NAME

INSTALL — Installation procedure for NetBSD/evbarm.

CONTENTS

About this Document....2
What is NetBSD?....3
Changes Between The NetBSD 3.0.2 and 3.0.3 updates....3
  Supported devices....3
  Kernel....3
  Networking....3
  File system....3
  Security....4
  Miscellaneous....4
  alpha specific....4
  mac68k specific....4
  sparc specific....4
Changes Between The NetBSD 3.0.1 and 3.0.2 updates....4
  Kernel....5
  Networking....5
  File system....5
  Libraries....5
  Security....5
  Miscellaneous....5
  amd64 specific....5
  sparc specific....5
  mac68k specific....5
Changes Between The NetBSD 3.0 release and 3.0.1 update....5
  Kernel....6
  Networking....6
  File system....6
  Libraries....7
  Security....7
  Miscellaneous....7
  acorn26 specific....7
  amiga specific....8
  hp300 specific....8
  i386 specific....8
  m68k specific....8
  mips specific....8
  powerpc specific....8
  sparc specific....8
  xen specific....8
The Future of NetBSD....8
Sources of NetBSD....9
NetBSD 3.0.3 Release Contents....9
  NetBSD/evbarm subdirectory structure....10
  Binary distribution sets....11
NetBSD/evbarm System Requirements and Supported Devices....12
  Supported devices (Technologic Systems TS-7200)....12
    TS-5620 Real Time Clock....12

Serial ports....12
Ethernet adapters....13
Other on-board functions....13
Supported devices (ARM, Ltd. Integrator) 13
PrimeCell PL030 Real Time Clock....13
Serial ports....13
SCSI host adapters....13
Ethernet adapters....13
Supported devices (Intel IQ80310) 13
Serial ports....13
Ethernet adapters....13
i80312 Companion I/O functions....13
Supported devices (Intel IQ80321) 13
Serial ports....13
Ethernet adapters....14
i80321 I/O Processor functions....14
Supported devices (Team ASA Npwr) 14
Serial ports....14
SCSI host adapters....14
Ethernet adapters....14
Supported devices (Intel IXM1200) 14
Serial ports....14
Ethernet adapters....14
Supported devices (Samsung SMDK2800) 14
Serial ports....14
Supported devices (ADI BRH) 14
Serial ports....14
Ethernet adapters....14
Getting the NetBSD System on to Useful Media....14
Preparing your System for NetBSD installation....16
Preparation for the Technologic Systems TS-7200 17
Preparation for the Intel IQ80310 17
Preparation for the Intel IQ80321 19
Installing the NetBSD System....21
Installation for the Technologic Systems TS-7200 21
Installation for the Intel IQ80310 22
Installation for the Intel IQ80321 23
Post installation steps....25
Upgrading a previously-installed NetBSD System....27
Compatibility Issues With Previous NetBSD Releases....27
Issues affecting an upgrade from NetBSD 2.1 and older 27
Using online NetBSD documentation....28
Administrivia....29
Thanks go to....29
We are....35
Legal Mumbo-Jumbo....43
The End....47

DESCRIPTION

About this Document

This document describes the installation procedure for NetBSD 3.0.3 on the *evbarm* 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 3.0.3 release contains complete binary releases for many different system architectures. (A few ports are not fully supported at this time and are thus not part of the binary distribution. For information on them, please see the NetBSD web site at .: **http://www.NetBSD.org/**)

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

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

Changes Between The NetBSD 3.0.2 and 3.0.3 updates

The NetBSD 3.0.3 update is the third security/critical update of the NetBSD 3.0 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 3.2 etc), together with other less-critical fixes and feature enhancements.

Specific updates are as follows:

Supported devices

- `wd`(4): fix broken Seagate drives ST3160021A and ST3160811A.

Kernel

- Don't accept a compressed `vnd`(4) image with block size 0 to avoid a kernel panic.

Networking

- Fix get-lease-hostnames option in `dhcpd.conf`(5) to make `dhcpd`(8) use the DNS hostname as the DHCP hostname given to the client.
- Fix TCP window scaling in `ipfilter`(4).
- Update BIND to version 9.3.2.

File system

- Prevent a kernel panic when creating snapshots on filesystems which use quota's.
- Fix a coredump in `dump`(8) when using -X (filesystem internal snapshots).
- Fix a panic when one user has 65536 active vnodes on a quota-enabled file system.

Security

- Fix a buffer overflow in `glob`(3), which affected `ftpd`(8) and possibly other programs (SA2006-027).
- X11: fix an integer overflow in the Render and DBE extensions (SA2007-002).
- BIND: fix multiple Denial of Service vulnerabilities (SA2007-003).
- Fix a buffer overflow in `iso`(4) which could potentially lead to a local root compromise (SA2007-004).
- Fix an integer underflow in `file`(1) which could lead to an exploitable heap overflow.
- X11: fix a number of possible memory corruptions due to integer overflows (CVE 2007-1003, 1351, 1352, 1667).
- Fix a denial of service vulnerability in `racoon`(8) which could allow an attacker to disrupt a connection between IPSec peers.
- Disable processing or routing header type 0 (RH0) IPv6 packets since they can be used for DoS attacks (CVE-2007-2242). Provide a sysctl (net.inet6.ip6.rht0) to re-enable them. Note: the sysctl may be removed completely in future NetBSD releases.
- Two `veriexec`(4) flaws have been fixed: users can no longer rename a file to a veriexec protected file or run unfingerprinted files at strict level two or above.
- `file`(1) has been updated to version 4.21, including and integer underflow and an integer overflow fix (CVE-2007-1536 and CVE-2007-2799).

Miscellaneous

- Fix some problems with `etcupdate`(8) for upgrading.
- Update time zoneinfo to tzdata2007f.
- Set /etc/postfix/post-install mode 555 (executable) instead of 444.
- `passwd`(1): display a message indicating who's password is being changed, to avoid confusion after `su`(1).
- The /etc/daily, weekly, monthly and security scripts now use $TMPDIR instead of /tmp.

alpha specific

- Fix alignment problems causing regular panics on NetBSD/alpha and NetBSD/sparc.

mac68k specific

- Avoid a kernel panic before reboot.

sparc specific

- Fix alignment problems causing regular panics on NetBSD/alpha and NetBSD/sparc.

Changes Between The NetBSD 3.0.1 and 3.0.2 updates

The NetBSD 3.0.2 update is the second security/critical update of the NetBSD 3.0 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 3.1 etc), together with other less-critical fixes and feature enhancements.

Specific updates are as follows:

Kernel

- Avoid a panic in page fault handling that could occer under low-memory conditions.

Networking

- Changed the default `sshd_config`(8) to enable SSH version 2 only.
- Don't accept TCP connections to broadcast addresses.

File system

- Fixed race condition in NFS renaming that could cause the renamed file to be deleted.

Libraries

- When a user is created, initialize the string of secondary groups to avoid it containing garbage when none is added.

Security

- Fixed a buffer overflow in the in-kernel PPP code share by ISDN PPP interfaces `ippp`(4) and `pppoe`(4) (SA2006-019).
- X11: fixed an integer overflow in FreeType (SA2006-020).
- Fixed a potential DoS attack with `sendmail`(8) (SA2006-017).
- Fixed a DoS vulnerability in BIND (SA2006-022).
- Fixed a RSA signature forgery in `openssl`(1) (SA2006-023).
- X11: fixed a vulnerability in Adobe Type 1 font handling (SA2006-021).
- Fixed a number of DoS vulnerabilities in `openssl`(1) (SA in preparation, CVE entries: 2006-2937, 2940, 3738 and 4343).
- Fixed a number of DoS vulnerabilities in `sshd`(8) (SA in preparation, CVE entries: 2006-4924 and 5051).

Miscellaneous

- Fixed cross-building from GCC 4.x hosts.

amd64 specific

- Make sure that the system reboots after a panic instead of halting.

sparc specific

- Disabled threading in `named`(8) on sparc and sparc64 to avoid a crash.

mac68k specific

- sysinst now newfs'es and mounts the target filesystem(s).

Changes Between The NetBSD 3.0 release and 3.0.1 update

The NetBSD 3.0.1 update is the first security/critical update of the NetBSD 3.0 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 3.1 etc), together with other less-critical fixes and feature enhancements.

Specific updates are as follows:

Kernel

- Hold kernel_lock while calling systrace_exit().
- In systrace_make_msg(), sleep uninterruptibly while waiting for the response from the systrace daemon, so that the message protocol between the kernel and the daemon doesn't get out of sync.
- RAIDframe: mark used spares as failed if they encounter IO errors.
- `wdc`(4): after a reset don't wait for drives to come ready if there are no drives (fixes a 30s hang after resume).
- Fix support in `wd`(4) for 1 and 2TB disks.
- Fix a bug in the fragment cache which could cause kernel panics (SA2006-004).
- Fix a crash caused by `azalia`(4) when a connection list has invalid NIDs.
- `aic`(4): work around an rbus resource allocation problem so cards work again.
- RAIDframe was erroneously re-initializing the Parity Stripe Status pool each time a new array was configured. This causes grief with things like 'vmstat -m' by causing it to loop. Make RAIDframe only initialize PSS bits once.
- `twe`(4): fix a memory leak in the TWEIO_GET_PARAM ioctl.
- Prevent system crashes caused by malformed ELF interpreters (SA2006-008).
- `usb`(4): Allow a NULL pointer as argument to usb_get_next_event(), and don't allocate a "struct usb_event" on stack in usb_add_event().
- Check the "oldlen" argument to `sysctl`(2) before passing it to `vslock`(9). This prevents a local DOS (SA2006-013).
- Use a `pmatch`(9) expression which should catch all present and future seagate drives larger than 200GB for the WD_QUIRK_FORCE_LBA48 quirks.
- Fix the 'audioctl of death' problem (SA2006-014).
- Limit the size of any kernel buffers allocated by the VOP_READDIR routines to MAXBSIZE.

Networking

- Make sure all `bridge`(4) structs are initialized to 0.
- Fix a memory disclosure in `bridge`(4) (SA2006-005).
- Use `sigaction`(2) to setup automatic disposal of child processes after daemonizing `rpc.statd`(8). This is more portable and avoids zombie `rpc.statd`(8) processes after an NFS client running e.g. Mac OS X shuts down.
- Prevent system crash when attempting to gather information about a non-existing alias of a network interface via the SIOCGIFALIAS ioctl (SA2006-012).
- Fix a panic caused by insufficient validation when parsing IPv6 socket options (SA2006-016).

File system

- Change union_unmount() to not play with the fs root vnode explicitly. Let it get recycled along with all of the others. This is important as if the root vnode has already been reclaimed, then we get a panic when we try to vget it.

Libraries

- xdr_rec.c missing a bugfix for an improper security check. The correct way to check for a zero record length is to check for it without the LAST_FRAG marker in it, since it's legal to send a LAST_FRAG marker with 0 bytes of data.
- `pam_nologin`(8): use the class of the user, not then default class, when checking for nologin and ignorelogin login.conf(5) capabilities.
- `pam_unix`(8): fix an uninitialized variable, and remove some unreachable code.
- Make PAM work on NetBSD ports without shared library support.
- Make password aging work again.
- Fix a bug in signal handling which could e.g get the MySQL daemon stuck in a tight loop after receiving a HUP signal.

Security

- Improve `rnd`(4) code probing for the Intel hardware RNG to avoid false detections (SA2006-009).
- `mail`(1) creates record file with insecure umask (SA2006-007).
- Fix a remote code execution vulnerability in sendmail (SA2006-010).
- A vulnerability was found in the `fast_ipsec`(4) stack that renders the IPSec anti-replay service ineffective under certain circumstances (SA2006-011).
- Fix a number of small security problems with the games.
- Fix an FPU Information leak on i386/amd64/Xen platforms with AMD CPUs (SA2006-015).
- Fix several integer overflows and NULL-pointer dereferences in freetype2.
- Fix a denial of service vulnerability in sendmail when handling malformed multipart MIME messages (SA2006-017).

Miscellaneous

- Sync the Nvidia drivers with XFree86's sources as of December 24th, 2005. These changes fix lots of problems (i.e. freezes) with the latest cards (such as a GeForce 6600GT).
- Give systrace its own version of realpath() that does what it wants, call it intercept_realpath(). Unbreaks systrace.
- `login`(1), `pam_securetty`(8): don't issue a different message for root login on an insecure terminal.
- Fix some unpaired sigblocks which possibly leave a process with all signals blocked (esp. apparant under Gnome).
- Fix a NULL pointer dereference in `ftp`(1).
- Make `sh`(1) expand "$@" correctly again.

acorn26 specific

- Avoid panics under NetBSD/acorn26 whenever a process exits.
- Stop NetBSD/acorn26 from allocating eight times as much pool memory as it needs, leading to a rapid exhaustion of physical memory. NetBSD/acorn26 now boots multi-user again.

amiga specific

- Installer: don't ask for the root device before the user had a chance to identify how it's called under NetBSD.

hp300 specific

- Fix the bootloader so it doesn't print garbage to the bitmap console.
- Fix a huge number of pkgsrc build problems.
- Fix boot failure problem on HP400t with fb console.
- Fix a problem with the probe of SCSI disks in the installer.

i386 specific

- Update pciide at pnpbios to work with the last changes to `wdc`(4), especially the deferral of drive probe.

m68k specific

- Fix a huge number of pkgsrc build problems.

mips specific

- Avoid kernel panics caused by binaries compiled with "-mips2" or better.
- Update FPE trapsignal functions for new siginfo.

powerpc specific

- Avoid pthreads program crashes.

sparc specific

- For GCC3 and later, use the __builtin_va* constructs. Avoids a build problem with Firefox.

xen specific

- Avoid dom0 kernel crash when destroying a domain with active I/O going.
- Fix the FPU problems detected by paranoia on a NetBSD/Xen guest.
- Check the destination ethernet address when not in promiscous mode. Fix a problem where packets would be duplicated, possibly looping, when a domU is doing IP routing.
- Avoid panics under high system load.

This is the third major release of NetBSD/evbarm.

The Future of NetBSD

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

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

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

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

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

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

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

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

Sources of NetBSD

Refer to .: **http://www.NetBSD.org/mirrors/**

NetBSD 3.0.3 Release Contents

The root directory of the NetBSD 3.0.3 release is organized as follows:

`.../NetBSD-3.0.3/`

`CHANGES`	Changes since earlier NetBSD releases.
`LAST_MINUTE`	Last minute changes.
`MIRRORS`	A list of sites that mirror the NetBSD 3.0.3 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 3.0.3 has a binary distribution.

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

gnusrc This set contains the "gnu" sources, including the source for the compiler, assembler, groff, and the other GNU utilities in the binary distribution sets.
79 MB gzipped, 367 MB uncompressed

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

sharesrc This set contains the "share" sources, which include the sources for the man pages not associated with any particular program; the sources for the typesettable document set; the dictionaries; and more.
5 MB gzipped, 20 MB uncompressed

src This set contains all of the base NetBSD 3.0.3 sources which are not in **gnusrc**, **sharesrc**, or **syssrc**.
37 MB gzipped, 176 MB uncompressed

syssrc This set contains the sources to the NetBSD 3.0.3 kernel for all architectures; `config`(8); and `dbsym`(8).
26 MB gzipped, 140 MB uncompressed

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

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

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

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

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

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

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

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

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

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

NetBSD/evbarm subdirectory structure

The evbarm-specific portion of the NetBSD 3.0.3 release is found in the `evbarm` subdirectory of the distribution: `.../NetBSD-3.0.3/evbarm/`. It contains the following files and directories:

`INSTALL.html`
`INSTALL.ps`
`INSTALL.txt`
`INSTALL.more` Installation notes in various file formats, including this file. The `.more` file contains underlined text using the `more`(1) conventions for indicating italic and bold display.
`binary/`
`kernel/`

```
                    netbsd-GENERIC.gz
                                A gzipped NetBSD kernel containing code for everything sup-
                                ported in this release.
              sets/     evbarm binary distribution sets; see below.
installation/
              misc/
```

Binary distribution sets

The NetBSD evbarm binary distribution sets contain the binaries which comprise the NetBSD 3.0.3 release for the evbarm. The binary distribution sets can be found in the `evbarm/binary/sets` subdirectory of the NetBSD 3.0.3 distribution tree, and are as follows:

base The NetBSD 3.0.3 evbarm **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.
18 MB gzipped, 50 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.
21 MB gzipped, 76 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-INTEGRATOR
(see below)
4 MB gzipped, 10 MB uncompressed

kern-IQ80310
(see below)
20 MB gzipped, 54 MB uncompressed

kern-IQ80321
(see below)
21 MB gzipped, 55 MB uncompressed

kern-TEAMASA_NPWR
12 MB gzipped, 35 MB uncompressed

These sets contain a NetBSD/evbarm 3.0.3 kernel, named `/netbsd`. Some of these sets also contain a binary format kernel named `/netbsd.bin` and an S-record format kernel named `/netbsd.srec` that your firmware may need to boot. You *must* install the kernel that matches your hardware.

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

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

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

NetBSD maintains its own set of sources for the X Window System in order to assure tight integration and compatibility. These sources are based on XFree86, and tightly track XFree86 releases. They are currently equivalent to XFree86 4.4.0. Binary sets for the X Window System are distributed with NetBSD. The sets are:

xbase The basic files needed for a complete X client environment. This does not include the X servers.
6 MB gzipped, 18 MB uncompressed

xcomp The extra libraries and include files needed to compile X source code.
13 MB gzipped, 44 MB uncompressed

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

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

xserver The X server.
4 MB gzipped, 9 MB uncompressed

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

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

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

NetBSD/evbarm System Requirements and Supported Devices

NetBSD 3.0.3 runs on the following ARM architecture evaluation boards:

- Technologic Systems TS-7200
- ARM, Ltd. Integrator
- Intel IQ80310
- Intel IQ80321
- Team ASA, Inc. Npwr
- Intel IXM1200
- Samsung SMDK2800
- ADI Engineering, Inc. BRH ("Big Red Head")

Supported devices (Technologic Systems TS-7200)

For devices that have machine-independent drivers, this is not an exhaustive list, but merely reflects the devices that have been tested on this platform.

- TS-5620 Real Time Clock
- Serial ports
  - On-chip EP9302 UARTs
  - TS-SER1, TS-SER2, TS-SER4 PC/104 serial adapters
  - TS-MODEM, TS-MODEM2 PC/104 land-line/cellular modems
- Ethernet adapters
  - On-chip EP9302 10/100 ethernet
  - TS-ETH10 PC/104 expansion ethernet adapters
- Other on-board functions
  - CompactFlash socket
  - USB OHCI host
  - Text-mode HD44780 LCD display header
  - Matrix keypad/GPIO header
  - Board temperature sensor and watchdog timer

Supported devices (ARM, Ltd. Integrator)

For devices that have machine-independent drivers, this is not an exhaustive list, but merely reflects the devices that have been tested on this platform.

- PrimeCell PL030 Real Time Clock
- Serial ports
  - PrimeCell PL010 UART
- SCSI host adapters
  - Adaptec AHA-2910, 2915, 2920, and 2930C adapters.
  - Adaptec AHA-2x4x[U][2][W] cards and onboard PCI designs using the AIC-7770, AIC-7850, AIC-7860, AIC-7870, AIC-7880 and AIC-789x chipsets.
  - Adaptec AHA-394x[U][W] cards [b]
  - Adaptec AHA-3950U2 cards
  - Adaptec AHA-3960, 19160 and 29160 Ultra-160 adapters
- Ethernet adapters
  - Intel i82557, i82558, i82559, i82550

Supported devices (Intel IQ80310)

For devices that have machine-independent drivers, this is not an exhaustive list, but merely reflects the devices that have been tested on this platform.

- Serial ports
  - On-board ST16c550 UARTs
- Ethernet adapters
  - Intel i82557, i82558, i82559, i82550
- i80312 Companion I/O functions
  - General Purpose I/O unit

Supported devices (Intel IQ80321)

For devices that have machine-independent drivers, this is not an exhaustive list, but merely reflects the devices that have been tested on this platform.

- Serial ports
  - On-board TI 16550 UART
- Ethernet adapters
  - Intel i82542, i82543, i85244 Gigabit Ethernet
- i80321 I/O Processor functions
  - Application Accelerator Unit
  - Watchdog timer

Supported devices (Team ASA Npwr)

The Npwr is a storage appliance based on the Intel IOP310 I/O processor chipset. This is an exhaustive list of devices supported on the Npwr as the Npwr has no expansion capability.

- Serial ports
  - On-board TI 16550 UART
- SCSI host adapters
  - On-board LSI Logic 53c1010 Ultra160 SCSI adapter
- Ethernet adapters
  - On-board Intel i82544 Gigabit Ethernet

Supported devices (Intel IXM1200)

For devices that have machine-independent drivers, this is not an exhaustive list, but merely reflects the devices that have been tested on this platform.

- Serial ports
  - IXP1200 on-chip serial port
- Ethernet adapters
  - Intel i82557, i82558, i82559, i82550

Supported devices (Samsung SMDK2800)

For devices that have machine-independent drivers, this is not an exhaustive list, but merely reflects the devices that have been tested on this platform.

- Serial ports
  - S3C2800 on-chip serial ports

Supported devices (ADI BRH)

For devices that have machine-independent drivers, this is not an exhaustive list, but merely reflects the devices that have been tested on this platform.

- Serial ports
  - On-board TI 16752 DUART
- Ethernet adapters
  - Intel i82557, i82558, i82559, i82550

- Intel i82544, i82546 Gigabit Ethernet

Getting the NetBSD System on to Useful Media

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

Installation is supported from several media types, including:

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

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

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

Proceed to the instruction on installation.

MS-DOS floppy NetBSD doesn't include split sets to keep the distribution size down. They can be created on a seperate machine using the `split`(1) command, running e.g. **`split -b 235k base.tgz base.`** to split the `base.tgz` file from `evbarm/binary/sets` into files named `base.aa`, `base.ab`, and so on. Repeat this for all `set_name.tgz` files, splitting them into `set_name.`*`xx`* files. Count the number of `set_name.`*`xx`* files that make up the distribution sets you want to install or upgrade. You will need one sixth that number of 1.44 MB floppies.

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

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

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

FTP The preparations for this installation/upgrade method are easy; all you need to do is make sure that there's an FTP site from which you can retrieve the NetBSD distribution when you're about to install or upgrade. If you don't have DHCP available on your network, you will need to know the numeric IP address of that site, and, if it's not on a network directly connected to the machine on which you're installing or upgrading NetBSD, you need to know the numeric IP address of the router closest to the NetBSD machine. Finally, you need to know the numeric IP address of the NetBSD machine itself. If you don't have access to a functioning nameserver during installation, the IPv4 address of **ftp.NetBSD.org** is `204.152.184.75` and the IPv6 address is `2001:4f8:4:7:2e0:81ff:fe21:6563` (as of June, 2004).

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

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

NFS

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

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

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

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

Tape

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

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

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

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

```
# cd .../NetBSD-3.0.3
# cd evbarm/binary
# tar -cf tape_device misc etc kern
```

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

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

Preparing your System for NetBSD installation

First and foremost, before beginning the installation process, *make sure you have a reliable backup* of any data on the target system that you wish to keep. Mistakes in e.g. partitioning your hard disk may lead to data loss.

Preparation for the Technologic Systems TS-7200

The TS-7200 uses RedBoot firmware which supports loading the kernel via serial port and the on-chip Ethernet. These notes assume that the Ethernet will be used to load the kernel. RedBoot can either use TFTP or HTTP to load the kernel via the network.

These notes also assume the the TS-7200 will be run from the CompactFlash; that a FFS filesystem on the CompactFlash card will provide the root file system. The minimum size of the CompactFlash card is 64MB with 512MB being the recommended minimum size if attempting a full installation.

The first step to install NetBSD is to interrupt normal system bootup and drop to the RedBoot prompt by hitting Ctrl-C immediately after applying power. To load both the installation kernel and the generic post-installation TS7200 kernel you will need network connectivity from RedBoot which you will have to configure using the RedBoot fconfig command; see the RedBoot documentation for more information.

A compressed kernel image and loader suitable for placing into flash is provided with the distribution. Use of this image is optional; a kernel that can be directly loaded over the network by RedBoot is also provided.

The image for the TS-7200 can be found under the NetBSD/evbarm 3.0.3 distribution directory in `evbarm/binary/gzimg/gzimg_TS7200_flash_0x60660000.gz`, and corresponds to the kernel in `evbarm/binary/kernel/netbsd-TS7200.gz`. The following steps describe how to copy the compressed kernel image into flash.

1. Uncompress the gzimg file and place it into the TFTP server's download area (note, the kernel inside the gzimg is still compressed after this step).

   ```
   server# gzcat gzimg_TS7200_flash_0x60660000.gz > /tftpboot/gzimg_TS7200_flash_0x60660000
   ```

2. Load the gzimg from the TFTP server.

   ```
   RedBoot load -r -b 0x00200000 gzimg_TS7200_flash_0x60660000
   Raw file loaded 0x00200000-0x00605fe0
   ```

3. Use the RedBoot **fis** command to copy the image into flash. For more information on the RedBoot Flash Image System, see the RedBoot documentation. The factory default Linux image may already be present in flash and must be deleted before continuing.

   ```
   RedBoot fis delete vmlinux
   RedBoot fis create -b 0x00200000 -f 0x60660000 -l 0x00160000 netbsd.gz
   ```

Once the compressed kernel image has been copied into flash, it may be started by jumping to the flash address of the image:

```
RedBoot g 0x60660000
```

Preparation for the Intel IQ80310

The IQ80310 uses RedBoot firmware from Red Hat, Inc. RedBoot supports loading the kernel via the serial port and the on-board Ethernet. These notes assume that the Ethernet will be used to load the kernel. RedBoot uses the Trivial File Transfer Protocol (TFTP) to load the kernel via the network.

These notes also assume that the IQ80310 will be run as a diskless system; that an NFS server will provide the root file system. The NetBSD kernel uses the Dynamic Host Configuration Protocol (DHCP) to obtain network address and root file system information.

The first step is to configure your DHCP, NFS, and TFTP server software. You will need the MAC address for the on-board Ethernet, which RedBoot can provide; see the RedBoot documentation for more information.

The next step is to configure RedBoot to obtain its IP address from your DHCP server; see the RedBoot documentation for more information.

Once your DHCP server and RedBoot are properly configured, you should see information similar to the following when the IQ80310 is reset or powered-up:

```
RedBoot(tm) debug environment - built 17:16:14, Feb 12 2001
Platform: IQ80310 (XScale)
Copyright (C) 2000, Red Hat, Inc.

RAM: 0xa0000000-0xa2000000
FLASH: 0x00000000 - 0x00800000, 64 blocks of 0x00020000 bytes each.
IP: 192.168.0.10, Default server: 192.168.0.1
RedBoot
```

A compressed kernel image and loader suitable for placing into flash is provided with the distribution. Use of this image is optional; a kernel that can be directly loaded over the network by RedBoot is also provided.

The image for the IQ80310 can be found under the NetBSD/evbarm 3.0.3 distribution directory in `evbarm/binary/gzimg/gzimg_IQ80310_flash_0x00080000.gz`, and corresponds to the kernel in `evbarm/binary/kernel/netbsd-IQ80310.gz`. The following steps describe how to copy the compressed kernel image into flash.

1. Uncompress the gzimg file and place it into the TFTP server's download area (note, the kernel inside the gzimg is still compressed after this step).

   ```
   server# gzcat gzimg_IQ80310_flash_0x00080000.gz > /tftpboot/gzimg_IQ80310_flash_0x00080000
   ```

2. Load the gzimg from the TFTP server.

   ```
   RedBoot load -r -b 0xa0200000 gzimg_IQ80310_flash_0x00080000
   Raw file loaded 0xa0200000-0xa0305fe0
   ```

3. Use the RedBoot **fis** command to copy the image into flash. For more information on the RedBoot Flash Image System, see the RedBoot documentation.

   ```
   RedBoot fis create -b 0xa0200000 -f 0x00080000 -l 0x00200000 netbsd.gz
   ... Erase from 0x00080000-0x00280000: ....................
   ... Program from 0xa0200000-0xa0400000 at 0x00080000: .....................
   ... Unlock from 0x007e0000-0x00800000:.
   ... Erase from 0x007e0000-0x00800000:.
   ... Program from 0xa1fd0000-0xa1ff0000 at 0x007e0000:.
   ... Lock from 0x007e0000-0x00800000:.
   ```

Once the compressed kernel image has been copied into flash, it may be started by jumping to the flash address of the image:

```
RedBoot g 0x00080000

 NetBSD/IQ80310 Gzip Boot, Revision 1.1
 (root@tgm, Thu Mar 28 18:32:45 PST 2002)
 RAM 0xa0000000 - 0xaffffffff, heap at 0xaffd0000
 Load address: 0xa0200000
```

```
 Image size: 741244
Uncompressing image...done.
Jumping to image @ 0xa0200000...

NetBSD/evbarm (IQ80310) booting ...
Resetting secondary PCI bus...
initarm: Configuring system ...
physmemory: 65536 pages at 0xa0000000 - 0xafffffff
init subsystems: stacks vectors undefined pmap irq done.
[ using 155084 bytes of netbsd ELF symbol table ]
Copyright (c) 1996, 1997, 1998, 1999, 2000, 2001, 2002
    The NetBSD Foundation, Inc.  All rights reserved.
Copyright (c) 1982, 1986, 1989, 1991, 1993
    The Regents of the University of California.  All rights reserved.

NetBSD 1.5ZC (IQ80310) #20: Fri Mar 29 10:25:53 PST 2002
    root@tgm:/u1/netbsd/src/sys/arch/evbarm/compile/IQ80310
total memory = 256 MB
avail memory = 232 MB
using 1228 buffers containing 13208 KB of memory
mainbus0 (root)
cpu0 at mainbus0: i80200 step A-0 (XScale core)
cpu0: DC enabled IC enabled WB enabled LABT
cpu0: 32KB/32B 32-way Instruction cache
cpu0: 32KB/32B 32-way write-back-locking Data cache
obio0 at mainbus0: board rev. F, CPLD rev. D, backplane present
com0 at obio0 addr 0xfe810000: ns16550a, working fifo
com0: console
com1 at obio0 addr 0xfe800000: ns16550a, working fifo
iopxs0 at mainbus0: i80312 Companion I/O, acting as PCI host
iopxs0: configuring Secondary PCI bus
pci0 at iopxs0 bus 1
pci0: i/o space, memory space enabled
ppb0 at pci0 dev 7 function 0: Digital Equipment DECchip 21154 PCI-PCI Bridge (rev. 0
pci1 at ppb0 bus 2
pci1: i/o space, memory space enabled
fxp0 at pci1 dev 0 function 0: i82559S Ethernet, rev 9
fxp0: interrupting at iq80310 irq 1
fxp0: Ethernet address 00:80:4d:46:0b:b9
inphy0 at fxp0 phy 1: i82555 10/100 media interface, rev. 4
inphy0: 10baseT, 10baseT-FDX, 100baseTX, 100baseTX-FDX, auto
clock: hz=100 stathz=0 profhz=0
boot device: unknown
root device:
```

Preparation for the Intel IQ80321

The IQ80321 uses RedBoot firmware from Red Hat, Inc. RedBoot supports loading the kernel via the serial port and the on-board Ethernet. These notes assume that the Ethernet will be used to load the kernel. RedBoot uses the Trivial File Transfer Protocol (TFTP) to load the kernel via the network.

These notes also assume that the IQ80321 will be run as a diskless system; that an NFS server will provide the root file system. The NetBSD kernel uses the Dynamic Host Configuration Protocol (DHCP) to obtain

network address and root file system information.

The first step is to configure your DHCP, NFS, and TFTP server software. You will need the MAC address for the on-board Ethernet, which RedBoot can provide; see the RedBoot documentation for more information.

The next step is to configure RedBoot to obtain its IP address from your DHCP server; see the RedBoot documentation for more information.

Once your DHCP server and RedBoot are properly configured, you should see information similar to the following when the IQ80321 is reset or powered-up:

```
Ethernet eth0: MAC address 00:07:e9:03:38:40
IP: 192.168.0.11, Default server: 192.168.0.1

RedBoot(tm) bootstrap and debug environment [ROM]
Non-certified release, version UNKNOWN - built 11:21:56, Feb  1 2002

Platform: IQ80321 (XScale)
Copyright (C) 2000, 2001, 2002, Red Hat, Inc.

RAM: 0x00000000-0x08000000, 0x00017008-0x01ddd000 available
FLASH: 0xf0000000 - 0xf0800000, 64 blocks of 0x00020000 bytes each.
RedBoot
```

A compressed kernel image and loader suitable for placing into flash is provided with the distribution. Use of this image is optional; a kernel that can be directly loaded over the network by RedBoot is also provided.

The image for the IQ80321 can be found under the NetBSD/evbarm 3.0.3 distribution directory in `evbarm/binary/gzimg/gzimg_IQ80321_flash_0xf0080000.gz`, and corresponds to the kernel in `evbarm/binary/kernel/netbsd-IQ80321.gz`. The following steps describe how to copy the compressed kernel image into flash.

1. Uncompress the gzimg file and place it into the TFTP server's download area (note, the kernel inside the gzimg is still compressed after this step).

   ```
   server# gzcat gzimg_IQ80321_flash_0xf0080000.gz > /tftpboot/gzimg_IQ80321_flash_0xf0080000
   ```

2. Load the gzimg from the TFTP server.

   ```
   RedBoot load -r -b 0x00200000 gzimg_IQ80321_flash_0xf0080000
   Raw file loaded 0x00200000-0x00305fe4
   ```

3. Use the RedBoot **fis** command to copy the image into flash. For more information on the RedBoot Flash Image System, see the RedBoot documentation.

   ```
   RedBoot fis create -b 0x00200000 -f 0xf0080000 -l 0x00200000 netbsd.gz
   * CAUTION * about to program 'netbsd.gz'
               at 0xf0080000..0xf027ffff from 0x00200000 - are you sure (y/n)? y
   ... Erase from 0xf0080000-0xf0280000: ......................
   ... Program from 0x00200000-0x00400000 at 0xf0080000: .....................
   ... Unlock from 0xf07e0000-0xf0800000:.
   ... Erase from 0xf07e0000-0xf0800000:.
   ... Program from 0x01ddf000-0x01dff000 at 0xf07e0000:.
   ... Lock from 0xf07e0000-0xf0800000:.
   ```

Once the compressed kernel image has been copied into flash, it may be started by jumping to the flash address of the image:

```
RedBoot g 0xf0080000

 NetBSD/IQ80321 Gzip Boot, Revision 1.1
 (root@tgm, Thu Mar 28 18:32:45 PST 2002)
 RAM 0xa0000000 - 0xa7ffffff, heap at 0xa7fd0000
 Load address: 0xa0200000
 Image size: 739495
Uncompressing image...done.
Jumping to image @ 0xa0200000...

NetBSD/evbarm (IQ80321) booting ...
initarm: Configuring system ...
physmemory: 32768 pages at 0xa0000000 - 0xa7ffffff
init subsystems: stacks vectors undefined pmap irq done.
[ using 155076 bytes of netbsd ELF symbol table ]
Copyright (c) 1996, 1997, 1998, 1999, 2000, 2001, 2002
    The NetBSD Foundation, Inc.  All rights reserved.
Copyright (c) 1982, 1986, 1989, 1991, 1993
    The Regents of the University of California.  All rights reserved.

NetBSD 1.5ZC (IQ80321) #1: Thu Mar 28 18:31:58 PST 2002
    root@tgm:/u1/netbsd/src/sys/arch/evbarm/compile/IQ80321
total memory = 128 MB
avail memory = 113 MB
using 1228 buffers containing 6656 KB of memory
mainbus0 (root)
cpu0 at mainbus0: i80321 step A-0 (XScale core)
cpu0: DC enabled IC enabled WB enabled LABT
cpu0: 32KB/32B 32-way Instruction cache
cpu0: 32KB/32B 32-way write-back-locking Data cache
obio0 at mainbus0
com0 at obio0 addr 0xfe800000: ns16550a, working fifo
com0: console
iopxs0 at mainbus0: i80321 I/O Processor, acting as PCI host
iopxs0: configuring PCI bus
pci0 at iopxs0 bus 0
pci0: i/o space, memory space enabled, rd/line, rd/mult, wr/inv ok
wm0 at pci0 dev 4 function 0: Intel i82544 1000BASE-T Ethernet, rev. 2
wm0: interrupting at iop321 irq 27
wm0: Ethernet address 00:07:e9:03:38:40
makphy0 at wm0 phy 1: Marvell 88E1000 Gigabit PHY, rev. 0
makphy0: 10baseT, 10baseT-FDX, 100baseTX, 100baseTX-FDX, 1000baseT, 1000baseT-FDX, au
clock: hz=100 stathz=0 profhz=0
boot device: unknown
root device:
```

Installing the NetBSD System

Installation for the Technologic Systems TS-7200

The following steps describe how to set up a CompactFlash card for the TS-7200. The TS-7200 can use the interactive installation program sysinst to perform an internet installation from ftp.netbsd.org on the CompactFlash.

1. Place the netbsd-TS7200_INSTALL.bin kernel into the TFTP server's download area. It must be decompressed from the gzipped image in the release directory evbarm/installation/instkernel/netbsd-TS7200_INSTALL.srec.gz

   ```
   server# gunzip netbsd-TS7200_INSTALL.srec.gz
   server# cp netbsd-TS7200_INSTALL.srec
   /tftpboot/netbsd-TS7200_INSTALL
   ```

2. Load the kernel from the TFTP server.

   ```
   RedBoot load netbsd-TS7200_INSTALL
   ```

3. Start the kernel running.

   ```
   RedBoot go
   ```

   The kernel will display boot messages, probe for devices, and invoke the interactive sysinst installation tool. Once sysinst has completed installation, the system will reboot back into RedBoot. If the preparation instructions above have been followed, the post-installation NetBSD kernel will be in the FIS ready to be jumped into. You can have RedBoot automatically start NetBSD by putting the command go 0x60660000 into the RedBoot bootscript using the fconfig command.

Installation for the Intel IQ80310

The following steps describe how to set up an NFS root environment for the IQ80310 and how to boot the IQ80310.

1. Place the kernel into the TFTP server's download area.

   RedBoot only loads S-Records, so select a kernel with the ".srec" filename suffix. In this example, we will boot the kernel hard-wired to use the on-board Ethernet as the root device.

   ```
   server# cp netbsd-fxp0.srec /tftpboot/netbsd-iq80310-fxp0
   ```

2. Unpack the distribution sets into the target's root directory.

   ```
   server# cd /export/client/iq80310
   server# gzcat .../base.tgz | tar xvpf -
   server# gzcat .../etc.tgz | tar xvpf -
   [repeat for all sets you wish to unpack]
   ```

3. Load the kernel from the TFTP server.

   ```
   RedBoot load netbsd-iq80310-fxp0
   Entry point: 0xa0200000, address range: 0xa0200000-0xa035e07c
   RedBoot
   ```

4. Start the kernel running. The kernel will display boot message, probe for devices, and mount the root file system. If you used a kernel hard-wired to use the on-board Ethernet, it will attempt to mount the root file system automatically. Otherwise, it will prompt for the root device to use.

   Since the system's run-time environment has not yet been configured, the system should boot into single-user mode.

```
RedBoot go

NetBSD/evbarm (IQ80310) booting ...
Resetting secondary PCI bus...
initarm: Configuring system ...
physmemory: 65536 pages at 0xa0000000 - 0xafffffff
init subsystems: stacks vectors undefined pmap irq done.
[ using 155084 bytes of netbsd ELF symbol table ]
Copyright (c) 1996, 1997, 1998, 1999, 2000, 2001, 2002
    The NetBSD Foundation, Inc.  All rights reserved.
Copyright (c) 1982, 1986, 1989, 1991, 1993
    The Regents of the University of California.  All rights reserved.

NetBSD 1.5ZC (IQ80310) #20: Fri Mar 29 10:25:53 PST 2002
    root@tgm:/u1/netbsd/src/sys/arch/evbarm/compile/IQ80310
total memory = 256 MB
avail memory = 232 MB
using 1228 buffers containing 13208 KB of memory
mainbus0 (root)
cpu0 at mainbus0: i80200 step A-0 (XScale core)
cpu0: DC enabled IC enabled WB enabled LABT
cpu0: 32KB/32B 32-way Instruction cache
cpu0: 32KB/32B 32-way write-back-locking Data cache
obio0 at mainbus0: board rev. F, CPLD rev. D, backplane present
com0 at obio0 addr 0xfe810000: ns16550a, working fifo
com0: console
com1 at obio0 addr 0xfe800000: ns16550a, working fifo
iopxs0 at mainbus0: i80312 Companion I/O, acting as PCI host
iopxs0: configuring Secondary PCI bus
pci0 at iopxs0 bus 1
pci0: i/o space, memory space enabled
ppb0 at pci0 dev 7 function 0: Digital Equipment DECchip 21154 PCI-PCI Bridge (rev
pci1 at ppb0 bus 2
pci1: i/o space, memory space enabled
fxp0 at pci1 dev 0 function 0: i82559S Ethernet, rev 9
fxp0: interrupting at iq80310 irq 1
fxp0: Ethernet address 00:80:4d:46:0b:b9
inphy0 at fxp0 phy 1: i82555 10/100 media interface, rev. 4
inphy0: 10baseT, 10baseT-FDX, 100baseTX, 100baseTX-FDX, auto
clock: hz=100 stathz=0 profhz=0
boot device: unknown
root on fxp0
nfs_boot: trying DHCP/BOOTP
nfs_boot: DHCP next-server: 192.168.0.1
nfs_boot: my_name=iq80310.lab.wasabisystems.com
nfs_boot: my_domain=wasabisystems.com
nfs_boot: my_addr=192.168.0.10
nfs_boot: my_mask=255.255.255.0
nfs_boot: gateway=192.168.0.254
root on 192.168.0.1:/export/client/iq80310
/etc/rc.conf is not configured.  Multiuser boot aborted.
Enter pathname of shell or RETURN for sh:
```

Installation for the Intel IQ80321

The following steps describe how to set up an NFS root environment for the IQ80321 and how to boot the IQ80321.

1. Place the kernel into the TFTP server's download area.

   RedBoot only loads S-Records, so select a kernel with the ".srec" filename suffix. In this example, we will boot the kernel hard-wired to use the on-board Ethernet as the root device.

   ```
   server# cp netbsd-wm0.srec /tftpboot/netbsd-iq80321-wm0
   ```

2. Unpack the distribution sets into the target's root directory.

   ```
   server# cd /export/client/iq80321
   server# gzcat .../base.tgz | tar xvpf -
   server# gzcat .../etc.tgz | tar xvpf -
   [repeat for all sets you wish to unpack]
   ```

3. Load the kernel from the TFTP server.

   ```
   RedBoot load netbsd-iq80321-wm0
   Entry point: 0x00200000, address range: 0x00200000-0x00396a40
   RedBoot
   ```

4. Start the kernel running. The kernel will display boot message, probe for devices, and mount the root file system. If you used a kernel hard-wired to use the on-board Ethernet, it will attempt to mount the root file system automatically. Otherwise, it will prompt for the root device to use.

   Since the system's run-time environment has not yet been configured, the system should boot into single-user mode.

   ```
   RedBoot go

   NetBSD/evbarm (IQ80321) booting ...
   initarm: Configuring system ...
   physmemory: 32768 pages at 0xa0000000 - 0xa7ffffff
   init subsystems: stacks vectors undefined pmap irq done.
   [ using 156468 bytes of netbsd ELF symbol table ]
   Copyright (c) 1996, 1997, 1998, 1999, 2000, 2001, 2002
       The NetBSD Foundation, Inc.  All rights reserved.
   Copyright (c) 1982, 1986, 1989, 1991, 1993
       The Regents of the University of California.  All rights reserved.

   NetBSD 1.5ZC (IQ80321) #1: Thu Mar 28 18:20:34 PST 2002
       root@tgm:/u1/netbsd/src/sys/arch/evbarm/compile/IQ80321
   total memory = 128 MB
   avail memory = 113 MB
   using 1228 buffers containing 6656 KB of memory
   mainbus0 (root)
   cpu0 at mainbus0: i80321 step A-0 (XScale core)
   cpu0: DC enabled IC enabled WB enabled LABT
   cpu0: 32KB/32B 32-way Instruction cache
   cpu0: 32KB/32B 32-way write-back-locking Data cache
   obio0 at mainbus0
   com0 at obio0 addr 0xfe800000: ns16550a, working fifo
   com0: console
   ```

```
iopxs0 at mainbus0: i80321 I/O Processor, acting as PCI host
iopxs0: configuring PCI bus
pci0 at iopxs0 bus 0
pci0: i/o space, memory space enabled, rd/line, rd/mult, wr/inv ok
wm0 at pci0 dev 4 function 0: Intel i82544 1000BASE-T Ethernet, rev. 2
wm0: interrupting at iop321 irq 27
wm0: Ethernet address 00:07:e9:03:38:40
makphy0 at wm0 phy 1: Marvell 88E1000 Gigabit PHY, rev. 0
makphy0: 10baseT, 10baseT-FDX, 100baseTX, 100baseTX-FDX, 1000baseT, 1000baseT-FDX,
clock: hz=100 stathz=0 profhz=0
boot device: unknown
root on wm0
nfs_boot: trying DHCP/BOOTP
nfs_boot: DHCP next-server: 192.168.0.1
nfs_boot: my_name=iq80321.lab.wasabisystems.com
nfs_boot: my_domain=wasabisystems.com
nfs_boot: my_addr=192.168.0.11
nfs_boot: my_mask=255.255.255.0
nfs_boot: gateway=192.168.0.254
root on 192.168.0.1:/export/client/iq80321
/etc/rc.conf is not configured.  Multiuser boot aborted.
Enter pathname of shell or RETURN for sh:
```

Post installation steps

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

1. Configuring `/etc/rc.conf`

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

2. Logging in

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

   Unless you have connected an unusual terminal device as the console you can just press `RETURN` when it prompts for `Terminal type? [...]`.

3. Adding accounts

   Use the `useradd`(8) command to add accounts to your system. Do not edit `/etc/passwd` directly! See `vipw`(8) and `pwd_mkdb`(8) if you want to edit the password database.

4. The X Window System

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

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

5. Installing third party packages

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

   - More information on the package system is at
     http://www.NetBSD.org/Documentation/software/packages.html
   - A list of available packages suitable for browsing is at
     ftp://ftp.NetBSD.org/pub/NetBSD/packages/pkgsrc/README.html

- Precompiled binaries can be found at ,:
  ftp://ftp.NetBSD.org/pub/NetBSD/packages/
  usually in the `3.0.3/evbarm/All` subdir. You can install them with the following commands under `sh`(1):

  ```
  # PKG_PATH=ftp://ftp.NetBSD.org/pub/NetBSD/packages/3.0.3/evbarm/All
  # export PKG_PATH
  # pkg_add -v tcsh
  # pkg_add -v bash
  # pkg_add -v perl
  # pkg_add -v apache
  # pkg_add -v kde
  # pkg_add -v mozilla
  ...
  ```

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

  ```
  # setenv PKG_PATH ftp://ftp.NetBSD.org/pub/NetBSD/packages/3.0.3/evbarm/All
  ...
  ```

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

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

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

  After extracting, see the `README` and `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/mail/sendmail.cf` file will almost definitely need to be adjusted; files aiding in this can be found in `/usr/share/sendmail`. See the `README` file there for more information. If you prefer postfix as MTA, adjust `/etc/mailer.conf`.
   - Edit `/etc/rc.local` to run any local daemons you use.
   - Many of the `/etc` files are documented in section 5 of the manual; so just invoking

     ```
     # man 5 filename
     ```

     is likely to give you more information on these files.

Upgrading a previously-installed NetBSD System

There is no previous release of NetBSD/evbarm from which to upgrade.

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 3.0.3.

Issues affecting an upgrade from NetBSD 2.1 and older releases.

It is **very important** that you populate the directory `/etc/pam.d` with appropriate configuration files for the 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/guide/en/chap-pam.html

for documentation about PAM.

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

```
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.

Using online NetBSD documentation

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

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

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

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

```
# man passwd
```

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

```
# man 5 passwd
```

instead.

If you are unsure of what man page you are looking for, enter

apropos *subject-word*

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

Administrivia

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

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

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

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

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

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

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

Thanks go to

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

  Keith Bostic
  Ralph Campbell
  Mike Karels
  Marshall Kirk McKusick

  for their ongoing work on BSD systems, support, and encouragement.
- The Internet Systems Consortium, Inc. for hosting the NetBSD FTP, CVS, AnonCVS, mail, mail archive, GNATS, SUP, Rsync and WWW servers.
- The Internet Research Institute in Japan for hosting the server which runs the CVSweb interface to the NetBSD source tree.

- The Helsinki University of Technology in Finland for hosting the NetBSD backup CVS and backup server.
- SSH Communications Security in Finland for operating the backup server.
- The many 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.
- The following individuals and organizations (each in alphabetical order) have made donations or loans of hardware and/or money, to support NetBSD development, and deserve credit for it:

  AMD - Advanced Micro Devices, Inc.
  ASA Computers
  Aaron Wall
  ABE Masayuki
  AboveNet Communications, Inc.
  Achim Grolms
  Adam Kasper
  Adaptec
  Advanced System Products, Inc.
  Akihiro IIJIMA
  Alex Poylisher
  Algorithmics, Ltd.
  Alistair Crooks
  Allegro Networks
  Aloys Keller
  Andreas Berg
  Andreas Jellinghaus
  Andrew Brown
  Andrew Gillham
  Andy Hagans
  Antonio Larripa
  Arend Harrewijne
  Armijn Hemel
  Atsushi YOKOYAMA
  Avalon Computer Systems
  Bay Area Internet Solutions
  Ben Collver
  Benoit Lepage
  Bernhard Moellemann
  Bill Coldwell
  Bill Sommerfeld
  Bill Squier
  Brad Salai
  Brains Corporation, Japan
  Brian Carlstrom
  Brian McGroarty
  Brian Mitchell
  Canada Connect Corporation
  Carl Shapiro
  Castor Fu

Central Iowa (Model) Railroad
Charles Conn
Charles D. Cranor
Charles M. Hannum
Chris Legrow
Chris Townsend
Christer O. Andersson
Christopher G. Demetriou
Christos Zoulas
Chuck Silvers
Cologne Chip AG
Computer und Literatur Verlag
Computertechnik Krienke & Nolte
Computing Services Department, The University of Liverpool
Convert Tools
Co-operative Research Centre for Enterprise Distributed
Curt Sampson
DAYOMON from Japan
Damicon Kraa, Finland
Daniel de Kok
Dave Burgess
Dave Rand
Dave Tyson
David Brownlee
Dayton Clark
Demon Internet, UK
Derek Fellion
Digital Equipment Corporation
Distributed Processing Technology
Distro Jockey
Douglas J. Trainor
Dr.ir. F.W. Dillema
Easynet, UK
Ed Braaten
Edward Richley
emuty
Eric and Rosemary Spahr
Erik Berls
Erik E. Fair
Erkki Ruohtula
Ernst Lehmann
Espen Randen
Ewald Kicker
Florent Parent
Frank Kardel
Free Hardware Foundation
Front Range *BSD User Group
FUKAUMI Naoki
Gan Uesli Starling
Garth R. Patil
Geert Hendrickx (ghen)
Geert Jan de Groot

GK Meier
Gordon Zaft
Grant Beattie
Greg Gingerich
Greg Girczyc
Guenther Grau
HP Sweden
Hanno Wagner
Hans Huebner
Harald Koerfgen
Haroon Khalid
Harry McDonald
Hauke Fath
Heiko W. Rupp
Herb Peyerl
Hernani Marques Madeira
Hidekichi Ookubo
Hideyuki Kido
Hisashi Fujinaka
Holger Weiss
Hubert Feyrer
IBM Corporation
IMAI Kiyoshi
Innovation Development Enterprises of America
Intel
Internet Software Consortium
Internet Users Forever IKI
Interoute Telecommunications, UK
JNUG (raised at JNUG meeting & BOF August 2005)
James Bursa
James Chacon
Jan Joris Vereijken
Jason Birnschein
Jason Brazile
Jason R. Thorpe
Jeff Rizzo
Jeff Woodall - Portland, OR
Jens Schoenfeld
Jim Wise
Joachim Nink
Joachim Thiemann
Joel CARNAT
John Heasley
John Kohl
John P. Refling
Jonathan P. Kay
Jordan K. Hubbard
Jorgen Lundman
Karl Wagner
Kenji Hisazumi
Kenneth Alan Hornstein
Kenneth P. Stox

Kevin Keith Woo
Kevin Sullivan
Klaus Lichti
Kimmo Suominen
Korea BSD User Forum
Krister Waldfridsson
Kwok Ng
Lars Mathiassen
Lehmanns Fachbochhandlung
Lex Wennmacher
LinuxFest Northwest
Luke Maurits
Luke Mewburn
MS Macro System GmbH, Germany
Maki Kato
Marc Tooley
Marcus Wyremblewski
Mark Brinicombe
Mark Houde
Mark Perkins
Mark S. Thomas
Mason Loring Bliss
Martin Cernohorsky
Martin J. Ekendahl
Matt Dainty
Matt Thomas
Matthew Jacob
Matthew Sporleder
Matthias Scheler
Mattias Karlsson
Mel Kravitz
Michael Graff
Michael "Kvedulv" Moll
Michael L. Hitch
Michael Richardson
Michael Thompson
Michael W. James
Mike Price
Mirko Thiesen (Thiesi)
Murphy Software BV, Netherlands
Neil J. McRae
Noah M. Keiserman
Norman R. McBride
Numerical Aerospace Simulation Facility, NASA Ames Research
Olaf "Rhialto" Seibert
Oliver Cahagne
Oppedahl & Larson LLP
Palle Lyckegaard
Paul Ripke
Paul Southworth
Pawel Rogocz
Pearson Education

Perry E. Metzger
Petar Bogdanovic
Peter C. Wallace
Peter J. Bui
Peter Postma
Petri T. Koistinen
Phil Thomas
Piermont Information Systems Inc.
Pierre-Philipp Braun
Precedence Technologies Ltd
Public Access Networks Corporation
Ralph Campbell
Randy Ray
Real Weasel
Reinoud Zandijk
Renewed Health Company
Richard Nelson
Rob Windsor
Robert Pankratz
Robert Thille
Roland Lichti
Ross Harvey
Ryan Campbell
SDF Public Access Unix, Inc. 501(c)(7)
SMC Networks Inc.
Salient Systems Inc.
Sander van Dijk
Scott Ellis
Scott Kaplan
Scott Walters
Sean Davis
Simon Burge
Soren Jacobsen
Soren Jorvang
Stephen Borrill
Stephen Early
Steve Allen
Steve Wadlow
Steven M. Bellovin
SunROOT# Project
Sylvain Schmitz
Takahiro Kambe
TAKEUCHI Yoji
Tamotsu Kanoh
Tasis Michalakopoulos (Athens, Greece)
Tatoku Ogaito
Ted Lemon
Ted Spradley
The Names Database
The NetBSD Mission
The People's Republic of Ames
Thierry Lacoste

Thierry Laronde
Thomas Runge
Thor Lancelot Simon
Tim Law
Timo Scholer
Tino Hanich
Tino Wildenhain
Tom Coulter
Tom Ivar Helbekkmo
Tom Lyon
Tomas Dabasinskas
Torsten Harenberg
Toru Nishimura
Toshiba
Turbocat's Development
Tyler Sarna
UTN Web Directory
VMC Harald Frank, Germany
Warped Communications, Inc.
Wasabi Systems, Inc.
Whitecross Database Systems Ltd.
William Gnadt
Worria Affordable Web Hosting
Worria Web Hosting
wwwTrace Traceroute Server Directory
Yusuke Yokota
Zach Metzinger

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

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

We are...

(in alphabetical order)

The NetBSD core group:

Allen Briggs	`briggs@NetBSD.org`
Matt Thomas	`matt@NetBSD.org`
Valeriy E. Ushakov	`uwe@NetBSD.org`
YAMAMOTO Takashi	`yamt@NetBSD.org`
Christos Zoulas	`christos@NetBSD.org`

The portmasters (and their ports):

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

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

The NetBSD 3.0.3 Release Engineering team:

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

NetBSD Developers:

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

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
D'Arcy J.M. Cain	darcy@NetBSD.org
Dave Carrel	carrel@NetBSD.org
Daniel Carosone	dan@NetBSD.org
James Chacon	jmc@NetBSD.org
Bill Coldwell	billc@NetBSD.org
Julian Coleman	jdc@NetBSD.org
Ben Collver	ben@NetBSD.org
Jeremy Cooper	jeremy@NetBSD.org
Chuck Cranor	chuck@NetBSD.org
Alistair Crooks	agc@NetBSD.org
Aidan Cully	aidan@NetBSD.org
Johan Danielsson	joda@NetBSD.org
John Darrow	jdarrow@NetBSD.org
Jed Davis	jld@NetBSD.org
Matt DeBergalis	deberg@NetBSD.org
Rob Deker	deker@NetBSD.org
Chris G. Demetriou	cgd@NetBSD.org
Tracy Di Marco White	gendalia@NetBSD.org
Jaromír Dolecek	jdolecek@NetBSD.org
Andy Doran	ad@NetBSD.org
Roland Dowdeswell	elric@NetBSD.org
Emmanuel Dreyfus	manu@NetBSD.org
Matthias Drochner	drochner@NetBSD.org
Jun Ebihara	jun@NetBSD.org
Elad Efrat	elad@NetBSD.org
Håvard Eidnes	he@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
Hubert Feyrer	hubertf@NetBSD.org
Jason R. Fink	jrf@NetBSD.org
Matt Fredette	fredette@NetBSD.org
Thorsten Frueauf	frueauf@NetBSD.org
Castor Fu	castor@NetBSD.org

Ichiro Fukuhara	ichiro@NetBSD.org
Quentin Garnier	cube@NetBSD.org
Thomas Gerner	thomas@NetBSD.org
Simon J. Gerraty	sjg@NetBSD.org
Justin Gibbs	gibbs@NetBSD.org
Chris Gilbert	chris@NetBSD.org
Eric Gillespie	epg@NetBSD.org
Adam Glass	glass@NetBSD.org
Michael Graff	explorer@NetBSD.org
Brian C. Grayson	bgrayson@NetBSD.org
Matthew Green	mrg@NetBSD.org
Andreas Gustafsson	gson@NetBSD.org
Jun-ichiro itojun Hagino	itojun@NetBSD.org
Juergen Hannken-Illjes	hannken@NetBSD.org
Charles M. Hannum	mycroft@NetBSD.org
Ben Harris	bjh21@NetBSD.org
Ross Harvey	ross@NetBSD.org
Eric Haszlakiewicz	erh@NetBSD.org
John Hawkinson	jhawk@NetBSD.org
HAMAJIMA Katsuomi	hamajima@NetBSD.org
HAYAKAWA Koichi	haya@NetBSD.org
John Heasley	heas@NetBSD.org
Geert Hendrickx	ghen@NetBSD.org
René Hexel	rh@NetBSD.org
Kouichirou Hiratsuka	hira@NetBSD.org
Michael L. Hitch	mhitch@NetBSD.org
Christian E. Hopps	chopps@NetBSD.org
Ken Hornstein	kenh@NetBSD.org
Marc Horowitz	marc@NetBSD.org
Eduardo Horvath	eeh@NetBSD.org
Nick Hudson	skrll@NetBSD.org
Shell Hung	shell@NetBSD.org
Martin Husemann	martin@NetBSD.org
Dean Huxley	dean@NetBSD.org
Love Hörnquist Åstrand	lha@NetBSD.org
Bernardo Innocenti	bernie@NetBSD.org
Tetsuya Isaki	isaki@NetBSD.org
ITOH Yasufumi	itohy@NetBSD.org
IWAMOTO Toshihiro	toshii@NetBSD.org
Matthew Jacob	mjacob@NetBSD.org
Soren Jacobsen	snj@NetBSD.org
Lonhyn T. Jasinskyj	lonhyn@NetBSD.org
Darrin Jewell	dbj@NetBSD.org
Chris Jones	cjones@NetBSD.org
Søren Jørvang	soren@NetBSD.org
Takahiro Kambe	taca@NetBSD.org
Antti Kantee	pooka@NetBSD.org
Masanori Kanaoka	kanaoka@NetBSD.org
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
Paul Kranenburg	pk@NetBSD.org
Martti Kuparinen	martti@NetBSD.org
Kentaro A. Kurahone	kurahone@NetBSD.org
Kevin Lahey	kml@NetBSD.org
Johnny C. Lam	jlam@NetBSD.org
Martin J. Laubach	mjl@NetBSD.org
Greg Lehey	grog@NetBSD.org
Ted Lemon	mellon@NetBSD.org
Christian Limpach	cl@NetBSD.org
Frank van der Linden	fvdl@NetBSD.org
Joel Lindholm	joel@NetBSD.org
Mike Long	mikel@NetBSD.org
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
Anders Magnusson	ragge@NetBSD.org
MAEKAWA Masahide	gehenna@NetBSD.org
David Maxwell	david@NetBSD.org
Dan McMahill	dmcmahill@NetBSD.org
Gregory McGarry	gmcgarry@NetBSD.org
Jared D. McNeill	jmcneill@NetBSD.org
Neil J. McRae	neil@NetBSD.org
Perry Metzger	perry@NetBSD.org
Simas Mockevicius	symka@NetBSD.org
Juan Romero Pardines	xtraeme@NetBSD.org
Julio M. Merino Vidal	jmmv@NetBSD.org
Minoura Makoto	minoura@NetBSD.org
Luke Mewburn	lukem@NetBSD.org
der Mouse	mouse@NetBSD.org
Joseph Myers	jsm@NetBSD.org
Ken Nakata	kenn@NetBSD.org
Takeshi Nakayama	nakayama@NetBSD.org
Phil Nelson	phil@NetBSD.org
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
Jesse Off	joff@NetBSD.org
Tatoku Ogaito	tacha@NetBSD.org
OKANO Takayoshi	kano@NetBSD.org
Masaru Oki	oki@NetBSD.org
Atsushi Onoe	onoe@NetBSD.org
Greg Oster	oster@NetBSD.org
Jonathan Perkin	sketch@NetBSD.org
Herb Peyerl	hpeyerl@NetBSD.org
Matthias Pfaller	matthias@NetBSD.org
Chris Pinnock	cjep@NetBSD.org
Adrian Portelli	adrianp@NetBSD.org
Rui Paulo	rpaulo@NetBSD.org
Peter Postma	peter@NetBSD.org
Dante Profeta	dante@NetBSD.org
Chris Provenzano	proven@NetBSD.org
Niels Provos	provos@NetBSD.org
Michael Rauch	mrauch@NetBSD.org
Marc Recht	recht@NetBSD.org
Darren Reed	darrenr@NetBSD.org
Jeremy C. Reed	reed@NetBSD.org
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
Jeff Rizzo	riz@NetBSD.org
Gordon Ross	gwr@NetBSD.org
Steve Rumble	rumble@NetBSD.org
Ilpo Ruotsalainen	lonewolf@NetBSD.org
Heiko W. Rupp	hwr@NetBSD.org
David Sainty	dsainty@NetBSD.org
SAITOH Masanobu	msaitoh@NetBSD.org
Kazuki Sakamoto	sakamoto@NetBSD.org
Curt Sampson	cjs@NetBSD.org
Wilfredo Sanchez	wsanchez@NetBSD.org
Ty Sarna	tsarna@NetBSD.org
SATO Kazumi	sato@NetBSD.org
Jan Schaumann	jschauma@NetBSD.org
Matthias Scheler	tron@NetBSD.org
Karl Schilke (rAT)	rat@NetBSD.org
Amitai Schlair	schmonz@NetBSD.org
Konrad Schroder	perseant@NetBSD.org
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
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
Bill Squier	groo@NetBSD.org
Jonathan Stone	jonathan@NetBSD.org
Bill Studenmund	wrstuden@NetBSD.org
Kevin Sullivan	sullivan@NetBSD.org
SUNAGAWA Keiki	kei@NetBSD.org
Kimmo Suominen	kim@NetBSD.org
Shin Takemura	takemura@NetBSD.org
TAMURA Kent	kent@NetBSD.org
Shin'ichiro TAYA	taya@NetBSD.org
Ian Lance Taylor	ian@NetBSD.org
Matt Thomas	matt@NetBSD.org
Jason Thorpe	thorpej@NetBSD.org
Christoph Toshok	toshok@NetBSD.org
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
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
Nathan Williams	nathanw@NetBSD.org
Rob Windsor	windsor@NetBSD.org
Dan Winship	danw@NetBSD.org
Jim Wise	jwise@NetBSD.org
Michael Wolfson	mbw@NetBSD.org

Steve Woodford	scw@NetBSD.org
Colin Wood	ender@NetBSD.org
YAMAMOTO Takashi	yamt@NetBSD.org
Yuji Yamano	yyamano@NetBSD.org
Reinoud Zandijk	reinoud@NetBSD.org
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:

This product includes software developed by the University of California, Berkeley and its contributors.
This product includes software developed by the NetBSD Foundation.
This product includes software developed by The NetBSD Foundation, Inc. and its contributors.
This product includes software developed for the NetBSD Project. See http://www.netbsd.org/ for information about NetBSD.
This product contains software developed by Ignatios Souvatzis for the NetBSD project.
This product contains software written by Ignatios Souvatzis and Michael L. Hitch for the NetBSD project.
This product contains software written by Michael L. Hitch for the NetBSD project.
This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)
This product includes cryptographic software written by Eric Young (eay@mincom.oz.au)
This product includes software designed by William Allen Simpson.
This product includes software developed at Ludd, University of Lulea, Sweden and its contributors.
This product includes software developed at Ludd, University of Lulea.
This product includes software developed at the Information Technology Division, US Naval Research Laboratory.
This product includes software developed by Berkeley Software Design, Inc.
This product includes software developed by David Jones and Gordon Ross
This product includes software developed by Gordon W. Ross and Leo Weppelman.
This product includes software developed by Hellmuth Michaelis and Joerg Wunsch
This product includes software developed by Internet Research Institute, Inc.
This product includes software developed by Leo Weppelman and Waldi Ravens.
This product includes software developed by Mika Kortelainen
This product includes software developed by Aaron Brown and Harvard University.
This product includes software developed by Adam Ciarcinski for the NetBSD project.
This product includes software developed by Adam Glass and Charles M. Hannum.
This product includes software developed by Adam Glass.
This product includes software developed by Advanced Risc Machines Ltd.
This product includes software developed by Alex Zepeda, and Colin Wood for the NetBSD Projet.
This product includes software developed by Alex Zepeda.
This product includes software developed by Alistair G. Crooks.
This product includes software developed by Alistair G. Crooks. for the NetBSD project.
This product includes software developed by Allen Briggs
This product includes software developed by Amancio Hasty and Roger Hardiman
This product includes software developed by Berkeley Software Design, Inc.
This product includes software developed by Berkeley Software Design, Inc.
This product includes software developed by Bill Paul.
This product includes software developed by Bodo Moeller. (If available, substitute umlauted o for oe)

This product includes software developed by Boris Popov.
This product includes software developed by Brad Pepers
This product includes software developed by Bradley A. Grantham.
This product includes software developed by Brini.
This product includes software developed by Causality Limited.
This product includes software developed by Charles D. Cranor and Seth Widoff.
This product includes software developed by Charles D. Cranor and Washington University.
This product includes software developed by Charles D. Cranor, Washington University, and the University of California, Berkeley and its contributors.
This product includes software developed by Charles D. Cranor, Washington University, the University of California, Berkeley and its contributors.
This product includes software developed by Charles D. Cranor.
This product includes software developed by Charles Hannum.
This product includes software developed by Charles M. Hannum, by the University of Vermont and State Agricultural College and Garrett A. Wollman, by William F. Jolitz, and by the University of California, Berkeley, Lawrence Berkeley Laboratory, and its contributors.
This product includes software developed by Charles M. Hannum.
This product includes software developed by Christian E. Hopps, Ezra Story, Kari Mettinen, Markus Wild, Lutz Vieweg and Michael Teske.
This product includes software developed by Christian E. Hopps.
This product includes software developed by Christian Limpach
This product includes software developed by Christopher G. Demetriou for the NetBSD Project.
This product includes software developed by Christopher G. Demetriou.
This product includes software developed by Christos Zoulas.
This product includes software developed by Chuck Silvers.
This product includes software developed by Colin Wood for the NetBSD Project.
This product includes software developed by Colin Wood.
This product includes software developed by Cybernet Corporation and Nan Yang Computer Services Limited
This product includes software developed by Daishi Kato
This product includes software developed by Dale Rahn.
This product includes software developed by Daniel Widenfalk and Michael L. Hitch.
This product includes software developed by Daniel Widenfalk for the NetBSD Project.
This product includes software developed by Darrin B. Jewell
This product includes software developed by David Miller.
This product includes software developed by Dean Huxley.
This product includes software developed by Eduardo Horvath.
This product includes software developed by Eric S. Hvozda.
This product includes software developed by Eric S. Raymond
This product includes software developed by Eric Young (eay@@mincom.oz.au)
This product includes software developed by Eric Young (eay@cryptsoft.com)
This product includes software developed by Eric Young (eay@mincom.oz.au)
This product includes software developed by Ezra Story and by Kari Mettinen.
This product includes software developed by Ezra Story, by Kari Mettinen and by Bernd Ernesti.
This product includes software developed by Ezra Story, by Kari Mettinen, Michael Teske and by Bernd Ernesti.
This product includes software developed by Ezra Story, by Kari Mettinen, and Michael Teske.
This product includes software developed by Ezra Story.
This product includes software developed by Frank van der Linden for the NetBSD Project.
This product includes software developed by Gardner Buchanan.
This product includes software developed by Gary Thomas.
This product includes software developed by Gordon Ross
This product includes software developed by Gordon W. Ross
This product includes software developed by HAYAKAWA Koichi.
This product includes software developed by Harvard University and its contributors.
This product includes software developed by Harvard University.
This product includes software developed by Herb Peyerl.
This product includes software developed by Hubert Feyrer for the NetBSD Project.
This product includes software developed by Iain Hibbert
This product includes software developed by Ian F. Darwin and others.
This product includes software developed by Ian W. Dall.
This product includes software developed by Ichiro FUKUHARA.
This product includes software developed by Ignatios Souvatzis for the NetBSD Project.
This product includes software developed by Internet Initiative Japan Inc.
This product includes software developed by James R. Maynard III.
This product includes software developed by Jared D. McNeill.
This product includes software developed by Jason L. Wright
This product includes software developed by Jason R. Thorpe for And Communications, http://www.and.com/

This product includes software developed by Joachim Koenig-Baltes.
This product includes software developed by Jochen Pohl for The NetBSD Project.
This product includes software developed by Joerg Wunsch
This product includes software developed by John Birrell.
This product includes software developed by John P. Wittkoski.
This product includes software developed by John Polstra.
This product includes software developed by Jonathan R. Stone for the NetBSD Project.
This product includes software developed by Jonathan Stone and Jason R. Thorpe for the NetBSD Project.
This product includes software developed by Jonathan Stone.
This product includes software developed by Jukka Marin.
This product includes software developed by Julian Highfield.
This product includes software developed by Kazuhisa Shimizu.
This product includes software developed by Kazuki Sakamoto.
This product includes software developed by Kenneth Stailey.
This product includes software developed by Kiyoshi Ikehara.
This product includes software developed by Klaus Burkert,by Bernd Ernesti, by Michael van Elst, and by the University of California, Berkeley and its contributors.
This product includes software developed by LAN Media Corporation and its contributors.
This product includes software developed by Leo Weppelman for the NetBSD Project.
This product includes software developed by Leo Weppelman.
This product includes software developed by Lloyd Parkes.
This product includes software developed by Luke Mewburn.
This product includes software developed by Lutz Vieweg.
This product includes software developed by MINOURA Makoto, Takuya Harakawa.
This product includes software developed by Manuel Bouyer.
This product includes software developed by Marc Horowitz.
This product includes software developed by Marcus Comstedt.
This product includes software developed by Mark Brinicombe for the NetBSD project.
This product includes software developed by Mark Brinicombe.
This product includes software developed by Mark Murray
This product includes software developed by Mark Tinguely and Jim Lowe
This product includes software developed by Markus Wild.
This product includes software developed by Martin Husemann and Wolfgang Solfrank.
This product includes software developed by Masanobu Saitoh.
This product includes software developed by Masaru Oki.
This product includes software developed by Mats O Jansson and Charles D. Cranor.
This product includes software developed by Mats O Jansson.
This product includes software developed by Matt DeBergalis
This product includes software developed by Matthew Fredette.
This product includes software developed by Matthias Pfaller.
This product includes software developed by Michael Graff for the NetBSD Project.
This product includes software developed by Michael Graff.
This product includes software developed by Michael L. Hitch.
This product includes software developed by Michael Shalayeff.
This product includes software developed by Michael Smith.
This product includes software developed by Mike Glover and contributors.
This product includes software developed by Mike Pritchard.
This product includes software developed by Minoura Makoto.
This product includes software developed by Nan Yang Computer Services Limited.
This product includes software developed by Niels Provos.
This product includes software developed by Niklas Hallqvist, Brandon Creighton and Job de Haas.
This product includes software developed by Niklas Hallqvist.
This product includes software developed by Onno van der Linden.
This product includes software developed by Paul Kranenburg.
This product includes software developed by Paul Mackerras.
This product includes software developed by Per Fogelstrom
This product includes software developed by Peter Galbavy.
This product includes software developed by Phase One, Inc.
This product includes software developed by Philip A. Nelson.
This product includes software developed by Philip L. Budne.
This product includes software developed by RiscBSD.
This product includes software developed by Roar Thronæs.
This product includes software developed by Rodney W. Grimes.
This product includes software developed by Roger Hardiman
This product includes software developed by Roland C. Dowdeswell.
This product includes software developed by Rolf Grossmann.

This product includes software developed by Ross Harvey for the NetBSD Project.
This product includes software developed by Ross Harvey.
This product includes software developed by Scott Bartram.
This product includes software developed by Scott Stevens.
This product includes software developed by Shingo WATANABE.
This product includes software developed by Softweyr LLC, the University of California, Berkeley, and its contributors.
This product includes software developed by Soren S. Jorvang.
This product includes software developed by Stephan Thesing.
This product includes software developed by Steve Woodford.
This product includes software developed by Takashi Hamada
This product includes software developed by Takumi Nakamura.
This product includes software developed by Tatoku Ogaito for the NetBSD Project.
This product includes software developed by Terrence R. Lambert.
This product includes software developed by Tetsuya Isaki.
This product includes software developed by Thomas Gerner
This product includes software developed by Thomas Klausner for the NetBSD Project.
This product includes software developed by Tobias Weingartner.
This product includes software developed by Todd C. Miller.
This product includes software developed by Tohru Nishimura and Reinoud Zandijk for the NetBSD Project.
This product includes software developed by Tohru Nishimura for the NetBSD Project.
This product includes software developed by Tohru Nishimura. for the NetBSD Project.
This product includes software developed by TooLs GmbH.
This product includes software developed by Toru Nishimura.
This product includes software developed by Trimble Navigation, Ltd.
This product includes software developed by WIDE Project and its contributors.
This product includes software developed by Waldi Ravens.
This product includes software developed by Wasabi Systems for Zembu Labs, Inc. http://www.zembu.com/
This product includes software developed by Winning Strategies, Inc.
This product includes software developed by Wolfgang Solfrank.
This product includes software developed by Yasushi Yamasaki
This product includes software developed by Zembu Labs, Inc.
This product includes software developed by the Alice Group.
This product includes software developed by the Center for Software Science at the University of Utah.
This product includes software developed by the Charles D. Cranor, Washington University, University of California, Berkeley and its contributors.
This product includes software developed by the Computer Systems Engineering Group at Lawrence Berkeley Laboratory.
This product includes software developed by the David Muir Sharnoff.
This product includes software developed by the Harvard University and its contributors.
This product includes software developed by the Kungliga Tekniska Högskolan and its contributors.
This product includes software developed by the Network Research Group at Lawrence Berkeley Laboratory.
This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.OpenSSL.org/)
his product includes software developed by the PocketBSD project and its contributors.
This product includes software developed by the RiscBSD kernel team
This product includes software developed by the RiscBSD team.
This product includes software developed by the SMCC Technology Development Group at Sun Microsystems, Inc.
This product includes software developed by the University of California, Berkeley and its contributors, as well as the Trustees of Columbia University.
This product includes software developed by the University of California, Lawrence Berkeley Laboratory and its contributors.
This product includes software developed by the University of California, Lawrence Berkeley Laboratory.
This product includes software developed by the University of Illinois at Urbana and their contributors.
This product includes software developed by the University of Vermont and State Agricultural College and Garrett A. Wollman.
This product includes software developed by the University of Vermont and State Agricultural College and Garrett A. Wollman, by William F. Jolitz, and by the University of California, Berkeley, Lawrence Berkeley Laboratory, and its contributors.
This product includes software developed for the FreeBSD project
This product includes software developed for the NetBSD Project by Bernd Ernesti.
This product includes software developed for the NetBSD Project by Christopher G. Demetriou.
This product includes software developed for the NetBSD Project by Christos Zoulas
This product includes software developed for the NetBSD Project by Emmanuel Dreyfus.
This product includes software developed for the NetBSD Project by Frank van der Linden
This product includes software developed for the NetBSD Project by Ignatios Souvatzis.
This product includes software developed for the NetBSD Project by Jason R. Thorpe.
This product includes software developed for the NetBSD Project by John M. Vinopal.

This product includes software developed for the NetBSD Project by Matthias Drochner.
This product includes software developed for the NetBSD Project by Michael L. Hitch.
This product includes software developed for the NetBSD Project by Perry E. Metzger.
This product includes software developed for the NetBSD Project by Scott Bartram and Frank van der Linden
This product includes software developed for the NetBSD Project by Allegro Networks, Inc., and Wasabi Systems, Inc.
This product includes software developed for the NetBSD Project by Genetec Corporation.
This product includes software developed for the NetBSD Project by Jonathan Stone.
This product includes software developed for the NetBSD Project by Piermont Information Systems Inc.
This product includes software developed for the NetBSD Project by SUNET, Swedish University Computer Network.
This product includes software developed for the NetBSD Project by Shigeyuki Fukushima.
This product includes software developed for the NetBSD Project by Wasabi Systems, Inc.
This product includes software developed under OpenBSD by Per Fogelstrom Opsycon AB for RTMX Inc, North Carolina, USA.
This product includes software developed under OpenBSD by Per Fogelstrom.
This software is a component of "386BSD" developed by William F. Jolitz, TeleMuse.
This software was developed by Holger Veit and Brian Moore for use with "386BSD" and similar operating systems. "Similar operating systems" includes mainly non-profit oriented systems for research and education, including but not restricted to "NetBSD", "FreeBSD", "Mach" (by CMU).
This software includes software developed by the Computer Systems Laboratory at the University of Utah.
This product includes software developed by Computing Services at Carnegie Mellon University (http://www.cmu.edu/computing/).

This product includes software developed by Advanced Risc Machines Ltd.
This product includes software developed by Neil Carson.
This product includes software developed by the RiscBSD kernel team.
This product includes software developed by Scott Stevens.

The End