

---

# Migrating WEBppliance™ 2.x for Linux® to WEBppliance 3.1 for Linux

December 30, 2002

---

## *Legal and copyright notice*

### *About this guide*

Introduction .....	v
Intended audience.....	v
Printer-friendly version .....	vi
Required skills.....	vi
Related documentation .....	vi
Additional resources.....	vi
Document conventions.....	vii
Feedback and support .....	viii

### *Chapter 1*

#### *What's new in WEBppliance 3.1 for Linux*

Introduction .....	1-1
Benefits of migrating to WEBppliance 3.1 .....	1-1
Benefits of the ServerXchange 3.1.3 Private Server platform.....	1-3
Differences in directory structure .....	1-3
Changes between WEBppliance 2.x and 3.1 .....	1-4
For Appliance Administrators.....	1-4
For Site Administrators and User Administrators.....	1-5
Login changes.....	1-5
An example of a sample email to Site Administrators.....	1-6

---

## **Chapter 2**

### ***Before you begin***

Introduction .....	2-1
About WEBppliance 3.1 for Linux .....	2-1
System requirements .....	2-2
Source server .....	2-2
Target server .....	2-2
Disk space .....	2-2
Script requirements for migration .....	2-3
Preparing to migrate.....	2-5
Setting up adequate disk space on the source server.....	2-5
Setting up Service Plans on the target server .....	2-6
Known issues of the migration process.....	2-7
What is migrated? .....	2-8
Appliance level configuration .....	2-8
Domain data .....	2-9
Archived data.....	2-10
What is not migrated? .....	2-10

## **Chapter 3**

### ***Migrating to WEBppliance 3.1 for Linux***

Introduction .....	3-1
An overview of the migration process .....	3-2
About the migration .....	3-3
Enabling legacy mode.....	3-3
Installing the migration scripts.....	3-5
Using the migration scripts.....	3-6
Using dmexporter .....	3-6
Using dmimporter .....	3-8
Performing the migration.....	3-10
Troubleshooting migration issues .....	3-11
Post-migration tasks.....	3-12
Upgrading Miva Merchant from 4.00 to 4.12 .....	3-12
Repairing FrontPage extensions .....	3-13

---

# Legal and copyright notice

This document contains information proprietary to Ensim Corporation and its receipt or possession does not convey any rights to reproduce, disclose, manufacture, or sell anything it might describe. Reproduction, disclosure, or use without Ensim's specific written authorization is strictly forbidden.

Ensim Corporation makes no representations or warranties with respect to the contents or use of this document. It also reserves the right to revise this publication and make changes to the content at any time, without the obligation to notify any person or entity of such revisions or changes.

Further, Ensim Corporation assumes no responsibility or liability for any errors or inaccuracies, makes no warranty of any kind (express, implied or statutory) with respect to the contents or use of the information, and expressly disclaims any and all warranties of merchantability, fitness for particular purposes, and non-infringement of third party rights.

This product includes software developed by Digital Creations/Zope Corporation for use in the Z Object Publishing Environment ([www.zope.org](http://www.zope.org)).

Ensim<sup>®</sup>, ServerXchange<sup>™</sup>, and WEBppliance<sup>™</sup> are trademarks of Ensim Corporation.

Netscape<sup>®</sup> is a registered trademark of Netscape Communications Corporation.

Windows<sup>®</sup> is a registered trademark of Microsoft Corporation.

Linux<sup>®</sup> is a registered trademark of Linus Torvalds.

All other trademarks and copyrights are the property of their respective owners.

Copyright<sup>©</sup> 1999 - 2002 by Ensim Corporation. All rights reserved.

## **CORPORATE HEADQUARTERS**

Ensim Corporation  
1366 Borregas Avenue  
Sunnyvale, CA 94089  
(408) 745-3300

[www.ensim.com](http://www.ensim.com)

*WEBppliance for Linux\_Migration Guide - Version 3.1.3*





---

# About this guide

## Introduction

Migration is the process of transferring data or configuration files from one server to another. Migration is usually done when a software (such as WEBppliance™) is upgraded to a higher version. The migration process uses commands and scripts to extract data from one server, then transfer it to another server with the correct configuration.

This guide contains instructions on migrating or moving domains from WEBppliance 2.x to WEBppliance 3.1.

Chapters in this guide include:

- [Chapter 1, “What’s new in WEBppliance 3.1 for Linux”](#). This chapter encapsulates the features introduced by WEBppliance 3.1 for Linux and the benefits of migrating.
- [Chapter 2, “Before you begin”](#). This chapter describes the requirements to be fulfilled in order to successfully migrate to WEBppliance 3.1 for Linux.
- [Chapter 3, “Migrating to WEBppliance 3.1 for Linux”](#). This chapter provides instructions on migrating to WEBppliance 3.1 for Linux.



**Important:** The migration tools work with “live” data. Please ensure that you read and understand all the instructions before you start the migration process.

Refer to the topic [“Performing the migration” on page 3-10](#) for a complete list of steps involved in the migration process.

---

## Intended audience

This guide is designed to aid ISPs and Appliance Administrators in the planning and implementation of the migration process. Detailed, procedural steps are provided to guide users through the migration process.

---

## Printer-friendly version

This document is presented in HTML and PDF formats. If you are viewing it in HTML and want to print it in a book style, use the Acrobat® PDF version.

## Required skills

To use this guide, you need to be familiar with the Red Hat® Linux® platform as well as general Internet applications. In addition, you should have some networking knowledge.

## Related documentation

For more information about WEBppliance 3.1 including system requirements, known issues, resolved issues, setting up, and using WEBppliance, refer to the following documents.

- *WEBppliance 3.1.3 for Linux Release Notes*
- *Installing and Setting Up WEBppliance 3.1 for Linux*
- *WEBppliance 3.1 for Linux Technical Reference Guide*

These documents can be found at <http://www.ensim.com/support/wpls/index.html>.

## Additional resources

For information about specific tasks and procedures in ServerXchange and WEBppliance 3.1 refer to the following:

- *ServerXchange 3.1.3 Online Help*

This document can be found at <http://www.ensim.com/support/sxc/index.html>.

- *WEBppliance 3.1 for Linux Appliance Administrator's Online Help*
- *WEBppliance 3.1 for Linux Reseller Administrator's Online Help*
- *WEBppliance 3.1 for Linux Site Administrator's Online Help*
- *WEBppliance 3.1 for Linux User Administrator's Online Help*

These documents can be found at <http://www.ensim.com/support/wpls/index.html>.

---

## Document conventions

Throughout this guide, specific fonts are used to identify computer input, output, and interface elements. [Table 1](#) lists the typographic conventions used in this guide.

**Table 1. Typographic conventions**

Appearance of text	How it is used
Narrow bold text (sans serif font)	Used for button names, column names, field names, file names, keystrokes, menu items, and path names. <i>Example:</i> From the <b>File</b> menu, choose <b>Delete</b> , then click <b>OK</b> .
Bold, italic text (serif font)	Used for information you type. <i>Example:</i> Type <code>cd &lt;directory name&gt;</code> on the command line.
Courier (typewriter-style) font	Used for system messages and screen text. <i>Example:</i> The following message is displayed: <code>The server has been added.</code>

Key information is sometimes displayed using special headings and formats to make it stand out from regular text. The following special headings are used in this guide:



**Tip:** Information to help you perform procedures

---



**Note:** Supplemental information about tasks or concepts

---



**Important:** Supplemental information that is more important than a note

---



**Caution:** Warnings about potential damage to data or systems

---

---

## Feedback and support

At Ensim, we value your feedback and suggestions. You can help Ensim improve its documentation by providing feedback about this guide's contents, format, usability, and information in the online help.

### WEBpliance 3.1 for Linux (LH)

To take advantage of Ensim's support services or to find technical information quickly, visit the Ensim support page at <http://support.ensim.com>.

To obtain the latest product documentation or to utilize Ensim's self-help resources, visit the Ensim product support page at <http://www.ensim.com/support/sxc/index.html>.

If you need additional online support or if you would like to provide feedback about Ensim documentation, please enter a CaseConnect ticket at <https://onlinesupport.ensim.com>.

### WEBpliance 3.1 for Linux (LS)

To take advantage of Ensim's support services or to find technical information quickly, visit the Ensim support page at <http://support.ensim.com>.

To obtain the latest product documentation or to utilize Ensim's self-help resources, visit the Ensim product support page at <http://www.ensim.com/support/wpls/index.html>.

If you need additional online support or if you would like to provide feedback about Ensim documentation, please enter a CaseConnect ticket at <https://onlinesupport.ensim.com>.

# What's new in WEBppliance 3.1 for Linux

## Introduction

This chapter describes the benefits of migrating to WEBppliance 3.1 and highlights the differences between WEBppliance 2.x and 3.1.

Topics include:

- “Benefits of migrating to WEBppliance 3.1” on page 1-1
- “Benefits of the ServerXchange 3.1.3 Private Server platform” on page 1-3
- “Differences in directory structure” on page 1-3
- “Changes between WEBppliance 2.x and 3.1” on page 1-4
- “Login changes” on page 1-5
- “An example of a sample email to Site Administrators” on page 1-6

## Benefits of migrating to WEBppliance 3.1

Migrating to WEBppliance 3.1 offers the following benefits.

- **Reseller Administrator**  
Earlier versions of WEBppliance consisted of three levels (Appliance Administrator, Site Administrator, and User Administrator level). A fourth level called the Reseller Administrator has been introduced in WEBppliance 3.1 for Linux. Through the Reseller interface, the Reseller Administrator can easily manage Web hosting administrative tasks and set up sites without having to worry about the risk and effort involved in maintaining server hardware and software.
- **Online and context-sensitive Help in multiple languages**  
Online and context-sensitive Help is available at all levels—Appliance, Reseller, Site, and User. WEBppliance 3.1 also supports online Help in multiple languages. The online Help is accessible through the **Help** link located on the left navigation bar of the WEBppliance 3.1 interface. A Help icon is also present at the top-right hand corner of individual pages for context-sensitive Help.
- **Red Hat® Linux® 7.2 compatible**  
Upgrading your server to WEBppliance 3.1 makes your server Red Hat Linux 7.2 compatible.

- **Tomcat 4 support**

WEBpliance 3.1 supports Tomcat (v-4.0.3) developed by the Jakarta-Apache project. Tomcat 4.0.3 implements the final released versions of the servlet 2.3 and JSP 1.2 specifications. As required by the specifications, Tomcat 4.0.3 also supports Web applications built for servlet 2.2 and JSP 1.1 specifications, with no changes. Additionally, the 4.0.3 servlet container Catalina provides greater flexibility and performance.

- **Miva® Merchant™ 4.12 support**

WEBpliance 3.1 includes version 4.12 of the browser-based storefront and cataloging development system - Miva Merchant.

- **Repair option for protected directories**

Repair option for protected directories has been introduced in WEBpliance 3.1 for Linux. The repair option is used for:

- Finding errors in .htaccess files and repairing them
- Checking for deleted groups with access to protected directories and repairing the required files

- **Multilingual support in WEBpliance 3.1 for Linux**

WEBpliance 3.1 for Linux currently supports English and German as the default languages. Future versions of WEBpliance will support more languages.

At each of the four levels (Appliance Administrator, Reseller Administrator, Site Administrator, and User Administrator) the user can select the language preference. The language preference is login-based, that is, it lasts as long as the user is logged on.

- **Domain preview and shared-SSL access are now reachable through an improved URL**

In WEBpliance 3.1, the domain (both HTML and cgi-bin) preview and shared-SSL access are reachable through:

**`https://<appliance server>/<domain>/<url>`**, and  
**`https://<appliance server>/~<domain>/<url>`**.

- **Language preference for Analog reports at the domain level**

WEBpliance 3.1 for Linux supports language selection at the domain owner level for Analog reports only. Site Administrators can choose the language in which they want the Analog reports to be generated. Once a language is selected, it is used to generate Analog reports and the effect is visible the next day. The language preference can subsequently be changed.

- **Development environment available to sites**

Sites with shell access now have gcc, g++, make, and java available to them to develop applications.

## Benefits of the ServerXchange 3.1.3 Private Server platform

The ServerXchange 3.1.3 Private Server platform provides the following benefits.

- The Private Server is based on the Red Hat Linux 7.2 operating system, and it allows the installation of any standard Red Hat 7.x application.
- The Private Server's resources can be changed through the user interface without restarting the Private Server (except for swap).
- The Private Server's firewall capability allows for usage and management of ipchains and iptables.
- The Private Server's refined Linux kernel provides better virtual memory management, including swap space.

## Differences in directory structure

The differences in directory structure of WEBppliance 2.x and 3.1 are summarized in [Table 2](#).

**Table 2. Differences in directory structure of WEBppliance 2.x and 3.1**

Description	WEBppliance 2.x	WEBppliance 3.1
Document root	/home/httpd/html	/var/www/html/
CGI directory	/home/httpd/cgi-bin/	/var/www/cgi-bin/
The domain's file system is accessed through:	Either /home/virtual/<domain_name>/ or /home/virtual/site<x>/fst/ where <domain_name> is the name of the domain and site<x> is the site identifier as returned by the sitelookup utility. For more information about sitelookup, see "System requirements" on page 2-2.	Either /home/virtual/<domain_name>/ or /home/virtual/site<x>/fst/ where <domain_name> is the name of the domain and site<x> is the site identifier as returned by the sitelookup utility. For more information about sitelookup, see "System requirements" on page 2-2.



**Note:** When Site Administrators or site users log on to WEBppliance 3.1 using SSH, Telnet, or FTP, they are chroot(ed) into their own private file system. This prevents them from accessing the file system outside their own domain.

## Changes between WEBpliance 2.x and 3.1

Once the migration is complete, administrators will see changes specific to their level.

### For Appliance Administrators

Appliance Administrators will see the following changes.

- The URL for logging on to the new appliance will now be **http://<IP address or host name>/admin/**.
- The Site Administrator user name will no longer be a UNIX account and will not be tied to the file system, allowing Appliance Administrators to change the Site Administrator login name or use the same user name to log on to all domains. This is especially useful for Web developers who manage their customers sites.
- In WEBpliance 3.1, Site Administrator login names are restricted to a maximum of 16 characters. During migration, Site Administrator names that are longer than 16 characters are replaced by a random name using the format "wpYYYYMMDDHHmmSS." The following message also appears during the migration process.

```
(verylongsite.com) The Site Administrator name
verylongsiteadministratorname for this domain
verylongsite.com is too long, replacing it with
wp20020615000304. Please login as Appliance Administrator
and change the Site Administrator name for this domain
appropriately.
```

Once the migration is complete, log on to the target server as the Appliance Administrator and change the Site Administrator login name.



**Important:** Scripts that rely on the old Site Administrator names for authentication will fail after the migration. You must fix the scripts to reflect the new names.

- WEBpliance 3.1 allows configuration of Anonymous FTP for IP-based sites only. Anonymous FTP is not supported in WEBpliance 2.x.
- All Site Administrator names are required to be within 16 characters.



**Note:** After running dmimporter, Anonymous FTP will still be disabled for name-based sites. You can enable it from the Appliance Administrator interface.

---

## For Site Administrators and User Administrators

Site Administrators and User Administrators will see the following changes.

- The Web site can now be previewed using the URL **http://<applianceHOST>:PORT/~<domain.com>/**.
- Backward compatibility will be enabled for existing domains, however, new domains will need to comply with the new directory structure.
- The document root for a Web site will now be as follows:
  - HTML: /var/www/html (as opposed to the earlier /home/httpd/html)
  - CGI scripts: /var/www/cgi-bin (as opposed to the earlier /home/httpd/cgi-bin)
- Email messages will now be authenticated using SMTP, so email users will need to switch or change their email client server settings to **Server Requires Authentication to send mail** (depending on their email client).

## Login changes

Due to the isolation of domains and the complete virtualization of services, the FTP, SSH, Telnet, or POP/IMAP, the user name for WEBppliance 3.1 is now `<user_name>@<domain.com>`.



**Note:** It is possible that users may not be able to log on using the above format, in which case they should use `<user_name>#<domain.com>`.

---

## An example of a sample email to Site Administrators

After performing the migration, Site Administrators will experience certain changes in the functioning and use of the new WEBppliance. The following is an example of a sample email that you can send to your Site Administrators to inform them of the changes.



---

### **Example: Sample email to Site Administrator**

---

Dear Site Administrator,

We are pleased to inform you that your domain <domain.com> hosted with us has been upgraded to WEBppliance 3.1.

As a part of this upgrade, you will see the following changes.

- Your domain address is now `http://<appliance_server>/<domain_name>/`
- Your login name is now `<user_name>@<domain.com>`
- Your login name when using the FTP, SSH, Telnet, or POP/IMAP client, is now `<user_name>@<domain.com>`. The password remains unchanged.
- The new document root for your domain is:
  - HTML: `/var/www/html`
  - CGI scripts: `/var/www/cgi-bin`
- Your Miva Merchant installation has been upgraded and is now at version 4.12. To activate the new version, please access `http://<domain.com>/Merchant2/upgrade.mv?`
- You can now subscribe to new services and features such as Tomcat, SquirrelMail, and Development tools (GNU C/C++ compiler)
- Your email will now be authenticated using SMTP. You will have to switch or change your email client server settings to Server Requires Authentication to send mail (the actual settings depend on the email client that you are using).

If you have any questions, send an email to `<appliance_admin_email>`.

Sincerely,

Appliance Administrator



# Before you begin

## Introduction

This chapter describes the requirements of the migration process and provides informative guidelines on the migration process.

Topics include:

- “System requirements” on page 2-2
- “Preparing to migrate” on page 2-5
- “Known issues of the migration process” on page 2-7
- “What is migrated?” on page 2-8
- “What is not migrated?” on page 2-10

## About WEBppliance 3.1 for Linux

WEBppliance 3.1 for Linux upgrades the Apache® Web server software to the Apache 1.3.27 release. This release is recommended by Apache as the best of the 1.3 series.

The Apache 1.3.27 version is chiefly a security and bug fix release, particularly notable for addressing a few critical security vulnerabilities evidenced with previous versions.

The vulnerabilities as described in CAN-2002-0839, CAN-2002-0840 and CAN-2002-0843 are briefly described here.

**CAN-2002-0839.** This vulnerability allows malicious exploit of the Apache shared memory scoreboard format that can subsequently result in denial-of-service attacks.

**CAN-2002-0840.** This cross-scripting vulnerability enables remote attackers to execute scripts on the host server.

**CAN-2002-0843.** This vulnerability allows a malicious Web server to cause denial-of-service attacks and possibly execute arbitrary code via a long response.

For more details, please visit the [www.apache.org](http://www.apache.org).

WEBppliance 3.1 for Linux also addresses a few bugs evidenced in the previous releases, thus providing a more reliable operational environment.

## System requirements

To migrate domains from WEBppliance 2.x to WEBppliance 3.1, you will need to meet the following requirements.

### Source server

- A Private Server with WEBppliance 2.x installed. This is the source server for migration.

### Target server

- A Private Server or Dedicated Server with WEBppliance 3.1 installed. This is the target server for migration.

For more information about installing and configuring WEBppliance, see the *Installing and Setting up WEBppliance 3.1 for Linux* guide.



**Important:** Tomcat runs 20+ processes when started. If you plan to include Tomcat as one of the services, ensure that you have a PII Processor (Intel) or higher, and at least 512 MB RAM.

---

## Disk space

Sufficient free disk space is needed to store the exported data. This space may be located on the source server, the target server, or a different server. If using a different server, ensure that the server is accessible to both the source and the target server, either through a local directory (possibly a NFS mount point) or through FTP.

If a local directory (as opposed to an NFS mount point or a FTP server) is being used to store the exported data, then the source server must have free disk space at least equal to the sum total of the disk space consumed by all domains.

On the target server you will also need to take into account the space consumed by new services such as SquirrelMail and Miva Merchant. The disk space requirements for new services for a single domain are given in [Table 3](#).



**Important:** If adequate disk space is not reserved on the target server, dmimporter will fail. The minimum disk space required is 20\*N MB on the /

home partition of the target server, where N = Number of sites being imported on the target server.

**Table 3. Disk space requirements for a single domain**

Service	Disk space used
SquirrelMail	9 MB
Miva Merchant	11 MB

## Script requirements for migration

The following scripts are required to perform the migration tasks.

- dmexporter**

This script exports appliance and domain information from WEBppliance 2.x. For more information about dmexporter options, see [“Using dmexporter” on page 3-6](#).
- dmimporter**

This script imports appliance and domain information to WEBppliance 3.1. For more information about dmimporter options, see [“Using dmimporter” on page 3-8](#).
- sitelookup**

This script gives the Site ID and Site Administrator ID for a particular domain. The usage is as follows:

```
sitelookup -d <domain.com>
```

where <domain.com> is the domain for which you want to look up the IDs. The command returns the Site Administrator ID: *admin<xxx>* and Site ID: *site<xxx>*.
- vklookup**

When run on a ServerXchange Multi Tenant Server, this script gives the ID of a Private Server. The usage is as follows:

```
vklookup -n <hostname>
```

where <hostname> is the full name of the Private Server, including the domain name.
- hide\_service**

This script is located in /usr/bin and is used to hide a service on WEBppliance 3.1. The script usage is as follows:

```
hide_service <service_name>
```

where <service\_name> is the name of the service to be hidden.

- **EditPlan**

This script is located in `/usr/local/bin` and modifies a specified Service Plan. This script can be run on the target server before the migration to set up the Service Plans as desired. The usage is as follows:

```
EditPlan [-p <template plan name>] [-c
<service>,<option>=<value>,[enable/
disable],<option>=<value>...]...<plan name>
```

where

`<template plan name>` is the name of the template or the Service Plan you want to use as the foundation for the Service Plan. If this option is omitted, the name defaults to the Service Plan name given at the end of the command line.

`<service>` is the name of the service you want to change or include in this Service Plan.

`<option>` is optional features, if any, that you want to specify for the service.

`<value>` is what the option specifies, such as a name, password, or measurement. The value must be a single string.

`enable/disable` enables or disables the service. To enable a service, specify either `on` or `enabled=1`, to disable a service specify either `off` or `enabled=0`.

`<plan name>` is the name of the Service Plan you are changing.

For example, to disable SquirrelMail, Miva Merchant, and Tomcat in the WEBppliance 3.1 default Service Plan, type the following command.

```
EditPlan -c sqmail,off -c tomcat4,off -c mivamerchant,off
default
```

- **upgrade-mivamerchant**

This script is located in `/usr/bin` and upgrades Miva Merchant from version 4.00 to 4.12. Run this script after the migration is complete. For information about upgrading Miva Merchant, see the topic [“Upgrading Miva Merchant from 4.00 to 4.12”](#) on page 3-12.

- **repair-frontpage**

This script is located in `/usr/bin` and repairs broken FrontPage® extensions. Run this script if the `dmimporter` script indicates that some domains contain broken FrontPage extensions. For information about repairing FrontPage extensions, see the topic [“Repairing FrontPage extensions”](#) on page 3-13.

## Preparing to migrate

Before you migrate, ensure that you fulfill the requirements enumerated in the succeeding section.

### Setting up adequate disk space on the source server

The exporter tool requires sufficient disk space to be available to the existing Private Server to store the exported files. To allow extra space, you can:

- Provide write access to a FTP server with sufficient space available. This is the optimum approach.
- Double the size of the existing source Private Server using the ServerXchange 2.x ControlCenter. For more information see the topic *Customizing Private Servers* in *ServerXchange 2.5 User's Guide*
- Mount an NFS file system onto your Private Server and save the exported data at that location. Since Private Servers do not have permission to perform mounts, you must mount remote directories on a Private Server from a Physical Server.

To mount a remote directory follow these steps:

- a** Determine the ID of the Private Server by typing the following command on the Multi Tenant Server.

```
vklookup -n <hostname>
```

where *<hostname>* is the full name of the Private Server, including the domain name.

- b** Create a new directory on the Private Server to act as a mount point.

For example, /mnt/nfs.

- c** Mount the NFS share into the "treeroot" of the Private Server. For example,

```
mount bigserver.isp.net:/export/huge /ServerXchange/  
vpses/<id>/treeroot/mnt/nfs
```

where *<id>* is the ID of the Private Server as determined in Step a.

## Setting up Service Plans on the target server

The default Service Plans of WEBppliance 2.x and 3.1 differ in the following respects.

- There are new services in WEBppliance 3.1 (for example, SquirrelMail and Miva Merchant)
- The status (enabled or disabled) of existing services may be different

The default Service Plan of WEBppliance 2.x will be migrated with all options kept intact. However, since there are new services in WEBppliance 3.1, the default Service Plan of WEBppliance 3.1 will be an union of the old Service Plan and the new services. The status of the new services will either be enabled or disabled based on their default installation configuration.

It is recommended that before starting the migration you change the settings of the new services in the default Service Plan on the target server. It is also recommended that you disable services such as SquirrelMail, Miva Merchant, and Tomcat in the default Service Plan of the target server, as they consume a considerable amount of disk space. As [Table 3](#) indicates, SquirrelMail and Miva Merchant together consume 20 MB of disk space for every domain that has these services enabled. This would require the target server to have an additional free space of  $20*N$  MB, where N is the number of domains being migrated.

If you need any of these services for the migrated sites, then ensure that for sites running such services adequate disk space is allocated before migration. If adequate space is not allocated, then it results in a “site disk quota is exceeded” error, causing dmimporter to fail.

To change the settings for a Service Plan on WEBppliance 3.1, use the EditPlan script. For information about the EditPlan script, see the topic [“Script requirements for migration”](#) on [page 2-3](#).

## Known issues of the migration process

This section describes the known issues of the migration process.

- The migration does **not** support non-standard port numbers. Please ensure that all domains are at 80, the standard port number.
- The **StartServers** directive in the Apache configuration file specifies the number of processes that your Apache Web server will create at startup time. The recommended range of values for this directive (depending on resource availability) is between 5 and 15. Exceeding this range would cause your system to slow down or become unstable.
- Any version of Miva Merchant lower than 4.0 is not exported. After migration, you need to upgrade the Miva Merchant version from 4.00 to 4.12 for the migrated domains. For more information about upgrading Miva Merchant, see [“Upgrading Miva Merchant from 4.00 to 4.12” on page 3-12](#).
- The ProFTPD Servername directive is migrated to WEBppliance 3.1, but can no longer be configured from its interface.
- WEBppliance 3.1, in order to be more compliant with DNS standards, does not allow multiple Pointer (PTR) records for a domain. The importer will retain only a single PTR record. Usually, this is the one pointing to the Web server name. For example, www.domain.com, or domain.com, if Apache is disabled.



**Note:** A Pointer record creates a “reverse” or address-to-host name mapping.

- WEBppliance 2.x services (like Apache, BIND, Sendmail, Telnet, and IMAP) do not retain their original status (started/stopped) after migration.

## What is migrated?

The migration script exports appliance level configuration, domain data, and archived data from the WEBppliance 2.x server and creates files that include the following:

### Appliance level configuration

This section enumerates the appliance configuration files that are exported by the migration script.

- OpenSSL server key, certificate, and certificate request files
  - Files: all files under the /etc/httpd/conf/ssl.crt, ssl.csr, and ssl.key directories
- Administrator accounts
  - Files: /etc/opcenter/webhost/passwd
- Allowed IP ranges
  - Files: /etc/virtualhosting/ipranges
- Apache logs
  - Files: all files under /var/log/httpd
- Navigation bar customizations
  - Files:
    - /usr/lib/opcenter/images/logo.gif
    - /etc/opcenter/webhost/appliancemenu.shtml
    - /etc/opcenter/webhost/sitemenu.shtml
    - /etc/opcenter/webhost/usermenu.shtml
- GUI customizations
  - Files: /etc/opcenter/display.ini
- Site welcome pages
  - Files:
    - /etc/virtualhosting/templates/index.shtml
    - /etc/virtualhosting/templates/miva1.gif
    - /etc/virtualhosting/templates/blank.gif
    - /etc/virtualhosting/templates/elogo.gif
    - /etc/virtualhosting/templates/left.gif
    - /etc/virtualhosting/templates/powered.gif
    - /etc/virtualhosting/templates/top.gif

- Apache, Sendmail, and ProFTPD directives that can be set in WEBppliance 2.x
- BIND zones for all domains
- Time zone
- Message of the day
  - Files: /etc/motd
- Site Service Plans

## Domain data

This section enumerates the domain data files that are exported by the migration script.

- Service status (enabled or disabled)
- Web pages and CGI scripts
- OpenSSL server key, certificate, and certificate request files
  - Files: all files under the etc/httpd/conf/ssl.crt, ssl.csr, and ssl.key directories in the domain
- Administrator and user accounts
- User home directories and user mailboxes
- Apache and ProFTPD logs
  - Files: all files under var/log in the domain
- Analog usage data
  - Files: all files under var/usage in the domain
- Majordomo mailing lists
  - Files: all files under usr/lib/majordomo in the domain
- Sendmail aliases, responders, and spam filters
  - Files: etc/aliases in the domain, all files under usr/share/opcenter/responders in the domain

## Archived data

The following data is exported from the source WEBppliance and imported, but not configured to be used on the destination WEBppliance. This data is unlikely to function properly on the destination WEBppliance without some modification.

- Customizations scripts
  - Files: all files under the directory /usr/local/customized
- Style sheet
  - Files: /usr/lib/opcenter/base/common/style.dhtml

## What is not migrated?

The following is a list of items that are not migrated.

- Any plans that you have created on WEBppliance 2.x using the Customer and Billing Manger (CBM) module.
- Any customizations that you have made to the WEBppliance 2.x HTML pages. These will have to be manually migrated.
- Any custom RPM packages or Perl modules that you have installed on WEBppliance 2.x.
- Existing databases on WEBppliance 2.x are not migrated. All migrated domains will have a new and empty database created for them, with the database name the same as domain name. All period characters (.) in the database name will be replaced with underscore characters (\_). However, you can manually migrate a database from WEBppliance 2.x to 3.1 using the following script.

```
mysqldump -uroot -p<root_mysql_passwd> --opt <dbase_name> | mysql  
--host=<target_machine> -uroot -p<target_machine_root_mysql_passwd>  
-C <new_dbase_name>
```

where

- <root\_mysql\_passwd> is the MySQL root password on the source server
- <dbase\_name> is the existing database name on the source server
- <target\_machine> is the name of the target WEBppliance 3.1 server
- <target\_machine\_root\_mysql\_passwd> is the MySQL root password on the WEBppliance 3.1 server
- <new\_dbase\_name> is the new name of the database migrated to WEBppliance 3.1

# Migrating to WEBppliance 3.1 for Linux

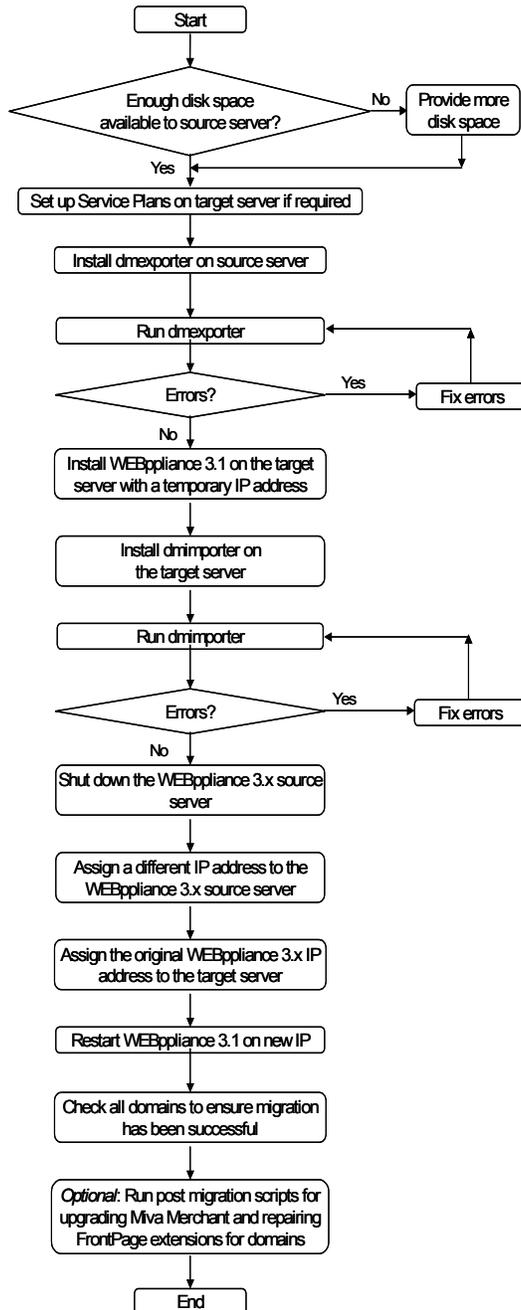
## Introduction

This chapter describes the migration process, post migration tasks, and legacy mode for migration.

Topics include:

- “An overview of the migration process” on page 3-2
- “About the migration” on page 3-3
- “Enabling legacy mode” on page 3-3
- “Installing the migration scripts” on page 3-5
- “Using the migration scripts” on page 3-6
- “Performing the migration” on page 3-10
- “Troubleshooting migration issues” on page 3-11
- “Post-migration tasks” on page 3-12

## An overview of the migration process



## About the migration

The migration process uses two scripts to move domains between servers.

- A Command Line script called **dmexporter** on WEBppliance 2.x to export the appliance and domain information for migration. For more information about using dmexporter see [“Using dmexporter” on page 3-6](#).
- A Command Line script called **dmimporter** on WEBppliance 3.1 to import the appliance and domain information. For more information about using dmimporter see [“Using dmimporter” on page 3-8](#).

## Enabling legacy mode

When domains are migrated from WEBppliance 2.x to WEBppliance 3.1, legacy mode is enabled on the WEBppliance 3.1 server. The purpose of legacy mode is to enable a site’s CGI scripts, which were developed under the WEBppliance 2.x environment with hardcoded paths, to run seamlessly in its new home on WEBppliance 3.1. Enabling legacy mode achieves this compatibility. After the migration:

- Sites will be created with the symlink `/home/virtual/<siteadmin name>`, in addition to the standard 3.1 symlink `/home/virtual/<domain>`
- A site’s file system will have the symlink `/home/httpd` available (pointing to `var/www`)
- Duplicate Site Administrator names will not allowed

Note that duplicate Site Administrator names are allowed in WEBppliance 3.1 when run in non-legacy mode.

The two symlinks are created for each migrated site and for all new sites that are created after the migration has completed.

The limitations of the legacy mode are as follows:

- Sites that have CGI scripts with hardcoded `/home/virtual/<siteadmin name>` paths will not function when the site is backed up and restored to another WEBppliance (using the backup and restore facility of WEBppliance 3.1).
- Future versions of WEBppliance may support creation of sites to locations other than `/home/virtual`. This is to support server configurations that have two or more partitions for user data. With this option, WEBppliance needs to be able to have the flexibility in choosing a root directory. Because of this, sites that rely on the legacy mode features must be modified to ensure their operation on newer versions of WEBppliance.
- Scripts that use hard coded paths, and work for the time being, may break in future versions of WEBppliance when legacy mode is deprecated.

Here are some suggestions on how to move away from legacy mode and to ensure your sites continue to work in the future versions.

- 1** Find the problem scripts: Appliance Administrators or Site Administrators can detect problem CGI scripts by running the following command.

```
fgrep -r -H "home/virtual" /home/virtual/site<n>/var/www/  
cgi-bin /home/virtual/site<n>/var/www/html
```

This shell command will do a string search on all files in the Web content directories for the string "home/virtual", assuming your scripts are written in interpreted languages like Perl or PHP.

- 2** Fix the problem scripts: Use the following environment variables instead of hardcoding the paths.

SITE\_ROOT- The root path of the domain

SITE\_CGIROOT- The top level CGI path of the domain

SITE\_HTMLROOT- The top level HTML CGI path of the domain

In Perl, to access these environment variables use the following:

```
$ENV{'SITE_ROOT'}
```

```
$ENV{'SITE_CGIROOT'}
```

```
$ENV{'SITE_HTMLROOT'}
```

In PHP, to access these environment variables use the following:

```
getenv("SITE_ROOT")
```

```
getenv("SITE_CGIROOT")
```

```
getenv("SITE_HTMLROOT")
```

- 3** Remove the following symbolic links.

```
/home/virtual/site<n>/home/httpd
```

```
/home/virtual/<siteadmin name for site<n>>
```

- 4** Test the site to ensure that CGI scripts for the site are working correctly.

## Installing the migration scripts



**Important:** Before installing the migration utilities from 2.x to WEBppliance 3.1 for Linux on a legacy Private Server (on a 2.2 MultiTenant Server), ensure that you update the RPM service on the Private Server to 4.0.2\_4, available on Ensim AppXchange™.

The **dmexporter** tool must be installed on WEBppliance 2.x source server and the **dmimporter** tool on WEBppliance 3.1 target server.



### Procedure: To download the tools using Ensim AppXchange

- 1 Connect to ServerXchange 3.1.3 and log on to the Server & Application Manager.
- 2 On the left navigation bar, click **Administration**.
- 3 Click the **Applications** tab.
- 4 Click the **AppXchange** link below the tabs.
- 5 Click **Connect to AppXchange**.  
The AppXchange interface opens in a new browser window.
- 6 Click **View** in the Web Hosting graphic box.
- 7 Select the check box in the **Action** column of the applications to mark them for downloading, namely, Ensim WEBppliance Export Tool [Legacy] and WEBppliance Import Tool.
- 8 When you have marked the applications, click on **Add selected items to cart**.  
The applications you marked for downloading appear in a list. To add more items click on **Add more items**. To clear the list and start again click on **Clear list**.
- 9 Click **Download**.
- 10 Read the license agreements, then click **Yes** if you agree to the terms.  
When the download is complete, AppXchange displays the results.
- 11 To return to the ServerXchange Server & Application Manager, click the **Logout** tab.
- 12 The downloaded services are registered and ready to install. They appear in the Services, Suites, or Packages lists.
- 13 Install the Ensim WEBppliance Export Tool [Legacy] service on a ServerXchange 2.5.2 Private Server.

- 14** Install the WEBpliance Import Tool service on a ServerXchange 3.1.3 Private Server or Dedicated Server.

## Using the migration scripts

Both the migration commands are executed from the Command Line. Use any Telnet or SSH client to access the server running WEBpliance and execute the required commands.

The syntax for these commands is described in the following sections.

### Using dmexporter

Use the **dmexporter** command to extract domain information from WEBpliance 2.x. and store it to a local or mounted NFS directory, or FTP directly to an FTP server.

#### Syntax

```
dmexporter [-l <log file>] -d <directory>
dmexporter [-l <log file>] [-p] -f ftp://[<username>[:<password>@]
<hostname>[:<port>]/[<path>/]
```



**Important:** The dmexporter command must be run as specified above, either with the -d, or -f options, but not both.

Table 4 lists the dmexporter options.

**Table 4. Options of dmexporter**

Option	Description
-l <log file>	The name of the log file, defaulting to dmexporter.log.
-d <directory>	Store the exported domains into the specified directory. The directory must exist and be empty.
-f <ftp_url>	Store the exported domains to the FTP server specified by the URL.
<username>	The username is optional, defaulting to anonymous.
<password>	The password is optional, defaulting to none.
<hostname>	The hostname is the hostname of the FTP server.
<port>	The port is optional, defaulting to 21.
<path>	The path is optional, and the default depends on the FTP server. The path must refer to a directory on the FTP server that exists and is empty.

**Table 4. Options of dmexporter**

Option	Description
-p	If -f is specified, use passive FTP.



**Note:** *<log file>* and *<directory>* are assumed to be relative to the current working directory, unless starting with */*.



### **Example: Using dmexporter to extract domain information**

The following is an example of the dmexporter command to extract domain information.

```
dmexporter -l mydom.log -d somedir
```

This command when run on a WEBppliance 2.x server, exports the appliance and domain information to a directory named somedir. This directory must be created before executing the command and must be empty. The log file created is named mydom.log.

```
dmexporter -f ftp://admin:testy123@ftpserver.com/tmp/export
```

This command when run on a WEBppliance 2.x server, exports the appliance and domain information to a FTP server named ftpserver.com in a directory named /tmp/export. This directory must exist and be empty. The log file created is named dmexporter.log.

## Using dmimporter

Use the **dmimporter** command to import the extracted domain information into the WEBppliance 3.1 target server.

### Syntax

```
dmimporter [-l <log_file>] -d <directory>
dmimporter [-l <log_file>] [-p] -f ftp://[<username>[:<password>@]
<hostname>[:<port>]/[<path>/]
```



**Note:** The dmimporter command must be run as specified above, either with the -d, or -f options, but not both.

Table 5 lists the dmimporter options.

**Table 5. Options of dmimporter**

Option	Description
-l <log_file>	The name of the log file, defaulting to dmimporter.log.
-d <directory>	Get the exported domains from the specified directory.
-f <ftp_url>	Store the exported domains to the FTP server specified by the URL.
<username>	The username is optional, defaulting to anonymous.
<password>	The password is optional, defaulting to none.
<hostname>	The hostname is the hostname of the FTP server.
<port>	The port is optional, defaulting to 21.
<path>	The path is optional, and the default depends on the FTP server. The path must refer to a directory on the FTP server that exists and is empty.
-p	If -f is specified, use passive FTP.



**Note:** *<log file>* and *<directory>* are assumed to be relative to the current working directory, unless starting with */*.



**Important:** The `dmimporter` command may print error messages while importing domains. For more information, see the topic [Troubleshooting migration issues](#).



### **Example: Using `dmimporter` to import domain information**

The following is an example of the `dmimporter` command to import the extracted domain information into the WEBpliance 3.1 target server.

```
dmimporter -l somedom.log -d somedir
```

This command when run on a WEBpliance 3.1 server, imports the appliance and domain information from a directory named `somedir`. The log file created is named `somedom.log`.

```
dmimporter -f ftp://admin:testy123@ftpserver.com/tmp/export
```

This command when run on a WEBpliance 3.1 server, imports the appliance and domain information from a FTP server named `ftpserver.com` from a directory named `/tmp/export`. The log file created is named `dmimporter.log`.

## Performing the migration

Follow the steps as under to migrate WEBppliance 2.x for Linux to WEBppliance 3.1 for Linux.



**Caution:** Please do not attempt to import data from two different WEBppliance 2.x source servers onto the same WEBppliance 3.1 target server.



### Procedure: To migrate to WEBppliance 3.1 for Linux

- 1 Install the **dmexporter** tool on WEBppliance 2.x. For more information, see [“Installing the migration scripts” on page 3-5](#).
- 2 Extract domain information from WEBppliance 2.x using **dmexporter**. For more information about using dmexporter, see [“Using dmexporter” on page 3-6](#).
- 3 Create a new WEBppliance 3.1 on a temporary IP address.
- 4 Install the **dmimporter** tool on WEBppliance 3.1. For more information, see [“Installing the migration scripts” on page 3-5](#).



**Important:** If you have hidden a service on the WEBppliance 2.x server, you must hide the same service on the WEBppliance 3.1 server (using the `hide_service` command) before you import the migrated data. For more information about the `hide_service` command, see [“Script requirements for migration” on page 2-3](#).

- 5 If the domains have been exported to a local directory on the source server, create a new directory on the target server and copy the exported data from the source server to this directory.



**Important:** Do not interrupt the `dmimporter` script. This may lead to some of the sites being in an inconsistent state. If `dmimporter` is interrupted due to any reason, it is recommended that you reinstall WEBppliance 3.1 on the target server and perform the migration again.

- 6 Import all domains to WEBppliance 3.1 using the **dmimporter** tool. For more information about using dmimporter, see [“Using dmimporter” on page 3-8](#).
- 7 Shut down WEBppliance 2.x.

- 8** Assign a different IP address to WEBppliance 2.x.
- 9** Assign the original WEBppliance 2.x IP address to the WEBppliance 3.1 target server.
- 10** Restart the target server on the new IP address.
- 11** Test the migration. To test:
  - a** Access the Appliance Administrator interface and check the status of all domains.
  - b** Access the domain and check the status of all services.

## Troubleshooting migration issues

The migration scripts, `dmimporter` and `dmexporter`, will output error messages if problems are encountered during execution. The `dmimporter` script may print the following error message.

```
<appliance> This name server is not authoritative for the given zone  
ERROR:This name server is not authoritative for the given zone
```

This error message does not affect the migration in any way and can be ignored.

You may also see error messages for domains with broken FrontPage extensions. After migrating the domains, run the `repair-frontpage` script to fix these domains. For more information, see the topic [“Repairing FrontPage extensions” on page 3-13](#).

Additionally, more information about the process is recorded in the log files. You can specify a log file name while executing the command. If no file name is specified, the default log file created by the `dmexporter` script is named `dmexporter.log` and the one created by the `dmimporter` script is named `dmimporter.log`.

## Post-migration tasks

To complete the migration process, you have to upgrade Miva Merchant from version 4.00 to 4.12, and repair broken FrontPage extensions for any domain as indicated by the dmimporter script.

### Upgrading Miva Merchant from 4.00 to 4.12

After the migration, you need to upgrade Miva Merchant from version 4.00 to 4.12 for the migrated domains. New domains created on the target WEBppliance server will automatically use Miva Merchant 4.12. The upgrade is a two step process that involves both Appliance Administrators and Site Administrators. Both these steps have to be followed for each domain that needs to be upgraded to Miva Merchant 4.12.



---

#### **Procedure:** To upgrade Miva Merchant for a domain

The Appliance Administrator must follow these steps.

- 1 Connect to the WEBppliance server using any Telnet or SSH client and log on as root.
- 2 Upgrade Miva Merchant. The upgrade can be done one domain at a time or all domains at once.

To upgrade one domain at a time:

```
/usr/bin/upgrade-mivamerchant -d <domain_name>
```

To upgrade all domains at once:

```
/usr/bin/upgrade-mivamerchant -a
```

The Site Administrator must click <http://domain.com/Merchant2/upgrade.mv?> to complete the upgrade from Miva Merchant 4.00 to 4.12.



**Note:** The following URLs will not be accessible until the Site Administrator completes the upgrade as described above:

<http://domain.com/Merchant2/admin.mv?>

<http://domain.com/Merchant2/merchant.mv?>

---

---

## Repairing FrontPage extensions

During the migration process, the dmimporter script may indicate that some migrated domains have broken FrontPage extensions. If this happens, you have the option of either ignoring these messages (for example, if the domains do not really intend to use FrontPage even though it is enabled), or repairing the FrontPage extensions.



---

### Procedure: To repair FrontPage extensions

- 1 Connect to the WEBppliance server using any Telnet or SSH client and log on as root.
- 2 You can either repair broken FrontPage extensions domain by domain, or repair all domains at once.

To repair FrontPage extensions domain by domain, run the following command.

```
/usr/bin/repair-frontpage -d <domain_name>
```

To repair FrontPage extensions for all domains at once, run the following command.

```
/usr/bin/repair-frontpage -a
```



**Note:** Repairing the FrontPage extensions in a domain renames all existing .htaccess files in the document root of the domain to .htaccess.old in the same directory.

---

