Automating Internet2's Nationwide Network with Cisco NSO

Karl Newell
Network Software Architect, Internet2
14-FEB-2024
Agenda

• Background
• Some History
• Migration
• Modern Times
• Keys to Success
Background

14-FEB-2024
Some History

14-FEB-2024
The Challenge

- Deploy a new nationwide network
  - Double the device count
- Change vendors
- Migrate legacy configuration
The Solution

• Cisco Network Services Orchestrator (NSO)
• Lots of Python
  • pyATS
• Google Sheets and Apps Script
• Amazing Network Engineers and Developers
What NSO Offers

• Configuration Orchestration
  • Multi-vendor support
  • Templates
  • Single config tree includes all devices
• System-wide transactions
• Coexistence with out of band changes
• Graceful evolution over time
• **Command line interface**
How NSO Works

- Service models
  - YANG
  - Vendor neutral
- Templates
  - XML
  - Vendor specific
- Declarative
  - NSO determines the minimal amount of changes needed to configure the device
  - Service instance deletion removes relevant configuration
Example Service Model

> show configuration services i2px-cust ALBA-TEST*

i2px-cust ALBA-TEST-1 {
  admin-state in-service;
  service-id 55668;
  entity TEST;
  pdp ALBA-CONN-TEST-1;
  encapsulation {
    dot1q {
      vlan-id 1091;
    }
  }
  address-ipv4 192.0.2.1/30;
  address-ipv6 2001:db8::1/64;
  remote-as 65505;
  neighbor 192.0.2.2 {
    maximum-prefix 20;
  }
  neighbor 2001:db8::2 {
    maximum-prefix 10;
  }
  password-md5 REDACTED;
  select-in {
    prefix 65505-CUST-V4-IN;
    prefix 65505-CUST-V6-IN;
  }
}

 NANOG™
How NSO Works

Service Manager

Device Manager

Network Element Drivers (NEDs)

Service Model

Device Model
Why We Chose NSO

• Vendor agnostic
• Declarative configuration
  • Service deletion removes all related configuration
• System wide commits
  • Problems don’t strand broken config
• CLI
• Engineers aren’t copying around their version of a configuration template
• Engineers spend less time implementing changes
Migration

14-FEB-2024
Network Automation and Migration

- NSO service models leveraged for service deployments
  - Minimal, simple parameters needed to produce complex device configuration
  - All changes are atomic across the network
- Layer 3 (BGP) service migrations - Cisco NSO, Google Sheets
  - Imported legacy network/DB data to produce NSO-generated service config
  - Allowed migration of 2000 peerings in ~30 days (up to 150/night)
Network Automation and Migration

• Validation tooling for quality assurance - Cisco pyATS
  • BGP prefix acceptance/rejection monitoring before and after migration
  • Allowed rapid validation of migrated service
• Test Driven Development - Robot Framework
  • Rapid iteration of NSO service models with testing to deter regression
  • Decrease time needed for new service deployments
Network Automation and Migration

- Configuration control and monitoring
  - Manual changes on device are identified and flagged for reintegration into NSO models
  - Ensures minimal drift
Network Automation and Migration

- Juniper MX to Cisco 8200 (IOS-XR)
- Translate services - legacy to NGI
  - Extract-Transform-Load
    - Scripts to pull config from legacy network and DB
    - Save into Google Sheet
    - Google Apps Script to convert data into NSO config
- Service validation scripts
  - pyATS
Network Automation and Migration

- pyATS validation scripts
  - pre and post migration BGP data (prefixes and counts)
  - diff and report anomalies

### re-participant CHIC-CUST-1 (192.0.2.1) ###
Neighbor migrated: rtsw.chic -> core2.chic

<table>
<thead>
<tr>
<th>Prefix</th>
<th>[ PRE ]</th>
<th>[ POST ]</th>
<th>INFO/Advice</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.0.0.0/24</td>
<td>*</td>
<td>*</td>
<td>INVESTIGATE - route no longer being accepted</td>
</tr>
<tr>
<td>192.0.2.0/24</td>
<td>*</td>
<td>*</td>
<td>INVESTIGATE - route no longer being accepted</td>
</tr>
<tr>
<td>192.168.0.0/16</td>
<td>*</td>
<td>*</td>
<td>INVESTIGATE - route no longer being accepted</td>
</tr>
<tr>
<td>198.51.100.0/24</td>
<td>*</td>
<td>*</td>
<td>INVESTIGATE - route no longer being accepted</td>
</tr>
<tr>
<td>203.0.113.0/24</td>
<td>*</td>
<td>*</td>
<td>INVESTIGATE - route no longer being accepted</td>
</tr>
</tbody>
</table>

Skipped 14 routes that did not change
NSO Stats

- 30 service models (19 edge service models)
- 4,000+ service instances
- 32,047 lines of NSO service config
  - resulting in 250,282 lines of device config
- **7.81 config compression ratio**
- 2,200+ commits
- 632 git merges
- 6,000+ build pipelines
Service Development

• Rapid iteration with Network Engineering
  • “paper prototypes” - what would an Engineer want to enter on the CLI to define a Service
• Keep Service options to a minimum - don’t need all the knobs
  • Reduces the amount of testing needed
• We spend a significant amount of time defining what a Service is and what input is needed to differentiate Service instances
Multi Vendor Support

- Cisco 8200 (IOS-XR)
- Cisco NCS 5500 (IOS-XR)
- Arista 7280R3
- Juniper EX 4600
- Juniper SRX 1500
- Juniper SRX 4100
User-facing Service Provisioning

- Developing the Internet2 Insight Console
  - Layer 2 and 3 circuit provisioning
  - Cloud Connections
  - Routing Intentions (prefix management)
- Looking Glass
- Future
  - Visualization and reporting of all member services
  - Management of all member services
Keys to Success

14-FEB-2024
Integrated Team

• Software Development and Network Engineering teams were tightly integrated
  • Met (and continue to meet) weekly
  • Developers and engineers worked together to develop service models and implementation
Iteration

- Don’t be afraid to start over
  - Some of our service models are on their third revision
Right Tools for the Job

- NSO is designed for network automation
- But NSO doesn’t meet all of our needs so we use other tools as well
  - pyATS
  - Nornir
- Spreadsheets work
  - Especially coupled with scripting (Google Apps Script)
Thank you

Karl Newell
Network Software Architect, Internet2
14-FEB-2024