

Time Machine: FreeBSD

Peter Kostouros

Every month we delve into the PC Update archive to take a look at what computing was like in the past...this month, a review of FreeBSD from February 2000.

Until recently my home workstation was host to several operating systems: Linux, my preferred environment; Windows 98 and Windows NT (I have legacy applications), and BeOS which I loaded to see if it really was 'cool'. An on-going episode with a sound card and Windows NT got me thinking it was time for something new.

FreeBSD is a freely available operating system primarily for Intel-based (386 and higher) PCs. A descendant of 386BSD, it has many features expected from a UNIX type operating system, for example multi-user support, preemptive multi-tasking, TCP/IP networking, and memory protection; it can also run SCO and Linux programs.

People may take these features for granted, but what also interested me was that FreeBSD was behind some popular and busy Internet sites including Yahoo! and Walnut Creek (the world's busiest ftp server which recently served out more than a terabyte of data in a day).

In this article I will describe installing and running FreeBSD.

Preamble

I purchased FreeBSD (3.2) online and received the package a few hours later. (A couple of days later FreeBSD 3.3 was released.) Since I was not rushed into installing the software I took time and read FreeBSD related news-groups (comp.unix.bsd.freebsd.misc) to prepare for the "Oh no, what have I done?" kind of feeling.

FreeBSD requires at least 5 MB RAM to install (though it will run with 4 MB RAM), and a minimum of about 100 MB of free hard disk space. Additionally, FreeBSD does not attempt to support as many devices as possible as its focus is elsewhere: it may be preferable to scan through the compatibility list.

Two problems arose immediately: FreeBSD must reside on a primary partition and to enable it to boot, certain files must be located within the first 1024 cylinders of the disk on it is installed. (Note some motherboards enable booting of operating systems loaded within the first 8 GB.) Disk partitioning software is supplied however it may be inadequate (FAT32 or VFAT partitions are not supported), so a commercial product may be required. In my case, I played a variation of the Towers of Hanoi problem with my partitions to allocate space.

Installation

The installation manual describes various install methods including installing over the Internet; the simplest method though is via CD-ROM, which I chose. If your system cannot boot from CD-ROM discs, boot floppies must be created.

When the computer was rebooted the first recognizable thing to pop up was a kernel configuration program that enables you to manually select and configure components such as hard disk controllers and network cards, or remove support for components: I removed support for components not present in my system. Once completed, FreeBSD probes the computer and after a few moments the install program is (automatically) started.

The install program is text-based and offers several install methods: beginners, impatient, and expert.

I chose expert as this option gives the most flexibility. With the expert option you set up partitions for FreeBSD; partitioning can be a complex process as hard disk performance and maintenance are dependent on the way partitions have been set up, so it may be worthwhile investigating. During the partitioning stage, a boot manager may be installed: the boot manager enables booting from various partitions or from different hard disks.

I chose to forgo this option as I have been happy with the Linux boot manager lilo. It is also possible to use NT's boot manager. (As no boot manager was installed, the installation process sets the partition FreeBSD was installed on as bootable.)

To assist users select programs to install, several prepackaged distributions are offered, for example, 'X-User' which installs a graphical user interface (the X Window System, which is more than a GUI) and general programs along with documentation,

and 'X-Kern-Developer' which along with the programs installed for 'X User', lots of source code and the necessary components to create the operating system itself are loaded. Alternatively, a custom distribution can be created where various components are selected for installation.

Once my selections the system started installing. This process took about an hour. If installing the X Window System you are required to select the appropriate X server that supports your graphics card and extra fonts to install.

Post installation

At this stage FreeBSD is about ready to run. Here several user accounts were created as was a group 'ppp' for those users who will have Internet access.

I also setup access to my Windows partitions and configured the X Window System with the program XF86Setup. This process involves selecting an appropriate graphics card, graphics modes, monitor timings, and keyboard. More programs can also be installed: I wanted to make the system resemble what I was accustomed to with Linux so I installed the Windowmaker window manager.

To enable my system to boot FreeBSD I restarted Linux and added a FreeBSD section in the lilo configuration files and ran the program lilo to commit the changes.

Working with FreeBSD

The first time I started FreeBSD I was presented with a login prompt on a text console. Once logged in, I started exploring the system by issuing commands like 'cd' to change directory and 'ls' to list the contents of directories. At first it may be difficult for those new to UNIX. However, once familiar with it you may find working with Windows difficult!



Figure 1. The files and directories under the root directory

For those unfamiliar with UNIX, '/' is a file delimiter and the topmost directory (the root directory) is identified by /. Figure 1 shows the files and directories under the root directory: /kernel and /kernel.GENERIC are kernels (programs that interact with hardware); /var contains log files, administration files, and mail; /usr contains system files and system programs; /usr/local is usually reserved for programs that you load, and; /home is the base directory for users, for example /home/peter is the work directory for the user 'peter'.

FreeBSD comes with a handbook that covers installation and basic use. It should be noted that when issuing certain commands (e.g., rm, remove a file), the system does not display counselling windows informing you of damage (if any) about to be caused. There are also many UNIX resources available on the Internet which cover basic to advanced system administration.

There are several ways to start the GUI, one being by running the program xinit. As I stated earlier I wanted the look and feel of what I was used to and spent some time putting things together: that meant setting up icons and menus. Figure 2 shows one of my sessions. If you are used to the Windows look and feel, there are window managers which imitate it.

There are several file managers, among them kfm (KDE's file manager and integrated web browser) and gmc (GNOME's file manager), and are used for exploring the system and basic maintenance. KDE and GNOME are efforts to make working with UNIX simple and come with many supporting programs such as text editors, word processors, user management utilities, network utilities, and of course, games.

Along with the KDE and GNOME applications there are about 2700 applications for FreeBSD, and as stated earlier, Linux programs can be run (this document was written with *WordPerfect for Linux* under FreeBSD). Additional components of the FreeBSD system can be installed by running the program 'sysinstall'. For those who enjoy creating applications instead of installing pre-built ones, FreeBSD has a ports collection. Many UNIX programs come with source code and the idea behind the ports collection is to get these programs working under FreeBSD: note that not every UNIX program is supported by the ports collection.

Connecting to the Internet

There are several programs that assist setting up PPP connections, however, I chose to do it the hard way. The handbook describes setting up connections to an ISP and essentially it requires editing the example PPP configuration files. My first attempt was unsuccessful: I did not give myself permission to dial out! (The person I logged in as was not given such privileges by the administrator, who also happens to be me.) Once the configuration files were modified (setting up users for PPP), the world was my oyster.

Initially dialling out was a manual process: I issued the command 'ppp - background melbpc' from the console. Once satisfied all was well, I created menu items from which to launch PPP sessions, and modified some files to display messages upon connection or disconnection; the example files also show how to play sounds dependent on the state of the connection.



Concluding Remarks

Some newsgroups have heated discussions on which is the best operating system as zealots wave their banners; occasionally some people add they have the best of most worlds by running several operating systems, and this is the camp I am in. With thousands of applications available, its ability to run Linux programs (Oracle for Linux, DB2 for Linux, WordPerfect for Linux), (limited) support for running Windows programs via the WINE project, and being behind some high profile Internet sites, FreeBSD has become the darling of operating systems on my PC.

To find out more about FreeBSD, visit <http://www.au.freebsd.org>.