Experiences and learning From APRICOT 2023

Ismat Zerin
Ex. Fellow of APRICOT 2023
Fiber@Home Ltd.

Why Fellowship

Professional Development

Fellowships often provide opportunities for personal and professional growth through specialized training, mentorship, and networking.

Financial Support

Financial Support is one of the major fact

Recognition

Being awarded a fellowship can be a significant achievement and can provide validation and recognition for work and accomplishments.

Community

Fellowships can also provide opportunities to connect with other professionals in our field and build a supportive community.

learning From APRICOT 2023

Network Automation

My workshop topic was **Network Automation for Network Engineers**

I have learned docker concept & some popular applications like smokeping, LibreNMS, Netbox, Uptime Kuma etc

Ansible, YAML,configured router via playbook etc.

Orchestration tool options

- Ansible
- Saltstack
- Puppet
- Chef

and more...

New Learning for me

INTRODUCTION OF YAML

YAML is known for its simplicity and readability, making it a popular choice for configuration files in various applications and systems.

Starts with ---
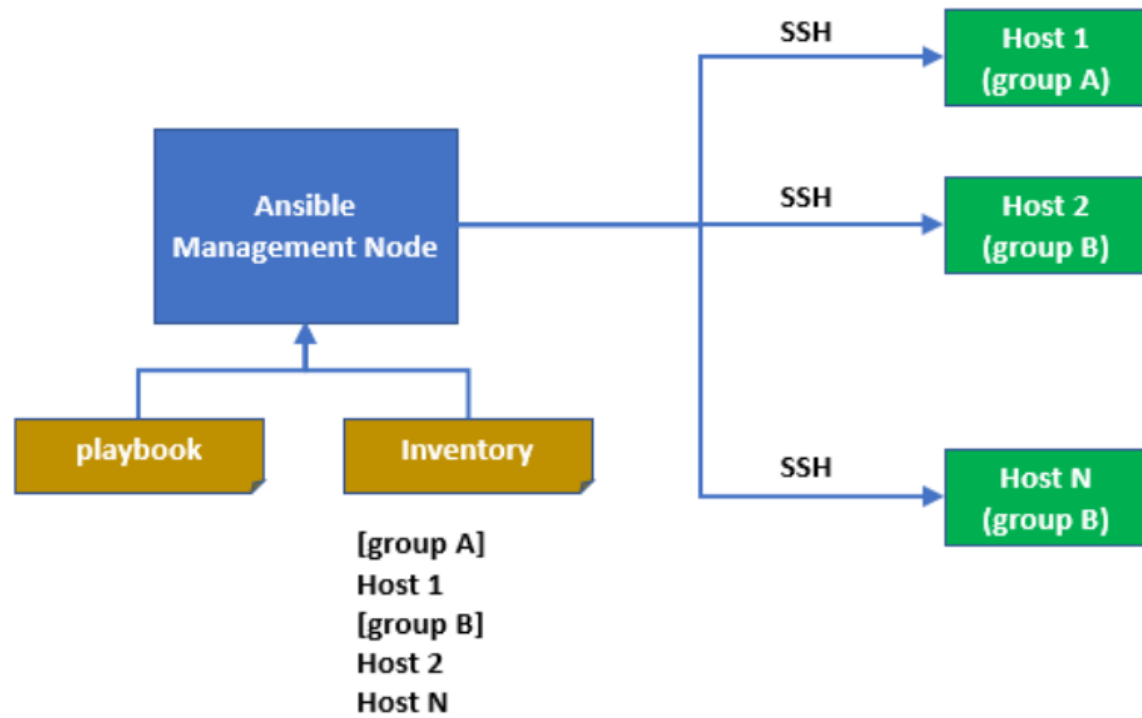
Indentation used to define nesting

Spaces Matters!!!

File extension: .yml or .yaml

FOR EXAMPLE

```
---
- name: Configure Web Server
  hosts: webserver
  become: yes
  tasks:
    - name: Install Apache
      package:
        name: apache2
        state: present
      tags:
        - webserver
```

# New Learning for me

Key elements of Ansible

After the workshop

I EXPERIMENTED THE AUTOMATION AND PREPARE A SMALL LAB IN GNS3
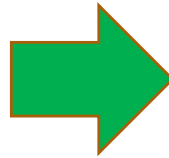
Network diagram in GNS3

Here in my diagram two Routers
connected through switch and in my lab I
will configure the routers via playbook
In Routers
Initially I have to configure router port
with IP address and SSH configuration as
well.
In Network Automation
I have to identify the hosts(R1,R2) and
check the hosts by
#ansible --hosts-list all
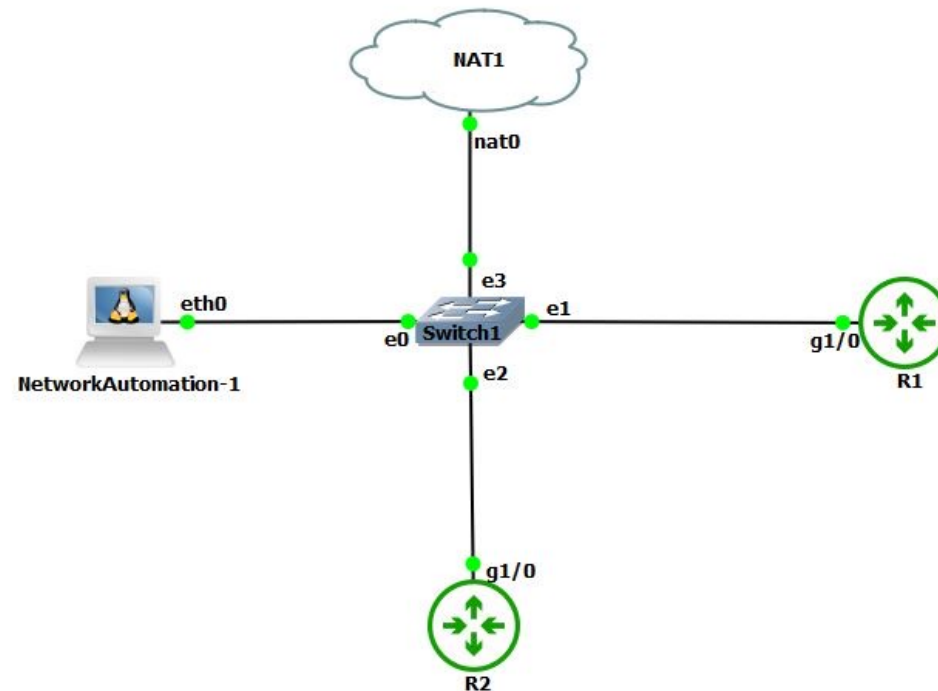
Lab Output

```
root@NetworkAutomation-1:~# cat newplaybook.yml
---
- name: Get status
  hosts: all
  gather_facts: false
  tasks:
    - name: Show Status
      raw: "show interface desc"
      register: print_output
    - debug: var=print_output.stdout_lines
```

output →

```
root@NetworkAutomation-1:~#
root@NetworkAutomation-1:~#
root@NetworkAutomation-1:~# ansible-playbook newplaybook.yml -u cisco -k
SSH password:

PLAY [Get status] ****************************************************************

TASK [Show Status] **************************************************************
changed: [R1]
changed: [R2]

TASK [debug] ********************************************************************
ok: [R2] => {
    "print_output.stdout_lines": [
        "",
        "Interface              Status      Protocol Description",
        "Fa0/0                  admin down  down        ",
        "Gi1/0                  up          up          ",
        "Gi2/0                  admin down  down        ",
        "Et3/0                  admin down  down        ",
        "Et3/1                  admin down  down        ",
        "Et3/2                  admin down  down        ",
        "Et3/3                  admin down  down        "
    ]
}
ok: [R1] => {
    "print_output.stdout_lines": [
        "",
        "Interface              Status      Protocol Description",
        "Fa0/0                  admin down  down        ",
        "Gi1/0                  up          up          ",
        "Gi2/0                  admin down  down        ",
        "Et3/0                  admin down  down        ",
        "Et3/1                  admin down  down        ",
        "Et3/2                  admin down  down        ",
        "Et3/3                  admin down  down        "
    ]
}

PLAY RECAP *********************************************************************
R1                         : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
R2                         : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

root@NetworkAutomation-1:~#
```

Here I have tried to show the interface
description of R1(Router 1) and R2(Router 2)
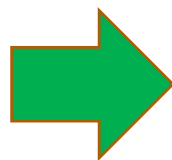
Lab Output for R1(Router 1)


```
root@NetworkAutomation-1:~# cat configplaybook.yml
---
- name: Configure Loopback Interfaces on R1 and R2
  hosts: all
  gather_facts: false
  connection: network_cli
  vars:
    ansible_network_os: ios
  tasks:
    - name: Configure Loopback Interface on R1
      ios_config:
        lines:
          - interface Loopback0
          - ip address 10.0.0.1 255.255.255.255
        parents: interface Loopback0
      register: output_R1

    - name: Configure Loopback Interface on R2
      ios_config:
        lines:
          - interface Loopback0
          - ip address 10.0.0.2 255.255.255.255
        parents: interface Loopback0
      register: output_R2

    - name: Print output from R1
      debug:
        var: output_R1

    - name: Print output from R2
      debug:
        var: output_R2
```

output

```
root@NetworkAutomation-1:~# ansible-playbook configplaybook.yml -u cisco -k
SSH password:

PLAY [Configure Loopback Interfaces on R1 and R2] ***********************

TASK [Configure Loopback Interface on R1] ******************************
changed: [R2]
changed: [R1]

TASK [Configure Loopback Interface on R2] ******************************
changed: [R2]
changed: [R1]

TASK [Print output from R1] ********************************************
ok: [R1] => {
    "output_R1": {
        "ansible_facts": {
            "discovered_interpreter_python": "/usr/bin/python3"
        },
        "banners": {},
        "changed": true,
        "commands": [
            "interface Loopback0",
            "interface Loopback0",
            "ip address 10.0.0.1 255.255.255.255"
        ],
        "failed": false,
        "updates": [
            "interface Loopback0",
            "interface Loopback0",
            "ip address 10.0.0.1 255.255.255.255"
        ]
    }
}
ok: [R2] => {
    "output_R1": {
        "ansible_facts": {
            "discovered_interpreter_python": "/usr/bin/python3"
        },
        "banners": {},
        "changed": true,
        "commands": [
            "interface Loopback0",
            "interface Loopback0",
            "ip address 10.0.0.1 255.255.255.255"
        ],
        "failed": false,
        "updates": [
            "interface Loopback0",
            "interface Loopback0",
            "ip address 10.0.0.1 255.255.255.255"
        ]
    }
}
```

Here I changed the loopback IP of R1 via Playbook and print the output as well
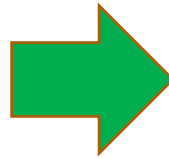
Lab Output For R2 (Router 2)


```
root@NetworkAutomation-1:~# cat configplaybook.yml
---
- name: Configure Loopback Interfaces on R1 and R2
  hosts: all
  gather_facts: false
  connection: network_cli
  vars:
    ansible_network_os: ios
  tasks:
    - name: Configure Loopback Interface on R1
      ios_config:
        lines:
          - interface Loopback0
          - ip address 10.0.0.1 255.255.255.255
        parents: interface Loopback0
      register: output_R1

    - name: Configure Loopback Interface on R2
      ios_config:
        lines:
          - interface Loopback0
          - ip address 10.0.0.2 255.255.255.255
        parents: interface Loopback0
      register: output_R2

    - name: Print output from R1
      debug:
        var: output_R1

    - name: Print output from R2
      debug:
        var: output_R2
```

output →

```
TASK [Print output from R2] ********************************************************
ok: [R1] => {
    "output_R2": {
        "banners": {},
        "changed": true,
        "commands": [
            "interface Loopback0",
            "interface Loopback0",
            "ip address 10.0.0.2 255.255.255.255"
        ],
        "failed": false,
        "updates": [
            "interface Loopback0",
            "interface Loopback0",
            "ip address 10.0.0.2 255.255.255.255"
        ]
    }
}
ok: [R2] => {
    "output_R2": {
        "banners": {},
        "changed": true,
        "commands": [
            "interface Loopback0",
            "interface Loopback0",
            "ip address 10.0.0.2 255.255.255.255"
        ],
        "failed": false,
        "updates": [
            "interface Loopback0",
            "interface Loopback0",
            "ip address 10.0.0.2 255.255.255.255"
        ]
    }
}

PLAY RECAP ************************************************************************
R1                         : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
R2                         : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

root@NetworkAutomation-1:~#
```

Here I changed the loopback IP of R2 via Playbook and print the output as well

Is APRICOT only about the Workshop?

There is a big part of my fellowship is socializing

Get to know network engineer's from different countries and communicate with them, share the knowledge.

And now ready to grab the opportunity to share our knowledge as well.

Thank You