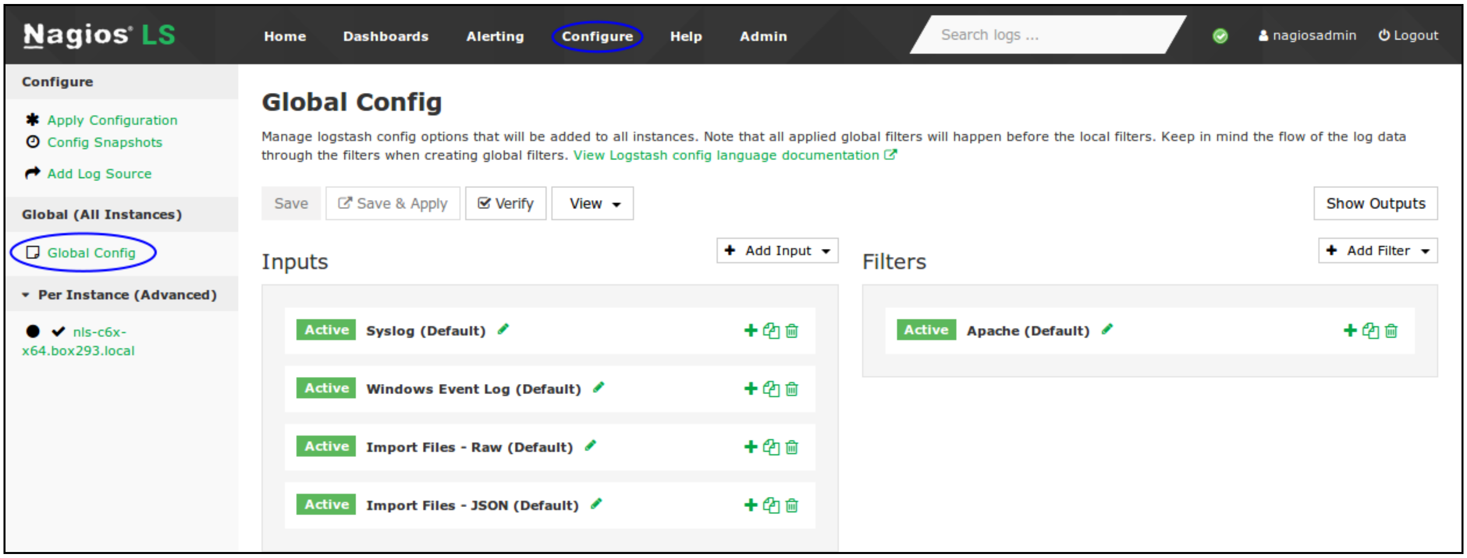
# Configuring Inputs in Nagios Log Server 2024

## What Are Inputs?

Inputs allow Nagios Log Server to collect data from various places, like TCP/UDP ports, SNMP Traps, Unix sockets, long running command pipes, etc.

## Input Configuration Location

Nagios Log Server is a cluster oriented application that uses Logstash to receive and process logs. The base configuration provided with a default installation of Nagios Log Server has all of the inputs defined as part of the Global Config. The Global Config is an easy way to set up the same Logstash configuration on all your instances. To access the configuration navigate to Configure > Global (All Instances) > Global Config.



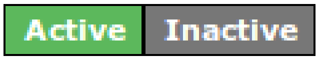
Global Config

Manage logstash config options that will be added to all instances. Note that all applied global filters will happen before the local filters. Keep in mind the flow of the log data through the filters when creating global filters. View Logstash config language documentation

## Input Configuration Options

On the Global Config page there are two main tables named Inputs and Filters (and a third table called Outputs which is hidden). Inputs, Filters and Outputs are all used by Logstash to process incoming log data and do something with it, which normally is to store it in the Elasticsearch database. This document will be focusing on Inputs.

In the Inputs table there are several pre-configured inputs that come as part of Nagios Log Sever, these are called blocks. The blocks have several icons which are explained as follows.



Each of these blocks are Active, which is indicated by the Active box next to each input

When you click the Active box the input will be marked as Inactive and the next time the configuration is applied this input will not be part of the running configuration.

This allows you to save inputs even if you don't want to use them right away. Clicking the box again will toggle it back to Active.

Rename

- Allows you to rename the block

Open / Close

- This will expand the block to show the input configuration in a text field
- Please refer to the Input Structure section of this document for more information
- The icon will change to a hyphen when open, clicking the hyphen will collapse the block

Copy

- This will create a duplicate of that block
- Allows you to easily create a new input based off the configuration of an existing input

🗑Remove

- Delete a block when it is no longer required
- Any changes you make will not be saved until you click the Save button. Keep this in mind as when you navigate away from the page all of your changes will be lost.

## Input Structure

The inputs are a structured format like this:

```
<plugin> {
```

```
<config_option> => <config_value>
```

```
<config_option> => <config_value>
```

```
}
```

<plugin> is the name of plugin that you wish to enable. Each plugin can have multiple <config_option> defined, depending on your requirements. Logstash allows a large amount of possible plugin types, here are two examples:

| | |
|---|---|
| syslog {<br>type => 'syslog'<br>port => 5544<br>} | file {<br>type => 'syslog'<br>path => ['/log/file/location/*.log']<br>start_position => 'beginning'<br>add_field => { 'program' => 'your_program' } |

| | |
|---|---|
| | } |

The example is using the syslog plugin that is configured by default in Nagios Log Server.

The other example is using the file plugin, you can see they have different options defined.

The purpose of showing you these two examples is to demonstrate how an input requires log data to come from "somewhere".

- In the first example the log data is coming via the network on port 5544 (TCP or UDP)
- In the other example the log data is coming from a file

In both examples you can see the option type has the value of 'syslog'. This will label any logs coming in as syslog so you can easily manage them through the dashboard and queries. It also provides the ability for filters and outputs to work with this log data based on the value of the type field.

Another example is the tcp plugin configured for receiving Windows Event Logs. This is configured by default in Nagios Log Server:

```
tcp {
```

```
type => 'eventlog'
```

```
port => 3515
```

```
codec => json {
```

```
charset => 'CP1252'
```

```
}
```

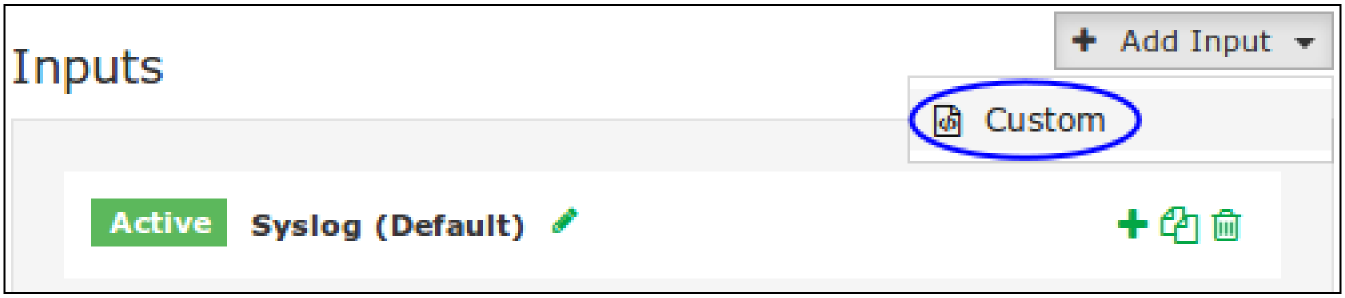
```
}
```

The first option is type and this has the value of 'eventlog'. This will label any logs coming in as eventlog so you can easily manage them through the dashboard and queries.

The next option is port and this has the value of 3515. This enables the Logstash listener to receive TCP network traffic on port 3515 (because this is the tcp plugin, it does not listen for UDP traffic).

The next option is codec and this has the value of json. The json option has additional configuration options that can be defined. You can see in this example that the configuration option of charset is defined as 'CP1252'.

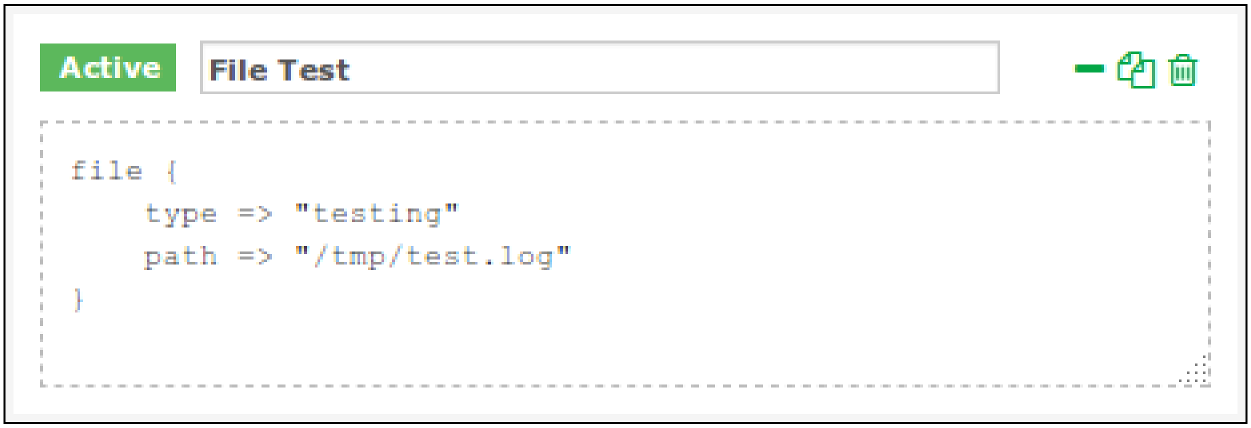
The purpose of this example was to demonstrate that inputs can have a sub-set of configuration settings available, it allows for very flexible configurations.



## Adding An Input

1. Click the Add Input drop down list and select Custom.

A new block will appear at the bottom of the list of Inputs.



File Test

```
file {
    type => "testing"
    path => "/tmp/test.log"
}
```

2. Type a unique name for the input.

3. In the text field you will need to define the input configuration. Here is a basic example for a local file on the Nagios Log Server machine itself:

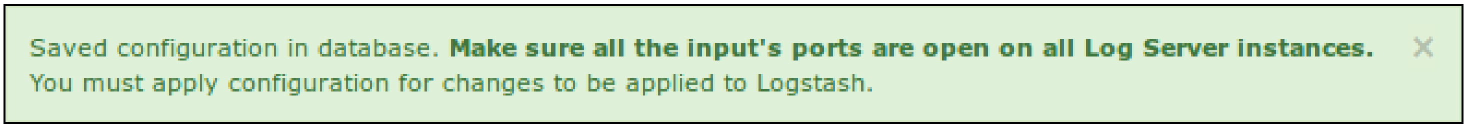
```
file {
```

```
type => "testing"
```

```
path => "/tmp/test.log"
```

```
}
```

4. Once you have finished click the Save button.

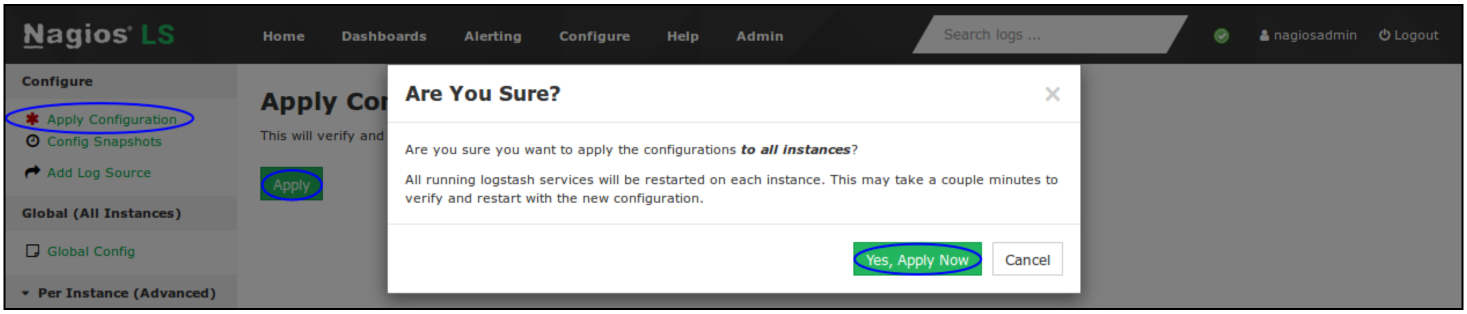


Saved configuration in database. **Make sure all the input's ports are open on all Log Server instances.**
You must apply configuration for changes to be applied to Logstash.

For the new input to become active you will need to Apply Configuration, however it is recommended that you Verify the configuration first. The next step will be to Apply Configuration to put this new input into the running configuration so it can be tested and demonstrated.

## Apply Configuration

To apply the configuration, in the left hand pane under Configure click Apply Configuration.



Are You Sure?

Are you sure you want to apply the configurations ***to all instances***?

All running logstash services will be restarted on each instance. This may take a couple minutes to verify and restart with the new configuration.

Click the Apply button and then on the modal that appears click the Yes, Apply Now button.

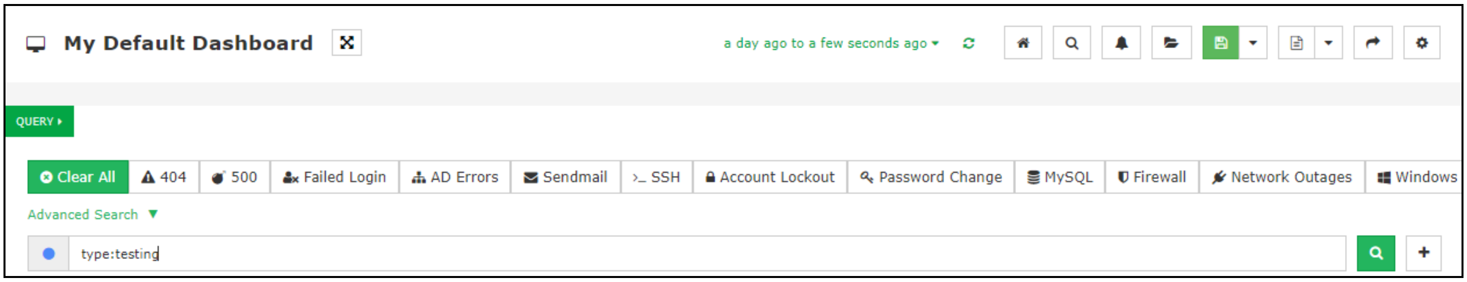
Wait while the configuration is applied. When it completes the screen will refresh with a completed message. Please proceed to the Test Input section.

## Test Input

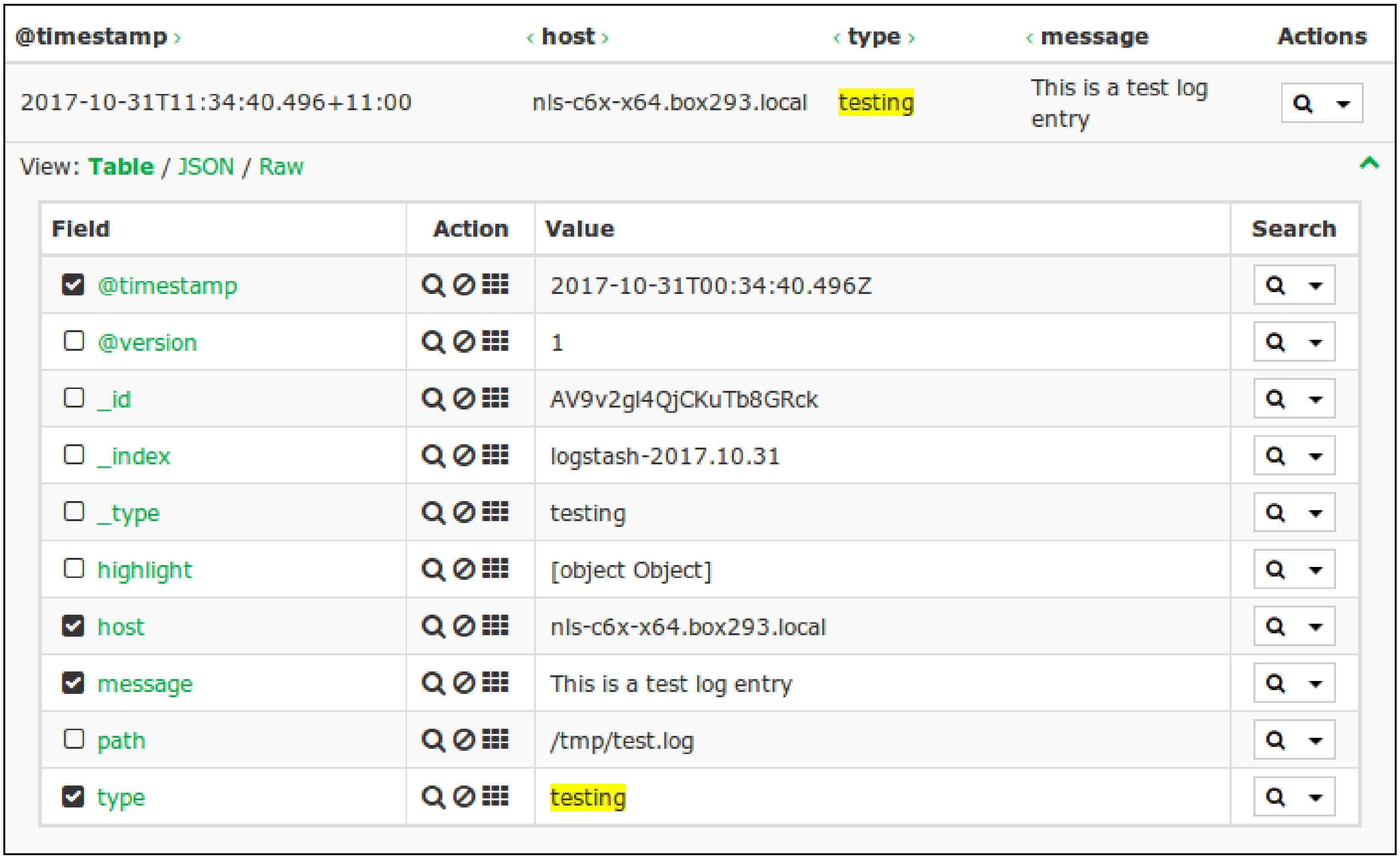
1. Using the input example created previously, these steps will show you how to confirm the input is working. Establish a terminal session to your Nagios Log Server instance and then execute the following command:

```
echo "This is a test log entry" >> /tmp/test.log
```

2. Now in Nagios Log Server open the Dashboards page, Advanced Search, and perform the query type:testing as per this screenshot:



type:testing



| @timestamp › | ‹ host › | ‹ type › | ‹ message | Actions |
|---|---|---|---|---|
| 2017-10-31T11:34:40.496+11:00 | nls-c6x-x64.box293.local | testing | This is a test log entry | 🔍 ▾ |

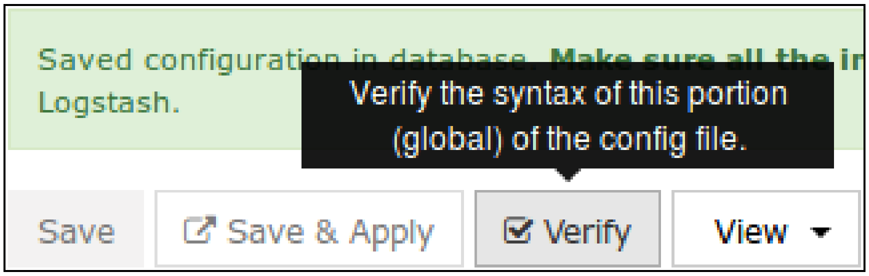
View: Table / JSON / Raw

| Field | Action | Value | Search |
|---|---|---|---|
| ☑ @timestamp | 🔍 ⊘ ▦ | 2017-10-31T00:34:40.496Z | 🔍 ▾ |
| □ @version | 🔍 ⊘ ▦ | 1 | 🔍 ▾ |
| □ _id | 🔍 ⊘ ▦ | AV9v2gl4QjCKuTb8GRck | 🔍 ▾ |
| □ _index | 🔍 ⊘ ▦ | logstash-2017.10.31 | 🔍 ▾ |
| □ _type | 🔍 ⊘ ▦ | testing | 🔍 ▾ |
| □ highlight | 🔍 ⊘ ▦ | [object Object] | 🔍 ▾ |
| ☑ host | 🔍 ⊘ ▦ | nls-c6x-x64.box293.local | 🔍 ▾ |
| ☑ message | 🔍 ⊘ ▦ | This is a test log entry | 🔍 ▾ |
| □ path | 🔍 ⊘ ▦ | /tmp/test.log | 🔍 ▾ |
| ☑ type | 🔍 ⊘ ▦ | testing | 🔍 ▾ |

The query should return one result in the ALL EVENTS panel.

3. Clicking on the log entry will show you the full details about the entry.Here you can see that the type is testing and the text has been stored in the message field. Obviously this test input we created isn't that useful however the purpose was to demonstrate how you can easily create an input and start receiving log data.

The remainder of this documentation explains the buttons available on the Global Config page.

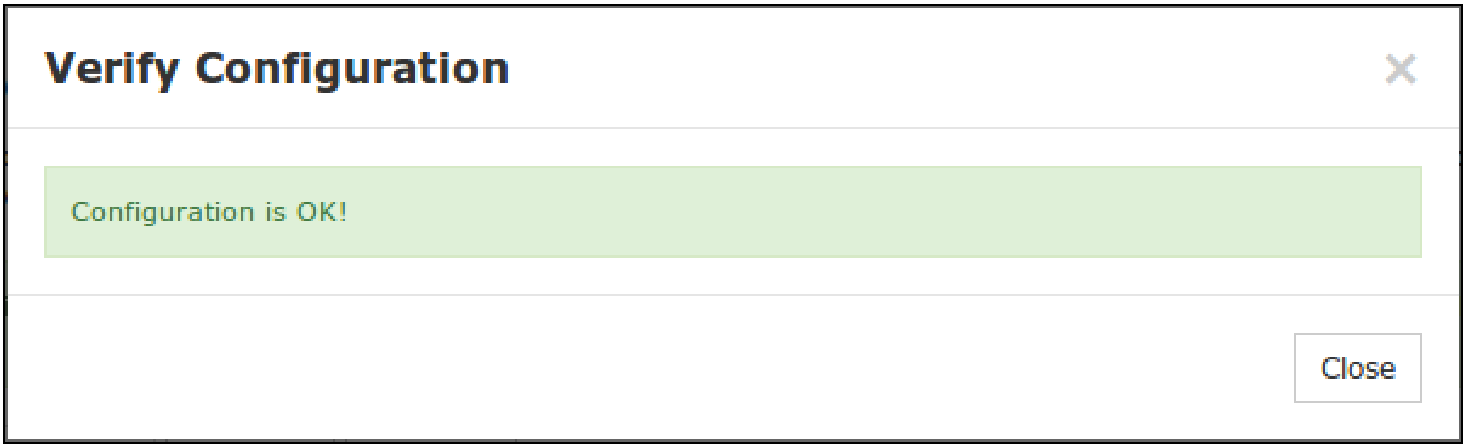


Saved configuration in database. Make sure all the in
Logstash.

Verify the syntax of this portion (global) of the config file.

## Verify

The Verify button ensures that the current saved configuration is valid. It can be useful when updating your configurations before attempting to Apply Configuration.

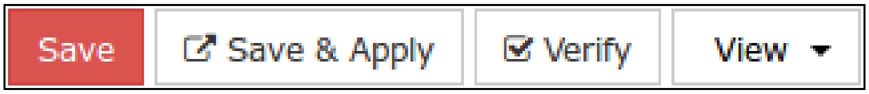


Verify Configuration

Configuration is OK!

Wait while the configuration is verified.

If you do not receive a Configuration is OK message then you will need to fix the problem before applying the configuration.

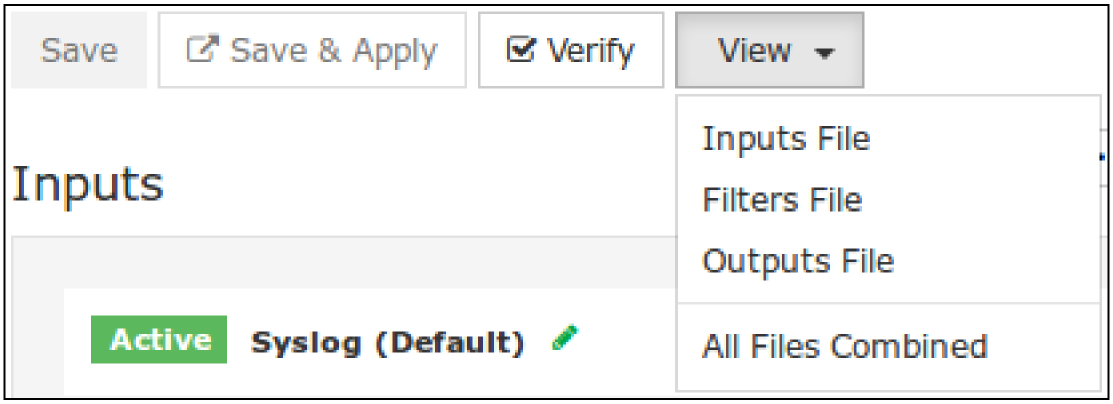


## Save vs Save & Apply

There are two separate buttons, Save and Save & Apply.

Save allows you to make changes but not make the changes become immediately active. It allows you to navigate away from the Configure screen without losing your work.

Save & Apply will save your changes and then make the changes become immediately active. This can be helpful if you changed a simple option in your config that does not need to be verified.



## View

The View button allows you to view the raw Logstash configuration. When you click the button you have a choice of selecting an individual config or a combination of all the configs.

This will open a new modal window where the configuration can be viewed or copied.

## Firewall Ports

If you add an input to Nagios Log Server that uses a network port to receive the data then the local operating system firewall will also require a rule to be added to open that network port.

A detailed explanation of this can be found in the Sending syslog With SSL/TLS documentation. In that documentation the TCP port 7778 needs to be opened, which can be done with the following commands:

**RHEL | CentOS | Oracle Linux**

```
firewall-cmd --zone=public --add-port=7778/tcp
```

```
firewall-cmd --zone=public --add-port=7778/tcp --permanent
```

**Debian:**

The local firewall is not enabled on Debian by default and no steps are required here. IF it is enabled then the commands are:

```
iptables -I INPUT -p tcp --destination-port 7778 -j ACCEPT
```

**Ubuntu:**

The local firewall is not enabled on Ubuntu by default and no steps are required here. IF it is enabled then the commands are:

```
sudo ufw allow 7778/tcp
```

```
sudo ufw reload
```

If you plan on creating the Input as part of the Global Config, you will need to create this firewall rule on all the instances in your Nagios Log Server cluster.

## External Resources

The following resources provide additional information on Logstash:

Structure Of A Logstash Configuration File:

https://www.elastic.co/guide/en/logstash/1.5/configuration-file-structure.html

Input Plugins Available For Logstash:

https://www.elastic.co/guide/en/logstash/1.5/input-plugins.html