NAME

INSTALL – Installation procedure for NetBSD/arm32.

DESCRIPTION

About this Document

This document describes the installation procedure for NetBSD 1.5 on the *arm32* 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 thirty-one different system architectures featuring twelve distinct families of CPUs, and is being ported to more. The NetBSD 1.5 release contains complete binary releases for fifteen different machine types. (The sixteen 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.

Changes Since The Last Release

The NetBSD 1.5 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 userland 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 one year of development that went into the NetBSD 1.5 release. Some highlights include:

Kernel

- Ports to new platforms including: arc, cobalt, hpcmips, news68k, sgimips, and sparc64.
- Improved performance and stability of the UVM virtual memory subsystem.
- Implementation of generic kernel locking code, as well as a restructure and re-tuning of the scheduler, to be used by the future symmetric multi-processing (SMP) implementation.
- Improved compatibility support for Linux, OSF1, and SVR4 programs.

- New compatibility support for Win32 programs.
- Support for dynamically loaded ELF kernel modules.
- Kernel process tracing using `ktruss`(1).
- Deletion of swap devices using `swapctl`(8).
- Easier hot-pluggability of keyboards and mice using a new wscons device - wsmux.
- Improved PCMCIA and Cardbus support, including support for detaching of devices and cards, resulting in better support for notebooks and PDA devices.
- Numerous hardware improvements, including areas such as: audio, UDMA/66 support for ATA drives, USB, and wireless networking.

Networking

- Addition of IP version 6 (IPv6) and IPsec to the networking stack, from the KAME project. This includes addition of kernel code for IPv6/IPsec, IPv4/v6 dual-stack user applications and supporting libraries. Due to this, the shlib major version for `pcap`(3) is incremented and you may need to recompile userland tools. The KAME IPv6 part includes results from the unified-ipv6 effort.

File system

- Significant Fast file system (FFS) performance enhancements via integration of Kirk McKusick's soft updates and trickle sync code.
- Support for the Windows NT 'NTFS' file system (read-only at this stage).
- Support for revision 1 of the Linux 'ext2fs' file system.
- Enhanced stability and usability of LFS (the BSD log-structured file system).
- Various RAIDframe enhancements including: auto-detection of RAID components and auto-configuration of RAID sets, and the ability to configure the root file system (`/`) on a RAID set.
- Support for Microsoft Joliet extensions to the ISO9660 CD file system.
- Improved file system vnode locking mechanisms, thus resolving a source of several panics in the past.
- Support for NFS and RPC over IPv6.
- Server part of NFS locking (implemented by `rpc.lockd`(8)) now works.

Security

- Strong cryptographic libraries and applications integrated, including the AES cipher Rijndael, the OpenSSL library, more complete Kerberos IV and Kerberos V support, and an SSH server and client.
- `sysctl`(3) interfaces to various elements of process and system information, allowing programs such as `ps`(1), `dmesg`(1) and the like to operate without recompilation after kernel upgrades, and remove the necessity to run setgid kmem (thus improving system security).
- Disable various services by default, and set the default options for disabled daemons to a higher level of logging.
- Several code audits were performed. One audit replaced string routines that were used without bound checking, and another one to identify and disable places where format strings were used in an unsafe way, allowing arbitrary data entered by (possibly) malicious users to overwrite application code, and leading from Denial of Service attacks to compromised system.

System administration and user tools

- Conversion of the `rc`(8) system startup and shutdown scripts to an 'rc.d' mechanism, with separate control scripts for each service, and appropriate dependency ordering provided by `rcorder`(8).
- `postfix`(1) provided as alternative mail transport agent to `sendmail`(8).
- User management tools `useradd`(8), `usermod`(8), `userdel`(8), `groupadd`(8), `groupmod`(8), and `groupdel`(8) added to the system.
- Incorporation of a login class capability database (`/etc/login.conf`) from BSD/OS.
- Improved support for usernames longer than eight characters in programs such as `at`(1) and `w`(1).
- Many enhancements to `ftpd`(8) providing features found in larger and less secure FTP daemons, such as user classes, connection limits, improved support for virtual hosting, transfer statistics, transfer rate throttling, and support for various IETF ftpext working group extensions.
- The `ftp`(1) client has been improved even further, including transfer rate throttling, improved URL support, command line uploads. See the man page for details.

Miscellaneous

- Updates to the NetBSD source code style code (located in `/usr/share/misc/style`) to use ANSI C only (instead of K&R) and reflect current (best) practice, and begin migrating the NetBSD source code to follow it.
- Implementation of many SUSv2 features to the `curses`(3) library, including support for color.
- Updates of most third party packages that are shipped in the base system, including `file`(1), `ipfilter`(4), `ppp`(4), and `sendmail`(8) to the latest stable release.
- Many new packages in the *pkgsrc* system, including standard desktops like KDE and GNOME as well as latest Tcl/Tk and perl and many of the components of the Java Enterprise platform. The package framework itself now has full wildcard dependency support.

As has been noted, there have also been innumerable bug fixes.

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.

NetBSD 1.5 on arm32 is, as usual, also fully backward compatible with old NetBSD arm32 binaries, so you don't need to recompile all your local programs provided you set the appropriate binary compatibility options in your kernel configuration.

New port-specific features include:

- Addition of Xarm32VIDC Xserver for RiscPC and ARM7500 based systems.

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 1.5 Release Contents

The root directory of the NetBSD 1.5 release is organized as follows:

`.../NetBSD-1.5/`

`CHANGES` Changes since earlier NetBSD releases.

`LAST_MINUTE` Last minute changes.

`MIRRORS` A list of sites that mirror the NetBSD 1.5 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 1.5 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.
22.3 MB gzipped, 98.8 MB uncompressed

pkgsrc This set contains the ''pkgsrc'' sources, which contain the infrastructure to build third-party packages.
5.6 MB gzipped, 57.0 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.
3.3 MB gzipped, 13.2 MB uncompressed

src This set contains all of the base NetBSD 1.5 sources which are not in **gnusrc**, **sharesrc**, or **syssrc**.
24.2 MB gzipped, 120.6 MB uncompressed

syssrc This set contains the sources to the NetBSD 1.5 kernel for all architectures, `config`(8), and `dbsym`(8).
17.6 MB gzipped, 88.6 MB uncompressed

xsrc This set contains the sources to the X Window System.
35.2 MB gzipped, 176.8 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. They 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/arm32 subdirectory structure

The arm32-specific portion of the NetBSD 1.5 release is found in the `arm32` subdirectory of the distribution:

`.../NetBSD-1.5/arm32/`

`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.
`sets/` arm32 binary distribution sets; see below.

`installation/`
`floppy/` arm32 boot and installation floppies; see below.
`misc/` Miscellaneous arm32 installation utilities; see installation section, below.

There are a collection of arm32 kernels in the `arm32/kernels` subdirectory of the NetBSD 1.5 distribution. Some of these kernels contain a root file system image and should only be used for the initial installation. Some of the kernels only support a particular subset of the platforms that arm32 supports (See `arm32/kernels/README` for more details.)

Binary distribution sets

The NetBSD arm32 binary distribution sets contain the binaries which comprise the NetBSD 1.5 release for the arm32. There are eight binary distribution sets. The binary distribution sets can be found in the `arm32/binary/sets` subdirectory of the NetBSD 1.5 distribution tree, and are as follows:

base The NetBSD 1.5 arm32 **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.
19.2 MB gzipped, 47.4 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.
13.8 MB gzipped, 45.4 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. (If you are upgrading, it's recommended that you get a copy of this set and *carefully* upgrade your configuration files by hand.)
0.1 MB gzipped, 0.6 MB uncompressed

games This set includes the games and their manual pages.
3.0 MB gzipped, 7.5 MB uncompressed

kern This set contains a NetBSD/arm32 1.5 `GENERIC` kernel, named `/netbsd`. You *must* install this distribution set.
1.1 MB gzipped, 2.3 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.
5.0 MB gzipped, 19.9 MB uncompressed

misc This set includes the (rather large) system dictionaries, the typesettable document set, and other files from `/usr/share`.
2.6 MB gzipped, 10.1 MB uncompressed

text This set includes NetBSD 's text processing tools, including `groff`(1), all related programs, and their manual pages.
1.4 MB gzipped, 4.8 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.
3.2 MB gzipped, 9.5 MB uncompressed

xcomp The extra libraries and include files needed to compile X source code.
2.1 MB gzipped, 8.2 MB uncompressed

xcontrib Programs that were contributed to X.
0.2 MB gzipped, 0.7 MB uncompressed

xfont Fonts needed by X.
6.2 MB gzipped, 7.5 MB uncompressed

xserver
6.8 MB gzipped, 17.4 MB uncompressed

The arm32 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 files are `/`-relative and therefore are extracted *below* the current directory. That is, 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 `/`.

Note: Each directory in the arm32 binary distribution also has its own checksum files, just as the source distribution does.

NetBSD/arm32 System Requirements and Supported Devices

NetBSD/arm32 1.5 runs on a number of systems with ARM6 or later processors, with or without FPU coprocessor. The minimal configuration is said to require 8 MB of RAM and 50 MB of disk space, though we do not know of anyone running with a system quite this minimal today. To install the entire system requires much more disk space (the unpacked binary distribution, without sources, requires at least 65 MB without counting space needed for swap space, etc), and to run X or compile the system, more RAM is recommended. (8 MB of RAM will actually allow you to run X and/or compile, but it won't be speedy. Note that until you have around 16 MB of RAM, getting more RAM is more important than getting a faster CPU.)

Supported devices

- RiscPC/A7000(+) floppy controller
- IDE controllers
  - Acorn motherboard IDE
  - Simtec IDE controller
  - RapIDE Issue 2 IDE controller
  - ICS V5 & V6 IDE controller
- SCSI host adapters
  - Cumana SCSI 2
  - PowerTec SCSI 2
  - MCS Connect32 SCSI 2
  - Acorn SCSI
  - Oak SCSI I
  - Morley SCSI I (uncached only)
- VIDC20 video
- RiscPC Motherboard serial port
- RiscPC Motherboard parallel port
- Ethernet adapters
  - Acorn Ether1
  - Atomwide Ether3
  - ANT Ether3
  - ANT Ether5
  - Atomwide EtherA
  - ANT EtherB
  - Acorn EtherH
  - I-cubed EtherH
  - ANT EtherM
- Most SCSI disk drives
- Most SCSI tape drives
- CD-ROM drives
  - Most SCSI CD-ROM drives
  - Most ATAPI CD-ROM drives

    Note: Some low-priced IDE CD-ROM drives are known for being not or not fully ATAPI compliant, and thus requires some hack (generally an entry to a quirk table) to work with NetBSD.
- Mice
  - RiscPC quadrature mouse
  - A7000 PS/2 mouse
- Processors
  - ARM 610
  - ARM 700
  - ARM 700 + FPA11

- ARM 710
- ARM 7500
- ARM 7500FE
- ARM 810. [*]
- SA110

- Motherboards
  - Acorn RiscPC
  - Acorn A7000
  - Acorn A7000+
  - VLSI RC7500
  - Digital DNARD
  - Intel EBSA285
  - Chalice CATS

- Other devices
  - RiscPC keyboard
  - A7000 keyboard
  - RiscPC realtime clock
  - VLSI RC7500 motherboard devices

Supported Digital DNARD devices

- IDE
- Keyboard
- Mouse
- Ethernet
- Smartcard
- Audio
- Joystick

Supported Chalice CATS devices

- ALI M1543 southbridge inc PS/2 keyboard & mouse, ide, serial
- Parallel, USB and ISA bus
- RTC

Supported Intel EBSA285 & Chalice CATS PCI devices

- PCI - PCI bridges
  - DC21150
  - DC21152
  - DC21153
  - DC21154
  - Hint HB1

- Digital DC21x4x-based PCI Ethernet adapters
  - Cogent EM1X0, EM960 (a.k.a. Adaptec ANA-69XX)
  - Cogent EM964 [b]
  - Cogent EM4XX [b]
  - Compex Readylink PCI
  - DANPEX EN-9400P3
  - Digital Celebris GL, GLST on-board ethernet
  - DEC (Digital) PCI Ethernet/Fast Ethernet adapters (all)

  - JCIS Condor JC1260
  - Linksys PCI Fast Ethernet
  - SMC EtherPower 10, 10/100 (PCI only!)
  - SMC EtherPower
  - SVEC
  - PN0455
  - SVEC FD1000-TP
  - Znyx ZX34X
- Qlogic ISP [12]0x0 SCSI/FibreChannel boards
- Adaptec AHA-2910, 2915, 2920, and 2930C adapters
- Adaptec AHA-2x4x[U][2][W] cards and onboard PCI designs using the AIC-7770, AIC-7850, AIC-7860, AIC-7870, AIC-7880 and AIC-789x chipsets
- Adaptec AHA-394x[U][W] cards [b]
- Adaptec AHA-3950U2 cards
- Adaptec AHA-3960, 19160 and 29160 Ultra-160 adapters
- NE2000 PCI ethernet adapters
- Universal Serial Bus
  - UHCI host controllers
  - OHCI host controllers
  - Hubs
  - Keyboards using the boot protocol
  - Mice
  - Printers
  - Generic support for HID devices
- Video cards
  - Diamond Stealth 3D 2000 Pro
  - ATI Charger 4MB
  - STB Velocity 128
  - Cirrus Logic 5446
  - IGS 2010
  - IGS 5000
  - S3 based interfaces

Drivers for hardware marked with ''[*]'' are *not* present in installation kernels.

Other PCI devices may be supported by Intel EBSA285 & Chalice CATS but have not been tested.

Support for some devices is limited to particular kernels. eg there is no SA110 support in A7000 kernels.

Unsupported devices

- Acorn/Aleph1 PC cards
- Any SCSI card using a PowerROM
- Podule based serial ports
- Castle SCSI/Ethernet cards

Drivers are planned for some of the above devices.

Getting the NetBSD System on to Useful Media

You will need to have an installation kernel and possibly a boot application. See `arm32/`*`platform`*`/prep` for details.

Installation is supported from several media types, including:

- 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.

CD-ROM 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 October, 2000).

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-1.5
# cd arm32/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

First and foremost, before beginning the installation process, *make sure you have a reliable backup* of any data on your hard disk that you wish to keep. Repartitioning your hard disk is an excellent way to destroy important data.

Second, read and perform the instructions in `arm32/`*`platform`*`/prep` that are specific to your platform for partitioning and booting (even if you're dedicating a device to NetBSD).

Finally, when you are happy with your NetBSD installation, do whatever is necessary to restore order to the partition you took space away from. This will most likely involve restoring files, but might involve some other ''house-work''.

Your hard disk is now prepared to have NetBSD installed on it, and you should proceed with the installation instructions.

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 should know the geometry of your hard disk, i.e. the sector size (note that sector sizes other than 512 bytes are not currently supported), the number of sectors per track, the number of tracks per cylinder (also known as the number of heads), and the number of cylinders on the disk. The NetBSD kernel will try to discover these parameters on its own, and if it can it will print them at boot time. If possible, you should use the parameters it prints. (You might not be able to because you're sharing your disk with another operating system, or because your disk is old enough that the kernel can't figure out its geometry.)

If NetBSD will be sharing the disk with RiscOS or another operating system, you should have already completed the section of these notes that instructed you on how to prepare your hard disk. You should know the size of the NetBSD area of the disk and its offset from the beginning of the disk. You will need this information when setting up your NetBSD partitions.

You should now be ready to install NetBSD. It might be handy for you to have a pencil, some paper, and a calculator handy.

The following is a walk-through of the steps you will take while getting NetBSD installed on your hard disk. If any question has a default answer, it will be displayed in brackets (''[]'') after the question. If you wish to stop the installation, you may press `CONTROL-C` at any time, but if you do, you'll have to begin the installation process again from scratch.

Boot your machine using the installation kernel for your platform. (Instructions for doing this on your platform can be found in the preparation section of this document.)

If this doesn't work, ensure that you're using the correct kernel for your hardware.

Depending upon your platform and the method of loading the, it may take a while to load the kernel.

You will then be presented with the NetBSD kernel boot messages. You will want to read them, to determine your disk's name and geometry. Its name will be something like `sd0` or `wd0` and the geometry will be printed on a line that begins with its name. As mentioned above, you will need your disk's geometry when creating NetBSD 's partitions. You will also need to know the name, to tell the install tools what disk to install on.

While booting, you will probably see several warnings. You should be warned that no swap space is present, and that `init`(8) cannot find `/etc/rc.` Do not be alarmed, these are completely normal. When you reach the prompt asking you for a shell name, just press `RETURN`.

You will be asked if you wish to install or upgrade your system or go to a shell prompt. Enter **`install`**.

You will be presented with a welcome message and a prompt, asking if you wish to proceed with the installation process. If you wish to proceed, enter '**`y`**' and press `RETURN`.

You will be asked what type of disk driver you have. The valid options are listed by the install program, to make sure you get it right.

The install program will then tell you which disks of that type it can install on, and ask you which it should use. Reply with the name of your disk. (The first disk of the type you selected, either `wd0` for IDE disks, or `sd0` for SCSI disks, is the default.)

You will then be asked to name your disk's disklabel. The default response is '**`mywd`**' or '**`mysd`**' depending on the type of your disk, and for most purposes it will be OK. If you choose to name it something different, make sure the name is a single word and contains no special characters. You don't need to remember this name.

You will be prompted for your disk's geometry information, i.e. the number of bytes per sector, cylinders on the disk, tracks per cylinder (heads), and sectors per track. Enter them when they are requested. If you make a mistake, press `CONTROL-C` and when you get to the shell prompt, restart the install process by running the **`install`** command. Once you have entered this data, the install program will tell you the total size of your disk, in both sectors, and cylinders. Remember this number; if you're installing on the whole disk, you'll need it again soon.

When describing your partitions, you will have the option of entering data about them in units of disk sectors or cylinders. If you choose to enter the information in units of sectors, remember that, for optimal performance, partitions should begin and end on cylinder boundaries. You will be asked about which units you wish to use, and you should reply with '**`c`**' for cylinders, or '**`s`**' for sectors.

You will be asked for the size of the NetBSD portion of the disk. If you're installing on the whole disk, reply with the size of the disk, as printed earlier by the install program. If you're using only part of the disk, reply with the size that you specified in the partition editor. (Don't forget to enter the size in the units you specified in the last step!)

If you are not installing on the whole disk, you will be asked for the offset of the NetBSD partition from the beginning of the disk. Reply with the appropriate offset (again, in whichever units you specified), as determined by how you set up your disk using the partition editor.

You will be asked to enter the size of your NetBSD root partition (`/`). It should be at least 13 MB, but if you are going to be doing development, 14-16 MB is a more desirable size. This size should be expressed in units of sectors or cylinders, depending on which you said you wanted to use.

Next, you will be asked for the size of your swap partition. You should probably allocate twice as much swap space as you have real memory. Systems that will be heavily used should have more swap space allocated, and systems that will be lightly used can get by with less. If you want the system to be able to save crash dumps when it panics, you will need at least as much swap space as you have RAM. Again, this number should be expressed in units of sectors or cylinders, as appropriate.

The install program will then ask you for information about the rest of the partitions you want on your disk. For most purposes, you will want only one more partition, `/usr`. (Machines used as servers will probably also want `/var` as a separate partition. That can be done with these installation tools, but is not covered here.) The install program will tell you how much space there is left to be allocated in the NetBSD area of the disk, and, if you only want one more partition (`/usr`), you should enter it at the prompt when the installer asks you how large the next partition should be. It will then ask you for the name of the mount point for that partition. If you're doing a basic installation, that is `/usr`.

You are now at the point of no return. Nothing has been written to your disk yet, but if you confirm that you want to install NetBSD, your hard drive will be modified, and its contents may be scrambled at the whim of the install program. This is especially likely if you have given the install program incorrect information. If you are sure you want to proceed, enter **`yes`** at the prompt.

The install program will now label your disk and make the file systems you specified. The file systems will be initialized to contain NetBSD bootstrapping binaries and configuration files. It will also create an `/etc/fstab` for your system, and mount all of the file systems under `/mnt`. (In other words, your `/` (root) partition will be mounted on `/mnt`, your `/usr` partition on `/mnt/usr`, and so on.) There should be no errors in this section of the installation. If there are, restart from the beginning of the installation process.

You will be placed at a shell prompt ('`#`'). The task is to install the distribution sets. The flow of installation differs depending on your hardware resources, and on what media the distribution sets reside.

To install from floppy

The first thing you should do is pick a temporary directory where the distribution files can be stored. To do this, enter the command **`Set_tmp_dir`**, and enter the name of the temporary directory. (Don't forget that if your disk is still mounted under `/mnt`; you should probably pick a directory under `/mnt/usr`.)

After you have picked a temporary directory, enter the **`Load_fd`** command, to load the distribution sets from your floppies.

You will be asked which floppy drive to use. Enter '**`0`**' (zero) if you're using the first floppy drive, or enter '**`1`**' if you're using the second.

You will be prompted to insert a floppy into the drive, to have its contents copied to your hard disk. Do so, and press `RETURN` to begin copying. When that is done, read the remainder of the floppies that contain the distribution sets that you want to install, one by one. When the last is read, and you are being prompted for another, press `CONTROL-C`.

Run the **`Extract`** command once for each distribution set you wish to install. For instance, if you wish to install the **base** distribution set, followed by the **text** distribution set, and finally the **etc** distribution set, use the commands:

```
# Extract base
# Extract text
# Extract etc
```

For each extraction, it will ask you if the extraction should be verbose. If you reply affirmatively, it will print out the name of each file that's being extracted.

Note: If you know that you will be running low on disk space when installing NetBSD, you can load and extract one distribution set at a time. To do this, load only the floppies which contain the files for the first distribution set, extract them, and then change to the temporary directory and remove them with the command **`rm set_name.??`**

Once you are finished extracting all of the sets that you wish to install, you should proceed to the instructions below (after the last install medium type-specific instructions), that explain how you should configure your system.

To install from tape

The first thing you should do is pick a temporary directory where the distribution files can be stored. To do this, enter the command **Set_tmp_dir**, and enter the name of the temporary directory. (Don't forget that your disk is mounted under `/mnt`; you should probably pick a directory under `/mnt/usr`.) The default is `/mnt/usr/distrib`.

After you have picked a temporary directory, enter the **Load_tape** command, to load the distribution sets from tape.

You will be asked which tape drive to use. The default is `rst0`, which is correct if you're using the SCSI tape drive with the lowest SCSI-ID number. (For the SCSI tape drive with the next lowest SCSI-ID number, you should use `rst1`, and so on.)

You will be prompted to press `RETURN` when you have inserted the tape into the tape drive. When you do, the contents of the tape will be extracted into the temporary directory, and the names of the files being extracted will be printed.

After the tape has been extracted, to go the directory containing the first distribution set you wish to install. (Depending on how you made the tape, it's probably a subdirectory of the temporary directory you specified above.) Once there, run the **Set_tmp_dir** command again, and accept its default answer by pressing return at the prompt.

Use the **Extract** command to extract the distribution set. For instance, if you're extracting the **base** set, use the command:

```
# Extract base
```

You will be asked if you wish the extraction to be verbose. If you reply affirmatively, the name of each file being extracted will be printed.

Repeat the previous two steps for each distribution set you wish to install. Change to the set's directory, run **Set_tmp_dir**, and then run **Extract** *set_name* to extract the set.

Once you are finished extracting all of the sets that you wish to install, you should proceed to the instructions below (after the last install medium type-specific instructions), that explain how you should configure your system.

To install via FTP or NFS

The first thing you should do is pick a temporary directory where the distribution files can be stored. To do this, enter the command **Set_tmp_dir**, and enter the name of the temporary directory. (Don't forget that your disk is mounted under `/mnt`; you should probably pick a directory under `/mnt/usr`.) The default is `/mnt/usr/distrib`.

Configure the appropriate ethernet interface (e.g. ea0, eb0, etc.) up, with a command like:

ifconfig *ifname ipaddr* [**netmask** *netmask*]

where *ifname* is the interface name, like those listed above, and *ipaddr* is the numeric IP address of the interface. If the interface has a special netmask, supply the word **netmask** and that netmask at the end of the command line. (The brackets indicate that those arguments are optional.) For instance, to configure interface *ea0* with IP address `129.133.10.10`, use the command:

```
# ifconfig ea0 129.133.10.10
```

and to configure interface *eb0* with IP address `128.32.240.167` and a special netmask, `0xffffff00`, use the command:

```
# ifconfig eb0 128.32.240.167 netmask 0xffffff00
```

If the NFS server or FTP server is not on a directly- connected network, you need to set up a route to it using a command like:

route add default *gate_ipaddr*

where *gate_ipaddr* is your gateway's numeric IP address.

If you are NFS-mounting the distribution sets, mount them on the temporary directory with a command like:

mount -t nfs *serv_ipaddr:dist_dir tmp_dir*

where *serv_ipaddr* is the server's numeric IP address, *dist_dir* is the path to the distribution files on the server, and *tmp_dir* is the name of the local temporary directory.

Once this is done, proceed as if you had loaded the files from tape, changing to the appropriate directories, running **Set_tmp_dir**, and running **Extract** as appropriate.

If you are retrieving the distribution sets using ftp, change into the temporary directory, and execute the command:

ftp *serv_ipaddr*

where *serv_ipaddr* is once again the server's numeric IP address. Get the files with FTP, taking care to use binary mode when transferring the files.

Once you have all of the files for the distribution sets that you wish to install, you can proceed using the instructions above, as if you had installed from a floppy. (Note that as with the floppy install, if you're short on disk space, you can transfer only one set at a time, extract it, then delete it, to save space.)

To install from CD-ROM

First create a mount point so that you can mount the CD-ROM:

```
# mkdir /mnt/cdrom
```

If you get an error here of ''mkdir: /mnt/cdrom'', don't worry it just means that you didn't need to create the directory.

Then all you need to do is mount the CD-ROM. For the first CD-ROM drive use:

```
# mount -rt cd9660 /dev/cd0a /mnt/cdrom
```

Or, for the second use:

```
# mount -rt cd9660 /dev/cd1a /mnt/cdrom
```

Once this is done, extract the required sets as described in the **To install from floppy** section, but ensure that you set the temporary directory to the location of the sets on the CD-ROM (usually `/cdrom/distrib`, but check the release notes that came with the CD).

Completing your installation

Once you have finished extracting all of the distribution sets that you wish to install, and are back at the '#' prompt, you are ready to configure your system. The configuration utility expects that you have installed the **base** and **etc** distribution sets. If you have not, you will not be able to run it successfully (nor will you have a functional system, in any case). To configure your newly-installed NetBSD system, run the command **Configure**. It will ask you for the system's host name, domain name, and other network configuration information. It will set up your configuration files and make the device nodes for the newly-installed system.

Congratulations, you have successfully installed NetBSD 1.5.

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`, 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-write. When the system asks you to choose a shell, simply press `RETURN` to get to a prompt. If you are asked for a terminal type, respond with **`vt220`** (or whatever is appropriate for your terminal type) and press `RETURN`. At this point, you need to configure at least one file in the `/etc` directory. 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 interface <int>, 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`.

   For Digital DNARD machines you will need to edit `/etc/ttys` and change the console terminal type from vt100 to pc3.

2. Logging in

   After reboot, you can log in as `root` at the login prompt. There is no initial password, but 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.

   You will need to set up a configuration file, see /usr/X11R6/lib/X11/XF86Config.eg for an example. The xf86config(1) utility can interactively create a first version of such a configuration file for you. See **http://www.xfree86.org/** and the XFree86 manual page for more 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 browsable listing of available packages 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/
   - Package sources for compiling packages 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), as with the command:

     ```
     # mkdir /usr/pkgsrc; tar -C /usr/pkgsrc -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 (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

Upgrading from a previous version of NetBSD isn't currently supported by NetBSD/arm32. If you are currently running NetBSD/arm32 then make a FULL BACKUP of your current installation, and install NetBSD 1.5 from scratch. Obviously some of the steps can be skipped (in particular, hard disk partitioning) as they will already be done.

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 1.5.

General issues

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

  In previous releases of NetBSD, `/etc/rc` was a traditional BSD style monolithic file. As of NetBSD 1.5, each discrete program or substem 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.

Issues affecting an upgrading from NetBSD 1.4 or later

- `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) pathname changed

  `sysctl`(8) is 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
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
Jim Wise
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.
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:

Alistair Crooks	agc@netbsd.org
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):

Mark Brinicombe	mark@netbsd.org	**arm32**
Jeremy Cooper	jeremy@netbsd.org	**sun3x**
Ross Harvey	ross@netbsd.org	**alpha**
Jun-ichiro itojun Hagino	itojun@netbsd.org	**sh3**
Ben Harris	bjh21@netbsd.org	**arm26**
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**
Anders Magnusson	ragge@netbsd.org	**vax**
Minoura Makoto	minoura@netbsd.org	**x68k**
Phil Nelson	phil@netbsd.org	**pc532**
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**
Frank van der Linden	fvdl@netbsd.org	**i386**
Leo Weppelman	leo@netbsd.org	**atari**
Nathan Williams	nathanw@netbsd.org	**sun3**
Steve Woodford	scw@netbsd.org	**mvme68k**

The NetBSD 1.5 Release Engineering team:

Chris G. Demetriou	cgd@netbsd.org
Havard Eidnes	he@netbsd.org
Ted Lemon	mellon@netbsd.org
John Hawkinson	jhawk@netbsd.org
Perry Metzger	perry@netbsd.org
Curt Sampson	cjs@netbsd.org
Jason Thorpe	thorpej@netbsd.org
Todd Vierling	tv@netbsd.org

Developers and other contributors:

Steve Allen	wormey@netbsd.org
Julian Assange	proff@netbsd.org
Lennart Augustsson	augustss@netbsd.org
Christoph Badura	bad@netbsd.org
Robert V. Baron	rvb@netbsd.org
Erik Berls	cyber@netbsd.org
John Birrell	jb@netbsd.org
Mason Loring Bliss	mason@netbsd.org
Manuel Bouyer	bouyer@netbsd.org
John Brezak	brezak@netbsd.org
Allen Briggs	briggs@netbsd.org
Aaron Brown	abrown@netbsd.org
David Brownlee	abs@netbsd.org
Frederick Bruckman	fredb@netbsd.org
Jon Buller	jonb@netbsd.org
Simon Burge	simonb@netbsd.org
Dave Burgess	burgess@cynjut.infonet.net
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
Chuck Cranor	chuck@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
Jaromir Dolecek	jdolecek@netbsd.org
Andy Doran	ad@netbsd.org
Roland Dowdeswell	elric@netbsd.org
Matthias Drochner	drochner@netbsd.org
Jun Ebihara	jun@netbsd.org
Havard Eidnes	he@netbsd.org
Enami Tsugutomo	enami@netbsd.org
Bernd Ernesti	veego@netbsd.org
Erik Fair	fair@netbsd.org
Hubert Feyrer	hubertf@netbsd.org
Thorsten Frueauf	frueauf@netbsd.org
Castor Fu	castor@netbsd.org
Ichiro Fukuhara	ichiro@netbsd.org
Brian R. Gaeke	brg@dgate.org
Thomas Gerner	thomas@netbsd.org
Simon J. Gerraty	sjg@netbsd.org
Justin Gibbs	gibbs@netbsd.org
Adam Glass	glass@netbsd.org
Michael Graff	explorer@netbsd.org
Brad Grantham	grantham@tenon.com
Brian C. Grayson	bgrayson@netbsd.org
Matthew Green	mrg@netbsd.org
Juergen Hannken-Illjes	hannken@netbsd.org
Charles M. Hannum	mycroft@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
Nick Hudson	skrll@netbsd.org
Martin Husemann	martin@netbsd.org
Dean Huxley	dean@netbsd.org
Bernardo Innocenti	bernie@netbsd.org
ITOH Yasufumi	itohy@netbsd.org
IWAMOTO Toshihiro	toshii@netbsd.org
Matthew Jacob	mjacob@netbsd.org
Lonhyn T. Jasinskyj	lonhyn@netbsd.org
Chris Jones	cjones@netbsd.org
Takahiro Kambe	taca@netbsd.org

Antti Kantee	pooka@netbsd.org
Lawrence Kesteloot	kesteloo@cs.unc.edu
Thomas Klausner	wiz@netbsd.org
Klaus Klein	kleink@netbsd.org
Wayne Knowles	wdk@netbsd.org
John Kohl	jtk@netbsd.org
Kevin Lahey	kml@netbsd.org
Johnny C. Lam	jlam@netbsd.org
Martin J. Laubach	mjl@netbsd.org
Ted Lemon	mellon@netbsd.org
Joel Lindholm	joel@netbsd.org
Mike Long	mikel@netbsd.org
Warner Losh	imp@netbsd.org
Federico Lupi	federico@netbsd.org
Brett Lymn	blymn@netbsd.org
Paul Mackerras	paulus@netbsd.org
David Maxwell	david@netbsd.org
Dan McMahill	dmcmahill@netbsd.org
Gregory McGarry	gmcgarry@netbsd.org
Neil J. McRae	neil@netbsd.org
Perry Metzger	perry@netbsd.org
der Mouse	mouse@netbsd.org
Joseph Myers	jsm@netbsd.org
Ken Nakata	kenn@netbsd.org
Bob Nestor	rnestor@netbsd.org
NONAKA Kimihiro	nonaka@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
Dante Profeta	dante@netbsd.org
Chris Provenzano	proven@netbsd.org
Waldi Ravens	waldi@moacs.indiv.nl.net
Darren Reed	darrenr@netbsd.org
Michael Richardson	mcr@netbsd.org
Tim Rightnour	garbled@netbsd.org
Gordon Ross	gwr@netbsd.org
Heiko W. Rupp	hwr@netbsd.org
SAITOH Masanobu	msaitoh@netbsd.org
Curt Sampson	cjs@netbsd.org
Wilfredo Sanchez	wsanchez@netbsd.org
Ty Sarna	tsarna@netbsd.org
SATO Kazumi	sato@netbsd.org
Matthias Scheler	tron@netbsd.org
Karl Schilke (rAT)	rat@netbsd.org
Konrad Schroder	perseant@netbsd.org
Reed Shadgett	dent@netbsd.org
Tim Shepard	shep@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
Bill Sommerfeld	sommerfeld@netbsd.org
Bill Squier	groo@netbsd.org
Bill Studenmund	wrstuden@netbsd.org
Kevin Sullivan	sullivan@netbsd.org
SUNAGAWA Keiki	kei@netbsd.org
Kimmo Suominen	kim@netbsd.org
Matt Thomas	matt@netbsd.org
Christoph Toshok	toshok@netbsd.org
UCHIYAMA Yasushi	uch@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
Assar Westerlund	assar@netbsd.org
Todd Whitesel	toddpw@netbsd.org
Rob Windsor	windsor@netbsd.org
Dan Winship	danw@netbsd.org
Jim Wise	jwise@netbsd.org
Michael Wolfson	mbw@netbsd.org
Colin Wood	ender@netbsd.org

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.

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 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 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 Neil Carson.

This product includes software developed by Scott Stevens.