**The NetBSD Guide**

by The NetBSD Developers

Published 2010/01/16 07:55:45

Copyright © 1999, 2000, 2001, 2002 Federico Lupi
Copyright © 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010 The NetBSD Foundation

All brand and product names used in this guide are or may be trademarks or registered trademarks of their respective owners.

# Table of Contents

**Purpose of this guide** ..... **xvii**
**I. About NetBSD** ..... **xviii**

4.3 Performing the upgrade ..... 36

14.6 Further Reading ..... 114
Bibliography ..... 114
15 Concatenated Disk Device (CCD) configuration ..... 116
15.1 Install physical media .....

18.2 Introduction .....159

18.10 Sample PAM Module ..... 175
18.11 Sample PAM Conversation Function ..... 178
18.12 Further Reading ..... 179
Bibliography ..... 179
19 Tuning NetBSD ..... 181
19.1 Introduction ..... 181
19.1.1 Overview ..... 181
19.1.1.1 What is Performance Tuning? .....

19.7.4 Summary ..... 202
19.8 System Tuning ..... 202
19.8.1 Using sysctl ..... 202
19.8.2 tmpfs & mfs ..... 203
19.8.3 Soft-dependencies ..... 203
19.8.4 Journaling ..... 204
19.8.5 LFS ..... 204
19.9 Kernel Tuning ..... 205
19.9.1 Preparing to Recompile a Kernel ..... 205
19.9.2 Configuring the Kernel ..... 205
19.9.2.1 Some example Configuration Items ..... 206
19.9.2.2 Some Drivers .....

24.7 A common LAN setup.....267

**D. Acknowledgements .....364**

.....

# List of Tables

## 2.2 Install preparations

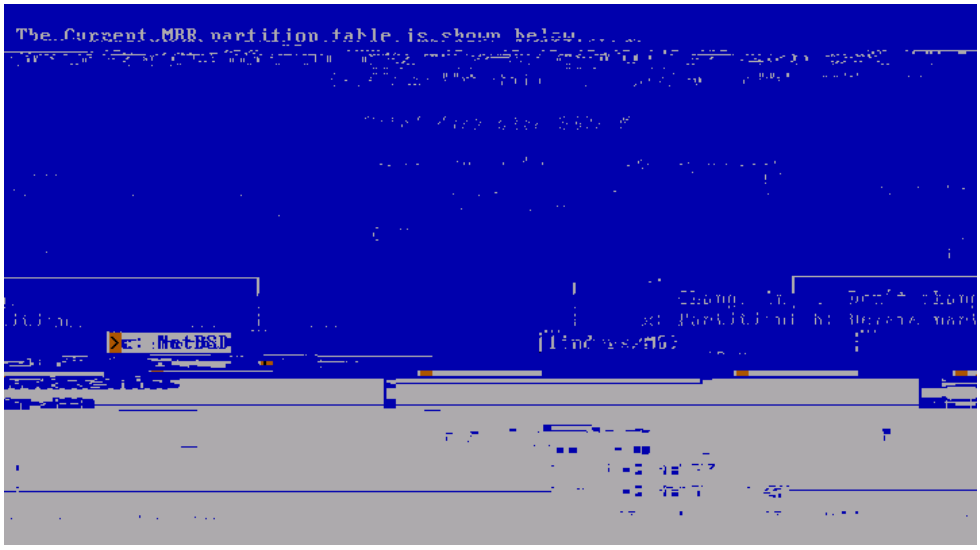
### 2.2.1 The INSTALL document

**Note:** If NetBSD shares the hard disk with another operating system (like in the previous example)

# Chapter 3

selecting a partition, a menu with options for that partition will appear, as shown in Figure 3-10.

**Figure 3-10. Partition options**



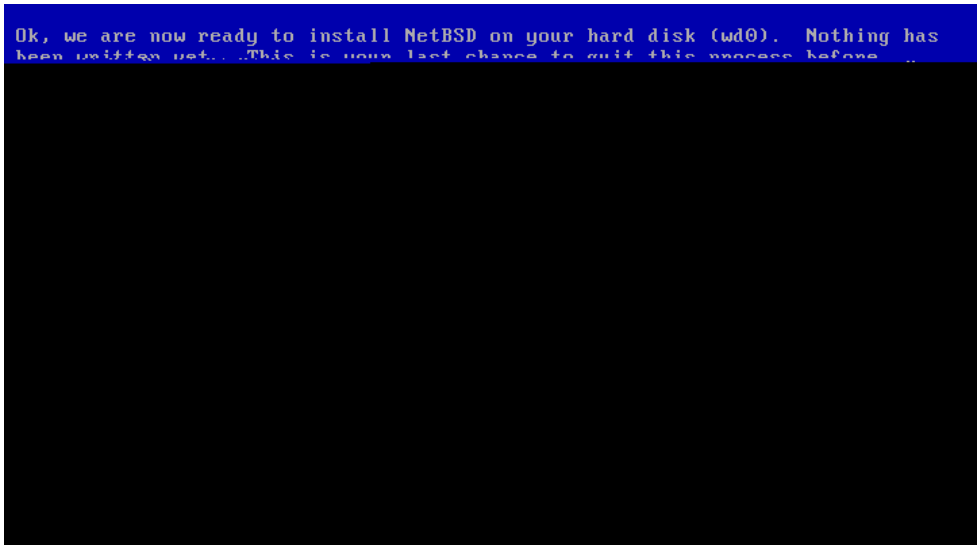

To create a new partition, the following information must be supplied:

•

**Figure 3-12. Edit partitions?**

When you choose to set the sizes of the NetBSD partitions you c

**Figure 3-17. Last chance to abort**




## 3.9 The disk preparation process

**Figure 3-30. NFS install screen**

Figure 3-31

## 3.13 Finishing the installation

**Figure 3-39. Reboot to finish installation**

# Chapter 4

# III. System configuration, administration and tuning

Chapter 5

# *The first steps on NetBSD*

```
login: root
password:
```

```
# pkg_add -uv firefox
```

The following command will force an update of firefox and all of its dependencies:

```
# pkg_add -fuuv firefox
```

All details about package management can be found in *The pkgsrc guide* (http://www.NetBSD.org/docs/pkgsrc/index.html)

# Chapter 6
# *Editing*

## 6.1 Introducing vi

a Append after cursor.

### 6.1.7 A Sample Session

- Ex/Vi Reference Manual by Keith Bostic
-

Chapter 7

# *The rc.d System*

# Chapter 8

For example, `screen 7` could be enabled in `/etc/wscons.conf` and X could be started with

```
encoding -> it
```

This setting will last until the next reboot. To make it permanent, add a `encoding` line to `/etc/wscons.conf`: it will be executed automatically the next time you reboot.

```
# cp /etc/wscons.conf /etc/wscons.conf.orig
# echo encoding it >>/etc/wscons.conf
```

Please be careful and type two **>** characters. If you type onlytw7yTd [(6305f52.25043(o)(a)]4343(n)x)-5.8887(t)-227.9(t)0.9

# Chapter 9

```
Driver      "kbd"
Option      "XkbRules" "xorg"
Option      "XkbModel" "pc105"
Option      "XkbLayout" "de"
Option      "XkbOptions" "ctrl:nocaps"
```

```
EndSection
```

## 9.7 Starting X

```
# pkg_add -v openbox
```

To build it with pkgsrc, run:

```
# cd /usr/pkgsrc/wm/openbox
# make install
```

Openbox is now installed; to start it you must modify your `.xinitrc` file: substitute the line which calls

Chapter 10

# *Linux emulation*

## 10.2 Directory structure

If we examine the outcome of the installation of the Linux libraries and programs we find that

*Emmnuee10.37085yuu Mr*

*Emmnuee10.37085yuu Mr*

### 11.5.3 audioplay(1)

1

```
!"#$%&'()
```

#

# 19erea

```
# file -s /dev/cd0d
```

### 14.2.3 Verification Methods

Another aspect of `cgd` that needs some attention are the verification methods **cgdconfig** provides. These

Once the installation is complete, you should examine the disklabel(8) and fdisk(8) / sunlabel(8) outputs on the system:

```
# df
Filesystem   1K-blocks        Used       Avail %Cap Mounted on
/dev/wd0a       9487886      502132     8511360   5% /
```

On i386:

```
#
```

On Sparc64 the command / output differs slightly:

#

```
fifo 100
```

Note that `absent`

```
rpm: 3600
interleave: 1
trackskew: 0
cylinderskew: 0
headswitch: 0 # microseconds
track-to-track seek: 0 # microseconds
drivedata: 0

#        size    offset     fstype [fsize bsize cpg/sgs]
 a:  19015680         0     4.2BSD      0     0     0  # (Cyl.      0 - 18569)
 b:    525184  19015680       swap                     # (Cyl.  18570 - 19082*)
 d:  19540864         0     unused      0     0        # (Cyl.      0 - 19082*)
```

On sparc64:

```
# disklabel -r -e -I raid0
[...snip...]
total sectors: 19539968
[...snip...]
3 partitions:
#        size    offset     fstype [fsize bsize cpg/sgs]
 a:  19251200         0     4.2BSD      0     0     0  # (Cyl.      0 -  18799)
 b:    288768  19251200       swap                     # (Cyl.  18800 -  19081)
 c:  19539968         0     unused      0     0        # (Cyl.      0 -  19081)
```

Next, format the newly created `/` partition as a 4.2BSD FFSv1 File System:

```
#
```

```
File system:         /dev/rwd1a
Primary bootstra
```

```
Device      1K-blocks     Used    Avail Capacity  Priority
```

```
# raidctl -v -s raid0
Components:
          component0: failed
           /dev/wd1a: optimal
Spares:
           /dev/wd0a: spare
[...snip...]
# raidctl -F component0 raid0
RECON: initiating reconstruction on col 0 -> spare at col 2
 11% |****                                   | ETA:    04:26 \
```

Depending on the speed of your hardware, the reconstruction



```
| ETA6 01:00
# raidctl -v --s raid1
Componenes:
```

**Figure 16-7. Award BIOS i386 Boot Disk0/wd0**

Save changes and exit.

```
>> NetBSD/i386 BIOS Boot, Revision 5.2 (from NetBSD 5.0.2)
>> Memory: 639/130048 k
Press return to boot now, any other key for boot menu
booting hd0a:netbsd - starting in 15
```

Notice how your custom kernel detects controller/bus/stock circuit.

2(u)-56071(c)-237365 9.96264 g(l)0.965528(o)-5.8911i

## 17.1 Anatomy of NetBSD Logical Volume Manager

```
# disklabel -r sd0
            [...snip...]
            bytes/sector: 512
            sectors/track: 63
            tracks/cylinder: 16
            sectors/cylinder: 1008
```

```
Root partition: No
Last configured as: raid0
Parity status: DIRTY
```

Chapter 18

# *Pluggable Authentication Modules (PAM)*

```
  }
  execve(*args, args, environ);
  warn("execve()");
  _exit(1);
 default:
  /* parent: wait for child to exit */
  waitpid(pid, &status, 0);

  /* close the session and release PAM resources */
  pam_err = pam_close_session(pamh, 0);
  pam_end(pamh, pam_err);

  exit(WEXITSTATUS(status));
 }

pamerr:
 fprintf(stderr, "Sorry\n");
err:
 pam_end(pamh, pam_err);
 exit(1);
}
```

## 18.10 Sample PAM Module

The following is a minimal implementation of pam_unix(8), offering only authentication services. It should build and run with most PAM implementations, but takes advantage of OpenPAM extensions if available: none

```
{
#ifndef _OPENPAM
 const void *ptr;
 const struct pam_conv *conv;
 struct pam_message msg;
 const struct pam_message *msgp;
 struct pam_response *resp;
#endif
 struct passwd *pwd;
 const char *user;
 char *
```

```
#ifdef PAM_MODULE_ENTRY
PAM_MODULE_ENTRY("pam_unix");
#endif
```

## 18.11 Sample PAM Conversation Function

## User Manuals

*PAM Administration (http://www.sun.com/software/solaris/pam/pam.admin.pdf)*

# Chapter 19

# *Tuning NetBSD*

#### 19.1.1.2 When does one tune?

Many NetBSD users rarely have to tune a system. The GENERIC kernel may run just fine and the layout/configuration of the system may do the job as well. By t

## 19.3 Visual Monitoring Tools

NetBSD ships a variety of performance monitoring tools with the system. Most of these tools are

```
 0   on retransmit timeout          0   urgent data only
 0   by keepalive                   0   control
 0   by persist
                                   29 total TCP packets received
11 potential rtt updates           17   in sequence
11 successful rtt updates           0   completely duplicate
 9 delayed acks sent                0   with some duplicate data
 0 retransmit timeouts              4   out of order
 0 persist timeouts                 0   duplicate acks
 0 keepalive probes                11   acks
 0 keepalive timeouts               0   window probes
                                    0   window updates
```

Now that is informative. The first poll is accumulative, so it is possible to see quite a lot of information in the output when sysstat is invoked. Now, while that may be int

network equipment. Not unlike ping, if the host itself is sus

```
ash                 0:b0:d0:de:49:df   UH
```

There are plenty of third party tools available, however, NetBSD comes shipped with a good tool set for tracking down network level performance problems.

#### 19.7.1.1 Using kgmon

To start kgmon:

```
$
```

At the end of the call graph right after the terms section is an

### 19.8.3 Soft-dependencies

```
options         INET6           # IPV6
options         IPFILTER_LOG    # ipmon(8) log support
```

IPFILTER_LOG is a nice one to have around since the server will be running ipf.

```
ex0: MAC address 00:50:04:83:ff:b7
UI 0x001018 model 0x0012 rev 0 at ex0 phy 24 not configured
ex1 at pci0 dev 19 function 0: 3Com 3c905B-TX 10/100 Ethernet (rev. 0x30)
ex1: interrupting at irq 11
ex1: MAC address 00:50:da:63:91:2e
exphy0 at ex1 phy 24: 3Com internal media interface
exphy0: 10baseT, 10baseT-FDX, 100baseTX, 100baseTX-FDX, auto
...
```

```
                 CONTENTS, ALLOC, LOAD, READONLY, DATA
11 link_set_evcnts 00000030  c083b7cc  c083b7cc  0073c7cc  2**2
                 CONTENTS, ALLOC, LOAD, READONLY, DATA
12 .data         00048ae4  c083c800  c083c800  0073c800  2**5
                 CONTENTS, ALLOC, LOAD, DATA
13 .bss          00058974  c0885300  c0885300  00785300  2**5
                 ALLOC
                                                          **
```

#

```
% sdpquery -d ubt0 -a phone search DUN

Record Handle: 0x00010000
Service Class ID List:
        Dial-Up Networking (0x1103)
        Generic Networking (0x1201)
Protocol Descriptor List:
        L2CAP (0x0100)
        RFCOMM (0x0003)
                Protocol specific parameter #1: u/int8/bool 1
Bluetooth Profile Descriptor List:
        Dial-Up Networking (0x1103) ver. 1.0
```

Most likely, the phone will request authentication before it allows connections to the DUN service, so before you make the first connection you may need to provTd [(t)-251ne P

## 22.3 Speaker

#

# IV. Networking and related issues

# Chapter 23

There are other protocol suites such as DECNET, Novell's IPX/SPX or Microsoft's NetBIOS, but these are not currently supported by NetBSD. These protocols diff

application using networking-calls. The resolver will then try to resolve (hence the name ;-) the

- Bigger address space
- Support for mobile devices
- Built-in security

#### 23.7.2.1 Bigger Address Space

The bigger address space that IPv6 offers is the most obvious

127.0.0.1

subnet, and one to address the hosts on that subnet. Usually, this is done on byte (8 bit) boundaries.

```
        file "db.reverse";
};
```

```
options         INET6           # IPV6
```

If you want to use IPv6, this is your option. If you don't want IPv6, which is part of NetBSD since the 1.5 release, you can remove/comment out that option. See the inet6(4) manpage and

2. Edit the file `/etc/resolv.conf` and check

```
ATZ
OK
~.

Disconnected.
#
```

**tcpdump**

### 24.8.2 Connecting NetBSD and Windows NT

e.g. ethernet or FDDI. 6over4 is documented in RFC2529. It's main drawback is that you do need

After these commands, you are connected to the IPv6-enabled

### 24.9.8 Quickstart using pkgsrc/net/hf6to4

So far, we have described how 6to4 works and how to set it up manually. For an automated way to make

The example assumes that you use the "ppp0" interface on your firewall to connect to the Internet.

# Chapter 25

and program name and arguments are "ftpd -ll". Notice in the last field, the program name is different from the service-name.

```
nfs             100003  nfsprog
```

## 25.7 Allowing and denying hosts -

Chapter 26

# *The Domain Name System*

## 26.2 The DNS Files

### 26.2.1 `/etc/namedb/named.conf`

cause lookup problems if a particular client decides it want

Line 7:

The minimum Time To Live.

Line 8:

This is the Nameserver line, which uses a "NS" resource record to show that "localhost" is the only DNS server handing out data for this zone (which is "@", which indicates the zone name used in the `named.conf` file, i.e. "diverge.org") is, well, "localhost".

Line 9:

This is the localhost entry, which7(h)-86180.96552Ans a "NS" resource record that "locale 8699.261538.04 -12.9s(n

```
;
;       This file holds the information on root name servers needed to
;       initialize cache of Internet domain name servers
;       (e.g. reference this file in the "cache  .  <file>"
```

As you've probably noticed, the address is the same, but the m

# Chapter 27

*Chapter 28 Introduct-5q 10-71(e2.69226153(n)-5.8521256742930.8 7e)-1.2.54964 Tf 1 0 0-1.666*

### 29.1.1 NFS setup example

messages from ntpdate witku(d)-5.888s(r)3.5582u(d)-5.8887(a)-1.6663l(t)0.96552l(i)0.96307y(e)-4.74374(e)-1.3899n(o)-5.

# V. Building the system

And for the “xsrc” module:

```
$
```

```
cvs     -q
```

## 30.5 Sources on CD (ISO)

If you prefer to download (and maybe burn) a CD-ROM image with the NetBSD source, just fetch `sourcecd-<RELEASE-NUMBER>.iso` from ftp.NetBSD.org or any other mirror.

The `sourcecd-<RELEASE-NUMBER>.iso` file is located under /pub/NetBSD/iso/<*RELEASE*>, where <*RELEASE-NUMBER*> is a release of NetBSD, for example, 5.1:

# Chapter 31

```
nbdb                    nbmknod                 sparc64--netbsd-g++
nbeqn                   nbmktemp                sparc64--netbsd-g77
nbfgen                  nbmsgc                  sparc64--netbsd-gcc
nbfile                  nbmtree                 sparc64--netbsd-gcc-3.3
nbgencat                nbnroff                 sparc64--netbsd-gccbug
nbgroff                 nbpax                   sparc64--netbsd-gcov
nbhexdump               nbpic                   sparc64--netbsd-ld
nbhost-mkdep            nbpwd_mkdb              sparc64--netbsd-lint
nbindxbib               nbrefer                 sparc64--netbsd-mdsetim
nbinfo                  nbrpcgen                sparc64--netbsd-nm
nbinfokey               nbsoelim                sparc64--netbsd-objcopy
nbinstall               nbstat                  sparc64--netbsd-objdump
nbinstall-info          nbsunlabel              sparc64--netbsd-ranlib
nbinstallboot           nbtbl                   sparc64--netbsd-readelf
nblex                   nbtexi2dvi              sparc64--netbsd-size
nblorder                nbtexi(907(p)-2.7(o)-2.2(t)-10789.8212.24907(i)-2.24907(b)864.24907(
```

```
LDSTATIC=-static
```

### 31.7.3 Using `build.sh` options

Besides variables in environment and

MKINFO

MKSHARE

INSTALLWORLDDIR

```
% file netbsd
netbsd: ELF 64-bit MSB executable, SPARC V9, version 1 (SYSV), statically linked, not stripped
```

Chapter 33

# *Updating an existing system from sources*

# Appendix B.

# *Contributing to the NetBSD guide*

When runniril60296264 -2mXthk

This software was developed for the FreeBSD Project by ThinkSec AS and Network Associates Laboratories, the Security Research Division of

# Appendix E.
# *Bibliography*

## Bibliography