Purpose

This document describes how to configure Nagios XI to run global event handler scripts in order to process host and service state changes, alerts, and notification information. Global event handlers allow admins to run custom-developed scripts with Nagios XI in order to create custom automation processes.

Target Audience

This document is intended for use by Nagios XI Administrators who have a good understanding of scripts and want to automate their monitoring environment based on host or service state changes.

Component Installation

If you do not have the Global Event Handler Component installed, you will need to download it and install it. If you are running Nagios XI 2012 or later, these components should already exist in your installation and you can skip this step. If you are running an earlier version of Nagios XI, you will need to download the Global Event Handler component. The component you need to install can be downloaded from the following URL:

https://assets.nagios.com/downloads/nagiosxi/components/globaleventhandler.zip

Once downloaded, the `globaleventhandler.zip` file can be uploaded and installed via **Admin** > **System Extensions** > **Manage Components**.

Once you have uploaded the component zip file, the component should be visible in the components list.

Configuring Global Event Handlers

Click on the component settings icon from the **Manage Components** page to access the component configuration settings.

The global event handler component configuration screen allows you to define specific commands that should be executed in sequence when host and service state changes and notification occur. Sample commands are displayed for reference.

In order to use your own handler commands, you must make sure you:

- Upload the handler command (shell or Perl script, etc.) to the Nagios XI server (**Admin** > **System Extensions** > **Manage Plugins**)

- Reference the full path to the handler command script in the configuration page

- Check the **Enabled** check box next to the command definition

- Check the **Don't Run in Down Time** check box if you want to suppress executing the event handler during downtime (only available in Nagios XI 2014 or later versions).

- Click the **Apply Settings** button when finished

Please refer to the following documentation for information on understanding event handlers within Nagios XI:

Introduction to Event Handlers in Nagios XI

General Considerations

Keep these considerations in mind when utilizing global event handler commands:

- Global event handlers will run with the privileges of the `nagios` user on the local Nagios XI server

- Event handlers should execute quickly so they can return control to Nagios XI to process other events

- Time-intensive global event handlers should be executed as a background or daemon process, this will prevent impacts to Nagios XI performance

Command Variables

Several variables (Macros) are available to Nagios XI global event handler commands. The example command definitions provided in the global event handler component demonstrate how to pass these variables to your commands. Variables available to global event handlers fall into several categories:

- General variables

- Host state change variables

- Service state change variables

General Variables

The following variables can be used in both host and service state change handlers and notification handlers.

Variable	Description
`%datetime%`	The current date and time, formatted according to the user's date format preference
`%xiserverurl%`	The URL to the Nagios XI server, as defined by the administrator

Host State Change Variables

The following host variables can be used in host state change handlers.

Variable	Description
`%currentattempt%`	A number indicating the current check attempt for the host. Valid values range from one (1) up to `%maxattempts%`
`%host%`	The host name
`%hostaddress%`	The address of the host, usually an IP address or FQDN
`%hosteventid%`	A unique id number for the current host event
`%hostoutput%`	The text output from the last check of the host.
`%hostproblemid%`	A unique id number for the current host problem
`%hoststate%`	A string indicating the current state of the host, valid values are: **UP**, **DOWN**, or **UNREACHABLE**
`%hoststateid%`	A number indicating the current state of the host, valid values and their string equivalents are: **0**=UP, **1**=DOWN, **2**=UNREACHABLE
`%hoststatetype%`	A string indicating the current state type for the host, valid values are **SOFT** and **HARD**. This value is almost always HARD for problem and recovery notifications. SOFT state occur when a host enters or recovers from a non-UP state before being re-checked `%maxattempts%` times.
`%lasthoststate%`	A string indicating the last state of the host, values are the same as for the `%hoststate%` variable

`%lasthoststateid%`	A number indicating the last state of the host, values are the same as for the `%hoststateid%` variable
`%maxattempts%`	A number indicating the maximum number of check attempts that will be made before the host is considered to be in a HARD non-UP state

Service State Change Variables

The following service variables can be used in service state change handlers.

Variable	Description
`%currentattempt%`	A number indicating the current check attempt for the service, valid values range from one (1) up to `%maxattempts%`
`%service%`	The service name
`%serviceeventid%`	A unique id number for the current service event
`%serviceoutput%`	The text output from the last check of the service
`%serviceproblemid%`	A unique id number for the current service problem
`%servicestate%`	A string indicating the current state of the service, valid values are: **OK**, **WARNING**, **CRITICAL** and **UNKNOWN**
`%servicestateid%`	A number indicating the current state of the service, valid values and their string equivalents are: **0**=OK, **1**=WARNING, **2**=CRITICAL, **3**=UNKNOWN
`%servicestatetype%`	A string indicating the current state type for the service, valid values are **SOFT** and **HARD**. This value is almost always HARD for problem and recovery notifications. SOFT state occur when a service enters or recovers from a non-OK state before being re-checked `%maxattempts%` times.
`%lastservicestate%`	A string indicating the last state of the service, values are the same as for the `%servicestate%` variable
`%lastservicestateid%`	A number indicating the last state of the service, values are the same as for the `%servicestateid%` variable

%maxattempts%	A number indicating the maximum number of check attempts that will be made before the service is considered to be in a HARD non-OK state

Service state change handler commands can also contain some host variables. When used, these variables refer to the host that is associated with the service. Valid host variables that can be used in service state change handlers are listed below.

Variable
%host%
%hostaddress%
%hosteventid%
%hostproblemid%
%hoststate%
%hoststateid%

Adding Nagios Core Macros To The Global Event Handler Component

In addition to referencing Nagios XI macros, you can also reference Nagios Core macros in the global event handler commands. The global event handler component uses a pair of commands to handle host and service state changes. By editing these commands in **Configure** > **Core Config Manager** > **Commands**, you can add additional Nagios Core macros that will be made available to the Nagios XI global event handler commands in the form of %<macro>%. Host state changes are controlled by the command: xi_host_event_handler and service state changes are controlled by the command: xi_service_event_handler.

On the following page are the default command definitions for each of these commands:

xi_host_event_handler:

```
/usr/bin/php /usr/local/nagiosxi/scripts/handle_nagioscore_event.php
--handler-type=host --host="$HOSTNAME$" --hostaddress="$HOSTADDRESS$"
--hoststate=$HOSTSTATE$ --hoststateid=$HOSTSTATEID$
--lasthoststate=$LASTHOSTSTATE$ --lasthoststateid=$LASTHOSTSTATEID$
--hoststatetype=$HOSTSTATETYPE$ --currentattempt=$HOSTATTEMPT$
--maxattempts=$MAXHOSTATTEMPTS$ --hosteventid=$HOSTEVENTID$
--hostproblemid=$HOSTPROBLEMID$ --hostoutput="$HOSTOUTPUT$"
--longhostoutput="$LONGHOSTOUTPUT$" --hostdowntime=$HOSTDOWNTIME$
```

xi_service_event_handler:

```
/usr/bin/php /usr/local/nagiosxi/scripts/handle_nagioscore_event.php
--handler-type=service --host="$HOSTNAME$" --service="$SERVICEDESC$"
--hostaddress="$HOSTADDRESS$" --hoststate=$HOSTSTATE$
--hoststateid=$HOSTSTATEID$ --hosteventid=$HOSTEVENTID$
--hostproblemid=$HOSTPROBLEMID$ --servicestate=$SERVICESTATE$
--servicestateid=$SERVICESTATEID$ --lastservicestate=$LASTSERVICESTATE$
--lastservicestateid=$LASTSERVICESTATEID$
--servicestatetype=$SERVICESTATETYPE$ --currentattempt=$SERVICEATTEMPT$
--maxattempts=$MAXSERVICEATTEMPTS$ --serviceeventid=$SERVICEEVENTID$
--serviceproblemid=$SERVICEPROBLEMID$ --serviceoutput="$SERVICEOUTPUT$"
--longserviceoutput="$LONGSERVICEOUTPUT$" --servicedowntime=$SERVICEDOWNTIME$
```

Any standard Nagios core macro can be assigned to a `%<macro>%` for use in the global event handler component. A list of available macros can be found in the Nagios Core Documentation. Adding a macro is as simple as editing the relevant event handler command above, adding an assignment in the form of:

```
--<variable name>="$<core macro>$"
```

Where `<variable name>` will be the name you reference in the component as `%<variable name>%` and

"$<core macro>$" will be the desired core macro from Nagios Core Documentation.

For example, to make the Nagios macro `$HOSTDURATION$` available to the global event handler component for a host event handler, edit the `xi_host_event_handler` command and add the following to the end of the command:

```
--hostduration="$HOSTDURATION$"
```

The `xi_host_event_handler` command should now resemble:

```
/usr/bin/php /usr/local/nagiosxi/scripts/handle_nagioscore_event.php
--handler-type=host --host="$HOSTNAME$" --hostaddress="$HOSTADDRESS$"
--hoststate=$HOSTSTATE$ --hoststateid=$HOSTSTATEID$
--lasthoststate=$LASTHOSTSTATE$ --lasthoststateid=$LASTHOSTSTATEID$
--hoststatetype=$HOSTSTATETYPE$ --currentattempt=$HOSTATTEMPT$
--maxattempts=$MAXHOSTATTEMPTS$ --hosteventid=$HOSTEVENTID$
--hostproblemid=$HOSTPROBLEMID$ --hostoutput="$HOSTOUTPUT$"
--longhostoutput="$LONGHOSTOUTPUT$" --hostdowntime=$HOSTDOWNTIME$
--hostduration="$HOSTDURATION$"
```

You should now be able to reference the `$HOSTDURATION$` macro in the global host event handler with the variable: `%hostduration%`

The same procedure can be performed for the `xi_service_event_handler` to make additional core macros available to the global service event handler.

Host And Service Notification Variables

The host and service notification variables that are available to global event handlers are the same as those available in notification messages that go out to end users. These variables are described in the following documentation:

[Nagios XI Notification Variables](#)

This document contains instructions to add and edit notification variables for host and service notification handlers (sections **Passing Host Variables To Notification Messages** and **Passing Service Variables To Notification Messages**). Once new variables are added to the notification handlers, they will be available to pass to external scripts in the notification handler tab of the global event handler component.

Finishing Up

This completes the documentation on configuring global event handlers 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