The Challenges Of Testing In Network CI/CD Pipelines

Anthony Miloslavsky
SE, Cumulus Networks

@permitanyany
Quick History Lesson

• Why Is Networking So Far Behind?
  – Shared infrastructure
  – If it ain’t broken…don’t fix it (until it breaks)
  – High risk, low reward culture
  – Innovation overshadowed by fear
Network Automation Today

- Initial provisioning
- Pushing one-off changes
- Automated validation?

(ntp server 1.1.1.1 configuration management tool)
Network Automation Tomorrow

• Infrastructure As Code
  – Source Of Truth
  – Imperative vs Declarative
  – Idempotency
Network Automation Tomorrow

- ntp: servers: 1.1.1.1
- code commit
- configuration management tool
- ntp server 1.1.1.1
Us And Them

- code
- build
- test
- deploy
- monitor

Continuous Integration

Continuous Deployment
Challenges Of Testing

Does This Translate To Networking?

Development Environment

Production Environment

configuration management tool

ntp server 1.1.1.1

configuration management tool

ntp server 1.1.1.1
Challenges Of Testing

This Looks More Realistic

Lab

Production Environment
Levels Of Testing

Static Post Change Testing

```python
tasks:
  - name: Check NTP State
    shell: "net show time ntp servers | grep 1.1.1.1"
  - name: Output NTP State
    debug:
      var: ntp_output.stdout_lines
```

Automated test
Levels Of Testing

Reusable Automated Tests

- ntp: servers: 1.1.1.1
- code commit
- configuration management tool
- ntp server 1.1.1.1
- automated tests
CI/CD Recap

- code
- build
- deploy
- test

Tools:
- Ansible

Flow:
User -> code -> build -> deploy -> test
CI/CD Recap

swp1:
    description: to Server01
swp2:
    description: to Server02
bgp:
    as: 65011
    neighbors:
        - swp51
        - swp52
ntp:
    servers:
        - 1.1.1.1
        - 2.2.2.2
CI/CD Recap

```
swp1:
  description: to Server01
swp2:
  description: to Server02
bgp:
  as: 65011
  neighbors:
    - swp51
    - swp52
ntp:
  servers:
    - 1.1.1.1
    - 2.2.2.2

auto swp1
iface swp1
  alias {{ swp1.description }}

auto swp2
iface swp2
  alias {{ swp2.description }}

router bgp {{ bgp.as }}
  {% for interface in bgp.neighbors -%}
    neighbor {{ interface }} interface remote-as internal
  {% endfor %}
```
CI/CD Recap

```
stages:
  - deploy

deploy:
  tags:
    - deploy
  stage: deploy
  script:
    - ansible-playbook main.yml
```
CI/CD Recap

- hosts: leaf
  remote_user: cumulus
  become: yes
  become_method: sudo

tasks:
  - name: populate interfaces file
    template: src=leaf_interfaces.j2 dest=/etc/network/interfaces
CI/CD Recap

tasks:
- name: Checking BGP State
  shell: net show bgp sum json | grep "state" | grep -v "Established"
  register: bgp_check
  ignore_errors: true

- name: Evaluate BGP State
  fail:
    msg: "BGP is currently in a bad state."
  when: bgp_check.rc == 0
CI/CD Recap

```
stages:
  - deploy
  - test

deploy:
  tags:
    - deploy
  stage: deploy
  script:
    - ansible-playbook main.yml

bgp_test:
  tags:
    - deploy
  stage: test
  script:
    - ansible-playbook test.yml
```
Levels Of Testing

Reusable Automated Tests

ncp: servers: - 1.1.1.1

code commit

configuration management tool

ntp server 1.1.1.1

Is This Good Enough?

automated tests
Levels Of Testing

Simulation

code commit → simulation → configuration management tool → ntp server 1.1.1.1 → automated tests

ntp: servers: - 1.1.1.1

configuration management tool
Simulation

- Microsoft CrystalNet

<table>
<thead>
<tr>
<th>Root Cause</th>
<th>Proportion</th>
<th>Examples</th>
<th>CrystalNet Coverage</th>
<th>Verification Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software Bugs</td>
<td>36%</td>
<td>bugs in routers, middleboxes, management tools</td>
<td>✓</td>
<td>X</td>
</tr>
<tr>
<td>Config. Bugs</td>
<td>27%</td>
<td>wrong ACL policies, traffic black holes, route leaking</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Human Errors</td>
<td>6%</td>
<td>mis-typing, unexpected design flaws</td>
<td>✓</td>
<td>X</td>
</tr>
<tr>
<td>Hardware Failures</td>
<td>29%</td>
<td>ASIC driver failures, silent packet drops, fiber cuts, power failures</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Unidentified</td>
<td>2%</td>
<td>transient failures</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Table 1: Root causes of O(100) significant and customer-impacting incidents in our network (2015 - 2017).
Simulation

• Is Network Simulation Ready For Primetime?
  – All vendors support some version
  – Vagrant and Docker
  – Bloated Images
  – Feature Parity
  – Simulation Speed
CI/CD Recap

```
stages:
- build
- test
- destroy
- deploy

build:
  stage: build
  before_script:
    - cd cicd-simulate
  script:
    - vagrant up leaf01 leaf02

test:
  stage: test
  script:
    - ansible-playbook playbook.yml -i hosts.yml
    - ansible-playbook test.yml -i hosts.yml

destroy:
  stage: destroy
  before_script:
    - cd cicd-simulate
  script:
    - vagrant destroy leaf01 leaf02

deploy:
  stage: deploy
  script:
    - ansible-playbook playbook.yml -i hosts.yml
    - ansible-playbook test.yml -i hosts.yml
```
End Goal

- Code commit
- Automated testing
- Configuration push
- Monitor
- Automated testing
- Provision
- Change
- Replace
Observations/Lessons Learned

– Are engineers ready for automated CD?
– Separate branch for simulation
– Separate inventory files