Integrating Nagios Mod-Gearman with Nagios XI

Purpose

This document describes how to integrate Nagios Mod-Gearman (NMG) with Nagios XI.

Overview

This document will provide instructions to:

- Install or Upgrade Nagios Mod-Gearman Server components on your Nagios XI server
- Create firewall rules for external workers (optional)
- Install or Upgrade Nagios Mod-Gearman Worker Components on external workers (optional)

Editing Files

In many steps of this documentation, you will be required to edit files. This documentation will use the vi text editor. When using the vi editor:

- To make changes press `i` on the keyboard first to enter insert mode
- Press **Esc** to exit insert mode
- When you have finished, save the changes in vi by typing `:wq` and press Enter

Nagios XI Version Notes and Remote Workers Web UI Option

If you are installing Nagios Mod-Gearman on Nagios XI, please make sure that you are running Nagios XI **2024R1.2+,** as Nagios Core version **>= 4.5.x** is required.

Also note that NMG is pre-loaded in Nagios XI **2024R2.1+**. You can easily configure and manage NMG from the web UI in the **Configure > Remote Workers** menu. This is a Premium feature, so is available as long as you have active Maintenance benefits.

Installing the Nagios Repository on Centos Stream / RHEL / Oracle Linux

In order to access the packages for Nagios Mod-Gearman , you will need to install the official Nagios repository. If you don't have this installed, please visit the Repository Home Page to view instructions for installing the repository on your distribution.

You will need to install the EPEL repository on the system if it is not already enabled.

https://www.redhat.com/en/blog/install-epel-linux

Server Installation – New Installation Centos Stream / RHEL / Oracle Linux instructions

These steps are performed on your Nagios XI server. Establish a terminal session to your Nagios XI server and execute the following commands to install Nagios Mod-Gearman and the gearmand server.

Centos Stream / RHEL / Oracle Linux instructions

To install gearmand and nagios-mod-gearman run the following as **root** :

```
dnf install gearmand nagios-mod-gearman
```

Notes About Server

You now have a Nagios Mod-Gearman server installed as an additional NEB (Nagios Event Broker) module (gearmand/gearman-job-server). Nagios XI will hand the execution of checks to Nagios-Mod-Gearman which will distribute it to the worker(s). Part of the installation is that a local worker was installed on the Nagios XI server and will by default execute checks. If your goal is to offload the checks from your Nagios XI server, you will need to stop the nagios-mod-gearman-worker service and disable it from being started at boot.

Run this to disable the nagios-mod-gearman-worker service as **root**.

```
systemctl disable nagios-mod-gearman-worker
```

To enable gearmand to start at boot and to start it, run this as **root**.

```
systemctl enable gearmand
systemctl start gearmand
```

To enable Nagios Core to use NMG, you need to add a broker line to the `nagios.cfg` file.

To add the broker line, edit the `/usr/local/nagios/etc/nagios.cfg` file and add the following line to it (note that this is a single long line, wrapped over two lines in this doc) :

```
broker_module=/usr/lib64/nagios-mod-gearman/nagios-mod-gearman.o config=/etc/nagios-mod-gearman/module.conf eventhandler=no
```

Restart nagios to load the broker.

```
systemctl restart nagios
```

Firewall Rules On Server

If you are going to use external workers, you will need to open the ports on the local firewall on the Nagios XI server to allow the workers to connect.

RHEL 8/9 | CentOS Stream 9 | Oracle Linux 8/9

```
firewall-cmd --zone=public --add-port=4730/tcp --permanent
firewall-cmd –reload
```

Installing the Nagios Repository Ubuntu / Debian instructions.

In order to access the packages for Nagios Mod Gearman , you will need to install the official Nagios repository. If you don't have this installed, please visit the Repository Home Page to view instructions for installing the repository on your distribution.

Server Installation – New Installation Ubuntu / Debian instructions.

These steps are performed on your Nagios XI server. Establish a terminal session to your Nagios XI server and execute the following commands to install Nagios Mod Gearman and the gearmand server.

Install `gearmand` and `nagios-mod-gearman` run the following as **root**:

```
apt-get install nagios-mod-gearman-module
apt-get install nagios-mod-gearman-tools
apt-get install gearman-job-server
```

Notes About Server

You now have a Nagios Mod-Gearman server installed as an additional NEB (Nagios Event Broker) module (gearmand/gearman-job-server). Nagios XI will hand the execution of checks to Nagios-Mod-Gearman which will distribute it to the worker(s). Part of the installation is that a local worker was installed on the Nagios XI server and will by default execute checks. If your goal is to offload the checks from your Nagios XI server, you will need to stop the `nagios-mod-gearman-worker` service and disable it from being started at boot if it is installed.

```
systemctl disable nagios-mod-gearman-worker
systemctl stop nagios-mod-gearman-worker
```

To enable `gearmand` to start at boot and to start it, run this as root.

```
systemctl enable gearman-job-server
systemctl start gearman-job-server
```

To enable Nagios Core to use NMG, you need to add a broker line to the `nagios.cfg` file.

To add the broker line, edit the `/usr/local/nagios/etc/nagios.cfg` file and add the following line to it (note that this is a single long line, wrapped over two lines in this doc) :

```
broker_module=/usr/lib64/nagios-mod-gearman/nagios-mod-gearman.o config=/etc/nagios-mod-gearman/module.conf eventhandler=no
```

Restart nagios to load the broker:

```
systemctl restart nagios
```

Firewall Rules On Server

If you are going to use external workers, you will need to open the ports on the local firewall on the Nagios XI server to allow the workers to connect.

Ubuntu | Debian

If the server is using `firewalld`, run these commands:

```
firewall-cmd --zone=public --add-port=4730/tcp --permanent
firewall-cmd –reload
```

If the server is using ufw (Uncomplicated Firewall), run these commands:

```
ufw allow 4370/tcp
```

Next up we'll discuss how to install the Nagios Mod-Gearman workers.

Worker Installation – New Installation

Installing the Nagios Repository

In order to access the packages for Nagios Mod-Gearman , you will need to install the official Nagios repository. If you don't have this installed, please visit the Repository Home Page to view instructions for installing the repository on your distribution.

You will need to enable `crb` (the CodeReady Linux Builder repository) and install the EPEL repository on the system if it is not already enabled.

https://www.redhat.com/en/blog/install-epel-linux

Installing Nagios Mod-Gearman

The following steps are completed on the remote worker system(s):

Centos Stream / RHEL / Oracle Linux

To install nagios-mod-gearman run the following as root:

```
dnf install nagios-mod-gearman
```

Ubuntu / Debian

To install nagios-mod-gearman run the following as root:

```
apt install nagios-mod-gearman-worker
```

To enable the Nagios Mod-Gearman Worker to start at boot and to start the worker, run the following as root (All operating systems).

```
systemctl enable nagios-mod-gearman-worker
systemctl start nagios-mod-gearman-worker
```

Important Note: the `nagios-mod-gearman-worker` service will not start if you have not set it to execute any host, service, or group checks in the `worker.conf` file. More information on these settings can be found in this guide:

Nagios Mod-Gearman Queues and Workers

Required Setting Changes on the Worker System

The `/etc/nagios-mod-gearman/worker.conf` file has to be changed to set the `Identifer` of the worker and to set the server option to point the Nagios Mod-Gearman server (Nagios XI server)

Modify the `/etc/nagios-mod-gearman/worker.conf` file

Search it for the identifier option

```
# Identifier, hostname will be used if undefined
identifier=hostname
```

Change it to a descriptive hostname name of the worker (Example below is using **worker1** as the name)

```
identifier=worker1
```

Also the server option has to be changed to the IP Address of the Nagios XI server.

```
# sets the addess of your gearman job server. Can be specified
# more than once to add more server.
server=localhost:4730
```

change it to this (**xxx.xxx.xxx.xxx** is the IP address of the Nagios XI server)

```
server=xxx.xxx.xxx.xxx:4370
```

Restart the worker to load the change

```
systemctl restart nagios-mod-gearman-worker
```

Notes About Worker

You now have a worker which can connect to the central Nagios Mod-Gearman installation on your Nagios XI server. You may deploy as many workers as needed though typically one worker system will give the desired results. Keep in mind you can also increase the amount of worker processes on this system by editing the worker configuration file by setting the `min-worker` and `max-worker` limits.

Another thing you will need to keep in mind is that if you use NRPE checks on certain hosts, their `nrpe.cfg` must have all of your worker server's IP addresses listed as allowed connections. This means if you are using the xinetd version you need to edit the `only_from` section, or if you are using the standard version you must edit the `allowed_hosts` section. Similarly if you are using NSClient++ you must add it in the `allowed_hosts` section as well to reflect this.

In addition to this you must have all of the plugins which your hosts use to run checks installed on your worker servers.

This guide explains plugins in Nagios XI:

https://assets.nagios.com/downloads/nagiosxi/docs/Managing-Plugins-in-Nagios-XI-2024.pdf

This guide explains how you can install Nagios Plugins on your worker(s):

Installing Nagios Plugins From Source

Notes About Queues

Understanding how queues work in Nagios-Mod-Gearman is important to getting the most out of Nagios-Mod-Gearman. The Nagios Mod-Gearman Queues and Workers document explains this in detail and is well worth reading.

Security Note

The shared key needs to be the same on both the workers and gearmand server. A generic key is used in the installation however it's recommended that you change it.

On your Nagios XI server edit the both the NEB and Worker configuration files to change the key:

```
vi /etc/nagios-mod-gearman/module.conf
```

On remote worker(s) edit the Worker Configuration file to change the key:

```
vi /etc/nagios-mod-gearman/worker.conf
```

The entry in both files is as follows:

```
key=should_be_changed
```

The key shown above is what allows Nagios Mod-Gearman to send and receive instructions, disabling this is not recommended as any set of instructions can then be injected into the daemon or workers. The key defaults to the value shown above but it is recommended that you change this to something more secure. The key must be at least 7 characters long but no longer than 32 characters.

Be sure to restart Nagios after making changes to the configuration files:

```
systemctl restart nagios
```

And the workers have to be restarted

```
systemctl restart  nagios-mod-gearman-worker
```

Service Start/Stop Order

When stopping or stopping Nagios or Nagios Mod-Gearman services the following order must be obeyed for proper functionality:

CentOS Stream / RHEL / OracleLinux service start order:

```
systemctl start gearmand
systemctl start nagios-mod-gearman-worker
systemctl start nagios
```

Service stop order:

```
systemctl stop nagios
systemctl stop nagios-mod-gearman-worker
systemctl start gearmand
```

Ubuntu / Debian service start order:

```
systemctl start gearman-job-server
systemctl start nagios-mod-gearman-worker
systemctl start nagios
```

Service stop order:

```
systemctl stop nagios
systemctl stop nagios-mod-gearman-worker
systemctl start gearman-job-server
```

Disable Worker

If you plan on using a remote worker server you may want to stop the local `nagios-mod_gearman_worker` service on your Nagios XI server, that way, your remote workers will take over processing. Execute the following commands in a terminal session on your Nagios XI Server:

```
systemctl stop nagios-mod-gearman-worker.service
systemctl disable nagios-mod-gearman-worker.service
```

Tips

You can view the jobs, processes, and workers by running the following on the Nagios XI server:

```
nagios-gearman-top
```

Or, you can run the following on a Nagios Mod-Gearman Worker:

```
nagios-gearman-top -H <hostname of Nagios XI Master Gearman server>[:port]
```

Finishing Up

This completes the documentation on Integrating Nagios Mod-Gearman with Nagios XI. If you have additional questions or other support-related questions, please visit us at our Nagios Support Forum, Nagios Knowledge Base, or Nagios Library:

Visit Nagios Support Forum Visit Nagios Knowledge Base Visit Nagios Library