Progress®
Kemp® LoadMaster®

Deployment Guide Apache Tomcat

Generated on: 16 September 2024

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

Document Purpose 4

Intended Audience 5

Chapter 2: Apache Tomcat Template 6

Chapter 3: Architecture 7

Chapter 4: Configure the LoadMaster 9

Enable Subnet Originating Requests Globally 9

Enable Check Persist Globally 10

Create the Apache Tomcat Virtual Services 10

Create an Apache Tomcat HTTP Virtual Service 11

Create an Apache Tomcat HTTPS Virtual Service 12

Create an Apache Tomcat HTTPS Offloaded Virtual Service 13

Create an Apache Tomcat HTTPS Re-encrypt Virtual Service 15

Chapter 5: References 18

1

Introduction

Introduction

Apache Tomcat, often referred to as Tomcat, is an open-source Java Servlet Container developed by the Apache Software Foundation. Tomcat implements several Java EE specifications including Java Servlet, JavaServer Pages, Java EL, and WebSocket, and provides a 'pure Java' HTTP web server environment in which Java code can run.

The LoadMaster is used to load balance the Apache Tomcat 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.

Related Links

- Document Purpose
- Intended Audience

Document Purpose

Document Purpose

This document provides the recommended LoadMaster settings used when load balancing the Apache Tomcat workload. The Progress Kemp Support Team is available to provide solutions for scenarios not

explicitly defined. The Progress Kemp support site can be found at: https://support.kemptechnologies.com

Intended Audience

Intended Audience

This document is intended to be read by anyone who is interested in configuring the LoadMaster to optimize the Apache Tomcat server.

2

Apache Tomcat Template

Apache Tomcat Template

Progress Kemp has developed a template containing our recommended settings for Apache Tomcat. You can install this template on the LoadMaster and use it when creating Virtual Services. Using a template automatically populates the settings in the Virtual Services, which is quicker and easier than manually configuring each Virtual Service. If needed, you can make changes to any of the Virtual Service settings after using the template.

Download released templates from the following page: LoadMaster Templates.

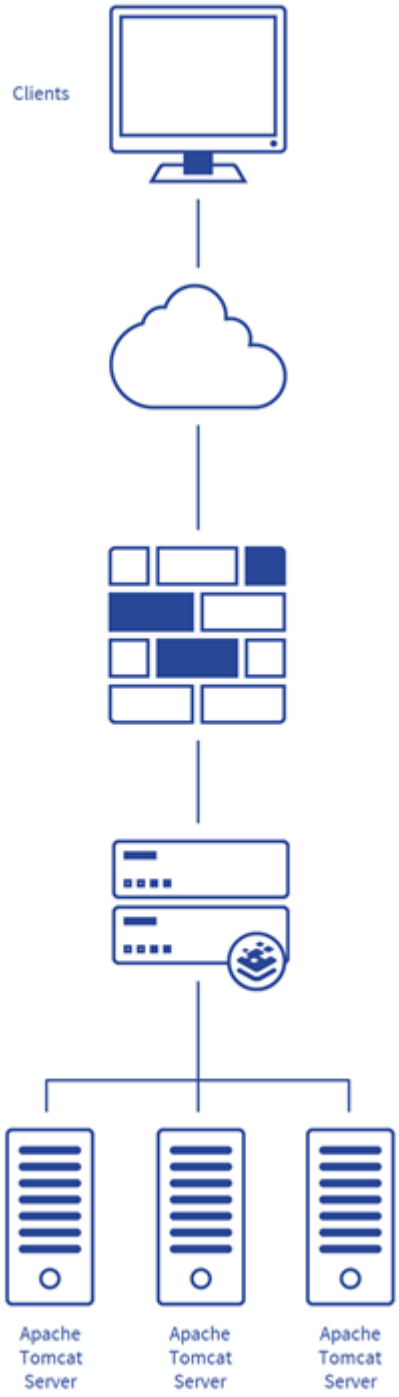
For more information and steps on how to import and use templates, refer to the Virtual Services and Templates, Feature Description.

For steps on how to manually add and configure each of the Virtual Services, refer to the other sections of this document.

3

Architecture

Architecture



4

Configure the LoadMaster

Configure the LoadMaster

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

Related Links

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

Enable Subnet Originating Requests Globally

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

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 Apache Tomcat Virtual Services

The following sections describe the recommended settings for the Apache Tomcat Virtual Services.

Note: It is not possible to load balance HTTP and HTTPS traffic on the same Virtual Service. Using additional ports is not a solution because these are different protocols. To load balance HTTP and HTTPS traffic, you must create two Virtual Services:

- One for HTTP traffic (normally port 80)
- One for HTTPS traffic (normally port 443)

Related Links

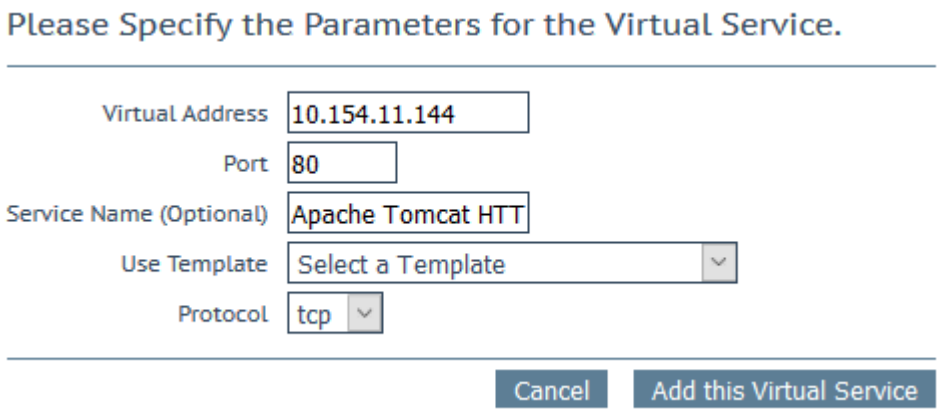
- Create an Apache Tomcat HTTP Virtual Service
- Create an Apache Tomcat HTTPS Virtual Service
- Create an Apache Tomcat HTTPS Offloaded Virtual Service
- Create an Apache Tomcat HTTPS Re-encrypt Virtual Service

Create an Apache Tomcat HTTP Virtual Service

Create an Apache Tomcat HTTP Virtual Service

The following are the steps involved and the recommended settings to configure the Apache Tomcat HTTP Virtual Service:

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



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

Section	Option	Value
Standard Options	Persistence Mode	Active Cookie
	Timeout	1 Hour

Section	Option	Value
	Cookie name	JSESSIONID
	Scheduling Method	least connection
	Idle Connection Timeout	900

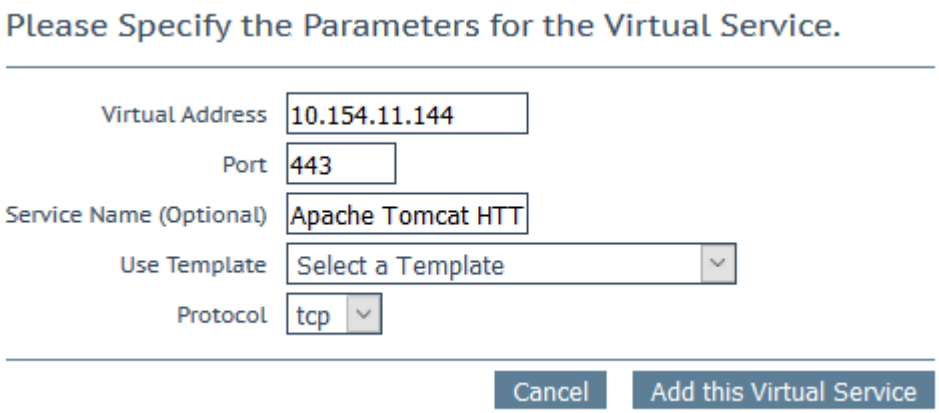
7. Add the Real Servers:
   1. Expand the **Real Servers** section.
   2. Click **Add New**.
   3. Enter the address of the relevant Real Server.
   4. Complete the other fields as required.
   5. Click **Add this Real Server** then click **OK** to the pop-up message.
   6. Repeat the steps above to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

Create an Apache Tomcat HTTPS Virtual Service

Create an Apache Tomcat HTTPS Virtual Service

The following are the steps involved and the recommended settings to configure the Apache Tomcat HTTPS Virtual Service:

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



2. Type a valid **Virtual Address**.
3. Type **443** as the **Port**.
4. Enter a recognizable **Service Name**, such as Apache Tomcat HTTPS Virtual Service.

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	900	
Advanced Properties	Add a Port 80 Redirector VS	https://%h%s	Click **Add HTTP Redirector**. This automatically creates a redirect on port 80.

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 an Apache Tomcat HTTPS Redirect Virtual Service

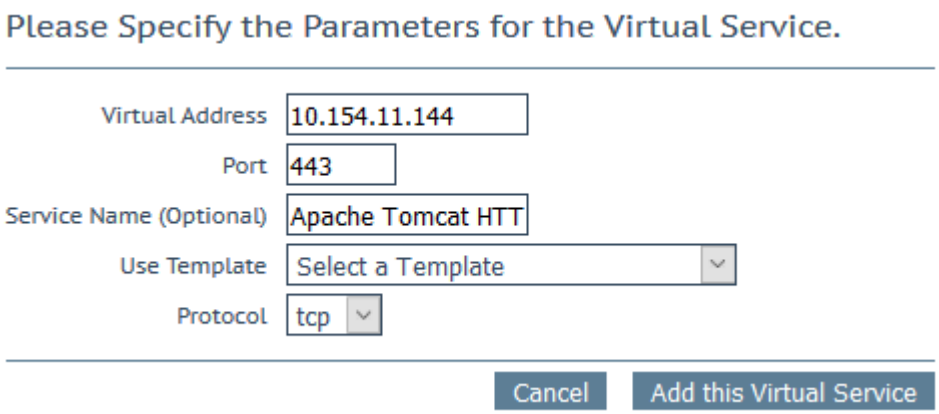
Clicking the **Add HTTP Redirector** button automatically creates a port 80 redirect Virtual Service. This is optional, but the purpose of this Virtual Service is to redirect any clients who have connected using HTTP to the HTTPS Virtual Service. We also recommend changing the **Real Server Check Method**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

Create an Apache Tomcat HTTPS Offloaded Virtual Service

Create an Apache Tomcat HTTPS Offloaded Virtual Service

The following are the steps involved and the recommended settings to configure the Apache Tomcat HTTPS Offloaded Virtual Service:

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



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

Section	Option	Value	Comments
Standard Options	Persistence Mode	Active Cookie	
	Timeout	1 Hour	
	Cookie name	JSESSSIONID	
	Scheduling Method	least connection	
	Idle Connection Timeout	900	
SSL Properties	SSL Acceleration	Enabled	
	Cipher Set	BestPractices	

Section	Option	Value	Comments
Advanced Properties	Add a Port 80 Redirector VS	https://%h%s	Click **Add HTTP Redirector**. This automatically creates a redirect on port 80.

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 an Apache Tomcat HTTPS Offloaded HTTP Redirect Virtual Service

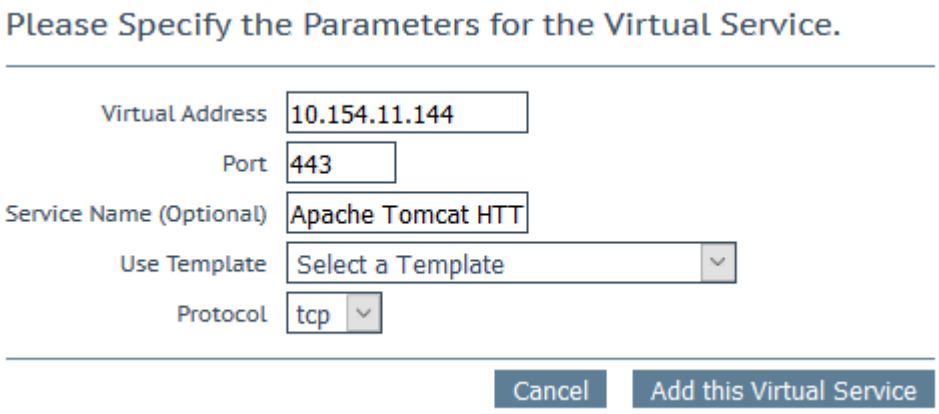
Clicking the **Add HTTP Redirector** button automatically creates a port 80 redirect Virtual Service. This is optional, but the purpose of this Virtual Service is to redirect any clients who have connected using HTTP to the HTTPS Virtual Service. We also recommend changing the **Real Server Check Method**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

Create an Apache Tomcat HTTPS Re-encrypt Virtual Service

Create an Apache Tomcat HTTPS Re-encrypt Virtual Service

The following are the steps involved and the recommended settings to configure the Apache Tomcat HTTPS Re-encrypt Virtual Service:

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



2. Type a valid **Virtual Address**.
3. Type **443** as the **Port**.
4. Enter a recognizable **Service Name**, such as **Apache Tomcat HTTPS Re-encrypt** Virtual Service.
5. Click **Add this Virtual Service**.
6. Configure the settings as recommended in the following table:

Section	Option	Value	Comments
Standard Options	Persistence Mode	Active Cookie	
	Timeout	1 Hour	
	Cookie name	JSESSSIONID	
	Scheduling Method	least connection	
	Idle Connection Timeout	900	
SSL Properties	SSL Acceleration	Enabled	
	Cipher Set	Best Practices	
Advanced Properties	Add a Port 80 Redirector VS	https://%h%s	Click **Add HTTP Redirector**. This automatically creates a redirect on port 80.

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 an Apache Tomcat HTTPS Re-encrypt 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. Progress Kemp also recommends changing the **Real Server Check Method** and **Persistence Mode** to **None**.

5

References

References

Unless otherwise specified, the following documents can be found at https://docs.progress.com/.

Virtual Services and Templates, Feature Description