Progress®
Kemp® LoadMaster®

Deployment Guide JBoss

Generated on: 21 April 2026

Copyright

Visit the following page online to see Progress Software Corporation's current Product Documentation Copyright Notice/Trademark Legend: Product Documentation Copyright Notice & Trademarks | Progress

Table of Contents

Chapter 1: Introduction 4

Chapter 2: Template 7

Chapter 3: Configure the LoadMaster 9

Enable Subnet Originating Requests Globally 9

Enable Check Persist Globally 10

Create the JBoss Virtual Services 10

Create a JBoss Web App HTTP Virtual Service 10

Create a JBoss Web App HTTPS Offloaded Virtual Service 12

Create a JBoss Web App HTTPS Reencrypted Virtual Service 13

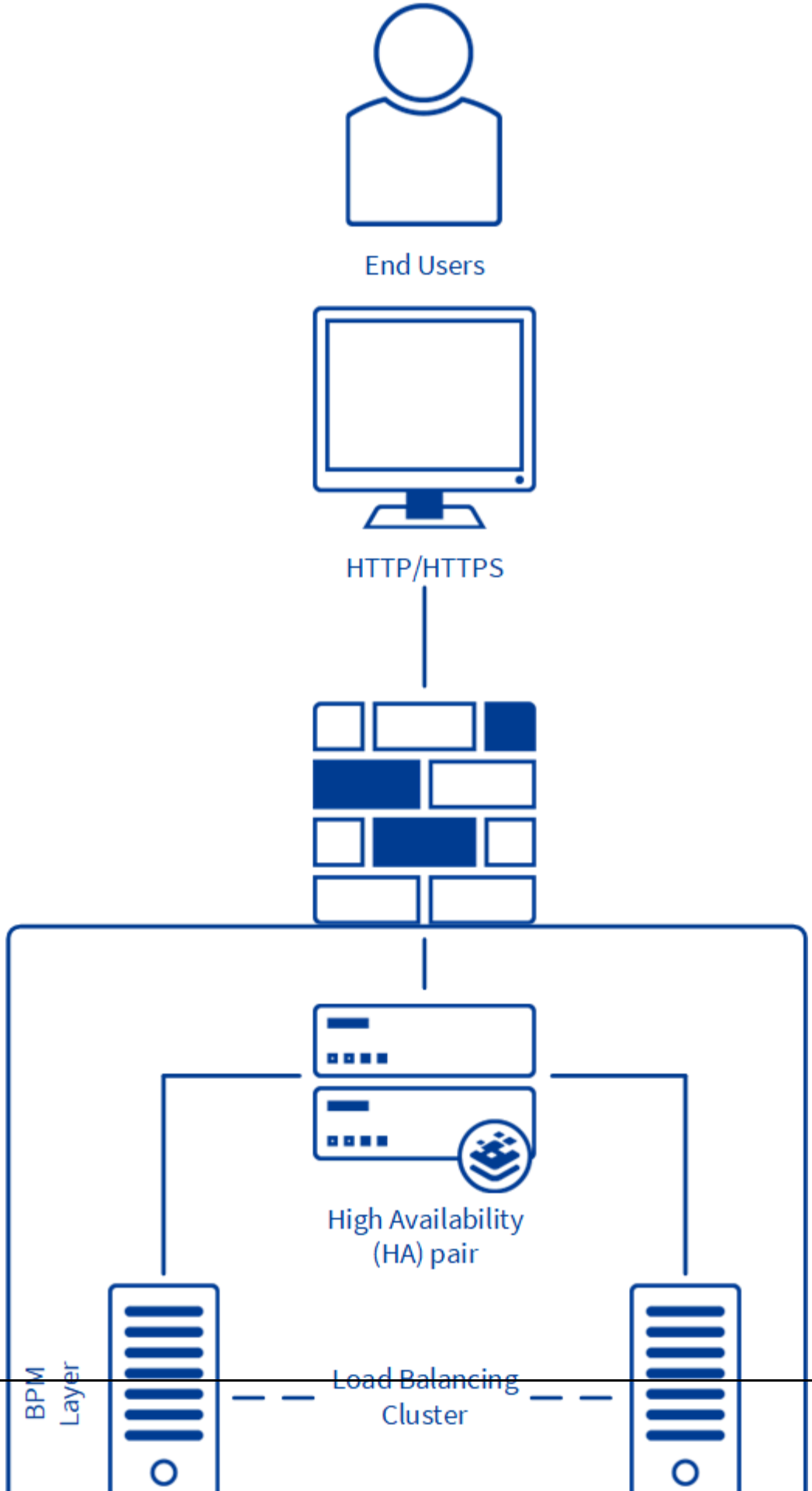
Create a JBoss Web App HTTPS Straight Through Virtual Service 15

Chapter 4: References 18

1

Introduction

JBoss is an application server authored by JBoss, which is now developed by Red Hat. It was renamed WildFly in November 2014. It is written in Java and implements the Java Platform, Enterprise Edition (Java EE) specification. It runs on multiple platforms and is open source.


End Users
HTTP/HTTPS
High Availability
(HA) pair
BPM
Layer
Load Balancing
Cluster

The LoadMaster is used to load balance the JBoss workload. The LoadMaster offers advanced Layer 4 and Layer 7 server load balancing, SSL Acceleration and a multitude of other advanced Application Delivery Controller (ADC) features. The LoadMaster intelligently and efficiently distributes user traffic among the application servers so that users get the best experience possible.

Document Purpose

This document provides the recommended LoadMaster settings used when load balancing the JBoss workload. The Progress Kemp Support Team is available to provide solutions for scenarios not explicitly defined. For further details, refer to the: LoadMaster Support and Learning Hub.

2

Template

Progress Kemp has developed a template containing our recommended settings for this workload. You can install this template to help create Virtual Services (VSs) because it automatically populates the settings. You can use the template to easily create the required VSs with the recommended settings. For some workloads, additional manual steps may be required such as assigning a certificate or applying port following. These steps are covered in the document, if needed.

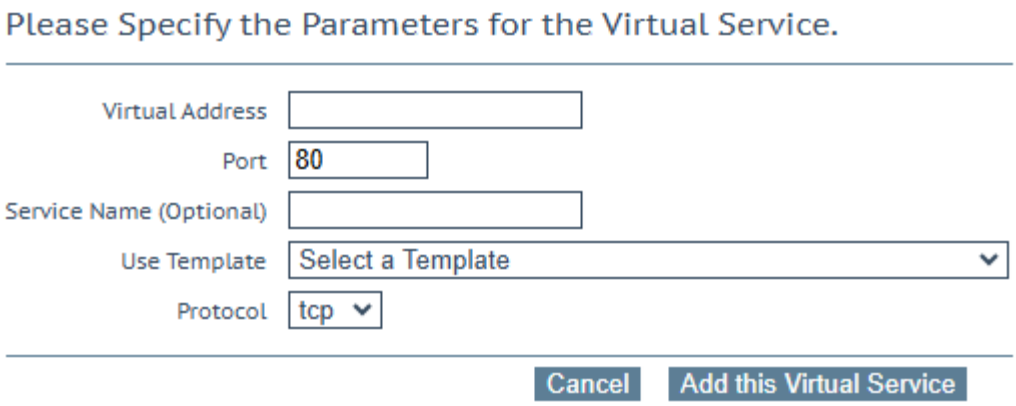
You can remove templates after use and this will not affect deployed services. If needed, you can make changes to any of the VS settings after using the template.

Download released templates from the following page: LoadMaster Templates.

You can import templates to the LoadMaster through the **Manage Templates** page under **Virtual Services** in the main menu of the LoadMaster User Interface (UI).

Click **Choose File**, select the template to install and click **Add New Template**. The template will then be available to select when you are adding a new Virtual Service.

To begin the process of creating a new Virtual Service, click **Virtual Services** and **Add New** in the main menu.



This opens the Virtual Service parameters page which prompts to enter the Virtual IP (VIP) address of the Virtual Service, the port, and the name to give the service and the protocol. You can select a previously-imported template in the **Use Template** drop-down list.

For more information and steps on how to use templates, refer to the Virtual Services and Templates, Feature Description.

3

Configure the LoadMaster

The deployed JBoss environment determines which of the following setups is used.

Contents

- Enable Subnet Originating Requests Globally
- Enable Check Persist Globally
- Create the JBoss Virtual Services

Enable Subnet Originating Requests Globally

It is best practice to enable the **Subnet Originating Requests** option globally.

In a one-armed setup (where the Virtual Service and Real Servers are on the same network/subnet) **Subnet Originating Requests** is usually not needed. However, enabling **Subnet Originating Requests** should not affect the routing in a one-armed setup.

In a two-armed setup where the Virtual Service is on network/subnet A, for example, and the Real Servers are on network B, **Subnet Originating Requests** should be enabled on LoadMasters with firmware version 7.1-16 and above.



When **Subnet Originating Requests** is enabled, the Real Server sees traffic originating from 10.20.20.21 (LoadMaster eth1 address) and responds correctly in most scenarios.

With **Subnet Originating Requests** disabled, the Real Server sees traffic originating from 10.0.0.15 (LoadMaster Virtual Service address on **eth0**) and responds to **eth0** which could cause asymmetric routing.

When **Subnet Originating Requests** is enabled globally, it is automatically enabled on all Virtual Services. If the **Subnet Originating Requests** option is disabled globally, you can choose whether to enable **Subnet Originating Requests** on a per-Virtual Service basis.

To enable **Subnet Originating Requests** globally, follow the steps below:

1. In the main menu of the LoadMaster User Interface (UI), go to **System Configuration > Miscellaneous Options > Network Options**.
2. Select the **Subnet Originating Requests** check box.

Enable Check Persist Globally

It is recommended that you change the **Always Check Persist** option to **Yes – Accept Changes**. Use the following steps:

1. Go to **System Configuration > Miscellaneous Options > L7 Configuration**.
2. Click the **Always Check Persist** drop-down arrow and select **Yes – Accept Changes**.

Create the JBoss Virtual Services

The following sections describe the recommended settings for the JBoss Virtual Services.

Contents

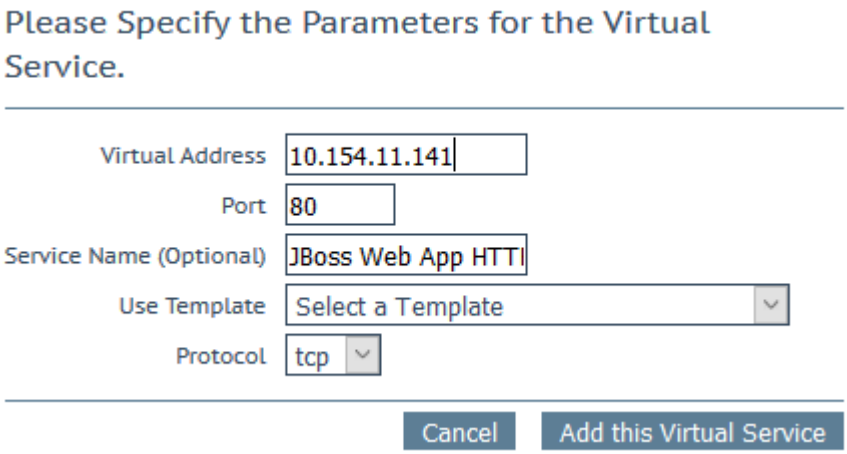
- Create a JBoss Web App HTTP Virtual Service
- Create a JBoss Web App HTTPS Offloaded Virtual Service
- Create a JBoss Web App HTTPS Reencrypted Virtual Service
- Create a JBoss Web App HTTPS Straight Through Virtual Service

Create a JBoss Web App HTTP Virtual Service

The following are the steps involved and the recommended settings to configure the JBoss Web App HTTP

Virtual Service:

1. In the main menu of the LoadMaster Web User Interface (WUI), go to **Virtual Services > Add New**.



2. Enter a valid **Virtual Address**.
3. Enter **80** as the **Port**.
4. Enter a recognizable **Service Name**, such as **JBoss Web App HTTP**.
5. Click **Add this Virtual Service**.
6. Configure the settings as recommended in the following table:

Section	Option	Value	Comment
Standard Options	Persistence Mode	Source IP Address	
	Timeout	1 Hour	
	Scheduling Method	least connection	
	Idle Connection Timeout	660 (seconds)	Click Set Idle Timeout.
Real Servers	Checked Port	8080	

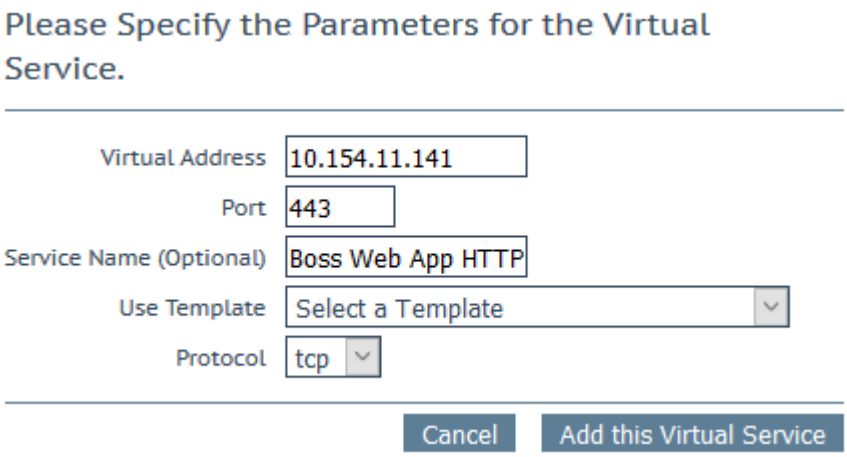
7. Add the Real Servers:
   1. Expand the **Real Servers** section.
   2. Click **Add New**.
   3. Enter the address of the relevant real server.
   4. Complete the other fields as required.

5. Click **Add this Real Server** then click **OK** to the pop-up message.
6. Repeat the steps above to add more Real Servers as needed, based on your environment.

Create a JBoss Web App HTTPS Offloaded Virtual Service

The following are the steps involved and the recommended settings to configure the JBoss Web App HTTPS Offloaded Virtual Service:

1. In the main menu of the LoadMaster WUI, go to **Virtual Services > Add New**.



2. Enter a valid **Virtual Address**.
3. Enter **443** as the **Port**.
4. Enter a recognizable Service Name, such as JBoss Web App HTTPS Offloaded.
5. Click **Add this Virtual Service**.
6. Configure the settings as recommended in the following table:

Section	Option	Value	Comment
Standard Options	Persistence Mode	Source IP Address	
	Timeout	1 Hour	
	Scheduling Method	least connection	
	Idle Connection Timeout	660 (seconds)	Click Set Idle Timeout.

Section	Option	Value	Comment
SSL Properties	SSL Acceleration	Enabled	
	Cipher Set	BestPractices	
Advanced Properties	Add a Port 80 Redirector VS	https://%h%s	Click the **Add HTTP Redirector** button. This automatically creates a redirect on port 80.
Real Servers	Checked Port	8080	

7. Add the Real Servers:
   1. Expand the **Real Servers** section.
   2. Click **Add New**.
   3. Enter the address of the relevant Real Server.
   4. Complete the other fields as required.
   5. Click **Add this Real Server** then click **OK** to the pop-up message.
   6. Repeat the steps above to add more Real Servers as needed, based on your environment.

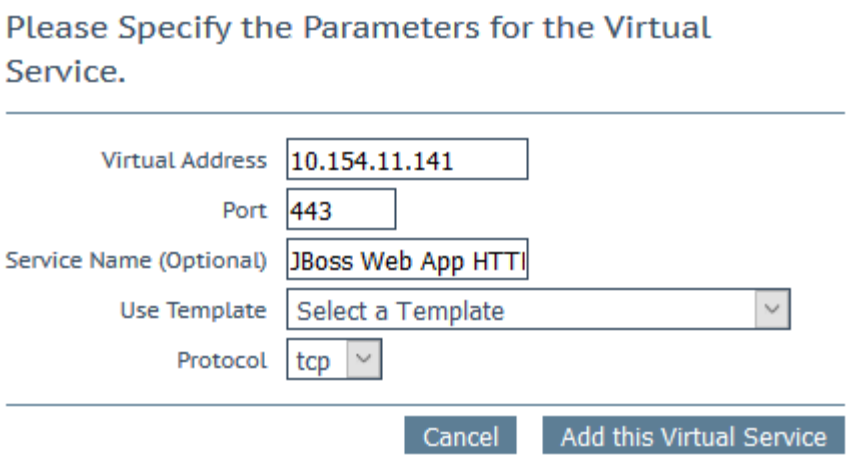
Create a JBoss Web App HTTPS Offloaded HTTP Redirect Virtual Service

Clicking the **Add HTTP Redirector** button automatically creates a port 80 redirect Virtual Service. This is optional, but the purpose of this Virtual Service is to redirect any clients who have connected using HTTP to the HTTPS Virtual Service. We also recommend changing the **Persistence Mode** to **None**.

Create a JBoss Web App HTTPS Reencrypted Virtual Service

The following are the steps involved and the recommended settings to configure the JBoss Web App HTTPS Reencrypted Virtual Service:

1. In the main menu of the LoadMaster WUI, go to **Virtual Services > Add New**.



2. Enter a valid **Virtual Address**.
3. Enter **443** as the **Port**.
4. Enter a recognizable Service Name, such as JBoss Web App HTTPS Reencrypted.
5. Click **Add this Virtual Service**.
6. Configure the settings as recommended in the following table:

Section	Option	Value	Comment
Standard Options	Persistence Mode	Source IP Address	
	Timeout	1 Hour	
	Scheduling Method	least connection	
	Idle Connection Timeout	660 (seconds)	Click Set Idle Timeout.
SSL Properties	SSL Acceleration	Enabled	
	Reencrypt	Enabled	
	Cipher Set	Best Practices	

Section	Option	Value	Comment
Advanced Properties	Add a Port 80 redirector VS	https://%h%s	Click the **Add HTTP Redirector** button. This automatically creates a redirect on port 80.
Real Servers	Checked Port	8443	

7. Add the Real Servers:
   1. Expand the **Real Servers** section.
   2. Click **Add New**.
   3. Enter the address of the relevant Real Server.
   4. Complete the other fields as required.
   5. Click **Add this Real Server** then click **OK** to the pop-up message.
   6. Repeat the steps above to add more Real Servers as needed, based on your environment.

Create a JBoss Web App HTTPS Reencrypted Redirect Virtual Service

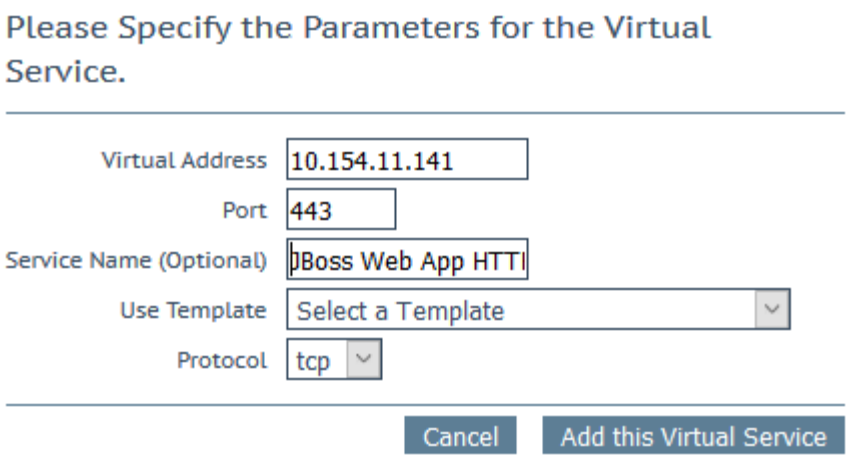
Clicking the **Add HTTP Redirector** button automatically creates a port 80 redirect Virtual Service. This is optional, but the purpose of this Virtual Service is to redirect any clients who have connected using HTTP to the HTTPS Virtual Service. We also recommend changing the **Persistence Mode** to **None**.

Create a JBoss Web App HTTPS Straight Through Virtual Service

Create a JBoss Web App HTTPS Straight Through Virtual Service

The following are the steps involved and the recommended settings to configure the JBoss Web App HTTPS straight through Virtual Service:

1. In the main menu of the LoadMaster WUI, go to **Virtual Services > Add New**.



2. Enter a valid **Virtual Address**.
3. Enter **443** as the **Port**.
4. Enter a recognizable Service Name, such as JBoss Web App HTTPS straight through.
5. Click Add this Virtual Service.
6. Configure the settings as recommended in the following table:

Section	Option	Value	Comment
Standard Options	Persistence Mode	Source IP Address	
	Timeout	1 Hour	
	Scheduling Method	least connection	
	Idle Connection Timeout	660	Click Set Idle Timeout.
Advanced Properties	Add a Port 80 Redirector VS		Click the **Add HTTP Redirector** button. This automatically creates a redirect on port 80.
Real Servers	Checked Port	8443	

7. Add the Real Servers:
   1. Expand the **Real Servers** section.

2. Click **Add New**.
3. Enter the address of the relevant Real Server.
4. Complete the other fields as required.
5. Click **Add this Real Server** then click **OK** to the pop-up message.
6. Repeat the steps above to add more Real Servers as needed, based on your environment.

Create a JBoss Web App HTTPS Straight Through HTTP Redirect Virtual Service

Clicking the **Add HTTP Redirector** button automatically creates a port 80 redirect Virtual Service. This is optional, but the purpose of this Virtual Service is to redirect any clients who have connected using HTTP to the HTTPS Virtual Service. We also recommend changing the **Persistence Mode** to **None**.

4

References

For further details on Virtual Services and Templates, refer to the following document:

Virtual Services and Templates Feature Description