TECHNOLOGICAL INSTITUTE OF THE PHILIPPINES

938 Aurora Blvd., Cubao, Quezon City

COMPUTER ENGINEERING DEPARTMENT

1st Semester S.Y. 2020 – 2021

FINAL CASE STUDY

Junaid M. Bantuas

Student

Engr. Alonica R. Villanueva

Instructor

Network Automation

Objectives:
Part 1: Launch the DEVASC V and CSR1kv
Part 2: Configure VLAN for Yaml
Part 3: Use ansible to utilize configured yaml file
Part 4: Use the pyATS to test the network.

Background / Scenario

In this procedure, we will use ansible playbooks to automate the network implementation process. Ansible is an open source IT automation platform that automates manual IT activities including provisioning, configuration management, application deployment, orchestration, and more. After implementing the configurations, we will use pyATS to test the network.

Required Resources

- 1 PC with operating system of your choice
- DEVASC Virtual Machine
- CSR1kv Virtual Machine

Planned Automation: Implementation of VLANS

Initial Design of Network structure on Packet Tracer

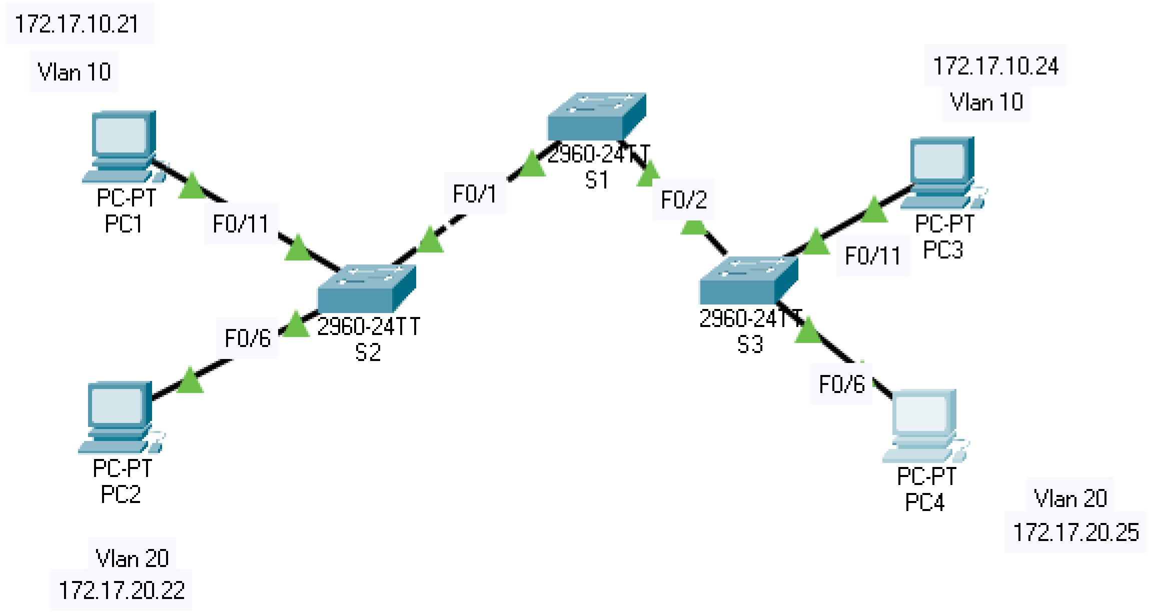


Figure 1.0 Initial Draft Design

This design was first tested in packet tracer to ensure that the planned system would work when it is finalized for use in the actual case study.

Table 1.0 Addressing Table

Device	Interface	Ip Address	Subnet Mask	Default Gateway
S1	VLAN 99	172.17.99.11	255.255.255.0	N/A
S2	VLAN 99	172.17.99.12	255.255.255.0	N/A
S3	VLAN 99	172.17.99.13	255.255.255.0	N/A
PC1	NIC	172.17.10.21	255.255.255.0	172.17.10.1
PC2	NIC	172.17.20.22	255.255.255.0	172.17.20.1
PC3	NIC	172.17.10.24	255.255.255.0	172.17.10.1
PC4	NIC	172.17.20.25	255.255.255.0	172.17.20.1

Table 1.1 Port Assignments

Ports Assignment Network	Ports Assignment Network	Ports Assignment Network
Fa0/1 – 0/5	802.1q Trunks (Native VLAN 99)	172.17.99.0 /24
Fa0/6 – 0/10	VLAN 20 – Students	172.17.20.0 /24
Fa0/11 – 0/17	VLAN 10 – Faculty	172.17.10.0 /24

Ansible Automation Code

```
---
- name: NETWORK AUTOMATION VLANS
  hosts: CSR1kv
  gather_facts: false
  connection: local


  tasks:
```

```yaml
- name: create vlan 10
  ios_config:
    parents:
      - vlan 10
    lines:
      - name Faculty
- name: create vlan 20
  ios_config:
    parents:
      - vlan 20
    lines:
      - name Students
- name: create vlan 99
  ios_config:
    parents:
      - vlan 99
    lines:
      - name Management

- name: show vlan brief
  ios_command:
    commands:
      - show vlan brief
  register: vlan_brief

- name: SAVE OUTPUT to ./backups/
  copy:
    content: "{{ config.stdout[0] }}"
    dest: "backups/show_run_{{ inventory_hostname }}.txt"
```

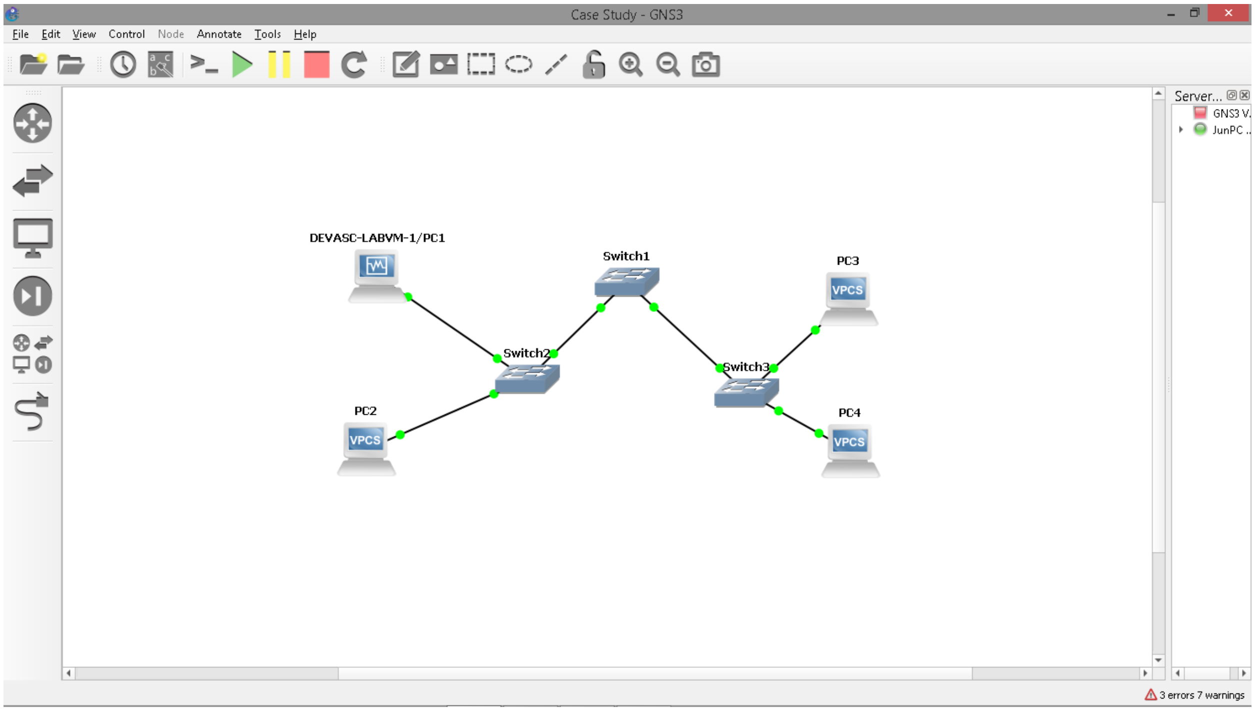
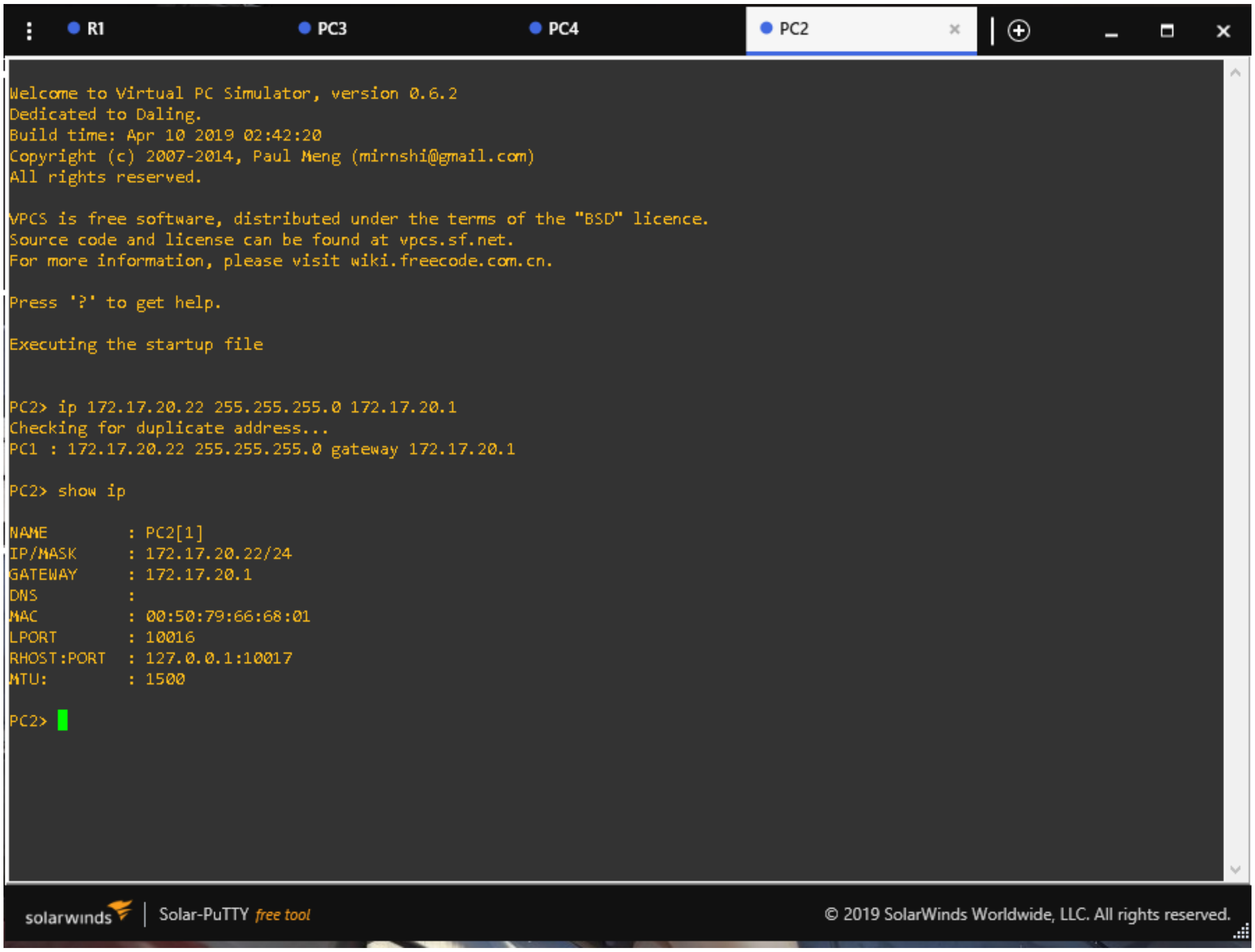


Figure 1.1 Implementing on GNS3

There are 4 PCs, with PC1 being the DEVASC Machine. With 3 switches connecting different sites. PC1 and PC3 are VLAN 10 while PC2 and PC4 are VLAN 20.

PART 1. IMPLEMENTING IP ADDRESS ON VPCs

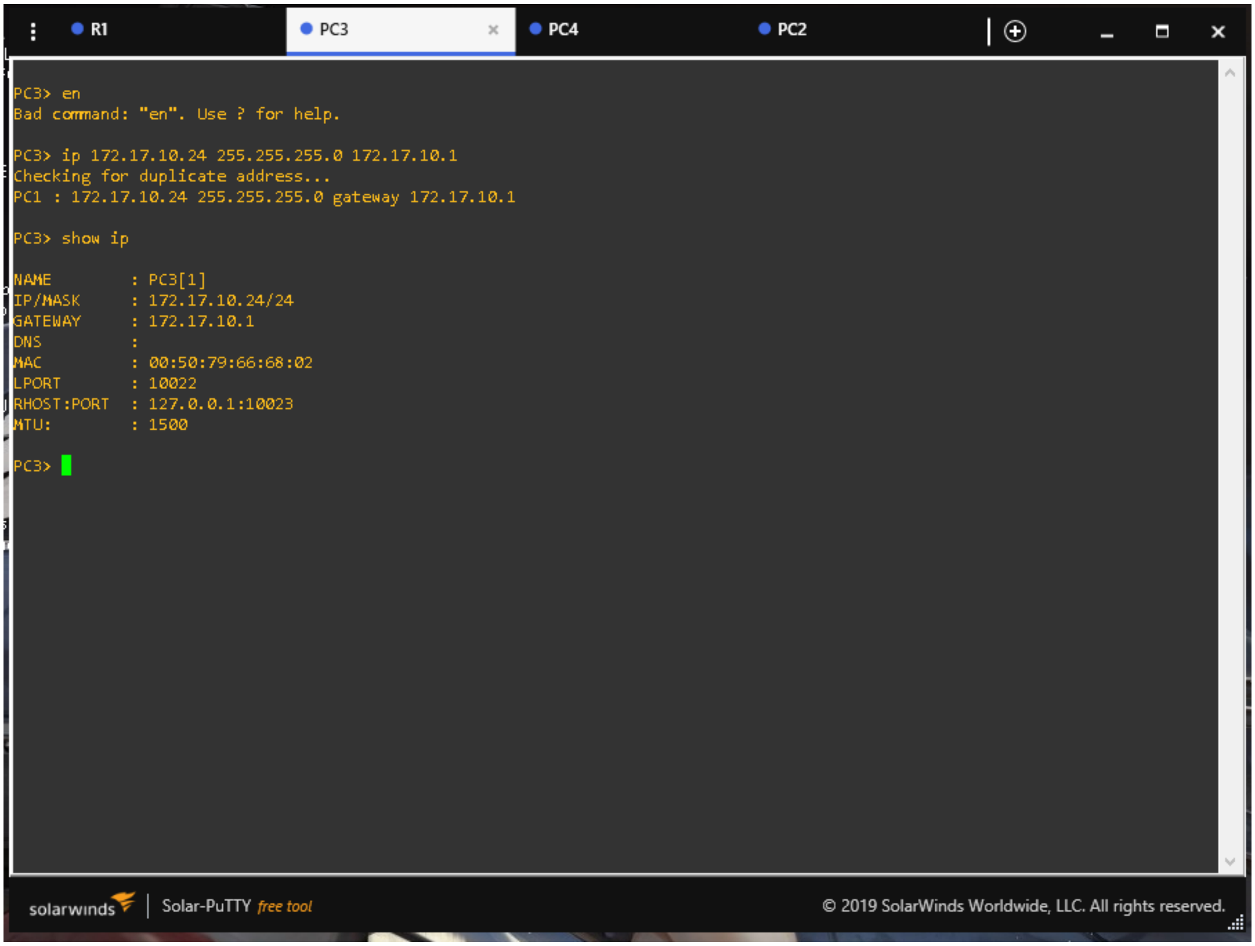
PC2



```
Welcome to Virtual PC Simulator, version 0.6.2
Dedicated to Daling.
Build time: Apr 10 2019 02:42:20
Copyright (c) 2007-2014, Paul Meng (mirnshi@gmail.com)
All rights reserved.

VPCS is free software, distributed under the terms of the "BSD" licence.
Source code and license can be found at vpcs.sf.net.
For more information, please visit wiki.freecode.com.cn.

Press '?' to get help.

Executing the startup file


PC2> ip 172.17.20.22 255.255.255.0 172.17.20.1
Checking for duplicate address...
PC1 : 172.17.20.22 255.255.255.0 gateway 172.17.20.1

PC2> show ip

NAME        : PC2[1]
IP/MASK     : 172.17.20.22/24
GATEWAY     : 172.17.20.1
DNS         :
MAC         : 00:50:79:66:68:01
LPORT       : 10016
RHOST:PORT  : 127.0.0.1:10017
MTU:        : 1500

PC2>
```

Figure 1.2 Implementing address on PC2

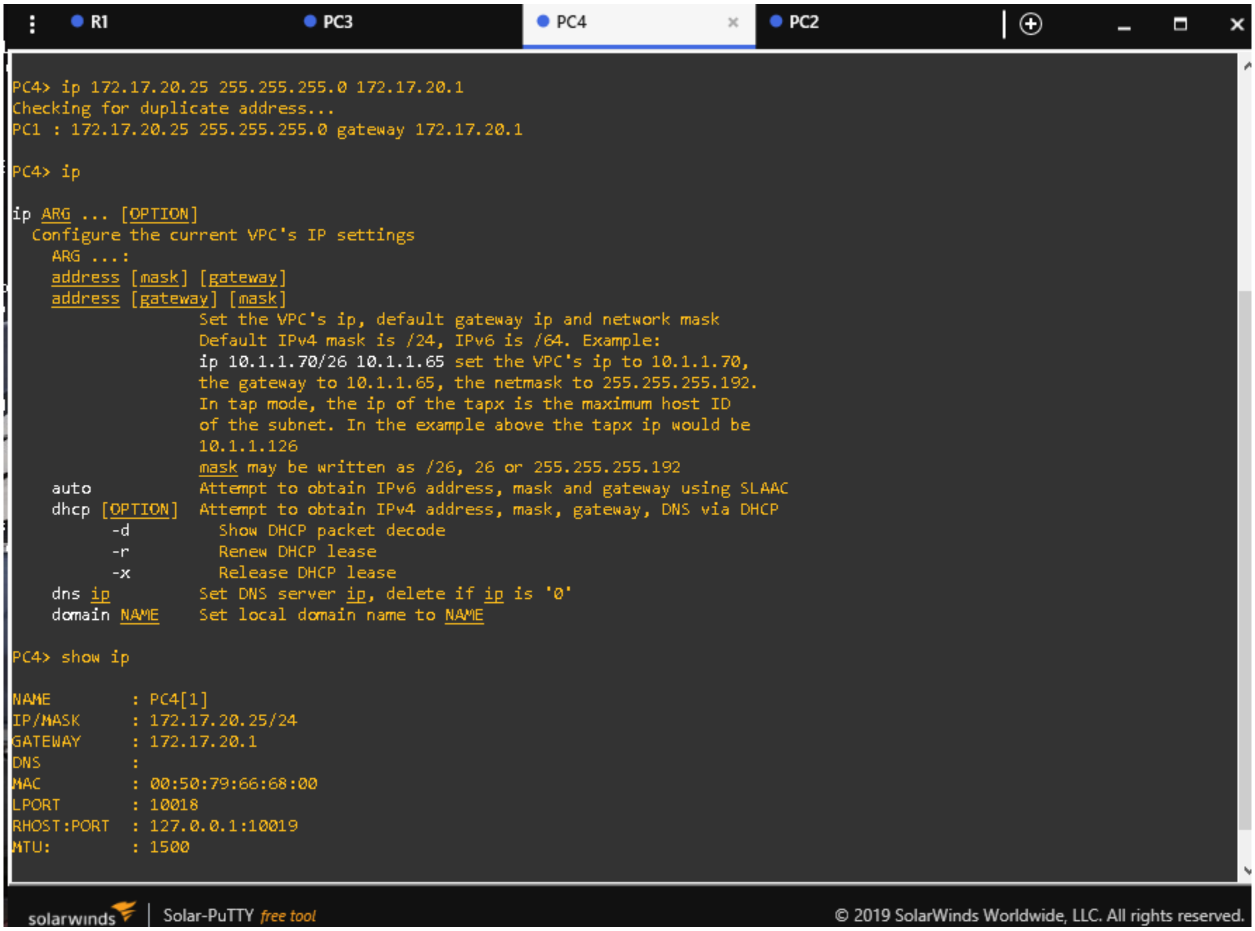
PC3



```
PC3> en
Bad command: "en". Use ? for help.

PC3> ip 172.17.10.24 255.255.255.0 172.17.10.1
Checking for duplicate address...
PC1 : 172.17.10.24 255.255.255.0 gateway 172.17.10.1

PC3> show ip

NAME        : PC3[1]
IP/MASK     : 172.17.10.24/24
GATEWAY     : 172.17.10.1
DNS         :
MAC         : 00:50:79:66:68:02
LPORT       : 10022
RHOST:PORT  : 127.0.0.1:10023
MTU:        : 1500

PC3>
```

Figure 1.3 Implementing address on PC3

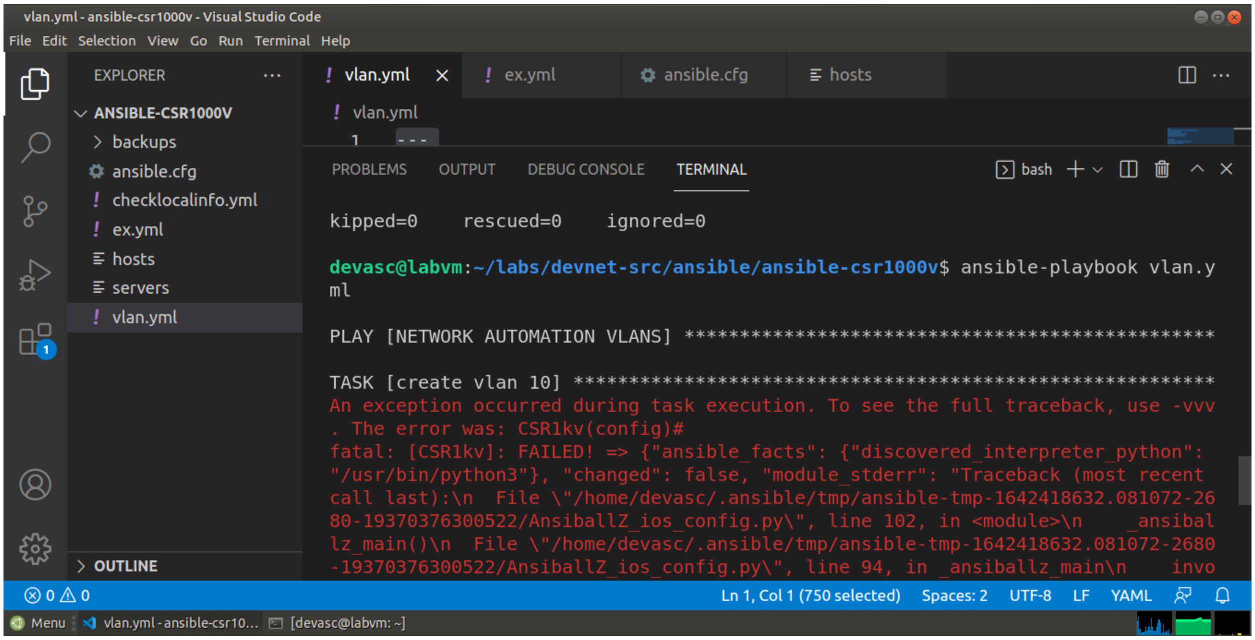
PC4



```
PC4> ip 172.17.20.25 255.255.255.0 172.17.20.1
Checking for duplicate address...
PC1 : 172.17.20.25 255.255.255.0 gateway 172.17.20.1

PC4> ip

ip ARG ... [OPTION]
  Configure the current VPC's IP settings
    ARG ...:
    address [mask] [gateway]
    address [gateway] [mask]
                   Set the VPC's ip, default gateway ip and network mask
                   Default IPv4 mask is /24, IPv6 is /64. Example:
                   ip 10.1.1.70/26 10.1.1.65 set the VPC's ip to 10.1.1.70,
                   the gateway to 10.1.1.65, the netmask to 255.255.255.192.
                   In tap mode, the ip of the tapx is the maximum host ID
                   of the subnet. In the example above the tapx ip would be
                   10.1.1.126
                   mask may be written as /26, 26 or 255.255.255.192
    auto           Attempt to obtain IPv6 address, mask and gateway using SLAAC
    dhcp [OPTION]  Attempt to obtain IPv4 address, mask, gateway, DNS via DHCP
          -d         Show DHCP packet decode
          -r         Renew DHCP lease
          -x         Release DHCP lease
    dns ip         Set DNS server ip, delete if ip is '0'
    domain NAME    Set local domain name to NAME

PC4> show ip

NAME        : PC4[1]
IP/MASK     : 172.17.20.25/24
GATEWAY     : 172.17.20.1
DNS         :
MAC         : 00:50:79:66:68:00
LPORT       : 10018
RHOST:PORT  : 127.0.0.1:10019
MTU:        : 1500
```

Figure 1.4 Implementing address on PC4

PART 2. USE ANSIBLE PLAYBOOK TO IMPLEMENT THE YAML



```
kipped=0    rescued=0    ignored=0

devasc@labvm:~/labs/devnet-src/ansible/ansible-csr1000v$ ansible-playbook vlan.y
ml

PLAY [NETWORK AUTOMATION VLANS] ************************************************

TASK [create vlan 10] **********************************************************
An exception occurred during task execution. To see the full traceback, use -vvv
. The error was: CSR1kv(config)#
fatal: [CSR1kv]: FAILED! => {"ansible_facts": {"discovered_interpreter_python":
"/usr/bin/python3"}, "changed": false, "module_stderr": "Traceback (most recent
call last):\n  File \"/home/devasc/.ansible/tmp/ansible-tmp-1642418632.081072-26
80-19370376300522/AnsiballZ_ios_config.py\", line 102, in <module>\n    _ansibal
lz_main()\n  File \"/home/devasc/.ansible/tmp/ansible-tmp-1642418632.081072-2680
-19370376300522/AnsiballZ_ios_config.py\", line 94, in _ansiballz_main\n    invo
```

Figure 1.5 Implementing the yaml playbook.

Conclusion:

Unfortunately, implementing the playbook seems to have encountered an error in order to fix some of the issues, ansible documentation was tackled to read on proper vlan configuration and implementation but reached to no avail. Though I have understood how to use ansible playbook to implement network automation as it requires the same process to perform it. We also failed to utilize the pyATS as there are no network to test.

VIDEO LINK:

https://drive.google.com/file/d/1VM7Mgs6TytQB5XPbwpAF0LVjofyKUg0y/view?usp=sharing

GITHUB LINK:

https://github.com/JunaidBantuas/FINAL-CASE-STUDY.git

HONOR PLEDGE:

“I affirm that I will not give or receive any unauthorized help on this exam, and that all work will be my own.”