



Purpose

The purpose of this document is to provide a guide to integrating Mod-Gearman with Nagios XI to introduce distributed monitoring in your environment. Mod-Gearman and its worker processes intercept checks made by Nagios and run them externally, process them, and send them back to the Nagios server. In this way Mod-Gearman has been shown to greatly reduce the load taken on by the Nagios server while still keeping its speed and efficiency. If you would like more information on Mod-Gearman please visit the following link:

http://labs.consol.de/nagios/mod-gearman/#_what_is_mod_gearman

Target Audience

This guide is directed towards Nagios XI administrators interested in integrating Mod-Gearman with their Nagios XI installation.

This guide has been tested on RHEL / CentOS versions 5 / 6 / 7.

Server Installation - New

These steps are performed on the Nagios XI server.

```
cd /tmp
wget http://assets.nagios.com/downloads/nagiosxi/scripts/ModGearmanInstall.sh
chmod +x ModGearmanInstall.sh
./ModGearmanInstall.sh --type=server
```

Server Installation - Upgrade

If you are already running a previous version of Mod-Gearman on your Nagios XI server, these steps will upgrade it to the latest version and maintain your existing configurations:

```
cd /tmp
cp /etc/mod_gearman/* /tmp/
wget http://assets.nagios.com/downloads/nagiosxi/scripts/ModGearmanInstall.sh
chmod +x ModGearmanInstall.sh
./ModGearmanInstall.sh --type=server --upgrade
```

Notes About Server

You now have a Mod-Gearman server installed as an additional Nagios broker module. Nagios will hand the execution of checks to 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 mod-gearman2-worker service and disable it from being started at boot.

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/CentOS 5.x/6.x:

```
iptables -I INPUT -p tcp --dport 4730 -j ACCEPT
iptables -I INPUT -p udp --dport 4730 -j ACCEPT
service iptables save
```

RHEL/CentOS 7.x

```
firewall-cmd --zone=public --add-port=4703/tcp --permanent
firewall-cmd --zone=public --add-port=4703/udp --permanent
firewall-cmd --reload
```

Worker Installation - New

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

```
cd /tmp
wget http://assets.nagios.com/downloads/nagiosxi/scripts/ModGearmanInstall.sh
chmod +x ModGearmanInstall.sh
./ModGearmanInstall.sh --type=worker
```

When you see “**What is the IP address of your Nagios master server?**”

Type your Master Nagios IP then press Enter.

XXX.XXX.XXX.XXX

Worker Installation - Upgrade

If you are already running a previous Mod-Gearman worker, these steps will upgrade it to the latest version and maintain your existing configurations:

```
cd /tmp
wget http://assets.nagios.com/downloads/nagiosxi/scripts/ModGearmanInstall.sh
chmod +x ModGearmanInstall.sh
./ModGearmanInstall.sh --type=worker --upgrade
```

Notes About Worker

You now have a worker which can connect to the central 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.

https://assets.nagios.com/downloads/nagiosxi/docs/Managing_Plugins_in_Nagios_XI.pdf

Notes About Queues

Understanding how queues work in Mod-Gearman is important to getting the most out of Mod-Gearman. The following KB article explains this in detail and is well worth reading:

[Documentation - Mod-Gearman Queues and Workers](#)

Security Note

Shared key or keyfile needs to be the same on both the workers and gearmand server:

Edit the both the NEB and Worker configuration to change the key on the gearman server “Nagios XI”:

```
vi /etc/mod_gearman2/module.conf
vi /etc/mod_gearman2/worker.conf
```

Then edit the Worker Configuration to change the key on the remote worker:

```
vi /etc/mod_gearman2/worker.conf
```

Shared key or keyfile:

```
key=should_be_changed
```

The key shown above is what allows 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.

An alternative to the key is the keyfile. If you do not want your key openly visible in a configuration you may supply a path to the keyfile from which the pass key will be read.

```
keyfile=/path/to/secret.file
```

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

```
service nagios restart
```

You may also start and stop the gearmand service and workers with the following commands on the Nagios XI server:

```
service gearmand "start/stop/status/restart/reload/condrestart/help"  
service mod-gearman2-worker "start/stop/status/restart"
```

Nagios and gearmand service start/stop order (must be obeyed for proper functionality)

Service start order:

```
service gearmand start
service mod-gearman2-worker start
service nagios start
```

Service stop order:

```
service nagios stop
service mod-gearman2-worker stop
service gearmand stop
```

Tips

If you plan on using a remote worker server you may want to stop the local `mod_gearman_worker` service on the master server, that way, your remote workers will take over processing.

You can view the jobs, processes, and workers by running the following on the Master server

```
gearman_top2
```

Plugins for monitoring the gearman process can be found at:

https://exchange.nagios.org/directory/Plugins/Clustering-and-High-2DAvailability/check_gearman/details

Finishing Up

This completes the documentation on how to integrate Mod-Gearman in Nagios XI.

If you have additional questions or other support related questions, please visit us at our Nagios Support Forums:

<https://support.nagios.com/forum>

The Nagios Support Knowledgebase is also a great support resource:

<https://support.nagios.com/kb>