Unix System Administration

Updating FreeBSD

Objectives

At the end of this session you should be able to:

1. Understand the differences between the CURRENT vs STABLE branches of FreeBSD
2. Use cvsup to update your source tree
3. Update your entire system to the latest STABLE release using buildworld.

Why update

- Fix bugs in the OS – some of them related to security issues that could potentially leave your system vulnerable to different kinds of attack.
- Get “drivers” for new hardware previously unsupported and/or updates to existing drivers.
- Other features that may be interesting like better memory management, background FSCK (introduced in the 5.X versions) etc

Branches of FreeBSD

- CURRENT

  The ``bleeding edge'' of FreeBSD development. FreeBSD-CURRENT users are expected to have a high degree of technical skill, and should be capable of solving difficult system problems on their own. If you are new to FreeBSD, PLEASE DO NOT install it.
- STABLE

  The development branch from which major releases are made. Changes go into this branch at a different pace, and with the general assumption that they have first gone into FreeBSD-CURRENT for testing. This is still a development branch, however.

Users of CURRENT

- Members of the FreeBSD community who are actively working on some part of the source tree and for whom keeping ``current'' is an absolute requirement.
- Members of the FreeBSD community who are active testers, willing to spend time solving problems in order to ensure that FreeBSD-CURRENT remains as sane as possible. These are also people who wish to make topical suggestions on changes and the general direction of FreeBSD, and submit patches to implement them.
- Those who merely wish to keep an eye on things, or to use the current sources for reference purposes (e.g. for reading, not running). These people also make the occasional comment or contribute code.
- Subscribe to freebsd-current@freebsd.org

STABLE

- Still developmental – but more tested than -CURRENT. Track STABLE with caution. This is what gives us the next release of FreeBSD.
- Possible to track a specific version after it's been released. E.g tracking only 5.3 which should get mostly security updates and almost nothing else.
- Advisable to test on a testing server before deployment in a live environment.
- Subscribe to freebsd-stable@freebsd.org

How to upgrade

- Reinstalling a new release from CD – this overwrites all your configuration files, settings etc. (Partition wisely for data preservation). 3rd Party applications need to be reinstalled.
- Binary upgrade feature available on the CDROM:
  - ✔ Fast and Easy
  - ✗ Very buggy and could break your system esp by not maintaining your configurations
  - ✗ Will not preserve custom options done at compile time e.g removing IPV6 from the kernel
  - ✗ Not optimised for your hardware – a problem if your hardware is very specific

- Binary update using FreeBSD update available online from the website http://www.daemonology.net/freebsd-update/ and also available in the ports
  - ✔ Tracks only security problems
  - ✔ Fast and easy to run
  - ✗ Still loses your compile time customisations
  - ✗ Potentially puts part of the system out of sync with the rest of the system if it doesn't handle changes to configuration files
  - ✗ Does not play nice if you upgrade your system any other way.
- Updating through source (using cvsup)
  - ✔ Most flexible – maintaining your run-time modifications.
  - ✔ Allows you to maintain your configuration files and/or update them manually when needed

How cvsup Works

- Anonymous CVS and CVSup use the pull model of updating sources. In the case of CVSup the user (or a cron script) invokes the cvsup program, and it interacts with a cvsupd server somewhere to bring your files up-to-date. The updates you receive are up-to-the-minute and you get them when, and only when, you want them. You can easily restrict your updates to the specific files or directories that are of interest to you. All this is specified in the supfile which we are going to edit in the next exercise.

At this point Go to exercise 1 that has been handed out.

•

Rebuilding World

- Once you have synchronized your local source tree against a particular version of FreeBSD (FreeBSD-STABLE, FreeBSD-CURRENT, and so on) you can then use the source tree to rebuild the system.
- Before you do anything else, read /usr/src/UPDATING (or the equivalent file wherever you have a copy of the source code). This file should contain important information about problems you might encounter, or specify the order in which you might have to run certain commands. If UPDATING contradicts something you read here, UPDATING takes precedence.

Exercise

- Now go to exercise 2 on the handout you have.

- Do source update

  cd /usr/src

  make -j2 buildworld

  make buildkernel

  make installkernel

  [reboot in single user mode]

  cd /usr/src

  mergemaster -p

  make installworld

  mergemaster

  [reboot]

Notes

- It is possible to specify a -j option to make which will cause it to spawn several simultaneous processes. ONLY use this on the buildworld NOT on any of the other make commands.
- If you want to build a custom kernel, and already have a configuration file, just use KERNCONF=MYKERNEL like this:
- # make buildkernel KERNCONF=MYKERNEL
- # make installkernel KERNCONF=MYKERNEL

Mergemaster

- The mergemaster(8) utility is a Bourne script that will aid you in determining the differences between your configuration files in /etc, and the configuration files in the source tree /usr/src/etc. This is the recommended solution for keeping the system configuration files up to date with those located in the source tree.
- running mergemaster(8) in pre-buildworld mode by providing the -p option will compare only those files that are essential for the success of buildworld or installworld.

Review Question 1

The most reliable way of updating FreeBSD is:

a) Using the CDs

b) Doing a binary update

c) using cvsup

d) using CTM

e) None of the above

Review Question 2

CVSUP updates the system sources by:

a) downloading all the system sources from the source server.

b) Downloading all the binaries from the source server.

c) patching the local sources against the remote sources.

d) comparing the local sources and remote sources and downloading only those that have changed

Review Question 3

Running make -j4 buildworld will:

a) Build the world for 4 different machines.

b) Compile the system sources 4 times

c) Run 4 simultaneous processes of make

d) None of the above

Review Question 4

When running FreeBSD as a production machine I should:

a) Use CURRENT rather than STABLE for my cvsup

b) Keep track of stable and run cvsup on my server regularly

c) Track stable, test on a test machine first before doing it on my production machines.

d) None of the above

Review Question 5

Make buildkernel and installkernel should be run:

a) Before building the world

b) after building the world but before installing the world.

c) after installing the world but before mergemaster.

d) after everything else but before rebooting.

Review Question 6

Mergemaster will:

a) copy the new configuration files into your /etc/directory

b) compare your current files to your previous files and prompt you on what it should do.

c) automatically merge the contents of the new file with the old files.

d)