Red Hat Ceph Storage 4

Monitoring Ceph with Nagios Guide

Monitoring Ceph with Nagios Core.

Last Updated: 2021-09-14

Red Hat Ceph Storage 4 Monitoring Ceph with Nagios Guide

Monitoring Ceph with Nagios Core.

Legal Notice

Abstract

This document provides instructions for installing and configuring Nagios to monitor a Red Hat Ceph Storage cluster. Red Hat is committed to replacing problematic language in our code, documentation, and web properties. We are beginning with these four terms: master, slave, blacklist, and whitelist. Because of the enormity of this endeavor, these changes will be implemented gradually over several upcoming releases. For more details, see our CTO Chris Wright's message.

Table of Contents

CHAPTER 1. NAGIOS AND CEPH

Nagios Core is an open source solution for monitoring nodes. Large Red Hat Ceph Storage clusters benefit from distributed monitoring systems such as Nagios Core. The Nagios Core checks each node in a cluster, including the health of the underlying operating system, as well as the health of the Red Hat Ceph Storage cluster daemons.

To deploy Nagios Core with Ceph requires:

- A running Red Hat Ceph Storage cluster.

Instead of Nagios Core, you can also substitute the more feature rich commercial version, Nagios XI.

IMPORTANT

Red Hat does not provide the Nagios packages.

IMPORTANT

Red Hat works with our technology partners to provide this documentation as a service to our customers. However, Red Hat does not provide support for this product. If you need technical assistance for this product, then contact Nagios for support.

CHAPTER 2. NAGIOS CORE INSTALLATION AND CONFIGURATION

As a storage administrator, you can install Nagios Core by downloading the Nagios Core source code; then, configuring, making and installing it on the node that will run Nagios Core instance.

2.1. INSTALLING AND CONFIGURING THE NAGIOS CORE SERVER FROM SOURCE

There is not a Red Hat Enterprise Linux package for the Nagios Core software, so the Nagios Core software must be compiled from source.

Prerequisites

- Access to OpenSSL.
- Internet access.

Procedure

1. Install the prerequisites:

```
[user@nagios]# yum install -y httpd php php-cli gcc glibc glibc-common gd gd-devel net-snmp openssl openssl-devel wget unzip
```

2. Open port **80** for **httpd**:

```
[user@nagios]# firewall-cmd --zone=public --add-port=80/tcp
[user@nagios]# firewall-cmd --zone=public --add-port=80/tcp --permanent
```

3. Create a user and group for Nagios Core:

```
[user@nagios]# useradd nagios
[user@nagios]# passwd nagios
[user@nagios]# groupadd nagcmd
[user@nagios]# usermod -a -G nagcmd nagios
[user@nagios]# usermod -a -G nagcmd apache
```

4. Download the latest version of Nagios Core and Plug-ins:

```
[user@nagios]# wget --inet4-only https://assets.nagios.com/downloads/nagioscore/releases/nagios-4.3.1.tar.gz
[user@nagios]# wget --inet4-only http://www.nagios-plugins.org/download/nagios-plugins-2.2.1.tar.gz
[user@nagios]# tar zxf nagios-4.3.1.tar.gz
[user@nagios]# tar zxf nagios-plugins-2.2.1.tar.gz
[user@nagios]# cd nagios-4.3.1
```

5. Run **./configure**:

```
[user@nagios]# ./configure --with-command-group=nagcmd
```

6. Compile the Nagios Core source code:

```
[user@nagios]# make all
```

7. Install Nagios source code:

```
[user@nagios]# make install
[user@nagios]# make install-init
[user@nagios]# make install-config
[user@nagios]# make install-commandmode
[user@nagios]# make install-webconf
```

8. Copy the event handlers and change their ownership:

```
[user@nagios]# cp -R contrib/eventhandlers/ /usr/local/nagios/libexec/
[user@nagios]# chown -R nagios:nagios /usr/local/nagios/libexec/eventhandlers
```

9. Run the pre-flight check:

```
[user@nagios]# /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg
```

10. Make and install the Nagios Core plug-ins:

```
[user@nagios]# cd ../nagios-plugins-2.2.1
[user@nagios]# ./configure --with-nagios-user=nagios --with-nagios-group=nagios
[user@nagios]# make
[user@nagios]# make install
```

11. Create a user for the Nagios Core user interface:

```
[user@nagios]$ sudo htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
```

IMPORTANT

If adding a user other than **nagiosadmin**, ensure the **/usr/local/nagios/etc/cgi.cfg** file gets updated with the username too.

Also modify the **/usr/local/nagios/etc/objects/contacts.cfg** file with the user name, full name and email address as needed.

2.2. STARTING THE NAGIOS CORE SERVICE

Start the Nagios Core service to monitor the Red Hat Ceph Storage cluster health.

Prerequisites

- Root-level access to the Nagios Core service.

Procedure

1. Add Nagios Core as a service and enable it:

```
[user@nagios]# chkconfig --add nagios
[user@nagios]# chkconfig --level 35 nagios on
```

2. Start the Nagios Core daemon and Apache:

```
[user@nagios]# systemctl start nagios
[user@nagios]# systemctl enable httpd
[user@nagios]# systemctl start httpd
```

2.3. LOGGING INTO THE NAGIOS CORE SERVER

Log in to the Nagios Core server to view the health status of the Red Hat Ceph Storage cluster.

Prerequisites

- User name and password for the Nagios web interface.

Procedure

1. With Nagios up and running, log in to the web user interface:

```
http://IP_ADDRESS/nagios
```

   Nagios Core will prompt for a user name and password.

2. Input the login and password of the default Nagios Core user.

CHAPTER 3. NAGIOS REMOTE PLUG-IN EXECUTOR INSTALLATION

As a storage administrator, you can monitor the Ceph storage cluster nodes, install Nagios plug-ins, the Ceph plug-ins and the Nagios remote plug-in executor (NRPE) add-on to each of the Ceph nodes.

For demonstration purposes, this section adds NRPE to a Ceph Monitor node with the hostname **mon**. Repeat the remaining procedures on all Ceph nodes that Nagios should monitor.

3.1. INSTALLING AND CONFIGURING NAGIOS REMOTE PLUG-IN EXECUTOR

Install the Nagios Remote Plug-in Executor (NPRE) and configure it to communicate with the Nagios Core server.

Prerequisites

- Access to OpenSSL.
- User-level access to Ceph Monitor node.

Procedure

1. Install these packages on the node:

```
[user@mon]# yum install openssl openssl-devel gcc make git
```

2. NRPE installation requires a Nagios user. So create the user first:

```
[user@mon]# useradd nagios
[user@mon]# passwd nagios
```

3. Download the latest version of the Nagios plug-ins. Then, make and install them:

```
[user@mon]# wget http://www.nagios-plugins.org/download/nagios-plugins-2.2.1.tar.gz
[user@mon]# tar zxf nagios-plugins-2.2.1.tar.gz
[user@mon]# cd nagios-plugins-2.2.1
[user@mon]# ./configure
[user@mon]# make
[user@mon]# make install
```

4. NRPE uses **xinetd** for communication. Install it before installing the NRPE module:

```
[user@mon]# yum install xinetd
```

5. Download the latest verion of the Ceph plug-ins:

```
[user@mon]# cd ~
[user@mon]# git clone --recursive https://github.com/valerytschopp/ceph-nagios-plugins.git
[user@mon]# cd ceph-nagios-plugins
[user@mon]# make dist
[user@mon]# make install
```

6. Download, make and install Nagios NRPE:

```
[user@mon]# cd ~
[user@mon]# wget https://github.com/NagiosEnterprises/nrpe/releases/download/nrpe-3.1.0/nrpe-3.1.0.tar.gz
[user@mon]# tar xvfz nrpe-3.1.0.tar.gz
[user@mon]# cd nrpe-3.1.0
[user@mon]# ./configure
[user@mon]# make all
[user@mon]# make install-groups-users
[user@mon]# make install
[user@mon]# make install-config
[user@mon]# make install-init
```

7. Edit the the **/etc/services** file, and add the service string **nrpe 5666/tcp**:

8. Open port **5666** to allow communication with NRPE:

```
[user@mon]# firewall-cmd --zone=public --add-port=5666/tcp
[user@mon]# firewall-cmd --zone=public --add-port=5666/tcp --permanent
```

Additional Resources

- See https://github.com/valerytschopp/ceph-nagios-plugins for details.

3.2. STARTING THE NAGIOS REMOTE PLUG-IN EXECUTOR SERVICE

Start the Nagios Remote Plug-in Executor service to collect data and report it back to the Nagios Core server.

Prerequisites

- User-level access to the Ceph Monitor node

Procedure

1. Enable, restart, and reload **xinetd**:

```
[user@mon]# systemctl enable xinetd
[user@mon]# systemctl restart xinetd
[user@mon]# systemctl reload xinetd
```

2. Enable and start NRPE:

```
[user@mon]# systemctl enable nrpe
[user@mon]# systemctl start nrpe
```

3.3. CONFIGURING NAGIOS CORE SERVER ACCESS TO REMOTE NODES

In order for the Nagios Core server to access Nagios Remote Plugin Executor (NPRE) on a remote machine, the remote machine's xinetd and NRPE configurations must be updated with the IP address of the Nagios Core server.

Prerequisites

- User-level access to the Nagios Core server.
- Internet access.
- Access to the Nagios Remote Plugin Executor.

Procedure

1. Edit the xinetd configuration with the Nagios server's IP address:

```
[user@mon]# vi /etc/xinetd.d/nrpe
```

```
# default: off
# description: NRPE (Nagios Remote Plugin Executor)
service nrpe
{
    disable         = yes
    socket_type     = stream
    port            = 5666
    wait            = no
    user            = nagios
    group           = nagios
    server          = /usr/local/nagios/bin/nrpe
    server_args     = -c /usr/local/nagios/etc/nrpe.cfg --inetd
    only_from       = 127.0.0.1,IP_ADDRESS_OF_NAGIOS_CORE_SERVER
    log_on_success  =
}
```

2. After adding the IP address of the Nagios Core server to the **only_from** option, restart the **xinetd** service:

```
[user@mon]# systemctl restart xinetd
```

3. Edit the NRPE configuration with the Nagios server's IP address:

```
[user@mon]# vi /usr/local/nagios/etc/nrpe.cfg
```

```
allowed_hosts=127.0.0.1,IP_ADDRESS_OF_NAGIOS_CORE_SERVER
```

4. Add the IP address of the Nagios Core server to the **allowed_hosts** setting. Then, restart **nrpe**:

```
[user@mon]# systemctl restart nrpe
```

5. Test the installation:

```
[user@host]# /usr/local/nagios/libexec/check_nrpe -H localhost
```

The check should echo **NRPE v3.1.0-rc1** if it is working correctly.

CHAPTER 4. CONFIGURING THE REMOTE NODE ON THE NAGIOS CORE SERVER

Configure the Nagios Core server to be aware of the remote nodes.

Prerequisites

- User-level access to the remote node on the Nagios Core server.
- Internet access.

Procedure

1. Install the **check_nrpe** plug-in:

```
[user@nagios]# cd ~
[user@nagios]# wget https://github.com/NagiosEnterprises/nrpe/releases/download/nrpe-3.1.0/nrpe-3.1.0.tar.gz
[user@nagios]# tar xvfz nrpe-3.1.0.tar.gz
[user@nagios]# cd nrpe-3.1.0
[user@nagios]# ./configure
[user@nagios]# make check_nrpe
[user@nagios]# make install-plugin
```

2. Create a configuration for the remote host:

```
[user@nagios]# cd /usr/local/nagios/etc/objects
[user@nagios]# cp localhost.cfg mon.cfg
```

   Replace **localhost** with the hostname of the remote host, and the loopback IP address with the IP address of the remote host. Finally, delete or comment out the Host Group definition.

3. Change the file ownership to Nagios:

```
[user@nagios]# chown nagios:nagios mon.cfg
```

4. Add a **cfg_file=** reference to the **mon.cfg** file in **/usr/local/nagios/etc/nagios.cfg**:

```
[user@nagios]# vi /usr/local/nagios/etc/nagios.cfg
```

   Example

```
cfg_file=/usr/local/nagios/etc/objects/mon.cfg
```

5. Restart the Nagios server:

```
[user@nagios]# systemctl restart nagios
```

6. Ensure that the make and install procedures worked and that there is connectivity between the Nagios Core server and the remote host containing NRPE:

```
[user@nagios]# /usr/local/nagios/libexec/check_nrpe -H
IP_ADDRESS_OF_REMOTE_HOST
```

It should echo **NRPE v3.1.0-rc1** if it is working correctly.

CHAPTER 5. CONFIGURING THE NAGIOS PLUGINS FOR CEPH

Configure the Nagios plug-ins for Red Hat Ceph Storage cluster.

Prerequisites

- User-level access to the Ceph Monitor node.
- A running Red Hat Ceph Storage cluster.
- Access to the Nagios Core Server.

Procedure

1. Log in to the monitor server and create a Ceph key and keyring for Nagios.

```
[user@mon]# ssh mon
[user@mon]# cd /etc/ceph
[user@mon]# ceph auth get-or-create client.nagios mon 'allow r' > client.nagios.keyring
```

   Each plug-in will require authentication. Repeat this procedure for each node that contains a plug-in.

2. Add a command for the **check_ceph_health** plug-in:

```
[user@mon]# vi /usr/local/nagios/etc/nrpe.cfg
```

   #### Example

```
command[check_ceph_health]=/usr/lib/nagios/plugins/check_ceph_health --id nagios --
keyring /etc/ceph/client.nagios.keyring
```

3. Enable and restart the **nrpe** service:

```
[user@mon]# systemctl enable nrpe
[user@mon]# systemctl restart nrpe
```

   Repeat this procedure for each Ceph plug-in applicable to the node.

4. Return to the Nagios Core server and define a **check_nrpe** command for the NRPE plug-in:

```
[user@nagios]# cd /usr/local/nagios/etc/objects
[user@nagios]# vi commands.cfg
```

```
define command{
 command_name check_nrpe
 command_line USER1/check_nrpe -H HOSTADDRESS -c ARG1
}
```

5. On the Nagios Core server, edit the configuration file for the node and add a service for the Ceph plug-in.

   #### Example

```
[user@nagios]# vi /usr/local/nagios/etc/objects/mon.cfg
```

```
define service {
  use                 generic-service
  host_name           mon
  service_description   Ceph Health Check
  check_command         check_nrpe!check_ceph_health
}
```

NOTE

The **check_command** setting uses **check_nrpe!** before the Ceph plug-in name. This tells NRPE to execute the **check_ceph_health** command on the remote node.

6. Repeat this procedure for each plug-in applicable to the node.

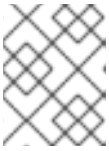
7. Restart the Nagios Core server:

```
[user@nagios]# systemctl restart nagios
```

8. Before proceeding with additional configuration, ensure that the plug-ins are working.

   Example

```
[user@mon]# /usr/lib/nagios/plugins/check_ceph_health --id nagios --keyring
/etc/ceph/client.nagios.keyring
```



NOTE

The **check_ceph_health** plug-in performs the equivalent of the **ceph health** command.

Additional Resources

- See the Ceph Nagios plugins web page for usage.