NAME

INSTALL — Installation procedure for NetBSD/mac68k.

CONTENTS

About this Document....2
What is NetBSD?....2
Changes Between The NetBSD 4.0 release and 4.0.1 update....2
  Security Advisories Fixes....3
  Other Security Fixes....3
  Networking....4
  Libraries....4
  Drivers....4
  Miscellaneous....4
  Platforms specific....4
Changes Between The NetBSD 3.0 and 4.0 Releases....5
  Networking....5
  File systems....5
  Drivers....6
  Platforms....8
  Kernel subsystems....8
  Security....9
  Userland....9
  Components removed from NetBSD....11
The Future of NetBSD....11
Sources of NetBSD....12
NetBSD 4.0.1 Release Contents....12
  NetBSD/mac68k subdirectory structure....13
  Binary distribution sets....14
NetBSD/mac68k System Requirements and Supported Devices....16
  Supported models....16
  Supported devices....16
  Unsupported models....17
  Known hardware issues with this release....17
Getting the NetBSD System on to Useful Media....17
Preparing your System for NetBSD installation....18
Installing the NetBSD System (Sysinst Method)....20
  Running the sysinst installation program....20
    Introduction....20
    Possible hardware-specific issues....20
    General....20
    Quick install....20
    Booting NetBSD....22
    Network configuration....23
    Preparing a disk for Mac OS and NetBSD....23
    Installation drive selection and parameters....23
    Partitioning the disk....23
    Preparing your hard disk....25
    Getting the distribution sets....25
    Installation using ftp....25
    Installation using NFS....26

Installation from CD-ROM.....26
Installation from Mac OS file systems.....26
Installation from an unmounted file system.....26
Installation from a local directory.....26
Extracting the distribution sets.....26
Finalizing your installation.....27
Installing the NetBSD System (Traditional Method).....27
Preparing the file system(s) 27
Installing the files 27
Installation of base files 28
Booting the system 29
Post installation steps.....29
Upgrading a previously-installed NetBSD System.....32
Compatibility Issues With Previous NetBSD Releases.....33
Issues affecting an upgrade from NetBSD 2.1 and older 33
Issues affecting an upgrade from NetBSD 3.x releases. 34
Issues with GDB 6.5.....34
Using online NetBSD documentation.....35
Administrivia.....35
Thanks go to.....36
We are.....36
Legal Mumbo-Jumbo.....45
The End.....49

DESCRIPTION

About this Document

This document describes the installation procedure for NetBSD 4.0.1 on the *mac68k* 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 4.0.1 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. Please see the NetBSD web site at **http://www.NetBSD.org/** for information on them.)

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 4.0 release and 4.0.1 update

The NetBSD 4.0.1 update is the first security/critical update of the NetBSD 4.0 release branch. This represents a selected subset of fixes deemed critical in nature for stability or security reasons.

These fixes will also appear in future releases (NetBSD 4.1), together with other less-critical fixes and feature enhancements.

The complete list of changes can be found in the CHANGES-4.0.1:

ftp://ftp.NetBSD.org/pub/NetBSD/NetBSD-4.0.1/CHANGES-4.0.1

file in the top level directory of the NetBSD 4.0.1 release tree.

Security Advisories Fixes

- NetBSD-SA2008-004, `bzip2`(1) Multiple issues (CVE-2008-1372 and CVE-2005-0953), has been fixed by upgrading bzip2 to 1.0.5.
- NetBSD-SA2008-005, OpenSSH Multiple issues (CVE-2008-1483 and CVE-2008-1657), has been fixed by applying patches from upstream.
- NetBSD-SA2008-006, Integer overflow in `strfmon`(3) function (CVE-2008-1391), has been fixed.
- NetBSD-SA2008-008, OpenSSL Montgomery multiplication (CVE-2007-3108), has been fixed.
- NetBSD-SA2008-009, BIND cache poisoning (CVE-2008-1447 and CERT VU#800113), has been fixed by updating BIND to 9.4.2-P2. Note there are two related changes to this advisory:
  - The default behavior of ipfilter's Port Address Translation has been changed to using random port allocation rather than sequential mappings, to avoid decreasing the randomness of source ports used for DNS queries which affects the BIND cache poisoning problem.
  - A '`query-source`' statement, which could allow the BIND cache poisoning attack, has been commented out in the default `named.conf`(5) file.
- NetBSD-SA2008-010, Malicious PPPoE discovery packet can overrun a kernel buffer (CVE-2008-3584), has been fixed.
- NetBSD-SA2008-011, ICMPv6 MLD query (CVE-2008-2464), has been fixed.
- NetBSD-SA2008-012, Denial of Service issues in `racoon`(8) (CVE-2008-3652), has been fixed by upgrading ipsec-tools to release 0.7.1. Note this also fixes CVE-2008-3651.
- upcoming NetBSD-SA2008-013, IPv6 Neighbor Discovery Protocol routing vulnerability (CVE-2008-2476), has been fixed.
- upcoming NetBSD-SA2008-014, Remote cross-site request forgery attack issue in `ftpd`(8) (CVE-2008-4247), has been fixed.
- upcoming NetBSD-SA2008-015, Remote kernel panics on IPv6 connections (CVE-2008-3530), has been fixed.

Note: NetBSD-SA2008-007 and advisories prior to NetBSD-SA2008-004 don't affect NetBSD 4.0.

Other Security Fixes

- Fix a buffer overrun which could crash a FAST_IPSEC kernel.
- `tcpdump`(8): Fix CVE-2007-1218, CVE-2007-3798 and CAN-2005-1278 in base-tcpdump.
- Fix a buffer overflow of PCF font parser in X11 libXfont library (CVE-2008-0006).
- Fix a buffer overflow of Tektronix Hex Format support in binutils (CVE-2006-2362).
- `machfb`(4), `voodoofb`(4): Introduce two missing KAUTH_GENERIC_ISSUSER check in the `mmap`(2) code.

Networking

- Update root.cache to 2008020400 version.
- Fix IP packet forwording code to make sure to send a reasonable fragment size when IPSEC is configured.
- Fix a bug in TCP SACK code which causes data corruption.
- Fix an `rc.d`(8) script for `amd`(8) not to shutdown gracefully since it seems to cause problems for more people than the old (also broken) behavior.
- `ftpd`(8): Fix and reorganize PAM support.

Libraries

- Pthread support of BIND has been disabled for future binary compatibility after removal of the scheduler activations.
- Fix coredump of gdtoa (conversion between binary floating-point and ASCII string) functions on out of memory conditions.

Drivers

- `fxp`(4): Fix random pool corruption and hangup problems.
- `wd`(4): Handle more LBA48 bug quirks on some Hitachi's SATA/IDE drives.

Miscellaneous

- Disable a NULL pointer check in zlib for standalone programs. This fixes errors on loading a gzipped kernel (including installation kernels) on several ports (news68k etc.) whose kernels are loaded at address zero.
- `awk`(1): Bring back an accidentally removed fix to allow escape of a newline in string literals.
- `gcc`(1):
  - Fix compilation of native sh3 gcc on 64-bit build machines.
  - Fix an internal compiler error on compiling m68k softfloat or m68010 targets on 64-bit build machines.
- `zgrep`(1): Make '**`-h`**' option (suppress filenames on output when multiple files are searched) actually work.
- Fix parallel build failure on building hpcarm, hpcmips and hpcsh releases.

Platforms specific

- acorn32: Fix a bootloader problem on some RiscPCs.
- cobalt:
  - Add a workaround to avoid panic on probing a multi function PCI device on Qube's PCI slot.
  - Fix a bug in the interrupt handler which causes network freeze if more than one interfaces are used.
- hp700: Fix potential kernel / userland memory corruption in `copyinstr`(9) and `copyoutstr`(9).
- sparc64: Fix a bug in `locore.s` which causes unexpected behavior.
- sun3: Fix a bug which might cause an occasional panic during boot.
- vax: Make syscall handler use proper `copyin`(9) function on parsing syscall args.

Changes Between The NetBSD 3.0 and 4.0 Releases

The NetBSD 4.0 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 more than a year of development that went into the NetBSD 4.0 release. The complete list of changes can be found in the CHANGES:

ftp://ftp.NetBSD.org/pub/NetBSD/NetBSD-4.0/CHANGES

and CHANGES-4.0:

ftp://ftp.NetBSD.org/pub/NetBSD/NetBSD-4.0/CHANGES-4.0

files in the top level directory of the NetBSD 4.0 release tree. Some highlights include:

Networking

- `agr`(4): new pseudo-device driver for link level aggregation.
- IPv6 support was extended with an RFC 3542-compliant API and added for `gre`(4) tunnels and the `tun`(4) device.
- A NDIS-wrapper was added to use Windows binary drivers on the i386 platform, see `ndiscvt`(8).
- The IPv4 source-address selection policy can be set from a number of algorithms. See "IPSRCSEL" in `options`(4) and `in_getifa`(9).
- Imported `wpa_supplicant`(8) and `wpa_cli`(8). Utilities to connect and handle aspects of 802.11 WPA networks.
- Imported `hostapd`(8). An authenticator for IEEE 802.11 networks.
- `carp`(4): imported Common Address Redundancy Protocol to allow multiple hosts to share a set of IP addresses for high availability / redundancy, from OpenBSD.
- ALTQ support for the PF packet filter.
- `etherip`(4): new EtherIP tunneling device. It's able to tunnel Ethernet traffic over IPv4 and IPv6 using the EtherIP protocol specified in RFC 3378.
- `ftpd`(8) can now run in standalone mode, instead of from `inetd`(8).
- `tftp`(1) now has support for multicast TFTP operation in open-loop mode, server is in progress.
- `tcp`(4): added support for RFC 3465 Appropriate Byte Counting (ABC) and Explicit Congestion Notification as defined in RFC 3168.

File systems

- scan_ffs(8), scan_lfs(8): utilities to find FFSv1/v2 and LFS partitions to recover lost disklabels on disks and image files.
- tmpfs: added a new memory-based file system aimed at replacing mfs. Contrary to mfs, it is not based on a disk file system, so it is more efficient both in overall memory consumption and speed. See mount_tmpfs(8).
- Added UDF support for optical media and block devices, see mount_udf(8). Read-only for now.
- NFS export list handling was changed to be filesystem independent.
- LFS: lots of stability improvements and new cleaner daemon. It is now also possible to use LFS as root filesystem.
- vnd(4): the vnode disk driver can be used on filesystems such as smbfs and tmpfs.
- Support for System V Boot File System was added, see newfs_sysvbfs(8) and mount_sysvbfs(8).

Drivers

- Audio:
  - Support for new models on drivers such as Intel ICH8/6300ESB, NVIDIA nForce 3/4, etc.
  - Added support for AC'97 modems.
  - auich(4): added support to handle the AC'97 modem as audio device, enabled with the kernel option "AUICH_ATTACH_MODEM"
  - azalia(4): added support for S/PDIF.
- Hardware Monitors:
  - amdpm(4): added support for the i2c bus on the AMD-8111 used on many Opteron motherboards and for the Analog Devices ADT7464 hardware monitor chip.
  - adt7467c(4): new driver for Analog Devices ADT7467 and ADM1030 hardware monitor chips.
  - ipmi(4): new driver for motherboards implementing the Intelligent Platform Management Interface 1.5 or 2.0, from OpenBSD.
  - it(4): new driver for iTE 8705F/8712F and SiS 950 hardware monitors.
  - The lm(4) driver was rewritten and support for more chips was added, for example for Winbond W83627HF, W83627THF, W83627DHG and Asus AS99127F.
  - owtemp(4): new driver for the 1-Wire temperature sensors.
  - tmp121temp(4): new driver for the Texas Instruments TMP121 temperature sensor.
  - ug(4): new driver for Abit uGuru hardware monitor found on newer Abit motherboards.
- Miscellaneous:
  - geodewdog(4): new AMD Geode SC1100 Watchdog Timer driver.
  - gscpcib(4): new AMD Geode SC1100 PCI-ISA bridge that provides support for the GPIO interface.
- Networking:

  - ath(4): updated HALs with support for WiSOC (AR531x) and 32bit SPARC.
  - bge(4): added support for the following chips: BCM5753, BCM5753M, BCM5715, BCM5754, BCM5755 and BCM5787. Numerous improvements and bugfixes were made too.
  - kse(4): new driver for Micrel KSZ8842/8841 PCI network cards.
  - msk(4): new driver for Marvell Yukon 2 GigE PCI network cards, from OpenBSD.
  - nfe(4): new driver for NVIDIA nForce Ethernet network cards, from OpenBSD.
  - ral(4): new 802.11 driver for PCI/Cardbus Ralink RT2500, RT2501, RT2600, RT2661 and RT2500 USB chipsets, from OpenBSD.
  - rum(4): new 802.11 driver for USB Ralink RT2501 and RT2601 chipsets, from OpenBSD.
  - sip(4): now works on sparc64.
  - tlp(4): added support for ASIX AX88140A and AX88141.
  - vr(4): added support for the VIA Rhine III.
  - wm(4): added support for i8003, ICH8, ICH9 and others. Support for IPv6 Rx TCP/UDP Checksum Offloading and more.
  - wpi(4): new driver for Intel PRO/Wireless 3945ABG PCI 802.11 network cards, from OpenBSD.

- Security:
  - glxsb(4): new driver for the AMD Geode LX AES Security Block that provides random numbers and AES acceleration, from OpenBSD.
- Power Management:
  - Support for Intel Speedstep SMI on PIIX4 PCI-ISA for i386.
  - Support for AMD PowerNow and Cool'n'Quiet Technology on K7 and K8 CPUs (both in 32 and 64 bit mode), including Athlon Mobile, Athlon64, Opteron or X2. See options(4) for more information.
  - Support for more Enhanced Speedstep CPUs, including VIA C7/Eden and Intel Core Solo/Duo/Duo2. See options(4) for more information.
  - The Enhanced Speedstep and PowerNow drivers were modified to be able to be scaled in all CPUs available, saving power on SMP systems.
- Storage:
  - ahcisata(4): new driver for AHCI 1.0 and 1.1 compliant SATA controllers.
  - ataraid(4): added support to handle Adaptec HostRAID and VIA V-Tech software RAID.
  - ciss(4): new driver for HP/Compaq 5th+ generation Smart ARRAY controllers, from OpenBSD.
  - fdc(4): added support for SBus based sparc64 machines and fixed formatting on sparc.
  - gcscide(4): new driver for the AMD Geode CS5535 Companion Device IDE controller.
  - jmide(4): new driver for JMicron Technology JMB36x PCIe to SATA II/PATA controllers.
  - mfi(4): new driver for LSI Logic and Dell MegaRAID SAS controllers, from OpenBSD.
  - mpt(4): added support for newer SAS and similar devices.

  - njata(4): new driver for Workbit NinjaATA-32 CardBus IDE controller.
  - pdcsata(4): added support for the Promise PDC20775, PDC20771, PDC40518, PDC40718 and some bugfixes.
  - piixide(4): added support for some ICH8/ICH8-M/ICH9 IDE and SATA controllers.
  - svwsata(4): new driver for Serverworks K2 SATA controllers, from OpenBSD.
  - viaide(4) added support for the VIA VT8237A SATA controller and AMD CS5536 Companion Device IDE Controller.
- USB:
  - ucycom(4): new driver for Cypress microcontroller based serial devices.
  - uipaq(4): new driver for the iPAQ devices.
  - uslsa(4): new driver for Silicon Labs CP210x series serial adapters.
  - utoppy(4): new driver for the Topfield TF5000PVR range of digital video recorders.

Platforms

- i386:
  - Added support for the for Multiboot specification. This means much improved support for loading the kernel by GRUB, including passing in parameters to the kernel.
  - Added the unichromefb framebuffer driver that supports the VIA Unichrome Graphics adapter.
  - vesafb(4): added new framebuffer driver that supports VESA BIOS (VBE) 2.0 extensions and up.
  - Added cd9660 file system support to the BIOS bootloader.
- evbarm: new platform support for Arcom Viper PXA255-based single board, Atmark Techno Armadillo-9 and Armadillo-210, Certance CP-3100, Linksys NSLU2 (a.k.a. "Slug") and I-O DATA HDL-G Giga LANDISK NAS devices.
- evbmips: added support for Alchemy Au1550 processors, DBAu1550 boards, Alchemy Au15XX PCI host, (OMS-AL400/128) and Atheros AR5312 SoC.
- New port ews4800mips: NEC's MIPS based EWS4800 workstations.
- cobalt: added support for booting off raidframe RAID1 mirrors.
- hpcmips: added the teliosio(4) driver for the Sharp Telios LCD screen and Battery unit.
- New port landisk: port to the SH4 processor based NAS appliances, supporting models by I-O DATA (USL-5P, HDL-U, HDL-AV, HDL-W and HDLM-U series, SuperTank LAN Tank, UHDL-160U and UHDL-300U) and Plextor PX-EH16L, PX-EH25L and PX-EH40L.
- macppc: this port has gained support to use accelerated wsdisplay drivers by default (if possible), and uses the appropriate driver rather than the Generic Open Firmware Framebuffer.
- prep: this port has been modernized, and support for five additional machines has been added, among them the IBM 7024-E20 and 7025-F30 models and Motorola Powerstack E1. Additionally, sysinst support was added, and the bootloader process was improved, allowing easy installation and upgrade to future releases.
- sparc: added support for booting off raidframe RAID1 mirrors.

- Xen: support for Xen3 domU and dom0 (Unprivileged domain and domain 0), including support for hardware virtualization on CPUs that support it.

Kernel subsystems

- Improved Firewire (IEEE1394) support imported from FreeBSD.
- The `midi`(4) framework got a complete overhaul for better support of Active Sensing and improved handling of tempo and timebase changes.
- Added a Bluetooth protocol stack including:
  - hardware drivers: `ubt`(4) for USB controllers, and `bt3c`(4) for the 3Com Bluetooth PC-Card.
  - socket based access to the HCI, L2CAP, RFCOMM and SCO protocols.
  - pseudo drivers for integrating services on remote Bluetooth devices such as Keyboards, Mice and SCO Audio into the NetBSD device framework.

  See `bluetooth`(4), `bthset`(1) and `btpin`(1).
- Imported the `bio`(4) framework from OpenBSD, to query/control block hardware RAID device controllers. Currently supporting the `mfi`(4) driver.
- Kernel uses stateful read-ahead algorithm.
- `dkctl`(8) can be used to switch buffer queuing strategies on the fly on `wd`(4) disks, see also `bufq`(9).
- `fileassoc`(9) is used by Veriexec, it adds in-kernel and file-system independent file meta-data association interface.
- `firmload`(9): an API for loading firmware images used by various hardware devices.
- `gpio`(4): imported General Purpose I/O framework from OpenBSD.
- `onewire`(4): imported Dallas Semiconductor 1-wire bus framework from OpenBSD.
- The `proplib`(3) protocol was added for sending property lists to/from the kernel using ioctls.
- `spi`(4): new SPI (Serial Peripherial Interface) framework.
- `timecounter`(9) adds a new time-keeping infrastructure along with NTP API 4 nanokernel implementation. Almost all platforms were changed to support this API.
- Start of 32bit-Linux-emulation for amd64 (COMPAT_LINUX32).
- `wscons`(4) console driver supports splash screens, scrolling, progress bar for kernel and boot messages.

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.

Security

- The FAST_IPSEC IPsec implementation was extended to use hardware acceleration for IPv6, in addition to the hardware accelerated IPv4 that was available before. See `fast_ipsec`(4) for more information.
- `mprotect`(2) got restrictions to enforce W^X policies, from PaX. See `options`(4), `sysctl`(3), and `paxctl`(1).
- GCC 4's support for stack smashing protection (SSP) was enabled by adding libssp, see `security`(8).
- The kernel authorization framework `kauth`(9) was added, replacing the traditional BSD credential management and privileged operation access control with an abstract layer, allowing the implementation of various security models either as part of the NetBSD distribution or as third-party LKMs.

NetBSD's kernel authorization is a hybrid clean-room implementation of a similar interface developed by Apple, extending its capabilities and combining concepts of credential inheritance control.

Userland

- 3rd party software updates:
  - BIND 9.4.1-P1
  - OpenSSL 0.9.8e
  - CVS 1.11.22
  - OpenSSH 4.4
  - gettext 0.14.4
  - PF from OpenBSD 3.7
  - (n)awk 20050424
  - Postfix 2.4.5
  - am-utils 6.1.3
  - file 4.21
  - zlib 1.2.3
  - GNU binutils 2.16.1
  - GNU groff 1.19.2
  - IPFilter 4.1.23
  - GNU gcc 4.1.2 prerelease
  - GNU gdb 6.5 (some architectures)
  - NTP 4.2.4p2
  - pppd 2.4.4
- `cdplay`(1): added digital transfer mode support.
- `cksum`(1) can now verify checksums.
- `csplit`(1): new utility that splits a file into pieces. From FreeBSD/OpenBSD.
- `identd`(1): added support for forwarding ident queries and receiving of proxied ident queries.
- `getent`(1): added support for the ethers database.
- `gkermit`(1): new program for transferring files using the Kermit protocol.
- `mail`(1): added support for Mime and multi-character set handling, command line editing and completion.
- `utoppya`(1): new utility to interface to the `utoppy`(4) driver.
- `init`(8): added support for running multi-user in a chroot() environment. Allows / file system on e.g., `cgd`(4), `vnd`(4) or `ccd`(4) volumes.
- `gpt`(8): new GUID partition table maintenance utility, from FreeBSD.

- iSCSI target (server) code added, see `iscsi-target`(8); Initiator (client) code is underway.
- `lockstat`(8): new command to display a summary of kernel locking events recorded over the lifetime of a called program.
- `ofctl`(8): new command to display the OpenPROM or OpenFirmware device tree for the macppc, shark and sparc64.
- Various utilities to support Bluetooth were added:
  - `btconfig`(8) for controller configuration.
  - `btdevctl`(8) to manage pseudo devices relating to remote services.
  - `bthcid`(8) and `btpin`(1) for authenticating radio connections.
  - `sdpd`(8) for providing service discovery to remote devices.
  - `sdpquery`(1) for querying services on remote devices.
  - `rfcomm_sppd`(1) to access remote services over RFCOMM via stdio or pty.
  - `bthset`(1) for making connections to Bluetooth headsets.

Besides this list, there have also been innumerable bug fixes and other miscellaneous enhancements of course.

Components removed from NetBSD

In this release of NetBSD, some software components known from previous releases were removed from the system. In some cases those were components that are not useful anymore or their utility does not justify maintenance overhead. Other components were not working properly and there was lack of interest in fixing them.

- Sushi was removed from the base system due to lack of interest and maintenance. If you really want it, it is available in the CVS repository at othersrc/usr.sbin/sushi. However, be warned that it is unmaintained and is most likely out of date.
- Vinum was removed due to lack of interest and maintenance. At the time of removal, it had several known serious issues (including not being compilable). RAIDframe provides similar functionality. If you were using Vinum you will need to back up your data, delete the Vinum partitions, create RAIDframe partitions, and restore your data to them. Details about RAIDframe can be found in `raid`(4), `raidctl`(8), and the NetBSD Guide:
  http://www.NetBSD.org/docs/guide/en/chap-rf.html.
- Sendmail was removed. Postfix is the MTA and provides the `sendmail`(1) command line tool. Postfix has been included with NetBSD since NetBSD 1.5 was released in December 2005. Details about Postfix can be found in the NetBSD Guide:
  http://www.NetBSD.org/docs/guide/en/chap-mail.html.
  For those who need Sendmail, it is available from pkgsrc in the `mail/sendmail` and `mail/sendmail813` packages.
- NETCCITT and NETNS were removed due to lack of interest and maintenance. They had known serious issues (including being out of date with respect to other network code) and there were no known users at the time of their removal. Unfortunately, there is no replacement or option for them.
- UUCP was removed. The NetBSD improvements were merged into the pkgsrc version. For those who use UUCP tools, they are available from pkgsrc in the `net/uucp` package. The `cu`(1) command is available as a frontend to `tip`(1).

- The Fortran 77 compiler (g77) has been removed with the transition from GCC 3 to GCC 4, which does not include it. For those who need it, it is available from pkgsrc in the `lang/gcc3-f77` package.
- The evbsh5 port has been removed from NetBSD due to lack of interest, compounded by a lack of available SH5 hardware.

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 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/mirrors/.**

NetBSD 4.0.1 Release Contents

The root directory of the NetBSD 4.0.1 release is organized as follows:

`.../NetBSD-4.0.1/`

`CHANGES`	Changes between NetBSD 3.0 and branching 4.0.
`CHANGES-4.0`	Changes since NetBSD 4.0 was branched.
`CHANGES-4.0.1`	Changes between NetBSD 4.0 and 4.0.1.
`CHANGES.prev`	Changes in earlier NetBSD releases.
`LAST_MINUTE`	Last minute changes.
`MIRRORS`	A list of sites that mirror the NetBSD 4.0.1 distribution.
`README.files`	README describing the distribution's contents.

`TODO` NetBSD's todo list (also somewhat incomplete and out of date).

`patches/` Post-release source code patches.

`source/` Source distribution sets; see below.

In addition to the files and directories listed above, there is one directory per architecture, for each of the architectures for which NetBSD 4.0.1 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.
95 MB gzipped, 484 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.
6 MB gzipped, 25 MB uncompressed

src This set contains all of the base NetBSD 4.0.1 sources which are not in **gnusrc**, **sharesrc**, or **syssrc**.
45 MB gzipped, 214 MB uncompressed

syssrc This set contains the sources to the NetBSD 4.0.1 kernel for all architectures; `config`(1); and `dbsym`(8).
33 MB gzipped, 165 MB uncompressed

xsrc This set contains the sources to the X Window System.
95 MB gzipped, 502 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 -a MD5 *file*.

`SHA512` SHA512 digests for the various files in that directory, in the format produced by the command:
cksum -a SHA512 *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 SHA512 digest is the safest checksum, followed by the MD5 digest, and finally the POSIX checksum. The other two checksums are provided only to ensure that the widest possible range of systems can check the integrity of the release files.

NetBSD/mac68k subdirectory structure

The mac68k-specific portion of the NetBSD 4.0.1 release is found in the `mac68k` subdirectory of the distribution: `.../NetBSD-4.0.1/mac68k/`. 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 kernel containing code for everything supported in this release using the standard SCSI driver.
`netbsd-GENERICSBC.gz`
A kernel containing code for everything supported in this release using the SBC variant of the SCSI driver.
`sets/` mac68k binary distribution sets; see below.

`installation/`
`instkernel/`
mac68k boot and installation kernels; see installation section (Sysinst Method), below.
`misc/` Miscellaneous mac68k installation utilities; see the Traditional method installation section, below.

Binary distribution sets

The NetBSD mac68k binary distribution sets contain the binaries which comprise the NetBSD 4.0.1 release for the mac68k. The binary distribution sets can be found in the `mac68k/binary/sets` subdirectory of the NetBSD 4.0.1 distribution tree, and are as follows:

base The NetBSD 4.0.1 mac68k **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.
24 MB gzipped, 68 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.
23 MB gzipped, 83 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, 8 MB uncompressed

kern-GENERIC
This set contains a NetBSD/mac68k 4.0.1 `GENERIC` kernel, named `/netbsd`. You *must* install either this distribution set or kern-GENERICSBC.
2 MB gzipped, 3 MB uncompressed

kern-GENERICSBC
This set contains a NetBSD/mac68k 4.0.1 `GENERICSBC` kernel, named `/netbsd`. You *must* install either this distribution set or kern-GENERIC.
2 MB gzipped, 4 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, 33 MB uncompressed

misc This set includes the (rather large) system dictionaries, the typesettable document set, and other files from `/usr/share`.
4 MB gzipped, 12 MB uncompressed

text This set includes NetBSD's text processing tools, including `groff`(1), all related programs, and their manual pages.
3 MB gzipped, 11 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 4.5.0. 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, 18 MB uncompressed

xcomp The extra libraries and include files needed to compile X source code.
11 MB gzipped, 39 MB uncompressed

xfont Fonts needed by the X server and by X clients.
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. This includes the Xmac68k *monochrome* server with man pages.
5 MB gzipped, 13 MB uncompressed

The mac68k 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.

Kernels suitable for booting from an AppleShare server may be found in the `mac68k/binary/kernels` subdirectory of the NetBSD 4.0.1 distribution tree. These kernels are generally named something like `netbsd-GENERIC.gz` and can be booted as-is by the NetBSD/mac68k Booter utility, if desired. Please note that these kernels are simply gzipped and are not in tar archives.

The Mac OS based utilities necessary for installing and running NetBSD can be found in the `mac68k/installation/misc` subdirectory of the NetBSD 4.0.1 distribution tree. The important files in this directory are as follows:

`Booter.sea.hqx` The NetBSD/mac68k Booter utility. This program is used to boot the NetBSD kernel from within Mac OS.
141 KB archived

`Installer.sea.hqx` The NetBSD/mac68k Installer utility. This program is used to install the distribution sets onto your NetBSD partition(s). This utility is used only in a Traditional method installation; it is not used or required for an installation using the **`sysinst`** method.
147 KB archived

`Mkfs.sea.hqx` The Mkfs utility. This program is used to format your chosen partitions so that they can be used with NetBSD. This utility is used only in a Traditional method installation; it is not used or required for an installation using the **`sysinst`** method.
76 KB archived

These files are all BinHexed, self-extracting archives. If you need them, the sources for these utilities are in the `src` subdirectory. This utility is used only in a Traditional method installation.

Note: Each directory in the mac68k binary distribution also has its own checksum files, just as the source distribution does.

NetBSD/mac68k System Requirements and Supported Devices

NetBSD/mac68k 4.0.1 runs on several of the older Macintosh computers. About 4 MB of RAM is sufficient to boot a stripped-down custom kernel, and a subset of the system can be squeezed onto a 40 MB hard disk with considerable creativity and persistence. However, an 80 MB disk should be considered a practical minimum, and to do anything more interesting than booting at least 8 MB of RAM and more disk space is recommended. About 180 MB will be necessary to install all of the NetBSD/mac68k 4.0.1 binary system distribution sets (note that this does not count swap space!). You can drop this to 150 MB if you choose not to install the binary X11 distribution sets. Much more disk space is required to install the source and objects as well; a complete source distribution, including X11, consumes nearly 1 GB of storage.

Please note that to install NetBSD/mac68k 4.0.1 using the **`sysinst`** method, your system must have a minimum of 6 MB of RAM and 60 MB of available disk space (i.e. not part of an in-use HFS partition).

Supported models

- Mac II, IIx, IIcx, SE/30, IIci, IIsi, IIvx, IIvi
- Performa 400/405/410/430, Performa 450, Performa 460/466/467
- Performa 520, Performa 550/560, Performa 600/600CD
- LC II, LC III, LC III+, LC 520, LC 550
- MacTV
- Classic II, Color Classic
- Centris 650
- Quadra 610, Quadra 630, Quadra 650, Quadra 700, Quadra 800
- Quadra/Centris 660AV, Quadra 840AV
- PowerBook 140, PowerBook 145/145B, PowerBook 170
- PowerBook 160, PowerBook 165, PowerBook 180
- PowerBook 165c, PowerBook 180c, PowerBook 550c

Supported devices

- Onboard SCSI bus and most SCSI tapes, hard drives, and CD-ROMs
- Internal sound — enough to beep on some machines, anyway
- Most basic NuBus video cards (there have been some problems with some 24-bit color cards and with most QuickDraw acclerators)
- Both internal serial ports
- ADB keyboards and mice (both Apple and a number of third party multi-button mice and trackballs are supported)
- Ethernet cards based on the National Semiconductor 8390 and the SONIC (DP83932) chips (Asante, Apple, and a few others — problems still with Ethernet and many NuBus video cards)
- Ethernet cards based on the SMC 91c92 and 91c100 (FEAST) chips. This includes the Asante-FAST 10/100 cards
- Onboard Ethernet based on the SONIC chip for Quadra-series Macs
- Onboard Ethernet based on the MACE (Am79C940) chip for the Quadra AV-series Macs
- Ethernet port on Asante NetDock and Newer Ether MicroDock, for PowerBook Duo series
- Comm-slot Ethernet should be working for most machines/cards

If your 68030 system is not listed above, it may be because of a problem with accessing onboard video, and it may still work with a serial console. Some of the known ones in this category:

- Mac Classic series
- PowerBook Duo series

If your 68LC040 system is not listed above, it is due to a problem with floating point emulation (FPE) for this type of processor. Machines in this category include:

- Newer LC-series machines (47x, 57x)
- Newer Performa-series machines (47x, 57x, 58x, 63x, 640)
- Some PowerBook 500-series Macs

Unsupported models

- Macintosh IIfx

  This machine has unusual custom chips for the ADB and serial interfaces which make support for it difficult. Work is in progress on this, though.

- Quadra 900/950

  These machines have I/O processor chips for their ADB interfaces similar to those used in the IIfx and thus face similar support problems. Note that you can use a serial console on these systems.

- PowerPC-based Macs

  This is a separate effort from the mac68k port. PowerMacs use hardware that is often fairly different from that of the mac68k port. If you are interested in this, you might want to take a look at the new NetBSD/macppc port:

  http://www.NetBSD.org/ports/macppc/index.html

Known hardware issues with this release

- Real Time Clock

  Due to oddities of the Macintosh hardware interrupt priority scheme, NetBSD/mac68k keeps very poor time. Under a high interrupt load (e.g. SCSI or serial port activity) , a machine can lose several minutes per hour. A consequence of this problem is that attempting to run **ntpd** is generally rather pointless.

- SCSI difficulties

  The NetBSD/mac68k SCSI drivers are not quite as robust as their Mac OS counterparts. Symptoms of these problems are that some SCSI disks will not work under NetBSD that work fine under Mac OS. Other problems include occasional file system corruption with some types of drives and the general unreliability of removable SCSI media. Keep in mind that there are no clear patterns with these problems, and they do not appear to affect the majority of users.

Getting the NetBSD System on to Useful Media

There are currently two installation methods available for initial installation of NetBSD on Apple Macintosh 68000-based systems. Neither supports all installation media types at this time, so the one you select must be compatible with the media you have available on your system.

- The **`sysinst`** method of installation uses an Installation Kernel which is a minimal NetBSD system with a memory resident set of utilities that are capable of partitioning the disk, initializing the file systems, and loading them from the archive files. Since the installation kernel does not currently support access to Mac OS HFS file systems this method requires that the Binary Distribution Sets be accessible from CD-ROM, remote NFS partition, or via FTP access.
- The Traditional method of installation uses Mac OS hosted utilities to partition your disk, initialize the partitions for use by NetBSD, and load the file systems from archive files stored on the Mac OS HFS file system. This method requires that the Binary Distribution Sets reside on a local Macintosh hard drive, a CD-ROM, or an AppleShare volume.

The Traditional method of installation is currently supported from the local Macintosh hard drive, from a CD-ROM, or from an AppleShare volume (however, you may ugrade a system from within NetBSD; see the **Upgrading a previously-installed NetBSD System** section for more details). If you are installing from a local hard drive, this means that you'll need at least enough room for the largest file that you will have to install. This is the 15.2 MB `base.tgz` file. There has been talk of allowing an install from split files. If you have the time, desire, and knowledge, please feel free to add that functionality.

If the install is being done from an AppleShare-mounted volume, the NetBSD/mac68k Installer must be in the same folder as the binary distribution sets.

Each distribution file is in *raw* archive format.

- Distribution files must be downloaded in binary mode. Common web browsers may not be suitable for this task; FTP clients such as Fetch and Anarchie work fine, but be sure to specify a binary file transfer.
- The files should not be unpacked. If you have the Internet Config extension installed, you can disable this in the "Helpers" dialog by removing the entry associated with ".tgz" files. Other FTP clients may require separate changes; consult your package's documentation.
- If you are installing from a CD-ROM, the distribution sets are already in the proper format. No special handling is required.

You will also need to collect the Mac OS installation tools from the `mac68k/installation/misc` subdirectory of the NetBSD 4.0.1 distribution: Mkfs, NetBSD/mac68k Installer, and NetBSD/mac68k Booter. These three are in BinHexed, self-extracting archives as Mkfs.sea.hqx, Installer.sea.hqx, and Booter.sea.hqx, respectively. Extract them as you would any other Macintosh application.

Note: The Booter is the only Mac OS application needed if the **`sysinst`** method of installation is used. The Traditional method of installation is deprecated and will be removed in a future release, and the Mkfs and Installer tools will be retired.

Preparing your System for NetBSD installation

No matter which installation method you use, there is some planning and preparation that is required beforehand. First and foremost, before beginning the installation process, make sure you have a reliable backup of any data on your hard disk that you wish to keep. *Mistakes in partitioning your hard disk may lead to data loss.*

NetBSD/mac68k uses the same disk mapping scheme as Mac OS: the Apple Disk Partition Map. This permits both systems to reside on the same disk, but introduces some installation problems unique to the Macintosh. There are very few, if any, reliable ways to reduce the size of an existing Mac OS disk partition, so partitioning a disk that currently contains Mac OS will almost always require a backup and reload step under Mac OS.

If you are using the **sysinst** method of installation you will be able to do most, if not all, of your disk partitioning during the install process. Partitioning the disk with **sysinst** will destroy any partition that is resized, deleted, converted, or designated for use by NetBSD. All space not planned to be used for Mac OS HFS partitions may be used by NetBSD and can be sub-divided by the **sysinst** process. This space may be defined within one or more existing disk partitions of any type, including HFS partitions that are no longer needed for Mac OS. However it is best if this space is physically contiguous on the disk as **sysinst** is not capable of merging non-contiguous disk partitions. If you are using the **sysinst** method and have sufficient disk space in one or more disk partitions you should skip forward to the section labeled **Installing the NetBSD System (Sysinst Method)** in this document.

If you are using the Traditional method of installation you must use a disk partitioning utility to designate the different partitions you will want in your final NetBSD configuration. It is not necessary to create NetBSD (or AU/X) type partitions at this stage; the Mkfs utility can convert a partition of any type to one usable for NetBSD.

If disk partitioning is required because you've selected the Traditional method of installation, or because disk space needs to be freed up for use for the **sysinst** method of installation, follow the directions in the remainder of this section.

Find your favorite disk partitioning utility. Any formatter capable of partitioning a SCSI disk should work. Some of the ones that have been tried and seem to work are:

- Apple HD SC Setup
- Hard Disk ToolKit from FWB
- SCSI Director Lite
- Disk Manager Mac from OnTrack
- Silverlining from LaCie
- APS Disk Tools

Apple's HD SC Setup is probably the easiest to use and the most commonly available. Instructions for patching HD SC Setup so that it will recognize non-Apple drives is available at:

http://www.euronet.nl/users/ernstoud/patch.html

First, you need to choose a drive on which to install NetBSD. Try to pick a drive with a low SCSI target number (or "SCSI ID"), especially if you are likely to add or remove drives to your SCSI chain in the future.

Note: Be certain you have a reliable backup of any data that you may want to keep. Repartitioning your hard drive is an excellent way to destroy important data.

Second, decide how you want to set up your partitions. At minimum, you need a partition to hold the NetBSD installation (the root partition — `/`) and a partition to serve as swap space. You may choose to use more than one partition to hold the installation. This allows you to separate the more vital portions of the file system (such as the kernel and the `/etc` directory) from the more volatile parts of the file system. Typical setups place the `/usr` directory on a separate partition from the root partition (`/`). Generally, `/` can be fairly

small while the `/usr` partition should be fairly large. If you plan to use this machine as a server, you may also want a separate `/var` partition.

Once you have decided how to lay out your partitions, you need to calculate how much space to allocate to each partition. A minimal install of NetBSD (i.e. `base.tgz`, `etc.tgz`, and either `kern-GENERIC.tgz` or `kern-GENERICSBC.tgz`) should just fit in a 56 MB partition. For a full installation, you should allocate at least 180 MB (150 MB if you do not wish to install the X sets). A general rule of thumb for sizing the swap partition is to allocate twice as much swap space as you have real memory. Having your swap + real memory total at least 20 MB is also a good idea. Systems that will be heavily used or that are low on real memory should have more swap space allocated. Systems that will be only lightly used or have a very large amount of real memory can get away with less.

Keep in mind that NetBSD currently requires Mac OS in order to boot, so it is likely that you will want to keep at least a minimal install of Mac OS around on an HFS partition for this purpose. The size of this partition may vary depending on the size requirements for the version of Mac OS you are using. Of course, if you have Mac OS on another hard drive or can boot from a floppy, feel free to dedicate the entire drive to NetBSD.

Next, use your favorite partitioning utility to make partitions of the necessary sizes. You can use any type of partition, but partitions of type `Apple_Free` might save you some confusion in the future.

You are now set to install NetBSD on your hard drive.

Installing the NetBSD System (Sysinst Method)

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 hardware-specific issues*

   - *SCSI driver problems*

     The SCSI driver used in the kernel on many older Macintosh systems is, by default, the ncrscsi driver. It contains a recognized but as yet unfixed bug that affects some disk drive/controller combinations, usually Quantum disks. Under heavy load these systems may hang or corrupt the file system; or, you may experience frequent *Segmentation fault* and *Illegal instruction* errors that may or may not be consistently repeatable. This latter condition is particularly prevalent on systems with minimal RAM installed.

     If either of these problems occur on your system you are advised to use the SBC variants of the Kernel and Installation Kernel. However, be aware that this issue does not affect e.g. Centris or Quadra systems.

   - *The 68LC040 processor*

     NetBSD has known but unresolved problems running on the 68LC040 processor, the variant of the 68040 that does not contain the floating point unit (FPU). The kernel is thus forced to emulate the missing operations in software. Unfortunately the 68LC040 processor has a design problem that causes the emulation to fail intermittently. We hope to provide a solution for this issue in a future NetBSD release.

     Software emulation of floating point operations is not a problem on the 68020 and 68030 processors.

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).
     - The Mac OS Booter application and an Installation Kernel
     - A Macintosh with a 68020 and MMU, 68030 or 68RC040 processor. An FPU is not required but will be used if present (but see the note above regarding the 68LC040).
     - A CD-ROM drive (SCSI), a hard disk and a minimum of 8 MB of memory installed.
     - The hard disk should have at least 120 + *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 NetBSD Boot Tools folder.
     - Create a Folder on your Mac OS disk for the NetBSD/mac68k components.
     - Copy the Booter application into the newly created Folder. Expand the file if necessary to create the Mac OS executable.
     - Copy the Installation Kernels into the newly created Folder. It is not necessary to **gunzip** compressed kernel files.
     - Single-click on the Booter application icon then select the "Get Info" from the File Menu list. Increase the memory allocation for the Booter to as much as possible for your system. Having a large number of fonts, extensions or sounds installed on your system can cause memory exhaustion problems for the Booter if you don't do this. Also, the extra memory is needed by the Booter to expand compressed kernels while booting.
   - The Quick Installation
     - Double-click on the Booter application icon to start executing it. From the **Options** pull-down menu, select **Monitors**, then select **Change Monitor Depth** and make sure BW is highlighted. Close the window using the **Close** button.
     - From the **Options** pull-down menu select **Boot Options**. This will bring up an option panel. Set the **Auto-set GMT Bias** checkbox in the lower left and then select the **Boot from Mac OS** option at the top of the window. The **Set** button on the right will become active. Use it to locate and select the Installation Kernel file appropriate for your hardware. This will be either netbsd-INSTALL.gz or netbsd-INSTALLSBC.gz. Close the window using the **Close** button.

- From the **Options** pull-down menu select the **Boot Now**, or use the Apple-B (Command-B) key combination to start the NetBSD boot process. Do not move the mouse while the boot operation is in progress as this may leave the keyboard locked to NetBSD.

  The main menu will be displayed.

```
.*********************************************.
* NetBSD-4.0.1 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.**
- 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*

   Prior to attempting to boot NetBSD/mac68k verify that all the following are done:

   - Enable 32-bit addressing in the Memory Control Panel [1].
   - Disable all forms of virtual memory (the Memory Control Panel, RAM Doubler, or other software-based memory enhancement products).
   - Place the system in BW Mode (1-bit color or grayscale) as shown in the Monitors Control Panel or in the Monitors options dialog of the Booter. You may choose to have the Booter do this for you automatically by selecting the appropriate check box and radio button in the **Monitors** dialog on the **Options** menu.

   It is probably best to boot your machine with all extensions turned off [1]. You can do this by booting into Mac OS with the SHIFT key held down. You may have to restart your Macintosh for changes to take effect before proceeding.

[1] If your version of the Memory control panel does not have a 32-bit addressing mode radio button, this means that your system is already 32-bit clean and is running in 32-bit addressing mode by default. If the Booter complains that your are not in 32-bit mode, it may be necessary for you to press the "Use Defaults" button in the Memory control panel to restore 32-bit addressing. You should probably reboot after doing so. If you have an older II-class system (including the II, IIx, IIcx, and SE/30), it is necessary to install Connectix's MODE32 to work around ROM issues which prevent you from enabling 32-bit addressing. Please see the NetBSD/mac68k FAQ at

http://www.NetBSD.org/ports/mac68k/faq/

for more information.

Double-click on the NetBSD/mac68k Booter icon to start the application. Select **Booting** from the **Options** menu. Select the Kernel Location to be from Mac OS with the filename corresponding to the name of the Installation Kernel you are using. Typically this will be netbsd-INSTALL.gz.

If you haven't already put your Macintosh into BW mode, select the **Monitor Options** from the **Options** menu and check the box for BW mode.

Try booting NetBSD by selecting **Boot Now** from the **Options** menu.

If the system does not come up, send mail to `port-mac68k@NetBSD.org` describing your software, your hardware, and as complete a description of the problem as you can. As an alternative, try using the Traditional method of installation described in the next section.

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*

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. *Preparing a disk for* Mac OS *and NetBSD*

sysinst can manipulate the Apple Disk Partition Map allowing you to partition your disk for use with NetBSD. It does not support resizing existing Mac OS HFS disk partitions. If there is insufficient Free space on the disk to support an installation of NetBSD you will need to backup, repartition and restore your existing Mac OS partitions before proceeding. You may choose to use a Traditional method of creating disk partitions for NetBSD if you wish. They can still be used by **sysinst** for a NetBSD installation.

8. *Installation drive selection and parameters*

To start the installation, select **Install NetBSD to hard disk** from the main menu. To start the installation, select the menu option in install NetBSD from the main menu.

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

sysinst next tries to figure out how the selected volume has been partitioned. It does this by reading the Apple Disk Partition Map from the disk. If the disk does not have a Partition Map, **sysinst** will give you the option of writing one, but doing so will not make the disk a Mac OS bootable volume. You will have the option of creating HFS partitions that may be subsequently initialized and used under Mac OS though.

9. *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.

   - Definition of the NetBSD disklabel.

     The Apple Disk Partition Map is used to create an in-core map of the disk called the disklabel. A minimum of two NetBSD partitions will be required, one for root and one for swap. Up to eight partitions may be used by NetBSD. Up to 32 partitions may exist on the disk which can be any combination of Mac OS HFS, Free, Scratch and NetBSD partitions, although only the first eight which meet the needs of NetBSD will be seen and mapped to the NetBSD disklabel.

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

   - Editing the NetBSD Disklabel (and the underlying Apple Disk Partition Map).

     You will be presented with the current layout of the disk as seen by NetBSD, and given a change to change it. (Even though NetBSD can only use the first eight qualified partitions, all partitions found on the disk will be displayed.) The partitions found on the disk will be shown in the top section of the display. Each will be identified with the name assigned by NetBSD, the current size, offset, type, use and mount point. The partition currently being modified will be highlighted in inverse video. The bottom part of the display will list the operations which may be performed on the selected (highlighted) partition. The options are:

     – Select next partition

       This highlights the next partition in the upper display list and makes it the current one selected for manipulation.

     – Change selected partition

       This changes the type assigned to the partition. A partition may be assigned for use as a NetBSD Root, SWAP, Usr, or Root&Usr; it may be assigned for use as a Mac OS HFS partition; a Scratch (for later reassignment); or a Free partition. Free partitions which are physically adjacent to each other will be collapsed into a single Free partition.

     – Set mount point for partition

       This designates the NetBSD file system mount point for the partition, and gets transferred into the /etc/fstab definition so NetBSD knows where to mount the file system on subsequent boots. The option only applies to NetBSD Root, Usr, Root&Usr or Mac OS HFS partitions, although currently HFS access is not supported without optional software components. A common set of predefined mount points (/usr, /home, /var, /tmp or None) will be presented to you to assist you in defining the most commonly used, but you may enter you own names if you choose. Selecting "None" will clear the mount point name and keep the partition from being defined in the resulting `/etc/fstab` file.

     – Split selected partition

       This option divides the selected partition into two separate partitions if there is space available in the Disk Partition Map. You will be prompted for the size of the first segment and the remaining portion will be allocated to the second segment. The first segment will be designated as a

Scratch type, and the second will be designated as a Free type. To clear a split, or remerge two adjacent partitions into a single one, change both to be Free types. **sysinst** will merge them and update the display.

This is the primary option used to partition the disk since it allows you to sub-divide the selected partition into two partitions. Changing the types associated with the resulting two parts, or splitting the second part further sub-divides the original partition.

– Page Up, Page Down

These entries allow you to scroll the upper display if more than eight partitions currently exist on the disk.

– Fix selected partition

This option reviews the partition's size and starting address and fixes the values if they overlap any adjacent partition. This is primarily a debugging option and shouldn't be necessary during a normal installation. However, some 3rd party disk formatters have been known to create bogus entries in the Apple Disk Partition Map, and this option can aid is repairing these entries.

– Exit

This option completes the disk partitioning and returns you to the previous installation menu. At that point you will be given one last opportunity to bail out before committing the changes to the Disk Partition Map recorded on the disk.

- Recommended approach to partitioning

The simplest method of approaching disk partitioning with **sysinst** is to convert everything that can be used for NetBSD into a Free type partition. This will allow **sysinst** to collapse and merge all the available space. Then cycle through the Select, Split, Select, Change, and Set Mount Point options for each of the NetBSD partitions that are desired. Since NetBSD/mac68k has a very specific mount order for partitions during system boot, it is best to create your NetBSD partitions in the following order: Root, SWAP, Usr where the partitions will be mounted in order on 'a', 'b', 'g', 'd', 'e', 'f', and 'h'.

At least one Root or Root&Usr is required, and a SWAP partition is highly desirable. As a general rule you will need twice as much swap space as you have RAM, more if you plan on running X, Web applications or doing heavy development in a multi-user environment. The Root partition, if it is separate from your Usr, usually requires about 24 MB.

If multiple Root partitions are defined, the second is usually mounted on /altusr by default. NetBSD/mac68k automatically mounts all Root partitions after the first as Usr type partitions. However, it is best to be very specific about mount points and partition type and use.

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 Mac OS file systems*

    NetBSD/mac68k does not currently have in-kernel support for Mac OS HFS/HFS+ or AppleShare filesystems. **sysinst** therefore can not access the file sets if they are on these filesystems.

16. *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.

17. *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.

18. *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. Alternatively, you will be asked if you wish to have a progress bar. This is the preferred option as it shows progress without significantly slowing down the installation process.

    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 then set a password for the "root" account, to prevent the machine coming up without access restrictions.

19. *Finalizing your installation*

    Congratulations, you have successfully installed NetBSD 4.0.1. You can now reboot the machine, and boot NetBSD from hard disk.

Installing the NetBSD System (Traditional Method)

The Traditional method of installation can be broken down into three basic steps:

- Run Mkfs to build a file system or file systems.
- Run the Installer to load the files onto your file systems.
- Run the Booter to boot the system.

Preparing the file system(s)

Double-click on the Mkfs application icon to start it up. It will ask you for the SCSI-ID (SCSI target number) of the drive that you are installing NetBSD on. Once this is selected, it will present a list of the partitions on that disk. You must first convert the partitions to a type which NetBSD can understand. Select each partition on which you wish to build a file system and click on the **`Change`** button. If you are placing the entire installation on a single partition, select the **`NetBSD RootUsr`** radio button. If you are using multiple partitions, select **`NetBSD Root`** for the root partition (**`/`**) and **`NetBSD Usr`** for all the other partitions. You should select **`NetBSD Swap`** for the swap partition.

When you have finished converting each partition, select each partition and click on the **`Format`** button. You will now be asked for a bunch of parameters for the hard drive and the file system. Usually, you can just take the defaults. If you are installing onto removable media (e.g. a Zip, Jaz, or Syquest), please see the FAQ. Note that although this dialog only has the **`OK`** button, you are not committed, yet. Once you get the values

you want, press the **OK** button. A dialog will be presented at this point with two options: **Format** and **Cancel**. If you choose **Cancel**, nothing will be written to your drive. If you choose **Format**, the program will proceed to make a file system.

Mkfs is not a well-behaved Macintosh application. It will not allow any other tasks to run while it does (cooperative multitasking at its best). When it's finished, the program will put up a dialog to ask if you have scanned the output for any error messages. Usually there won't have been any errors, but do scan the output to make sure. Simply click on the **I Read It** button and the program will quit.

Repeat as necessary for any extra partitions that you wish to make file systems on. Note that you do *not* need a file system on your swap partition.

When you are finished, click on the **Done** button and choose **Quit** from the **File** menu to exit Mkfs.

Installing the files

Before using the Installer, it is probably a good idea to increase its memory allocation. Select the Installer icon by clicking on it and choose **Get Info** from the File menu. Increase both the Minimum and Preferred sizes to as much as you can spare.

Double-click on the Installer icon to start it up. The Installer will present the same SCSI-ID menu that Mkfs did. Select the same SCSI-ID (SCSI target number) that you did for Mkfs - i.e., the one you are installing NetBSD on.

If you are installing onto a single root partition (`/`), proceed to the **Installation of base files** section, below.

If you have not created file systems for `/` (root), `usr`, and any other file systems, go back to **Preparing the file system(s)** above.

When you started the Installer, it mounted your root partition (`/`). Just before it printed

```
Mounting partition 'A' as /
```

it printed lines like:

```
sd1 at scsi ID 5
```

This means that the device for SCSI target 5 ("SCSI ID 5") is `sd1`. The partitions are signified by a trailing letter. For instance, `sd1a` would be the root partition (`/`) of the second SCSI disk in the chain, and `sd0g` would be the first Usr partition on the first SCSI disk.

You will need to know the proper device to mount the remaining partition(s) by hand:

1. Select **Build Devices** from the **File** menu.
2. Select **Mini Shell** from the **File** menu.
3. You can use the **disklabel** command to get a listing of the available partitions and their types and sizes.
4. Create the directory mount point(s) with the command:

   ```
   # mkdir path
   ```

   E.g. for the `/usr` partition type:

   ```
   # mkdir /usr
   ```

5. Mount the file systems you wish with the command:

   ```
   # mount device path
   ```

   For example, if you wish to mount a `/usr` partition from the first SCSI disk `sd0`, on `/usr`, you would type:

```
# mount /dev/sd0g /usr
```

6. Type
   ```
   # fstab force
   ```
   to create a proper `/etc/fstab` file.

7. Type **quit** after you have mounted all the file systems.

Installation of base files

Select the **Install** menu item from the **File** menu and install `base.tgz`, `etc.tgz`, either `kern-GENERIC.tgz` or `kern-GENERICSBC.tgz`, and any other sets you wish to install at this time (see the **NetBSD 4.0.1 Release Contents** for information about what's in each set). The Installer will print out the filename of each file as it is installed, and will take quite some time to install everything (the base package alone can take over two hours on a slow hard drive).

As is the case with Mkfs, this is not a particularly well-behaved Macintosh application and the machine will be completely tied up while the installation takes place.

At some point after installing the base set, select the **Build Devices** option from the **File** menu if you have not already done so. This will create a bunch of device nodes for you and will create your initial `/etc/fstab`. The Installer program also has an option to give you a mini-shell. Do not use this unless you are sure know what you are doing.

When you are finished installing all of the sets you wish to install, exit the Installer by choosing **Quit** from the **File** menu.

Booting the system

Prior to attempting to boot NetBSD/mac68k, please verify that all of the following are true:

1. 32-bit addressing is enabled [2] in the Memory control panel;
2. All forms of virtual memory are disabled (the Memory control panel, RAM Doubler, or other software-based memory enhancement products); and
3. Your system is in BW mode (1-bit color or grayscale) as shown by the Monitors control panel. You may choose to have the Booter do this for you automatically by selecting the appropriate check box and radio button in the **Monitors** dialog on the **Options** menu.

It is probably best to boot your machine with all extensions turned off [2]. You can do this by booting into Mac OS with the SHIFT key held down. You may have to restart your Macintosh for changes to take effect before proceeding.

[2] If your version of the Memory control panel does not have a 32-bit addressing mode radio button, this means that your system is already 32-bit clean and is running in 32-bit addressing mode by default. If the Booter complains that you are not in 32-bit mode, it may be necessary for you to press the **Use Defaults** button in the Memory control panel to restore 32-bit addressing. You should probably reboot after doing so. If you have an older II-class system (including the II, IIx, IIcx, and SE/30), it is necessary to install Connectix's MODE32 to work around ROM issues which prevent you from enabling 32-bit addressing. Please see the NetBSD/mac68k FAQ at
http://www.NetBSD.org/ports/mac68k/faq/
for more information.

Double-click on the NetBSD/mac68k Booter icon to start the application. Select **Booting** from the **Options** menu. Check that all of the items in the resulting dialog look sane - especially the SCSI target number. If not, correct them to your preference (the SCSI target number, or "SCSI ID", should be the only thing you need to change). When you are satisfied with your choices, try booting NetBSD by selecting **Boot Now** from the **Options** menu.

If you wish to save your preferences, choose **Save Options** from the **File** menu before Booting (your preferences will not be saved if you forget to do this).

If the system does not come up, send mail to port-mac68k@NetBSD.org describing your software, your hardware, and as complete a description of the problem as you can.

If the system does come up, congratulations, you have successfully installed NetBSD 4.0.1.

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, where your on-board, NuBus or PDS interface may be ae0, mc0 or sn0. For example:

   ```
   ifconfig_sn0="inet 123.45.67.89 netmask 255.255.255.0"
   ```

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

   ```
   ifconfig_sn0="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/docs/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/pkgsrc/packages/NetBSD/
     usually in the mac68k/4.0/All subdir. You can install them with the following commands under sh(1):

     ```
     # PKG_PATH=ftp://ftp.NetBSD.org/pub/pkgsrc/packages/NetBSD/mac68k/4.0/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 firefox
     ...
     ```

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

```
# setenv PKG_PATH ftp://ftp.NetBSD.org/pub/pkgsrc/packages/NetBSD/mac68k/4.0/Al
...
```

Note: Some mirror sites don't mirror `/pub/pkgsrc` directory. If you would like to use such mirrors, you could also try the `/pub/NetBSD/packages/current-packages/NetBSD/mac68k/4.0/All` directory which may have the same contents.

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

Note: In some case the `pkg_add`(1) command will complain about a version mismatch of packages with a message like the following:

```
Warning: package 'foo' was built for a different version
of the OS:
NetBSD/i386 4.0 (pkg) vs. NetBSD/i386 4.0.1 (this host),
```

This warning would be harmless if the formal major release numbers are same between the pkg and your host. Please refer "the NetBSD release glossary and graphs":

http://www.NetBSD.org/releases/release-map.html

for details of the release numbering scheme of NetBSD.

– Package sources for compiling packages on your own can be obtained by retrieving the file

ftp://ftp.NetBSD.org/pub/NetBSD/packages/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 `doc/pkgsrc.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/postfix/main.cf` file will almost definitely need to be adjusted. If you prefer a different MTA, then install it using the NetBSD package system or by hand and 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 4.0.1 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 boot from the installer kernel using one of the methods described above. 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 4.0.1 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/mac68k 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 4.0.1.

N.B. when using **sysinst** for upgrading, it will automatically invoke

```
postinstall fix
```

and thus all issues that are fixed by **postinstall** by default (see below) will be handled.

Issues affecting an upgrade from NetBSD 2.1 and older releases.

See the section below on upgrading from NetBSD 3.x as well.

It is **very important** that you populate the directory `/etc/pam.d` with appropriate configuration files for Pluggable Authentication Modules (PAM) because you will not be able to login any more otherwise. Using *postinstall* as described below will take care of this. Please refer to

http://www.NetBSD.org/docs/guide/en/chap-pam.html

for documentation about PAM.

The following issues can generally be resolved by running *postinstall* with the **etc** set :

```
postinstall -s /path/to/etc.tgz check
postinstall -s /path/to/etc.tgz fix
```

Issues fixed by *postinstall*:

- Various files in `/etc` need upgrading. These include:
  - `/etc/defaults/*`
  - `/etc/mtree/*`
  - `/etc/pam.d/*`
  - `/etc/daily`
  - `/etc/weekly`
  - `/etc/monthly`
  - `/etc/security`
  - `/etc/rc.subr`
  - `/etc/rc`
  - `/etc/rc.shutdown`
  - `/etc/rc.d/*`

The following issues need to be resolved manually:

- The user '_pflogd' and the groups '_pflogd' and 'authpf' need to be created.

Issues affecting an upgrade from NetBSD 3.x releases.

The following issues can generally be resolved by running *postinstall* with the **etc** set :

```
postinstall -s /path/to/etc.tgz check
postinstall -s /path/to/etc.tgz 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/*`
  - `/etc/envsys.conf`

The following issues need to be resolved manually:

- The users '_proxy', '_rwhod', and '_sdpd' and the groups '_proxy', '_rwhod' and '_sdpd' need to be created and the user 'uucp' needs to be updated.
- A number of things have been removed from the NetBSD 4.0 release including: the evbsh5 port, the Fortran 77 compiler (g77), NETCCITT, NETNS, Sendmail, Sushi, UUCP, and Vinum. If you were using any of these, then please see the "Components removed from NetBSD" section near the beginning of this document.
- The replacement of Sendmail by Postfix can be handled automatically by *postinstall* but it is not done by default. If you want to transition to Postfix, the command

  ```
  postinstall -s /path/to/etc.tgz fix mailerconf
  ```

  will update your `/etc/mailer.conf` file to use Postfix as the MTA. When using **`sysinst`** to upgrade the system, it will ask if you want this to be done.

Note that if you have a customized Sendmail setup, you need to set up Postfix in an equivalent way; there is no tool for automatic conversion of Sendmail configuration to a Postfix one.

Postfix will be started up automatically when the system boots. You may see messages like "$sendmail is not set properly" at boot. You can suppress them by removing `/etc/rc.d/sendmail` and `/etc/rc.d/smmsp`. Those files and other parts of sendmail configuration like files under `/usr/share/sendmail` are not removed by default while upgrading for those who want to continue using sendmail from outside the base system. If you want to delete them, *postinstall* can be used:

```
postinstall -s /path/to/etc.tgz fix sendmail
```

Issues with GDB 6.5

Some architectures (arm, i386, powerpc and sparc64) have switched to a newer gdb version (6.5) in this release. Unfortunately support for debugging programs using the SA (scheduler activation) based thread library, is incomplete in this gdb version. Furthermore kernel crashdumps can not be debugged due to a missing identification in the kernel binaries.

Both issues have been addressed on the wrstuden-fixsa branch, but did not make it into the NetBSD release. Both will be fixed in the next patch release.

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-users@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 Luleå Academic Computer Society for providing the backup services server.
- The Columbia University Computer Science Department for hosting the NYC build cluster.
- The Western Washington University Computer Science Department for running the WWU build cluster that produces daily snapshots.
- The many organizations 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.
- We list the individuals and organizations that have made donations or loans of hardware and/or money, to support NetBSD development, and deserve credit for it at **http://www.NetBSD.org/donations/.** (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 be mentioned, tell us!)

We are...

(in alphabetical order)

The NetBSD core group:

Alistair Crooks	agc@NetBSD.org
Quentin Garnier	cube@NetBSD.org
Matt Thomas	matt@NetBSD.org
YAMAMOTO Takashi	yamt@NetBSD.org
Christos Zoulas	christos@NetBSD.org

The portmasters (and their ports):

Erik Berls	cyber@NetBSD.org	**cobalt**
Manuel Bouyer	bouyer@NetBSD.org	**xen**
Simon Burge	simonb@NetBSD.org	**evbmips**
Simon Burge	simonb@NetBSD.org	**pmax**
Simon Burge	simonb@NetBSD.org	**sbmips**
Julian Coleman	jdc@NetBSD.org	**atari**
Marcus Comstedt	marcus@NetBSD.org	**dreamcast**
Andrew Doran	ad@NetBSD.org	**amd64**
Andrew Doran	ad@NetBSD.org	**i386**
Matthias Drochner	drochner@NetBSD.org	**cesfic**
Gavan Fantom	gavan@NetBSD.org	**iyonix**
Jaime A Fournier	ober@NetBSD.org	**zaurus**
Matt Fredette	fredette@NetBSD.org	**sun2**
Ichiro FUKUHARA	ichiro@NetBSD.org	**hpcarm**
Chris Gilbert	chris@NetBSD.org	**cats**
Ben Harris	bjh21@NetBSD.org	**acorn26**
Ross Harvey	ross@NetBSD.org	**alpha**
Nick Hudson	skrll@NetBSD.org	**hp700**
Martin Husemann	martin@NetBSD.org	**sparc64**
IWAMOTO Toshihiro	toshii@NetBSD.org	**hpcarm**
Darrin Jewell	dbj@NetBSD.org	**next68k**
Søren Jørvang	soren@NetBSD.org	**sgimips**
Wayne Knowles	wdk@NetBSD.org	**mipsco**
Takayoshi Kochi	kochi@NetBSD.org	**ia64**
Paul Kranenburg	pk@NetBSD.org	**sparc**
Michael Lorenz	macallan@NetBSD.org	**macppc**
Anders Magnusson	ragge@NetBSD.org	**vax**
Cherry G. Mathew	cherry@NetBSD.org	**ia64**
NISHIMURA Takeshi	nsmrtks@NetBSD.org	**x68k**
Tohru Nishimura	nisimura@NetBSD.org	**luna68k**
Tohru Nishimura	nisimura@NetBSD.org	**sandpoint**
Andrey Petrov	petrov@NetBSD.org	**sparc64**
Scott Reynolds	scottr@NetBSD.org	**mac68k**
Tim Rightnour	garbled@NetBSD.org	**ofppc**
Tim Rightnour	garbled@NetBSD.org	**prep**
Tim Rightnour	garbled@NetBSD.org	**rs6000**
Noriyuki Soda	soda@NetBSD.org	**arc**
Ignatios Souvatzis	is@NetBSD.org	**amiga**

Jonathan Stone	jonathan@NetBSD.org	**pmax**
Shin Takemura	takemura@NetBSD.org	**hpcmips**
Matt Thomas	matt@NetBSD.org	**alpha**
Matt Thomas	matt@NetBSD.org	**netwinder**
Jason Thorpe	thorpej@NetBSD.org	**algor**
Jason Thorpe	thorpej@NetBSD.org	**evbarm**
Jason Thorpe	thorpej@NetBSD.org	**shark**
Izumi Tsutsui	tsutsui@NetBSD.org	**ews4800mips**
Izumi Tsutsui	tsutsui@NetBSD.org	**hp300**
Izumi Tsutsui	tsutsui@NetBSD.org	**news68k**
Valeriy E. Ushakov	uwe@NetBSD.org	**landisk**
Nathan Williams	nathanw@NetBSD.org	**sun3**
Steve Woodford	scw@NetBSD.org	**evbppc**
Steve Woodford	scw@NetBSD.org	**mvme68k**
Steve Woodford	scw@NetBSD.org	**mvmeppc**
Reinoud Zandijk	reinoud@NetBSD.org	**acorn32**

The NetBSD 4.0.1 Release Engineering team:

Grant Beattie	grant@NetBSD.org
Manuel Bouyer	bouyer@NetBSD.org
David Brownlee	abs@NetBSD.org
James Chacon	jmc@NetBSD.org
Julian Coleman	jdc@NetBSD.org
Håvard Eidnes	he@NetBSD.org
Liam J. Foy	liamjfoy@NetBSD.org
John Heasley	heas@NetBSD.org
Geert Hendrickx	ghen@NetBSD.org
Soren Jacobsen	snj@NetBSD.org
Daniel de Kok	daniel@NetBSD.org
Phil Nelson	phil@NetBSD.org
Jeff Rizzo	riz@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
Alan Barrett	apb@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
Stephen Borrill	sborrill@NetBSD.org
Sean Boudreau	seanb@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
Pavel Cahyna	pavel@NetBSD.org
D'Arcy J.M. Cain	darcy@NetBSD.org
Daniel Carosone	dan@NetBSD.org
Dave Carrel	carrel@NetBSD.org
James Chacon	jmc@NetBSD.org
Mihai Chelaru	kefren@NetBSD.org
Bill Coldwell	billc@NetBSD.org
Julian Coleman	jdc@NetBSD.org
Ben Collver	ben@NetBSD.org
Marcus Comstedt	marcus@NetBSD.org
Jeremy Cooper	jeremy@NetBSD.org
Chuck Cranor	chuck@NetBSD.org
Alistair Crooks	agc@NetBSD.org
Aidan Cully	aidan@NetBSD.org
Garrett D'Amore	gdamore@NetBSD.org
Johan Danielsson	joda@NetBSD.org
John Darrow	jdarrow@NetBSD.org
Jed Davis	jld@NetBSD.org
Matt DeBergalis	deberg@NetBSD.org
Arnaud Degroote	degroote@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
Jaime A Fournier	ober@NetBSD.org
Stoned Elipot	seb@NetBSD.org

Michael van Elst	mlelstv@NetBSD.org
Enami Tsugutomo	enami@NetBSD.org
Bernd Ernesti	veego@NetBSD.org
Erik Fair	fair@NetBSD.org
Gavan Fantom	gavan@NetBSD.org
Hauke Fath	hauke@NetBSD.org
Hubert Feyrer	hubertf@NetBSD.org
Jason R. Fink	jrf@NetBSD.org
Matt J. Fleming	mjf@NetBSD.org
Marty Fouts	marty@NetBSD.org
Liam J. Foy	liamjfoy@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
Brian Ginsbach	ginsbach@NetBSD.org
Paul Goyette	pgoyette@NetBSD.org
Michael Graff	explorer@NetBSD.org
Brian C. Grayson	bgrayson@NetBSD.org
Matthew Green	mrg@NetBSD.org
Andreas Gustafsson	gson@NetBSD.org
Ulrich Habel	rhaen@NetBSD.org
Jun-ichiro itojun Hagino	itojun@NetBSD.org
HAMAJIMA Katsuomi	hamajima@NetBSD.org
Adam Hamsik	haad@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
Geert Hendrickx	ghen@NetBSD.org
René Hexel	rh@NetBSD.org
Iain Hibbert	plunky@NetBSD.org
Kouichirou Hiratsuka	hira@NetBSD.org
Michael L. Hitch	mhitch@NetBSD.org
Ádám Hóka	ahoka@NetBSD.org
Jachym Holecek	freza@NetBSD.org
David A. Holland	dholland@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
Roland Illig	rillig@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
Nicolas Joly	njoly@NetBSD.org
Chris Jones	cjones@NetBSD.org
Søren Jørvang	soren@NetBSD.org
Takahiro Kambe	taca@NetBSD.org
Masanori Kanaoka	kanaoka@NetBSD.org
Antti Kantee	pooka@NetBSD.org
Frank Kardel	kardel@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
John Klos	jklos@NetBSD.org
Wayne Knowles	wdk@NetBSD.org
Takayoshi Kochi	kochi@NetBSD.org
John Kohl	jtk@NetBSD.org
Daniel de Kok	daniel@NetBSD.org
Jonathan A. Kollasch	jakllsch@NetBSD.org
Paul Kranenburg	pk@NetBSD.org
Lubomir Kundrak	lkundrak@NetBSD.org
Jochen Kunz	jkunz@NetBSD.org
Martti Kuparinen	martti@NetBSD.org
Kentaro A. Kurahone	kurahone@NetBSD.org
Arnaud Lacombe	alc@NetBSD.org
Kevin Lahey	kml@NetBSD.org
David Laight	dsl@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
Tonnerre Lombard	tonnerre@NetBSD.org
Mike Long	mikel@NetBSD.org
Michael Lorenz	macallan@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
MAEKAWA Masahide	gehenna@NetBSD.org
Anders Magnusson	ragge@NetBSD.org
Cherry G. Mathew	cherry@NetBSD.org
David Maxwell	david@NetBSD.org
Gregory McGarry	gmcgarry@NetBSD.org
Dan McMahill	dmcmahill@NetBSD.org
Jared D. McNeill	jmcneill@NetBSD.org
Neil J. McRae	neil@NetBSD.org
Julio M. Merino Vidal	jmmv@NetBSD.org
Perry Metzger	perry@NetBSD.org
Luke Mewburn	lukem@NetBSD.org
Brook Milligan	brook@NetBSD.org
Minoura Makoto	minoura@NetBSD.org
Simas Mockevicius	symka@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
John Nemeth	jnemeth@NetBSD.org
Bob Nestor	rnestor@NetBSD.org
NISHIMURA Takeshi	nsmrtks@NetBSD.org
Tohru Nishimura	nisimura@NetBSD.org
NONAKA Kimihiro	nonaka@NetBSD.org
Takehiko NOZAKI	tnozaki@NetBSD.org
Tobias Nygren	tnn@NetBSD.org
OBATA Akio	obache@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
Rui Paulo	rpaulo@NetBSD.org
Jonathan Perkin	sketch@NetBSD.org
Andrey Petrov	petrov@NetBSD.org
Herb Peyerl	hpeyerl@NetBSD.org
Matthias Pfaller	matthias@NetBSD.org
Chris Pinnock	cjep@NetBSD.org
Adrian Portelli	adrianp@NetBSD.org

Peter Postma	peter@NetBSD.org
Dante Profeta	dante@NetBSD.org
Chris Provenzano	proven@NetBSD.org
Niels Provos	provos@NetBSD.org
Mindaugas Rasiukevicius	rmind@NetBSD.org
Michael Rauch	mrauch@NetBSD.org
Marc Recht	recht@NetBSD.org
Darren Reed	darrenr@NetBSD.org
Jeremy C. Reed	reed@NetBSD.org
Antoine Reilles	tonio@NetBSD.org
Tyler R. Retzlaff	rtr@NetBSD.org
Scott Reynolds	scottr@NetBSD.org
Michael Richardson	mcr@NetBSD.org
Tim Rightnour	garbled@NetBSD.org
Alan Ritter	rittera@NetBSD.org
Jeff Rizzo	riz@NetBSD.org
Hans Rosenfeld	hans@NetBSD.org
Gordon Ross	gwr@NetBSD.org
Steve Rumble	rumble@NetBSD.org
Ilpo Ruotsalainen	lonewolf@NetBSD.org
Heiko W. Rupp	hwr@NetBSD.org
Blair J. Sadewitz	bjs@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
Silke Scheler	silke@NetBSD.org
Karl Schilke (rAT)	rat@NetBSD.org
Amitai Schlair	schmonz@NetBSD.org
Konrad Schroder	perseant@NetBSD.org
Georg Schwarz	schwarz@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
Daniel Sieger	dsieger@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
Jörg Sonnenberger	joerg@NetBSD.org
Ignatios Souvatzis	is@NetBSD.org
T K Spindler	dogcow@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
Robert Swindells	rjs@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
Greg Troxel	gdt@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
Valeriy E. Ushakov	uwe@NetBSD.org
Todd Vierling	tv@NetBSD.org
Aymeric Vincent	aymeric@NetBSD.org
Paul Vixie	vixie@NetBSD.org
Mike M. Volokhov	mishka@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
Frank Wille	phx@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
Colin Wood	ender@NetBSD.org
Steve Woodford	scw@NetBSD.org
YAMAMOTO Takashi	yamt@NetBSD.org
Yuji Yamano	yyamano@NetBSD.org
David Young	dyoung@NetBSD.org
Reinoud Zandijk	reinoud@NetBSD.org

S.P.Zeidler	spz@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:

NetBSD is a registered trademark of The NetBSD Foundation, Inc.
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 Juan Romero Pardines for the NetBSD Foundation, Inc. and its contributors.
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/)
This 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 by Marshall M. Midden.
This product includes software developed or owned by Caldera International, Inc.
The Institute of Electrical and Electronics Engineers and The Open Group, have given us permission to reprint portions of their documentation.

In the following statement, the phrase ''this text'' refers to portions of the system documentation.

Portions of this text are reprinted and reproduced in electronic form in NetBSD, from IEEE Std 1003.1, 2004 Edition, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 6, Copyright (C) 2001-2004 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between these versions and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document.

The original Standard can be obtained online at http://www.opengroup.org/unix/online.html.

This notice shall appear on any product containing this material

This product includes software developed by the Alice Group.
This product includes software developed by Allen Briggs.
This product includes software developed by Bradley A. Grantham.
This product includes softwarre developed by Daishi Kato.
This product includes software developed by Takashi Hamada.
This product includes software developed by Scott Reynolds.
This product includes software developed by John P. Wittkoski.
This product includes software developed by Colin Wood.

The End

Contributions

The following people have made contributions of various sorts specifically for the Macintosh port (in alphabetical order):

- All of the users who have supplied us with good bug reports and moral support.
- The Alice Group (Allen K. Briggs, Chris P. Caputo, Michael L. Finch, Bradley A. Grantham, and Lawrence A. Kesteloot), without whom there would be no NetBSD port for the Macintosh.
- Steven R. Allen for keeping our snapshot distributions up-to-date.
- Stephen C. Brown for maintaining the Installer application.
- Denton Gentry and Yanagisawa Takeshi for their work on the SONIC Ethernet driver.
- Paul Goyette, Taras Ivanenko, Ken Nakata, and Michael R. Zucca for invaluable work towards supporting color X.
- Takashi Hamada and John Wittkoski beating the direct ADB hardware driver into submission.
- David Huang for getting MACE Ethernet and basic DMA working on the AV Macs.
- Scott Jann for acquiring a IIx and a IIci, used for building and testing release sets.
- Scott Kaplan for lending his IIci and Kensington Turbo Mouse for IIci/IIsi banked memory and internal video as well as non-Apple ADB devices.
- Noah M. Kieserman for lending a PowerBook 520C for tracking down several bugs on that platform.
- Markus Krummenacker for monetary donations.

- Glan Lalonde for an invaluable IIci page table dump.
- Dan McMahill for lending a PowerBook 165 to tweak ADB support on the PowerBook 160 and 180 family laptops.
- Bob Nestor for (unofficially) maintaining the Mkfs utility, and providing a lot of useful information about the ROM vectors used by different systems.
- Brad Parker for serial and Ethernet drivers/improvements.
- Brian R. Gaeke and Nigel Pearson for tweaking, polishing, and performing the occasional major refit on the Booter application.
- Scott Redman for lending Brad Grantham a PowerBook 160.
- Craig Ruff for assembling an '030 `pmove ttx` instruction.
- Brad Salai for lending an Ethernet card to help resolve interrupt conflicts.
- Larry Samuels for monetary donations.
- Peter Siebold for lending his IIvx in support of ADB and IIvx internal video.
- Glen Stewart for lending a Carrera040 accelerator which, while still unsupported, helped to track down memory management bugs for '030-based machines.
- Bill Studenmund for providing a stable front end to the machine- independent serial driver.
- Schuyler Stultz for the loan of his Macintosh II when we desperately needed another machine on which to compile and test during the '93 Xmas vacation.
- Tenon Intersystems for monetary donations, MachTen, and Brad's access to several machines and documentation after hours.
- Virginia Tech English Department for loan of a IIci w/ NuBus video and 32 MB of RAM — the first IIci to run NetBSD/mac68k.
- Rob Windsor for donating a variety of Macintosh II-family systems, a Centris 650, a Quadra 700, and several boxes full of miscellaneous peripherals and parts in the interest of ensuring adequate testing and working out minor (and not-so-minor) problems.
- Colin Wood for maintaining a host of NetBSD/mac68k documentation, including the FAQ, Meta-FAQ, and OS Info documents.