Progress®
Kemp® LoadMaster®

Deployment Guide MinIO Object Storage

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 5

Intended Audience 5

Document Purpose 5

Chapter 2: Template 6

Chapter 3: MinIO Object Storage 7

Chapter 4: LoadMaster Global Settings 9

Enable Subnet Originating Requests Globally 9

Chapter 5: LoadMaster Virtual Services. 11

Create a Virtual Service using a Template 12

MinIO S3 Virtual Services 12

S3 HTTP Virtual Service Recommended API Settings (optional) 12

S3 HTTPS Passthrough Virtual Service Recommended API Settings (optional). 13

S3 HTTPS Offloaded Virtual Service Recommended API Settings (optional) 13

S3 HTTPS Reencrypted Virtual Service Recommended API Settings (optional). 14

Chapter 6: Troubleshooting. 16

Connections Rejected 16

Source/Client IP Visibility 16

Chapter 7: References 17

1

Introduction

MinIO is a software-defined, flexible, and expandable object storage solution that runs seamlessly in the private cloud, public cloud, and at the edge. In combination with a Progress Kemp load balancer, a MinIO solution can provide the necessary availability and scale for object storage using the S3 protocol.

Related Links

For details, see the following topics:

- Intended Audience
- Document Purpose

Intended Audience

Anyone interested in configuring the LoadMaster to load balance MinIO.

Document Purpose

This deployment guide provides instructions on how to configure the LoadMaster to load balance MinIO nodes using Progress Kemp application templates. You should only use this guide as a reference for the load balancing configuration of MinIO services because each environment is unique and may have different requirements. This guide outlines the load balancing configuration using standard ports (80 and 443) in the Progress Kemp application templates which connect back to the MinIO default port of tcp 9000, but you can also leverage MinIO custom ports based on the environment.

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.

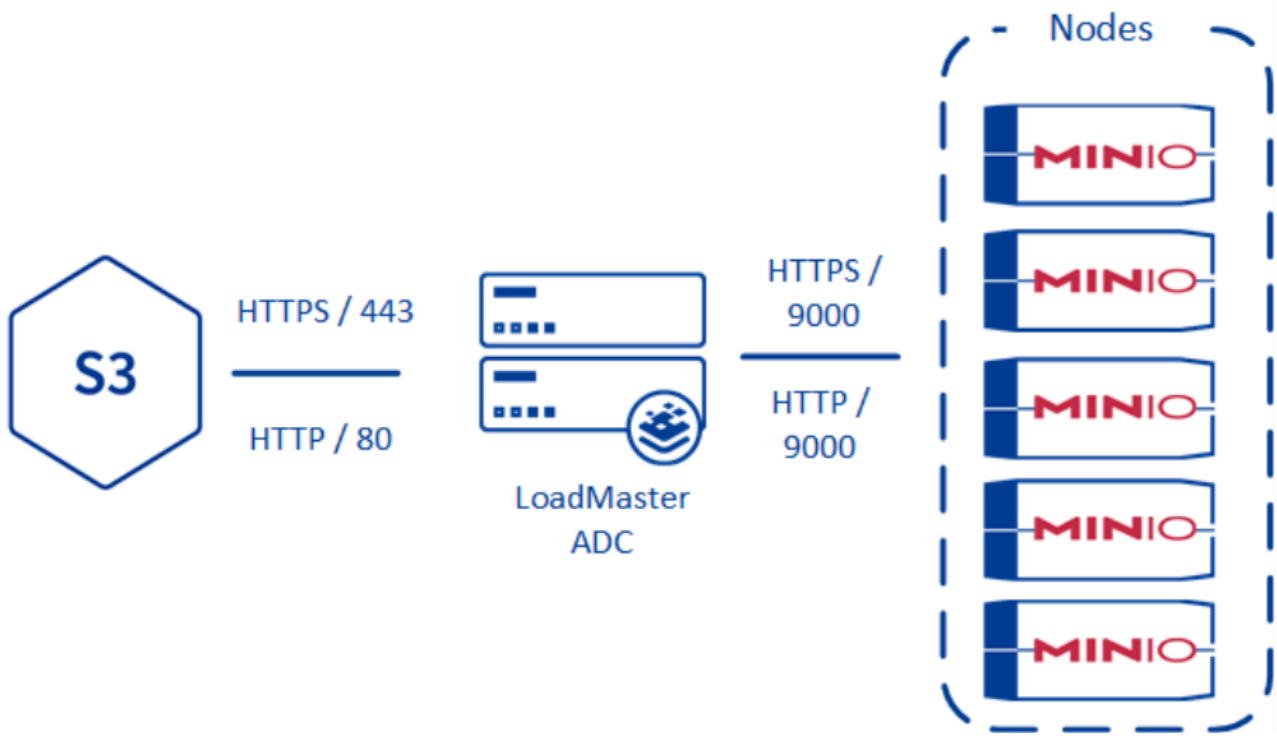
3

MinIO Object Storage

MinIO is a software-defined object storage solution for large archives, media, and web data storage. Policy-driven data management allows MinIO to move data seamlessly between on-premises and public cloud storage to optimize availability, protection, performance, and cost.

The following table provides a list of the MinIO default ports and protocols used for accessing the storage.

Protocols	Transport Protocol or Daemon Service	Port
S3	HTTP/HTTPS	9000


Nodes
S3
HTTPS / 443
HTTP / 80
LoadMaster
ADC
HTTPS /
9000
HTTP /
9000
MINIO
MINIO
MINIO
MINIO
MINIO

4

LoadMaster Global Settings

Before setting up the Virtual Services, you should configure the following global settings to support the workload.

Related Links

For details, see the following topics:

- 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.

5

LoadMaster Virtual Services

This step-by-step setup of Virtual Services leverages the Progress Kemp application template for MinIO Object Storage with Layer 7.

Layer 7 by default does not use transparency and therefore the IP address of the LoadMaster is used when accessing the MinIO nodes. The X-Forwarded-For header is leveraged to provide the original source IP address in the logs for troubleshooting purposes. When a secure connection is used, a certificate must be installed on the LoadMaster to decrypt the traffic for the X-Forwarded-For header insertion. This traffic can then be re-encrypted or offloaded depending on the security requirements.

The tables in the sections below outline the settings configured by the application template. You can use this information to manually configure Virtual Services or using the LoadMaster Application Programming Interface (API) and automation tools.

There are three supported configurations:

- **SSL pass-through:** The SSL certificate is installed on the MinIO nodes as a custom server certificate.
- **SSL termination and reencryption:** This might be beneficial if you are already doing SSL certificate management on the load balancer rather than installing the SSL certificate on the MinIO nodes. This configuration provides the additional security benefit of limiting the attack surface to the LoadMaster.
- **SSL termination with HTTP:** In this configuration, SSL is terminated on the LoadMaster and communication from the LoadMaster to MinIO is non-encrypted to take advantage of SSL offload.

Related Links

For details, see the following topics:

- Create a Virtual Service using a Template
- MinIO S3 Virtual Services

Create a Virtual Service using a Template

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

1. In the main menu of the LoadMaster WUI, go to **Virtual Services > Add New**.
2. Type a valid **Virtual Address**.
3. Select the appropriate template in the **Use Template** drop-down list.
4. Click **Add this Virtual Service**.
5. **Required only for TLS/SSL Offload and Reencrypt:** Expand the **SSL Properties** section.
6. **Required only for TLS/SSL Offload and Reencrypt:** Select the certificate to use from **Available Certificates** and click the arrow (**>**) to move it to **Assigned Certificates**.
7. Expand the **Real Servers** section.
8. Click **Add New**.
9. Type the **Real Server Address**.
10. Confirm that the correct port is entered.
11. Click **Add This Real Server**.
12. Add any additional Real Servers.

MinIO S3 Virtual Services

The following section outlines the Layer 7 configuration options for using S3 with MinIO Object Storage.

S3 HTTP Virtual Service Recommended API Settings (optional)

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.

API Parameter	API Value
port	80
prot	tcp
VStype	http
SubnetOriginating	1
Schedule	lc

API Parameter	API Value
AddVia	1
CheckType	http
CheckPort	9000
CheckUseGet	0
CheckURL	/minio/health/ready

S3 HTTPS Passthrough Virtual Service Recommended API Settings (optional)

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.

API Parameter	API Value
port	443
prot	tcp
VStype	http
SubnetOriginating	1
Schedule	lc
CheckType	https
CheckPort	9000
CheckUseGet	0
CheckURL	/minio/health/ready

S3 HTTPS Offloaded Virtual Service Recommended API Settings (optional)

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.

API Parameter	API Value
port	443
prot	tcp
VStype	http
SubnetOriginating	1
Schedule	lc
SSLAcceleration	1
AddVia	1
TLSType	1
CipherSet	BestPractices
CheckType	http
CheckPort	9000
CheckUseGet	0
CheckURL	/minio/health/ready

S3 HTTPS Reencrypted Virtual Service Recommended API Settings (optional)

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.

API Parameter	API Value
port	443
prot	tcp
VStype	http
SubnetOriginating	1
Schedule	lc
SSLAcceleration	1

API Parameter	API Value
SSLReencrypt	1
AddVia	1
TLSType	1
CipherSet	BestPractices
CheckType	https
CheckPort	9000
CheckUseGet	0
CheckURL	/minio/health/ready

6

Troubleshooting

Refer to the sections below for details on some common issues seen when load balancing the MinIO workload.

Related Links

For details, see the following topics:

- Connections Rejected
- Source/Client IP Visibility

Connections Rejected

When using a non-default TCP port or offloading for MinIO services, you must ensure the Real Server port is correct. This is a common mistake when configuring the Real Servers when the Virtual Services port is different from the Real Server port. See the table in the MinIO Object Storage section of this document for the required Real Server port for MinIO.

Source/Client IP Visibility

If the client source IP address is required for audit logging, configure your load balancer so that it passes the requests through with the original requesting IP address.

7

References

Some useful resources are listed below:

MinIO Storage Solutions

Useful, related documents are listed below:

SSL Accelerated Services Feature Description

Transparency Feature Description

RESTful API documentation