

What is Automation?

Automation is a process to do a task with least manual intervention. A process can be created for a task which is done by humans repeatedly. Repeated tasks or standard tasks for which have any sort of SOP can be automated easily.

Why Automation?

- **Tasks in code**
- **Collaboration**
- **Eliminate errors**
- **Write once**
- **Laziness**
- **Etc.**

Why Automation is Required in Network Space?

Differences between Legacy Method and Automation

Our discussion will be focused around following:

- **Configuration Management**
- **Maintenance**
- **Configuration Audit**
- **Upgradation**

Everybody is talking about Network Automation or NetDevOps. Let's see what it's all about.

Before jumping into netdevops, lets first see how we perform tasks normally in our current environment then we will look at netdevops.

What all tasks we normally do in network? Let`s see some of them.

- Device Provisioning / Configuration Management
- Information Gathering / Monitoring / Troubleshooting.
- Management

1. Device Provisioning / Configuration Management

Device Provisioning

- There is golden template for initial device provisioning. We have multiple templates based on the Vendor (Cisco/HP/Juniper) and Device Role (Access Switch/ Core Switch).
- When we provision a new device, we take reference from golden template and generate the required configuration for each device. Its basically a copying, pasting and editing the golden template. Generally, we use excel for this. We add the certain values (ip address/ vlan numbers), which varies device to device.
- The we login to each device and pastes the configuration.
- Fair enough, the task is done. **Now what if we have to do the same task for 100s of devices?**

Configuration Management

- We manage/change configuration. Some changes are restricted to particular device, where as some are across all the devices.
- This also includes the same procedure, manually generate the configuration and then copy it to the devices.

Issues with this process of manually generating configuration and copying it to devices.

- Error prone – Error can occur while generating configuration or copying & pasting the configuration to the devices.
- Time consuming – What if we have to configure 10, 20 Or 50 devices? Then login and log off from these many devices.
- Uniformity – Everybody has its own way of generating configuration and working on the device.

2. Information Gathering / Monitoring / Troubleshooting.

There are many scenarios when a network engineer has to fetch the information from n number of devices. We login to the device and run the same set of commands and then save the output to a file.

- Let's say there were a power maintenance and we need to check the up time of all the devices and need to check the status of all the links. We need to fetch the output for the following commands.

show runn

show version

show int status

show ip ospf neighbor

show ip bgp summary

show module

show environment

show standby brief

Login to each device and run these commands. Then save the output to a file for each device.

- When ever a change is performed, there is always a requirement of pre-checks and post-checks. Manually doing that involve the same process as above.

Issues with this process of manually gathering information from the devices.

- Time consuming

3. Management

Lets say you are managing 100s or 1000s of devices.

- Updating configuration against golden template which keep changes as per the requirement.

Validating configuration against golden template.

- Upgrading IOS or firmware on multiple devices.

As we have seen the legacy method of doing certain tasks. Let`s see we can automate it and its benefits. We will go through the same points once again.

1. Device Provisioning / Configuration Management

Earlier we talked about the golden template, we will still use that to generate the configuration.

We will decouple the variable part (ip address, vlan number) from the CLI based commands. Now, instead of using a single file for values and commands, we will use two separate files. One file will have the cli based commands and another file will have the variables. With help of variable file, we can generate the configuration for multiple devices.

We have created a template which has standard configuration (cli based command) with variables (variable part is inside curly braces). All the commands are taken from golden template or from documents available on line for the vendors.

```
vlan 10
name data
!
vlan 20
name voice
!
vlan 30
name ap
|!
```

```
# configure vlans
{% for vlan in item.vlans %}
vlan {{ vlan.number }}
  name {{ vlan.name }}
!
{% endfor %}
```

```
# variables:
---
- vlans:
  - number: 10
    name: Data
  - number: 20
    name: Voice
  - number: 30
    name: ap
```

My example is based on Ansible. We will go through the complete process in coming section.

Benefits of using Automation

- **The best part is that we only need to pass the variable through variable file whenever we want to configure new vlans. Our template remains intact.**
- **Less error prone** – We are not copying and pasting the complete configuration and then editing it. We are only passing the variables and rest is taken care by the tool itself.
Only a certain user will have the access to edit the template, so there are less chances to make changes in the which can cause issues later.
- **Scaling** – With this same template we can generate configuration for any number of devices.
- **Standardization:** As we are not changing our code our configuration will be a standard for all the devices.

2. Information Gathering / Monitoring / Troubleshooting.

- We talked about a power maintenance, where we need to fetch the output of certain commands. We need to fetch the output for the following commands.

show runn

show version

show int status

show ip ospf neighbor

show ip bgp summary

show module

show environment

show standby brief

Again, my example to fetch information from multiple devices is based on Ansible.

```
---
- name: Show Running Configuration
  hosts: core-switches
  connection: local
  gather_facts: false

  tasks:
  - name: Show Runn
    ios_command:
      authorize: yes
      commands:
        - show runn
        - show version
        - show int status
        - show ip ospf neighbor
        - show ip bgp summary
        - show module
        - show environment
        - show standby brief

    register: print_output
  - debug: var=print_output.stdout_lines
```

With this single source of code, we can fetch information from multiple devices. This can also, save the configuration in a text file for each device. We don't even need to login to the switches.

Benefits of using Automation

- Time saving

Why Automation?
The Big Picture – Orchestration

- We are slowly moving towards Cloud – Public, Private and Hybrid Cloud. All the services offered by cloud are deployed automatically. There is a self-service portal, where a user fills all the required information and at the backend it gets deployed automatically.
- The virtualization and automation in server/compute, storage and other technologies apart from network is not new.
- Network was lagging as far as the virtualization and automation is concerned but not anymore. There are many ways to do virtualization and automation in network. Many vendors are offering these capabilities.
- When we use self-service portal, be it in-house or hosted on public cloud, we only need to fill in required information and all the underline tasks are implemented automatically.
- There is an orchestration tool which sits above all the underlying technologies which delicates the tasks to each underlying service like compute, storage and network, which are further done automatically with help of an automation tools or script.
- Automation in server and storage space were already there. Automation in network was badly needed because to deploy any service automatically with the information filled in on self-service portal all tasks have to be achieved without manual intervention otherwise the application has to wait for a pending task to be complete by a human, which can break the cycle and the purpose of self-service portal.