22.1 Synopsis 541
22.2 Installation 541
22.3 Installing Mathematica 544
22.4 Installing Maple 546
22.5 Installing Oracle 548
22.6 Installing SAP R/3 551
22.7 Advanced Topics 572
III. Appendices 574
A. Obtaining FreeBSD 575
A.1 CDROM Publishers 575
A.2 DVD Publishers 576
A.3 FTP Sites 577
A.4 Anonymous CVS 584
A.5 Using CTM 586
A.6 Using CVSup 590
A.7 CVS Tags 608
A.8 AFS Sites 611
A.9 rsync sites 611
B. Bibliography 613
B.1 Books & Magazines Specific to FreeBSD 613
B.2 Users' Guides 613
B.3 Administrators' Guides 614
B.4 Programmers' Guides 614
B.5 Operating System Internals 615
B.6 Security Reference 616
B.7 Hardware Reference 616
B.8 Unix History 616
B.9 Magazines and Journals 617
C. Resources on the Internet 618
C.1 Mailing Lists 618
C.2 Usenet Newsgroups 629
C.3 World Wide Web Servers 630
C.4 Email Addresses 632
C.5 Shell Accounts 632
D. PGP Keys 634
D.1 Officers 634
D.2 Core Team Members 634
D.3 Developers 635
Colophon 655

1994 to stop shipping its own Net/2 based product. Under the terms of that agreement, the project was allowed one last release before the deadline, that release being FreeBSD 1.1.5.1.

FreeBSD then set about the arduous task of literally re-inventing itself from a completely new and rather incomplete set of 4.4BSD-Lite bits. The “Lite” releases were light in part because Berkeley’s CSRG had removed large chunks of code required for actually constructing a bootable running system (due to various legal requirements) and the fact that the Intel port of 4.4 was highly incomplete. It took the project until November of 1994 to make this transition, at which point it released FreeBSD 2.0 to the net and on CDROM (in late December). Despite being still more than a little rough around the edges, the release was a significant success and was followed by the more robust and easier to install FreeBSD 2.0.5 release in June of 1995.

We released FreeBSD 2.1.5 in August of 1996, and it appeared to be popular enough among the ISP and commercial communities that another release along the 2.1-STABLE branch was merited. This was FreeBSD 2.1.7.1, released in February 1997 and capping the end of mainstream development on 2.1-STABLE. Now in maintenance mode, only security enhancements and other critical bug fixes will be done on this branch (RELENG_2_1_0).

FreeBSD 2.2 was branched from the development mainline (“-CURRENT”) in November 1996 as the RELENG_2_2 branch, and the first full release (2.2.1) was released in April 1997. Further releases along the 2.2 branch were done in the summer and fall of ’97, the last of which (2.2.8) appeared in November 1998. The first official 3.0 release appeared in October 1998 and spelled the beginning of the end for the 2.2 branch.

The tree branched again on Jan 20, 1999, leading to the 4.0-CURRENT and 3.X-STABLE branches. From 3.X-STABLE, 3.1 was released on February 15, 1999, 3.2 on May 15, 1999, 3.3 on September 16, 1999, 3.4 on December 20, 1999, and 3.5 on June 24, 2000, which was followed a few days later by a minor point release update to 3.5.1, to incorporate some last-minute security fixes to Kerberos. This will be the final release in the 3.X branch.

There was another branch on March 13, 2000, which saw the emergence of the 4.X-STABLE branch, now considered to be the "current -stable branch". There have been several releases from it so far: 4.0-RELEASE came out in March 2000, 4.1 was released in July 2000, 4.2 in November 2000, 4.3 in April 2001, and 4.4 in September 2001. There will be more releases along the 4.X-stable (RELENG_4) branch well into 2002.

Long-term development projects continue to take place in the 5.0-CURRENT (trunk) branch, and SNAPshot releases of 5.0 on CDROM (and, of course, on the net) are continually made available from the snapshot server (ftp://current.FreeBSD.org/pub/FreeBSD/snapshots/) as work progresses.

1.3.2 FreeBSD Project Goals

Contributed by Jordan Hubbard.

The goals of the FreeBSD Project are to provide software that may be used for any purpose and without strings attached. Many of us have a significant investment in the code (and project) and would certainly not mind a little financial compensation now and then, but we are definitely not prepared to insist on it. We believe that our first and foremost “mission” is to provide code to any and all comers, and for whatever purpose, so that the code gets the widest possible use and provides the widest possible benefit. This is, I believe, one of the most fundamental goals of Free Software and one that we enthusiastically support.

That code in our source tree which falls under the GNU General Public License (GPL) or Library General Public License (LGPL) comes with slightly more strings attached, though at least on the side of enforced access rather than the usual opposite. Due to the additional complexities that can evolve in the commercial use of GPL software we do, however, prefer software submitted under the more relaxed BSD copyright when it is a reasonable option to do so.

for the firewall to accept any packet. Make sure you know how this changes the meaning of your ruleset before you set it.

19.3.3.4 Traffic Shaping Support

If you want to use the bridge as a traffic shaper, you will need to add the `DUMMYNET` option to your kernel configuration. Read dummynet(4) for further information.

19.3.4 Enabling the Bridge

Add the line:

```
net.link.ether.bridge=1
```

to `/etc/sysctl.conf` to enable the bridge at runtime, and the line:

```
net.link.ether.bridge_cfg=if1,if2
```

to enable bridging on the specified interfaces (replace *`if1`* and *`if2`* with the names of your two network interfaces). If you want the bridged packets to be filtered by ipfw(8), you should add:

```
net.link.ether.bridge_ipfw=1
```

as well.

19.3.5 Performance

My bridge/firewall is a Pentium 90 with one 3Com 3C900B and one 3C905B. The protected side of the network runs at 10 mbps half duplex and the connection between the bridge and my router (a Cisco 675) runs at 100 mbps full duplex. With no filtering enabled, I have found that the bridge adds about 0.4 milliseconds of latency to pings from the protected 10 mbps network to the Cisco 675.

19.3.6 Other Information

If you want to be able to telnet into the bridge from the network, it is OK to assign one of the network cards an IP address. The consensus is that assigning both cards an address is a bad idea.

If you have multiple bridges on your network, there cannot be more than one path between any two workstations. Technically, this means that there is no support for spanning tree link management.

Colophon

This book is the combined work of hundreds of contributors to “The FreeBSD Documentation Project”. The text is authored in SGML according to the DocBook DTD and is formatted from SGML into many different presentation formats using **Jade**, an open source DSSSL engine. Norm Walsh’s DSSSL stylesheets were used with an additional customization layer to provide the presentation instructions for Jade. The printed version of this document would not be possible without Donald Knuth’s **TeX** typesetting language, Leslie Lamport’s **LaTeX**, or Sebastian Rahtz’s **JadeTeX** macro package.