Progress®
Kemp® LoadMaster®

Deployment Guide Evertz Mediator X

Generated on: 10 March 2025

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 6

Chapter 3: Virtual Service - Mediator-X Global Access 7

Using the Template 7

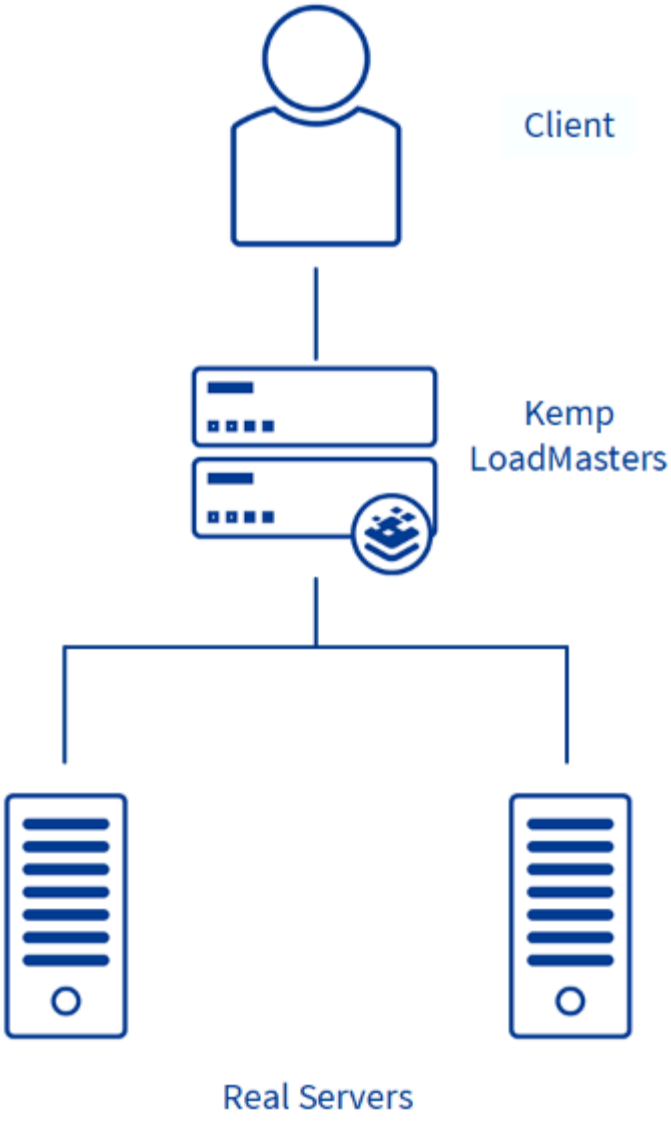
Configuring Evertz Mediator-X for Load Balancing 8

API Configuration 9

1

Introduction

This guide details the steps required to load balance the Mediator-X user interface traffic and Application Programming Interface (API) endpoint traffic using the LoadMaster. Evertz Mediator-X unifies content acquisition, content processing, media management, production, playout, and delivery into a single, integrated environment. The unification of these services on a single platform delivers optimized media workflows and increased operational efficiency.



The LoadMaster offers advanced Layer 4 and Layer 7 server load balancing, SSL Acceleration, and a multitude of other advanced Application Delivery and Optimization (ADC) features. The LoadMaster can load balance the Evertz Mediator-X servers. The LoadMaster intelligently and efficiently distributes user traffic among the application servers so that users get the best experience possible.

This document provides guidance and recommended settings on how to load balance Evertz Mediator-X servers with a LoadMaster. The Progress Kemp Support Team is available to provide solutions for scenarios not explicitly defined.

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.

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

3

Virtual Service - Mediator-X Global Access

For instructions on how to use the template and details on the API configuration settings, refer to the sections below.

Related Links

For details, see the following topics:

- Using the Template
- Configuring Evertz Mediator-X for Load Balancing
- API Configuration

Using the Template

This step-by-step setup of the Virtual Service leverages the Progress Kemp application template for Evertz Mediator-X.

The table in the API Configuration section outlines the settings configured by the application template. You can use this information to manually configure the Virtual Service or use the LoadMaster API and automation tools.

To configure a Virtual Service using the application template, perform the following steps:

1. In the main menu of the LoadMaster User Interface (UI), go to **Virtual Services > Add New**.
2. Type a valid **Virtual Address**.

3. Select **Mediator X Global Access** in the **Use Template** drop-down list.
4. Click **Add this Virtual Service**.
5. In the left-hand navigation select **View/Modify Services**.
6. Click **Modify** on the **Mediator X Global Access** Virtual Service on port tcp 80.
7. Expand the **Real Servers** section.
8. Click **Add New**.
9. Type the **Real Server Address**. (These are the Evertz Mediator servers.)
10. Confirm that port **80** is entered.
11. Click **Add This Real Server**.
12. Repeat these steps to add more Real Servers as needed.
13. In the left-hand navigation select **View/Modify Services**
14. Click **Modify** on the **Mediator X Global Access Secure** Virtual Service on port tcp 443.
15. Expand the **Real Servers** section.
16. Click **Add New**.
17. Type the **Real Server Address**. (These are the Evertz Mediator servers.)
18. Confirm that port **443** is entered.
19. Click **Add This Real Server**.
20. Repeat these steps to add more Real Servers as needed.

Configuring Evertz Mediator-X for Load Balancing

Some changes must be made to the Mediator-X back-end servers for them to be correctly load balanced. These changes should be configured by an Evertz Deployment Team. Contact your Evertz representative for further information.

1. On the Mediator-X deployment, connect to Node0 and log in as the root user. You can do this by executing the command **sudo su** and then entering the system-specific shell access password.
2. Navigate to the directory **/srv/salt/pillar**. You can do this by executing the command **cd /srv/salt/pillar**.
3. Edit the file **system.sls**, for example, using a text editor such as Nano or Vim: **nano system.sls**.
4. Find the **virtual_ips** parameter and add the virtual IP address that will be used for the load balanced deployment. This is the user-facing IP address that all clients will connect to when accessing the load balanced Mediator-X services.
5. The result should look like the following:

```
virtual_ips : 10.0.1.50

# Set the loadbalancer virtual ip or leave blank
```

6. Save and exit the **system.sls** file.
7. Run the **salt** command and call the **state.highstate** function, which automatically applies the changed configuration across all Mediator-X nodes. The full command to execute is:

```
sudo salt "*" state.highstate
```

8. Run the **salt** command and call a function to restart the NGINX service across all Mediator-X nodes. The full command to execute is:

```
sudo salt -G 'is_mediatorx:True' service.restart nginx
```

API Configuration

This table outlines the API parameters and values set using the Progress Kemp application template. You can use these settings with scripts and automation tools.

Mediator X Global Access

The API parameter and value settings for the Mediator X Global Access Virtual Service are listed in the table below:

API Parameter	API Value
port	80
prot	tcp
Schedule	rr
Persist	header
cookie	X-Forwarded-For
Persisttimeout	300

Mediator X Global Access Secure

The API parameter and value settings for the Mediator X Global Access Secure Virtual Service are listed in the table below:

API Parameter	API Value
port	443
prot	tcp
Schedule	rr
Persist	header
cookie	X-Forwarded-For

API Parameter	API Value
Persisttimeout	300