NAME

INSTALL – Installation procedure for NetBSD/vax

DESCRIPTION

About this Document

This document describes the installation procedure for NetBSD 1.4.2 on the *vax* 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 UN*X-like operating system derived from the Berkeley Networking Release 2 (Net/2), 4.4BSD-Lite, and 4.4BSD-Lite2 sources. NetBSD runs on twenty different system architectures featuring eight distinct families of CPUs, and is being ported to more. The NetBSD 1.4.2 release contains complete binary releases for fifteen different machine types. (The remaining ones 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, 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 1.4.2

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 1.4.2.

NetBSD 1.4.2 is an upgrade of NetBSD 1.4.1, NetBSD 1.4 and earlier releases of NetBSD such as versions 1.3.3, 1.2 etc.

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 1.4 was started are designated by version identifiers such as 1.4A, 1.4B, 1.4P etc. These identifiers do not designate releases, but indicate major changes in internal kernel APIs. Note that the kernel from NetBSD 1.4.2 can *not* be used to upgrade a system running one of those intermediate development versions. Trying to use the NetBSD 1.4.2 kernel on such a system *will* in all probability 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 1.4.2 to determine if a given feature is present or absent in 1.4.2. The development of 1.4, 1.4.1 and 1.4.2 is done on a separate branch in the CVS repository which was created when the release cylcle for 1.4 was started, and during the release cycles for all these versions selective fixes (with minor impact on the stability of the code on the release branch) have been imported from the main development trunk. So, there are features in 1.4.2 which were not in e.g. 1.4H, but the reverse is also true.

Major Changes Between 1.4.1 and 1.4.2

The complete list of changes between NetBSD 1.4.1 and 1.4.2 can be found in the file CHANGES-1.4.2 in the top directory of the source tree. The following are the highlights only.

- A driver for the Alteon Gigabit Ethernet cards has been added, see `ti`(4).
- A driver for the Realtek 8129/8239 Fast Ethernet PCI cards have been added, see `rl`(4).
- A driver for the DPT SmartCache and SmartRAID III or IV SCSI adapters has been added, see `dpt`(4).
- A driver for the BOCA IOAT66 6-port ISA serial adapter has been added, see `ioat`(4).
- Support for the X-surf Amiga board has been added.
- Support has been added for ext2fs revision 1, with read-only support for the ''sparse super'' and the ''file-type'' options.
- BIND has been upgraded to version 8.2.2-P5.
- The IPF packet filtering software has been updated to version 3.3.5.
- Tcpdump now does hex/ascii dumps of packet contents if asked to do so.
- An implementation of the System V user management utilities has been added.
- The name ''errno'' is now always a macro which expands to a function call. This is done to ease the integration of thread libraries with the code in both system and third-party libraries. Please include <errno.h> to access the correct definition of ''errno''.
- A utility for making temporary files for shell scripts has been added, see `mktemp`(1).
- The automounter utility `amd`(8) has been updated to fix a security problem.
- A security problem in procfs has been fixed. Procfs is not used by default in NetBSD.
- The floating point emulation on the ports using the m68k CPU has been reverted to the version in NetBSD 1.4 (the version in 1.4.1 had problems).
- Several subsystems have received substantial work, such as RAIDframe, LFS, and the package tools.

In addition, many, many bugs have been fixed -- more than 100 problems reported through our problem tracking system have been fixed, and many other non-reported problems have also been found and fixed. See the CHANGES-1.4.2 file for the complete list.

Known caveats with 1.4.2

The following are the major known issues with NetBSD 1.4.2.

- The upgrade of `ipf`(8) caused a change of the kernel API. Thus, if you are using `ipf`(8) you need to upgrade both the kernel and the user-land utilities to control that feature in order for it to work.
- As part of fixing a kernel bug, ''struct vfsnode'' needed to change size. This will unfortunately, and contrary to tradition for patch releases, cause incompatibilities for users of file system LKMs compiled for 1.4 and 1.4.1.
- Although LFS has been improved between 1.4.1 and 1.4.2, it can probably still be characterized as being of ''beta test'' quality, e.g. filling the file system still gives ''interesting'' effects. It should probably not be used to store critical data quite yet.
- The version of the IPF packet filtering software in NetBSD 1.4.2 must be enabled with ''ipf -E'' before filtering or NAT rules can be set. Failure to do so may result in a panic.

Major Changes Between 1.4 and 1.4.1

The complete list of changes between NetBSD 1.4 and 1.4.1 can be found in the file CHANGES-1.4.1 in the top directory of the source tree.

- The NetBSD/alpha port's compatibility for Digital UNIX executables has been greatly improved.
- Many bug fixes and improvements of the installation tools and utilities.
- Support for more PCI serial/parallel cards has been added.
- It is now possible to boot NetBSD/i386 on systems with 1GB of RAM.
- The floating point emulation on the ports using the m68k CPU has been upgraded.
- A fatal problem with /dev/random has been found and fixed.
- Support for Alpha 21264 ev6 based systems has been added to NetBSD/alpha.

Major Changes Between 1.3 and 1.4

The NetBSD 1.4 release is a substantial improvement over its predecessors. We have provided 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 results of these improvements is a stable operating system fit for production use that rivals most commercially available systems.

It is impossible to completely summarize the nearly two years of development that went into the NetBSD 1.4 release. Some highlights include:

- Substantial improvements in the TCP/IP implementation, including numerous performance enhancements and bug fixes by Jason Thorpe and others.
- A new, high efficiency kernel memory pool allocator by Paul Kranenburg. This has been integrated into most kernel subsystems.
- A new, totally rewritten virtual memory subsystem, UVM, created by Chuck Cranor, which is substantially cleaner and better performing than the old Mach derived VM subsystem.
- Improved POSIX and XPG standards compliance.
- Completion of the integration of all remaining 4.4BSD Lite-2 kernel improvements and bug fixes that had not been previously integrated. (Integration of all userland components was completed before NetBSD 1.3)
- Several new ports, including macppc, bebox, sparc64, next68k, and others, have been integrated into the source tree.
- The system compilers have been upgraded to egcs 1.1.1, and the system compiler toolchain now (mostly) uses the latest versions of GNU binutils instead of the obsolete versions left over from 4.4BSD Lite.
- Everyone's favorite `ftp`(1) client has been improved even further. See the man page for details.
- A new architecture independent console driver, `wscons`(4), has been integrated into many ports.
- Numerous improvements have been made to the audio subsystem support, including support for MIDI device drivers.
- Linux compatibility support has been improved.
- A number of scheduler enhancements have yielded dramatic improvements in interactive performance and better control of background tasks.

- Several network tunneling protocols, including GRE and IP in IP, have been implemented.
- Kernel support for the CODA distributed file system has been added.
- Manuel Bouyer completed major changes to the IDE support. It is now architecture independent. Major changes have been made to the IDE code for better error handling, improved ATAPI support, 32 bit data I/O support and bus-master DMA support on PCI IDE controllers.
- Lennart Augustsson has added full USB support, permitting the use of a wide variety of Universal Serial Bus peripherals. The drivers should easily port to any future platforms that support the PCI bus. See `usb`(4) for an overview.
- RAIDframe, version 1.1, from the Parallel Data Laboratory at Carnegie Mellon University, has been integrated. Supports RAID levels 0, 1, 4, 5, and more.
- Luke Mewburn added `nsswitch.conf`(5) functionality to the system to specify the search order for system databases.
- `syslogd`(8) now supports listening on multiple sockets, to make the chrooting of servers easier.
- Most third party packages have been updated to the latest stable release.

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.

This is the third public release of NetBSD for the VAX.

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. In addition, Anonymous CVS access to the NetBSD source tree has been added since NetBSD 1.4.1; see

http://www.netbsd.org/Changes/#anoncvs-available

We have also added a browsable CVS repository on the web at

http://cvsweb.netbsd.org/cgi-bin/cvsweb.cgi/

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

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

NetBSD 1.4.2 Release Contents

The root directory of the NetBSD 1.4.2 release is organized as follows:

`.../NetBSD-1.4.2/`

`BUGS` Known bugs list (somewhat incomplete and out of date).

`CHANGES` Changes since earlier NetBSD releases.

`LAST_MINUTE` Last minute changes.

`MIRRORS` A list of sites that mirror the NetBSD 1.4.2 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 1.4.2 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 (i.e. the 'domestic' portion) that may be subject to export regulations of the United States. It is your responsibility to determine whether or not it is legal for you to export these portions and to act accordingly. The NetBSD project maintains a web page at

http://www.NetBSD.ORG/Misc/crypto-export.html

which should contain up-to-date information on this issue.

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:

secrsrc.tgz: This set contains the "domestic" sources. These sources may be subject to United States export regulations.
421K gzipped, 2M uncompressed

gnusrc.tgz: This set contains the "gnu" sources, including the source for the compiler, assembler, groff, and the other GNU utilities in the binary distribution sets.
19M gzipped, 84.3M uncompressed

syssrc.tgz: This set contains the sources to the NetBSD 1.4.2 kernel, config(8), and dbsym(8).
13.9M gzipped, 68.3M uncompressed

sharesrc.tgz: 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.
3M gzipped, 11.9M uncompressed

src.tgz: This set contains all of the NetBSD 1.4.2 sources which are not mentioned above.
14.4M gzipped, 67.4M uncompressed

Most of the above source sets are located in the `source/sets` subdirectory of the distribution tree. The secrsrc.tgz set is contained in the `source/security` subdirectory. This set, which may only be available to users in the United States and Canada, contains the sources normally found in `/usr/src/domestic` – primarily Kerberos and other cryptographic security related software. (Remember, because of United States law, it may not be legal to distribute this set to locations outside of the United States and Canada. Again, see

http://www.NetBSD.ORG/Misc/crypto-export.html

for updated information on this issue.)

The source sets are distributed as compressed tar files. They may be unpacked into `/usr/src` with the command:

```
cat set_name.tgz | gunzip | (cd /; tar xpf - )
```

The `sets/Split/` and `security/Split/` subdirectories contain 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 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.?? | gunzip | (cd /; tar xpf - )
```

In each of the source distribution set directories, there is a file named `CKSUMS` which contains the checksums of the files in that directory, as generated by the `cksum`(1) utility. You can use cksum to check the integrity of the archives, if you suspect that one of the files is corrupt and have access to a cksum binary. Checksums based on other algorithms may also be present – see the `release`(7) man page for details.

NetBSD/vax Subdirectory Structure

The vax-specific portion of the NetBSD 1.4.2 release is found in the `vax` subdirectory of the distribution:

`.../NetBSD-1.4.2/vax/`
`INSTALL.html`
`INSTALL.ps`
`INSTALL.txt`
`INSTALL.more` Installation notes; this file. The `.more` file contains underlined text using the `more`(1) conventions for indicating italic and bold display.
`binary/`
`sets/` vax binary distribution sets; see below.
`security/` vax security distribution; see below.
`installation/`

Binary Distribution Sets

The NetBSD vax binary distribution sets contain the binaries which comprise the NetBSD 1.4.2 release for the vax. There are eight binary distribution sets and the *security* distribution set. The binary distribution sets can be found in the `vax/binary/sets` subdirectory of the NetBSD 1.4.2 distribution tree, and are as follows:

base The NetBSD 1.4.2 vax **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.
10.1M gzipped, 24.8M 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.
8.1M gzipped, 27.3M 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.)
60K gzipped, 350K uncompressed

games This set includes the games and their manual pages.
2.8M gzipped, 6.8M uncompressed

kern This set contains a NetBSD/vax 1.4.2 GENERIC kernel, named `/netbsd`. You *must* install this distribution set.
517k gzipped, 1M 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.
4.2M gzipped, 17.2M uncompressed

misc This set includes the (rather large) system dictionaries, the typesettable document set, and other files from `/usr/share`.
2.1M gzipped, 8.1M uncompressed

text This set includes NetBSD's text processing tools, including `groff`(1), all related programs, and their manual pages.
1.2M gzipped, 4.3M uncompressed

The vax security distribution set is named **secr** and can be found in the

`vax/binary/security`

subdirectory of the NetBSD 1.4.2 distribution tree. It contains security-related binaries which depend on cryptographic source code. You do not need this distribution set to use encrypted passwords in your password file; the **base** distribution includes a crypt library which can perform only the one-way encryption function. The security distribution includes a version of the Kerberos IV network security system, and a Kerberized version of `telnet`(1) program. The **secr** distribution set can be found only on those sites which carry the complete NetBSD distribution and which can legally obtain it. Because of United States law, it may not be legal to distribute this set to locations outside of the United States and Canada. See

http://www.NetBSD.ORG/Misc/crypto-export.html

for updated information on this issue.

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.

xcomp The extra libraries and include files needed to compile X source code.

xcontrib Programs that were contributed to X.

xfont Fonts needed by X.

The vax 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 xfp`** command from /.

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

All BSDSUM files are historic BSD checksums for the various files in that directory, in the format produced by the command: **`cksum -o 1`** *`file`*

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

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

All SYSVSUM files are 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/vax System Requirements and Supported Devices

Hardware supported by NetBSD/vax 1.4.2 includes (but may not be limited to):

CPU types:
VAX 11/750, VAX 11/78{0,5}, VAX 8200/8250/8300/8350, VAX 8600/8650, MicroVAX II/VAXstation II, MicroVAX III, MicroVAX 3300/3400, MicroVAX 3500/3600, MicroVAX 3800/3900, MicroVAX 2000/VAXstation 2000, VAXstation 3100 model 30,38,40,48,76, VAXstation 3200/3500 and VAXstation 4000/60.

Networking:
DEUNA/DELUA Unibus ethernet, DEQNA/DELQA Q22 bus ethernet, and LANCE chip where available.

Serial lines:
DHU11/DHV11/DL11/DLV11/DZ11/DZQ11/DZV11 Unibus/Q22 bus asynchronous lines and DZ11-compatible lines on VAXstations.

Disks: UDA50 Unibus MSCP controller, KDA50/RQDX1/2/3 Q22 bus MSCP controller, KDB50 BI-bus MSCP controller, MFM and SCSI controllers on VAXstations, RP04/05/06/07 and RM02/03/05/80 Massbus disks, and Console RL02 on VAX 8600.

Some third-party controllers are also known to work, other do not.

Tapes: TMSCP on Q22 bus (TK50/70, TU81), TMSCP on BI bus and SCSI tapes on VAXstations.

Floppy: RX33/RX50 on RQDX controllers. RX50 on VAX 8200.

The minimal configuration requires 2M of RAM and ~40MB of disk space, but the installation really requires 6MB RAM unless you plan on using Jedi powers.

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

Partition	Advised	Needed
root (/)	32M	16M
user (/usr)	110M	45M
swap	2 or 3 * RAM	

Anything else is up to you!

Getting the NetBSD System on to Useful Media

Installation is supported from several media types, including:

- Floppy
- FTP
- Tape
- NFS

Note Installing on a "bare" machine requires some bootable device; either a tape or floppy drive or a NFS server together with a machine that can act as a MOP loader, such as another machine running NetBSD. NetBSD/vax can use both BOOTP/DHCP and BOOTPARAMS for netboot installations.

The procedure for transferring the distribution sets onto installation media depends on the type of media. most of it is up to you, depending what you want to install, but preferred are to do the installation over network as soon as the install kernel is booted.

Creating boot tapes

Fetch the bootable bootfs image from

`.../NetBSD-1.4.2/vax/installation/bootfs/boot.fs.gz`

Gunzip boot.fs.gz and write it on the beginning of the tape. Under NetBSD this is done via:

```
gunzip boot.fs.gz
mt -f /dev/nrmt0 rewind
dd if=boot.fs of=/dev/nrmt0
mt -f /dev/nrmt0 rewoffl
```

Under Ultrix the tape name is different:

```
gunzip boot.fs.gz
mt -f /dev/rmt0h rewind
dd if=boot.fs of=/dev/rmt0h
mt -f /dev/rmt0h rewoffl
```

Of course, if you have a tape unit other than unit 0 you have to use the corresponding unit number.

If you wish to install the sets from tape then stage you will need to download the *.tgz files from

`.../NetBSD-1.4.2/vax/binary/sets{50,33}`

(if your disk is less than 200MB you will probably want to exclude the X sets) and then before the 'mt ... rewoffl' run

```
tar cvf /dev/nrmt0 *.tgz
```

When you have booted the bootfs and completed the disk partitioning you will be prompted to 'select medium' for install. At this point you will need to press ^Z (Ctrl+Z) to suspend the install tool, then

```
cd /mnt
mt -f /dev/nrmt0 rewind
mt -f /dev/nrmt0 fsf
tar xvf /dev/nrmt0
fg
```

then select "install from local dir" and give "/mnt". Note: If your disk is small you will need to be careful about filling it up.

If you are using any other OS to create bootable tapes, remember that the blocksize *must* be 512 for the file to be bootable! Otherwise it just won't work.

Creating boot floppies

Fetch the bootable bootfs image from

`.../NetBSD-1.4.2/vax/installation/bootfs/boot.fs.gz`

and gunzip it. It is a 1MB bootable image that will boot from any floppy of size 1MB and bigger. Note that you *cannot* install from RX50 floppies due to the small size. This may change in the future.

All floppies except RX50 use a standardized format for storing data so writing the bootfs to the floppy can be done on any PC. From DOS the preferred way to do this is using RAWRITE.EXE.

Booting from NFS server

All VAXen that can boot over network uses MOP, a DEC protocol. To be able to use MOP, a MOP daemon must be present on one of the machines on the local network. The boot principle is:

- The VAX broadcast a wish to load an image.
- A mopd answers and send the boot program to the VAX.
- The boot program does rarp/bootp requests, mounts the root filesystem and loads the kernel.
- The kernel is loaded and starts executing.

If your machine has a disk and network connection, it may be convenient for you to install NetBSD over the network. This involves temporarily booting your machine over NFS, just long enough so you can initialize its disk. This method requires that you have access to an NFS server on your network so you can configure it to support diskless boot for your machine. Configuring the NFS server is normally a task for a system administrator, and is not trivial.

If you are using a NetBSD system as the boot-server, have a look at the `diskless`(8) manual page for guidelines on how to proceed with this. If the server runs another operating system, consult the documentation that came with it. (I.e. add_client(8) on SunOS.)

There is also very useful documentation at

http://www.netbsd.org/Documentation/network/netboot/

You also must install a MOP loader. If you are booting from another NetBSD machine, the MOP daemons are included in the distribution, otherwise you may have to install a MOP loader. A loader can be found at

ftp.stacken.kth.se:/pub/OS/NetBSD/mopd

Fetch the latest and read the installation instructions.

The file that should be loaded is called boot.mop and is located in

`.../NetBSD-1.4.2/vax/installation/netboot/boot.mop`

The kernel to load is the same kernel as the bootfs uses and can be found in

`.../NetBSD-1.4.2/vax/installation/netboot/netbsd.ram.gz`

From the install program started in the kernel the rest of the system can be installed. There is also a very good (if somewhat out of date) FAQ for netbooting VAXen at

http://world.std.com/~bdc/projects/vaxen/VAX-netboot-HOWTO.html

that describes netbooting of VAXen from many different OS'es.

Preparing your System for NetBSD Installation

VAX machines usually need little or no preparation before installing NetBSD, other than the usual, well advised precaution of BACKING UP ALL DATA on any attached storage devices.

If you are on a Qbus or Unibus system the disk controller(s) need be at the standard CSR addresses for DUA (772150) or DUB (760334) to be recognized by the kernel and boot programs.

If you are installing on a VAXstation you *may* require a serial console.

Installing the NetBSD System

Installation of NetBSD/vax is now easier than ever! For the latest news, problem reports, and discussion, join the port-vax mainlist by mailing a line saying

```
subscribe port-vax
```

to `majordomo@netsbd.org`. Also, see **http://www.netbsd.org** for more information.

If you encounter any problems, please report them via the mailing list or the `send-pr`(1) program so that they can be fixed for the next release.

To install or upgrade NetBSD, you need to first boot the installation program and then interact with the screen-menu program **`sysinst`**. The installation program actually consists of the NetBSD kernel plus an in-memory file system of utility programs.

The usual procedure is to write the installation system to the install media, as described earlier.

Booting from install media

The built-in console monitor understands a bunch of commands, dependent of which VAX you have. To just boot from a device, type 'B' at the '>>>' prompt. Device naming in the console monitor differs a lot from the Unix counterparts. A device looks like "ddcu", where dd is the device type, c is the controller number and u is the device unit. Many console monitors also support the 'SHOW DEV' command, which shows available units to boot from.

A summary of the most common boot devices and their name:

```
B DUA0 - first MSCP controller, unit 0.
B MUA0 - first TMSCP tape controller, unit 0.
B DKB0 - second SCSI bus, unit 0.
B XQA0 - first Q22 bus Ethernet controller.
B ESA0 - first LANCE Ethernet controller on VAXstations.
```

Other devices may be appropriate for your configuration.

Then just proceed with the program sysinst. The install descriptions for sysinst that follows can be easily adopted to vax.

Running the Sysinst Installation Program

1. *Introduction*

   Using **`sysinst`**, installing NetBSD is a relatively easy process. You still should read this document and have it in hand when doing the installation process. This document tries to be a good guideline for the installation and as such covers many details to be completed. Do not let this discourage you, the install program is not hard to use.

2. *Possible* PCMCIA *issues*

   There is a serious bug that may make installation of NetBSD on PCMCIA machines difficult. This bug does not make *use* of PCMCIA difficult once a machine is installed. If you do not have PCMCIA on your machine (PCMCIA is only really used on laptop machines), you can skip this section, and ignore the "[PCMCIA]" notes.

   This will explain how to work around the installation problem.

   What is the bug: The kernel keeps careful track of what interrupts and i/o ports are in use during auto-configuration. It then allows the PCMCIA devices to pick unused interrupts and ports. Unfortunately, not all devices are included in the `INSTALL` kernels in order to save space. Let's say your laptop has a

soundblaster device built in. The `INSTALL` kernel has no sound support. The PCMCIA code might allocate your soundblaster's IRQ and I/O ports to PCMCIA devices, causing them not to work. This is especially bad if one of the devices in question is your ethernet card.

This problem will impact some, but not all, users of PCMCIA. If this bug is affecting you, watch the [PCMCIA] notes that will appear in this document.

3. *General*

The following is a walk-through of the steps you will take while getting NetBSD installed on your hard disk. **`sysinst`** is a menu driven installation system that allows for some freedom in doing the installation. Sometimes, questions will be asked and in many cases the default answer will be displayed in brackets ("[ ]") after the question. If you wish to stop the installation, you may hit Control-C at any time, but if you do, you'll have to begin the installation process again from scratch.

4. *Quick install*

First, let's describe a quick install. The other sections of this document go into the installation procedure in more detail, but you may find that you do not need this. If you want detailed instructions, skip to section 3. This section describes a basic installation, using a CD-ROM install as an example.

- What you need.
  - The distribution sets (in this example, they are on CD).
  - One 1.44M 3.5" floppy.
  - A CD-ROM drive (SCSI or ATAPI), a harddisk and a minimum of of memory installed.
  - The harddisk should have at least + *n* megabytes of space free, where *n* is the number of megabytes of main memory in your system. If you wish to install the X window system as well, you will need at least 60Mb more.
- The Quick Installation
  - Insert the boot floppy you just created. Boot the computer. The main menu will be displayed.
  - If you wish, you can configure some network settings immediately by choosing the **utilities** menu and then **configure network**. It isn't actually required at this point, but it may be more convenient. Go back to the main menu.
  - Choose **install**
  - You will be guided through some steps regarding the setup of your disk, and the selection of distributed components to install. When in doubt, refer to the rest of this document for details.
  - After your disk has been prepared, choose **CD-ROM** as the medium. The default values for the path and device should be ok.
  - After all the files have been unpacked, go back to the main menu and select **reboot**, after you have removed the bootfloppy from the drive.
  - NetBSD will now boot. You should log in as `root`, and set a password for that account. You are also advised to edit the file `/etc/rc.conf` to match your system needs.
  - Your installation is now complete.

– For configuring the X window system, if installed, see the files in
`/usr/X11R6/lib/X11/doc`.
Further information can be found on **http://www.xfree86.org/**

5. *Booting* NetBSD

[PCMCIA] Unplug your PCMCIA devices, so that they won't be found by NetBSD.

Boot your machine using the boot floppy. The boot loader will start, and will print a countdown and begin booting.

If the boot loader messages do not appear in a reasonable amount of time, you either have a bad boot floppy or a hardware problem. Try writing the install floppy image to a different disk, and using that.

It will take a while to load the kernel from the floppy, probably around a minute or so, then, the kernel boot messages will be displayed. This may take a little while also, as NetBSD will be probing your system to discover which hardware devices are installed. The most important thing to know is that `wd0` is NetBSD's name for your first IDE disk, `wd1` the second, etc. `sd0` is your first SCSI disk, `sd1` the second, etc.

Note that, once the system has finished booting, you need not leave the floppy in the disk drive.

Once NetBSD has booted and printed all the boot messages, you will be presented with a welcome message and a main menu. It will also include instructions for using the menus.

6. *Network configuration*

[PCMCIA] You can skip this section, as you will only get data from floppy in the first part of the install.

If you will not use network operation during the installation, but you do want your machine to be configured for networking once it is installed, you should first go to the utilities menu, and select **`Configure network option`**. If you only want to temporarily use networking during the installation, you can specify these parameters later. If you are not using Domain Name Service (DNS), you can give an empty response in reply to answers relating to this.

7. *Installation drive selection and parameters*

To start the installation, select the menu option to install NetBSD from the main menu.

The first thing is to identify the disk on which you want to install NetBSD. **`sysinst`** will report a list of disks it finds and ask you for your selection. Depending on how many disks are found, you may get a different message. You should see disk names like `wd0`, `wd1`, `sd0`, or `sd1`.

Next, depending on whether you are using a wd*x* or wd*x* disk, you will either be asked for the type of disk (wd*x*) you are using or you will be asked if you want to specify a fake geometry for your SCSI disk (sd*x*). The types of disk are be IDE, ST-506 or ESDI. If you're installing on an ST-506 or ESDI drive, you'll be asked if your disk supports automatic sector forwarding. If you are *sure* that it does, reply affirmatively. Otherwise, the install program will automatically reserve space for bad144 tables.

8. *Partitioning the disk.*

- Which portion of the disk to use.

  You will be asked if you want to use the entire disk or only part of the disk. If you decide to use the entire disk for NetBSD, it will be checked if there are already other systems present on the disk, and you will be asked to confirm whether you want to overwrite these.

  If you want to use the entire disk for NetBSD, you can skip the following section and go to *Editing the* NetBSD *disklabel*.

- *Editing the* NetBSD *disklabel*.

  The partition table of the NetBSD part of a disk is called a *disklabel*. There are 3 layouts for the NetBSD part of the disk that you can pick from: **`Standard, Standard with X`** and **`Custom`**. The first two use a set of default values (that you can change) suitable for a normal installation, possibly including X. The last option lets you specify everything yourself.

  You will be presented with the current layout of the NetBSD disklabel, and given a chance to change it. For each partition, you can set the type, offset and size, block and fragment size, and the mount point. The type that NetBSD uses for normal file storage is called **4.2BSD**. A swap partition has a special type called **swap**.

  Some partitions in the disklabel have a fixed purpose. Partition **a** is always the root partition, **b** is the swap partition, and **c** is the whole disk. Partitions **e-h** are available for other use. Traditionally, **d** is the partition mounted on the `/usr` directory, but this is historical practice, not a fixed value.

  You will then be asked to name your disk's disklabel. The default response is **mydisk**. For most purposes this 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.

9. *Preparing your hard disk*

   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. 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 filesystems will be initialized to contain NetBSD bootstrapping binaries and configuration files. You will see messages on your screen from the various NetBSD disk preparation tools that are running. There should be no errors in this section of the installation. If there are, restart from the beginning of the installation process. Otherwise, you can continue the installation program after pressing the return key.

10. *Getting the distribution sets.*

    The NetBSD distribution consists of a number of *sets*, that come in the form of gzipped tarfiles. A few sets must be installed for a working system, others are optional. At this point of the installation, you will be presented with a menu which enables you to choose from one of the following methods of installing the sets. Some of these methods will first load the sets on your hard disk, others will extract the sets directly.

    For all these methods, the first step is making the sets available for extraction, and then do the actual installation. The sets can be made available in a few different ways. The following sections describe each of those methods. After reading the one about the method you will be using, you can continue to section 9

11. *Installation using ftp*

    To be able to install using ftp, you first need to configure your network setup, if you haven't already at the start of the install procedure. **`sysinst`** will do this for you, asking you to provide some data, like IP number, hostname, etc. If you do not have name service set up for the machine that you are installing on, you can just press return in answer to these questions, and DNS will not be used.

    You will also be asked to specify the host that you want to transfer the sets from, the directory on that host, and the account name and password used to log into that host using ftp. If you did not set up DNS when answering the questions to configure networking, you will need to specify an IP number instead of a hostname for the ftp server.

sysinst will proceed to transfer all the default set files from the remote site to your hard disk.

12. *Installation using NFS*

    To be able to install using NFS, you first need to configure your network setup, if you haven't already at the start of the install procedure. **sysinst** will do this for you, asking you to provide some data, like IP number, hostname, etc. If you do not have name service set up for the machine that you are installing on, you can just press return in answer to these questions, and DNS will not be used.

    You will also be asked to specify the host that you want to transfer the sets from, and the directory on that host that the files are in. This directory should be mountable by the machine you are installing on, i.e. correctly exported to your machine.

    If you did not set up DNS when answering the questions to configure networking, you will need to specify an IP number instead of a hostname for the NFS server.

13. *Installation from CD-ROM*

    When installing from a CD-ROM, you will be asked to specify the device name for your CD-ROM player (usually `cd0`), and the directory name on the CD-ROM where the distribution files are.

    sysinst will then check if the files are indeed available in the specified location, and proceed to the actual extraction of the sets.

14. *Installation from an unmounted filesystem*

    In order to install from a local filesystem, you will need to specify the device that the filesystem resides on (for example `wd1e`) the type of the filesystem, and the directory on the specified filesystem where the sets are located. **sysinst** will then check if it can indeed access the sets at that location.

15. *Installation from a local directory*

    This option assumes that you have already done some preparation yourself. The sets should be located in a directory on a filesystem that is already accessible. **sysinst** will ask you for the name of this directory.

16. *Extracting the distribution sets*

    After the install sets containing the NetBSD distribution have been made available, you can either extract all the sets (a full installation), or only extract sets that you have selected. In the latter case you will be shown the currently selected sets, and given the opportunity to select the sets you want. Some sets always need to be installed (**kern, base** and **etc**) they will not be shown in this selection menu.

    Before extraction begins, you can elect to watch the files being extracted; the name of each file that is extracted will be shown. This can slow down the installation process considerably, especially on machines with slow graphics consoles or serial consoles.

    After all the files have been extracted, all the necessary device node files will be created. If you have already configured networking, you will be asked if you want to use this configuration for normal operation. If so, these values will be installed in the network configuration files.

17. *Finalizing your installation.*

    Congratulations, you have successfully installed NetBSD 1.4.2. You can now reboot the machine, and boot from harddisk.

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 propperly configured state, with the most important ones described below.

1. Configuring `/etc/rc.conf`

   If you 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 filesystem mounted read-write. When the system asks you to choose a shell, simply hit 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 hit 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. If your `/usr` directory is on a separate partition and you do not know how to use 'ed' or 'ex', you will have to mount your `/usr` partition to gain access to 'vi'. Do the following:

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

   If you have `/var` on a seperate 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 are new to NetBSD 1.4 and 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. 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.

3. Adding accounts

   Use the `vipw`(8) command to add accounts to your system, *do not* edit `/etc/passwd` directly. See `adduser`(8) for more information on the process of 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 3rd party packages

   There is a lot of software freely available for Unix-based systems, almost all of which can run on NetBSD. Modifications are usually needed to when transferring programs between different Unix-like systems, so the NetBSD packages collection incorporates any such changes necessary to make that software run on NetBSD, and makes the installation (and deinstallation) of the software packages easy. There's also the option of building a package from source, in case there's no precompiled binary available.

   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
   and extracting it into `/usr/pkgsrc`. See `/usr/pkgsrc/README` then for more information.

6. Misc

   - To adjust the system to your local timezone, point the `/etc/localtime` symlink to the appropriate file under `/usr/share/zoneinfo`.
   - Edit `/etc/aliases` to forward root mail to the right place (run `newaliases`(1) afterwards.)
   - The `/etc/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 *filename*
     is likely to give you more information on these files.

Upgrading a previously-installed NetBSD System

There is no upgrade program for NetBSD/vax; you will have to upgrade your system by hand. Here are some hints about how to do it:

- First replace the kernel, /boot and the boot blocks with new ones. The boot system has changed since 1.3.
- Reboot the system so you are running with the new kernel.
- Untar the wanted distributions. Remember to use the **--unlink** flag to `tar`(1), otherwise you will run into trouble.
- Reboot the system once again.
- Now you will (hopefully) be up and running in the new world.

Note You *should not* extract the **etc** set if upgrading. Instead, you should extract that set into another area and carefully merge the changes by hand.

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 1.4.2

Note Only issues effecting an upgrade from NetBSD 1.3 or NetBSD 1.3.x are decribed here.

- "machine" directory/link in "/usr/include"

  Description
  Some architecture may fail to install the **comp** set because the
  `/usr/include/machine`
  directory changed to a symbolic link in NetBSD 1.4.

  Fix If this happens, you can use the command
  # **rm -r /usr/include/machine**
  to remove the old directory and it contents and reinstall the **comp** set.

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)m 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. 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 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, WWW and SUP server.

- Redback Networks, Inc. for hosting the NetBSD Mail server.

- 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 organiztions (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
Soren Jorvang
Scott Kaplan
Noah M. Keiserman
John Kohl
Chris Legrow
Ted Lemon
Neil J. McRae
Perry E. Metzger
Herb Peyerl
Mike Price
Dave Rand
Michael Richardson
Heiko W. Rupp
Brad Salai
Chuck Silvers
Thor Lancelot Simon
Bill Sommerfeld
Paul Southworth
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
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.
VMC Harald Frank, Germany
Warped Communications, Inc.

(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**
Ignatios Souvatzis	is@NetBSD.ORG	**amiga**
Jun-ichiro itojun Hagino	itojun@NetBSD.ORG	**sh3**
Eduardo Horvath	eeh@NetBSD.ORG	**sparc64**
Paul Kranenburg	pk@NetBSD.ORG	**sparc**
Anders Magnusson	ragge@NetBSD.ORG	**vax**
Tsubai Masanari	tsubai@NetBSD.ORG	**macppc**
Tsubai Masanari	tsubai@NetBSD.ORG	**newsmips**
Minoura Makoto	minoura@NetBSD.ORG	**x68k**
Phil Nelson	phil@NetBSD.ORG	**pc532**
Scott Reynolds	scottr@NetBSD.ORG	**mac68k**
Darrin Jewell	dbj@NetBSD.ORG	**next68k**
Kazuki Sakamoto	sakamoto@NetBSD.ORG	**bebox**
Wolfgang Solfrank	ws@NetBSD.ORG	**ofppc**
Jonathan Stone	jonathan@NetBSD.ORG	**pmax**
Shin Takemura	takemura@NetBSD.ORG	**hpcmips**
Jason Thorpe	thorpej@NetBSD.ORG	**hp300**
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 1.4.2 Release Engineering team:

Chris G. Demetriou	cgd@NetBSD.ORG
Havard Eidnes	he@NetBSD.ORG
Ted Lemon	mellon@NetBSD.ORG
Perry Metzger	perry@NetBSD.ORG
Curt Sampson	cjs@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
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
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
Brian C. Grayson	bgrayson@NetBSD.ORG
Brad Grantham	grantham@tenon.com
Matthew Green	mrg@NetBSD.ORG
Juergen Hannken-Illjes	hannken@NetBSD.ORG
Charles M. Hannum	mycroft@NetBSD.ORG
Eric Haszlakiewicz	erh@NetBSD.ORG
HAYAKAWA Koichi	haya@NetBSD.ORG
Rene 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
Dean Huxley	dean@netbsd.org
ITOH Yasufumi	itohy@NetBSD.ORG
Matthew Jacob	mjacob@NetBSD.ORG
Lonhyn T. Jasinskyj	lonhyn@NetBSD.ORG
Chris Jones	cjones@NetBSD.ORG
Soren Jorvang	soren@NetBSD.ORG
Antti Kantee	pooka@NetBSD.ORG
Lawrence Kesteloot	kesteloo@cs.unc.edu
Thomas Klausner	wiz@NetBSD.ORG
Klaus Klein	kleink@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
Brett Lymn	blymn@NetBSD.ORG
Paul Mackerras	paulus@NetBSD.ORG
Dan McMahill	dmcmahill@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
Tohru Nishimura	nisimura@NetBSD.ORG
Masaru Oki	oki@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
Tim Shepard	shep@NetBSD.ORG
Takao Shinohara	shin@NetBSD.ORG
Chuck Silvers	chs@NetBSD.ORG
Thor Lancelot Simon	tls@NetBSD.ORG
Noriyuki Soda	soda@NetBSD.ORG
Bill Sommerfeld	sommerfeld@NetBSD.ORG
Bill Studenmund	wrstuden@NetBSD.ORG
Kevin Sullivan	sullivan@NetBSD.ORG
Kimmo Suominen	kim@NetBSD.ORG
Matt Thomas	matt@NetBSD.ORG
Christoph Toshok	toshok@NetBSD.ORG
Izumi Tsutsui	tsutsui@NetBSD.ORG
UCHIYAMA Yasushi	uch@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
Rob Windsor	windsor@NetBSD.ORG
Dan Winship	danw@NetBSD.ORG
Jim Wise	jwise@NetBSD.ORG
Colin Wood	ender@NetBSD.ORG

Legal Mumbo-Jumbo

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 Computer Systems Engineering Group at Lawrence Berkeley Laboratory.

This product includes software developed by the NetBSD Foundation, Inc. and its contributors.

This product includes software developed by Adam Glass and Charles Hannum.

This product includes software developed by Adam Glass.

This product includes software developed by Berkeley Software Design, Inc.

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 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 Herb Peyerl.

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 Mark Brinicombe.

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 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 Scott Bartram.

This product includes software developed by SigmaSoft, Th. Lockert.

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 TooLs GmbH.

This product includes software developed by Winning Strategies, Inc.

This product includes software developed by the Center for Software Science 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 at Ludd, University of Lule}, Sweden.

This product includes software developed by Gordon W. Ross.

This product includes software developed by Ben Harris.

This product includes software developed by Adam Glass, David Jones, and Gordon Ross.