vFabric Web Server Installation and Configuration

VMware vFabric Web Server 5.3

VMware vFabric Suite 5.3

This document supports the version of each product listed and supports all subsequent versions until the document is replaced by a new edition. To check for more recent editions of this document, see http://www.vmware.com/support/pubs.

vmware®

You can find the most up-to-date technical documentation on the VMware Web site at: https://www.vmware.com/support/.

The VMware Web site also provides the latest product updates.

If you have comments about this documentation, submit your feedback to: docfeedback@vmware.com

Table of Contents

1. About vFabric Web Server Installation and Configuration

Revised November 13, 2013.

VMware vFabric Web Server Installation and Configuration describes product concepts and product-specific configuration tasks for VMware® vFabric™ Web Server. Fully compatible with Apache Web Server, vFabric Web Server is a dynamic load-balancing service that is available as a standalone product and as a part of VMware® vFabric Suite™ Standard and vFabric Suite Advanced editions. Read this documentation for an overview of vFabric Web Server features, installation instructions, and information on how to configure functionality that is specific to vFabric Web Server.

Intended Audience

vFabric Web Server Installation and Configuration is intended for experienced Windows and Unix developers and system administrators who want to install a Web Server to serve static Web content, act as a proxy, or load balance between application servers such as VMware® vFabric™ tc Server.

2. Overview of vFabric Web Server

vFabric Web Server is the Web server and load-balancing component of VMware vFabric Suite. vFabric Web Server 5.3 is based on Apache HTTP Server version 2.2.

In addition to the standard features of Apache HTTP Server, vFabric Web Server provides the following benefits:

- Ability to easily install multiple instances of vFabric Web Server running on a single computer.

- Scalable management of multiple Web sites and servers; you can run and manage hundreds of instances of vFabric Web Server.

- Support for heterogeneous environments (32- and 64-bit architectures): Red Hat Enterprise Linux (RHEL), Solaris, Ubuntu, and Microsoft Windows. vFabric Web Server can be updated and patched across all servers at once. See Supported Configurations and System Requirements for details.

Subtopics

Information for International Users

Complete Packages and Modules in vFabric Web Server 5.3

Differences Between vFabric Web Server and vFabric ERS

Information for International Customers

If you require multi-byte character-set support, VMware recommends that you follow these guidelines for using multi-byte filenames and multi-byte characters in configuration files. These guidelines provide the most flexibility and most consistent results:

- Use the UTF-8 encoding when running a Unix terminal or Windows PowerShell. For example, to set this encoding on Unix, you can use the following environment variable:

```
LANG=en_US.UTF-8
```

On Windows, you can run the following command in your PowerShell window:

```
PS prompt> chcp 65001
```

Important: The preceding Windows command does not always work as expected on pre-R2 versions of Windows 2008. For this reason, VMware strongly recommends that you install vFabric Web Server on Windows 2008 R2.

Setting your encoding to UTF-8 results in better display of `httpctl` output, easier examination of the log files, and so on.

- Edit and save all vFabric Web Server configuration files, such as `conf/httpd.conf`, in UTF-8 format. This improves the legibility of the Web Server access and error log files.

Complete Packages and Modules in vFabric Web Server 5.3

This section lists the complete contents of vFabric Web Server 5.3.

This document describes features and functionality for vFabric Web Sever 5.3. For general information about new features in Apache HTTP Server 2.2, see the Apache Web site.

- Apache Web Server httpd 2.2.24

- Apache mod_ftp 0.9.6

- Apache mod_fcgid 2.3.7

- Apache mod_jk 1.2.37

- Apache APR Library 1.4.6

- Apache APR-util Library 1.3.12

- Apache tcnative connector

- cURL 7.29

- Expat 2.1.0

- GNU libiconv 1.11

- mod_bmx 0.9.4 (Hyperic plug-in for monitoring support)

- OpenSSL 1.0.1e

- OpenSSL/FIPS 2.0

- OpenLDAP 2.3.43

- PCRE 8.32

- zlib 1.2.7

You typically install vFabric Web Server from the `vfabric-web-server-version-platform` package, which contains all the preceding compiled modules. For your convenience, VMware also makes a `vfabric-web-server-devel-version-platform` package available which you can use to build HTTPD modules. Typically, you install the `-devel` package only on development computers, and not on production computers.

mod_fcgid Implementation of Connector to FastCGI

The mod_fcgid distributed with vFabric Web Server 5.3 is an implementation of the connector to FastCGI applications. This module allows the user to provision FastCGI providers such as PHP or Ruby on Rails from third parties, running out-of-process from the server itself.

Many applications can be built to support FastCGI; consult your language or application documentation for details. The application providing FastCGI services is launched by mod_fcgid on the initial request, and reused for subsequent requests to that application or language environment.

For details on configuring an application, including the number of persistent processes created, see [Apache Module mod_fcgid](#).

Differences Between vFabric Web Server and vFabric ERS

The vFabric Cloud Application Platform includes two HTTP server and load-balancing products: vFabric Web Server and vFabric Enterprise Ready Server (ERS). vFabric ERS is nearing its end-of-life and VMware highly recommends that ERS customers migrate to vFabric Web Server. The following table describes the major differences between the two products and provides high-level actions that existing ERS customers can take as they prepare for the migration.

Table 2.1. Differences Between vFabric Web Server and vFabric ERS

vFabric Web Server	vFabric ERS	Customer Action
Available as part of vFabric Suite (Standard and Advanced) or as a standalone product.	Not included in vFabric Suite. Originally designed for only physical computers.	Select licensing based on vFabric integration or dedicated hardware.
Strictly an Apache HTTPD Server-based product. vFabric tc Server, a separate	Includes both Apache HTTPD and Apache Tomcat packages.	Migrate ERS Apache HTTPD instances to vFabric Web Server. Separately migrate ERS Tomcat instances to vFabric tc Server.

vFabric Web Server	vFabric ERS	Customer Action
product, is strictly an Apache Tomcat-based product.		
Runs on current, vendor-supported 32- and 64-bit releases of RHEL, Microsoft Windows, Solaris, and AIX. See [Supported Configurations and System Requirements](#) for the exact versions.	Runs on now-unsupported, or "twilighted", versions of RHEL, Windows, Solaris, AIX, and HPUX. See [ERS Supported Platforms](#) for the exact versions.	Upgrade to a vendor-supported operating system version for all vFabric Web Server instances, and apply all patch releases (such as service packs) issued by that vendor no later than 12 months from their vendor release.
Includes the current enterprise-ready release of Apache HTTPD Server 2.2.	Includes the current release of Apache HTTPD Server 2.2, as well as the now-deprecated 2.0 and 1.3 versions.	Migrate all Apache HTTPD 2.0 and 1.3 instances to vFabric Web Server 2.2 instances. The migration requires updates to the `*.conf` file.
During installation or upgrade, the Apache HTTPD binaries are written to a path in the format `vfabric-web-server/httpd-2.2.xx.x-32`. This preserves any existing Apache HTTPD binaries without overwriting them.	During installation or upgrade, the Apache HTTPD binaries are always written to the same directory (`ers-install-path/apache2.2-64`), which means on upgrade any existing binaries are overwritten.	Point all server instances to the common symlink `vfabric-web-server/httpd-2.2`, modify it to revert/roll back/change 32-64 bit modes.
Includes the most commonly-used modules. See [Complete Packages and Modules in vFabric Web Server 5.3](#).	Included additional modules, such as mod_perl, mod_php, and mod_snmp.	Migrate PHP and Perl applications to the supported, and more optimal, mod_fcgid environment.
Closely tracks Apache Software Foundation (ASF) naming and directory layout conventions. In particular: • `install-dir`/httpd-2.2/modules/ directory contains the loadable modules • Binaries and configuration file names use `httpd` prefix • `install-dir`/newserver creates new instances • `instance-dir`/bin/httpdctl controls each deployed instance • `instance-dir`/conf/extras/ offers feature-based small config templates	Has a number of now-stale, legacy file and path conventions. In particular: • `install-dir`/apache2.2/modules/standard directory contains the loadable modules • Binaries and configuration file names use `httpsd` prefix • `install-dir`/ers-server.pl creates new instances • `install-dir`/servers/`instance-dir`/bin/apache_startup.sh|bat controls each deployed instance • `install-dir`/servers/`instance-dir`/conf/httpsd.conf is one large, monolithic configuration template.	Create a new vFabric Web Server instance, then migrate customizations from your existing vFabric ERS instance. Alternatively, modify a copy of the deployed vFabric ERS instance tree to use vFabric Web Server path and file name conventions. Use smaller functional `.conf` snippets to make the configuration more organized and maintainable.

3. Installing vFabric Web Server

vFabric Web Server has several options for installation and setup. Installation options vary according to whether you install Web Server standalone or as part of vFabric Suite installation and whether your operating system is Unix, Windows, or another supported platform.

Subtopics

Installation Note for vFabric Suite Customers

Available Distribution Packages

RHEL: Install vFabric Web Server from an RPM

Ubuntu: Install vFabric Web Server from a Debian Package

Unix: Install vFabric Web Server from a Self-Extracting ZIP

Windows: Install vFabric Web Server from a Self-Extracting ZIP File

Activate a vFabric Web Server Local License

Description of the vFabric Web Server Installation

Installation Note for vFabric Suite Customers

If you will be installing vFabric Web Server as part of a vFabric Suite Standard package, complete the procedures for installing vFabric License Server and activating vFabric Suite licenses in *Getting Started with vFabric Suite*. You may also want to read the other vFabric licensing sections in that guide, to understand how licensing works with vFabric Suite.

Available Distribution Packages

vFabric Web Server is split into the following two distribution packages to simplify the installation and deployment to your datacenter:

- `vfabric-web-server`: Base package entirely sufficient for all production environments.

- `vfabric-web-server-devel`: Supplemental package that developers can use to compile and link http modules with the same headers and libraries as httpd itself using the `httpd-2.2/bin/apxs` tool. This package requires that you also install the base package.

RHEL: Install vFabric Web Server from an RPM

VMware recommends that you install vFabric Web Server on a Red Hat Linux Enterprise (RHEL) computer by first installing the VMware RPM repository and then using `yum` to perform the actual installation. See Install vFabric Web Server from the VMware RPM Repository.

You can also download the RPM from the VMware Download page and install it on your RHEL computer using the `rpm` command, as described in Install vFabric Web Server from a Downloaded RPM. This installation option is necessary if you are installing a version of Web Server that is not yet certified for vFabric Suite, which means that it has not yet been added to the VMware RPM repository.

Install vFabric Web Server from the VMware RPM Repository

VMware recommends that you install Web Server on RHEL computers using the VMware RPM repository.

Prerequisites

- Verify that your system meets the supported configurations and installation requirements. See Supported Configurations and System Requirements.

- Install the vFabric repository RPM, which makes it easier for you to browse the vFabric RPMs, including the vFabric Web Server RPM. The vFabric repository contains all RPMs that are certified with this release of vFabric Suite. You install the vFabric repository RPM on each RHEL computer on which you want to install one or more vFabric components, such as vFabric Web Server.

  1. On the RHEL computer, start a terminal as the `root` user.

  2. Install the vFabric repository RPM using the following `wget` command.

      Important: You must run the entire `wget` command on a single line. Be sure you include the `|` `sh` at the end, or the RPM installation will not work.

      ```
      prompt# wget -q -O - packages.pivotal.io | sh
      ```

      The command performs the following tasks:

      - Imports the vFabric GNU Privacy Guard (GPG) key.

      - Installs the vFabric 5.3 repository RPM.

      - Launches the VMware End User License Agreement (EULA) acceptance and repository configuration script.

      - Outputs the EULA for you to read; you must answer `yes` to accept the terms and continue.

  3. Use the `yum search vfabric` or `yum search vmware` command to view the list of vFabric components that you can install from the VMware repository. For example (output truncated for clarity):

      ```
      prompt# yum search vfabric
      ...
      ======================================== Matched: vfabric ========================================
      vfabric-rabbitmq-java-client-bin.noarch : The RabbitMQ Java Client Library
      vfabric-rabbitmq-server.x86_64 : The RabbitMQ server
      vfabric-tc-server-standard.noarch : VMware vFabric tc Server Standard
      vfabric-web-server.x86_64 : VMware vFabric Web Server
      ...
      ```

      The vFabric Web Server RPM is called `vfabric-web-server`.

Procedure

1. From the RHEL computer on which you will install vFabric Web Server, log in as the `root` user and start a terminal.

2. Execute the following `yum` command:

    ```
    prompt# yum install vfabric-web-server
    ```

    The `yum` command begins the install process, resolves dependencies, and displays the packages it will install.

    The `yum` command automatically chooses the appropriate RPM package based on your architecture (32- or 64-bit).

    To install the developer's package:

    ```
    prompt# yum install vfabric-web-server-devel
    ```

3. Enter `y` at the prompt to begin the actual installation.

    If the installation is successful, you see a `Complete!` message at the end.

What the yum install command does

The `yum install` command:

- Installs vFabric Web Server into the `/opt/vmware/vfabric-web-server` directory.

- Sets the owner of installation directory, along with all child directories and files, to `root:root.`

What to do next

- If you installed Web Server standalone, activate a local Web Server license as described in Activate a vFabric Web Server Local License.

  If you installed vFabric Web Server as part of vFabric Suite, you should have already activated licensing as described in the procedures for installing vFabric License Server and activating vFabric Suite licenses in *Getting Started with vFabric Suite*. (You may also want to read the other vFabric licensing sections in that guide, to understand how licensing works with vFabric Suite.)

- Read Description of the vFabric Web Server Installation for a brief tour of what was installed.

- Create and start using a vFabric Web Server as described in Creating and Using vFabric Web Server Instances.

Install vFabric Web Server From a Downloaded RPM

You can install vFabric Web Server on RHEL by downloading the RPM from the VMware download center and executing the `rpm` command. This procedure is necessary if you want to install a version of Web Server that has not yet been certified for vFabric Suite.

Prerequisites

- Verify that your system meets the supported configurations and installation requirements. See Supported Configurations and System Requirements.

Procedure

1. Log in to the RHEL computer on which you will install vFabric Web Server as the `root` user.

2. Browse to the VMware Download page and download the Web Server RPM file to a directory on your computer. The RPM file is called `vfabric-web-server-version.architecture.rpm`.

3. Start a terminal and change to the directory in which you downloaded the RPM.

4. Execute the following `rpm` command to install Web Server:

   ```
   prompt# rpm -ivhf vfabric-web-server-version.architecture.rpm
   ```

5. In the previous section, see What the yum install command does for post-installation information, such as the installation directory. (The `yum` install command corresponds to the `rpm` command in this procedure.)

See What to do next for the suggested next steps.

Ubuntu: Install vFabric Web Server from a Debian Package

It is recommended that you install vFabric Web Server on a computer running Ubuntu by first installing the vFabric Debian package repository and then using `apt-get` to perform the Web Server installation.

Prerequisites

- Verify that your system meets the supported configurations and installation requirements. See Supported Configurations and System Requirements.

- Install the vFabric Debian package repository that contains the vFabric Web Server `*.deb` packages.

  1. On the Ubuntu computer, start a terminal either as the `root` user or as an unprivileged user who has `sudo` privileges.

2.   Install the vFabric Debian package repository using the following `wget` command:

```
prompt# wget -q -O - http://packages.pivotal.io/ | sh
```

If necessary, use `sudo` to run the preceding commands if you are not logged in as the `root` user. Use the following `sudo` command:

```
prompt$ wget -q -O - packages.pivotal.io/ | sudo sh
```

The command performs the following tasks:

- Imports the vFabric GNU Privacy Guard (GPG) key.

- Installs the vFabric Debian package repository.

- Launches the VMware End User License Agreement (EULA) acceptance and repository configuration script.

- Outputs the EULA for you to read; you must answer `yes` to accept the terms and continue.

3.   Use the `apt-cache search vfabric` command to view the list of vFabric components that you can install from the vFabric Debian package repository. For example:

```
prompt# apt-cache search vfabric
vfabric-tc-server-standard - Commercial Java application server based on Apache Tomcat
vfabric-web-server - VMware vFabric Web Server
vfabric-web-server-devel - VMware vFabric Web Server Libraries and Headers
vfabric-repo-precise - vFabric EULA and APT repo configuration utility
...
```

Procedure

1.   From the Ubuntu computer on which you will install vFabric Web Server, log in as the `root` user (or as an unprivileged user who has `sudo` privileges) and start a terminal.

2.   Execute the following `apt-get` command:

```
prompt# apt-get install vfabric-web-server
```

The `apt-get` command begins the install process, resolves dependencies, and displays the packages it will install.

If necessary, use `sudo` to run the preceding command if you are not logged in as the `root` user. For example:

```
prompt$ sudo apt-get install vfabric-web-server
```

To install the supplemental developer's package:

```
prompt# apt-get install vfabric-web-server-devel
```

What the apt-get install command does

The `apt-get install` command:

- Installs vFabric Web Server into the `/opt/vmware/vfabric-web-server` directory.

- Sets the owner of the directory, along with all child directories and files, to `root:root`.

What to do next

- If you installed Web Server standalone, activate a local Web Server license as described in [Activate a vFabric Web Server Local License](#).

If you installed vFabric Web Server as part of vFabric Suite, you should have already activated licensing as described in the procedures for installing vFabric License Server and activating vFabric Suite licenses in *Getting Started with vFabric Suite*. (You may also want to read the other vFabric licensing sections in that guide, to understand how licensing works with vFabric Suite.)

- Read Description of the vFabric Web Server Installation for a brief tour of what was installed.

- Create and start using a vFabric Web Server as described in Creating and Using vFabric Web Server Instances.

Unix: Install vFabric Web Server from a Self-Extracting ZIP

You can install vFabric Web Server on Unix computers with a self-extracting ZIP file that you download from the VMware Web site. Self-extracting zip files expand themselves, or you can use `unzip` if your platform supports it. Using `unzip` explicitly enables you to specify options in addition to what is executed by default when the ZIP self-extracts.

Prerequisites

- Verify that your system meets the supported configurations and installation requirements. See Supported Configurations and System Requirements.

- If you want to use `unzip` so you can specify additional options, but your platform does not support `unzip`, obtain an unzip command.

- If your operating system is configured to support only 64-bit operation, an external unzip utility is required. Do not use the `jar` utility to unpack these zip files, because the file system permissions will not be unpacked correctly.

- Be sure you have installed Perl on your computer, and that it is at least version 5.8. VMware recommends that you run the Perl script (described in the procedure) in multibyte character encoding such as UTF-8, especially if you are an international customer.

Procedure

1. Log in as the `root` user on to the Unix computer on which you want to install vFabric Web Server.

2. Create the directory in which you will install vFabric Web Server. For example:

   ```
   prompt# mkdir /opt/vmware
   ```

3. Download the appropriate vFabric Web Server self-extracting ZIP from the VMware Download Web site and place it in the directory you created.

   Be sure to choose the correct Unix operating system and chip architecture. For example, the file to install vFabric Web Server on a 64-bit Linux platform is `vfabric-web-server-version-x86_64-linux-glibc2.zip.sfx`.

4. From your terminal window, change to the directory in which you downloaded the ZIP file:

   ```
   prompt# cd /opt/vmware
   ```

5. If necessary, change the permissions of the downloaded ZIP file to make it executable:

   ```
   prompt# chmod 755 vfabric-web-server-version-x86_64-linux-glibc2.zip.sfx
   ```

6. Self-extract the files from the downloaded ZIP by using the file name as a command. For example:

   ```
   prompt# ./vfabric-web-server-version-x86_64-linux-glibc2.zip.sfx
   ```

   When it completes, the vFabric Web Server files are located in the `vfabric-web-server` subdirectory.

7. Change to the `vfabric-web-server` directory and run the `fixrootpath.pl` Perl script to correct the root paths. For example:

```
prompt# cd vfabric-web-server
prompt# perl fixrootpath.pl
```

What to do next

- If you installed Web Server standalone, activate a local Web Server license as described in Activate a vFabric Web Server Local License.

  If you installed vFabric Web Server as part of vFabric Suite, you should have already activated licensing as described in the procedures for installing vFabric License Server and activating vFabric Suite licenses in *Getting Started with vFabric Suite*. (You may also want to read the other vFabric licensing sections in that guide, to understand how licensing works with vFabric Suite.)

- Read Description of the vFabric Web Server Installation for a brief tour of what was installed.

- Create and start using a vFabric Web Server as described in Creating and Using vFabric Web Server Instances.

Windows: Install vFabric Web Server from a ZIP File

You install vFabric Web Server on Windows computers using a self-extracting ZIP file (`*.zip.exe`) that you download from the VMware Web site.

Prerequisites

- Verify that your system meets the supported configurations and installation requirements. See Supported Configurations and System Requirements.

- As of version 5.3 of vFabric Web Server, you must use Windows PowerShell 2.0 to execute the vFabric Web Server scripts on Windows computers. Most recent Windows versions, such as Windows 7, have PowerShell installed by default, but some older versions of Windows do not. To check whether your version of Windows has PowerShell installed, go to Start > All Programs > Accessories and check for `Windows PowerShell` in the list.

  If Windows PowerShell 2.0 is not installed on your Windows computer, download and install it from the Windows Management Framework (Windows PowerShell 2.0, WinRM 2.0, and BITS 4.0) Web page on the Microsoft Support.

- If necessary, enable Windows PowerShell for script processing; by default, script processing is disabled.

  1. Start PowerShell from the Start Menu as an Administrator by opening Start > All Programs > Accessories > Windows PowerShell, then right-clicking on Windows PowerShell and selecting Run as Administrator. A PowerShell window starts.

  2. Check the current PowerShell setting by executing the following command:

     ```
     PS prompt> Get-ExecutionPolicy
     ```

     If the command returns `Restricted`, it means that PowerShell is not yet enabled. Enable it to allow local script processing at a minimum by executing the following command:

     ```
     PS prompt> Set-ExecutionPolicy RemoteSigned
     ```

     You can choose a different execution policy for your organization if you want, as well as enable PowerShell using Group and User policies. Typically, only the Administrator will be using the vFabric Web Server scripts, so the `RemoteSigned` execution policy should be adequate in most cases.

Procedure

1. From the Windows computer on which you want to install vFabric Web Server, log in as the `Administrator` user.

2. Start PowerShell from the Start Menu as an Administrator by opening Start > All Programs > Accessories > Windows PowerShell, then right-clicking on Windows PowerShell and selecting Run as Administrator. A PowerShell window starts.

3. Create the directory into which you will install vFabric Web Server. Do not create a directory name that contains spaces. For example:

    ```
    PS prompt> mkdir c:\opt\vmware
    ```

4. Download the appropriate vFabric Web Server self-extracting ZIP from the VMware Download Web site and place it in the directory you created.

    Be sure to choose the correct architecture (32- or 64-bit). For example, the file to install vFabric Web Server on a 64-bit Windows platform is `vfabric-web-server-version-x64-windows.zip.exe`.

5. Execute the downloaded `*.zip.exe` file to self-extract the files into the directory you created. You can do this, for example, by opening Window Explorer, navigating to the directory, and double-clicking on the `*.zip.exe` file.

    When the extraction completes, the vFabric Web Server files are located in the `vfabric-web-server` subdirectory.

6. From your command window, change to the main vFabric Web Server directory:

    ```
    PS prompt> cd c:\opt\vmware\vfabric-web-server
    ```

7. Run the `fixrootpath.ps1` PowerShell script to correct the root paths:

    ```
    PS prompt> .\fixrootpath.ps1
    ```

8. Create a symbolic link from the existing `httpd-2.2.version` directory to one called `httpd-2.2`, where `version` refers to the minor version and architecture of vFabric Web Server.

    PowerShell does not include `mklink` intrinsically, so you must use the command `cmd /c mklink`. For example, to use the 64-bit edition of vFabric Web Server, run the following command:

    ```
    PS prompt> cmd /c mklink /d httpd-2.2 httpd-2.2.23.0-64
    ```

What to do next

• If you installed Web Server standalone, activate a local Web Server license as described in Activate a vFabric Web Server Local License.

  If you installed Web Server as part of vFabric Suite, you should have already activated licensing as described in the procedures for installing vFabric License Server and activating vFabric Suite licenses in *Getting Started with vFabric Suite*. (You may also want to read the other vFabric licensing sections in that guide, to understand how licensing works with vFabric Suite.)

• Read Description of the vFabric Web Server Installation for a brief tour of what was installed.

• Create and start using a vFabric Web Server as described in Creating and Using vFabric Web Server Instances.

Activate a vFabric Web Server Local License

When you purchase a local production vFabric Web Server license, your account manager provides you with one or more serial numbers. This section describes how to activate this local license.

Important: This section does not apply if you purchased vFabric Web Server as part of vFabric Suite. To activate licensing for vFabric Suite (including vFabric Web Server), complete the procedures for installing vFabric License Server and activating vFabric Suite licenses in *Getting Started with vFabric Suite*. You may also want to read the other vFabric licensing sections in that guide, to understand how licensing works with vFabric Suite.

Procedure

1. On the computer on which you installed Web Server, paste the serial number into a file named `vf.ws-serial-numbers.txt`.

2. Save this file to the directory appropriate to your operating system:

   • Windows: `%ALLUSERSPROFILE%\VMware\vFabric\`

   • Unix: `/etc/opt/vmware/vfabric/`

3. Ensure that the file is readable by the user who runs the Web Server instance. Because this user is typically `root`, no additional steps are typically required on Unix. Windows users who adjust the vFabric Web Server service to run as an account other than `LocalSystem` will need to verify these permissions.

Description of the vFabric Web Server Installation

The main vFabric Web Server directory structure, although similar in many ways to the standard Apache HTTP directory layout, differs from it in a very fundamental way: vFabric Web Server separates the runtime binaries from the configuration data.

To implement this separation, you use the `newserver` command to create a vFabric Web Server *instance* that lives in a subdirectory of the main vFabric Web Server home directory. The name of the instance is the name of the new subdirectory. You then configure this instance as you want, using the standard Apache httpd files in the `server-name/conf` directory, such as `httpd.conf`.

> Perform all configuration work inside the server instance (`server-name/conf`) directory. Never modify any files under the binary directory (such as `httpd-2.2`).

Keeping the runtime binary files apart from the files that are configured by administrators or end users makes it easier to upgrade or apply patches to the code without the risk of overwriting or corrupting user data. It also enables administrators to run multiple server instances independently.

Directly after installing vFabric Web Server, you see the following files and directories:

• `httpd-2.2/`: Symbolic link to a sibling directory that actually contains the Apache 2.2 binary runtime files.

• `httpd-2.2.version/`: Directory that contains the actual Apache 2.2 binary runtime files. The `2.2.version` string specifies the Apache HTTP version, such as `2.2.23.0` or `2.2.23.0-64`.

• `licenses/`: EULA and open source license files.

• `newserver`: (Unix) Perl script for creating vFabric Web Server instances.

• `newserver.ps1`: (Windows) PowerShell script for creating vFabric Web Server instances.

• `fixrootpath.pl`: (Unix) Perl script for fixing root paths; run only once.

• `fixrootpath.ps1`: (Windows) PowerShell script for fixing root paths; run only once.

After you use `newserver` to create a new vFabric Web Server instance, the command creates a new directory that contain a separately configurable instance of vFabric Web Server. An *instance* is a complete, discrete server configuration. You can create multiple instances. You can run multiple instances at the same time if you are careful not to use the same ports in two different instances. For example, the default HTTP listening port is 80, and only one instance on any computer is allowed to communicate on port 80 at any one time. So if you wanted to have two vFabric Web Server instances running at the same time on the same computer, you would configure one instance to use a port other than 80.

Each instance directory contains subdirectories that contain all the data required to run a given vFabric Web Server instance. This includes configuration data as well as all other data that is associated with that instance's configuration. For example, assume you installed vFabric Web Server in `/opt/vmware/vfabric-web-server` and create an instance called `myserver`:

```
prompt$ cd /opt/vmware/vfabric-web-server/myserver
prompt$ ls
bin  cgi-bin  conf  ftpdocs  htdocs  logs  proxy ssl  var
```

The `conf` directory contains the vFabric Web Server configuration files, such as `httpd.conf`. The `bin` directory contains the startup script used to start and stop the `myserver` instance (`httpdctl`. Each of these directories is specific to the `myserver` instance. Each instance that you create has a similar set of directories.

4. Upgrading vFabric Web Server

The instructions for upgrading vFabric Web Server differ slightly depending on your operating system and how you installed the original version. See:

*   RHEL: Upgrade Using the VMware RPM Repository

*   Unix: Upgrade Using a Self-Extracting ZIP

*   Windows: Upgrade Using a Self-Extracting ZIP

*   Ubuntu: Upgrade Using the vFabric Debian Package Repository

RHEL: Upgrade Using the VMware RPM Repository

You use `yum upgrade` to upgrade an existing RHEL installation of vFabric Web Server to the latest version. It is assumed that you used `yum install` to originally install the older version.

The `yum upgrade` process:

*   Installs the latest version of the Apache HTTPD Web Server binaries in a new directory (such as `httpd-2.2.24.0`) and removes the older versions.

*   Adjusts the `httpd-2.2` symbolic link to point to the new binaries.

*   Automatically updates all existing Web Server instances to start using the new binaries.

Prerequisites

*   Stop all running Web Server instances in the installation that you are upgrading. See Unix: Start and Stop vFabric Web Server Instances

*   If necessary, install the vFabric repository RPM that contains the version of vFabric Web Server to which you are upgrading. For example, vFabric Web Server 5.3 is certified for vFabric Suite 5.3, so install the 5.3 vFabric repository RPM.

Procedure

1.   From the RHEL VM on which you will upgrade vFabric Web Server, log in as the `root` user.

2.   Execute the following `yum` command:

```
prompt# yum upgrade vfabric-web-server
```

The `yum` command begins the upgrade process, resolves dependencies, and displays the packages it plans to upgrade.

3.   Enter `y` at the prompt to begin the actual upgrade.

When the upgrade process finishes, you will see a `Complete!` message at the end. Check the output of the command to ensure that the upgrade was successful.

Unix: Upgrade Using a Self-Extracting ZIP

Upgrading vFabric Web Server on Unix is very similar to installing, except you unzip the self-extracting `ZIP` file into the existing vFabric Web Server installation directory.

Prerequisites

*   Stop all running Web Server instances in the installation that you are upgrading. See Unix: Start and Stop vFabric Web Server Instances

- Read the Unix installation prerequisites which also apply to upgrading. See <u>Unix: Install vFabric Web Server from a Self-Extracting ZIP</u> .

Procedure

1. Download the appropriate vFabric Web Server self-extracting ZIP from the <u>VMware Download</u> Web site and place it in the *parent* directory of the main Web Server installation directory (`vfabric-web-server.`)

    For example, if you originally installed Web Server in `/opt/vmware/vfabric-web-server`, which in turn means that the HTTPD binaries are located in `/opt/vmware/vfabric-web-server/httpd-2.2.`*`version`*, then place the downloaded ZIP file in `/opt/vmware`.

    Be sure to choose the correct Unix operating system and chip architecture. For example, the file to install vFabric Web Server on a 64-bit Linux platform is `vfabric-web-server-`*`version`*`-x86_64-linux-glibc2.zip.sfx`.

2. Log in as the `root` user on to the Unix computer on which you want to upgrade vFabric Web Server and start a terminal.

3. Change to the parent directory of the main Web Server installation directory, which is also the directory in which you downloaded the ZIP file of the new Web Server version. For example:

    ```
    prompt# cd /opt/vmware
    ```

4. If necessary, change the permissions of the downloaded ZIP file to make it executable:

    ```
    prompt# chmod 755 vfabric-web-server-version-x86_64-linux-glibc2.zip.sfx
    ```

5. Self-extract the files from the downloaded ZIP by using the file name as a command. For example:

    ```
    prompt# ./vfabric-web-server-version-x86_64-linux-glibc2.zip.sfx
    ```

    At the prompt to replace `vfabric-web-server/fixrootpath.pl`, answer `All`.

    When it completes, the new vFabric Web Server files are located in the `vfabric-web-server` subdirectory. If the new version of Web Server includes a more recent version of the Apache HTTPD binaries, you will see a new corresponding directory, such as `vfabric-web-server/httpd-2.2.24.0-64`.

6. Change to the `vfabric-web-server` directory and run the `fixrootpath.pl` Perl script to correct the root paths and adjust the httpd-2.2 symbolic link:

    ```
    prompt# cd vfabric-web-server
    prompt# perl fixrootpath.pl
    ```

When you next start your existing Web Server instances, they will automatically use the new Apache HTTPD binaries you just installed.

Windows: Upgrade Using a Self-Extracting ZIP

Upgrading vFabric Web Server on Windows is very similar to <u>installing</u>, except you unzip the self-extracting `ZIP` file into the existing vFabric Web Server installation directory.

Prerequisites

- Stop all running Web Server instances in the installation that you are upgrading and uninstall them as Windows services. See <u>Windows: Start and Stop vFabric Web Server Instances</u>

- Read the Windows installation prerequisites which also apply to upgrading. See <u>Windows: Install vFabric Web Server from a Self-Extracting ZIP</u> .

Procedure

1. Log in as the `Administrator` user on the Windows computer on which you want to upgrade vFabric Web Server.

2. Start PowerShell from the Start Menu as an Administrator by opening Start > All Programs > Accessories > Windows PowerShell, then right-clicking on Windows PowerShell and selecting Run as Administrator. A PowerShell window starts.

3. Download the appropriate vFabric Web Server self-extracting ZIP from the VMware Download Web site and place it in the *parent* directory of the main Web Server installation directory (`vfabric-web-server`.)

    For example, if you originally installed Web Server in `c:\opt\vmware\vfabric-web-server`, which in turn means that the HTTPD binaries are located in `c:\opt\vmware\vfabric-web-server\httpd-2.2.`*version*, then place the downloaded ZIP file in `c:\opt\vmware`.

    Be sure to choose the correct architecture (32- or 64-bit). For example, the file to install vFabric Web Server on a 64-bit Windows platform is `vfabric-web-server-`*version*`-x64-windows.zip.exe`.

4. Execute the downloaded `*.zip.exe` file to self-extract the files into the installation directory. You can do this, for example, by opening Window Explorer, navigating to the directory, and double-clicking on the `*.zip.exe` file.

    At the prompt to replace any files, answer `All`.

    When the extraction completes, the new vFabric Web Server files are located in the `vfabric-web-server` subdirectory. If the new version of Web Server includes a more recent version of the Apache HTTPD binaries, you will see a new corresponding directory, such as `vfabric-web-server\httpd-2.2.23.0-64`.

5. From your PowerShell window, change to the main Web Server installation directory. Following our example:

    ```
    PS prompt> cd c:\opt\vmware\vfabric-web-server
    ```

6. Run the `fixrootpath.ps1` PowerShell script to correct the root paths using the `cscript` command:

    ```
    PS prompt> .\fixrootpath.ps1
    ```

7. Remove the existing `httpd-2.2` symbolic link and recreate it so that it points to the new Apache HTTP binary directory.

    PowerShell does not include `mklink` intrinsically, so you must use the command `cmd /c mklink`. For example:

    ```
    PS prompt> rmdir -Force httpd-2.2
    PS prompt> cmd /c mklink /d httpd-2.2 httpd-2.2.23.0-64
    ```

When you next start your existing Web Server instances, they will automatically use the new Apache HTTPD binaries you just installed.

What to do next

*   Re-install your existing Web Server instances as Windows services; see Windows: Start and Stop vFabric Web Server Instances .

    Important: As of version 5.2 of vFabric Web Server, the name of the script to control 5.2 and later Web Server instances changed from `httpdctl.bat` to `httpdctl.ps1`. The PowerShell script is now UTF-8 compatible, and it requires that you use Windows PowerShell. However, if you upgraded a pre-5.2 Web Server installation to 5.2 or later, you continue to use `httpdctl.bat` to control your pre-5.2 Web Server instances; refer to the appropriate 5.0 or 5.1 vFabric Web Server documentation.

Ubuntu: Upgrade Using the vFabric Debian Package Repository

As of version 5.3.1, vFabric Web Server provided Debian packages in a repository from which you could install on an Ubuntu computer. If you originally installed Web Server on the Ubuntu computer using `apt-get install`, then you use `apt-get install` again to upgrade the installation to the latest version.

The `apt-get upgrade` process:

- Installs the latest version of the Apache HTTPD Web Server binaries in a new directory (such as `httpd-2.2.24.0`) and removes the older versions.

- Adjusts the `httpd-2.2` symbolic link to point to the new binaries.

- Automatically updates all existing Web Server instances to start using the new binaries.

Prerequisites

- Stop all running Web Server instances in the installation that you are upgrading. See [Unix: Start and Stop vFabric Web Server Instances](#)

Procedure

1. From the Ubuntu computer on which you will upgrade vFabric Web Server, log in as the `root` user or as an unprivileged user who has `sudo` privileges.

2. Execute the `apt-get update` command so your computer is synchronized with recent changes to the vFabric Debian package repository:

   ```
   prompt# apt-get update
   ```

   If necessary, use `sudo` to run the preceding command if you are not logged in as the `root` user:

   ```
   prompt$ sudo apt-get update
   ```

3. Upgrade Web Server using the `apt-get install` command:

   ```
   prompt# apt-get install vfabric-web-server
   ```

   The `apt-get` command begins the upgrade process, resolves dependencies, and displays the packages it plans to upgrade.

   Enter `y` at the prompt to begin the actual upgrade.

5. Migrating Enterprise Ready Server to vFabric Web Server

This section has information about migrating Enterprise Ready Server (ERS) to to vFabric Web Server. For information about migrating ERS to vFabric tc Server, see Migrate an ERS Tomcat Instance to tc Server.

VMware currently provides two supported distributions of the open-source Apache HTTPD Server: vFabric Enterprise Ready Server (ERS) and vFabric Web Server, which is part of vFabric Suite. vFabric Web Server is fully supported by VMware on common platforms such as Red Hat Enterprise Linux and Microsoft Windows, and support is available for additional, less-common platforms. See Supported Configurations and System Requirements. Please contact VMware support if your platform is not listed on the vFabric Web Server product download page.

vFabric Enterprise Ready Server is deprecated. VMware will no longer provide support for HTTPD version 2.0 as of November 30, 2013 and HTTPD version 2.2 as of November 30, 2014. For more information, see VMware Support Policies. Please contact your account manager to learn about converting your existing active ERS licenses to vFabric Web Server licenses.

This document describes how to migrate Enterprise Ready Server instances to use vFabric Web Server. Although each product is based on the standard, open-source Apache HTTP Server, there are differences in VMware's implementations. See Differences Between vFabric Web Server and vFabric ERS.

This document describes the migration procedure for Enterprise Ready Server based on Apache version 2.2. You can use the same procedures to migrate from Apache version 2.0, but be aware of additional changes. See Upgrading to 2.2 from 2.0.

Subtopics

Preparing to Migrate

Migration Procedure

Preparing to Migrate

Before you migrate, read the information in this section. You may also want to review Differences Between vFabric Web Server and vFabric ERS.

Audit your Configurations

Migration from ERS to vFabric Web Server is an opportune time for you to examine your vFabric Web Server configurations for best practices and to update them where necessary. Note the following recommendations:

• VMware recommends that your deployment use the latest security modules and procedures.

• VMware recommends that you use the latest version of plug-in modules.

  VMware does not guarantee compatibility of third-party plug-in modules between ERS and vFabric Web Server. VMware recommends that you recompile third-party modules using the Developer build of vFabric Web Server. You may need to contact your vendor to obtain the latest versions of the modules.

• VMware recommends that you use up-to-date SSL libraries and that you review your SSL cipher configurations and protocol directives. .

• VMware recommends that you use modular configuration files.

  VMware recommends that you use the `Include` directive to add external configurations to your `httpd.conf` file. When you use the `Include` directive to reference external configuration files, the `httpd.conf` file is more readable and your configurations are easier to manage. For example, the following directive references an external configuration file:

```
Include conf/extra/httpd-languages.conf
```

The default `httpd.conf` file that is created when you create a new vFabric Web Server instance contains examples of including external configuration files. The external files are located in the *`<instance>`*`/conf/extra` directory) and are referenced with `Include` directives in the main `httpd.conf` file.

For more information, see the [Apache HTTPD](#) documentation.

To view a summary of Web server instance configurations, sorted by module, use the following procedure to enable the **mod_info** module to display configuration information for your Enterprise Ready Server or vFabric Web Server instances:

1. Make sure the **mod_info** module is enabled in your configuration file (`httpsd.conf` for ERS and `httpd.conf` for vFabric Web Server) with a configuration similar to the following. (You may need to change the path to include the correct path for 32-bit or 64-bit installations.)

   ```
   LoadModule info_module "<installation_directory>/apache2.2-64/modules/
   standard/mod_info.so"
   ```

2. Add the following code block to the configuration file:

   ```
   <Location /hidden-info>
       SetHandler server-info
   </Location>
   ```

3. Restart the server instance.

4. Use a Web browser to open the following URL:

   ```
   http://<server_hostname:port>/hidden-info
   ```

Directory Structure Changes

The directory structure of a vFabric Web Server installation closely matches a standard Apache 2.2 installation, whereas vFabric ERS uses a more proprietary file-naming and directory convention. Migrating to vFabric Web Server requires that you change configuration files and other scripts in your environment to reflect the new naming conventions. These changes make it easier to integrate third-party extensions with vFabric Web Server.

Table 5.1, "Directory Structure and File Name Changes" describes the changes in directory structure, configuration file names, and script file names.

Table 5.1. Directory Structure and File Name Changes

Purpose	Enterprise Ready Server	vFabric Web Server 5.3

Apache modules	*`<install_dir>`*`/`
	`httpd-2.2.`*`<version>`*`/modules/`
	`standard`
	
	`<install_dir>``/`
	`httpd-2.2.`*`<version>`*`/modules/`
	`covalent`
	
	`<install_dir>``/`
	`httpd-2.2.`*`<version>`*`/modules/`
	`jakarta`
	
	`<install_dir>``/`
	`httpd-2.2.`*`<version>`*`/modules/`
	`perl`
	
	`<install_dir>``/`
	`httpd-2.2.`*`<version>`*`/modules/`
	`php`
		`<install_dir>``/httpd-2.2`*`<version>`*`/`
		`modules`
		
		All modules reside in this single directory. Note that the **mod_perl** and **mod_php** modules are not included with vFabric Web Server.
		
		See [Apache Module Changes](#).

Purpose	Enterprise Ready Server	vFabric Web Server 5.3
Configuration file	`<web_server_instance>`/conf/ `httpsd.conf`	`<web_server_instance>`/conf/ `httpd.conf` Additional configuration files for various Apache modules are located in the `<web_server_instance>`/conf/extras subdirectory.
Script for creating new Apache HTTPD server instances	Linux Perl script: `<install_directory>`/ers-server.pl	Linux symlink to Perl script:`<install_directory>`/newserver
	Windows Perl script: `<install_directory>`\ers-server.pl	Windows PowerShell script:`<install_directory>`\newserver.ps1
Start-up script	Linux: `<web_server_instance>`/ bin/apache-startup.sh	Linux: `<web_server_instance>`/bin/ `httpdctl`
	Windows: `<web_server_instance>`\bin \apache-startup.bat	Windows PowerShell script: `<web_server_instance>`\bin \httpdctl.ps1
Tools	`<install_directory>`/tools/*	`<install_directory>`/ `httpd-<version>`/bin/*
Apache HTTPD binaries	HTTPD binaries are written to the installation directory. When the server is upgraded, the binaries are overwritten.	HTTPD binaries are written to a directory in the following format: `vfabric-web-server/httpd.-2.2.`*`xx.x-`*`bb`. (Where *`xx.x`* is the minor version number and *`bb`* is either 32 or 64 (for 32-bit or 64-bit versions). When the server is upgraded, the binaries are not overwritten.
Windows service names	`ERS <instance name> httpsd`	`vFabric httpd <instance name>`

Apache Module Changes

The compiled Apache modules for vFabric Web Server differ from those that are compiled and provided with Enterprise Ready Server. VMware recommends you substitute the modules described in Table 5.2, "Apache Modules". If necessary, you can compile and implement the modules yourself using the Developer build of vFabric Web Server. To ensure compatibility, VMware recommends that you recompile all third-party modules. See Complete Packages and Modules in vFabric Web Server 5.3.

Table 5.2. Apache Modules

Module	Changes in vFabric Web Server
mod_perl	VMware recommends that you use the `mod_fcgid` module to serve Perl-based applications. For more information, see the following article: FastCGI rocks mod_perl and FCGID.
mod_php (ERS)	The `mod_php` module is not included with vFabric Web Server. VMware recommends that you use the `mod_fcgid` module to serve PHP-based applications. For more information on the benefits of using `mod_fcgid`, see Apache with fcgid: acceptable performance and better resource utilization.
mod_snmp (ERS)	VMware recommends that you use the Apache BMX framework to provide internal runtime information about your vFabric Web Server instances to monitoring systems such as vFabric Hyperic. The following modules are included and are enabled by default: • `mod_bmx` • `mod_bmx_status` • `mod_bmx_vhost`

Module	Changes in vFabric Web Server
	You can disable this functionality by commenting out the `LoadModule` directives for the BMX modules in the `httpd.conf` file. For more information, see Configure BMX for Monitoring vFabric Web Server Instances.
mod_bmx (vFabric Web Server)	`mod_bmx` is a module in vFabric Web Server. For a sample configuration, see the following directory in a newly created vFabric Web Server instance: `<instance>/conf/extras/conf/extra/httpd-info.conf`
mod_jk	Although `mod_jk` is still included with Apache version 2.2, it has limitations on header fields sizes, cannot be secured with SSL cryptography, and requires separate configuration of proxy worker pools. VMware recommends that you use `mod_proxy_balancer`, which supports both the ajp protocol used by `mod_jk`, and also the http and https protocol to legibly decipher the back-end connection stream and secure the back-end connections to Tomcat and tc Server. `mod_proxy_balancer` is configured using the `httpd.conf` file syntax. For `mod_proxy` configuration options, see `http://httpd.apache.org/docs/2.2/mod/mod_proxy.html`. For configuration instructions, see Configure Load Balancing Between Two or More tc Runtime Instances.
mod_version	The `mod_version` module is compiled into vFabric Web Server. If your ERS `httpdsd.conf` file contains a `LoadModule` line, delete the line. The functionality of `mod_version` is still available in your migrated vFabric Web Server instances. For example, delete the following line: `LoadModule version_module <path>`
mod_ssl	In ERS instances, you conditionally configure SSL in the `httpdsd.conf` file by wrapping the `LoadModule mod_ssl` directive within an `<IfDefine SSL>` directive. When you start the server with the `startssl` command, SSL is enabled. In vFabric Web Server, you configure SSL in the `httpd.conf` file by adding the `LoadModule ssl_module <path>` line without a conditional `<IfDefine SSL>` directive. When you start the server instance using the `start` command, the server starts using SSL. See Startup Script Changes. vFabric Web Server now uses Open SSL 1.01 and supports Transport Layer Security (TLS) 1.1 and 1.2.

Startup Script Changes

The name of the script that you use to start vFabric Web Server instances has changed, and has some new options. In addition, configuration of the multi-process module (MPM) has changed. The startup scripts are located in the `bin` subdirectory of a server instance.

Table 5.3. Startup Script Changes

Item	Enterprise Ready Server	vFabric Web Server 5.3
Startup (control) script name	Linux: `<web_server_instance>/bin/apache_startup.sh`	Linux: `<web_server_instance>/bin/httpdctl`
	Windows: `<web_server_instance>\bin\apache_startup.bat`	Windows: `<web_server_instance>\bin\httpdctl.ps1`
Startup (control) script parameters	The following commands are available with the **apache_startup** script: • **start** • **startssl** • **stop** • **restart**	The following commands are available with the **httpdctl** script: • **start** • **stop** • **restart** • **gracefulstop**

Item	Enterprise Ready Server	vFabric Web Server 5.3
	• **graceful** • **configtest** • **configtestssl**	• **graceful** • **status** • **install** • **uninstall** • **configtest** See [httpdctl Reference]().
Startup script command with SSL enabled	To use SSL in an ERS instance you configure and load an SSL module in the `httpsd.conf` file, and you use the `startssl` option when you run the `apache_startup` script.	To use SSL in a vFabric Web Server instance, you configure a SSL module in the `httpd.conf` file and use the regular **start** option when you run the `httpdctl` script.
Multi-process module (MPM) configuration	You configure MPM in the `startup.properties` file.	You configure MPM when you create a vFabric Web Server instance. Use the following option of the **newserver** command: `--mpm=worker|prefork|event` (The default value is `worker`.) See [newserver Prompts and Command Reference](). If you need to modify MPM after creating a server, change the value of the `default_mpm` variable to `worker`, `prefork`, or `event` by changing the following line in the `httpdctl` script: `default_mpm="worker"`

Migration Procedure

To migrate an Enterprise Ready Server installation to a vFabric Web Server installation:

1. Install vFabric Web Server on each host where you will run a vFabric Web Server instance. See [Installing vFabric Web Server]().

2. Create a new vFabric Web server instance. See [Creating and Using vFabric Web Server Instances]().

   > If you specified mpm options in the Enterprise Ready Server `startup.properties` file, you must specify the `--mpm=worker|prefork|event` option when creating the new Web server instance. See [Startup Scripts]() and [newserver Prompts and Command Reference]().

3. Make a backup copy of the default configuration file from the new vFabric Web Server instance. The file is in the following location:

   `<Web Server instance>/conf/httpd.conf .`

4. Copy the following file from the Enterprise Ready Server instance directory:

   `servers/<Web server instance>/conf/httpsd.conf`

   into the `conf` directory of your new vFabric Web Server instance and rename the file to `httpd.conf`.

5. Open the newly-copied `httpd.conf` configuration file in a text editor and modify all of the paths to point to the new Web server instance. Make sure to change the following paths:

   • `ServerRoot` property.

- Apache modules—modify the appropriate `LoadModule` directives.

- `DocumentRoot` property.

- `<Directory>` statements.

- `Include` directives.

- Paths to any scripts referenced in the configuration.

- Paths to any log files referenced in the configuration.

6. If your vFabric Web server instance uses any of the Apache modules that have changed, change the `LoadModule` directives to point to the new modules. Review the changes listed in [Apache Modules Changes](#) for any changes that apply to your deployment.

7. Move static content from the previous Document Root directory to the new Document Root (or change `DocumentRoot` to point to the correct location.)

8. Make sure that the user and group name used to run vFabric Web Server instances exist on the host system. Use the host operating system commands to create the required user and group as indicated in the following table.

    The default user and group names used to run the httpd server have changed.

Table 5.4. User and group name changes

Default Name	Enterprise Ready Server	vFabric Web Server
Default group name	`nobody`	`vfhttpd`
Default user name	`nobody`	`vfhttpd`

9. Review the configuration file for any modules that you are not using in your deployment, and remove any `LoadModule` directives for these modules.

10. If you are using SSL, move the certificates and key file to your new vFabric Web Server instance directory. Make sure the permissions allow access to these files. (The `root` user should have read-only access.)

11. If a previous instance of Enterprise Ready Server was running on the host computer, make sure you uninstall the obsolete service and install the new vFabric Web Server instance as a service. See [Windows: Start and Stop vFabric Web Server Instances](#) or [Installing vFabric Web Server Instances as Unix Services](#).

12. Start the vFabric Web Server instance, using the **httpdctl** script. See [Startup Script Changes](#).

13. Test the server using a Web browser.

6. Creating and Using vFabric Web Server Instances

To start using vFabric Web Server, you explicitly create a new instance after you install it. An instance is not created for you by default.

Subtopics

Description of vFabric Web Server Instances

Create vFabric Web Server Instances

newserver Prompts and Command Reference

Unix: Start and Stop vFabric Web Server Instances

Windows: Start and Stop vFabric Web Server Instances

httpdctl Reference

Serve a Sample HTML File from Your vFabric Web Server Instance

Description of vFabric Web Server Instances

A vFabric Web Server *instance* is a complete, discrete HTTP server configuration.

You can create multiple instances that you can run simultaneously on the same computer if you do not use the same ports in two different instances. For example, the default HTTP listening port on Unix is 80, and only one instance on any computer is allowed to communicate on port 80 at any one time. So if you wanted to have two vFabric Web Server instances running at the same time on the same Unix computer, you configure one instance to use a port other than 80.

After you create an instance, its corresponding directory contains subdirectories that in turn contain all the data required to run a given vFabric Web Server instance. This data includes configuration information and all other data that is associated with that instance's configuration. For example, assume you installed vFabric Web Server in `/opt/vmware/vfabric-web-server` and create an instance called `myserver`:

```
prompt$ cd /opt/vmware/vfabric-web-server/myserver
prompt$ ls
bin  cgi-bin  conf  ftpdocs  htdocs  logs  proxy ssl  var
```

The `conf` directory contains the vFabric Web Server configuration files, such as `httpd.conf`. The `bin` directory contains the startup script used to start and stop the `myserver` instance (`httpdctl`. Each of these directories is specific to the `myserver` instance. Each instance you create will have a similar set of directories.

Create vFabric Web Server Instances

You create a new vFabric Web Server instance with the `newserver` command. The command creates a new directory that contains the instance-specific configuration files.

The `newserver` command format depends on your operating system:

• `newserver` : Perl script for Unix operating systems.

• `newserver.ps1`: Powershell script for Windows operating systems.

The command-line options for the two flavors are exactly the same. Where appropriate, the following procedure calls out the different usage depending on whether you are on Unix or Windows.

Prerequisites

- Complete the appropriate procedure in <u>Installing vFabric Web Server.</u>

- As of version 5.2 of vFabric Web Server, you must use Windows PowerShell to execute vFabric Web Server scripts on Windows computers. See the prerequisites section of <u>Windows: Install vFabric Web Server from a ZIP File</u> for information on installing PowerShell (if it is not already installed on your Windows computer) and enabling it for script processing.

Procedure

1. Log on to your computer as `root` (Unix) or the Administrator user (Windows) and start a terminal (Unix) or PowerShell window (Windows).

   To start a PowerShell window, go to Start > All Programs > Accessories > Windows PowerShell, then right-click on Windows PowerShell and select Run as Administrator.

2. Change to the directory in which you installed vFabric Web Server. For example, on Unix:

   ```
   prompt# cd /opt/vmware/vfabric-web-server
   ```

3. Run the `newserver` command to create the new instance; the command prompts you for information about the new server.

   The only required command option is `--server`, with which you specify the name of your vFabric Web Server instance. On Unix, use the Perl flavor; for example:

   ```
   prompt# ./newserver --server=myserver
   ```

   On Windows, use the PowerShell script:

   ```
   PS prompt> .\newserver.ps1 --server=myserver
   ```

   In both preceding examples, the way you specify the options is exactly the same. In the examples, the new instance is called `myserver` and its server directory is `/opt/vmware/vfabric-web-server/myserver`.

   For additional options, see <u>newserver Prompts and Command Reference</u>.

4. Enter values for the `newserver` prompts as the command requests information about your new instance. You can use the default values for many of the prompts, or even leave them blank.

   <u>newserver Prompts and Command Reference</u> provides additional information about the prompts.

What to do next

- Start the vFabric Web Server instance and verify that it is working correctly. See <u>Unix: Start and Stop vFabric Web Server Instances</u> and <u>Windows: Start and Stop vFabric Web Server Instances</u>.

- Complete the procedure in <u>Serve a Sample HTML File from your vFabric Web Server Instance.</u>

- Configure your instance as described in <u>Configuring vFabric Web Server</u>.

newserver Prompts and Command Reference

The `newserver` command has a number of options and prompts, as described in the two tables that follow.

The `newserver` command format depends on your operating system:

- `newserver` : Perl script for Unix.

- `newserver.ps1`: PowerShell script for Windows.

The command-line options for the two flavors are exactly the same.

Table 6.1. Options of the newserver Command

Option	Description	Required?
`--server=`*`servername`*	Name of the new vFabric Web Server instance. The value of this option becomes the name of the directory that contains the instance configuration files, and by default is the name of the host. The value of *`servername`* must be a valid DNS value and consist only of ASCII letters, digits or the dash character. Illegal characters include but are not limited to control characters below ASCII 32 as well as the following symbols: < > : " / \	? *. If you want to use an internationalized (i18n) name for the instance directory, you must enter the correct Punycode domain name provided by the registrar for the actual hostname. Do this by either specifying the `--set HostName=`*`punycode-hostname`* option at the command-line or entering the value interactively when the `newserver` command prompts you for the hostname.	Yes.

`--httpddir=`*`httpddir`*	Directory that contains the Apache HTTP binaries.
	
	The default value is `rootdir/`
	`httpd-`*`httpdver`*, such as `/opt/`
	`vmware/vfabric-web-server/`
	`httpd-2.2.23.0-64`.
		No.

`--httpdver=`*`httpdver`*	Version of the Apache HTTP binaries you want your instance to use. The default value is `2.2`, which is a symbolic link to the actual installed version of the binaries, such as `2.2.23.0-64`.	No.
`--mpm=`*`mpm`*	Specifies the type of multi-processing module (MPM) that the instance uses. Valid values are: • `worker`: Threaded MPM, ideal if you need a great deal of scalability. By using threads to serve requests, the instance can serve many requests with fewer system resources than a process-based server. • `prefork`: Non-threaded, pre-forking MPM if you require stability or compatibility with older software. • `event`: Less proven but higher-efficiency asynchronous connection-keepalive MPM. The `event` MPM offers little benefit for HTTPS connections, but is able to handle more simultaneous kept-alive and pending HTTP connections. The default value is `worker`.	No.
`--overlay`	Specifies that, if `serverdir` exists, you want to overwrite the existing files with new ones.	No. If you do not specify this option, and `serverdir` exists, the `newserver` command returns an error and suggests you specify a

Option	Description	Required?
		unique name and directory location for the new instance.
`--quiet`	Specifies that the `newserver` command should use default values for all prompts.	No. If you do not specify this option, `newserver` interactively prompts for all answers.
`--rootdir=rootdir`	Directory that contains the `httpd-2.2.version` directory, which in turn contains the Apache HTTP binaries. The default value is the current directory.	No.
`--serverdir=serverdir`	Directory in which you want the new instance directory to be created. The default value is `rootdir`.	No.
`--set token=value`	Specifies one or more tokens for which you would like to specify a custom value. The tokens are variables in the templates used to create new vFabric Web Server instances and correspond to a directive, or part of a directive, in the new instance's configuration. Each available token has a default value (listed below) that is automatically configured if you do not override it using the `--set` option. You can specify the following tokens; note that the token names are case-sensitive: • `User`: User that the vFabric Web Server processes run as. Corresponds to the `User` directive in `conf/httpd.conf`. Default value is `vfhttpd`. • `Group`: Group to which the user who runs the vFabric Web Server processes belongs. Corresponds to the `Group` directive in `conf/httpd.conf`. Default value is `vfhttpd`. • `Port`: HTTP port that the vFabric Web Server instance listens to. Corresponds to the port number in the `Listen` directive in `conf/httpd.conf`. Default value is `80`. • `SSLPort`: HTTPS port that the vFabric Web Server instance listens to for secure communications. Corresponds to the port number in the `Listen https` directive in `conf/extras/httpd-ssl.conf`. Default value is `443`. • `HostName`: Name of the host that the instance uses to identify itself. Corresponds to the hostname part of the `ServerName` directive in the `conf/httpd.conf` file. Default value is the value you specified for the required `--server` option. • `ServerAdmin`: Email address of the administration user who should get emails when there are problems with the instance. Corresponds to the `ServerAdmin` directive in the `conf/httpd.conf` file. Default value is `webmaster@HostName.`.	No.

Option	Description	Required?
	The following example shows how to specify that the new vFabric Web Server instance run as the `newhttpd` user in the `newhttpd` group: `prompt# ./newserver --server=myserver --set User=newhttpd --set Group=newhttpd`	
`--sourcedir=sourcedir`	Name of the directory that contains the template that `newserver` uses to create the new vFabric Web Server instance. The default value is `httpdir/_instance`.	No.

Table 6.2. newserver Prompts

Prompt	Description
Enable SSL and create a default key [y/n]?	Enabling SSL provides secure communication between client and server by allowing mutual authentication; the use of digital signatures for integrity; and encryption for privacy. If you answer yes, you are later asked for information that will be used to create a certificate.
Server hostname (e.g. www.example.com) [myserver]?	Name that the vFabric Web Server instance uses to identify itself. If your host does not have a registered DNS name, enter its IP address. The default value is the value you entered for the `--server` option.
Administrator email [webmaster@myserver]?	Email address to which vFabric Web Server instances send problems. This address appears on some instance-generated pages, such as error documents.
Port for http:// traffic [80]?	HTTP port to which the vFabric Web Server instance listens. Default value is `80` when running the `newserver` command as the `root` user on Unix, `8080` otherwise.
Port for https:// SSL traffic [443]?	HTTPS port to which the vFabric Web Server instance listens. Default value is `443` when running the `newserver` command as the `root` user on Unix, `8443` otherwise.
If you previously specified that you want to enable SSL...	The `newserver` command prompts you for information required to create the private key, such as the size of the SSL RSA key in bits and the PEM pass phrase you specify when you start the instance. You also are prompted to enter information for your certificate. The information is mostly about your Distinguished Name, or DN, that will be incorporated into your certificate request. As indicated, some fields have default values. You can also leave some fields blank by entering a '.' (period.) When `newserver` completes, it generates the following SSL files in the `ssl` subdirectory of the instance directory: • `instancename.key`: Unencrypted private key. The file has a permission code of 0600 for additional security. • `instancename.pem`: DES 3 encrypted private key. • `instancename.csr`: Certificate-signing request. Submit this file to the Certificate Authority. • `instancename.crt`: Self-signed certificate. Replace this certificate with a signed certificate by the CA. **Important** Be sure to record the passphrase to decrypt the `*.pem` file and back up the file. Never transmit the `.key` file or cause it to be readable by others.

Unix: Start and Stop vFabric Web Server Instances

You interactively start, stop, or restart a vFabric Web Server instance on Unix with the `httpdctl` shell script in the `bin` directory of the instance.

Warning: You always use the start script in the `bin` of the instance directory, such as `/opt/vmware/vfabric-web-server/myserver/bin`. Do not use the start script in the `httpd-2.2/bin` sub-directory of the main installation directory.

You can also install a vFabric Web Server instance as a Unix service so it automatically starts and stops when the operating system itself is started and stopped. If you install it as a service, you can also start and stop the service by using the `/etc/init.d/service-name` script. For more information, see [Installing vFabric Web Server Instances as Unix Services](#).

Prerequisites

- Complete the appropriate procedure in [Create vFabric Web Server Instances.](#)

Procedure

1. Log in to your Unix computer as the `root` user.

2. Start a terminal window and change to the `bin` sub-directory of your vFabric Web Server instance's root directory. For example, if you created an instance called `myserver` that lives in the installation directory `/opt/vmware/vfabric-web-server`:

    ```
    prompt# cd /opt/vmware/vfabric-web-server/myserver/bin
    ```

3. Start the instance using the `./httpdctl start` command:

    ```
    prompt# ./httpdctl start
    ```

    You should see a message as follows:

    ```
    Starting Apache:
    Server started OK
    ```

4. To test that the vFabric Web Server instance actually started, navigate to the `http://host:port` URL in your browser, where `host` refers to the host computer (you can use `localhost` if your browser is on the same computer) and `port` refers to the HTTP listen port number you provided when you created the instance. The default value is `80`

    For example, if you are using the default ports on your local computer, you can use this URL:

    ```
    http://localhost:80
    ```

    If the instance started successfully, you should see the Welcome page.

5. To get status about the instance:

    ```
    prompt# ./httpctl status
    ```

6. To stop the instance immediately, even if there are current connections in use:

    ```
    prompt# cd /opt/vmware/vfabric-web-server/myserver/bin
    prompt$ ./httpdctl stop
    ```

    To stop the instance gracefully:

    ```
    prompt$ ./httpdctl gracefulstop
    ```

See [httpdctl Reference](#) for the full list of available `httpdctl` commands.

What to do next

- Complete the procedure in <u>Serve a Sample HTML File from your vFabric Web Server Instance.</u>

- Configure your instance as described in <u>Configuring vFabric Web Server</u>.

Installing vFabric Web Server Instances as Unix Services

You can install a vFabric Web Server instance as a system service on Unix, Linux, so it automatically starts and stops when the operating system itself starts and stops. If you install it as a service, you start and stop the service by using the `/etc/init.d/`*`ServiceName`* script.

Procedure

1. Log in to your computer as the `root` user.

2. Start a terminal window and change to the `bin` sub-directory of your vFabric Web Server instance's root directory. For example, if you created an instance called `myserver` that lives in the installation directory `/opt/vmware/vfabric-web-server`:

   ```
   prompt# cd /opt/vmware/vfabric-web-server/myserver/bin
   ```

3. Install the instance as a service using the `./httpdctl install` command:

   ```
   prompt# ./httpdctl install
   ```

   You should see the following output:

   ```
   Installing vFabric httpd myserver
   as the unix service vFabric-httpd-myserver
   ```

   The display name for the service name is `vFabric httpd `*`InstanceName`*. The service name is the display name, with spaces replaced by dashes: `vFabric-httpd-`*`InstanceName`*.

4. After creating the service, you can control it (1) with the system-config-services GUI utility, (2) by running the `bin/httpdctl` shell script from the instance directory, or (3) by running the `/etc/init.d/`*`ServiceName`* script, passing it the same commands as `httpdlctl`. For example, to start the `myserver` instance:

   ```
   prompt# /etc/init.d/vFabric-httpd-myserver start
   ```

5. To uninstall the instance as a Unix service:

   ```
   prompt# ./httpdctl uninstall
   ```

Windows: Start and Stop vFabric Web Server Instances

You start, stop, or restart a vFabric Web Server instance on Windows by first installing it as Windows service using the `httpdctl.ps1` PowerShell script in the `bin` directory of the instance directory, and subsequently using the Windows Services console to start or stop it.

Warning: You always use the start script in the `bin` of the instance directory, such as `c:\opt\vmware\vfabric-web-server\myserver\bin`. Do not use the start script in the `httpd-2.2\bin` sub-directory of the main installation directory.

Prerequisites

- Complete the appropriate procedure in <u>Create vFabric Web Server Instances.</u>

- As of version 5.2 of vFabric Web Server, you are required to use Windows PowerShell to execute vFabric Web Server scripts on Windows computers. See the prerequisites section of <u>Windows: Install vFabric Web Server from a ZIP File</u> for information on installing PowerShell (if it is not already installed on your Windows computer) and enabling it for script processing.

Procedure

1. Log in to your Windows computer as the `Administrator` user and start a PowerShell window by going to Start > All Programs > Accessories > Windows PowerShell, then right-clicking on Windows PowerShell and selecting Run as Administrator.

2. Change to the `bin` subdirectory of the root directory for the vFabric Web Server instance.

    For example, if you created an instance called `myserver` that lives in the installation directory `c:\opt\vmware` `\vfabric-web-server`:

    ```
    PS prompt> cd c:\opt\vmware\vfabric-web-server\myserver\bin
    ```

3. Install the instance as a Windows service by running the `httpdctl.ps1 install` command:

    ```
    PS prompt> .\httpdctl.ps1 install
    ```

    The display name for the service name is vFabric httpd *InstanceName*. The service name is the display name, with spaces removed: vFabric-httpd *InstanceName*.

    After installing the service, you can control and further configure the service in several ways: (1) `httpdctl` command options, (2) the Windows Services console, or (3) the `sc` command

4. To test that the vFabric Web Server instance actually started, navigate to the `http://host:port` URL in your browser, where *host* is the host computer (you can use `localhost` if your browser is on the same computer), and *port* is the HTTP port number you provided when you created the instance. The default value on Windows is `80`.

    For example, if you are using the default ports on your local computer, use this URL:

    ```
    http://localhost:80
    ```

    If the vFabric Web Server instance started successfully, you should see the Welcome page.

5. To get status about a running vFabric Web Server instance, execute the following command:

    ```
    PS prompt> .\httpdctl.ps1 status
    ```

6. To uninstall the vFabric Web Server instance as a Windows service, use the following command:

    ```
    PS prompt> cd c:\opt\vmware\vfabric-web-server\myserver\bin
    PS prompt> .\httpdctl.ps1 uninstall
    ```

See [httpdctl Reference](#) for the full list of available `httpdctl` commands.

What to do next

*   Complete the procedure in [Serve a Sample HTML File from your vFabric Web Server Instance.](#)

*   Configure your instance as described in [Configuring vFabric Web Server](#).

httpdctl Reference

Use the `httpdctl` script to control a vFabric Web Server instance: start 'n' stop it, install it as a service, and so on. The script is located in the `bin` directory of the instance directory, such as `/opt/vmware/vfabric-web-server/myserver/bin`.

The `httpdctl` script format depends on your operating system:

*   `httpdctl`: Perl script for Unix. See [Unix: Start and Stop vFabric Web Server Instances](#) for usage examples.

- `httpdctl.ps1`: PowerShell script for Windows. See <u>Windows: Start and Stop vFabric Web Server Instances</u> for usage examples.

Script commands are the same for both. Commands are described in the following table.

Table 6.3. httpdctl Script Commands

Command	Description
`start`	Starts the vFabric Web Server instance. If the instance is already running, the command returns an error.
`stop`	Forcibly stops the vFabric Web Server instance. All currently opened connections are aborted.
`gracefulstop`	Gracefully stops the vFabric Web Server instance, which means that the script waits until all currently open connections are closed rather than aborting them forcibly.
`restart`	Restarts the instance. If the instance was not originally running, the script starts it. If the instance was not originally running, the script starts it. The script also runs a `configtest` before starting the instance.
`graceful`	Gracefully restarts the instance. If the instance is not running, it is started. This command differs from a normal restart in that currently open connections are not aborted. A side effect is that old log files will not be closed immediately. This means that if you use this command in a log rotation script, a substantial delay may be necessary to ensure that the old log files are closed before processing them. This command runs a `configtest` before starting the instance.
`status`	Displays basic status information about the instance, such as whether it is running and its process id (PID) if so.
`install`	Installs the instance as a service on Windows, Unix, and Linux. After installing the service on Windows, you use the Windows Services console or the `sc` command to start, stop, and restart the vFabric Web Server instance, and configure whether the service starts automatically when Windows starts, and so on. For more information about the the `sc` command, see `http://technet.microsoft.com/en-us/library/bb490995.aspx` On Unix, the instance is installed as a script file in the `/etc/init.d` directory with name `vFabric-httpd-`*`instance-name`*. The service automatically starts and stops when Unix is started or stopped.
`uninstall`	Uninstalls the instance as Windows, Unix, or Linux service. On Windows, the instance is removed from the Service registry. On Unix, the command deletes the `/etc/init.d/vFabric-httpd-`*`instance-name`* script file.
`configtest`	Runs a syntax test against the configuration files, such as `conf/httpd.conf`. The script parses the configuration files and either reports Syntax OK or outputs detailed information about the particular syntax error.

Serve a Sample HTML File from Your vFabric Web Server Instance

After you install vFabric Web Server and create an instance, you can use it to host your entire Web site. This section does not describe the entire process; rather, it simply shows how to serve an HTML file from the default document root of your instance.

Prerequisites

- Create and start a vFabric Web Server instance. See <u>Create vFabric Web Server Instances</u>.

- Create or download one or more sample HTML pages that you want to serve from the instance.

Procedure

1. Open the configuration file for your vFabric Web Server instance and make note of the value of the `DocumentRoot` directive, which is the directory out of which the instance serves your documents. By default, vFabric Web Server takes all requests from this directory.

    The configuration file is called `httpd.conf` and is located in the `INSTANCE-DIR`/conf, such as `/opt/vmware/ vfabric-web-server/myserver/conf/httpd.conf`. The `DocumentRoot` directive looks like the following:

    ```
    DocumentRoot "/opt/vmware/vfabric-web-server/myserver/htdocs"
    ```

2. Copy your sample HTML pages to the document root.

    For example, if you have a `hello.html` page in the `/home/samples` directory that you want to serve up:

    ```
    prompt# cp /home/samples/hello.html /opt/vmware/vfabric-web-server/myserver/htdocs
    ```

3. Invoke the HTML page in your browser using the vFabric Web Server instance.

    For example, if your browser is running on the same computer as vFabric Web Server and the instance is listening at the default port `80`, the URL is as follows:

    ```
    http://localhost/hello.html
    ```

    Because the instance is using the default port of 80, you do not have to explicitly specify it in the URL. If you set a different port, such as 8000, then the URL would be:

    ```
    http://localhost:8000/hello.html
    ```

    You should see your `hello.html` page in your browser.

4. You can create a directory hierarchy under the document root to better organize your HTML pages.

    For example:

    ```
    prompt# cd /opt/vmware/vfabric-web-server/myserver/htdocs
    prompt# mkdir fun
    prompt# cp /home/samples/hello.html fun
    ```

    The URL to invoke the HTML page would now be:

    ```
    http://localhost/fun/hello.html
    ```

What to do next

-   Configure vFabric Web Server instances to take advantage features such as load balancing, virtual hosts, and SSL. See Configuring vFabric Web Server Instances.

7. Configuring vFabric Web Server Instances

The default configuration of a newly created vFabric Web Server instance is fairly simple. Although the configuration is likely adequate for your needs, sometimes you might need to further configure the instance to enable one of its many useful features, such as load-balancing between two or more tc Runtime instances. This chapter provides some information to get you started.

For complete documentation on how to configure vFabric Web Server instances, see Apache HTTP Server Version 2.2 Documentation. Because vFabric Web Server is based on Apache HTTP server, the general configuration documentation on the Apache Web site applies to vFabric Web Server as well.

Subtopics

Using the Sample Configuration Files to Enable Features and Modify Configuration

Configure Load Balancing Between Two or More tc Server Instances

Configure SSL Between vFabric Web Server and vFabric tc Server

Configure BMX for Monitoring vFabric Web Server Instances

Metrics

Using Sample Configuration Files to Enable Features and Modify Configuration

All vFabric Web Server instances include sample configuration files that you can use to enable extra features in the server instance or to modify its default configuration. These files are located in the `INSTANCE-DIR`/conf/extra directory, where `INSTANCE-DIR` refers to the instance directory, such as /opt/vmware/vfabric-web-server/myserver.

For example, the httpd-info.conf sample configuration file shows how you can get information about the requests being processed by the vFabric Web Server instance as well as information about the configuration of the instance. The httpd-ssl.conf file shows how to provide SSL support. It contains the configuration directives to instruct the instance how to serve pages over an HTTPS connection.

For your convenience, the main vFabric Web Server configuration file for a particular instance (`INSTANCE-DIR`/conf/httpd.conf) already includes commented-out lines for including each sample configuration file. For example, the line to include the httpd-info.conf configuration file is as follows:

```
#Include conf/extra/httpd-info.conf
```

To include the configuration file, simply uncomment the Include directive:

```
Include conf/extra/httpd-info.conf
```

You do not have to use Include in this way; you can simply copy and paste the information in a sample configuration file into the main configuration file.

The sample configuration files are full of comments on how exactly to enable the feature they configure. Be sure to read these comments before you proceed further.

What to do next

- Restart the vFabric Web Server instance for the configuration changes to take effect. For example, on Unix:

```
prompt# cd /opt/vmware/vfabric-web-server/myserver
prompt# bin/httpdctl restart
```

Configure Load Balancing Between Two or More tc Runtime Instances

You can configure a vFabric Web Server instance to perform simple load balancing between two or more tc Server instances.

tc Runtime is the runtime component of vFabric tc Server.

In the procedure that follows, you configure a vFabric Web Server instance to run *in front* of the tc Runtime instances; this vFabric Web Server instance receives all requests from users, and then passes them back to the tc Runtime instances using a specified load-balancing algorithm. Responses from the tc Runtime instances are then routed back through this same vFabric Web Server instance. For this reason, the vFabric Web Server instance acts like a proxy (both reverse and forward) so that the users never know the URLs of the backend tc Runtime instance that are actually doing all the work. Additionally, the vFabric Web Server instance ensures that the load on each tc Runtime instance is balanced. You can specify that each tc Runtime instance take on an equal work load, or you can specify that one instance work twice as hard as the others.

In the procedure, the following scenario pertains. These assumptions are *not* requirements; your environment might be very different. The assumptions are listed only to make the procedure easier to understand.

- Two tc Runtime instances are running at the following two hosts and port numbers:

  - `http://192.168.0.203:8081`

  - `http://192.168.0.203:8082`

  The two tc Runtime instances are running on the same computer, are part of the same installation and their respective `CATALINA_BASE` variables are as follows:

  - `/var/opt/vmware/vfabric-tc-server-standard/instanceOne`

  - `/var/opt/vmware/vfabric-tc-server-standard/instanceTwo`

- Each tc Runtime instance is configured exactly the same (other than the value of the various ports).

- You have deployed the same application to both tc Runtime instances and the URL context is the same in both instances: `/my-app`.

- You want all users of the application to first go through the front-end vFabric Web Server instance, and any evidence of the backend tc Runtime instances upon which the application is actually deployed should be hidden from the user.

- vFabric Web Server is installed on a different computer than vFabric tc Server. The name of the particular vFabric Web Server instance is `lb-server` and its home directory is `/opt/vmware/vfabric-web-server/lb-server`.

- You want to configure *sticky sessions*, which means that the vFabric Web Server instance always routes the requests for a particular session to the same tc Runtime instance that serviced the first request for that session.

- You want to use the HTTP protocol for all communication between the vFabric Web Server and the tc Runtime instances.

The load balancing described in this procedure is very simple, although you have many options available to further customize it. At appropriate locations in the procedure, links to the Apache HTTP Server documentation are provided for additional configuration options not covered by this specific scenario. Adapt the procedure for your particular environment.

As part of the procedure, you update the configuration files of both the vFabric Web Server instance and the two tc Runtime instances.

Prerequisites

- [Install vFabric Web Server](#) on your platform and [create a new instance](#).

* Install vFabric tc Server on the same or different computer as vFabric Web Server, and create two more instances. Make note of the host and port numbers of the two instances. See the vFabric tc Server documentation for details.

* Deploy the same application to the two tc Runtime instances.

Procedure

To configure load balancing for the scenario described in the introduction to this section, follow these steps:

1. On the computer on which vFabric Web Server is installed, stop the instance, if it is currently running. Following the example and assumptions:

```
prompt# cd /opt/vmware/vfabric-web-server/lb-server
prompt# bin/httpdctl stop
```

2. Open the `httpd.conf` configuration file of the vFabric Web Server instance and ensure that the three required `LoadModule` directives (`proxy_balancer_module`, `mod_proxy`, and `mod_proxy_http`, are present and enabled (in other words, are *not* commented out):

```
LoadModule proxy_balancer_module "VFWS-INSTALL/httpd-2.2/modules/mod_proxy_balancer.so"
LoadModule proxy_module          "VFWS-INSTALL/httpd-2.2/modules/mod_proxy.so"
LoadModule proxy_http_module     "VFWS-INSTALL/httpd-2.2/modules/mod_proxy_http.so"
```

where *VFWS-INSTALL* refers to the directory in which you installed vFabric Web Server. If they are not in the file, add them in the same location as the other `LoadModule` directives.

Following our example, the directive configurations would be:

```
LoadModule proxy_balancer_module  "/opt/vmware/vfabric-web-server/httpd-2.2/modules/mod_proxy_balancer.so"
LoadModule proxy_module           "/opt/vmware/vfabric-web-server/httpd-2.2/modules/mod_proxy.so"
LoadModule proxy_http_module      "/opt/vmware/vfabric-web-server/httpd-2.2/modules/mod_proxy_http.so"
```

The vFabric Web Server configuration file is located in the `conf` directory of your vFabric Web Server instance (`/opt/vmware/vfabric-web-server/lb-server/conf` in our example).

3. In the same `httpd.conf` file, add the proxy configuration.

Use the `<Proxy>` element to specify the list of tc Runtime instances and the method of load balancing you want to use. Then use the `ProxyPass` and `ProxyPassReverse` directives to specify the URLs that will use this proxy and load-balancing (both for requests and responses.) For example:

```
<Proxy balancer://my-balancer>
     BalancerMember http://192.168.0.203:8081 route=instanceOne loadfactor=1
     BalancerMember http://192.168.0.203:8082 route=instanceTwo loadfactor=1
     ProxySet lbmethod=byrequests
</Proxy>

ProxyPass /my-app balancer://my-balancer/my-app
ProxyPassReverse /my-app http://192.168.0.203:8081/my-app
ProxyPassReverse /my-app http://192.168.0.203:8082/my-app
```

In the preceding example:

* The `balancer` parameter of the `<Proxy>` element specifies a unique identifier for this load balancer configuration.

* Each tc Runtime instance that is serviced by this load balancer must have its own `BalancerMember`; the first parameter of this directive specifies the full IP address (including port number) of the tc Runtime instance.

* The `route` parameter contains session ID information. You later use the value of this parameter in the tc Runtime configuration file to configure sticky sessions; for now, just ensure that the values are unique for each `BalancerMember`.

- The `loadfactor` parameter specifies how much load a particular member carries. If you want each member to carry the same load, set the numbers equal to each other (as in the example above). If, however, you want one member to work three times harder than the other, set the load factors to 3 and 1.

- Use the `lbmethod` parameter of the `ProxySet` directive to specify the load balancing algorithm. The possible values are as follows:

  - `byrequests`: performs weighted request counting. This is the default value.

  - `bytraffic`: performs weighted traffic byte count balancing.

  - `bybusyness`: performs pending request balancing.

- Use the `ProxyPass` and `ProxyPassReverse` to specify the context URLs of the application that will be routed to the tc Runtime instances that you have configured in the load balancing scheme. `ProxyPass` specifies that when the vFabric Web Server instance receives a request at the `/my-app` URL, it routes the request to the load balancer that will in turn route it to the tc Runtime instance. `ProxyPassReverse` does the reverse: when the tc Runtime instance sends a response to a user who is using `/my-app`, the response appears to come from the vFabric Web Server instance, and not the tc Runtime instance. Thus the details of the tc Runtime instance are *hidden* from the user.

4. **Optional**. If you want to enable the balancer manager Web application to watch the load balancing activity and control the behavior, add the following to the `httpd.conf` configuration file of your vFabric Web Server instance:

```
<Location /balancer-manager>
    SetHandler balancer-manager
    Order Deny,Allow
    Deny from all
    # BE VERY RESTRICTIVE with YOUR ALLOW STATEMENT
    Allow from 127.0.0.1
</Location>
```

5. **Optional**. If you want to enable sticky sessions, follow these steps:

   a. In the `httpd.conf` file of the vFabric Web Server instance, update the `ProxySet` directive of the `<Proxy>` element you configured in a preceding step by adding the `stickysession=JSESSIONID|jsessionid` parameter. This parameter configures the cookie/path that will be used for stickiness. For example (update shown in bold):

```
<Proxy balancer://my-balancer>
     BalancerMember http://192.168.0.203:8081 route=instanceOne loadfactor=1
     BalancerMember http://192.168.0.203:8082 route=instanceTwo loadfactor=1
     ProxySet lbmethod=byrequests stickysession=JSESSIONID|jsessionid
</Proxy>
```

   b. Go to the computer on which vFabric tc Server is running and update the `server.xml` configuration file of both tc Runtime instances by adding the `jvmRoute=value` attribute to the Catalina `<Engine>` element. Set the value of this attribute equal to the value you specified (in a preceding step) for the `route` parameter of the `BalancerMember` directive in the vFabric Web Server `httpd.conf` file that describes the tc Runtime instance.

   Following our example, the updated `<Engine>` entry for the `instanceOne` tc Runtime instance (that uses port `8081`) would be as follows (new attribute in bold):

```
  <Engine name="Catalina" defaultHost="localhost" jvmRoute="instanceOne">
```

If you configure sticky sessions, VMware recommends that you also configure session replication for the tc Runtime instances. For details, see the section titled *Enabling Clustering for High Availability* in the vFabric tc Server documentation.

6. Start the vFabric Web Server instance. Following our example:

```
prompt# cd /opt/vmware/vfabric-web-server/lb-server
prompt# bin/httpdctl start
```

7. Start (or restart) the two tc Runtime instances for the configuration changes to take effect. Following our example:

```
prompt$ cd /var/opt/vmware/vfabric-tc-server-standard
prompt$ ./tcruntime-ctl.sh instanceOne restart
prompt$ ./tcruntime-ctl.sh instanceTwo restart
```

You have now configured load balancing for the two tc Runtime instance using the front-end vFabric Web Server.

What to do next

*   For full reference documentation on the directives described in step 3, along with additional parameters you can use, see Apache Module mod_proxy on the Apache Software Foundation Web site.

*   Ensure that you can access your application through the vFabric Web Server instance, which in turn routes the request to one of the tc Runtime instances. Do this by invoking your application in a browser, but specify the vFabric Web Server instance rather than the tc Runtime instance. For example, if the URL to access the vFabric Web Server is `http://www.myhost.com`, invoke the `/my-app` application using the following URL in your browser:

```
http://www.myhost.com/my-app
```

    If you see your application, then you have correctly set up the vFabric Web Server instance to route requests to the `/my-app` application to one of the two tc Runtime instances. The vFabric Web Server instance will also balance the load between the two instances.

*   If you enabled the balancer manager Web application, use it to watch and control load-balancing activity. Access the balancer manager application by navigating to the following URL in your browser:

```
http://localhost:port/balancer-manager
```

    where `port` is the port number of the vFabric Web Server instance (`80` by default.) For security, the balancer manager configuration allows access only to users who navigate to the application using a browser installed on the same computer on which the vFabric Web Server instance is actually running.

Configure SSL Between vFabric Web Server and vFabric tc Server

For additional security, it is often desirable to configure SSL between a vFabric Web Server instance and one or more tc Runtime instances, although it's not required. (*tc Runtime* is the runtime component of vFabric tc Server.)

SSL certificates are frequently used to confirm the identity of a server before consuming its services and to secure communications with the server. Typically, if you use a vFabric Web server instance to load balance requests to one or more tc Runtime instances, the SSL encryption and certificate authentication is terminated at the Web Server instance. Communication between the Web Server and tc Runtime instances is then trusted and in clear text.

However, there are organizational security policies and B2B scenarios that might mandate secure communication between the vFabric Web Server and tc Runtime instances. Furthermore, it might be important to restrict access to the tc Runtime instances to known instances of vFabric Web Server.

This section provides details for configuring SSL communication and client certificate authentication between vFabric Web Server and tc Server. The high-level steps are as follows, with detailed information about each step in its own sub-section:

1. Configure tc Runtime Instances to Use SSL

2. Configure the vFabric Web Server instance to Use SSL

3. Update the Web Server Configuration for HTTPS Connections to tc Runtime Instances

4.   [Restrict Communication With tc Runtime Instances to Known Clients](#)

5.   [Configure vFabric Web Server to Authenticate Using a Specific Client Certificate](#)

Important. It is assumed that you have already installed vFabric Web Server and vFabric tc Server, created instances, and set up unsecured load balancing between them. If you have not already done this, see [Configure Load Balancing Between vFabric Web Server and vFabric tc Server](#).

Configure tc Runtime Instances to Use SSL

VMware recommends that you configure a tc Runtime instance to use SSL by specifying the `bio-ssl` template when you create or modify an instance; this template adds the correct configuration to the `conf/server.xml` file and automatically generates a keystore based on your inputs. You specify the `bio-ssl` template when you create a new tc Runtime instance using the `tcruntime-instance` command. Additionally, as of version 2.8 of vFabric tc Server, you can also apply the template to an existing instance.

The following example shows how to create a new tc Runtime instance that uses the `bio-ssl` template:

```
prompt$ ./tcruntime-instance.sh create instanceOne -t bio-ssl -i /var/opt/vmware/vfabric-tc-server-standard
```

In the preceding example, the tc Runtime instance will be located in the `/var/opt/vmware/vfabric-tc-server-standard` directory and will use default values when creating the keystore. If you want to customize the keystore, use the `--interactive` option and the command will prompt you for specific information:

```
prompt$ ./tcruntime-instance.sh create instanceOne -t bio-ssl -i /var/opt/vmware/vfabric-tc-server-standard --interactive
```

The following example shows how to apply the `bio-ssl` template to an existing tc Runtime instance called `instanceTwo`:

```
prompt$ ./tcruntime-instance.sh apply-template instanceOne -t bio-ssl -i /var/opt/vmware/vfabric-tc-server-standard
```

Note: The `apply-template` option of `tcruntime-instance` is available as of version 2.8 of vFabric tc Server.

To invoke an application deployed to the tc Runtime instance using HTTPS, specify the HTTPS port. The default HTTPS port is `8443`, although you might have configured a different port for your particular instance. For example:

```
https://host:8443/my-app
```

See *Create and Modify a tc Runtime Instance* in the *Getting Started with vFabric tc Server* guide in this Documentation Center for details.

If you chose not to use the `bio-ssl` template, you can create your own keystore using the [keytool](#) command, as shown in the following example:

```
prompt$ keytool -genkey -alias tomcat -keyalg RSA -keystore CATALINA_BASE/conf/tomcat.keystore
```

In the preceding example, `CATALINA_BASE` refers to the instance directory, such as `/var/opt/vmware/vfabric-tc-server-standard/instanceOne`.

Update the appropriate `<Connector />` element in the instance's `conf/server.xml` file by adding the `keyAlias`, `keystoreFile`, and `keystorePass` attributes, setting the values to those you specified when you created the keystore using `keytool` as shown above. For example:

```
<Connector SSLEnabled="true"
          acceptCount="100"
          connectionTimeout="20000"
          executor="tomcatThreadPool"
          keyAlias="tomcat"
          keystoreFile="${catalina.base}/conf/tomcat.keystore"
          keystorePass="changeme"
```

```
            maxKeepAliveRequests="15"
            port="8443"
            protocol="org.apache.coyote.http11.Http11Protocol"
            redirectPort="8443"
            scheme="https"
            secure="true"/>
```

Configure the vFabric Web Server Instance to Use SSL

The easiest way to do configure SSL for a Web Server instance is to use the `newserver` interactive command to create a new instance and specify that you want to enable SSL for the instance. The command performs configuration tasks and creates a private key. See Create vFabric Web Server Instances.

If you want to enable SSL for an *existing* Web Server instance, you can uncomment the `Include conf/extras/httpd-ssl.conf` directive in the main `conf/httpd.conf` file and then follow directions in the `conf/extra/httpd-ssl.conf` file. See Using the Sample Configuration Files to Enable Features and Modify Configuration for details.

Restrict Communication With tc Runtime Instances to Known Clients

This section describes how to specify that the tc Runtime instances require a valid certificate from a client before it accepts a connection.

Procedure

1. Log in to the computer on which you installed tc Server and open a terminal window.

2. Create a certificate authority file. The following examples show how to create the CA file `ca.crt` by using the `openssl` command:

    ```
    prompt$ openssl genrsa -out ca.key 1024
    prompt$ openssl req -new -key ca.key -out ca.csr
    prompt$ openssl x509 -req -days 365 -in ca.csr -signkey ca.key -out ca.crt
    ```

3. Generate a JKS formatted certificate authority file. The following example shows how to use the keytool command to create the file `cacerts.jks` from the CA file `ca.crt` you created in the preceding step:

    ```
    prompt$ keytool -importcert -keystore cacerts.jks -storepass changeme -alias my_ca -file ca.crt
    ```

4. Copy the `cacerts.jks` file to the *CATALINA_BASE*`/conf` directory of each tc Runtime instance.

5. Update the appropriate `<Connector />` element in each instance's `conf/server.xml` file by adding the `clientAuth` and `truststoreFile` attributes, setting their values as shown in the example:

    ```
    <Connector SSLEnabled="true"
            clientAuth="true"
            truststoreFile="${catalina.base}/conf/cacerts.jks"
            acceptCount="100"
            connectionTimeout="20000"
            executor="tomcatThreadPool"
            keyAlias="tomcat"
            keystoreFile="${catalina.base}/conf/tomcat.keystore"
            keystorePass="changeme"
            maxKeepAliveRequests="15"
            port="8443"
            protocol="org.apache.coyote.http11.Http11Protocol"
            redirectPort="8443"
            scheme="https"
            secure="true"/>
    ```

6. Restart each tc Runtime instance for the changes to take effect:

    ```
    prompt$ cd /var/opt/vmware/vfabric-tc-server-standara/instanceOne/bin
    ```

```
prompt$ ./tcruntime-ctl.sh restart
```

7. Test that you have configured each tc Runtime instance correctly by navigating to an application deployed on the instance using your browser.

    The tc Runtime instance should deny you access because your browser does not have the required client certificate configured.

Update the Web Server Configuration for HTTPS Connections to tc Runtime Instances

As specified earlier, it is assumed that you have already configured your vFabric Web Server instance for unsecured load balancing between two or more tc Runtime instances. If you have not already done this, see Configure Load Balancing Between vFabric Web Server and vFabric tc Server.

Update the Web Server configuration to communicate with the tc Runtime instances securely by editing the conf/http.conf file in the Web Server instance directory (such as /opt/vmware/vfabric-web-server/lb-server) and changing the tc Runtime URLs so they use HTTPS and specify the HTTPS port. Following the example from the load balancing section, if you specified that the HTTPS ports for instanceOne and instanceTwo were 8443 and 8553, respectively, the updated file would look like this:

```
<Proxy balancer://my-balancer>
     BalancerMember https://192.168.0.203:8443 route=instanceOne loadfactor=1
     BalancerMember https://192.168.0.203:8553 route=instanceTwo loadfactor=1
     ProxySet lbmethod=byrequests
</Proxy>

ProxyPass /my-app balancer://my-balancer/my-app
ProxyPassReverse /my-app https://192.168.0.203:8443/my-app
ProxyPassReverse /my-app https://192.168.0.203:8553/my-app
```

Configure vFabric Web Server to Authenticate Itself Using a Specific Client Certificate

This section describes how to configure the Web Server instance to authenticate itself using the client certificate you created in a previous step and configured for each tc Runtime instance.

1. On the computer on which you installed vFabric Web Server, create a client certificate and key. Use the *same* certificate authority file (called ca.crt in the example) that you created in a previous step.

    The following example shows how to do this using the openssl command:

    ```
    prompt$ openssl genrsa -out client.key 1024
    prompt$ openssl req -new -key client.key -out client.csr -config your-openssl.cnf-file
    prompt$ openssl x509 -req -days 365 -CA ca.crt -CAkey ca.key -CAcreateserial -in client.csr -out client.crt
    ```

    In the preceding example, the newly generated client key is called client.key and the client certificate file is called client.crt. Replace your-openssl.cnf-file with the full path name of the openssl.cnf file on your computer, such as /etc/pki/tls/openssl.cnf.

2. Concatenate the generated client key and client certificate files into a single file. In the following example, the new file is called client.crtkey:

    ```
    prompt$ cat client.crt client.key > client.crtkey
    ```

3. Copy the generated client.crtkey file to the ssl directory of the Web Server instance directory. For example:

    ```
    prompt$ cp client.crtkey /opt/vmware/vfabric-web-server/lb-server/ssl
    ```

4. Configure the mod_ssl module of the Web Server instance to use SSL for the proxy engine and to use the generated client certificate and key file by adding the following directives to the file that contains the SSL configuration, such as conf/extra/httpd-ssl.conf:

```
SSLProxyMachineCertificateFile "ssl/client.crtkey"
SSLProxyEngine on
```

5. Restart the vFabric Web Server instance for the configuration changes to take effect. For example:

```
prompt# cd /opt/vmware/vfabric-web-server/lb-server
prompt# bin/httpdctl restart
```

6. Test that everything is working correctly by accessing your application through the vFabric Web Server host or IP address and HTTPS port. For example, if the Web Server IP address is 192.11.22.33 and configured an HTTPS port of 8663:

```
https://192.11.22.33:8663/my-app
```

   Because you have configured your Web Server instance with the client certificate required by the tc Runtime instances, you will see your application, and yet all communication from your browser to the tc Runtime instance is using SSL.

Configure BMX for Monitoring vFabric Web Server Instances

As of version 5.1, all new vFabric Web Server instances are configured with BMX by default.

BMX is an Apache HTTPD framework that provides internal runtime information (performance metrics, status, configuration, and current capacity) to monitoring applications such as vFabric Hyperic. In turn, these types of applications monitor the health of vFabric Web Server instances by running BMX queries to gather metrics and configuration information.

New vFabric Web Server instances have the following default BMX configuration:

*   Three main BMX modules (mod_bmx, mod_bmx_status, and mod_bmx_vhost) are all enabled. Together, these modules provide overall runtime statistics of the Web Server instance, as well as the virtual hosts running within the instance.

*   Access is allowed only to processes running on http://localhost (IP address 127.0.0.1), or in other words, only to monitoring applications running on the same computer as the Web Server instance.

*   Access requires no authentication.

*   BMX access is enabled for all virtual hosts defined for the Web Server instance.

The default BMX configuration for vFabric Web Server instances make them immediately available for monitoring by monitoring applications.

The BMX-related modules are loaded into the Web Server instance using appropriate LoadModule directives in the conf/httpd.conf configuration file. Additional BMX configuration is in the conf/extra/httpd-info.conf file, which the main conf/httpd.conf file includes using the Include conf/extra/httpd-info.conf directive.

Procedure

1. To disable BMX access to your vFabric Web Server instance, comment out the appropriate LoadModule directives in the conf/httpd.conf configuration file for your instance as shown:

```
#LoadModule bmx_module c:/opt/vmware/vfabric-web-server/httpd-2.2/modules/mod_bmx.so
#LoadModule bmx_status_module c:/opt/vmware/vfabric-web-server/httpd-2.2/modules/mod_bmx_status.so
#LoadModule bmx_vhost_module c:/opt/vmware/vfabric-web-server/httpd-2.2/modules/mod_bmx_vhost.so
```

2. To allow BMX access to processes running on hosts other than the localhost, edit the <Location /bmx> directive in the conf/extra/httpd-info.conf file and add the IP address or fully qualified domain name to the Allow from directive. For example, to allow myhost.com access in addition to localhost:

```
<Location /bmx>
```

```
     SetHandler bmx-handler
     Order Deny,Allow
     Deny from all
     Allow from 127.0.0.1 myhost.com
</Location>
```

3. To restrict BMX access to a particular virtual host, put the `<Location /bmx>` directive inside the appropriate `<VirtualHost>` directive. For example:

```
<VirtualHost 10.1.2.3:80>
    DocumentRoot "/opt/vmware/vfabric-web-server/myserver/myhost.com/htdocs"
    ServerName status.myhost.com
    ...
    <Location /bmx>
        SetHandler bmx-handler
        Order Deny,Allow
        Deny from all
        Allow from 127.0.0.1 myhost.com
    </Location>
</VirtualHost>
```

What to do next

• Restart the vFabric Web Server instance for the configuration changes to take effect. For example, on Unix:

```
prompt# cd /opt/vmware/vfabric-web-server/myserver
prompt# bin/httpdctl restart
```

Metrics

This section lists the metrics reported by the Hyperic plugin for vFabric Web Server.

vFabric Web Server Server Metrics

This section lists the metric available for a server.

Table 7.1. Metric Definitions

Metric	Alias	Units	Category	Default On	Interval
Availability	Availability	percentage	Availability	true	5 min
Server Uptime	ServerUptimeSeconds	none	Availability	false	5 min
Busy Workers	BusyWorkers	none	Utilization	true	5 min
Idle Workers	IdleWorkers	none	Utilization	true	5 min
Bytes Served Per Second	KilobytesPerSec	KB	Throughput	false	5 min
Bytes Served Per Request	KilobytesPerReq	KB	Throughput	false	5 min
Requests Served Per Second	ReqPerSec	none	Throughput	true	5 min
Requests Served	TotalAccesses	none	Throughput	false	30 min
Requests Served per Minute	TotalAccesses1m	none	Throughput	true	10 min
Bytes Served	TotalTrafficKilobyes	KB	Throughput	false	30 min
Bytes Served per Minute	TotalTrafficKilobyes1m	KB	Throughput	true	10 min

Metric	Alias	Units	Category	Default On	Default Interval
Parent Server Generation	ParentServerGeneration	none	Throughput		

vFabric Web Server Virtual Host Metrics

This section lists the metrics available for a virtual host.

Table 7.2. Metric Definitions

Metric	Alias	Units	Category	Default On	Default Interval
Availability	Availability	percentage	Availability	true	10 min
Start Elapsed	StartElapsed	mu	Availability	true	10 min
Start Time	StartTime	epoch-millis	Availability	true	10 min
In Bytes GET	InBytesGET	none	Throughput	false	10 min
In Bytes GET per Minue	InBytesGET1m	none	Throughput		10 min
In Bytes HEAD	InBytesHEAD	none	Throughput	false	10 min
In Bytes HEAD per Minute	InBytesHEAD1m	none	Throughput	true	10 min
In Bytes POST	InBytesPOST	none	Throughput	false	10 min
In Bytes POST per Minute	InBytesPOST1m	none	Throughput	true	10 min
In Bytes PUT	InBytesPUT	none	Throughput	false	10 min
In Bytes PUT per Minute	InBytesPUT1m	none	Throughput	true	10 min
In Requests GET	InRequestsGET	none	Throughput	false	10 min
In Requests GET per Minute	InRequestsGET1m	none	Throughput	true	10 min
In Requests HEAD	InRequestsHEAD	none	Throughput	false	10 min
In Requests HEAD per Minute	InRequestsHEAD1m	none	Throughput	true	10 min
In Requests POST	InRequestsPOST	none	Throughput	false	10 min
In Requests POST per Minute	InRequestsPOST1m	none	Throughput	true	10 min
In Requests PUT	InRequestsPUT	none	Throughput	false	10 min
In Requests PUT per Minute	InRequestsPUT1m	none	Throughput	true	10 min
Out Bytes 200	OutBytes200	none	Throughput	false	10 min
Out Bytes 200 per Minute	OutBytes2001m	none	Throughput	true	10 min
Out Bytes 301	OutBytes301	none	Throughput	false	10 min
Out Bytes 301 per Minute	OutBytes3011m	none	Throughput	true	10 min
Out Bytes 302	OutBytes302	none	Throughput	false	10 min

Metric	Alias	Units	Category	Default On	Default Interval
Out Bytes 302 per Minute	OutBytes3021m	none	Throughput	true	10 min
Out Bytes 403	OutBytes403	none	Throughput	false	10 min
Out Bytes 403 per Minute	OutBytes4031m	none	Throughput	true	10 min
Out Bytes 404	OutBytes404	none	Throughput	false	10 min
Out Bytes 404 per Minute	OutBytes4041m	none	Throughput	true	10 min
Out Bytes 500	OutBytes500	none	Throughput	false	10 min
Out Bytes 500 per Minute	OutBytes5001m	none	Throughput	true	10 min
Out Responses 200	OutResponses200	none	Throughput	false	10 min
Out Responses 200 per Minute	OutResponses2001m	none	Throughput	true	10 min
Out Responses 301	OutResponses301	none	Throughput	false	10 min
Out Responses 301 per Minute	OutResponses3011m	none	Throughput	true	10 min
Out Responses 302	OutResponses302	none	Throughput	false	10 min
Out Responses 302 per Minute	OutResponses3021m	none	Throughput	true	10 min
Out Responses 401	OutResponses401	none	Throughput	false	10 min
Out Responses 401 per Minute	OutResponses4011m	none	Throughput	true	10 min
Out Responses 403	OutResponses403	none	Throughput	false	10 min
Out Responses 403 per Minute	OutResponses4031m	none	Throughput	true	10 min
Out Responses 404	OutResponses404	none	Throughput	false	10 min
Out Responses 404 Per Minute	OutResponses4041m	none	Throughput	true	10 min
Out Responses 500	OutResponses500	none	Throughput	false	10 min
Out Responses 500 per Minute	OutResponses5001m	none	Throughput	true	10 min
In Low Bytes	InLowBytes	none	Throughput	false	10 min
In Low Bytes per Minute	InLowBytes1m	none	Throughput	true	10 min
In Requests	InRequests	none	Throughput	false	10 min
In Requests per Minute	InRequests1m	none	Throughput	true	10 min
Out Responses	OutResponses	none	Throughput	false	10 min
Out Responses per Minute	OutResponses1m	none	Throughput	true	10 min

8. Security Information

VMware is committed to providing products and solutions that allow you to assess the security of your information, secure your information infrastructure, protect your sensitive information, and manage security information and events to assure effectiveness and regulatory compliance. As part of this commitment, the following vFabric Web Server-specific security information is provided to help you secure your environment:

- External Ports

- Resources That Must Be Protected

- Log File Locations

- User Accounts Created at Installation

- Obtaining and Installing Latest Version of Product

External Interfaces, Ports, and Services

A vFabric Web Server instance uses TCP/IP ports to receive incoming requests and send outgoing responses. Different protocols (such as HTTP or HTTPS) listen on different ports. You can change these port numbers when you create the Web Server instance using the `newserver` script, but these are the default values:

- HTTP: 80

- HTTPS: 443

If you have already created the Web Server instance, you can change its HTTP listen port by updating the `Listen 90 http` directive in the *INSTANCE-DIR*`/conf/httpd.conf` file, where *INSTANCE-DIR* refers to the directory in which the Web Server instance is located, such as `/opt/vmware/vfabric-web-server/myserver`. To update the HTTPS port, update the `Listen 443 https` directive in the *INSTANCE-DIR*`/conf/extra/httpd-ssl.conf` file.

vFabric Web Server does not have any external interfaces or services that need to be enabled or opened.

Resources That Must Be Protected

The following vFabric Web Server configuration files should be readable and writable *only* by the `root` (Unix) or `Administrator` (Windows) user:

- `conf/httpd.conf`

- `conf/userfile`

- All files in the `ssl` directory (if you have enabled SSL for the instance)

- `extra/conf/httpd-ssl.conf` (if you have enabled SSL for the instance)

- Any other `conf/httpd-XX.conf` file that you have for which there is an uncommented `Include` in the main `conf/httpd.conf` configuration file.

These configuration files are specific to a Web Server instance and are stored in the *INSTANCE-DIR* directory, where *INSTANCE-DIR* refers to the directory in which the Web Server instance is located, such as `/opt/vmware/vfabric-web-server/myserver`.

Log File Locations

The most important log files for a vFabric Web Server instance are as follows:

- `error_log`: Contains errors and diagnostic information that occurred while the Web Server instance was serving requests.

- `access_log`: Contains information about all Web Server requests.

- `ssl_request_log`: Applies only if you enabled SSL. Contains information about requests that came over HTTPS.

These log files are specific to a Web Server instance and are stored by default in the `INSTANCE-DIR`/`logs` directory, where `INSTANCE-DIR` refers to the directory in which the Web Server instance is located, such as `/opt/vmware/vfabric-web-server/myserver`.

The preceding log files should be readable and writable *only* by the `root` (Unix) or `Administrator` (Windows) user.

The `logs` directory also contains other log files associated with BMX and the vFabric License server.

User Accounts Created at Installation

If you install vFabric Web Server on Red Hat Enterprise Linux (RHEL) using the RPM, a user with the following characteristics is automatically created:

- ID: `vfhttpd`

- Group: `vfhttpd`

- Non-interactive, which means that you cannot directly log in to the RHEL computer as this user. Rather, you must log in as `root` or user with appropriate `sudo` privileges and `su - vfhttpd`.

When installing from RPM on RHEL, the installation directory will be owned by the `root` user, with group `root`.

When installing Web Server on Windows or Unix from a self-extracting `*.zip` file, a user account is *not* automatically created for you. Rather, you should install as `root` on Unix and `Administrator` on Windows.

Obtaining and Installing Security Updates

vFabric Web Server an HTTPD server based on open-source Apache HTTPD. A particular version of vFabric Web Server includes a particular version of Apache HTTPD, such as `httpd-2.2.22.1`. New versions of vFabric Web Server typically include an updated version of Apache HTTPD, some of which might fix important security vulnerabilities. To install these security updates, you install the new version of vFabric Web Server and then upgrade your existing instances.

To download the latest `*.zip` distributions of the vFabric Web Server, go to the [VMware Download Center](#).

When using RPMs on RHEL, use the `yum upgrade` command to upgrade to the latest vFabric Web Server version.

See [Upgrading vFabric Web Server](#) for details.

9. Managing Planned and Unplanned Outages

This chapter describes how to manage both planned and unplanned outages of vFabric Web Server.

Subtopics

[Managing Planned Outages](#)

[Managing Unplanned Outages](#)

[Backing Up vFabric Web Server](#)

Managing Planned Outages

In a planned outage, you schedule a time when vFabric Web Server instances will be briefly unavailable so that you can perform maintenance on the instance, create cold backups, and so on.

1. On the computer on which the vFabric Web Server instances are installed, log in as the `root` (Unix) or `Administrator` (Windows) user and stop all instances. For example, on Unix:

    ```
    prompt# cd /opt/vmware/vfabric-web-server/myserver/bin
    prompt# ./httpctl stop
    ```

    In the preceding example, the vFabric Web Server instance is located in the `/opt/vmware/vfabric-web-server/myserver` directory.

    The `stop` command forcibly ends all sessions. To specify that you want the Web Server instance to wait until all sessions end gracefully, use the `gracefulstop` command:

    ```
    prompt# ./httpdctl gracefulstop
    ```

    See [Creating and Using vFabric Web Server Instances](#) for additional details, such as Windows instructions.

2. Perform any required maintenance on the instance, such as updating its configuration and creating a [cold backup.](#) While the Web Server instances are stopped, the content deployed to the instances is not available to users.

3. If the Web Server instance is acting as a proxy or load balancer to one or more application server instances, such as tc Runtime instances, start them if they are not already running.

4. Start the vFabric Web Server instances. For example:

    ```
    prompt# cd /opt/vmware/vfabric-web-server/myserver/bin
    prompt# ./httpctl start
    ```

Managing Unplanned Outages

An unplanned outage is one that you do not schedule. Examples include fairly minor outages such as an unexpected power outage that causes the Web Server computer to shut down ungracefully to more critical outages such as a hard-disk failure.

Typically, if you have fully restored and restarted the computer on which vFabric Web Server is installed, all you need to do next is start the instances. Check the `error.log` log file in the `INSTANCE-DIR/logs` directory to ensure that failures do not occur during startup and that the configuration files are not corrupted. Invoke your deployed content to verify that you can access it.

If, however, the log file indicates that the Web Server instance did not start because, for example, the configuration files are corrupted, or your deployed content does not seem to be working correctly, you should restore the instance directory from a recent cold backup. The following procedure describes how to do this.

Procedure

1. Ensure that you have a recent cold backup of the Web Server instance that contains the last known good configuration and deployed content.

2. If necessary, log in as the `root` (Unix) or `Administrator` (Windows) user and stop all Web Server instances. For example, on Unix:

```
prompt# cd /opt/vmware/vfabric-web-server/myserver/bin
prompt# ./httpctl stop
```

    In the preceding example, the vFabric Web Server instance is located in the `/opt/vmware/vfabric-web-server/myserver` directory.

    The `stop` command forcibly ends all sessions. To specify that you want the Web Server instance to wait until all sessions end gracefully, use the `gracefulstop` command:

```
prompt# ./httpdctl gracefulstop
```

    See Creating and Using vFabric Web Server Instances for additional details, such as Windows instructions.

3. Change to the parent directory of the instance, then rename the instance directory. For example:

```
prompt$ cd /opt/vmware/vfabric-web-server
prompt$ mv myserver myserver-backup
```

    Note: This is just a precautionary step; you can remove this temporary backup after you fully restore the instance from the cold backup.

4. Unzip or un-tar your backup appropriately. For example, if you created a TAR file on Unix as described in Backing Up vFabric Web Server and the TAR file is called `myserverBackup-20120922.tar`, execute the following commands:

```
prompt$ cd /var/opt/vmware/vfabric-web-server
prompt$ tar xvf myserverBackup-20120922.tar
```

    If you compressed the tarball when creating the backup, use the appropriate option to untar it. For example, for bz2 compression:

```
prompt$ tar xjvf myserverBackup-20120922.bz2
```

5. If the Web Server instance is acting as a proxy or load balancer to one or more application server instances, such as vFabric tc Server, start them if they are not already running.

6. Start the instance to make your Web content is available again:

```
prompt# cd /opt/vmware/vfabric-web-server/myserver/bin
prompt# ./httpctl start
```

7. Check the `logs/error.log` file to ensure that the instance started without errors, then invoke your Web content and ensure that it is working correctly.

If you lost all data on the computer on which vFabric Web Server was installed, first re-install Web Server and then follow the preceding procedure to restore each Web Server instance.

Backing Up vFabric Web Server

When backing up vFabric Web Server, you need to create only backups of your instances; you do not need to back up the Web Server installation itself because you can always re-install it from your original distribution if necessary.

VMware recommends that you always take *cold backups* of your instances, which means you ZIP or TAR up the instance directory after stopping the instance.

A *hot backup* refers to creating a ZIP or TAR file of the instance directory without first stopping the instance. On Unix this method might be possible, and you likely can fully restore the instance from the hot backup. However, Windows may prevent you from even creating the hot backup in the first place if Web Server processes are holding locks on files that you are trying to back up. For this reason, VMware does not recommend hot backups.

The procedure describes how to perform a cold backup.

Procedure

1. Fully shut down the Web Server instances as described in [Managing Planned Outages](#).

2. Create a ZIP or a TAR file of each Web Server instance directory. For example, if your instances are located in the `/opt/vmware/vfabric-web-server` directory, and you want to create a TAR file on Unix of the `myserver` instance:

```
prompt$ cd /opt/vmware/vfabric-web-server
prompt$ tar cvf myserverBackup-20120922.tar myserver
```

This creates a TAR file called `myserverBackup-20120922.tar` with the top-most level being the instance directory (`myserver` in this case.)

When using `tar`, optionally use the `j` or `z` option to compress the result using bz2 or gzip, respectively. Compression results in more efficient tarballs. For example:

```
prompt$ tar cjvf myserverBackup-20120922.bz2 myserver
```

```
prompt$ tar czvf myserverBackup-20120922.gz myserver
```

10. Additional Documentation

The documentation in this guide provides information about what vFabric Web Server contains; how to install it; and how to create, start, and stop instances. The vFabric Web Server documentation does not, however, provide details about configuring and using the core Apache HTTP component; for that you must go elsewhere, such as the Apache documentation.

- Apache HTTP Server 2.2 http://httpd.apache.org/docs/2.2/

- OpenSSL http://www.openssl.org/docs/

- ASF Bugzilla page (search for known bugs in Apache HTTP Server) https://issues.apache.org/bugzilla/

- Searchable archive of Apache HTTP Users mail list http://marc.info/?l=apache-httpd-users&r=1&w=2