



Purpose

This document provides a solution for generating audio alerts in Nagios XI. The solution is based on using a Raspberry Pi to output the audio alert via HDMI or 3.5mm headphone jack.

Target Audience

This document is intended for use by Nagios Administrators who wish to provide an alternate way to receive notification alerts from Nagios XI.

Overview

An audio alert from Nagios is just another type of notification. Instead of receiving an email informing you of a problem, an audio alert would be heard without needing to look at computer. So what exactly is an audio alert?

- This solution generates the audio using a program called Flite, it is a text to speech synthesis engine
- In real time the Pi will generate the audio and output it to the via HDMI or 3.5mm headphone jack
- This means you can customize the alert to say whatever you want, for example:
 - The host server01 is currently down
 - The service disk usage C: for the host server01 is currently in a warning state, the last output was C:\ - total: 99.51 Gb - used: 25.89 Gb (26%) - free 73.62 Gb (74%)

A Raspberry Pi is used in this solution because:

- It is a cheap device that is commonly used to display Nagios health on a TV or screen in a NOC
- It has HDMI and or 3.5mm headphone jack audio output
- The audio alerts are not physically tied to the Nagios XI Server
 - Commands are sent to the Pi using an SSH connection
 - This allows you to easily scale up this solution to multiple Pi's in different rooms/locations
- It provides an easy to implement solution

This solution is not limited to a Raspberry Pi, as you follow the documentation you will see that it can be implemented on any Linux Distribution.

Requirements

To be able to implement this solution the following requirements must be met:

- Raspberry Pi (3 Model B used in this example)
- A TV/Screen with built in speakers to play audio or external speakers that can connect to the 3.5mm headphone jack
- Raspbian Stretch Desktop (Lite also works if you wanted an audio only solution without a display)
- A static IP address on the Pi as the Nagios Server will be sending commands to it to play the audio
- An active internet connection as several components will need to be installed

Once you have met all of those requirements you will be able to proceed with the steps in this document.

Enable SSH / Create Nagios User

The first step is to enable the SSH service and create a nagios user. SSH is how Nagios will send the commands to play the audio. It will also provide the ability to monitor the Pi itself using the SSH Proxy configuration wizard in Nagios XI.

On your Pi Desktop open a terminal session. Execute this command to become the root user:

```
sudo su
```

Enable the SSH service so it starts on boot and also start the service by executing these commands:

```
systemctl enable ssh.service  
systemctl start ssh.service
```

Now create the `nagios` user along with adding the user account to the `audio` group so it has permission to output the audio. You will need to provide a password for the `nagios` user. Execute these commands:

```
useradd -m nagios
usermod -a -G audio nagios
passwd nagios
```

Once it is completed you need to configure the SSH keys between the Nagios XI server and the Pi (this allows passwordless authentication). Establish a terminal session to your **Nagios XI** server as the root user. Execute this command to become the `nagios` user on the Nagios XI server:

```
su nagios
```

Execute this command to create an SSH key:

```
ssh-keygen
```

Press ENTER (accepting defaults) when prompted for a filename and passphrase. Public and private SSH keys will be generated and saved in the following directory:

```
/home/nagios/.ssh
```

In the following commands replace `xxx.xxx.xxx.xxx` with the IP address of your Pi. This command will copy the SSH key to the Pi:

```
ssh-copy-id -i ~/.ssh/id_rsa.pub nagios@xxx.xxx.xxx.xxx
```

You will be prompted to add the host to the list of new hosts, you need to type **yes** to proceed and then you will need to type the password for the `nagios` user (the `nagios` user you created on the Pi).

Now to verify that you can login to the Pi without supplying a password. Continuing with the terminal session on the Nagios XI server execute the following command:

```
ssh nagios@xxx.xxx.xxx.xxx
```

If the SSH keys are configured properly you should be able to login to the Pi machine without supplying credentials. Simply type `exit` to close the SSH session.

Also test the `check_by_ssh` plugin, run the following command:

```
/usr/local/nagios/libexec/check_by_ssh -H xxx.xxx.xxx.xxx -C uptime
```

If things are setup properly, you should get output from the "uptime" command on the remote server that looks similar to the following:

```
12:01:42 up 35 min, 1 user, load average: 0.00, 0.01, 0.05
```

If your output is similar then you are ready to proceed to the next step.

Install Flite

Next is to install the Flite application and test that the audio works.

Return to your Pi Desktop terminal session and execute the following command:

```
apt-get install -y flite
```

Once this has been installed you can test it straight away.

Make sure your speakers are turned on and execute the following command:

```
flite -t "testing 1 2 3"
```

If you heard that come through your speakers then everything is working correctly. If it did not, double check your display or external speakers have the volume turned up and execute the command again. If you still do not hear the audio then please continue to the [Troubleshooting](#) section.

There are several voices available in Flite, personally I like the "slt" voice, for example:

```
flite -t "testing 1 2 3" -voice slt
```

In the remainder of this documentation I am going to use this voice.

Test Flite From Nagios XI

Now you can test from the Nagios XI server to see if everything is configured correctly. In you terminal session on your Nagios XI server execute the following commands (replace `xxx.xxx.xxx.xxx` with the IP address of your Pi):

```
cd /usr/local/nagios/libexec/  
./check_by_ssh -H xxx.xxx.xxx.xxx -C '/usr/bin/flite -t "testing 1 2 3" -voice slt'
```

You should hear the audio and receive output similar to:

```
OK - check_by_ssh: Remote command '/usr/bin/flite -t "testing 1 2 3" -voice  
slt' returned status 0
```

If you heard the audio then you are ready to proceed to the next step. If you do not hear the audio then please continue to the [Troubleshooting](#) section.

Configure Nagios XI

The way Nagios XI will decide if it needs to generate an audio alert is exactly the same way it decides if it needs to send an email alert. A Nagios XI user account will be created and used to send notifications to, this is how you will be able to selectively define what hosts and services will generate audio alerts. The next step will create the user and define the host and service notification commands that will be triggered when a notification is sent.

Login to your Nagios XI server and navigate to **Admin > Users > Manage Users**.

The screenshot shows the Nagios XI Admin interface. The top navigation bar includes Home, Views, Dashboards, Reports, Configure, Tools, Help, and Admin (circled in blue). The left sidebar has System Information and Users sections. Under Users, 'Manage Users' is circled in blue. The main content area is titled 'Manage Users' and contains three buttons: 'Add New User' (circled in blue), 'Add users from LDAP/AD', and 'Email All Users'. Below the buttons, it says 'Showing 1-1 of 1 total records'. A table lists the user details:

	Username	Name	Email	Phone Number	Auth Level	Auth Type	Last Login	Actions
<input type="checkbox"/>	nagiosadmin	Nagios Administrator	root@localhost	-	Admin	Local	2017-11-30 17:25:16	

Click the **Add New User** button and the populate the fields as required.

The check boxes highlighted in the screenshot are the most important ones that need to be enabled.

Add New User

General Settings

Username:

Password:

Repeat Password:

Force Password Change at Next Login:

Email User Account Information:

Name:

Email Address:

Create as Monitoring Contact:

Enable Notifications:

Account Enabled:

Security Settings

Authorization Level:

Can see all objects:

Can (re)configure hosts and services:

Can control all objects:

Can see/control monitoring engine:

Can access advanced features:

Has read-only access:

Has API access:

Click the **Add User** button to create the new user.

Now you need to create commands for the notifications, navigate to **Configure > Core Config Manager**.

In the left pane expand **Commands** and click **>_ Commands**.

Click the **+Add New** button.

The screenshot shows the Nagios XI Core Config Manager interface. The top navigation bar includes 'Home', 'Views', 'Dashboards', 'Reports', and 'Configure'. The left sidebar is titled 'CCM Core Config Manager' and contains a tree view with categories like 'Quick Tools', 'Monitoring', 'Alerting', 'Templates', 'Commands', 'Advanced', 'Tools', and 'CCM Admin'. The 'Commands' category is expanded, and the '>_ Commands' sub-item is selected and circled in blue. The main content area is titled 'Commands' and features a '+ Add New' button circled in blue. Below the button, it says 'Displaying 1-15 of 128 results'. A table lists several commands with their names and command lines.

<input type="checkbox"/>	Command Name	Command Line
<input type="checkbox"/>	check-host-alive	\$USER1\$/check_icmp -H \$HOSTADD
<input type="checkbox"/>	check-host-alive-http	\$USER1\$/check_http -H \$HOSTADD
<input type="checkbox"/>	check-host-alive-tftp	tftp \$HOSTNAME\$ 69
<input type="checkbox"/>	check_bpi	/usr/bin/php \$USER1\$/check_bpi.pl
<input type="checkbox"/>	check_dhcp	\$USER1\$/check_dhcp \$ARG1\$

Populate the **Command Name** field with `audio_host_notification_handler`. The **Command Line** field is shown on the following page. Make sure the Command Type is defined as `"misc command"`.

Command Name *

Example: check_example

Command Line *

Example: \$USER1\$/check_example -H \$HOSTADDRESS\$ -P \$ARG1\$ \$ARG2\$

Command Type:

Active ?

Click the **Save** button once you have populated the fields.

You will need to repeat the same steps to create the `audio_service_notification_handler` command. The full details of both commands is shown below. The commands are one long line, they are wrapped over several lines due to their size.

Command Name:

```
audio_host_notification_handler
```

Command Line:

```
$USER1$/check_by_ssh -H $CONTACTPAGER$ -C '/usr/bin/flite -t "$NOTIFICATIONTYPE$ $NOTIFICATIONTYPE$ $NOTIFICATIONTYPE$ the host $HOSTNAME$ has a $HOSTSTATE$ state" -voice slt'
```

Command Name:

```
audio_service_notification_handler
```

Command Line:

```
$USER1$/check_by_ssh -H $CONTACTPAGER$ -C '/usr/bin/flite -t "$NOTIFICATIONTYPE$ $NOTIFICATIONTYPE$ $NOTIFICATIONTYPE$ the service $SERVICEDESC$ on host $HOSTNAME$ has a $SERVICESTATE$ state" -voice slt'
```

These commands are how Nagios XI dynamically generates the audio alert. It is able to pull the values like `$HOSTNAME$` and `$SERVICEDESC$` to provide alerts specific to the object the alert is for.

You can see that `$NOTIFICATIONTYPE$` has been used three times at the beginning of each message, there are two reasons for this:

- To give you a moment to realize you need to listen to the audio alert that is going to be announced
- Depending on the HDMI / audio equipment being used the beginning of the message can be cut off, this can do with the audio equipment "waking up". Having `$NOTIFICATIONTYPE$` there three times ensures you don't miss the important information.

You will also notice the command uses `-H $CONTACTPAGER$` for the Pi address. In the next step you will configure the `audioalerts` contact object with the IP address of the of the Pi. By doing it this way you can easily use multiple Pi's to send audio alerts to, it allows the solution to scale easily.

Here is a screenshot of both commands that have been created:

<input type="checkbox"/>	Command Name	Command Line	Active	Actions	ID
<input type="checkbox"/>	audio_host_notification_handler	<code>\$USER1\$/check_by_ssh -H \$CONTACTPAGER\$ -C '/usr/bin/flite -t "\$NOTIFICATIONTYPE\$ \$NOTIFICATIONTYPE\$ \$NOTIFICATIONTYPE\$ the host \$HOSTNAME\$ has a \$HOSTSTATE\$ state" -voice slt'</code>	Yes		128
<input type="checkbox"/>	audio_service_notification_handler	<code>\$USER1\$/check_by_ssh -H \$CONTACTPAGER\$ -C '/usr/bin/flite -t "\$NOTIFICATIONTYPE\$ \$NOTIFICATIONTYPE\$ \$NOTIFICATIONTYPE\$ the service \$SERVICEDESC\$ on host \$HOSTNAME\$ has a \$SERVICESTATE\$ state" -voice slt'</code>	Yes		129

Now that the commands have been created they need to be assigned to the contact. In the left pane expand **Alerting** and click **Contacts**.

The screenshot shows the Nagios XI interface. The top navigation bar includes Home, Views, Dashboards, Reports, Configure, Tools, Help, and Admin. The left sidebar has 'Alerting' expanded, with 'Contacts' selected. The main content area is titled 'Contacts' and shows a table with two contacts: 'audioalerts' (ID 2) and 'nagiosadmin' (ID 1). The 'audioalerts' contact is circled in blue. Below the table are buttons for '+ Add New', 'Apply Configuration', and a filter dropdown set to 'With checked'. A search bar is visible at the top right of the main content area.

On the **Common Settings** tab you will need to provide the IP address of the Pi in the Pager Number field.

Contact Management

Common Settings | Alert Settings | Misc Settings

Contact Name *

Description

Email Address

Pager Number

Click the **Alert Settings** tab.

There are two buttons here:

Manage Host Notification Commands

Manage Service Notification Commands

The screenshot shows the 'Contact Management' interface with the 'Alert Settings' tab active. It features several sections for configuring notifications, including 'Host Notifications Enabled', 'Service Notifications Enabled', 'Host Notifications Timeperiod', 'Service Notifications Timeperiod', 'Host Notification options', 'Service Notification options', 'Retain status information', 'Retain non-status information', and 'Can Submit Commands'. Two prominent buttons are visible: 'Manage Host Notification Commands' and 'Manage Service Notification Commands'. A blue arrow points from the left button to the right button.

Click the **Manage Host Notification Commands** button, this will bring up the **Manage** window.

In the Manage window you need to select the `audio_host_notification_handler` command in the left pane and then click the **Add Selected** button so it appears in the right hand pane.

This is for HOST notifications, hence why you added the `audio_host_XXX` command only.

Click the **Close** button.

The 'Manage' window is shown with a list of notification handlers on the left. The 'audio_host_notification_handler' is selected. The 'Add Selected' button is highlighted in blue. The right pane shows the 'Assigned' list with 'audio_host_notification_handler' added. A red 'X' icon is next to it. The 'Remove All' button is visible in the top right of the Assigned list.

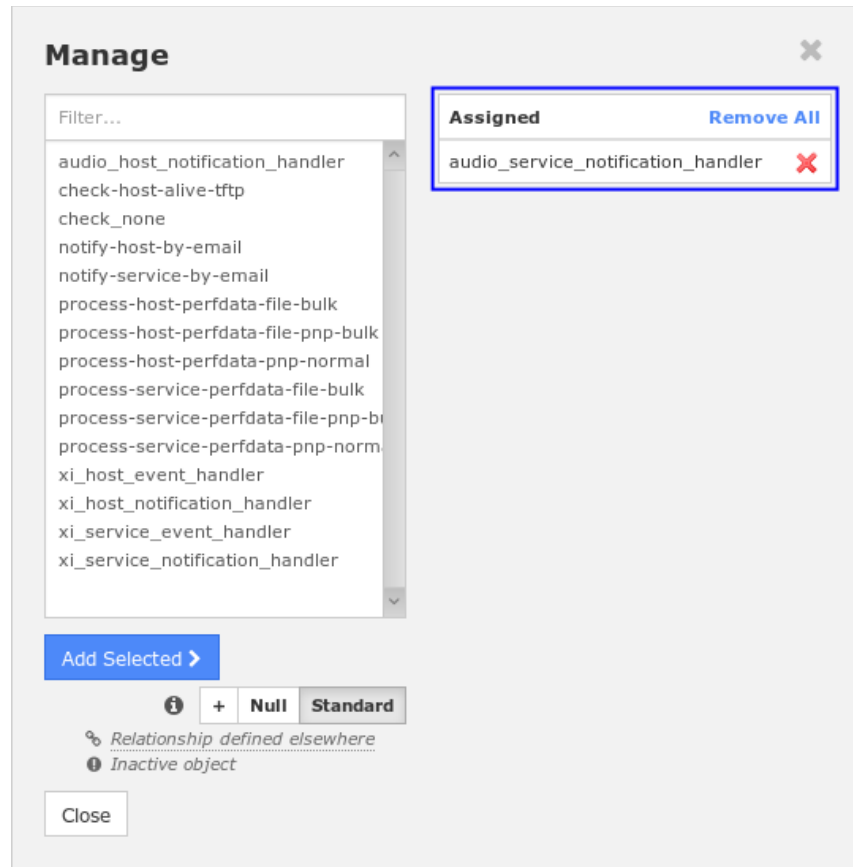
Click the **Manage Service Notification Commands** button, this will bring up the **Manage** window.

In the Manage window you need to select the `audio_service_notification_handler` command in the left pane and then click the **Add Selected** button so it appears in the right hand pane.

This is for SERVICE notifications, hence why you added the `audio_service_xxx` command only.

Click the **Close** button.

Click the **Save** button, this completes the changes required for the Contact object.



Once you've made your selections click the **Apply Configuration** button to push these changes into the running monitoring configuration.

Assign Contact To Objects And Test

At this point all that is remaining is to assign the `audioalerts` contact to the Nagios Host and Service objects that you want to receive audio alerts for. This example is going to use the **Current Users** service for the `localhost` object. The thresholds will be adjusted so the service can be put into a warning state and you will be able to hear the audio alert.

In the left pane expand **Monitoring** and click **Services**. Click the **Current Users** service object to edit it.

Service Management

Common Settings	Check Settings	Alert Settings	Misc Settings
<p>Config Name *</p> <input type="text" value="localhost"/>	<p>Check command</p> <input type="text" value="check_local_users"/>		
<p>Description *</p> <input type="text" value="Current Users"/>	<p>Command view</p> <pre>\$USER1\$/check_users -w \$ARG1\$ -c \$ARG2\$</pre>		
<p>Display name</p> <input type="text"/>	<p>\$ARG1\$ 1</p>		
<p>Manage Hosts 1</p>	<p>\$ARG2\$ 50</p>		

Change the `$ARG1$` field to 1, this will cause it to go into a warning state when more than 1 user is logged in.

Service Management

Click the **Alert Settings** tab and then click the **Manage Contacts** button.

Common Settings	Check Settings	Alert Settings	Misc Settings
<p>Manage Contacts 1</p>	<p>Manage Contact Groups 0</p>		
<p>Notification period *</p> <input type="text" value="xi_timeperiod_24x7"/>			

In the Manage Contacts window you need to select the `audioalerts` contact in the left pane and then click the **Add Selected** button so it appears in the right hand pane.

Click the **Close** button and then the Save button.

Once you've made your selections click the **Apply Configuration** button to push these changes into the running monitoring configuration.

Now you are ready to generate an audio alert. Navigate to **Home > Service Detail** and click the **Current Users** service.

Host	Service	Status	Duration	Attempt	Last Check	Status Information
localhost	Current Load	Ok	23d 22h 16m 30s	1/4	2017-12-01 09:01:30	OK - load average: 0.17, 0.11, 0.09
	Current Users	Ok	23d 22h 16m 5s	1/4	2017-12-01 09:01:55	USERS OK - 1 users currently logged in

The service will have an OK state. Earlier on the thresholds were changed so that this service will enter a WARNING state when more than 1 user is logged onto the Nagios XI server. To force it into a WARNING state all you need to do is establish 2 SSH sessions to your Nagios XI server (you should already have one open).

Once you have the 2 SSH sessions open click the **Force an immediate check** link under **Quick Actions**.

The service should go into a WARNING state, however a notification will not be sent yet as it is in SOFT state. It must be in a WARNING state 4 times before it is in a HARD state.

Force an immediate check 3 more times to force the notification to be sent.

Once it goes into a hard state you will hear the audio notification.

If you close one of the SSH sessions and then force another check you will hear the recovery audio notification.

Service Status Detail

Current Users

localhost



USERS OK - 1 users currently logged in

Status Details

Service State:	● Ok
Duration:	23d 22h 17m 4s
Service Stability:	Unchanging (stable)
Last Check:	2017-12-01 09:01:55
Next Check:	2017-12-01 09:06:55

Quick Actions

- Disable notifications
- Force an immediate check



USERS WARNING - 2 users currently logged in

Status Details

Service State:	● Warning
Duration:	6s
Service Stability:	○ Changing
Current Check:	1 of 4
Last Check:	2017-12-01 09:13:43
Next Check:	2017-12-01 09:14:43

Quick Actions

- Acknowledge this problem
- Disable notifications
- Force an immediate check



USERS WARNING - 2 users currently logged in

Status Details

Service State:	● Warning
Duration:	1m 26s
Service Stability:	Unchanging (stable)
Last Check:	2017-12-01 09:14:30
Next Check:	2017-12-01 09:19:30

Quick Actions

- Acknowledge this problem
- Disable notifications
- Force an immediate check

Now the service will return back to an OK state.

You can also verify that the notification was generated by clicking the cell phone icon above the **Overview** tab.

Current Users

localhost



USERS OK - 1 users currently logged in

Status Details

Service State:	● Ok
Duration:	7s
Service Stability:	Unchanging (stable)
Last Check:	2017-12-01 12:07:30
Next Check:	2017-12-01 12:12:30

Quick Actions

- Disable notifications
- Force an immediate check

Date / Time	Host	Service	Reason	Escalated	State	Contact	Dispatcher	Information
2017-12-01 12:07:30	localhost	Current Users	Service Recovery	No	OK	nagiosadmin	Nagios XI	USERS OK - 1 users currently logged in
2017-12-01 12:07:30	localhost	Current Users	Service Recovery	No	OK	audioalerts	Custom: audio_service_notification_handler	USERS OK - 1 users currently logged in
2017-12-01 12:04:53	localhost	Current Users	Service Problem	No	WARNING	nagiosadmin	Nagios XI	USERS WARNING - 2 users currently logged in
2017-12-01 12:04:53	localhost	Current Users	Service Problem	No	WARNING	audioalerts	Custom: audio_service_notification_handler	USERS WARNING - 2 users currently logged in

You can see in the screenshot above how a notification was sent to the `audioalerts` contact for the WARNING and OK states.

This completes all the steps required to implement audio alerts in Nagios XI using a Raspberry Pi.

Considerations

This documentation has provided you with an solution for generating audio alerts with Nagios XI however it has only scratched the surface of what is possible.

Multiple Pi's

- Simply create more Nagios XI users and define the IP address in the Pager field of the contact
- Assign the different contact accounts to the relevant host and service objects, different teams in different rooms can receive audio alerts that only they need to hear

Selective Alerts

- You can change the time period assigned to a contact so that audio alerts are only generated when you want them to
- The types of alerts for a contact can be defined on the contact object, only receive DOWN, WARNING and CRITICAL events if that is what you require

Descriptive Alerts

- Any of the Nagios Macros can be used to define the audio alert messages, they can be found here:
- <https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/macrolist.html>

Extensive Notification Handler

- Currently the `audio_host_notification_handler` and `audio_service_notification_handler` are oriented towards host/service problem/recovery notifications
- Other types of alerts will also be sent to these contacts, like:
 - flapping start/stop
 - custom notifications
 - acknowledgments
 - however they might not have all the required text content
- You can define exactly what text string is sent to the Flite program however you will need to write a wrapper script that performs the logic behind all of that
- If you look at some of the other commands like the `xi_host_notification_handler` you can see that it sends all the valid macros to the `handle_nagioscore_notification.php` script, it is this script that performs all the logic and then actually sends the notification

Speech Synthesis Markup Language (SSML)

- Flite supports reading text that is in SSML format, this allows you to customize the audio output

Monitor The Pi

You should start monitoring the Pi from Nagios XI, it has been configured ready to run the SSH Proxy configuration wizard against it. Documentation on this is located here:

[Monitoring Hosts Using SSH](#)

You will need to have some plugins installed for the monitoring. This KB article shows you how to install Nagios Plugins:

[Installing Nagios Plugins From Source](#)

The check library has a list of different ways to monitor a device, this will help you with the SSH Proxy configuration wizard. This is located in the Nagios Knowledgebase:

[Check Library](#)

Troubleshooting

If you are not receiving audio then there are several troubleshooting steps you can follow.

Reference: https://elinux.org/R-Pi_Troubleshooting#Sound

Audio output to HDMI or 3.5mm headphone jack is automatic however you can override this. If you execute this command you can get a list of the devices on the Pi:

```
/usr/bin/amixer controls
```

The output will be something like:

```
numid=3,iface=MIXER,name='PCM Playback Route'  
numid=2,iface=MIXER,name='PCM Playback Switch'  
numid=1,iface=MIXER,name='PCM Playback Volume'
```

```
numid=5,iface=PCM,name='IEC958 Playback Con Mask'  
numid=4,iface=PCM,name='IEC958 Playback Default'
```

The `numid=3,iface=MIXER,name='PCM Playback Route'` device is what is used to output the audio. You can change the preferred output with the following command:

```
amixer cset numid=3 2
```

Where 0=auto, 1=headphones, 2=hDMI. In that command above you can see that the output was defined as number 2 which is the HDMI. After making the change you can perform a test to see if it is working:

```
flite -t "testing 1 2 3" -voice slt
```

It has been observed that if the display is turned off when the Pi boots up the HDMI audio may not work, a reboot with the display turned on is the resolution to the problem.

Finishing Up

This completes the documentation on how to generate audio alerts from 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>