# How To Use the Negate Plugin in Nagios XI 2024

## Purpose

This document is designed to assist Nagios administrators in understanding and using the Negate plugin in Nagios XI. The Negate plugin allows for any standard plugin output to be reversed and is very useful with hosts or services that are expected to be in a Critical or Warning state, but you wish to show them as OK. This function can be used for the opposite effect (i.e. showing a CRITICAL state when the actual state is OK).

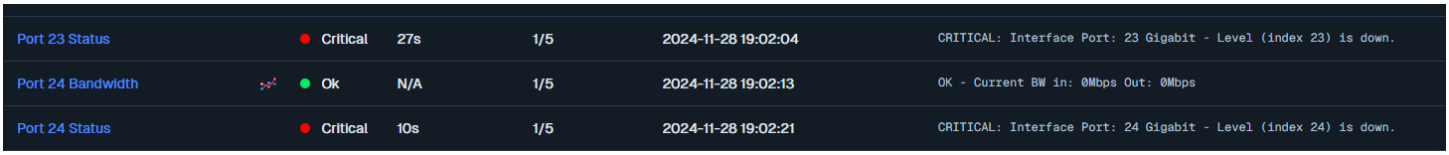
## What Is The Negate Plugin?

Negate is used to execute other plugins, the state returned by the other plugin can be changed by the negate plugin. For example, when a check is normally considered to be in a Critical or Warning state, but the system administrator would instead prefer to see an OK when in such a state. Understand that this will not turn any check consistently to an OK state, but will reverse a critical to OK or an OK to critical, depending on the actual check being run.

## Negate Plugin Example

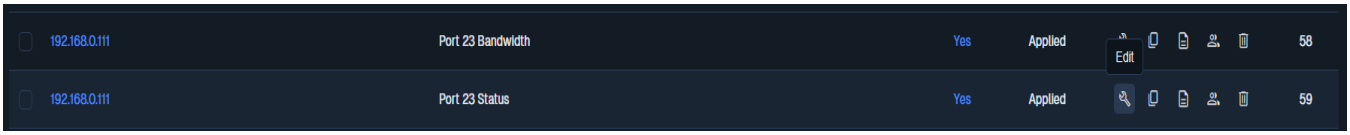
For our example we will use a service check for Port 4 Status on a network switch. Below is the service check for Port 4 Status which is currently not being used and is in a Critical state.

Before showing you how to use the negate plugin with this service, lets us understand how the Port 4 Status service works at the command line.



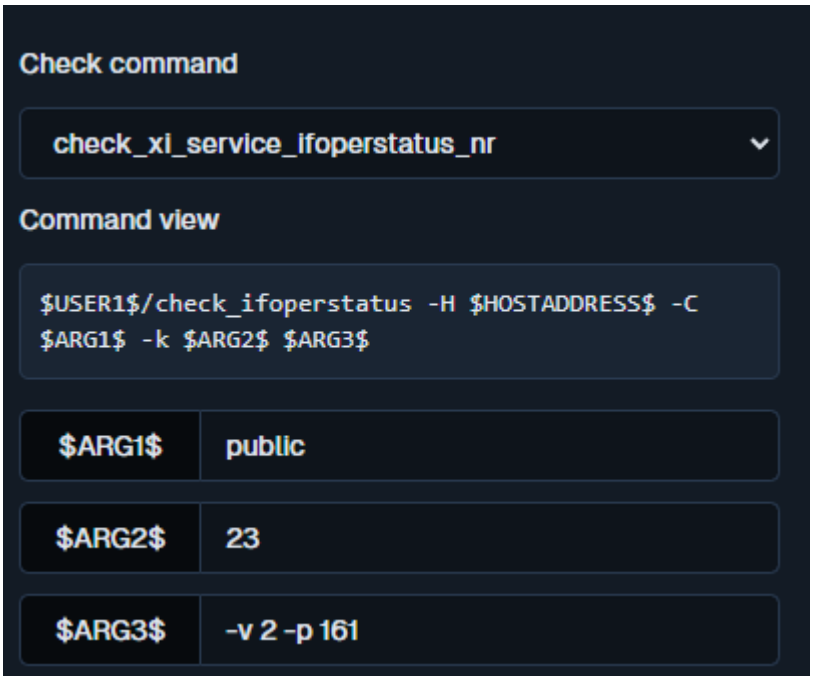

To do that we need to view the service definition in **Core Config Manager (CCM).** Navigate to **Configure > Core Config Manager > Monitoring > Services** and locate the Port 23 Status check.

Click the **modify** icon to view the service configuration.



Notice the Check command drop down has **xi_service_check_ifoperstatus** selected.

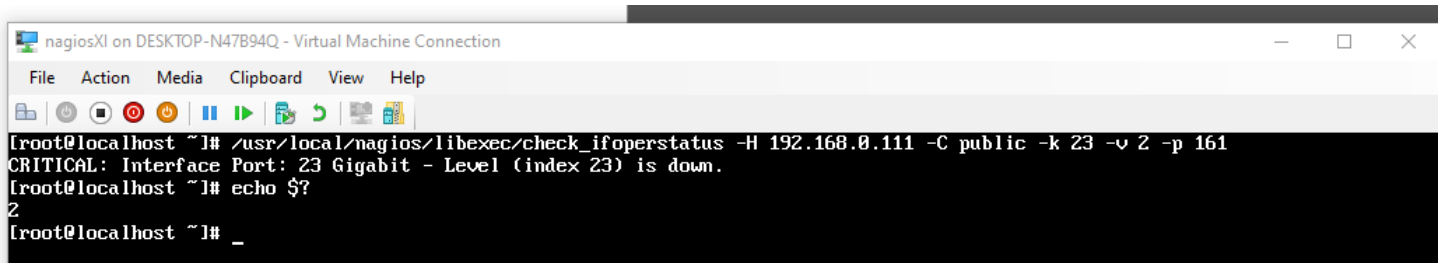
The **Command View** field shows what arguments are used for this command.




The `$ARGx$` fields are the values being used for the command. When Nagios XI executes this command it replaces the `$variables$` with actual values, which results in something like:

```
/usr/local/nagios/libexec/check_ifoperstatus -H 192.168.0.111 -C public -k 23 -v 2 -p 161
```

When this is executed at the command line, the output is:




You'll notice the first line of output is the CRITICAL: Interface tengigabitethernet1/0/4 (index 4) is down. This is only for us humans to understand what the result of the plugin was.

The second line `echo $?` is telling us what the exit code of the plugin was, which is the value 2. The exit code is what tells Nagios XI that the service is in a critical state.

Now let's execute that same command again, but this time use the negate command to turn that 2 state into a 0 state. The command below is one long command, it is just wrapped over two lines:

```
/usr/local/nagios/libexec/negate/usr/local/nagios/libexec/check_ifoperstatus -H 192.168.0.111 -C\
public -k 23 -v 2 -p 161
```

```
[root@localhost ~]# /usr/local/nagios/libexec/negate /usr/local/nagios/libexec/check_ifoperstatus -H 192.168.0.111 -C pu
blic -k 23 -v 2 -p 161
CRITICAL: Interface Port: 23 Gigabit - Level (index 23) is down.
[root@localhost ~]# echo $?
0
[root@localhost ~]#
```

The exit state returned by the plugin is what tells Nagios XI that it's in an OK state, because it's a 0.

You'll notice that the text is still saying CRITICAL, this doesn't affect Nagios but it can be confusing for us humans. There is an additional argument -s that will substitute the output text as well:

```
/usr/local/nagios/libexec/negate -s /usr/local/nagios/libexec/check_
ifoperstatus -H 192.168.0.111 -C public -k 23 -v 2 -p 161
```

```
[root@localhost ~]# /usr/local/nagios/libexec/negate -s /usr/local/nagios/libexec/check_ifoperstatus -H 192.168.0.111 -C
 public -k 23 -v 2 -p 161
OK: Interface Port: 23 Gigabit - Level (index 23) is down.
[root@localhost ~]# echo $?
0
[root@localhost ~]#
```

Now you can see the text output says OK.

## Update Service To Use Negate

Now that you have tested negate from the command line and know how it works you can now implement it in your service definition.

Looking at the original service definition, the command used is `xi_service_check_ifoperstatus`, and the command definition for this is:

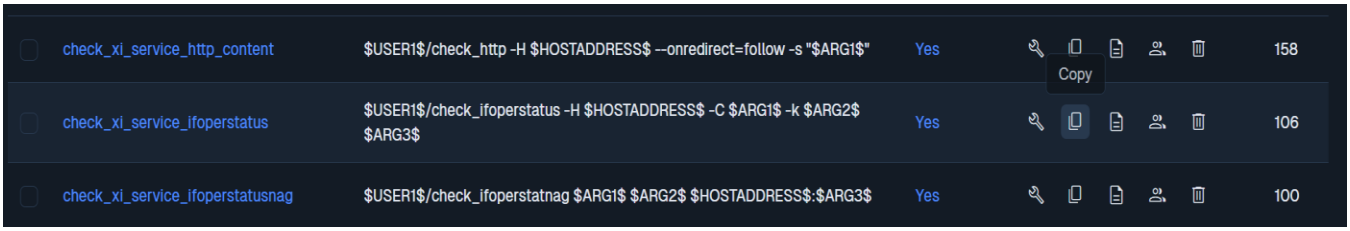
```
$USER1$/check_ifoperstatus -H $HOSTADDRESS$ -C $ARG1$ -k $ARG2$ $ARG3$
```

All that is required is to put the negate command in front of this like so:

```
$USER1$/negate -s $USER1$/check_ifoperstatus -H $HOSTADDRESS$ -C $ARG1$ -k
$ARG2$ $ARG3$
```

However, you shouldn't change the original `xi_service_check_ifoperstatus` command definition as it'll affect all services, instead you can copy the existing command to create a new command.

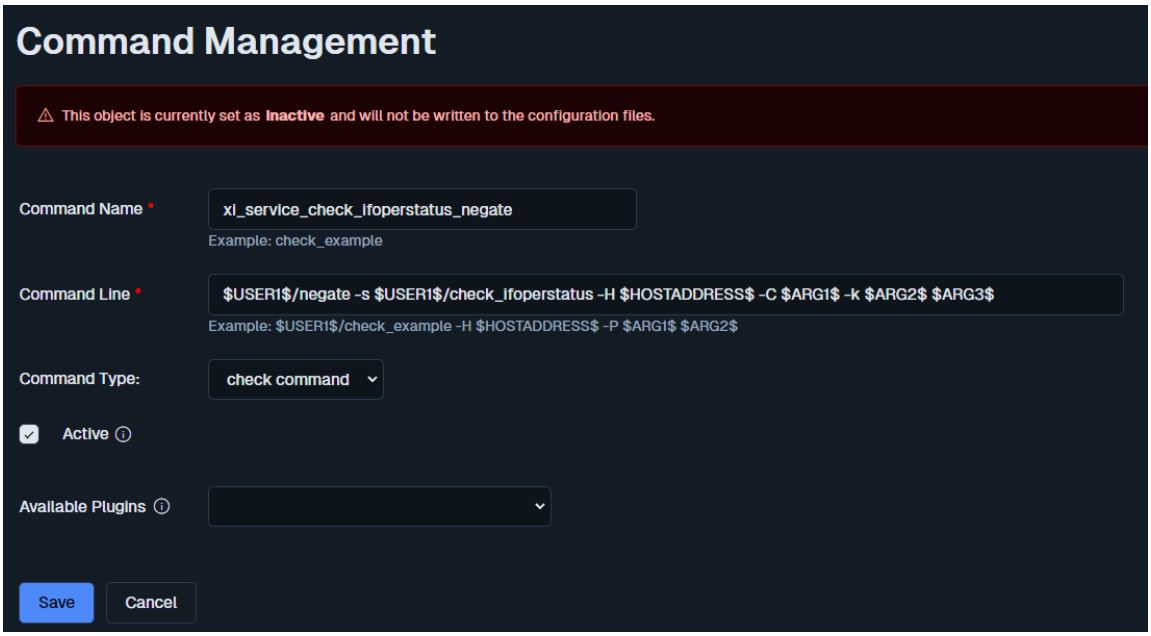
1. Navigate to **Configure > Core Config Manager > Commands > >_Commands**.
2. Locate the `xi_service_check_ifoperstatus` command and click the copy icon.




3. When the screen refreshes you'll have a duplicate command appended with `_ copy_1`.


4. Click the `modify` icon to edit this command.

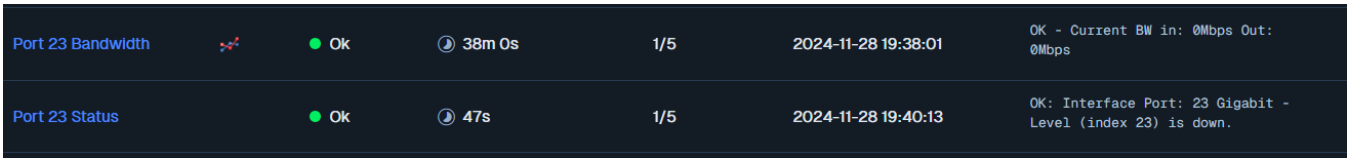
5. You will need to give the command a new name. In this example it seems logical to append the name with `_negate` so the command name is `xi_service_check_ifoperstatus_negate`




6. Then you need to add the negate command (`$USER1$/negate -s`) to the beginning of the command line.
7. Lastly you need to click the **Active** checkbox.
8. Click the **Save** button.

9. The last step is to update the Port 23 Status service with the new check command. Navigate to **Configure > Core Config Manager > Monitoring > Services** and edit the Port 23 Status check.


10. Use the Check command drop down list to select the new command `xi_service_check_ifoperstatus_negate`.
11. Once selected you'll see the **Command view** update, it shows the negate command being used.
12. Click **Save** button and then **Apply Configuration**.
13. After the configuration is applied and the Port 23 Status service is checked, the service will be in an OK state:




You can see that the Port 23 Status service check is in an OK state and the status information shows that the port is down.

## Finishing Up

This completes the documentation on using the Negate Plugin in Nagios XI. If you have additional questions or other support-related questions, please visit us at our Nagios Support Forum, Nagios Knowledge Base, or Nagios Library:

Visit Nagios Support Forum  Visit Nagios Knowledge Base  Visit Nagios Library