# The LMA Infrastructure Alerting plugin for Fuel Documentation

*Release 0.9-0.9.0-1*

**Mirantis Inc.**

April 22, 2016

# CONTENTS

CHAPTER

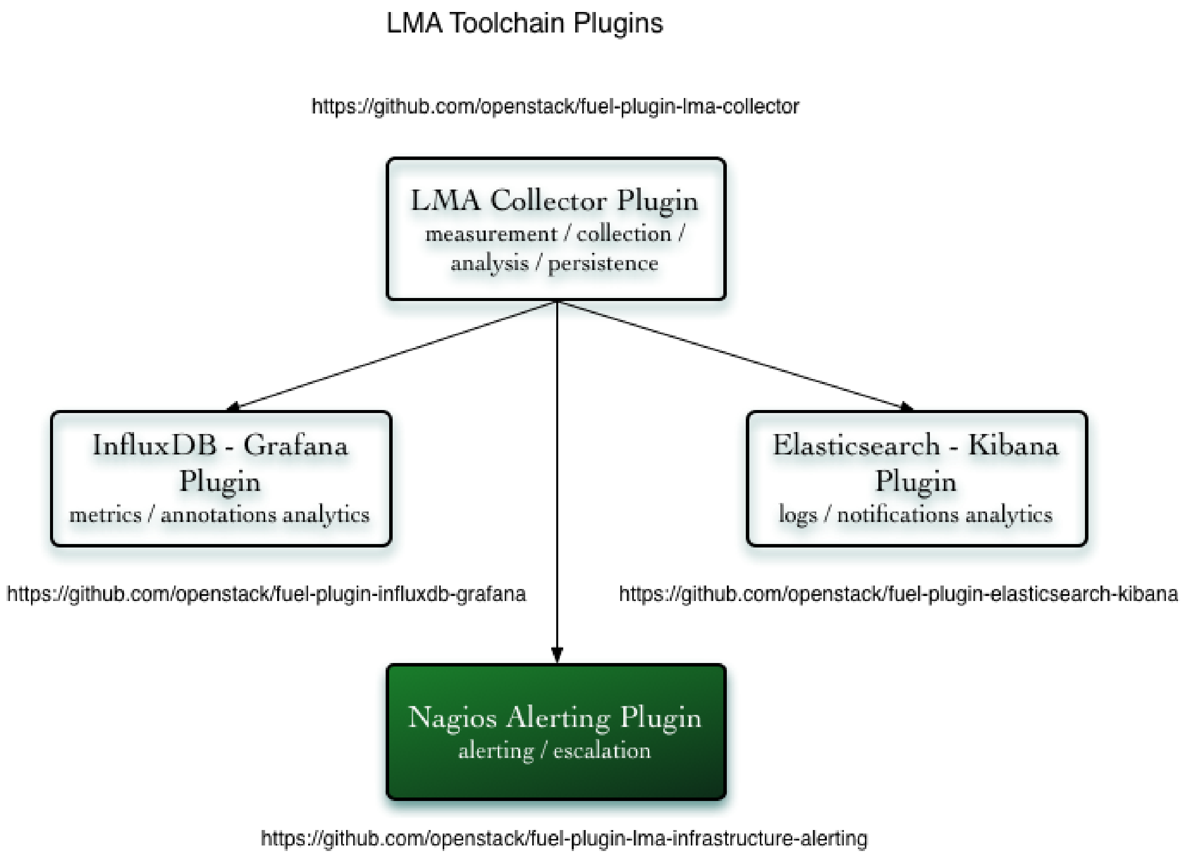
# ONE

# USER DOCUMENTATION

## 1.1 Overview

The **LMA Infrastructure Alerting Plugin** is used to install and configure Nagios™ which provides the alerting and escalation functionalities of the LMA Toolchain.

Nagios is a key component of the LMA Toolchain project as shown in the figure below.



LMA Toolchain Plugins
https://github.com/openstack/fuel-plugin-lma-collector
LMA Collector Plugin
measurement / collection / analysis / persistence
InfluxDB - Grafana Plugin
metrics / annotations analytics
Elasticsearch - Kibana Plugin
logs / notifications analytics
https://github.com/openstack/fuel-plugin-influxdb-grafana
https://github.com/openstack/fuel-plugin-elasticsearch-kibana
Nagios Alerting Plugin
alerting / escalation
https://github.com/openstack/fuel-plugin-lma-infrastructure-alerting

### 1.1.1 Requirements

| Requirement | Version/Comment |
|---|---|
| Disk space | The plugin's specification requires to provision at least 15GB of disk space for the system, 10GB for the logs and 20GB for Nagios™. As a result, the installation of the plugin will fail if there is less than 45GB of disk space available on the node. |
| Hardware configuration | The hardware configuration (RAM, CPU, disk) required by this plugin depends on the size of your cloud environment and other parameters like the retention period of the data.<br>A typical setup would at least require a quad-core server with 8GB of RAM and fast disks (ideally, SSDs). |
| Mirantis OpenStack | 8.0 |
| The LMA Collector Fuel Plugin | 0.9 |
| The LMA InfluxDB Grafana Fuel Plugin | 0.9 This is optional and only needed if you want to create alarms in Nagios™ for time-series stored in InfluxDB. |

### 1.1.2 Limitations

If Nagios is installed on several nodes for high availability, the alerts history will be lost in case of a server failover.

## 1.2 Release Notes

### 1.2.1 0.9.0

- Support Nagios clustering for high availability.
- Specify contact_groups for HTTP checks (#1559151).
- Specify contact_groups for SSH checks (#1559153).

### 1.2.2 0.8.0

- Initial release of the plugin.

## 1.3 Installation Guide

### 1.3.1 LMA Infrastructure Alerting Fuel Plugin install using the RPM file of the Fuel Plugins Catalog

To install the LMA Infrastructure Alerting Fuel Plugin using the RPM file of the Fuel Plugins Catalog, you need to follow these steps:

1. Download the RPM file from the Fuel Plugins Catalog.
2. Copy the RPM file to the Fuel Master node:

```
[root@home ~]# scp lma_infrastructure_alerting-0.9-0.9.0-0.noarch.rpm \
root@<Fuel Master node IP address>:
```

3. Install the plugin using the Fuel CLI:

```
[root@fuel ~]# fuel plugins --install \
lma_infrastructure_alerting-0.9-0.9.0-0.noarch.rpm
```

4. Verify that the plugin is installed correctly:

```
[root@fuel ~]# fuel plugins --list
id | name                          | version  | package_version
---|-------------------------------|----------|----------------
1  | lma_infrastructure_alerting | 0.9.0    | 4.0.0
```

### 1.3.2 LMA Infrastructure Alerting Fuel Plugin install from source

Alternatively, you may want to build the RPM file of the plugin from source if, for example, you want to test the latest features, modify some built-in configuration or implement your own customization. But note that running a Fuel plugin that you have built yourself is at your own risk.

To install the LMA Infrastructure Alerting Plugin from source, you first need to prepare an environment to build the RPM file. The recommended approach is to build the RPM file directly onto the Fuel Master node so that you won't have to copy that file later on.

**Prepare an environment for building the plugin on the Fuel Master Node**

1. Install the standard Linux development tools:

```
[root@home ~] yum install createrepo rpm rpm-build dpkg-devel
```

2. Install the Fuel Plugin Builder. To do that, you should first get pip:

```
[root@home ~] easy_install pip
```

3. Then install the Fuel Plugin Builder (the *fpb* command line) with *pip*:

```
[root@home ~] pip install fuel-plugin-builder
```

**Note**: You may also need to build the Fuel Plugin Builder if the package version of the plugin is higher than package version supported by the Fuel Plugin Builder you get from *pypi*. In this case, please refer to the section "Preparing an environment for plugin development" of the Fuel Plugins wiki if you need further instructions about how to build the Fuel Plugin Builder.

4. Clone the plugin git repository:

```
[root@home ~] git clone \
https://github.com/openstack/fuel-plugin-lma-infrastructure-alerting.git
```

5. Check that the plugin is valid:

```
[root@home ~] fpb --check ./fuel-plugin-lma-infrastructure-alerting
```

6. And finally, build the plugin:

```
[root@home ~] fpb --build ./fuel-plugin-lma-infrastructure-alerting
```

7. Now that you have created the RPM file, you can install the plugin using the *fuel plugins –install* command:

```
    [root@fuel ~] fuel plugins --install \
    ./fuel-plugin-lma-infrastructure-alerting/*.rpm
```

### 1.3.3 LMA Infrastructure Alerting Fuel Plugin software components

List of software components installed by the plugin

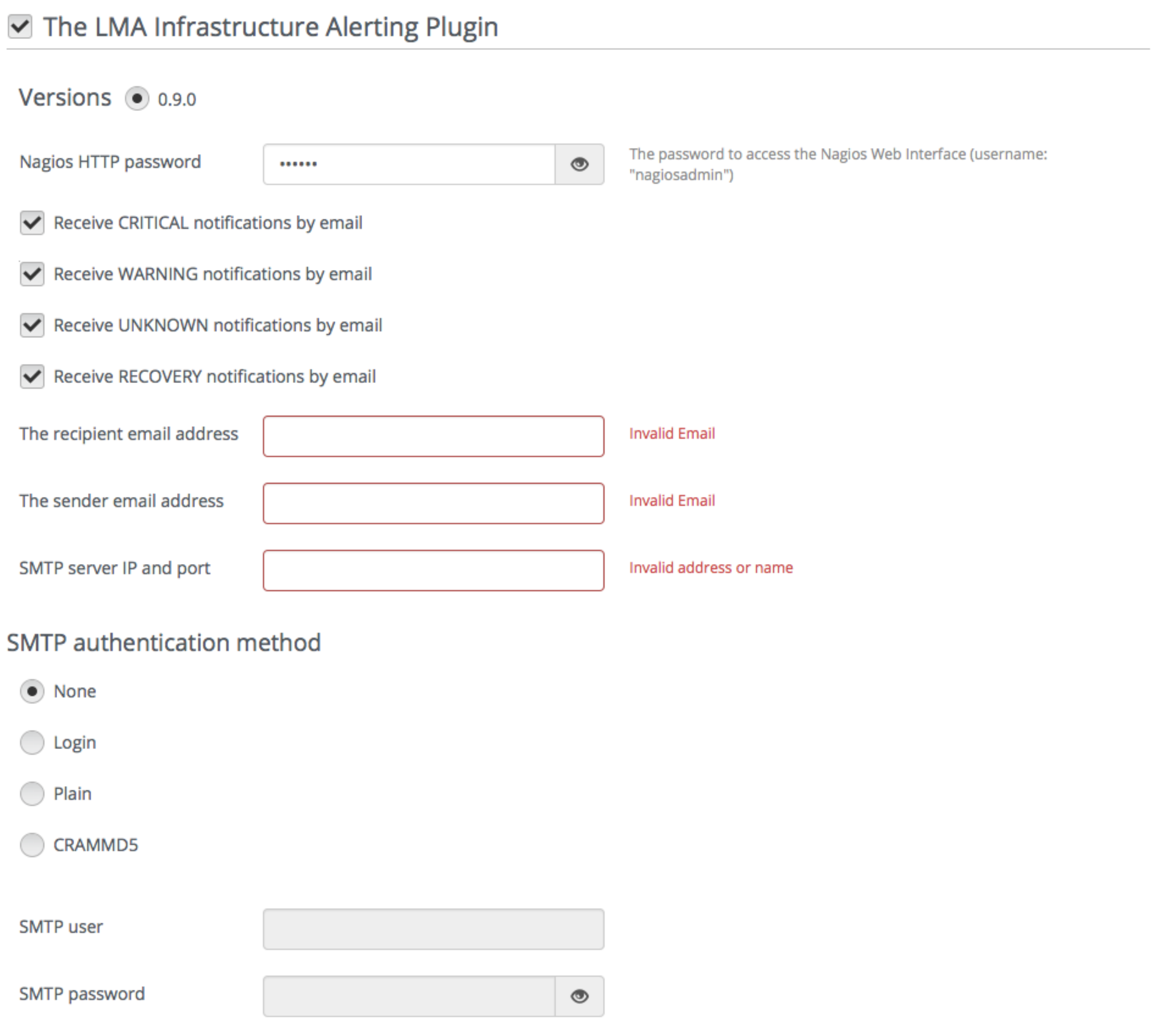
| Component | Version |
|---|---|
| Nagios | v3.5.1 for Ubuntu (64-bit) |
| Apache | Version coming with the Ubuntu distribution |

## 1.4 User Guide

### 1.4.1 Plugin configuration

To configure your plugin, you need to follow these steps:

1. Create a new environment with the Fuel web user interface.
2. Click the **Settings** tab and select the **Other** category.
3. Scroll down through the settings until you find the **LMA Infrastructure Alerting Plugin** section. You should see a page like this.



[✓] The LMA Infrastructure Alerting Plugin

Versions ● 0.9.0

Nagios HTTP password: •••••• — The password to access the Nagios Web Interface (username: "nagiosadmin")

[✓] Receive CRITICAL notifications by email

[✓] Receive WARNING notifications by email

[✓] Receive UNKNOWN notifications by email

[✓] Receive RECOVERY notifications by email

The recipient email address: — Invalid Email

The sender email address: — Invalid Email

SMTP server IP and port: — Invalid address or name

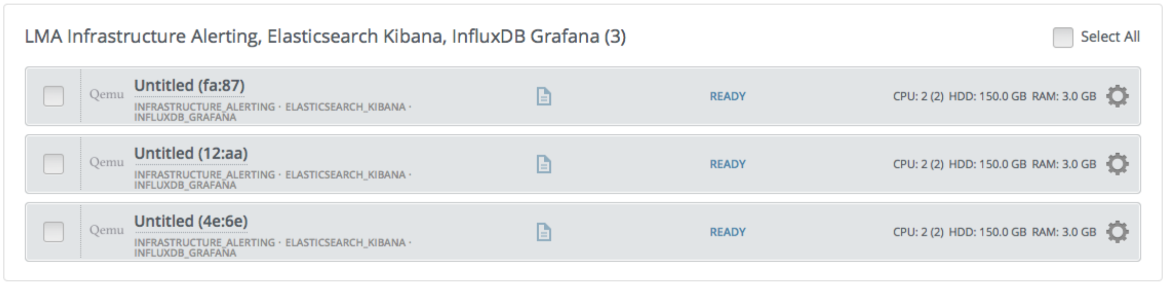
SMTP authentication method

[✓ None] Login Plain CRAMMD5

SMTP user:

SMTP password:

4. Check the *LMA Infrastructure Alerting Plugin* box and fill-in the required fields as indicated below.
   (a) Change the Nagios web interface password (recommended).
   (b) Check the boxes corresponding to the type of notification you would like to be alerted for by email (*CRITICAL*, *WARNING*, *UNKNOWN*, *RECOVERY*).
   (c) Specify the recipient email address for the alerts.
   (d) Specify the sender email address for the alerts.
   (e) Specify the SMTP server address and port.
   (f) Specify the SMTP authentication method.
   (g) Specify the SMTP username and password (required if the authentication method isn't *None*).
5. When you are done with the settings, scroll down to the bottom of the page and click the **Save Settings** button.
6. Click the *Nodes* tab and assign the *LMA Infrastructure Alerting* role to nodes as shown below. You can see in this example that the *Infrastructure_Alerting* role is assigned to three different nodes along with the *Elasticsearch_Kibana* role and the *InfluxDB_Grafana* role. This means that the three plugins of the LMA toolchain can be installed on the same nodes.



LMA Infrastructure Alerting, Elasticsearch Kibana, InfluxDB Grafana (3) Select All

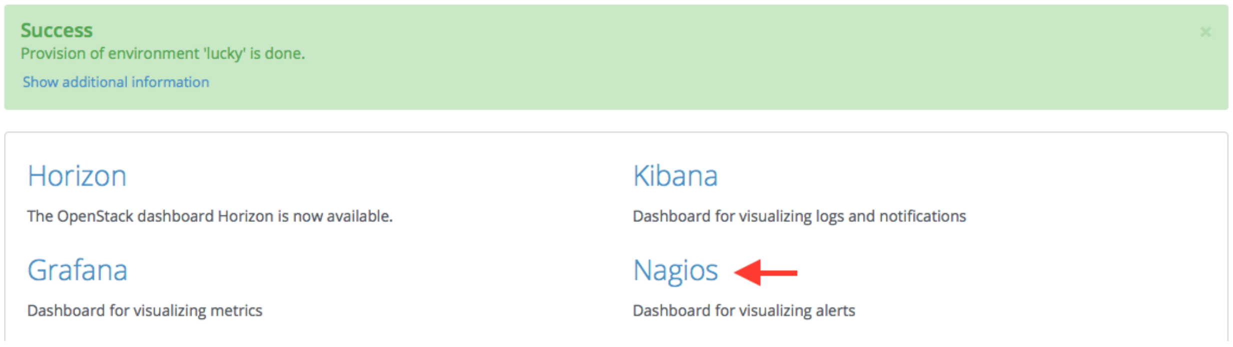
| | | | |
|---|---|---|---|
| Untitled (fa:87) INFRASTRUCTURE_ALERTING · ELASTICSEARCH_KIBANA · INFLUXDB_GRAFANA | READY | CPU: 2 (2) HDD: 150.0 GB RAM: 3.0 GB |
| Untitled (12:aa) INFRASTRUCTURE_ALERTING · ELASTICSEARCH_KIBANA · INFLUXDB_GRAFANA | READY | CPU: 2 (2) HDD: 150.0 GB RAM: 3.0 GB |
| Untitled (4e:6e) INFRASTRUCTURE_ALERTING · ELASTICSEARCH_KIBANA · INFLUXDB_GRAFANA | READY | CPU: 2 (2) HDD: 150.0 GB RAM: 3.0 GB |

**Note:** You can assign the *Infrastructure_Alerting* role up to three nodes. Nagios clustering for high availability requires that you assign the *Infrastructure_Alerting* role to at least three nodes. Note also that it is possible to add or remove a node with the *Infrastructure_Alerting* role after deployment.

7. Click on **Apply Changes**.
8. Adjust the disk configuration if necessary (see the Fuel User Guide for details). By default, the *LMA Infrastructure Alerting Plugin* allocates:
   - 20% of the first available disk for the operating system by honoring a range of 15GB minimum and 50GB maximum,
   - 10GB for */var/log*,
   - At least 20 GB for the Nagios data in */var/nagios*.
9. Configure your environment as needed.
10. Verify the networks on the Networks tab of the Fuel web UI.
11. And finally, Deploy your changes.

### 1.4.2 Plugin verification

Be aware, that depending on the number of nodes and deployment setup, deploying a Mirantis OpenStack environment can typically take anything from 30 minutes to several hours. But once your deployment is complete, you should see a deployment success notification message with a link to the Nagios dashboard as shown below.



Success
Provision of environment 'lucky' is done.
Show additional information

Horizon
The OpenStack dashboard Horizon is now available.

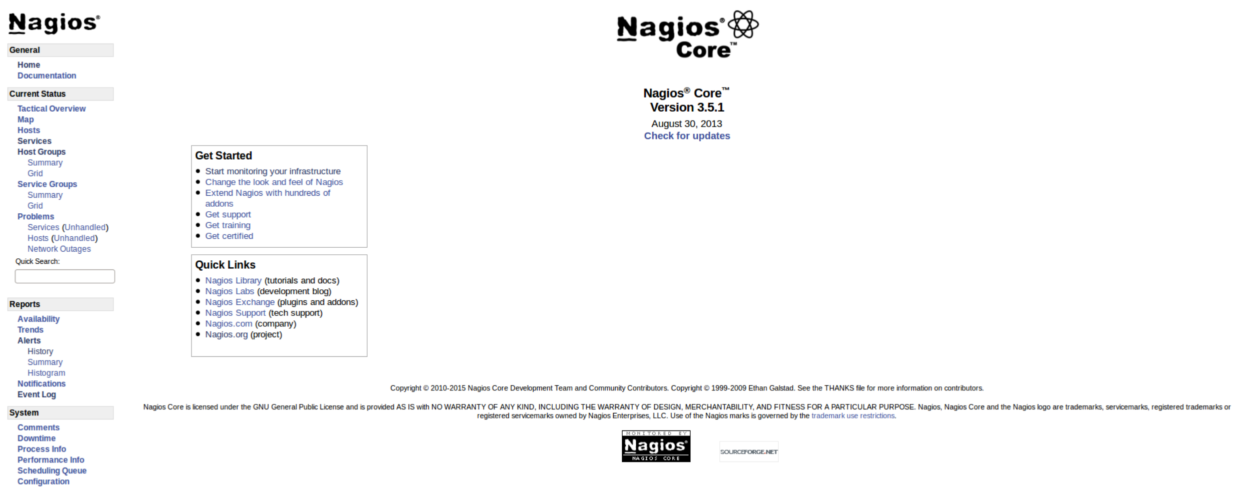
Kibana
Dashboard for visualizing logs and notifications

Grafana
Dashboard for visualizing metrics

Nagios
Dashboard for visualizing alerts

From the Fuel web UI **Dashboard** view, click on the **Nagios** link. Once you have authenticated (username is `nagiosadmin` and the password is defined in the settings of the plugin), you should be directed to the *Nagios Home Page* as shown below.

**Note:** Be aware that Nagios is attached to the *management network*. Your desktop machine must have access to the OpenStack environment's *management network* you just created to get access to the Nagios dashboard.



Nagios® Core™
Version 3.5.1
August 30, 2013
Check for updates

Get Started

- Start monitoring your infrastructure
- Change the look and feel of Nagios
- Extend Nagios with hundreds of addons
- Get support
- Get training
- Get certified

Quick Links

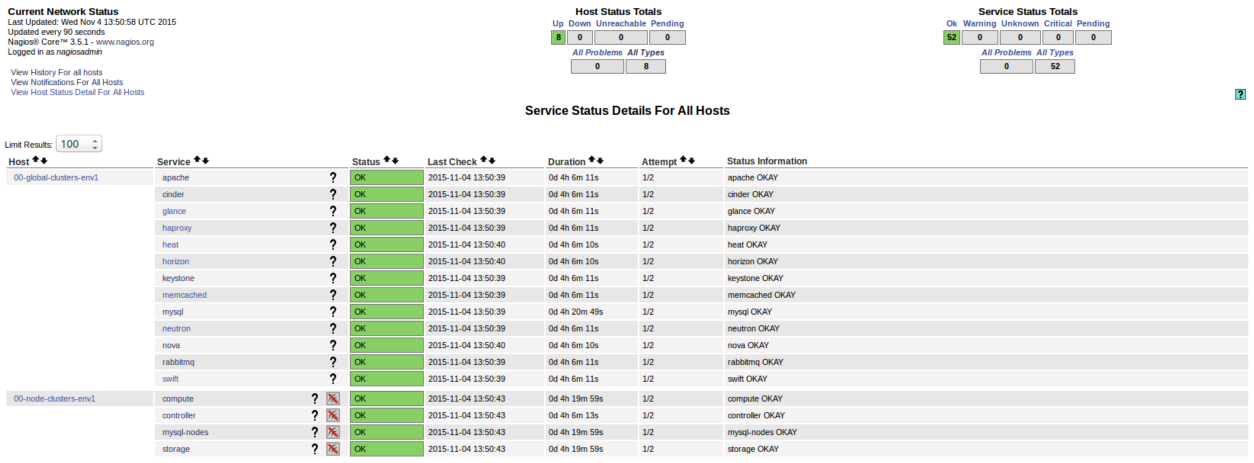
- Nagios Library (tutorials and docs)
- Nagios Labs (development blog)
- Nagios Exchange (plugins and addons)
- Nagios Support (tech support)
- Nagios.com (company)
- Nagios.org (project)

Copyright © 2010-2015 Nagios Core Development Team and Community Contributors. Copyright © 1999-2009 Ethan Galstad. See the THANKS file for more information on contributors.

Nagios Core is licensed under the GNU General Public License and is provided AS IS with NO WARRANTY OF ANY KIND, INCLUDING THE WARRANTY OF DESIGN, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. Nagios, Nagios Core and the Nagios logo are trademarks, servicemarks, registered trademarks or registered servicemarks owned by Nagios Enterprises, LLC. Use of the Nagios marks is governed by the trademark use restrictions.

## 1.4.3 Managing Nagios

You can get the current status of the OpenStack environment by clicking on the *Services* menu item as shown below.


Current Network Status
Last Updated: Wed Nov 4 13:50:58 UTC 2015
Updated every 90 seconds
Nagios® Core™ 3.5.1 - www.nagios.org
Logged in as *nagiosadmin*

View History For all hosts
View Notifications For All Hosts
View Host Status Detail For All Hosts

Host Status Totals

| Up | Down | Unreachable | Pending |
|---|---|---|---|
| 8 | 0 | 0 | 0 |

| All Problems | All Types |
|---|---|
| 0 | 8 |

Service Status Totals

| Ok | Warning | Unknown | Critical | Pending |
|---|---|---|---|---|
| 52 | 0 | 0 | 0 | 0 |

| All Problems | All Types |
|---|---|
| 0 | 52 |

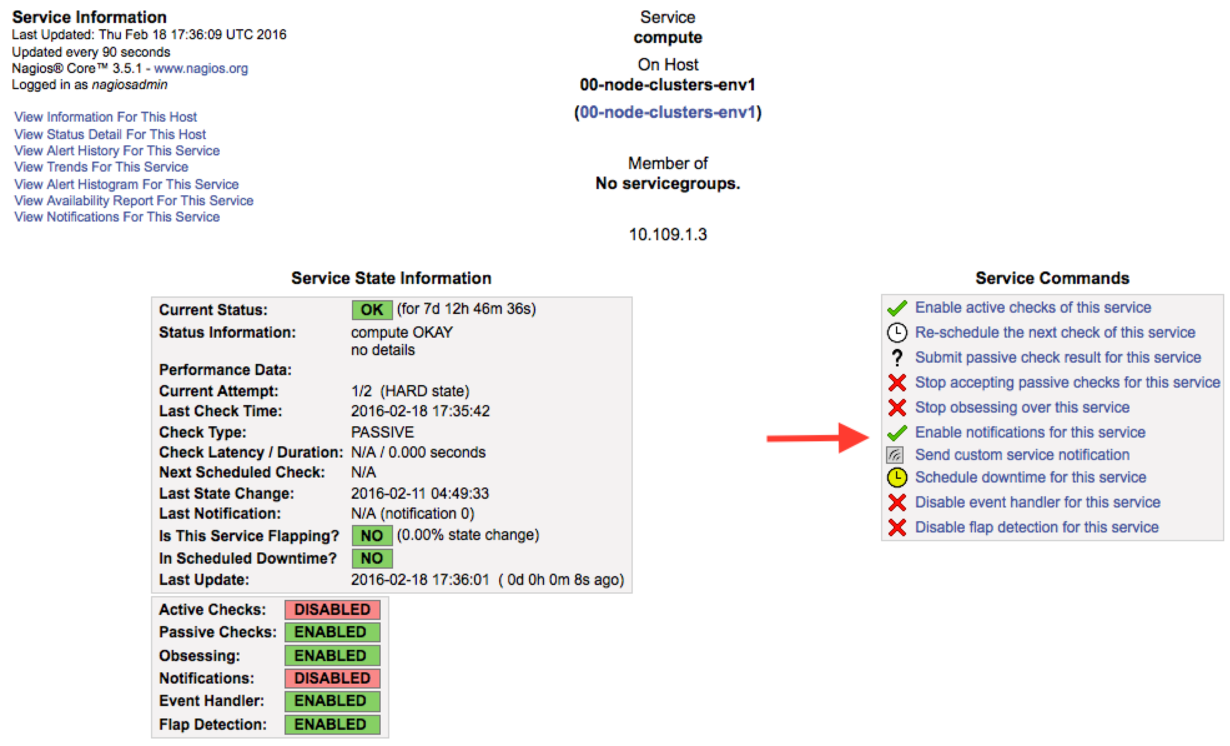
Service Status Details For All Hosts

Limit Results: 100

| Host | Service | Status | Last Check | Duration | Attempt | Status Information |
|---|---|---|---|---|---|---|
| 00-global-clusters-env1 | apache | OK | 2015-11-04 13:50:39 | 0d 4h 6m 11s | 1/2 | apache OKAY |
| | cinder | OK | 2015-11-04 13:50:39 | 0d 4h 6m 11s | 1/2 | cinder OKAY |
| | glance | OK | 2015-11-04 13:50:39 | 0d 4h 6m 11s | 1/2 | glance OKAY |
| | haproxy | OK | 2015-11-04 13:50:39 | 0d 4h 6m 11s | 1/2 | haproxy OKAY |
| | heat | OK | 2015-11-04 13:50:40 | 0d 4h 6m 10s | 1/2 | heat OKAY |
| | horizon | OK | 2015-11-04 13:50:40 | 0d 4h 6m 10s | 1/2 | horizon OKAY |
| | keystone | OK | 2015-11-04 13:50:39 | 0d 4h 6m 11s | 1/2 | keystone OKAY |
| | memcached | OK | 2015-11-04 13:50:39 | 0d 4h 6m 11s | 1/2 | memcached OKAY |
| | mysql | OK | 2015-11-04 13:50:39 | 0d 4h 20m 49s | 1/2 | mysql OKAY |
| | neutron | OK | 2015-11-04 13:50:39 | 0d 4h 6m 11s | 1/2 | neutron OKAY |
| | nova | OK | 2015-11-04 13:50:40 | 0d 4h 6m 10s | 1/2 | nova OKAY |
| | rabbitmq | OK | 2015-11-04 13:50:39 | 0d 4h 6m 11s | 1/2 | rabbitmq OKAY |
| | swift | OK | 2015-11-04 13:50:39 | 0d 4h 6m 11s | 1/2 | swift OKAY |
| 00-node-clusters-env1 | compute | OK | 2015-11-04 13:50:43 | 0d 4h 19m 59s | 1/2 | compute OKAY |
| | controller | OK | 2015-11-04 13:50:43 | 0d 4h 6m 13s | 1/2 | controller OKAY |
| | mysql-nodes | OK | 2015-11-04 13:50:43 | 0d 4h 19m 59s | 1/2 | mysql-nodes OKAY |
| | storage | OK | 2015-11-04 13:50:43 | 0d 4h 19m 59s | 1/2 | storage OKAY |

The *LMA Infrastructure Alerting Plugin* configures Nagios for all the hosts and services that have been deployed in the environment. The alarms (or service checks in Nagios terms) are created in **passive mode** as they are received from the *LMA Collector* and *Aggregator* (see the LMA Collector documentation for more details).

**Note:** The alert notifications for the nodes and clusters of nodes are disabled by default to avoid the alert fatigue and because they are not necessarily indicative of a condition affecting the overall health state of an OpenStack service cluster. If you nonetheless want to enable those alerts, go to the service details page and click on the *Enable* *notifications for this service* link within the *Service Commands* panel as shown below.



**Service Information**
Last Updated: Thu Feb 18 17:36:09 UTC 2016
Updated every 90 seconds
Nagios® Core™ 3.5.1 - www.nagios.org
Logged in as *nagiosadmin*

View Information For This Host
View Status Detail For This Host
View Alert History For This Service
View Trends For This Service
View Alert Histogram For This Service
View Availability Report For This Service
View Notifications For This Service

Service
**compute**
On Host
**00-node-clusters-env1**
**(00-node-clusters-env1)**

Member of
**No servicegroups.**

10.109.1.3

**Service State Information**

| | |
|---|---|
| **Current Status:** | OK (for 7d 12h 46m 36s) |
| **Status Information:** | compute OKAY<br>no details |
| **Performance Data:** | |
| **Current Attempt:** | 1/2 (HARD state) |
| **Last Check Time:** | 2016-02-18 17:35:42 |
| **Check Type:** | PASSIVE |
| **Check Latency / Duration:** | N/A / 0.000 seconds |
| **Next Scheduled Check:** | N/A |
| **Last State Change:** | 2016-02-11 04:49:33 |
| **Last Notification:** | N/A (notification 0) |
| **Is This Service Flapping?** | NO (0.00% state change) |
| **In Scheduled Downtime?** | NO |
| **Last Update:** | 2016-02-18 17:36:01 ( 0d 0h 0m 8s ago) |

| | |
|---|---|
| **Active Checks:** | DISABLED |
| **Passive Checks:** | ENABLED |
| **Obsessing:** | ENABLED |
| **Notifications:** | DISABLED |
| **Event Handler:** | ENABLED |
| **Flap Detection:** | ENABLED |

**Service Commands**

✓ Enable active checks of this service
Re-schedule the next check of this service
? Submit passive check result for this service
✗ Stop accepting passive checks for this service
✗ Stop obsessing over this service
✓ Enable notifications for this service
Send custom service notification
Schedule downtime for this service
✗ Disable event handler for this service
✗ Disable flap detection for this service

There are also two *Virtual Hosts* representing the health state of the *service clusters* and *node clusters*:

- *00-global-clusters-env${ENVID}* for the service clusters like the Nova cluster, the Keystone cluster, the RabbiMQ cluster and so on.
- *00-node-clusters-env${ENVID}* for the physical node clusters like the cluster of controller nodes, the cluster of storage nodes and so on.

These *Virtual Hosts* entities offer a high-level health state view for those clusters in the OpenStack environment.

## 1.4.4 Configuring service checks on InfluxDB metrics

You can configure additional alarms (other than those already defined in the *LMA Collector*) based on the metrics stored in the InfluxDB database. You can, for example, define an alert to be notified when the CPU activity for a particular process crosses a particular threshold. Say for example, you would like to set a 'warning' alarm at 30% of system CPU usage and a 'criticial' alarm at 50% system CPU usage for the Elasticsearch process. The steps to define those alarms in Nagios would be as follow:

1. Connect to the *LMA Infrastructure Alerting* node.
2. Install the Nagios plugin for querying InfluxDB:

```
[root@node-13 ~]# pip install influx-nagios-plugin
```

3. Define the command and the service check in the `/etc/nagios3/conf.d/influxdb_services.conf` file:

```
# Replace <INFLUXDB_HOST>, <INFLUXDB_USER> and <INFLUXDB_PASSWORD> by
# the appropriate values for your deployment
define command {
  command_line /usr/local/bin/check_influx \
      -h <INFLUXDB_HOST> -u <INFLUXDB_USER> -p <INFLUXDB_PASSWORD> -d lma \
      -q "select max(value) from lma_components_cputime_syst \
      where time > now() - 5m and service='$ARG1$' \
      group by time(5m) limit 1" \
      -w $ARG2$ -c $ARG3$
  command_name check_cpu_metric
}

define service {
  service_description Elasticsearch system CPU
  host                node-13
  check_command       check_cpu_metric!elasticsearch!30!50:
  use                 generic-service
}
```

4. Verify that the Nagios configuration is valid:

```
  [root@node-13 ~]# nagios3 -v /etc/nagios3/nagios.cfg

     [snip]

  Total Warnings: 0
  Total Errors:   0
```

   Here, things look okay. No serious problems were detected during the pre-flight check.

5. Restart the Nagios server,:

```
  [root@node-13 ~]# /etc/init.d/nagios3 restart
```

6. Go the Nagios dashboard and verify that the service check has been added.

From there, you can define additional service checks for different hosts or host groups using the same `check_influx` command. You will just need to provide these three required arguments for defining new service checks:

- A valid InfluxDB query that should return only one row with a single value. Check the InfluxDB documentation to learn how to use the InfluxDB's query language.
- A range specification for the warning threshold.
- A range specification for the critical threshold.

**Note:** Threshold ranges are defined following the Nagios format.

### 1.4.5 Using an external SMTP server with STARTTLS

If your SMTP server requires STARTTLS, you need to make some manual adjustements to the Nagios configuration after the deployment of your environment.

**Note:** Prior to enabling STARTTLS, you need to configure the *SMTP Authentication method* parameter in the plugin's settings to use either *Plain*, *Login* or *CRAM-MD5*.

1. Login to the *LMA Infrastructure Alerting* node.

2. Edit the `/etc/nagios3/conf.d/cmd_notify-service-by-smtp-with-long-service-output.cfg` file to add the `-S smtp-use-starttls` option to the *mail* command. For example:

```
define command{
  command_name    notify-service-by-smtp-with-long-service-output
  command_line    /usr/bin/printf "%b" "***** Nagios *****\n\n"\
    "Notification Type: $NOTIFICATIONTYPE$\n\n"\
    "Service: $SERVICEDESC$\nHost: $HOSTALIAS$\nAddress: $HOSTADDRESS$\n"\
    "State: $SERVICESTATE$\n\nDate/Time: $LONGDATETIME$\n\n"\
    "Additional Info:\n\n$SERVICEOUTPUT$\n$LONGSERVICEOUTPUT$\n" | \
    /usr/bin/mail -s "** $NOTIFICATIONTYPE$ "\
    "Service Alert: $HOSTALIAS$/$SERVICEDESC$ is $SERVICESTATE$ **" \
    -r 'nagios@localhost' \
    -S smtp="smtp://<SMTP_HOST>" \
    -S smtp-auth=<SMTP_AUTH_METHOD> \
    -S smtp-auth-user='<SMTP_USER>' \
    -S smtp-auth-password='<SMTP_PASSWORD>' \
    -S smtp-use-starttls \
    $CONTACTEMAIL$
}
```

**Note:** If the server certificate isn't present in the standard directory (eg `/etc/ssl/certs` on Ubuntu), you can specify its location by adding the `-S ssl-ca-file=<FILE>` option.

If you want to disable the verification of the SSL/TLS server certificate altogether, you should add the `-S ssl-verify=ignore` option instead.

3. Verify that the Nagios configuration is correct:

```
[root@node-13 ~]# nagios3 -v /etc/nagios3/nagios.cfg
```

4. Restart the Nagios service:

```
[root@node-13 ~]# /etc/init.d/nagios3 restart
```

## 1.4.6 Troubleshooting

If you cannot access the Nagios UI, follow these troubleshooting tips.

1. Check that the *LMA Collector* nodes are able to connect to the Nagios VIP address on port *8001*.
2. Check that the Nagios configuration is valid:

```
[root@node-13 ~]# nagios3 -v /etc/nagios3/nagios.cfg

   [snip]

Total Warnings: 0
Total Errors:   0
```

Here, things look okay. No serious problems were detected during the pre-flight check.

1. Check that the Nagios server is up and running:

```
[root@node-13 ~]# /etc/init.d/nagios3 status
```

2. If Nagios is down, restart it:

```
[root@node-13 ~]# /etc/init.d/nagios3 start
```

3. Check if Apache is up and running:

```
[root@node-13 ~]# /etc/init.d/apache2 status
```

4. If Apache is down, restart it:

```
[root@node-13 ~]# /etc/init.d/apache2 start
```

5. Look for errors in the Nagios log file (located at /var/nagios/nagios.log).
6. Look for errors in the Apache log file (located at /var/log/apache2/nagios_error.log).

Finally, Nagios may report a host or service state as *UNKNOWN*. Two cases can be distinguished:

- 'UNKNOWN: No datapoint have been received ever',
- 'UNKNOWN: No datapoint have been received over the last X seconds'.

Both cases indicate that Nagios doesn't receive regular passive checks from the *LMA Collector*. This may be due to different problems:

- The 'hekad' process of the *LMA Collector* fails to communicate with Nagios,
- The 'collectd' and/or 'hekad' process of the *LMA Collector* has crashed,
- One or several alarm rules are misconfigured.

To remedy to the above situations, follow the troubleshooting tips of the *LMA Collector Plugin User Guide*.

## 1.5 Licenses

### 1.5.1 Third Party Components

| Name | Project Web Site | License |
|---|---|---|
| Nagios | https://www.nagios.org/ | GPLv2 |
| Apache HTTP server | http://httpd.apache.org | Apache v2 |

### 1.5.2 Puppet modules

| Name | Project Web Site | License |
|---|---|---|
| puppetlabs-apache | https://github.com/puppetlabs/puppetlabs-apache | Apache v2 |
| puppetlabs-concat | https://github.com/puppetlabs/puppetlabs-concat | Apache v2 |
| puppetlabs-stdlib | https://github.com/puppetlabs/puppetlabs-stdlib | Apache v2 |
| leinaddm-htpasswd | https://github.com/leinaddm/puppet-htpasswd | Apache v2 |

## 1.6 Appendix

- The LMA Infrastructure Alerting plugin project at GitHub.
- The LMA Collector plugin project at GitHub.
- The InfluxDB-Grafana plugin project at GitHub.
- The official Nagios documentation.

CHAPTER

TWO

# INDICES AND TABLES

- search