Howto upgrade a NetBSD-1.5.x system with mirrored disks to NetBSD-1.6

It isn't possible to just boot a NetBSD 1.6 CD and upgrade with sysinst. This document describes the steps I took, to manually upgrade my NetBSD machines from 1.5.3 to 1.6.

Use at your own Risk!

Version 1.1
30. October 2002

Prerequisites

- an up to date backup of your systems data
- the binary sets from ftp.netbsd.org/pub/NetBSD/NetBSD-1.6/i386/binary/sets
- a NetBSD-1.5.x system willing to upgrade (older releases may also work, but I didn't test those)

You can also upgrade a remote (i386) machine through an ssh session. You just have to be prepared to go on location if something goes wrong. (like for example: I had a machine that saw its NIC's (DEC DE450-CA) as de0 and de1 with my NetBSD-1.5 custom kernel, but as tlp0 and tlp1 with the 1.6 GENERIC kernel...)

Sparc machines have to go single-user in order to install new bootblocks, so you can't upgrade the sparcs remotely.

Overview

The binary upgrade process is really easy. But it is also easy to ruin you system by doing something wrong. So I usually create checklists for such tasks.

- Step1: Save some files and directories
- Step2: Extract the sets
- Step3: Set a root password
- Step4: Create device files
- Step5: Update bootblocks
- Step6: Recreate your /etc/* files and other things to clean up
- Step7: reboot and test

Step1: Save some files and directories

Unzipping/untarring the binary sets has the effect of overwriting almost everything on your system. Especially the /etc directory, which holds all your confi guration data.

mv /etc /etc.old

I don’t know exactly what fi les will be overwritten in /root, but /root/.profi le certainly is, so I also save a copy of /root:

cp -pR /root /root.old

The crontabs seem also to disappear, so we save them in our (saved) root directory:

tar -czvf /root.old/crontabs.backup.tar.gz /var/cron/tabs

And just in case: save the current kernel:

cp /netbsd /netbsd-1.5

Step2: Extract the sets

Now, with your backup in a secure place, extract shiny new NetBSD 1.6 fi les onto your harddisks.

cd /
gzip -dc base.tgz | tar -xpvf -
gzip -dc comp.tgz | tar -xpvf -
gzip -dc etc.tgz | tar -xpvf -
gzip -dc games.tgz | tar -xpvf -
gzip -dc kern-GENERIC.tgz | tar -xpvf -
gzip -dc man.tgz | tar -xpvf -
gzip -dc misc.tgz | tar -xpvf -
gzip -dc text.tgz | tar -xpvf -
gzip -dc xbase.tgz | tar -xpvf -
gzip -dc xcomp.tgz | tar -xpvf -
gzip -dc xcontrib.tgz | tar -xpvf -
gzip -dc xfont.tgz | tar -xpvf -
gzip -dc xmisc.tgz | tar -xpvf -
gzip -dc xserver.tgz | tar -xpvf -

Instead of extracting a generic kernel, you can also put a custom kernel in /netbsd.

You can shoot yourselve in the foot (like I did :-) by having an umask of 007 and not giving the -p flag to tar. Very strange and hard to track down problems can arise from such things...

Take care NOT to reboot your system until after step5!

Step3: Set a root password

You got a new account database where root has no password. Change this now.

Step4: Create device files

```
cd /dev
./MAKEDEV all
```

Step5: Update bootblocks

AFAIK, you will not be able to boot NetBSD-1.6 kernels with 1.5.x or older bootblocks. So you'll have to upgrade those too. If you have RAID1 setup according to my paper "Mirroring with NetBSD 1.5.2 and RaidFrame" your bootblocks will be on special boot partitions (/dev/rwd0f and /dev/rwd1f for example) So the commands you'll have to enter will be something like:

```
/usr/mdec/installboot /usr/mdec/biosboot.sym /dev/rwd0f
/usr/mdec/installboot /usr/mdec/biosboot.sym /dev/rwd1f
```

But don't just type that in blindly, make sure to specify the partitions where your bootblocks are!

Step6: Recreate your /etc/* files and other things to clean up

This is the step that will take the most time. You can go the save way and edit the new fi les in /etc and add the values you had in the /etc.old fi les.

Or you can try and just copy the fi les from /etc.old over to /etc.

Here is a list of fi les, I had to look at. Some need special treatment, more words on those follow after the table.

File or directory to inspect	X
/etc/rc.conf	
/etc/rc.d/* scripts	
/etc/hosts	
/etc/postfi x directory	
/etc/gettytab	
/etc/ntp.conf	
/etc/sysctl.conf	
/etc/ssh/* directory (see below)	
/etc/syslog.conf	
/etc/localtime (link)	
/etc/mailer.conf	
/etc/raid?.conf	
/etc/ipsec.conf	
/etc/smb.conf	
/etc/samba directory	
/etc/mk.conf	
/etc/printcap	
/etc/aliases	
/etc/passwd (see below)	
/etc/group	
/etc/inetd.conf	
/etc/ipf.conf	
/etc/ipnat.conf	
/etc/ifconfi g.*	
/etc/myname	
/etc/mygate	
/etc/newsyslog.conf	
/etc/XF86Confi g (see below)	
/etc/fstab	
/etc/resolv.conf	
/etc/isdn directory (see below)	
/etc/ttys	

You may have additional fi les, you need to change.

/etc/ssh/* directory You will want to copy your old ssh_* keys (Public and private Keys) over. Also, have a look at sshd_confi g.

/etc/passwd (Useraccounts)	Well, not exactly /etc/passwd. But you need to add the users you had before manually with *vipw* or *adduser*. There were changes in the User Account Database structure, so you can't just copy them over.
/etc/XF86Config	On the i386 architecture, XFree86 is now version 4.2. I had to recreate /etc/XF86Confi g by running *xf86confi g*.
/etc/isdn directory	(if you are using ISDN) Now, this is a mean one. They changed the isp device name (Sync PPP over ISDN) to ippp! Change this in /etc/isdn/isdnd.rc and possibly /etc/rc.conf and /etc/rc.d/isdnd.
/root/.profile	This gets overwritten. Copy it back from /root.old
crontabs	Those get deleted. Restore them by doing *cd /; gzip -dc /root.old/crontabs.backup.tar.gz | tar -xpvf -*
/usr/share/groff_font/devps/DE-SC	This fi le contains among other things the size of A4 paper and fontnames you added. If you're working with groff, you may want to restore this from a backup, as it gets overwritten too.
/usr/share/groff_font/de-vps/download	This fi les contains fontmappings. If you changed this, you will want to restore it also.
/etc/rc.d/swap1	With 1.5 having swap on RAID1 you needed to have "# KEYWORD: shutdown" in /etc/rc.d/swap1 for the swap device to be unconfi gured at shutdown time. If you didn't do this, the parity of the mirror was bad on startup. Now, you have an option "swapoff=yes" to put in /etc/rc.conf.

Step7: Reboot and test

Reboot and hope that your system comes up again :-)

Now you can clean up all the little oddities, that we missed.

The end

I hope, this helped you going. I have verifi ed the steps by upgrading several machines with this procedure, so I hope it doesn't contain severe bugs.
If you have ideas on how to improve this paper, please send it to me at ck at neverland dot ch.

This document has been created using vim, groff and ghostscript.