Technical Notes

EMC ScaleIO

Version 2.x

High Availability for the ScaleIO Gateway

P/N 302-002-687
REV 04
September 2017

This document explains how to set up high availability for the ScaleIO Gateway when using REST API commands. Topics include:

- Preface 2
- Introduction 3
- Overview of the high-availability solution 3
- Configuration workflow 4
- Example using Apache httpd and Keepalived on Ubuntu 5
- Troubleshooting and getting help 11

EMC²

Preface

As part of an effort to improve its product lines, EMC periodically releases revisions of its software and hardware. Therefore, some functions described in this document might not be supported by all versions of the software or hardware currently in use. The product release notes provide the most up-to-date information on product features.

Contact your EMC technical support professional if a product does not function properly or does not function as described in this document.

Note

This document was accurate at publication time. Go to EMC Online Support (https://support.emc.com) to ensure that you are using the latest version of this document.

Related documentation

The release notes for your version includes the latest information for your product.

The following EMC publication sets provide information about your ScaleIO or ScaleIO Ready Node product:

- ScaleIO software (downloadable as ScaleIO Software <version> Documentation set)
- ScaleIO Ready Node with AMS (downloadable as ScaleIO Ready Node with AMS Documentation set)
- ScaleIO Ready Node no AMS (downloadable as ScaleIO Ready Node no AMS Documentation set)
- VxRack Node 100 Series (downloadable as VxRack Node 100 Series Documentation set)

You can download the release notes, the document sets, and other related documentation from EMC Online Support.

Typographical conventions

EMC uses the following type style conventions in this document:

Bold	Used for names of interface elements, such as names of windows, dialog boxes, buttons, fields, tab names, key names, and menu paths (what the user specifically selects or clicks)
Italic	Used for full titles of publications referenced in text
`Monospace`	Used for: • System code • System output, such as an error message or script • Pathnames, filenames, prompts, and syntax • Commands and options
Monospace italic	Used for variables
`Monospace bold`	Used for user input
[ ]	Square brackets enclose optional values
|	Vertical bar indicates alternate selections - the bar means “or”

{ }	Braces enclose content that the user must specify, such as x or y or z
...	Ellipses indicate nonessential information omitted from the example

Where to get help

EMC support, product, and licensing information can be obtained as follows:

Product information

For documentation, release notes, software updates, or information about EMC products, go to EMC Online Support at https://support.emc.com.

Technical support

Go to EMC Online Support and click Service Center. You will see several options for contacting EMC Technical Support. Note that to open a service request, you must have a valid support agreement. Contact your EMC sales representative for details about obtaining a valid support agreement or with questions about your account.

Your comments

Your suggestions will help us continue to improve the accuracy, organization, and overall quality of the user publications. Send your opinions of this document to techpubcomments@emc.com.

Introduction

ScaleIO is a software-only solution that uses existing servers' local disks and LAN to create a virtual SAN that has all the benefits of external storage—but at a fraction of the cost and complexity.

ScaleIO utilizes the existing local internal storage and turns it into internal shared block storage. For many workloads, ScaleIO storage is comparable to, or better than, external shared block storage.

The ScaleIO Gateway, which runs on Apache Tomcat, hosts a number of ScaleIO features, including a REST gateway, Installation Manager, and SNMP trap sender. For high availability when two Apache Tomcat instances (Gateway servers) are configured, use Apache httpd.

Note

This document discusses procedures needed only in environments in which the REST API commands are used.

Overview of the high-availability solution

To prevent a single point of failure in the Apache httpd, use a failover httpd instance that is clustered using one of the many available clustering stacks—for example, Keepalived, which is a lightweight open source clustering stack for Linux.

A shared (virtual) IP address is used for both Apache httpd servers. Both of the Apache httpd servers have the same configuration, except that one is defined in the Keepalived configuration as the master, while the other is defined as the slave. If the master fails, the slave becomes the master, which means the users do not notice any loss of service.

The Gateway servers are configured the same way. According to the configuration in the Apache httpd, one of the Gateway servers (machine2) is the master and the other (machine1) is the slave. The mod_jk module in the Apache httpd monitors the Gateway servers and decides to which server to forward the request from the client.

The following figures illustrate the configuration and topology for this solution:

Figure 1 High availability Gateway cluster topology

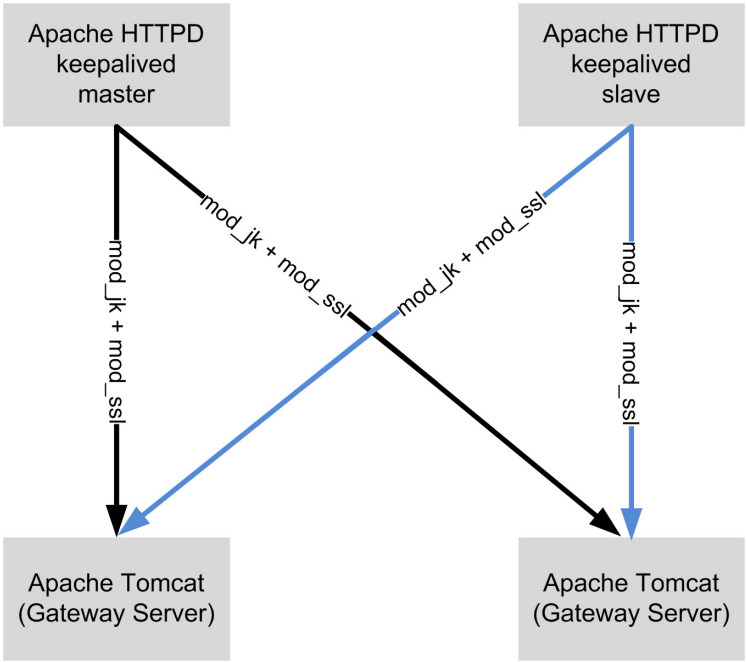
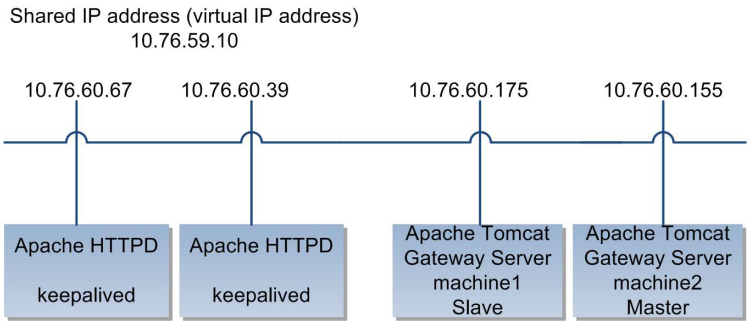


Figure 2 High availability networking example



Configuration workflow

1. Install Apache httpd.
2. Install mod_jk.
3. Install and enable mod_ssl.
4. Configure the Apache httpd virtual host to use the virtual IP address that is defined for both Apache httpd servers, instead of the default IP addresses.

5. Update the `server.xml` file on the Tomcat servers (ScaleIO Gateways) to listen on port 8009 + jvmRoute according to the configuration of the `workers.properties` file used by mod_jk.
6. Install and configure the clustering solution.

Example using Apache httpd and Keepalived on Ubuntu

The following is a full configuration example for a system running on Ubuntu 14.04 with Apache httpd 2.4.7., with the Keepalived open source tool used for the keep-alive component.

For more information about Keepalived, see http://www.keepalived.org. For more information about Ubuntu and the Apache HTTP Server, see their respective product documentation.

These procedures are necessary for enabling the solution:

Note

The following procedures involve scaleio-gateway service restarts.

- Task 1: Update server.xml on page 5
- Task 2: Create certificates on page 6
- Task 3: Configure the virtual IP address on page 8
- Task 4: Install and enable mod_sll on page 8
- Task 5: Configure Keepalived on page 9
- Task 6: Finish the configuration on page 10

Task 1: Update server.xml

Update the `server.xml` file on the Tomcat servers (scaleio-gateway) to listen on port 8009 + jvmRoute according to the configuration of the `workers.properties` file used by the mod_jk module:

Procedure

1. Edit the `server.xml` file, located at *<gateway installation directory>*`/conf/server.xml`:

   a. Uncomment and edit the following line:

   ```
   "<!-- <Connector port="8009" protocol="AJP/1.3"
   redirectPort="443" /> -->"
   ```

   After uncommenting and editing, it should look like this:

   ```
   "<Connector port="8009" protocol="AJP/1.3" redirectPort="$
   {ssl.port}" />"
   ```

   b. Add jvmRoute="machine1" to one gateway and jvmRoute="machine2" to the other.

   Make sure that the name you used in the `workers.properties` file is the same as the `jvmRoute` value:

SIO-GW #1:

```
<Engine name="Catalina" defaultHost="localhost"
jvmRoute="machine1">
```

SIO-GW #2:

```
<Engine name="Catalina" defaultHost="localhost"
jvmRoute="machine2">
```

c. If you intend to run Apache httpd on the same machine as a Gateway server (not recommended), ensure that the following is commented out to disabled the HTTP connector:

```
<Connector port="${http.port}" protocol="HTTP/1.1"
               connectionTimeout="20000"
               redirectPort="${ssl.port}"
               compression="force"
compressableMimeType="application/json,application/octet-
stre am" />
```

Note

To prevent port collision with the Apache default SSL port, change the default SSL port to something other than 443 at either Apache or Tomcat.

d. Restart the scaleio-gateway service on both Gateway servers:

2. Install Apache httpd:

```
- sudo apt-get install apache2
```

Task 2: Create certificates

Procedure

1. Create a working directory called `certs` at `etc/apache2/certs`.
2. Create `.pem` files:

```
"C:\Program Files\Java\jdk1.7.0_25\bin\keytool.exe" -genkey -
alias scaleio -keyalg RSA -validity 360 -keysize 2048 -dname
"CN=<Virtual ip>, OU=ASD, O=EMC, L=Hopkinton,
S=Massachusetts, C=US" -storepass <keystore password> -
keypass <keystore password> -keystore .keystore
```

a. Replace the values for `OU`, `O`, `L`, `S`, and `C` with your credentials.

b. Replace *<keystore password>* with the password you want to use. For example:

```
b2aa330a-e7c7-4e89-86c3-6f2565973cab
```

3. Run the following commands:

Note

A password is required.

```
keytool -importkeystore -storepass <keystore password> -
keypass <keystore password> -srckeystore .keystore -
destkeystore
intermediate.p12 -deststoretype PKCS12
```

```
openssl pkcs12 -in intermediate.p12 -nocerts -nodes -out
filename-key.pem
```

```
openssl pkcs12 -in intermediate.p12 -clcerts -nokeys -out
filename-cert.pem
```

4. Copy the `filename-key.pem` and `filename-cert.pem` files you created in the previous step to `/etc/apache2/certs`.
5. Install the mod_jk module:

```
apt-get install libapache2-mod-jk
```

6. Create a `workers.properties` file:

```
touch /etc/apache2/workers.properties
```

7. Add the following content to the `workers.properties` file:

```
worker.list=balance1
worker.machine1.type=ajp13
worker.machine1.host=<machine1_IP>
worker.machine1.port=8009
worker.machine1.lbfactor=1
worker.machine2.type=ajp13
worker.machine2.host=<machine2_IP>
worker.machine2.port=8009
worker.machine2.lbfactor=10
worker.balance1.type=lb
worker.balance1.balance_workers=machine1,machine2
```

Where:

- *<machine1_IP>* and *<machine2_IP>* are the IP addresses of the Gateway Servers.
- `worker.machine2` is the master gateway,
- `worker.machine1` is the slave gateway

8. Update the `/etc/apache2/mods-available/jk.conf` file:
   a. Change `JkWorkersFile` to `JkWorkersFile /etc/apache2/workers.properties`.

b. Add the following:

```
# Map a worker to a namespace.
JkMount /api/* balance1
JkOptions +ForwardSSLCertChain
```

c. Delete the following:

```
<Location /jk-status>
        # Inside Location we can omit the URL in JkMount
        JkMount jk-status
        Order deny,allow
        Deny from all
        Allow from 127.0.0.1
</Location>
<Location /jk-manager>
        # Inside Location we can omit the URL in JkMount
        JkMount jk-manager
        Order deny,allow
        Deny from all
        Allow from 127.0.0.1
</Location>
```

Task 3: Configure the virtual IP address

Configure the virtual IP address for the Gateway servers.

Procedure

1. Add the virtual IP address to the `/etc/hosts` file.

   Note

   Add only the hostname related to the virtual IP.

2. Edit the `/etc/apache2/sites-available/default-ssl.conf` file:

   a. Find this line in the file:

   ```
   Update <VirtualHost _default_:443>
   ```

   b. Replace `_default_` with the virtual IP address.

   c. Under the `VirtualHost _default` section, add the following lines:

   ```
   SSLCertificateFile /etc/apache2/certs/filename-cert.pem
   SSLCertificateKeyFile /etc/apache2/certs/filename-key.pem
   JkMountCopy On
   JkMount /* balance1
   ```

Task 4: Install and enable mod_sll

Procedure

1. Run the following commands:

```
a2enmod ssl
service apache2 restart
a2ensite default default-ssl
service apache2 reload
```

Task 5: Configure Keepalived

Install Keepalived on every server on which Apache httpd is installed.

Procedure

1. Install Keepalived:

```
apt-get install keepalived
```

2. Clear the example in the `keepalived.conf` file:

```
cat /dev/null > /etc/keepalived/keepalived.conf
```

3. Update the `/etc/keepalived/keepalived.conf` file:

   a. Edit `/etc/keepalived/keepalived.conf` on the Apache https master server:

```
vrrp_script chk_apache_httpd {
   script "killall -0 apache2"   # verify the pid is exist
or not
    interval 2                   # check every 2 seconds
    weight 2                     # add 2 points of prio if
OK
}

vrrp_instance VI_1 {
    interface eth0               # interface to monitor
    state MASTER
    virtual_router_id 51         # Assign one ID for this
route
    priority 101                 # 101 on master, 100 on
slave
    virtual_ipaddress {
         <virtual_ip>            # the virtual IP
    }
    track_script {
        chk_apache_httpd
    }
}
```

   b. Edit the `/etc/keepalived/keepalived.conf` file on the Apache https slave server:

```
vrrp_script chk_apache_httpd {
   script "killall -0 apache2"   # verify the pid is exist
```

```
or not
    interval 2                    # check every 2 seconds
    weight 2                      # add 2 points of prio if
OK
}

vrrp_instance VI_1 {
    interface eth0                # interface to monitor
    state MASTER
    virtual_router_id 51          # Assign one ID for this
route
    priority 101                  # 101 on master, 100 on
slave
    virtual_ipaddress {
         <virtual_ip>             # the virtual IP
    }
    track_script {
        chk_apache_httpd
    }
}
```

Task 6: Finish the configuration

Procedure

1. Add this line to the `/etc/sysctl.conf` file:

```
net.ipv4.ip_nonlocal_bind = 1
```

2. Apply the changes:

```
sysctl -p
```

Note

This instructs the kernel to bind the non-local IP to the Apache httpd service.

3. Create a symlink to the `/etc` directory:

```
/usr/sbin/keepalived -f /etc/keepalived/keepalived.conf
```

4. Stop, then start, Keepalived:

```
service keepalived stop
service keepalived start
```

Troubleshooting and getting help

EMC support, product, and licensing information can be obtained as follows:

- Product information — For documentation, release notes, software updates, or information about EMC products, go to EMC Online Support at: https://support.emc.com
- Technical support — Go to EMC Online Support and click **Service Center**. You will see several options for contacting EMC Technical Support. Note that to open a service request, a valid support agreement is required. Contact your EMC sales representative for details about obtaining a valid support agreement or with questions about your account.

Copyright © 2017 Dell Inc. or its subsidiaries. All rights reserved.

Published September 2017

Dell believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS-IS." DELL MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. USE, COPYING, AND DISTRIBUTION OF ANY DELL SOFTWARE DESCRIBED IN THIS PUBLICATION REQUIRES AN APPLICABLE SOFTWARE LICENSE.

Dell, EMC, and other trademarks are trademarks of Dell Inc. or its subsidiaries. Other trademarks may be the property of their respective owners. Published in the USA.