Linux VDS

Getting Started Guide

Table of Contents

Introduction

This document provides information for basic administration of Linux Virtual Dedicated Server (VDS). It provides important information about creating users, configuring email and subhosts (or *virtual hosts*), and program management to help you begin using Linux VDS in a short period of time.

The Linux VDS product offers the benefits of a managed hosting solution with the flexibility of a dedicated Linux server. The proprietary technology of the Linux VDS provides a solution where the operating system and core services are managed for you. However, you have full root access to your files and programs.

This guide discusses server management from a shell command line and assumes you understand shell clients and command-line interaction with Linux servers. If you do not use shell command line or are not familiar with shell interaction, you can install and use the CPX: Control Panel Web interface. For more information about CPX: Control Panel, see the getting started guide for CPX: Control Panel.

Linux VDS includes several *v-commands*. These commands, created specifically for your product, enable easier server administration. Some examples discussed in this guide are `vadduser`, `vrmuser`, `vaddhost`, and `vinstall`.

This document includes the following sections:

Configuring Users

As part of the automatic maintenance provided with Linux VDS, your account contains several necessary root and application users by default, such as `root`, `ftp`, `pop`, and the administrative user (created when you ordered the server). You can create additional users and offer them different services, such as FTP, email, and shell access. This section discusses adding, managing, and removing users.

Note: When adding, configuring, or removing users, you must be the root user. You can become the root user by typing `su` at the command line and supplying the root user password.

Adding Users

To add a server user, follow these steps from the command line:

1. Type `vadduser` and press **Enter**.
2. Further instructions and information for vadduser will display during this step and throughout the vadduser process. Press **Enter** to continue after reviewing the information.
3. **Type** the username for the user you wish to create and press **Enter**. Usernames consist of alphanumeric characters up to 16 characters. The first character cannot be a number.

Note: Red Hat Enterprise Linux (RHEL) supports multi-byte characters in usernames, but will not support the following multi-byte characters because they are reserved for system use: `@`, `$`, `#`, `/` or `\`.

4. **Type** the user's password and press **Enter**. Use a password that will not easily be guessed. A combination of uppercase and lowercase letters, numbers, and symbols is an example of a more secure password. If a particularly insecure password is entered, the system will prompt you to enter a more secure password. You will need to type the same password twice for confirmation.
5. **Type** the user's full name and press **Enter**. This information will display when using system tools such as `vlist` (used to list all server users).
6. **Type** the user's home directory path and press **Enter**. The `vadduser` command simplifies this step and provides a recommended path for you. You can press **Enter** without typing a path to select this default.
7. **Select** the services to offer the user. This prompt is a toggle-style one, with the FTP and email services enabled for the user by default. Available services include `ftp`, `mail`, and `shell`. **Type** the service name at the prompt and press **Enter** to toggle the service on or off. Press **Enter** when done.

Caution:
Shell access enables control over many aspects of the server. Shell access should only be offered to trusted users. With shell access users can potentially change files or settings affecting your entire server. Use caution when offering the shell service to users.

8. **Type** the user's allotted file system quota in megabytes (MB). The quota must be a whole number. **Enter** `0` to give the user an unlimited quota (up to the free space available for your plan). Press **Enter** to complete the addition of the user.

Managing and Configuring Users

To list all existing server users, **type** `vlist` at the command prompt.

To change the full name, services offered, or quota for a user, follow these steps from the command line:

1. **Type** `vedituser` and press **Enter**
2. **Type** the name of the user in question and press **Enter**.
3. **Select** the services to offer the user. This prompt is a toggle-style one, with the FTP and email services enabled for the user by default. Available services include `ftp`, `mail`, and `shell`. **Type** the service name at the prompt and press **Enter** to toggle the service on or off. Press **Enter** when done.

To change a server user's password, follow these steps from the command line:

1. **Type** `passwd` *name*, where *name* is the username of the user in question. Press **Enter**.
2. **Type** the new password and press **Enter**. **Type** the new password again for confirmation and press **Enter**. If the two passwords do not match, you will need to type them again.

Note: Use a password that will not easily be guessed. A combination of uppercase and lowercase letters, numbers, and symbols is an example of a more secure password. If a particularly insecure password is entered, the system will prompt you to enter a more secure password.

Removing Users

To remove a server user, follow these steps from the command line:

1. **Type** `vrmuser` and press **Enter**.
2. **Type** the username of the user to remove. If the user does not exist, the system will indicate the user does not exist in the password database and exit the removal process.
3. The system will display the password entry for the user to be removed and ask you to confirm the removal of the user. **Type** `yes` if the information is correct or type `no` to exit the process. Press **Enter**.

Caution:
Use extreme caution when removing the home directory of users (step 4). You will not be warned if Web content or other important information is stored in the user's home directory.

4. The system will display the user's home directory and ask if you wish to remove it. **Type** `yes` and press **Enter** to remove the directory and complete the removal of the user, or type `no` and press **Enter** to remove the user, but keep the user's home directory and its contents.

Note: If you remove a user, but not the files or home directory they own, the files or directories of the removed user will be owned by a system-generated four digit ID (for example: 1007).

Configuring Email

All user email boxes reside in the `/var/spool/mail` directory. The system automatically creates email boxes for server users. If no other email routing settings (such as aliases or virtual user tables) are configured, by default the username of each user functions as a valid email for all domains that resolve to the server. For example, if *domain1.com* and *domain2.com* resolve to the server, for the user *bob*, both *bob@domain1.com* and *bob@domain2.com* would deliver email to the `/var/spool/mail/bob` email box. The system checks for matches in email routing and addresses in the following order and delivers to the first match it finds:

- Virtual user tables (or *virtusertable*)
- aliases
- users

This section discusses virtusertable, catchall, and alias configurations.

Note: To edit the virtusertable and aliases files, you must be the root user. You can become the root user by typing `su` at the command line and supplying the root user password.

Virtual User Tables

Virtual user tables (or *virtusertable*) route the full email address (both sides of the "@") to a local user, alias, or remote email address, but cannot direct to files or programs. Only aliases can route email to a file or program. A virtusertable consists of the virtusertable and its destination, or direction. For example, a virtusertable for *bob@domain1.com* could be directed to the user *bob* with the following virtusertable entry:

```
bob@domain1.com      bob
```

Virtusertable enables you to create email addresses without the need to create a corresponding user. For example, a virtusertable for *bob@domain2.com* could be directed to *bob@remotedomain.com* with the following virtusertable entry:

```
bob@domain2.com      bob@remotedomain.com
```

In this example, *remotedomain.com* would indicate a remote domain which does not resolve to your Linux VDS account.

Specify virtusertable in the `/etc/mail/virtusertable` file, with one virtusertable entry per line. The `virtusertable.sample` file provided with each server contains example virtusertable. You can change virtusertable settings by editing the `virtusertable` file and then executing the `vnewvirtusertable` command from the command prompt to load the virtusertable into the system.

Any time you make a change to the /etc/mail/virtusertable text file, you will need to create a db file that sendmail can read. The following command will create the `/etc/mail/virtusertable.db` file when run by a root user:

```
# makemap hash /etc/mail/virtusertable < /etc/mail/virtusertable
```

Catchalls

Caution:
Catchall virtusertable should be used sparingly. Spammers often send many emails to every conceivable address for a domain, often numbering in the thousands. A catchall virtusertable would cause the system to receive all of these emails and direct them to the recipient.

A catchall virtusertable directs all email not otherwise configured with a virtusertable for a given domain to a single recipient. For example, the following catchall virtusertable would direct all email not configured with another virtusertable for the domain *company.com* to the *joe* user:

```
@company.com   joe
```

If no virtusertable exists for an email address, and a catchall virtusertable is configured for the domain, the system would route all email sent to that address to the catchall virtusertable.

Aliases

Email aliases simply forward email to a user, another alias, email address, list of addresses, file, or program. Aliases enable you to create email addresses without needing to create a corresponding user. For example, the following alias would forward email sent to *webmaster@domain.com* to the *stan* user:

```
webmaster:  stan
```

Aliases also enable you to send email to a list or to a program. For example, the following alias would forward email sent to *sales@corporation.com* to the *bob*, *joe*, and *stan* users:

```
sales:  bob,joe,stan
```

For long email lists, place the emails in a file, one address per line, and use the *include* option. For example, the following alias would forward email sent to *promotion@website.com* to all addresses in the `/lists/promotion` file:

```
promotion:     :include:  /lists/promotion
```

Specify aliases in the `/etc/aliases` file, with one alias per line. You configure the aliases by editing the `aliases` file according to your needs. After making changes to the file, execute the `newaliases` command from the command prompt to load the aliases into the system.

Note: The system only considers the first portion of the email address (before `@`) for aliases. To direct an entire email address to a certain recipient, use a virtusertable.

Configuring Subhosts

Linux VDS enables you to configure multiple Web sites and domains in addition to the main domain of the server. The main domain or site for the server is called the hostname. Place the content for this site in the `/var/www/html` directory.

You may have configured a custom hostname during the order process. To aid with configuring and testing your account, all Linux VDS accounts receive a temporary domain name, or *temp domain*, which resolves to your Linux VDS account. This domain can be used if the hostname is inaccessible or does not yet point to your account.

Other domains or sites hosted by your account are called subhosts. This section explains adding, removing, and configuring subhosts.

Note: To execute the `vaddhost` command or edit the `httpd.conf` file, you must be the root user. You can become the root user by typing `su` at the command line and supplying the root user password.

Adding a Subhost

Subhosts are configured with the Apache Virtual Host directive. The Apache Web server looks for Virtual Host entries in the `/www/conf/httpd.conf` file. The `vaddhost` custom installation script assists in the creation of the subhost configuration Virtual Host tags in your Apache configuration file.

Subhost configuration includes many variables. While separated here into three sections, the entire process includes all three and you must complete all of them to entirely configure the subhost.

To begin the subhost configuration and specify the domain and administrator, follow these steps from the command line:

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 (*domain.com*, for example), any other variations of the domain (*www.domain.com*, *store.domain.com*, etc), and any other domains used for this subhost, pressing **Enter** after each. The first domain entered will be the main domain for the subhost. Additional variations will be aliases that point to the main domain. Press **Enter** without any text after providing all variations to move to the next step.

Note: The hostname or subhost usually consists of the top-level domain only (*domain.com*) instead of a canonical name such as *www.domain.com*. Canonical names are usually added as secondary domains or aliases.

3. The system displays the list of domains and variations for verification. **Type** `y` and press **Enter** to continue or **type** `n` and press **Enter** to input the domains again.
4. **Type** the username of the user who will administer the subhost and press **Enter**. (Press **Enter** without any text to display a list of existing users.) This user should be the owner of the site files and folders, otherwise the Web server will not be able to load the site.
5. Verify the information, **type** `y` and press **Enter** to continue or **type** `n` and press **Enter** to input the username again.

To continue the configuration and specify SSL, administrative email, and document root (Web directory) settings, follow these steps from the command line:

1. To enable SSL for the subhost **type** `y` and press **Enter** or **type** `n` and press **Enter** to not enable SSL for the subhost.
2. Verify the information and **type** `y` and press **Enter** to continue or **type** `n` and press **Enter** to input the SSL information again.
3. **Type** the email address of the subhost administrator and press Enter.
4. Verify the information and **type** `y` and press **Enter** to continue or **type** `n` and press **Enter** to input the address again.

5. **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, which will create a subhosted directory in the home directory of the user specified in step 4.
6. Verify the information and **type** `y` and press **Enter** to continue or **type** `n` and press **Enter** to input the path again.

To continue the configuration and specify log and 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 and **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 and **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 Virtual Host 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 abort 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.

Note: Press **ctrl+c** to exit the `vaddhost` process at any time. This immediately cancels `vaddhost` and any subhost configuration entered during the `vaddhost` process is lost.

Configuring a Subhost

You may configure a subhost further by editing the `Virtual Host` entries for the subhost in the `/www/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.

Removing a Subhost

The `/www/conf/httpd.conf` file contains the configuration for subhosts. To remove the subhost configuration from the Web server, follow these steps:

1. **Edit** the `httpd.conf` file and **remove** the `Virtual Host` entries for the subhost in question.
2. After modifying the `httpd.conf` file, **execute** the `restart_apache` command from the command line to restart the Web server.
3. **Remove** any unneeded files or directories for the subhost from the server.

Note: Please refer to “Removing Users” on page 3 for further information about subhost files and directories assigned to a user and considerations when removing that user

Using the Vinstall Utility Library

Your server contains another feature unique to Linux VDS which makes configuration of your account easier. The proprietary `vinstall` custom installation script enables you to install and configure programs for your account. Programs available for installation through `vinstall` have been configured specifically for Linux VDS and enable you to utilize a variety of programs without complicated manual configuration.

The `vuninstall` utility removes files and configurations created by the `vinstall` utility. This section discusses using `vinstall` and `vuninstall` to list available programs, find information about them, install, and/or remove programs.

You can execute the `vinstall` utility using interactive commands, or supply the needed information as command-line options to avoid the interactive `vinstall` prompts. Substitute `vuninstall` for `vinstall` when removing programs.

Note: Some programs prompt you to provide further information as part of their own installation process, even when using the non-interactive `vinstall` commands.

Refer to the following table for a list of the possible `vinstall` actions you can execute and the interactive and non-interactive commands for each:

Actions	Interactive Command	Non-interactive Command
Enter interactive mode	`vinstall`	n/a
List available programs	`?`	`vinstall -l`
Find additional information about a program	*program*	n/a
Enter install mode	`install`	n/a
Install a program	*program* (while in install mode)	`vinstall` *program*
Exit interactive mode	`quit`	n/a

Note: Press **ctrl+c** at any time to exit the interactive or non-interactive `vinstall` process. Use this option with caution as it immediately cancels `vinstall` and may leave installations and programs incomplete and non-functional.

Document Conventions

The conventions used in this document are designed to be completely predictable and are used for the following specific purposes.

Conventions List

Typeface	Usage
Italic	Used to indicate the following: • The first mention of new terms in any information unit. For example: The *rudaplex* and the *strataguide* have been the modified for this model. • References to titles of books, chapters, headings, CDs, diskettes, or software programs. For example: Refer to *The Technical Manual* for technical term descriptions. • Variables that the user types. For example: Type the *User ID* in the User ID text box.
Bold	Used to indicate the following: • Exact text strings typed. For example: Type **ABCDEFG**. • Keyboard keys pressed. For example: **Press Ctrl+a**, then press **Enter**.
Blue Underline	Used to indicate linked email, IP, Network, or Web addresses. For example: Go to http://www.microsoft.com for more information about Microsoft products.
Cross-Reference	Used to indicate a reference to another part of the same document. The grey portion of the cross-reference is hot linked to the appropriate section of the document, followed by a page number, also hot-linked to the same portion of the document. For example: For more information about the Document Conventions, see the "Document Conventions" on page 9.
`Operating System Text`	Used to indicate text that appears in a shell session for an operating system. The displayed text pertains to operating system text only, not application elements. For example: Type `LIST MAIN FOLDER`. The screen displays the Main folder.

`Program Code`	Used to indicate code listings. For example:
	`{`
	`# do something;`
	`}`
	`# check to see if $user has the attrib 'atrib'`
	`if (hasKey($user_obj, 'atrib', $dbh) != 1)`
	`{`
	`print "User not Authorized to update!";25`
	`}`

Window Element	Window elements consist of anything that is displayed on window (exclusive of the operating system). This includes toolbar menu items, drop-down lists and items in a drop-down list, buttons, or anything else a user sees on screen. For example: • From the Printer drop-down list, choose Local Printer. The Are You Sure? dialog box appears. Click OK. • The following message appears: User Not Authorized

Special Elements

These elements provide a variety of information ranging from warnings that the reader should not neglect to supplementary information and tips that will simply enhance their reading experience.

Tip: Used to point out helpful ideas, some not-so-obvious features, quick or alternate ways to get a particular job done, and techniques you might not discover by yourself. The **Tip List** special element is used when multiple tips are used.

Note: Used to highlight certain information for the reader. Generally, the Note element provides additional information on the current topic. The **Notes:** special element is used when multiple notes are required.

Important:
Used for information that is considered more pertinent to the reader than information presented in Note elements.

Caution:
Used as a hazard light in customer documents. Information included in a Caution element could save the reader from hours of lost work.