>>> network.toCode()

Golden Config

Maintaining your configurations with Nautobot

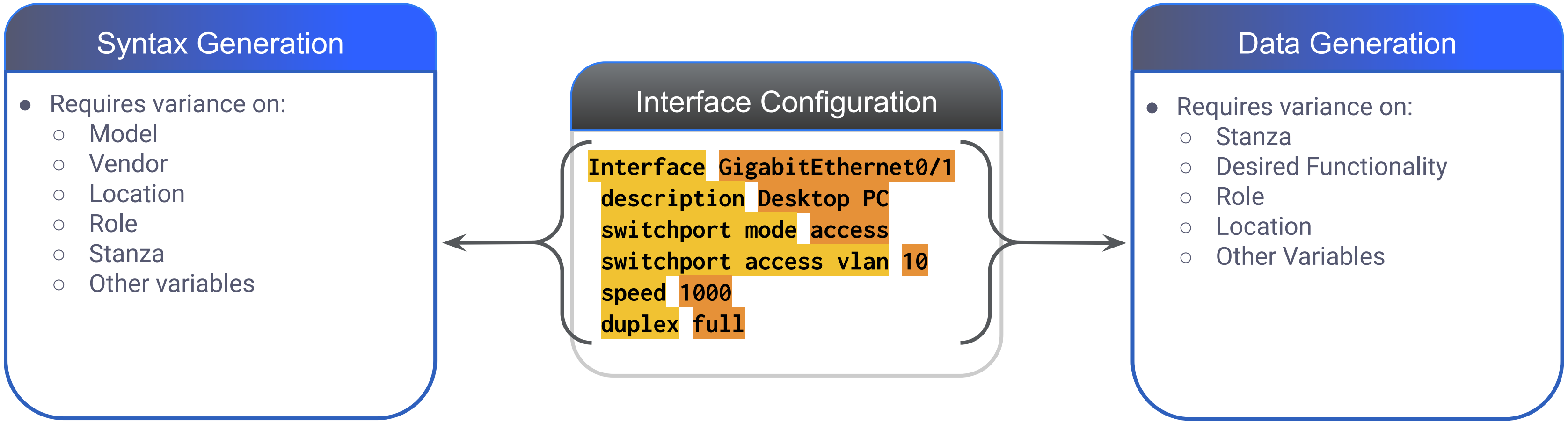
>>> Agenda

- Configuration Compliance Overview
- Implementation Details
- Getting Started
- Demo

Configuration Compliance Overview

Why Maintaining Compliance is Hard

Traditional scripting cannot handle the volume or frequency of adjustments



Syntax Generation

- Requires variance on:
  - Model
  - Vendor
  - Location
  - Role
  - Stanza
  - Other variables

Interface Configuration

```
Interface GigabitEthernet0/1
 description Desktop PC
 switchport mode access
 switchport access vlan 10
 speed 1000
 duplex full
```

Data Generation

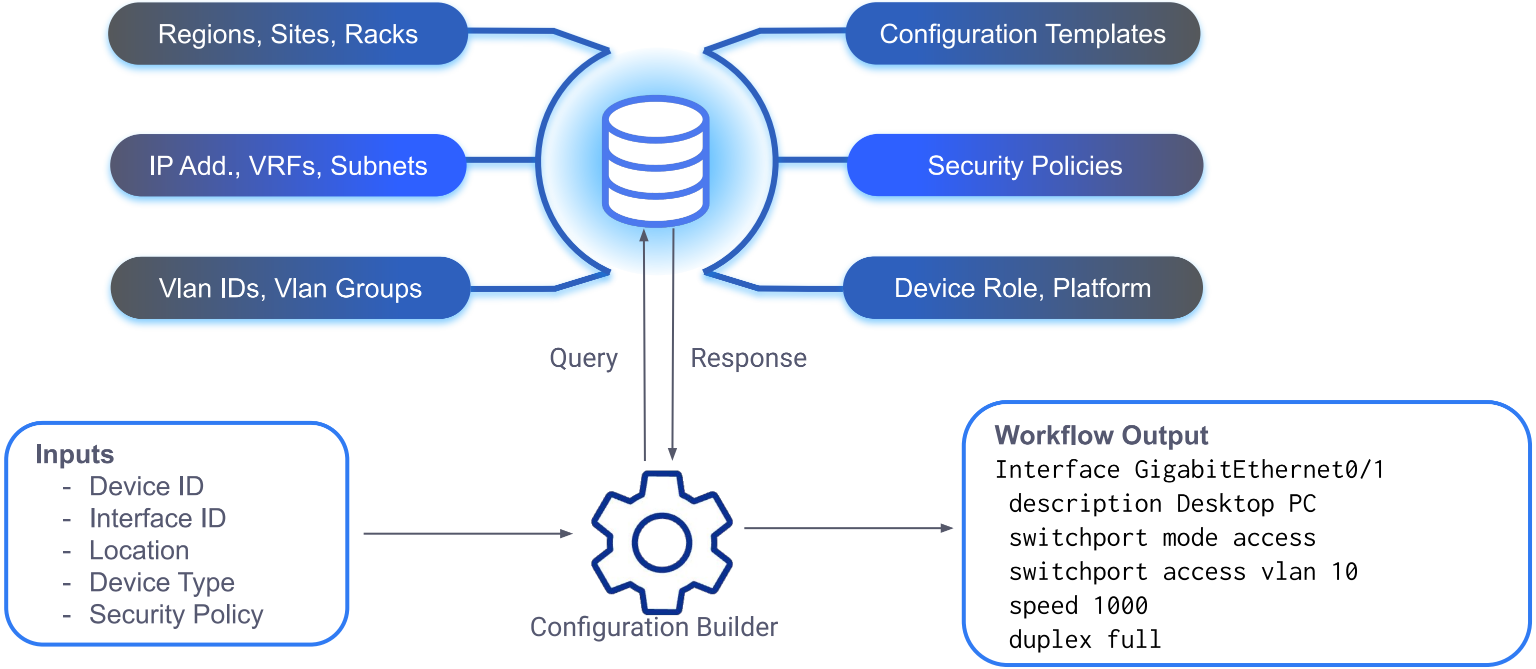
- Requires variance on:
  - Stanza
  - Desired Functionality
  - Role
  - Location
  - Other Variables

- *Must be adjusted after every change*
- *Must accommodate:*
  - *One-off configs*
  - *Variances based on location/site/region/etc*
  - *Variances in syntax (platform/model/vendor/etc)*

How is compliance Accomplished? Part 1

Source of Truth Explained

Data is defined in an authoritative location and retrieved as necessary



Regions, Sites, Racks

IP Add., VRFs, Subnets

Vlan IDs, Vlan Groups

Configuration Templates

Security Policies

Device Role, Platform

Query

Response

Inputs

- Device ID
- Interface ID
- Location
- Device Type
- Security Policy

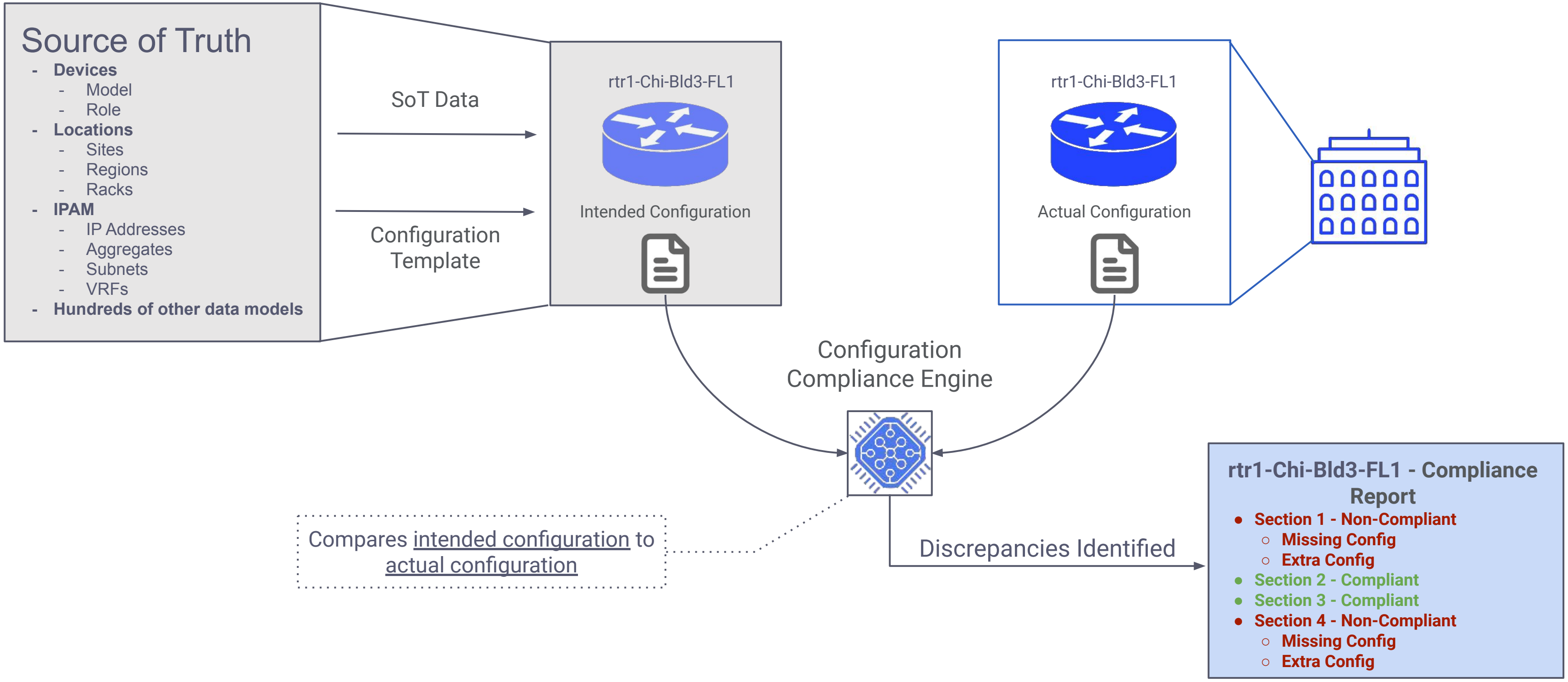
Configuration Builder

Workflow Output

```
Interface GigabitEthernet0/1
 description Desktop PC
 switchport mode access
 switchport access vlan 10
 speed 1000
 duplex full
```

>>> How is compliance Accomplished? Part 2

Standardization & Continuous Verification



Source of Truth

- **Devices**
  - Model
  - Role
- **Locations**
  - Sites
  - Regions
  - Racks
- **IPAM**
  - IP Addresses
  - Aggregates
  - Subnets
  - VRFs
- **Hundreds of other data models**

SoT Data

Configuration Template

rtr1-Chi-Bld3-FL1

Intended Configuration

rtr1-Chi-Bld3-FL1

Actual Configuration

Configuration Compliance Engine

Compares intended configuration to actual configuration

Discrepancies Identified

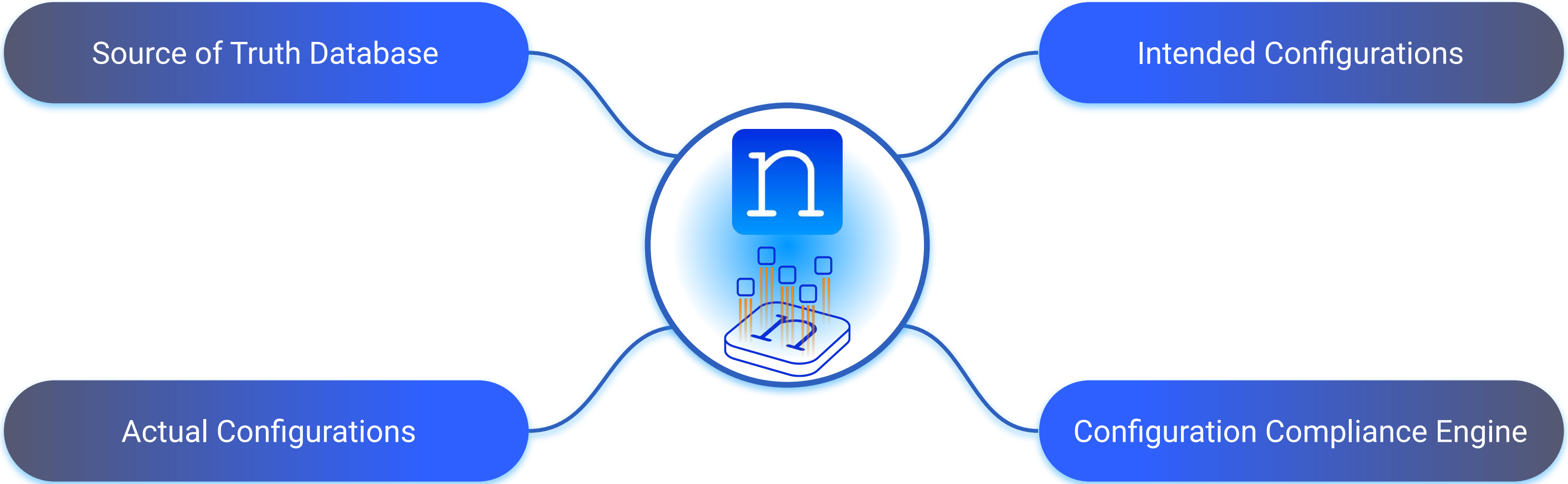
rtr1-Chi-Bld3-FL1 - Compliance Report

- **Section 1 - Non-Compliant**
  - **Missing Config**
  - **Extra Config**
- **Section 2 - Compliant**
- **Section 3 - Compliant**
- **Section 4 - Non-Compliant**
  - **Missing Config**
  - **Extra Config**

Nautobot’s Solution

>>> Nautobots Holistic Solution

Nautobot is pre-packaged with the essential technologies



Source of Truth Database

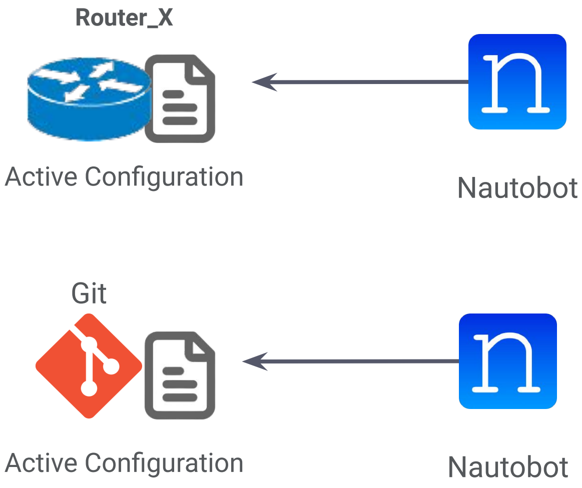
Intended Configurations

Actual Configurations

Configuration Compliance Engine

>>> Nautobot Actual Configuration

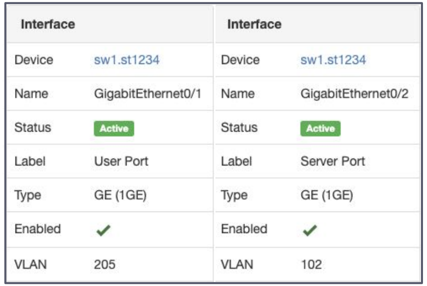
Nautobot captures the actual configuration from the Device Itself or your existing backups



>>> Nautobot Intended Configurations

Using Jinja and data to generate intended configurations

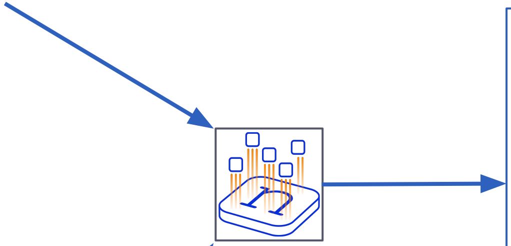
SoT
Database



Interface		Interface	
Device	sw1.st1234	Device	sw1.st1234
Name	GigabitEthernet0/1	Name	GigabitEthernet0/2
Status	Active	Status	Active
Label	User Port	Label	Server Port
Type	GE (1GE)	Type	GE (1GE)
Enabled	✓	Enabled	✓
VLAN	205	VLAN	102

Jinja

```
{% for item in cfg_vars %}
interface {{ item['interface'] }}
 description {{ item['description'] }}
 switchport mode access
 switchport mode access vlan {{ item['vlan'] }}
 spanning-tree portfast
 spanning-tree guard root
{% endfor %}
```



Golden
Configuration App

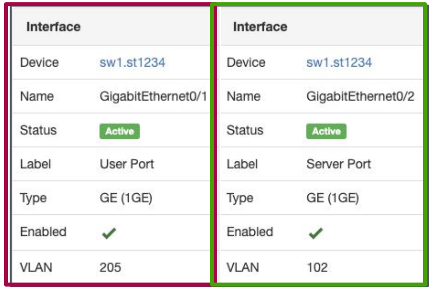
Intended Configurations

```
interface GigabitEthernet 0/1
 description User Port
 switchport mode access
 switchport mode access vlan 205
 spanning-tree portfast
 spanning-tree guard root
interface GigabitEthernet 0/2
 description Server Port
 switchport mode access
 switchport mode access vlan 102
 spanning-tree portfast
 spanning-tree guard root
```

>>> Nautobot Intended Configurations

Using Jinja and data to generate intended configurations

SoT
Database

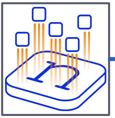


Interface	
Device	sw1.st1234
Name	GigabitEthernet0/1
Status	Active
Label	User Port
Type	GE (1GE)
Enabled	✓
VLAN	205

Interface	
Device	sw1.st1234
Name	GigabitEthernet0/2
Status	Active
Label	Server Port
Type	GE (1GE)
Enabled	✓
VLAN	102

Jinja

```
{% for item in cfg_vars %}
interface {{ item[‘interface’] }}
 description {{ item[‘description’] }}
 switchport mode access
 switchport mode access vlan {{ item[‘vlan’] }}
 spanning-tree portfast
 spanning-tree guard root
{% endfor %}
```

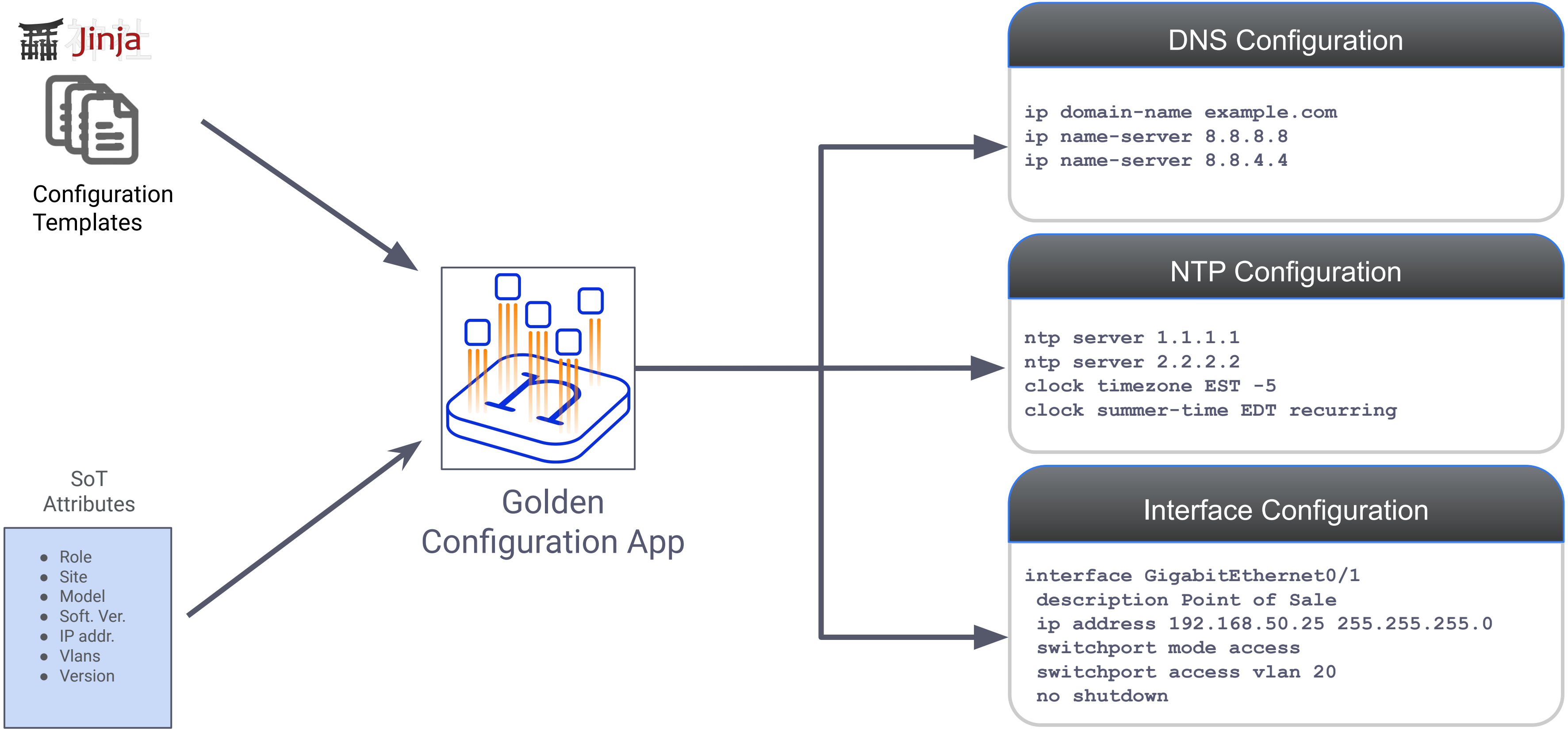


Golden
Configuration App

Intended Configurations

```
interface GigabitEthernet 0/1
 description User Port
 switchport mode access
 switchport mode access vlan 205
 spanning-tree portfast
 spanning-tree guard root
interface GigabitEthernet 0/2
 description Server Port
 switchport mode access
 switchport mode access vlan 102
 spanning-tree portfast
 spanning-tree guard root
```

Nautobot Intended Configurations Con’t



Configuration Templates

SoT Attributes

- Role
- Site
- Model
- Soft. Ver.
- IP addr.
- Vlans
- Version

Golden Configuration App

DNS Configuration

```
ip domain-name example.com
ip name-server 8.8.8.8
ip name-server 8.8.4.4
```

NTP Configuration

```
ntp server 1.1.1.1
ntp server 2.2.2.2
clock timezone EST -5
clock summer-time EDT recurring
```

Interface Configuration

```
interface GigabitEthernet0/1
 description Point of Sale
 ip address 192.168.50.25 255.255.255.0
 switchport mode access
 switchport access vlan 20
 no shutdown
```

Nautobot Configuration Compliance Engine

Comparing Actual Configurations to Intended Configurations

Intended Interface Configuration

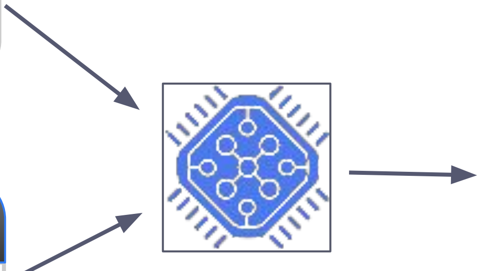
```
interface GigabitEthernet0/1
 description Point of Sale
 ip address 192.168.30.25 255.255.255.0
 switchport mode access
 switchport access vlan 30
 no shutdown
```

Obtained from Golden Config app

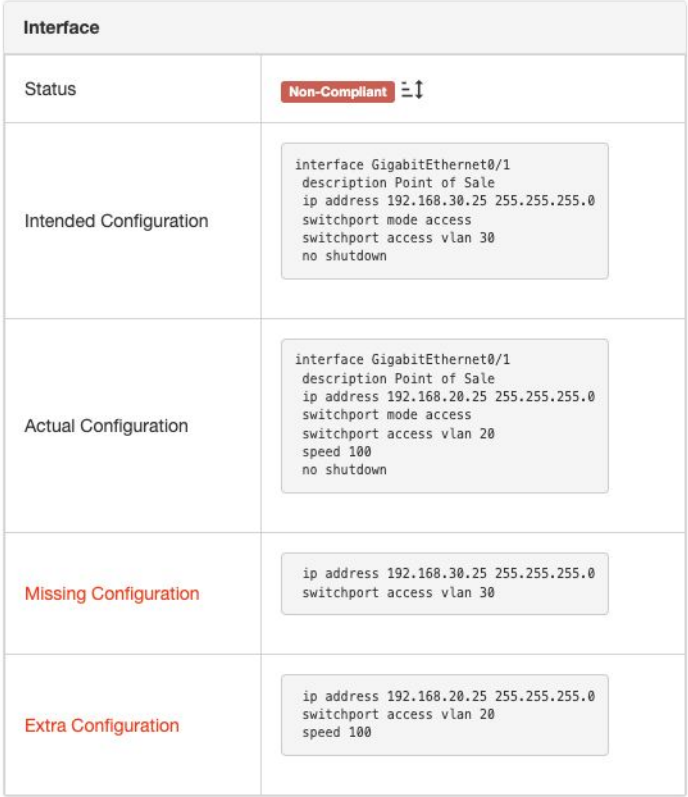
Actual Interface Configuration

```
interface GigabitEthernet0/1
 description Point of Sale
 ip address 192.168.20.25 255.255.255.0
 switchport mode access
 switchport access vlan 20
 speed 100
 no shutdown
```

Obtained from backup



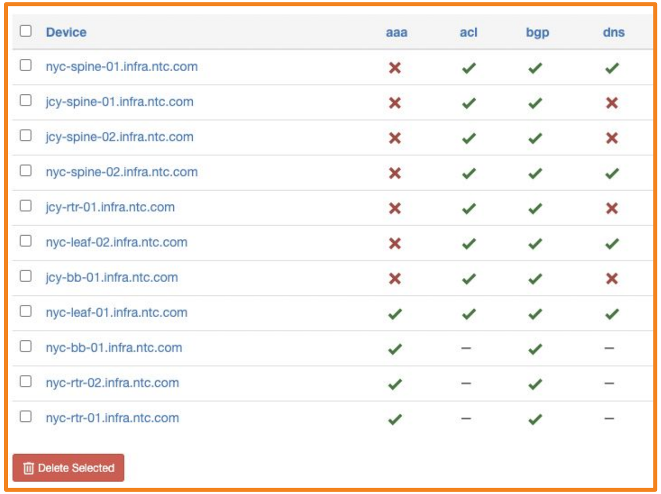
Configuration Compliance Engine



Interface	
Status	Non-Compliant
Intended Configuration	interface GigabitEthernet0/1 description Point of Sale ip address 192.168.30.25 255.255.255.0 switchport mode access switchport access vlan 30 no shutdown
Actual Configuration	interface GigabitEthernet0/1 description Point of Sale ip address 192.168.20.25 255.255.255.0 switchport mode access switchport access vlan 20 speed 100 no shutdown
Missing Configuration	ip address 192.168.30.25 255.255.255.0 switchport access vlan 30
Extra Configuration	ip address 192.168.20.25 255.255.255.0 switchport access vlan 20 speed 100

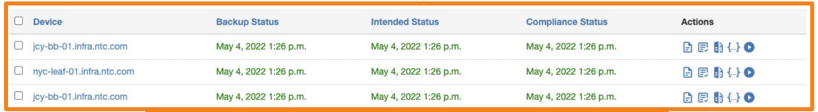
Golden Configuration - UI Screenshots

Dashboard View



☐	Device	aaa	acl	bgp	dns
☐	nyc-spine-01.infra.ntc.com	✗	✓	✓	✓
☐	jcy-spine-01.infra.ntc.com	✗	✓	✓	✗
☐	jcy-spine-02.infra.ntc.com	✗	✓	✓	✗
☐	nyc-spine-02.infra.ntc.com	✗	✓	✓	✓
☐	jcy-rtr-01.infra.ntc.com	✗	✓	✓	✗
☐	nyc-leaf-02.infra.ntc.com	✗	✓	✓	✓
☐	jcy-bb-01.infra.ntc.com	✗	✓	✓	✗
☐	nyc-leaf-01.infra.ntc.com	✓	✓	✓	✓
☐	nyc-bb-01.infra.ntc.com	✓	–	✓	–
☐	nyc-rtr-02.infra.ntc.com	✓	–	✓	–
☐	nyc-rtr-01.infra.ntc.com	✓	–	✓	–

Status Page



☐	Device	Backup Status	Intended Status	Compliance Status	Actions
☐	jcy-bb-01.infra.ntc.com	May 4, 2022 1:26 p.m.	May 4, 2022 1:26 p.m.	May 4, 2022 1:26 p.m.	
☐	nyc-leaf-01.infra.ntc.com	May 4, 2022 1:26 p.m.	May 4, 2022 1:26 p.m.	May 4, 2022 1:26 p.m.	
☐	jcy-bb-01.infra.ntc.com	May 4, 2022 1:26 p.m.	May 4, 2022 1:26 p.m.	May 4, 2022 1:26 p.m.	

Backup Config

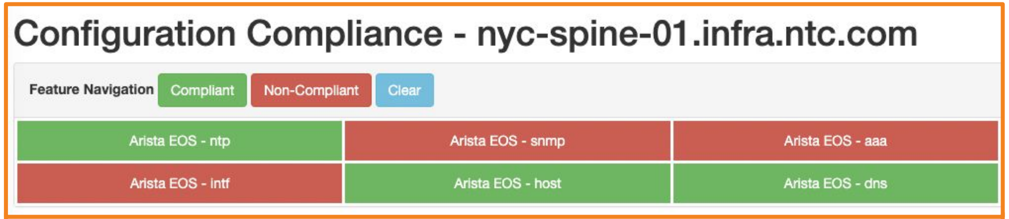
{...} Aggregate Data

Intended Config

Run Job

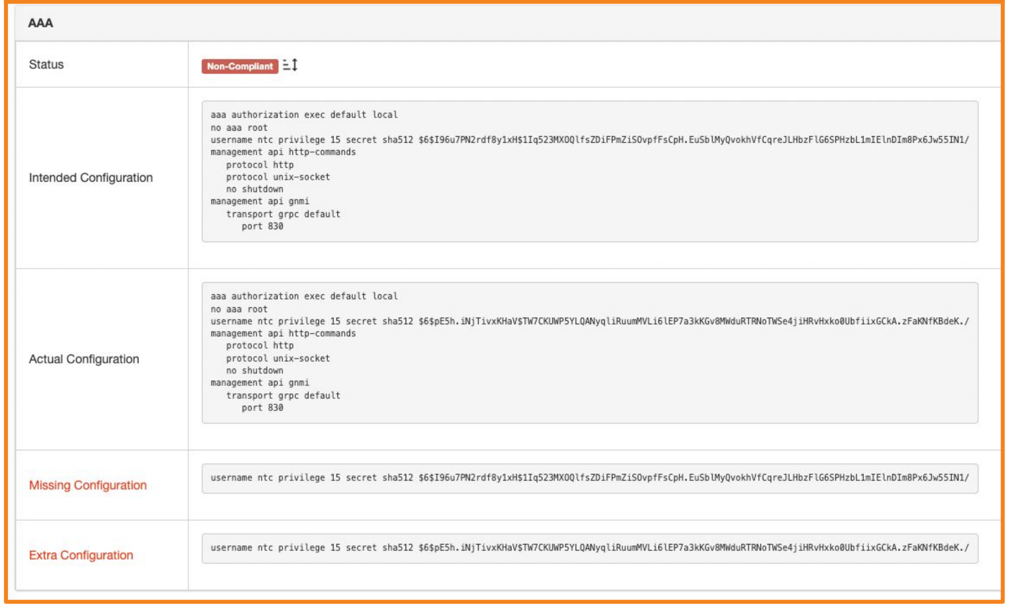
Compliance Details

Device Compliance Views



Configuration Compliance - nyc-spine-01.infra.ntc.com

🟩 Arista EOS - ntp	🟥 Arista EOS - snmp	🟥 Arista EOS - aaa
🟥 Arista EOS - intf	🟩 Arista EOS - host	🟩 Arista EOS - dns



AAA

Status: Non-Compliant

Intended Configuration

```
aaa authorization exec default local
no aaa root
username ntc privilege 15 secret sha512 $6$I96u7PN2rdf8y1xH$1Iq523MXOQlfsZDiFPmZiSOvpfFsCpH.EuSblMyQvokhVfCqreJLHbzFlG6SPHzbL1mIElnDIm8Px6Jw55IN1/
management api http-commands
   protocol http
   protocol unix-socket
   no shutdown
management api gnmi
   transport grpc default
      port 830
```

Actual Configuration

```
aaa authorization exec default local
no aaa root
username ntc privilege 15 secret sha512 $6$pE5h.iNjTivxKHaV$TW7CKUWP5YLQANyqliRuumMVLi6lEP7a3kKGv8MWduRTRNoTWSe4jiHRvHxko0UbfiixGCkA.zFaKNfKBdeK./
management api http-commands
   protocol http
   protocol unix-socket
   no shutdown
management api gnmi
   transport grpc default
      port 830
```

Missing Configuration

```
username ntc privilege 15 secret sha512 $6$I96u7PN2rdf8y1xH$1Iq523MXOQlfsZDiFPmZiSOvpfFsCpH.EuSblMyQvokhVfCqreJLHbzFlG6SPHzbL1mIElnDIm8Px6Jw55IN1/
```

Extra Configuration

```
username ntc privilege 15 secret sha512 $6$pE5h.iNjTivxKHaV$TW7CKUWP5YLQANyqliRuumMVLi6lEP7a3kKGv8MWduRTRNoTWSe4jiHRvHxko0UbfiixGCkA.zFaKNfKBdeK./
```

Getting Started

Getting Started is Easy

Start by modeling simple stanzas and venture out from there

Basic Features

- DNS
- NTP
- Logging
- Hostname

Intermediate Features

- Interfaces
- SNMP
- AAA
- Vlans

Platforms

- Cisco IOS
- Ruckus Switches
- Ruckus APs
- Ruckus vSZ
- Mikrotik

Golden Config - Demo

>>> network.toCode()

Thank you

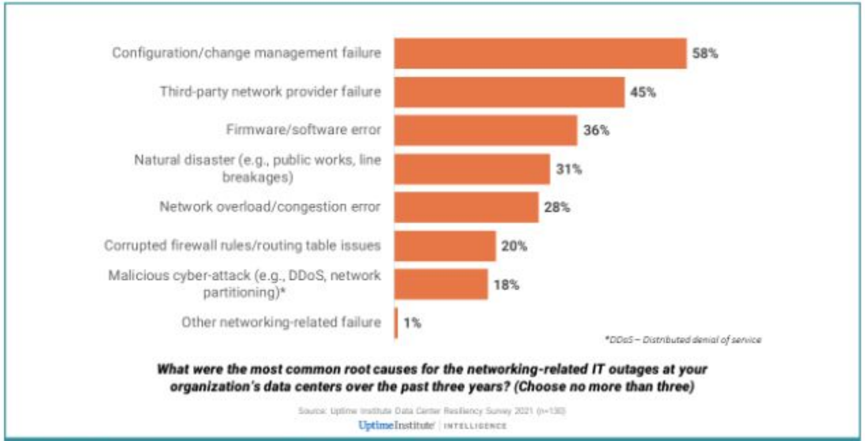
Bonus

Configuration Remediation Overview

>>> Configuration Remediation Overview

Motivation - Automation Drivers

- ❏ Consistency and Standardization
- ❏ Time and Resource Efficiency
- ❏ Increased Compliance
- ❏ Rapid Response to Security Threats
- ❏ Error Reduction and Risk Mitigation
- ❏ Scalability
- ❏ Auditing and Reporting

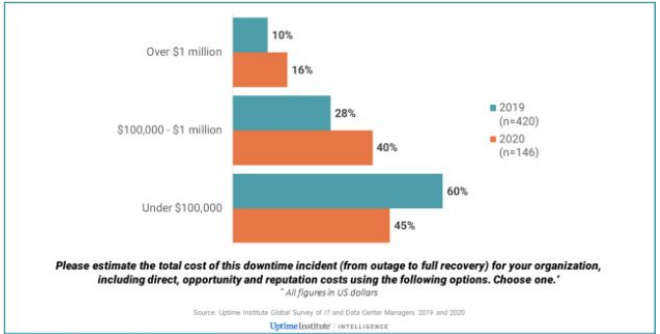


*DDoS – Distributed denial of service

What were the most common root causes for the networking-related IT outages at your organization's data centers over the past three years? (Choose no more than three)

Source: Uptime Institute Data Center Resiliency Survey 2021 (n=130)

Figure 11. **Most common causes of networking-related IT outages**



Please estimate the total cost of this downtime incident (from outage to full recovery) for your organization, including direct, opportunity and reputation costs using the following options. Choose one.*

** All figures in US dollars*

Source: Uptime Institute Global Survey of IT and Data Center Managers 2019 and 2020

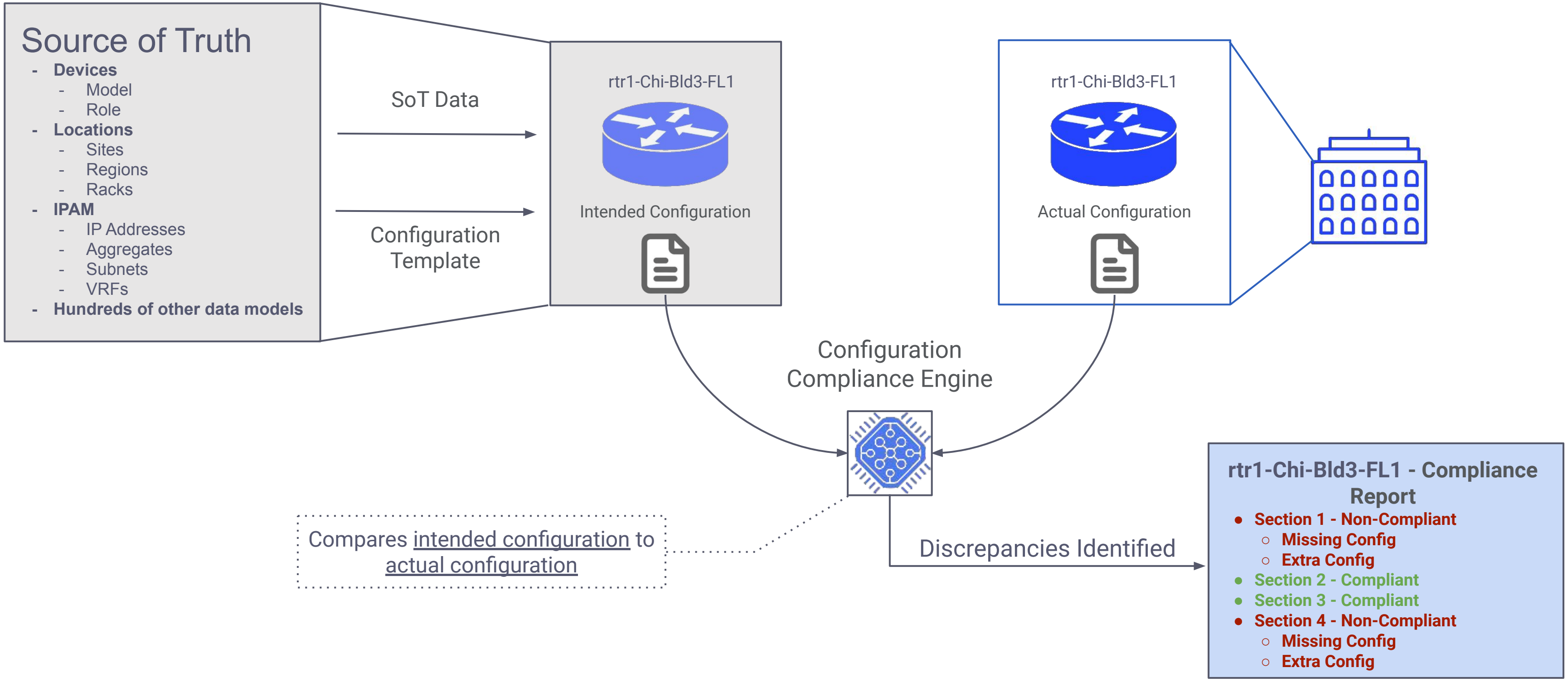
Figure 13. **Outage costs are rising**

Source: Uptime Institute

Achieving Compliance with Automated Remediation

>>> Configuration Remediation Overview

My Network device configurations are non-compliant, what now?



Source of Truth

- **Devices**
  - Model
  - Role
- **Locations**
  - Sites
  - Regions
  - Racks
- **IPAM**
  - IP Addresses
  - Aggregates
  - Subnets
  - VRFs
- **Hundreds of other data models**

SoT Data

Configuration Template

rtr1-Chi-Bld3-FL1

Intended Configuration

rtr1-Chi-Bld3-FL1

Actual Configuration

Configuration Compliance Engine

Compares intended configuration to actual configuration

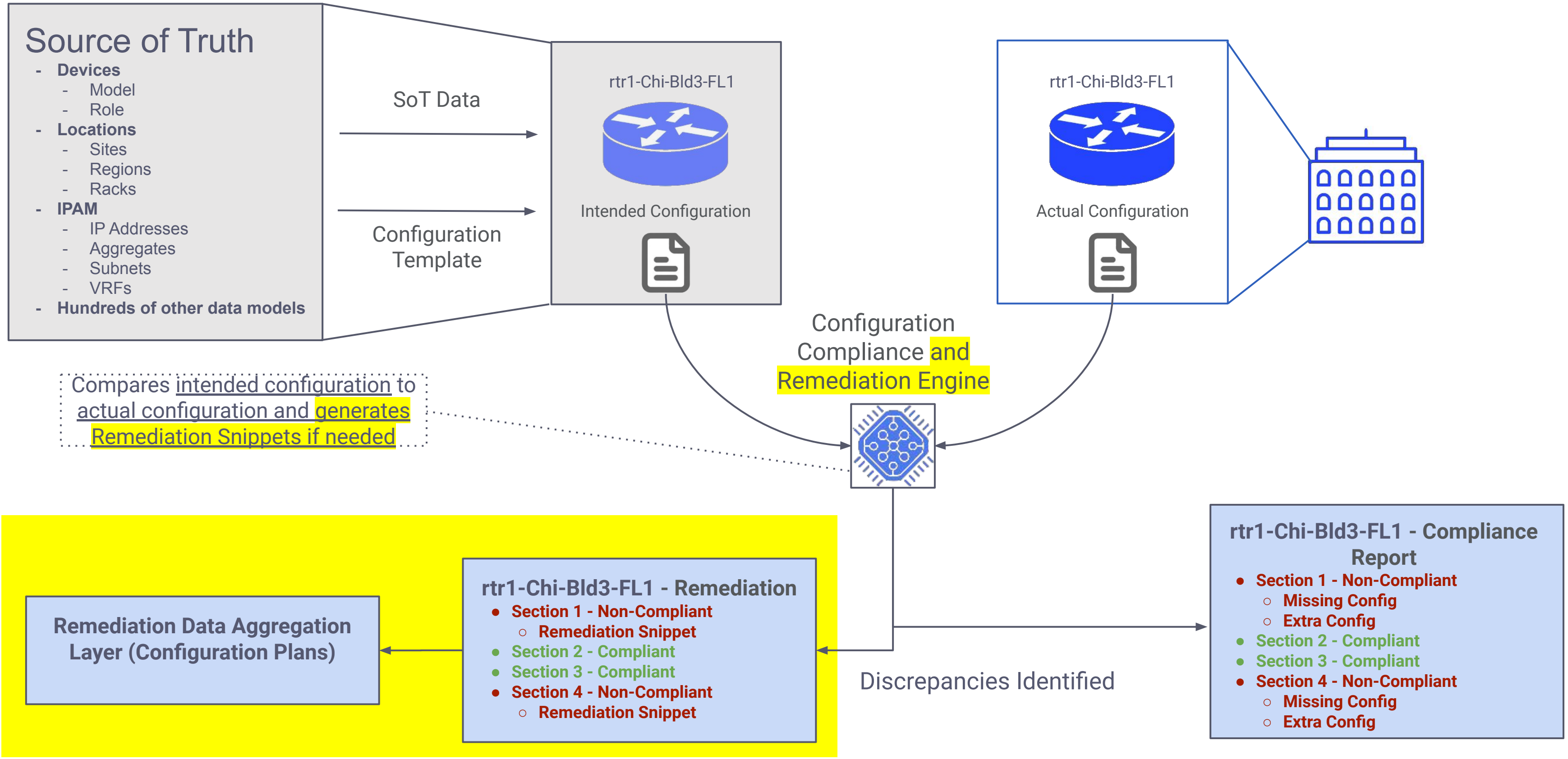
Discrepancies Identified

rtr1-Chi-Bld3-FL1 - Compliance Report

- **Section 1 - Non-Compliant**
  - **Missing Config**
  - **Extra Config**
- **Section 2 - Compliant**
- **Section 3 - Compliant**
- **Section 4 - Non-Compliant**
  - **Missing Config**
  - **Extra Config**

Configuration Remediation Overview

My Network device configurations are non-compliant, what now?



Source of Truth

- **Devices**
  - Model
  - Role
- **Locations**
  - Sites
  - Regions
  - Racks
- **IPAM**
  - IP Addresses
  - Aggregates
  - Subnets
  - VRFs
- **Hundreds of other data models**

SoT Data

Configuration Template

rtr1-Chi-Bld3-FL1

Intended Configuration

rtr1-Chi-Bld3-FL1

Actual Configuration

Configuration Compliance and Remediation Engine

Compares intended configuration to actual configuration and generates Remediation Snippets if needed

Discrepancies Identified

rtr1-Chi-Bld3-FL1 - Compliance Report

- **Section 1 - Non-Compliant**
  - **Missing Config**
  - **Extra Config**
- **Section 2 - Compliant**
- **Section 3 - Compliant**
- **Section 4 - Non-Compliant**
  - **Missing Config**
  - **Extra Config**

rtr1-Chi-Bld3-FL1 - Remediation

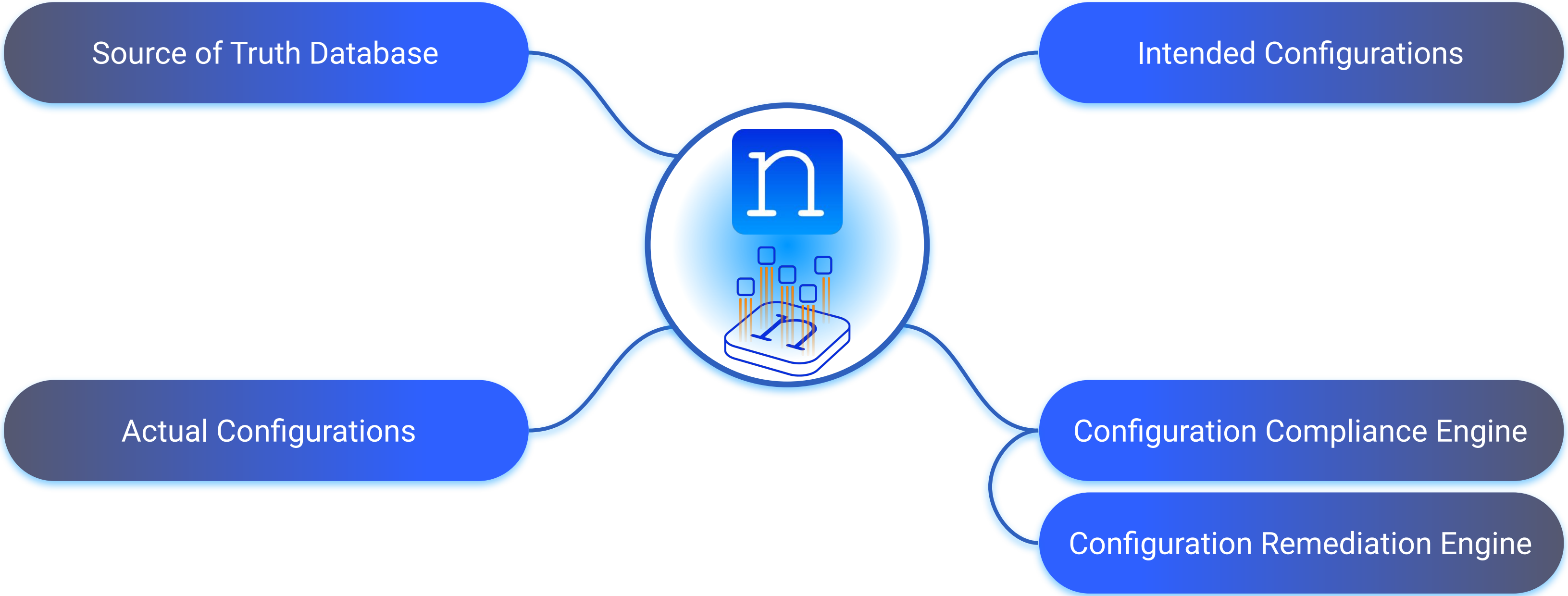
- **Section 1 - Non-Compliant**
  - **Remediation Snippet**
- **Section 2 - Compliant**
- **Section 3 - Compliant**
- **Section 4 - Non-Compliant**
  - **Remediation Snippet**

Remediation Data Aggregation Layer (Configuration Plans)

Nautobot’s Solution

>>> Nautobots Holistic Solution

Nautobot is pre-packaged with the essential technologies



Source of Truth Database

Intended Configurations

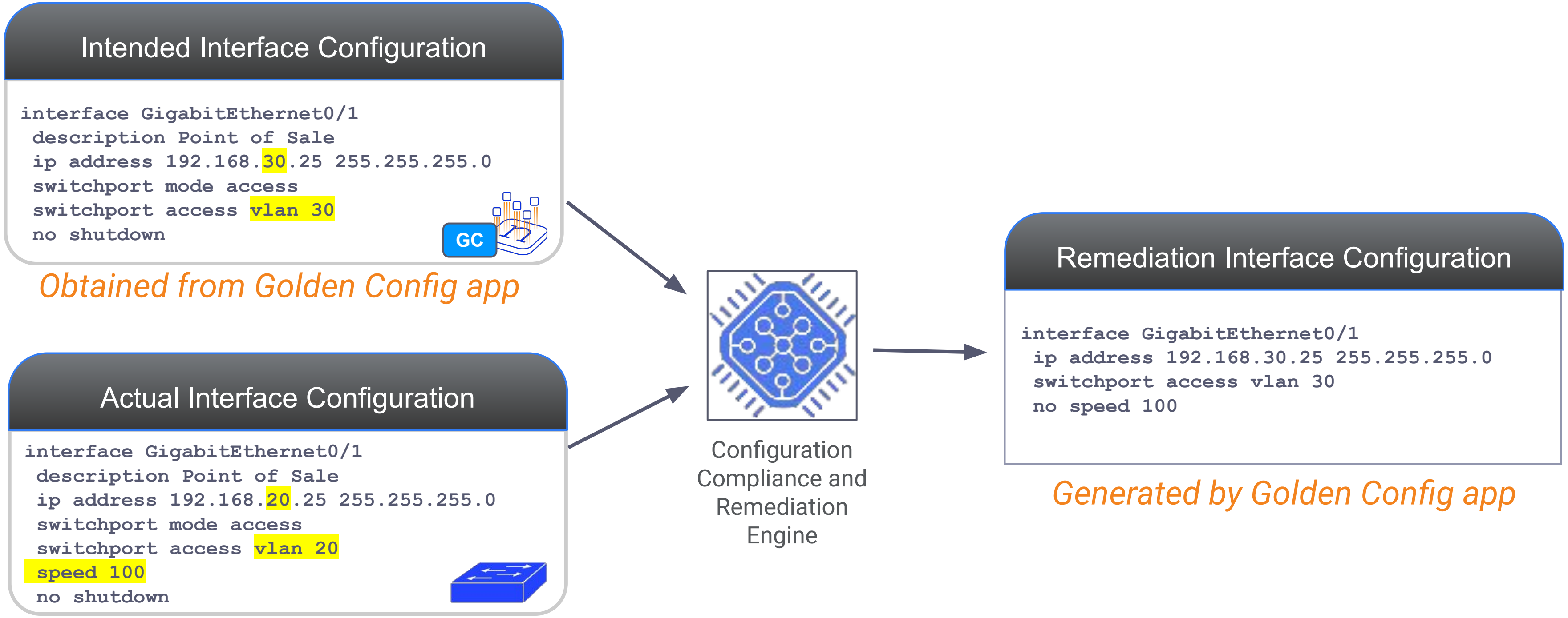
Actual Configurations

Configuration Compliance Engine

Configuration Remediation Engine

Nautobot Configuration Remediation Engine

Creating Remediation Configs from Actual and Intended



Intended Interface Configuration

```
interface GigabitEthernet0/1
 description Point of Sale
 ip address 192.168.30.25 255.255.255.0
 switchport mode access
 switchport access vlan 30
 no shutdown
```

Obtained from Golden Config app

Actual Interface Configuration

```
interface GigabitEthernet0/1
 description Point of Sale
 ip address 192.168.20.25 255.255.255.0
 switchport mode access
 switchport access vlan 20
 speed 100
 no shutdown
```

Obtained from backup

Configuration Compliance and Remediation Engine

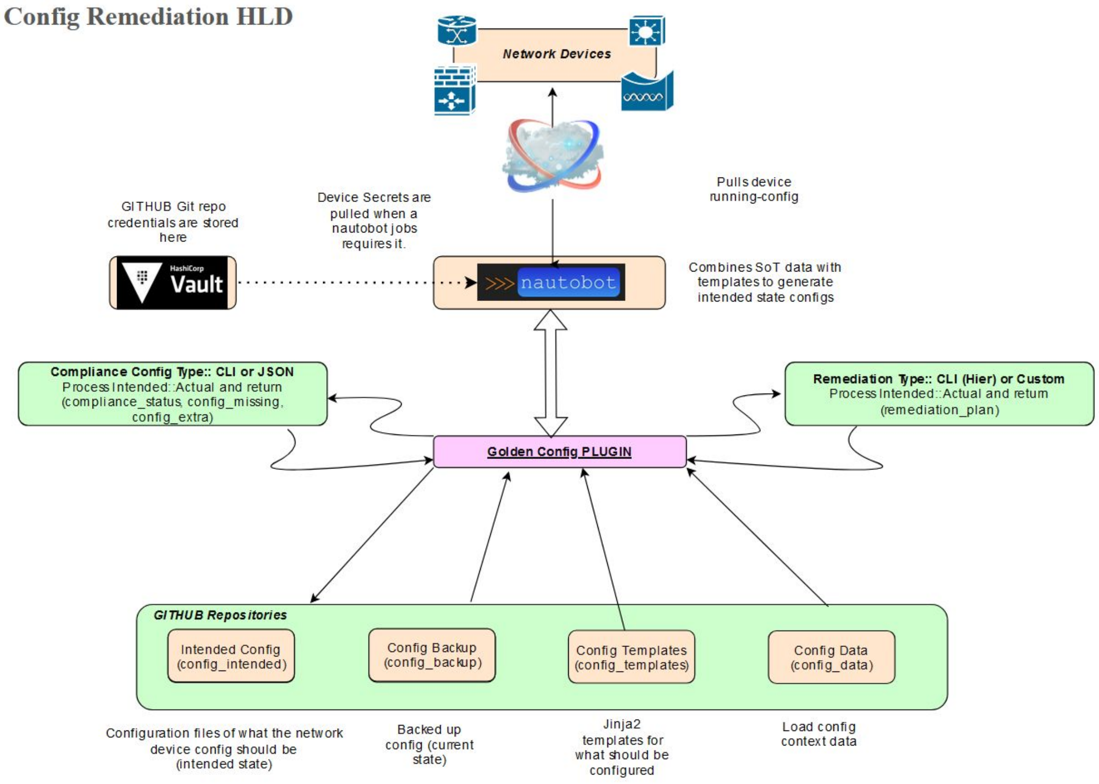
Remediation Interface Configuration

```
interface GigabitEthernet0/1
 ip address 192.168.30.25 255.255.255.0
 switchport access vlan 30
 no speed 100
```

Generated by Golden Config app

>>> Nautobot Configuration Remediation Engine

Solution High Level Design



Config Remediation HLD

Network Devices

GITHUB Git repo credentials are stored here

HashiCorp Vault

Device Secrets are pulled when a nautobot jobs requires it.

>>> nautobot

Pulls device running-config

Combines SoT data with templates to generate intended state configs

Compliance Config Type:: CLI or JSON
Process Intended::Actual and return (compliance_status, config_missing, config_extra)

Remediation Type:: CLI (Hier) or Custom
Process Intended::Actual and return (remediation_plan)

Golden Config PLUGIN

GITHUB Repositories

Intended Config (config_intended)

Config Backup (config_backup)

Config Templates (config_templates)

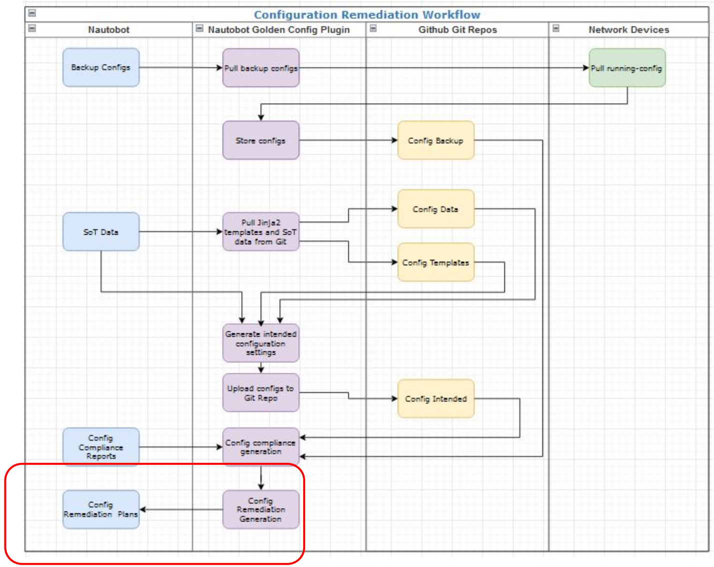
Config Data (config_data)

Configuration files of what the network device config should be (intended state)

Backed up config (current state)

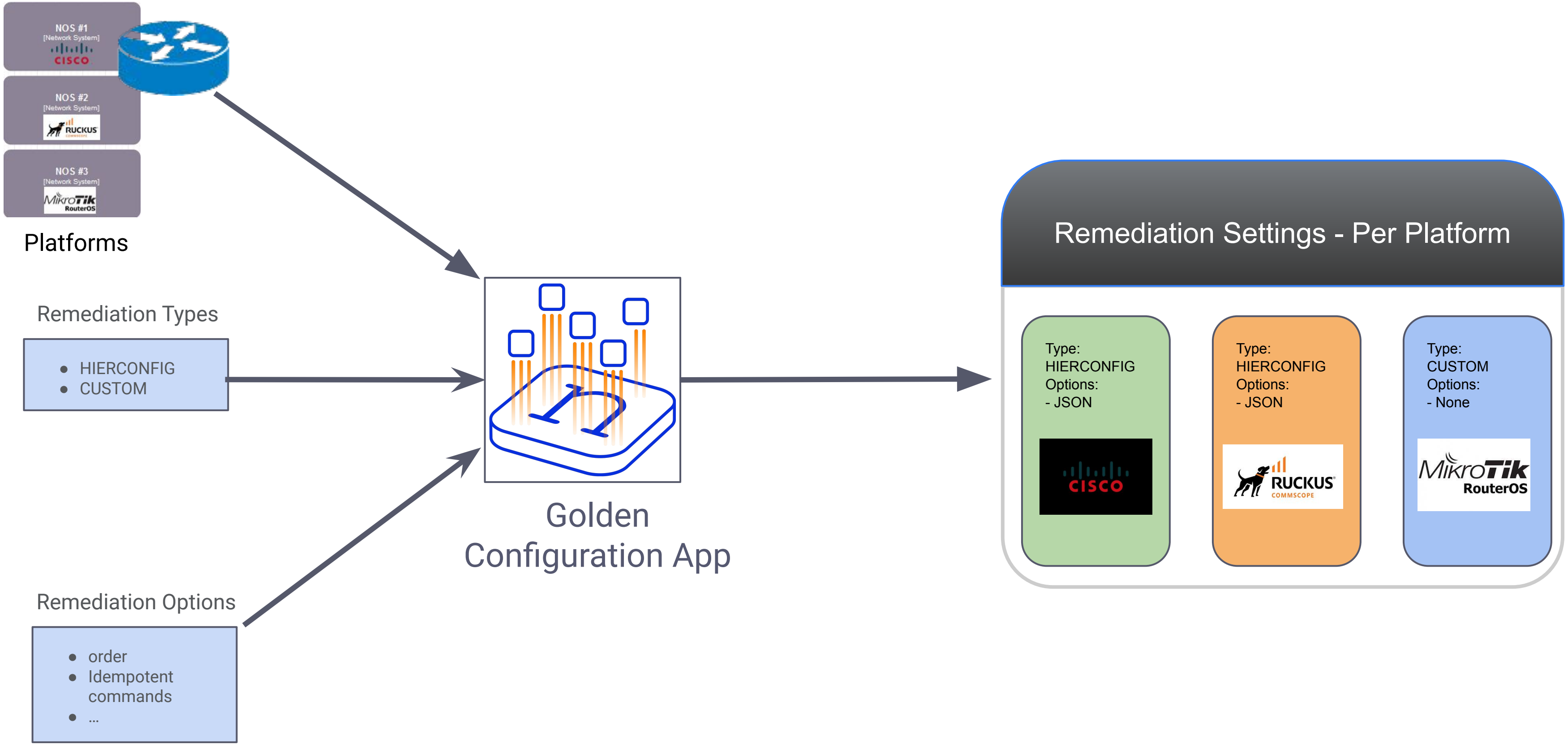
Jinja2 templates for what should be configured

Load config context data


Configuration Remediation Workflow
Nautobot
Nautobot Golden Config Plugin
Github Git Repos
Network Devices
Backup Configs
Pull backup configs
Pull running-config
Store configs
Config Backup
Config Data
SoT Data
Pull Jinja2 templates and SoT data from Git
Config Templates
Generate intended configuration settings
Upload configs to Git Repo
Config Intended
Config Compliance Reports
Config compliance generation
Config Remediation Plans
Config Remediation Generation

Nautobot Configuration Remediation

Golden Config Remediation Settings



Platforms

Remediation Types

- HIERCONFIG
- CUSTOM

Remediation Options

- order
- Idempotent commands
- …

Golden Configuration App

Remediation Settings - Per Platform

Type:
HIERCONFIG
Options:
- JSON

Type:
HIERCONFIG
Options:
- JSON

Type:
CUSTOM
Options:
- None

>>> Nautobot Configuration Remediation

Remediation Types - HIERCONFIG

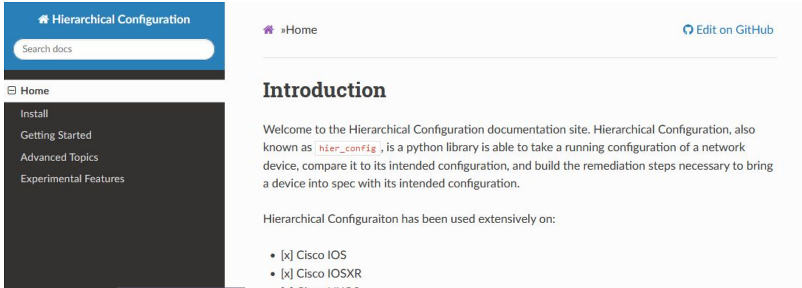
HIERCONFIG: python library that builds the remediation steps necessary to bring a device into spec with its intended configuration.

Hierarchical Configuration native options for:

- Cisco IOS
- Cisco IOSXR
- Cisco NXOS
- Arista EOS

Any NOS that utilizes a CLI syntax that is structured in a similar fashion to IOS can be customized to work with HIERCONFIG.

Code documentation can be found at:
https://netdevops.io/hier_config/

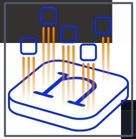


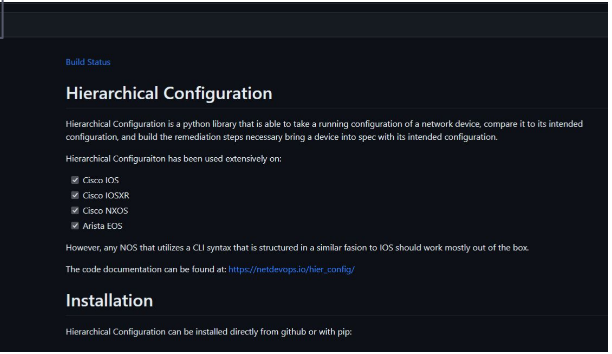
Introduction

Welcome to the Hierarchical Configuration documentation site. Hierarchical Configuration, also known as `hier_config`, is a python library is able to take a running configuration of a network device, compare it to its intended configuration, and build the remediation steps necessary to bring a device into spec with its intended configuration.

Hierarchical Configuraiton has been used extensively on:

- [x] Cisco IOS
- [x] Cisco IOSXR
- [x] Cisco NXOS
- [x] Arista EOS





Hierarchical Configuration

Hierarchical Configuration is a python library that is able to take a running configuration of a network device, compare it to its intended configuration, and build the remediation steps necessary bring a device into spec with its intended configuration.

Hierarchical Configuraiton has been used extensively on:

☑ Cisco IOS
☑ Cisco IOSXR
☑ Cisco NXOS
☑ Arista EOS

However, any NOS that utilizes a CLI syntax that is structured in a similar fasion to IOS should work mostly out of the box.

The code documentation can be found at: https://netdevops.io/hier_config/

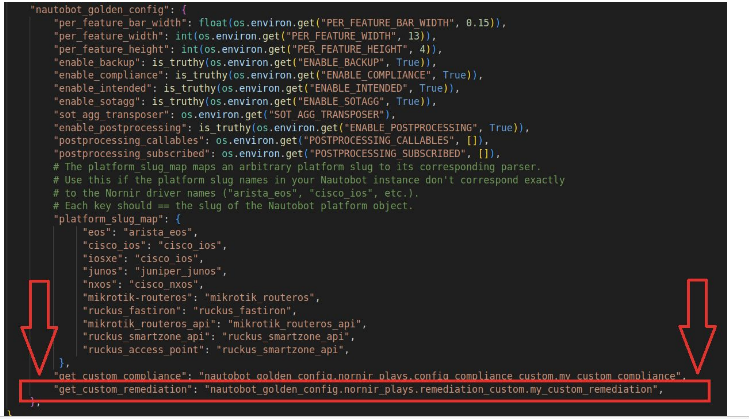
Installation

Hierarchical Configuration can be installed directly from github or with pip:

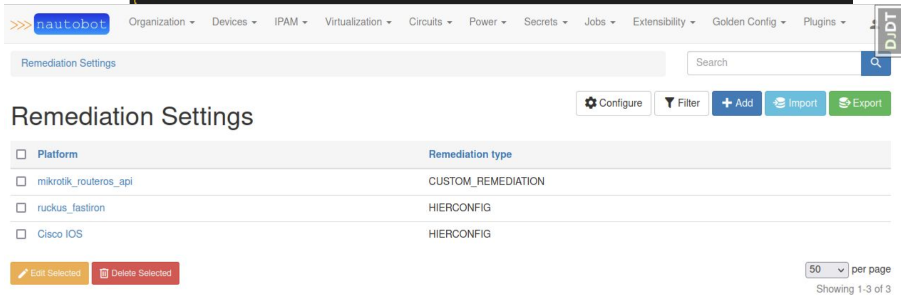
Nautobot Configuration Remediation

Remediation Types - CUSTOM

- CUSTOM_REMEDIATION: Allows Nautobot users to define their own remediation logic.
- Custom Remediation logic is implemented using a python function.
- Python function is then referenced in Nautobot Golden Config settings.
- USE CASE: Mikrotik remediation, actual/backup config is in JSON format (not suitable for HIERCONFIG) and output should be routerOS CLI.



```
"nautobot_golden_config": {
    "per_feature_bar_width": float(os.environ.get("PER_FEATURE_BAR_WIDTH", 0.15)),
    "per_feature_width": int(os.environ.get("PER_FEATURE_WIDTH", 13)),
    "per_feature_height": int(os.environ.get("PER_FEATURE_HEIGHT", 4)),
    "enable_backup": is_truthy(os.environ.get("ENABLE_BACKUP", True)),
    "enable_compliance": is_truthy(os.environ.get("ENABLE_COMPLIANCE", True)),
    "enable_intended": is_truthy(os.environ.get("ENABLE_INTENDED", True)),
    "enable_sotagg": is_truthy(os.environ.get("ENABLE_SOTAGG", True)),
    "sot_agg_transposer": os.environ.get("SOT_AGG_TRANSPOSER"),
    "enable_postprocessing": is_truthy(os.environ.get("ENABLE_POSTPROCESSING", True)),
    "postprocessing_callables": os.environ.get("POSTPROCESSING_CALLABLES", []),
    "postprocessing_subscribed": os.environ.get("POSTPROCESSING_SUBSCRIBED", []),
    # The platform_slug_map maps an arbitrary platform slug to its corresponding parser.
    # Use this if the platform slug names in your Nautobot instance don't correspond exactly
    # to the Nornir driver names ("arista_eos", "cisco_ios", etc.).
    # Each key should == the slug of the Nautobot platform object.
    "platform_slug_map": {
        "eos": "arista_eos",
        "cisco_ios": "cisco_ios",
        "iosxe": "cisco_ios",
        "junos": "juniper_junos",
        "nxos": "cisco_nxos",
        "mikrotik-routeros": "mikrotik_routeros",
        "ruckus_fastiron": "ruckus_fastiron",
        "mikrotik_routeros_api": "mikrotik_routeros_api",
        "ruckus_smartzone_api": "ruckus_smartzone_api",
        "ruckus_access_point": "ruckus_smartzone_api",
    },
    "get_custom_compliance": "nautobot_golden_config.nornir_plays.config_compliance_custom.my_custom_compliance",
    "get_custom_remediation": "nautobot_golden_config.nornir_plays.remediation_custom.my_custom_remediation",
},
```



Remediation Settings

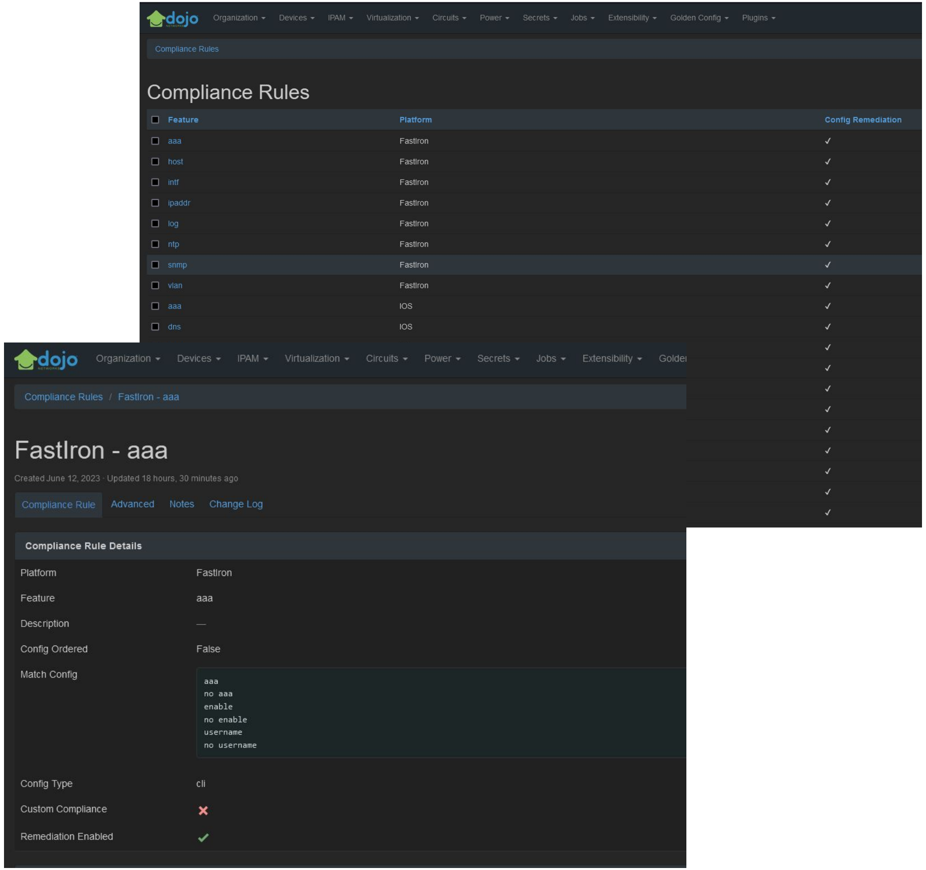
□	Platform	Remediation type
□	mikrotik_routeros_api	CUSTOM_REMEDIATION
□	ruckus_fastiron	HIERCONFIG
□	Cisco IOS	HIERCONFIG

Showing 1-3 of 3

>>> Nautobot Configuration Remediation Engine

Golden Config - Activate Remediation

- Platform Settings need to be set up first, enabling remediation at the platform level.
- Individual features can then be activated/enabled for remediation.
- Administrators have the ability to decide which features should be enabled.
- Features without remediation enabled will not generate remediation snippets when the compliance job runs.



Compliance Rules

Compliance Rules

□	Feature	Platform	Config Remediation
□	aaa	FastIron	✓
□	host	FastIron	✓
□	intf	FastIron	✓
□	ipaddr	FastIron	✓
□	log	FastIron	✓
□	ntp	FastIron	✓
□	snmp	FastIron	✓
□	vlan	FastIron	✓
□	aaa	IOS	✓
□	dns	IOS	✓
			✓
			✓
			✓
			✓
			✓
			✓
			✓
			✓
			✓

Compliance Rules / FastIron - aaa

FastIron - aaa

Created June 12, 2023 · Updated 18 hours, 30 minutes ago

Compliance Rule Details

Platform	FastIron
Feature	aaa
Description	—
Config Ordered	False
Match Config	aaa no aaa enable no enable username no username
Config Type	cli
Custom Compliance	✗
Remediation Enabled	✓

>>> Nautobot Configuration Remediation Engine

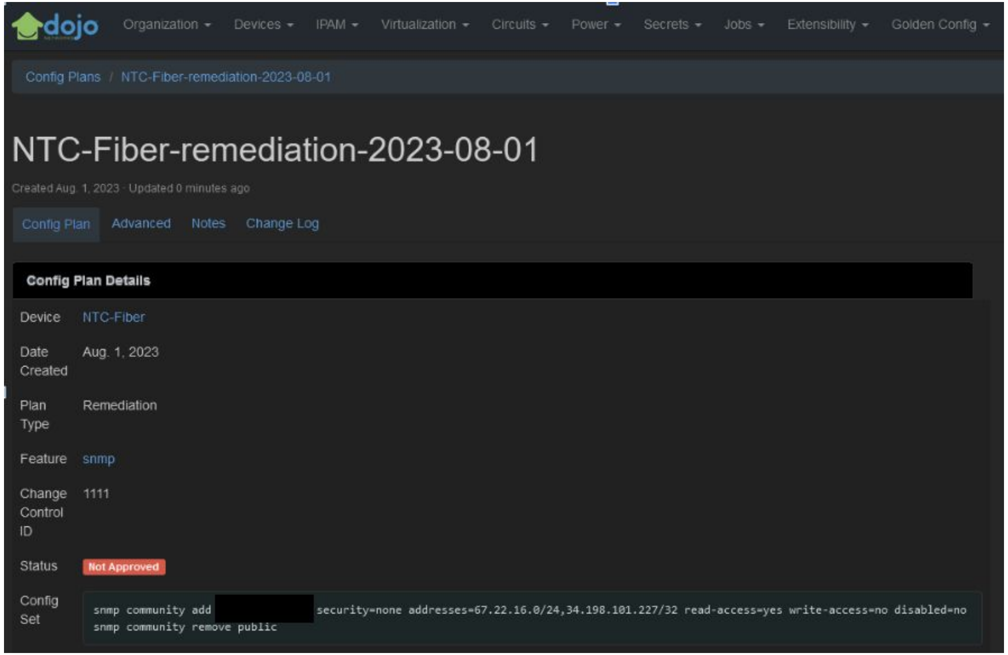
Golden Config - Config Plans

The natural progression for the Golden Config application is providing the ability to execute config deployments. Config Plans aggregate deployment data and provide LCM.

The Golden Config application has the ability to generate plans containing sets of configuration commands from various sources with the intent of deploying them to devices.

The current sources of these plans (i.e. plan types) are as follows:

- The **Intended** configuration of a specific Compliance Feature
- The **Missing** configuration of a specific Compliance Feature
- The **Remediation** configuration of a specific Compliance Feature(s).
- A **Manual** set of configuration commands



NTC-Fiber-remediation-2023-08-01

Created Aug. 1, 2023 · Updated 0 minutes ago

Config Plan Details

Device	NTC-Fiber
Date Created	Aug. 1, 2023
Plan Type	Remediation
Feature	snmp
Change Control ID	1111
Status	Not Approved

Config Set

```
snmp community add [redacted] security=none addresses=67.22.16.0/24,34.198.101.227/32 read-access=yes write-access=no disabled=no
snmp community remove public
```

Getting Started

Q&A