

Nagios World Conference 2025

ERIC LOYD

Seriously Advanced
Nagios XI Configuration

- ▶ Started working on Nagios in 2004
- ▶ Speaker at every Nagios World Conference since 2012
- ▶ Received Nagios MVP award in 2014
- ▶ Received another in 2015
- ▶ Winner in the 2015 Nagios Log Server contest
- ▶ COO of EverWatch Global, a Nagios Authorized partner & 100% pure-play Nagios professional services provider
- ▶ IT professional, entrepreneur, musician, and astrophotographer

INTRODUCTION – ABOUT ERIC

Nagios World Conference 2025

AGENDA

1. A very brief overview of the Nagios XI API
2. “Command line” version of the GUI
3. Automated federation Nagios XI Core*
4. Self-aware Nagios XI services

*The new Nagios Mod Gearman of 2024R2.2 effectively supersedes this, but it's still cool and can be done with all versions of XI.



WESTERN VEIL NEBULA (NGC 6960)

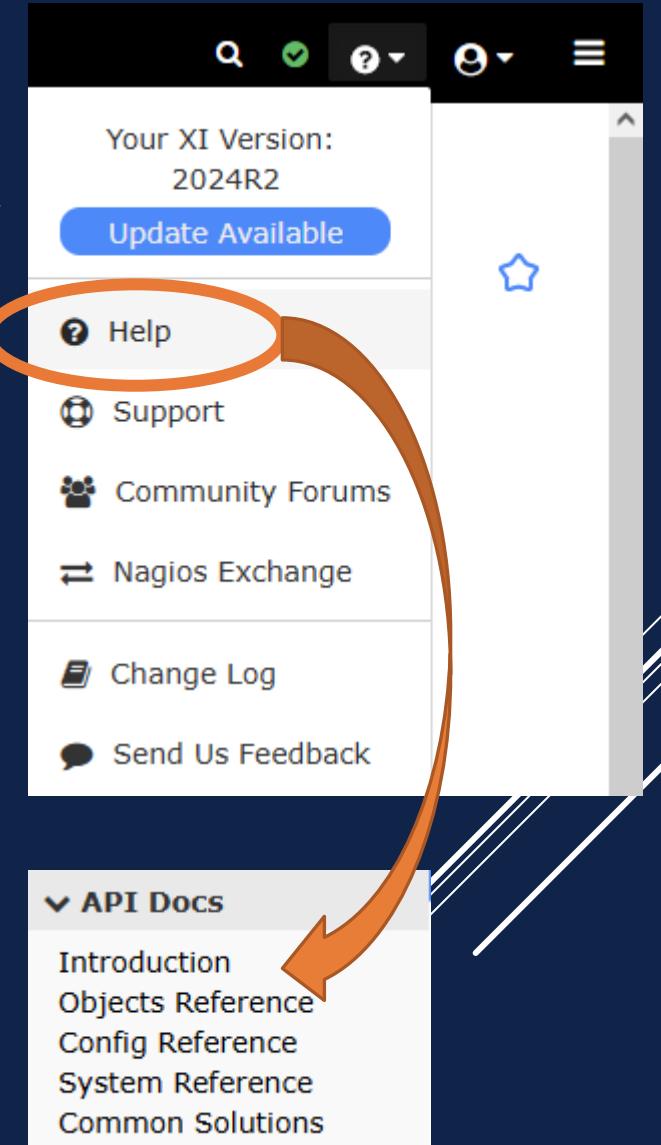
Nagios World Conference 2025

THE NAGIOS XI API

Have you ever actually clicked on the help icon?

- ▶ Click on the “help” button sometime, and you’ll get documentation
- ▶ The API provides full* read and write access to hosts, services, users, configuration, history, and more
- ▶ You need to have API access activated for a user
 - ▶ And you need that user’s API key
- ▶ With this, you can get all sorts of information via `#!/bin/bash`

*Some very esoteric things can’t be done [yet?] via the API (Aaron?)



THE NAGIOS XI API

Nagios World Conference 2025



NORTH AMERICA NEBULA (NGC 7000)

Nagios World Conference 2025

COMMAND LINE GUI

Leverage the API to get all status information
over your crappy hotel hotspot (aka, text)

Edit User: eloyd



Account Settings

Username:

eloyd

New Password:



Email User New Password: [?](#)

Set to a random secure password

Force Password Change at Next Login:



General Settings

Alias (Name):

Eric Loyd

Email Address:

eloyd@everwatch.global

Phone Number:

1234567980



Enable Notifications:



Account Enabled:



Security Settings

Authorization Level: [?](#)

Admin

API Settings

API Key:

9RTG9Aq2NTfefE3XeAiDUbacmNvOJPWfRV3Vbs5DS8ZCCAr6XrIU

[Generate new API key](#)

WHAT YOU'LL NEED

Nagios World Conference 2025

- ▶ Execute the API via *CURL* to get current service status as JSON data
- ▶ Parse that data with *JQ* to get what we want
- ▶ Optionally, only show the bad stuff
- ▶ Convert to CSV
- ▶ Parse with *CSVTOOL* (optional)
- ▶ Display in our SSH session
- ▶ Easy peasy, lemon squeezy
- ▶ cron it all into an email and send to the IT Director every 30 seconds



LET'S GET A COMMAND LINE GUI

Host	Address	State	Type	Last Check	Attempt	Normal	Retry	Max	Command
Host 1	10.20.30.41	1	1	2025-05-07 08:47:38	3	5	10	3	check-host-alive!!!!!!
Host 2	10.20.30.42	1	1	2025-05-07 08:48:46	5	5	1	5	check_xi_host_ping!3000.0!80%!5000.0!100%
Host 3	10.20.30.43	1	1	2025-05-07 08:47:35	3	5	10	3	check_snmp!-C public -p 2161 -o sysUpTimeInstance!!!!!!
Host 4	10.20.30.44	1	1	2025-05-07 08:49:22	3	5	10	3	check-host-alive!!!!!!
Host 5	10.20.30.45	1	1	2025-05-07 08:49:42	3	5	10	3	check-host-alive!!!!!!

Service	Host	State	Type	Last Check	Attempt	Normal	Retry	Max	Output
Service Status - ntpd	Host 1	3	1	2025-05-07 08:33:37	4	5	1	4	ntpd.service - Network Time Service
Light Mode: Twilight	Host 1	2	1	2025-05-07 08:35:54	3	5	10	3	SNMP CRITICAL - *"Day"*
Light Mode: Night	Host 1	2	1	2025-05-07 08:36:25	3	5	10	3	SNMP CRITICAL - *"Day"*
Night Mode	Host 2	2	1	2025-05-07 08:34:16	3	5	10	3	SNMP CRITICAL - *"Day"*
Day/Night Transition Fail	Host 2	2	1	2025-05-07 08:33:39	5	5	1	5	CRITICAL - Plugin timed out
Volts DC	Host 2	2	1	2025-05-07 08:36:33	5	5	1	5	CRITICAL - Plugin timed out
Trans COMM	Host 3	2	1	2025-05-07 08:34:46	5	5	1	5	CRITICAL - Plugin timed out
Night Mode	Host 3	2	1	2025-05-07 08:34:53	3	5	10	3	SNMP CRITICAL - *0*
Beacon Volts Out of Range	Host 4	2	1	2025-05-07 08:34:00	5	5	1	5	CRITICAL - Plugin timed out
Main: Red Fault	Host 5	2	1	2025-05-07 08:32:18	3	5	10	3	CRITICAL - Plugin timed out

SAMPLE OUTPUT

Nagios World Conference 2025



WHIRPOOL GALAXY (M 51)

Nagios World Conference 2025



AUTOMATED FEDERATION

For when it absolutely, positively,
must be done without human intervention
at all hours of the day and night



- ▶ A Nagios XI server with your hosts and services configured passively
- ▶ Decent knowledge of the Nagios XI API
- ▶ Some hosts in hostgroups (*SITE_ONE*, *SITE_TWO* etc)
- ▶ A host template to create a Nagios Core hosts.cfg file [optional]
- ▶ A service template to create a Nagios Core services.cfg file [optional]
- ▶ Some software on your Nagios Core (client) box:
 - ▶ curl
 - ▶ jq

WHAT YOU'LL NEED

- ▶ Nagios Core box at XXX calls API to get hosts in *SITE_XXX* hostgroup
- ▶ Loop through the results and for each host, pull config information for that host from the API and use JQ to reproduce the data structure as a valid "define host {...}" hosts.cfg configuration stanza (whew!)
- ▶ Pull service status information from API
- ▶ Grab entries that are on hosts that match *SITE_XXX* members
- ▶ Pull appropriate fields (service_description, check_interval, etc) and create valid "define service {...}" services.cfg stanzas (double whew!)
- ▶ Repeat for *SITE_YYY*, etc.
- ▶ Reload your Nagios Core and send stuff back to Nagios XI via NRDP



LET'S AUTOMATE SOME STUFF



ELEPHANT TRUNK NEBULA (IC 1396A)

Nagios World Conference 2025



SELF-AWARE SERVICES

\$\$(\$USER1\$/magic happens here)

Nagios World Conference 2025

- ▶ Either:
 - ▶ A narcissistic, megalomaniacal desire to make things as difficult as possible for the next person who comes along and has to read your Nagios configuration after you and your Federated Bots move to Aruba
- ▶ Or:
 - ▶ A zen state of mind that clears all obstacles from having *service_description* fields capable of dictating what the *check_command* actually does
 - ▶ Make one change (or add) to /usr/local/nagios/etc/nagios.cfg:
 - ▶ enable_environment_macros=1
 - ▶ The magic of shell substitution and the headache of dollar signs

WHAT YOU'LL NEED

- ▶ I used XI so this was all done in CCM but you can do this in Core as well
- ▶ Create a custom plugin that will parse **\$NAGIOS_varName** env vars
 - ▶ I called mine **ENV** and it has *very convoluted* parameter processing
- ▶ Create a service template called “**Check Disk: disk warn crit**”
 - ▶ Use it to create a service called “**Disk Usage on /home 80 90**”
- ▶ Set your check_command to be something like (this will be ugly):
 - ▶ **check_xi_ncpa!-t \$\$(\$USER1\$/ENV -NT) -P \$\$(\$USER1\$/ENV -NP)
-M "disk/logical/\$\$\$(\$USER1\$/ENV -F 1 -p)" -w \$\$(\$USER1\$/ENV -F 2)
-c \$\$(\$USER1\$/ENV -F 3)!\$ARG8\$**

LET'S OBFUSCATE SOME STUFF

- ▶ `check_xi_ncpa!-t $$($USER1$/ENV -NT) -P $$($USER1$/ENV -NP) -M "disk/logical/$$$($USER1$/ENV -F 1 -p)" -w $$($USER1$/ENV -F 2) -c $$($USER1$/ENV -F 3)!$ARG8$`
- ▶ **-t \$\$(\$USER1\$/ENV -NT)** Looks for NAGIOS_HOSTNCPATOKEN custom host macro (no default, will break, as written, if not defined)
- ▶ **-P \$\$(\$USER1\$/ENV -NP)** Looks for a NAGIOS_HOSTNCPAPORT custom host macro (defaults to 5693 if not found)
- ▶ **-M "disk/logical/\$\$\$(\$USER1\$/ENV -F 1 -p)"** Looks at the first of the last three fields separated by spaces (/home) [Design choice when making ENV]
- ▶ **-w \$\$(\$USER1\$/ENV -F 2)** Second of the last three fields (80)
- ▶ **-c \$\$(\$USER1\$/ENV -F 3)** Third of the last three fields (90)
- ▶ **\$ARG8\$** So we can append something to the end to override whatever came before it (say, "-w 50" to change warning to 50%); take advantage of shell arg processing overrides

LET'S UNPACK THIS AMAZING CHECK_XI_NCPA COMMAND

Nagios World Conference 2025

Macro names were design choices when creating “ENV”

Parameter processing logic was also a design choice base on typical usage:
Name: Thing Warn Crit

You design your version of “ENV” with whatever makes sense for you.

- ▶ Beware of crazy people with API access
 - ▶ Watch your audit log if you start handing out API keys
- ▶ If you're doing Automated Federation via API calls this way, you'll need a "flatter" configuration approach; no multi-layer inheritance templates, no hostgroups that include hostgroups, etc. The simpler, the better.
 - ▶ Remember: The goal is to distribute processing not simplify configuration
- ▶ If you are going to use self-aware services, then I beg of you, please use service templates. Otherwise you'll spend way more time making your self-aware *service_descriptions* and *check_commands* than you need to.
 - ▶ Also, seek professional mental health counselling. Think of the children.

THINGS TO WATCH OUT FOR



THANK YOU

Eric Loyd

eloyd@everwatch.global

<https://github.com/everwatch> <https://astropotamus.com>

Nagios World Conference 2025



DOWNLOAD MY ASTRO IMAGES

Nagios World Conference 2025

The background image shows a stunning display of the Northern Lights (aurora borealis) in a dark sky. Below the lights, a small town or village is visible along a coastline, with lights from houses and street lamps reflecting on the water. A tall flagpole stands prominently on the right side of the frame.

Q&A SESSION

Eric Loyd

eloyd@everwatch.global

<https://github.com/everwatch> <https://astropotamus.com>

NORTHERN LIGHTS, OCTOBER 2024

Nagios World Conference 2025