Red Hat Ceph Storage 3

Monitoring Ceph for Ubuntu with Nagios

Monitoring Ceph for Ubuntu with Nagios Core.

Last Updated: 2018-09-07

Red Hat Ceph Storage 3 Monitoring Ceph for Ubuntu with Nagios

Monitoring Ceph for Ubuntu with Nagios Core.

Legal Notice

Abstract

This document provides instructions for installing and configuring Nagios to monitor an Ubuntu-based Ceph Storage cluster.

Table of Contents

CHAPTER 1. INTRODUCTION

Nagios Core is an open source solution for monitoring hosts. Large Ceph Storage clusters benefit from distributed monitoring systems such as Nagios Core that check each host in a cluster, including the health of the underlying operating system, as well as the health of the Ceph Storage Cluster daemons.

To deploy Nagios Core with Ceph requires:

- A running Ceph cluster.
- A running Nagios core server.

In lieu of Nagios Core, you may also substitute the more feature rich commercial version, Nagios XI.

CHAPTER 2. INSTALLING NAGIOS CORE

Installing Nagios Core involves downloading the Nagios Core source code; then, configuring, making and installing it on the host that will run Nagios Core instance.

The following sections describe the process for Ubuntu 16.04 and later releases.

For Ubuntu 16.04, execute all commands as **root**:

```
[user@nagios]$ sudo -i
```

2.1. INSTALLING NAGIOS PREREQUISITES

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
```

```
[user@nagios]# apt-get install wget build-essential apache2 php apache2-
mod-php7.0 openssl libssl-dev gcc make php-gd libgd-dev unzip
```

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
```

Open port **80** for **apache2**:

```
[user@nagios]# iptables -A INPUT -p tcp --dport 80 -j ACCEPT
[user@nagios]# apt-get install iptables-persistent
```

2.2. CREATING A NAGIOS USER AND GROUP

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
```

Then, execute the following:

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

```
[user@nagios]# usermod -a -G nagios,nagcmd www-data
```

2.3. DOWNLOAD NAGIOS SOURCE CODE AND PLUG-INS

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
```

2.4. MAKE AND INSTALL NAGIOS CORE

To make and install Nagios Core, first run **`./configure`**.

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

```
[user@nagios]# ./configure --with-command-group=nagcmd --with-httpd-
conf=/etc/apache2/
```

After running **`./configure`**, compile the Nagios Core source code.

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

After making Nagios Core, install it.

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

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
```

Finally, run the pre-flight check.

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

2.5. MAKE AND INSTALL NAGIOS CORE PLUG-INS

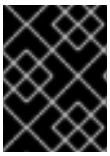
Make and install the Nagios Core plug-ins.

```
[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
```

2.6. CREATE A DEFAULT NAGIOS CORE USER

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.

2.7. START NAGIOS

Add Nagios Core as a service and enable it. Then start the Nagios Core daemon and Apache.

```
[user@nagios]# chkconfig --add nagios
[user@nagios]# chkconfig --level 35 nagios on
[user@nagios]# systemctl start nagios
[user@nagios]# systemctl enable httpd
[user@nagios]# systemctl start httpd
```

```
[user@nagios]# a2ensite nagios
[user@nagios]# a2enmod rewrite cgi
[user@nagios]# sudo cp /etc/init.d/skeleton /etc/init.d/nagios
```

Then, edit the initialization file:

```
[user@nagios]# vi /etc/init.d/nagios
```

Add the following lines:

```
DESC="Nagios"
NAME=nagios
DAEMON=/usr/local/nagios/bin/$NAME
DAEMON_ARGS="-d /usr/local/nagios/etc/nagios.cfg"
PIDFILE=/usr/local/nagios/var/$NAME.lock
```

Finally, start Nagios:

```
[user@nagios]# systemctl restart apache2
[user@nagios]# systemctl start nagios
```

2.8. LOG IN TO NAGIOS CORE

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

CHAPTER 3. INSTALLING NAGIOS REMOTE PLUG-IN EXECUTOR (NRPE)

To monitor Ceph Storage cluster hosts, install Nagios Plug-ins, the Ceph plug-ins and the NRPE add-on to each of the Ceph cluster's hosts.

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. INSTALL PREREQUISITES

NRPE requires OpenSSL. Install the following libraries first.

Execute the following:

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

```
[user@host]# apt install openssl libssl-dev gcc make git
```

3.2. CREATE A NAGIOS USER

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

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

3.3. DOWNLOAD, MAKE AND INSTALL THE NAGIOS PLUG-INS

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
```

3.4. DOWNLOAD, MAKE AND INSTALL THE NAGIOS CEPH PLUG-INS

Download the latest verion of the Ceph plug-ins. See https://github.com/valerytschopp/ceph-nagios-plugins for details.

```
[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
```

3.5. INSTALL XINETD

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

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

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

3.6. 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
```

Then, add **nrpe 5666/tcp** to the **/etc/services** file.

3.7. ENABLE, RESTART AND RELOAD XINETD.

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

3.8. ENABLE AND START NRPE

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

3.9. OPEN PORT 5666

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
```

```
[user@mon]# iptables -A INPUT -p tcp --dport 5666 -j ACCEPT
[user@mon]# apt-get install iptables-persistent
```

3.10. ADD THE NAGIOS CORE SERVER IP ADDRESS

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

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

```
[user@mon]# vim /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>
    log_on_success  =
}
```

Add the IP address of the Nagios Core server to the **only_from** setting. Then, restart **xinetd**.

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

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

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

```
allowed_hosts=127.0.0.1,<ip-address-of-nagios-core>
```

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

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

3.11. TEST THE INSTALLATION

Ensure that the make and install procedures worked.

```
[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. CONFIGURE THE NAGIOS CORE SERVER

After configuring NRPE on a Ceph host, configure the Nagios Core Server to recognize and monitor the host.

4.1. 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
```

4.2. CHECK TO ENSURE CONNECTIVITY

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.

4.3. 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.

Change the file ownership to nagios.

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

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

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

For example:

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

Then, restart the Nagios server.

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

CHAPTER 5. CONFIGURE CEPH PLUG-INS

There are some open source Ceph plug-ins provided at https://github.com/valerytschopp/ceph-nagios-plugins. They include:

- **check_ceph_df**: This plug-in outputs messages related to **ceph df** for the entire cluster or for individual pools. This plug-in only needs to run on Ceph monitor hosts. Multiple instances may be configured to monitor individual pools.
- **check_ceph_health**: This plug-in outputs the result of **ceph health**. This plug-in only needs to run on Ceph monitor hosts.
- **check_ceph_mon**: This plug-in checks a single monitor and returns **OK** if the monitor is up and running or **WARN** if it is down or missing. This plug-in only needs to run on Ceph monitor hosts.
- **check_ceph_osd**: This plug-in checks an OSD host or a single OSD and returns **OK** if the OSD is up and running or **WARN** if it is down. This plug-in only needs to run on Ceph OSD hosts.
- **check_ceph_rgw**: This plug-in checks a single Ceph Object Gateway and returns **OK** and the buckets and data usage if it is up and running or **WARN** if it is down or missing. This plug-in only needs to run on Ceph Object Gateway hosts.
- **check_ceph_mds**: This plug-in checks a single metadata server and returns **OK** if it is up and running, **WARN** if it is laggy and **Error** if it is down or missing. This plug-in only needs to run on Ceph metadata server hosts. These plug-ins get installed on the appropriate Ceph hosts. The following sections describe how to configure the **ceph health** plug-in on a monitor host.

5.1. CREATE KEYRING AND KEY

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 host that contains a plug-in.

5.2. TEST THE CEPH PLUG-IN INSTALLATION

Before proceeding with additional configuration, ensure that the plug-ins are working. For example:

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

The **check_ceph_health** plug-in performs the the equivalent of:

```
[user@mon]# ceph health
```

5.3. ADD A COMMAND FOR THE CEPH PLUG-IN

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

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

For example:

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

Save and restart NRPE.

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

Repeat this procedure for each Ceph plug-in applicable to the host. See https://github.com/valerytschopp/ceph-nagios-plugins for usage.

5.4. DEFINE THE CHECK_NRPE COMMAND

Return to the Nagios 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 $ARG1$
}
```

5.5. DEFINE A SERVICE FOR THE PLUG-IN

On the Nagios server, edit the configuration file for the host and add a service for the Ceph plug-in. For example:

```
[user@nagios]# vim /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 that 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 host.

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

Then, restart the Nagios server.

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

CHAPTER 6. SUMMARY

After completing the foregoing procedures, return to the Nagios web user interface and click on the "Hosts" link. The host should appear in the list of hosts. Click on the host to see additional details. Click on the **View Status Detail** hyperlink. It should display the checks it performs. In the instant example, there should be a **Ceph Health Check** service with status information on the Ceph cluster.