Remote Upgrading OpenBSD (historical)

http://www.openbsd.org/faq/upgrade51.html

[3.4 -> 3.5] [2] | [3.5 -> 3.6] [3] | [3.6 -> 3.7] [4] | [3.7 -> 3.8] [5] | [3.8 -> 3.9] [6] | [3.9 -> 4.0] [7] | [4.0 -> 4.1] [8] | [4.1 -> 4.2] [9] | [4.2 -> 4.3] [10] | [4.3 -> 4.4] [11] | [4.4 -> 4.5] [12] | [4.5 -> 4.6] [13] | [4.6 -> 4.7] [14] | [4.7 -> 4.8] [15] | [4.8 -> 4.9] [16] | [4.9 -> 5.0] [17] | [FAQ Index] [18]

Notes for remote upgrades, in place, of OpenBSD from distribution

Last updated 8 February 2008, Now the OpenBSD project adequately documents upgrading and should be your primary reference.

> **The following notes use as example of remotely upgrading to current versions from earlier versions on an i386 architecture. We started these notes before comprehensive upgrade notes became part of the standard OpenBSD documentation. You should review the official "Upgrade Guide" which is hyperlinked on the left of the** OpenBSD FAQ Index **[18] which now includes notes for "Upgrading without install media" which can be referenced for remote upgrading.**

The experience of upgrading one or two revisions in place is most of the time successful. An example of a good gotcha that caused an remote upgrade to fail years back was that one of the remote machines had a network card that was recognized as a "de0" for version 3.0 and after reboot the new kernel identified it as "dc0" ... so after the reboot it was no longer available remote. We had to drive to the remote location and use the console. After copying /etc/hostname.de0 to /etc/hostname.dc0 it still would not work. We may have found the fix by fooling around with ifconfig and how it did the media, but uptime was more important that being Sherlock Holmes so we replaced the network card with a different model that happened to recognize as dc0 and it worked. Lesson learned: remote upgrades are a risk.

- Use sudo where appropriate
- Read and familiarize yourself with the OpenBSD FAQ http://www.openbsd.org/faq/index.html [18]
- If feasible when doing remote administration it is really helpful to have the test machine that has the exact same hardware and setup to practice on. As mentioned above we got stung by the fact that the newer version tried to use a newer driver for the network card and failed.
- If you have made any changes to the usr land, for example if you need a custom kernel, recompiled sendmail to support smtp authentication, you enabled suexec with "chmod u+s

/usr/sbin/suexec" or the like better test the steps that may cause problems and document on a local machine before trying remote!

- Recommend you '**man release**' This man page discusses in some detail several steps involved with upgrading a release that may or may not apply.
- Recommend you review related documentation created by the author of the OpenBSD Upgrade FAQ. Other Links:
  http://www.openbsd.org/faq/upgrade-minifaq.html [19]
  http://pintday.org/hack/ [20]
  http://pintday.org/hack/docs/remote-upgrade-howto.shtml [21]
  And another remote upgrade faq: http://a.area51.dk/openbsd/upgrade [22]
- Recommend you use the latest version on CDROM [23], rather than a snapshot [24].
  We recommend upgrading to stable, but don't recommend upgrading to snapshot nor current unless you like to be on the edge and you are good at researching answers for yourself. If you like being on the edge, don't please do not ask questions of the mail lists [25] till you have exhausted google.com, the faqs etc.
- Be prepared and have available the files you will need for target distribution; source tree (/usr/src), Mergemaster package, and the generic /bsd or your custom kernel made on the test box ahead of time. It is helpful to have them on a the hard drive of the computer before you start and on cdrom in case the remote becomes an non-remote upgrade...
- Expect the typical issues to give you headaches like permissions and paths for wrappers changing, like smrsh for sendmail, if you modified apachectl for the -u flag will have to do it again, for apache suexec will have to enabled with "chmod u+s /usr/sbin/suexec", other headaches, and it helps to have access to www.google.com/bsd [26], www.undeadly.org [27] and other openbsd literature like searching the past mail lists [28] to hunt fixes with.
- Which version, rather update of src tree to use? It seems logical to use the exact same src tree that matches the distribution tar files that are used, that way mergemaster is proper configuration files from /usr/src/etc. To do this would mean using the src.tar.gz and srssys.tar.gz not using anoncvs to get the stable version till after upgrade to target version number. However we have jumped this step for machines not excessively remote and used anoncvs and updated /usr/src to stable prior to doing the upgrade rather than using the exact matching src to the distribution files and have not had significant issues arise.
- Familiarize yourself with the port mergemaster and it's man page.
- Do not forget that after using the distribution files, mergemaster and rebooting you still need to update the src, rebuild or use new stable kernel , redo usr land (make obj && make build), and redo mergmaster just in case configuration files changed significantly on the stable since the distribution files were created.
- We are not sure that it is a good idea but on the drive back on night from a remote location to make fixed from the console we thought that maybe if the machine has only one network card what might be useful is to have one /etc/hostname.anycard file and link all variations of /etc/hostname.xxx to it. If anyone does this, send us know your experience and maybe script or tar of the symbolic link files.
- Your procedure for handling migration of ports packages should also be tested before you try it on a 'must be up' machine or a remote machine.
- And oh by the way, this documentation does not address the issue of ports packages nor the X4 [29]. You should consider removing all unnecessary installed packages and perhaps all packages then reinstall necessary packages after upgrade.

Attempt to anticipate probable problems that may arise because of things like resource limitations and hardware compatibility.

Try to anticipate things that may cause problems and ways to work around them. If appropriate we would like to add your work around notes to these tips.

You may want to check hardware compatibility. Over all we would expect hardware that works for past versions to continue to be supported for new versions, but even so you may experience

issues like the driver designation changed.

Is you hard drive big enough? You may need to clean all the cruft out you can find and verify you partition sizes are adequate. Here is a link to some notes by a contributor for remote computers with small hard drives to minimize space normally taken up by unpacking src.tar.gz and by mergemaster:
http://www.cocoavillagepublishing.com/development/tools/ [30]
openbsd/tips/upgrading/notes_minimize_space_on_etcvar.html [30]

Open session(s) for remote administration.

You may want to create more than one ssh terminal session just in case on a computer with battery backup and a dependable remote connection. The following commands are examples run from the remote computer.

Backup anything important just to be safe.

We recommend you consider doing a comprehensive backup, however if you want to just copy the directories most likely to be edited by mergemaster then try backing up /etc, /var, /usr/local if you have stuff their and perhaps /usr/share (some sendmail stuff may reside there).
An example of backing up the folders with configuration information from the remote server, for the paranoid:

```
### The following four examples use 'pax' with cpio format so you can backup long path and file names.
### If you have SAMBA [31] in use, you may find really long file and path names created by windows
### software and tar may not be able to handle them.
### You may also have to set up ssh keys [32] so that using ssh to run pax on the remote server
### does not require passphrases during the session, which is a security problem that
### may be temporarily acceptable, else the pipe not work because of the authentication request.

### no compression on remote, compress afterwards on this computer.
ssh root@remotecomputer "pax -w -x cpio -b 5120 /etc /root /var /home /usr/local /usr/share" | gzip > /storage/remotecomputer.`date +%Y%m%d%H%M`.pax.gz

### no compression on remote, compress afterwards on this computer.
### with -s option to remove root /
ssh root@remotecomputer "pax -w -x cpio -b 5120 -s ',^//*,,' /root /var /home /usr/local /usr/share" | gzip > /storage/remotecomputer.`date +%Y%m%d%H%M`.pax.gz

### compression on remote with output redirected to be saved on this computer
ssh root@remotecomputer "pax -wz -x cpio -b 5120 /etc /root /var /home /usr/local /usr/share" > /storage/remotecomputer.`date +%Y%m%d%H%M`.pax.gz

### compression on remote with output redirected to be saved on this computer,
### with -s option to remove leading /
ssh root@remotecomputer "pax -wz -x cpio -b 5120 -s ',^//*,,' /etc /root /var /home /usr/local /usr/share" > /storage/remotecomputer.`date +%Y%m%d%H%M`.pax.gz
```

Prepare source on remote computer for upgrade.

To be safe it may be best to have the necessary distribution files, mergemaster package and kernel available on the remote machine.. You could mount a cdrom on the remote machine prior *(if you do put a bootable cdrom in the server, make sure the bios does not boot from it on restarts)* or copy them from the cdrom or across the network to a partitions with enough space to spare
(we will use /usr for our example).

```
###  Example using 4.0
mkdir -p /usr/4.0/`arch -s`
### /usr/4.0/`arch -s` is the same as /usr/4.0/i386 for most all computers
### with  intel compatible processors.
mkdir /usr/4.0/packages

### mount the distribution cdrom of the local computer and scp to the remote machine
### if you have it, or modify procedure appropriately.
ssh root@localcomputer mount -t cd9660 -r /dev/cd0a /mnt/cdrom
scp root@localcomputer:/mnt/cdrom/4.0/`arch -s`/*  /usr/4.0/i386/
scp root@localcomputer:/mnt/cdrom/4.0/src.tar.gz  /usr/4.0/
scp root@localcomputer:/mnt/cdrom/4.0/srcsys.tar.gz  /usr/4.0/
scp root@localcomputer:/mnt/cdrom/4.0/ports.tgz  /usr/4.0/

### or if you must be a gorilla, ftp like below
### but try using a mirror to not load primary openbsd ftp server.
# ftp -n ftp://ftp.openbsd.org/pub/OpenBSD/4.0/`arch -s`/*

### just to be safe because we don't want to untar etc**.tgz
### we will rename it for now, mergemaster will use src so it doesn't need this file.
mv /usr/4.0/i386/etc40.tgz  /usr/4.0/i386/etc40.donotusenow
```

Prepare ports on remote computer for upgrade.

You may also want to get all the ports packages you will need to replace available. Maybe checkout /usr/ports or have the precompiled binaries copies somewhere locally.

Mergemaster

You'll need mergemaster unless you are doing it by hand and it is a port package, if the machine your upgrading from is not to old you may can use that mergemaster but it seems logical to use a mergemaster version that is released with the distribution. Mergemaster is a Bourne shell script which is designed to aid you in up- dating the various configuration and other files associated with OpenBSD. It appears to have it's beginnings with FreeBSD but no homepage on the Internet. The man page and a test run on an unimportant machine should suffice for documentation and how-to for someone familiar with Unix system administration.

```
### example for version 4.3 w/ i386
###  if you do not already have the latest version of mergemaster installed:
cd /usr/4.3/i386/packages/
```

```
ftp ftp://ftp.openbsd.org/pub/OpenBSD/4.3/packages/i386/mergemaster-*.tgz
pkg_add /usr/4.0/packages/mergemaster-*.tgz
```

You may want to also get other packages you think you may need. You may want to use pkg_info to see what you are already using and try to have the equivalent packages and dependent packages already on hand for use after upgrading the operating system.

```
# scp root@localcomputer:/mnt/cdrom/4.3/packages/*  /usr/4.3/packages/
```

Stop, kill, unnecessary processes that could get confused by usr land and configuration files changing. Also un mount any unnecessary file systems.

Sendmail and named will most like be upset by in place upgrades.
Best to stop any processes you don't need running.

```
### look at processes and identify some you can kill like sendmail and named
ps -ax | more
### killing sendmail and named using info they stored on file
kill `head -1 /var/run/sendmail.pid`
kill `head -1 /var/named/named.pid`
### stop apache
/usr/sbin/apachectl stop
### df -k
### umount any unnecessary file systems in case of crash.
### others?
```

Prepare the source tree to be used later

You should prepare the source tree under /usr/src.
You can extract the src from the installation cdrom that has the src in two different files, src.tar.gz and srcsys.tar.gz.

```
### get the old version stuff out of src if untaring new src
rm -rf /usr/src/*
cd /usr/src

### IF you want stable or say a certain date of current you could check out src from an anoncvs server [33].
### For example if you are crazy enough to use a current snapshot, know the target date and
### need the source for that snapshot date you may could do something like this:
###      cvs checkout -D "2003-07-22 12:00" src
###
### Getting the new src from

 the installation cdrom by extracting src.tar.gz & srcsys.tar.gz
tar xvzpf  /usr/4.3/src.tar.gz
tar xvzpf  /usr/4.3/srcsys.tar.gz
```

Taking care of groups, users, permissions, ownerships, /dev and mtree after massive changes to usr land

You may or may not want to do these steps before replacing kernel and untarring the new binaries [34]. We have done it before and after with out major issues arising, yet.

Upgrading from previous versions frequently requires additional users, groups, permission changes, and ownership changes that can cause mergemaster to fail or mysterious problems. Sometimes mergemaster can do it automagically, but not always. This requires a human to figure out what is going on and lean heavily on the information in the OpenBSD FAQs.

The following indented notes are are in series so if you are upgrading by a jump of more than one release the notes may be used consecutively. If you are jumping more than one or two releases for your remote upgrade it would be safer to get another computer and do a fresh install then migrate the data accordingly, but that is not always feasible :(. The following notes are a guide and are not meant to be comprehensive. The notes may have missed something so you should read the OpenBSD upgrade-minfaq [35] carefully in addition to using them.

The original 'MINI-FAQ: Upgrading OpenBSD *[19]*' *by* Kjell Wooding *[36], was heavily referenced to create the following notes, however it has been replaced with official documentation as of version 3.5 as noted below. For following current see* http://www.openbsd.org/faq/current.html *[37]*

- From 2.9 to 3.0 [38]
- From 3.0 to 3.1 [39]
- From 3.1 to 3.2 [40]
- From 3.2 to 3.3 [41]
- From 3.3 to 3.4 [42]
- NOTE: STARTING version 3.5, the OpenBSD FAQ officially posted notes for upgrading, notes for versions hereafter are included as a reference, however we recommend you reference primarily the notes from OpenBSD FAQ pages.
- From 3.4 to 3.5 [43], see also the official openbsd website notes http://www.openbsd.org/faq/upgrade35.html [2]
- From 3.5 to 3.6. [44] see also the official openbsd website notes http://www.openbsd.org/faq/upgrade36.html [3].
- From 3.6 to 3.7 [45], see also the official openbsd website notes http://www.openbsd.org/faq/upgrade37.html [4].
- From 3.7 to 3.8 [46], see also the official openbsd website notes http://www.openbsd.org/faq/upgrade38.html [5].
- From 3.8 to 3.9 [47] see also the official OpenBSD website notes http://www.openbsd.org/faq/upgrade39.html [6]
- From 3.9 to 4.0 [48] see official OpenBSD website notes http://www.openbsd.org/faq/upgrade40.html [7]
- From 4.0 to.4.1 [49] see also the official OpenBSD website notes http://www.openbsd.org/faq/upgrade41.html [8]
- From 4.1 to 4.2 [50] see also the official OpenBSD website notes http://www.openbsd.org/faq/upgrade42.html [9]
- From 4.2 to 4.3(current) [51] see also the official OpenBSD website notes http://www.openbsd.org/faq/current.html [6]

Notes for changes from release 2.9 to 3.0 that may still need doing after extracting binaries and before mergemaster

```
####################
### from 2.9. to 3.0 ###
####################
###
### a new user and group named proxy were added to the system.
### To support this addition,
### add the following user entry using vipw(8):
#    proxy:*:71:71::0:0:Proxy Services:/nonexistent:/sbin/nologin
### Also add the proxy group to /etc/group:
#    proxy:*:71:
### Second, as part of the Sendmail 8.12 upgrade, sendmail no longer runs setuid
root.
### Both a new user and a new group, named smmsp, have been added to the
system.
### Add a line like the following to your /etc/group:
###       smmsp:*:25:
### Then, run vipw(8) and add the following line for the smmsp user:
#   smmsp:*:25:25::0:0:Sendmail Message Submission
Program:/nonexistent:/sbin/nologin
### Make sure this line appears before any yp(8) settings line.
### Finally, a new user and group were added for Solar Designer's popa3d server,
### now part of the core system. Add the following to /etc/group:
#   popa3d:*:26:
### And using vipw(8), add
#   popa3d:*:26:26::0:0:POP3 server:/var/empty:/sbin/nologin
```

Notes for changes from release 3.0 to 3.1 that may still need doing after extracting binaries and before mergemaster

```
####################
### from 3.0 to 3.1 ###
####################

### ssh config files moved to /etc/ssh/
mkdir /etc/ssh
### Move your /etc/ssh*_* files
cd /etc
mv ssh*_* ssh/
#### If you don't plan on replacing
#### the rc files when doing mergemaster
#### you may need to make sure the paths
#### is correct like: HostKey /etc/ssh/ssh_host_key
```

Notes for changes from release 3.1 to 3.2 that may still need doing after extracting binaries and before mergemaster

```
### Several new users/groups have been added.
### Add the following user entries using vipw(8) [52]:


  sshd:*:27:27::0:0:sshd privsep:/var/empty:/sbin/nologin  _portmap:*:2
8:28::0:0:portmap:/var/empty:/sbin/nologin  _identd:*:29:29::0:0:identd
:/var/empty:/sbin/nologin  _rstatd:*:30:30::0:0:rpc.rstatd:/var/empty:/
sbin/nologin  _rusersd:*:32:32::0:0:rpc.rusersd:/var/empty:/sbin/nologi
n  _fingerd:*:33:33::0:0:fingerd:/var/empty:/sbin/nologin  _x11:*:35:35
::0:0:X server:/var/empty:/sbin/nologin

### Add the following to /etc/group:

 sshd:*:27:
 _portmap:*:28:
 _identd:*:29:
 _rstatd:*:30:
 _rusersd:*:32:
 _fingerd:*:33:
 _sshagnt:*:34:
 _x11:*:35:
 authpf:*:72:

### New group for crontab(1) and at(1)
### The crontab(1) [53] and at(1) [54] commands are no longer setuid root,
### they are now setgid crontab.
### Add a line like the following to your /etc/group file:

 crontab:*:66:

### The permissions may or may not get corrected,
### Here are some commands by hand just in case
###  they are needed  (/bin/csh shell)

 chgrp crontab /var/at/at.{allow,deny} /var/cron/{allow,deny}
 chmod 0640 /var/at/at.{allow,deny} /var/cron/{allow,deny}
 foreach f ( /var/cron/tabs/* )
 set u=`basename $f`
   chown $u.crontab $f
 end

### Note that you probably will not have all of the allow/deny files;
### this is not expected to be a problem.
```

Notes for changes from release 3.2 to 3.3 that may still need doing after extracting binaries and before mergemaster

```
##################
#### from 3.2 to 3.3#
##################
```

```
### more users and groups needed for new features
groupadd -g 61 _lkm
groupadd -g 62 _spamd
groupadd -g 63 _radius
groupadd -g 64 _token
groupadd -g 65 _shadow
chgrp _lkm /dev/lkm
chgrp _radius /etc/raddb /etc/raddb/servers
chmod g+x /etc/raddb
chmod g+r /etc/raddb/servers
chgrp _token /etc/activ.db /etc/crypto.db /etc/snk.db
chmod 0640 /etc/activ.db /etc/crypto.db /etc/snk.db
chgrp _shadow /etc/spwd.db
chmod 0640 /etc/spwd.db

### as root add the user entry using vipw(8):
### _spamd:*:62:62::0:0:Spam daemon:/var/empty:/sbin/nologin

### may or may not need to remove unused libraries,
### libc_r and libnptherad replaced by libpthread
rm /usr/lib/libc_r* /usr/lib/libnpthread*

### may or may not need to do something about the fact that the
### contents of /var/at have been merged into /var/cron now
### that at has been integrated into cron
mv /var/at/* /var/cron
mv /var/cron/jobs /var/cron/atjobs
mv /var/cron/allow /var/cron/cron.allow
mv /var/cron/deny /var/cron/cron.deny
rm -rf /var/at
kill `cat /var/run/cron.pid`
/usr/sbin/cron

###  Also be forewarned, that now OpenBSD uses
###  version 9.x of BIND instead of 4.x and you the
###  human may need to read up on BIND if unfamiliar.
```

Notes for changes from release 3.3 to 3.4 that need to be done.

```
##################
#### from 3.3 to 3.4  - didn't work very well for us for those done
#### remotely because of the change to ELF [55],
#### in-place upgrades are not recommended.
####  Some enterprising admins have successfully worked around this
####  using a 'installboot' that understands elf [56].
####  Todd Miller posted a message:
####       "You can copy:
ftp://ftp.courtesan.com/pub/millert/OpenBSD/3.4-a.out/bsd.rd [57]
####       to your 3.3 machine's root filesystem and boot off it. It is an a.out 3.4
```

```
kernel
####       with the normal 3.4 ELF filesystem embedded in it. This is how I upgraded
a
####       machine remotely via serial console..."
####  See also http://undeadly.org/cgi?action=article&sid=20031028101040 [58]
####   BEST ADVICE: DON'T TRY IT, go to the machine, with a cdrom
####   and schedule downtime and do it by booting by cdrom!
############

### Following are commented out notes from the upgrade mini-faq,
###  From 3.3 to 3.4 the following will need to be done, and might or might not
###  be caught by Mergemaster.

### New user and group _syslogd (2003/07/31)
### The syslogd(8) daemon now runs in privilege separated mode,
### and requires a new user and group _syslogd. Add the group by running
 groupadd -g 73 _syslogd
### as root, and add the user entry using vipw(8):
         _syslogd:*:73:73::0:0:Syslog Daemon:/var/empty:/sbin/nologin

###  of note several files have been added that may not be yet mentioned,
###  like    /etc/pf.os   which is used for passive OS fingerprinting
###  using mergemaster helps catch theses
```

Notes for changes from release 3.4 to 3.5 that need to be done.

```
##################
### from 3.4 to 3.5
##################

#### new dev stuff
#### cp /usr/src/etc/etc.`machine`/MAKEDEV /dev
cd /dev
rm svnd* rsvnd*
./MAKEDEV vnd

### The pflogd(8) daemon now runs in privilege separated mode,
### and requires a new user and group _pflogd. Add the group by running

 groupadd -g 74 _pflogd

###   as root, and add the user entry using vipw(8):
###       _pflogd:*:74:74::0:0:pflogd privsep:/var/empty:/sbin/nologin

### Several pseudo network drivers (like gif(4), lo(4), and tun(4)) have been
### moved to support cloning.  A snapshot will likely overwrite so they don't
### have to be made, but if unsure, with new kernel can do theses steps:
# cd /usr/src && make includes
```

```
# cd sbin/ifconfig
# make obj depend
# make
# make install
# cp /usr/src/etc/netstart  /etc

### The new bgpd(8) daemon runs in privilege separated mode,
### and requires a new user and group _bgpd.
#Add the group by running
groupadd -g 75 _bgpd
# and as root add this entry using vipw
###    _bgpd:*:75:75::0:0:BGP Daemon:/var/empty:/sbin/nologin

### The join(1) command has been updated to follow the POSIX
### standard when writing non-matching lines. As a consequence
## /etc/security has to be updated.  Runing mergemaster should catch
## the change, but if you have the source you may want to:
# cp /usr/src/etc/security /etc

### The way the sudo(8) command is built has changed.
### A side-effect of this is that you may need to remove and old
### obj code if later doing a 'make build'
# rm -f /usr/obj/usr.bin/sudo/*

### new user noted during mergemasteruse....
groupadd -g 76 _tcpdump
###  use vipw to add
###    _tcpdump:*:76:76::0:0:tcpdump:/var/empty:/sbin/nologin

### ... this may not need to be done if replacing binaries rather than
### trying to rebuild, however it may need doing before rebuilding a kernel:
### machdep.c change (i386) (2004/02/01) A change to machdep.c requires that
###  binutils be rebuilt  before the kernel can be compiled. This only affects i386.
cd /usr/src/gnu/usr.bin/binutils
make -f Makefile.bsd-wrapper cleandir
make -f Makefile.bsd-wrapper obj
make -f Makefile.bsd-wrapper depend
make -f Makefile.bsd-wrapper
make -f Makefile.bsd-wrapper install
### Now you can rebuild the kernel and rest of the system

######  THIS IS ONLY FOR SPARC64 ############
### : Switch to gcc3 (sparc64) (2004/02/20)
### OpenBSD/sparc64 has converted to gcc3. To upgrade,
#
# cd /usr/src/share/mk
# make install
# rm -rf /usr/include/g++
### Now, build gcc3, (using the gcc3 directory)
###  then build the kernel and system as usual.
#################################

### Update bsd.own.mk (2004/02/20), suggested to update
### /usr/share/mk/bsd.own.mk to get the USE_GCC3 definition:
### (may not be necessary with binary install, may be...
```

```
cd /usr/src/share/mk
make install

########### not thought to be applicable for i386:
### Switch to gcc3 (sparc64) (2004/02/20)
###OpenBSD/sparc64 has converted to gcc3. To upgrade,
# rm -rf /usr/include/g++
###Now, build gcc3, (using the gcc3 directory) then build
###the kernel and system as usual.
```

Notes for changes from release 3.5 to 3.6 that need to be done.

```
################################
### from 3.5 to 3.6
################################

########################################
#### New user and group _dhcp (2004/04/12)
#### The various dhcp related daemons are being reworked
#### to drop privileges or to use privilege separation.
#### Thus a new user and group _dhcp are required.
### Add the user and group by running as root:
groupadd -g 77 _dhcp
useradd -u77 -g=uid -c"DHCP programs" -d/var/empty -s/sbin/nologin _dhcp

########################################
#### New user and group _mopd (2004/04/14)
#### The mopd(8) daemon now drops privileges after initialization,
#### and requires a new user and group _mopd.
###  Add the user and group by running as root
groupadd -g 78 _mopd
useradd -u78 -g=uid -c"MOP Daemon" -d/var/empty -s/sbin/nologin _mopd

########################################
#### New user and group _tftpd (2004/04/17)
#### The tftpd(8) daemon now drops privileges after initialization
#### to a new user and group _tftpd instead of nobody.
###   Add the guser and roup by running as root
groupadd -g 79 _tftpd
useradd -u79 -g=uid -c"TFTP Daemon" -d/var/empty -s/sbin/nologin _tftpd

### New user and group _rbootd (2004/04/30). The rbootd(8) daemon now drops
### privileges after initialization to a  new user and group _rbootd.
### Add the user and  group by running
groupadd -g 80 _rbootd
useradd -u80 -g=uid -c"rbootd Daemon" -d/var/empty -s/sbin/nologin _rbootd

### New user and group _afs (2004/05/06). The afsd(8) daemon is about to be
```

```
### privilege seperated. A new user and group _afs are required for this.
### Add the user and group by running
groupadd -g 81 _afs
useradd -u81 -g=uid -c"afs Daemon" -d/var/empty -s/sbin/nologin _afs

### New user and group _ppp (2004/05/06). The pppoe(8) program
### now drops privileges to a new user and group _ppp.
### Add the user and group by running
groupadd -g 82 _ppp
useradd -u82 -g=uid -c"PPP utilities" -d/var/empty -s/sbin/nologin _ppp

###New user and group _ntp (2004/05/30), The upcoming ntpd(8) program
###needs a new user and group _ntp. Add the user and group by running
groupadd -g 83 _ntp
useradd -u83 -g=uid -c"NTP Daemon" -d/var/empty -s/sbin/nologin _ntp

##########################################
#### pty device minor numbers changed (2004/04/11)
#### The device minor numbers of pty devices have changed
#### to increase the maximum number of pseudo-terminals
#### from 256 to 992, so you will need to run the updated
#### /dev/MAKEDEV before booting the new kernel:
###
### * Update your source code tree
### * Build and install a new kernel
### * Build the rest of the system (make build)
### * Copy the new MAKEDEV script to the /dev
###   directory and create the new pty devices.
cp /usr/src/etc/etc.`machine`/MAKEDEV /dev
cd /dev && ./MAKEDEV pty0

### * Boot the new kernel
### * Remove the old pty devices no longer in use:

cd /dev && rm -f [pt]ty[rq]*

### * Or, for the vax platform:
### cd /dev && rm -f [pt]tyr*

###  cksum moves (2004/05/02) The cksum(1) utility has moved from /usr/bin to
/bin.
### After you have completed a make build or extracted the new chksum,
### run the following as root:
  rm -f /usr/bin/cksum /usr/bin/sum

### Library bump flag day (2004/07/13)
####    This may cause you remote difficulties if you did not reboot with new
####     generic kernel.....
### Library and system interface changes have been made that require
### a bump of the major version number of every library.
### If you are extracting from you most likely will not have to do the following.
#* update your source code tree, * Build a new kernel,
#* Install the new kernel and reboot, * install new header files:
# cd /usr/src && make obj && make cleandir && make includes
#* build and install a new version of libc
```

```
## cd /usr/src/lib/libc && make depend && make && env NOMAN=1 make install
#* Build and install the system as usual
# cd /usr/src && make build

### Changes in pf anchors (2004/08/07)
### Users of authpf should change their anchor rule in the main ruleset from
#  anchor authpf
#  to
#  anchor "authpf/*"
```

Notes for changes from release 3.6 to 3.7

```
####################
### from 3.6 to 3.7  ###
####################
```

New users and groups

The FTP daemon has been reworked to drop privileges, and an OSPF daemon has been added, so new users and groups are required. As *root*, add the following users and groups, using useradd(8) [59]:

```
useradd -u84 -g=uid -c"FTP Daemon" \ -d/var/empty -s/sbin/nologin
 _ftp
```

```
useradd -u85 -g=uid -c"OSPF Daemon" \ -d/var/empty -s/sbin/nologi
n _ospfd
```

X.org replaces XFree86: Due the XFree86 project's change of license, OpenBSD has switched to the X.org project's [60] X Window System. The X11 configuration files must be updated, however.

> If you have an `/etc/X11/XF86Config` file that works on OpenBSD 3.6, renaming it to `/etc/X11/xorg.conf` will probably give you a fully functioning X Window system on OpenBSD 3.7. You may also find that the newer versions of X are much more "self-configuring" than earlier versions, and you may not need an `/etc/X11/xorg.conf` file at all now, depending on your hardware.

New directories also, which using mergemaster utility should catch with mtree invoked.

Notes for changes from release 3.7 to 3.8

```
####################
### from 3.7 to 3.8  ###
####################
```

Note that, with OpenBSD 3.8, the pkg tools now support in-place updating using pkg_add -r. This has been checked to work with most packages, in particular with the CD packages available in 3.6 or 3.7.
Some important points apply:

- pkg_add -r does not do global updates, but needs to be told which packages to update. A new option pkg_add -u can be used to find out the exact list of packages to pass to pkg_add -r.
- pkg_add -r -F update -F updatedepends -q list_of_new_pkgs should work in most cases.
- mozilla packages in 3.7 can now safely be updated to a 3.8 version.

Install new `/etc/firmware` files: Due to the fact that some uploaded "firmware" blobs were relocated from the kernel to files in the `/etc/firmware` directory, there are a few drivers which will break if there is no uploadable firmware file available when the new kernel boots. This will impact users of only a few devices, though all users can use this step without harm.
To extract the firmware files from `base38.tgz`, use the following as root:

```
cd /tar xzpf /path/base38.tgz "*etc/firmware/*"
```

New users and groups

An HostAP daemon has been added, so new users and groups are required.
As *root*, add the following users and groups, using useradd(8) [59]:

```
useradd -u86 -g=uid -c"HostAP Daemon" \ -d/var/empty -s/sbin/nolo
gin _hostapd
```

This step will add both the new user and its corresponding group.
Your environment may allow you to copy/paste those commands.

Userland Applications

`/etc` file changes (mergemaster package may be able to do all this later)

You will want to extract the `etc38.tgz` files to a temporary location:

```
cd /tmptar xzpf /path/etc38.tgz
```

Files that can probably be copied from `etc38.tgz` or from src "as is":

```
hostapd.confnetstartpf.osrcservicesmtree/*
```

Note that it IS possible to locally modify these files, if this has been done, manual merging will be needed. Here are copy/paste lines for copying these files, assuming you unpacked `etc38.tgz` in the above recommended place:

```
cd /tmp/etccp hostapd.conf netstart pf.os rc services /etccp mtre
e/* /etc/mtree/
```

Files that must be manually merged, respecting any local changes made to them, if they were modified from the default, otherwise, just copy them over, too:

```
ftpusersinetd.conflogin.confrc.confsysctl.confsyslog.confmail/ali
ases
```

The changes to these files are in this patch file [61].
NOTE mergemaster maybe able to handle all this...
You can attempt to use this by executing the following as root:

```
cd /patch -C -p0 <upgrade38.patch
```

This will test the patch to see how well it will apply to YOUR system, to actually apply it, leave off the "`-C`" option. Note that it is likely that if you have customized files or not kept them closely updated, or are upgrading from an snapshot of 3.7, they may not accept the patch cleanly. In those cases, you will need to manually apply the changes.
Please test this process before relying on it for a machine you can not easily get to.
The following files have had changes which should be looked at, but it is unlikely they should be directly copied or merged (i.e., if you are using pf.conf, look at the suggested change of strategy, and decide if it is appropriate for your use).

```
pf.confspamd.conf/root/.profile
```

New directories also, which using mergemaster utility should catch with mtree invoked.

Finally, use mtree(8) [62] to create any new directories:

```
mtree -qdef /etc/mtree/4.4BSD.dist -p / -u
```

Notes for changes from release 3.8 to 3.9

```
####################
### from 3.8 to 3.9  ###
####################

# We repeat you may do better to follow the official notes# on the Open
BSD website. The notes now include local# local booting from media and
remote
 upgrade not
es.#  http://www.openbsd
.org/faq/upgrade39.html[6]#  We repeat similar information here.

# For the /etc you will need to get some files from the new distribution,
# and it may be a good # idea to do so before doing the install because
# they may be needed by the kernel.
# Install new /etc/firmware files: Due to the fact that some
# uploaded "firmware" blobs were relocated from the kernel to files in
# the /etc/firmware directory, there are a few drivers which
# will break if there is no uploadable firmware file available
# when the new kernel boots. This will impact users of only a
# few devices, though all users can use this step without harm.
# To extract the firmware files from base39.tgz, use the following as root:


       cd /
       tar xzpf ${RELEASEPATH}/base39.tgz "*etc/firmware/*"


# Reboot on the new kernel before doing extracting # the upgrade tars:
# This might be a tempting step to skip, # but it should be done now, a
s usually, # the new kernel will run old userland apps



# New users and groups  * no new users or # groups  (that was unusual)

#### Operational changes

# * ftp-proxy ftp-proxy(8) was replaced by what was previously called pftpx.
#  The new ftp-proxy runs stand-alone and not from inetd.conf(5) as it used to.
#  You will have to update /etc/inetd.conf to no longer invoke ftp-proxy(8),
#  and update /etc/rc.conf and /etc/rc to run the new one.


       #  Edit rc.conf or rc.conf.local to invoke the new program, for example:
```

```
        echo 'ftpproxy_flags=""' >> /etc/rc.conf.local

        # The new proxy uses anchors to allow data connections,
        # which means that your existing /etc/pf.conf must be adapted.
        # In the NAT section you need:

        nat-anchor "ftp-proxy/*"
        rdr-anchor "ftp-proxy/*"

        #  They are mandatory, even if you don't use NAT otherwise.
        # The following rule, that is  probably already there for the
        # old ftp-proxy, must stay:

        rdr pass on $int_if proto tcp from $lan to any port 21 -> \
        127.0.0.1 port 8021

        # In the rules section, this is needed:

        anchor "ftp-proxy/*"

        # Rules that allow the proxy to make FTP control connections
        #(destination port 21/tcp) must stay. Rules that allow FTP data
        #connections are no longer needed. Those rules may contain
        #"user proxy" or "to port > 49151". Care has been taken to keep the
        #commandline switches similar, but some differ.
        #See the ftp-proxy man page.

        # One case warrants special mention: if you have old clients that rely
        #on active mode data connections which use 20/tcp as a source port,
        #you need the '-r' switch
        #(for this you had to run the old proxy with "-u root").

        # Run ftp-proxy with "-d -D7" if you run into trouble and
        # want to diagnose what's happening.


########   /etc file changes
########    If you use mergemaster it should catch these changes, but it
########    would not be a bad idea to do it manually afore.

# You will want to extract the etc39.tgz files to a temporary location:


        cd /tmp
        tar xzpf ${RELEASEPATH}/etc39.tgz


# Files that can probably be copied from etc39.tgz "as is":


        daily
        ipsec.conf
        magic
        monthly
        netstart
```

```
        rc
        security
        services
        weekly
        mtree/*


# Note that it IS possible to locally modify these files, if this has
# been done, manual merging will be needed. Here are copy/paste
# lines for copying these files, assuming you unpacked etc39.tgz
# in the above recommended place:


        cd /tmp/etc
        cp daily ipsec.conf magic monthly netstart rc security services weekly /etc
        cp mtree/* /etc/mtree/


# Files that must be manually merged, respecting any local changes
# made to them,  if they were modified from the default,
# otherwise, just copy them over, too:


        changelist
        inetd.conf
        lynx.cfg
        rc.conf
        ssh/ssh_config
        ssh/sshd_config
        sysctl.conf


# The changes to these files are in this patch file. You can attempt to
# use this by executing the following as root:


        cd /
        patch -C -p0 < upgrade39.patch


# This will test the patch to see how well it will apply to YOUR system,
# to actually apply it, leave off the "-C" # option. Note that it is likely that
# if you have customized files or not kept them closely updated, or are
# upgrading from a snapshot of 3.8, they may not  accept the patch cleanly.
# In those cases, you will need to manually apply the changes.
# Please test this process before relying on it for a machine you
# can not easily get to.

# The following files have had changes which should be looked at,
# but it is unlikely they should be directly copied or merged (i.e., if
# you are using pf.conf, look at the suggested change of strategy, and
# decide if it is appropriate for your use).


        hostapd.conf
```

```
        pf.conf
        spamd.conf


# Finally, use mtree(8) to create any new directories:


        mtree -qdef /etc/mtree/4.4BSD.dist -p / -u
```

Notes for changes from release 3.9 to 4.0

```
#########################
### from 3.9 to 4.0     ###
#########################

# We repeat you may do better to follow the official# notes at http://w
ww.openbsd.org/faq/upgrade40.html [7] # the notes are referenced as follows



#2006/03/27 - New iwi(4) firmware#iwi(4) now requires firmware version
3.0 to run. #Due to license problems the #firmware is not currently dis
tributed with OpenBSD. #The firmware can be installed via the pkg tools
 #with the package at  http://damien.bergamini.free.fr/iwifw/\OpenBSD/i
wi-firmware-3.0.tgz.#2006/05/18 - spppcontrol(8) removal#In-kernel PPP
configuration is now fully supported #by ifconfig(8). The obsolete sppp
control(8) utility #has been removed from the tree.#See the manual page
s sppp(4) and pppoe(4) #for configuration examples.#Previously, the /et
c/hostname.pppoe0 file # looked like this:##pppoedev ne0#!/sbin/ifconfi
g ne0 up#!/usr/sbin/spppcontrol \# $if myauthproto=pap myauthname=testc
aller \# myauthkey=donttell#!/sbin/ifconfig \$if inet 0.0.0.0 0.0.0.1 \
# netmask 0xffffffff#!/sbin/route add default 0.0.0.1#up##This should b
e updated according to the following example:##inet 0.0.0.0 255.255.255
.255 0.0.0.1 pppoedev ne0 \#authproto pap authname testcaller authkey d
onttell up#!/sbin/route add default 0.0.0.1##And the physical interface
 must be marked UP:### echo "up" > /etc/hostname.ne0#2006/05/26 - ipsec
adm(8) removal#IPsec configuration is now fully supported by ipsecctl(8
).#The obsolete ipsecadm(8) utility has been removed #from the tree. Se
e the ipsec.conf(5) manual page for #configuration examples.#2006/06/21
 - dvmrpd(8) added#A mrouted(8) replacement has been started # and impo
rted, in order to use # (note: not very useful yet)this new daemon #the
 user _dvmrpd must be added: useradd -u87 -g=uid -c"DVMRP Daemon" \   -
d/var/empty -s/sbin/nologin _dvmrpd#2006/06/25 - wicontrol(8) removal#W
ireless configuration for wi(4) is now fully supported #by ifconfig(8).
 The obsolete wicontrol(8) utility has #been removed from the tree.#200
6/07/27 - new compiler option#The kernel on all platforms is now built
with the # recently introduced # -Wstack-larger-than compiler # option.
 # Before building a kernel, the compiler needs # to be built and insta
lled.#Follow the instructions in FAQ 5 - # How do I bootstrap a newer v
ersion # of the compiler for details. # http://www.openbsd.org/faq/faq5
```

.html#NewCompiler [63]
--- this may not apply if you booted # using a cdrom w/ snapshot# and are doing an upgrade, but it does sound # like trouble for a remote upgrade

Notes for changes from release 4.0 to 4.1

##########################
from 4.0 to 4.1
##########################
QUOTED FROM http://www.openbsd.org/faq/upgrade41.html [8]

- **Install new /etc/firmware files:** Due to the fact that some uploaded "firmware" files may have been updated, you may need to update the files in the /etc/firmware directory. This will impact users of only a few devices, though all users can use this step without harm. To extract the firmware files from base41.tgz, The firmware files hopefully will extract when extractng:

  base41.tgz ./etc/firmware

- Note: the files in /etc may best be manually merged or with mergemaster, so etc41.tgz and xetc41.tgz should NOT be unpacked over existing.

Note: During this process, sendmail(8) [64] may produce some error messages like the following:

Nov 1 12:47:05 puffy sm-mta[16733]: filesys_update failed: No such file or dire ctory, fs=., avail=-1, blocksize=380204

These messages can be safely ignored, or you may wish to halt sendmail(8) during the upgrade process.

New users and groups

- A RIP daemon and a host state daemon have been imported, and in order to use these new daemons, new users and groups are required. As *root*, add the following users and groups, using useradd(8) [59]:

  useradd -u88 -g=uid -c"RIP Daemon" -d/var/empty -s/sbin/nologin _ripd
  useradd -u89 -g=uid -c"HostState Daemon" \-d/var/empty -s/sbin/nologin _hoststated

  These steps will add both the new users and their corresponding groups. Your environment may allow you to copy/paste those commands.

Operational changes

This is *not* a complete list of the changes that took place between 4.0 and 4.1, but rather some of the important things that will impact a large number of users in the upgrade process. For a more complete list of changes, see plus41.html [65] and the CVS change logs.

- **"flags S/SA keep state" implicit in pf.conf(5)**
  flags S/SA keep state is now the default for pass rules in pf.conf(5) [66], and new *no state* and *flags any* options have been added to override these defaults. Current rulesets will continue to load, but the behaviour may be slightly changed as these defaults are more restrictive. Rulesets with stateless filtering (*no state*) or a requirement to create states on intermediate packets (*flags any*) should be updated to explicitly request the desired behaviour.
  In particular care should be taken with the enc0 [67] interface, as floating states are a potential problem for filtering IPsec traffic: states need to be interface bound, to avoid permitting unencrypted traffic should isakmpd(8) [68] exit. Therefore all rules on the enc0 interface should explicitly set *keep state (if-bound)*

- **FreeBSD and NetBSD DOS MBR partitions are no longer used to store disklabels, and are no longer searched for a disklabel**
  As a result, disks with only FreeBSD or NetBSD DOS MBR partitions will no longer have an 'a' partition created in the spoofed disklabel. The disklabel on such disks will be stored and loaded from sector 1, and this could find an old accidental disklabel. Use fdisk(8) to change the partition type to OpenBSD ('A6') to use the existing disklabel.

- **amd64, i386, macppc, and mvmeppc no longer try to boot from the first NetBSD partition when no OpenBSD partition is found.**
  Again, use fdisk(8) [69] to change the partition type to OpenBSD ('A6').

- **New way to enable/disable process accounting on boot**
  In order to enable accounting on boot you need to set the new rc.conf variable "accounting=YES". The presence of the /var/account/acct file does not make it start automatically anymore.

- **spamd(8) now runs in greylist mode by default**
  Whereas before greylisting had to be enabled via the `-g` flag, it is now the default runtime mode. The `-b` flag (which used to indicate which IP spamd(8) should listen on) can be used to run spamd in blacklist-only mode. The new `-l` flag is used to indicate which IP spamd should listen on.

- **spamd.conf(5) now stored in /etc/mail**
  The spamd.conf(5) file which used to be stored in `/etc` has been moved to `/etc/mail`. Move your spamd.conf file into `/etc/mail` so that spamd(8) loads the proper configuration.

- **Kernel sensors sysctl(3) MIB is now hierarchical**
  The sensors sysctl(3) MIB now contains the name of the sensor device as well as the type and index of the actual sensor. So before when running "sysctl hw.sensors" you might've seen "`hw.sensors.12=it0, Temp 1, 25.00 degC`" you will now see "`hw.sensors.it0.temp0=25.00 degC`". Update your sensorsd.conf(5) [70] file to reflect the new names of your sensors.

Upgrading packages

If you installed any packages on your system, you should upgrade them after completing the upgrade of the base system. Be aware, however, many packages will require further setup before and/or after upgrading the package. Check with the application's upgrade guide for details.
The following package is known to have significant upgrade issues that will impact a large number of users. The fact that a package is not on this list doesn't mean it will have a trivial upgrade. You must do some homework on the applications YOU use.

- **Postgresql:** you must dump your database before upgrade and restore it after

The package tools support in-place updating using `pkg_add -u`. For instance, to update all your packages, make sure `PKG_PATH` is pointing to the 4.1 packages directory on your CD or nearest FTP mirror, and use something like

```
# pkg_add -ui -F update -F updatedepends
```

where the `-u` indicates update mode, and `-i` specifies interactive mode, so pkg_add will prompt you for input when it encounters some ambiguity. Read the pkg_add(1) [71] manual page and the package management [72] chapter of the FAQ for more information.

Notes for changes from release 4.1 to 4.2

```
####################
### from 4.1 to 4.2  ###
####################
```

Most of this information is quotedfrom http://www.openbsd.org/faq/upgrade42.html [9]

- **Packages are now using libexpat shipped with X; xbase42.tgz is your friend:**
  The libexpat port has been removed. Packages now use the libexpat which is shipped with X instead. Many packages depend on libexpat through other dependencies. For instance, many packages require gettext which depends on libexpat. Therefore, these packages also depend on libexpat.
  This means that many systems that weren't using X before will now need to

have `xbase42.tgz` installed. If you don't do this and try to install a package that requires libexpat, pkg_add(1) will give an error message.

Also note that building ports is only supported with a full installation, including *all* X file sets.

Finally, after upgrading all your packages to the 4.2 versions, clean up by removing the old expat package from your system:

```
# pkg_delete expat
```

This will impact a large number of users! This was an unfortunate decision whose ramifications were not recognized earlier in the process. For 4.3, libexpat will be part of `base43.tgz`, solving this problem.

- **Updated** ahci **[73] driver may change how disks are handled:**
  Systems using the ahci(4) driver may find SATA disks which had been recognized and handled by the wd(4) [74] (i.e., wd0) to become sd(4) [75] (i.e., sd0) devices. Watch the `dmesg` output at boot. A drive which turned up earlier as:
  ```
  pciide1 at pci0 dev 31 function 2 "Intel 82801GBM AHCI
  SATA" rev 0x02: DMA, channel 0 wired to native-PCI,
  channel 1 wired to native-PCI
  pciide1: using apic 2 int 11 (irq 11) for native-PCI interrupt
  wd0 at pciide1 channel 0 drive 0: <FUJITSU MHV2080BH>
  wd0: 16-sector PIO, LBA48, 76319MB, 156301488 sectors
  wd0(pciide1:0:0): using PIO mode 4, Ultra-DMA mode 5
  ```
  now instead turns up as:
  ```
  ahci0 at pci0 dev 31 function 2 "Intel 82801GBM
  AHCI SATA" rev 0x02: AHCI 1.1: apic 2 int 16 (irq 11)
  scsibus1 at ahci0: 32 targets
  sd0 at scsibus1 targ 0 lun 0: <ATA, FUJITSU MHV2080B,
  0084> SCSI2 0/direct fixed
  sd0: 76319MB, 76319 cyl, 64 head, 32 sec, 512 bytes/sec,
  156301488 sec total
  ```
  This will cause problems for people doing remote upgrades of these machines, as if the fstab file is not "correct", the system will not complete booting. Unfortunately, how the driver handles the disk is dependent upon a lot of things including BIOS configurations, so IF one has an AHCI SATA interface, one will have to experiment with a similarly configured local machine to see if the `/etc/fstab` file will need to be re-worked.
  It is believed very few people will be hit by this now, as ahci(4) devices are moderately rare on existing hardware, though becoming much more common now.

- **bgpd filter language change:**
  bgpd filters using only `prefixlen` as filter parameter now need to include an address family identifier like `inet` or `inet6`:

```
allow from any inet prefixlen 8 - 24
deny from any inet6 prefixlen > 64
```

As a reminder, bgplg and associated binaries are disabled at install/upgrade time. If you use them, they must be re-enabled as described in bgplg(8) [76].

SSH protocol 1 discouraged on new installs:
New OpenBSD installs will default to not accepting SSH v1 connections. This upgrade process will not replace the `/etc/ssh` configuration files, but you may wish to manually do this to gain the new behavior. The following diff shows what has changed on new installs:

```
--- ./etc/ssh/sshd_config       Sat Mar 10 20:31:32 2007+++ ../42
/etc/ssh/sshd_config   Tue Aug 28 11:59:52 2007@@ -11,3 +11,2 @@
#Port 22-#Protocol 2,1 #AddressFamily any@@ -15,2 +14,7 @@   #Lis
tenAddress :: ++# Disable legacy (protocol version 1) support in
the server for new+# installations. In future the default will ch
ange to require explicit+# activation of protocol 1+Protocol 2
```

- Again, this change is NOT part of the standard upgrade process.
- **Changes in the way sudo(8) passes environment:**
  For security reasons [77] sudo(8) [78] will now reset the environment to a small default set with only certain variables preserved from the previous environment.

  In order for many things to continue to work as expected, the patch file below [79] will add a "`Defaults env_keep`" line to your `/etc/sudoers` file and otherwise try to make the file look like the one in `etc42.tgz`, but will possibly fail. You will want to make sure your `sudoers` file contains a line that looks something like:

  ```
  %wheel        ALL=(ALL)
  SETENV: ALL   -- or --
  %wheel        ALL=(ALL) NOPASSWD: SETENV: ALL
  ```

  assuming you wish "wheel" group users to have full sudo rights. It would probably be wise to test sudo(8) for proper operation before logging out of the system after the patch file is applied.

- **[alpha Only] Some de(4) NICs will become dc(4):**
  On the alpha [80] platform, some NICs which had been supported by the de(4) [81] driver will now be supported by dc(4) [82].
  IF your NIC is one of these, you will need to alter at least your `/etc/hostname.deX` (hint: hard link) and your pf.conf files as appropriate.

Again, this is *only* on the alpha platform.

- **[i386 Only] apm(4) takes precedence over acpi(4)**
  The device detection on i386 has been modified to make apm(4) [83] take precedence over acpi(4) [84]. It means that if your hardware has both apm and acpi devices, only apm will attach. If for some reason you prefer to use acpi, disable `apm` on your acpi-enabled kernel using `config(8)` or `boot -c`.

- **rc.conf:**
  Unlike earlier versions of this process, since 4.1 [8] it is assumed that `/etc/rc.conf` is not a user-altered file. If you have made changes to your `/etc/rc.conf` file, merge those changes into `/etc/rc.conf.local`. If you have NO `/etc/rc.conf.local`, simply copy your existing `/etc/rc.conf` file to `/etc/rc.conf.local` and *delete the last line of the script!* Otherwise, pull your existing `rc.conf` into the top of your existing `rc.conf.local` file *and remove the last line* before doing the rest of this process.

- **Modified kernel:**
  Check whether you have made any modifications to your kernel. For example, you might have modified your network device to use a non-default setting using config(8). Note your changes, so you can repeat them for the new 4.2 kernel.

Upgrading without install kernel

This is NOT the recommended process. Use the install kernel method if at all possible!
Sometimes, one needs to do an upgrade of a machine when one can't easily use the normal upgrade process. The most common case is when the machine is in a remote location and you don't have easy access to the system console. One can usually do this by carefully following this process:

- **Place install files in a "good" location**. Make sure you have sufficient space!

- **Stop any "insecure" applications from starting at boot:** There will be a time when PF will be unlikely to be running during this upgrade process, but your applications may still start and run properly. Any application dependent upon PF for security should be disabled before this happens, and should not be re-enabled until proper PF operation is verified after upgrade. There may be other applications which you wish to keep from running during the upgrade, stop and disable them as well.

- **Check the kernel:** Although **most people can skip this step**, if you had a modified kernel in 4.1, it is likely you will need to modify the stock kernel of 4.2. Especially when you are performing the upgrade process remotely, now is the time to make sure the new kernel will work upon rebooting the

machine. If any changes must be made to the kernel, the safest thing to do is to make those changes on a local 4.2 system. This can be as simple as modifying a specific device using config(8), or it can involve a recompilation if the option you need is not included in the GENERIC kernel. Please consult FAQ 5 - Building the system from source [85] before considering to recompile your kernel.

- **Install new kernel(s)**

```
export RELEASEPATH=/usr/rel
# where you put the filescd ${RELEASEPATH}
rm
 /ob
sd ; ln
/bsd /obsd && cp
 bsd /nbsd && mv /nbsd /bsdcp bsd.rd bsd.mp /
```

Note the extra steps for copying over the primary kernel: those are done to ensure that there is always a valid copy of the kernel on the disk that the system can boot from should there be a really badly timed power outage or system crash.

- **Install new `/etc/firmware` files:** Due to the fact that some uploaded "firmware" files may have been updated, you may need to update the files in the `/etc/firmware` directory. This will impact users of only a few devices, though all users can use this step without harm. To extract the firmware files from `base42.tgz`, use the following as root:

```
cd /
tar -C / -xzphf ${RELEASEPATH}/base42.tgz ./etc/firmware
```

- **Reboot on the new kernel:** This might be a tempting step to skip, but it should be done now, as usually, the new kernel will run old userland apps (such as the soon to be important `reboot`!), but often a new userland will NOT work on the old kernel.

- **Install new userland applications.** *Do NOT install `etc42.tgz` and `xetc42.tgz` now, because that will overwrite your current configuration files!*

```
export RELEASEPATH=/usr/rel
cd ${RELEASEPATH}
tar -C / -xzphf base42.tgz
tar -C / -xzphf comp42.tgz
tar -C / -xzphf game42.tgz
tar -C / -xzphf man42.tgz
tar -C / -xzphf misc42.tgz
```

```
tar -C / -xzphf xbase42.tgz
tar -C / -xzphf xfont42.tgz
tar -C / -xzphf xserv42.tgztar -C / -xzphf xshare42.tgz
```

Note: not all file sets will need to be installed for all applications, however if you installed a file set originally, you should certainly upgrade it with the new file set now.
Note: the files in `/etc` are handled separately below, so `etc42.tgz` and `xetc42.tgz` are NOT unpacked here.

- **Upgrade `/dev`.** The new MAKEDEV [86] file will be copied to /dev by the installation of `base42.tgz`, so you simply need to do the following:

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

- **Upgrade `/etc` as below**.

- **Reboot**

During this process, sendmail(8) [64] may produce some error messages like the following:

```
    Nov 1 12:47:05 puffy sm-mta[16733]: filesys_update failed: No such
file or directory, fs=., avail=-1, blocksize=380204
```

These messages can be safely ignored for the moment, or you may wish to halt sendmail(8) during the upgrade process. Note that sendmail is not working properly at this point, and will need to be restarted (as part of the reboot) before mail is expected to be handled properly.

Final steps

Whether you upgrade by using an install kernel and doing a formal "upgrade" process, or do a "in-place" binary upgrade, there are certain manual steps that have to be performed.

1. Upgrading `/etc`

You will want to extract the `etc42.tgz` files to a temporary location:

```
tar -C /tmp -xzphf ${RELEASEPATH}/etc42.tgz
```

Files that can probably be copied from `etc42.tgz` "as is":

```
etc/magicetc/man.confetc/netstartetc/rcetc/rc.confetc/rpcetc/serv
icesetc/mail/helpfileetc/mail/localhost.cfetc/mail/sendmail.cfetc
/mail/submit.cfetc/mtree/4.4BSD.distetc/mtree/BSD.local.distetc/m
tree/special
```

Note that it IS possible to locally modify these files, if this has been done, manual merging will be needed. Pay special attention to `mail/*` if you are using something other than the default Sendmail(8) configuration. Here are copy/paste lines for copying these files, assuming you unpacked `etc42.tgz` in the above recommended place:

```
cd /tmp/etccp magic man.conf netstart rc rc.conf rpc services /et
ccp mtree/* /etc/mtree/cp mail/helpfile mail/localhost.cf mail/su
bmit.cf /etc/m
ailcp mail/sendmail.cf /etc/mail     # Careful on this one!!
```

Files that must be manually merged, respecting any local changes made to them, if they were modified from the default, otherwise, just copy them over, too:

```
etc/ntpd.confetc/sensorsd.confetc/ssh/ssh_configetc/ssl/x509v3.cn
fetc/sudoersetc/sysctl.confetc/wsconsctl.confvar/www/conf/httpd.c
onf
```

The changes to these files are in this patch file [87]. You can attempt to use this by executing the following as root:

```
cd /patch -C -p0 < upgrade42.patch
```

This will test the patch to see how well it will apply to YOUR system, to actually apply it, leave off the "`-C`" option. Note that it is likely that if you have customized files or not kept them closely updated, or are upgrading from a snapshot of 3.9, they may not accept the patch cleanly. In those cases, you will need to manually apply the changes. Please test this process before relying on it for a machine you can not easily get to.
The following files have had changes which should be looked at, but it is unlikely they should be directly copied or merged (i.e., if you are using bgpd.conf, look at the suggested change of strategy, and decide if it is appropriate for your use).

```
etc/bgpd.confetc/mail/spamd.confetc/ospfd.confetc/ssh/sshd_config
```

Finally, use newaliases(8) [88] to update the aliases database and mtree(8) [62] to create any new directories:

```
newaliasesmtree -qdef /etc/mtree/4.4BSD.dist -p / -u
```

2. Checking the kernel

Note: **most people can skip this step!**
If you followed the instructions for the upgrade process without install kernel, you have already completed this step. However, if you used the install kernel, and if you had a modified kernel in 4.1, it is likely you will need to modify the stock kernel of 4.2. This can be as simple as modifying a specific device using config(8), or it can involve a recompilation if the option you need is not included in the GENERIC kernel. Please consult FAQ 5 - Building the system from source [85] before considering to recompile your kernel.

3. X configuration files

Due to the significant changes in X for this release, the easiest way to upgrade your X for 4.2 may be to back up your existing X configuration files, unpack xetc42.tgz, and manually merge back any changes you had made.
The files you are most likely to want to save a copy of are `/etc/X11/xorg.conf` and `/etc/X11/xinit/xinitrc`. As X now often works with NO xorg.conf file, you may wish to try it without one before you copy your file back.

Unpack xetc42.tgz as you would other file sets:

```
export RELEASEPATH=/usr/relcd ${RELEASEPATH}tar -C / -xzphf xetc4
2.tgz
```

4. Upgrading packages

If you installed any packages on your system, you should upgrade them after completing the upgrade of the base system. Be aware, however, many packages will require further setup before and/or after upgrading the package. Check with the application's upgrade guide for details.
The following package is known to have significant upgrade issues that will impact a large number of users. The fact that a package is not on this list doesn't mean it will have a trivial upgrade. You must do some homework on the applications YOU use.

- **xfce:** Configuration base directory and files have completely changed. You will probably have to re-create your configuration from scratch.

Before continuing, there are some major changes in the 4.2 release which you should know about:

- The libexpat port has been removed, and packages are now using the libexpat shipped with X11 instead. Many packages depend on libexpat through other dependencies. For instance, all the packages which require gettext also depend on libexpat. This means that before you can add or

upgrade packages, you most likely need to have `xbase42.tgz` installed, even if the packages you are installing don't have a graphical operation. Refer to the FAQ [89] about adding `xbase42.tgz` to an installed system. If you don't do this and try to install a package that requires libexpat, pkg_add(1) will give an error message.
- If you are using pkg_add(1) in combination with sudo(8), you will need to change your `/etc/sudoers` as the patch file [79] attempts to. See changes in sudo(8) [90] for more info.

The package tools support in-place updating using `pkg_add -u`. For instance, to update all your packages, make sure `PKG_PATH` is pointing to the 4.2 packages directory on your CD or nearest FTP mirror, and use something like

```
# pkg_add -ui -F update -F updatedepends
```

where the `-u` indicates update mode, and `-i` specifies interactive mode, so pkg_add will prompt you for input when it encounters some ambiguity. Read the pkg_add(1) [91] manual page and the package management [72] chapter of the FAQ for more information.
You will very possibly see something like this when running the above command:

```
Looking for updates: completeCannot find updates for expat-2.0.0P
roceed? [y/N]
```

This is indicating you have run into the libexpat problem and must now install `xbase42.tgz` as mentioned above. If you have not installed xbase42.tgz, it is recommended that you stop the package update, install xbase42.tgz and then re-run the package upgrade.
Finally, after upgrading all your packages, clean up by removing the old expat package from your system:

```
# pkg_delete expat
```

Notes for changes from release 4.2 to 4.3

```
#########################
### from 4.2 to 4.3 beta  ###
#########################
```

2007/09/03 - _cxa_atexit support and sys/net headers flag day

Support for _cxa_atexit has been added to csu. Using the new kernel is imperiative and a new "csu" which hopefully will unfold from the tar

2007/10/20 - libexpat moved to base

libexpat has been moved from xenocara (xbase) to src (base); if you are using binary snapshots and packages, things should just update during extraction...

2007/10/08 - new user for ospf6d

A ospf6d daemon has been imported in order to use this daemon, a new user and group are required. As *root*, add the following user and group, using useradd(8) [59]:

```
# useradd -u90 -g=uid -c"OSPF6 Daemon" -d/var/empty -s/sbin/nologin _os
pf6d
```

This step will add both the new user and its corresponding group.

2007/11/28 - new AGP driver

The new agp(4) [92] driver requires an update to the X server after installing a new kernel if your machine requires AGP support for X or XVideo. A new upgrade of X hopefully will extract what you need, if not:

```
 # cd /usr/src/include # make includes # cd /usr/src/xenocara/xserver #
 make -f Makefile.bsd-wrapper build # cd /usr/src/etc/etc.`uname -m` #
install -c MAKEDEV /dev # cd /dev # ./MAKEDEV agp0
```

2007/12/07 - _hoststated user gets renamed to _relayd

The hoststated(8) daemon has been renamed to relayd(8) [93], so the user and group need to be changed accordingly.
Use vipw(8) [52], edit the password files and your favorite editor to edit the `/etc/group` file, changing "`_hoststated`" to "`_relayd`" in both. Leave the rest of the line alone in each case.

Notes for changes from release 4.3 to 4.4

```
#########################
### from 4.3 to 4.4. beta  ###
#########################
```
not available as of this documentation.

PORTS: Before upgrading, some users choose to remove all packages [94], and installing new versions after upgrade. You may want to rename the old /usr/ports to something like /usr/ports-old just in case you need it to roll back.

To quickly remove all packages from your system:

```
pkg_delete -q /var/db/pkg/*
```

After the upgrade, install the new versions of these applications.

Note that, with OpenBSD 3.8, the pkg tools now support in-place updating using pkg_add -r. This has been checked to work with most packages, in particular with the CD packages available in 3.6 or 3.7. Some important points apply:

- pkg_add -r does not do global updates, but needs to be told which packages to update. A new option pkg_add -u can be used to find out the exact list of packages to pass to pkg_add -r.
- pkg_add -r -F update -F updatedepends -q list_of_new_pkgs should work in most cases.
- mozilla packages in 3.7 can now safely be updated to a 3.8 version.

OpenBSD 3.9

If you installed any packages on your system, you may want to update them after completing the update of the base system. In OpenBSD 3.9, the pkg tools now support in-place updating using `pkg_add -u`. This has been checked to work with most packages, in particular with the CD packages available in 3.8. For instance, to update all your packages, make sure `PKG_PATH` is pointing to the 3.9 packages directory on your nearest CD or FTP mirror, and use something like

```
# pkg_add -ui -F update -F updatedepends
```

where the `-u` indicates update mode, and `-i` specifies interactive mode, so pkg_add will prompt you for input when it encounters some ambiguity. Read the pkg_add(1) [95] manual page and the package management [72] chapter of the FAQ for more information.

DEVICES: As a general rule, it is a good idea to copy and run the MAKEDEV script from your source tree when performing an upgrade. Mergemaster might handle this okay, but if unsure you may want to do yourself. Do note however when you remake the devices it is probable you won't be able to ssh back in again till reboot with matching usr land and kernel and /dev so it doesn't hurt to have an extra terminal open before doing this.

```
cd /dev
cp /usr/src/etc/etc.i386/MAKEDEV ./
./MAKEDEV all
```

DIRECTORIES: From time to time, files or directories are added to, or removed from the file hierarchy. Also, ownership information for portions of the filesystem may changes. Mergemaster might handle this okay, but if unsure you may want to do yourself. An easy way to ensure that your file hierarchy is up-to-date is to use the mtree(8) utility. First, fetch the latest source, then do the following:

```
cd /usr/src/etc/mtree
install -c -o root -g wheel -m 600 special /etc/mtree

### note, if users or groups like auth need added the next command will fail
mtree -qdef /etc/mtree/4.4BSD.dist -p / -u
```

Hopefully now your file hierarchy should now be up to date.

Replace the kernel

Save existing kernel and copy the new replacement kernel into place. If you have a custom kernel made on a similar machine already upgraded, change the procedure accordingly. After this step, you will be committed if you are remote. Also of note some unix administrators may do this much earlier before previous steps, some will reboot after this point, others wait till later to reboot and use the new kernel. This will be a decision to make by situation, because sometimes the old user land does not like the new kernel and sometimes the new userland being untarred does not like the old kernel. Rebooting to late or to early may require a on site reboot and maybe a console upgrade from media. No one promised you a rose garden! Of note, for 3.9 to 4.0 we rebooted after untarring the new files in the next step.

```
cp -p /bsd /bsd.old
cp /usr/4.0/i386/bsd  /bsd
chown root.wheel /bsd
```

Prepare and untar the new files

Don't untar etc**.tgz! You should at least untar the base**.tgz, comp**.tgz, and man**tgz. We think it is wise, space permitting, to untar the xserver related files to avoid potential issues when doing make build from source. (We have noticed that sometimes if you don't install x related distribution you may get errors when doing make build from soure).

Now to extract the binaries, you can use a for loop (but make sure etc does not have tgz extension!) or untar each manually . Note this probably will not work jumping from 3.3 to 3.4 for intel platforms. ALSO NOTE that going from 3.5 to 3.6/3.7 probably because of the Library bump part of the way through the untar lost control of computer and had to have the remote machine unplugged and powered back up causing a messy fsck. So you may want to try to reboot with generic kernel before untarring, but that to may be an issue. Good luck...

```
cd /usr/4.0/i386/; for i in *.tgz; do tar xvzpf $i -C /; done
```

(Of note, for 3.9 to 4.0 we remotely rebooted, reboot, here after untarring the new files)

Editing the configuration files; mergemaster is your buddy

```
As of OpenBSD 4.4,  A new tool sysmerge(1), derived from the old mergemaster port
, makes
      it easier to merge configuration files changes during an upgrade.
```

Hopefully now we can use mergemaster package to updage configuration files. else we hand do the changes as mentioned in the "Long Answer" section of the http://www.openbsd.org/faq/upgrade-minifaq.html#1.9 [96] Probably can lean on info in the upgrade minfaq to help decide on some of the merging. And we can use pkg_info look at where mergemaster was put because it's probably not in your path. Some of the information in the Upgrading OpenBSD FAQ [19] may be very helpful when having to decide on what parts to merge, replace or leave alone. BE VERY CAREFUL - THINK BEFORE YOU REPLACE, MERGE, OR NOT REPLACE. A mistake here can show up when you reboot and a remote upgrade could turn into a local console upgrade or even install if fubar [97]. Oh and if you are do some line editing with mergemaster you might want to refresh your "ed [98]" editor skills.

By the way, before running mergemaster you may want to replace some of the documents for apache so you don't have to answer a hundred or requests if okay to replace or merge, example:

```
### Copying the whole manual also copies a lot of unecessary pages with other languages.
### cp -Rp /usr/src/usr.sbin/httpd/htdocs/manual/*  /var/www/htdocs/manual/*
### so we can do this
cp /var/www/conf/httpd.conf /var/www/conf/httpd.conf.backup
cd /usr/src/usr.sbin/httpd; make -f Makefile.bsd-wrapper distribution

### install and run mergemaster
pkg_info -L mergemaster-*
/usr/local/sbin/mergemaster
```

Worry, Reboot, & Pray

A few last things to consider before rebooting....

- If you are paranoid or very far away from the remoter server, you may want to check /etc/rc.conf and /etc/rc.local one more time with an editor.
- If you must use httpd with the -u flag, you may want to edit /usr/sbin/apachectl, because it got replaced.
  ```
  # the path to your httpd binary, including options if necessary
  HTTPD="/usr/sbin/httpd -u "
  ```
- If you are using suexec, you'll want to enable it because it also got replaced

```
chmod u+s /usr/sbin/suexec
```

- If you manually copied some libraries or other includes from within /usr/src, you better check and do manually again if needed for remaking ports perhaps. For example, if you are using the libmilter feature of sendmail *(can set in /etc/mk.conf)* and an port of clamav [99] with milter flavor, the port will need mfapi.h and perhaps it is not using the latest and greatest but an old copy mapi.h that was copied into a folder like /usr/include/libmilter/mfapi.h

```
rm -rf  /usr/include/libmilter
cp -pr /usr/src/gnu/usr.sbin/sendmail/include/libmilter /usr/include/
```

- What else have we/you forgot?

Now that you have it done, and hopefully it runs after a reboot. *Note that next time you use ssh or scp to the machine the keys may have changed!*

```
reboot
```

If it does not finish rebooting remotely, maybe the problem is a simple mistake on /etc/rc.conf, /etc/rc.local or the new kernel didn't recognize hardware the same way. If the remote machine is very remote, that will be very sad. Maybe you can have some one nearer put a monitor on it and tell you want the console says or maybe turn of the power and restart. Good luck.

Upgrading from distribution to recent stable or current

Assuming that everything is otay after the reboot, now you should upgrade yet again to get changes since your distribution. If you are chasing the edge, current, you better read and reread the most recent notes added to Upgrading OpenBSD FAQ [19]. For upgrading to stable read the OpenBSD literature associated.

http://www.openbsd.org/stable.html **[100]**

- What is the patch branch? [101]
- Getting the patch branch [102]
- Building from the patch branch [103]

We updated using source by anoncvs [104] to modify /usr/src and redid the make obj && make build then used mergmaster to merge configuration changes yet again (may want to delete any temp stuff from last use of mergemaster if haven't already). If you already have a src tree and need to change tags, or just to be safe declare Tag and environmental variables like below for bourne shell. Also we may want to clean all the cruft out of the existing source tree. A very important thing you can do is to clean out your obj directories. (Note you could also tar a know good /usr/src that was updated or checkout recently and use that, but if you do be sure to clean the 'cruft' out of it).

```
###  -- example with anoncvs@anoncvs1.usa.openbsd.org as server,
###      you may want to change
/bin/sh
cd /usr/src
```

```
export CVSROOT= "anoncvs@anoncvs1.usa.openbsd.org:/cvs"
export CVS_RSH=/usr/bin/ssh
### NOTE: If you are updating a source tree that you initially fetched from a
### different server, or from a CD, you must add the -d $CVSROOT options to cvs.
#cd /usr/src
#cvs -d $CVSROOT -q up -Pd
cd /usr/src
cvs -q up -rOPENBSD_3_5 -Pd
##### example using 4 aug2003 current update:
### cvs -q up -D "2003-08-04 12:00" -Pd
######  of note, if use -A will make current
### cvs up -Pd -A
find . -type l -name obj | xargs rm
make -k cleandir
rm -rf /usr/obj/*

### If you're having trouble with left over stuff in /usr/src, per suggestion of the
###  Upgrading OpenBSD FAQ [19] you may want to
###  try adding -I ! -I CVS -I obj to your CVS updates.
```

And now to bring everything in usr land up to date for stable version. You may want to modify the commands below to use nohup with make build since it takes a long time to rebuild the usr land and you can watch it with a tail -f nohup.out

```
### note, may want to edit /etc/mk.conf if want things like libmilter support made during build


cd /usr/src
make obj
make build
cd /dev
cp /usr/src/etc/etc.`machine`/MAKEDEV ./
./MAKEDEV all
cd /usr/src/etc/mtree
install -c -o root -g wheel -m 600 special /etc/mtree
install -c -o root -g wheel -m 444 4.4BSD.dist /etc/mtree
mtree -qdef /etc/mtree/4.4BSD.dist -p / -u
mergemaster
```

And it probably would not hurt to recompile your kernel with the latest and replace the generic distribution kernel. Note if you are using samba [105] or doing a gateway you will want to research tuning the kernel parameters. Another excellent source of information on OpenBSD is the site by nomoa, http://nomoa.com/bsd/kernel.htm [106] :

OpenBSD FAQ 5.0 - Kernel configuration
http://www.openbsd.org/faq/faq5.html [85]

And as afore mentioned **[107]**, you may have to reset some file permissions like suexec, and copy some includes out of source that a port may need.

Reboot yet again and verify all is otay then document findings

```
reboot
```

It is a good idea to shutdown the computer completely and restart while you are available to monitor so as to make darn sure some strange error doesn't show up days, weeks later when an ups outage or forced reboot happens! Last thing is make notes. And if you could be so kind please send us corrections or improvements to our notes here, openbsd@webengr.com. - tia.

.....PHEW.....

More Concerns:

Some packages have security issues and may need upgrading. Upgrading packages even if conveniently made in /usr/ports may become tricky because of dependencies and changes in configuration files.

You may want to research and if appropriate upgrade X4 and your packages and custom applications

Please note that the above will not build and install X4, make and validate the XF4 release, nor make the third party packages! If your machine is a server and not using X windows this may not be a worry for you, but do worry about upgrading the packages and ports you use! Read the man page on 'release' for more information. See also a script created by FenderQ, that may or may not work for you... release_building.sh [108]

Excerpt from man release

"...
At this point you have most of an OpenBSD release. The only thing missing is the X Window System (which is covered in the next section).

5. Build and install XF4 The XF4 tree is primarily imake(1)-based and doesn't contain the ``obj'' directory mechanism that comes with Berkeley make(1). While the tree can be built in place, it's better to refrain from polluting the cvs sources. An alternate build location needs to be selected, large enough to hold the X Window System object files, libraries, and binaries. Call this location XF4BLD. XF4SRC is the path to your X Window System source files. Once you've selected XF4BLD the build process is ...."

If you like this document make a donation to OpenBSD [109],
and recommend our services [110] ;-)

Source URL: https://cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading

Links
[1] http://www.openbsd.org/faq/upgrade51.html

[2] http://www.openbsd.org/faq/upgrade35.html
[3] http://www.openbsd.org/faq/upgrade36.html
[4] http://www.openbsd.org/faq/upgrade37.html
[5] http://www.openbsd.org/faq/upgrade38.html
[6] http://www.openbsd.org/faq/upgrade39.html
[7] http://www.openbsd.org/faq/upgrade40.html
[8] http://www.openbsd.org/faq/upgrade41.html
[9] http://www.openbsd.org/faq/upgrade42.html
[10] http://www.openbsd.org/faq/upgrade43.html
[11] http://www.openbsd.org/faq/upgrade44.html
[12] http://www.openbsd.org/faq/upgrade45.html
[13] http://www.openbsd.org/faq/upgrade46.html
[14] http://www.openbsd.org/faq/upgrade47.html
[15] http://www.openbsd.org/faq/upgrade48.html
[16] http://www.openbsd.org/faq/upgrade49.html
[17] http://www.openbsd.org/faq/upgrade50.html
[18] http://www.openbsd.org/faq/index.html
[19] http://www.openbsd.org/faq/upgrade-minifaq.html
[20] http://pintday.org/hack/
[21] http://pintday.org/hack/docs/remote-upgrade-howto.shtml
[22] http://a.area51.dk/openbsd/upgrade
[23] http://www.openbsd.org/orders.html
[24] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/cdrom/
[25] http://www.openbsd.org/mail.html
[26] http://www.google.com/bsd
[27] http://www.undeadly.org/
[28] http://www.monkey.org/cgi-bin/wilma/openbsd-misc
[29] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#X4
[30] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/notes_minimize_space_on_etcvar.html
[31] http://www.samba.org/
[32] http://open.bsdcow.net/tutorials/ssh_pubkey_auth
[33] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#anoncvsupgrade
[34] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#Replacekernel
[35] http://www.openbsd.org/faq/upgrade-minifaq.html#3.3.0
[36] http://pintday.org/
[37] http://www.openbsd.org/faq/current.html
[38] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#2.9%20to%203.0
[39] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#3.0%20to%203.1
[40] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#3.1%20to%203.2
[41] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#3.2%20to%203.3
[42] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#3.3%20to%203.4
[43] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#3.4%20to%203.5
[44] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#3.5%20to%203.6
[45] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#3.6%20to%203.7
[46] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#3.7%20to%203.8

[47]
http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#3.8%20to%203.9
[48]
http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#3.9%20to%204.0
[49]
http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#4.0%20to%204.1
[50]
http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#4.1%20to%204.2
[51]
http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#4.2%20to%204.3
[52] http://www.openbsd.org/cgi-bin/man.cgi?query=vipw&sektion=8
[53] http://www.openbsd.org/cgi-bin/man.cgi?query=crontab&sektion=1
[54] http://www.openbsd.org/cgi-bin/man.cgi?query=at&sektion=1
[55] http://www.openbsd.org/faq/upgrade-minifaq.html#3.3.1
[56] http://graveland.net/openbsd/upgrade-33-34/
[57] ftp://ftp.courtesan.com/pub/millert/OpenBSD/3.4-a.out/bsd.rd
[58] http://undeadly.org/cgi?action=article&sid=20031028101040
[59] http://www.openbsd.org/cgi-bin/man.cgi?query=useradd&sektion=8
[60] http://www.x.org/
[61] http://www.openbsd.org/faq/upgrade38.patch
[62] http://www.openbsd.org/cgi-bin/man.cgi?query=mtree&sektion=8
[63] http://www.openbsd.org/faq/faq5.html#NewCompiler
[64] http://www.openbsd.org/cgi-bin/man.cgi?query=sendmail&sektion=8
[65] http://www.openbsd.org/plus41.html
[66] http://www.openbsd.org/cgi-bin/man.cgi?query=pf.conf&sektion=5
[67] http://www.openbsd.org/cgi-bin/man.cgi?query=enc&sektion=4
[68] http://www.openbsd.org/cgi-bin/man.cgi?query=isakmpd&sektion=8
[69] http://www.openbsd.org/cgi-bin/man.cgi?query=fdisk&sektion=8
[70] http://www.openbsd.org/cgi-bin/man.cgi?query=sensorsd.conf&sektion=5
[71] http://www.openbsd.org/cgi-
bin/man.cgi?query=pkg_add&sektion=1&manpath=OpenBSD+4.1
[72] http://www.openbsd.org/faq/faq15.html#PkgMgmt
[73] http://www.openbsd.org/cgi-bin/man.cgi?query=ahci&sektion=4
[74] http://www.openbsd.org/cgi-bin/man.cgi?query=wd&sektion=4
[75] http://www.openbsd.org/cgi-bin/man.cgi?query=sd&sektion=4
[76] http://www.openbsd.org/cgi-bin/man.cgi?query=bgplg&sektion=8
[77] http://www.openbsd.org/cgi-bin/man.cgi?query=sudo&sektion=8#SECURITY+NOTES
[78] http://www.openbsd.org/cgi-bin/man.cgi?query=sudo&sektion=8
[79] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#patchfile
[80] http://www.openbsd.org/alpha.html
[81] http://www.openbsd.org/cgi-bin/man.cgi?query=de&sektion=4
[82] http://www.openbsd.org/cgi-bin/man.cgi?query=dc&sektion=4
[83] http://www.openbsd.org/cgi-bin/man.cgi?query=apm&sektion=4&arch=i386
[84] http://www.openbsd.org/cgi-bin/man.cgi?query=acpi&sektion=4
[85] http://www.openbsd.org/faq/faq5.html
[86] http://www.openbsd.org/cgi-bin/man.cgi?query=MAKEDEV&sektion=8&arch=i386
[87] http://www.openbsd.org/faq/upgrade42.patch
[88] http://www.openbsd.org/cgi-bin/man.cgi?query=newaliases&sektion=8
[89] http://www.openbsd.org/faq/faq4.html#AddFileSet
[90] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#sudo
[91] http://www.openbsd.org/cgi-
bin/man.cgi?query=pkg_add&sektion=1&manpath=OpenBSD+4.2
[92] http://www.openbsd.org/cgi-bin/man.cgi?query=agp
[93] http://www.openbsd.org/cgi-bin/man.cgi?query=relayd&sektion=8
[94] http://www.openbsd.org/faq/faq8.html#Packages

[95] http://www.openbsd.org/cgi-bin/man.cgi?query=pkg_add&sektion=1&manpath=OpenBSD+3.9
[96] http://www.openbsd.org/faq/upgrade-minifaq.html#1.9
[97] http://whatis.techtarget.com/definition/0,,sid9_gci748437,00.html
[98] http://www.openbsd.org/cgi-bin/man.cgi?query=ed
[99] http://www.fatbsd.com/openbsd/clamav/index.php
[100] http://www.openbsd.org/stable.html
[101] http://www.openbsd.org/stable.html#whatis
[102] http://www.openbsd.org/stable.html#getting
[103] http://www.openbsd.org/stable.html#building
[104] http://www.openbsd.org/anoncvs.html
[105] http://nomoa.com/bsd/samba.htm
[106] http://nomoa.com/bsd/kernel.htm
[107] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/#worry-reboot-pray
[108] http://old.cocoavillagepublishing.com/development/tools/openbsd/tips/upgrading/release_building.sh
[109] http://www.openbsd.org/donations.html
[110] http://old.cocoavillagepublishing.com/services/