NAME

INSTALL — Installation procedure for NetBSD/mac68k.

CONTENTS

DESCRIPTION

About this Document

This document describes the installation procedure for NetBSD 5.1 on the *mac68k* platform. It is available in four different formats titled `INSTALL.`*`ext`*, where *`.ext`* is one of `.ps`, `.html`, `.more`, or `.txt`:

`.ps` PostScript.

`.html` Standard Internet HTML.

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

`.txt` Plain old ASCII.

You are reading the *PostScript* version.

What is NetBSD?

The NetBSD Operating System is a fully functional Open Source UNIX-like operating system derived from the University of California, Berkeley Networking Release 2 (Net/2), 4.4BSD-Lite, and 4.4BSD-Lite2 sources. NetBSD runs on 57 different system architectures (ports) across 15 distinct CPU families, and is being ported to more. The NetBSD 5.1 release contains complete binary releases for many different system architectures. (A few ports are not fully supported at this time and are thus not part of the binary distribution. Please see the NetBSD web site at **http://www.NetBSD.org/** for information on them.)

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

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

Dedication

NetBSD 5.1 is dedicated to the memory of Martti Kuparinen, who was the victim of a traffic accident in June 2010.

Martti's technical contributions are too many to list here in full. He created and maintained numerous packages in pkgsrc, updated two packet filter solutions distributed with NetBSD and improved several hardware drivers. Beyond that he was always helpful and friendly. His example encouraged users to contribute to the project and share their work with the community. Some of these users later became NetBSD developers themselves thanks to Martti's efforts.

Changes Between the NetBSD 5.0 and 5.1 Releases

The NetBSD 5.1 release is the first feature update of the NetBSD 5.0 release branch. It represents a selected subset of fixes deemed critical for security or stability reasons, as well as new features and enhancements.

Please note that all fixes in security/critical updates (i.e., NetBSD 5.0.1, 5.0.2, etc.) are cumulative, so the latest update contains all such fixes since the corresponding minor release. These fixes also appear in minor releases (i.e., NetBSD 5.1, 5.2, etc.).

The complete list of changes can be found in the CHANGES-5.1:

http://ftp.NetBSD.org/pub/NetBSD/NetBSD-5.1/CHANGES-5.1

file in the top level directory of the NetBSD 5.1 release tree. An abbreviated list is as follows:

Security Advisory Fixes

- NetBSD-SA2009-004 (NetBSD OpenPAM `passwd`(1) changing weakness):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2009-004.txt.asc
- NetBSD-SA2009-005 (Plaintext Recovery Attack Against SSH):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2009-005.txt.asc
- NetBSD-SA2009-006 (Buffer overflows in ntp):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2009-006.txt.asc
- NetBSD-SA2009-007 (Buffer overflows in `hack`(6)):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2009-007.txt.asc
- NetBSD-SA2009-008 (OpenSSL ASN1 parsing denial of service and CMS signature verification weakness):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2009-008.txt.asc
- NetBSD-SA2009-009 (OpenSSL DTLS Memory Exhaustion and DSA signature verification vulnerabilities):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2009-009.txt.asc
- NetBSD-SA2009-010 (ISC dhclient subnet-mask flag stack overflow):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2009-010.txt.asc
- NetBSD-SA2009-011 (ISC DHCP server Denial of Service vulnerability):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2009-011.txt.asc
- NetBSD-SA2009-012 (SHA2 implementation potential buffer overflow):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2009-012.txt.asc
- NetBSD-SA2009-013 (BIND named dynamic update Denial of Service vulnerability):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2009-013.txt.asc
- NetBSD-SA2010-002 (OpenSSL TLS renegotiation man in the middle vulnerability):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2010-002.txt.asc
- NetBSD-SA2010-003 (azalia(4)/hdaudio(4) negative mixer index panic):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2010-003.txt.asc

- NetBSD-SA2010-004 (amd64 per-page No-execute (NX) bit disabled):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2010-004.txt.asc
- NetBSD-SA2010-005 (NTP server Denial of Service vulnerability):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2010-005.txt.asc
- NetBSD-SA2010-006 (Buffer length checking errors in CODA):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2010-006.txt.asc
- NetBSD-SA2010-007 (Integer overflow in libbz2 decompression code):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2010-007.txt.asc
- NetBSD-SA2010-008 (sftp(1)/ftp(1)/glob(3) related resource exhaustion):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2010-008.txt.asc
- NetBSD-SA2010-010 (Buffer Length Handling Errors in netsmb):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2010-010.txt.asc
- NetBSD-SA2010-011 (OpenSSL Double Free Arbitrary Code Execution):
  http://ftp.NetBSD.org/pub/NetBSD/security/advisories/NetBSD-SA2010-011.txt.asc

Advisories prior to NetBSD-SA2009-004 do not affect NetBSD 5.0: *.:*
http://www.NetBSD.org/support/security/patches-5.0.html

Other Security Fixes

- openssl: Fix CVE-2009-4355 and CVE-2010-0740.
- Fix crash in openssl: handshake_dgst[] may be used without being allocated, causing NULL pointer dereference.
- Update BIND server and tools to 9.5.2-P2, fixing CVE-2009-0025, CVE-2009-4022, and CVE-2010-0097.
- `ntpd`(8): Fix CVE-2009-3563.
- expat: Fix SA36425 and CVE-2009-3560.
- `fts`(3): Avoid possible integer overflow on really deep dirs, and subsequent collateral damage. Received from OpenBSD via US-CERT as VU #590371.
- Fix a couple issues with POSIX message queues:
  - An invalid signal number passed to mq_notify() could crash the kernel on delivery -- add a boundary check.
  - A user could set mq_maxmsg (the maximal number of messages in a queue) to a huge value on mq_open(O_CREAT) and later use up all kernel memory by mq_send() -- add a sysctl'able limit which defaults to 16∗mq_def_maxmsg.
- `arc4random`(3): Keep arc4_i and arc4_j synchronised after a rekeying. This prevents accidentally ending up in a short ARC4 cycle.
- freetype: Fix CVE-2009-0946.
- `ftpd`(8): Fix a remote crash. PR 43023.
- openldap: Fix CVE-2009-3767.
- Fix an NX regression observed on amd64 kernels, where per-page execution right was disabled (therefore leading to the inability of the kernel to detect fraudulent use of memory mappings marked as not being executable).

Kernel

- Fix random "filesystem full" messages on large FFS file systems.
- Fix a regression in the 4.4BSD scheduler, improving interactive performance under load.
- Remove a race where physio_done() may use memory already freed. PR 39536.
- Fix a crash observed when trying to load a corrupted ELF kernel module.
- Fix PR 41566, where writes on the controlling tty were not being awoken from blocks.
- Various fixes for POSIX message queues.
- Fix a couple deadlocks in the VFS subsystem. PR 41374.
- Fix an issue on some architectures where a softint could fire on the wrong CPU.
- Fixes for POSIX advisory locks.
- A number of other stability fixes.
- Fix a case where `setpriority`(2) returned EACCES instead of EPERM. PR 41489.
- `lockf`(3) passes its arguments through to `fcntl`(2) but is supposed to support -ve lengths (lock area before current offset). Nothing in libc or the kernel allowed for this, so some random part of the file would get locked. PR 41620.
- Fix ktrace of data from iovec based system calls. PR 41819.
- Fix stack size enforcement.

File Systems

- `fsck_ext2fs`(8): Ignore the "-P" option as intended, to make this work with, e.g., "fsck_flags=-pP" in `/etc/rc.conf`. PR 41490.
- UFS quotas: Add missing mutex_destroy() before pool_cache_put(). Prevents a "Mutex error: lockdebug_alloc: already initialized" panic.
- `tunefs`(8): Allow tunefs to clear any type of WAPBL log, not only in-filesystem ones.
- `fsck_ffs`(8): Do some basic checks of the WAPBL journal, and fail if the kernel would refuse to mount the filesystem read/write. Add code to clear the bogus journal when not run in preen (-p) mode.
- `wapbl`(4): If the WAPBL journal can't be read (ffs_wapbl_replay_start() fails), mount the filesystem anyway if MNT_FORCE is present. This allows to still boot single-user a system with a corrupted WAPBL on /, and so get a chance to run fsck to fix it.
- smbfs:
  - Fix some panics while trying to umount a smbfs share.
  - Fix detection of SMB capabilities according to the CIFS spec:
    - SMB_CAP_LARGE_FILES advertises support for 64-bit file offsets.
    - SMB_CAP_LARGE_READX and SMB_CAP_LARGE_WRITEX advertise support for large reads and writes (larger than 64KB). PR 42175.
  - Add support for file sizes greater than 4GB.
  - Prevent malicious local program from causing a kernel crash.

- A number of fixes for ext2fs. PR 28712.
- procfs: Fix memory leak. PR 42053.
- Fix `cp`(1) from NTFS. PR 38531.
- cd9660: Change cd9660_mount, in MNT_UPDATE case, to check dev_t's for equality instead of just vnode pointers. Fixes erroneous "Invalid argument" errors from `mount`(8) with -u against cd9660 root in the presence of mfs or tmpfs `/dev` prepared after initial mountroot.
- A number of fixes to ffs snapshots.
- sysctlfs: Fix a crash while trying to read nodes on amd64. PR 41494.
- Various improvements to UDF.
- xattr: Fix system crash which could be triggered by a malicious command.
- coda: Fix incomplete ioctl parameter verification.
- ffs: Fix a race condition which could lead to data corruption.
- `wapbl`(4): Fix replay problems which could corrupt the fs. PR 43336.

Networking

- `ath`(4): Remove the binary HAL and update to the open source HAL.
- Add the `age`(4) driver for Attansic L1.
- Add the `ale`(4) driver for Atheros AR8121/AR8113/AR8114.
- `brgphy`(4): Add support for BCM5462, BCM54K2 and BCM5722.
- `arp`(4): Don't require the gateway address to have room for both an interface name and address. Fixes a regression in 'arp -s ...' on interfaces such as xennet0 with unusually long names. PR 41878.
- Make tcp msl (max segment life) tunable via sysctl net.inet.tcp.msl.
- Fixed a number of `bge`(4) bugs.
- `ifpci`(4): Fix endianness issues when accessing the B-channel fifos.
- `wm`(4):
  - Numerous fixes for various chips.
  - Add SIOCSIFADDR support for setting the AF_LINK address, necessary for agr to be able to set the mac addresses of each port to the agr address (i.e., so it can receive all intended traffic at the hardware level).
  - Enable hardware VLAN support.
- `re`(4):
  - Detect RTL8169CP, RTL8168D/8111D, and RTL8103E variants.
  - Add hardware checksum support for newer 8168/8111/8102 chips.
  - Fix RX hardware checksum for DESCV2 chips. PR 40605.
- `le`(4): Don't immediately switch UTP/AUI ports on lost carrier.
- `agr`(4): Add vlan support and hardware offload capabilities. Add support for disabling the LACP protocol by setting LINK1 on the agr interface.

- `bnx`(4):
  - Protect against spurious "bnx0: Double mbuf allocation failure!" panics.
  - Add support for BCM5709 and BCM5716.
- Follow exactly the recommendation of draft-ietf-tcpm-tcpsecure-11.txt: Don't check against the last ack received, but the expected sequence number. This makes RST handling independent of delayed ACK.
- `vr`(4): Add suspend/resume support.
- `carp`(4): Improve logging. PR 38260.
- Give 100BASE-TX full duplex higher priority than 100BASE-T4.
- Fix wpa with `ral`(4).
- `vlan`(4): Inherit the parent's TCP segmentation offload capability.
- dhcpcd:
  - Update to 5.1.3.
  - Add an rc.d script.
  - Only start dhcpcd per interface if not running the full dhcpcd daemon. Only stop dhcpcd per interface if it's running for the interface. PR 40320.
- Fix a panic when trying to disable IPFilter before enabling it. PR 41364.
- `ping`(8): Deal with source route and record route specially giving a meaningful error message when remote side doesn't support record route. PR 41111.
- `ifconfig`(8): Don't require a "vlan" argument with "-vlanif". "ifconfig vlan0 -vlanif" now works as one would expect.
- `tap`(4): Fix a potential leak on device close. `bpf`(4): Prevent malicious bytecode from crashing the kernel with a divide-by-zero trap.
- UDPv6: Prevent local crash by malicious user program.

Miscellaneous Drivers

- `ehci`(4): Add a workaround for ATI SB600 and SB700 revisions A12 and A13 to avoid a USB subsystem hang when the system has multiple USB devices connected to it or one device is re-connected often.
- `uftdi`(4): Add support for multiple channel cards, specifically quad channel FT4232H.
- `ums`(4): Add quirks to make MS Wireless Laser Mouse 6000 work. PR 41737. Add support for USB HID devices that report absolute coordinates instead of relative movement data, e.g. touchpanels. Add support for the Microsoft Natural Ergonomic Desktop 7000 mouse.
- `apm`(4): Fix suspend/resume.

Audio

- Add `hdaudio`(4), a standards-compliant driver for Intel High Definition Audio. It will replace `azalia`(4) eventually.
- Add `gcscaudio`(4), an AMD Geode CS5536 audio driver.
- Add support for playback- or capture-only devices. PR 42050.

- pad(4): Catch up to audio(4) device_t/softc split to prevent pad from corrupting its child device's softc.
- audio(4): OSS audio allows mixer operations on the dsp device. NetBSD would previously return EINVAL in these circumstances. This can break audio in apps running under Linux emulation. Select whether to call mixer_ioctl() or audio_ioctl() based on whether the command smells like a mixer ioctl or not.

Security

- pam(3): Restore the good old UNIX behavior of root password changing: only root may change the root password.
- racoon(8): Fix a bug where racoon used a wrong IPsec-SA handle that was for another peer if it received an ISAKMP message for IPsec-SA that has the same message-id as the message-id that was received before.
- ipsec(4): Add a missing splx() call. PR 41701.
- opencrypto(9):
  - Extend the API to allow the destination buffer size to be specified when it is not the same size as the input buffer.
  - Add user-space access to compression features.
  - Add software gzip support (CRYPTO_GZIP_COMP).
  - Add the fast version of crc32 from zlib to libkern.
  - Fix PRs 41069 and 41070.

Storage

- Add support for RAIDframe parity maps. Drastically reduces the amount of time spent checking parity after an unclean shutdown by keeping better track of which regions might have had outstanding writes. Enabled by default; can be disabled on a per-set basis, or tuned, with the new raidctl(8) commands.
- Add sdmmc framework.
- Add sdhc(4), a driver for SD controllers following the SD Host Controller Standard Simplified Specification.
- Add wb(4), a driver for Winbond W83L518D SD/MMC readers.
- siisata(4): Sync with HEAD, fixing a number of bugs. PR 41579.
- mfi(4): Fix command list corruption seen on heavy I/O load. Add support for MFI gen2 devices.
- twa(4): Disable completely bogus DIAGNOSTIC check.
- wdc(4): use 8bits access to legacy IDE registers through the SATA interface (except data registers). Stops errors such as the following when probing SATA drives through controllers that offer the legacy pciide interface: viaide1 channel 0: reset failed for drive 0
- piixide(4): Add Intel 3400 support.

Platform specific

- x86 (amd64 and i386)
  - Add a workaround for a bug with some Opteron revisions where locked operations sometimes do not serve as memory barriers, allowing memory references to bleed outside of critical sections.

- ichlpcib(4): Fix watchdog code:
  - The timer bound constants are in tick, so convert period to tick before checking it against the bounds.
  - For ICH5 or older, fix code that would have always written a 0 period to the register.
- Add CPU topology detection support for AMD processors.
- asus(4):
  - Add experimental cpu fan/voltage switching support (sysctl hw.asus0.*).
  - Add fan sensor.
- Add hw.wake.* sysctl subtree for toggling which devices are allowed to restore the system from sleep. By default, the following devices are enabled for wake: sleep/power buttons, lid switch, pc kbd controller.
- agp(4): Add support for Intel G35, G45, and Q45.
- Cut down on the number of lines used in acpi autoconf messages.
- Extend CPU probe of Intel processors to handle extended-models: new Intel 45nm processors, Core i7, Atom, and the 45nm Xeon MP. Properly decode several new Intel cache descriptors, as listed in the most recent (March 2009) edition of Intel's Application Note 485. PRs 41289 and 41290.
- cpuctl(8): Add newer VIA C7 core and VIA Nano. When printing an unknown VIA CPU, default to 'Unknown IDT/VIA' instead of 'C3'.
- Fix NetBSD under qemu with ACPI enabled. PR 38729.
- acpi(4): If the firmware describes duplicate keyboard controller nodes, don't panic when the driver fails to map registers. PR 39671.
- Fix a bug where mapping the very end of iomem accidentally returns an address in the ISA hole. Fixes ohci on VirtualPC 7 for Mac, which places OHCI at base address 0xfffff000 size 0x1000.
- int 15h/AH=86h (WAIT) doesn't work properly on all hardware and emulators, so for the countdown use the more coarsely grained sleep implementation based on int 1ah/AH=00h (GET SYSTEM TIME).
- Use the TSC and current multiplier to calculate bus clock on VIA C7 Esther.
- Add support for VIA C7 temperature sensors (options VIA_C7TEMP) and enable in i386 GENERIC kernel.
- Fix a regression in the boot loader where pressing a letter not bound to a menu item would select a numeric item.
- Enable cgd(4) support on all amd64 and i386 XEN kernels.

- amd64
  - Handle protection faults properly, returning SIGSEGV instead of SIGBUS.
  - Ensure FP state is reset, if FP is used in a signal handler. PR 39299.
  - Build kernel modules with -mno-red-zone like the kernel is built.
  - On amd64, add a third free list distinct from the default free list, holding RAM between 16Mb and 4Gb. This helps preventing bus_dma(9) memory allocation failures for 32bit DMA on large-memory machines.

- i386
  -
  - The FPU Tag word is a 16bit register, in FPU (387) mode it defines 2-bit tags for each FPU data register, in MMX mode it defines 1-bit tags for each data register. The single bit tags are stored in the lower 8 bits and thus in the first byte of the save frame.
  - Fix a local user crash.
- ARM: Work-around a possible process exit corner case which can leave stale data in the cache after a context-switch. PR 41058.
- sgimips
  - haltwo(4): Set delta value for the mixer's master channel.
  - Add a driver for the Indy's front panel buttons. Power button presses are reported to sysmon, volume control buttons are reported to PMF.
  - Fixes for newport video.
- alpha
  - A number of multiprocessor fixes. PRs 41106 and 42174.
  - Add support for booting off a couple of common RAID adapters found on several models of alpha systems: mlx [Mylex DAC060] and iop [I2O]. PR 25829.
  - Change the kernel text to 0xfffffc0000430000 (which is where Tru64 has its kernel).
  - The tsc(4) bus initialization was using a single statically allocated extent storage for each tsp(4), which caused a LOCKDEBUG kernel to fail because the extent storage contained a mutex which panics when the second mutex_init() is attempted. Put the extent storage into the tsp_config structure so each tsp(4) gets its own. PR 38358.
- amiga
  - Switch amiga to the common m68k pmap.
  - Fix a problem with the timecounter running backwards everytime the hardware counter wraps and the clock-interrupt is not yet serviced by hardclock().
  - Improve precision of small kernel delays.
  - Fix keyboard handshaking problems on Amiga 1200.
- atari
  - Rewrite the `binpatch`(8) utility to add support for ELF binaries, old src/usr.sbin/mdsetimage sources which support misc executable formats without LGPL'ed libbfd.
  - Add a workaround for annoying "WARNING: negative runtime; monotonic clock has gone backwards" message.
  - `fd`(4): Correctly detect the default density.
- hp300: Make install.md probe `cd`(4) devices properly.
- hpcmips: Fix pcic kthread creation timing. PRs 41791 and 41164.
- m68k: Ensure functions like `mmap`(2), `mremap`(2), `shmat`(2) or `sbrk`(2) return -1 in case of an error. A side benefit of this is to fix a segfault caused by jemalloc when `mmap`(2) failed.

- macppc: pbms(4): Avoid an immediate crash during attach, and fix the aspect ratio of the trackpad on the geyser2 model.
- pmax: Make `ksyms`(4) actually work.
- sh3: Fix logic error in copyinstr() when deciding whether to return EFAULT or ENAMETOOLONG.
- sparc64:
  - Fix long double support in 32bit libc. PR 41406.
  - When preparing the initial trap frame for a new forked lwp, explicitly clear condition code. Otherwise we might catch a signal before we ever return to userland. PR 41302.
  - Fix a disk I/O regression under heavy load.
  - Build the sunleo X driver.
  - Add `lom`(4), a driver for LOMlite lights out management hardware monitor and watchdog timer.
  - sab(4): Allow the RSC to be the console on an E250, by checking for RSC-specific properties, and by not changing the port baud rate.
- vax
  - binutils: Allocate relocation section using bfd_zalloc() to ensure no garbage relocations when not all the entries are used. PR 39182.
  - Keep track of the previous ICR value and hardclock_ticks to ensure the 32 bit counter doesn't go backwards. Also, the ICR runs from -10000 to -1, so adjust the value when reading it. Now mfpr works quite nicely on a 4000/90.
- sparc
  - Enable `ddb`(4) in GENERIC-like kernels.
  - Fix a number of issues with floppies.
  - Add apc(4), a driver for the Aurora Personality Chip (APC) found on SPARCstation-4/5, and emulated by qemu to idle the simulator when the CPU is idle. Only the CPU idle part implemented at this time.
- xen
  - Now a XEN3_DOM0 kernel properly updates the CMOS time.
  - Implement DIOCGDISKINFO for xbd disk driver.
  - xbdback: implement and publish "feature-flush-cache". xbd: if feature-flush-cache is present, use it for DIOCCACHESYNC. Should improve WAPBL reliability of Xen guests on a NetBSD dom0.
  - xennet: Write a "feature-rx-notify" to the xenstore, which is used by recent linux dom0 kernels. This reduces packet loss when using a NetBSD domU on such linux dom0.
  - xennetback: Announce feature-rx-copy and feature-rx-flip. Add support for request-rx-copy. PR 40650.
  - Add i368PAE support to Xen3 dom0.
  - Add Xen3 PCI pass-through support.
  - Fix stalled xbdback detach that would stall the whole xenbus thread, preventing new domUs from being created.

  – Make it possible to use netbsd-5 domUs running on a Xen2 hypervisor.

Miscellaneous

- sh(1): Make the cd builtin accept and ignore -P. PR 42557.
- fdisk(8): Fix issues with large disks.
- savecore(8): Instead of exiting with an obscure error message if -N /kernelname isn't specified, blithely assume the kernel will consume around 20 megs.
- newfs(8): Issue a better error message if attempting to create a file system on a block device. Inspired by PR 41127.
- newfs_msdos(8): Make fs size detection get proper size rather than disk size.
- termcap(3): Only add the ZZ capability for termcap entries that are larger than 1023 bytes.
- dkctl(8): List the partition types addwedge understands. PR 37252. Make dkctl conform to its man page and print the device name on addwedge when the addition was successful (as well as indicating success).
- If the current locale doesn't define the 'thousands' grouping info then use sane defaults (',' every 3 digits). PR 40714.
- pthread(3):
  – Make nanosleep cancelable again.
  – Improve the algorithm used in pthread creation so that it does a better job of reusing dead threads.
  – Make sure thread id is set correctly in case a threaded program forks from a thread other than the main thread. Fixes issues with, e.g., ruby.
- newsyslog(8): Reset ziptype on each line. Failure to do this caused any log file to be compressed if it was listed after a line using Z or J flag.
- nvi(1):
  – Fix a problem where the pattern / didn't match a dollar sign. PR 41781.
  – Make :ESC, #+ and #- work again.
  – Rename "expandtabs" to "expandtab" to match documentation.
  – Implement the "et" abbreviation for expandtab.
  – Fix regexp on LP64 systems. PR 41924.
  – Fix format string bug: filenames may contain % characters.
  – Fix ˜ on big-endian architecturs.
- pstat(8): Distinguish between UFS1 and UFS2 inodes by reading the ufsmount structure, the previous heuristic of comparing the size fields of inode and dinode failed.
- btpin(1): Add a -P flag to attempt immediate pairing.
- ksh(1): Support 0xnn for hexadecimal constants, as well as 16#nn. PR 40512.
- Add support for fr_∗.UTF-8 locale. Setting LANG to fr_∗.UTF-8 won't get the message catalog right (they're encoded in iso-8859-1), but other locale functions should work properly.

- Add Solaris-like dlinfo() interface to the ELF dynamic linker. Implement RTLD_DI_LINKMAP which returns a pointer to the linkmap chain at the given object.
- kill(1): Make sure that numerical signals and pids are in range for their types. PR 42143.
- pcap(3): pcap_lookupnet(): reset ifr before SIOCGIFNETMASK. Without it we get back a bogus netmask. PR 41367.
- man(1): Accept a pathname to a man file, e.g., "man ./man.1"
- setlocale(3): Handle nonexistent locales properly. PR 42124.
- amldb(8): Check validity of parent to avoid a potential segfault.
- sort(1): Fix a ton of bugs including but not limited to PR 18614 PR 27257 PR 25551 PR 221 82 PR 31095 PR 30504 PR 36816 PR 37860 PR 39308 PR 42094.
- user(8): Fix -p flag: rm_eo is the first character *after* the match, so no need for a +1. Blowfish hashes are only 53 chars long, not 54.
- Update libevent to 1.4.11-stable.
- hunt(6): Fix possible remote DoS of a running hunt game, and prevent a possible theoretical attack involving >= 1 billion ammo.
- systat(1): Fix problems with SIGWINCH. PR 42161.
- gzip(1):
  - Fix support for multi-section bzip2 files, as created by pbzip2.
  - Add "pack" uncompression support.
  - Avoid an overflow in suffix handling.
- audio{ctl,play,record}: Add extended WAVE header support, attempt to play a bunch more WAV files.
- Update Postfix to 2.6.5.
- df(1): Block numbers are measured in f_frsize units. Make -P option use this instead of f_bsize. Also account for reserved blocks like normal non-P output. PR 41541.
- Update libfetch to 2.30.
- window(1): Fix a SEGV with certain terminal types. PR 41581.
- Update pkg_install to 20100204:
  - audit-packages.conf(5) has been superseded by pkg_install.conf(5). The default configuration is the same.
  - Support for pkg_view(1) has been retired.
  - The functionality of audit-packages(1) and download-vulnerability-list(1) has moved into pkg_admin(1). Wrapper scripts that handle the common use cases are provided.
- racoonctl(8): Adjust ADMINPORTDIR to match that of racoon (/var/run). PR 41376.
- schedctl(8): Skip LSIDL and LSZOMB threads when retrieving info.
- postinstall(8) now knows about /etc/dhcpcd.conf.
- drvctl(8): Allow querying for root devices in the tree by specifying an empty device name. Ensure that l_devname is NUL-terminated and fail otherwise. If drvctl -l is used without argument, print the root device nodes.

- paste(1): Null-terminate the delimiter list string after processing escapes (which can shorten it) because the code that issues delimiters depends on it being null-terminated. This caused e.g. paste -d ' ' a b to print a '0' at the beginning of each line. PR 41159.
- Don't attempt to read or write ~/.lesshst if it's not a regular file or a symlink to a regular file. Previously, symlinking to /dev/null would cause less to trash /dev/null if run with sufficient privileges. PR 42237.
- Make rtsold_flags default to -a. PR 39657.
- libevent: Add -fno-strict-aliasing to work around problems with GCC 4 and strict-aliasing.
- Renamed a number of internal getline() functions to get_line() so as to compile under -current.
- Install the Xvidtune app-defaults file. PR 41577.
- Fixes to Linux compat:
  - In sendmsg(2), copy the msghdr structure before trying to use it.
  - In linux_sys_sched_getaffinity(), do not leak memory on error.
- Various METALOG fixes, including sorting entries. PRs 24457 and 41155.
- Update DRM code.
- cdce(4): Add support for the Openmoko Freerunner. PR 40049.
- pms(4): Add elantech multi-touch support.
- Big X.Org update.
- ugen(4): ugen_get_cdesc() can return NULL. When this happens, return EINVAL error instead of crashing.
- Fix a race between nfsd and local rm. PR 41147.
- puc(4): Add support for Oxford Semiconductor OXmPCI952 2 port UARTs. Add support for the IBM 481033H SCC.
- sysinst(8): Try ATA/SCSI identify commands and when successful, use the model information along with the disk size when creating the 'Available disks' menu. PR 41925.
- Fix PR kern/41659: add missing splx() in FAST_IPSEC code.
- Add hungarian keyboard layout.
- Add Swiss German keyboard layout.
- Introduce MKSUBPIXEL, which allows enabling subpixel rendering code in FreeType.
- Support drives over 1TB in size in sysinst. Error if drive is over 2TB in in size (and thus over the disklabel limit).
- Update and add some TNF ssh keys to /etc/ssh/ssh_known_hosts.
- Prevent makefs(8) from creating invalid ISO format on rockridge support which causes fatal errors in ARC BIOS firmware on MIPS Magnum R4000. PR 42410.
- sysinst(8): When creating /etc/fstab: for the first swap partition use type "sw,dp" instead of "sw", so dump device gets configured correctly if swap is not on the second partition. PR 42148.
- sys/atomic.h: Make atomics usable from C++.

- wc(1): Add support for "-L" option (longest line) as present in the GNU and FreeBSD versions of "wc".
- libm: Add f{min,max,dim}{,l,f}.
- jemalloc: Fix race condition on reallocation of huge category. PR 42876.
- sys/null.h: Enclose (void *)0 in an extra set of parenthese to make the result usable in arbitrary expressions. PR 41890.

Known Problems

Using block device nodes (e.g., wd0a) directly for I/O may cause a kernel crash when the file system containing /dev is FFS and is mounted with -o log. Workaround: use raw disk devices (e.g., rwd0a), or remount the file system without -o log.

Occassionally, gdb may cause a process that is being debugged to hang when "single stepped". Workaround: kill and restart the affected process.

gdb cannot debug running threaded programs correctly. Workaround: generate a core file from the program using gcore(1) and pass the core to gdb, instead of debugging the running program.

The sparc port does not have functional SMP support in this release.

Features to be removed in a later release

The following features are to be removed from NetBSD in the future:

- Support for soft dependencies, also knows as soft updates (see "softdep" in mount(8)) will be removed in the next major release.

  NetBSD 5.1 includes a preview of WAPBL (Write Ahead Physical Block Logging), which will replace soft dependencies in the next major release. See wapbl(4) and
  http://mail-index.netbsd.org/netbsd-announce/2008/12/14/msg000051.html
  for details.

- Support for Xen 2.0.x. The Xen-3 and hypervisor interface is diverging from Xen-2 as development is ongoing, increasing the maintenance cost for NetBSD.

  It should be considered as deprecated. Users are expected to not rely on it any more beyond this major release.

  Further, at least version 3.1 of Xen will be required to run NetBSD as Dom0 or DomU.

The NetBSD Foundation

The NetBSD Foundation is a tax exempt, not-for-profit 501(c)(3) corporation that devotes itself to the traditional goals and Spirit of the NetBSD Project and owns the trademark of the word "NetBSD". It supports the design, development, and adoption of NetBSD worldwide. More information on the NetBSD Foundation, its composition, aims, and work can be found at: **http://www.NetBSD.org/foundation/**

Sources of NetBSD

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

NetBSD 5.1 Release Contents

The root directory of the NetBSD 5.1 release is organized as follows:

`.../NetBSD-5.1/`

`CHANGES`	Changes between the 4.0 and 5.0 releases.
`CHANGES-5.0`	Changes between the initial 5.0 branch and final release of 5.0.
`CHANGES-5.1`	Changes between the 5.0 and 5.1 releases.
`CHANGES.prev`	Changes in previous NetBSD releases.
`LAST_MINUTE`	Last minute changes and notes about the release.
`README.files`	README describing the distribution's contents.
`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 5.1 has a binary distribution.

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

gnusrc This set contains the "gnu" sources, including the source for the compiler, assembler, groff, and the other GNU utilities in the binary distribution sets.
79 MB gzipped, 450 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.
7 MB gzipped, 32 MB uncompressed

src This set contains all of the base NetBSD 5.1 sources which are not in **gnusrc**, **sharesrc**, or **syssrc**.
59 MB gzipped, 350 MB uncompressed

syssrc This set contains the sources to the NetBSD 5.1 kernel for all architectures as well as the `config`(1) utility.
34 MB gzipped, 197 MB uncompressed

xsrc This set contains the sources to the X Window System.
127 MB gzipped, 694 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:

`MD5`	MD5 digests in the format produced by the command: **cksum -a MD5** *file*.
`SHA512`	SHA512 digests in the format produced by the command: **cksum -a SHA512** *file*.

The SHA512 digest is safer, but MD5 checksums are provided so that a wider range of operating systems can check the integrity of the release files.

NetBSD/mac68k subdirectory structure

The mac68k-specific portion of the NetBSD 5.1 release is found in the `mac68k` subdirectory of the distribution: `.../NetBSD-5.1/mac68k/`. It contains the following files and directories:

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

`binary/`
- `kernel/`
  - `netbsd-GENERIC.gz`
    A kernel containing code for everything supported in this release using the standard SCSI driver.
  - `netbsd-GENERICSBC.gz`
    A kernel containing code for everything supported in this release using the SBC variant of the SCSI driver.
- `sets/` mac68k binary distribution sets; see below.

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

Binary distribution sets

The NetBSD mac68k binary distribution sets contain the binaries which comprise the NetBSD 5.1 release for mac68k. The binary distribution sets can be found in the `mac68k/binary/sets` subdirectory of the NetBSD 5.1 distribution tree, and are as follows:

base The NetBSD 5.1 mac68k **base** binary distribution. You *must* install this distribution set. It contains the base NetBSD utilities that are necessary for the system to run and be minimally functional.
26 MB gzipped, 75 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.
36 MB gzipped, 130 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.
4 MB gzipped, 8 MB uncompressed

kern-GENERIC
This set contains a NetBSD/mac68k 5.1 `GENERIC` kernel, named `/netbsd`. You *must* install either this distribution set or kern-GENERICSBC.
2 MB gzipped, 4 MB uncompressed

kern-GENERICSBC
This set contains a NetBSD/mac68k 5.1 `GENERICSBC` kernel, named `/netbsd`. You *must* install either this distribution set or kern-GENERIC.
2 MB gzipped, 4 MB uncompressed

man This set includes all of the manual pages for the binaries and other software contained in the **base** set. Note that it does not include any of the manual pages that are included in the other sets.
12 MB gzipped, 50 MB uncompressed

misc This set includes the system dictionaries, the typesettable document set, and other files from `/usr/share`.
4 MB gzipped, 13 MB uncompressed

text This set includes NetBSD's text processing tools, including `groff`(1), all related programs, and their manual pages.
3 MB gzipped, 10 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 4.5.0. Binary sets for the X Window System are distributed with NetBSD. The sets are:

xbase The basic files needed for a complete X client environment. This does not include the X servers.
6 MB gzipped, 19 MB uncompressed

xcomp The extra libraries and include files needed to compile X source code.
11 MB gzipped, 43 MB uncompressed

xfont Fonts needed by the X server and by X clients.
31 MB gzipped, 39 MB uncompressed

xetc Configuration files for X which could be locally modified.
1 MB gzipped, 1 MB uncompressed

xserver The X server. This includes the Xmac68k *monochrome* server with man pages.
5 MB gzipped, 13 MB uncompressed

The mac68k binary distribution sets are distributed as gzipped tar files named with the extension **.tgz**, e.g. `base.tgz`.

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

Kernels suitable for booting from an AppleShare server may be found in the `mac68k/binary/kernels` subdirectory of the NetBSD 5.1 distribution tree. These kernels are generally named something like `netbsd-GENERIC.gz` and can be booted as-is by the NetBSD/mac68k Booter utility, if desired. Please note that these kernels are simply gzipped and are not in tar archives.

The Mac OS based utilities necessary for installing and running NetBSD can be found in the `mac68k/installation/misc` subdirectory of the NetBSD 5.1 distribution tree. The important files in this directory are as follows:

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

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

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

These files are all BinHexed, self-extracting archives. If you need them, the sources for these utilities are in the `src` subdirectory.

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

NetBSD/mac68k System Requirements and Supported Devices

NetBSD/mac68k 5.1 runs on several of the older Macintosh computers. About 4 MB of RAM is sufficient to boot a stripped-down custom kernel, and a subset of the system can be squeezed onto a 40 MB hard disk with considerable creativity and persistence. However, 100 MB of disk should be considered a practical minimum, and to do anything more interesting than booting at least 8 MB of RAM and more disk space is recommended.

Please note that to install NetBSD/mac68k 5.1 using the **sysinst** method, your system must have a minimum of 6 MB of RAM and 60 MB of available disk space (i.e. not part of an in-use HFS partition).

Supported models

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

Supported devices

- Onboard SCSI bus and most SCSI tapes, hard drives, and CD-ROMs
- Internal sound — enough to beep on some machines, anyway
- Most basic NuBus video cards (there have been some problems with some 24-bit color cards and with most QuickDraw accelerators)
- Both internal serial ports
- ADB keyboards and mice (both Apple and a number of third party multi-button mice and trackballs are supported)
- Ethernet cards based on the National Semiconductor 8390 and the SONIC (DP83932) chips (Asante, Apple, and a few others — problems still with Ethernet and many NuBus video cards)

- Ethernet cards based on the SMC 91c92 and 91c100 (FEAST) chips. This includes the Asante-FAST 10/100 cards
- Onboard Ethernet based on the SONIC chip for Quadra-series Macs
- Onboard Ethernet based on the MACE (Am79C940) chip for the Quadra AV-series Macs
- Ethernet port on Asante NetDock and Newer Ether MicroDock, for PowerBook Duo series
- Comm-slot Ethernet should be working for most machines/cards

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

- Mac Classic series
- PowerBook Duo series

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

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

Unsupported models

- Macintosh IIfx

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

- Quadra 900/950

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

- PowerPC-based Macs

  This is a separate effort from the mac68k port. PowerMacs use hardware that is quite different from that of the mac68k port. See the NetBSD/macppc port webpage at **http://www.NetBSD.org/ports/macppc/** for more information.

Known hardware issues with this release

- Real Time Clock

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

- SCSI difficulties

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

Getting the NetBSD System on to Useful Media

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

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

The Traditional method of installation is currently supported from the local Macintosh hard drive, from a CD-ROM, or from an AppleShare volume (however, you may upgrade a system from within NetBSD; see the **Upgrading a previously-installed NetBSD System** section for more details). If you are installing from a local hard drive, this means that you'll need at least enough room for the largest file that you will have to install.

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

Each distribution file is in *raw* archive format.

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

You will also need to collect the Mac OS installation tools from the `mac68k/installation/misc` subdirectory of the NetBSD 5.1 distribution: Mkfs, NetBSD/mac68k Installer, and NetBSD/mac68k Booter. These three are in BinHexed, self-extracting archives as Mkfs.sea.hqx, Installer.sea.hqx, and Booter.sea.hqx, respectively. Extract them as you would any other Macintosh application.

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

Preparing your System for NetBSD installation

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

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

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

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

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

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

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

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

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

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

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

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

Once you have decided how to lay out your partitions, you need to calculate how much space to allocate to each partition. A minimal install of NetBSD (i.e. `base.tgz`, `etc.tgz`, and either `kern-GENERIC.tgz` or `kern-GENERICSBC.tgz`) requires about 100MB. A general rule of thumb for sizing the swap partition is to allocate twice as much swap space as you have real memory. Having your swap + real memory total at least 20 MB is also a good idea. Systems that will be heavily used or that are low on real memory should have more swap space allocated. Systems that will be only lightly used or have a very large amount of real memory can get away with less.

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

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

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

Installing the NetBSD System (Sysinst Method)

Running the sysinst installation program

1. *Introduction*

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

2. *Possible hardware-specific issues*

   - *SCSI driver problems*

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

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

   - *The 68LC040 processor*

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

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

3. *General*

   The following is a walk-through of the steps you will take while getting NetBSD installed on your hard disk. **`sysinst`** is a menu driven installation system that allows for some freedom in doing the installation. Sometimes, questions will be asked and in many cases the default answer will be displayed in brackets ("[ ]") after the question. If you wish to stop the installation, you may press `CONTROL-C` at any time, but if you do, you'll have to begin the installation process again from scratch by running the `/sysinst` program from the command prompt. It is not necessary to reboot.

4. *Quick install*

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

- What you need.
  - The distribution sets (in this example, they are on CD).
  - The Mac OS Booter application and an Installation Kernel
  - A Macintosh with a 68020 and MMU, 68030 or 68RC040 processor. An FPU is not required but will be used if present (but see the note above regarding the 68LC040).
  - A CD-ROM drive (SCSI), a hard disk and a minimum of 8 MB of memory installed.
  - The hard disk should have at least 120 + *n* megabytes of space free, where *n* is the number of megabytes of main memory in your system. If you wish to install the X Window System as well, you will need at least 215 MB more.
- The NetBSD Boot Tools folder.
  - Create a Folder on your Mac OS disk for the NetBSD/mac68k components.
  - Copy the Booter application into the newly created Folder. Expand the file if necessary to create the Mac OS executable.
  - Copy the Installation Kernels into the newly created Folder. It is not necessary to **gunzip** compressed kernel files.
  - Single-click on the Booter application icon then select the "Get Info" from the File Menu list. Increase the memory allocation for the Booter to as much as possible for your system. Having a large number of fonts, extensions or sounds installed on your system can cause memory exhaustion problems for the Booter if you don't do this. Also, the extra memory is needed by the Booter to expand compressed kernels while booting.
- The Quick Installation
  - Double-click on the Booter application icon to start executing it. From the **Options** pull-down menu, select **Monitors**, then select **Change Monitor Depth** and make sure B&W is highlighted. Close the window using the **Close** button.
  - From the **Options** pull-down menu select **Boot Options**. This will bring up an option panel. Set the **Auto-set GMT Bias** checkbox in the lower left and then select the **Boot from Mac OS** option at the top of the window. The **Set** button on the right will become active. Use it to locate and select the Installation Kernel file appropriate for your hardware. This will be either `netbsd-INSTALL.gz` or `netbsd-INSTALLSBC.gz`. Close the window using the **Close** button.
  - From the **Options** pull-down menu select the **Boot Now**, or use the Apple-B (Command-B) key combination to start the NetBSD boot process. Do not move the mouse while the boot operation is in progress as this may leave the keyboard locked to NetBSD.

    The main menu will be displayed. Insert the first boot floppy you just created and boot the computer. After language selection, the main menu will be displayed.

```
.**********************************************.
* NetBSD-5.1 Install System                    *
*                                              *
*>a: Install NetBSD to hard disk               *
* b: Upgrade NetBSD on a hard disk             *
* c: Re-install sets or install additional sets *
* d: Reboot the computer                       *
* e: Utility menu                              *
* x: Exit Install System                       *
.**********************************************.
```

– If you wish, you can configure some network settings immediately by choosing the **Utility menu** and then **Configure network**. It isn't actually required at this point, but it may be more convenient. Go back to the main menu.

– Choose **install**.

– You will be guided through some steps regarding the setup of your disk, and the selection of distributed components to install. When in doubt, refer to the rest of this document for details.

– After your disk has been prepared, choose **CD-ROM** as the medium. The default values for the path and device should be ok.

– After all the files have been unpacked, go back to the main menu and select **reboot.**

– NetBSD will now boot. If you haven't already done so in **sysinst**, you should log in as `root` and set a password for that account. You are also advised to edit `/etc/rc.conf` to match your needs.

– Your installation is now complete.

5. *Booting NetBSD*

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

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

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

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

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

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

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

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

6. *Preparing your hard disk*

   You are now at the point of no return. Nothing has been written to your disk yet, but if you confirm that you want to install NetBSD, your hard drive will be modified. If you are sure you want to proceed, select **yes**.

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

7. *Getting the distribution sets*

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

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

8. *Installation from CD-ROM*

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

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

9. *Installation using ftp*

   To be able to install using ftp, you first need to configure your network setup if you haven't already done so. **sysinst** will do this for you, asking you if you want to use DHCP. If you do not use DHCP, you can enter network configuration details yourself. If you do not have DNS set up for the machine that you are installing on, you can just press RETURN in answer to this question, and DNS will not be used.

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

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

10. *Installation using NFS*

    To be able to install using NFS, you first need to configure your network setup if you haven't already done so. **sysinst** will do this for you, asking you if you want to use DHCP. If you do not use DHCP, you can enter network configuration details yourself. If you do not have DNS set up for the machine that you are installing on, you can just press RETURN in answer to this question, and DNS will not be used.

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

    If you did not set up DNS, you will need to specify an IP address instead of a hostname for the NFS server.

11. *Installation from Mac OS file systems*

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

12. *Installation from an unmounted file system*

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

13. *Installation from a local directory*

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

14. *Extracting the distribution sets*

    Before extraction begins, you can elect to watch the files being extracted; the name of each file that is extracted will be shown. This can slow down the installation process considerably on machines with slow graphics consoles or serial consoles. Alternatively, you can choose to see a progress bar. This is the preferred option as it shows progress without significantly slowing down the installation process.

    After all the files have been extracted, the device node files will be created. If you have already configured networking, you will be asked if you want to use this configuration for normal operation. If so, these values will be installed in the network configuration files. The next menu will allow you to select the time zone that you're in, to make sure your clock has the right offset from UTC. Finally you will be asked to select a password encryption algorithm and can then set a password for the "root" account, to prevent the machine from coming up without access restrictions.

15. *Finalizing your installation*

    Congratulations, you have successfully installed NetBSD 5.1. You can now reboot the machine and boot NetBSD from hard disk.

Installing the NetBSD System (Traditional Method)

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

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

Preparing the file system(s)

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

When you have finished converting each partition, select each partition and click on the **Format** button. You will now be asked for a bunch of parameters for the hard drive and the file system. Usually, you can just take the defaults. If you are installing onto removable media (e.g. a Zip, Jaz, or Syquest), please see the FAQ. Note that although this dialog only has the **OK** button, you are not committed, yet. Once you get the values you want, press the **OK** button. A dialog will be presented at this point with two options: **Format** and **Cancel**. If you choose **Cancel**, nothing will be written to your drive. If you choose **Format**, the program will proceed to make a file system.

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

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

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

Installing the files

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

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

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

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

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

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

it printed lines like:

```
sd1 at scsi ID 5
```

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

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

1. Select **Build Devices** from the **File** menu.

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

   `# mkdir path`

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

   `# mkdir /usr`
5. Mount the file systems you wish with the command:

   `# mount device path`

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

   `# mount /dev/sd0g /usr`
6. Type

   `# fstab force`

   to create a proper `/etc/fstab` file.
7. Type **quit** after you have mounted all the file systems.

Installation of base files

Select the **Install** menu item from the **File** menu and install `base.tgz, etc.tgz, either kern-GENERIC.tgz or kern-GENERICSBC.tgz`, and any other sets you wish to install at this time (see the **NetBSD 5.1 Release Contents** for information about what's in each set). The Installer will print out the filename of each file as it is installed, and will take quite some time to install everything (the base package alone can take over two hours on a slow hard drive).

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

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

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

Booting the system

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

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

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

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

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

for more information.

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

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

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

If the system does come up, congratulations, you have successfully installed NetBSD 5.1.

Post installation steps

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

   When you have finished editing `/etc/rc.conf`, type **exit** at the prompt to leave the single-user shell and continue with the multi-user boot.

   Other values that may need to be set in `/etc/rc.conf` for a networked environment are *hostname* and possibly *defaultroute*. You may also need to add an *ifconfig_int* for your ⟨int⟩ network interface, where your on-board, NuBus or PDS interface may be `ae0`, `mc0` or `sn0`. For example:

```
ifconfig_sn0="inet 192.0.2.123 netmask 255.255.255.0"
```

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

```
ifconfig_sn0="inet myname.my.dom netmask 255.255.255.0"
```

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

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

2. Logging in

   After reboot, you can log in as `root` at the login prompt. Unless you've set a password in **sysinst**, there is no initial password. 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.

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 installed the X Window System, you may want to read the chapter about X in the NetBSD Guide:
   http://netbsd.org/docs/guide/en/chap-x.html

   [Color_X] Some systems are capable of supporting a color X Server in either thousands or millions of colors. The mac68k port does not support resolution switching at this time. If your system is capable of running the color X Server it can be installed at this point and the Booter options can be modified to match the resolution depth.

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, pkgsrc. pkgsrc automatically handles any changes necessary to make the software run on NetBSD. This includes the retrieval and installation of any other packages on which the software may depend.

   – More information on the package system is available at
     http://www.NetBSD.org/docs/software/packages.html

   – A list of available packages suitable for browsing is at
     ftp://ftp.NetBSD.org/pub/NetBSD/packages/pkgsrc/README.html

   – Precompiled binaries can be found at
     ftp://ftp.NetBSD.org/pub/pkgsrc/packages/NetBSD/
     usually in the `mac68k/5.1/All` subdir. You can install them with the following commands under `sh`(1):

```
# PKG_PATH=ftp://ftp.NetBSD.org/pub/pkgsrc/packages/NetBSD/mac68k/5.1/All
# export PKG_PATH
# pkg_add -v tcsh
# pkg_add -v bash
# pkg_add -v perl
```

```
# pkg_add -v apache
# pkg_add -v kde
# pkg_add -v firefox
...
```

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

```
# setenv PKG_PATH ftp://ftp.NetBSD.org/pub/pkgsrc/packages/NetBSD/mac68k/5.1/Al
```

Note: Some mirror sites don't mirror the /pub/pkgsrc directory. If you would like to use such mirrors, you could also try the /pub/NetBSD/packages/current-packages/NetBSD/mac68k/5.1/All directory, which may have the same contents.

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

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

```
Warning: package 'foo' was built for a different version
of the OS:
NetBSD/i386 M.N (pkg) vs. NetBSD/i386 5.1 (this host),
```

This warning is harmless if the formal major release numbers are the same between the pkg and your host. Please refer to the NetBSD release glossary and graphs at

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

for more information about NetBSD's release numbering scheme.

– The framework for compiling packages can be obtained by retrieving the file

ftp://ftp.NetBSD.org/pub/NetBSD/packages/pkgsrc.tar.gz

It is typically extracted into /usr/pkgsrc (though other locations work fine) with the commands:

```
# cd /usr
# tar -zxpf pkgsrc.tar.gz
```

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

6. Misc

– Edit /etc/mail/aliases to forward root mail to the right place. Don't forget to run newaliases(1) afterwards.

– The /etc/postfix/main.cf file will almost definitely need to be adjusted. If you prefer a different MTA, then install it using pkgsrc or by hand and adjust /etc/mailer.conf.

– Edit /etc/rc.local to run any local daemons you use.

– Many of the /etc files are documented in section 5 of the manual; so just invoking

```
# man 5 filename
```

is likely to give you more information on these files.

Upgrading a previously-installed NetBSD System

The easiest way to upgrade to NetBSD 5.1 is with binaries, and that is the method documented here.

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

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

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

After a new kernel has been copied to your hard disk, your machine is a complete NetBSD 5.1 system. However, that doesn't mean that you're finished with the upgrade process. You will probably want to update the set of device nodes you have in `/dev`. If you've changed the contents of `/dev` by hand, you will need to be careful about this, but if not, you can just cd into `/dev`, and run the command:

```
# sh MAKEDEV all
```

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

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 5.1.

If your port uses X.Org and you see messages from the X server indicating that no devices were found, you may need to run **X -configure** and update your existing xorg.conf to use the BusID line from the newly-generated config file.

Dual-head support for PC systems has become broken for many configurations with the update to xorg-server 1.6.x, which has removed the userland PCI configuration mechanism, and needs to rely upon the OS. We hope to correct this for future releases. Workaround: The only workaround is non-trivial and requires programming several PCI BAR registers as they previously were in NetBSD 5.0.

If you are updating to NetBSD 5.1 without the aid of sysinst or postinstall and your port uses X.Org, be sure to remove `/usr/X11R7/lib/X11/xkb/symbols/pc` before extracting the xbase set. In the version of X.Org shipped with 5.0, this was a directory, but in more recent X.Org versions it is a file.

pkg_install now depends on the pkgdb cache for automatic conflict detection. It is recommended to rebuild the cache with

```
# pkg_admin rebuild
```

`audit-packages.conf`(5) has been superseded by `pkg_install.conf`(5). The default configuration is the same.

Support for `pkg_view`(1) has been retired.

The functionality of `audit-packages`(1) and `download-vulnerability-list`(1) has moved into `pkg_admin`(1). However, wrapper scripts that handle the common use cases are provided.

Issues when running older binaries on NetBSD 5.1

The pthread libraries from previous versions of NetBSD require that the `sysctl`(3) node `kern.no_sa_support` be set to `0`. This affects the following environments:

– Running a 5.x kernel with an older userland.

– Running an older userland inside a chroot'ed environment on a 5.x system.

– Running older statically linked pthread applications.

The 5.x kernel defaults to `0` for `kern.no_sa_support`, which covers the first case. However, please note that a full installation of 5.x (either from scratch or through an upgrade) will set `kern.no_sa_support` to 1 during the boot process. This means that for the last two cases, you will have to manually set `kern.no_sa_support` to `0`, using either the `sysctl`(8) command or through `sysctl.conf`(5).

Note that **sysinst** will automatically invoke

```
postinstall fix
```

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

Issues affecting an upgrade from NetBSD 3.x releases

See the section below on upgrading from NetBSD 4.x as well.

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

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

Issues fixed by *postinstall*:

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

The following issues need to be resolved manually:

- The users '_proxy', '_rwhod', and '_sdpd' and the groups '_proxy', '_rwhod' and '_sdpd' need to be created and the user 'uucp' needs to be updated.

- A number of things were removed in the NetBSD 4.0 release, including: the evbsh5 port, the Fortran 77 compiler (g77), NETCCITT, NETNS, Sendmail, Sushi, UUCP, and Vinum. If you were using any of these, please see the "Components removed from NetBSD" at

  http://www.NetBSD.org/releases/formal-4/NetBSD-4.0.html#removals

- The replacement of Sendmail by Postfix can be handled automatically by *postinstall* but it is not done by default. If you want to transition to Postfix, the command

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

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

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

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

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

Issues affecting an upgrade from NetBSD 4.x releases

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

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

Issues fixed by *postinstall*:

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

The following issues need to be resolved manually:

- The users '_httpd' and '_timedc' and the groups '_httpd' and '_timedc' need to be created.
- Unprivileged use of the `mount`(8) command now requires the *nosuid* and *nodev* options to be explicitly specified. Previously, these options were automatically enforced even if they were not explicitly specified.
- A number of things have been removed from the NetBSD 5.1 release. See the "Components removed from NetBSD" section near the beginning of this document for a list.

Using online NetBSD documentation

Documentation is available if you installed 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. See **http://www.NetBSD.org/mailinglists/** for a web interface.

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.

Bugs also can be submitted and queried with the web interface at
http://www.NetBSD.org/support/send-pr.html

There are also port-specific mailing lists, to discuss aspects of each port of NetBSD. Use majordomo to find their addresses, or visit **http://www.NetBSD.org/mailinglists/**

If you're interested in doing a serious amount of work on a specific port, you probably should contact the 'owner' of that port (listed below).

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

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

Thanks go to

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

Keith Bostic
Ralph Campbell
Mike Karels
Marshall Kirk McKusick

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

- The Internet Systems Consortium, Inc. for hosting the NetBSD FTP, CVS, AnonCVS, mail, mail archive, GNATS, SUP, Rsync and WWW servers.
- The Internet Research Institute in Japan for hosting the server which runs the CVSweb interface to the NetBSD source tree.
- The Luleå Academic Computer Society for providing the backup services server.
- The Columbia University Computer Science Department for hosting the NYC build cluster.
- The Western Washington University Computer Science Department for running the WWU build cluster.
- The many organizations that provide NetBSD mirror sites.
- Without CVS, this project would be impossible to manage, so our hats go off to Brian Berliner, Jeff Polk, and the various other people who've had a hand in making CVS a useful tool.
- We list the individuals and organizations that have made donations or loans of hardware and/or money, to support NetBSD development, and deserve credit for it at **http://www.NetBSD.org/donations/** (If you're not on that list and should be, tell us! We probably were not able to get in touch with you, to verify that you wanted to be listed.)
- Finally, we thank all of the people who've put sweat and tears into developing NetBSD since its inception in January, 1993. (Obviously, there are a lot more people who deserve thanks here. If you're one of them, and would like to be mentioned, tell us!)

We are...

(in alphabetical order)

The NetBSD core group:

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

The portmasters (and their ports):

Erik Berls	`cyber@NetBSD.org`	**cobalt**
Manuel Bouyer	`bouyer@NetBSD.org`	**xen**
Simon Burge	`simonb@NetBSD.org`	**evbmips**
Simon Burge	`simonb@NetBSD.org`	**pmax**
Simon Burge	`simonb@NetBSD.org`	**sbmips**
Julian Coleman	`jdc@NetBSD.org`	**atari**
Marcus Comstedt	`marcus@NetBSD.org`	**dreamcast**
Andrew Doran	`ad@NetBSD.org`	**amd64**
Andrew Doran	`ad@NetBSD.org`	**i386**
Matthias Drochner	`drochner@NetBSD.org`	**cesfic**
Gavan Fantom	`gavan@NetBSD.org`	**iyonix**
Jaime A Fournier	`ober@NetBSD.org`	**zaurus**
Matt Fredette	`fredette@NetBSD.org`	**sun2**
Ichiro FUKUHARA	`ichiro@NetBSD.org`	**hpcarm**

Chris Gilbert	chris@NetBSD.org	**cats**
Ben Harris	bjh21@NetBSD.org	**acorn26**
Ross Harvey	ross@NetBSD.org	**alpha**
Nick Hudson	skrll@NetBSD.org	**hp700**
Martin Husemann	martin@NetBSD.org	**sparc64**
IWAMOTO Toshihiro	toshii@NetBSD.org	**hpcarm**
Darrin Jewell	dbj@NetBSD.org	**next68k**
Søren Jørvang	soren@NetBSD.org	**sgimips**
Wayne Knowles	wdk@NetBSD.org	**mipsco**
Takayoshi Kochi	kochi@NetBSD.org	**ia64**
Paul Kranenburg	pk@NetBSD.org	**sparc**
Michael Lorenz	macallan@NetBSD.org	**macppc**
Anders Magnusson	ragge@NetBSD.org	**vax**
Cherry G. Mathew	cherry@NetBSD.org	**ia64**
NISHIMURA Takeshi	nsmrtks@NetBSD.org	**x68k**
Tohru Nishimura	nisimura@NetBSD.org	**luna68k**
Tohru Nishimura	nisimura@NetBSD.org	**sandpoint**
Andrey Petrov	petrov@NetBSD.org	**sparc64**
Scott Reynolds	scottr@NetBSD.org	**mac68k**
Tim Rightnour	garbled@NetBSD.org	**ofppc**
Tim Rightnour	garbled@NetBSD.org	**prep**
Tim Rightnour	garbled@NetBSD.org	**rs6000**
Noriyuki Soda	soda@NetBSD.org	**arc**
Ignatios Souvatzis	is@NetBSD.org	**amiga**
Jonathan Stone	jonathan@NetBSD.org	**pmax**
Shin Takemura	takemura@NetBSD.org	**hpcmips**
Matt Thomas	matt@NetBSD.org	**alpha**
Matt Thomas	matt@NetBSD.org	**netwinder**
Jason Thorpe	thorpej@NetBSD.org	**algor**
Jason Thorpe	thorpej@NetBSD.org	**evbarm**
Jason Thorpe	thorpej@NetBSD.org	**shark**
Izumi Tsutsui	tsutsui@NetBSD.org	**ews4800mips**
Izumi Tsutsui	tsutsui@NetBSD.org	**hp300**
Izumi Tsutsui	tsutsui@NetBSD.org	**news68k**
Valeriy E. Ushakov	uwe@NetBSD.org	**landisk**
Nathan Williams	nathanw@NetBSD.org	**sun3**
Steve Woodford	scw@NetBSD.org	**evbppc**
Steve Woodford	scw@NetBSD.org	**mvme68k**
Steve Woodford	scw@NetBSD.org	**mvmeppc**
Reinoud Zandijk	reinoud@NetBSD.org	**acorn32**

The NetBSD 5.1 Release Engineering team:

Manuel Bouyer	bouyer@NetBSD.org
David Brownlee	abs@NetBSD.org
James Chacon	jmc@NetBSD.org
Julian Coleman	jdc@NetBSD.org
Håvard Eidnes	he@NetBSD.org
Liam J. Foy	liamjfoy@NetBSD.org

John Heasley	heas@NetBSD.org
Geert Hendrickx	ghen@NetBSD.org
Soren Jacobsen	snj@NetBSD.org
Phil Nelson	phil@NetBSD.org
Jeff Rizzo	riz@NetBSD.org

NetBSD Developers:

Nathan Ahlstrom	nra@NetBSD.org
Steve Allen	wormey@NetBSD.org
Jukka Andberg	jandberg@NetBSD.org
Julian Assange	proff@NetBSD.org
Lennart Augustsson	augustss@NetBSD.org
Christoph Badura	bad@NetBSD.org
Bang Jun-Young	junyoung@NetBSD.org
Dieter Baron	dillo@NetBSD.org
Robert V. Baron	rvb@NetBSD.org
Alan Barrett	apb@NetBSD.org
Grant Beattie	grant@NetBSD.org
Jason Beegan	jtb@NetBSD.org
Erik Berls	cyber@NetBSD.org
Hiroyuki Bessho	bsh@NetBSD.org
John Birrell	jb@NetBSD.org
Mason Loring Bliss	mason@NetBSD.org
Charles Blundell	cb@NetBSD.org
Rafal Boni	rafal@NetBSD.org
Stephen Borrill	sborrill@NetBSD.org
Sean Boudreau	seanb@NetBSD.org
Manuel Bouyer	bouyer@NetBSD.org
John Brezak	brezak@NetBSD.org
Allen Briggs	briggs@NetBSD.org
Mark Brinicombe	mark@NetBSD.org
Aaron Brown	abrown@NetBSD.org
Andrew Brown	atatat@NetBSD.org
David Brownlee	abs@NetBSD.org
Frederick Bruckman	fredb@NetBSD.org
Jon Buller	jonb@NetBSD.org
Simon Burge	simonb@NetBSD.org
Robert Byrnes	byrnes@NetBSD.org
Pavel Cahyna	pavel@NetBSD.org
D'Arcy J.M. Cain	darcy@NetBSD.org
Daniel Carosone	dan@NetBSD.org
Dave Carrel	carrel@NetBSD.org
James Chacon	jmc@NetBSD.org
Mihai Chelaru	kefren@NetBSD.org
Bill Coldwell	billc@NetBSD.org
Julian Coleman	jdc@NetBSD.org
Ben Collver	ben@NetBSD.org
Marcus Comstedt	marcus@NetBSD.org

Jeremy Cooper	jeremy@NetBSD.org
Chuck Cranor	chuck@NetBSD.org
Alistair Crooks	agc@NetBSD.org
Aidan Cully	aidan@NetBSD.org
Garrett D'Amore	gdamore@NetBSD.org
Johan Danielsson	joda@NetBSD.org
John Darrow	jdarrow@NetBSD.org
Jed Davis	jld@NetBSD.org
Matt DeBergalis	deberg@NetBSD.org
Arnaud Degroote	degroote@NetBSD.org
Rob Deker	deker@NetBSD.org
Chris G. Demetriou	cgd@NetBSD.org
Tracy Di Marco White	gendalia@NetBSD.org
Jaromír Dolecek	jdolecek@NetBSD.org
Andy Doran	ad@NetBSD.org
Roland Dowdeswell	elric@NetBSD.org
Emmanuel Dreyfus	manu@NetBSD.org
Matthias Drochner	drochner@NetBSD.org
Jun Ebihara	jun@NetBSD.org
Håvard Eidnes	he@NetBSD.org
Jaime A Fournier	ober@NetBSD.org
Stoned Elipot	seb@NetBSD.org
Michael van Elst	mlelstv@NetBSD.org
Enami Tsugutomo	enami@NetBSD.org
Bernd Ernesti	veego@NetBSD.org
Erik Fair	fair@NetBSD.org
Gavan Fantom	gavan@NetBSD.org
Hauke Fath	hauke@NetBSD.org
Hubert Feyrer	hubertf@NetBSD.org
Jason R. Fink	jrf@NetBSD.org
Matt J. Fleming	mjf@NetBSD.org
Marty Fouts	marty@NetBSD.org
Liam J. Foy	liamjfoy@NetBSD.org
Matt Fredette	fredette@NetBSD.org
Thorsten Frueauf	frueauf@NetBSD.org
Castor Fu	castor@NetBSD.org
Ichiro Fukuhara	ichiro@NetBSD.org
Quentin Garnier	cube@NetBSD.org
Thomas Gerner	thomas@NetBSD.org
Simon J. Gerraty	sjg@NetBSD.org
Justin Gibbs	gibbs@NetBSD.org
Chris Gilbert	chris@NetBSD.org
Eric Gillespie	epg@NetBSD.org
Brian Ginsbach	ginsbach@NetBSD.org
Paul Goyette	pgoyette@NetBSD.org
Michael Graff	explorer@NetBSD.org
Brian C. Grayson	bgrayson@NetBSD.org
Matthew Green	mrg@NetBSD.org
Andreas Gustafsson	gson@NetBSD.org

Ulrich Habel	rhaen@NetBSD.org
Jun-ichiro itojun Hagino	itojun@NetBSD.org
HAMAJIMA Katsuomi	hamajima@NetBSD.org
Adam Hamsik	haad@NetBSD.org
Juergen Hannken-Illjes	hannken@NetBSD.org
Charles M. Hannum	mycroft@NetBSD.org
Ben Harris	bjh21@NetBSD.org
Ross Harvey	ross@NetBSD.org
Eric Haszlakiewicz	erh@NetBSD.org
John Hawkinson	jhawk@NetBSD.org
HAYAKAWA Koichi	haya@NetBSD.org
John Heasley	heas@NetBSD.org
Geert Hendrickx	ghen@NetBSD.org
René Hexel	rh@NetBSD.org
Iain Hibbert	plunky@NetBSD.org
Kouichirou Hiratsuka	hira@NetBSD.org
Michael L. Hitch	mhitch@NetBSD.org
Ádám Hóka	ahoka@NetBSD.org
Jachym Holecek	freza@NetBSD.org
David A. Holland	dholland@NetBSD.org
Christian E. Hopps	chopps@NetBSD.org
Ken Hornstein	kenh@NetBSD.org
Marc Horowitz	marc@NetBSD.org
Eduardo Horvath	eeh@NetBSD.org
Nick Hudson	skrll@NetBSD.org
Shell Hung	shell@NetBSD.org
Martin Husemann	martin@NetBSD.org
Dean Huxley	dean@NetBSD.org
Love Hörnquist Åstrand	lha@NetBSD.org
Roland Illig	rillig@NetBSD.org
Bernardo Innocenti	bernie@NetBSD.org
Tetsuya Isaki	isaki@NetBSD.org
ITOH Yasufumi	itohy@NetBSD.org
IWAMOTO Toshihiro	toshii@NetBSD.org
Matthew Jacob	mjacob@NetBSD.org
Soren Jacobsen	snj@NetBSD.org
Lonhyn T. Jasinskyj	lonhyn@NetBSD.org
Darrin Jewell	dbj@NetBSD.org
Nicolas Joly	njoly@NetBSD.org
Chris Jones	cjones@NetBSD.org
Søren Jørvang	soren@NetBSD.org
Takahiro Kambe	taca@NetBSD.org
Masanori Kanaoka	kanaoka@NetBSD.org
Antti Kantee	pooka@NetBSD.org
Frank Kardel	kardel@NetBSD.org
Mattias Karlsson	keihan@NetBSD.org
KAWAMOTO Yosihisa	kawamoto@NetBSD.org
Mario Kemper	magick@NetBSD.org
Min Sik Kim	minskim@NetBSD.org

Thomas Klausner	wiz@NetBSD.org
Klaus Klein	kleink@NetBSD.org
John Klos	jklos@NetBSD.org
Wayne Knowles	wdk@NetBSD.org
Takayoshi Kochi	kochi@NetBSD.org
John Kohl	jtk@NetBSD.org
Daniel de Kok	daniel@NetBSD.org
Jonathan A. Kollasch	jakllsch@NetBSD.org
Paul Kranenburg	pk@NetBSD.org
Lubomir Kundrak	lkundrak@NetBSD.org
Jochen Kunz	jkunz@NetBSD.org
Martti Kuparinen	martti@NetBSD.org
Kentaro A. Kurahone	kurahone@NetBSD.org
Arnaud Lacombe	alc@NetBSD.org
Kevin Lahey	kml@NetBSD.org
David Laight	dsl@NetBSD.org
Johnny C. Lam	jlam@NetBSD.org
Martin J. Laubach	mjl@NetBSD.org
Greg Lehey	grog@NetBSD.org
Ted Lemon	mellon@NetBSD.org
Christian Limpach	cl@NetBSD.org
Frank van der Linden	fvdl@NetBSD.org
Joel Lindholm	joel@NetBSD.org
Tonnerre Lombard	tonnerre@NetBSD.org
Mike Long	mikel@NetBSD.org
Michael Lorenz	macallan@NetBSD.org
Warner Losh	imp@NetBSD.org
Tomasz Luchowski	zuntum@NetBSD.org
Federico Lupi	federico@NetBSD.org
Brett Lymn	blymn@NetBSD.org
Paul Mackerras	paulus@NetBSD.org
MAEKAWA Masahide	gehenna@NetBSD.org
Anders Magnusson	ragge@NetBSD.org
Cherry G. Mathew	cherry@NetBSD.org
David Maxwell	david@NetBSD.org
Gregory McGarry	gmcgarry@NetBSD.org
Dan McMahill	dmcmahill@NetBSD.org
Jared D. McNeill	jmcneill@NetBSD.org
Neil J. McRae	neil@NetBSD.org
Julio M. Merino Vidal	jmmv@NetBSD.org
Perry Metzger	perry@NetBSD.org
Luke Mewburn	lukem@NetBSD.org
Jean-Yves Migeon	jym@NetBSD.org
Brook Milligan	brook@NetBSD.org
Minoura Makoto	minoura@NetBSD.org
Simas Mockevicius	symka@NetBSD.org
der Mouse	mouse@NetBSD.org
Joseph Myers	jsm@NetBSD.org
Ken Nakata	kenn@NetBSD.org

Takeshi Nakayama	nakayama@NetBSD.org
Phil Nelson	phil@NetBSD.org
John Nemeth	jnemeth@NetBSD.org
Bob Nestor	rnestor@NetBSD.org
NISHIMURA Takeshi	nsmrtks@NetBSD.org
Tohru Nishimura	nisimura@NetBSD.org
NONAKA Kimihiro	nonaka@NetBSD.org
Takehiko NOZAKI	tnozaki@NetBSD.org
Tobias Nygren	tnn@NetBSD.org
OBATA Akio	obache@NetBSD.org
Jesse Off	joff@NetBSD.org
Tatoku Ogaito	tacha@NetBSD.org
OKANO Takayoshi	kano@NetBSD.org
Masaru Oki	oki@NetBSD.org
Atsushi Onoe	onoe@NetBSD.org
Greg Oster	oster@NetBSD.org
Rui Paulo	rpaulo@NetBSD.org
Jonathan Perkin	sketch@NetBSD.org
Andrey Petrov	petrov@NetBSD.org
Herb Peyerl	hpeyerl@NetBSD.org
Matthias Pfaller	matthias@NetBSD.org
Chris Pinnock	cjep@NetBSD.org
Adrian Portelli	adrianp@NetBSD.org
Peter Postma	peter@NetBSD.org
Dante Profeta	dante@NetBSD.org
Chris Provenzano	proven@NetBSD.org
Niels Provos	provos@NetBSD.org
Mindaugas Rasiukevicius	rmind@NetBSD.org
Michael Rauch	mrauch@NetBSD.org
Marc Recht	recht@NetBSD.org
Darren Reed	darrenr@NetBSD.org
Jeremy C. Reed	reed@NetBSD.org
Antoine Reilles	tonio@NetBSD.org
Tyler R. Retzlaff	rtr@NetBSD.org
Scott Reynolds	scottr@NetBSD.org
Michael Richardson	mcr@NetBSD.org
Tim Rightnour	garbled@NetBSD.org
Alan Ritter	rittera@NetBSD.org
Jeff Rizzo	riz@NetBSD.org
Hans Rosenfeld	hans@NetBSD.org
Gordon Ross	gwr@NetBSD.org
Steve Rumble	rumble@NetBSD.org
Ilpo Ruotsalainen	lonewolf@NetBSD.org
Heiko W. Rupp	hwr@NetBSD.org
Blair J. Sadewitz	bjs@NetBSD.org
David Sainty	dsainty@NetBSD.org
SAITOH Masanobu	msaitoh@NetBSD.org
Kazuki Sakamoto	sakamoto@NetBSD.org
Curt Sampson	cjs@NetBSD.org

Wilfredo Sanchez	wsanchez@NetBSD.org
Ty Sarna	tsarna@NetBSD.org
SATO Kazumi	sato@NetBSD.org
Jan Schaumann	jschauma@NetBSD.org
Matthias Scheler	tron@NetBSD.org
Silke Scheler	silke@NetBSD.org
Karl Schilke (rAT)	rat@NetBSD.org
Amitai Schlair	schmonz@NetBSD.org
Konrad Schroder	perseant@NetBSD.org
Georg Schwarz	schwarz@NetBSD.org
Lubomir Sedlacik	salo@NetBSD.org
Christopher SEKIYA	sekiya@NetBSD.org
Reed Shadgett	dent@NetBSD.org
John Shannon	shannonjr@NetBSD.org
Tim Shepard	shep@NetBSD.org
Takeshi Shibagaki	shiba@NetBSD.org
Naoto Shimazaki	igy@NetBSD.org
Takao Shinohara	shin@NetBSD.org
Takuya SHIOZAKI	tshiozak@NetBSD.org
Daniel Sieger	dsieger@NetBSD.org
Chuck Silvers	chs@NetBSD.org
Thor Lancelot Simon	tls@NetBSD.org
Jeff Smith	jeffs@NetBSD.org
Noriyuki Soda	soda@NetBSD.org
Wolfgang Solfrank	ws@NetBSD.org
SOMEYA Yoshihiko	someya@NetBSD.org
Bill Sommerfeld	sommerfeld@NetBSD.org
Jörg Sonnenberger	joerg@NetBSD.org
Ignatios Souvatzis	is@NetBSD.org
T K Spindler	dogcow@NetBSD.org
Bill Squier	groo@NetBSD.org
Jonathan Stone	jonathan@NetBSD.org
Bill Studenmund	wrstuden@NetBSD.org
Kevin Sullivan	sullivan@NetBSD.org
SUNAGAWA Keiki	kei@NetBSD.org
Kimmo Suominen	kim@NetBSD.org
Robert Swindells	rjs@NetBSD.org
Shin Takemura	takemura@NetBSD.org
TAMURA Kent	kent@NetBSD.org
Shin'ichiro TAYA	taya@NetBSD.org
Ian Lance Taylor	ian@NetBSD.org
Matt Thomas	matt@NetBSD.org
Jason Thorpe	thorpej@NetBSD.org
Christoph Toshok	toshok@NetBSD.org
Greg Troxel	gdt@NetBSD.org
Tsubai Masanari	tsubai@NetBSD.org
Izumi Tsutsui	tsutsui@NetBSD.org
UCHIYAMA Yasushi	uch@NetBSD.org
Masao Uebayashi	uebayasi@NetBSD.org

Shuichiro URATA	ur@NetBSD.org
Valeriy E. Ushakov	uwe@NetBSD.org
Todd Vierling	tv@NetBSD.org
Aymeric Vincent	aymeric@NetBSD.org
Paul Vixie	vixie@NetBSD.org
Mike M. Volokhov	mishka@NetBSD.org
Krister Walfridsson	kristerw@NetBSD.org
Lex Wennmacher	wennmach@NetBSD.org
Leo Weppelman	leo@NetBSD.org
Assar Westerlund	assar@NetBSD.org
Todd Whitesel	toddpw@NetBSD.org
Frank Wille	phx@NetBSD.org
Nathan Williams	nathanw@NetBSD.org
Rob Windsor	windsor@NetBSD.org
Dan Winship	danw@NetBSD.org
Jim Wise	jwise@NetBSD.org
Michael Wolfson	mbw@NetBSD.org
Colin Wood	ender@NetBSD.org
Steve Woodford	scw@NetBSD.org
YAMAMOTO Takashi	yamt@NetBSD.org
Yuji Yamano	yyamano@NetBSD.org
David Young	dyoung@NetBSD.org
Reinoud Zandijk	reinoud@NetBSD.org
S.P.Zeidler	spz@NetBSD.org
Maria Zevenhoven	maria7@NetBSD.org
Christos Zoulas	christos@NetBSD.org

Other contributors:

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

Legal Mumbo-Jumbo

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

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

NetBSD is a registered trademark of The NetBSD Foundation, Inc.
This product includes software developed by the University of California, Berkeley and its contributors.
This product includes software developed by the NetBSD Foundation.
This product includes software developed by The NetBSD Foundation, Inc. and its contributors.
This product includes software developed for the NetBSD Project. See http://www.netbsd.org/ for information about NetBSD.
This product contains software developed by Ignatios Souvatzis for the NetBSD project.
This product contains software written by Ignatios Souvatzis and Michael L. Hitch for the NetBSD project.
This product includes software developed by Intel Corporation and its contributors.
This product contains software written by Michael L. Hitch for the NetBSD project.
This product includes software developed by Microsoft.
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 Adrian Portelli for the NetBSD project.
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 Bruce M. Simpson.
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 Daan Vreeken.
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 K. Kobayashi.
This product includes software developed by K. Kobayashi and H. Shimokawa.
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 Florian Stoehr.
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 Garrett D'Amore.
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 Henrik Vestergaard Draboel.
This product includes software developed by Herb Peyerl.
This product includes software developed by Hidetoshi Shimokawa.
This product includes software developed by Hubert Feyrer for the NetBSD Project.
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 Jeremy C. Reed for the NetBSD project.
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 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 Tinguely and Jim Lowe
This product includes software developed by Markus Wild.
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 Pritchard.
This product includes software developed by Mike Pritchard and contributors.
This product includes software developed by Minoura Makoto.
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 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 Steven M. Bellovin.
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 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 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 Yen Yen Lim and North Dakota State University.
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 Network Research Group at Lawrence Berkeley Laboratory.
This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.OpenSSL.org/)
This product includes software developed by the PocketBSD project and its contributors.
This product includes software developed by the RiscBSD kernel team
This product includes software developed by the RiscBSD team.
This product includes software developed by the SMCC Technology Development Group at Sun Microsystems, Inc.
This product includes software developed by the University of California, Berkeley and its contributors, as well as the Trustees of Columbia University.
This product includes software developed by the University of California, Lawrence Berkeley Laboratory and its contributors.
This product includes software developed by the University of California, Lawrence Berkeley Laboratory.
This product includes software developed by the University of Illinois at Urbana and their contributors.
This product includes software developed by the Urbana-Champaign Independent Media Center.
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 by Kyma Systems.
This product includes software developed for the NetBSD Project by Kyma Systems LLC.
This product includes software developed for the NetBSD Project by Matthias Drochner.
This product includes software developed for the NetBSD Project by Michael L. Hitch.
This product includes software developed for the NetBSD Project by Perry E. Metzger.
This product includes software developed for the NetBSD Project by Scott Bartram and Frank van der Linden
This product includes software developed for the NetBSD Project by Allegro Networks, Inc., and Wasabi Systems, Inc.
This product includes software developed for the NetBSD Project by Genetec Corporation.
This product includes software developed for the NetBSD Project by Jonathan Stone.
This product includes software developed for the NetBSD Project by Piermont Information Systems Inc.
This product includes software developed for the NetBSD Project by SUNET, Swedish University Computer Network.
This product includes software developed for the NetBSD Project by Shigeyuki Fukushima.
This product includes software developed for the NetBSD Project by Wasabi Systems, Inc.
This product includes software developed under OpenBSD by Per Fogelstrom Opsycon AB for RTMX Inc, North Carolina, USA.
This product includes software developed under OpenBSD by Per Fogelstrom.
This software is a component of "386BSD" developed by William F. Jolitz, TeleMuse.
This software was developed by Holger Veit and Brian Moore for use with "386BSD" and similar operating systems. "Similar operating systems" includes mainly non-profit oriented systems for research and education, including but not restricted to "NetBSD", "FreeBSD", "Mach" (by CMU).
This software includes software developed by the Computer Systems Laboratory at the University of Utah.
This product includes software developed by Computing Services at Carnegie Mellon University (http://www.cmu.edu/computing/).
This product includes software developed by Marshall M. Midden.
This product includes software developed or owned by Caldera International, Inc.
The Institute of Electrical and Electronics Engineers and The Open Group, have given us permission to reprint portions of their documentation.

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

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

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

This notice shall appear on any product containing this material

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

This product includes software developed by John P. Wittkoski.
This product includes software developed by Colin Wood.

The End

Contributions

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

- All of the users who have supplied us with good bug reports and moral support.
- The Alice Group (Allen K. Briggs, Chris P. Caputo, Michael L. Finch, Bradley A. Grantham, and Lawrence A. Kesteloot), without whom there would be no NetBSD port for the Macintosh.
- Steven R. Allen for keeping our snapshot distributions up-to-date.
- Stephen C. Brown for maintaining the Installer application.
- Denton Gentry and Yanagisawa Takeshi for their work on the SONIC Ethernet driver.
- Paul Goyette, Taras Ivanenko, Ken Nakata, and Michael R. Zucca for invaluable work towards supporting color X.
- Takashi Hamada and John Wittkoski beating the direct ADB hardware driver into submission.
- David Huang for getting MACE Ethernet and basic DMA working on the AV Macs.
- Scott Jann for acquiring a IIx and a IIci, used for building and testing release sets.
- Scott Kaplan for lending his IIci and Kensington Turbo Mouse for IIci/IIsi banked memory and internal video as well as non-Apple ADB devices.
- Noah M. Kieserman for lending a PowerBook 520C for tracking down several bugs on that platform.
- Markus Krummenacker for monetary donations.
- Glan Lalonde for an invaluable IIci page table dump.
- Dan McMahill for lending a PowerBook 165 to tweak ADB support on the PowerBook 160 and 180 family laptops.
- Bob Nestor for (unofficially) maintaining the Mkfs utility, and providing a lot of useful information about the ROM vectors used by different systems.
- Brad Parker for serial and Ethernet drivers/improvements.
- Brian R. Gaeke and Nigel Pearson for tweaking, polishing, and performing the occasional major refit on the Booter application.
- Scott Redman for lending Brad Grantham a PowerBook 160.
- Craig Ruff for assembling an '030 `pmove ttx` instruction.
- Brad Salai for lending an Ethernet card to help resolve interrupt conflicts.
- Larry Samuels for monetary donations.
- Peter Siebold for lending his IIvx in support of ADB and IIvx internal video.
- Glen Stewart for lending a Carrera040 accelerator which, while still unsupported, helped to track down memory management bugs for '030-based machines.
- Bill Studenmund for providing a stable front end to the machine- independent serial driver.

- Schuyler Stultz for the loan of his Macintosh II when we desperately needed another machine on which to compile and test during the '93 Xmas vacation.
- Tenon Intersystems for monetary donations, MachTen, and Brad's access to several machines and documentation after hours.
- Virginia Tech English Department for loan of a IIci w/ NuBus video and 32 MB of RAM — the first IIci to run NetBSD/mac68k.
- Rob Windsor for donating a variety of Macintosh II-family systems, a Centris 650, a Quadra 700, and several boxes full of miscellaneous peripherals and parts in the interest of ensuring adequate testing and working out minor (and not-so-minor) problems.
- Colin Wood for maintaining a host of NetBSD/mac68k documentation, including the FAQ, Meta-FAQ, and OS Info documents.