NAME

INSTALL – Installation procedure for NetBSD/sun3.

DESCRIPTION

About this Document

This document describes the installation procedure for NetBSD 1.5 on the *sun3* platform. It is available in four different formats titled `INSTALL.`*ext*, where *ext* is one of `.ps`, `.html`, `.more`, or `.txt`:

`.ps` PostScript.

`.html` Standard Internet HTML.

`.more` The enhanced text format used on UNIX-like systems by the `more`(1) and `less`(1) pager utility programs. This is the format in which the on-line *man* pages are generally presented.

`.txt` Plain old ASCII.

You are reading the *PostScript* version.

What is NetBSD?

The NetBSD Operating System is a fully functional Open Source UNIX-like operating system derived from the University of California, Berkeley Networking Release 2 (Net/2), 4.4BSD-Lite, and 4.4BSD-Lite2 sources. NetBSD runs on thirty-one different system architectures featuring twelve distinct families of CPUs, and is being ported to more. The NetBSD 1.5 release contains complete binary releases for fifteen different machine types. (The sixteen remaining are not fully supported at this time and are thus not part of the binary distribution. For information on them, please see the NetBSD web site at **http://www.netbsd.org/**)

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

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

Changes Since The Last Release

The NetBSD 1.5 release provides numerous significant functional enhancements, including support for many new devices, integration of hundreds of bug fixes, new and updated kernel subsystems, and many userland enhancements. The result of these improvements is a stable operating system fit for production use that rivals most commercially available systems.

It is impossible to completely summarize over one year of development that went into the NetBSD 1.5 release. Some highlights include:

Kernel

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

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

Networking

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

File system

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

Security

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

System administration and user tools

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

Miscellaneous

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

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

Kernel interfaces have continued to be refined, and more subsystems and device drivers are shared among the different ports. You can look for this trend to continue.

This is the fifth major release of NetBSD/sun3.

The Future of NetBSD

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

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

- providing better organization to keep track of development efforts, including co-ordination with groups working in related fields.
- providing a framework to receive donations of goods and services and to own the resources necessary to run the NetBSD Project.

- providing a better position from which to undertake promotional activities.
- periodically organizing workshops for developers and other interested people to discuss ongoing work.

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

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

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

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

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

Sources of NetBSD

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

NetBSD 1.5 Release Contents

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

`.../NetBSD-1.5/`

`CHANGES` Changes since earlier NetBSD releases.

`LAST_MINUTE` Last minute changes.

`MIRRORS` A list of sites that mirror the NetBSD 1.5 distribution.

`README.files`
README describing the distribution's contents.

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

`patches/` Post-release source code patches.

`source/` Source distribution sets; see below.

In addition to the files and directories listed above, there is one directory per architecture, for each of the architectures for which NetBSD 1.5 has a binary distribution. There are also `README.export-control` files sprinkled liberally throughout the distribution tree, which point out that there are some portions of the distribution that may be subject to export regulations of the United States, e.g. code under `src/crypto` and `src/sys/crypto`. It is your responsibility to determine whether or not it is legal for you to export these portions and to act accordingly.

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

gnusrc This set contains the ''gnu'' sources, including the source for the compiler, assembler, groff, and the other GNU utilities in the binary distribution sets.
22.3 MB gzipped, 98.8 MB uncompressed

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

sharesrc This set contains the ''share'' sources, which include the sources for the man pages not associated with any particular program, the sources for the typesettable document set, the dictionaries, and more.
3.3 MB gzipped, 13.2 MB uncompressed

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

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

xsrc This set contains the sources to the X Window System.
35.2 MB gzipped, 176.8 MB uncompressed

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

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

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

The `sets/Split/` subdirectory contains split versions of the source sets for those users who need to load the source sets from floppy or otherwise need a split distribution. The split sets are named `set_name.`*xx* where `set_name` is the distribution set name, and *xx* is the sequence number of the file, starting with ''aa'' for the first file in the distribution set, then ''ab'' for the next, and so on. All of these files except the last one of each set should be exactly 240,640 bytes long. (The last file is just long enough to contain the remainder of the data for that distribution set.)

The split distributions may be reassembled and extracted with **cat** as follows:

```
# cat set_name.?? | ( cd / ; tar -zxpf - )
```

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

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

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

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

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

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

NetBSD/sun3 subdirectory structure

The sun3-specific portion of the NetBSD 1.5 release is found in the `sun3` subdirectory of the distribution:
`.../NetBSD-1.5/sun3/`
`INSTALL.html`

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

`binary/`
`kernel/`
`netbsd.GENERIC.gz` A gzipped NetBSD kernel containing code for everything supported in this release.
`sets/` sun3 binary distribution sets; see below.

`installation/`
`floppy/` sun3 boot and installation floppies; see below.
`miniroot/` sun3 miniroot file system image; see below.
`misc/` Miscellaneous sun3 installation utilities; see installation section, below.
`netboot/` Two programs needed to boot sun3 kernels over the network.
`tapeimage/` Tape boot programs, and a RAMDISK kernel.

Binary distribution sets

The NetBSD sun3 binary distribution sets contain the binaries which comprise the NetBSD 1.5 release for the sun3. There are eight binary distribution sets. The binary distribution sets can be found in the `sun3/binary/sets` subdirectory of the NetBSD 1.5 distribution tree, and are as follows:

base The NetBSD 1.5 sun3 **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.
14.5 MB gzipped, 38.1 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.
9.5 MB gzipped, 32.5 MB uncompressed

etc This distribution set contains the system configuration files that reside in `/etc` and in several other places. This set *must* be installed if you are installing the system from scratch, but should *not* be used if you are upgrading. (If you are upgrading, it's recommended that you get a copy of this set and *carefully* upgrade your configuration files by hand.)
0.1 MB gzipped, 0.6 MB uncompressed

games This set includes the games and their manual pages.
3.0 MB gzipped, 7.4 MB uncompressed

kern This set contains a NetBSD/sun3 1.5 `GENERIC` kernel, named `/netbsd`. You *must* install this distribution set.

man This set includes all of the manual pages for the binaries and other software contained in the **base** set. Note that it does not include any of the manual pages that are included in the other sets.
5.0 MB gzipped, 19.9 MB uncompressed

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

text This set includes NetBSD 's text processing tools, including `groff`(1), all related programs, and their manual pages.
1.3 MB gzipped, 4.7 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 3.3.6. Binary sets for the X Window system are distributed with NetBSD. The sets are:

xbase The basic files needed for a complete X client environment. This does not include the X servers.
2.6 MB gzipped, 8.2 MB uncompressed

xcomp The extra libraries and include files needed to compile X source code.
1.7 MB gzipped, 7.2 MB uncompressed

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

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

The sun3 binary distribution sets are distributed as gzipped tar files named with the extension **.tgz**, e.g. `base.tgz`.

The instructions given for extracting the source sets work equally well for the binary sets, but it is worth noting that if you use that method, the files are `/`-relative and therefore are extracted *below* the current directory. That is, if you want to extract the binaries into your system, i.e. replace the system binaries with them, you have to run the **`tar -xpf`** command from `/`.

There is a collection of Sun3 and Sun3X kernels in the `sun3/binary/kernels` subdirectory of the NetBSD 1.5 distribution. The ones named `netbsd-ramdisk*.gz` contain a root file system image and should only be used for the initial installation. The others are included for convenience. (Most people will want to use `netbsd-generic.gz` or `netbsd-generic3x.gz` as appropriate.) Please note that these kernels are simply gzipped and are not tar archives.

Note: Each directory in the sun3 binary distribution also has its own checksum files, just as the source distribution does.

NetBSD/sun3 System Requirements and Supported Devices

NetBSD/sun3 1.5 runs on most Sun3 machines, including:

3/50	3/60	3/110
3/75	3/150	3/160
3/260	3/280	3/E
3/80	3/470	

Note that NetBSD/sun3 now includes support for 'Sun3X' machines, which used to be supported with a separate NetBSD/sun3x distribution.

The minimal configuration requires 4 MB of RAM and about 80 MB of disk space. To install the entire system requires much more disk space (approx. 100 MB additional space is necessary for full sources). To run X or compile the system, more RAM is recommended. Good performance requires 8 MB of RAM, or 16 MB when running the X Window System.

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

Partition	Suggested	+ X	Needed	+ X
/ (root)	20 MB	20 MB	15 MB	15 MB
/usr	175 MB	205 MB	75 MB	105 MB
/var	20 MB	20 MB	5 MB	5 MB
swap	*2*RAM ...*			

Anything else is up to you!

As you may note, the recommended size of `/usr` is 100 MB greater than needed. This is to leave room for a kernel source and compile tree as you will probably want to compile your own kernel. (`GENERIC` is large and bulky to accommodate all people).

Note that the sun3 installation procedure uses a *miniroot* image which is placed into the swap area of the disk. The swap partition must be at least as large as the miniroot image (10 MB).

Supported hardware

- Serial ports (RS232)
  - built-in ttya, ttyb
- Video adapters
  - bwtwo
  - cgtwo
  - cgfour
- Network interfaces:
  - On-board Lance Ethernet
  - On-board or VME Intel Ethernet
  - Sun3/E SCSI/Ethernet board
- SCSI
  - Most SCSI disks, tapes, CD-ROMs, etc
  - On-board Sun3/80 SCSI (*esp*)
  - On-board SCSI-3 (*si*)
  - VME SCSI-3 board (*si*)
  - Sun3/E SCSI/Ethernet board
- SMD Disks
  - Xylogics 450/451
  - Xylogics 753/7053
- Input devices
  - Sun keyboard and mouse
- Miscellaneous
  - Battery-backed real-time clock.
  - On-board floppy controller (Sun3/80 floppy)

If it's not on this list, there is no support for it in this release.

Getting the NetBSD System on to Useful Media

Installation is supported from several media types, including:

- Tape
- NFS
- CD-ROM

- FTP

Note: Installing on a 'bare' machine requires some bootable device; either a tape drive or Sun-compatible NFS server.

The procedure for transferring the distribution sets onto installation media depends on the type of media. Instructions for each type of media are given below.

In order to create installation media, you will need all the files in the directory

```
.../NetBSD-1.5/sun3/
```

Creating boot/install tapes

Installing from tape is the simplest method of all. This method uses two tapes; one called the *boot* tape, and another called the *install* tape.

The boot tape is created as follows:

```
# cd .../NetBSD-1.5/sun3/installation/tapeimage
# sh MakeBootTape /dev/nrst0
```

The install tape is created as follows:

```
# cd .../NetBSD-1.5/sun3/installation/tapeimage
# sh MakeInstallTape /dev/nrst0
```

If the tapes do not work as expected, you may need to explicitly set the EOF mark at the end of each tape segment. It may also be necessary to use the **conv=osync** argument to dd(1). Note that this argument is incompatible with the **bs=** argument. Consult the tape-related manual pages on the system where the tapes are created for more details.

Boot/Install from NFS server

If your machine has a disk and network connection, but no tape drive, it may be convenient for you to install NetBSD over the network. This involves temporarily booting your machine over NFS, just long enough so you can initialize its disk. This method requires that you have access to an NFS server on your network so you can configure it to support diskless boot for your machine. Configuring the NFS server is normally a task for a system administrator, and is not trivial.

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

When instructed to boot over the network, your sun3 expects to be able to download a second stage bootstrap program via TFTP after it has acquired its IP address through RARP. It will attempt to download a file using a name derived from the machine's recently aquired IP address. (It may be handy to have a hexadecimal calculator for this next step.) The filename is created by converting the machine's assigned IP address into hexadecimal, most-significant octet first, using uppercase characters for the non-decimal (A-F) digits.

For example, a sun3 which has been assigned IP address 130.115.144.11 will make an TFTP request for 8273900B. Normally, this file is a symbolic link to the NetBSD/sun3 netboot program, which should be located in a place where the TFTP daemon can find it. (Remember, many TFTP daemons run in a chroot'ed environment.) The netboot program may be found in the install directory of this distribution.

The netboot program will query a bootparamd server to find the NFS server address and path name for its root, and then load a kernel from that location. The server should have a copy of the netbsd-rd kernel in the root area for your client (no other files are needed in the client root) and /etc/bootparams on the server should have an entry for your client and its root directory. The client will need access to the miniroot image, which can be provided using NFS or remote shell.

If you will be installing NetBSD on several clients, it may be useful to know that you can use a single NFS root for all the clients as long as they only use the netbsd-rd kernel. There will be no conflict between clients because the RAM-disk kernel will not use the NFS root. No swap file is needed; the RAM-disk kernel does not use that either.

Install/Upgrade from CD-ROM

This method requires that you boot from another device (i.e. tape or network, as described above). You may need to make a boot tape on another machine using the files provided on the CD-ROM. Once you have booted netbsd-rd (the RAM-disk kernel) and loaded the miniroot, you can load any of the distribution sets directly from the CD-ROM. The **install** program in the miniroot automates the work required to mount the CD-ROM and extract the files.

Install/Upgrade via FTP

This method requires that you boot from another device (i.e. tape or network, as described above). You may need to make a boot tape on another machine using the files in `.../install` (which you get via FTP). Once you have booted netbsd-rd (the RAM-disk kernel) and loaded the miniroot, you can load any of the distribution sets over the net using FTP. The **install** program in the miniroot automates the work required to configure the network interface and transfer the files.

The preparations for this installation/upgrade method are easy; all you make sure that there's some FTP site from which you can retrieve the NetBSD distribution when you're about to install or upgrade. You need to know the numeric IP address of that site, and, if it's not on a network directly connected to the machine on which you're installing or upgrading NetBSD, you need to know the numeric IP address of the router closest to the NetBSD machine. Finally, you need to know the numeric IP address of the NetBSD machine itself.

Preparing your System for NetBSD installation

Sun3 machines usually need little or no preparation before installing NetBSD, other than the usual, well advised precaution of *backing up all data* on any attached storage devices.

You will need to know the SCSI target ID of the drive on which you will install NetBSD.

Note: SunOS on the sun3 uses confusing names for the SCSI devices: target 1 is `sd2`, target 2 is `sd4`, etc.

It might be a good time to run the diagnostics on your Sun3. First, attach a terminal to the `ttya` serial port, then set the ''Diag/Norm'' switch to the Diagnostic position, and power-on the machine. The Diag. switch setting forces console interaction to occur on ttya. Note that the 3/80 has a ''software'' diag switch you can set at the PROM monitor prompt. To turn on diag boot mode, do: **q 70b 12** To return to normal boot mode, do: **q 70b 6**.

The console location (ttya, ttyb, or keyboard/display) is controlled by address 0x1F in the EEPROM, which you can examine and change in the PROM monitor by entering **q1f** followed by a numeric value (or just a '**.**' if you don't want to change it). Console values are:

`00` Default graphics display

`10` tty a (9600-N-8-1)

`11` tty b (1200-N-8-1)

`20` Color option board on P4

Installing the NetBSD System

Installing NetBSD is a relatively complex process, but if you have this document in hand it should not be too difficult.

There are several ways to install NetBSD onto your disk. If your machine has a tape drive the easiest way is **Installing from tape** (details below). If your machine is on a network with a suitable NFS server, then **Installing from NFS** is the next best method. Otherwise, if you have another Sun machine running SunOS you can initialize the disk on that machine and then move the disk. (Installing from SunOS is not recommended.)

Installing from tape

Create the NetBSD/sun3 1.5 boot tape as described in the section entitled **Preparing a boot tape** and boot the tape. At the PROM monitor prompt, use one of the commands:

```
>b st()
>b st(0,8,0)
```

The first example will use the tape on SCSI target 4, where the second will use SCSI target 5. The > is the monitor prompt.

After the tape loads, you should see many lines of configuration messages, and then the following 'welcome' screen:

```
          Welcome to the NetBSD/sun3 RAMDISK root!

This environment is designed to do only three things:
1:  Partititon your disk (use the command:  edlabel /dev/rsd0c)
2:  Copy a miniroot image into the swap partition  (/dev/rsd0b)
3:  Reboot (using the swap partition, i.e. /dev/sd?b).

Copying the miniroot can be done several ways, allowing
the source of the miniroot image to be on any of these:
    boot tape,  NFS server, TFTP server, rsh server

The easiest is loading from tape, which is done as follows:
        mt -f /dev/nrst0 rewind
        mt -f /dev/nrst0 fsf 2
        dd bs=32k if=/dev/nrst0 of=/dev/rsd0b
(For help with other methods, please see the install notes.)

To reboot using the swap partition, first use "halt",
then at the PROM monitor prompt use a command like:
        b sd(,,1) -s

To view this message again, type:  cat /.welcome
```

Copy the miniroot as described in the welcome message, and reboot from that just installed miniroot. See the section entitled **Booting the miniroot** for details.

Installing from NFS

Before you can install from NFS, you must have already configured your NFS server to support your machine as a diskless client. Instructions for configuring the server are found in the section entitled **Getting the NetBSD System onto Useful Media** above.

First, at the Sun PROM monitor prompt, enter a boot command using the network interface as the boot device. On desktop machines this is *le*, and *ie* on the others. Examples:

```
>b le() -s
>b ie() -s
```

After the boot program loads the RAMDISK kernel, you should see the welcome screen as shown in the **Installing from tape** section above. You must configure the network interface before you can use any network resources. For example the command:

```
ssh> ifconfig le0 inet 192.233.20.198 up
```

will bring up the network interface with that address. The next step is to copy the miniroot from your server. This can be done using either NFS or remote shell. (In the examples that follow, the server has IP address 192.233.20.195.) You may then need to add a default route if the server is on a different subnet:

```
ssh> route add default 192.233.20.255 1
```

You can look at the route table using:

```
ssh> route show
```

Now mount the NFS file system containing the miniroot image:

```
ssh> mount -r 192.233.20.195:/server/path /mnt
```

The procedure is simpler if you have space for an expanded (not compressed) copy of the miniroot image. In that case:

```
ssh> dd if=/mnt/miniroot of=/dev/rsd0b bs=8k
```

Otherwise, you will need to use **zcat** to expand the miniroot image while copying. This is tricky because the **ssh** program (small shell) does not handle sh(1) pipeline syntax. Instead, you first run the reader in the background with its input set to /dev/pipe and then run the other program in the foreground with its output to /dev/pipe. The result looks like this:

```
ssh> run -bg dd if=/dev/pipe of=/dev/rsd0b obs=8k
ssh> run -o /dev/pipe zcat /mnt/install/miniroot.gz
```

To load the miniroot using rsh to the server, you would use a pair of commands similar to the above. Here is another example:

```
ssh> run -b dd if=/dev/pipe of=/dev/rsd0b obs=8k
ssh> run -o /dev/pipe rsh 192.233.20.195 zcat miniroot.gz
```

Installing from SunOS

To install NetBSD/sun3 onto a machine already running SunOS, you will need the miniroot image (miniroot.gz) and some means to decompress it.

First, boot SunOS and place the miniroot file onto the hard drive. If you do not have **gzip** for SunOS, you will need to decompress the image elsewhere before you can use it.

Next, bring SunOS down to single user mode to insure that nothing will be using the swap space on your drive. To be extra safe, reboot the machine into single-user mode rather than using the **shutdown** command.

Now copy the miniroot image onto your swap device (here `/dev/rsd0b`) with the command

```
gzip -dc miniroot.gz | dd of=/dev/rsd0b obs=32k
```

or if you have already decompressed the miniroot

```
dd if=miniroot.gz of=/dev/rsd0b obs=32k
```

Finally, reboot the machine and instruct the ROM to boot from the swap device as described in the next section.

Booting the Miniroot

If the miniroot was installed on partition 'b' of the disk with SCSI target ID=0 then the PROM boot command would be:

```
>b sd(0,0,1) -s
```

With SCSI target ID=2, the PROM is:

```
>b sd(0,10,1) -s
```

The numbers in parentheses above are:

1. controller (usually zero)
2. unit number (SCSI-ID * 8, in hexadecimal)
3. partition number

Miniroot install program

The miniroot's install program is very simple to use. It will guide you through the entire process, and is well automated. Additional improvements are planned for future releases.

The miniroot's install program will:

- Allow you to place disklabels on additional disks. The disk we are installing on should already have been partitioned using the RAMDISK kernel.
- Create file systems on target partitions.
- Allow you to set up your system's network configuration. Remember to specify host names without the domain name appended to the end. For example use `foo` instead of `foo.bar.org`. If, during the process of configuring the network interfaces, you make a mistake, you will be able to re-configure that interface by simply selecting it for configuration again.
- Mount target file systems. You will be given the opportunity to manually edit the resulting `/etc/fstab`.
- Extract binary sets from the media of your choice.
- Copy configuration information gathered during the installation process to your root file system (`/`).
- Make device nodes in your root file system under `/dev`.
- Copy a new kernel onto your root partition (`/`).
- Install a new boot block.
- Check your file systems for integrity.

First-time installation on a system through a method other than the installation program is possible, but strongly discouraged.

Post installation steps

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

1. Configuring /etc/rc.conf

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

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

   and with the root file system (/) mounted read-write. When the system asks you to choose a shell, simply press RETURN to get to a prompt. If you are asked for a terminal type, respond with **vt220** (or whatever is appropriate for your terminal type) and press RETURN. At this point, you need to configure at least one file in the /etc directory. Change to the /etc directory and take a look at the /etc/rc.conf file. Modify it to your tastes, making sure that you set rc_configured=YES so that your changes will be enabled and a multi-user boot can proceed. Default values for the various programs can be found in /etc/defaults/rc.conf, where some in-line documentation may be found. More complete documentation can be found in rc.conf(5).

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

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

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

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

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

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

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

   To enable proper hostname resolution, you will also want to add an /etc/resolv.conf file or (if you are feeling a little more adventurous) run named(8). See resolv.conf(5) or named(8) for more information.

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

2. Logging in

   After reboot, you can log in as root at the login prompt. There is no initial password, but if you're using the machine in a networked environment, you should create an account for yourself (see below) and protect it and the "root" account with good passwords. Unless you have connected an unusual terminal device as the console you can just press RETURN when it prompts for Terminal type? [...]

3. Adding accounts

   Use the useradd(8) command to add accounts to your system, *do not* edit /etc/passwd directly. See useradd(8) for more information on how to add a new user to the system.

4. The X Window System

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

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

5. Installing third party packages

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

   - More information on the package system is at
     http://www.netbsd.org/Documentation/software/packages.html
   - A browsable listing of available packages is at
     ftp://ftp.netbsd.org/pub/NetBSD/packages/pkgsrc/README.html
   - Precompiled binaries can be found at
     ftp://ftp.netbsd.org/pub/NetBSD/packages/
   - Package sources for compiling packages can be obtained by retrieving the file
     ftp://ftp.netbsd.org/pub/NetBSD/NetBSD-current/tar_files/pkgsrc.tar.gz
     They are typically extracted into `/usr/pkgsrc` (though other locations work fine), as with the command:

     ```
     # mkdir /usr/pkgsrc; tar -C /usr/pkgsrc -zxpf pkgsrc.tar.gz
     ```

     After extracting, then see the `README` file in the extraction directory (e.g. `/usr/pkgsrc/README`) for more information.

6. Misc

   - Edit `/etc/mail/aliases` to forward root mail to the right place (run `newaliases`(1) afterwards.)
   - The `/etc/mail/sendmail.cf` file will almost definitely need to be adjusted; files aiding in this can be found in `/usr/share/sendmail`. See the README file there for more information.
   - Edit `/etc/rc.local` to run any local daemons you use.
   - Many of the `/etc` files are documented in section 5 of the manual; so just invoking

     ```
     # man 5 filename
     ```

     is likely to give you more information on these files.

Upgrading a previously-installed NetBSD System

It is possible to easily upgrade your existing NetBSD/sun3 system using the upgrade program in the miniroot. If you wish to upgrade your system by this method, simply select the **upgrade** option once the miniroot has booted. The upgrade program with then guide you through the procedure. The upgrade program will:

- Enable the network based on your system's current network configuration.
- Mount your existing file systems.

- Extract binary sets from the media of your choice.
- Make new device nodes in your root file system under `/dev`.
- Copy a new kernel onto your root partition (`/`).

  Note: the existing kernel *will not* be backed up; doing so would be pointless, since older kernels may not be capable of running NetBSD 1.5 executables.
- Install a new boot block.
- Check your file systems for integrity.

Using the miniroot's upgrade program is the preferred method of upgrading your system.

However, it is possible to upgrade your system manually. To do this, follow the following procedure:

- Place *at least* the **base** binary set in a file system accessible to the target machine. A local file system is preferred, since the NFS subsystem in the NetBSD 1.5 kernel may be incompatible with your old binaries.
- Back up your pre-existing kernel and copy the 1.5 kernel into your root partition (`/`).
- Reboot with the 1.5 kernel into single-user mode. (Otherwise you can not install the boot block.)
- Check all file systems:

  ```
  # /sbin/fsck -pf
  ```
- Mount all local file systems:

  ```
  # /sbin/mount -a -t nonfs
  ```
- If you keep `/usr` or `/usr/share` on an NFS server, you will want to mount those file systems as well. To do this, you will need to enable the network:

  ```
  # sh /etc/rc.d/network start
  ```
- Make sure you are in the root file system (`/ and extract the`) **base** binary set:

  ```
  # cd /
  # pax -zrvpe -f Ar /path/to/base.tgz
  ```
- Install a new boot block: (assuming root is on `/dev/rsd0a`)

  ```
  # cd /usr/mdec
  # cp -p ./ufsboot /mnt/ufsboot
  # sync ; sleep 1 ; sync
  # ./installboot -v /ufsboot bootxx /dev/rsd0a
  ```
- Sync the file systems:

  ```
  sync
  ```
- At this point you may extract any other binary sets you may have placed on local file systems, or you may wish to extract additional sets at a later time. To extract these sets, use the following commands:

  ```
  # Ic cd /
  # Ic pax -zrvpe -f  path to set""
  ```

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

Compatibility Issues With Previous NetBSD Releases

Users upgrading from previous versions of NetBSD may wish to bear the following problems and compatibility issues in mind when upgrading to NetBSD 1.5.

General issues

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

  In previous releases of NetBSD, `/etc/rc` was a traditional BSD style monolithic file. As of NetBSD 1.5, each discrete program or substem from `/etc/rc` and `/etc/netstart` has been moved into separate scripts in `/etc/rc.d/`.

  At system startup, `/etc/rc` uses `rcorder`(8) to build a dependency list of the files in `/etc/rc.d` and then executes each script in turn with an argument of 'start'. Many `rc.d` scripts won't start unless the appropriate `rc.conf`(5) entry in `/etc/rc.conf` is set to 'YES.'

  At system shutdown, `/etc/rc.shutdown` uses `rcorder`(8) to build a dependency list of the files in `/etc/rc.d` that have a ''KEYWORD: shutdown'' line, reverses the resulting list, and then executes each script in turn with an argument of 'stop'. The following scripts support a specific shutdown method: `cron`, `inetd`, `local`, and `xdm`.

  Local and third-party scripts may be installed into `/etc/rc.d` as necessary. Refer to the other scripts in that directory and `rc`(8) for more information on implementing `rc.d` scripts.

Issues affecting an upgrading from NetBSD 1.4 or later

- `named`(8) leaks version information

  Previous releases of NetBSD disabled a feature of `named`(8) where the version number of the server could be determined by remote clients. This feature has not been disabled in NetBSD 1.5, because there is a `named.conf`(5) option to change the version string:

```
option {
        version "newstring";
};
```

- `sysctl`(8) pathname changed

  `sysctl`(8) is moved from `/usr/sbin/sysctl` to `/sbin/sysctl`. If you have hardcoded references to the full pathname (in shell scripts, for example) please be sure to update those.

- `sendmail`(8) configuration file pathname changed

  Due to `sendmail`(8) upgrade from 8.9.x to 8.10.x, `/etc/sendmail.cf` is moved to `/etc/mail/sendmail.cf`. Also, the default `sendmail.cf`(5) refers different pathnames than before. For example, `/etc/aliases` is now located at `/etc/mail/aliases`, `/etc/sendmail.cw` is now called `/etc/mail/local-host-names`, and so forth. If you have customized `sendmail.cf`(5) and friends, you will need to move the files to the new locations. See `/usr/share/sendmail/README` for more information.

Using online NetBSD documentation

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

- `intro`(1),

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

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

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

```
# man passwd
```

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

```
# man 5 passwd
```

instead.

If you are unsure of what man page you are looking for, enter **apropos** *subject-word*

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

Administrivia

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

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

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

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

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

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

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

Thanks go to

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

  Keith Bostic
  Ralph Campbell
  Mike Karels
  Marshall Kirk McKusick

for their ongoing work on BSD systems, support, and encouragement.

- Also, our thanks go to:

  Mike Hibler
  Rick Macklem
  Jan-Simon Pendry
  Chris Torek

  for answering lots of questions, fixing bugs, and doing the various work they've done.

- UC Berkeley's Experimental Computing Facility provided a home for sun-lamp in the past, people to look after it, and a sense of humor. Rob Robertson, too, has added his unique sense of humor to things, and for a long time provided the primary FTP site for NetBSD.
- Vixie Enterprises for hosting the NetBSD FTP, SUP, and WWW servers.
- Redback Networks, Inc. for hosting the NetBSD mail and GNATS server.
- The Helsinki University of Technology in Finland for hosting the NetBSD CVS server.
- The Internet Research Institute in Japan for hosting the server which runs the CVSweb interface to the NetBSD source tree.
- The many organisations that provide NetBSD mirror sites.
- Without CVS, this project would be impossible to manage, so our hats go off to Brian Berliner, Jeff Polk, and the various other people who've had a hand in making CVS a useful tool.
- Dave Burgess `burgess@cynjut.infonet.net` has been maintaining the 386BSD/NetBSD/FreeBSD FAQ for quite some time, and deserves to be recognized for it.
- The following individuals and organizations (each in alphabetical order) have made donations or loans of hardware and/or money, to support NetBSD development, and deserve credit for it:

  Steve Allen
  Jason Birnschein
  Mason Loring Bliss
  Jason Brazile
  Mark Brinicombe
  David Brownlee
  Simon Burge
  Dave Burgess
  Ralph Campbell
  Brian Carlstrom
  James Chacon
  Bill Coldwell
  Charles Conn
  Tom Coulter
  Charles D. Cranor
  Christopher G. Demetriou
  Scott Ellis
  Hubert Feyrer
  Castor Fu
  Greg Gingerich
  William Gnadt
  Michael Graff
  Guenther Grau

Ross Harvey
Charles M. Hannum
Michael L. Hitch
Kenneth Alan Hornstein
Jordan K. Hubbard
Søren Jørvang
Scott Kaplan
Noah M. Keiserman
Harald Koerfgen
John Kohl
Chris Legrow
Ted Lemon
Norman R. McBride
Neil J. McRae
Perry E. Metzger
Toru Nishimura
Herb Peyerl
Mike Price
Dave Rand
Michael Richardson
Heiko W. Rupp
Brad Salai
Chuck Silvers
Thor Lancelot Simon
Bill Sommerfeld
Paul Southworth
Eric and Rosemary Spahr
Ted Spradley
Kimmo Suominen
Jason R. Thorpe
Steve Wadlow
Krister Walfridsson
Jim Wise
Christos Zoulas

AboveNet Communications, Inc.
Advanced System Products, Inc.
Avalon Computer Systems
Bay Area Internet Solutions
Brains Corporation, Japan
Canada Connect Corporation
Co-operative Research Centre for Enterprise Distributed Systems Technology
Demon Internet, UK
Digital Equipment Corporation
Distributed Processing Technology
Easynet, UK
Free Hardware Foundation
Innovation Development Enterprises of America
Internet Software Consortium
MS Macro System GmbH, Germany
Numerical Aerospace Simulation Facility, NASA Ames Research Center
Piermont Information Systems Inc.

Salient Systems Inc.
VMC Harald Frank, Germany
Warped Communications, Inc.
Whitecross Database Systems Ltd.

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

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

We are...

(in alphabetical order)

The NetBSD core group:

Alistair Crooks	agc@netbsd.org
Jun-ichiro itojun Hagino	itojun@netbsd.org
Frank van der Linden	fvdl@netbsd.org
Luke Mewburn	lukem@netbsd.org
Christos Zoulas	christos@netbsd.org

The portmasters (and their ports):

Mark Brinicombe	mark@netbsd.org	**arm32**
Jeremy Cooper	jeremy@netbsd.org	**sun3x**
Ross Harvey	ross@netbsd.org	**alpha**
Jun-ichiro itojun Hagino	itojun@netbsd.org	**sh3**
Ben Harris	bjh21@netbsd.org	**arm26**
Eduardo Horvath	eeh@netbsd.org	**sparc64**
Darrin Jewell	dbj@netbsd.org	**next68k**
Søren Jørvang	soren@netbsd.org	**cobalt**
Søren Jørvang	soren@netbsd.org	**sgimips**
Wayne Knowles	wdk@netbsd.org	**mipsco**
Paul Kranenburg	pk@netbsd.org	**sparc**
Anders Magnusson	ragge@netbsd.org	**vax**
Minoura Makoto	minoura@netbsd.org	**x68k**
Phil Nelson	phil@netbsd.org	**pc532**
Tohru Nishimura	nisimura@netbsd.org	**luna68k**
NONAKA Kimihiro	nonaka@netbsd.org	**prep**
Scott Reynolds	scottr@netbsd.org	**mac68k**
Kazuki Sakamoto	sakamoto@netbsd.org	**bebox**
Noriyuki Soda	soda@netbsd.org	**arc**
Wolfgang Solfrank	ws@netbsd.org	**ofppc**
Ignatios Souvatzis	is@netbsd.org	**amiga**
Jonathan Stone	jonathan@netbsd.org	**pmax**
Shin Takemura	takemura@netbsd.org	**hpcmips**
Jason Thorpe	thorpej@netbsd.org	**alpha**
Jason Thorpe	thorpej@netbsd.org	**hp300**
Tsubai Masanari	tsubai@netbsd.org	**macppc**

Tsubai Masanari	tsubai@netbsd.org	**newsmips**
Izumi Tsutsui	tsutsui@netbsd.org	**news68k**
Frank van der Linden	fvdl@netbsd.org	**i386**
Leo Weppelman	leo@netbsd.org	**atari**
Nathan Williams	nathanw@netbsd.org	**sun3**
Steve Woodford	scw@netbsd.org	**mvme68k**

The NetBSD 1.5 Release Engineering team:

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

Developers and other contributors:

Steve Allen	wormey@netbsd.org
Julian Assange	proff@netbsd.org
Lennart Augustsson	augustss@netbsd.org
Christoph Badura	bad@netbsd.org
Robert V. Baron	rvb@netbsd.org
Erik Berls	cyber@netbsd.org
John Birrell	jb@netbsd.org
Mason Loring Bliss	mason@netbsd.org
Manuel Bouyer	bouyer@netbsd.org
John Brezak	brezak@netbsd.org
Allen Briggs	briggs@netbsd.org
Aaron Brown	abrown@netbsd.org
David Brownlee	abs@netbsd.org
Frederick Bruckman	fredb@netbsd.org
Jon Buller	jonb@netbsd.org
Simon Burge	simonb@netbsd.org
Dave Burgess	burgess@cynjut.infonet.net
Robert Byrnes	byrnes@netbsd.org
D'Arcy J.M. Cain	darcy@netbsd.org
Dave Carrel	carrel@netbsd.org
James Chacon	jmc@netbsd.org
Bill Coldwell	billc@netbsd.org
Julian Coleman	jdc@netbsd.org
Chuck Cranor	chuck@netbsd.org
Aidan Cully	aidan@netbsd.org
Johan Danielsson	joda@netbsd.org
Matt DeBergalis	deberg@netbsd.org
Rob Deker	deker@netbsd.org
Chris G. Demetriou	cgd@netbsd.org

Jaromir Dolecek	jdolecek@netbsd.org
Andy Doran	ad@netbsd.org
Roland Dowdeswell	elric@netbsd.org
Matthias Drochner	drochner@netbsd.org
Jun Ebihara	jun@netbsd.org
Havard Eidnes	he@netbsd.org
Enami Tsugutomo	enami@netbsd.org
Bernd Ernesti	veego@netbsd.org
Erik Fair	fair@netbsd.org
Hubert Feyrer	hubertf@netbsd.org
Thorsten Frueauf	frueauf@netbsd.org
Castor Fu	castor@netbsd.org
Ichiro Fukuhara	ichiro@netbsd.org
Brian R. Gaeke	brg@dgate.org
Thomas Gerner	thomas@netbsd.org
Simon J. Gerraty	sjg@netbsd.org
Justin Gibbs	gibbs@netbsd.org
Adam Glass	glass@netbsd.org
Michael Graff	explorer@netbsd.org
Brad Grantham	grantham@tenon.com
Brian C. Grayson	bgrayson@netbsd.org
Matthew Green	mrg@netbsd.org
Juergen Hannken-Illjes	hannken@netbsd.org
Charles M. Hannum	mycroft@netbsd.org
Eric Haszlakiewicz	erh@netbsd.org
John Hawkinson	jhawk@netbsd.org
HAYAKAWA Koichi	haya@netbsd.org
René Hexel	rh@netbsd.org
Michael L. Hitch	mhitch@netbsd.org
Christian E. Hopps	chopps@netbsd.org
Ken Hornstein	kenh@netbsd.org
Marc Horowitz	marc@netbsd.org
Nick Hudson	skrll@netbsd.org
Martin Husemann	martin@netbsd.org
Dean Huxley	dean@netbsd.org
Bernardo Innocenti	bernie@netbsd.org
ITOH Yasufumi	itohy@netbsd.org
IWAMOTO Toshihiro	toshii@netbsd.org
Matthew Jacob	mjacob@netbsd.org
Lonhyn T. Jasinskyj	lonhyn@netbsd.org
Chris Jones	cjones@netbsd.org
Takahiro Kambe	taca@netbsd.org
Antti Kantee	pooka@netbsd.org
Lawrence Kesteloot	kesteloo@cs.unc.edu
Thomas Klausner	wiz@netbsd.org
Klaus Klein	kleink@netbsd.org
Wayne Knowles	wdk@netbsd.org
John Kohl	jtk@netbsd.org
Kevin Lahey	kml@netbsd.org

Johnny C. Lam	jlam@netbsd.org
Martin J. Laubach	mjl@netbsd.org
Ted Lemon	mellon@netbsd.org
Joel Lindholm	joel@netbsd.org
Mike Long	mikel@netbsd.org
Warner Losh	imp@netbsd.org
Federico Lupi	federico@netbsd.org
Brett Lymn	blymn@netbsd.org
Paul Mackerras	paulus@netbsd.org
David Maxwell	david@netbsd.org
Dan McMahill	dmcmahill@netbsd.org
Gregory McGarry	gmcgarry@netbsd.org
Neil J. McRae	neil@netbsd.org
Perry Metzger	perry@netbsd.org
der Mouse	mouse@netbsd.org
Joseph Myers	jsm@netbsd.org
Ken Nakata	kenn@netbsd.org
Bob Nestor	rnestor@netbsd.org
NONAKA Kimihiro	nonaka@netbsd.org
Masaru Oki	oki@netbsd.org
Atsushi Onoe	onoe@netbsd.org
Greg Oster	oster@netbsd.org
Herb Peyerl	hpeyerl@netbsd.org
Matthias Pfaller	matthias@netbsd.org
Dante Profeta	dante@netbsd.org
Chris Provenzano	proven@netbsd.org
Waldi Ravens	waldi@moacs.indiv.nl.net
Darren Reed	darrenr@netbsd.org
Michael Richardson	mcr@netbsd.org
Tim Rightnour	garbled@netbsd.org
Gordon Ross	gwr@netbsd.org
Heiko W. Rupp	hwr@netbsd.org
SAITOH Masanobu	msaitoh@netbsd.org
Curt Sampson	cjs@netbsd.org
Wilfredo Sanchez	wsanchez@netbsd.org
Ty Sarna	tsarna@netbsd.org
SATO Kazumi	sato@netbsd.org
Matthias Scheler	tron@netbsd.org
Karl Schilke (rAT)	rat@netbsd.org
Konrad Schroder	perseant@netbsd.org
Reed Shadgett	dent@netbsd.org
Tim Shepard	shep@netbsd.org
Takao Shinohara	shin@netbsd.org
Takuya SHIOZAKI	tshiozak@netbsd.org
Chuck Silvers	chs@netbsd.org
Thor Lancelot Simon	tls@netbsd.org
Jeff Smith	jeffs@netbsd.org
Bill Sommerfeld	sommerfeld@netbsd.org
Bill Squier	groo@netbsd.org

Bill Studenmund	wrstuden@netbsd.org
Kevin Sullivan	sullivan@netbsd.org
SUNAGAWA Keiki	kei@netbsd.org
Kimmo Suominen	kim@netbsd.org
Matt Thomas	matt@netbsd.org
Christoph Toshok	toshok@netbsd.org
UCHIYAMA Yasushi	uch@netbsd.org
Shuichiro URATA	ur@netbsd.org
Todd Vierling	tv@netbsd.org
Aymeric Vincent	aymeric@netbsd.org
Paul Vixie	vixie@netbsd.org
Krister Walfridsson	kristerw@netbsd.org
Lex Wennmacher	wennmach@netbsd.org
Assar Westerlund	assar@netbsd.org
Todd Whitesel	toddpw@netbsd.org
Rob Windsor	windsor@netbsd.org
Dan Winship	danw@netbsd.org
Jim Wise	jwise@netbsd.org
Michael Wolfson	mbw@netbsd.org
Colin Wood	ender@netbsd.org

Legal Mumbo-Jumbo

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

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

This product includes software developed by the University of California, Berkeley and its contributors.

This product includes software developed by The NetBSD Foundation, Inc.

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

This product includes software developed by the Computer Systems Engineering Group at Lawrence Berkeley Laboratory.

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

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

This product includes software developed by Adam Glass.

This product includes software developed by Alistair G. Crooks.

This product includes software developed by Amancio Hasty and Roger Hardiman.

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

This product includes software developed by Bill Paul.

This product includes software developed by Charles D. Cranor and Washington University.

This product includes software developed by Charles D. Cranor.

This product includes software developed by Charles Hannum, by the University of Vermont and State Agricultural College and Garrett A. Wollman, by William F. Jolitz, and by the University of California, Berkeley, Lawrence Berkeley Laboratory, and its contributors.

This product includes software developed by Charles Hannum.

This product includes software developed by Charles M. Hannum.

This product includes software developed by Chris Provenzano.

This product includes software developed by Christian E. Hopps.

This product includes software developed by Christopher G. Demetriou for the NetBSD Project.

This product includes software developed by Christopher G. Demetriou.

This product includes software developed by Christos Zoulas.

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

This product includes software developed by Dean Huxley.

This product includes software developed by Eric S. Hvozda.

This product includes software developed by Ezra Story.

This product includes software developed by Gardner Buchanan.

This product includes software developed by Gordon Ross.

This product includes software developed by Gordon W. Ross and Leo Weppelman.

This product includes software developed by Gordon W. Ross.

This product includes software developed by Hauke Fath.

This product includes software developed by HAYAKAWA Koichi.

This product includes software developed by Hellmuth Michaelis and Joerg Wunsch.

This product includes software developed by Herb Peyerl.

This product includes software developed by Holger Veit and Brian Moore for use with "386BSD" and similar operating systems.

This product includes software developed by Hubert Feyrer for the NetBSD Project.

This product includes software developed by Iain Hibbert.

This product includes software developed by Ian W. Dall.

This product includes software developed by Ignatios Souvatzis for the NetBSD Project.

This product includes software developed by Jason R. Thorpe for And Communications, http://www.and.com/.

This product includes software developed by Joachim Koenig-Baltes.

This product includes software developed by Jochen Pohl for The NetBSD Project.

This product includes software developed by John Polstra.

This product includes software developed by Jonathan Stone and Jason R. Thorpe for the NetBSD Project.

This product includes software developed by Jonathan Stone for the NetBSD Project.

This product includes software developed by Jonathan Stone.

This product includes software developed by Julian Highfield.

This product includes software developed by Kenneth Stailey.

This product includes software developed by Leo Weppelman.

This product includes software developed by Lloyd Parkes.

This product includes software developed by Manuel Bouyer.

This product includes software developed by Marc Horowitz.

This product includes software developed by Mark Brinicombe.

This product includes software developed by Mark Tinguely and Jim Lowe.

This product includes software developed by Markus Wild.

This product includes software developed by Martin Husemann and Wolfgang Solfrank.

This product includes software developed by Mats O Jansson and Charles D. Cranor.

This product includes software developed by Mats O Jansson.

This product includes software developed by Matthias Pfaller.

This product includes software developed by Michael L. Hitch.

This product includes software developed by Niels Provos.

This product includes software developed by Paul Kranenburg.

This product includes software developed by Paul Mackerras.

This product includes software developed by Peter Galbavy.

This product includes software developed by Philip A. Nelson.

This product includes software developed by Rodney W. Grimes.

This product includes software developed by Roland C. Dowdeswell.

This product includes software developed by Rolf Grossmann.

This product includes software developed by Scott Bartram.

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

This product includes software developed by Tatoku Ogaito for the NetBSD Project.

This product includes software developed by Terrence R. Lambert.

This product includes software developed by Theo de Raadt and John Brezak.

This product includes software developed by Theo de Raadt.

This product includes software developed by Tohru Nishimura for the NetBSD Project.

This product includes software developed by TooLs GmbH.

This product includes software developed by Winning Strategies, Inc.

This product includes software developed by Zembu Labs, Inc.

This product includes software developed by the Center for Software Science at the University of Utah.

This product includes software developed by the Computer Systems Laboratory at the University of Utah.

This product includes software developed by the University of Calgary Department of Computer Science and its contributors.

This product includes software developed by the University of Vermont and State Agricultural College and Garrett A. Wollman.

This product includes software developed for the FreeBSD project.

This product includes software developed for the Internet Software Consortium by Ted Lemon.

This product includes software developed for the NetBSD Project by Frank van der Linden.

This product includes software developed for the NetBSD Project by Jason R. Thorpe.

This product includes software developed for the NetBSD Project by John M. Vinopal.

This product includes software developed for the NetBSD Project by Matthias Drochner.

This product includes software developed for the NetBSD Project by Matthieu Herrb.

This product includes software developed for the NetBSD Project by Perry E. Metzger.

This product includes software developed for the NetBSD Project by Piermont Information Systems Inc.

This product includes software developed for the NetBSD Project by Ted Lemon.

This product includes software developed by LAN Media Corporation and its contributors.

This product includes software developed by Michael Graff for the NetBSD Project.

This product includes software developed by Niklas Hallqvist, C Stone and Job de Haas.

This product includes software developed by Eric Young (eay@mincom.oz.au).

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/).

This product includes software developed by the University of Oregon.

This product includes software developed by the University of Southern California and/or Information Sciences Institute.

This product includes software developed by Internet Initiative Japan Inc.