
Migrating to FreeBSD VDS v3

**First Edition
December 2008**

Table of Contents

- Introduction 1**
 - Executive Summary 1
 - Overview of the Migration 1
 - Process Timeline..... 1
 - Regarding Specialized Configurations..... 2
- Prior to Requesting a Shadow Account 3**
 - Becoming Familiar With FreeBSD VDS v3..... 3
 - Reacquainting Yourself with FreeBSD 3
 - FreeBSD Files and Directories 3
 - FreeBSD User Permissions..... 4
 - Accessing the FreeBSD Manual Pages 6
- Using Your Shadow Account..... 7**
 - Regarding IMAP..... 7
 - Completing the Migration..... 7
 - Canceling the Migration 7
 - Connecting to Your Shadow Account 7
 - Using SSH to Connect 8
 - Using FTP to Connect..... 8
 - Controlling Access to Your Shadow Account 9
 - Managing Groups 9
 - Vinstall Add-Ons 10
 - Perl Modules 10
- Before Terminating Your Original Account 11**
 - Verifying Website Configurations 11
 - Verifying your Email Settings 12
 - Maintaining Your Digital Certificate..... 12
 - Finalizing the Migration 13
- Caveats..... 14**

Table of Tables

Table 1: Files Created on Your Shadow Account4
Table 2: User Permission Bits4
Table 3: User Permission Bit Combinations.....5
Error! Reference source not found. **Error! Bookmark not defined.**
Table 4: Verifications and Guidelines for your Website 12
Table 5: Caveats 14

Introduction

Note: At the time of this release, migration scripts and tools are designed to successfully copy the content of your FreeBSD VDS v2 (*original*) account to a special migrations directory of on a newer, FreeBSD VDS v3 (*shadow*) account. It is your responsibility to configure the content on your newer account. Verify the configuration of the following critical or important applications before you terminate your original account and complete your successful migration to FreeBSD VDS v3:

- Antivirus package: ClamAV
- Databases: MySQL, PostgreSQL
- E-Commerce: ShopSite
- Mail Services: Aliases, Lists, POP, IMAP, SpamAssassin
- Multimedia: Shockwave Flash, Podcasting, WordPress blog
- Server Access: CPX:Control Panel, Secure Shell (SSH), Root
- Statistics Packages: Urchin, The Webalizer
- Web development tools: PHP Hypertext Processor (PHP) Miva Empresa, Java
- Web services: Apache HTTP Server, Common Gateway Interface (CGI) binaries,

Using the instructions included in this document and by applying your FreeBSD Virtual Server experiences, you can conduct a successful migration to FreeBSD VDS, version three (FreeBSD VDS v3). By means of a structured and guided migration, you can conduct this on your own and without extensive technical support.

After this process is completed, you can expect that the content and users, email, SSH keys and SSL certificates from FreeBSD VDS v2 will be copied to a newer, FreeBSD VDS v3.

Executive Summary

This version of the VDS account adheres closely to the UNIX-style FreeBSD operating system. This ensures that knowledge an administrator has about other FreeBSD operating system environments transfers seamlessly.

This version provides you with support for newer features and enhancements. This is true not only for the features which enable you to manage users and content but also for external, open source database, scripting, mail, and Web analytics. Improvements to the process of moving to this version decrease the time an administrator must invest to learn a customized operating system.

Overview of the Migration

The migration begins when you request a duplicate (or *shadow*) account. For the duration, you will have two accounts. At the end, it is your FreeBSD VDS v2 (or *original*) account which terminates. At that point, your shadow account becomes your updated, FreeBSD VDS v3 account. And it is that account which utilizes all of the features of the newer version.

Process Timeline

The standard duration of your migration is up to 21 days. You can complete the migration in less time that and you can, if necessary request an extension. During the migration, there are up to four sets of tasks to perform, as follows:

- Configure content

- Test all content,
- Update domain name service (DNS)
- Complete your migration

After you have tested all content, you should request a DNS update for all domains associated with your original account.

Note: If we do not provide DNS service for your domain names, you must update your DNS to point to the IP address of your shadow account.

At any time after you have successfully requested a shadow account and conducted the migration, you can complete the process. Completing it terminates your original account. On the twenty-first day, the process is automatically completed without your intervention.

Regarding Specialized Configurations

The automated portion of the process copies content to your shadow account and enables you to begin the process of configuring your new account. This document does not address issues associated with a highly customized VDS account configuration including significant changes to the file and naming structures as well as services of a VDS account. As the capabilities of the VDS account allow for multiple configuration options. After you have initiated the migration process, you will need to perform additional steps. In fact, whether you have a near-default or a highly-customized configuration, you must plan time to manually configure features you will utilize.

Prior to Requesting a Shadow Account

This section familiarizes you with the migration and the FreeBSD operating system. It also provides you with an overview of shadow-related options available in the Backroom.

Becoming Familiar With FreeBSD VDS v3

Familiarize yourself with the following customer documentation, offered at no extra charge and as a feature of FreeBSD VDS v3, prior to beginning your migration:

- *FreeBSD VDS v3 Firewall Supplement*
- *FreeBSD VDS v3 Getting Started Guide*
- *FreeBSD VDS v3 IPv6 Supplement*
- *FreeBSD VDS v3 New Feature Supplement* (for Multiple IP Address support)
- *FreeBSD VDS v3 Release Notes*
- *FreeBSD VDS v3 User's Guide*
- *FreeBSD VDS v3: Frequently Asked Questions* (available on the Web)

You might also refer to the following documentation offered on the Web by the FreeBSD Documentation Project (<http://www.freebsd.org/docs.html>):

- *Frequently Asked Questions for FreeBSD 5.x and 6.x*
- *FreeBSD Handbook*
- *FreeBSD Hypertext Man Pages* (or *Manual Pages*)

Note: For more information about how to access the FreeBSD General Commands Manual, see “Accessing the FreeBSD Manual Pages” on page 6.

Reacquainting Yourself with FreeBSD

This section provides you with information about using the FreeBSD directory structure and user permissions. In addition, this section refers you to additional resources to know more about FreeBSD.

FreeBSD Files and Directories

Your shadow account, just as your original account, utilizes a file structure that follows that of FreeBSD, a UNIX-style operating system. It mounts all directories from a *root* directory appearing as a single forward slash character (/). The operating system provides mount points, directories where you add additional file systems onto the root file system.

Initiating a shadow account creates the following files and directories on your shadow account:

| File/Directory | Description |
|-----------------------|--|
| /.migrate | Directory where the process automatically places content and data from your original account. Caution: Do not edit the content of the /.migrate directory. |
| /.migrate/accountname | A subdirectory of .migrate containing a copy of the file system from your original account. |
| /.migrate/bin | A subdirectory of .migrate containing binary files from your original account. User utilities fundamental to both single-user and multi-user environments. |

Table 1: Files Created on Your Shadow Account

FreeBSD User Permissions

UNIX provides a basis for FreeBSD, a multi-user operating system. FreeBSD enables several users to work simultaneously on unrelated tasks. The operating system enables you to configure your VDS account so that these users' requests can share hardware devices, peripherals, memory and processing capacity.

The FreeBSD directory uses a directory structure that includes control of the permissions you grant to *users*, *groups*, and *everyone else* (neither users nor groups). The structure enables you to configure permissions for users and groups. Control user access by configuring permissions to your precise needs.

The following table describes user permission bits and the abilities they specify for each user.

| Permission Bit | Specifies |
|----------------|--|
| r | User can read files |
| w | User can write to files |
| x | User can execute files. |
| en dash (-) | A null value, no permission granted for the ability. |

Table 2: User Permission Bits

The following table provides you with a guide to permission combinations you can issue to users, groups, and users.

| Permission | Directory listing |
|-------------------------------|-------------------|
| No read, no write, no execute | --- |
| No read, no write, execute | --X |
| No read, write, no execute | -w- |
| No read, write, execute | -wX |
| Read, no write, no execute | r-- |
| Read, no write, execute | r-X |
| Read, write, no execute | rw- |
| Read, write, execute | rwX |

Table 3: User Permission Bit Combinations

Following is an example of a typical setup for permissions on a user file, including group permissions:

```
rw-r--r--
```

In the previous example, *rw* specifies read (*r*) and write (*w*) permissions for the owner of the file. The group and others have read and null permissions (*r--*). This way, the group and others cannot write to the owner's file.

Use the `ls -l` command line argument to see a directory listing including a column with information about a file's permissions for the owner, group, and everyone else. Issuing an `ls -l` command line argument in a directory may show results as in the following:

```
% ls -l
total 530
-rw-r--r-- 1 root  wheel    512 Sep  5 12:31 myfile
-rw-r--r-- 1 root  wheel    512 Sep  5 12:31 otherfile
-rw-r--r-- 1 root  wheel   7680 Sep  5 12:31 email.txt
...
```

You can use the `chmod` utility to change permissions. For more information about the `chmod` utility, refer to the *FreeBSD Manual Pages*.

Accessing the FreeBSD Manual Pages

You can access a version of the FreeBSD *Manual Pages* several ways. You can run the `man` command on your account command prompt, as follows:

```
yourv3 > man command
```

```
BUILTIN(1)                FreeBSD General Commands Manual
BUILTIN(1)
```

NAME

`alias`, `alloc`, `bg`, `bindkey`, `break`, `breaksw`, `builtins`, `case`, `cd`, `chdir`, `command`, `complete`, `continue`, `default`, `dirs`, `do`, `done`, `echo`, `echotc`, `elif`, `else`, `end`, `endif`, `endsw`, `esac`, `eval`, `exec`, `exit`, `export`, `false`, `fc`, `fg`, `filetest`, `fi`, `for`, `foreach`, `getopts`, `glob`, `goto`, `hash`, `hashstat`, `history`, `hup`, `if`, `jobid`, `jobs`, `kill`, `limit`, `log`, `lo88gin`, `logout`, `ls-F`, `nice`, `nohup`, `notify`, `onintr`, `popd`, `printenv`, `printf`, `pushd`, `pwd`, `read`, `readonly`, `rehash`, `repeat`, `sched`, `set`, `setenv`, `settc`, `setty`, `setvar`, `shift`, `source`, `stop`, `suspend`, `switch`, `telltc`, `test`, `then`, `time`, `trap`, `true`, `type`, `ulimit`, `umask`, `unalias`, `uncomplete`, `unhash`, `unlimit`, `unset`, `unsetenv`, `until`, `wait`, `where`, `which`, `while` - shell builtin commands

SYNOPSIS

```
builtin [-options] [args ...]
```

DESCRIPTION

Shell builtin commands are commands that can be executed within the running shell's process.

You might also refer to the documentation offered on the Web by the FreeBSD Documentation Project as well as other locations on the Web.

Using Your Shadow Account

Regarding IMAP

Note: The default configuration for FreeBSD VDS v3 currently utilizes the `mbox` format. However, the `maildir` format (and Dovecot) is supported by means of a `vinstall`. The examples included in this document refer to the `mbox` format only.

If your original account is configured to utilize IMAP exclusively, the mailbox on that account is a file named as in the following example:

```
/var/mail/username
```

In addition, mailbox information is located in the following files:

```
/usr/home/username/mbox
```

```
/usr/home/username/mail/saved-messages
```

```
/usr/home/username/mail/my-favorites
```

```
/usr/home/username/mail/other_names_of_choice
```

Notes: On your original account:

- IMAP clients can automatically empty the contents of `/var/mail/username` and add it to `/usr/home/username/mbox` when the IMAP client begins to run.
- You have additional mailbox files, each file is located in the following location:
`/usr/home/username/mail`.
- Some IMAP clients change the `/usr/home/username/mail` directory.

Completing the Migration

At the completion of the migration the original account is terminated and the shadow account becomes the live, active VDS account.

Note: This removes all content from your original account. Once you select this option, you cannot revive the original account or return to that configuration.

Verify you performed the following tasks before you request to complete the migration:

- Manually configured your shadow account for all configurations.
- Verified executable files and other contents located in the `cgi-bin` directory.
- Tested your shadow account to ensure it is functioning properly.

Canceling the Migration

You can cancel your shadow account and stop the migration to FreeBSD VDS v3. This option simply cancels the migration. It does not begin or finish any further shadow-related tasks. Since this option terminates your shadow account, any configuration on that account is, potentially, lost.

Connecting to Your Shadow Account

Note: When you connect to your shadow account to perform shadow tasks, you must first connect as the Admin User. Use the IP address of your shadow account. Once connected, use the `su` command to access the root user profile.

Connecting to your shadow account is only slightly different than connecting to your original account. Your shadow account has both an Admin User and a root user. This section includes

information about using Secure Shell (SSH) and File Transfer Protocol (FTP) to connect to your shadow account.

Using SSH to Connect

In a UNIX-style operating system, a root user (also called *super user*) has unlimited abilities including the ability to execute commands that the Admin User cannot. You make most of the changes on your server as the root user, not as a user with administrative permissions. As a security measure, you cannot use a Secure Shell (SSH) client to connect directly to the server as the root user even when you use a correct password.

Instead, you must use an SSH client to connect to the server as the Admin User and after you have performed the necessary task, use the `su` command to become the root user. Do this to assure your root password is secure, to execute the commands you need, and to navigate from the Admin User to become the root user.

After you connect using the new IP address of your shadow account with the Admin User password authentication, run the `su -l root` command to become the root user. After issuing the command, enter the root password which is the same as the Admin User password.

As the Admin User you have a limited set of privileges, compared to the root user. For most administrative tasks, you will need to `su` to the root user. As Administrative User you can do some simple administrative tasks such as adding users and Web administration. The `sudo` command enables you to perform the following commands:

| | | | | |
|-----------------------------|------------------------|------------------------|--------------------------|---------------------|
| <code>adduser</code> | <code>vadduser</code> | <code>vedituser</code> | <code>pw</code> | <code>rmuser</code> |
| <code>vmuser</code> | <code>vlistuser</code> | <code>vlist</code> | <code>edquota</code> | <code>quota</code> |
| <code>restart_apache</code> | <code>apachectl</code> | <code>vaddhost</code> | <code>su webadmin</code> | <code>su -l</code> |
| <code>webadmin</code> | | | | |

The following example shows how to add a user using `vadduser` with the `sudo` command:

```
% sudo vadduser
```

By default, you can perform the above commands without a password; however, `sudo` can be set to require a password for additional security. To add additional functionality to the `sudo` command, edit the `/usr/local/etc/sudoers` file as root. For more information on `sudo` see the `sudo` man pages:

```
% man sudo
```

```
Examplev3 >
```

As you navigate, you can verify the user with which you logged in most recently by issuing the `whoami` command, as in the following example:

```
Examplev3 /home/examplev3# whoami
root
examplev3 /home/examplev3#
```

Using FTP to Connect

Your shadow account includes an assigned virtual host (appears as *VirtualHost*) for each user. All files under each `VirtualHost` directory have the same ownership as the user to which it was assigned. When you connect as a user you read, write, and execute only the files owned by the user. `VirtualHost` files appear in the Web (`www`) directory. When you connect as the Admin User, you read only the files owned by that user, including any user's `VirtualHost`

files. However, an Admin User cannot write to any other user's VirtualHost files. Only the root user modifies the Apache configuration file (`www/conf/httpd.conf`) on your shadow account.

Controlling Access to Your Shadow Account

As with your original account, your FreeBSD VDS v3 account enables each user to own files and directories. Users can change permissions to specific files or directories that they own.

Your new account provides you with the option to assign shell access to user permissions. They also automatically assign new users to a group when you do not. Once configured, users with shell access can directly connect to their home directory on the server, work with files, and run commands.

The `/www/htdocs` directory on your shadow account is assigned to the `webadmin` user. The `webadmin` user is a part of a standard, FreeBSD configuration. If your main Website is migrated to the main `/www/htdocs` directory, you cannot log in with as Admin User and edit files those files. Instead, change the ownership of the `htdocs` directory or set up the Main domain as a VirtualHost.

Managing Groups

Note: In most cases, you do not need to manually edit the `/etc/group` file. Use the `vadduser` command to add users.

Your FreeBSD VDS v3 account automatically assigns all users to a group. A group consists of users identified by their user name or by their group identification (GID). Groups appear in the `/etc/group` file. When users are migrated they keep the same permissions as they did on the original account. For example, if a user had FTP access on your original account, they are automatically placed in the FTP group (in `/etc/group`) on your shadow account.

Vinstall Add-Ons

Since your shadow account is a new one, use `vinstall` for additional programs that you may have installed on your original account. There may not be a `vinstall` for some of the programs that you previously installed in your FreeBSD VDS v3 account with `vinstall`. If this is the case, install the program using FreeBSD ports collection. For information on how to install programs using the ports collection refer to the customer documentation for your account.

Perl Modules

If you are using Perl and you have installed modules that you use, install these modules on your FreeBSD VDS v3 account as well. Instead, FreeBSD VDS v3 uses the default Perl method of installing modules. To install a module using Comprehensive Perl Archive Network (CPAN), type the following at the shell prompt:

```
% perl -MCPAN -e shell
```

The first time you run CPAN, you are prompted for your preferences regarding where to download your modules and other settings. For most settings, just accept the default. When you select your mirror preferences, select from four to five mirrors.

Note: You do not have to know anything about the mirror sites or where they are located. They serve your account just as well if you simply choose the first five (1, 2, 3, 4, and 5) as any others.

Configuring your CPAN settings rarely require reconfiguration, making this a one-time task, in general. From a CPAN command line (`cpan>`), to install a module type the following command in order to get and install the module:

```
cpan> install <Module_name>
```

To save space on your account, be sure to do a clean of the module after your installation by typing the following at the CPAN command line to delete all unnecessary files the server used during installation:

```
cpan> clean <Module_name>
```

Before Terminating Your Original Account

As noted previously in this document, you should perform a series of verification tasks prior to the end of the 21-day period allowed for your shadow account. This section provides guides to verifying Website configurations, email setup, and Urchin Web analytics. This section also provides you with descriptions of how to move your digital certificate and, finally, how to terminate your original account and make your shadow the active, FreeBSD VDS v3 account.

Verifying Website Configurations

Verify the following areas of your Website configurations before you point DNS to the new IP address and direct your traffic to the Websites:

- Website access is unobstructed
- All Web content saved from your original account to a local computer
- All necessary Web content uploaded to your shadow account
- Virtual hosting operates correctly
- Common Gateway Interface (CGI) scripts run without conflict

Before you verify Website configurations, ensure you have access to the correct IP address for your shadow account. Email notifications begin once you establish a shadow account. Those emails include the address. Also, the Shadow Details page located in the Backroom includes the address. Refer to either recourse to ensure you have the correct address.

The following table provides you with verifications and guidelines for your recently migrated Website configurations:

| Verifications | Guidelines |
|-------------------------------------|--|
| Website access is unobstructed | Use your Web browser to verify your main Website functions correctly. The correct URL for this task does not include a Web resource identifier (<i>www</i>) in the URL. Access your main Website by using the IP address of your shadow account. |
| Virtual hosting functions correctly | <p>Always test a virtual host before moving DNS services to your shadow account. Access the virtual host by using an Web address you compose of the following elements:</p> <p><i>ip.add.re.ss/user_name/</i></p> <p>Notes:</p> <ul style="list-style-type: none"> • The IP address of your shadow account (<i>ip.add.re.ss/</i>) • The user that the virtual host is under (<i>user_name/</i>) <p>This address provides an index of sites under <i>user_name</i> or a listing of <i>/home/user_name/www directory</i>. Click the domain that you want to test to show the results for the Website you are testing.</p> |

| Verifications | Guidelines |
|----------------------------------|--|
| CGI scripts run without conflict | <p>Check active scripts, including CGI scripts, for any conflicts. For example, the original account stores a virtual host's CGI information in the following directory: /www/cgi-bin</p> <p>For a shadow account, move CGI scripts to cgi-bin directory for the VirtualHost on your shadow account. By default, the migration assigns a cgi-bin directory for each VirtualHost directory. Following is an example of the directory path: /home/userid/www/example.com/cgi-bin</p> <p>If a virtual host had its own cgi-bin directory on the original account then any scripts in that cgi-bin directory are moved to the correct cgi-bin directory automatically.</p> |

Table 4: Verifications and Guidelines for your Website

Verifying your Email Settings

Check the following files to verify you have configured them correctly; they affect email delivery.

```
/etc/mail/aliases
/etc/mail/virtusertable
/etc/mail/access
/etc/mail/local-host-names
```

Maintaining Your Digital Certificate

Initiating the migration process does not move any custom or multiple digital certificates to your shadow account. Follow these steps to copy your custom digital or multiple certificates:

1. Copy the certificate and private key files to the /etc directory from the .migrate directory of your shadow account as follows:
%cp /.migrate/account_name/etc/ssl.cert /etc
%cp /.migrate/account_name/etc/ssl.pk /etc
2. Add the following lines to the Apache configuration file (usr/local/apache/conf/httpd.conf) outside of all Apache directives:
SSLCertificateFile /etc/ssl.cert
SSLCertificateKeyFile /etc/ssl.pk
3. Restart Apache run the following command as root from the command line:
%restart_apache

Finalizing the Migration

The duration of your shadow account is up to 21 days. At the end of that period, the migration completes and your original account terminates automatically. Prior to the conclusion of the 21 day period, verify you have configured and tested all content, settings, and applications. Also, verify and test time-sensitive aspects of your shadow account such as DNS, mail, and database configurations.

Caveats

The following table provides information about special concerns regarding FreeBSD VDS v3 accounts and the migration:

| Topic | Special Concerns |
|------------------------------|--|
| Manual configuration changes | Once you initiate a shadow account, changes you make to the configuration of your original account do not automatically become part of your shadow account configuration. During the migration, any changes you manually perform for your original account, you must also perform for your shadow account. |
| Disk space | If you have added additional disk space to your original account the additional space is added to your shadow account also. You cannot remove the additional space until after the migration is complete |
| Databases | By default, FreeBSD VDS v3 is configured to utilize version 5.0.x of MySQL. You can configure your account to utilize the default version (5.0.x) or you can use the provided vinstall to configure your new account to utilize the legacy version of MySQL (4.1.x). Since FreeBSD VDS v2 includes support for version 4.1.x by default, you must take action: verify the configuration of your databases for the newer version, or configure your FreeBSD VDS v3 account to support the legacy version. For PostgreSQL, be aware that your original account supported version 7.4.x where FreeBSD VDS v3 supports version 8.2.x by default. There is not a vinstall to assist you with a configuration of FreeBSD VDS v3 to support a legacy version of PostgreSQL. |
| CPX:Control Panel | If you use the CPX: Control Panel, refer to release notes and other guides, available from the FreeBSD VDS v3 Documentation Library, to verify the versions of Web development tools, Web services, and databases supported by CPX: Control Panel. |
| Web development tools | The default configuration of your FreeBSD VDS v3 server includes an installation of PHP Hypertext Processor (PHP), version 5.2.x where FreeBSD VDS v2 includes version 4.4.x. |
| Web services | The default configuration of your FreeBSD VDS v3 server includes an installation of Apache HTTP Server, version 2.2.x where FreeBSD VDS v2 includes version 1.3.x. |

Table 5: Caveats