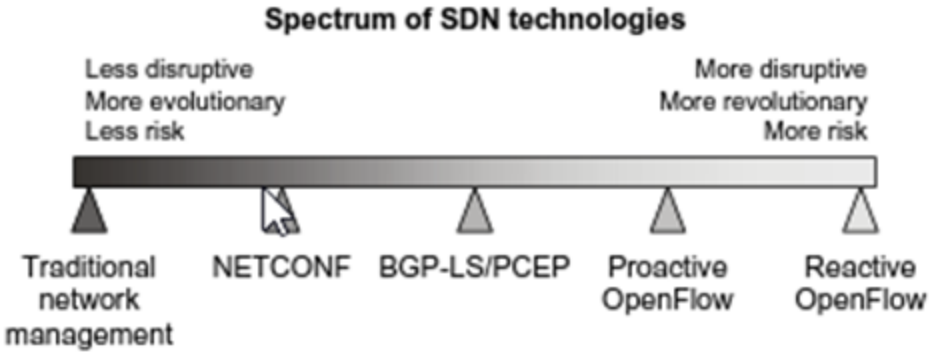
Ansible

Created	@November 19, 2022 12:49 PM
Week	Week 1

Traditioneel CLI vs SDN



Voordelen:

Minder verstoringen in het netwerk;
Minder scholing;

Voordelen:

Veranderingen;
Innovatie;

Minder kosten; Minder riskant	Business kansen;
Nadelen:	Nadelen:
Beperkte veranderingen; Beperkte innovatie; Beperkte business kansen;	Meer verstoringen; Meer scholing; Meer kosten; Meer riskant

Voordelen van IT Automation met Ansible

De mogelijkheid om snel ‘machines’ aan te bieden en te gebruiken. Zelfs nieuwe machines in minuten;
Het controleren, aanpassen en configureren van de infrastructuur;
Saai, herhalend werk wordt minder.

Vormen van Automation

Agent-based:

- Twee systemen: 1) een server en 2) een client ofwel een agent
- Meestal één server en meerdere agents.
- De agent zal periodiek contact zoeken met de server om te vragen of er een nieuwe configuratie is.

Agent-less:

- Geen ‘echt’ client-server model;
- Communicatie op initiatief van een server’;
- Communicatie via standaard protocollen als Telnet. SSH, PowerShell

Voor en Nadelen Agent-based / - less

Voordelen

- Eenvoudige(re) integratie binnen de infrastructuur;

- Geen additionele configuratie (anders dan bijv ssh) nodig.

Nadelen

- Mogelijk verminderde veiligheid omdat alle agents een verbinding met de server kunnen initiëren;
- Performance problemen doordat agents, de server overvragen
- Vaak geen push, maar een pull mogelijkheid waardoor het ophalen van nieuwe configuratie langer kan duren

Wat kun je met Ansible?

- Automatiseren, standardiseren en versnellen van …
- Simpel, krachtig en ‘agent less’ taken uitvoeren.
- Veilige communicatie via SSH of HTTPS
- (Her)gebruiken van playbooks en rollen van de community en fabrikanten.
- Naast het gebruik op ‘netwerkhardware’ kun je Ansible ook gebruiken op Linux en Windows.

Modules: Uitvoerbare code van Ansible
Task: ad-hoc = enkele taak (ansible all -m ping)
Playbook: enkele / meervoudige taken (ansible-playbook ….)

Inventory/Ansible-host file

locatie: /etc/ansible/hosts

kan bevatten: IP-addressen en FQDN + Variabelen en aliassen

opmaak: YAML

Default groups: All en Ungrouped

Host kan lid zijn van 1 of meerdere groepen

```
mail.example.com

[webservers]
foo.example.com
bar.example.com

[dbservers]
one.example.com
two.example.com
three.example.com
```

Commong inventory variables:

- ansible_connection (hoe verbinden met netwerkapparatuur) vb: ansible.netcommon.network_cli
- ansible_network_os
- ansible_user
- ansible_password
- ansible_become
- ansible_become_method

Privilege Escalation & group variables

Adminrechten krijgen (enable, sudo)

- become: yes
- become_method: enable

```
ansible_connection: network_cli
ansible_network_os: ios
ansible_become: yes
ansible_become_method:  enable
```

Variablelen

- Voor wachtwoord, gebruiker, etc.
- Ansible-vault: beveiligen van individuele variabelen

Ad-hoc

Voor taken die je weinig gaat uitvoeren zoals:
Reboot, kopiëren bestanden, specifiek info opvragen

ansible [pattern] -m [module] -a "[module options]“

Let op de comma na HOST, anders is het geen lijst (en wordt er in een inventoryfile naar de host gezocht)
ansible all -i HOST, -m ping

ansible R1 -m raw -a "show ip route" -u root –k (dit gebruik je wanneer R1 in een inventoryfile staat)

Connection Methods

Value of ansible_connection	Protocol	Requires	Persistent?
network_cli	CLI over SSH	network_os setting	yes
netconf	XML over SSH	network_os setting	yes
httpapi	API over HTTP/HTTPS	network_os setting	yes
Local	depends on provider	provider setting	no

Playbooks

Is geschreven in YAML format (key - value pairs)
Is goed leesbaar
Begint met --- en bijvoorkeur eindigt met …

Elke taak in een PB is gekoppeld aan code (een module);
Condition / output is JSON;
Uitvoer van taken is één voor één;

`ansible-playbook ospf.yml –u root -k`

Verifieer Ansible playbook syntax met:

`ansible-lint playbook.yml`

Playbook examples

```
---

- name: Network Getting Started First Playbook
  connection: ansible.netcommon.network_cli
  gather_facts: false
  hosts: all
  tasks:

    - name: Get config for VyOS devices
      vyos.vyos.vyos_facts:
        gather_subset: all

    - name: Display the config
      debug:
        msg: "The hostname is {{ ansible_net_hostname }} and the OS is {{ ansible_net_version }}"
```

```yaml
---
- name: OSPF configuratie
  hosts: all
  gather_facts: false
  connection: local

#-----------------------------------------------

  vars:
    cli:
      username: root
      password: cisco

#-----------------------------------------------

  tasks:
    - name: enable ospf
      ios_config:
        provider: "{{ cli }}"
        authorize: yes
        parents: router ospf 1
        lines:
          - network 172.10.0.0 0.0.255.255 area 0

      register: print_output

    -  debug: var=print_out
```

```yaml
---
- name: configure cisco routers
  hosts: routers
  connection: network_cli
  gather_facts: no
  vars:
    dns: "8.8.8.8 8.8.4.4"

  tasks:
   - name: configure hostname
     ios_config:
       lines: hostname {{ inventory_hostname }}

   - name: configure DNS
     ios_config:
       lines: ip name-server {{dns}}
```