Nagios Log Server – How To Configure SSL

The Industry Standard in IT Infrastructure Monitoring

Purpose

This document will describe how to setup Nagios Log Server to use SSL/TLS to provide encrypted connections to the Nagios Log Server. This document can also be used as an initial point for troubleshooting SSL/TLS connections.

Target Audience

This document is intended for use by Nagios Log Server Administrators who require encrypted connections to their Nagios Log Server.

Terminology

For your information:

- SSL = Secure Sockets Layer
- TLS = Transport Layer Security

TLS replaces SSL, however the tools used to implement both generally use SSL in their name/directives. For simplicity reasons, the rest of this document will use the term SSL.

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

Installing Necessary Components

Establish a terminal session to your Nagios Log Server and as root and execute the following command:

```
yum install -y mod_ssl openssl
```

You will continue to use this terminal session throughout this documentation.

Updated – March, 2017

Generating A Key

In this demonstration, we will be using a self-signed key. If you are running a bigger production environment you will want to get a key from a company like VeriSign. However, for smaller uses, self-generated keys should be sufficient. First thing you should do is generate the key, execute the following command:

```
openssl genrsa -out ca.key 2048
```

That would have generated some random text. Next you will create a request by executing the following command:

```
openssl req -new -key ca.key -out ca.csr
```

You will need to supply some values, some can be left blank, the following is an example:

- Country Name (2 letter code) [XX]:**AU**
- State or Province Name (full name) []:**NSW**
- Locality Name (eg, city) [Default City]:**Sydney**
- Organization Name (eg, company) [Default Company Ltd]:**My Company Pty Ltd**
- Organizational Unit Name (eg, section) []:
- Common Name (eg, your name or your server's hostname) []:**nls-c7x-x64.domain.local**
- Email Address []:
-
- Please enter the following 'extra' attributes
- to be sent with your certificate request
- A challenge password []:
- An optional company name []:

As you can see above, I did not supply a password, it is not necessary. One more command is required to sign the key, execute the following command:

```
openssl x509 -req -days 365 -in ca.csr -signkey ca.key -out ca.crt
```

Which should produce output saying the Signature was OK and it was Getting Private Key.

Copy Keys

You need to copy the certificate files to the correct location and set permissions, execute the following commands:

```
        cp ca.crt /etc/pki/tls/certs
        cp ca.key /etc/pki/tls/private/
        chmod go-rwx /etc/pki/tls/certs/ca.crt
        chmod go-rwx /etc/pki/tls/private/ca.key
```

Update Apache Configuration

Now you have to tell the Apache web server (httpd) where to look for it.

Edit the `/etc/httpd/conf.d/ssl.conf` file, find the following lines and update them as follows:

```
        SSLCertificateFile /etc/pki/tls/certs/ca.crt
        SSLCertificateKeyFile /etc/pki/tls/private/ca.key
```

In that same file, navigate to the end (press **SHIFT** + **G**), and before `</VirtualHost>` add the following:

```
<IfModule mod_rewrite.c>
RewriteEngine on
RewriteCond $1 !^(index\.php|scripts|media|app|js|css|img|font|vendor|config.js)
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule nagioslogserver/(.*)$ /var/www/html/nagioslogserver/www/index.php/$1 [L,QSA]
</IfModule>
```

Save the changes, you have finished editing this file.

Edit the file `/etc/httpd/conf/httpd.conf` and add the following lines to the end of the file (press **SHIFT** + **G**):

```
        RewriteEngine On
        RewriteCond %{HTTPS} off
        RewriteRule (.*) https://%{HTTP_HOST}%{REQUEST_URI}
```

Save the changes, you have finished editing this file.

Restart Apache

We need to restart the Apache for the new certificate key to be used.

RHEL/CentOS 5.x/6.x:

```
service httpd restart
```

RHEL/CentOS 7.x:

```
systemctl restart httpd.service
```

Firewall Rules

The following firewall rules may need to be added. If you cannot access the Nagios Log Server in the next step (Test Certificate) then it's likely you'll need to run these commands:

RHEL/CentOS 5.x/6.x:

```
iptables -I INPUT -p tcp --dport 443 -j ACCEPT
service iptables save
```

RHEL/CentOS 7.x:

```
firewall-cmd --zone=public --add-port=443/tcp
firewall-cmd --zone=public --add-port=443/tcp --permanent
```

Test Certificate

Now test your connection to the server by directing your web browser to https://yourservername/.

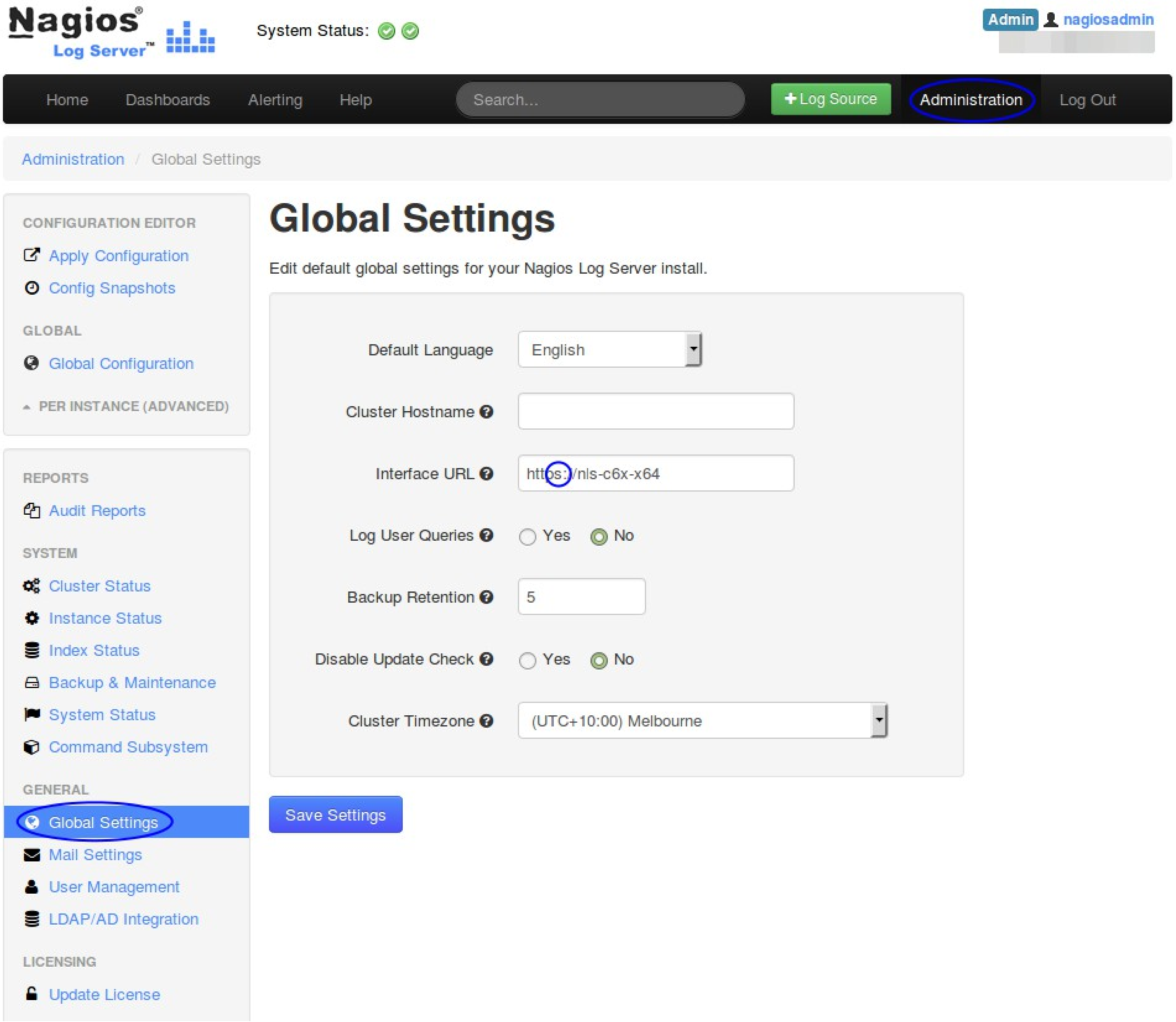
Note: There is no `nagioslogserver/` extension in the URL, we are just testing a connection to Apache to see if the certificate works.

You may get a self signed certificate warning, but that is OK, you can just add a security exception. If is working you'll see the Nagios Log Server welcome page.

If it returns an error check your firewall and backtrack through this document, making sure you've performed all the steps listed.

Update Nagios Log Server Configuration

The Nagios Log Server GUI settings also need updating. Open up the Nagios Log Server interface to `https://yourservername/nagioslogserver/` and navigate to **Administration** > **General** > **Global Settings**.



Global Settings

Edit default global settings for your Nagios Log Server install.

Default Language: English

Cluster Hostname:

Interface URL: https://nls-c6x-x64

Log User Queries: ○ Yes ● No

Backup Retention: 5

Disable Update Check: ○ Yes ● No

Cluster Timezone: (UTC+10:00) Melbourne

Change the **Interface URL** to https instead of the default http and click the **Save Settings** button.

Note: It's very important that the IP Address / DNS name is the same here as it was typed in the certificate key “common name”.

You are now set to use https with your Nagios Log Server web interface.

Notes On Redirecting

With this configuration, if a user types `http://logserver` in their web browser, it will redirect them to `https://logserver` which can cause certificate warnings. If you wanted to redirect them to `https://logserver.yourdomain.com` then you simply need to change the RewriteRule in the `/etc/httpd/conf/httpd.conf` file.

```
RewriteRule (.*) https://logserver.yourdomain.com%{REQUEST_URI}
```

Then restart the httpd service.

Finishing Up

If you have any problems configuring SSL or any other support related questions about Nagios Log Server, please report them to the Nagios Support Forum at:

https://support.nagios.com/forum