NAME

INSTALL – Installation procedure for NetBSD/vax.

CONTENTS

About this Document....2
What is NetBSD?....2
Changes Between The NetBSD 2.0.1 and 2.0.2 updates....2
  Kernel....3
  Networking....3
  System administration and user tools....3
  vax specific....3
Changes Between The NetBSD 2.0 release and 2.0.1 update....3
  Kernel....4
  Networking....4
  System administration and user tools....4
  vax specific....4
Changes Between The NetBSD 1.6 and 2.0 Releases....4
  Kernel....4
  Networking....5
  File system....5
  Security....5
  System administration and user tools....5
  Miscellaneous....6
  vax specific....6
Important notes about NetBSD 2.0.2....7
The Future of NetBSD....7
Sources of NetBSD....8
NetBSD 2.0.2 Release Contents....8
  NetBSD/vax subdirectory structure....9
  Binary distribution sets....9
NetBSD/vax System Requirements and Supported Devices....11
  Supported hardware....11
Getting the NetBSD System on to Useful Media....12
  Creating boot tapes....13
  Creating boot floppies....14
  Booting from NFS server....14
Preparing your System for NetBSD installation....15
Installing the NetBSD System....15
  Running the sysinst installation program....16
    Introduction....16
    Possible PCMCIA issues....16
    General....16
    Quick install....17
    Booting NetBSD....18
    Network configuration....18
    Installation drive selection and parameters....18
    Partitioning the disk....18
    Preparing your hard disk....19
    Getting the distribution sets....19
    Installation using ftp....19

Installation using NFS.....19
Installation from CD-ROM.....20
Installation from an unmounted file system.....20
Installation from a local directory.....20
Extracting the distribution sets.....20
Finalizing your installation.....20
Post installation steps.....20
Upgrading a previously-installed NetBSD System.....23
Compatibility Issues With Previous NetBSD Releases.....24
Issues affecting an upgrade from NetBSD 1.6.....24
Using online NetBSD documentation.....24
Administrivia.....25
Thanks go to.....25
We are.....28
Legal Mumbo-Jumbo.....35
The End.....39

DESCRIPTION

About this Document

This document describes the installation procedure for NetBSD 2.0.2 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 UNIX -like operating system derived from the University of California, Berkeley Networking Release 2 (Net/2), 4.4BSD-Lite, and 4.4BSD-Lite2 sources. NetBSD runs on fifty four different system architectures (ports), featuring seventeen machine architectures across fifteen distinct CPU families, and is being ported to more. The NetBSD 2.0.2 release contains complete binary releases for many different system architectures. (A few ports 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 Between The NetBSD 2.0.1 and 2.0.2 updates

The NetBSD 2.0.2 release is the second security/critical update of the NetBSD 2.0 release branch. This represents a selected subset of fixes deemed critical in nature for stability or security reasons. All of these fixes will also appear on future NetBSD releases (e.g. NetBSD 2.1 , etc) from the NetBSD 2.0 release branch as well.

Specific updates are as follows:

Kernel

- Avoid endless loop in F_CLOSEM fnctl which could be abused for a local DOS.
- Avoid possible security issues on amd64, x86 and xen by properly checking range for copyinstr and copyoutstr.
- Make pmap_map() work on sun4 machines so these are usable again.
- Fix a UVM problem that causes hangs when large processes fork.
- Fix pthreads on sun4c hardware by fixing a problem with detecting faults in atomic load/store instructions.
- Address further pthread issues on sparc and sparc64 by defining a new LWP flag which indicates that we're in the process of doing a context switch.
- Make sun3 port functional again by restoring the definition for MAXPHYS as it is now used rather than MAXBSIZE to limit page cache I/O sizes.
- Prevent lockups/panics if the VOP_LOOKUP() call unlocked the parent directory node.
- Prevent a kernel panic on boot with a PX graphics card on DECstations.
- Fix prep port to be functional by ensuring that the OpenPIC register window is mapped during startup.
- Clear freed memory in `cgd`(4) code to avoid possibly security issues.

Networking

- Disabled the oow test in ipf because it is killing valid packets.
- Prevent deadlock/panic on NFS clients after NFS server reboots and caches aren't in sync.
- Fix an NFS panic caused by truncating a file while another client is writing data to it.
- Avoid infinite loops when getting NFS readdir response without any entries or EOF.
- Fix possible remote DOS via IPSec AH packets.

System administration and user tools

- Teach groff about NetBSD versions that aren't on the branches documentation.
- Add description for Solaris 10 dual boot installation.
- Document PTHREAD_CONCURRENCY.
- Fix bug in binutils which broke Firefox under NetBSD-sparc64.
- Address xpm security problems reported in CAN-2005-0605.

vax specific

Changes Between The NetBSD 2.0 release and 2.0.1 update

The NetBSD 2.0.1 release is the first security/critical update of the NetBSD 2.0 release branch. This represents a selected subset of fixes deemed critical in nature for stability or security reasons. All of these fixes will also appear on future NetBSD releases (e.g. NetBSD 2.1 , etc) from the NetBSD 2.0 release branch as well.

Specific updates are as follows:

Kernel

- Prevent panics on powerpc with DIAGNOSTIC kernels and trap handling.
- Make the macppc INSTALL kernel bootable again by moving the load address to the correct location.
- Fix a major issue with sparc64 pmap to prevent crashes under heavy load.
- Fix major performance issue with xen port so idle loop doesn't consume 100% of cpu time.
- Fix xen port so it can reboot cleanly instead of hanging.
- MMU fixes for sh3 based ports to prevent reboots under heavy load.

Networking

- Fix major performance issues with the i82547 Gig-E chip which improves performance with `wm`(4)
- Fix problems with `sk`(4) performance on ASUS A8V motherboards.
- Prevent panics in ipf when receiving IPv6 packets.
- NFS fixes to address incorrect atimes updates and cache coherancy issues.

System administration and user tools

- Fix `swapcontext`(3) for amd64 so that it works correctly.

vax specific

Changes Between The NetBSD 1.6 and 2.0 Releases

The NetBSD 2.0 release provides numerous significant functional enhancements, including support for many new devices, integration of hundreds of bug fixes, new and updated kernel subsystems, and many user-land enhancements. The result of these improvements is a stable operating system fit for production use that rivals most commercially available systems.

It is impossible to completely summarize over two years of development that went into the NetBSD 2.0.2 release. Some highlights include:

Kernel

- Ports to new platforms including: amd64, evbsh5, and xen.
- Native thread support has been added, based on Scheduler Activations. Applications which support native threads can now take full advantage of the high-performance NetBSD POSIX threads implementation.
- Kernel events notification framework - kqueue kqueue provides a stateful and efficient event notification framework. Currently supported events include socket, file, directory, fifo, pipe, tty and device changes, and monitoring of processes and signals. kqueue is supported by all writable filesystems in the NetBSD tree (with the exception of Coda) and all device drivers supporting poll(2).
- Improvements have been made to NetBSD's Linux emulation to support the latest Sun JDK/JRE for Linux. Testing has shown that it now runs as well as it does on Linux natively.
- NetBSD 2.0 enforces non-executable mappings on many platforms. This means that parts of the stack and heap are made non-executable when they are marked writable. This makes exploiting potential buffer overflows harder.
- For the i386 port it now supports SMP and has a new ACPI and power management framework which takes advantage of Intel's ACPI implementation.

- The macppc port also now supports SMP as well as hardware support for newer G4 models has been added.
- SMP support has been added to the sparc port as well.
- New generic i2c framework, supporting bit-bang and "intelligent" controllers.
- `sysctl`(9) was switched from a static binding to a dynamic implementation.
- Add a new driver `satalink`(4) and move SATA support from other controllers into this along with adding support for new controllers.

Networking

- `ipf`(8) has been upgraded to version 4.1.3.
- `tcp`(4) now implements path MTU discovery blackhole detection (i.e. it will turn off path MTU discovery if the connection is losing).
- Socket buffer insertion is now O(C). This can provide a substantial performance boost to some applications which use large socket buffers.
- `wi`(4) has support for Host-AP mode, allowing Intersil Prism2/2.5/3-based boards to be used to make an 802.11 Access Point.
- Support for `ipf`(8) has been added to `bridge`(4) and `brconfig`(8)
- Change port allocation from linked list to a hash table for better performance.

File system

- FreeBSD's UFS2 has been ported to NetBSD. UFS2 is an extension to FFS, adding 64 bit block pointers and support for extended file storage. Among other enhancements, UFS2 allows for file systems larger than 1Terabyte.
- The cryptographic disk driver (cgd) can be used to encrypt disks or partitions, using some strong encryption algorithms, like AES (Rijndael) and Blowfish. cgd can also be configured to encrypt swap.

Security

- The systrace utility has been added to the system. systrace monitors and controls an application access to the system by enforcing access policies for system calls. The systrace utility might be used to trace an untrusted application's access to the system. In addition, it can be used to protect the system from software bugs (such as buffer overflows) by constraining a daemon's access to the system. The privilege elevation feature of systrace can be used to obviate the need to run large, untrusted programs as root when only one or two system calls require the elevated privilege.
- Verified Exec support has been added in this release. Verified Exec verifies a cryptographic hash before allowing execution of binaries and scripts. This can be used to prevent a system from running binaries or scripts which have been illegally modified or installed. In addition, Verified Exec can also be used to limit the use of script interpreters to authorized scripts only and disallow interactive use.

System administration and user tools

- Switched from the GPL version to a non-GPL version of various tools including gzip and awk.

Miscellaneous

- NetBSD 2.0 supports a new toolchain based on gcc 3.3.3 and binutils 2.14. gcc 3.3.1 adds support for a number of CPU targets and greatly improved support for i386 and other targets. The support for new platforms in gcc 3.3.3 has enabled the porting of NetBSD to even more architectures.
- Updates of most third party packages that are shipped in the base system to the following latest stable releases:
  - bind 8.3.7
  - binutils 2.14
  - cvs 1.11.17
  - diffutils 2.8.1
  - file 4.08
  - gcc 3.3.3
  - gdb 5.3
  - grep 2.5.1
  - groff 1.19
  - less 381
  - openssl 0.9.7d
  - postfix 2.0.19
  - sendmail 8.12.11
  - tcpump 3.7.1
  - texinfo 4.6
- Many new packages in the *pkgsrc* system, including the latest open source desktop KDE3, OpenOffice, perl, Apache and many more. A number of new platforms are supported, including Darwin, FreeBSD, IRIX, Linux, OpenBSD and Solaris. Support for various other platforms (among them AIX, BSD/OS and HP-UX) is currently being worked on thanks to our new, portable bootstrap kit which makes it much simpler to port pkgsrc support to new operating systems. At the time of writing, there are over 4500 third party packages available in pkgsrc.

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.

vax specific

This is the sixth major release of NetBSD for the VAX.

Support for the following machines has been added:

- MicroVAX 3100m90
- MicroVAX 3100m95
- VAX 4000/100
- VAX 4000/105A
- VAX 4000/108
- VAX 4000/400
- VAX 4000/500
- VAX 4000/600
- VAX 4000/705A

The following drivers were added:

- Interlan NI1010 ethernet
- DEC DDCMP DMC-11/DMR-11 interface

Important notes about NetBSD 2.0.2

- As noted above, the sendmail third-party program has been upgraded since the last release. The new version of sendmail runs without requiring being set-user-ID to ''root''.

  In order to retain the functionality that a newly installed system can send mail ''out of the box'', the default has changed so that sendmail will now start by default, and listen for host-local connections.

  If this behaviour is not desired, you can either
  1. explicitly set the sendmail variable to ''NO'' in `/etc/rc.conf`,
  2. modify `/etc/mail/submit.cf` to point to another host,
  3. modify `/etc/mailer.conf` to point to something else than sendmail, or
  4. make sendmail set-user-ID to ''root'', remove `/etc/mail/submit.cf`, and set the sendmail_suid variable to ''YES'' in `/etc/rc.conf`.

  The latter method is strongly discouraged.

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 2.0.2 Release Contents

The root directory of the NetBSD 2.0.2 release is organized as follows:

`.../NetBSD-2.0.2/`

`CHANGES`	Changes since earlier NetBSD releases.
`LAST_MINUTE`	Last minute changes.
`MIRRORS`	A list of sites that mirror the NetBSD 2.0.2 distribution.
`README.files`	README describing the distribution's contents.
`TODO`	The NetBSD 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 2.0.2 has a binary distribution.

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.
79 MB gzipped, 367 MB uncompressed

pkgsrc This set contains the "pkgsrc" sources, which contain the infrastructure to build third-party packages.
24 MB gzipped, 200 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.
5 MB gzipped, 20 MB uncompressed

src This set contains all of the base NetBSD 2.0.2 sources which are not in **gnusrc**, **sharesrc**, or **syssrc**.
37 MB gzipped, 176 MB uncompressed

syssrc This set contains the sources to the NetBSD 2.0.2 kernel for all architectures; `config`(8); and `dbsym`(8).
26 MB gzipped, 140 MB uncompressed

xsrc This set contains the sources to the X Window System.
84 MB gzipped, 450 MB uncompressed

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

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

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

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 ATT 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 subdirectory structure

The vax-specific portion of the NetBSD 2.0.2 release is found in the `vax` subdirectory of the distribution: `.../NetBSD-2.0.2/vax/`. It contains the following files and directories:

`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/   vax binary distribution sets; see below.
installation/
        misc/
```

Binary distribution sets

The NetBSD vax binary distribution sets contain the binaries which comprise the NetBSD 2.0.2 release for the vax. The binary distribution sets can be found in the `vax/binary/sets` subdirectory of the NetBSD 2.0.2 distribution tree, and are as follows:

base The NetBSD 2.0.2 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.
15 MB gzipped, 44 MB uncompressed

comp Things needed for compiling programs. This set includes the system include files (`/usr/include`) and the various system libraries (except the shared libraries, which are included as part of the **base** set). This set also includes the manual pages for all of the utilities it contains, as well as the system call and library manual pages.
14 MB gzipped, 54 MB uncompressed

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

1 MB gzipped, 1 MB uncompressed

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

kern-GENERIC
This set contains a NetBSD/vax 2.0.2 `GENERIC` kernel, named `/netbsd`. You *must* install this distribution set.
1 MB gzipped, 3 MB uncompressed

man This set includes all of the manual pages for the binaries and other software contained in the **base** set. Note that it does not include any of the manual pages that are included in the other sets.
8 MB gzipped, 30 MB uncompressed

misc This set includes the (rather large) system dictionaries, the typesettable document set, and other files from `/usr/share`.
3 MB gzipped, 9 MB uncompressed

text This set includes NetBSD's text processing tools, including `groff`(1), all related programs, and their manual pages.
2 MB gzipped, 7 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 4.4.0. 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.
6 MB gzipped, 16 MB uncompressed

xcomp The extra libraries and include files needed to compile X source code.
8 MB gzipped, 29 MB uncompressed

xfont Fonts needed by X.
31 MB gzipped, 39 MB uncompressed

xetc Configuration files for X which could be locally modified.
0.03 MB gzipped, 0.17 MB uncompressed

xserver The X server.
2 MB gzipped, 4 MB uncompressed

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 filenames stored in the sets are relative and therefore the files are extracted *below the current directory*. Therefore, if you want to extract the binaries into your system, i.e. replace the system binaries with them, you have to run the **`tar -xpf`** command from the root directory (`/`) of your system. This utility is used only in a Traditional method installation.

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

NetBSD/vax System Requirements and Supported Devices

Supported hardware

- CPU types
  - VAX 11/750
  - VAX 11/78{0,5}
  - VAX 4000 model 100(A),105A,106A,108,200,300,400,500(A),505A,600(A),700
  - VAX 8200/8250/8300/8350
  - VAX 8600/8650
  - MicroVAX/VAXstation II
  - MicroVAX III
  - MicroVAX 3100 model 10,10e,20,20e,30,40,80,85,90,95
  - MicroVAX 3300/3400
  - MicroVAX 3500/3600
  - MicroVAX 3800/3900
  - MicroVAX 2000/VAXstation 2000
  - VAXstation 3100 model 30,38,40,48,76
  - VAXserver 3100 KA41-B and KA41-E
  - VAXserver 3600
  - VAXserver 3900
  - VAXstation 3200/3500
  - VAXstation 4000 models 60,90,90A,96 and VLC
- Networking
  - DEUNA/DELUA Unibus ethernet
  - DEQNA/DELQA Q22 bus ethernet
  - LANCE chip (where available)
  - SGEC chip (where available)
- Serial lines
  - DHU11/DHV11/DL11/DLV11/DZ11/DZQ11/DZV11 Unibus/Q22 bus asynchronous lines
  - 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
  - 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
  - SCSI tapes on VAXstations.
- Floppy
  - RX33/RX50 on RQDX controllers
  - RX50 on VAX 8200

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

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

Partition	Suggested	Needed
/ (root)	32 MB	16 MB
/usr	110 MB	45 MB
/var	20 MB	5 MB
swap	*2-3 *RAM*	

Anything else is up to you!

Getting the NetBSD System on to Useful Media

Installing on a 'bare' machine requires some bootable device; either a tape, CD-ROM 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.

Note that if you are installing or upgrading from a writable media, the media can be write-protected if you wish. These systems mount a root image from inside the kernel, and will not need to write to the media. If you booted from a floppy, the floppy disk may be removed from the drive after the system has booted.

Installation is supported from several media types, including:

- CD-ROM / DVD
- 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 / DVD Find out where the distribution set files are on the CD-ROM or DVD. Likely locations are `binary/sets` and `vax/binary/sets`.

Proceed to the instruction on installation.

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. If you don't have DHCP available on your network, you will 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 IPv4 address of **ftp.NetBSD.org** is `204.152.184.75` and the IPv6 address is `2001:4f8:4:7:2e0:81ff:fe21:6563` (as of June, 2004).

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 for those 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 you don't have DHCP available on your network and 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 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.

Note: the tape devices with which NetBSD/vax is believed to work is the DEC TK-50. This is a very slow device. Installation via disk or network is recommended if possible.

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-2.0.2
# cd vax/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.

Creating boot tapes

Fetch the bootable bootfs image from

```
.../NetBSD-2.0.2/vax/installation/bootfs/boot.fs
```

and write it on the beginning of the tape. Under NetBSD this is done (for MSCP tape, with SCSI tape the name is `nrst0`) via:

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

Under Ultrix the tape name is different:

```
# 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-2.0.2/vax/binary/sets{50,33}
```

(if your disk is less than 200 MB 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 CONTROL-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-2.0.2/vax/installation/bootfs/boot.fs
```

and **gunzip** it. It is a 1.2MB bootable image that will boot from any floppy of size 1.2MB 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 MS-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 file system 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 and is located in

```
.../NetBSD-2.0.2/vax/installation/netboot/boot
```

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

```
.../NetBSD-2.0.2/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 Q-bus 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 maillist by mailing a line saying

```
subscribe port-vax
```

to majordomo@NetBSD.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.

B EZA0 first SGEC 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 for the sake of completeness. Do not let this discourage you; the install program is not hard to use.

2. *Possible* PCMCIA *issues*

   Machines with PCMCIA slots may have problems during installation. 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. If you do have PCMCIA in your machine, you can safely ignore this section and the "[PCMCIA]" the first time, as you are likely to not have problems. Should troubles occur during floppy boot, they may be PCMCIA specific. You should then re-read this section and try again, following the instructions in the "[PCMCIA]" notes.

   This section explains how to work around the installation problem.

   The kernel keeps careful track of what interrupts and I/O ports are in use during autoconfiguration. It then allows the PCMCIA devices to pick unused interrupts and I/O ports. Unfortunately, the INSTALL kernel may not detect all devices in your system. This may be because the INSTALL kernel only supports the minimum set of devices to install NetBSD on your system, or it may be that NetBSD does not have support for the device causing the conflict.

   For example, suppose 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, or to lock up the system. This is especially bad if one of the devices in question is your ethernet card.

   The kernel attempts to probe for available interrupts that are usable by the PCIC (PCMCIA interrupt controller), which should alleviate interrupt conflicts; however, I/O port conflicts are still possible.

   This problem will impact some, but not all, users of PCMCIA. If this problem 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 press CONTROL-C at

any time, but if you do, you'll have to begin the installation process again from scratch by running the `/sysinst` program from the command prompt. It is not necessary to reboot.

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 the next section. 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 1.44 MB 3.5" floppy.
     - A CD-ROM drive (SCSI), a hard disk and a minimum of of memory installed.
     - The hard disk 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 120 MB more.
   - The Quick Installation
     - Insert the first boot floppy you just created. Boot the computer. After language selection, the main menu will be displayed.

```
.*********************************************.
* NetBSD-2.0 Install System                    *
*                                              *
*a: Install NetBSD to hard disk                *
* b: Upgrade NetBSD on a hard disk             *
* c: Re-install sets or install additional sets *
* d: Reboot the computer                       *
* e: Utility menu                              *
* x: Exit Install System                       *
.*********************************************.
```

     - If you wish, you can configure some network settings immediately by choosing the **Utility 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. If you haven't already done so in **`sysinst`**, 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. 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 the NetBSD 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 **Utility menu**, and select the **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 the Domain Name System (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 **Install NetBSD to hard disk** 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 `sd0` or `sd1`.

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.

9. *Editing the NetBSD disklabel*

   The partition table of the NetBSD part of a disk is called a *disklabel*. There are 4 layouts for the NetBSD part of the disk that you can pick from: **`Standard, Standard with X, Custom`** and **`Use Existing`**. The first two use a set of default values (that you can change) suitable for a normal installation, possibly including X. With the **`Custom`** option you can specify everything yourself. The last option uses the partition info already present on the disk.

   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.

`a` Root partition (`/`)

`b` Swap partition.

`c` The NetBSD portion of the disk.

`d`–`h` Available for other use. Traditionally, `d` is the partition mounted on `/usr`, but this is historical practice and not a fixed value.

You will then be asked to name your disk's disklabel. The default response will be ok for most purposes. 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.

10. *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 file systems 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.

11. *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 the section labeled 'Extracting the distribution sets'.

12. *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 if you want to use DHCP, and if not to provide data like IP address, 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, the account name and password used to log into that host using ftp, and optionally a proxy server to use. If you did not set up DNS when answering the questions to configure networking, you will need to specify an IP address 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.

13. *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 if you want to use DHCP, and if not to provide data like IP address, 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 address instead of a hostname for the NFS server.

14. *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.

15. *Installation from an unmounted file system*

    In order to install from a local file system, you will need to specify the device that the file system resides on (for example `sd1e`) the type of the file system, and the directory on the specified file system where the sets are located. **sysinst** will then check if it can indeed access the sets at that location.

16. *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 file system that is already accessible. **sysinst** will ask you for the name of this directory.

17. *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. The next menu will allow you to select the time zone that you're in, to make sure your clock has the right offset from UTC. Finally you will be asked to select a password encryption algorithm and can than set a password for the "root" account, to prevent the machine coming up without access restrictions.

18. *Finalizing your installation*

    Congratulations, you have successfully installed NetBSD 2.0.2. You can now reboot the machine, and boot NetBSD from hard disk.

Post installation steps

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

1. Configuring `/etc/rc.conf`

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

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

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

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

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

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

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

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

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

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

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

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

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

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

To enable proper hostname resolution, you will also want to add an /etc/resolv.conf file or (if you are feeling a little more adventurous) run named(8). See resolv.conf(5) or named(8) for more information. Instead of manually configuring network and naming service, DHCP can be used by setting dhclient=YES in /etc/rc.conf.

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

2. Logging in

After reboot, you can log in as root at the login prompt. Unless you've set a password in **sysinst**, there is no initial password. If you're using the machine in a networked environment, you should create an account for yourself (see below) and protect it and the "root" account with good passwords. By default, root login from the network is disabled (even via ssh(1)). One way to become root over the network is to log in as a different user that belongs to group "wheel" (see group(5)) and use su(1) to become root.

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 `vipw`(8) and `pwd_mkdb`(8) if you want to edit the password database.

4. The X Window System

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

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

5. Installing third party packages

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

   - More information on the package system is at
     http://www.NetBSD.org/Documentation/software/packages.html
   - A list of available packages suitable for browsing is at
     ftp://ftp.NetBSD.org/pub/NetBSD/packages/pkgsrc/README.html
   - Precompiled binaries can be found at ,:
     ftp://ftp.NetBSD.org/pub/NetBSD/packages/
     usually in the `2.0.2/vax/All` subdir. You can install them with the following commands under `sh`(1):

     ```
     # PKG_PATH=ftp://ftp.NetBSD.org/pub/NetBSD/packages/2.0.2/vax/All
     # export PKG_PATH
     # pkg_add -v tcsh
     # pkg_add -v bash
     # pkg_add -v perl
     # pkg_add -v apache
     # pkg_add -v kde
     # pkg_add -v mozilla
     ...
     ```

     If you are using `csh`(1) then replace the first two lines with the following:

     ```
     # setenv PKG_PATH ftp://ftp.NetBSD.org/pub/NetBSD/packages/2.0.2/vax/All
     ...
     ```

     The above commands will install the Tenex-csh and Bourne Again shell, the perl programming language , Apache web server, KDE desktop environment and the Mozilla web browser as well as all the packages they depend on.

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

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

After extracting, see the `README` and `Packages.txt` files in the extraction directory (e.g. `/usr/pkgsrc/README`) for more information.

6. Misc

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

     ```
     # man 5 filename
     ```

     is likely to give you more information on these files.

Upgrading a previously-installed NetBSD System

The upgrade to NetBSD 2.0.2 is a binary upgrade; it can be quite difficult to update the system from an earlier version by recompiling from source, primarily due to interdependencies in the various components.

To do the upgrade, you must have the boot floppy available. You must also have at least the **base** and **kern** binary distribution sets available, so that you can upgrade with them, using one of the upgrade methods described above. Finally, you must have sufficient disk space available to install the new binaries. Since files already installed on the system are overwritten in place, you only need additional free space for files which weren't previously installed or to account for growth of the sets between releases. If you have a few megabytes free on each of your root (`/`) and `/usr` partitions, you should have enough space.

Since upgrading involves replacing the kernel, the boot blocks on your NetBSD partition, and most of the system binaries, it has the potential to cause data loss. You are strongly advised to *back up* any important data on the NetBSD partition or on another operating system's partition on your disk before beginning the upgrade process.

The upgrade procedure using the **`sysinst`** tool is similar to an installation, but without the hard disk partitioning. **`sysinst`** will attempt to merge the settings stored in your `/etc` directory with the new version of NetBSD. Getting the binary sets is done in the same manner as the installation procedure; refer to the installation part of the document for how to do this. Also, some sanity checks are done, i.e. file systems are checked before unpacking the sets.

After a new kernel has been copied to your hard disk, your machine is a complete NetBSD 2.0.2 system. However, that doesn't mean that you're finished with the upgrade process. You will probably want to update the set of device nodes you have in `/dev`. If you've changed the contents of `/dev` by hand, you will need to be careful about this, but if not, you can just cd into `/dev`, and run the command:

```
# sh MAKEDEV all
```

Finally, you will want to delete old binaries that were part of the version of NetBSD that you upgraded from and have since been removed from the NetBSD distribution.

NetBSD/vax has switched its executable format from the old a.out format to ELF, the now more commonly used and supported format. Your old binaries will continue to work just fine. The installation procedure will try to take the necessary steps to accomplish this. The most important step is to move the old a.out shared libraries in `/usr/lib` and `/usr/X11R6/lib` (if X was installed) to `/emul/aout`, where they will be automatically found if an older a.out binary is executed. **`sysinst`** will use an existing `/emul` and / or `/emul/aout` directory if available, and will create it (as a symbolic link to `/usr/aout`) if necessary.

If you already had a /emul directory, or a symbolic link by that name, **sysinst** should rename it and tell you about it.

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 2.0.2.

Issues affecting an upgrade from NetBSD 1.6

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

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

Issues fixed by *postinstall*:

- Various files in /etc need upgrading. These include:
  - /etc/defaults/*
  - /etc/mtree/*
  - /etc/daily
  - /etc/weekly
  - /etc/monthly
  - /etc/security
  - /etc/rc.subr
  - /etc/rc
  - /etc/rc.shutdown
  - /etc/rc.d/*
- The following files are now obsolete: /etc/rc.d/fsck.sh /etc/rc.d/gated /etc/rc.d/kerberos /etc/rc.d/NETWORK /etc/rc.d/systemfs /etc/rc.d/xntpd and /etc/rc.d/ypset.
- The users and groups 'smmsp', needs to be created, 'news' is no longer part of the system.

The following issues need to be resolved manually:

- postfix(8) configuration files require upgrading.

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

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.

- The Internet Systems Consortium, Inc. for hosting the NetBSD FTP, CVS, AnonCVS, mail, mail archive, GNATS, SUP, Rsync and WWW servers.
- The Internet Research Institute in Japan for hosting the server which runs the CVSweb interface to the NetBSD source tree.
- The Helsinki University of Technology in Finland for hosting the NetBSD backup CVS and backup server.
- SSH Communications Security in Finland for operating the backup server.
- 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.
- 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:

AboveNet Communications, Inc.
Advanced System Products, Inc.
Alex Poylisher
Alistair Crooks
Andrew Brown
Atsushi YOKOYAMA
Avalon Computer Systems
Bay Area Internet Solutions
Ben Collver
Bill Coldwell
Bill Sommerfeld
Brad Salai
Brains Corporation, Japan
Brian Carlstrom
Brian McGroarty
Canada Connect Corporation
Castor Fu
Central Iowa (Model) Railroad
Charles Conn
Charles D. Cranor
Charles M. Hannum
Chris Legrow
Christer O. Andersson
Christopher g. Demetriou
Christos Zoulas
Chuck Silvers
Co-operative Research Centre for Enterprise Distributed
Curt Sampson
Dave Burgess
Dave Rand
David Brownlee
Demon Internet, UK
Derek Fellion
Digital Equipment Corporation
Distributed Processing Technology
Douglas J. Trainor

Easynet, UK
Ed Braaten
Edward Richley
Eric and Rosemary Spahr
Free Hardware Foundation
Greg Gingerich
Guenther Grau
Harald Koerfgen
Harry McDonald
Heiko W. Rupp
Herb Peyerl
Hubert Feyrer
Innovation Development Enterprises of America
Internet Software Consortium
James Chacon
Jan Joris Vereijken
Jason Birnschein
Jason Brazile
Jason R. Thorpe
Jim Wise
John Kohl
Jonathan P. Kay
Jordan K. Hubbard
Kenneth Alan Hornstein
Kevin Keith Woo
Kimmo Suominen
Krister Waldfridsson
Lex Wennmacher
LinuxFest Northwest
Luke Mewburn
MS Macro System GmbH, Germany
Mark Brinicombe
Mark S. Thomas
Mason Loring Bliss
Mattias Karlsson
Michael Graff
Michael L. Hitch
Michael Richardson
Michael Thompson
Michael W. James
Mike Price
Neil J. McRae
Noah M. Keiserman
Norman R. McBride
Numerical Aerospace Simulation Facility, NASA Ames Research
Oliver Cahagne
Perry E. Metzger
Petri T. Koistinen
Piermont Information Systems Inc.
Precedence Technologies Ltd
Ralph Campbell
Reinoud Zandijk

Richard Nelson
Rob Windsor
Ross Harvey
SDF Public Access Unix, Inc. 501(c)(7)
Salient Systems Inc.
Scott Ellis
Scott Kaplan
Simon Burge
Soren Jacobsen
Soren Jorvang
Steve Allen
Steve Wadlow
SunROOT# Project
Ted Lemon
Ted Spradley
Thor Lancelot Simon
Tim Law
Tom Coulter
Toru Nishimura
VMC Harald Frank, Germany
Warped Communications, Inc.
Wasabi Systems
Whitecross Database Systems Ltd.
William Gnadt
Worria Web Hosting

(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:

Allen Briggs	`briggs@NetBSD.org`
Frank van der Linden	`fvdl@NetBSD.org`
Luke Mewburn	`lukem@NetBSD.org`
Matt Thomas	`matt@NetBSD.org`
Christos Zoulas	`christos@NetBSD.org`

The portmasters (and their ports):

Allen Briggs	briggs@NetBSD.org	**sandpoint**
Anders Magnusson	ragge@NetBSD.org	**vax**
Andrey Petrov	petrov@NetBSD.org	**sparc64**
Ben Harris	bjh21@NetBSD.org	**acorn26**
Chris Gilbert	chris@NetBSD.org	**cats**
Christian Limpach	cl@NetBSD.org	**xen**
Eduardo Horvath	eeh@NetBSD.org	**evbppc**
Frank van der Linden	fvdl@NetBSD.org	**amd64**
Frank van der Linden	fvdl@NetBSD.org	**i386**
IWAMOTO Toshihiro	toshii@NetBSD.org	**hpcarm**
Ichiro Fukuhara	ichiro@NetBSD.org	**hpcarm**
Ignatios Souvatzis	is@NetBSD.org	**amiga**
Izumi Tsutsui	tsutsui@NetBSD.org	**hp300**
Izumi Tsutsui	tsutsui@NetBSD.org	**news68k**
Jason Thorpe	thorpej@NetBSD.org	**algor**
Jason Thorpe	thorpej@NetBSD.org	**shark**
Jeremy Cooper	jeremy@NetBSD.org	**sun3**
Jonathan Stone	jonathan@NetBSD.org	**pmax**
Julian Coleman	jdc@NetBSD.org	**atari**
Jun-ichiro itojun Hagino	itojun@NetBSD.org	**evbsh3**
Jun-ichiro itojun Hagino	itojun@NetBSD.org	**mmeye**
Kazuki Sakamoto	sakamoto@NetBSD.org	**bebox**
Lennart Augustsson	augustss@NetBSD.org	**pmppc**
Marcus Comstedt	marcus@NetBSD.org	**dreamcast**
Martin Husemann	martin@NetBSD.org	**sparc64**
Matt DeBergalis	deberg@NetBSD.org	**next68k**
Matt Fredette	fredette@NetBSD.org	**hp700**
Matt Fredette	fredette@NetBSD.org	**sun2**
Matt Thomas	matt@NetBSD.org	**alpha**
Matt Thomas	matt@NetBSD.org	**netwinder**
Matthias Drochner	drochner@NetBSD.org	**cesfic**
NISHIMURA Takeshi	nsmrtks@NetBSD.org	**x68k**
NONAKA Kimihiro	nonaka@NetBSD.org	**prep**
Nathan Williams	nathanw@NetBSD.org	**sun3**
Noriyuki Soda	soda@NetBSD.org	**arc**
Paul Kranenburg	pk@NetBSD.org	**sparc**
Phil Nelson	phil@NetBSD.org	**pc532**
Reinoud Zandijk	reinoud@NetBSD.org	**acorn32**
Ross Harvey	ross@NetBSD.org	**alpha**
S(/oren Jørvang	soren@NetBSD.org	**cobalt**
S(/oren Jørvang	soren@NetBSD.org	**sgimips**
Scott Reynolds	scottr@NetBSD.org	**mac68k**
Shin Takemura	takemura@NetBSD.org	**hpcmips**
Simon Burge	simonb@NetBSD.org	**evbmips**
Simon Burge	simonb@NetBSD.org	**evbppc**
Simon Burge	simonb@NetBSD.org	**pmax**
Simon Burge	simonb@NetBSD.org	**sbmips**

Steve Woodford	scw@NetBSD.org	**evbsh5**
Steve Woodford	scw@NetBSD.org	**mvme68k**
Steve Woodford	scw@NetBSD.org	**mvmeppc**
Tohru Nishimura	nisimura@NetBSD.org	**luna68k**
Tsubai Masanari	tsubai@NetBSD.org	**macppc**
Tsubai Masanari	tsubai@NetBSD.org	**newsmips**
UCHIYAMA Yasushi	uch@NetBSD.org	**hpcsh**
UCHIYAMA Yasushi	uch@NetBSD.org	**playstation2**
Wayne Knowles	wdk@NetBSD.org	**mipsco**
Wolfgang Solfrank	ws@NetBSD.org	**ofppc**

The NetBSD 2.0.2 Release Engineering team:

Grant Beattie	grant@NetBSD.org
Erik Berls	cyber@NetBSD.org
James Chacon	jmc@NetBSD.org
Julian Coleman	jdc@NetBSD.org
Håvard Eidnes	he@NetBSD.org
Jun-ichiro itojun Hagino	itojun@NetBSD.org
SAITOH Masanobu	msaitoh@NetBSD.org
Luke Mewburn	lukem@NetBSD.org
Matthias Scheler	tron@NetBSD.org
Curt Sampson	cjs@NetBSD.org
Jim Wise	jwise@NetBSD.org

NetBSD Developers:

Nathan Ahlstrom	nra@NetBSD.org
Steve Allen	wormey@NetBSD.org
Jukka Andberg	jandberg@NetBSD.org
Julian Assange	proff@NetBSD.org
Lennart Augustsson	augustss@NetBSD.org
Christoph Badura	bad@NetBSD.org
Bang Jun-Young	junyoung@NetBSD.org
Dieter Baron	dillo@NetBSD.org
Robert V. Baron	rvb@NetBSD.org
Grant Beattie	grant@NetBSD.org
Jason Beegan	jtb@NetBSD.org
Erik Berls	cyber@NetBSD.org
Hiroyuki Bessho	bsh@NetBSD.org
John Birrell	jb@NetBSD.org
Mason Loring Bliss	mason@NetBSD.org
Charles Blundell	cb@NetBSD.org
Rafal Boni	rafal@NetBSD.org
Manuel Bouyer	bouyer@NetBSD.org
John Brezak	brezak@NetBSD.org
Allen Briggs	briggs@NetBSD.org
Mark Brinicombe	mark@NetBSD.org
Aaron Brown	abrown@NetBSD.org

Andrew Brown	atatat@NetBSD.org
David Brownlee	abs@NetBSD.org
Frederick Bruckman	fredb@NetBSD.org
Jon Buller	jonb@NetBSD.org
Simon Burge	simonb@NetBSD.org
Robert Byrnes	byrnes@NetBSD.org
D'Arcy J.M. Cain	darcy@NetBSD.org
Dave Carrel	carrel@NetBSD.org
Daniel Carosone	dan@NetBSD.org
James Chacon	jmc@NetBSD.org
Bill Coldwell	billc@NetBSD.org
Julian Coleman	jdc@NetBSD.org
Ben Collver	ben@NetBSD.org
Jeremy Cooper	jeremy@NetBSD.org
Chuck Cranor	chuck@NetBSD.org
Alistair Crooks	agc@NetBSD.org
Aidan Cully	aidan@NetBSD.org
Johan Danielsson	joda@NetBSD.org
John Darrow	jdarrow@NetBSD.org
Matt DeBergalis	deberg@NetBSD.org
Rob Deker	deker@NetBSD.org
Chris G. Demetriou	cgd@NetBSD.org
Tracy Di Marco White	gendalia@NetBSD.org
Jaromír Dolecek	jdolecek@NetBSD.org
Andy Doran	ad@NetBSD.org
Roland Dowdeswell	elric@NetBSD.org
Emmanuel Dreyfus	manu@NetBSD.org
Matthias Drochner	drochner@NetBSD.org
Jun Ebihara	jun@NetBSD.org
Håvard Eidnes	he@NetBSD.org
Stoned Elipot	seb@NetBSD.org
Enami Tsugutomo	enami@NetBSD.org
Bernd Ernesti	veego@NetBSD.org
Erik Fair	fair@NetBSD.org
Gavan Fantom	gavan@NetBSD.org
Hubert Feyrer	hubertf@NetBSD.org
Jason R. Fink	jrf@NetBSD.org
Matt Fredette	fredette@NetBSD.org
Thorsten Frueauf	frueauf@NetBSD.org
Castor Fu	castor@NetBSD.org
Ichiro Fukuhara	ichiro@NetBSD.org
Quentin Garnier	cube@NetBSD.org
Thomas Gerner	thomas@NetBSD.org
Simon J. Gerraty	sjg@NetBSD.org
Justin Gibbs	gibbs@NetBSD.org
Chris Gilbert	chris@NetBSD.org
Eric Gillespie	epg@NetBSD.org
Adam Glass	glass@NetBSD.org
Michael Graff	explorer@NetBSD.org

Brian C. Grayson	bgrayson@NetBSD.org
Matthew Green	mrg@NetBSD.org
Andreas Gustafsson	gson@NetBSD.org
Jun-ichiro itojun Hagino	itojun@NetBSD.org
Juergen Hannken-Illjes	hannken@NetBSD.org
Charles M. Hannum	mycroft@NetBSD.org
Ben Harris	bjh21@NetBSD.org
Ross Harvey	ross@NetBSD.org
Eric Haszlakiewicz	erh@NetBSD.org
John Hawkinson	jhawk@NetBSD.org
HAYAKAWA Koichi	haya@NetBSD.org
John Heasley	heas@NetBSD.org
René Hexel	rh@NetBSD.org
Michael L. Hitch	mhitch@NetBSD.org
Christian E. Hopps	chopps@NetBSD.org
Ken Hornstein	kenh@NetBSD.org
Marc Horowitz	marc@NetBSD.org
Eduardo Horvath	eeh@NetBSD.org
Nick Hudson	skrll@NetBSD.org
Shell Hung	shell@NetBSD.org
Martin Husemann	martin@NetBSD.org
Dean Huxley	dean@NetBSD.org
Love Hörnquist Åstrand	lha@NetBSD.org
Bernardo Innocenti	bernie@NetBSD.org
Tetsuya Isaki	isaki@NetBSD.org
ITOH Yasufumi	itohy@NetBSD.org
IWAMOTO Toshihiro	toshii@NetBSD.org
Matthew Jacob	mjacob@NetBSD.org
Soren Jacobsen	snj@NetBSD.org
Lonhyn T. Jasinskyj	lonhyn@NetBSD.org
Darrin Jewell	dbj@NetBSD.org
Chris Jones	cjones@NetBSD.org
Søren Jørvang	soren@NetBSD.org
Takahiro Kambe	taca@NetBSD.org
Antti Kantee	pooka@NetBSD.org
Masanori Kanaoka	kanaoka@NetBSD.org
Mattias Karlsson	keihan@NetBSD.org
KAWAMOTO Yosihisa	kawamoto@NetBSD.org
Mario Kemper	magick@NetBSD.org
Min Sik Kim	minskim@NetBSD.org
Thomas Klausner	wiz@NetBSD.org
Klaus Klein	kleink@NetBSD.org
Wayne Knowles	wdk@NetBSD.org
Takayoshi Kochi	kochi@NetBSD.org
John Kohl	jtk@NetBSD.org
Daniel de Kok	daniel@NetBSD.org
Paul Kranenburg	pk@NetBSD.org
Martti Kuparinen	martti@NetBSD.org
Kevin Lahey	kml@NetBSD.org

Johnny C. Lam	jlam@NetBSD.org
Martin J. Laubach	mjl@NetBSD.org
Greg Lehey	grog@NetBSD.org
Ted Lemon	mellon@NetBSD.org
Christian Limpach	cl@NetBSD.org
Frank van der Linden	fvdl@NetBSD.org
Joel Lindholm	joel@NetBSD.org
Mike Long	mikel@NetBSD.org
Warner Losh	imp@NetBSD.org
Tomasz Luchowski	zuntum@NetBSD.org
Federico Lupi	federico@NetBSD.org
Brett Lymn	blymn@NetBSD.org
Paul Mackerras	paulus@NetBSD.org
Anders Magnusson	ragge@NetBSD.org
MAEKAWA Masahide	gehenna@NetBSD.org
David Maxwell	david@NetBSD.org
Dan McMahill	dmcmahill@NetBSD.org
Gregory McGarry	gmcgarry@NetBSD.org
Jared D. McNeill	jmcneill@NetBSD.org
Neil J. McRae	neil@NetBSD.org
Perry Metzger	perry@NetBSD.org
Simas Mockevicius	symka@NetBSD.org
Juan Romero Pardines	xtraeme@NetBSD.org
Julio M. Merino Vidal	jmmv@NetBSD.org
Minoura Makoto	minoura@NetBSD.org
Luke Mewburn	lukem@NetBSD.org
der Mouse	mouse@NetBSD.org
Joseph Myers	jsm@NetBSD.org
Ken Nakata	kenn@NetBSD.org
Takeshi Nakayama	nakayama@NetBSD.org
Phil Nelson	phil@NetBSD.org
Bob Nestor	rnestor@NetBSD.org
NISHIMURA Takeshi	nsmrtks@NetBSD.org
Tohru Nishimura	nisimura@NetBSD.org
NONAKA Kimihiro	nonaka@NetBSD.org
Jesse Off	joff@NetBSD.org
Tatoku Ogaito	tacha@NetBSD.org
OKANO Takayoshi	kano@NetBSD.org
Masaru Oki	oki@NetBSD.org
Atsushi Onoe	onoe@NetBSD.org
Greg Oster	oster@NetBSD.org
Jonathan Perkin	sketch@NetBSD.org
Herb Peyerl	hpeyerl@NetBSD.org
Matthias Pfaller	matthias@NetBSD.org
Chris Pinnock	cjep@NetBSD.org
Adrian Portelli	adrianp@NetBSD.org
Dante Profeta	dante@NetBSD.org
Chris Provenzano	proven@NetBSD.org
Niels Provos	provos@NetBSD.org

Michael Rauch	mrauch@NetBSD.org
Marc Recht	recht@NetBSD.org
Darren Reed	darrenr@NetBSD.org
Jeremy C. Reed	reed@NetBSD.org
Tyler R. Retzlaff	rtr@NetBSD.org
Scott Reynolds	scottr@NetBSD.org
Michael Richardson	mcr@NetBSD.org
Tim Rightnour	garbled@NetBSD.org
Gordon Ross	gwr@NetBSD.org
Steve Rumble	rumble@NetBSD.org
Ilpo Ruotsalainen	lonewolf@NetBSD.org
Heiko W. Rupp	hwr@NetBSD.org
David Sainty	dsainty@NetBSD.org
SAITOH Masanobu	msaitoh@NetBSD.org
Kazuki Sakamoto	sakamoto@NetBSD.org
Curt Sampson	cjs@NetBSD.org
Wilfredo Sanchez	wsanchez@NetBSD.org
Ty Sarna	tsarna@NetBSD.org
SATO Kazumi	sato@NetBSD.org
Jan Schaumann	jschauma@NetBSD.org
Matthias Scheler	tron@NetBSD.org
Karl Schilke (rAT)	rat@NetBSD.org
Amitai Schlair	schmonz@NetBSD.org
Konrad Schroder	perseant@NetBSD.org
Lubomir Sedlacik	salo@NetBSD.org
Christopher SEKIYA	sekiya@NetBSD.org
Reed Shadgett	dent@NetBSD.org
John Shannon	shannonjr@NetBSD.org
Tim Shepard	shep@NetBSD.org
Takeshi Shibagaki	shiba@NetBSD.org
Naoto Shimazaki	igy@NetBSD.org
Takao Shinohara	shin@NetBSD.org
Takuya SHIOZAKI	tshiozak@NetBSD.org
Chuck Silvers	chs@NetBSD.org
Thor Lancelot Simon	tls@NetBSD.org
Jeff Smith	jeffs@NetBSD.org
Noriyuki Soda	soda@NetBSD.org
Wolfgang Solfrank	ws@NetBSD.org
SOMEYA Yoshihiko	someya@NetBSD.org
Bill Sommerfeld	sommerfeld@NetBSD.org
Ignatios Souvatzis	is@NetBSD.org
Bill Squier	groo@NetBSD.org
Jonathan Stone	jonathan@NetBSD.org
Bill Studenmund	wrstuden@NetBSD.org
Kevin Sullivan	sullivan@NetBSD.org
SUNAGAWA Keiki	kei@NetBSD.org
Kimmo Suominen	kim@NetBSD.org
Shin Takemura	takemura@NetBSD.org
TAMURA Kent	kent@NetBSD.org

Shin'ichiro TAYA	taya@NetBSD.org
Ian Lance Taylor	ian@NetBSD.org
Matt Thomas	matt@NetBSD.org
Jason Thorpe	thorpej@NetBSD.org
Christoph Toshok	toshok@NetBSD.org
Tsubai Masanari	tsubai@NetBSD.org
Izumi Tsutsui	tsutsui@NetBSD.org
UCHIYAMA Yasushi	uch@NetBSD.org
Masao Uebayashi	uebayasi@NetBSD.org
Shuichiro URATA	ur@NetBSD.org
Todd Vierling	tv@NetBSD.org
Aymeric Vincent	aymeric@NetBSD.org
Paul Vixie	vixie@NetBSD.org
Krister Walfridsson	kristerw@NetBSD.org
Lex Wennmacher	wennmach@NetBSD.org
Leo Weppelman	leo@NetBSD.org
Assar Westerlund	assar@NetBSD.org
Todd Whitesel	toddpw@NetBSD.org
Nathan Williams	nathanw@NetBSD.org
Rob Windsor	windsor@NetBSD.org
Dan Winship	danw@NetBSD.org
Jim Wise	jwise@NetBSD.org
Michael Wolfson	mbw@NetBSD.org
Steve Woodford	scw@NetBSD.org
Colin Wood	ender@NetBSD.org
YAMAMOTO Takashi	yamt@NetBSD.org
Yuji Yamano	yyamano@NetBSD.org
Reinoud Zandijk	reinoud@NetBSD.org
Maria Zevenhoven	maria7@NetBSD.org
Christos Zoulas	christos@NetBSD.org

Other contributors:

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

Legal Mumbo-Jumbo

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

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

This product includes software developed by the University of California, Berkeley and its contributors.
This product includes software developed by the NetBSD Foundation.
This product includes software developed by The NetBSD Foundation, Inc. and its contributors.
This product includes software developed for the NetBSD Project. See http://www.netbsd.org/ for information about NetBSD.
This product contains software developed by Ignatios Souvatzis for the NetBSD project.
This product contains software written by Ignatios Souvatzis and Michael L. Hitch for the NetBSD project.
This product contains software written by Michael L. Hitch for the NetBSD project.
This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)

This product includes cryptographic software written by Eric Young (eay@mincom.oz.au)
This product includes software designed by William Allen Simpson.
This product includes software developed at Ludd, University of Lulea, Sweden and its contributors.
This product includes software developed at Ludd, University of Lulea.
This product includes software developed at the Information Technology Division, US Naval Research Laboratory.
This product includes software developed by Berkeley Software Design, Inc.
This product includes software developed by David Jones and Gordon Ross
This product includes software developed by Gordon W. Ross and Leo Weppelman.
This product includes software developed by Hellmuth Michaelis and Joerg Wunsch
This product includes software developed by Internet Research Institute, Inc.
This product includes software developed by Leo Weppelman and Waldi Ravens.
This product includes software developed by Mika Kortelainen
This product includes software developed by Aaron Brown and Harvard University.
This product includes software developed by Adam Ciarcinski for the NetBSD project.
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 Advanced Risc Machines Ltd.
This product includes software developed by Alex Zepeda, and Colin Wood for the NetBSD Projet.
This product includes software developed by Alex Zepeda.
This product includes software developed by Alistair G. Crooks.
This product includes software developed by Alistair G. Crooks. for the NetBSD project.
This product includes software developed by Allen Briggs
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 Berkeley Software Design, Inc.
This product includes software developed by Bill Paul.
This product includes software developed by Bodo Moeller. (If available, substitute umlauted o for oe)
This product includes software developed by Boris Popov.
This product includes software developed by Brad Pepers
This product includes software developed by Bradley A. Grantham.
This product includes software developed by Brini.
This product includes software developed by Causality Limited.
This product includes software developed by Charles D. Cranor and Seth Widoff.
This product includes software developed by Charles D. Cranor and Washington University.
This product includes software developed by Charles D. Cranor, Washington University, and the University of California, Berkeley and its contributors.
This product includes software developed by Charles D. Cranor, Washington University, the University of California, Berkeley and its contributors.
This product includes software developed by Charles D. Cranor.
This product includes software developed by Charles Hannum.
This product includes software developed by Charles M. 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 M. Hannum.
This product includes software developed by Christian E. Hopps, Ezra Story, Kari Mettinen, Markus Wild, Lutz Vieweg and Michael Teske.
This product includes software developed by Christian E. Hopps.
This product includes software developed by Christian Limpach
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 Chuck Silvers.
This product includes software developed by Colin Wood for the NetBSD Project.
This product includes software developed by Colin Wood.
This product includes software developed by Cybernet Corporation and Nan Yang Computer Services Limited
This product includes software developed by Daishi Kato
This product includes software developed by Dale Rahn.
This product includes software developed by Daniel Widenfalk and Michael L. Hitch.
This product includes software developed by Daniel Widenfalk for the NetBSD Project.
This product includes software developed by Darrin B. Jewell
This product includes software developed by David Miller.
This product includes software developed by Dean Huxley.
This product includes software developed by Eduardo Horvath.
This product includes software developed by Eric S. Hvozda.
This product includes software developed by Eric S. Raymond
This product includes software developed by Eric Young (eay@@mincom.oz.au)

This product includes software developed by Eric Young (eay@cryptsoft.com)
This product includes software developed by Eric Young (eay@mincom.oz.au)
This product includes software developed by Ezra Story and by Kari Mettinen.
This product includes software developed by Ezra Story, by Kari Mettinen and by Bernd Ernesti.
This product includes software developed by Ezra Story, by Kari Mettinen, Michael Teske and by Bernd Ernesti.
This product includes software developed by Ezra Story, by Kari Mettinen, and Michael Teske.
This product includes software developed by Ezra Story.
This product includes software developed by Frank van der Linden for the NetBSD Project.
This product includes software developed by Gardner Buchanan.
This product includes software developed by Gary Thomas.
This product includes software developed by Gordon Ross
This product includes software developed by Gordon W. Ross
This product includes software developed by HAYAKAWA Koichi.
This product includes software developed by Harvard University and its contributors.
This product includes software developed by Harvard University.
This product includes software developed by Herb Peyerl.
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 F. Darwin and others.
This product includes software developed by Ian W. Dall.
This product includes software developed by Ichiro FUKUHARA.
This product includes software developed by Ignatios Souvatzis for the NetBSD Project.
This product includes software developed by Internet Initiative Japan Inc.
This product includes software developed by James R. Maynard III.
This product includes software developed by Jared D. McNeill.
This product includes software developed by Jason L. Wright
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 Joerg Wunsch
This product includes software developed by John Birrell.
This product includes software developed by John P. Wittkoski.
This product includes software developed by John Polstra.
This product includes software developed by Jonathan R. Stone for the NetBSD Project.
This product includes software developed by Jonathan Stone and Jason R. Thorpe for the NetBSD Project.
This product includes software developed by Jonathan Stone.
This product includes software developed by Jukka Marin.
This product includes software developed by Julian Highfield.
This product includes software developed by Kazuhisa Shimizu.
This product includes software developed by Kazuki Sakamoto.
This product includes software developed by Kenneth Stailey.
This product includes software developed by Kiyoshi Ikehara.
This product includes software developed by Klaus Burkert,by Bernd Ernesti, by Michael van Elst, and by the University of California, Berkeley and its contributors.
This product includes software developed by LAN Media Corporation and its contributors.
This product includes software developed by Leo Weppelman for the NetBSD Project.
This product includes software developed by Leo Weppelman.
This product includes software developed by Lloyd Parkes.
This product includes software developed by Luke Mewburn.
This product includes software developed by Lutz Vieweg.
This product includes software developed by MINOURA Makoto, Takuya Harakawa.
This product includes software developed by Manuel Bouyer.
This product includes software developed by Marc Horowitz.
This product includes software developed by Marcus Comstedt.
This product includes software developed by Mark Brinicombe for the NetBSD project.
This product includes software developed by Mark Brinicombe.
This product includes software developed by Mark Murray
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 Masanobu Saitoh.
This product includes software developed by Masaru Oki.
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 Matt DeBergalis
This product includes software developed by Matthew Fredette.

This product includes software developed by Matthias Pfaller.
This product includes software developed by Michael Graff for the NetBSD Project.
This product includes software developed by Michael Graff.
This product includes software developed by Michael L. Hitch.
This product includes software developed by Michael Shalayeff.
This product includes software developed by Michael Smith.
This product includes software developed by Mike Glover and contributors.
This product includes software developed by Mike Pritchard.
This product includes software developed by Minoura Makoto.
This product includes software developed by Nan Yang Computer Services Limited.
This product includes software developed by Niels Provos.
This product includes software developed by Niklas Hallqvist, Brandon Creighton and Job de Haas.
This product includes software developed by Niklas Hallqvist.
This product includes software developed by Onno van der Linden.
This product includes software developed by Paul Kranenburg.
This product includes software developed by Paul Mackerras.
This product includes software developed by Per Fogelstrom
This product includes software developed by Peter Galbavy.
This product includes software developed by Phase One, Inc.
This product includes software developed by Philip A. Nelson.
This product includes software developed by Philip L. Budne.
This product includes software developed by RiscBSD.
This product includes software developed by Roar Thronæs.
This product includes software developed by Rodney W. Grimes.
This product includes software developed by Roger Hardiman
This product includes software developed by Roland C. Dowdeswell.
This product includes software developed by Rolf Grossmann.
This product includes software developed by Ross Harvey for the NetBSD Project.
This product includes software developed by Ross Harvey.
This product includes software developed by Scott Bartram.
This product includes software developed by Scott Stevens.
This product includes software developed by Shingo WATANABE.
This product includes software developed by Softweyr LLC, the University of California, Berkeley, and its contributors.
This product includes software developed by Soren S. Jorvang.
This product includes software developed by Stephan Thesing.
This product includes software developed by Steve Woodford.
This product includes software developed by Takashi Hamada
This product includes software developed by Takumi Nakamura.
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 Tetsuya Isaki.
This product includes software developed by Thomas Gerner
This product includes software developed by Thomas Klausner for the NetBSD Project.
This product includes software developed by Tobias Weingartner.
This product includes software developed by Todd C. Miller.
This product includes software developed by Tohru Nishimura and Reinoud Zandijk for the NetBSD Project.
This product includes software developed by Tohru Nishimura for the NetBSD Project.
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 Toru Nishimura.
This product includes software developed by Trimble Navigation, Ltd.
This product includes software developed by WIDE Project and its contributors.
This product includes software developed by Waldi Ravens.
This product includes software developed by Wasabi Systems for Zembu Labs, Inc. http://www.zembu.com/
This product includes software developed by Winning Strategies, Inc.
This product includes software developed by Wolfgang Solfrank.
This product includes software developed by Yasushi Yamasaki
This product includes software developed by Zembu Labs, Inc.
This product includes software developed by the Alice Group.
This product includes software developed by the Center for Software Science at the University of Utah.
This product includes software developed by the Charles D. Cranor, Washington University, 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 David Muir Sharnoff.
This product includes software developed by the Harvard University and its contributors.
This product includes software developed by the Kungliga Tekniska Högskolan and its contributors.

This product includes software developed by the Network Research Group at Lawrence Berkeley Laboratory.
This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.OpenSSL.org/)
his product includes software developed by the PocketBSD project and its contributors.
This product includes software developed by the RiscBSD kernel team
This product includes software developed by the RiscBSD team.
This product includes software developed by the SMCC Technology Development Group at Sun Microsystems, Inc.
This product includes software developed by the University of California, Berkeley and its contributors, as well as the Trustees of Columbia University.
This product includes software developed by the University of California, Lawrence Berkeley Laboratory and its contributors.
This product includes software developed by the University of California, Lawrence Berkeley Laboratory.
This product includes software developed by the University of Illinois at Urbana and their contributors.
This product includes software developed by the University of Vermont and State Agricultural College and Garrett A. Wollman.
This product includes software developed 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 for the FreeBSD project
This product includes software developed for the NetBSD Project by Bernd Ernesti.
This product includes software developed for the NetBSD Project by Christopher G. Demetriou.
This product includes software developed for the NetBSD Project by Christos Zoulas
This product includes software developed for the NetBSD Project by Emmanuel Dreyfus.
This product includes software developed for the NetBSD Project by Frank van der Linden
This product includes software developed for the NetBSD Project by Ignatios Souvatzis.
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 Michael L. Hitch.
This product includes software developed for the NetBSD Project by Perry E. Metzger.
This product includes software developed for the NetBSD Project by Scott Bartram and Frank van der Linden
This product includes software developed for the NetBSD Project by Allegro Networks, Inc., and Wasabi Systems, Inc.
This product includes software developed for the NetBSD Project by Genetec Corporation.
This product includes software developed for the NetBSD Project by Jonathan Stone.
This product includes software developed for the NetBSD Project by Piermont Information Systems Inc.
This product includes software developed for the NetBSD Project by SUNET, Swedish University Computer Network.
This product includes software developed for the NetBSD Project by Shigeyuki Fukushima.
This product includes software developed for the NetBSD Project by Wasabi Systems, Inc.
This product includes software developed under OpenBSD by Per Fogelstrom Opsycon AB for RTMX Inc, North Carolina, USA.
This product includes software developed under OpenBSD by Per Fogelstrom.
This software is a component of "386BSD" developed by William F. Jolitz, TeleMuse.
This software was developed by Holger Veit and Brian Moore for use with "386BSD" and similar operating systems. "Similar operating systems" includes mainly non-profit oriented systems for research and education, including but not restricted to "NetBSD", "FreeBSD", "Mach" (by CMU).
This software includes software developed by the Computer Systems Laboratory at the University of Utah.
This product includes software developed by Computing Services at Carnegie Mellon University (http://www.cmu.edu/computing/).

This product includes software developed at Ludd, University of Lule}, Sweden.
This product includes software developed by Adam Glass, David Jones, and Gordon Ross.

The End