
FreeBSD MPS v3

User's Guide

First Edition
May 2007

Table of Contents

Introduction	1
How to Use this Document	1
Audience for this Document	1
Overview of this Document	1
Shell Prompts in Command Examples	2
Overview of Your Server	2
Functional Overview of Features	2
Services and Features Overview	3
Web Services	3
Mail Services	3
FTP Services	3
Web Development Tools	4
E-Commerce	4
Databases	5
Multimedia	5
Statistics and Log Analyzer Packages	5
FreeBSD is a UNIX Operating System	5
FreeBSD UNIX and Your Server	5
Use the Features of Your Server	6
Configure Your Server	7
Connect to Your Server the First Time	7
Access Your Server	8
Creating and Editing User Accounts	8
Configure Virtual Sub Hosts	9
Create a Virtual Host	9
Default Applications for Your Server	10
Apache HTTP Server	11
Autoresponder	11
FTP	12
IMAP	12
OpenSSH	13
Securing Root Access by Means of SSH	13
Access Authorized SSH Keys	13
Perl	14
POP 3 Server	14
Python Programming Language	14
Ruby Scripting Language	15
Embed Ruby Code	15
Sendmail SMTP Server	15
SSL	16
Create a Signing Request and Private Key	16
Custom Digital Certificate	18
Obtain a Signed Digital Certificate	18
Install your Custom Digital Certificate	19
Move your Custom SSL Certificate	20
Change Operating Systems	20
Move a Certificate to a New Server	20
Renew Custom digital certificates	21
Install Additional Features	22
Accrisoft Freedom	22
Apache DSO Modules	22
Aspell	22
ClamAV	22

CPX: Control Panel	23
Email List Package	24
Dada Mail	24
Majordomo	25
Mailman	25
Email Service.....	25
Expect.....	25
FormMail	25
Installing FormMail	25
Using FormMail.....	26
FrontPage.....	27
HTTP-Analyze.....	27
iManager Web-based Server Utility	27
Java	28
MajorCool Web Interface Maintenance Tool	28
Metamail.....	28
Multiple IP Addresses.....	28
How Multiple IP Addresses Work with Other Features	28
Overview of the New Multiple IP Address Feature	29
Potential Uses for Multiple IP Addresses	29
How Your Server Utilizes Multiple IP Addresses	29
Overview of Configuring Multiple IP Addresses	29
Managing Multiple IP Addresses, Subhosts, and Certificates	30
New and Updated Command-line Utilities	30
Adding a Subhost.....	30
Administrative Email and Document Root settings	31
Log and cgi-bin Settings	31
Assigning a New SSL Certificate	32
Going Beyond the Basics.....	32
Your Responsible Use of IP Addresses	32
Configure Provisioned IP Addresses Only	33
CPX: Control Panel and Multiple IP Addresses	33
MySQL	33
Namazu	34
Open WebMail.....	34
PHP	34
phpMyAdmin.....	34
PGP/GnuGP	34
PostgreSQL.....	35
Multi-Language Abilities in PostgreSQL	35
Procmal.....	36
Samba	36
SpamAssassin	36
Savelogs.....	37
Shockwave.....	37
SquirrelMail.....	38
Swish-e	38
TCL.....	38
Time Zone Custom Installation Utility	39
Tomcat	39
TWIG.....	39
Urchin 5 (Google Analytics).....	39
Urchin 5 Web Log Analyzer Features	39
Install Urchin	39
Configure Urchin	40
Vinstall Utilities Library.....	40

Removing Software Packages.....	41
Software Packages Included in the Vinstall Utilities Library	41
WordPress.....	42
Available Features	42
Before you Install WordPress	42
Get Started	42
Go Beyond the Basics with WordPress	43
The Webalizer.....	43
Webmin	43
WebTrends.....	43
Wpoison.....	44
Install Wpoison	44
Use Wpoison.....	44
Zend Optimizer.....	45
Install Zend Optimizer	45
Go Beyond Zend Optimizer Basics	45
Zope.....	45
Install Zope	45
Use Zope.....	46
Go Beyond the Basics of Zope	46
Troubleshoot Your Server	47
General Issues.....	47
Failure to Create a Virtual Host.....	47
Check Quotas.....	47
Check Log Files.....	47
Check for Idle Processes.....	48
Custom Digital Certificate Problems	48

Introduction

Use the instructions included in this document and apply your previous system administration experience to configure your of a FreeBSD Managed Private Server, version three (FreeBSD MPS v3), administer all features of your server, and troubleshoot common concerns. By using this document, conduct these tasks at your own pace, on your own, and without extensive technical support. This introduction provides you with descriptions of how to use this document, the audience it is intended to reach, and the product's features.

In addition to this introduction, this document includes the following sections:

- “Configure Your Server” on page 7.
- “Default Applications for Your Server” on page 10.
- “Install Additional Features” on page 22.
- “Troubleshoot Your Server” on page 47.

How to Use this Document

Note: Some additional, late-breaking information regarding installation, administration, and troubleshooting tasks are included in release notes and FreeBSD MPS v3 support-related Web content such as frequently asked questions (FAQ). Always verify you have acquired the latest information available prior to installing, administering, or troubleshooting your server.

This document provides you with an overview of your server. This document describes the details of how to install, maintain, and troubleshoot your server. When applicable, the document describes these tasks by instructing you to use product-specific commands and operations. However, not all features of your server use product-specific commands and operations. In those cases, this document describes the details of how the features function and refers you to the correct resources provided by the FreeBSD operating system or the provider of software package provider.

Audience for this Document

This document provides information useful to FreeBSD MPS v3 server administrators located in, but not limited to, any of the following types of organizations:

- Hosting service provider (HSP)
- Application service provider (ASP)
- Independent software vendor (ISV)
- Value-added reseller (VAR)
- Small-sized business
- Medium-sized businesses

The instructions describe tasks assuming you have moderate knowledge and familiarity with UNIX, the FreeBSD operating system, as well as some broad knowledge of Internet and Web hosting technologies.

Overview of this Document

This document provides you with the information you need to configure FreeBSD MPS v3 and also to install additional software packages, as well as providing troubleshooting guidelines.

This document is a companion to other print-ready customer documentation which is included, at no cost, as a feature of your server:

- *FreeBSD MPS v3 Getting Started Guide*
- *FreeBSD MPS v3 Release Notes*
- *FreeBSD MPS v3 New Feature Supplement*
- *FreeBSD MPS v3 Technical Overview* (access limited)

There are also Web site resources such as a documentation library and Frequently-Asked Questions (FAQ). The documentation library is updated to include information about new features introduced to your server.

Shell Prompts in Command Examples

Command line examples included in this document assume you use C shell (*cs*) or TENEX C shell (*tcsh*). Wherever a command is able to be issued by a user, this document provides a dollar sign (\$) prompt. When a command is meant to be issued as root, this document provides a hash mark (#).

When you follow the instructions in this document, type the double-quotes or single quotes as displayed. The root path typically includes */bin*, */sbin*, */usr/bin*, or */usr/sbin* directories. The instructions using commands from these directories show the commands in these directories without absolute path names. Instructions which use commands in other directories show the absolute paths in examples.

Overview of Your Server

FreeBSD MPS v3 is the third major release of custom, proprietary technology. Your server provides access to system administration services and technical support. In addition, you can configure and customize your server exactly as you wish. Your server provides resources for high Internet traffic levels, and your server is well-suited for applications which use extensive Random Access Memory (RAM). For example, your server provides the benefits of dedicated speed and performance levels. A traffic-intensive site would benefit from a dedicated database server. One of the most useful advantages of your server is that it offers you the ability to control access while exercising complete control over all the Web sites you host on your server. By following suggested guidelines, you can create as many accounts as you require, for both Post Office Protocol, version three (POP3) and File Transfer Protocol (FTP), without any additional costs. Also, using your server as a Web server, you can immediately troubleshoot and solve end-user concerns.

Functional Overview of Features

There are several FreeBSD MPS v3 plans available with different disk space allocations available. All of the plan levels do not charge additional fees for data transfers and offer unlimited user accounts, mailboxes, and virtual hosts. Your server utilizes Intel hardware.

The following list provides you with a functional overview of the features of your server:

- Utilize and control root access to your server
- Configure multiple shell accounts
- Host unlimited email accounts
- Add multiple IP addresses
- Securely support multiple Web sites
- Utilize to the FreeBSD Ports Collection of applications
- Install and configure the applications of your choice

- Monitor and ensure your server's stability
- Utilize CPX: Control Panel Web interface for simplified account administration.

Services and Features Overview

The following list provides you with an overview of the services and features of your server:

- Web Services
- Mail Services
- FTP Services
- SFTP
- Web Development Tools
- E-Commerce
- Databases
- Multimedia
- Statistics and Log Analyzer Packages

Web Services

Following are the Web services your server provides:

- SSL Secure Server Support
- Complete configuration files
- Raw log files
- Full `cgi-bin` access
- Dynamic module support
- Create and manage multiple podcasts

Mail Services

Following are the mail services your server provides:

- Unlimited POP mailboxes
- Unlimited IMAP mailboxes
- Email quota (disk space) limits
- Unlimited email aliases (forwarding)
- Autoresponder support
- Mailing list support
- Anti-spam features
- Web Mail

FTP Services

Your server's support for FTP provides anonymous FTP and unlimited non-anonymous FTP accounts. It also enables you to set upload quota limits and customize welcome and directory messages. For more, see "FTP" on page 12.

Note: Your server's support for SFTP provides FTP access through SSH.

Web Development Tools

Your server supports the following Web development tools:

- Microsoft FrontPage extensions – FrontPage provides tool for Web pages designed and implemented with the Microsoft Web development software (<http://office.microsoft.com/en-us/frontpage/default.aspx>). For more, see “FrontPage” on page 27.
- Java, Java Servlets, and JavaServer Pages – Your server provides support for Java, Java Servlets, and JavaServer Pages (<http://java.sun.com/>). Java is a trademark of the Sun Corporation. Java products are developed, maintained, and distributed by that organization. For more, see “Java” on page 28.
- iManager Web-based Server Utility – Comparable to the CPX: Control Panel, iManager is a Web-based server utility, which enables you to manage many of the common tasks involved in server administration. In addition to basic user and subhost configuration tools, it includes an IMAP style email manager and an easy to use file manager. Your server provides a `vinstall` utility for iManager.
- PHP4, PHP5 – Your server supports PHP: Hypertext Preprocessor (<http://www.php.net/>), the widely-used, general-purpose, and open-source scripting language distributed with most UNIX binaries. Your server includes a set of `vinstall` utilities for PHP. For more, see “PHP” on page 34.
- MIVA Empresa – Your server supports MIVA Empresa to provide Web development and also includes a set of `vinstall` utilities for the software.
- Perl5 – By default, Perl is pre-installed on your server as a core service. Your server supports Perl (<http://www.perl.org/>), the widely-used, open-source cross platform programming language distributed with most UNIX binaries. Your server includes a set of `vinstall` utilities for `mod_perl`. For more, see “Perl” on page 14.

E-Commerce

FreeBSD MPS v3 supports the following e-commerce software packages:

- ShopSite – For an additional fee, you can add ShopSite (<http://shopsite.com/>) shopping cart software to your server. Once you have made the purchase, refer to ShopSite customer documentation and Web site content.
- MIVA Empresa – MIVA Empresa (<http://smallbusiness.miva.com/products/empresa/>) provides an e-commerce solution. The server features are the same as other plans but also include the MIVA license and software. A set of `vinstall` utilities are available to assist you as you install or upgrade your e-commerce plan.

Databases

The following add-on database software packages are supported by your server:

- MySQL – Your server supports the current, stable release of MySQL, an open source database server and tool distributed under the terms of the GNU General Public License (GPL). For more, see “MySQL” on page 33.
- PostgreSQL – Your server supports the current, stable release of PostgreSQL, an open source relational database system distributed by PostgreSQL Global Development Group under the Berkley Software Distribution (BSD) license. For more, see “PostgreSQL” on page 35.
- Oracle Gateways – Your server supports Oracle Open Gateways (previously called SQL*Connect). You can use the product set to access data from non-Oracle databases and file systems. The product set is developed, maintained, and distributed by Oracle (<http://www.oracle.com/technology/products/gateways/index.html>).

Multimedia

Shockwave/Flash provides support for multimedia playback on your server. The Shockwave Player enables you to view interactive Web content. For more, see “Shockwave” on page 37.

Statistics and Log Analyzer Packages

The following add-on statistics and log analyzer packages are supported by your server:

- WebTrends – Your server supports WebTrends (<http://www.webtrends.com/>) provides a Web Log Analyzer that will gather and report valuable information about your Web site and the users that access it.
- Webmin – Your server supports Webmin (<http://www.webmin.com/>), a Web-based interface for system administration for UNIX. There is a custom installation utility (`vinstall webmin`) to assist you as you install the interface. Webmin is available from the FreeBSD Ports Collection.
- Urchin – The software provides Web analytics and analyzes traffic for one or more Web sites and provides accurate, easy-to-understand reports.

FreeBSD is a UNIX Operating System

FreeBSD MPS v3 utilizes FreeBSD UNIX (FreeBSD 6.x), a widely implemented UNIX standard. Your server also utilizes the UNIX File System 2 (UFS2). FreeBSD is a derivative of the Berkley Software Distribution (BSD) originally developed by the Computer Systems Research Group (CSRG) at the University of California, Berkeley in the United States. The operating system is distributed under the terms of the FreeBSD Foundation (<http://www.freebsdoundation.org/>). The operating system is based on open standards and is derived from the community-supported, open source FreeBSD Project. The operating system provides support for numerous open-source communication, database, and software applications.

FreeBSD UNIX and Your Server

As you perform configuration, administration and trouble-shooting tasks, you will be able to apply your previous knowledge of open-source software applications to FreeBSD MPS v3. By utilizing root access you can grant access to any port. The server supports multiple users and user based applications. With access to all of your server's logs, administration per service is easy to do. Data backups, server security and software updates are updated through server software updates which often require no intervention on your part. Your server can be

remotely rebooted and runs with server monitoring software applications. Configure your server to support multiple users with shell, Web, FTP and/or email privileges. The FreeBSD operating system provides a compatible base for operating system level server virtualization, a template based UFS2 file system (or *skel*) package, and copy-on-write (COW) file system optimization.

For information about the full command set provided on your server, consult the manual pages (or *man pages*). Man pages also provide information about system calls, library calls, special files, as well as file formats and conventions.

Use the Features of Your Server

Following are examples of how to use the features of your server:

- Host an e-commerce Web site
- Support a corporate intranet
- Build a custom development environment
- Provide Web-based calendaring
- Provide multimedia applications
- Host an online game site
- Manage an email system
- Create a customer support tracking system
- Backup important data
- Host multiple Web sites

Configure Your Server

Important: If you are migrating or transferring services to a FreeBSD MPS v3, verify you have a backup, local copy of the files which are essential to your Web site. For example, if you have essential content and graphics. Save them in so that they are accessible even when you are unable access to your server. Do this prior to following any of the subsequent instructions.

As you begin to configure your server, consider the processing power, memory and disk space available on your local system. The following are basic, network requirements for operating your server:

- Local Area Network (LAN).
- Internet connection.
- Valid IP addresses.
- IP addresses are open for access from the outside if you do not apply a firewall.

Connect to Your Server the First Time

Important: Always carefully protect root access to your server as well as the passwords you assign to root, administrative, and user accounts.

When you ordered your server, you provided a username and password for your administrative user account. This account is the one you will use to connect to your server to perform administrative tasks.

Your administrative user is the primary user for managing your account. The administrative user enjoys email and FTP permissions as well as the ability to manage virtual user accounts. The administrative user manages FTP, Web, and email configurations. The administrative user is a member of the wheel group, which means that the administrative user can use the `su` command to become the root user.

When you connect to your server to perform administrative tasks, always connect using a secure protocol such as Secure Shell (SSH), SFTP, or Secure Copy (SCP). Avoid connecting to your server directly as the root user, and never use an insecure protocol when doing so.

A successful login places you in the user's home directory. Only the user's files and directories are accessible here. To access the main server directories you will need to change your current directory to the server directory.

Keep in mind that the user `root` is the primary administrative user on your server. To modify many system files, including adding or modifying users, you must be root. Because root is such an important user with so much power, you should be especially careful about selecting a root password and maintaining its security. Only after you configure SSH keys are you able to connect directly to your server as the user root. Until then, any user who belongs to the wheel group, such as the *Administrative User* that was created when your server was provisioned, can connect to the server and then use the `su` command to become root. Never use an insecure protocol such as Telnet for administrative tasks. If you do, any non-encrypted data could be sniffed by malicious hackers. Because the root user should only be used for administrative purposes, root does not have email or Web permissions.

All users with shell access are able to change user identifications by means of the UNIX substitute user command (`su`). This enables authorized users to become the root user without being prompted to provide a password. Other users can do so only if they are able to provide a password.

Access Your Server

Shell provides a powerful tool for your server administration tasks. You have SSH access to your server. Your server benefits from a security hardened environment which ensures that your data is not compromised. Because SSH provides complete shell capability over a secure channel, it is the useful tool for managing your server. While SSH is preferable to Telnet, most operating systems include a Telnet client. Your shell login also includes a built-in Telnet client program.

Once you have located an SSH client, connecting to your server requires you to specify a remote host. Your remote host is your server, so you would specify your domain name, your temporary domain (if applicable), or your IP address.

At some point, you are prompted for your login name and login password. Use the login name and login password when you ordered your server. After the login process is successful, you will have gained access to your server and can now issue commands at the command prompt.

Follow these steps to access your server by means of SSH:

1. Log into your server by means of SSH. For example, Connect to a server named *example.example.net* by issuing the address, as follows:
ssh root@example.example.net
2. Once you have accessed the server, show existing accounts by issuing the following command:
vlist -a
3. Use an Internet browser to access Web sites provisioned on the account, as follows:
http://example.example.net or *http://ip.address/*

Creating and Editing User Accounts

Your server enables you to create new users by manually editing the files that contain user information. To make the task easier, your server supports commands which guide you through the process.

The `vadduser` command is a custom script with which to add user accounts. If you are not familiar with the command, refer to the manual pages (`#man vadduser`).

To run the `vadduser` command, connect to your server by means of SSH and then type `vadduser` at the command prompt. The on screen instructions prompt you for the required information.

The `vedituser` command is a custom script that modifies an existing user account. It prompts you to modify the user information, including permissions and quota.

Because user account information is stored in several locations, including in hashed databases, it is important to use the tools listed above, rather than attempting to modify account information by editing the files directly.

When a user account is no longer needed, remove the account using the `vrmsuser` command. This gives you the option to keep or remove the home directory as well. Do not use this command to disable a user who you intend to reestablish at a later time. In those cases, it is better to change the password or to disable a user's privileges.

User information is stored in several different files on your server. First, the `/etc/passwd` file contains a list of user names, along with some account information. The following is a sample entry for the user `test`:

```
test:*:1001:1001:Test User Account:/home/test:/usr/local/bin/tcsh
```

The entry contains seven fields in a colon (`:`) delimited list. The first field is the username, followed by an asterisk (`*`), which represents the password. As a security measure, passwords are not actually stored in the `/etc/passwd` file, so you see an asterisk instead. Next are two

numbers, the User ID number and the Group ID number. These are used by the account to track file access and ownership rights. After the numbers, the *real name* or a description of the user account, followed by the user's home directory, and finally the shell they are allowed to use.

User passwords are stored in a hash format in the `spwd.db/master.passwd` file. This file is similar to the `passwd` file, although there are a few extra fields that the system uses.

Additional user information is stored in files such as `/etc/group` and `/quota.user`.

Administrators can view users and user quota information with the `vlistuser` command. It displays a list of all the user accounts (excluding the system users).

Configure Virtual Sub Hosts

Virtual sub hosting is one of the most powerful features of your server and the Apache HTTP Server. This feature enables you to support multiple domain names that each resolve to their own unique subdirectories on a single Account. You can host *example1.com* and *example2.com* on the same account, each with its own domain name and unique site content. Provide each virtual sub host customer their own unique FTP login with access to their own subdirectory and email addresses using their own domain name.

Create a Virtual Host

The `vaddhost` utility is an interactive, command-line program that automates the process of configuring virtual sub hosts. After launching `vaddhost`, it will ask you several questions about the configuration of your virtual sub host and provide you with default responses. As you answer each question, `vaddhost` will display the Virtual Host definition with each new piece of information.

Once you have responded to all questions, `vaddhost` will create necessary directories, add the virtual host entry to your main Web server configuration file (`/www/conf/httpd.conf`), and create a backup of your old `/www/conf/httpd.conf` file in your `/www/conf` directory. Remove these backup files at your discretion.

Note: If you replaced the default (`/www/conf/httpd.conf`) and it does not already have the `NameVirtualHost` directive, you will need to add it before adding any virtual sub hosts.

To run the `vaddhost` command, connect to your server by means of SSH and follow these steps:

1. Run the `vaddhost` command.
2. Specify one or more domain names for each virtual sub host definition. Typically, Virtual Host Names will at the very least include `www.SUBHOST-DOMAIN.NAME` and `SUBHOST-DOMAIN.NAME`.
3. Enter the administrative email address for the virtual sub host. This identifies the person responsible for the virtual sub host Web site. If the email address you specify is an email user account, run the `vadduser` command to add the email account separately.

Default Applications for Your Server

The operating system of your server supports the FreeBSD Ports and Packages (<http://www.freebsd.org/ports/>).

In addition, your server includes default applications. The following table describes the applications which are installed by default in the configuration of your server. In addition, the table provides an overview of the usage of the application.

Application	Usage
Apache DSO modules	Apache DSO modules are dynamic stored objects which are written to comply with the Apache API specification and can be loaded into the Apache Web Server. Apache modules can be loaded in one the following ways. The modules are dynamically loaded in the Web server configuration file.
Apache Web Server	Apache HTTP (or <i>Web</i>) Server and Web Server Modules provided by the Apache Hypertext Transfer Protocol (HTTP) Server.
Autoresponder support	Autoresponder provides an email alias which executes a program that automatically replies to any email sent to the specified address.
FTP server and users	ProFTPD provides anonymous configuration and support for FTPS (<i>FTP/SSL</i>) which includes Transport Layer Security (TLS) protocols such as anonymous FTP and FTP server processes. Your account supports unlimited FTP users.
IMAP server and email accounts	Your account uses University of Washington IMAP (UW IMAP) software. By default, your account also supports unlimited POP/IMAP email accounts.
OpenSSH	OpenSSH provides connectivity tools that encrypt all traffic (including passwords) to eliminate eavesdropping, connection hijacking, and other attacks. OpenSSH also supports secure tunneling capabilities and several authentication methods, and supports commonly used versions of the SSH protocol.
Perl and mod_perl	Perl provides a cross-platform programming language.
POP3 server and email accounts	Your account supports Post Office Protocol, version three (POP3) services for the purpose of handling email. Also, unlimited POP email accounts are supported by default.
Python	Python provides a dynamic programming language for your server.
Ruby	Ruby provides interpreted scripting language for object-oriented programming.

Sendmail SMTP Server	Sendmail with unlimited aliases provides support for the Simple Mail Transfer Protocol (SMTP).
SSL and SSL mail encryption	Privacy and encryption provided by support for the Secure Sockets Layer (SSL) protocol. Your account defaults to utilize a shared SSL certificate.
Unauthorized relay protection	Your account supports the SSH service under <code>inetd</code> and enables the ability to limit the rate of incoming SSH connections to eliminate this additional load. If you modify your <code>inetd</code> configuration, verify your configuration continues to work with the new configuration.

Apache HTTP Server

As a core service, your server supports the Apache HTTP Server, version 2.x. The open-source software is distributed by the Apache Software Foundation (<http://www.apache.org/>), under the terms of the Apache License. Apache HTTP Server maintains ongoing compliance with the HTTP standard which provides an application-level protocol for distributed, collaborative, hypermedia information systems.

Autoresponder

Note: The information included in this document applies only if you have not installed the CPX: Control Panel Web interface. If you have installed the Web interface, refer to the release notes and users guides for CPX: Control Panel.

Automatic responses to electronic email (also referred to as *autoresponder* and *autoreply*) are part of the core services of your account and installed by default on your server. The autoresponder provides an email alias which executes a program that automatically replies to any email sent to it. You can configure the email address to automatic reply with any message in the specified text, such as an FAQ, some marketing information, or a product list.

To configure an autoresponder, create and store an appropriate reply message named `/.autoreply` in the user's home directory. You can use an online file editor, such as Pico, vi, and Emacs. Or you can transfer the file from your local system in order to add the alias. If you transfer the file, be sure to download and upload the `/.autoreply` file in ASCII mode. For example, you might establish a user named *information* and create an automated reply for the email address associated with that user (*information@example.com*):

Thanks for requesting information about *Example* products and services. One of our capable representatives will be in contact with you within 24-hours. In the meantime, do not hesitate to refer to the frequently-asked questions on the <http://www.example.com/faq> Web page.

After you have created and stored the message for a user, add information to the `/etc/aliases` file. This creates an automated reply for *information@example.com*, as in the following example:

```
info:you@example.com, "|usr/local/bin/autoreply -f info-reply -a info"
```

Your server provides a `vnewaliases` command which updates the `/etc/aliases.db` file, as in the following example:

```
# vnewaliases
```

When your server receives email at *info@example.com*, your server sends an automated reply containing the message you stored in the `/.autoreply` file. In the previous example, email sent to *info@example.com* will also be sent to *you@example.com*. Without the

you@example.com, mail from the customer would not be sent to *you@example.com* and you would then need to assure that you (or someone you assign) check the email address *information@example.com*.

In the previous example, two optional arguments are added to the information included in the `/etc/aliases` file. Following is a full list of the available options:

- The `-m` option specifies a different message file (for example, `autoreply -m /etc/mymessage`). Be sure you use the full path to the user's home directory.
- The `-f` option allows you to change who the autoreply message will be from (in the previous example the `From:` field the customer gets will read *info-reply@example.com*).

Note: When creating an autoreply, make sure to make the `From:` address different than the autoreply recipient name. This prevents your autoreply from getting caught in an *autoreply loop* with another autoresponder.

- The `-a` option specifies a user that an autoreply can reply for. The user specified should be the same as the user configured for the autoreply (for example, `info: ... -a info`).
- The `-h` option can be added to an autoreply to turn off X-info headers.

For more information, refer to Recommendations for Automatic Responses to Electronic Mail published as an Internet Engineering Task Force Request for Comment (<http://www.faqs.org/rfcs/rfc3834.html>).

FTP

FTP enables you to copy files from one computer to another. As a core service, your private server supports ProFTPD with the Transport Layer Security (TLS) protocol as well as anonymous configuration for unlimited users. The software is installed by default as a core service your server provides. Your server's FTP services are secure and configurable. The software is distributed by the ProFTPD Project (<http://www.proftpd.org>) and is available for free under the terms of the GPL. As you configure ProFTPD, you must implement only the application features supported by the current release.

To use FTP to transfer files between your private server and your own local computer system, you must have an FTP client (or *program*) installed on your local computer system.

For your private server, configure ProFTPD to suit your use of the software. The ProFTPD configuration file is located at the following location:

```
/etc/proftpd.conf
```

Use an online file editor or transfer the file to your local computer system to make any configuration changes. ProFTPD runs as a daemon on your private server. The software reads its configuration file each time a process is spawned.

IMAP

By default, as a core service, your server provides IMAP server processes. You can add as many IMAP users as you want to. The protocol provides a method of accessing email or other kinds of electronic messages. IMAP enables the messages to be stored on a mail server. The protocol enables a client email program to access remote messages in a manner that appears to the user to be local. For example, the Mail user can manipulate email stored on an IMAP server from your personal computer at home, your workstation, and your notebook computer. The protocol enables you to do this without storing the mail on your local computer. This way, you can access the mail from any location.

- **Email address** — This is often listed as *IMAP Account, return address, or reply address*, and some programs may request this more than once. In every case, however, this is simply your username at your host domain (such as *username@example.com*).

- **Username** — Also often called *IMAP ID* or *Account Name*, this is your username.
- **Password** — This is the password associated with your username. Some programs do not ask for the password until you check your mail.
- **Incoming mail server** — This phrase (or a similar one) refers to the domain name where your mail is stored. Your mail is stored on your server; enter your host domain name.

For more information about IMAP, refer to the IMAP Connection (<http://www.imap.org/>).

OpenSSH

OpenSSH software is installed by default as a core service which provides SSH connectivity tools. The software gives your server an alternative to Telnet, rlogin, and FTP. Other connectivity tools can transmit passwords across the Internet unencrypted.. OpenSSH encrypts all traffic, including your passwords. The software reduces the possibility of successful eavesdropping and connection hijacking. OpenSSH supports all SSH protocol versions. OpenSSH (<http://www.openssh.com/>) is developed by and distributed under the terms of the OpenBSD Project (<http://www.openbsd.org/>).

Securing Root Access by Means of SSH

Note: Verify you have an alternate method for connecting to root (while testing access using SSH keys).

The most important security measure you can take for your server to prevent unauthorized access to the root (or *superuser*) user. As you establish security for your server, follow these general guidelines:

- Use only secure tools such as OpenConnect to access root or administrative user accounts.
- Apply passwords for root and administrative users which are strong and difficult to surmise. Protect the root password, in particular, as much as possible.
- Closely monitor root and administrative user accounts.
- Remove shell access from any user who does not require it.
- Require that any user who does require shell access always connects securely (by means of SSH).

Access Authorized SSH Keys

Remote root shell access may be available over the network by means of SSH, but only by using an SSH public/private key pair. Password logins as root are disallowed. Follow these steps to configure this method to create public/private key pairs.

First, create a public/private key pair by running this command, as an ordinary user:

```
%ssh-keygen -t dsa
```

The command prompts you for information which you should provide. During key creation, you are prompted for a passphrase which is used to protect the private key from unauthorized use. The guidelines for choosing strong passphrases are the same as those for choosing strong passwords: use a mix of upper- and lower-case letters, numbers, and symbols.

Passphrases must be 10 characters or more in length and unique from your root password.

Two files will result: the private key and the public key. The following are the default values for OpenSSH and the FreeBSD UNIX operating system:

```
id_dsa – private key
```

```
id_dsa.pub – public key
```

After you have created a private/public key pair, you must secure the private key on the client. Secure the private key from outside access, and place it where the SSH client program can access it, depending on which client is being used. For FreeBSD UNIX, the default location for the private key follows:

```
$HOME/.ssh/id_dsa
```

Cut and paste the contents of the public key file into your server's `authorized_keys` file. Verify that the key is on a single line. (Line breaks inside the key prevent it from being recognized.) The `authorized_keys` file can contain multiple keys, one key per line. Also, verify `/etc/ssh/sshd_config` includes a `PermitRootLogin` set as follows (the default):
`PermitRootLogin without-password`

If the `PermitRootLogin` line is not set as in the previous example, login as the root user and edit the line. After you have edited the line, restart SSHD.

Take great care regarding who has access to your private key and whose key is stored on your server. Any key in the `/root/.ssh/authorized_keys` provides access to your root login account.

Perl

By default, Perl is pre-installed on your server as a core service. Your server supports Perl (<http://www.perl.org/>), the widely-used, open-source cross platform programming language distributed with most UNIX systems. As you configure Perl, you must implement only the application features supported by the current, stable production release. The performance of the CPX: Control Panel is dependent upon support for Perl Modules.

POP 3 Server

By default, your server supports the Post Office Protocol, version three (*POP3*) to enable access and retrieval of email stored remotely on your server. The POP3 mail client enables you do download email from your server to a local computer system. Other protocols, such as IMAP, leave the email on the server rather than downloading it to the local computer system. When you configure email settings, each program may be different. However, most email programs require some of the same basic information, as follows:

- **Email address** — This is often listed as *POP account* or *IMAP Account*, return address, or reply address, and some programs may request this more than once. In every case, however, this is simply your username at your host domain (such as *username@example.com*).
- **Username** — Also often called POP ID or Account Name, this is your username.
- **Password** — This is the password associated with your username. Some programs do not ask for the password until you check your mail.
- **Incoming mail server** — This phrase (or a similar one) refers to the domain name where your mail is stored. Your mail is stored on your server; enter your Host domain name.

Compare POP to other protocols which are also supported by your server. For example see “IMAP” on page 12.

Python Programming Language

Python is a programming language comparable to Tcl and Perl. The FreeBSD UNIX operating system supports the current production (or *stable*) version of Python. The software is distributed for free by Python Software Foundation (<http://www.python.org/psf/>) under the terms of the Python license. Although the software is pre-installed on your server, as you

configure Python, you must implement only the application features supported by the current production release.

There are custom installation utilities for Python (`vininstall python`, `vininstall python-2`).

Ruby Scripting Language

Ruby (<http://www.ruby-lang.org/en/>) is an open-source interpreted scripting language primarily developed on the FreeBSD UNIX operating system. It is available for free under the terms of the GPL. Your server supports the current, stable release. As you configure Ruby, you must implement only the application features supported by the current, stable production release.

Embed Ruby Code

Your server supports the eRuby implementation as an add-on feature. The implementation enables you to embed a Ruby code to a HTML file. To install eRuby, connect to your server by means of SSH, `su` to root, and run the following command:

```
# vininstall eruby
```

Refer to FreeBSD UNIX Man Pages regarding eRuby by typing the following during an SSH session with your server:

```
# man eruby
```

eRuby is developed, maintained, and distributed by the Apache/Ruby integration project (<http://modruby.net>) under project's terms.

Sendmail SMTP Server

By default, the Sendmail SMTP server (<http://Sendmail.org/>) is installed as a core service. The SMTP server manages FreeBSD MPS v3 email services. It processes all incoming and outgoing messages for all user accounts with email permissions. In order to check or send email from a remote client (such as Outlook or Eudora) the user must have POP or IMAP permissions.

As a daemon, the Sendmail program is always running on your server. For Sendmail to run correctly there should be two daemons as well as the Simple Authentication and Security Layer authentication daemon (referred to as *saslauthd*) running. When you make changes to any of the Sendmail configurations you must restart Sendmail to load the new settings. To restart Sendmail, you must be the root user. Connect to your server by means of SSH, `su` to root, and run the following command:

```
# restart_sendmail
```

Note: There are several other Sendmail-related commands you can use, as follows:

```
sendmailctl stop
sendmailctl start
sendmailctl restart
```

You must change to the `/etc/mail` directory and run the following command:

```
# make restart
```

You must add `start` and `stop` to the command, as in the following examples:

```
# make restart stop
# make restart start
```

A `check_sendmail` tool displays the status of the two Sendmail daemons and the SASL authentication (`saslauthd`) daemon. For more information on the SASL authentication daemon refer to the FreeBSD UNIX Man Pages by issuing the following command:

```
# man saslauthd
```

A `check_sendmail` tool restarts any of these daemons if they are not running. To run the `check_sendmail` tool run the following command:

```
# check_sendmail
```

You can also use the `cron` program scheduler to specify that the `check_sendmail` tool runs to at regular intervals. Following is an example of an entry for a `crontab` file:

```
*/30 * * * * /usr/local/sbin/check_sendmail
```

The previous example indicates that the server runs the command `/usr/local/sbin/check_sendmail` every thirty minutes.

When an email message arrives to be processed by your server, Sendmail checks the incoming domain and determines if it is either a local domain or an authorized relay domain. In addition to your server's hostname (or primary domain) you can have any number of local domains on your server. In order to be considered local you must have the domain listed in the `/etc/mail/local-host-names` file on your server. Once a domain is determined to be local, Sendmail checks your server's `virtusertable` and then the aliases to determine if there are any special delivery conditions for the recipient address.

Once an email recipient has been checked against the `virtusertable` for domain based delivery, the domain name of the recipient is ignored by Sendmail. The username, without the domain, is then checked against the aliases, and finally delivered to the correct local user's mailbox.

You can configure your server to relay email messages for authorized domains. This allows you to use your server as a secondary mail server. When an email message arrives with a recipient domain that is listed in the `/etc/mail/relay-domains` file, Sendmail attempts to deliver the message to the primary server. If the primary server is unable to accept the mail, Sendmail holds the messages in a special queue, and delivers them to the primary mail server once the primary server is online again. This feature is especially useful to companies using exchange servers, or who have limited network access to their primary mail server.

While there are many possible reasons to filter mail, a common use is to prevent spam. In addition to recipient based email routing, you can configure Sendmail to filter or route messages based on other elements of the message. The easiest way to do this is with Procmail, an easy to configure mail filtering add-on. For more information, see "Procmail" on page 36. You can also configure Sendmail to filter by using the M4 macro set (http://www.sendmail.org/m4/intro_m4.html).

SSL

Your server supports the privacy and encryption provided by the Secure Sockets Layer (SSL) protocol. You can also change operating system and maintain SSL support, move a certificate to a new server, and renew a custom digital certificate.

Create a Signing Request and Private Key

To obtain a signed Digital Certificate you must create a Certificate Signing Request (CSR). At the same time your CSR is created, you will also generate a Private Key. The CSR is used by the signing authority to create a signed digital certificate which works with your Private Key to provide secure access to your Web site. There is some necessary information that you gather before generating the CSR and Private Key.

The following information is required as part of the CSR and must be entered exactly as you want them to appear in your certificate:

- PEM Passphrase – This is a security phrase which, like a password, ensures that only you can use your digital certificate. Be sure to use a phrase which you can easily remember but which is not easily guessed. Enter the passphrase in the future to install your signed certificate.
- Company Location – Know the country, province or state, and city where you want the certificate to display as your company location.
- Company Contact Information – This includes the complete company or organization name and the organizational unit or department (if applicable).
- Your Domain Name – Determine the exact domain name you want to use to access your Web site securely.
- Contact Email Address – The contact email address that you want to have the signing authority use when corresponding with you.
- Extra Information – This information can include a challenge password which some signing authorities use to allow you access to your certificate and which they require when interacting with them. You can also enter additional company information.

Connect to your server by means of SSH and run the following command:

```
# mkdir /usr/local/certs
# cd /usr/local/certs
# openssl req -new
```

You are prompted to provide the information you gathered earlier. *Common name* refers to the domain name that you want to use when you access your site using SSL. For example, `domain.com`, `www.domain.com`, `cname.domain.com`, or `*.domain.com`. The domain must be used exactly as it appears in the certificate.

When you have entered all the data, your CSR is shown. It is a good idea to save the CSR by copying and pasting it exactly as it appears on the screen, with line breaks and no extra lines before or after into a file on your local computer. You will need it when you are ordering your SSL certificate from a signing authority's Web site. The following is an example of a CSR.

```
-----BEGIN CERTIFICATE REQUEST-----
MIIB2jCCAUMCAQAwgYExCzAJBgNVBAYTAlVTMQ0wCwYDVQQIEwRVdGFoMQ4wDAYD
VQQHEwVQcm92bzETMBEGA1UEChMKU3R1bmt3b3JrczEVMBMGAlUEAxMMTWfYayBT
cGVuY2VyMScwJQYJKoZIhvcNAQkBFhh3ZWJtYXN0ZXJAc3R1bmt3b3Jrcy5jb20w
gZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAKIkMHnII4uNDwgTYsBYdiOBLY
NOsTfXp/5sG1VXj1YhDMoLzWxBbaulx2hEufj1Sfkm65MrD8j4nMFVIGf1sGnFCj
ClgxQ/5DJtV22jgnqQfKq7se32r9INoPWjFfjD1JC+4zry5LRiSPNImCYq2E1578
h6S6i6auDlnTDD0LAgMBAAGgGDAWBgkqhkiG9w0BCQcxCRMHZ3JvYmtpbWpANBgkq
hkiG9w0BAQQFAAOBgQANwQ7wudkfKxrrZA41XbOYeXWLnGhtNdzPJ8WyzOjGof4h
jkpDPV6SJqHEszpmZljEqb6fxgeiM4cpWSFGJAlQNFz+Ra8/msrLLBMM+zPuHpeR
OPFCsrIErmaBgnmymGok/DiHvhV+LqCkAgjcS2Kpn0cOy8KRyXzUc4k+TTw0Uw==
-----end CERTIFICATE REQUEST-----
```

In the directory where you ran the `openssl` command you will also find a new file called *privkey.pm*. This is your private key which you will need at a later time.

The following is an example of a private key:

```
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4, ENCRYPTED
DEK-Info: DES-EDE3-CBC, BCC23A5E16582F3D

hfWypkea3gnVCHCZJ / zgQpCH9RZF7WjYXGYohdbfkJY0ETLwXaqjvnnHQ1LomwIt
CvAzXhq8wnHur6SK21SO0ry3aSCvrBezH99miSJvtnt0HVlRJDnvaYQDbe01Z26D
hy2Yqha56Z8pvrTTolJfNL0sW4ewdws1wR4kxYDYkpusoe / Wed9Wg+i6xr9YmIjT
le9bbQ1PK2D / 3gJDhWW / aZHiMmLcYJtmWmf0wUMdmlibWYuq0UH1EefiLq3SLKK2
izvYpWDGhxVgtmzupvoc2E6CS3rQeRN3QQ9RqhzqdGqP8Xy / x11LMuDRUbPY54Kp
3a4gqZCXdlxctK70XX5TdhimsFEb5LlwA8CsnKE69nzs8MOLiz6mjtAhGB6KVKb4
dod3Wn6z20cus21SY5LxFkf6JZrAsqSZFzETN9n2Fbel2pTp3IRWx7Q+WBTlrME
uIMgUSKszpvgzg0Tf2Kxfw6Yw15EpEGA8PeiGrM1NeT2TftgiQBRQdAy7TQxgBlF
LOW2r5 / 1347ZgafacXLzpDBHnQrn / OtZijzleeoIwcvVwCOKz1oufEAN1ZTJbG6F
WYJuFt fopM5swoUYK3JgT582ziAeu4jcPdrNHCxqcInkNG+ib3dHdy8yccWRehD
VnSX2hr1MDd2cpFFT177Bc2 / neNyUieqiHkrTOZicD9oBSxFd0fP9QxLWEMCDWHT
N5UK1n29+TFgm / aXjZNjsIE5DSjTTBGTY2fPwtnefQaFk23ppV5VQypmZjxcwt2f
Eekjhl1vEiQChKULQCXFAaxL61HvBRqe3iJwJ+niOBuGpYnjdc80oIA==
-----END RSA PRIVATE KEY-----
```

Custom Digital Certificate

The Default Certificate is a generic way to provide secure access to your server. However, if you want to use your own domain name to provide secure access to your server get a custom digital certificate. This not only provides secure access to your Virtual Server, but provides an additional level of customer confidence by using your own domain name in the secure area of your site.

Obtain a Signed Digital Certificate

Once you have created a CSR, decide what signing authority and digital certificate to use.

There are a large number of different signing authorities. Each one offers several different types of digital certificated that have different capabilities and options associated with it. It is very important you select the certificate that best suits your needs. Because most signing authorities also sign additional types of certificates and products, verify that you are obtaining an SSL digital certificate.

There are a number of signing authorities, each with different methods for verifying your company's authenticity and with different levels of customer awareness and trust. The following is a list of a few of the signing authorities.

- GeoTrust
- GlobalSign
- VeriSign
- Thawte

When you have decided which signing authority and SSL Certificate type you want, and have created a CSR, you are ready to order your signed certificate.

The ordering process for obtaining a signed digital certificate is different for each vendor and certificate type. There are, however, some things that will remain the same throughout all of them. The following is a list of useful tips for ordering your certificate.

At some point in the ordering process, you are asked for a Server Type or the Server Software you are running; when this occurs, select Apache-SSL or Apache with OpenSSL.

When you are prompted to enter the CSR, be sure to paste it exactly as it appeared on the screen when you generated it, including the first (*BEGIN CERTIFICATE*) and last (*END CERTIFICATE*) lines.

An example of a certificate signing request appears as follows:

```
-----BEGIN CERTIFICATE REQUEST-----
MIIB2jCCAUMCAQAwYExCZAJBgNVBAYTAlVTMQ0wCwYDVQQIEwRVdGFoMQ4wDAYD
VQQHEwVQcm92bzETMBEGA1UEChMKU3R1bmt3b3JrczEVMBMGAlUEAxMMTWfYayBT
cGvUy2VyMScwJQYJKoZIhvcNAQkBFhh3ZWJtYXN0ZXJAc3R1bmt3b3Jrcy5jb20w
gZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBAKIkMHnII4uNDwgTYsBYdiIOBLTY
NOsTfXp/5sG1VXj1YhDMoLzWxBbaulx2hEufj1Sfkm65Mrd8j4nMFVIGf1sGnFCj
ClgxQ/5DJtV22jgnqQfKq7se32r9INoPWjFfjD1JC+4zry5LRiSPNImCYq2E1578
h6S6i6auDlnTDD0LAgMBAAGGDAWBgkqhkiG9w0BCQcxCRMHZ3JvYmxbpJANBgkq
hkiG9w0BAQQFAAOBgQANwQ7wudkfkxrrZA4lXbOYeXWLnGhtNdZPJ8WyzOjGof4h
jKpDPV6SjQHESzpmZ1jEqb6fxgeiM4cpWSFGJA1QNFz+Ra8/msrLLBMM+zPuHpeR
OPFCsrIermaBgnmymGOk/DiHvhV+LqCkAgjcs2Kpn0cOy8KRyXzUc4k+TTw0Uw==
-----END CERTIFICATE REQUEST-----
```

You are required to enter information about your company, including the official company name and address.

After you have ordered your certificate and sent in the requested documents, the signing authority will issue you a signed certificate. Once you have your signed certificate, you can install your signed digital certificate.

Install your Custom Digital Certificate

Once you have obtained a signed digital certificate, install it and configure SSL to use your certificate and private key instead of the default.

When you got your certificate, you most likely saved it to a file on your local computer. Copy the file onto your server by means of SCP. Be sure to copy the file using ASCII format to avoid corrupting the file.

Once the certificate is on your server, get the Private Key, which you generated at the same time as you generated the CSR, and confirm it is in the `/usr/local/certs/` directory with the name `ssl.pk`. Verify to keep a copy of the Private Key in a different location as well so if you make a mistake you don't lose your Private Key. Create a directory on your server and store a copy of both your Private Key and the Certificate until you are certain that the new certificate is working properly.

Connect to your server by means of SSH and run the following:

```
# cd /usr/local/certs
# openssl rsa -in ssl.pk -out ssl.pk
```

The `openssl rsa` command removes the default encryption on your key, and makes it useable by the Apache HTTP server. Verify your Private Key has been decrypted or not by looking at the file. When your key is generated, the first few lines are similar to the following example:

```
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4, ENCRYPTED
DEK-Info: DES-EDE3-CBC, BCC23A5E16582F3D
hfWyPkea3gnVCHCZJ/zgQpCH9RZF7WjYXGYohdbfkJY0ETLwXaqjvvnNHQlLomwIt
```

After decrypting your key, the key changed as in the following example:

```
-----BEGIN RSA PRIVATE KEY-----
MIICXQIBAAKBgQCot9aa9R38QevFSWqU718VFxqEDcY4gJfdZ6sBy282jdgCVcwU
q92tQ5V3amQanoSIWxI/O9GYm5kJS03b2qGib2sqLiHZFav/brjL5IDFOMwCSTyp
00I9otCK72/rrxMl+Gt8b5saEiIdmG04ar9AM2DYYQCFKYR62mDZ7mRa6wIDAQAB
AoGBAJWyoCqblGhvgSeCdZwCK+ZFopRKuHcHu jeLtRKZk2rfPisMPLCUEdObJLJY
5ssrnUJzM+SBSf5TCN1Slj3dZg2NRBq+68L1dR+3voEWv2ebPhzicjw8110xuVoX
HbXhM052Bmhp8XWzd3VdKXyQuTQeh17F4R2o39r9vP88pGnRAkEA40xTu4p6gAxF
l4JwiqFeswdoq/jEj9KkKGy/wM4psGQqUrZwZgKmN+R1NpSRWcyohpS0sU8yFchb
bydNYvYj0wJBAMAHGQENrGx+3XEzcCx3uY8vvlgvCNFou0RKKcoaHyf8n028AJAF
ZAM/7h+cFcJVYEEb8n54ED4979c+gr3ttYkCQD444okVLAJUYSQhL6UKMzpvqEM6
1JW8/fc490sPnXTQoOy21030yarYppxsyTEAbvacDkV61S4zrNK5Gq1vzkUCQF45
0GVR7k92mPZZBSvsu5K1HTEKZlN7DpjdW0+2LZ+TaB/epnARlyN5FUFRd6PZ/Npm
```

```
fUDtbRr9jViTBdhocfECQQDfxT3bUNjvJUeWQieQg2ooj7yzbjMD5MjA+9z+qh1V
Cb+4kQSEWrP7EdJk4cOHOH+ZYjinf77x8v2PbnaKE5Dc
-----END RSA PRIVATE KEY-----
```

Edit your `/www/conf/httpd.conf` file to look for your certificate file by adding the following command:

```
SSLCertificateFile /usr/local/certs/example.com.pem
```

Once you have added the certificate directive to your `/www/conf/httpd.conf` file, issue `restart_apache` to make Apache start and utilize the new certificate.

Check to verify the new certificate is working by connecting to the domain your certificate is configured to use by means of HTTPS. For example, if the domain name were `www.example.com`, you would type `https://www.example.com` into your browser's location bar. If the page loads without any errors, find the lock icon on your browser and click (or possibly double-click) on it. This brings up the certificate information or a window that lets you view certificate information. Check that the certificate is using the correct domain name and has the correct information.

If you intend to use your SSL certificate with email as well, make links so that the POP and IMAP services are able to find the file as well:

```
# ln /usr/local/certs/example.com.pem /usr/local/certs/imapd.pem
# ln /usr/local/certs/example.com.pem /usr/local/certs/ipop3d.pem
```

Move your Custom SSL Certificate

If you are moving your secure Web site from one server to another, there are a few specific concerns to be aware of in order for the certificate to work on the new server.

Change Operating Systems

Digital certificates work differently with different operating systems and Web Server software. Because of this, a certificate generated for a Windows2000 server running the IIS Web server does not work on a FreeBSD UNIX server running Apache. Likewise, a FreeBSD UNIX server running Netscape Web Server can not use a certificate designed to operate on a FreeBSD UNIX server running Apache.

If your current certificate is not compatible with your new server, obtain a certificate for the new operating system and Web server. Most certificate authorities will issue a transfer certificate at a lesser cost than obtaining a new certificate.

The signing authority provides you with instructions on how to install a transfer certificate.

Move a Certificate to a New Server

If your current certificate is compatible with the server you are moving your secure Web site to, you do not need a new certificate. Simply move your certificate to the new server and ensure that it works.

1. Connect to your server by means of SSH and run the following command:


```
# mkdir /usr/local/certs
# cd /usr/local/certs
```
2. Using FTP or another method, copy the certificate and Private Key files to the new server. Copy the files to the `/usr/local/certs/` directory. The certificate is in a file named `ssl.cert`, and the key is in a file named `ssl.pk`. If you use FTP, be sure to copy the file using ASCII format to avoid corrupting the file.
3. Verify the Private Key has been decrypted by looking at the file. If the key has not been decrypted the first few lines appear as in the following example:


```
Proc-Type: 4, ENCRYPTED
DEK-Info: DES-EDE3-CBC, BCC23A5E16582F3D
```

4. To decrypt the key connect to your server by means of SSH and run the following commands:
cd /usr/local/certs
openssl rsa -in ssl.pk -out ssl.pk
Create a PEM file that contains both the certificate and key. To do this, run the following commands:
cd /usr/local/certs
cp ssl.pk YOUR-DOMAIN.NAME.pem
cat ssl.cert >> YOUR-DOMAIN.NAME.pem
5. Edit your /www/conf/httpd.conf file to look for your certificate file by adding the following command:
SSLCertificateFile /usr/local/certs/MY-DOMAIN.NAME.pem
6. Once you have added the certificate directive to your /www/conf/httpd.conf file, issue `restart_apache` to make Apache start using the new certificate.

Renew Custom digital certificates

Order signed digital certificates for periods of one to three years depending on the signing authority. It is important to renew digital certificates no less than 30 days prior to the expiration date to avoid any interruptions with your SSL Service. The renewal process is different for each vendor and certificate type.

After you have completed the renewal process, the signing authority will issue a new signed certificate. Once you have received the renewed certificate, replace the original certificate on your server, and restart Apache. Follow the instructions to install your signed digital certificate to complete this process.

Install Additional Features

There are additional features actively supported by your server. Most are offered without additional fees. Some, such as support multiple IP addresses, require you to pay additional fees. Many include `vinstall` and `vininstall` utilities you can use to more easily install, configure, and update your server. Further, server software updates continuously apply the latest stable versions of the features.

Accrisoft Freedom

Accrisoft Freedom (also referred to as *Accrisoft RBT*) provides you with a suite of tools to build and manage your Web sites. The Accrisoft suite is available as a fee-based, additional feature for your account. Once you purchase the suite and verify the installation, refer to Web-based information, documentation, and instructions provided with the purchase of the suite for more information.

Accrisoft Freedom is developed, maintained, and distributed by Accrisoft Corporation (<http://www.accrisoft.com/>), its partners, and resellers.

Apache DSO Modules

Apache Dynamic Server Object (DSO) modules are code segments that are written to comply with the Apache API specification and can be loaded into the Apache Web Server. Apache modules can be loaded in the following ways:

- Statically loaded in the compiled `httpd` daemon
- Dynamically loaded in the Web server configuration file

This modular design for adding Web server features gives Web administrators tremendous power and flexibility. A wide variety of Apache modules have been created supporting all kinds of exciting Web server features. Web server speed and efficiency is improved when using Apache modules since your Web server can internally process instruction sets rather than relying on external applications.

Dynamic module support is one of the key features of the Apache Web Server. The ability to dynamically load modules is known as DSO support. DSO allows you to extend the features and capabilities of Apache by adding the specific module you need, when you need it, without recompiling the Web server binary.

Note: If you try to load all the modules at the same time you will probably get a resource error. Simply load the modules you need one at a time.

Aspell

Aspell (<http://aspell.net/>) is an open-source command-line spell checker. It can either be used as a library or as an independent spell checker. Your server provides a `vinstall` utility for installation, re-installation, and upgrades of the spell checker. Aspell upgrades are included in Server Software Distributions and upgrades do not usually require your intervention. For more information refer to the Aspell Man Pages.

ClamAV

Your server supports Clam Antivirus (or *ClamAV*), a free, open-source virus scanner distributed by the ClamAV Team (<http://www.clamav.net/>), under the terms of the GPL. Your server provides a `vinstall` utility for the virus scanner (`vinstall clamav`).

Note: Do not use ClamAV to replace antivirus software on your local computer system. ClamAV is designed to supplement such programs and provide additional safeguards. It does not provide the antivirus capabilities such as protection from Web based or TCP/IP-based attacks. Only a local antivirus program installed to your computer system provides sufficient protection.

If you do not have Procmail installed on your server, the ClamAV installation script will install it and configure it as your local delivery agent (LDA). If you already have Procmail installed and have your own recipes in use, check your `/etc/procmailrc` file to see that the ClamAV configurations are in the proper order.

When ClamAV is installed, a table of utilities configured to operate in the background at regular intervals (or *crontab*) is added to the system to update your virus database twice daily using the ClamAV Freshclam program.

For more documentation of ClamAV, consult the `clamscan`, `clamd`, `freshclam`, and `clamav.conf` manual pages. Find documentation on the ClamAV Web site (<http://www.clamav.net/>).

CPX: Control Panel

Comparable to iManager, the CPX: Control Panel provides an intuitive Web interface to administer your server. The interface enables you to perform user and domain management tasks. It also provides a Web-based email interface and mail management modules and empowers virtual sub hosting on your server. It enables you to create domain administrators with user management control. This enables each sub host and its respective end users the ability to configure and control their own accounts.

CPX: Control Panel includes the following modules:

- File Management – This module enables you to navigate through directories, view and edit text files, download and upload files, create or delete files and directories, rename or move files and directories, and view and edit permissions.
- Webmail – An email management interface to read, store and compose email, manage folders, apply spam filters, store contact information, and manage automated replies (Autoreply).
- User Management – The user management module enables you to add or delete users, manage domain admin accounts, and view the status of user accounts.
- Domain Management – Manage your domains easily with the ability to add or delete sub hosts, specify limits on the number of users and email accounts, manage logs, and specify catchall email rules.
- Mail Management – This module provides the management of email to add or delete email aliases, edit account settings, or even configure broadcast lists.
- Profile and Preferences – Customize your settings to your personal preferences. Change your password, shell, and the date/time display for your server.

Note: Due to the high number of possible account configurations or modifications, there is no guarantee that CPX: Control Panel will perform reliably on previously configured accounts. CPX: Control Panel is designed and tested for new server configurations and a small number of existing configurations.

A CPX: Control Panel `vinstall` utility makes the following changes to your server:

- Upgrade of Perl.
- Installation of `mod_perl` and `mod_rewrite`.
- Installation of the CPX: Control Panel handler for `mod_perl`.

- Installation of ClamAV, SpamAssassin, and Procmail (configured as the Sendmail local delivery agent).
- Modification existing ClamAV and SpamAssassin installations.
- Install savelogs (or upgrade if previously installed).
- Initiation of the CPX: Control Panel daemon `vsapd`.
- Creation of `virtusertable` entries for existing mail users, as well as addition of default catchalls for all domains (as found in `/etc/mail/local-host-names`).

Follow these steps to install CPX: Control Panel on your server.

1. Connect to your server by means of SSH and issue this command:
`# vinstall cpx`
2. Access CPX: Control Panel by going to the following URL:
`https://YOUR-DOMAIN.NAME/ControlPanel/`

You control whether virtual users are enabled to use the Webmail and Profile/Preferences features of CPX: Control Panel. Add new users by using the CPX: Control Panel or by command line issuing the following command:

```
# vadduser --cpx
```

Email List Package

Automate the management of Internet email lists on your server by installing and utilizing Dada Mail, Majordomo, or Mailman.

Dada Mail

Your server supports the Dada Mail Web-based email list management system (<http://mojo.skazat.com/>) as an add-on feature. There is a `vinstall` utility to assist with the installation, re-installation, or upgrade of the management system.

Note: Dada Mail was formerly known as *Mojomail*.

To install Dada Mail, connect to your server by means of SSH, `su` to root, and run the following command:

```
# vinstall dada
```

During the installation, you are prompted to enter an administrator password, which you use when accessing the administration utility. You are also prompted to enter the domain name to use for the mailing lists.

Access the list administration utility accessing the following URL:

```
http://example.com/cgi-dada/mail.cgi
```

When you log in to the administration utility for the first time, you are prompted to create a new mailing list. After you have created a list, accessing the administration utility enables people to add themselves to one of the lists. You can also select Administration to manage lists and users, or to change the administrator password.

Majordomo

Note: Majordomo is best configured by administrators with advanced skills who carefully research the software capabilities before installing the feature.

Majordomo is community-supported software you use to automate the management of Internet email lists. The software is written in Perl and is compatible with the current, stable version of the language. Correct operations of the software on your server are dependent upon the versions of Majordomo, Perl, operating system software, as well as the email software (such as Sendmail) and the versions you are operating. Great Circle Associates (<http://www.greatcircle.com/majordomo/>) distributes the free software but offers no technical support.

Mailman

Your server supports Mailman, free software, distributed under the GNU General Public License. Mailman is written in the Python programming language the versions of the software and the programming language must both be stable, current versions installed on your server.

Email Service

As a core service, your server supports mail services by means of the Simple Mail Transfer Protocol (SMTP), Post Office Protocol (POP or *POP3*), and Internet Message Access Protocol (IMAP or, more precisely, *IMAP4*).

- SMTP provides a standard method to send email messages between servers.
- POP provides a standard method to retrieve email from a mail server.
- IMAP provides a standard method of accessing electronic mail or bulletin board messages kept on a shared mail server.

These standards are maintained and updated as Internet industry standards by the Internet Engineering Task Force (<http://www.ietf.org/>).

Expect

Expect (<http://expect.nist.gov/>), the UNIX automation and testing tool, enables your server to interact with other interactive programs according to a script. In addition, the user can take control and interact directly when desired, afterward returning control to the script. Your server provides a `vinstall` utility for Expect. The tool is maintained and distributed, with coordination with Don Libes, by the National Institute for Standards and Technology (NIST).

FormMail

FormMail is a CGI program designed to generate email based on the input from an HTML form.

Installing FormMail

To install the FormMail CGI on your server, connect to your server by means of SSH, `su` to root, and run the following command:

```
# vinstall formmail
```

This command installs three files, `FormMail.pl`, `FormMail.examples` and `FormMail.readme`, into your `/www/cgi-bin` directory. The examples and readme files contain various information and examples on using FormMail.

Set up the script to use your account information. Open the file `FormMail.pl` file and modify the following lines in the user configuration section.

- Find the `@referers` line and replace the information inside the parentheses with your own server's domain name(s) and IP address. You can leave the `localhost` value.
- In the `@allow_mail_to` line, remove the original email addresses and put either the domain, or a full email address for every account that should be allowed to receive email messages from this form. For security reasons, unless you have a large number of email accounts at a single domain, it is better to list the full address for each recipient.

Once you have modified these two fields, save the file.

Using FormMail

Create a form that you would like the contents mailed to some address. The form should include the following field (at the very least):

- `recipient` – specifies who mail is sent to

Other optional fields can also be used to enhance the operation of FormMail for you site, for example:

- `subject` – specify the subject included in email sent back to you.
- `email` – allow the user to specify a return email address.
- `realname` – allow the user to input their real name.
- `redirect` – URL of page to redirect to instead of echoing form input.
- `required` – list of field names that are required input (comma delimited).

Several other fields are supported. See the `FormMail.readme` file for a complete presentation of the supported fields.

The following is an example of HTML source markup:

```
<form method="POST" action="/cgi-bin/formmail.pl">
<input type="hidden" name="recipient"
  value="order@yourdomain.com">
<input type="hidden" name="subject"
  value="Order Request">
<input type="hidden" name="required"
  value="realname,email,phone">
Please Enter Your Name:<br>
<input name="realname" size="40">
<p>
Please Enter Your Email Address:<br>
<input name="email" size="40">
<p>
Please Enter Your Phone Number:<br>
<input name="phone" size="40">
<p>
<input type="submit" value="Submit">
<input type="reset" value="Reset">
</form>
```

Once your form is complete, you should be able to send email messages using it

FrontPage

FrontPage provides tool for Web pages designed and implemented with the Microsoft Web development software (<http://office.microsoft.com/en-us/frontpage/default.aspx>). In order to use Microsoft FrontPage in conjunction with your server, you must install the FrontPage Server Extensions. Your server supports the extensions and provides them without additional of charges. You can use a `vinstall` utility to ensure your server includes the extensions you need.

HTTP-Analyze

HTTP-Analyze is a Web Log Analyzer that watches the transfer log file of your Web server and creates a comprehensive summary report from the information found there. HTTP-analyze has been optimized to process large log files as fast as possible.

HTTP-Analyze is available from the FreeBSD Ports Collection in `/ports/www/http-analyze/` directory.

Connect to your server by means of SSH, `su` to root, and run the following command:

```
# cd /ports/www/http-analyze/  
# make install clean
```

HTTP-Analyze is installed into the `/usr/local/bin` directory of your server.

Run HTTP-Analyze using a configuration file or with options from the command line. The following is an example of how you could do this:

```
# http-analyze -vm -S YOUR-DOMAIN.NAME -o /www/htdocs/http-analyze  
/www/logs/access_log
```

The directory specified in the output path (`-o`) must exist. After running the command above you will find several pages of your server's Web statistics at the URL:

```
http://example.com/http-analyze/
```

Here is an explanation of the command line arguments used above and several others you may find useful. (Check the Man Page for HTTP-analyze for full usage information.)

<code>-h</code>	print the help list
<code>-d</code>	generate short statistics (default)
<code>-m</code>	generate full statistics (includes <code>-d</code>)
<code>-v</code>	verbose mode: comment ongoing processing
<code>-o outdir</code>	name of the directory for HTML output files
<code>-S srvname</code>	set server name (default: system name)

The final command line argument used in the previous example is the path and filename of the Web server access log file.

iManager Web-based Server Utility

Comparable to the CPX: Control Panel, iManager is a Web-based server utility, which enables you to manage many of the common tasks involved in server administration. In addition to basic user and subhost configuration tools, it includes an IMAP style email manager and an easy to use file manager. Your server provides a `vinstall` utility for iManager.

Java

Java technology, created and distributed by Sun Microsystems, offers many benefits to Internet and application programmers. A set of `vinstall` utilities includes the following Java applications:

- Java SE Development Kit (JDK)
- Java Runtime Environment (JRE)
- Java Sun Developer Kit (SDK)

MajorCool Web Interface Maintenance Tool

MajorCool (<http://tldp.org/>) is a web interface maintenance tool for Majordomo. There is a custom installation utility (`vinstall majorcool`) available for the tool. For more, see “Majordomo” on page 25.

Metamail

The Metamail (<http://packages.debian.org/stable/mail/metamail>) program reads a *mailcap* file to determine how to display non-text at the local site. Every mail-reading interface needs to call Metamail whenever non-text mail is being viewed, unless the mail is of a type that is already understood by the mail-reading program. There is a custom installation utility (`vinstall metamail`).

Multiple IP Addresses

By default, your private server is assigned a single Internet Protocol (IP) address. For some customers, a FreeBSD MPS v3 which is configured to utilize a single, base IP address provides all they need. However, you can now assign additional IP addresses. And you can now assign additional IP addresses for both new and existing servers.

How Multiple IP Addresses Work with Other Features

In order to provide support for the new multiple IP address feature, your server includes support for the following other features which are also compatible with assigning multiple IP addresses:

- Apache Hypertext Transfer Protocol (HTTP) Secure Server
- Dedicated Secure Socket Layer (SSL) Certificates
- Shared SSL Certificates
- Multiple SSL Certificates (on a standard port)
- Secure FTP
- Post Office Protocol (POP) over SSL
- POP email encryption
- Internet Message Access Protocol (IMAP) email encryption
- Sendmail mail transfer agent (MTA)

Overview of the New Multiple IP Address Feature

With the introduction of this new feature, you can assign additional IP addresses for your server v3 server. There is a monthly fee to associate each IP address with your account and the addresses are available individually, without any kind of bundling required. Following are the additional number of IP addresses, for each plan:

- **FreeBSD MPS v3 Basic** – You can now assign 29 additional IP addresses to your Basic server. This is in addition to the base IP address for a maximum of 30 IP addresses.
- **FreeBSD MPS v3 Pro** – You can now assign 39 additional IP addresses to your Pro server. This is in addition to the base IP address for a maximum of 40 IP addresses.
- **FreeBSD MPS v3 Pro Plus** – You can now assign 49 additional IP addresses to your Pro Plus server. This is in addition to the base IP address for a maximum of 50 IP addresses.

Potential Uses for Multiple IP Addresses

Use the Multiple IP address feature to specify more than one unique SSL certificate. This enables groups of customers to utilize the features offered by your server without visibility or compromise to other groups of customers. Following are some examples of groups which might require access to the same server but also require the separate, distinct authentication of unique SSL certificates:

- Internal employees, including administrators, who require access to an organization's intranet features.
- External clients, vendors, and contractors who require access to an organization's intranet or other Web content and features.
- The public which requires unfettered access to some portions of your organization's Web site but not to others.
- Customers who require access to retail e-commerce features.
- Sales representatives who require access to wholesale e-commerce features.
- Managers who require access to e-commerce (or other) statistics.
- Customers who are located in regions where a unique pricing or taxation structures apply.
- Customers you wish to offer products under several distinct brands.

How Your Server Utilizes Multiple IP Addresses

Once you configure your server to utilize multiple IP addresses, you can utilize a link from the account information interface. For accounts which utilize domains managed under the terms of `secure.net` name servers, you can manage Domain Name Service (DNS) for domains associated with the additional IP addresses. If you are a reseller, you can do this from the Reseller Backroom. In general, the services bind to all IP addresses. However, Apache and SSL recognize and operate using a specific IP address.

Overview of Configuring Multiple IP Addresses

The following provides an overview checklist of the tasks you must perform in order to utilize support for Multiple IP addresses.

- Set up DNS for additional IP addresses.
- Set domains for DNS services.

- Assign each IP addresses to a virtual host.
- Install a SSL certificate for a virtual host.

Managing Multiple IP Addresses, Subhosts, and Certificates

Prior to the introduction of support for multiple IP addresses, your server enabled configuration of multiple Web sites and domains in addition to the main domain of the server. (or *hostname*). The hostname and subhosts were associated with the single, base IP address for the account. You may have placed the Web content for your hostname in the `/usr/local/apache2/htdocs` directory. You may also have configured a custom hostname during the order process. To assist with the process of configuring and testing your server, all FreeBSD MPS v3 servers receive a temporary domain name (or *temp domain*) which resolves to your server. Use this domain if the custom hostname is temporarily inaccessible or does not yet resolve to your server. Other domains or sites hosted by your server are called *subhosts*. This section explains adding, removing, and configuring subhosts with the additional consideration of multiple IP addresses.

Because a standard, default FreeBSD MPS v3 server supports just one IP address, you can only associate one SSL certificate with the standard SSL port (443) for the Web server. You can, however, configure your Web server to use the Apache Listen directive to monitor other ports for SSL requests, and associate other certificates with these different ports. Doing this requires you to indicate the port number in the Universal Resource Locator (URL). With the addition of support for multiple IP addresses, this non-standard type of configuration is no longer necessary for those who purchase the use of additional IP addresses.

New and Updated Command-line Utilities

The assistance provided by the `vaddhost` command-line utility continues with the addition of prompts to enable you to associate a subhost with the base IP address or another IP address associated with your account. A command-line utility, `vaddcert`, is added to enable you to install certificates for different domains which can now utilize different IP addresses.

Note: To execute the `vaddhost` and `vaddcert` commands or to edit to the `httpd.conf` file, as instructed in the following sections, you must verify you are the root user. You can become the root user by typing `su -` at the command line and supplying the root user password. Also, you can press `ctrl+c` to exit the `vaddhost` or the `vaddcert` process at any time. This immediately cancels `vaddhost` and any subhost configuration entered during the `vaddhost` process is lost.

Adding a Subhost

The hostname or subhost typically consists of the top-level domain (*example.com*) only instead of a *canonical* name such as *www.example.com*. Canonical names are usually added as secondary domains or aliases. With the assistance of a command-line utility, you can configure subhosts (and canonical variations) to comply with the Apache `VirtualHost` directive. The Apache software looks for `VirtualHost` entries in the following file:

```
/usr/local/apache2/conf/httpd.conf
```

The `vaddhost` command assists you as you create a subhost configuration `VirtualHost` tags in your Apache configuration file. While the configuration task is presented in three sections you must complete all of the steps to complete the configuration of the subhost which complies with the Apache `VirtualHost` directive.

From your server's command-line interface, follow these steps to begin configuring the subhost. After you have completed these steps, you will have specified the domain and administrator:

Note: Throughout the following steps, the system periodically displays the list of domains, canonical variations, and IP addresses for verification.

1. Type `vaddhost` and press **Enter**. Instructions and information for `vaddhost` will display during this step and throughout the `vaddhost` process.
2. Type the domain for the subhost (such as *example.com*), any secondary domains (such as *www.example.com* or *store.example.com*), and any other domains used for this subhost, pressing **Enter** after each. The first domain entered will be the `ServerName` (or *main domain*) for the subhost. Additional variations will be aliases that point to the main domain.
(Or press **Enter** without any text after providing all variations to move to the next step. The system displays the list of domains and variations for verification.)
3. Type the username of the administrative user for the subhost and press **Enter**. This user should be the owner of the Web site files and folders; otherwise the Web server will not be able to load the site.
4. Verify the information and type **y** and press **Enter** to continue.
(Or type **n** and press **Enter** to input the username again.)
5. Type the IP address with which you wish to associate the subhost and press **Enter**.
(Press **Enter** prior to typing an IP address to see a list of available IP addresses.)
6. If the listings of domains, canonical variations, and IP addresses are correct, type **y** and press **Enter**.
(Or, type **n** and press **Enter** to input the information again.)

Administrative Email and Document Root settings

After you have added a subhost, continue the configuration and specify administrative email as well as document root (or *Web directory*) settings. Follow these steps from the command line:

1. Type the email address of the subhost administrator and press **Enter**.
2. Verify the information and type **y** and press **Enter** to continue.
(If the information is incorrect, type **n** and press **Enter** to input the address correctly.)
3. Type the path for the subhost Web directory, or document root, on the server. The `vaddhost` command simplifies this step and provides a recommended path for you. You can press **Enter** without typing a path to select this default and create a subhosted directory in the home directory of the user specified in the previous step.
4. Verify the information, type **y** and press **Enter** to continue.
(If the information is incorrect, type **n** and press **Enter** to input the path correctly.)

Log and cgi-bin Settings

After you have configured administrative email and document root settings, specify log and Common Gateway Interface Binaries (`cgi-bin`) settings. Follow these steps from the command line:

1. Select an option for the subhost transfer log and press **Enter**.
2. Verify the information, type **y**, and press **Enter** to continue.
(Or type **n** and press **Enter** to choose the transfer log configuration again.)
3. Select an option for the subhost error log and press **Enter**.
4. Verify the information, type **y**, and press **Enter** to continue.
(Or type **n** and press **Enter** to choose the error log configuration again.)
5. Select an option for the subhost `cgi-bin` and press **Enter**. This will enable the subhost to execute scripts and programs.

6. Verify the information and type **y** and press **Enter** to continue.
(Or type **n** and press **Enter** to choose the `cgi-bin` configuration again.)
7. The system will display the `VirtualHost` entry to be added to the `httpd.conf` file for confirmation. Type **y** and press **Enter** to add the entry to the `httpd.conf` file.
(Or type **n** and press **Enter** to halt the `vaddhost` process.)
8. If you typed **y** to accept the entry, type **y** and press **Enter** to restart the Web server and complete the subhost addition.

Assigning a New SSL Certificate

This release offers a new command-line utility (`vaddcert`) which enables you to assign a new SSL certificate to a host. Follow these steps to use `vaddcert` to assign a new SSL certificate.

1. Select the host to which the new SSL certificate will be assigned.
If the host is not listed, check the Apache configuration to verify that another SSL certificate is not previously assigned to the IP address and port. Also verify that the host's `SSLEngine` directive is set to 'on'.
2. Enter the file path of the SSL certificate file to be installed.
3. Enter the file path of the SSL certificate key file to be installed.
The following lines will be added to Apache configuration for `ServerName`.
`example.securesites.net`
`SSLCertificateFile /usr/local/apache2/conf/ssl2.crt`
`SSLCertificateKeyFile /usr/local/apache2/conf/ssl2.key`
4. If the information is correct, press **y** to continue.
5. Press **y** to restart Apache now.
A `Syntax OK` message is displayed.

Going Beyond the Basics

You may configure a subhost further by editing the `VirtualHost` entries for the subhost in the `/usr/local/apache2/conf/httpd.conf` file. Execute the `restart_apache` command from the command line after editing the file to restart the Web server and make the changes effective. In addition, refer to the customer documentation for information about using the full range of features for your server.

Your Responsible Use of IP Addresses

Note: FreeBSD MPS v3 supports IP version four (IP v4) and is available in the San Jose, California and Sterling, Virginia datacenters located in the United States.

All IP addresses are on loan from a Regional Internet Registry (RIR). The number of the IP addresses on loan can vary and is solely based on the requirements you demonstrate and document at the time you request them. Your name and justification for utilizing each IP address may be disclosed to certain registries, including, but not limited to, the American Registry of Internet Numbers (ARIN). For more information refer to the ARIN Web site (<http://www.arin.net/index.shtml>). The ARIN Web site includes a Search WHOIS feature.

The guidelines regarding your responsible use of IP addresses are offered with guidance from various Internet Engineering Task Force (IETF) documentation as well as the regional registries. The guidelines are subject to change in the future. For example, ordering systems for additional IP addresses will be updated to ease the burden of disclosing and demonstrating your requirement to use an IP address. All future updates to the policies and procedures will be based upon the following guidelines:

- **Conservation** – The objective distribution of globally unique IP address space according to the operational needs of customers. No stockpiling designed to maximize the utilization of IP address space is acceptable.
- **Registration** – ARIN requires information on which entity is using an IP address. This information includes: your name, company name (if a business), postal address, email address, IP address, and telephone number.
- **Routability** – The distribution of globally unique IP address space in a hierarchical manner, which permits scalability in the internet routing table.

Configure Provisioned IP Addresses Only

Caution: When configuring your subhosts and certificates, do not specify different IP addresses than the ones associated with your server. If you do, your subhosts and certificates will not function. This is true of any IP addresses, even those you might see displayed as unused IP addresses for the subnet of your server. Your server will not function with any other IP addresses and you will not be able to see information about any aspect of the data traffic associated with any other IP addresses.

The FreeBSD operating system supports a standard, UNIX command which enables you to see the status of network interfaces for your server. Use the command for diagnostic and configuration tasks. For example, if you issue `ifconfig` command without any additional argument, all of the currently active interfaces are displayed. If you run the command with the additional argument of `-a`, all of the interfaces, including inactive ones are displayed. Beyond the simple display of interfaces, the command includes additional arguments which enable you to specify IP addresses for each of the network interfaces. If you do specify another address in error, the remedy is to log into your server as root and use the `ifconfig` command to specify a correct, provisioned IP addresses. For more details about the `ifconfig` command, refer to the FreeBSD operating system *Man Pages*.

CPX: Control Panel and Multiple IP Addresses

You can configure Multiple IP Address settings with the assistance of the command-line interface your server offers. The ability to utilize the CPX: Control Panel to configure multiple IP address features is coming soon.

MySQL

Your server supports the current, stable release of MySQL, an open source database server and tool distributed under the terms of the GPL.

To use the MySQL client, connect to your server by means of SSH and run the following command:

```
% /usr/local/mysql/bin/mysql -u root
```

This command will start the MySQL client as the root user. Add more users by following the directions in the *MySQL Reference Manual* or another, reliable MySQL resource.

To make starting MySQL easier, create a file with all your start-up options instead of having to type in all the different flags at the command prompt. To do this, create a file in your `/etc/` directory named `my.cnf`. The contents of the file would appear as follows if you wanted MySQL to report error messages in Japanese:

```
[mysqld]
language = japanese
default-character-set = ujis
```

Access manual pages by typing the following during an SSH session with your server:

```
% man mysql
```

For more information, refer to the MySQL Developer Zone Web site (<http://dev.mysql.com/doc/>).

Namazu

Your server supports Namazu, an open-source, full-text search engine maintained by the Namazu Project (<http://www.namazu.org/>). The software is available for free under the terms of the GPL. Your server includes a `vinstall` utility for the search engine software.

Open WebMail

Your server provides support for Open WebMail (<http://openwebmail.org/>), a Webmail system designed to manage very large mail folder files in a memory efficient way. It also provides a range of features to help users migrate smoothly from Microsoft Outlook to Open WebMail. Your server provides a custom installation utility (`vinstall openwebmail`).

PHP

Your server supports PHP: Hypertext Preprocessor (<http://www.php.net/>), the widely-used, general-purpose, and open-source scripting language distributed with most UNIX binaries. As you configure PHP, you must implement only the application features supported by the current, stable production release. The custom installation utility for PHP includes prompts for you to include the Zend Optimizer and the Apache Perl Module (`mod_php`).

phpMyAdmin

Your server supports phpMyAdmin, a PHP software package which enables you to administer of MySQL over the Web. PhpMyAdmin is distributed by the PhpMyAdmin Project (http://www.phpmyadmin.net/home_page/index.php) under the terms of the GPL. You can install and uninstall the software package using custom installation utilities. Once the package is installed, your server receives automatic updates which do not require your intervention.

PGP/GnuPG

For the purposes of signing and encrypting your data communications, Pretty Good Privacy (PGP) and Gnu Privacy Guard (GnuPG) are both pre-installed on your server. PGP, originally developed by Phil Zimmerman, is a high security cryptographic software application for MSDOS, UNIX, VAX/VMS, and other computers. PGP enables you to exchange files or messages with privacy, authentication, and convenience.

Note: You must agree to the PGP 5.0 License before installing this version of PGP on your server. This version of PGP is for non-commercial use only. If you are going to use PGP for commercial use, you must purchase a license from Network Associates. This version of PGP has also been modified so that it will work in both the virtual and non-virtual environments. Modifications have also been made to the PGP executable provided such that it will only run on your server. Please do not attempt to export this version off of your server. It will not operate.

An alternative to PGP, GnuPG is distributed under the terms of the GNU General Public License. For more information, refer to the PGP GnuPG Web site (<http://www.gnupg.org/>). GnuPG (The GNU Privacy Guard) is a tool for secure communication and data storage. It can be used to encrypt data and to create digital signatures. It includes an advanced key management facility and is compliant with the proposed OpenPGP Internet standard as

described in RFC2440. GnuPG is a complete and free alternative to PGP. Because it does not use the patented IDEA algorithm, it can be used without any restrictions.

PostgreSQL

Your server supports the current, stable release of PostgreSQL, an open source relational database system distributed by PostgreSQL Global Development Group under the Berkley Software Distribution (BSD) license. The database system was formerly known as *Postgres* and *Postgres95*.

If you choose to configure PostgreSQL, add the following lines to your shell startup file, according to which shell your server is running.

Note: To find out which shell your server is running, run the following command:

```
% echo $SHELL
```

- `/bin/csh` - If you are using `/bin/csh` or one of its variants, then add the following lines to the `.cshrc` file on your server.


```
setenv PGDATA /usr/local/pgsql/data
setenv PGLIB /usr/local/pgsql/lib
set path = (/usr/local/pgsql/bin $path)
```
- `/bin/sh` & `/bin/bash` - If you are using the Bourne shell (`/bin/sh` or `/bin/bash`) then add the following lines to the `.profile` file on your server:


```
PATH=$PATH:/usr/local/pgsql/bin
PGDATA=/usr/local/pgsql/data
PGLIB=/usr/local/pgsql/lib
export PGDATA PGLIB
```

The tool for managing PostgreSQL is the `psql` client. To start `psql` run the following command:

```
% psql
```

The `psql` client starts, and then you can to run SQL-related commands and for help.

Note: Look for the following error:

```
Connection to database '(null)' failed.
FATAL: PQsetdb: Unable to determine a Postgres username!
```

To resolve this, run the following command:

```
% vpwd_mkdb /etc/passwd
```

This program will read your password file at `/etc/passwd` and create a Berkeley DB format file. PostgreSQL uses this new file to look up user names and account information.

Multi-Language Abilities in PostgreSQL

PostgreSQL enables for a number of languages by enabling specific character-sets in the databases. When you create a database in PostgreSQL, you can use the `-E` flag to enable support for a specific character set.

```
% initdb -E SET
```

The following list provides the available character sets and the character set name to use to enable support for it.

- ALT (Windows CP866).
- EUC (JP Japan EUC).
- EUC (CN China EUC).
- EUC KR (Korea EUC).
- EUC TW (Taiwan EUC).
- MULE_INTERVAL (Mule internal code).

- LATIN1 ISO 8859-1, LATIN2 ISO 8859-2, LATIN3 ISO 8859-3, LATIN4 ISO 8859-4, LATIN5 ISO 8859-5 (Latin alphabets one through five for Western Europe, Eastern Europe, Turkey, Northern and Western Europe, Cyrillic character sets).
- SQL_ASCII (ASCII).
- UNICODE (Unicode or UTF-8).
- WIN (Windows CP1251).

To remove PostgreSQL, connect to your server by means of SSH and run the following command:

```
% vuninstall pgsq1
```

Edit your `/etc/rc` file, removing the line that contains postmaster.

Run the `ps` command, as follows:

```
% ps -x
```

Determine the process ID of the PostgreSQL daemon and use `kill` to stop the PostgreSQL daemon:

```
% kill PROCESS-ID
```

Procmail

Your server supports Procmail (<http://www.procmail.org/>), a free, open-source mail delivery agent (MDA) distributed under the terms of the GPL. You can configure Procmail to call email filter programs, such as SpamAssassin.

You can customize the behavior of Procmail by creating a `procmailrc` file. The file must be located in your `/usr/local/etc/` directory, or a user can have a `.procmailrc` file in the user's home directory.

Samba

Windows File Sharing enables you to map a Windows network drive to your server home directory across the Internet. Once you have mapped the Windows network drive to your server, you can drag-and-drop files to and from your server as if it were a local drive.

The Windows File Sharing feature for server is made possible by Samba (<http://us4.samba.org/samba/>), a Server Message Block (SMB) client and server for UNIX. Your server provides a custom installation utility (`vinstall samba`)

SpamAssassin

Your server supports SpamAssassin (<http://spamassassin.apache.org/>), a free, open-source email filter distributed under the terms of the Apache Software license.

SpamAssassin applies a number of tests to an incoming message, and each test returns a score. If enough tests return a combined score that is high enough. The default setting is five (5). Once a message has been tagged, there are a number of possible actions that can be taken with the message. Both tagging and actions can be handled either as a system-wide or as a user specific filter.

- **System-wide Filters** apply SpamAssassin tests to every email message that arrives on your server, regardless of the intended recipient. This avoids accidentally losing the occasional legitimate message that has spam-like characteristics.

- `User Specific Filters` enables individual users to use different methods of dealing with spam. The user-specific settings enable you to configure specific users with different ways of dealing with messages tagged as spam. Once you tag a message, SpamAssassin will do one of the following with the message, depending on your system and user settings.
- `Deliver Tagged messages along with Untagged messages` enables the user to see if a message is tagged as spam and enables them to make the final decision to read the message or not. If you have system-wide filtering on, it is a good idea to use this option for the system-level filtering.
- `Deliver Spam to a special mailbox` delivers untagged messages and delivers tagged messages to a special mailbox (or IMAP folder). This is a good user-level setting for all users who don't want potential spam cluttering the user's inbox but want to have the option to check through to see if there is anything important among the tagged messages.
- `Deliver spam to a special mailbox and forward non-spam to another address` specifies that if a user has another account that they forward the user's messages to, this enables you to filter out spam before forwarding the messages to the user's account
- `Forward Spam to another address` specifies non-spam is delivered normally, but spam can be forwarded to an account on a different server.
- `Delete Spam` specifies that all messages tagged as spam are deleted, either on a system level, or just for specific users. This is not suggested, as messages (and possible false positives) would be permanently thrown away
- `Delete Spam and forward non-spam to another address` specifies that the tagged messages are deleted before forwarding untagged messages to a remote email account.

You can configure SpamAssassin to keep a log of activity. Logs can be useful in tracking down problems and errors but, like any other log file, your SpamAssassin logs must be cleared out occasionally to prevent them from using up all your disk space. You can run the `cron` command to archive or empty your spam log files.

There are a number of sources of documentation for SpamAssassin. You can access the manual pages issuing the following commands.

```
% man spamassassin
% man Mail::SpamAssassin::Conf
```

Locate further information about the SpamAssassin filtering engine at the SpamAssassin Project Web site (<http://spamassassin.apache.org/>).

Savelogs

Savelogs provide a complete Web server log rotation program. Savelogs can rename, archive, compress, delete, and provide a `newsyslog`-type of log rotation. You can specify options on the command-line or in a configuration file. Besides archiving single logs, savelogs can search your Web server configuration file to automatically rotate logs defined there.

Shockwave

Shockwave/Flash provides support for multimedia playback on your server. The Shockwave Player enables you to view interactive Web content. The software is developed, maintained, and distributed by Adobe (<http://www.adobe.com/shockwave/download/download.cgi>). You can include Flash multimedia presentations on your Web sites. You can use Flash content to add high-impact graphics, animation, and interactivity to your Web pages.

You may need to add the following MIME types to your `/www/conf/mime.types` file:

```
application/x-shockwave-flash  swf cab
application/futuresplash        spl
```

After making changes to the `mime.types` file, you must restart your Web server.

In order to embed your `filename.swf` Flash content in a Web page, you must include at least the following HTML code:

```
<OBJECT WIDTH="550" HEIGHT="400">
<PARAM NAME="MOVIE" VALUE="filename.swf">
<EMBED SRC="filename.swf">
</EMBED>
</OBJECT>
```

The `OBJECT` tags are for Microsoft browsers and the `EMBED` tags are for Mozilla browsers. Substitute the filename of your Flash content for `filename.swf`.

SquirrelMail

Your server supports SquirrelMail for Web mail processes. The open-source software is distributed by the SquirrelMail Project Team (<http://www.squirrelmail.org/>) under the terms of the GPL.

Swish-e

Your server supports Simple Web Indexing System for Humans - Enhanced (*Swish-e*), an open source system which enables you to index Web page and other types of files. A Swish-e development community (<http://swish-e.org/>) distributes the system under the terms of the GPL.

Swish-e provides you with a number of powerful indexing tools that you can modify and use however you want. Refer to the Swish-e Readme file for details of what is possible with Swish-e, and how to do what you want.

To install Swish-e, connect to your server by means of SSH, `su` to root, and run the following command:

```
# vinstall swish-e
```

This will install a number of files on your server. First, the `swish-e` program itself, which will be installed to the `/usr/local/bin/` directory. In addition, you will have access to several example configuration files in your `/usr/local/share/examples/swish-e/` directory, and the documentation for Swish-e in your `/usr/local/share/doc/swish-e` directory.

TCL

TCL (<http://sourceforge.net/projects/tcl/>) is an embeddable command programming language for interactive tools. As a scripting language, Tcl is similar to other UNIX shell languages such as the Bourne Shell (`sh`), the C Shell (`csh`), the Korn Shell (`ksh`), and Perl. Your server provides a custom installation utility (`vinstall tcl`) to assist you with the installation of the programming languages.

Time Zone Custom Installation Utility

A custom installation utility to interactively set the time zone is supported by your server. This enables you to set the time zone based on a major city in the desired time zone. To take advantage of this update, connect to your server through SSH and execute the following command from the prompt:

```
# vinstall timezone
```

Tomcat

Java Servlets and JSPs are made available on your server by means Tomcat, a software package distributed by the Apache Jakarta Project (<http://jakarta.apache.org/>). Tomcat is an implementation of the Java Servlet and JavaServer Pages specifications.

Note: Java applications consume significant CPU and memory resources and may not be appropriate for use on your server.

TWIG

Your server provides support for The Web Interface Gateway (TWIG, <http://www.informationgateway.org/>), a Web-based intranet/groupware tool and application framework. It is implemented using PHP, an HTML-embedded scripting language, and the MySQL database application. There is a custom installation utility for TWIG (`vinstall twig`) for the tool.

Urchin 5 (Google Analytics)

Urchin (<http://www.google.com/support/urchin45/>) is provided as Web analytics software which analyzes traffic for one or more Web sites and provides accurate and easy-to-understand reports. The software is developed, maintained, and distributed by Google Analytics (<http://www.google.com/analytics/>).

Urchin 5 Web Log Analyzer Features

The features of Urchin are continuously updated. The following provides a list of features provided by Urchin 5:

- Installs directly on your server with a `vinstall` utility
- Creates HTML-based graphical reports of Web server traffic
- Provides multi-language reporting
- E-commerce log reporting
- Provides the ability to track up to 100 Web sites or profiles

Note: Development and documentation for Urchin 3 is no longer available. If you are currently running Urchin 3 consider upgrading to Urchin 5.

Install Urchin

Connect to your server by means of SSH, `su` to root, and run the following command:

```
# vinstall urchin5
```

At the end of the `vinstall` utility, your Urchin 5 installation should be fully licensed (with a permanent license) and ready to configure. Be sure to note the URL of the Urchin administration interface, as well as the Username and Password of the administrative user. You will need them to configure Urchin 5.

It is important to note that the main difference between your server's installation and standard Urchin 5 is that your server's runs on the main Apache Web server, rather than on a stand-alone Web server used by Urchin exclusively. This difference means that some administrative tasks might require minor changes to the `/usr/local/apache/conf/httpd.conf` configuration file and a restart of the Apache daemons. Additionally, some sections of the README file, `install.txt` file, and the documentation may not be applicable to your Urchin 5 installation.

Configure Urchin

To configure Urchin you will need to go to the URL listed at the end of the `vinstall` utility. You will also need the listed login and password. The URL will be similar to the following example:

```
https://example.com:9878/
```

Login and follow the prompts. Click on the help icon for assistance with individual configuration screens.

Note: The `vinstall` utility configures Urchin 5 to listen on Web port 9878. This port may be blocked by firewalls. See instructions below to change the port.

Vinstall Utilities Library

The `vinstall` utilities library enables you to add supported software packages (utilities, database programs, and other software) to your server. The library provides a custom FreeBSD MPS v3 command-line tool. A root user can use the library from the shell on your server. To begin using library, connect to your server by means of SSH, `su` to root, and run the following command:

```
# vinstall
```

If you know the name of the package you want to install, you can install it directly by indicating the name of the package.

```
# vinstall package_name
```

If you do not indicate a package name, the `vinstall` utility enters an interactive mode which prompts you for more information, as in the following example:

```
Select an option:
```

```
? view list of programs
```

```
install enter install mode
```

```
module_name view information about program_name
```

```
quitexit vinstall program
```

```
-->
```

You can view the available programs available to install using the library, enter a question mark (?) at the prompt. Or you can run the following command:

```
# vinstall -l
```

You can install a program by entering install mode. Type `install` at the prompt, and you will enter install mode. You can then enter the package name at the next prompt, and `vinstall` utility begins installing the package. Typing the name of a program in the list will bring up a short dialog about what the program is. You can leave install mode without installing anything. To do this type `quit` at the prompt and you will return to the standard shell prompt.

Removing Software Packages

Most packages that can be installed using a `vinstall` utility can be removed using `vuninstall` utility. The `vuninstall` utility follows the same format as the `vinstall` utility.

Software Packages Included in the Vinstall Utilities Library

The following table provides you with information regarding the software packages which are included with the `vinstall` utilities library.

Note: Refer to updates provided on the Web, and other electronic communications regarding additions and modifications to the library.

Software Package	Install (<code>vinstall</code>)
Apache HTTP Server (1.3, 2.0)	Yes
Aspell (or <i>GNU Aspell</i>)	Yes
Clam AntiVirus	Yes
CPX: Control Panel	Yes
Dada Mail	Yes
eRuby	Yes
Expect	Yes
Formmail	Yes
FrontPage	Yes
iManager 2.0	Yes
Java JDK, JRE	Yes
Mailman	Yes
MajorCool	Yes
Majordomo	Yes
Metamail	Yes
MIVA (Empresa, Merchant, and upgrades)	Yes
mod_perl	Yes
mod_python	Yes
mod_ruby	Yes
MySQL (4, 5), MySQL check	Yes
Namaz	Yes
Open WebMail	Yes
PGP5Formmail, PGPFormmail	Yes
PHP (4,5)	Yes
phpMyAdmin	Yes
PostgreSQL	Yes
Procmal, Procmal LDA	Yes
Python (python-2)	Yes
System Quota Checker	Yes
Samba (2.x, 3.x)	Yes
savelogs	Yes
Sendmail (Sendmail RBL, sendmailcert)	Yes
SpamAssassin	Yes
SquirrelMail	Yes
Swish-e	Yes
Tcl	Yes
Time zone	Yes
Tomcat	Yes
TWIG	Yes

Urchin, Urchin 5	Yes
Webmin	Yes
WordPress	Yes
Wpoison	Yes
Zend Optimizer PHP enhancing application	Yes
Zope open source content management package	Yes

WordPress

WordPress is open-source software distributed under the terms of the GPL. WordPress utilizes PHP and MySQL. The software is highly customizable and provides you with the capability to deliver information by means of audio, video, and other media, including blogs and podcasts.

A blog is a collection of short articles, essays, or loosely-formatted thoughts, usually written by one individual. A podcast is a multimedia file (audio, video, or multimedia) distributed in a series of episodes. A customer can subscribe to your podcast, download it as soon as it is available, and then play it on their compatible devices (such as MP3 players).

Available Features

The following list provides an overview of some of the available features included with WordPress:

- Integrated theme system.
- Trackback support.
- Pingback support.
- Spam protection.
- Full user registration.
- Password protected blog postings.
- Support for importing content from previously-designed blogs (such as MoveableType).
- Common blog XML-RPC support.
- Workflow, post, and draft tools.
- Intelligent text formatting.
- Support for services (such as Ping-O-Matic) designed to update Web search engines.

As an open-source application, WordPress is not limited to this set of features. There are numerous extensions, or plug-ins developed by the community of WordPress users. Refer to the WordPress Web site for more information about standard WordPress features, extensions, or plug-ins.

Before you Install WordPress

You must uninstall any previously installed version of WordPress present on your account prior to installation using the `vinstall` utility. Also, make a backup of your previous configuration of blog or podcast software, as well as of the databases to which they refer. The `vinstall` utility provides for installing WordPress to any sub host configured in the Apache configuration file (`httpd.conf`).

Get Started

The `vinstall` utility for WordPress runs a script which places the WordPress version 2.0.2 on your account. To install the software, run the following command from a SSH prompt:

```
# vinstall wordpress
```

Note: If you are upgrading WordPress from a previous installation, ignore any warnings you receive regarding your existing MySQL database. After the installation completes, use your preferred browser to access the following location:

```
https://YOURDOMAIN/WORDPRESS/upgrade.php
```

Replace *YOURDOMAIN* and *WORDPRESS* with the domain and directory, respectively, in which you installed WordPress. After visiting the upgrade page, replace your customizations by utilizing the backup file you made before you began this process.

Refer to the WordPress Web site and documentation for further information regarding maintenance, administration, and troubleshooting.

Go Beyond the Basics with WordPress

Following are links to Web sites you can use to learn more about WordPress software, blogging, and other related services. These Web sites inform you about concerns in the Internet development community regarding how these applications interact with each other. In addition, many of the Web sites provide opportunities for you to subscribe to topical email lists and RSS Web feeds.

- MySQL Developer Zone – <http://dev.mysql.com/>
- PHP Group – <http://www.php.net/>
- WordPress Open-Source Software Wiki – http://codex.wordpress.org/Main_Page
- WordPress Open-Source Software homepage – <http://wordpress.org/>

The Webalizer

Your server supports The Webalizer (<http://Webalizer.domainunion.de/>). The Web server log file analysis program distributed under the terms of the GNU General Public License as published by the Free Software Foundation.

Manual pages are installed on your server when you install The Webalizer. Use the following `man` command to access them:

```
% man Webalizer
```

Webmin

Your server supports Webmin (<http://www.webmin.com/>), a UNIX Web-based interface for system administration. There is a custom installation utility (`vinstall webmin`) to assist you as you install the interface. Webmin is available from the FreeBSD Ports Collection.

WebTrends

WebTrends provides a Web Log Analyzer that will provides valuable information about your Web site and the users that access it. Reports generated by WebTrends Log Analyzer include statistical information as well as colorful graphs that show usage, trends, market share and much more.

WebTrends Log Analyzer will help you determine:

- Interest level in specific services you offer
- Local, national, and international activity
- Specific organizations to which your services appeal

- How users are referred to your Web site
- Activity at your site during any time period

Note: WebTrends Log Analyzer is a third party application. In order to use WebTrends Log Analyzer, you must purchase a license from WebTrends.

WebTrends Log Analyzer works very well with your server and is compatible with log files created by the Apache Web Server, as well as many other Web servers. Reports can be generated as HTML files that can be viewed by any browser on your own computer or remotely from anywhere on the Internet with any browser. You can also create the reports in Microsoft Word, Excel, text, and comma-delimited formats.

Wpoison

Wpoison is a Common Gateway Interface (CGI) program which you can use to reduce the quantity of bulk, junk email (or *spam*). Wpoison combats spam by thwarting the efforts of spammers who scan Web pages, looking for target email addresses which they subsequently bombard with spam.

Install Wpoison

To install Wpoison, connect to your server by means of SSH, and verify you are signed into the server as root (`su` to root), and run the following command:

```
# vinstall wpoison
```

Then add the `mod_rewrite` Apache Module to your Web server by including the following line in your Web server configuration file (`/www/conf/httpd.conf`):

```
LoadModule rewrite_module modules/mod_rewrite.so
```

Use Wpoison

In order to capture

In order to properly implement a site inoculation, you will want to use a combination of empty `<a href>` tags on your home page and throughout your Web site similar to this:

```
<a href="/traps/index.html"></a>
```

Add lines similar to the following to your Web server configuration file:

```
RewriteEngine On
RewriteCond %{HTTP_USER_AGENT} ^ExtractorPro.*
RewriteRule ^.* /spammers/index.html [L]
RewriteCond %{HTTP_USER_AGENT} ^EmailSiphon.*
RewriteRule ^.* /spammers/index.html [L]
RewriteCond %{HTTP_USER_AGENT} ^eCatch.*
RewriteRule ^.* /spammers/index.html [L]
```

To prevent legitimate Web robots such as Webcrawler (<http://www.Webcrawler.com/info.wbcrawl/>) from indexing the Wpoison generated pages, create an entry in your `robots.txt` file (or create the file in your `/www/htdocs/` directory if it does not already exist) to disallow the `/spammers/` directory, as in the following example:

```
User-agent: *
Disallow: /spammers/
```

Zend Optimizer

Zend Optimizer enables you to run encoded files and enhance the performance of your PHP scripts. The package is a passive module which runs within the framework of PHP and uses multi-pass code optimizations to potentially double the running speed of current PHP applications. This add-on is appropriate for all PHP users, who can benefit from the better response time featured by the package. The increase in speed for running PHP code reduces the CPU load for the server, and cuts latency time in half. Once you install the package, the version is updated automatically by means of server software updates. Zend is a trademark of Zend Technologies Ltd (http://www.zend.com/products/zend_optimizer) and is distributed under the terms of that organization.

Install Zend Optimizer

Follow these steps to install Zend Optimizer:

1. Connect to your server by means of SSH.
2. Verify you are signed into your account as root (su to root).
3. Run the following command:
`# vinstall zendoptimizer`
4. The install will ask the location of your php.ini file. By default this file will be located in the `/usr/local/php5/lib` directory.
5. Run the following command:
`# restart_apache`

Go Beyond Zend Optimizer Basics

To learn more about configuring and using Zend Optimizer features, refer to the online Optimizer forum (<http://www.zend.com/zend/optimizer-forum.php>).

Zope

Zope is an open-source content management package. The package is designed and developed using Python and uses a Web-based interface to enable you to quickly and easily develop a content management solution to suit your content management needs. Zope is developed, maintained, and distributed by the Zope Community (<http://www.zope.org/>) under the terms of the Zope Public License (ZPL) and with support, as well as funding from, the Zope Corporation.

Install Zope

To install Zope, SSH or telnet to your server and run the following custom installation utility:

```
% vinstall zope
```

The Zope installation starts by checking for and installing (if they don't already exist) some packages that are required for Zope to function. These include Python and ZMySQLda (which allows Zope to work with a MySQL database). You will then be guided through some basic configuration step, as follows:

1. For the administrative password, enter a memorable password. When you type the password, you will not see the cursor move. After you enter the password once, you will need to confirm it by entering it again.
2. When the system asks if you want to configure an emergency access user, select Yes. Create the emergency user with a username and password you will be able to remember; you will need to use the emergency user to fix your admin user if you ever get locked

out.

3. When prompted to select the encryption format, you should select CRYPT, the standard UNIX encryption method.
4. Domain restrictions enable you to configure specific domains with permission to access Zope. Add any domain names you want to allow access to your Zope server.
5. Verify you include the one you are connecting from.

Note: You can change the domain access restrictions later if you wish to add or remove any domains from the list.

Use Zope

Once Zope you install on your server, you can access the Web-based administration section by going to the following URL:

```
http://YOUR-DOMAIN.NAME/zope/manage/
```

When you are prompted, use *admin* for the user name, and the administrative password you previously configured during the install.

Note: Zope runs on a different port than your Web server. If you are unable to access Zope using the `/zope/` path, try connecting directly to the port:

```
http://YOUR-DOMAIN.NAME:8080/manage/
```

Go Beyond the Basics of Zope

There is a large community of Zope users and extensive information provided by them for Zope users, administrators, and developers. For more, refer to the following Web sites:

- Zope Community (<http://www.zope.org/>)
- Zope Developers Guide (<http://www.zope.org/Documentation/ZDG>)
- Zope Documentation (<http://www.zope.org/Documentation>)

Troubleshoot Your Server

This section describes how to troubleshoot general issues as well specific problems you encounter as you operate your server. This section provides information about troubleshooting the following problems on your account:

- “General Issues” on page 47.
- “Failure to Create a Virtual Host” on page 47.
- “Check Quotas” on page 47.
- “Check Log Files” on page 47.
- “Check for Idle Processes” on page 48.
- “Custom Digital Certificate Problems” on page 48.

General Issues

Always remember where you are located now in your command interface. Check it periodically using the `pwd`, `hostname`, `ifconfig` commands. The same command executed inside your server, under a different level of access, can lead to different results. Subscribe to bug tracking lists for FreeBSD UNIX and the additional, supported features you install on your server. Keep track of new public denial-of-service attack tools or remote exploits for the software and install them into your server or at the server level.

Failure to Create a Virtual Host

If your attempt to create a new virtual host fails and you see a message indicating that the operating system template is absent or inaccessible, verify the location of the template on your system and, if necessary, re-install the template.

Check Quotas

When your server meets quota limits, the disk cannot be written to. Your server cannot accept email, log files, or complete installations. Your quota has a soft limit (which you temporarily exceed) and a hard limit (which you do not exceed).

Check Log Files

Your server records all errors and system messages in log files. If you or your users are having problems on the account, first check the quota; then check the log files. If the problems concern email, check the `/var/log/maillog` file. Problems with the Web site are recorded in the `/www/logs/error_log` file.

Use the `tail` command to watch error messages as they are added to log files. Note what is being added to the log files as the user duplicates the error. Follow these steps to use the `tail` command:

1. Connect to your server using SSH.
2. At the command prompt type `tail -f /var/log/maillog`. (If necessary, substitute the messages directory with `/www/logs/error_log`, `/access_log`, or the `/ssl_error_log` files.)
3. Have the user duplicate the error while you are running the `tail` command.

Check for Idle Processes

If you are receiving errors, use the `top` command to check the length of time a current process has been running. If the process is idle or has been running an unusually lengthy period of time, the process could be suspended and causing problems. For example, an FTP process can hang if you improperly disconnect from your server. Use the `kill` command to shut down a suspended process.

Custom Digital Certificate Problems

There are a number of warnings or errors that can come up when accessing Web pages by means of SSL. Your SSL digital certificate is configured to use a very specific domain name, which must match exactly to avoid an error. For example, if your certificate is for the domain *www.my-domain.name*, and you type *https://my-domain.name* into the browser, you will get this warning. Likewise, if your certificate is for *my-domain.name* and you enter *https://www.my-domain.name* into your browser, you will get the same warning. To avoid this warning, verify the exact domain name on the certificate when making links to secure pages. Following are suggestions to use as you troubleshoot for digital .certificate problems:

- When you make links or include images in pages, the URL is an absolute link and includes the protocol, domain, and path to a file. If you include an image in a page using an absolute URL, see an error when the page is viewed using a different protocol than the one indicated in the image URL. For example, include an image as follows:
http://www.my-domain.name/images/myimage.gif
When you access this page through secure protocol such as HTTP over SSL (HTTPS), you will see a warning that the page has encrypted as well as unencrypted content. The easiest way to avoid this error is to use relative paths, as in the following example:
/images/myimage.gif
- Many older Web browsers only support 40 or 52 bit encryption. Because modern SSL certificates use 128 bit encryption, older browsers may not be able to view pages securely. If many of your customers are likely to be using older browsers, you must acquire a special low-encryption certificate. Several current browsers are available free of charge. Encourage any users having problems with your SSL certificate to upgrade to a current browser.
- When you install a custom signed digital certificate, there are a number of possible mistakes or errors that can cause problems. In most cases, the Apache HTTP server will not start up when one of these errors occurs. If your site will not load in a browser, check if there are any HTTPS processes running on your server.
Connect to your server by means of SSH and run the following command:
`# top`
Restart Apache and try loading the page again even if there are HTTPS processes running. If restarting the Apache does not cause HTTPS processes to start on your server, it is possible your custom certificate is not installed properly.
- Verify the account's private key is not decrypted. View the file; if the key file includes the following lines, the key is still encrypted:
`Proc-Type: 4, ENCRYPTED`
`DEK-Info: DES-EDE3-CBC, BCC23A5E16582F3D`
To decrypt your server's private key, run the following command from the SSH command prompt:
`% openssl rsa -in /etc/ssl.pk -out /etc/ssl.pk`
When prompted, type the PEM Passphrase, after which the key is decrypted.
- Verify you uploaded the certificate using an ASCII format. Check if your certificate was uploaded properly by reviewing it in a text editor. If each line includes character which

indicate it was uploaded the file in a binary format (^M), you must upload the file again using ASCII format.

- Verify that the certificate and private key match. For example, if you have multiple accounts which utilize SSL, verify you are using the private key which was generated at the same time as the CSR for the domain of the account you are configuring.
- Verify if you ordered a certificate that is correct for your server. For example, if you are transferring your certificate from a previous account, verify that the previous account uses Apache with SSL as the Web server software.
- Verify your certificate or key are complete. Check that the certificate or key is complete, that the beginning and ending lines of the key or certificate are present. Both the certificate and private key begin and end with specific as in the following example:
-----BEGIN RSA PRIVATE KEY.