NANOG 81

Automation without Config Deployment
Introduction

Ken Celenza
• Managing Director at Network to Code
• Traditional network engineer by day, coder by night
• Converted full time network Automator in 2016
• 20 years in the industry, primarily supporting enterprises
Automation is the art of deploying configurations... right?
The Common Approach
Let’s Automate the Fun Part!

Let’s face it, the configuration is the fun part... why is it the first thing we try to automate?

• Well, I configure [“IPSec tunnels”, “firewall rules”, “switchports”] that must be where my time is spent.

• Let’s automate the configuration deployment of the tedious tasks.
  – Develop conf_ipsec_tunnel.py, deploy_fw_rule.yml, etc.

• The configuration required only requires a few variables.
Why isn’t Anyone Using my Automation?

• Change window was an hour, and now “I am done a few minutes early.”
• “My change window is so short, I need to make sure everything works in time”
• “If I’m going to make the change, I want to know what configuration is going to be deployed”
• “Automation can’t be run unattended, I still need to verify everything myself”
What are the Issues?

**Source of Truth**

Time is spent curating the “correct data” and configuration.
- Data is kept transactionally, and not via the SoT.
- Results in re-doing the same analysis every time there is a change.
What are the Issues?

Verification

Time is spent on verifying the network is “healthy”

• This takes experience and institutional knowledge to know what that means
What are the Issues?

*Network Management*

Need to add to all other monitoring and inventory systems.
What are the Issues?

The actual configuration doesn’t take long to deploy.
Workflow Analysis
Why is it Important?

- Most networking groups are not actually aware of their own workflows.
- What should be tracked?
  - Number of times a type request happens
  - Amount of engineering time (hours worked)
  - Amount of time from request to completion
  - Opportunity Cost
- Should ask yourself, “how would I explain process to new engineer on the team”
First Take

Tips:
- Do not discount the work being done.
- Take a system view of the workflow
- Consider all groups and approvals
- Consider all tasks!
- ... Don’t do this -

---

**IPSec Tunnel Creation**

<table>
<thead>
<tr>
<th>ServiceNow</th>
<th>Network Technician</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPSec Request Made</td>
<td>Configure IPSec Tunnel</td>
</tr>
<tr>
<td>Close Ticket</td>
<td></td>
</tr>
</tbody>
</table>

---

www.networktoeode.com
Second Take
Looking Better… still…

---

www.networktocode.com
Time Required

Now we can see where the time is spent.

What does the data tell us?
Deploying configuration, has low ROI
Data Curation
Getting Better Data

- Limit free-form fields from requestors (work with ServiceNow developers as an example)
- Move tunnel assignments to programmatic accessible attribute
  - Database, NetBox, Git, gsheet, etc.
- Develop automation to update the Source of Truth for next available tunnels and subnets
- Develop automation to verify resources are free, by checking actual devices
- Develop automation to create configuration snippets and test plan
Generating Configuration

Populating a proper Source of Truth is the cornerstone to automation
python bp_tun_create_vars.py >> vars.json

[Reserve Tunnel, Subnets, generate Configuration, generate test plan for Ticket, manually copy over]
Verification
Pre & Post Checks

- Each engineer has their own tests
- There is no standardized definition of healthy
- There is no baseline for operational data (non-snmp, e.g. optic-levels)
- Data intends to change, 100 tunnels before change, 101 after
- Raw text is too large to compare
  - Timers and counters make it impossible to use diff
- There still needs to be evidence for change control
Pre & Post Checks

Rethink how checks are done
- Build queries against structured data
- Compare to "healthy" not just the change
  - Run all checks every time
- Remove manual diff review
Business Partner - IPSec Tunnel Creation

Request for new Business Partner

15 min

Assigned to Network Engineering

5 min

Schedule Change

[ CAB Denied ] [ CAB Approved ]

10 min

Verify Data Points

30 min

IT Operations requests more data from user

15-60 min

Python bp_tun_create_vars.py >> vars.json

[ Reserve Tunnel, Subnets, generate Configuration, generate test plan for Ticket, manually copy over ]

10 min

View Monitoring Health

10 min

Remove Alarms / Notify Operations

10 min

Reactivate Alarms & Monitoring

10 min

Rediscover new Tunnel

10 min

Python bp_tun_pre_check.py

[ capture pre checks, compare to post checks ]

20 min

Close Ticket

[ Upload Evidence ]

perform analysis
Network Management
Network Management

- Maintaining systems is difficult and tedious
- Large amount of false positives removes trust in monitoring
  - How many NOC’s are filled with dozens of unanswered alarms?
Business Partner - IPSec Tunnel Creation

Request for new Business Partner

15 min

IT Operations requests more data from user

15-60 min

Verify Data Points

30 min

Assigned to Network Engineering

5 min

Schedule Change

[Reserve Tunnel, Subnets, generate Configuration, generate test plan for Ticket, manually copy over]

python bp_tun_create_vars.py >> vars.json

Manually Apply Configurations

python bp_tun_pre_check.py
[capture pre checks, compare to post checks]

python bp_tun_start_maint.py
[Bring systems out of monitoring]

Remove Alarms / Notify Operations

python bp_tun_end_maint.py
[Bring systems back into monitoring, rediscover devices for tunnels]

Close Ticket

20 min

Upload Evidence

perform analysis

www.networktoCode.com
Final Thoughts
Automation without Config Deployment

What are the Benefits?

- Network Engineer is in control of what commands they send
  - This allows them to trust automation, engineers need to see the configs that are being sent
  - Automation cannot be blamed for issues
- Concentrate on tasks that take the most time
- Quicker to get into production and easier adoption
- Helps to build out a Source of Truth
- Allows automation to be introduced with less pressure
## Final Design

Processed from 4-5 engineers hours to **less than 1!**

### Business Partner - Automated Design

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>servicenow Request for new Business Partner</td>
</tr>
<tr>
<td>2.</td>
<td>Obtain Next Available Network</td>
</tr>
<tr>
<td>3.</td>
<td>servicenow Send Config, vars, and plan</td>
</tr>
<tr>
<td>4.</td>
<td>servicenow Schedule Change</td>
</tr>
<tr>
<td>5.</td>
<td>servicenow Close Ticket</td>
</tr>
<tr>
<td>6.</td>
<td>[API Call] Engineer runs post checks, monitoring</td>
</tr>
<tr>
<td>7.</td>
<td>[API Call] Engineer runs pre checks, monitoring</td>
</tr>
</tbody>
</table>

- [CAB Approved]
- [CAB Denied]