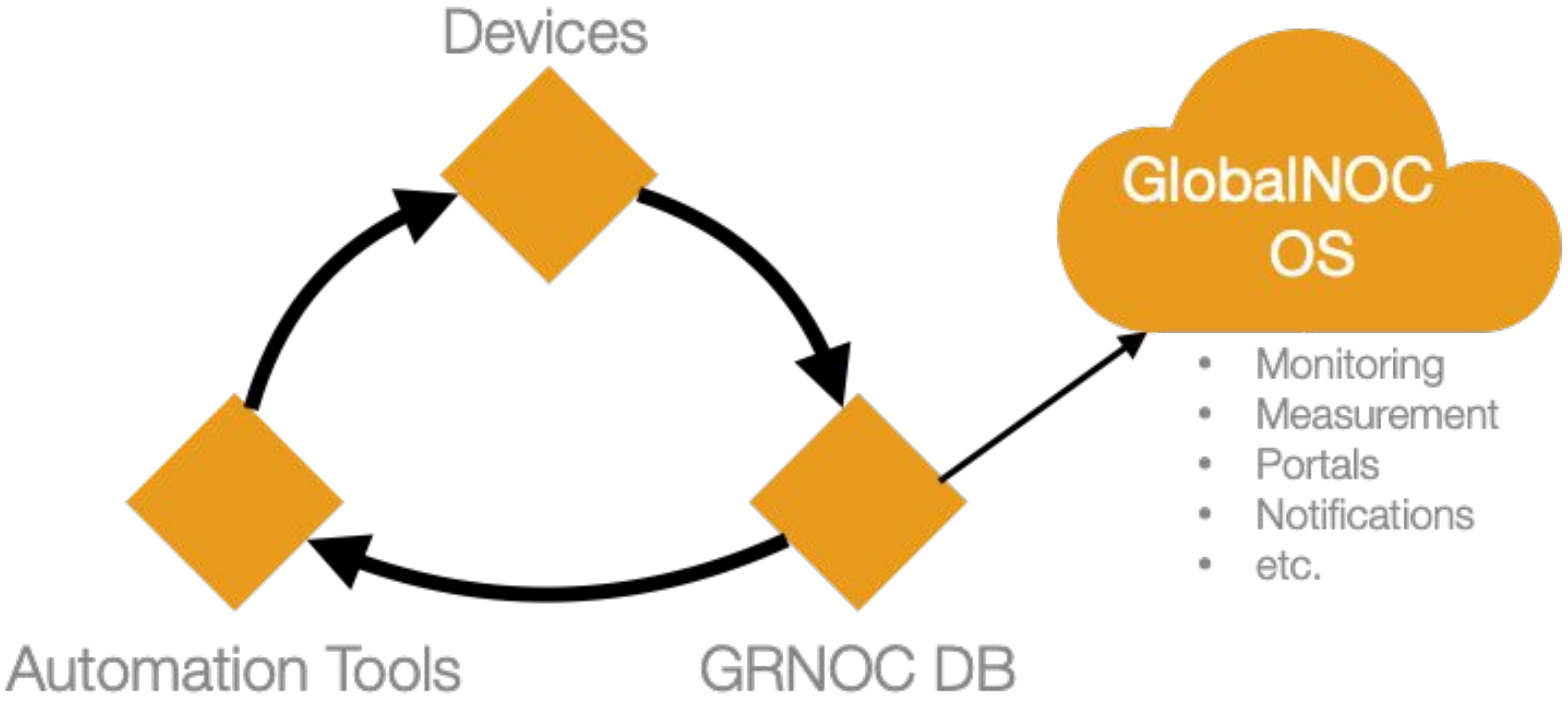
GlobalNOC Network Automation Tools

AJ Ragusa

Philosophy / Goals

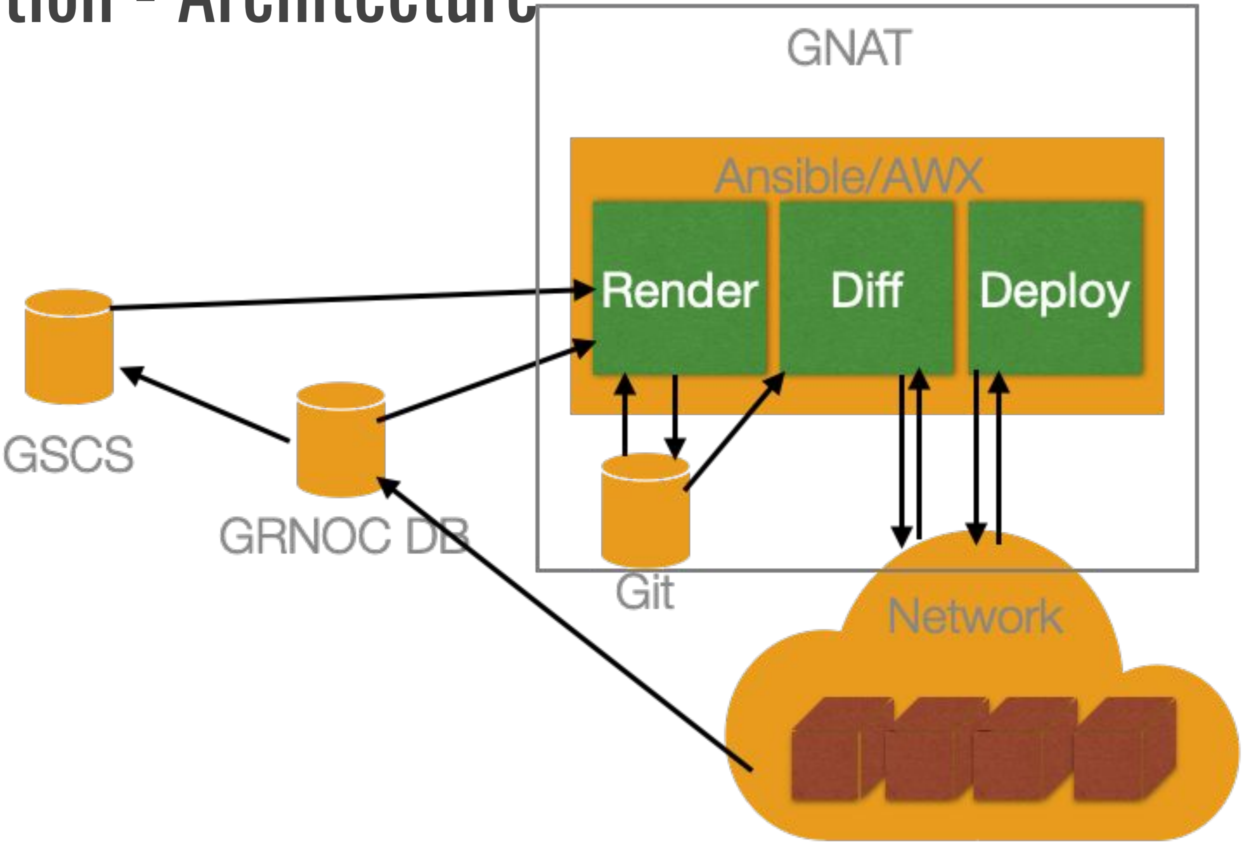
- Flexibility
    - Automate as much or as little of your network as you want
    - Ability to modify the workflow (add in pre/post checks, reviewers, etc.)
    - Software changes not required for new services/models
- Fix Security, and reduce troubleshooting times through standardization
- Treat diverse parts of the configuration as a single “block” of config
- Model/Store this data in a vendor agnostic format, but show vendor specific
- Speed up deployment of new services
- Get the Network Engineers to WANT to use the automation tools!

Automation Lifecycle



- Monitoring
- Measurement
- Portals
- Notifications
- etc.

Network Automation - Architecture



GNAT - GlobalNOC Network Automation Tool

- A combination of “static” templates filled with information from GRNOC DB
- Ansible playbooks / roles are included in the repository
- AWX / Ansible tower is how we launch our playbooks (backend)
- Custom Web-UI facilitates workflow with Network Engineers
- Network Engineers now have full ability to edit the Templates/Plays
- Can integrate other sources (like GSCS, or RADB)
- Add reviewers and add pre-post checks or other checkpoints into workflow

GNAT - Many different tools - 1 workflow

- AWX/Ansible - provides the automation engine for GNAT
  - AWX provides a centralized location for adding new playbooks, storing credentials for logging in, and launching jobs via Web-Services
- Git/GitHub - provides a central repository for storing device configuration and playbooks
  - Stores templates, configurations, passwords and playbooks
  - Versioned! We can roll back to a different version of the network
- GNAT WebUI - The place to launch the workflows

Jinja - YAML - JSON training

———

See Other slide deck!

Git/GitHub - Version control for Network Engineers

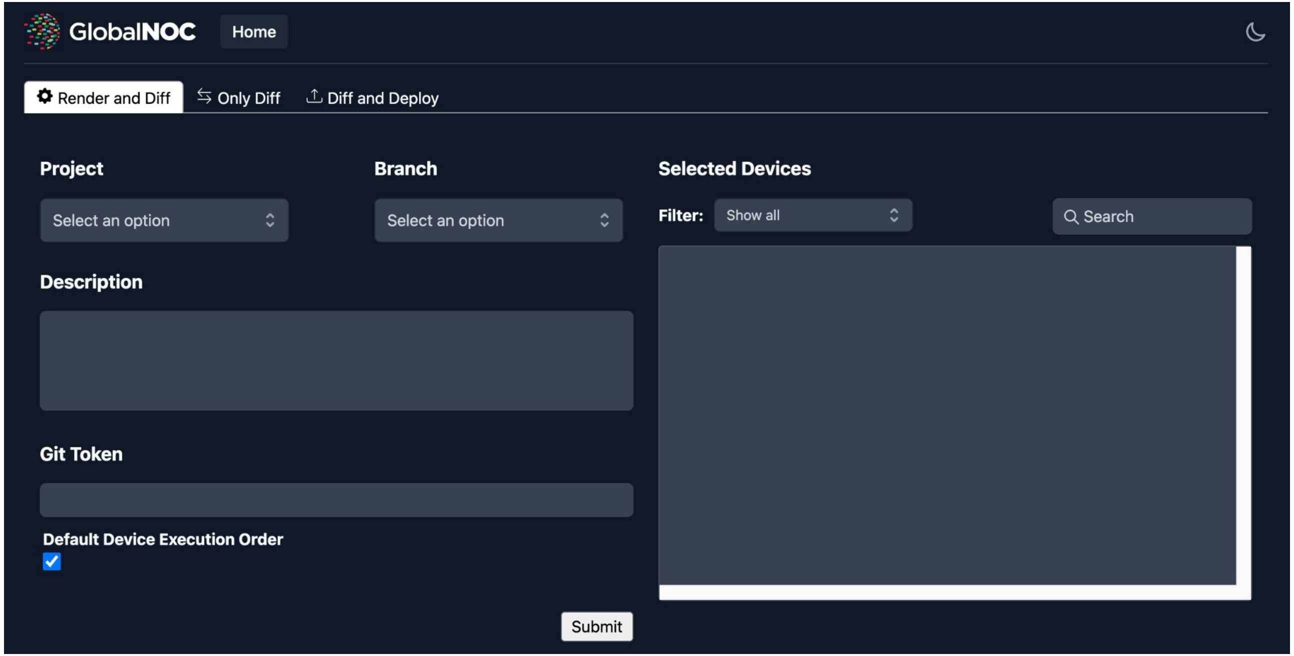
See Slide deck!

AWX/Ansible

See more slides!

Finally GNAT!

- Git Repository Structure
- Templates
- Variables
- The Jobs
  - Render + Diff
  - Diff + Deploy



Filter: Show all

Default Device Execution Order ☑

What is GNAT’s Role?

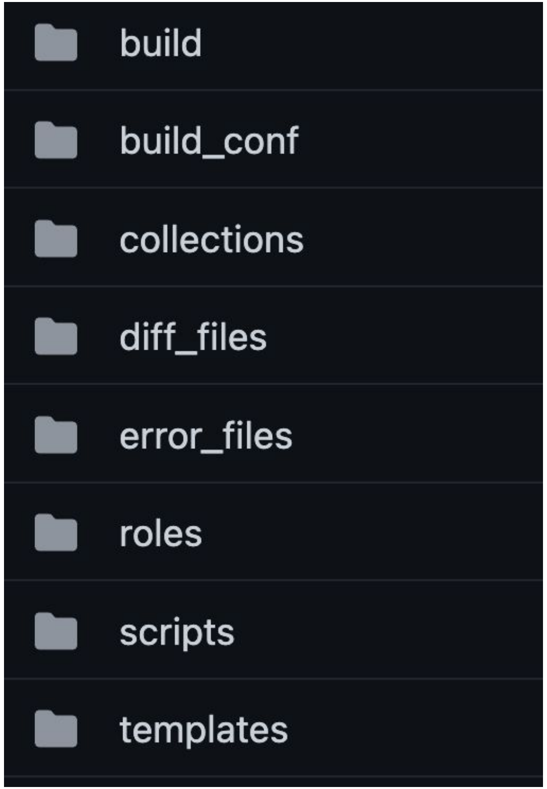
- All of the things that are “the same” across a node role
  - Eg. RADIUS config, NTP, Syslog
  - iBGP, root password, prefix-lists, ACLs, etc…
- Most Networks call this the “base-line”
- What GNAT is not for
  - Each interface’s description
  - Individual BGP peers
  - Services that aren’t on every node! (this is what GSCS is for)

Common Use Cases

- I need to update the root password on all the devices
- I need to add an allowed SSH login host to all devices
- The address of the syslog server has changed
- Diff expected configuration to what is deployed

All of these use cases are great for this tool. Even more we can verify the configuration deployed on a device MATCHES what we expect to be there (raise an alarm in Alertmon if it doesn’t match)

GitHub Repository Structure - GNAT

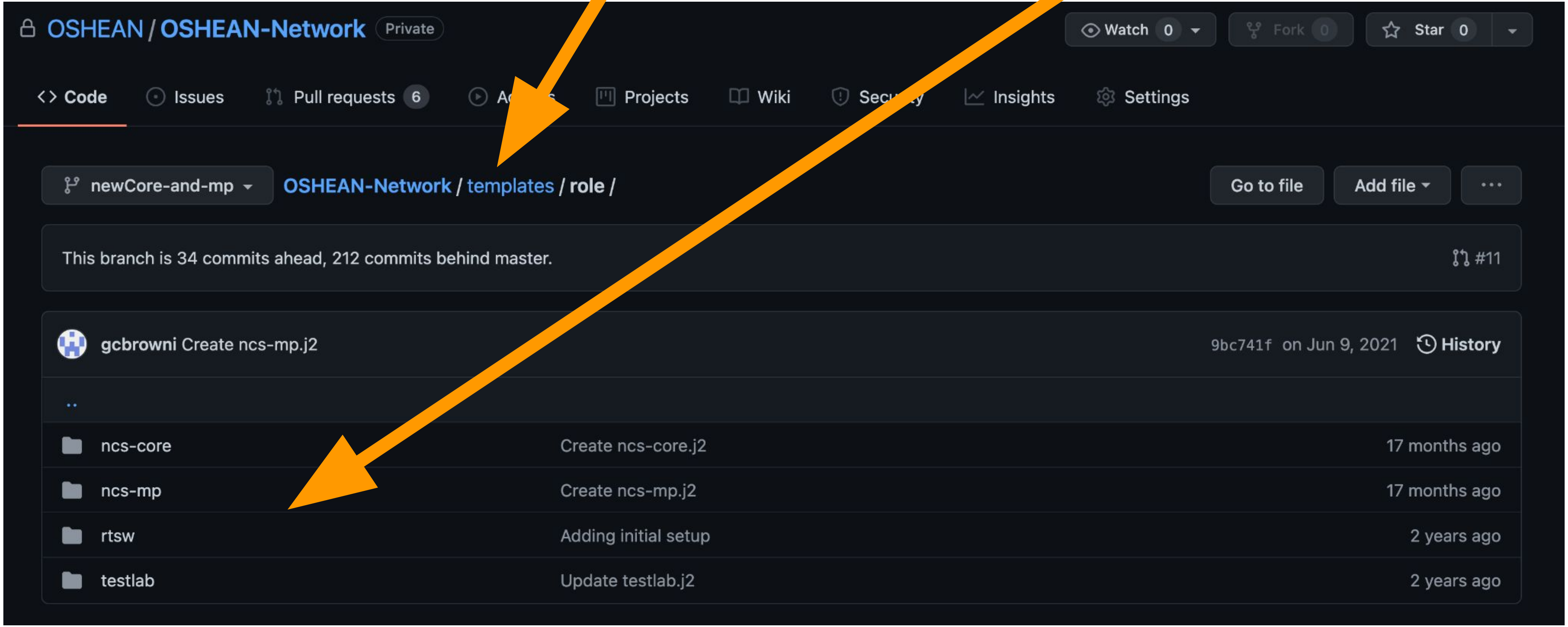


build	The branches rendered configuration - each device will have its own .conf file
build_conf	Configuration variables used to build the configurations (passwords, globals, etc.)
collections	For Ansible - you can ignore
diff_files	Location of Diff files (when diff runs)
error_files	For Ansible - you can ignore
roles	For Ansible - you can ignore
scripts	Scripts maintained by syseng (you can ignore)
templates	Where templates are stored! - YOU WANT THIS!

Templates! - Finally we have enough to work with these!

Please note the path!

Each role code has its own directory



OSHEAN / OSHEAN-Network Private

OSHEAN-Network / templates / role /

This branch is 34 commits ahead, 212 commits behind master. #11

gcbrowni Create ncs-mp.j2 9bc741f on Jun 9, 2021 History

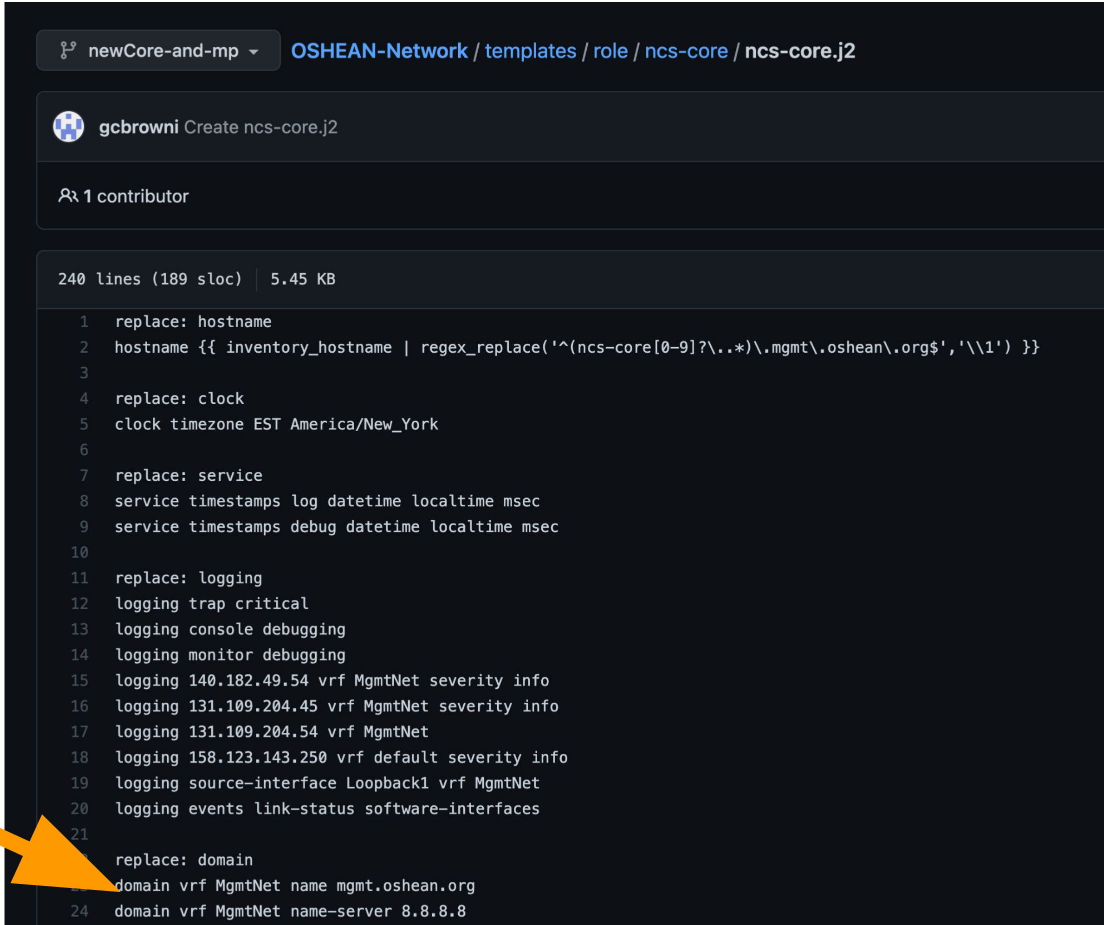
..

ncs-core	Create ncs-core.j2	17 months ago
ncs-mp	Create ncs-mp.j2	17 months ago
rtsw	Adding initial setup	2 years ago
testlab	Update testlab.j2	2 years ago

Lets do a quick example of how we would change something!

Specifically lets change the name-servers for the ncs-core devices!

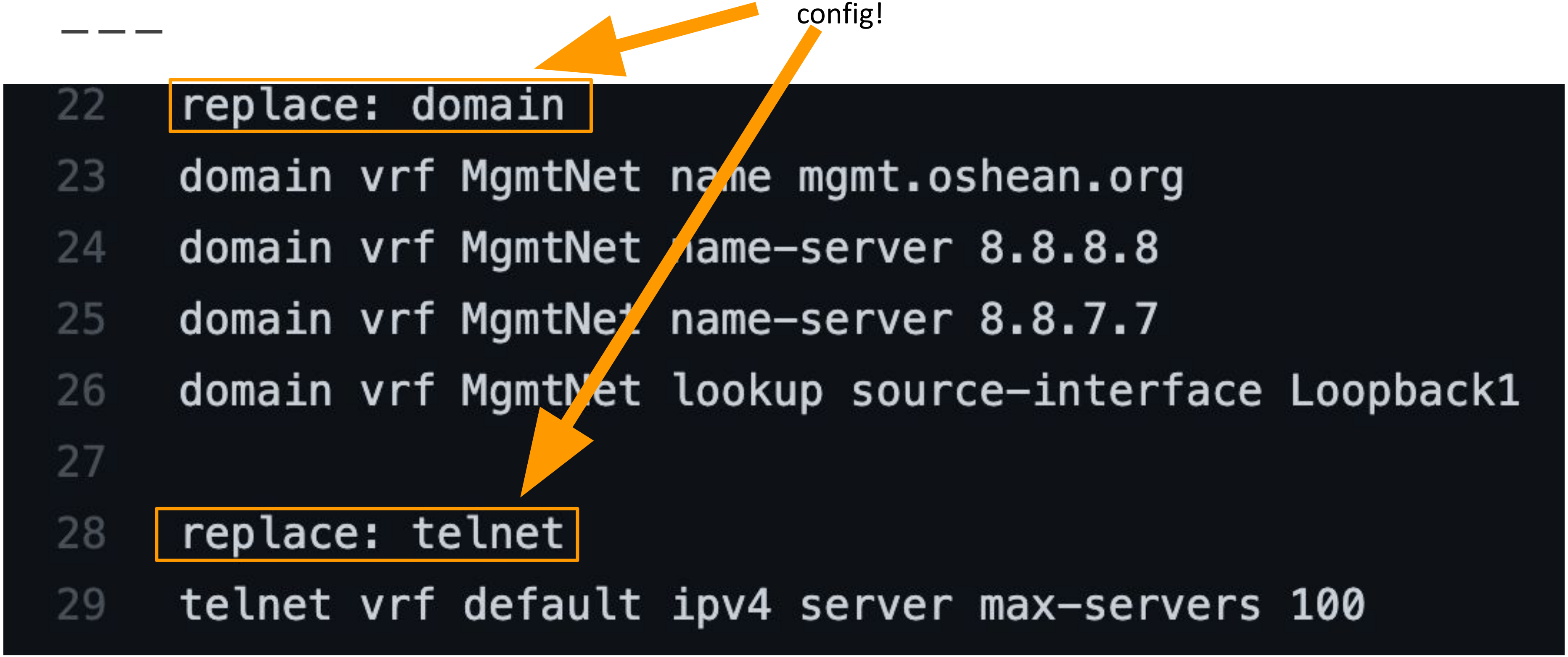
Change this!



newCore-and-mp
OSHEAN-Network / templates / role / ncs-core / ncs-core.j2
gcbrowni Create ncs-core.j2
1 contributor
240 lines (189 sloc) | 5.45 KB

```
replace: hostname
hostname {{ inventory_hostname | regex_replace('^(ncs-core[0-9]?\..*)\.mgmt\.oshean\.org$','\\1') }}

replace: clock
clock timezone EST America/New_York

replace: service
service timestamps log datetime localtime msec
service timestamps debug datetime localtime msec

replace: logging
logging trap critical
logging console debugging
logging monitor debugging
logging 140.182.49.54 vrf MgmtNet severity info
logging 131.109.204.45 vrf MgmtNet severity info
logging 131.109.204.54 vrf MgmtNet
logging 158.123.143.250 vrf default severity info
logging source-interface Loopback1 vrf MgmtNet
logging events link-status software-interfaces

replace: domain
domain vrf MgmtNet name mgmt.oshean.org
domain vrf MgmtNet name-server 8.8.8.8
```

Closer look



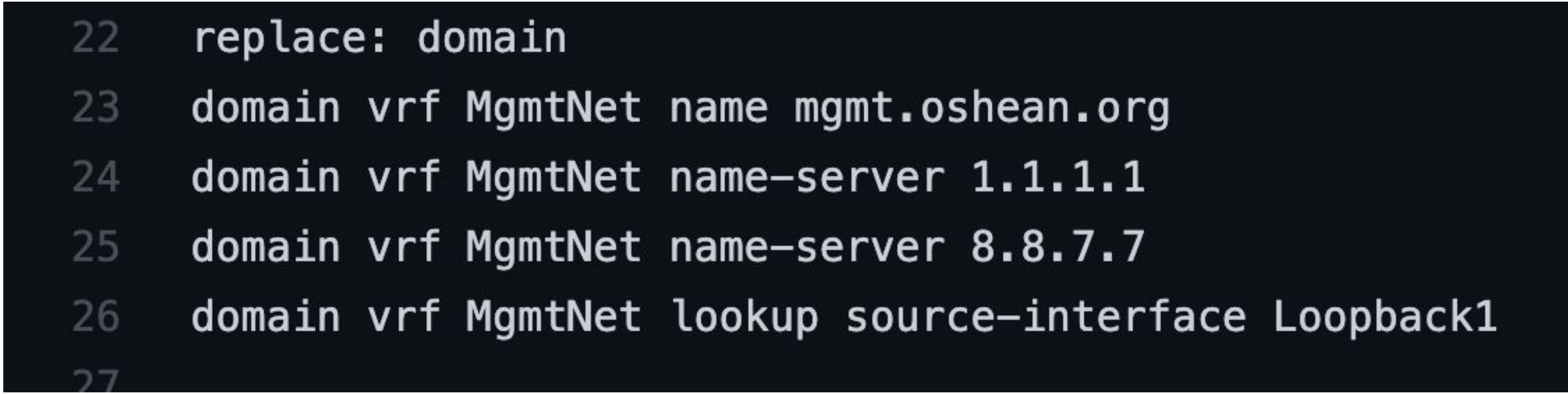
```
replace: domain
domain vrf MgmtNet name mgmt.oshean.org
domain vrf MgmtNet name-server 8.8.8.8
domain vrf MgmtNet name-server 8.8.7.7
domain vrf MgmtNet lookup source-interface Loopback1

replace: telnet
telnet vrf default ipv4 server max-servers 100
```

Hey What is this! Its not valid config!

Replace statement

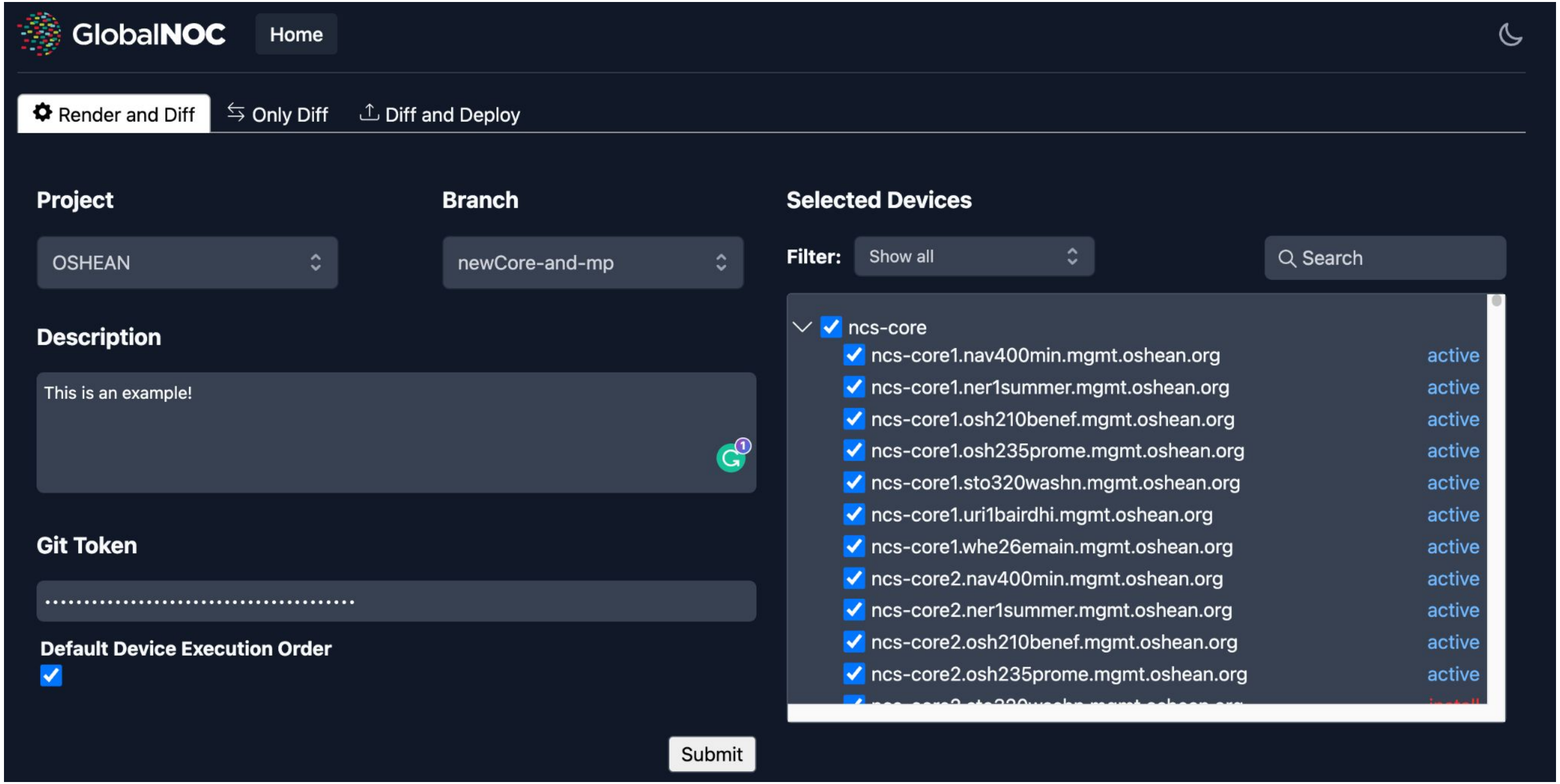
- In IOS-XR and JunOS devices the replace statement operates similarly.
  - In Junos it will replace everything underneath that structure with whatever is specified there
  - In IOS-XR it essentially will append a ‘no’ to every line of the current config that matches (ie… ‘replace domain’ will read the current configuration of the device and find all statements that match ‘domain*’ and prepend them with a ‘no’ before the rest of your config goes in

We can edit the config



```
replace: domain
domain vrf MgmtNet name mgmt.oshean.org
domain vrf MgmtNet name-server 1.1.1.1
domain vrf MgmtNet name-server 8.8.7.7
domain vrf MgmtNet lookup source-interface Loopback1
```

Now with our change we go to the GNAT UI



Project

OSHEAN

Branch

newCore-and-mp

Description

This is an example!

Git Token

••

Default Device Execution Order

☑

Selected Devices

Filter: Show all

Search

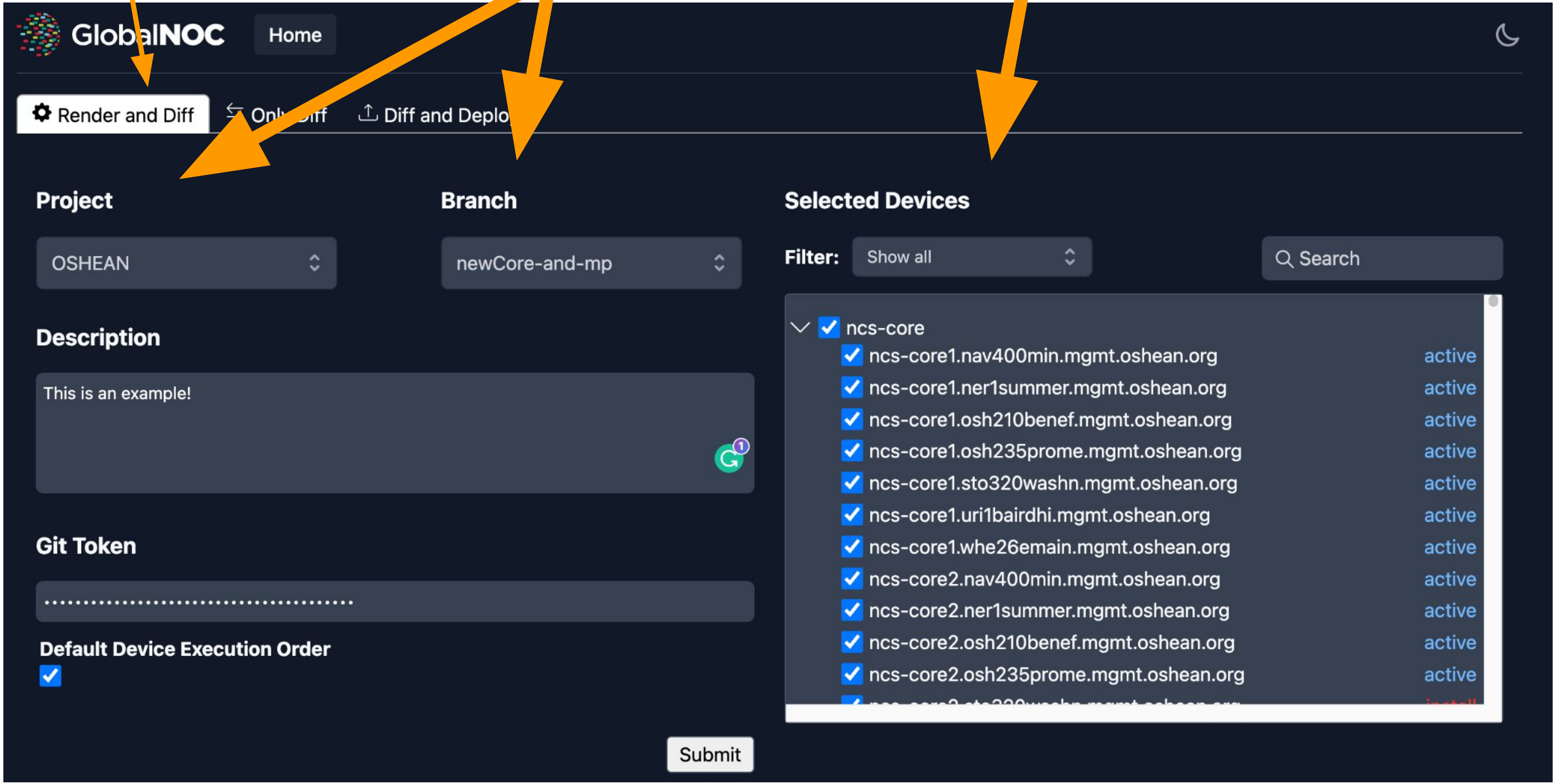
☑ ncs-core

☑ ncs-core1.nav400min.mgmt.oshean.org	active
☑ ncs-core1.ner1summer.mgmt.oshean.org	active
☑ ncs-core1.osh210benef.mgmt.oshean.org	active
☑ ncs-core1.osh235prome.mgmt.oshean.org	active
☑ ncs-core1.sto320washn.mgmt.oshean.org	active
☑ ncs-core1.uri1bairdhi.mgmt.oshean.org	active
☑ ncs-core1.whe26emain.mgmt.oshean.org	active
☑ ncs-core2.nav400min.mgmt.oshean.org	active
☑ ncs-core2.ner1summer.mgmt.oshean.org	active
☑ ncs-core2.osh210benef.mgmt.oshean.org	active
☑ ncs-core2.osh235prome.mgmt.oshean.org	active

Select Render and Diff

Select the Project and Branch

Select Devices to run it on!



Project

OSHEAN

Branch

newCore-and-mp

Description

This is an example!

Git Token

••

Default Device Execution Order

☑

Submit

Selected Devices

Filter: Show all

Search

☑ ncs-core

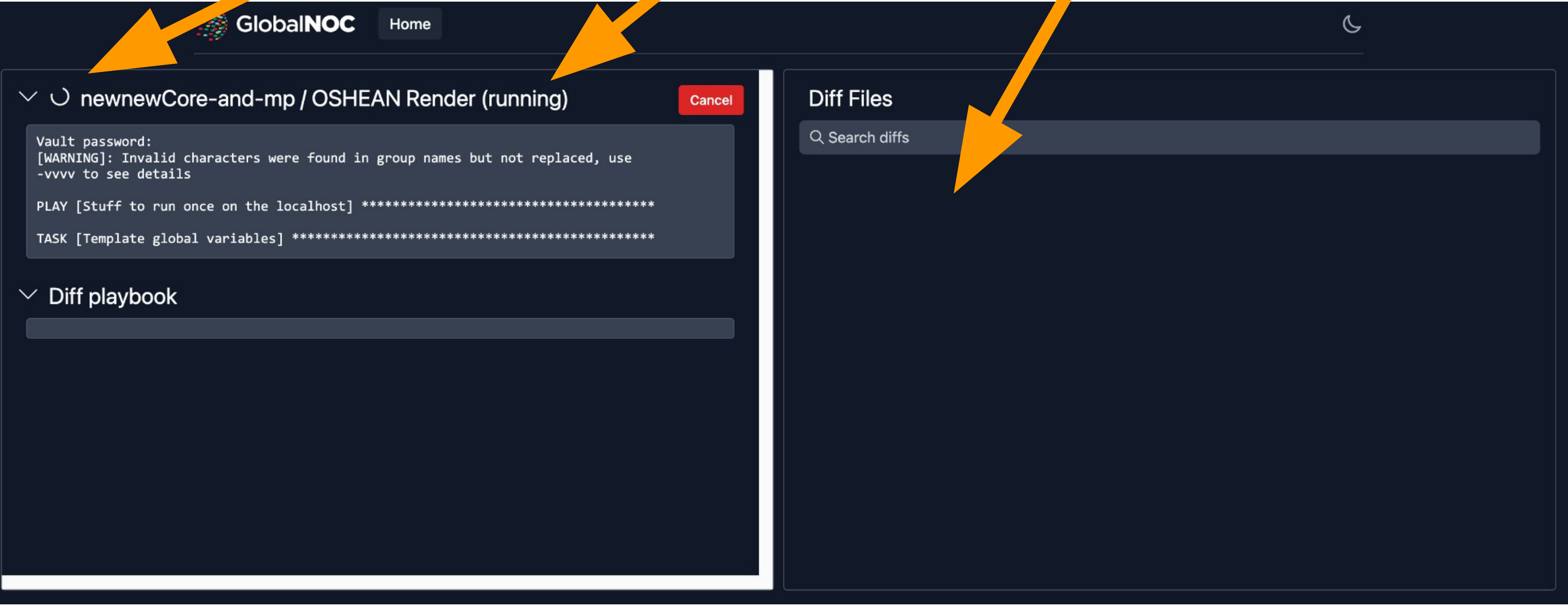
☑ ncs-core1.nav400min.mgmt.oshean.org	active
☑ ncs-core1.ner1summer.mgmt.oshean.org	active
☑ ncs-core1.osh210benef.mgmt.oshean.org	active
☑ ncs-core1.osh235prome.mgmt.oshean.org	active
☑ ncs-core1.sto320washn.mgmt.oshean.org	active
☑ ncs-core1.uri1bairdhi.mgmt.oshean.org	active
☑ ncs-core1.whe26emain.mgmt.oshean.org	active
☑ ncs-core2.nav400min.mgmt.oshean.org	active
☑ ncs-core2.ner1summer.mgmt.oshean.org	active
☑ ncs-core2.osh210benef.mgmt.oshean.org	active
☑ ncs-core2.osh235prome.mgmt.oshean.org	active
☑ ncs-core2.sto320washn.mgmt.oshean.org	install

While it is running

Shows its running, complete or failed

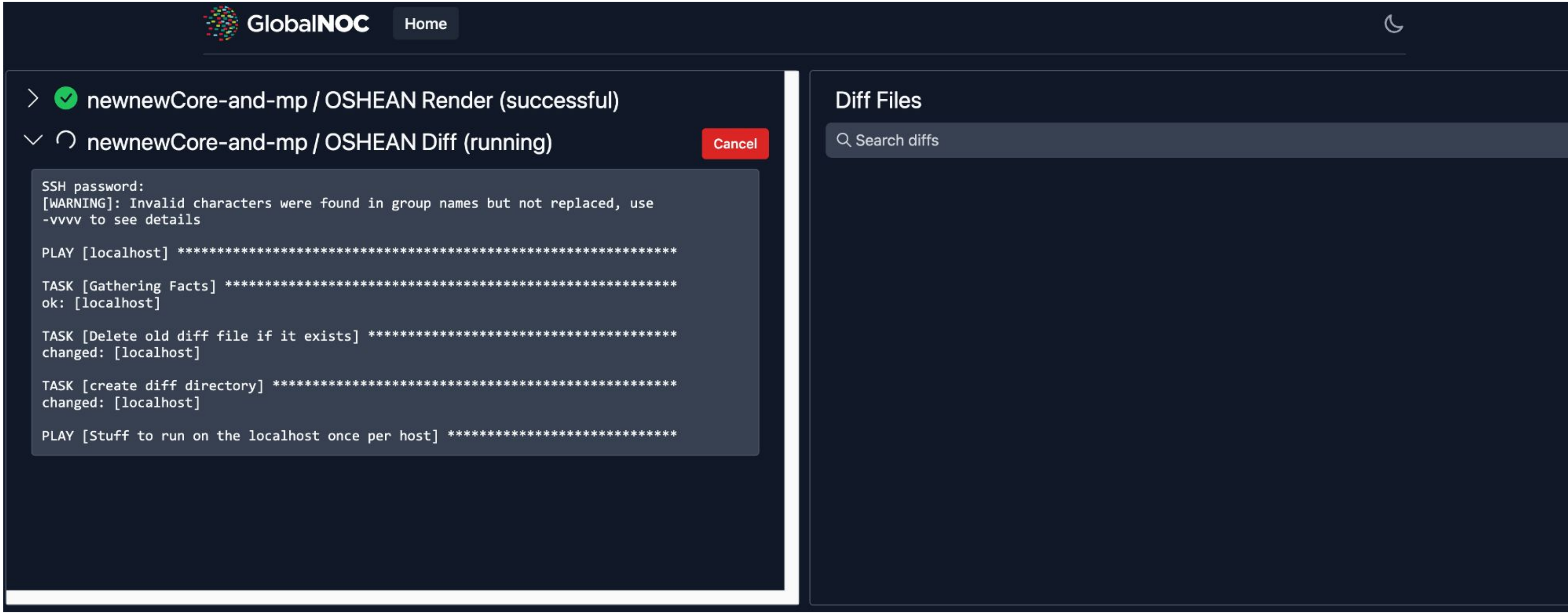
Real-time job status

Resulting diffs will appear here!



```
Vault password:
[WARNING]: Invalid characters were found in group names but not replaced, use
-vvvv to see details

PLAY [Stuff to run once on the localhost] ***************************************

TASK [Template global variables] ************************************************
```

Success running Render - now doing diff

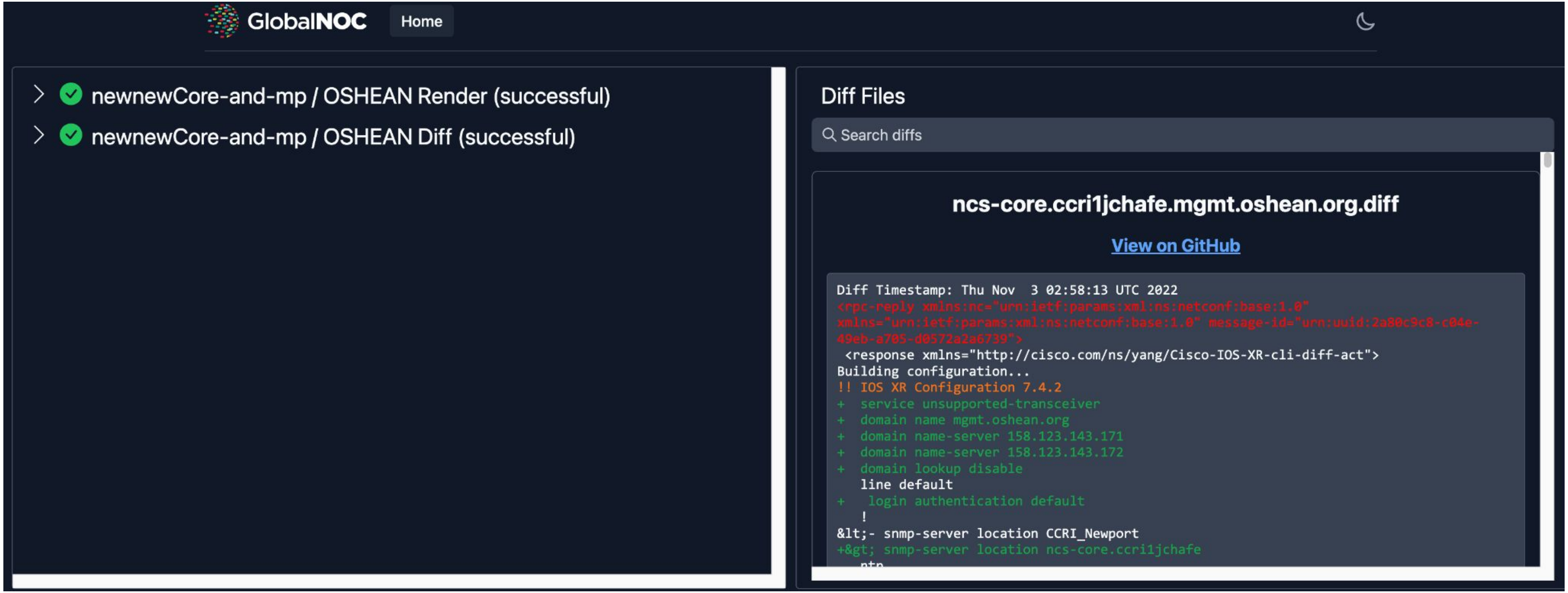


newnewCore-and-mp / OSHEAN Render (successful)

newnewCore-and-mp / OSHEAN Diff (running)

```
SSH password:
[WARNING]: Invalid characters were found in group names but not replaced, use
-vvvv to see details

PLAY [localhost] **************************************************************

TASK [Gathering Facts] ********************************************************
ok: [localhost]

TASK [Delete old diff file if it exists] **************************************
changed: [localhost]

TASK [create diff directory] **************************************************
changed: [localhost]

PLAY [Stuff to run on the localhost once per host] ****************************
```

Finally Results!



newnewCore-and-mp / OSHEAN Render (successful)

newnewCore-and-mp / OSHEAN Diff (successful)

ncs-core.ccri1jchafe.mgmt.oshean.org.diff

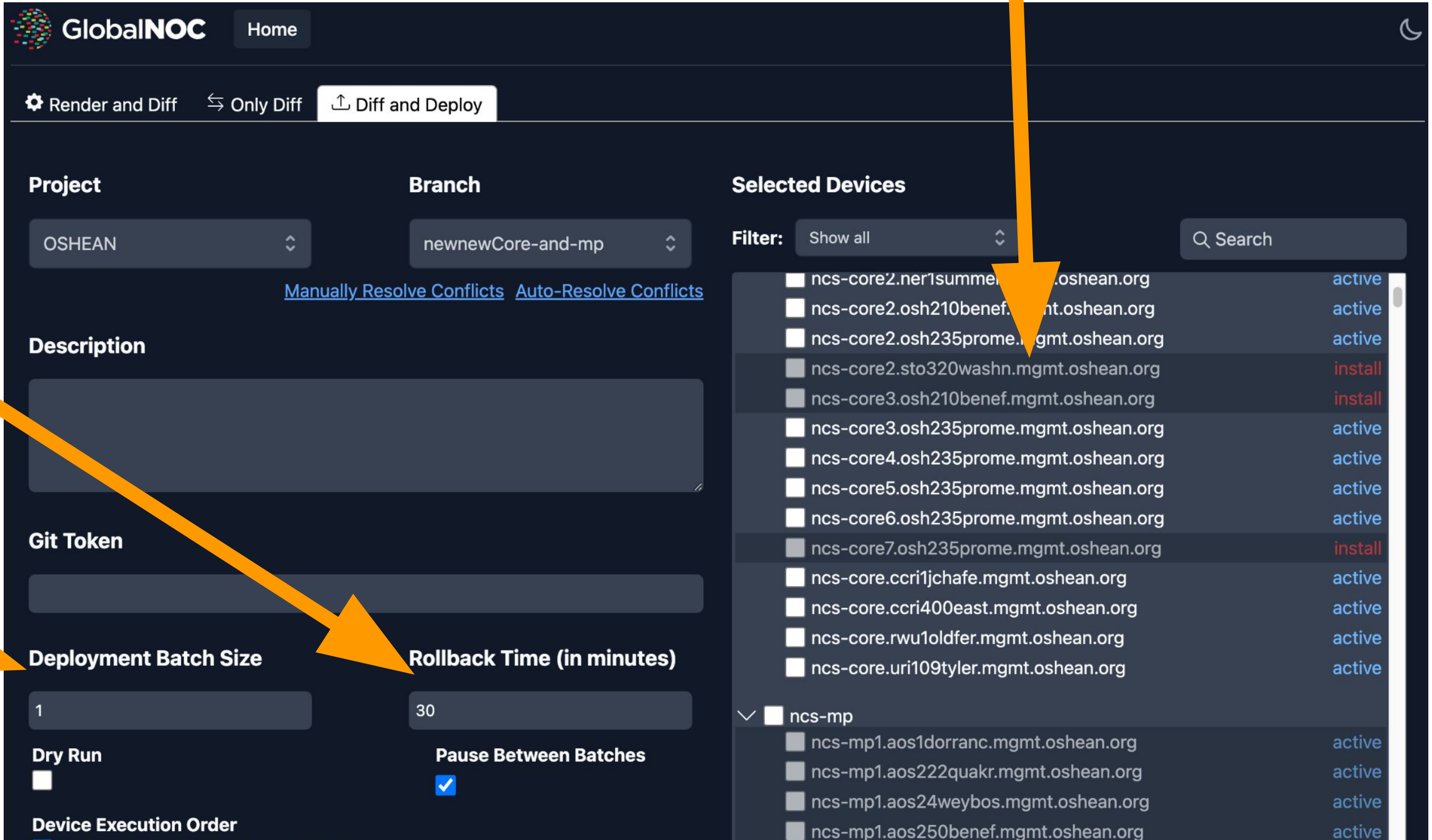
```
Diff Timestamp: Thu Nov  3 02:58:13 UTC 2022
<rpc-reply xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:2a80c9c8-c04e-
49eb-a705-d0572a2a6739">
 <response xmlns="http://cisco.com/ns/yang/Cisco-IOS-XR-cli-diff-act">
Building configuration...
!! IOS XR Configuration 7.4.2
+  service unsupported-transceiver
+  domain name mgmt.oshean.org
+  domain name-server 158.123.143.171
+  domain name-server 158.123.143.172
+  domain lookup disable
   line default
+   login authentication default
   !
&lt;- snmp-server location CCRI_Newport
+&gt; snmp-server location ncs-core.ccri1jchafe
   ntp
```

Lets look at Diff and Deploy

Only devices currently rendered on this branch can be deployed

How long to wait for a confirmation or rollback

Batch size (how to break apart the



Project

OSHEAN

Branch

newnewCore-and-mp

Manually Resolve Conflicts Auto-Resolve Conflicts

Description

Git Token

Deployment Batch Size

1

Rollback Time (in minutes)

30

Dry Run

□

Pause Between Batches

☑

Device Execution Order

Selected Devices

Filter: Show all

Search

□ ncs-core2.ner1summe .oshean.org	active
□ ncs-core2.osh210benef. nt.oshean.org	active
□ ncs-core2.osh235prome. gmt.oshean.org	active
□ ncs-core2.sto320washn.mgmt.oshean.org	install
□ ncs-core3.osh210benef.mgmt.oshean.org	install
□ ncs-core3.osh235prome.mgmt.oshean.org	active
□ ncs-core4.osh235prome.mgmt.oshean.org	active
□ ncs-core5.osh235prome.mgmt.oshean.org	active
□ ncs-core6.osh235prome.mgmt.oshean.org	active
□ ncs-core7.osh235prome.mgmt.oshean.org	install
□ ncs-core.ccri1jchafe.mgmt.oshean.org	active
□ ncs-core.ccri400east.mgmt.oshean.org	active
□ ncs-core.rwu1oldfer.mgmt.oshean.org	active
□ ncs-core.uri109tyler.mgmt.oshean.org	active
□ ncs-mp	
□ ncs-mp1.aos1dorranc.mgmt.oshean.org	active
□ ncs-mp1.aos222quakr.mgmt.oshean.org	active
□ ncs-mp1.aos24weybos.mgmt.oshean.org	active
□ ncs-mp1.aos250benef.mgmt.oshean.org	active

Thats it!

- I know its a lot of information (these slides will all be made available)
- There is a dedicated slack channel where me (and the rest of the NAP team) are available
- You can put in a ticket and we’ll get back with you
- Ultimately GNAT is fully production and the NAP team will work to resolve problems 24x7 and help you complete your maintenances