net
Brian R. Gaeke	brg@dgate.org
Brad Grantham	grantham@tenon.com
Lawrence Kesteloot	kesteloo@cs.unc.edu
Waldi Ravens	waldi@moacs.indiv.nl.net

Legal Mumbo-Jumbo

All product names mentioned herein are trademarks or registered trademarks of their respective owners.

The following notices are required to satisfy the license terms of the software that we have mentioned in this document:

This product includes software developed by the University of California, Berkeley and its contributors.
This product includes software developed by The NetBSD Foundation, Inc.
This product includes software developed by the NetBSD Foundation, Inc. and its contributors.
This product includes software developed by the Computer Systems Engineering Group at Lawrence Berkeley Laboratory.
This product includes software developed by Adam Glass and Charles Hannum.
This product includes software developed by Adam Glass and Charles M. Hannum.
This product includes software developed by Adam Glass.
This product includes software developed by Alistair G. Crooks.
This product includes software developed by Amancio Hasty and Roger Hardiman.
This product includes software developed by Berkeley Software Design, Inc.
This product includes software developed by Bill Paul.
This product includes software developed by Charles D. Cranor and Washington University.
This product includes software developed by Charles D. Cranor.
This product includes software developed by Charles Hannum, by the University of Vermont and State Agricultural College and Garrett A. Wollman, by William F. Jolitz, and by the University of California, Berkeley, Lawrence Berkeley Laboratory, and its contributors.
This product includes software developed by Charles Hannum.
This product includes software developed by Charles M. Hannum.
This product includes software developed by Chris Provenzano.
This product includes software developed by Christian E. Hopps.
This product includes software developed by Christopher G. Demetriou for the NetBSD Project.
This product includes software developed by Christopher G. Demetriou.
This product includes software developed by Christos Zoulas.
This product includes software developed by David Jones and Gordon Ross.
This product includes software developed by Dean Huxley.
This product includes software developed by Eric S. Hvozda.
This product includes software developed by Ezra Story.
This product includes software developed by Gardner Buchanan.
This product includes software developed by Gordon Ross.
This product includes software developed by Gordon W. Ross and Leo Weppelman.
This product includes software developed by Gordon W. Ross.
This product includes software developed by Hauke Fath.
This product includes software developed by HAYAKAWA Koichi.
This product includes software developed by Hellmuth Michaelis and Joerg Wunsch.
This product includes software developed by Herb Peyerl.
This product includes software developed by Holger Veit and Brian Moore for use with "386BSD" and similar operating systems.
This product includes software developed by Hubert Feyrer for the NetBSD Project.
This product includes software developed by Iain Hibbert.
This product includes software developed by Ian W. Dall.
This product includes software developed by Ignatios Souvatzis for the NetBSD Project.
This product includes software developed by Jason R. Thorpe for And Communications, http://www.and.com/.
This product includes software developed by Joachim Koenig-Baltes.
This product includes software developed by Jochen Pohl for The NetBSD Project.
This product includes software developed by John Polstra.
This product includes software developed by Jonathan Stone and Jason R. Thorpe for the NetBSD Project.
This product includes software developed by Jonathan Stone for the NetBSD Project.
This product includes software developed by Jonathan Stone.
This product includes software developed by Julian Highfield.
This product includes software developed by Kenneth Stailey.
This product includes software developed by Leo Weppelman.
This product includes software developed by Lloyd Parkes.
This product includes software developed by Manuel Bouyer.
This product includes software developed by Marc Horowitz.
This product includes software developed by Mark Brinicombe for the NetBSD Project.
This product includes software developed by Mark Brinicombe.
This product includes software developed by Mark Tinguely and Jim Lowe.
This product includes software developed by Markus Wild.
This product includes software developed by Martin Husemann and Wolfgang Solfrank.
This product includes software developed by Mats O Jansson and Charles D. Cranor.
This product includes software developed by Mats O Jansson.
This product includes software developed by Matthias Pfaller.
This product includes software developed by Michael L. Hitch.
This product includes software developed by Niels Provos.
This product includes software developed by Paul Kranenburg.
This product includes software developed by Paul Mackerras.
This product includes software developed by Peter Galbavy.
This product includes software developed by Philip A. Nelson.

This product includes software developed by Rodney W. Grimes.
This product includes software developed by Roland C. Dowdeswell.
This product includes software developed by Rolf Grossmann.
This product includes software developed by Scott Bartram.
This product includes software developed by SigmaSoft, Th. Lockert.
This product includes software developed by Tatoku Ogaito for the NetBSD Project.
This product includes software developed by Terrence R. Lambert.
This product includes software developed by Theo de Raadt and John Brezak.
This product includes software developed by Theo de Raadt.
This product includes software developed by Tohru Nishimura for the NetBSD Project.
This product includes software developed by TooLs GmbH.
This product includes software designed by William Allen Simpson.
This product includes software developed by Winning Strategies, Inc.
This product includes software developed by Zembu Labs, Inc.
This product includes software developed by the Center for Software Science at the University of Utah.
This product includes software developed by the Computer Systems Laboratory at the University of Utah.
This product includes software developed by the University of Calgary Department of Computer Science and its contributors.
This product includes software developed by the University of Vermont and State Agricultural College and Garrett A. Wollman.
This product includes software developed for the FreeBSD project.
This product includes software developed for the Internet Software Consortium by Ted Lemon.
This product includes software developed for the NetBSD Project by Frank van der Linden.
This product includes software developed for the NetBSD Project by Jason R. Thorpe.
This product includes software developed for the NetBSD Project by John M. Vinopal.
This product includes software developed for the NetBSD Project by Matthias Drochner.
This product includes software developed for the NetBSD Project by Matthieu Herrb.
This product includes software developed for the NetBSD Project by Perry E. Metzger.
This product includes software developed for the NetBSD Project by Piermont Information Systems Inc.
This product includes software developed for the NetBSD Project by Ted Lemon.
This product includes software developed for the NetBSD Project by Wasabi Systems, Inc.
This product includes software developed by LAN Media Corporation and its contributors.
This product includes software developed by Michael Graff for the NetBSD Project.
This product includes software developed by Niklas Hallqvist, C Stone and Job de Haas.
This product includes software developed by Eric Young (eay@mincom.oz.au).
This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/).
This product includes software developed by the University of Oregon.
This product includes software developed by the University of Southern California and/or Information Sciences Institute.
This product includes software developed by Internet Initiative Japan Inc.
This product includes software developed by Reinoud Zandijk.
This product includes software developed at the Information Technology Division, US Naval Research Laboratory.

This product includes software developed by Tobias Abt.
This product includes software developed by Klaus Burkert.
This product includes software developed by Michael van Elst.
This product includes software developed by Bernd Ernesti.
This product includes software developed by Markus Illenseer.
This product includes software developed by Mika Kortelainen.
This product includes software developed by Jukka Marin.
This product includes software developed by Kari Mettinen.
This product includes software developed by Brad Pepers.
This product includes software developed by Ignatios Souvatzis.
This product includes software developed by Michael Teske.
This product includes software developed by Lutz Vieweg.
This product includes software developed by Daniel Widenfalk.

The End