

Network Engineering Capstone Functionality Report

Introduction

Provide a functionality report detailing the ten test case scenarios used to verify the utility of your network project. Seven of the test case scenarios must be from the provided predefined list, with the remaining three test cases created by you. The functionality report should be written so that a networking peer could replicate the steps for a successful test of your networking solution.

Student Name	[REDACTED]
WGU Student ID	[REDACTED]
WGU Student Email	[REDACTED]



WESTERN GOVERNORS UNIVERSITY.

Test Case #1: Device Discovery and Reachability

Your network solution must include multiple network segments with access controls that allow traffic from a device on one network to access the resources of a device on another network. Similarly, there must be devices on one network that cannot access resources on a different network.

Functionality

Describe the **functionality** of the test case in relation to your network project. Identify the relevant tools (devices, subnets, etc.) used in this test case and their specific interactions.

The network is divided into several networks, both internally, and in the mock internet. The internal networks include network device networks between network devices, and endpoint networks which are assigned VLANs and utilize DHCP to obtain IP addresses.

The internal network device networks are:

EdgeRouter1 <-> L3Switch1 10.0.0.4/30

EdgeRouter1 <-> L3Switch2 10.0.0.8/30

L3Switch1 <-> L3Switch2 10.0.0.40/30

The internal endpoint networks are:

Admin Network 10.0.0.16/29 – VLAN 10

Office Network 10.0.0.24/29 – VLAN 20

Guest Network 10.0.0.32 - VLAN 30

L3Switch Tagging VLAN 40 10.0.0.40/30

The mock internet networks are:

ISP <-> EdgeRouter1 8.8.8.0/30

ISP <-> SasSPlatform 4.4.4.0/30

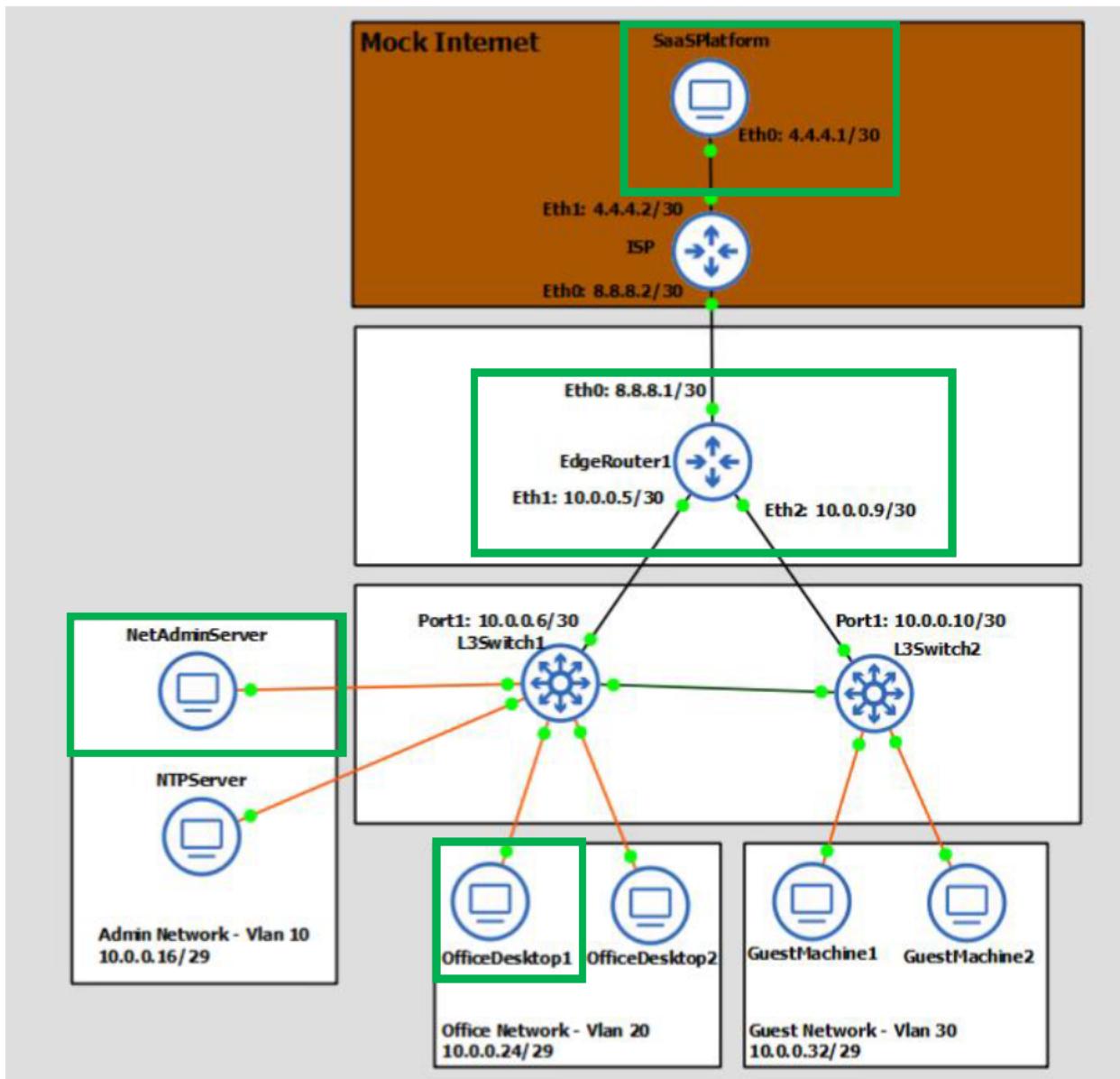
Network Diagram or Segment

Provide a **network diagram or segment** visualizing the topology and devices used in this test case.

All network devices are applicable to this test case, but the ones that are directly applicable to the testing method are enclosed in green boxes.



WESTERN GOVERNORS UNIVERSITY



Legend

SaaS	Software as a Service
Eth*	Ethernet + port number
ISP	Internet Service Provider
L3	Layer 3
NetAdminServer	Network Administration Server
Admin	Administration
VLAN	Virtual Local Area Network

Testing Method

Summarize the **testing method** used to verify functionality of the network project within the virtual lab environment, including any metrics of success.



WESTERN GOVERNORS UNIVERSITY®

There are several ACLs in place that both deny access and allow access. To demonstrate access to another network, I will show that OfficeDesktop1 can access SaaSPlatform by creating a HTTP server on SaaSPlatform and connecting to it over port 80 on OfficeDesktop1. OfficeDesktop1 is on the 10.0.0.24/29 network, while SaaSPlatform is on the 4.4.4.0/30 network. This connectivity is enabled because traffic is only denied to other VLANs from OfficeDesktop1, and the ACL on EdgeRouter1 eth0 inbound allows traffic back from SaaSPlatform if a TCP connection has already been established.

To demonstrate a case where resources is denied access to one on a different network, I will show that SaaSPlatform cannot access a HTTP server on NetAdminServer. NetAdminServer is on the 10.0.0.16/29 network and SaaSPlatform is on the 4.4.4.0/30 network. This connectivity is denied because EdgeRouter1 eth0 inbound denies all new TCP connections.

Process List

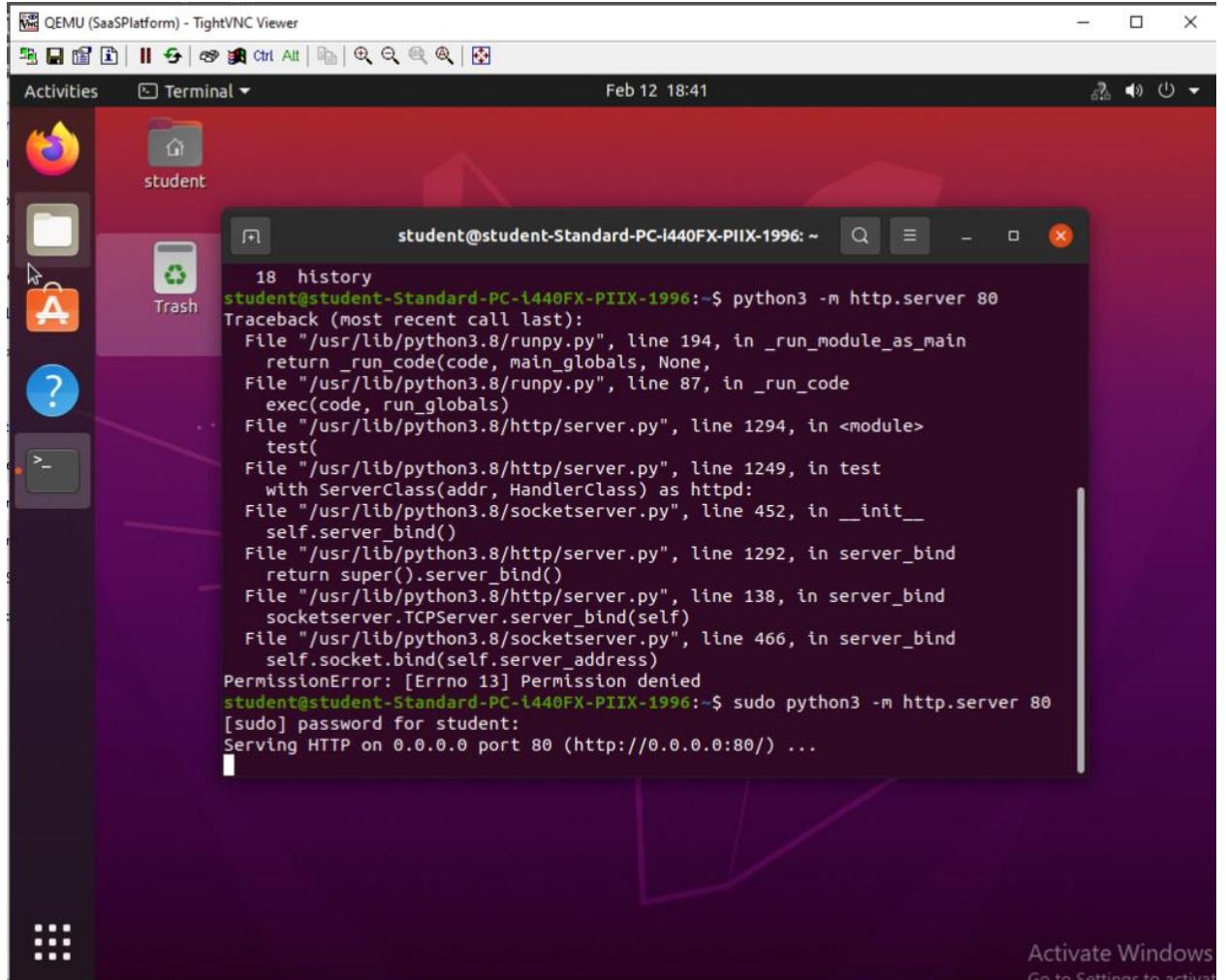
*Provide a comprehensive **process list** of the steps taken within the network project to run the testing method. Include screenshots to illustrate the process and ensure clarity for others attempting to replicate the test.*

Access to resource on another network:

1. Login to SaaSPlatform.
2. Open a terminal.
3. Run "sudo python3 -m http.server 80" to start an http server.



WESTERN GOVERNORS UNIVERSITY.

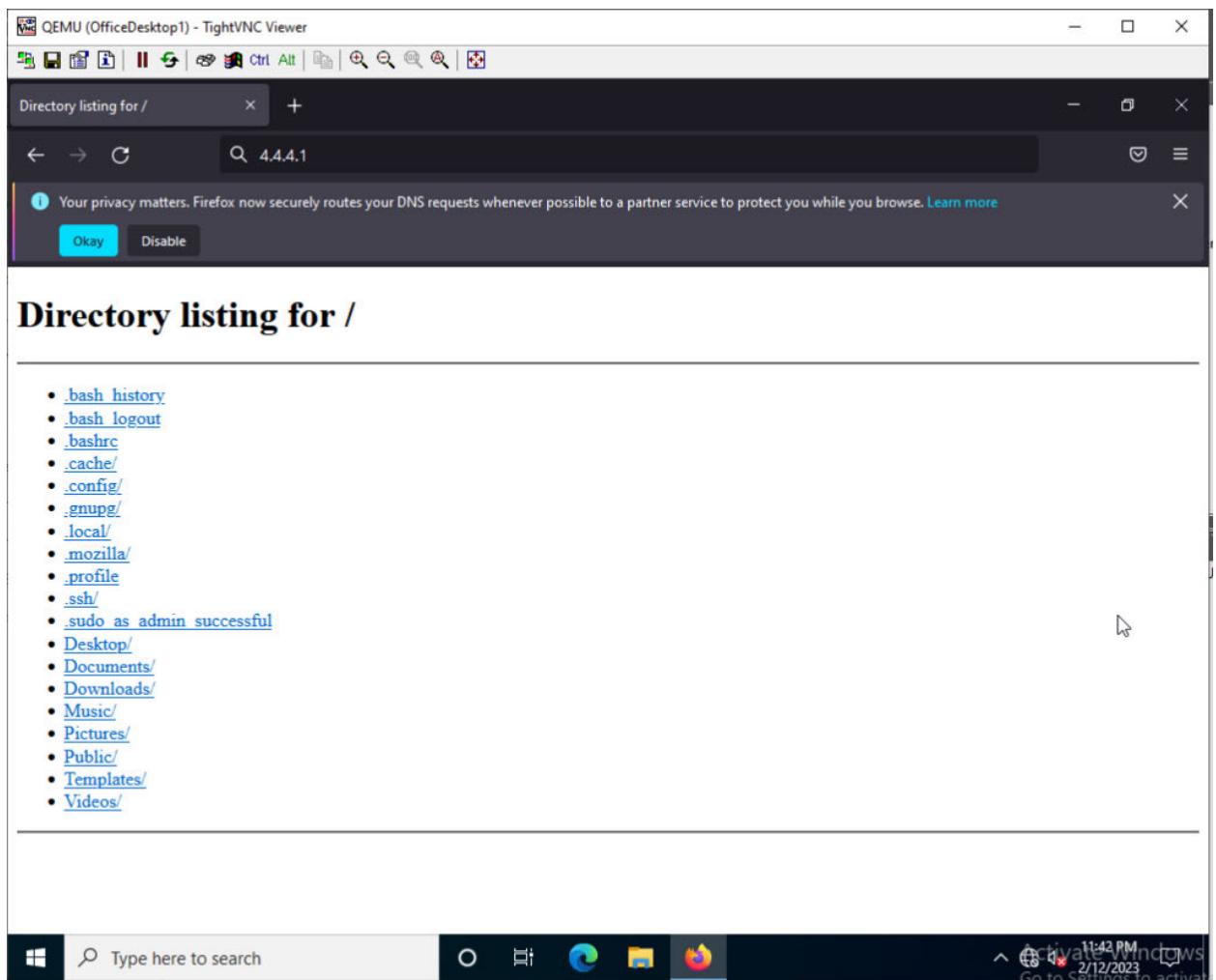


The screenshot shows a Linux desktop environment with a purple-themed interface. A terminal window is open, displaying the following command-line session:

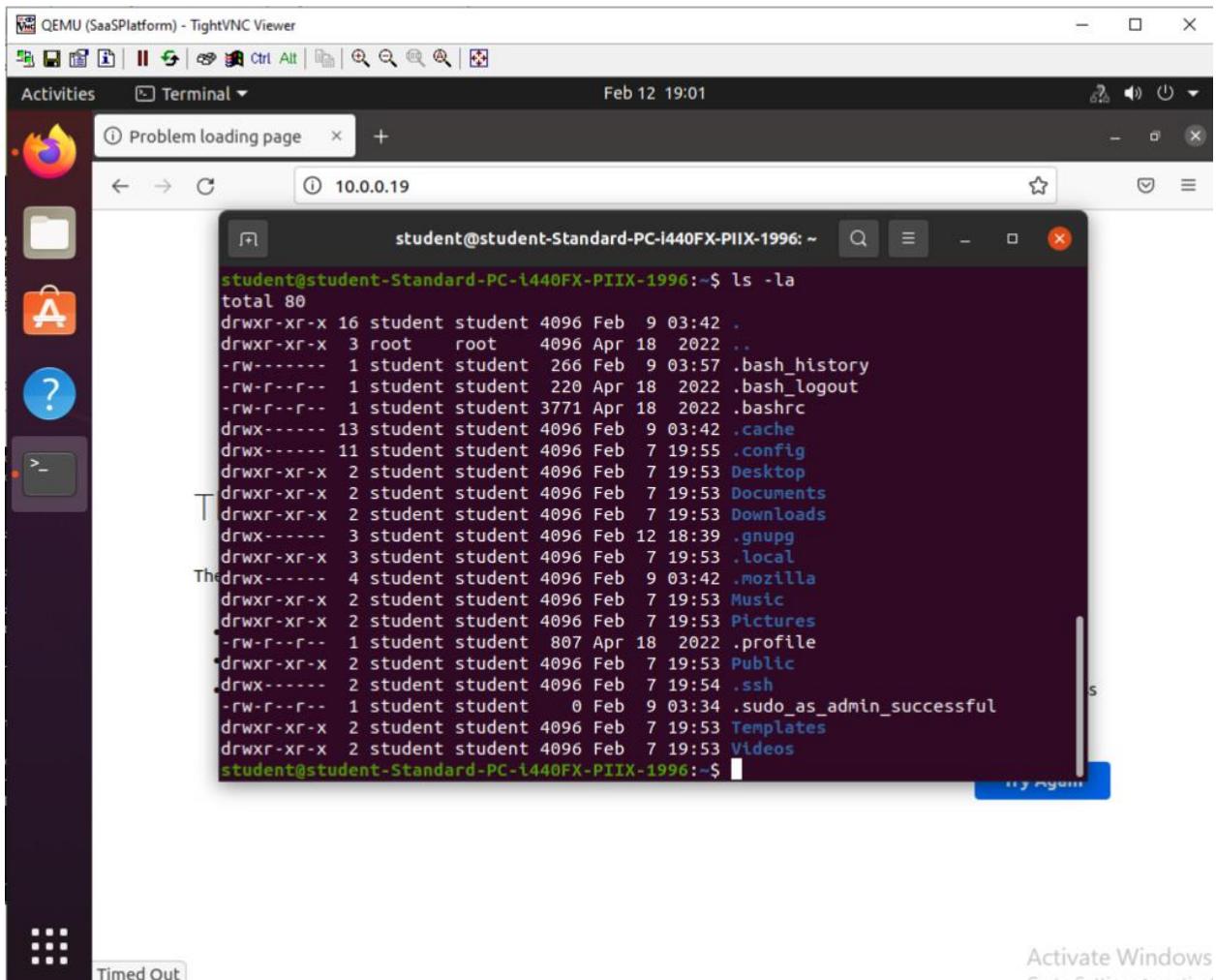
```
student@student-Standard-PC-i440FX-PIIX-1996: ~
18 history
student@student-Standard-PC-i440FX-PIIX-1996:~$ python3 -m http.server 80
Traceback (most recent call last):
  File "/usr/lib/python3.8/runpy.py", line 194, in _run_module_as_main
    return _run_code(code, main_globals, None,
  File "/usr/lib/python3.8/runpy.py", line 87, in _run_code
    exec(code, run_globals)
  File "/usr/lib/python3.8/http/server.py", line 1294, in <module>
    test()
  File "/usr/lib/python3.8/http/server.py", line 1249, in test
    with ServerClass(addr, HandlerClass) as httpd:
  File "/usr/lib/python3.8/socketserver.py", line 452, in __init__
    self.server_bind()
  File "/usr/lib/python3.8/http/server.py", line 1292, in server_bind
    return super().server_bind()
  File "/usr/lib/python3.8/http/server.py", line 138, in server_bind
    socketserver.TCPServer.server_bind(self)
  File "/usr/lib/python3.8/socketserver.py", line 466, in server_bind
    self.socket.bind(self.server_address)
PermissionError: [Errno 13] Permission denied
student@student-Standard-PC-i440FX-PIIX-1996:~$ sudo python3 -m http.server 80
[sudo] password for student:
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```

4. Login to OfficeDesktop1.
5. Open Firefox.
6. Connect to 4.4.4.1:80. It is successful and we can see the contents of the HTTP servers working directory.





WESTERN GOVERNORS UNIVERSITY[®]

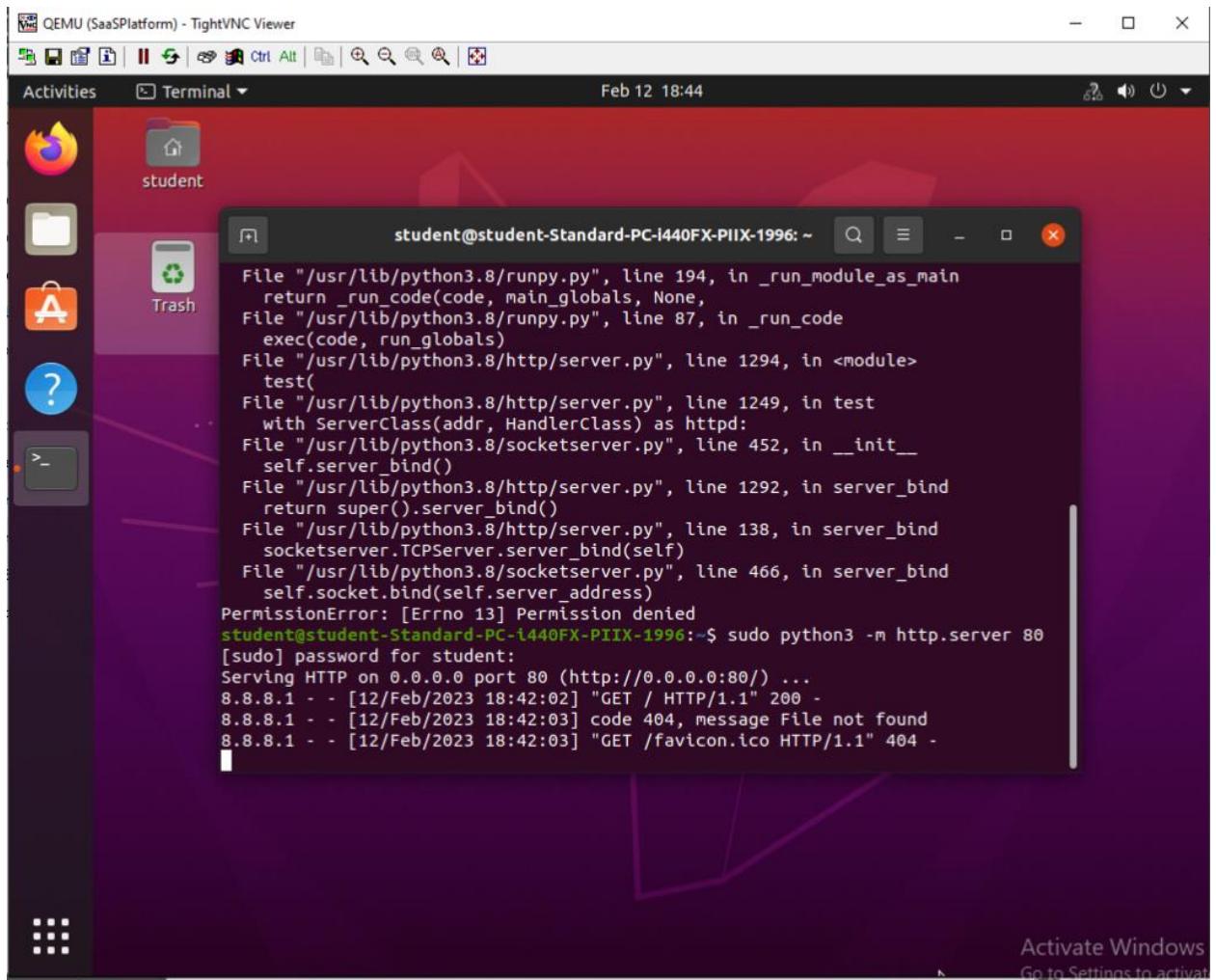


The screenshot shows a terminal window titled "student@student-Standard-PC-i440FX-PIIX-1996: ~" running on a Linux desktop environment. The terminal displays the output of the command "ls -la", listing the contents of the current directory (~). The output shows various files and directories with their permissions, ownership, and timestamps. The desktop interface includes a dock with icons for a file manager, a browser, and other applications, and a system tray at the bottom.

```
student@student-Standard-PC-i440FX-PIIX-1996:~$ ls -la
total 80
drwxr-xr-x 16 student student 4096 Feb  9 03:42 .
drwxr-xr-x  3 root   root   4096 Apr 18 2022 ..
-rw-r--r--  1 student student 266 Feb  9 03:57 .bash_history
-rw-r--r--  1 student student 220 Apr 18 2022 .bash_logout
-rw-r--r--  1 student student 3771 Apr 18 2022 .bashrc
drwx----- 13 student student 4096 Feb  9 03:42 .cache
drwx----- 11 student student 4096 Feb  7 19:55 .config
drwxr-xr-x  2 student student 4096 Feb  7 19:53 Desktop
drwxr-xr-x  2 student student 4096 Feb  7 19:53 Documents
drwxr-xr-x  2 student student 4096 Feb  7 19:53 Downloads
drwx-----  3 student student 4096 Feb 12 18:39 .gnupg
drwxr-xr-x  3 student student 4096 Feb  7 19:53 .local
drwx-----  4 student student 4096 Feb  9 03:42 .mozilla
drwxr-xr-x  2 student student 4096 Feb  7 19:53 Music
drwxr-xr-x  2 student student 4096 Feb  7 19:53 Pictures
-rw-r--r--  1 student student 807 Apr 18 2022 .profile
drwxr-xr-x  2 student student 4096 Feb  7 19:53 Public
drwx-----  2 student student 4096 Feb  7 19:54 .ssh
-rw-r--r--  1 student student    0 Feb  9 03:34 .sudo_as_admin_successful
drwxr-xr-x  2 student student 4096 Feb  7 19:53 Templates
drwxr-xr-x  2 student student 4096 Feb  7 19:53 Videos
student@student-Standard-PC-i440FX-PIIX-1996:~$
```

7. Connection is seen on 4.4.4.1 from the company's NAT public IP address 8.8.8.1.





The screenshot shows a Linux desktop environment with a purple gradient background. A terminal window titled "student" is open, displaying the following command-line session:

```
student@student-Standard-PC-i440FX-PIIX-1996: ~
File "/usr/lib/python3.8/runpy.py", line 194, in _run_module_as_main
    return _run_code(code, main_globals, None,
File "/usr/lib/python3.8/runpy.py", line 87, in _run_code
    exec(code, run_globals)
File "/usr/lib/python3.8/http/server.py", line 1294, in <module>
    test(
File "/usr/lib/python3.8/http/server.py", line 1249, in test
    with ServerClass(addr, HandlerClass) as httpd:
File "/usr/lib/python3.8/socketserver.py", line 452, in __init__
    self.server_bind()
File "/usr/lib/python3.8/http/server.py", line 1292, in server_bind
    return super().server_bind()
File "/usr/lib/python3.8/http/server.py", line 138, in server_bind
    socketserver.TCPServer.server_bind(self)
File "/usr/lib/python3.8/socketserver.py", line 466, in server_bind
    self.socket.bind(self.server_address)
PermissionError: [Errno 13] Permission denied
student@student-Standard-PC-i440FX-PIIX-1996:~$ sudo python3 -m http.server 80
[sudo] password for student:
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
8.8.8.1 - - [12/Feb/2023 18:42:02] "GET / HTTP/1.1" 200 -
8.8.8.1 - - [12/Feb/2023 18:42:03] "code 404, message File not found"
8.8.8.1 - - [12/Feb/2023 18:42:03] "GET /favicon.ico HTTP/1.1" 404 -
```

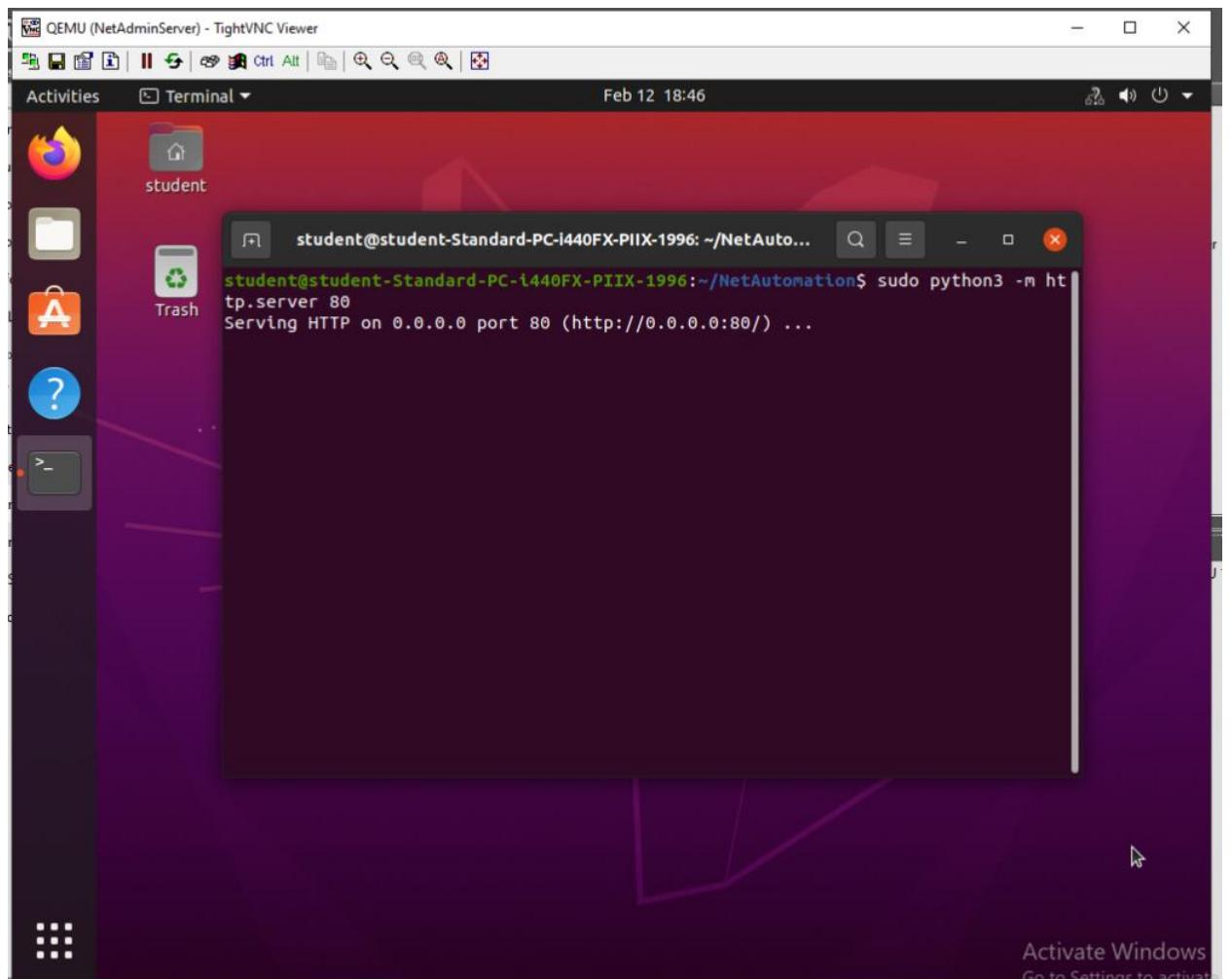
The terminal window has a dark theme with white text. The desktop interface includes a dock with icons for a browser, file manager, terminal, and system tools. A system tray at the top shows standard icons for network, battery, and volume. The bottom right corner features a watermark for "Activate Windows" and a link to "Go to Settings to activate".

Deny access to another network:

1. Login to NetAdminServer.
2. Open a terminal.
3. Run "sudo python3 -m http.server 80" to start an HTTP server.



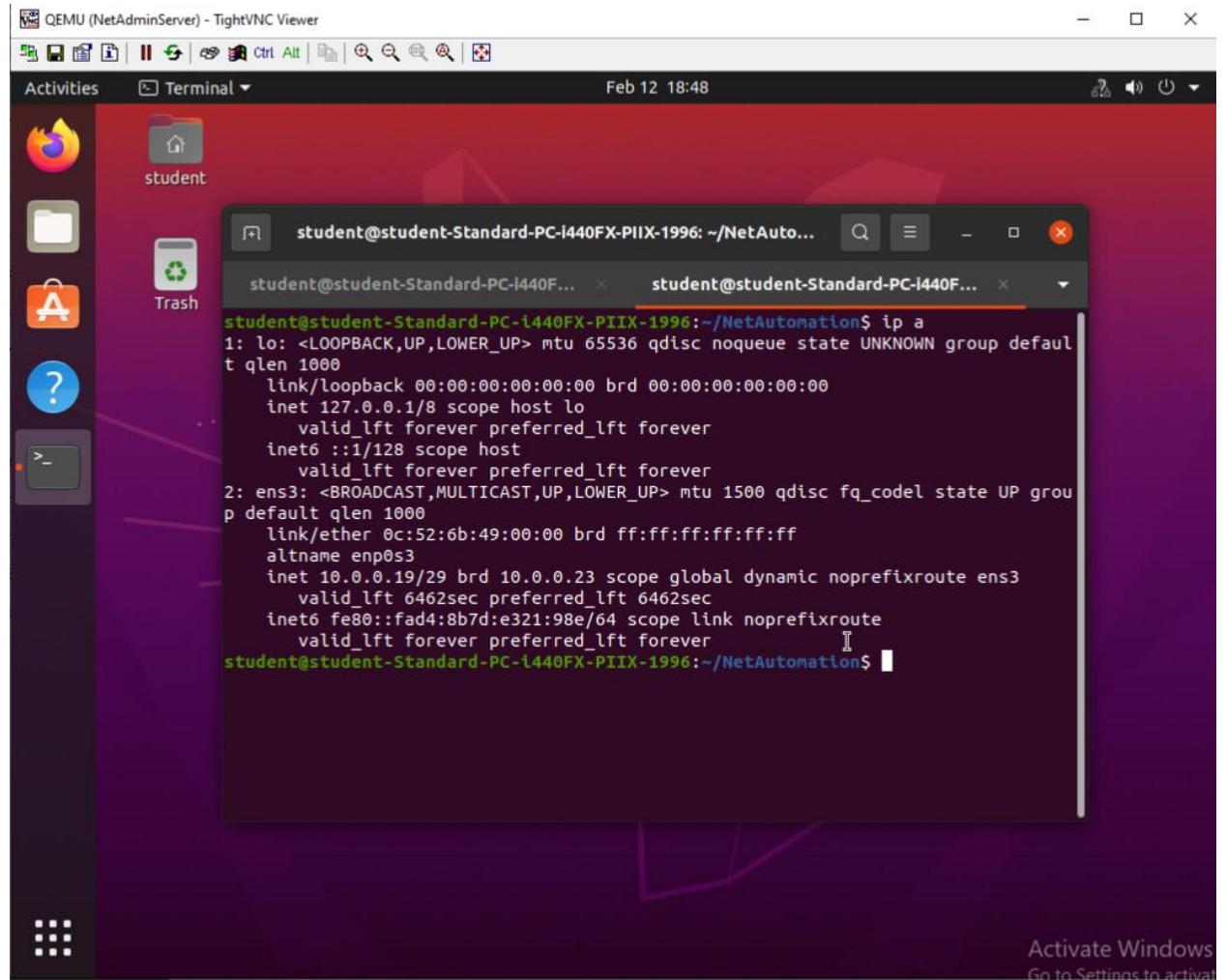
WESTERN GOVERNORS UNIVERSITY



4. Get the IP address of NetAdminServer with "ip a".



WESTERN GOVERNORS UNIVERSITY

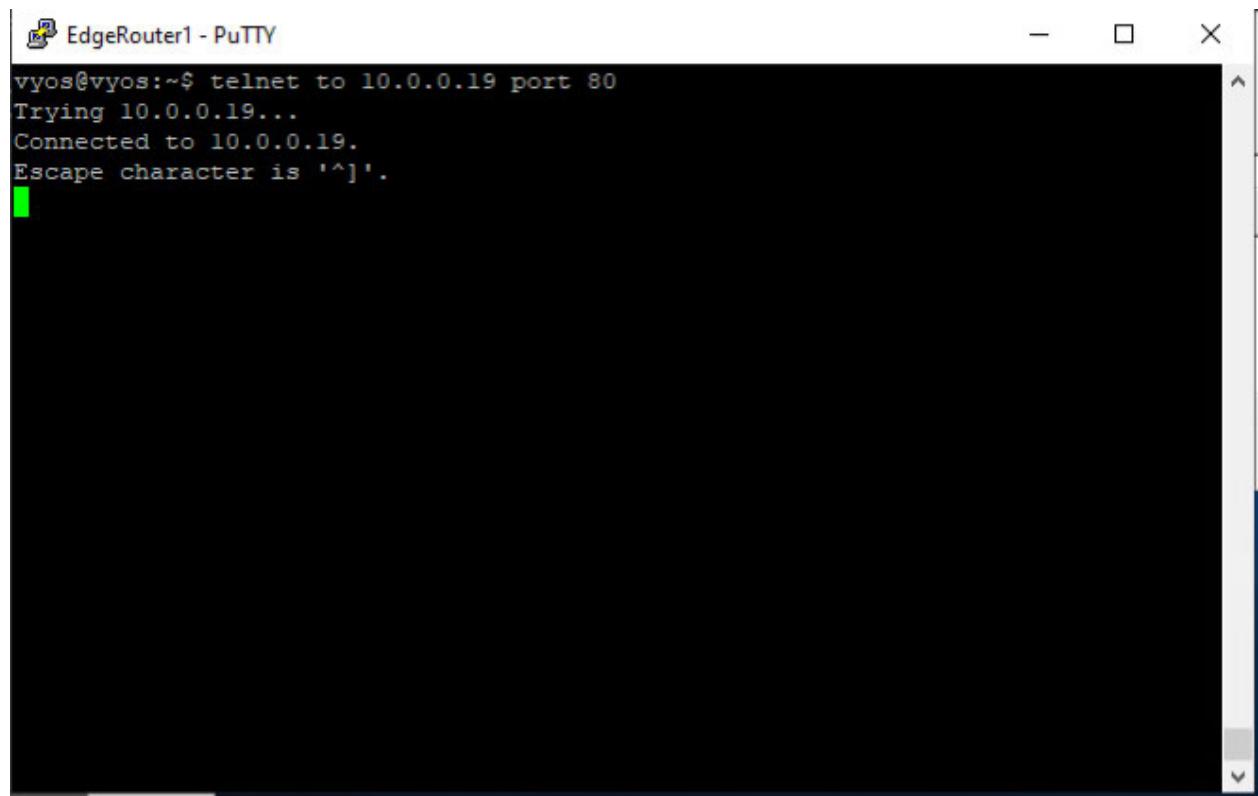


The screenshot shows a Linux desktop environment with a purple background. A terminal window titled "student@student-Standard-PC-i440FX-PIIX-1996: ~/NetAutomation" is open, displaying the output of the command "ip a". The terminal shows two network interfaces: "lo" (loopback) and "ens3" (ethernet). The "lo" interface has an IP of 127.0.0.1/8. The "ens3" interface has an IP of 10.0.0.19/29 and is connected to a dynamic route to 10.0.0.23 via "ens3". The "inet6" entry for "ens3" shows a link-local address fe80::fad4:8b7d:e321:98e/64.

```
student@student-Standard-PC-i440FX-PIIX-1996:~/NetAutomation$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 0c:52:6b:49:00:00 brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    inet 10.0.0.19/29 brd 10.0.0.23 scope global dynamic noprefixroute ens3
        valid_lft 6462sec preferred_lft 6462sec
    inet6 fe80::fad4:8b7d:e321:98e/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
student@student-Standard-PC-i440FX-PIIX-1996:~/NetAutomation$
```

5. Login to SaaSPlatform.
6. Open Firefox.
7. Try to access the HTTP server on NetAdminServer's IP address at 10.0.0.19.
8. To verify that this HTTP server works, I run "telnet to 10.0.0.19 port 80" on EdgeRouter1. It successfully connects.





The screenshot shows a PuTTY terminal window titled "EdgeRouter1 - PuTTY". The window contains the following text:

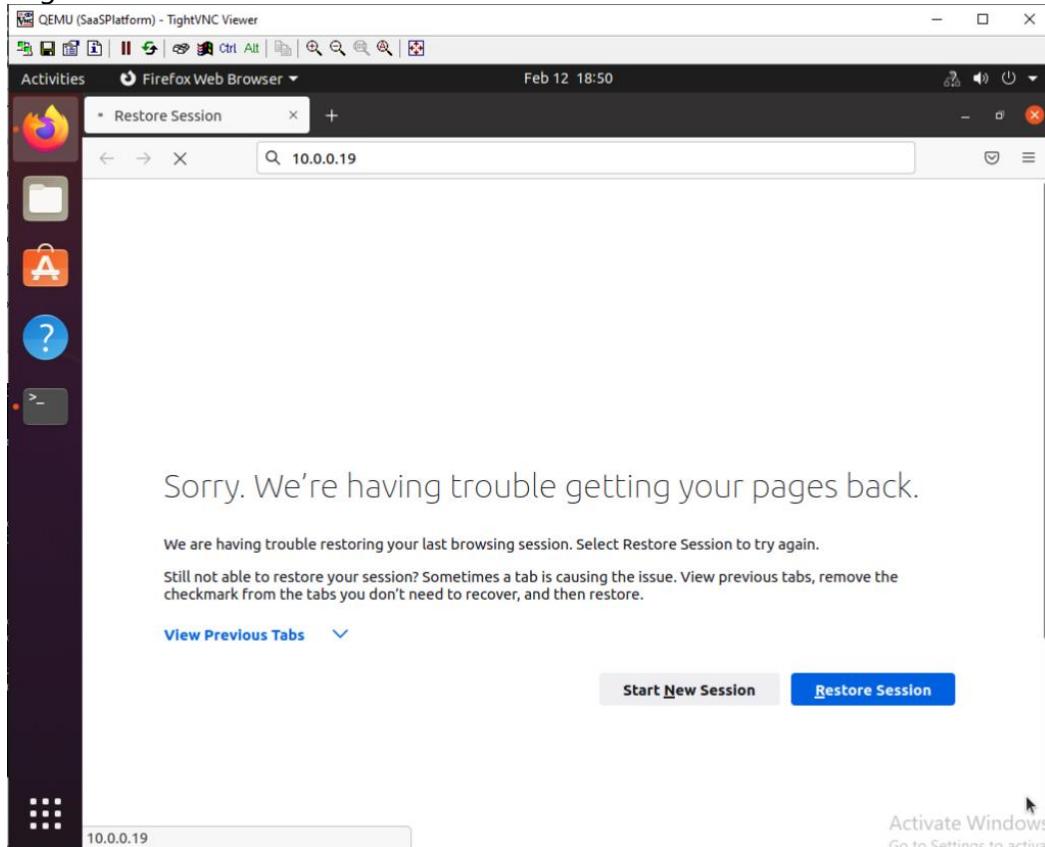
```
vyos@vyos:~$ telnet to 10.0.0.19 port 80
Trying 10.0.0.19...
Connected to 10.0.0.19.
Escape character is '^]'.  
[REDACTED]
```

The terminal window has standard window controls (minimize, maximize, close) at the top right. A vertical scroll bar is visible on the right side of the terminal window.



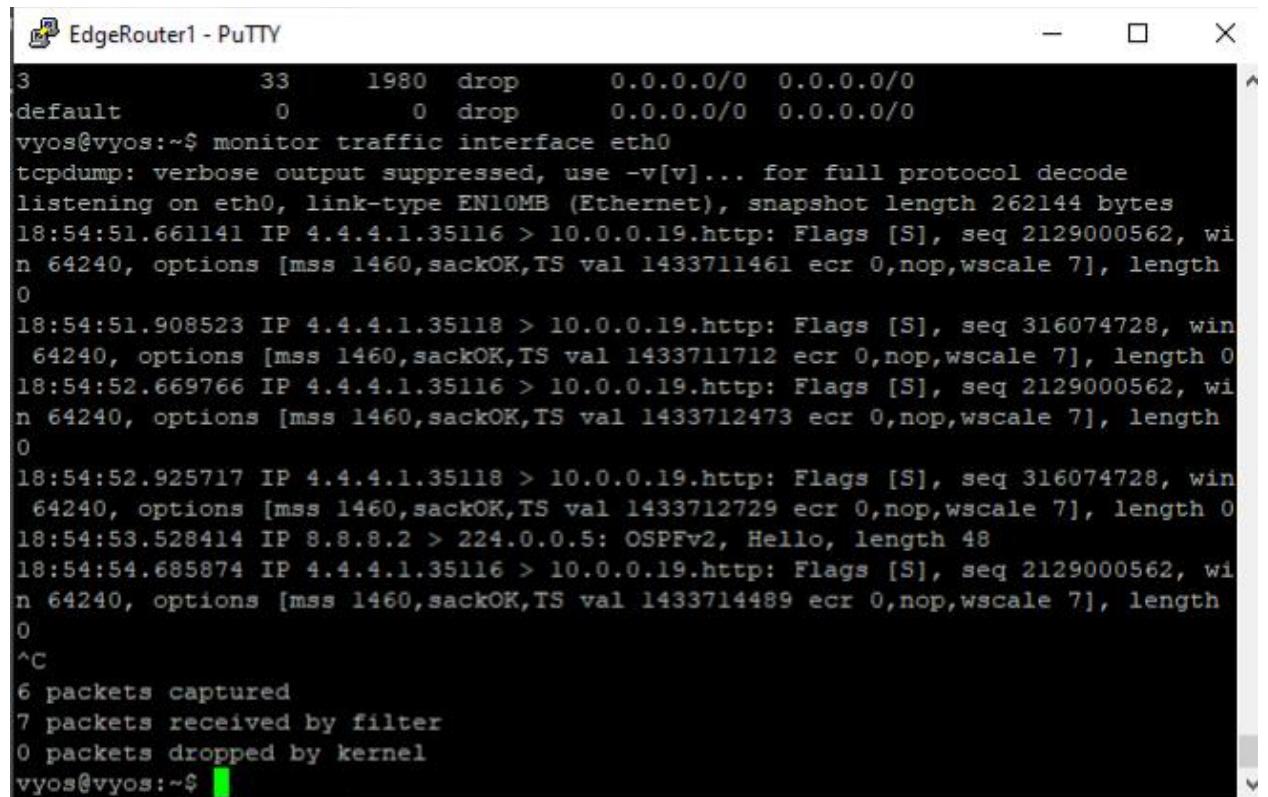
WESTERN GOVERNORS UNIVERSITY.

9. The connection from SaaSPlatform fails because of the ACL configured on EdgeRouter1.



10. To verify this, I run "monitor traffic interface eth0" on EdgeRouter1 which shows HTTP traffic from 4.4.4.1 to 10.0.0.19.





```

EdgeRouter1 - PuTTY
vyos@vyos:~$ monitor traffic interface eth0
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
18:54:51.661141 IP 4.4.4.1.35116 > 10.0.0.19.http: Flags [S], seq 2129000562, wi
n 64240, options [mss 1460,sackOK,TS val 1433711461 ecr 0,nop,wscale 7], length
0
18:54:51.908523 IP 4.4.4.1.35118 > 10.0.0.19.http: Flags [S], seq 316074728, win
64240, options [mss 1460,sackOK,TS val 1433711712 ecr 0,nop,wscale 7], length 0
18:54:52.669766 IP 4.4.4.1.35116 > 10.0.0.19.http: Flags [S], seq 2129000562, wi
n 64240, options [mss 1460,sackOK,TS val 1433712473 ecr 0,nop,wscale 7], length
0
18:54:52.925717 IP 4.4.4.1.35118 > 10.0.0.19.http: Flags [S], seq 316074728, win
64240, options [mss 1460,sackOK,TS val 1433712729 ecr 0,nop,wscale 7], length 0
18:54:53.528414 IP 8.8.8.2 > 224.0.0.5: OSPFv2, Hello, length 48
18:54:54.685874 IP 4.4.4.1.35116 > 10.0.0.19.http: Flags [S], seq 2129000562, wi
n 64240, options [mss 1460,sackOK,TS val 1433714489 ecr 0,nop,wscale 7], length
0
^C
6 packets captured
7 packets received by filter
0 packets dropped by kernel
vyos@vyos:~$ 

```

11. To verify that these packets hit the ACL, I run "show firewall statistics" on EdgeRouter1.

IPv4 Firewall "deny_inbound_rfc1918"					
Active on: (eth0,in)					
Rule	packets	bytes	Action	Source	Destination
1	32	5027	accept	0.0.0.0/0	0.0.0.0/0
2	754	63040	drop	0.0.0.0/0	0.0.0.0/0
3	47	2820	drop	0.0.0.0/0	0.0.0.0/0
default	0	0	drop	0.0.0.0/0	0.0.0.0/0
:					

12. Lastly, I run "show firewall name deny_inbound_rfc1918" on EdgeRouter1 to see the rule conditions. We can see that rule 3 checks if the connection state is new, and if so, it drops the packets for any protocol.



IPv4 Firewall "deny_inbound_rfc1918"						
Active on: (eth0,in)						
Rule	Action	Protocol	Packets	Bytes	Conditions	
1	accept	all	32	5027	ct state { established }	
2	drop	icmp	754	63040	meta l4proto icmp	
3	drop	all	61	3660	ct state { new }	
default	drop	all	0	0		

Test Case #2: Administering an Access Control List for Guest Access

Your network must utilize an Access Control List that allows guest access. Guest access should be limited to internet traffic only.

Functionality

Describe the **functionality** of the test case in relation to your network project. Identify the relevant tools (devices, subnets, etc.) used in this test case and their specific interactions.

There is a guest network configured on 10.0.0.32/29. There is an ACL configured on L3Switch2 ports 3 and 4 that only allow packets that have a destination IP address of the default gateway for the computers 10.0.0.33, 255.255.255.255 for DHCP, and the SaaSPlatform 4.4.4.1 in the Mock Internet.

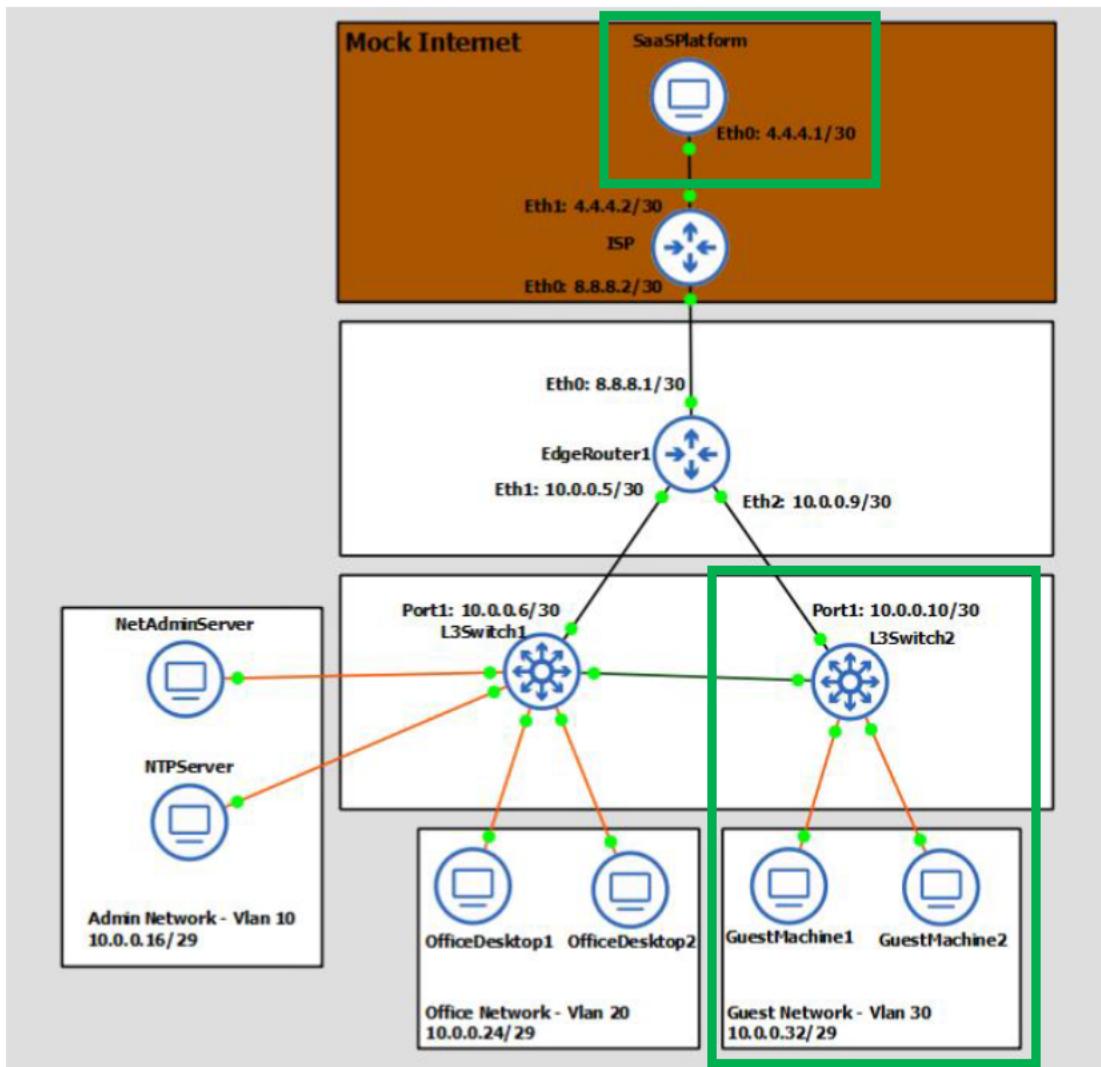
Network Diagram or Segment

Provide a **network diagram or segment** visualizing the topology and devices used in this test case.

Devices that are directly applicable to the testing method are enclosed in green boxes.



WESTERN GOVERNORS UNIVERSITY



Legend

SaaS	Software as a Service
Eth*	Ethernet + port number
ISP	Internet Service Provider
L3	Layer 3
NetAdminServer	Network Administration Server
Admin	Administration
VLAN	Virtual Local Area Network

Testing Method

Summarize the **testing method** used to verify functionality of the network project within the virtual lab environment, including any metrics of success.

To verify that the ACL works properly for the Guest Network, I will perform a traceroute to 10.0.0.33 and 4.4.4.1 from GuestMachine1, both of which will be successful. To verify that I am

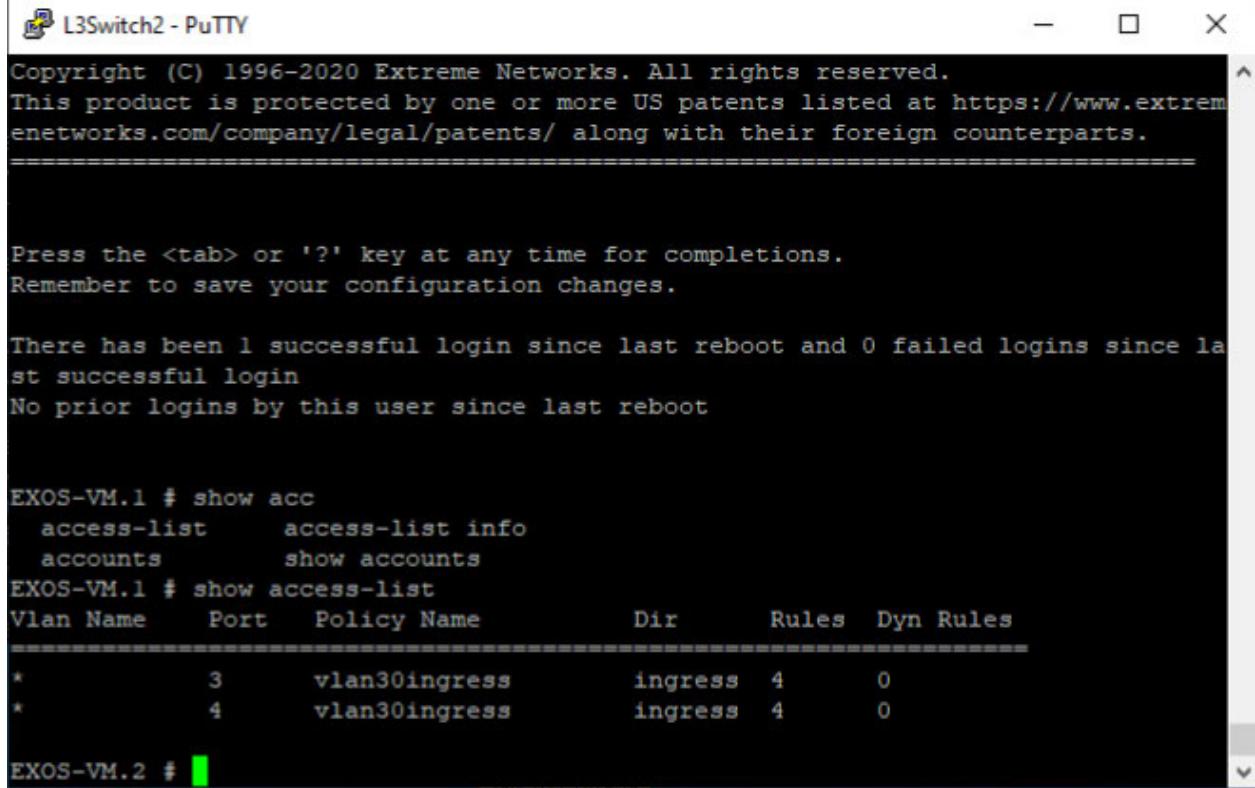


unable to reach any other devices, I will perform a traceroute destined to each hop returned by the traceroute that is not 10.0.0.33 and 4.4.4.1. This will fail because the ACL will deny packets not destined for those IP addresses.

Process List

Provide a comprehensive **process list** of the steps taken within the network project to run the testing method. Include screenshots to illustrate the process and ensure clarity for others attempting to replicate the test.

1. Verify the access lists are in place on L3Switch2 for ports 3 and 4 by logging into the switch and running "show access-list".

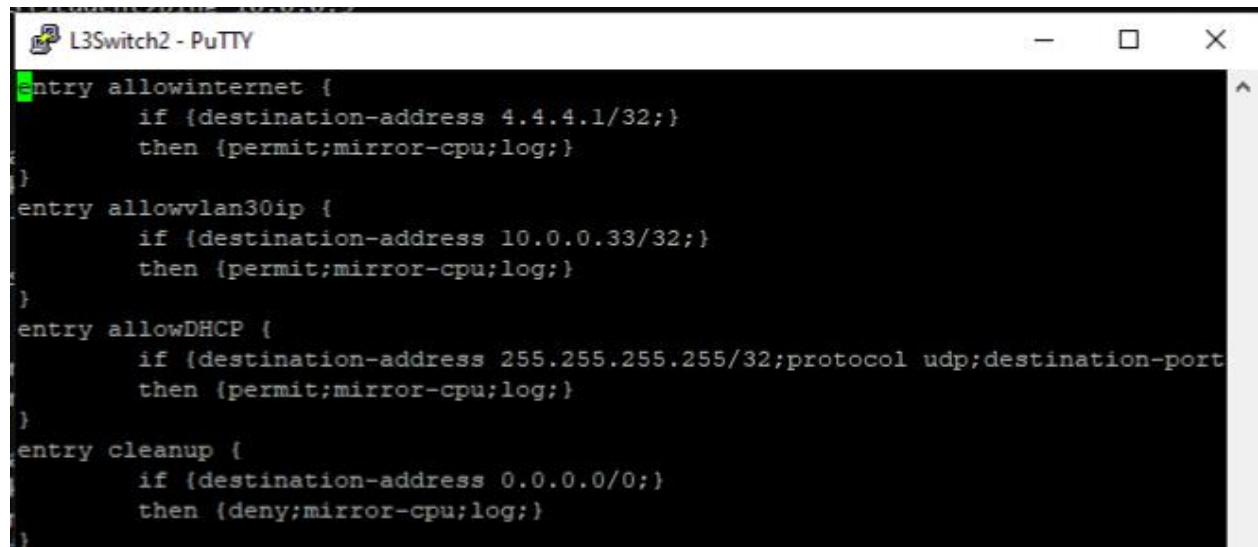


L3Switch2 - PuTTY

```
Copyright (C) 1996-2020 Extreme Networks. All rights reserved.  
This product is protected by one or more US patents listed at https://www.extremenetworks.com/company/legal/patents/ along with their foreign counterparts.  
=====  
  
Press the <tab> or '?' key at any time for completions.  
Remember to save your configuration changes.  
  
There has been 1 successful login since last reboot and 0 failed logins since last successful login  
No prior logins by this user since last reboot  
  
EXOS-VM.1 # show acc  
  access-list      access-list info  
  accounts        show accounts  
EXOS-VM.1 # show access-list  
Vlan Name      Port    Policy Name          Dir      Rules  Dyn Rules  
=====  
*             3       vlan30ingress      ingress  4      0  
*             4       vlan30ingress      ingress  4      0  
  
EXOS-VM.2 #
```

2. Verify the ACLs are blocking all traffic other than DHCP and destination addresses 10.0.0.33 (gateway) and 4.4.4.1 by opening the policy file by running "vi vlan30ingress.pol". We can see that this is correct.

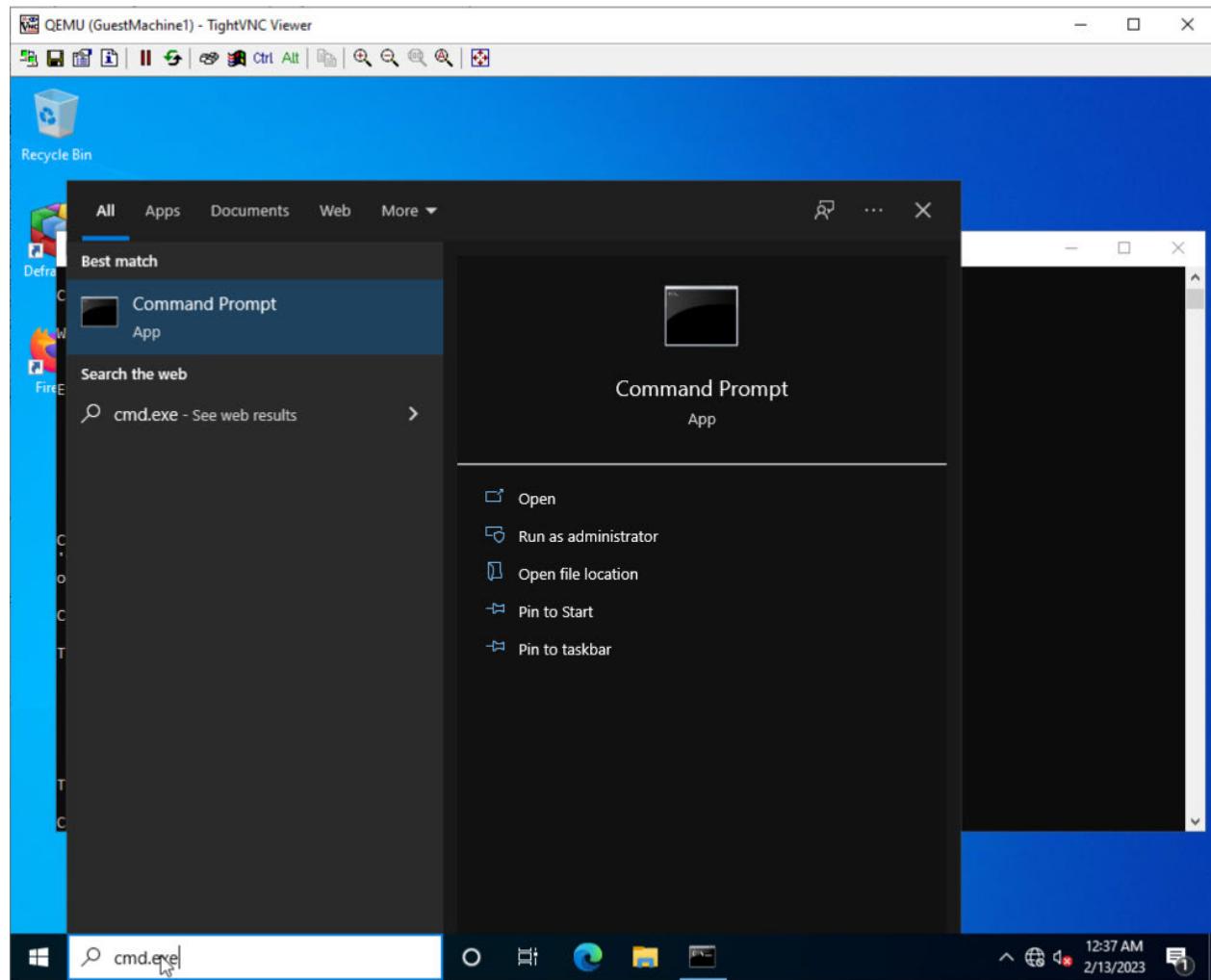




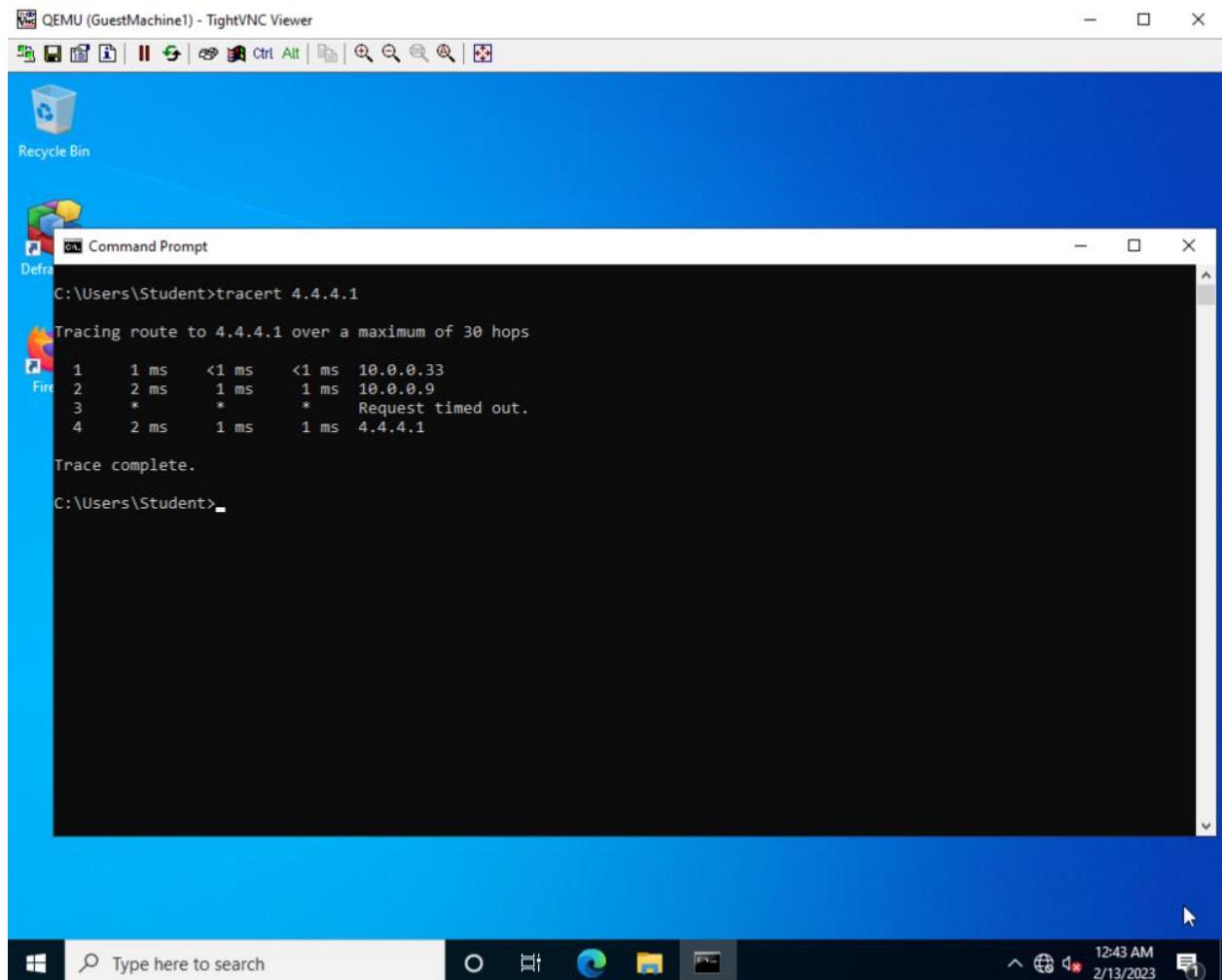
L3Switch2 - PuTTY

```
entry allowinternet {
    if {destination-address 4.4.4.1/32;}
    then {permit;mirror-cpu;log;}
}
entry allowvlan30ip {
    if {destination-address 10.0.0.33/32;}
    then {permit;mirror-cpu;log;}
}
entry allowDHCP {
    if {destination-address 255.255.255.255/32;protocol udp;destination-port 67}
    then {permit;mirror-cpu;log;}
}
entry cleanup {
    if {destination-address 0.0.0.0/0;}
    then {deny;mirror-cpu;log;}
}
```

3. Login to GuestMachine1
4. Run cmd.exe

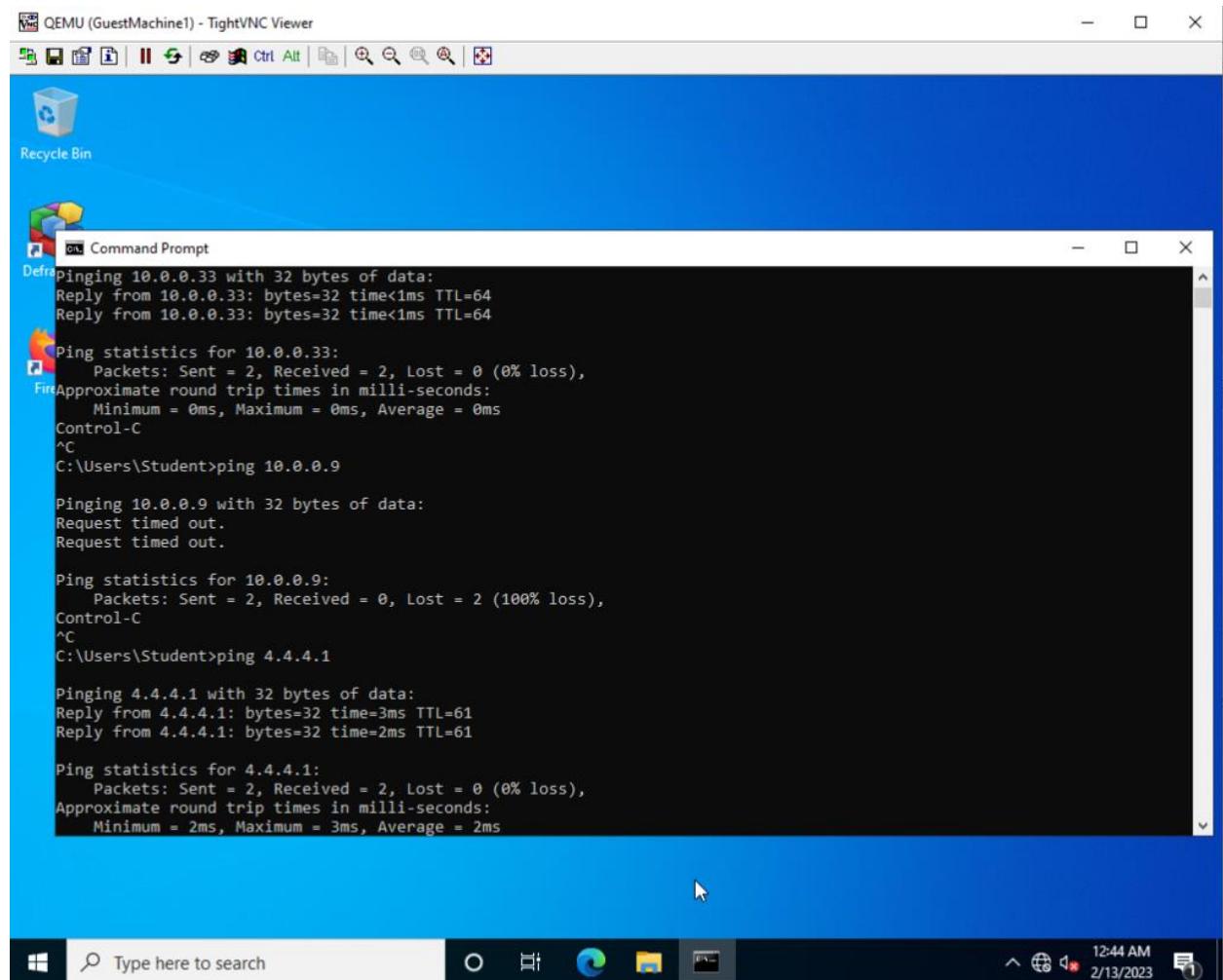


5. In the command prompt, run "tracert 4.4.4.1" to discover all layer 3 hops between GuestMachine1 and SaaSPlatform. This shows that 10.0.0.33, 10.0.0.9, a device which blocks inbound ICMP traffic, and 4.4.4.1 are in between.



6. Run "ping 10.0.0.33" and let it complete. This will be successful because it is explicitly allowed in the ACL.
7. Run "ping 10.0.0.9" and let it complete. This will fail and is caught by the deny all rule because it is not explicitly allowed.
8. Run "ping 4.4.4.1" and let it complete. This will be successful because it is explicitly allowed in the ACL.





WESTERN GOVERNORS UNIVERSITY®

Test Case #3: Security Compliance—Log-in Banners and Automation

Display a log-in banner when accessing each device on the network. The log-in banner should notify users of an acceptable use policy (AUP) or other security-based policies when attempting to Login to the network.

Additionally, establish an automated process to update the log-in banner for multiple devices. Clearly identify the devices that will be updated, and provide a step-by-step guide for initiating the automated updates.

Note: Remove the banners from the devices that will be updated by your automation process prior to submission so the evaluators can run the process and verify the new configuration.

Functionality

*Describe the **functionality** of the test case in relation to your network project. Identify the relevant tools (devices, subnets, etc.) used in this test case and their specific interactions.*

There are three network devices that would require banners on the network. EdgeRouter1, L3Switch1, and L3Switch2. The ISP router will not receive a banner because it is in the Mock Internet and would not be controlled by the customer. The login banner currently in the lab is based on the “Short-Form Banner” from Kellep Charles article on SecurityOrb.com “Warning banner sample for systems and network devices”.

The banners are configured through a Python script hosted on NetAdminServer and located at “/home/student/NetAutomation/update_banner.py”. This script utilizes the paramiko and netmiko libraries to handle SSH connections to the router and switches to update their banners. The banner that will be configured on the devices is stored as a constant in the Python script. The constant is named “BANNER” and is at the top of the script.

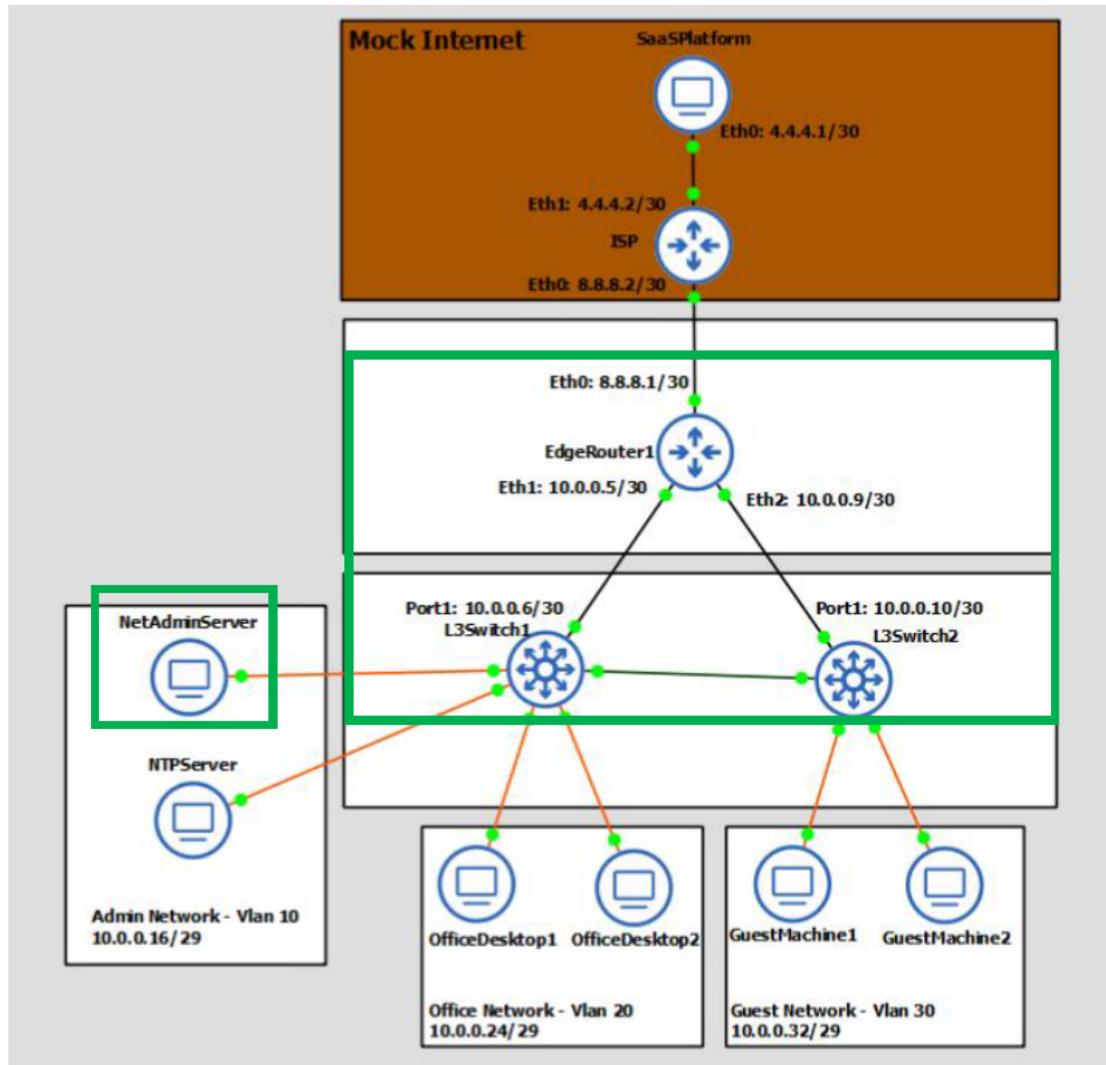
Network Diagram or Segment

*Provide a **network diagram or segment** visualizing the topology and devices used in this test case.*

Devices that are directly applicable to the testing method are enclosed in green boxes.



WESTERN GOVERNORS UNIVERSITY



Legend

SaaS	Software as a Service
Eth*	Ethernet + port number
ISP	Internet Service Provider
L3	Layer 3
NetAdminServer	Network Administration Server
Admin	Administration
VLAN	Virtual Local Area Network

Testing Method

Summarize the **testing method** used to verify functionality of the network project within the virtual lab environment, including any metrics of success.

I will remove the banner from each device and save the configuration. I will then verify that there is no banner configured. I will then run the Python script to configure a banner that meets the requirements

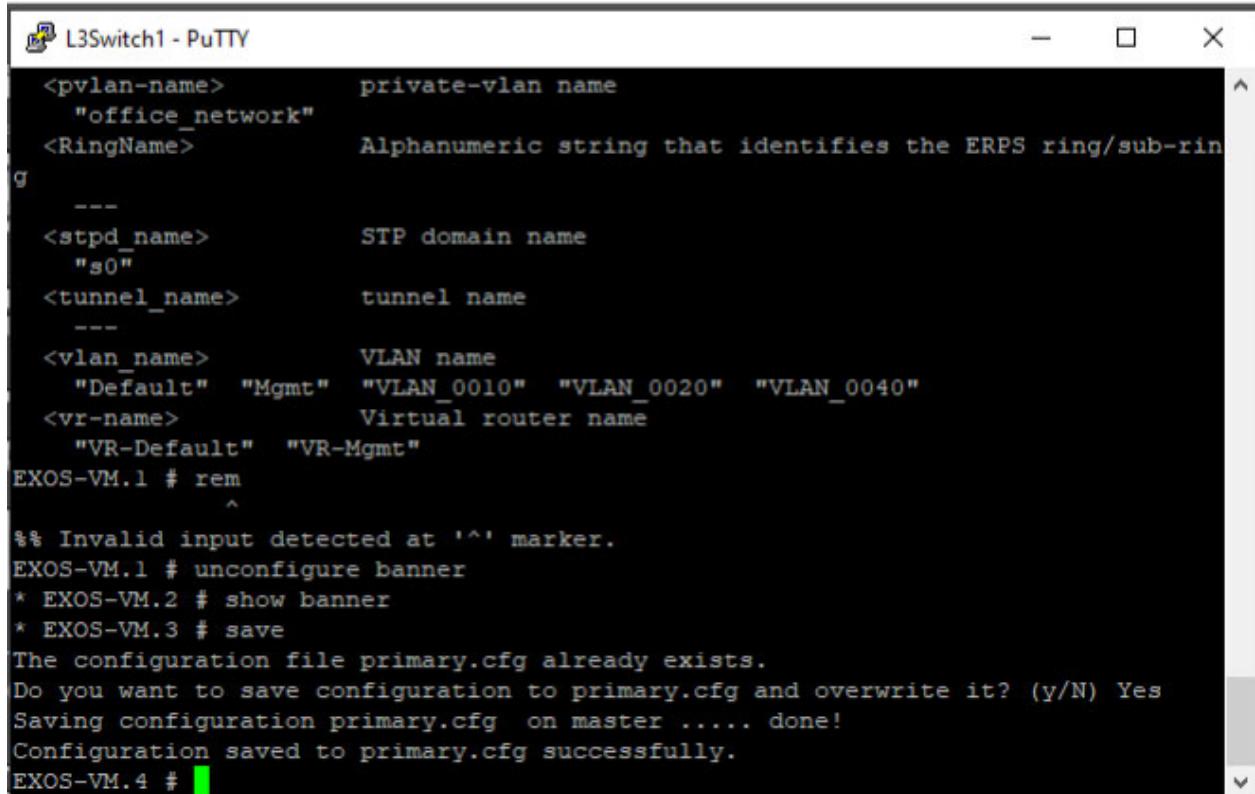


stated in the test case on each device. I will then reboot each device and show the new banner to demonstrate that each device has the proper banner that persists after reboot.

Process List

Provide a comprehensive **process list** of the steps taken within the network project to run the testing method. Include screenshots to illustrate the process and ensure clarity for others attempting to replicate the test.

1. Login to L3Switch1.
2. Run "unconfigure banner" to remove any configured banners.
3. Run "show banner" to verify that this was successful.
4. Run "save" to save the configuration through reboot.

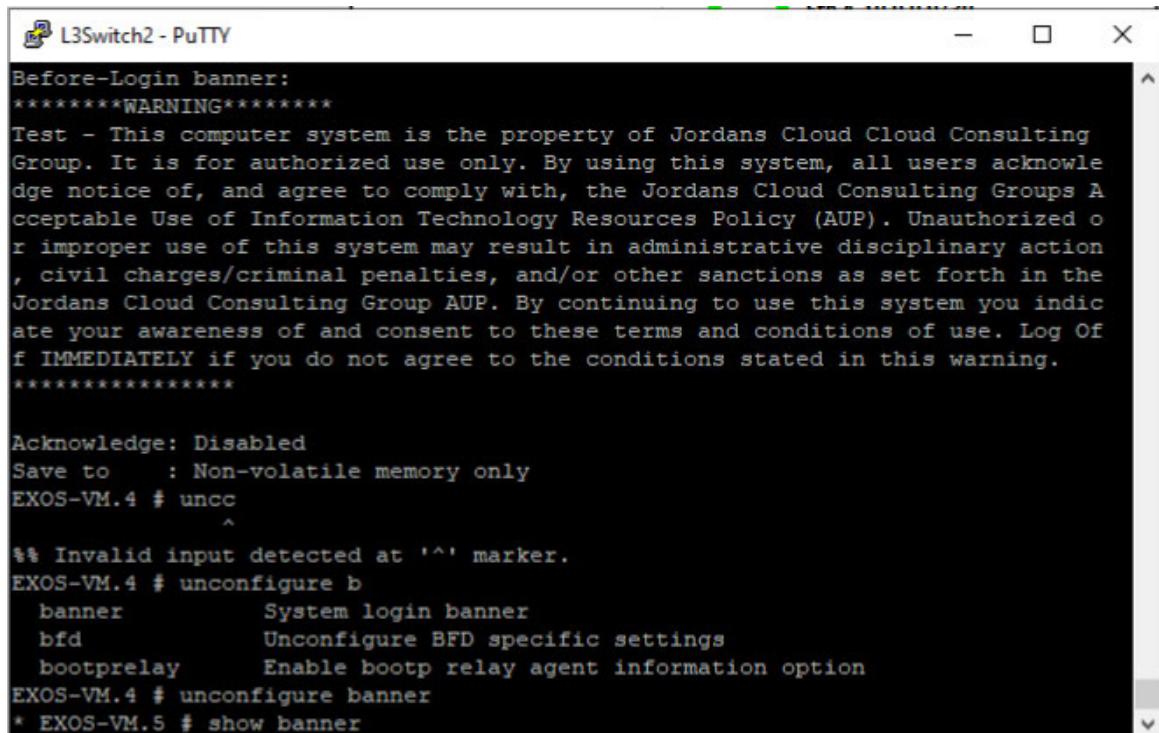


```
L3Switch1 - PuTTY

<pvlan-name>          private-vlan name
  "office_network"
<RingName>            Alphanumeric string that identifies the ERPS ring/sub-ring
g
  ---
<stpd_name>           STP domain name
  "s0"
<tunnel_name>         tunnel name
  ---
<vlan_name>            VLAN name
  "Default"  "Mgmt"  "VLAN_0010"  "VLAN_0020"  "VLAN_0040"
<vr-name>              Virtual router name
  "VR-Default"  "VR-Mgmt"
EXOS-VM.1 # rem
  ^
%% Invalid input detected at '^' marker.
EXOS-VM.1 # unconfigure banner
* EXOS-VM.2 # show banner
* EXOS-VM.3 # save
The configuration file primary.cfg already exists.
Do you want to save configuration to primary.cfg and overwrite it? (y/N) Yes
Saving configuration primary.cfg on master ..... done!
Configuration saved to primary.cfg successfully.
EXOS-VM.4 #
```

5. Repeat steps 1-3 for L3Switch2



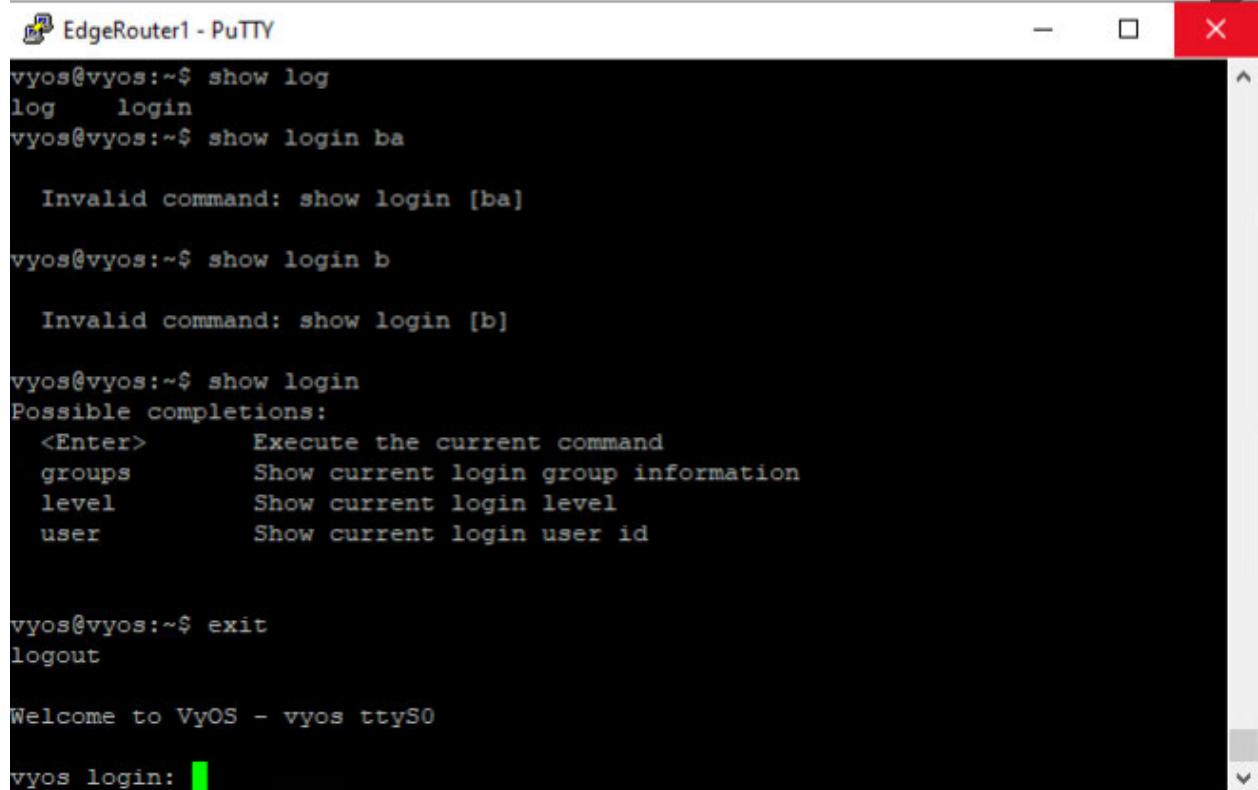


L3Switch2 - PuTTY

```
Before-Login banner:  
*****WARNING*****  
Test - This computer system is the property of Jordans Cloud Cloud Consulting Group. It is for authorized use only. By using this system, all users acknowledge notice of, and agree to comply with, the Jordans Cloud Consulting Groups Acceptable Use of Information Technology Resources Policy (AUP). Unauthorized or improper use of this system may result in administrative disciplinary action, civil charges/criminal penalties, and/or other sanctions as set forth in the Jordans Cloud Consulting Group AUP. By continuing to use this system you indicate your awareness of and consent to these terms and conditions of use. Log Off IMMEDIATELY if you do not agree to the conditions stated in this warning.  
*****  
Acknowledge: Disabled  
Save to : Non-volatile memory only  
EXOS-VM.4 # uncc  
^  
%% Invalid input detected at '^' marker.  
EXOS-VM.4 # unconfigure b  
    banner      System login banner  
    bfd         Unconfigure BFD specific settings  
    bootprelay  Enable bootp relay agent information option  
EXOS-VM.4 # unconfigure banner  
* EXOS-VM.5 # show banner
```

6. Login to EdgeRouter1.
7. Run "config" to enter configuration mode.
8. Run "delete system login banner" to delete any configured banners.
9. Run "commit" then "save" to save the configuration through reboot.
10. Run "exit" to leave configuration mode
11. Run "exit" again to log out. There should be no login banner.





```
EdgeRouter1 - PuTTY
vyos@vyos:~$ show log
log    login
vyos@vyos:~$ show login ba

        Invalid command: show login [ba]

vyos@vyos:~$ show login b

        Invalid command: show login [b]

vyos@vyos:~$ show login
Possible completions:
<Enter>      Execute the current command
groups        Show current login group information
level         Show current login level
user          Show current login user id

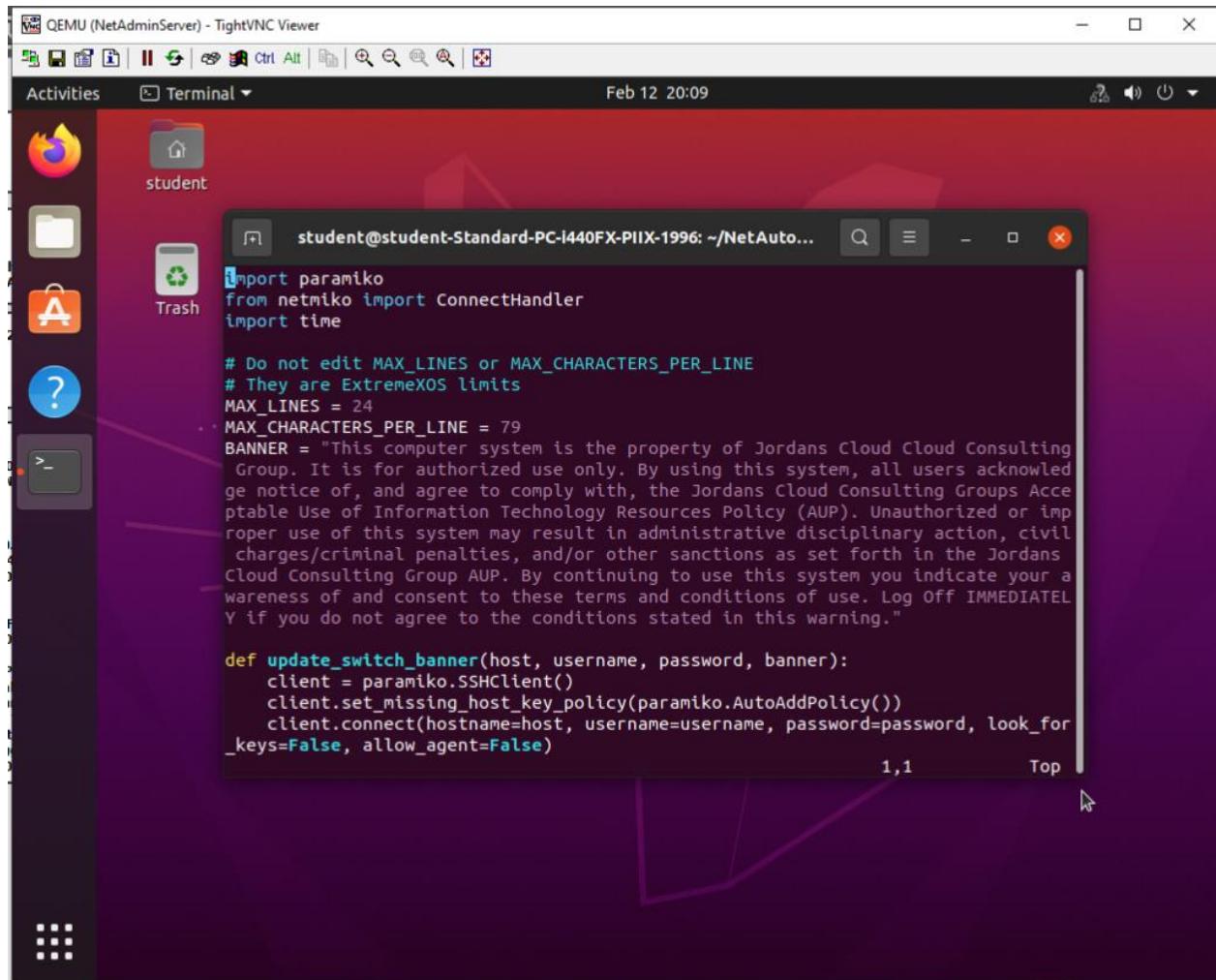
vyos@vyos:~$ exit
logout

Welcome to VyOS - vyos ttyS0

vyos login: [REDACTED]
```

12. Login to NetAdminServer.
13. Open a terminal and run "cd /home/student/NetAutomation".
14. Run "ls" to view available scripts. You should see "update_banner.py" which is the script we will use.
15. Use a text editor on "update_banner.py" to look at the banner that will be used on the network devices. The "BANNER" constant is on line 9 of the file.
16. Modify the banner to suit the organizations needs. Keep in mind that there are limitations on banner size, and characters. ExtremeXOS limits the allowed lines and characters, and VyOS has issues handling certain characters. The script handles ExtremeXOS's limitations natively, but not VyOS because there is not a list I found of all excluded characters.





The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "student@student-Standard-PC-i440FX-PIIX-1996: ~/NetAuto...". The terminal content displays Python code using the paramiko library to set an SSH banner. The banner text is a warning about the system's ownership and usage policies, including administrative action and legal consequences for unauthorized use.

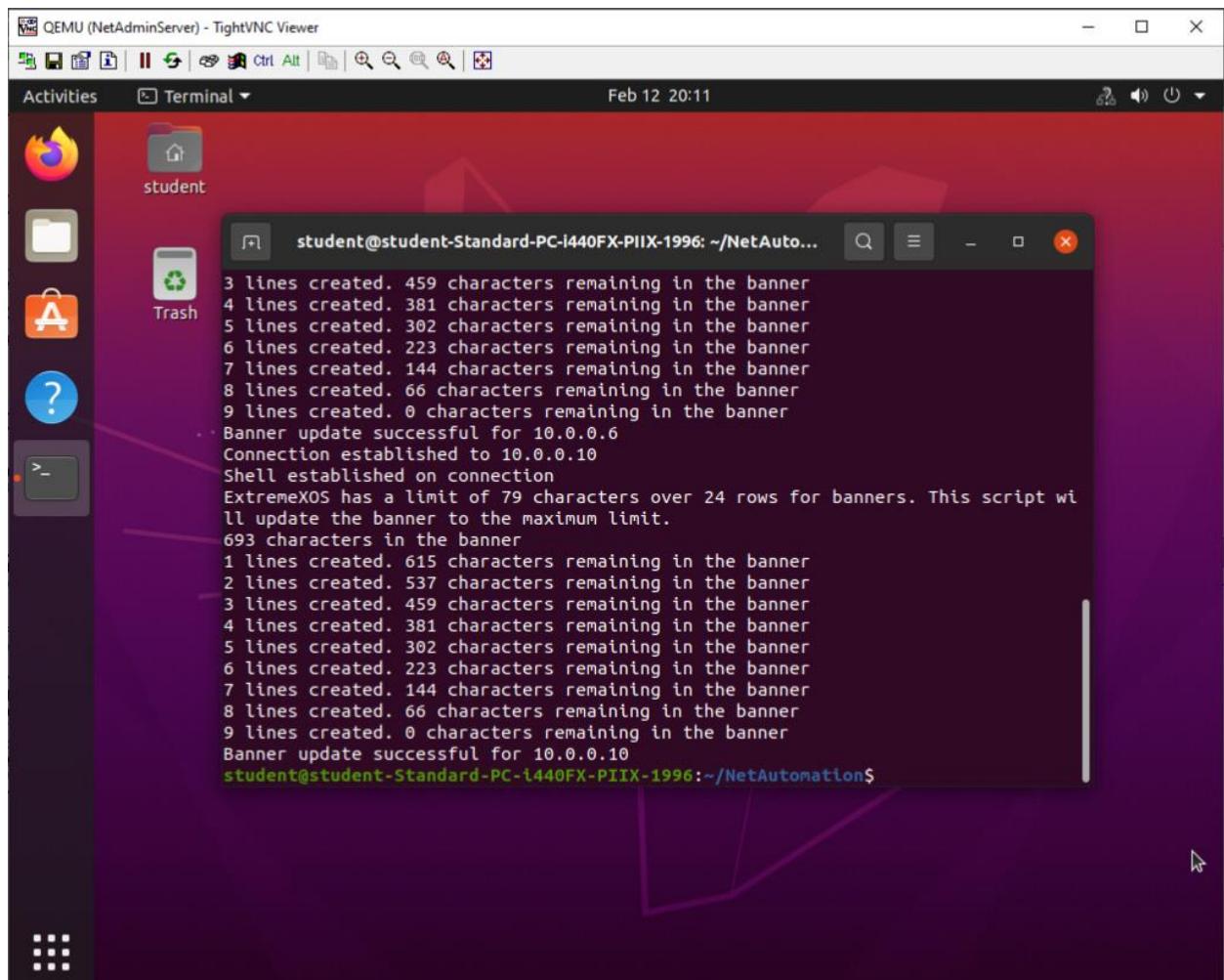
```
import paramiko
from netmiko import ConnectHandler
import time

# Do not edit MAX_LINES or MAX_CHARACTERS_PER_LINE
# They are ExtremeOS limits
MAX_LINES = 24
MAX_CHARACTERS_PER_LINE = 79
BANNER = "This computer system is the property of Jordans Cloud Consulting Group. It is for authorized use only. By using this system, all users acknowledge notice of, and agree to comply with, the Jordans Cloud Consulting Groups Acceptable Use of Information Technology Resources Policy (AUP). Unauthorized or improper use of this system may result in administrative disciplinary action, civil charges/criminal penalties, and/or other sanctions as set forth in the Jordans Cloud Consulting Group AUP. By continuing to use this system you indicate your awareness of and consent to these terms and conditions of use. Log Off IMMEDIATELY if you do not agree to the conditions stated in this warning."

def update_switch_banner(host, username, password, banner):
    client = paramiko.SSHClient()
    client.set_missing_host_key_policy(paramiko.AutoAddPolicy())
    client.connect(hostname=host, username=username, password=password, look_for_keys=False, allow_agent=False)
```

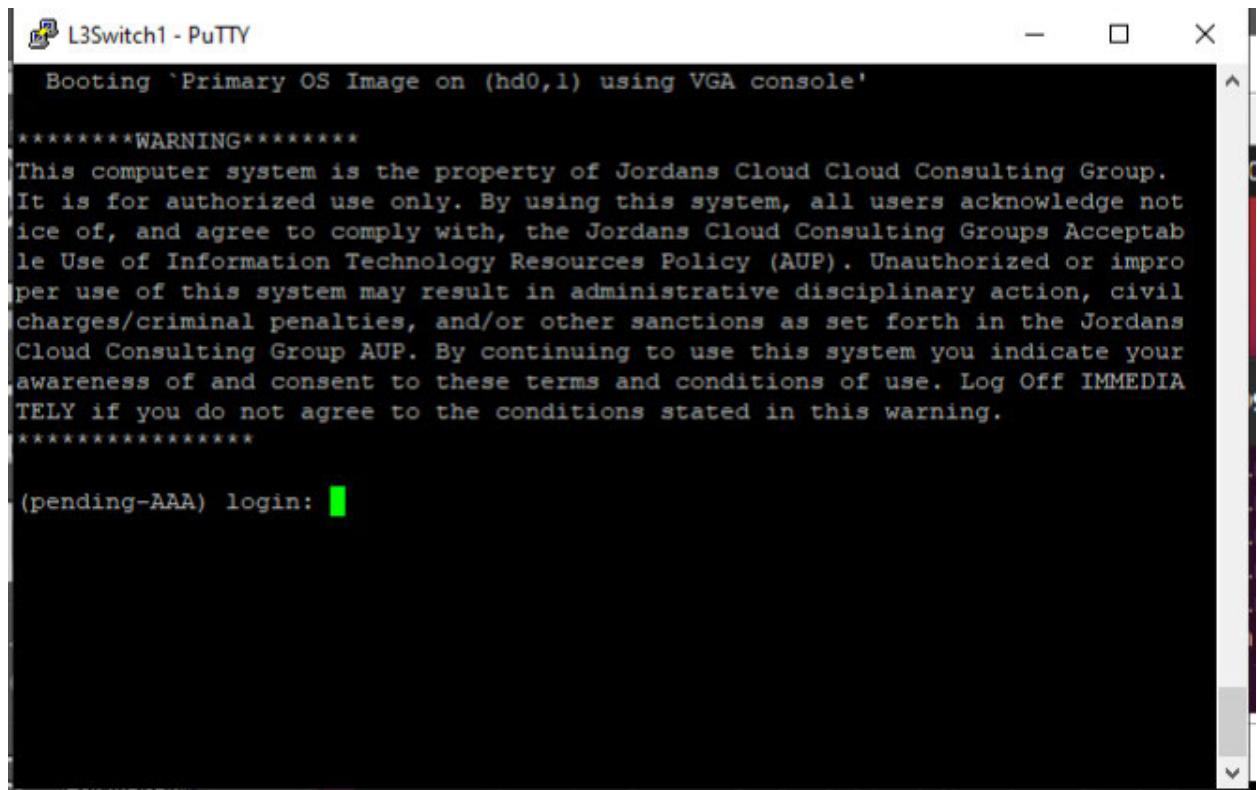
17. When you have created the banner to use, save and exit the file.
18. Run “python3 update_banner.py” to update the banners across the three network devices. It will show progress updates while it is running.





19. To verify that the script updated the devices successfully, reboot EdgeRouter1, L3Switch1 and L3Switch2 using the “reboot” command on each.
20. After rebooting each device, we can see that the banner configuration was successful. The banner will appear right after reboot, before login. No additional commands are needed.



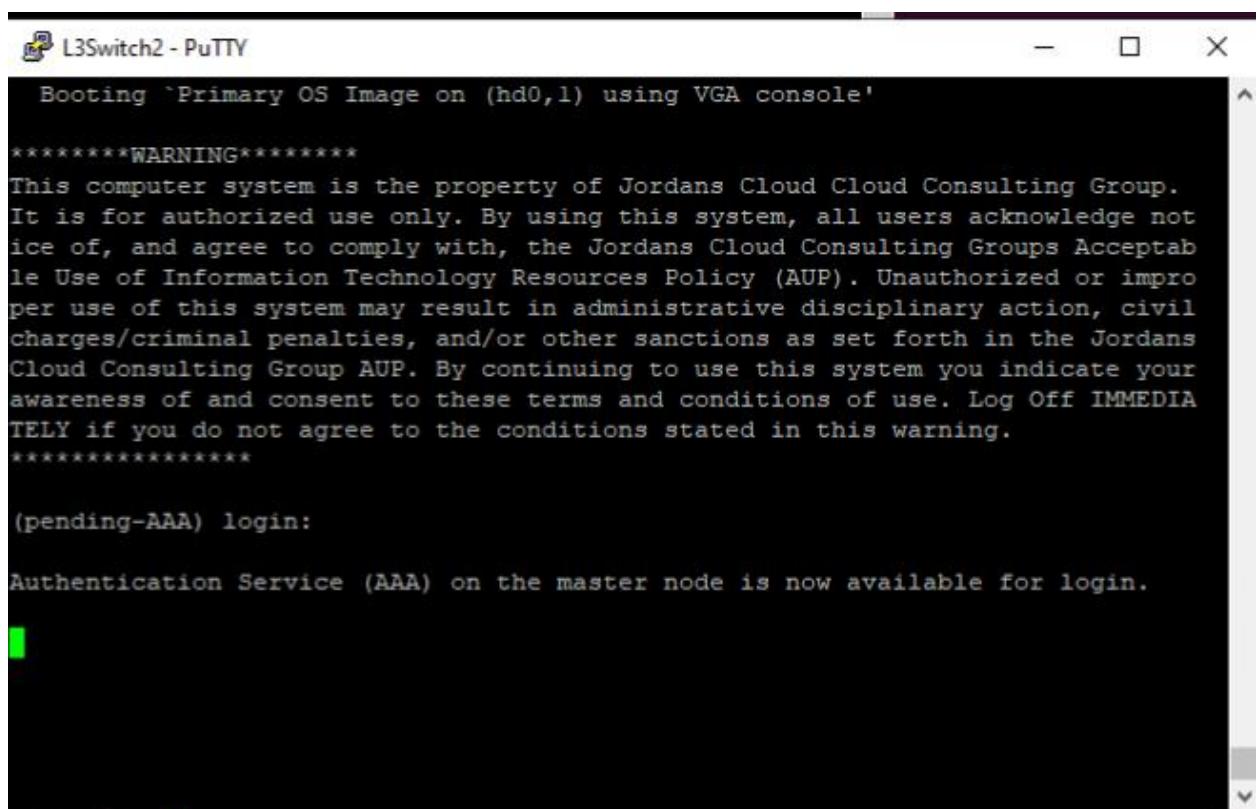


```
L3Switch1 - PuTTY
Booting `Primary OS Image on (hd0,1) using VGA console'

*****WARNING*****
This computer system is the property of Jordans Cloud Cloud Consulting Group.
It is for authorized use only. By using this system, all users acknowledge not
ice of, and agree to comply with, the Jordans Cloud Consulting Groups Acceptab
le Use of Information Technology Resources Policy (AUP). Unauthorized or impro
per use of this system may result in administrative disciplinary action, civil
charges/criminal penalties, and/or other sanctions as set forth in the Jordans
Cloud Consulting Group AUP. By continuing to use this system you indicate your
awareness of and consent to these terms and conditions of use. Log Off IMMEDIA
TELY if you do not agree to the conditions stated in this warning.
*****



(pending-AAA) login:
```



```
L3Switch2 - PuTTY
Booting `Primary OS Image on (hd0,1) using VGA console'

*****WARNING*****
This computer system is the property of Jordans Cloud Cloud Consulting Group.
It is for authorized use only. By using this system, all users acknowledge not
ice of, and agree to comply with, the Jordans Cloud Consulting Groups Acceptab
le Use of Information Technology Resources Policy (AUP). Unauthorized or impro
per use of this system may result in administrative disciplinary action, civil
charges/criminal penalties, and/or other sanctions as set forth in the Jordans
Cloud Consulting Group AUP. By continuing to use this system you indicate your
awareness of and consent to these terms and conditions of use. Log Off IMMEDIA
TELY if you do not agree to the conditions stated in this warning.
*****

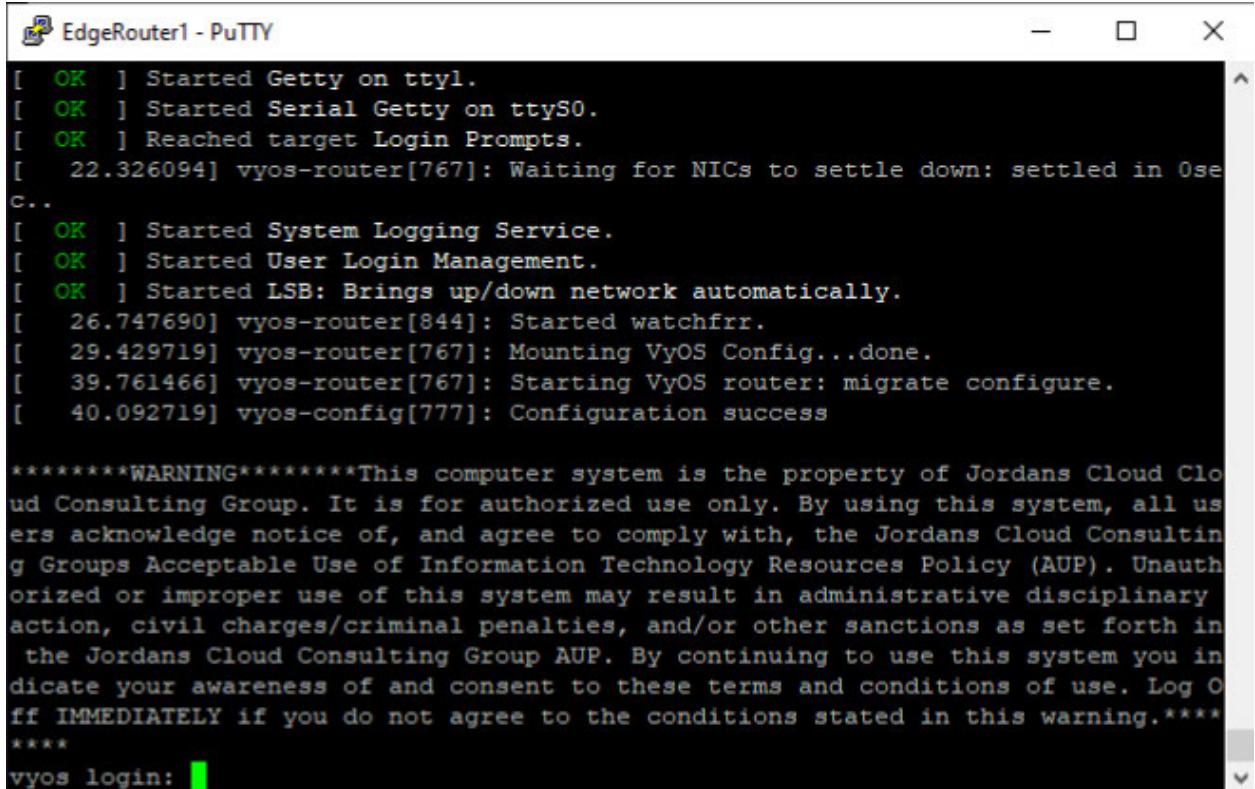


(pending-AAA) login:

Authentication Service (AAA) on the master node is now available for login.
```



WESTERN GOVERNORS UNIVERSITY



```
[ OK ] Started Getty on ttyl.
[ OK ] Started Serial Getty on ttyS0.
[ OK ] Reached target Login Prompts.
[ 22.326094] vyos-router[767]: Waiting for NICs to settle down: settled in 0sec..
[ OK ] Started System Logging Service.
[ OK ] Started User Login Management.
[ OK ] Started LSB: Brings up/down network automatically.
[ 26.747690] vyos-router[844]: Started watchfrr.
[ 29.429719] vyos-router[767]: Mounting VyOS Config...done.
[ 39.761466] vyos-router[767]: Starting VyOS router: migrate configure.
[ 40.092719] vyos-config[777]: Configuration success

*****WARNING*****This computer system is the property of Jordans Cloud Consulting Group. It is for authorized use only. By using this system, all users acknowledge notice of, and agree to comply with, the Jordans Cloud Consulting Groups Acceptable Use of Information Technology Resources Policy (AUP). Unauthorized or improper use of this system may result in administrative disciplinary action, civil charges/criminal penalties, and/or other sanctions as set forth in the Jordans Cloud Consulting Group AUP. By continuing to use this system you indicate your awareness of and consent to these terms and conditions of use. Log Off IMMEDIATELY if you do not agree to the conditions stated in this warning.*****
vyos login:
```

*Banners have been removed after configuration in the lab per instructions

Test Case #4: Accessing External Resources—Routing and Traffic Security

User devices on your network should have dynamic addresses that are assigned through DHCP unless they provide a service that requires a static address. You must also have at least one network resource that requires a static address.

Functionality

*Describe the **functionality** of the test case in relation to your network project. Identify the relevant tools (devices, subnets, etc.) used in this test case and their specific interactions.*

There are several interfaces that utilize static IP addresses and several networks that utilize DHCP. The static IP addresses configured are:

NTPServer eth0 10.0.0.18/29

L3Switch1 eth1 10.0.0.6/30

L3Switch2 eth1 10.0.0.42/30

L3Switch2 eth1 10.0.0.10/30

L3Switch2 eth2 10.0.0.41/30

EdgeRouter1 eth0 8.8.8.1/30 - eth1 10.0.0.5/30 - eth2 10.0.0.9/30

ISP eth0 8.8.8.2/30 - eth1 4.4.4.2/30

SaaSPlatform eth0 4.4.4.1/30



WESTERN GOVERNORS UNIVERSITY

DHCP is configured on L3Switch1 and L3Switch2. DHCP will assign an IP address and default gateway to each device that requests a lease. The default gateway for each lease is the VLAN IP address for each VLAN. Example: 10.0.0.25/29 VLAN 20 – Default Gateway address is VLAN Address 10.0.0.25.

The switches assign IP addresses in the following ranges on the following ports:

L3Switch1 VLAN_0010 Ports 3 6 - 10.0.0.19 - 10.0.0.22

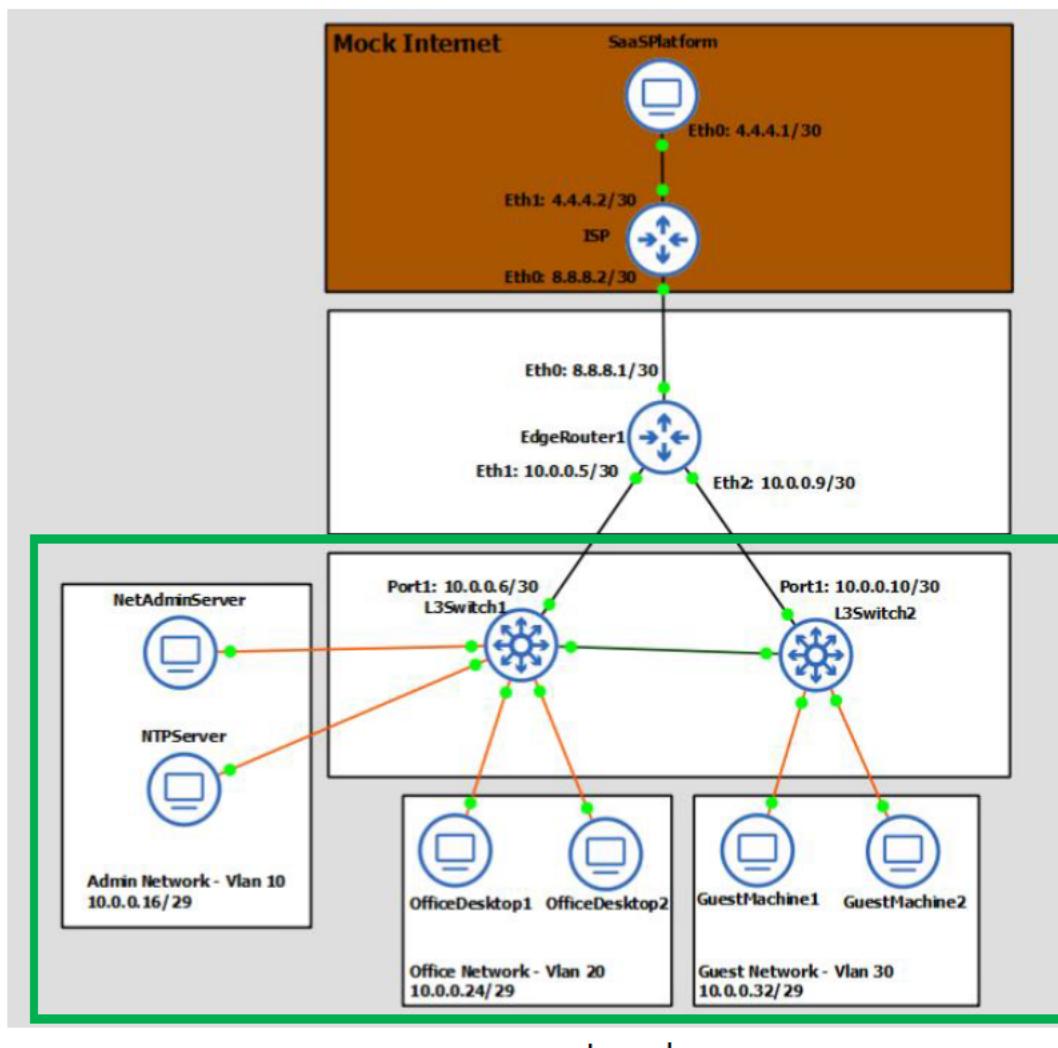
L3Switch1 VLAN_0020 Ports 4 5 – 10.0.0.26 - 10.0.0.30

L3Switch2 VLAN_0030 Ports 3 5 – 10.0.0.34 - 10.0.0.38

Network Diagram or Segment

*Provide a **network diagram or segment** visualizing the topology and devices used in this test case.*

Devices that are directly applicable to the testing method are enclosed in green boxes.



SaaS	Software as a Service
Eth*	Ethernet + port number
ISP	Internet Service Provider
L3	Layer 3
NetAdminServer	Network Administration Server
Admin	Administration
VLAN	Virtual Local Area Network

Testing Method

Summarize the **testing method** used to verify functionality of the network project within the virtual lab environment, including any metrics of success.

To demonstrate that DHCP is working as expected, I will show the DHCP service details on each switch. I will then log onto a computer from each network receiving a DHCP lease and verify that one has been received, and it falls within the configured range. I will also discuss NTPServer's static address and how it affects the DHCP address pool for VLAN 10.

Process List

Provide a comprehensive **process list** of the steps taken within the network project to run the testing method. Include screenshots to illustrate the process and ensure clarity for others attempting to replicate the test.

1. Login to L3Switch1.
2. Run "show dhcp-server" to view the server config information.

```

L3Switch1 - PuTTY

* EXOS-VM.3 # show bann
* EXOS-VM.4 # save
The configuration file primary.cfg already exists.
Do you want to save configuration to primary.cfg and overwrite it? (y/N) Yes
Saving configuration primary.cfg on master ... done!
Configuration saved to primary.cfg successfully.
EXOS-VM.5 # show dhcp
% Ambiguous command: "show dhcp"
EXOS-VM.6 # show dhcp-server
VLAN "VLAN_0010":
    DHCP Address Range      : 10.0.0.19->10.0.0.22
    Netlogin Lease Timer   : Not configured (Default = 10 seconds)
    DHCP Lease Timer       : Not configured (Default = 7200 seconds)
    Default Gateway        : 10.0.0.17
    Ports DHCP Enabled    : 3

VLAN "VLAN_0020":
    DHCP Address Range      : 10.0.0.26->10.0.0.30
    Netlogin Lease Timer   : Not configured (Default = 10 seconds)
    DHCP Lease Timer       : Not configured (Default = 7200 seconds)
    Default Gateway        : 10.0.0.25
    Ports DHCP Enabled    : 4-5

EXOS-VM.7 #

```



3. Login to NetAdminServer on VLAN 10. It is the only device receiving DHCP leases on the network. The reason for this is because NTPServer is configured statically as 10.0.0.18. This is why the DHCP address pool for VLAN 10 starts at 10.0.0.19. We can view the static address for NTPServer by opening "cmd.exe" and running "ipaddress /all". We can see that "DHCP Enabled" is "No". NTPServer uses a static address so network devices know where to reach it to receive NTP information.

```
C:\Users\Administrator>ipconfig /all
Windows IP Configuration

Host Name . . . . . : WIN-PU6TC6U10GF
Primary Dns Suffix . . . . . :
Node Type . . . . . : Hybrid
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No

Ethernet adapter Ethernet:

Connection-specific DNS Suffix . . . . . :
Description . . . . . : Intel(R) PRO/1000 MT Network Connection
Physical Address. . . . . : 0C-10-9E-30-00-00
DHCP Enabled. . . . . : No
Autoconfiguration Enabled . . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::2dea:c050:3eaa:f100%11(Preferred)
IPv4 Address. . . . . : 10.0.0.18(PREFERRED)
Subnet Mask . . . . . : 255.255.255.248
Default Gateway . . . . . : 10.0.0.17
DHCPv6 IAID . . . . . : 84676766
DHCPv6 Client DUID. . . . . : 00-01-00-01-2B-7A-6D-50-0C-10-9E-30-00-00
DNS Servers . . . . . : fec0:0:0:ffff::1%1
                           fec0:0:0:ffff::2%1
                           fec0:0:0:ffff::3%1
NetBIOS over Tcpip. . . . . : Enabled

C:\Users\Administrator>
```

4. I run a ping to 10.0.0.19 and 4.4.4.1 to verify that even though a static address is configured, internet access is still possible.



```
C:\ Administrator: Command Prompt

Pinging 10.0.0.19 with 32 bytes of data:
Reply from 10.0.0.19: bytes=32 time=1ms TTL=64
Reply from 10.0.0.19: bytes=32 time=1ms TTL=64
Reply from 10.0.0.19: bytes=32 time=1ms TTL=64

Ping statistics for 10.0.0.19:
    Packets: Sent = 3, Received = 3, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 1ms, Average = 1ms
Control-C
^C
C:\Users\Administrator>ping 4.4.4.1

Pinging 4.4.4.1 with 32 bytes of data:
Reply from 4.4.4.1: bytes=32 time=3ms TTL=61
Reply from 4.4.4.1: bytes=32 time=2ms TTL=61

Ping statistics for 4.4.4.1:
    Packets: Sent = 2, Received = 2, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 3ms, Average = 2ms
Control-C
^C
C:\Users\Administrator>ping 10.0.0.26

Pinging 10.0.0.26 with 32 bytes of data:
Control-C
^C
C:\Users\Administrator>
```

5. Open a terminal and run "ip a". An IP address should have already been received by the DHCP server.
6. Release the DHCP lease by running "sudo dhclient -r". You should no longer have an IP address after running "ip a".
7. Obtain a DHCP lease by running "sudo dhclient". You should now have an address after seeing the output of "ip a".



WESTERN GOVERNORS UNIVERSITY[®]

```

valid_lft forever preferred_lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 0c:52:6b:49:00:00 brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    inet 10.0.0.19/29 brd 10.0.0.23 scope global dynamic ens3
        valid_lft 7199sec preferred_lft 7199sec
        inet6 fe80::fad4:8b7d:e321:98e/64 scope link noprefixroute
            valid_lft forever preferred_lft forever
student@student-Standard-PC-I440FX-PIIX-1996:~/NetAutomation$ sudo dhclient -r
Killed old client process
student@student-Standard-PC-I440FX-PIIX-1996:~/NetAutomation$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 0c:52:6b:49:00:00 brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    inet6 fe80::fad4:8b7d:e321:98e/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
student@student-Standard-PC-I440FX-PIIX-1996:~/NetAutomation$ sudo dhclient
student@student-Standard-PC-I440FX-PIIX-1996:~/NetAutomation$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 0c:52:6b:49:00:00 brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    inet 10.0.0.19/29 brd 10.0.0.23 scope global dynamic ens3
        valid_lft 7199sec preferred_lft 7199sec
        inet6 fe80::fad4:8b7d:e321:98e/64 scope link noprefixroute
            valid_lft forever preferred_lft forever
student@student-Standard-PC-I440FX-PIIX-1996:~/NetAutomation$ 
```

8. Verify the that L3Switch1 gave this lease by running "show dhcp-server" on it. We can see that the lease it (10.0.0.19) gave corresponds with the lease received on NetAdminServer and the interface's MAC address.

```

EXOS-VM.7 # show dhcp-
    dhcp-client      Show information on DHCP client
    dhcp-server      Show information related to the dhcp-server
EXOS-VM.7 # show dhcp-server
VLAN "VLAN_0010":
    DHCP Address Range   : 10.0.0.19->10.0.0.22
    Netlogin Lease Timer : Not configured (Default = 10 seconds)
    DHCP Lease Timer     : Not configured (Default = 7200 seconds)
    Default Gateway       : 10.0.0.17
    Ports DHCP Enabled   : 3

=====
IP                  MAC                      State        Lease Time Left
=====
10.0.0.19          0c:52:6b:49:00:00      Assigned    0001:56:36 
```



- To verify DHCP is working correctly for VLAN 20, Login to a host on the VLAN. For this example, I will use OfficeDesktop1.
 - Run "cmd.exe" and in the command prompt run "ipconfig /release" and "ipconfig /renew" to release the old DHCP lease and get a new one.

The screenshot shows a Windows 10 desktop environment. At the top, there's a taskbar with various icons. The main window is a Command Prompt titled "Command Prompt" with the path "C:\Users\Student>". The user has run the command "ipconfig /release" followed by "ipconfig /renew", which outputs the current network configuration for the "Ethernet adapter Ethernet". The configuration includes a Link-local IPv6 Address of fe80::e8be:b6ab:793:9ba4%5, an IPv4 Address of 10.0.0.26, and a Default Gateway of 10.0.0.25. The desktop background is blue, and the taskbar includes icons for File Explorer, Task View, Start, and other system functions.

```
Microsoft Windows [Version 10.0.19044.1645]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Student>ipconfig /release

Windows IP Configuration

Ethernet adapter Ethernet:

Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . : fe80::e8be:b6ab:793:9ba4%5
Default Gateway . . . . . :

C:\Users\Student>ipconfig /renew

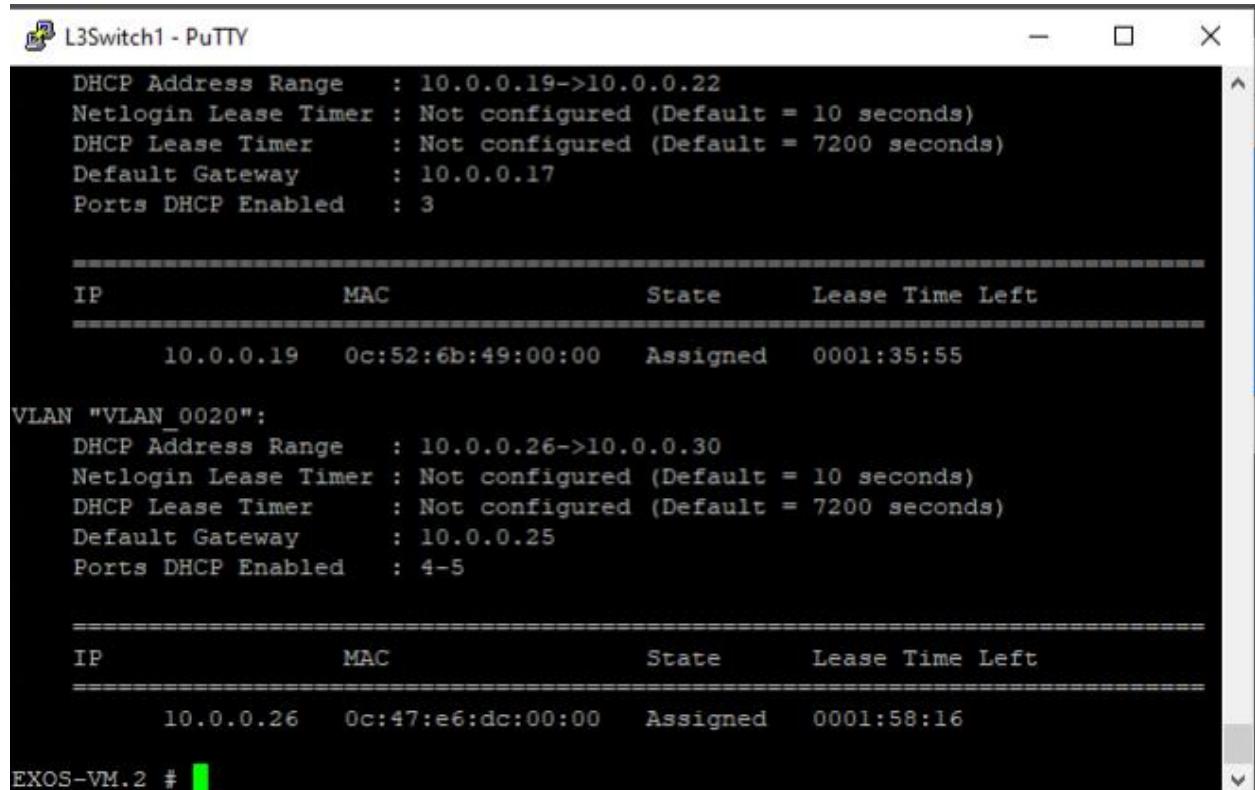
Windows IP Configuration

Ethernet adapter Ethernet:

Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . : fe80::e8be:b6ab:793:9ba4%5
IPv4 Address . . . . . : 10.0.0.26
Subnet Mask . . . . . : 255.255.255.248
Default Gateway . . . . . : 10.0.0.25

C:\Users\Student>
```

11. To verify this lease was given by L3Switch1, run "show dhcp-server" on the switch. This shows that the lease given to the host (10.0.0.26) was given by the server.



```

L3Switch1 - PuTTY

DHCP Address Range : 10.0.0.19->10.0.0.22
Netlogin Lease Timer : Not configured (Default = 10 seconds)
DHCP Lease Timer : Not configured (Default = 7200 seconds)
Default Gateway : 10.0.0.17
Ports DHCP Enabled : 3

=====

IP           MAC             State      Lease Time Left
=====
10.0.0.19    0c:52:6b:49:00:00  Assigned   0001:35:55

VLAN "VLAN_0020":
DHCP Address Range : 10.0.0.26->10.0.0.30
Netlogin Lease Timer : Not configured (Default = 10 seconds)
DHCP Lease Timer : Not configured (Default = 7200 seconds)
Default Gateway : 10.0.0.25
Ports DHCP Enabled : 4-5

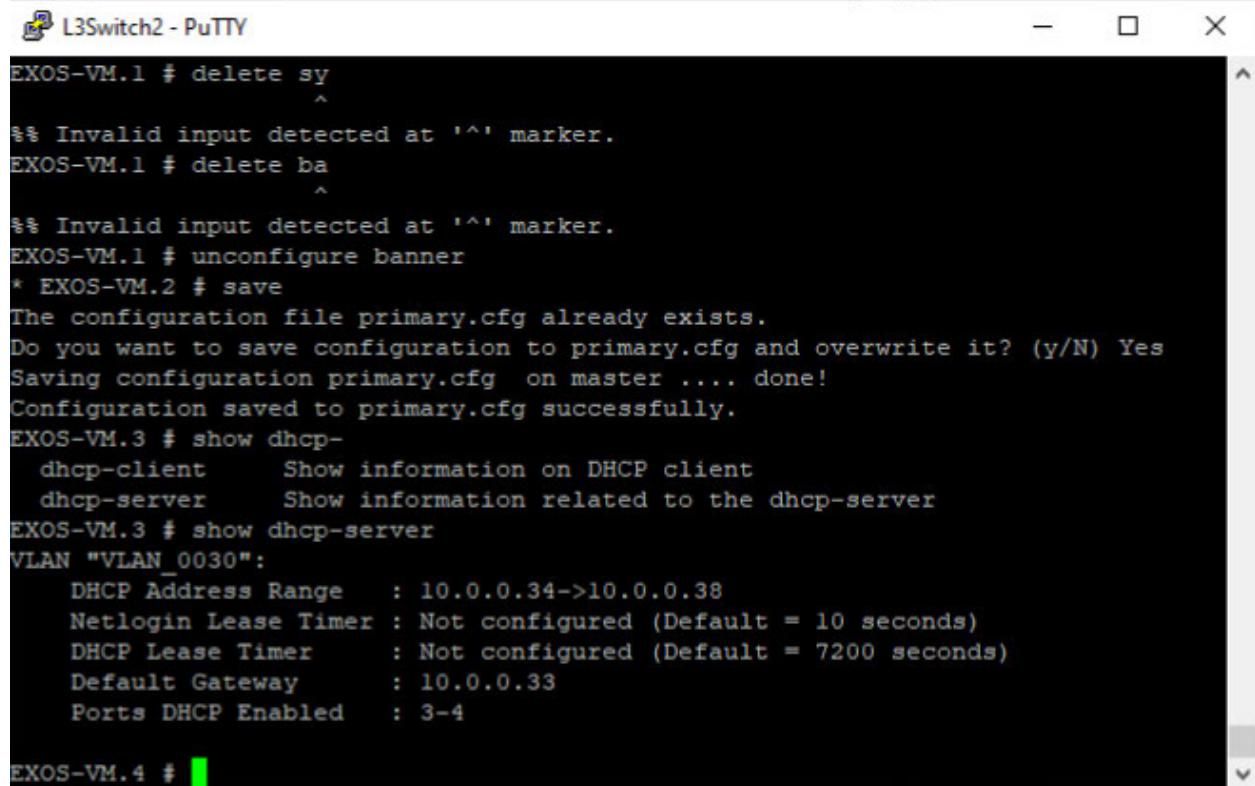
=====

IP           MAC             State      Lease Time Left
=====
10.0.0.26    0c:47:e6:dc:00:00  Assigned   0001:58:16

EXOS-VM.2 #

```

12. To verify L3Switch2's DHCP configuration, Login to L3Switch2 and run "show dhcp-server"



```

L3Switch2 - PuTTY

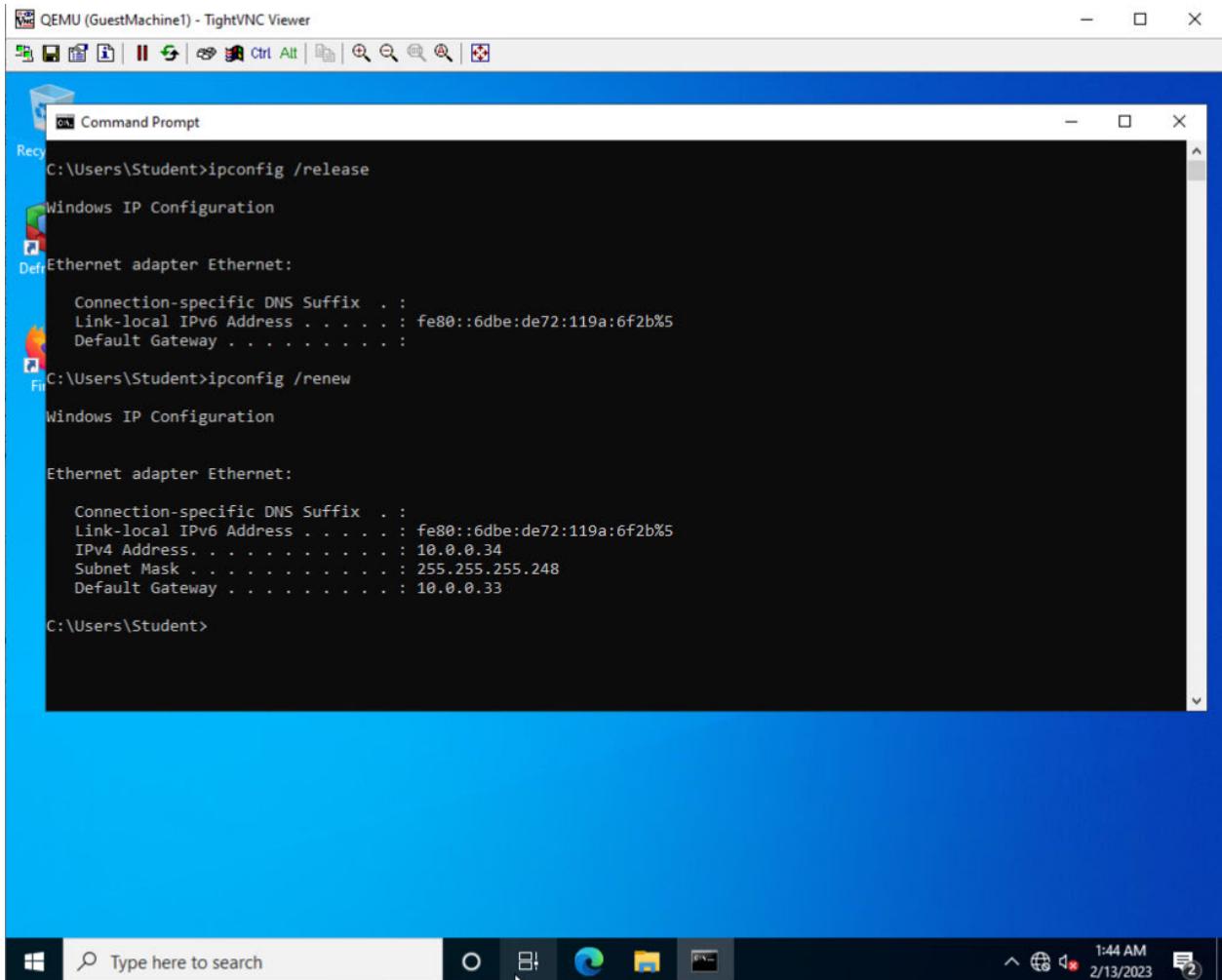
EXOS-VM.1 # delete sy
^
%% Invalid input detected at '^' marker.
EXOS-VM.1 # delete ba
^
%% Invalid input detected at '^' marker.
EXOS-VM.1 # unconfigure banner
* EXOS-VM.2 # save
The configuration file primary.cfg already exists.
Do you want to save configuration to primary.cfg and overwrite it? (y/N) Yes
Saving configuration primary.cfg on master .... done!
Configuration saved to primary.cfg successfully.
EXOS-VM.3 # show dhcp-
    dhcp-client      Show information on DHCP client
    dhcp-server      Show information related to the dhcp-server
EXOS-VM.3 # show dhcp-server
VLAN "VLAN_0030":
    DHCP Address Range : 10.0.0.34->10.0.0.38
    Netlogin Lease Timer : Not configured (Default = 10 seconds)
    DHCP Lease Timer : Not configured (Default = 7200 seconds)
    Default Gateway : 10.0.0.33
    Ports DHCP Enabled : 3-4

EXOS-VM.4 #

```



13. Login to a host on VLAN 30. For this example, I will use GuestMachine1.
14. Run "cmd.exe".
15. In the command prompt, run "ipconfig /release" and "ipconfig /renew" to release the old DHCP and receive a new one from the switch.



```
C:\Users\Student>ipconfig /release

Windows IP Configuration

Defining IP address for interface: Ethernet
Link-local IPv6 Address . . . . . : fe80::6dbe:de72:119a:6f2b%5
Default Gateway . . . . . : 0.0.0.0

C:\Users\Student>ipconfig /renew

Windows IP Configuration

Ethernet adapter Ethernet:

Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . : fe80::6dbe:de72:119a:6f2b%5
IPv4 Address . . . . . : 10.0.0.34
Subnet Mask . . . . . : 255.255.255.248
Default Gateway . . . . . : 10.0.0.33

C:\Users\Student>
```

16. We can verify that this lease was given by L3Switch2 by running "show dhcp-server" on it. We can see that the lease given by the switch matches the one received by the computer.



```
L3Switch2 - PuTTY
Remember to save your configuration changes.

There has been 1 successful login since last reboot and 0 failed logins since last successful login
No prior logins by this user since last reboot

EXOS-VM.1 # show dhcp-
    dhcp-client      Show information on DHCP client
    dhcp-server      Show information related to the dhcp-server
EXOS-VM.1 # show dhcp-server
VLAN "VLAN_0030":
    DHCP Address Range   : 10.0.0.34->10.0.0.38
    Netlogin Lease Timer : Not configured (Default = 10 seconds)
    DHCP Lease Timer     : Not configured (Default = 7200 seconds)
    Default Gateway      : 10.0.0.33
    Ports DHCP Enabled   : 3-4

=====
IP                  MAC                State        Lease Time Left
=====
10.0.0.34          0c:9d:c9:93:00:00  Assigned      0001:56:41

EXOS-VM.2 #
```

Test Case #5: Layer 2 Link Redundancy and Spanning-Tree Protocol (802.1w)

Enable and manage the Spanning-Tree Protocol to establish redundant Layer 2 paths while avoiding possible loops and broadcast storms. Identify the Layer 2 devices that will become the Root Bridge.

Functionality

*Describe the **functionality** of the test case in relation to your network project. Identify the relevant tools (devices, subnets, etc.) used in this test case and their specific interactions.*

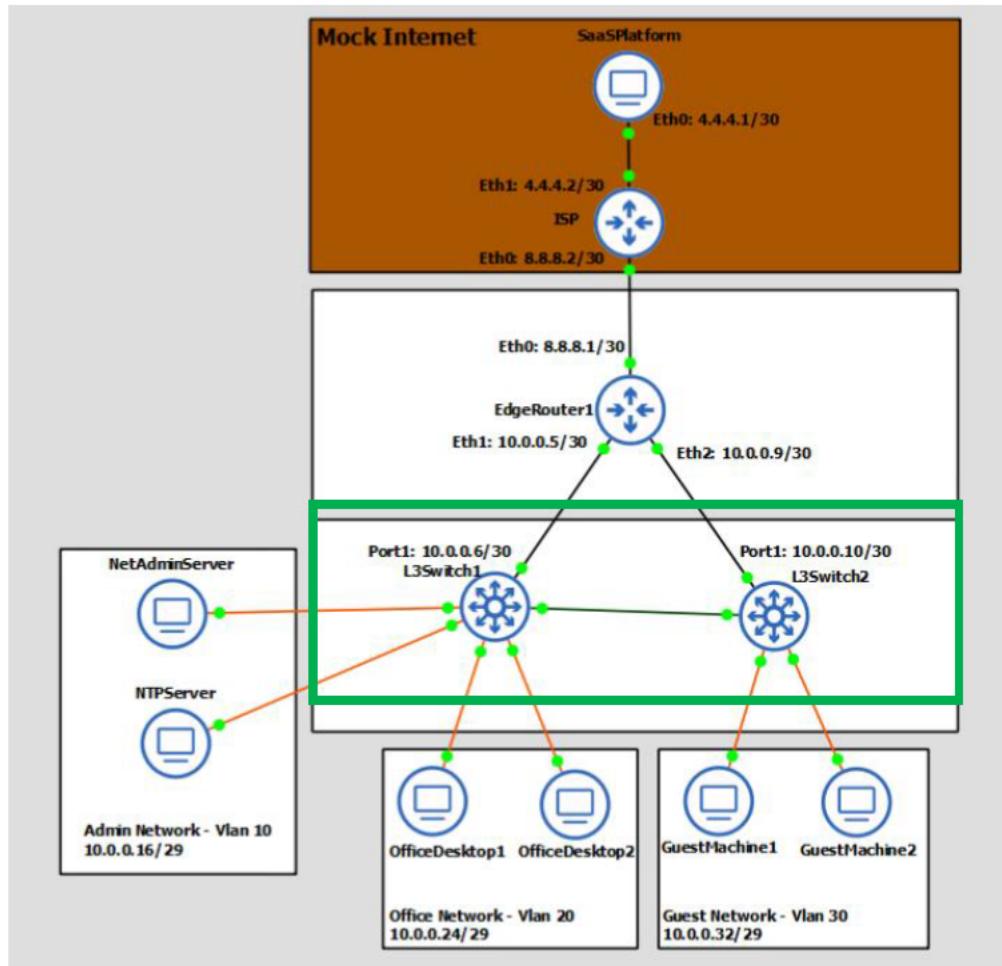
There are two switches on the network, L3Switch1 and L3Switch2. There are no switching loops in the architecture since there are only 2 switches only connected by 1 link. STP is enabled on both switches.

Network Diagram or Segment

*Provide a **network diagram or segment** visualizing the topology and devices used in this test case.*

Devices that are directly applicable to the testing method are enclosed in green boxes.





Legend

SaaS	Software as a Service
Eth*	Ethernet + port number
ISP	Internet Service Provider
L3	Layer 3
NetAdminServer	Network Administration Server
Admin	Administration
VLAN	Virtual Local Area Network

Testing Method

Summarize the **testing method** used to verify functionality of the network project within the virtual lab environment, including any metrics of success.

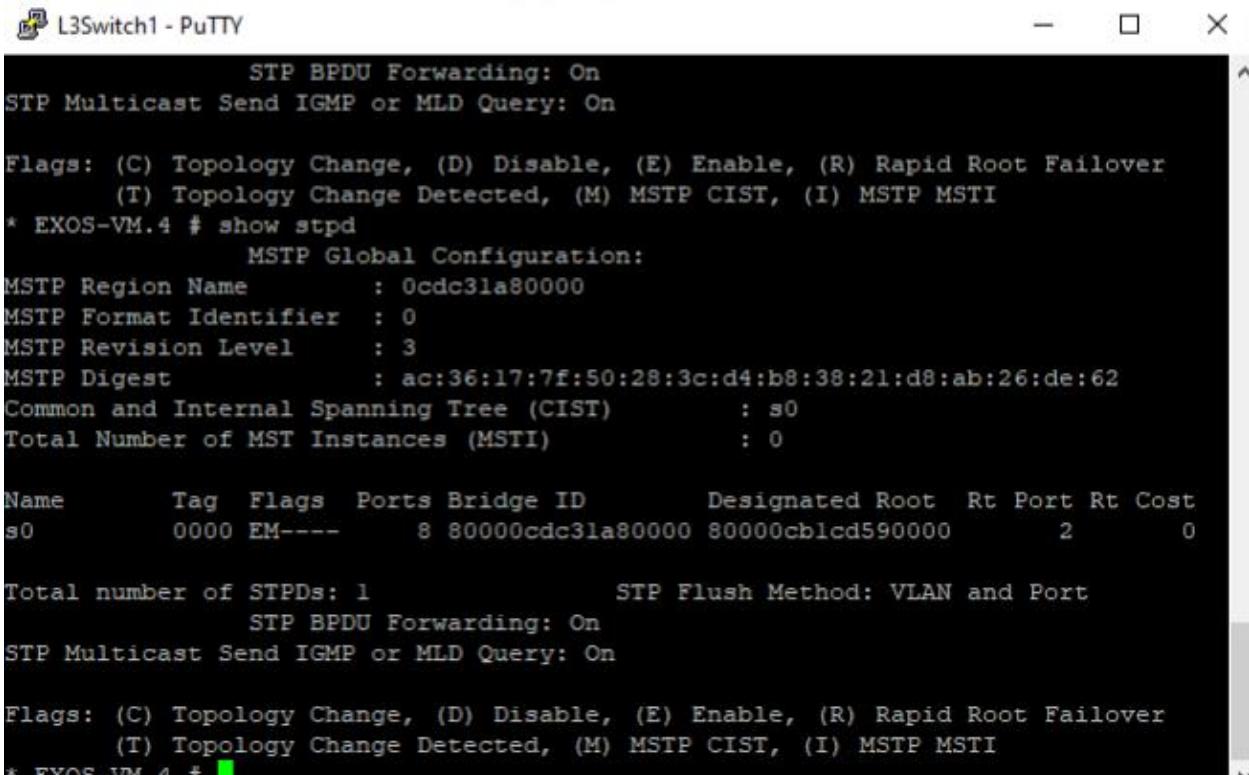
To demonstrate that STP is working as expected, I will show the STP configuration on each switch. Viewing the configuration, I will be able to see the elected root bridge. I will also capture packets on the link between L3Switch1 and L3Switch2 to view the exchange of STP information.



Process List

Provide a comprehensive **process list** of the steps taken within the network project to run the testing method. Include screenshots to illustrate the process and ensure clarity for others attempting to replicate the test.

1. Login to L3Switch1
2. Run "show stud". We can see that STP is not selecting L3Switch1 as the root bridge.



```
STP BPDU Forwarding: On
STP Multicast Send IGMP or MLD Query: On

Flags: (C) Topology Change, (D) Disable, (E) Enable, (R) Rapid Root Failover
      (T) Topology Change Detected, (M) MSTP CIST, (I) MSTP MSTI
* EXOS-VM.4 # show stpd

    MSTP Global Configuration:
    MSTP Region Name      : 0cdc31a80000
    MSTP Format Identifier : 0
    MSTP Revision Level   : 3
    MSTP Digest           : ac:36:17:7f:50:28:3c:d4:b8:38:21:d8:ab:26:de:62
    Common and Internal Spanning Tree (CIST)       : s0
    Total Number of MST Instances (MSTI)          : 0

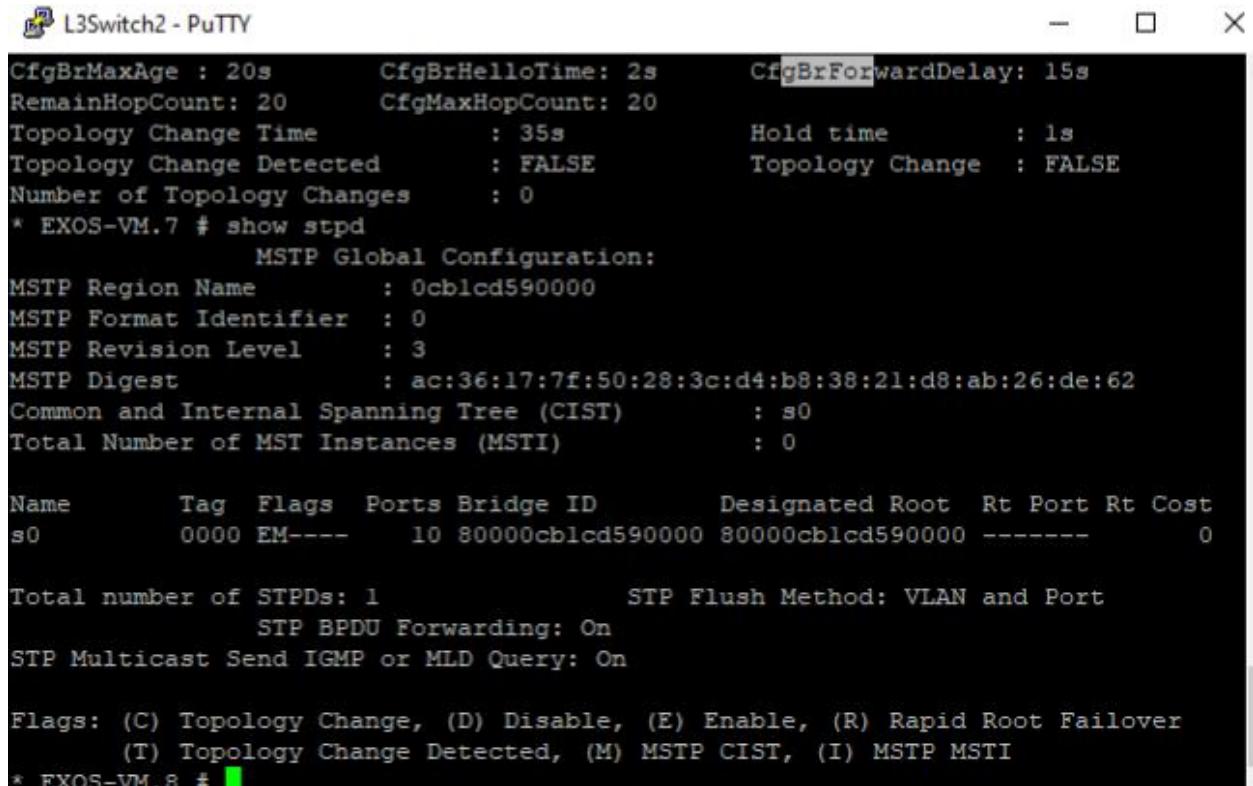
    Name      Tag  Flags  Ports Bridge ID      Designated Root  Rt Port Rt Cost
    s0        0000 EM---- 8 80000cdc31a80000 80000cb1cd590000      2      0

Total number of STPDs: 1                      STP Flush Method: VLAN and Port
                        STP BPDU Forwarding: On
                        STP Multicast Send IGMP or MLD Query: On

Flags: (C) Topology Change, (D) Disable, (E) Enable, (R) Rapid Root Failover
      (T) Topology Change Detected, (M) MSTP CIST, (I) MSTP MSTI
* EXOS-VM.4 #
```

3. Login to L3Switch2
4. Run "show stpd". We can see that STP is selecting L3Switch2 as the root bridge.





```

L3Switch2 - PuTTY

CfgBrMaxAge : 20s      CfgBrHelloTime: 2s      CfgBrForwardDelay: 15s
RemainHopCount: 20      CfgMaxHopCount: 20
Topology Change Time     : 35s      Hold time       : 1s
Topology Change Detected   : FALSE    Topology Change  : FALSE
Number of Topology Changes : 0
* EXOS-VM.7 # show stpd

        MSTP Global Configuration:
MSTP Region Name          : 0cblcd590000
MSTP Format Identifier     : 0
MSTP Revision Level        : 3
MSTP Digest                : ac:36:17:7f:50:28:3c:d4:b8:38:21:d8:ab:26:de:62
Common and Internal Spanning Tree (CIST)      : s0
Total Number of MST Instances (MSTI)           : 0

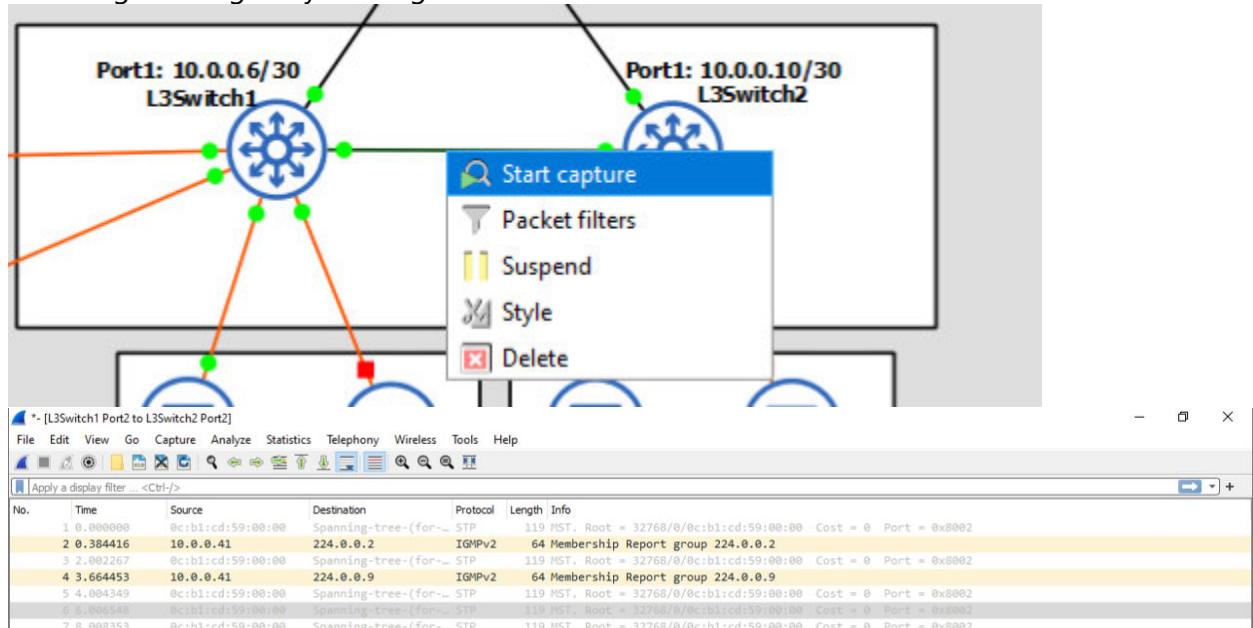
Name      Tag  Flags  Ports Bridge ID      Designated Root  Rt Port Rt Cost
s0        0000 EM---  10 80000cblcd590000  80000cblcd590000  -----  0

Total number of STPDs: 1      STP Flush Method: VLAN and Port
                           STP BPDU Forwarding: On
                           STP Multicast Send IGMP or MLD Query: On

Flags: (C) Topology Change, (D) Disable, (E) Enable, (R) Rapid Root Failover
       (T) Topology Change Detected, (M) MSTP CIST, (I) MSTP MSTI
* EXOS-VM.8 #

```

5. We can also see on the link from L3Switch1 eth2 to L3Switch2 eth2 that STP packets are being exchanged by running Wireshark on the link.



Test Case #6: Edge Device Syslog and NTP

Configure perimeter devices to generate system logs that capture unwanted traffic. Additionally, those perimeter devices should utilize Network Time Protocol (NTP) for clock synchronization.

Functionality

*Describe the **functionality** of the test case in relation to your network project. Identify the relevant tools (devices, subnets, etc.) used in this test case and their specific interactions.*

EdgeRouter1, L3Switch1, and L3Switch2 have ACLs configured which generate logs. A Windows 2019 Server is configured to be an NTP server which the three network devices use. The Windows Server has a static IP address of 10.0.0.18.

The only traffic allowed to EdgeRouter1 itself is OSPF. The only traffic allowed to EdgeRouter1 from the internet is traffic from established TCP connections which would have to be initiated from the internal network.

L3Switch1 blocks traffic from VLAN 10 to VLANs 20 and 30 on ports 3 and 6. It also blocks traffic from VLAN 20 to VLANs 10 and 30 on ports 4 and 5.

L3Switch1 blocks traffic from VLAN 30 to VLANs 10 and 20 on ports 3 and 4.

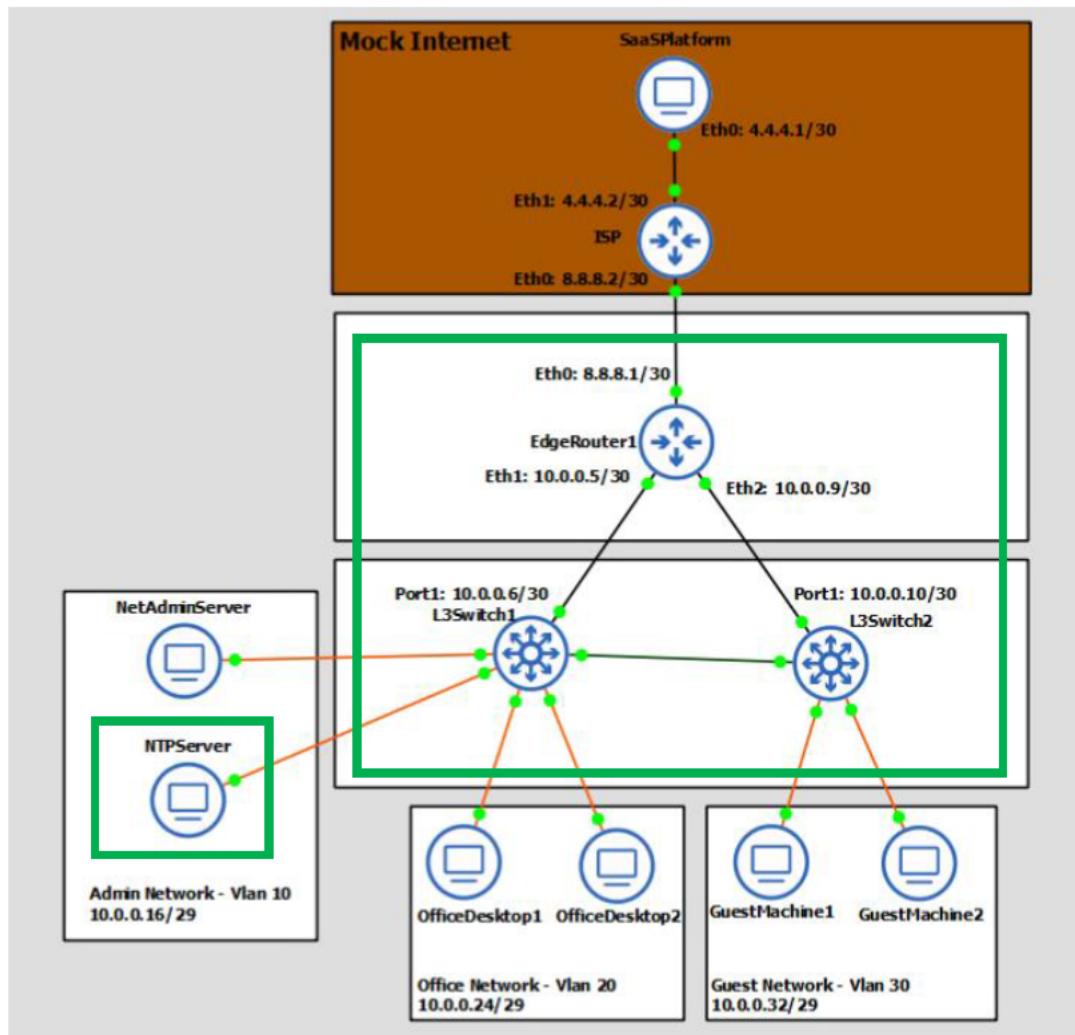
Network Diagram or Segment

*Provide a **network diagram or segment** visualizing the topology and devices used in this test case.*

Devices that are directly applicable to the testing method are enclosed in green boxes.



WESTERN GOVERNORS UNIVERSITY



Legend

SaaS	Software as a Service
Eth*	Ethernet + port number
ISP	Internet Service Provider
L3	Layer 3
NetAdminServer	Network Administration Server
Admin	Administration
VLAN	Virtual Local Area Network

Testing Method

Summarize the **testing method** used to verify functionality of the network project within the virtual lab environment, including any metrics of success.

To show that NTP is configured and being utilized on the three network devices, I will show the NTP server configuration, and show the traffic statistics to the server on the switches, and force a sync to the NTP service on the router.



To show that the ACLs are working on each device and logging, I will generate unwanted traffic from:

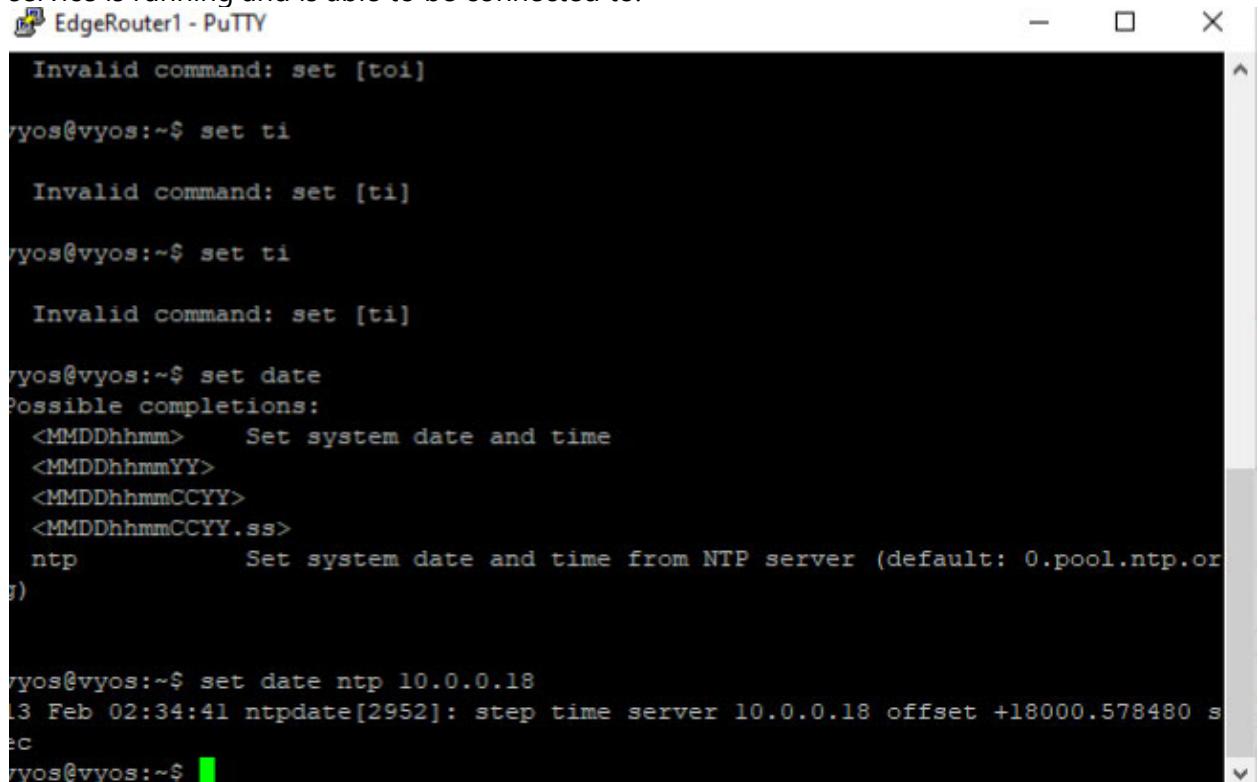
VLAN 10 to VLAN 20
VLAN 20 to VLAN 10
VLAN 30 to VLAN 10
SaaSPlatform to NetAdminServer

and show the associated logging on the relevant network device.

Process List

Provide a comprehensive **process list** of the steps taken within the network project to run the testing method. Include screenshots to illustrate the process and ensure clarity for others attempting to replicate the test.

1. Login to EdgeRouter1.
2. Run "set date ntp 10.0.0.18". This will force a sync with NTPServer to demonstrate the service is running and is able to be connected to.



```
EdgeRouter1 - PuTTY

Invalid command: set [toi]

vyos@vyos:~$ set ti

Invalid command: set [ti]

vyos@vyos:~$ set ti

Invalid command: set [ti]

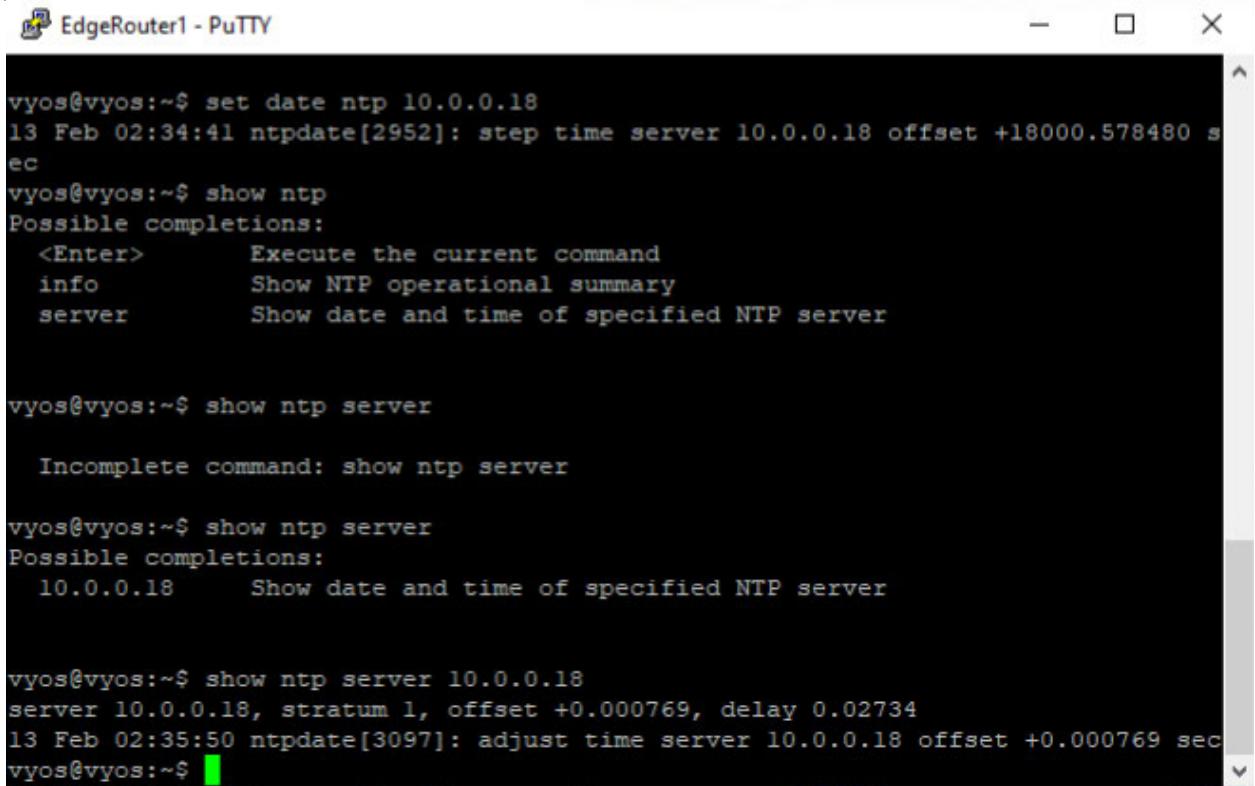
vyos@vyos:~$ set date
Possible completions:
<MMDDhhmm>      Set system date and time
<MMDDhhmmYY>
<MMDDhhmmCCYY>
<MMDDhhmmCCYY.ss>
ntp              Set system date and time from NTP server (default: 0.pool.ntp.or
g)

vyos@vyos:~$ set date ntp 10.0.0.18
13 Feb 02:34:41 ntpdate[2952]: step time server 10.0.0.18 offset +18000.578480 s
ec

vyos@vyos:~$
```



3. Run "show ntp server 10.0.0.18". This shows that the NTP configuration is persistent, not just a one-time sync.



A screenshot of a PuTTY terminal window titled "EdgeRouter1 - PuTTY". The window displays the following command-line session:

```
vyos@vyos:~$ set date ntp 10.0.0.18
13 Feb 02:34:41 ntpdate[2952]: step time server 10.0.0.18 offset +18000.578480 s
ec
vyos@vyos:~$ show ntp
Possible completions:
<Enter>      Execute the current command
info           Show NTP operational summary
server         Show date and time of specified NTP server

vyos@vyos:~$ show ntp server

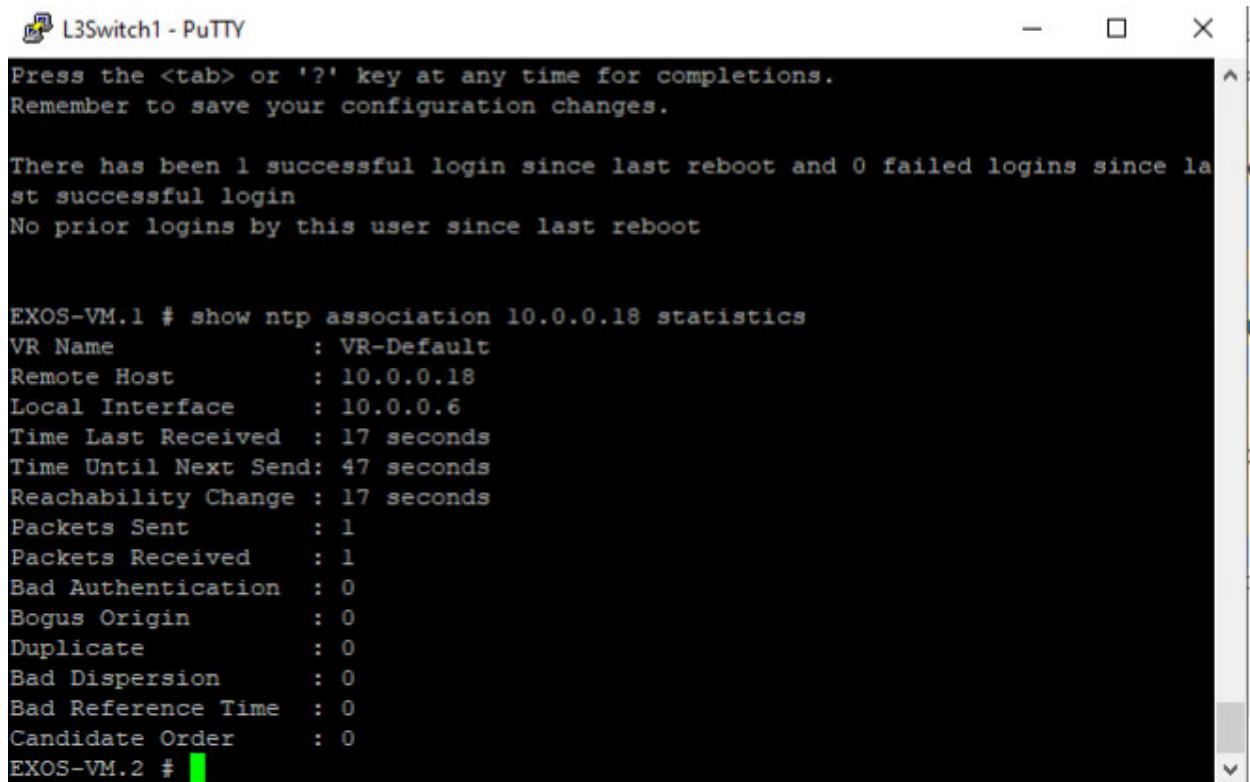
Incomplete command: show ntp server

vyos@vyos:~$ show ntp server
Possible completions:
10.0.0.18      Show date and time of specified NTP server

vyos@vyos:~$ show ntp server 10.0.0.18
server 10.0.0.18, stratum 1, offset +0.000769, delay 0.02734
13 Feb 02:35:50 ntpdate[3097]: adjust time server 10.0.0.18 offset +0.000769 sec
vyos@vyos:~$
```

4. Login to L3Switch1 and run "show ntp associations 10.0.0.18 statistics" which will show that packets are being exchanged between the NTP server and switch.





```

L3Switch1 - PuTTY

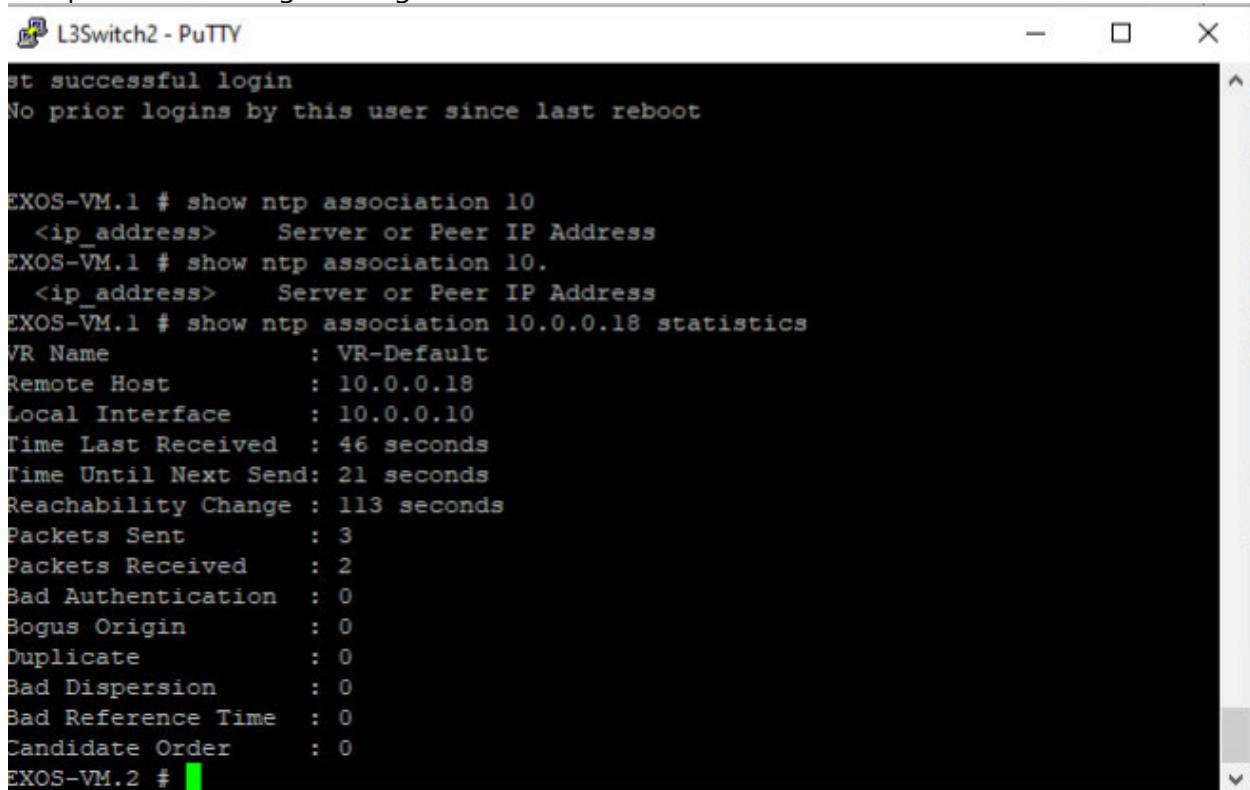
Press the <tab> or '?' key at any time for completions.
Remember to save your configuration changes.

There has been 1 successful login since last reboot and 0 failed logins since last
successful login
No prior logins by this user since last reboot

EXOS-VM.1 # show ntp association 10.0.0.18 statistics
VR Name          : VR-Default
Remote Host      : 10.0.0.18
Local Interface   : 10.0.0.6
Time Last Received : 17 seconds
Time Until Next Send: 47 seconds
Reachability Change : 17 seconds
Packets Sent     : 1
Packets Received  : 1
Bad Authentication : 0
Bogus Origin      : 0
Duplicate          : 0
Bad Dispersion     : 0
Bad Reference Time : 0
Candidate Order    : 0
EXOS-VM.2 #

```

5. Login to L3Switch2 and run "show ntp associations 10.0.0.18 statistics" which will show that packets are being exchanged between the NTP server and switch.



```

L3Switch2 - PuTTY

st successful login
No prior logins by this user since last reboot

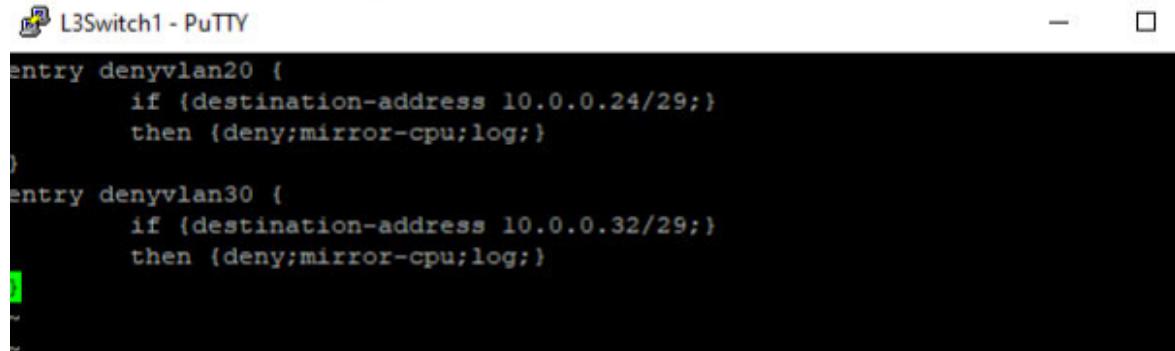
EXOS-VM.1 # show ntp association 10
<ip_address>  Server or Peer IP Address
EXOS-VM.1 # show ntp association 10.
<ip_address>  Server or Peer IP Address
EXOS-VM.1 # show ntp association 10.0.0.18 statistics
VR Name          : VR-Default
Remote Host      : 10.0.0.18
Local Interface   : 10.0.0.10
Time Last Received : 46 seconds
Time Until Next Send: 21 seconds
Reachability Change : 113 seconds
Packets Sent     : 3
Packets Received  : 2
Bad Authentication : 0
Bogus Origin      : 0
Duplicate          : 0
Bad Dispersion     : 0
Bad Reference Time : 0
Candidate Order    : 0
EXOS-VM.2 #

```



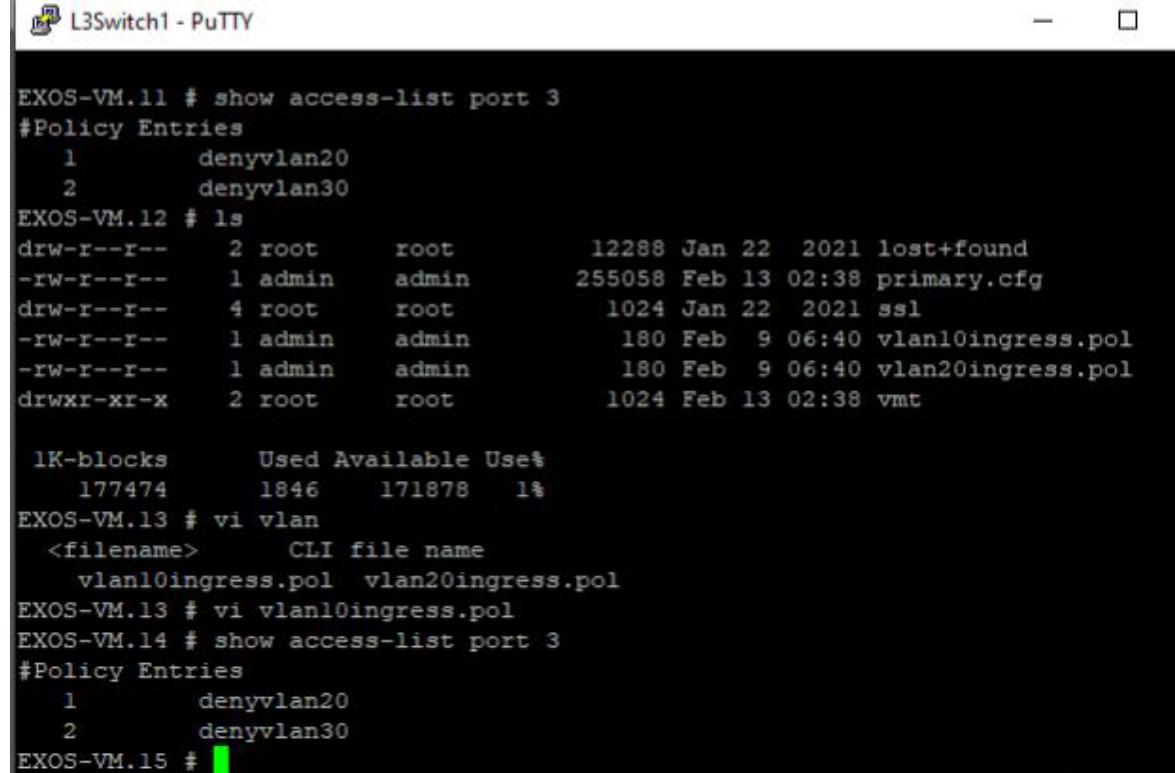
Viewing logs of unwanted traffic from VLAN 10 to VLAN 20:

1. Login to L3Switch1.
2. Run "vi vlan10ingress.pol" to view the ACL configured on VLAN 10. This shows the rules that will be triggered when sending traffic from VLAN 10 to VLAN 20.



```
L3Switch1 - PuTTY
entry denyvlan20 {
    if {destination-address 10.0.0.24/29;}
    then {deny;mirror-cpu;log;}
}
entry denyvlan30 {
    if {destination-address 10.0.0.32/29;}
    then {deny;mirror-cpu;log;}
}
```

3. Verify that this rule is applied to the NetAdminServer port by running "show access-list port 3". We can see that the two rules are applied.



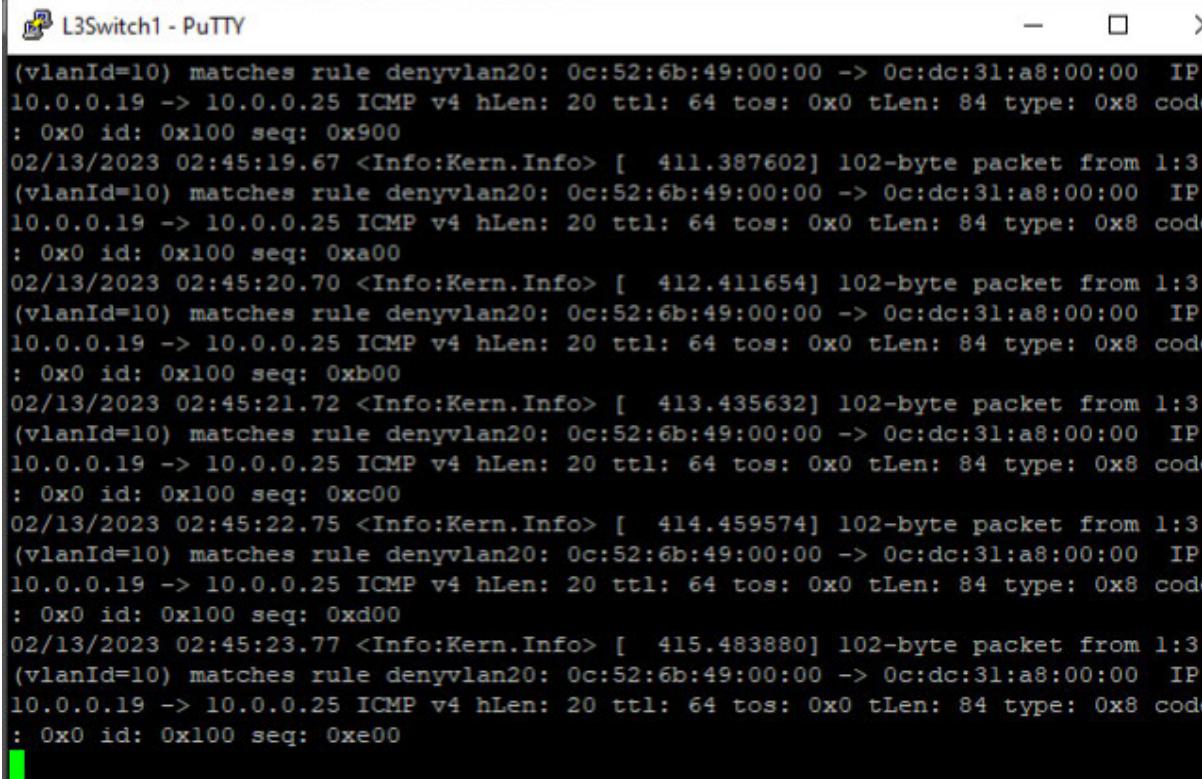
```
EXOS-VM.11 # show access-list port 3
#Policy Entries
 1      denyvlan20
 2      denyvlan30
EXOS-VM.12 # ls
drw-r--r--  2 root      root      12288 Jan 22  2021 lost+found
-rw-r--r--  1 admin     admin     255058 Feb 13  02:38 primary.cfg
drw-r--r--  4 root      root      1024 Jan 22  2021 ssl
-rw-r--r--  1 admin     admin      180 Feb   9 06:40 vlan10ingress.pol
-rw-r--r--  1 admin     admin      180 Feb   9 06:40 vlan20ingress.pol
drwxr-xr-x  2 root      root      1024 Feb 13  02:38 vmt

 1K-blocks      Used Available Use%
 177474       1846      171878   1%
EXOS-VM.13 # vi vlan
<filename>      CLI file name
  vlan10ingress.pol  vlan20ingress.pol
EXOS-VM.13 # vi vlan10ingress.pol
EXOS-VM.14 # show access-list port 3
#Policy Entries
 1      denyvlan20
 2      denyvlan30
EXOS-VM.15 #
```

4. Run "enable log target session" to display system logs in the terminal
5. Login to NetAdminServer.
6. Open a terminal and ping the VLAN 20 IP address 10.0.0.25. We can see that the ICMP traffic is being denied by rule "denyvlan20" in the system logs that



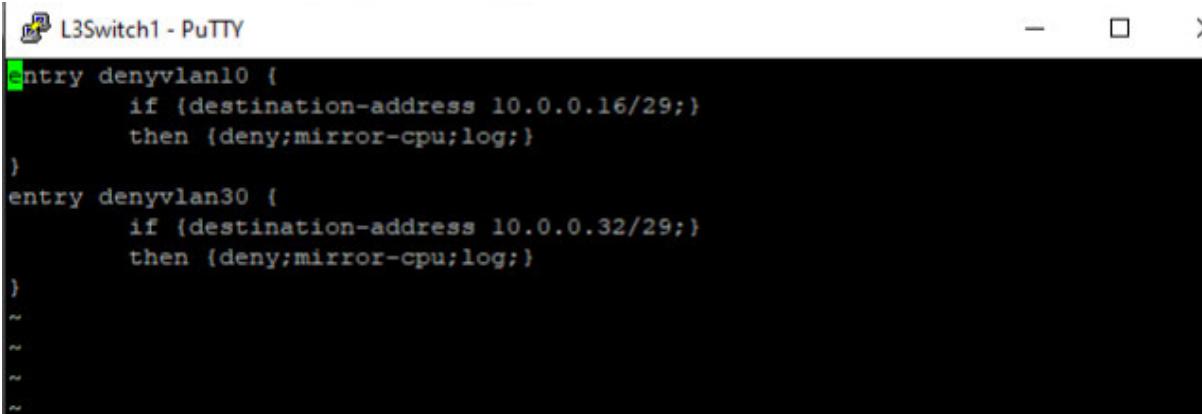
display on screen.



```
(vlanId=10) matches rule denyvlan20: 0c:52:6b:49:00:00 -> 0c:dc:31:a8:00:00  IP
10.0.0.19 -> 10.0.0.25 ICMP v4 hLen: 20 ttl: 64 tos: 0x0 tLen: 84 type: 0x8 code: 0x0 id: 0x100 seq: 0x900
02/13/2023 02:45:19.67 <Info:Kern.Info> [ 411.387602] 102-byte packet from 1:3
(vlanId=10) matches rule denyvlan20: 0c:52:6b:49:00:00 -> 0c:dc:31:a8:00:00  IP
10.0.0.19 -> 10.0.0.25 ICMP v4 hLen: 20 ttl: 64 tos: 0x0 tLen: 84 type: 0x8 code: 0x0 id: 0x100 seq: 0xa00
02/13/2023 02:45:20.70 <Info:Kern.Info> [ 412.411654] 102-byte packet from 1:3
(vlanId=10) matches rule denyvlan20: 0c:52:6b:49:00:00 -> 0c:dc:31:a8:00:00  IP
10.0.0.19 -> 10.0.0.25 ICMP v4 hLen: 20 ttl: 64 tos: 0x0 tLen: 84 type: 0x8 code: 0x0 id: 0x100 seq: 0xb00
02/13/2023 02:45:21.72 <Info:Kern.Info> [ 413.435632] 102-byte packet from 1:3
(vlanId=10) matches rule denyvlan20: 0c:52:6b:49:00:00 -> 0c:dc:31:a8:00:00  IP
10.0.0.19 -> 10.0.0.25 ICMP v4 hLen: 20 ttl: 64 tos: 0x0 tLen: 84 type: 0x8 code: 0x0 id: 0x100 seq: 0xc00
02/13/2023 02:45:22.75 <Info:Kern.Info> [ 414.459574] 102-byte packet from 1:3
(vlanId=10) matches rule denyvlan20: 0c:52:6b:49:00:00 -> 0c:dc:31:a8:00:00  IP
10.0.0.19 -> 10.0.0.25 ICMP v4 hLen: 20 ttl: 64 tos: 0x0 tLen: 84 type: 0x8 code: 0x0 id: 0x100 seq: 0xd00
02/13/2023 02:45:23.77 <Info:Kern.Info> [ 415.483880] 102-byte packet from 1:3
(vlanId=10) matches rule denyvlan20: 0c:52:6b:49:00:00 -> 0c:dc:31:a8:00:00  IP
10.0.0.19 -> 10.0.0.25 ICMP v4 hLen: 20 ttl: 64 tos: 0x0 tLen: 84 type: 0x8 code: 0x0 id: 0x100 seq: 0xe00
```

Viewing logs of unwanted traffic from VLAN 20 to VLAN 10:

1. Login to L3Switch1
2. Run "vi vlan20ingress.pol" to view the ACL configured on VLAN 20. This shows the rules that will be triggered when sending traffic from VLAN 20 to VLAN 10.



```
entry denyvlan10 {
    if {destination-address 10.0.0.16/29;}
    then {deny;mirror-cpu;log;}
}
entry denyvlan30 {
    if {destination-address 10.0.0.32/29;}
    then {deny;mirror-cpu;log;}
}
```

3. Verify that this rule is applied to the NetAdminServer port by running "show access-list port 4" and "show access-list port 5". We can see that the two rules are applied to both ports in VLAN 20.



```
L3Switch1 - PuTTY
177474      1846      171878   1%
EXOS-VM.13 # vi vlan
<filename>      CLI file name
vlan10ingress.pol  vlan20ingress.pol
EXOS-VM.13 # vi vlan10ingress.pol
EXOS-VM.14 # show access-list port 3
#Policy Entries
1      denyvlan20
2      denyvlan30
EXOS-VM.15 # show ace
^
%% Invalid input detected at '^' marker.
EXOS-VM.15 # show access-list port 4 5
^
%% Invalid input detected at '^' marker.
EXOS-VM.16 # show access-list port 4
#Policy Entries
1      denyvlan10
2      denyvlan30
EXOS-VM.17 # show access-list port 5
#Policy Entries
1      denyvlan10
2      denyvlan30
EXOS-VM.18 #
```

4. Run "enable log target session" to display system logs in the terminal
5. Login to a host on the VLAN 20, the Office Network.
6. Open a terminal and ping the VLAN 10 IP address 10.0.0.17. We can see that the ICMP traffic is being denied by rule "denyvlan10" in the system logs that display on screen.



```

1      denyvlan10
2      denyvlan30
EXOS-VM.18 # vi vlan
<filename>      CLI file name
    vlan10ingress.pol  vlan20ingress.pol
EXOS-VM.18 # vi vlan20ingress.pol
EXOS-VM.19 # 02/13/2023 02:54:22.23 <Info:Kern.Info> [ 465.659738] 102-byte pa
ket from 1:3 (vlanId=10) matches rule denyvlan20: 0c:52:6b:49:00:00 -> 0c:dc:31:a8:00:00
IP 10.0.0.19 -> 10.0.0.25 ICMP v4 hLen: 20 ttl: 64 tos: 0x0 tLen: 84 type: 0x8 code: 0x0 id: 0x100 seq: 0x3f00
02/13/2023 02:54:27.24 <Info:Kern.Info> [ 954.967681] 78-byte packet from 1:4 (v
lanId=20) matches rule denyvlan10: 0c:47:e6:dc:00:00 -> 0c:dc:31:a8:00:00
IP 10.0.0.26 -> 10.0.0.17 ICMP v4 hLen: 20 ttl: 128 tos: 0x0 tLen: 60 type: 0x8 code: 0x0 id: 0x100 seq: 0x100
02/13/2023 02:54:32.24 <Info:Kern.Info> [ 959.972209] 78-byte packet from 1:4 (v
lanId=20) matches rule denyvlan10: 0c:47:e6:dc:00:00 -> 0c:dc:31:a8:00:00
IP 10.0.0.26 -> 10.0.0.17 ICMP v4 hLen: 20 ttl: 128 tos: 0x0 tLen: 60 type: 0x8 code: 0x0 id: 0x100 seq: 0x200
02/13/2023 02:54:37.23 <Info:Kern.Info> [ 964.972393] 78-byte packet from 1:4 (v
lanId=20) matches rule denyvlan10: 0c:47:e6:dc:00:00 -> 0c:dc:31:a8:00:00
IP 10.0.0.26 -> 10.0.0.17 ICMP v4 hLen: 20 ttl: 128 tos: 0x0 tLen: 60 type: 0x8 code: 0x0 id: 0x100 seq: 0x300
^C
EXOS-VM.19 #

```

Viewing logs of unwanted traffic from VLAN 30 to VLAN 10:

1. Login to L3Switch2
2. Run "vi vlan30ingress.pol" to view the ACL configured on VLAN 30. This shows the rules that will be triggered when sending traffic from VLAN 30 to VLAN 10. In this case, it will hit the "cleanup" rule which is a default deny.

```

entry allowinternet {
    if {destination-address 4.4.4.1/32;}
    then {permit;mirror-cpu;log;}

entry allowvlan30ip {
    if {destination-address 10.0.0.33/32;}
    then {permit;mirror-cpu;log;}

entry allowDHCP {
    if {destination-address 255.255.255.255/32;protocol udp;destination-port}
    then {permit;mirror-cpu;log;}

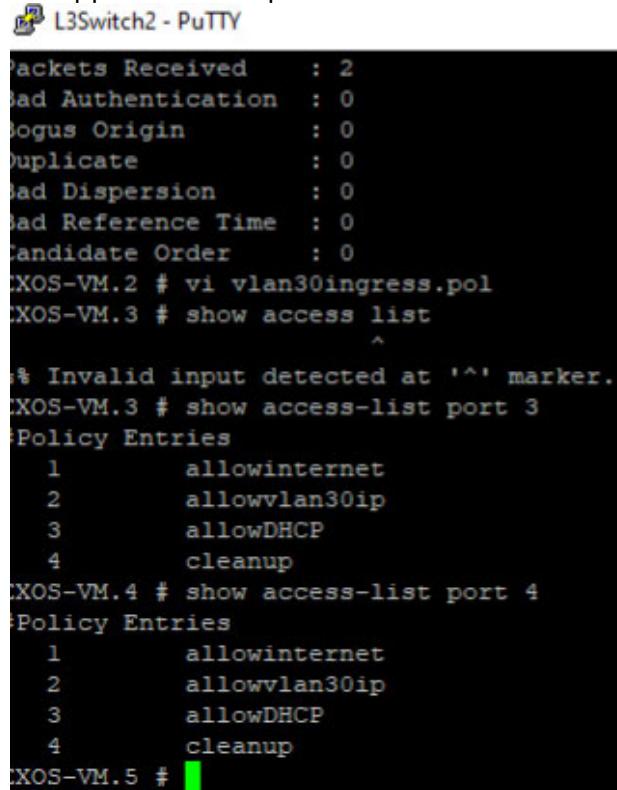
entry cleanup {
    if {destination-address 0.0.0.0/0;}
    then {deny;mirror-cpu;log;}
}

/usr/local/cfg/vlan30ingress.pol 1/16 6%

```



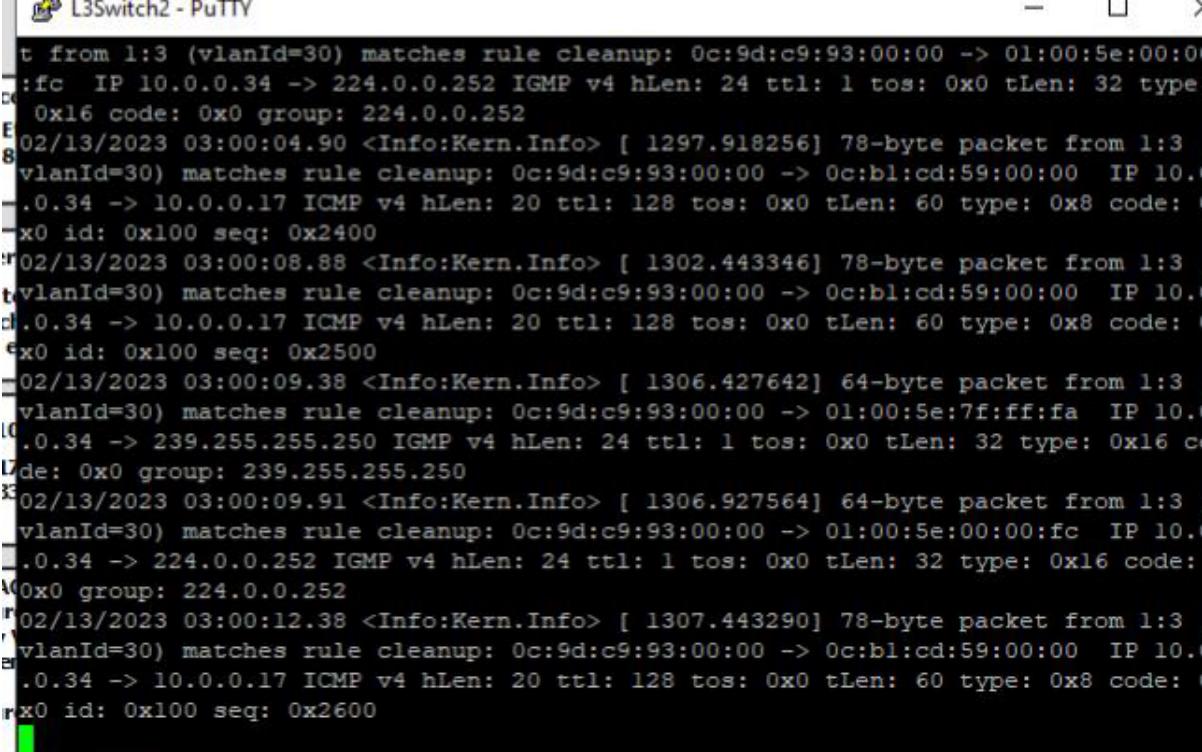
3. Verify that this rule is applied to the NetAdminServer port by running "show access-list port 3" and "show access-list port 3". We can see that the two rules are applied to both ports in VLAN 30.



```
L3Switch2 - PuTTY
Packets Received      : 2
Bad Authentication    : 0
Bogus Origin          : 0
Duplicate             : 0
Bad Dispersion         : 0
Bad Reference Time    : 0
Candidate Order        : 0
XOS-VM.2 # vi vlan30ingress.pol
XOS-VM.3 # show access list
^
% Invalid input detected at '^' marker.
XOS-VM.3 # show access-list port 3
#Policy Entries
  1      allowinternet
  2      allowvlan30ip
  3      allowDHCP
  4      cleanup
XOS-VM.4 # show access-list port 4
#Policy Entries
  1      allowinternet
  2      allowvlan30ip
  3      allowDHCP
  4      cleanup
XOS-VM.5 #
```

4. Run "enable log target session" to display system logs in the terminal
5. Login to a host on the VLAN 30, the Guest Network.
6. Open a terminal and ping the VLAN 10 IP address 10.0.0.17. We can see that the ICMP traffic is being denied by rule "cleanup" in the system logs that display on screen.





L3Switch2 - PuTTY

```
t from 1:3 (vlanId=30) matches rule cleanup: 0c:9d:c9:93:00:00 -> 01:00:5e:00:00:fc IP 10.0.0.34 -> 224.0.0.252 IGMP v4 hLen: 24 ttl: 1 tos: 0x0 tLen: 32 type 0x16 code: 0x0 group: 224.0.0.252
02/13/2023 03:00:04.90 <Info:Kern.Info> [ 1297.918256] 78-byte packet from 1:3
(vlanId=30) matches rule cleanup: 0c:9d:c9:93:00:00 -> 0c:bl:cd:59:00:00 IP 10.0.34 -> 10.0.0.17 ICMP v4 hLen: 20 ttl: 128 tos: 0x0 tLen: 60 type: 0x8 code: 0x0 id: 0x100 seq: 0x2400
02/13/2023 03:00:08.88 <Info:Kern.Info> [ 1302.443346] 78-byte packet from 1:3
(vlanId=30) matches rule cleanup: 0c:9d:c9:93:00:00 -> 0c:bl:cd:59:00:00 IP 10.0.34 -> 10.0.0.17 ICMP v4 hLen: 20 ttl: 128 tos: 0x0 tLen: 60 type: 0x8 code: 0x0 id: 0x100 seq: 0x2500
02/13/2023 03:00:09.38 <Info:Kern.Info> [ 1306.427642] 64-byte packet from 1:3
(vlanId=30) matches rule cleanup: 0c:9d:c9:93:00:00 -> 01:00:5e:7f:ff:fa IP 10.0.34 -> 239.255.255.250 IGMP v4 hLen: 24 ttl: 1 tos: 0x0 tLen: 32 type: 0x16 code: 0x0 group: 239.255.255.250
02/13/2023 03:00:09.91 <Info:Kern.Info> [ 1306.927564] 64-byte packet from 1:3
(vlanId=30) matches rule cleanup: 0c:9d:c9:93:00:00 -> 01:00:5e:00:00:fc IP 10.0.34 -> 224.0.0.252 ICMP v4 hLen: 24 ttl: 1 tos: 0x0 tLen: 32 type: 0x16 code: 0x0 group: 224.0.0.252
02/13/2023 03:00:12.38 <Info:Kern.Info> [ 1307.443290] 78-byte packet from 1:3
(vlanId=30) matches rule cleanup: 0c:9d:c9:93:00:00 -> 0c:bl:cd:59:00:00 IP 10.0.34 -> 10.0.0.17 ICMP v4 hLen: 20 ttl: 128 tos: 0x0 tLen: 60 type: 0x8 code: 0x0 id: 0x100 seq: 0x2600
```

Viewing logs of unwanted traffic from SaaSPlatform to NetAdminServer:

1. Login to EdgeRouter1
2. Run "show firewall name deny_inbound_rfc1918" to view firewall rules active on the eth0 inbound interface. Notice that you can also see the logging of packets and bytes that each rule processes.



```
EdgeRouter1 - PuTTY

-----
deny_inbound      Denies all non OSPF traffic to the router (eth0,local)
deny_inbound_rfc1918 (eth0,in)

vyos@vyos:~$ show firewall name de
deny_inbound      deny_inbound_rfc1918
vyos@vyos:~$ show firewall name deny_in
deny_inbound      deny_inbound_rfc1918
vyos@vyos:~$ show firewall name deny_inbound_rfc1918
Ruleset Information

-----
IPv4 Firewall "deny_inbound_rfc1918"

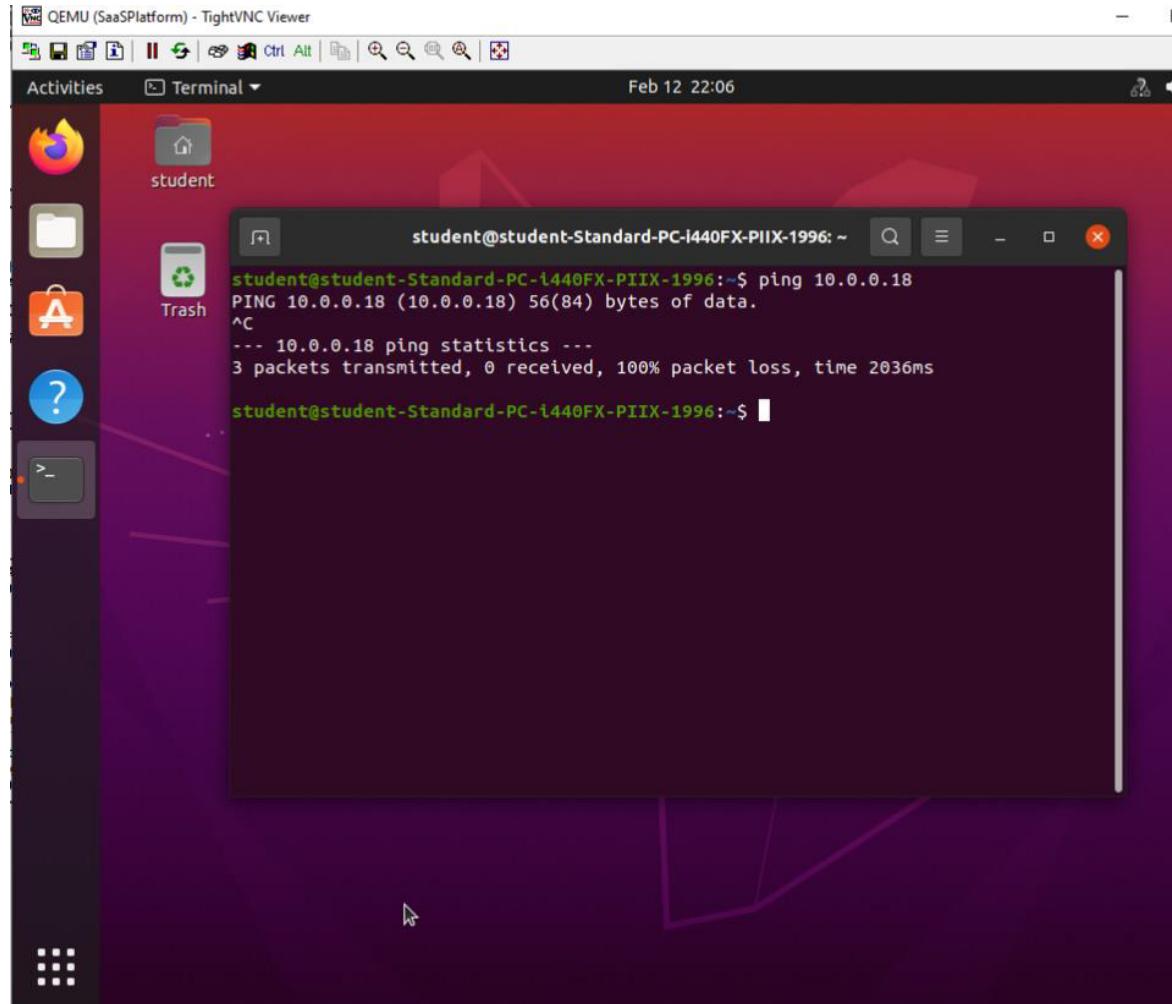
Active on: (eth0,in)

Rule    Action    Protocol    Packets    Bytes    Conditions
-----  -----
1       accept    all          2           120      ct state { established }
2       drop      icmp         0            0        meta l4proto icmp
3       drop      all          0            0        ct state { new }
default  drop      all          0            0

vyos@vyos:~$
```

3. Login to SaaSPlatform.
4. Open a terminal and run “ping 10.0.0.18”. We can see that there was no response.





5. Run "show firewall name deny_inbound_rfc1918" to see if rule 2 dropped the ICMP packets. We can see that rule 2 logged and dropped the 3 packets that



WESTERN GOVERNORS UNIVERSITY®

SaaSPlatform sent and logged the number of bytes sent.

EdgeRouter1 - PuTTY

```
Rule      Action     Protocol    Packets    Bytes    Conditions
-----  -----  -----  -----  -----  -----
1        accept    all          2          120    ct state { established }
2        drop      icmp         0           0    meta l4proto icmp
3        drop      all          0           0    ct state { new }
default   drop      all          0           0

vyos@vyos:~$ show firewall name deny_inbound_rfcl918
Ruleset Information

-----
IPv4 Firewall "deny_inbound_rfcl918"

Active on: (eth0,in)

Rule      Action     Protocol    Packets    Bytes    Conditions
-----  -----  -----  -----  -----  -----
1        accept    all          2          120    ct state { established }
2        drop      icmp         3          252    meta l4proto icmp
3        drop      all          0           0    ct state { new }
default   drop      all          0           0

vyos@vyos:~$
```



WESTERN GOVERNORS UNIVERSITY

Test Case #7: Basic Network Segmentation at Layer 2 via VLANs and 802.1q

Your network traffic should be segmented per department or service function at Layer 2 to enhance security and reduce network congestion at the switching layer while allowing segmented traffic to traverse between switches (VLAN trunking).

Functionality

*Describe the **functionality** of the test case in relation to your network project. Identify the relevant tools (devices, subnets, etc.) used in this test case and their specific interactions.*

There are 3 VLANs configured:

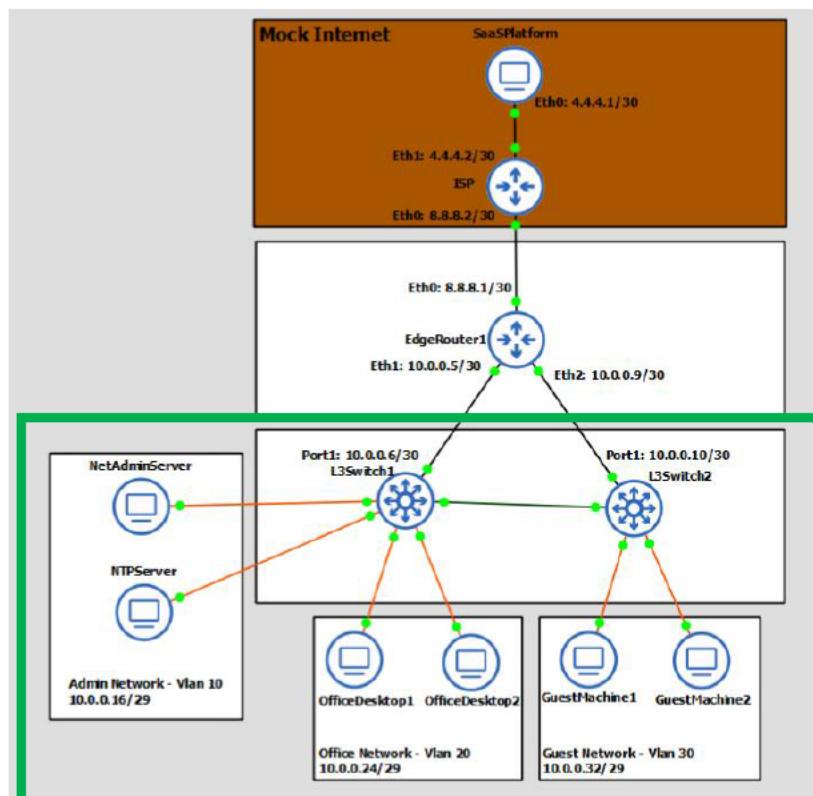
L3Switch1 Admin Network VLAN 10 10.0.0.16/29
 L3Switch1 Office Network VLAN 20 10.0.0.24/29
 L3Switch2 Guest Network VLAN 30 10.0.0.32/29

On L3Switch1, port 2 is connected to L3Switch2 and is configured as a tagged port.
 On L3Switch2, port 2 is connected to L3Switch1 and is configured as a tagged port.

Network Diagram or Segment

*Provide a **network diagram or segment** visualizing the topology and devices used in this test case.*

Devices that are directly applicable to the testing method are enclosed in green boxes.



Legend

SaaS	Software as a Service
Eth*	Ethernet + port number
ISP	Internet Service Provider
L3	Layer 3
NetAdminServer	Network Administration Server
Admin	Administration
VLAN	Virtual Local Area Network

Testing Method

Summarize the **testing method** used to verify functionality of the network project within the virtual lab environment, including any metrics of success.

To demonstrate that VLANs are configured, I will show the VLAN that each port on the switches is assigned to, the VLANs IP addresses and tagged ports.

To verify that VLAN trunking works, I will capture the packets sent on VLAN 40 between L3Switch1 eth2 and L3Switch2 eth2. This will contain an 802.1q tag in the packet.

Process List

Provide a comprehensive **process list** of the steps taken within the network project to run the testing method. Include screenshots to illustrate the process and ensure clarity for others attempting to replicate the test.

1. Login to L3Switch1
2. Run "show vlan". This will show each VLAN configured and its IP address.

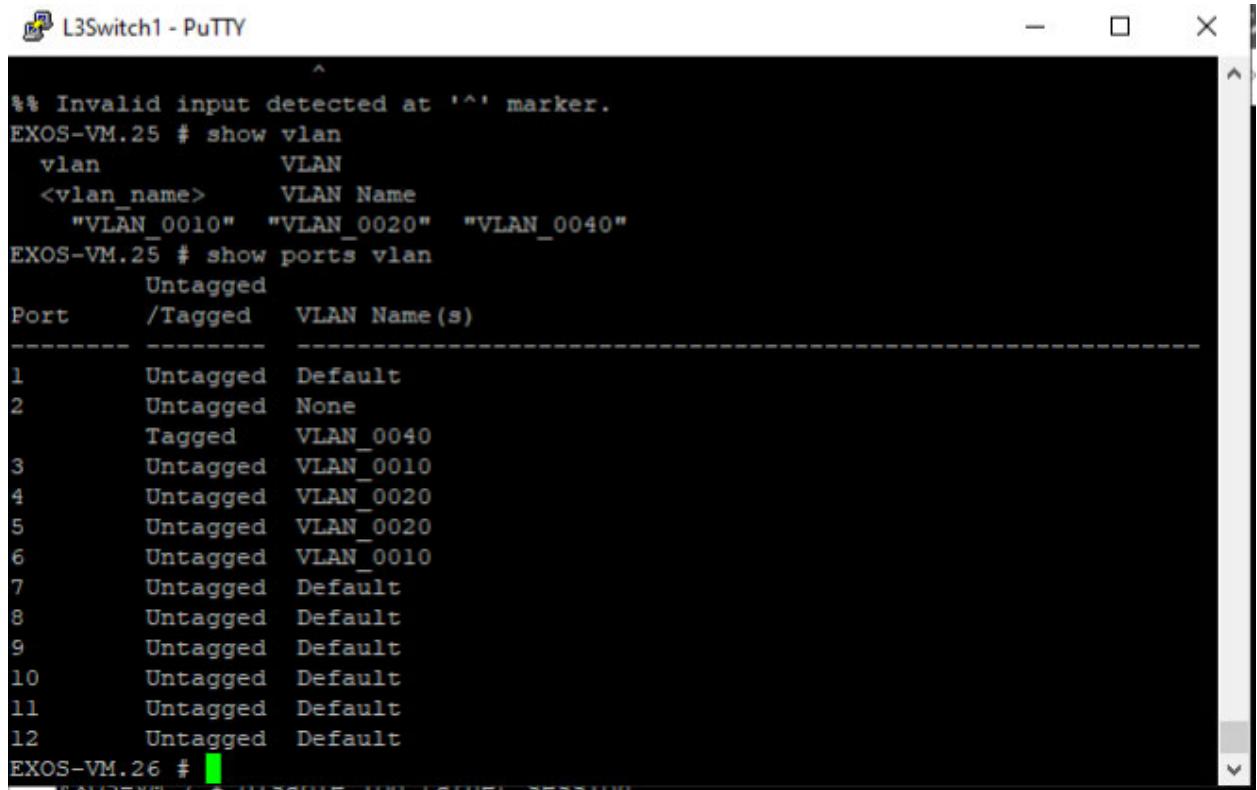
```

tive router
otal

-----
Default      1    10.0.0.6      /30  -f-----rT----- ANY   1
/7  VR-Default
Mgmt        4095 -----
/1  VR-Mgmt
VLAN_0010    10   10.0.0.17     /29  -f-----r----- ANY   2
/2  VR-Default
VLAN_0020    20   10.0.0.25     /29  -f-----r----- ANY   2
/2  VR-Default
VLAN_0040    40   10.0.0.42     /30  -f-----rT----- ANY   1
/1  VR-Default
-----
```

3. Run "show ports vlan" to show which VLAN each port is assigned to. We can see that



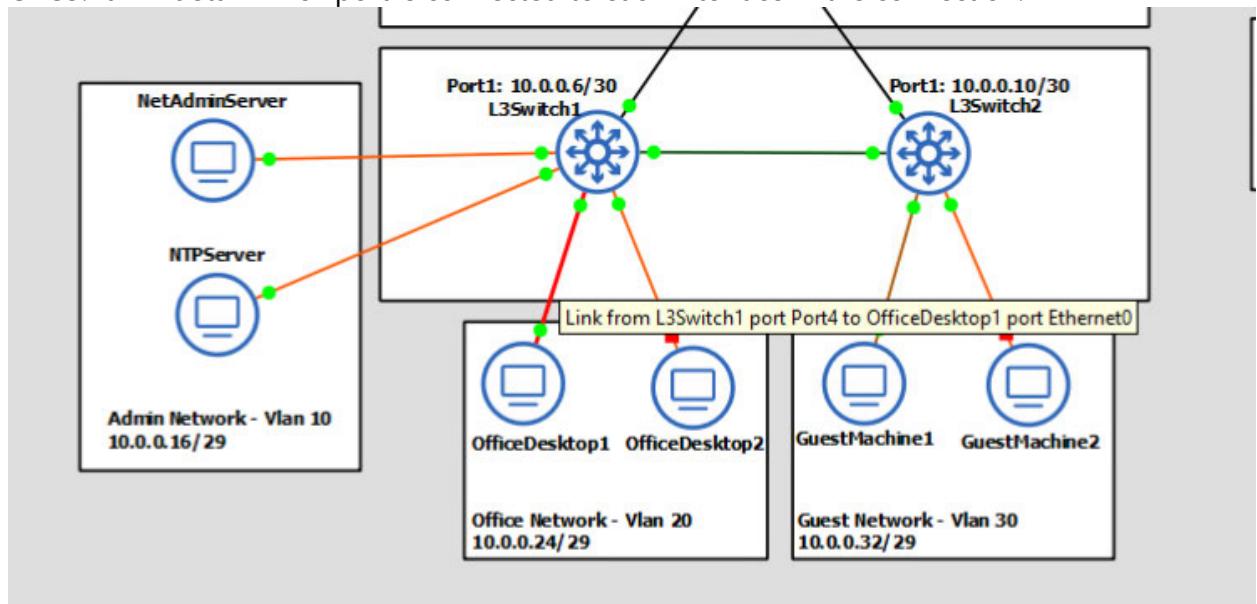


```

% Invalid input detected at '^' marker.
EXOS-VM.25 # show vlan
  vlan          VLAN
  <vlan_name>  VLAN Name
    "VLAN_0010"  "VLAN_0020"  "VLAN_0040"
EXOS-VM.25 # show ports vlan
      Untagged
Port   /Tagged  VLAN Name(s)
-----
1     Untagged  Default
2     Untagged  None
      Tagged   VLAN_0040
3     Untagged  VLAN_0010
4     Untagged  VLAN_0020
5     Untagged  VLAN_0020
6     Untagged  VLAN_0010
7     Untagged  Default
8     Untagged  Default
9     Untagged  Default
10    Untagged  Default
11    Untagged  Default
12    Untagged  Default
EXOS-VM.26 #

```

4. To verify that the network topology matches the output, hover over each connection in GNS3. It will detail which port is connected to each interface in the connection.



5. Port 2 is tagged with VLAN 40. We can verify this with "show ports tag 40". Port 2 is connected to L3Switch2 which is why it is a tagged interface.

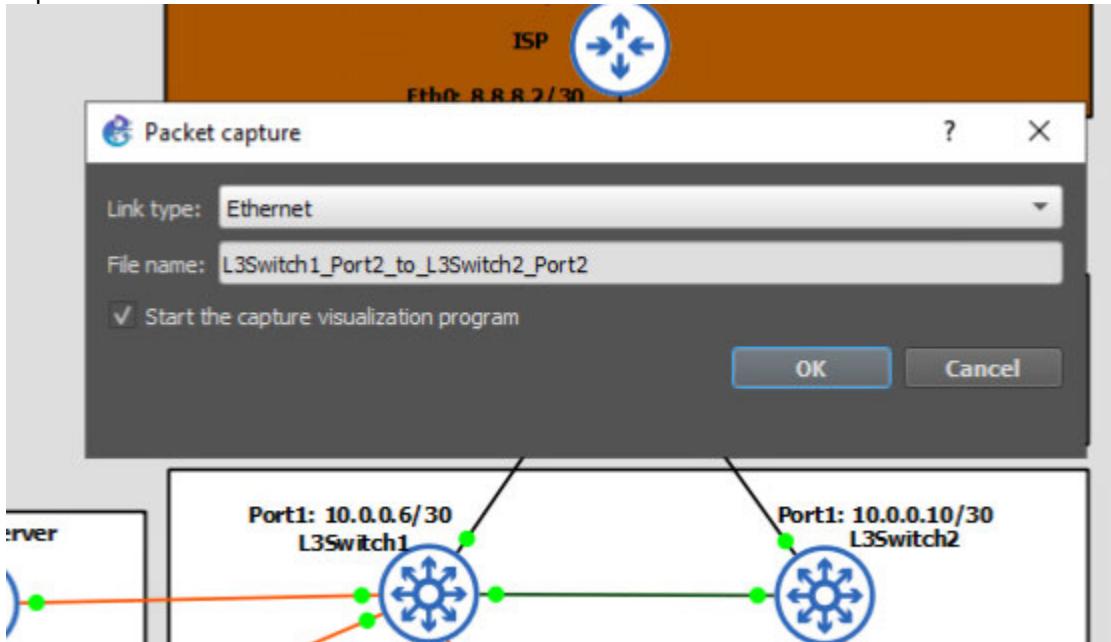


```

transceiver          DDMI info
txerrors            Displays transmit error statistics
EXOS-VM.31 # show ports t
tag                  Display ports in VMAN/VLANs with this IEEE 802.1Q or 802.1ad tag
ag
transceiver          DDMI info
txerrors            Displays transmit error statistics
EXOS-VM.31 # show ports tag
%% Incomplete command
EXOS-VM.32 # show ports tag
<tag>              tag ID between 1 and 4095
EXOS-VM.32 # show ports tag 40
Port Summary Monitor           Mon Feb 13 03:15:54 2023
Port      Display          VLAN Name       Port Link Speed Duplex
#        String            (or # VLANs)   State State Actual Actual
=====
2                   VLAN_0040          E     A    100   FULL

```

6. Repeat steps 1-5 for L3Switch2.
7. To verify that traffic is being tagged on port 2 of each interface, start a Wireshark packet capture on the link between L3Switch1 eth2 and L3Switch2 eth2.



8. By watching the packets, we can see that they are appropriately tagged. For example, L3Switch 1 and L3Switch2 are configured to broadcast routes with RIP. L3Switch2's VLAN 40 IP is 10.0.0.41. Looking at traffic from that source, we can see in the packet that there



is an 802.1q tag for VLAN ID 40. Likewise, if we look for L3Switch1's VLAN 40 IP address, 10.0.0.41, we can see that it is also tagged with VLAN ID 40.

The table below summarizes the captured traffic details from the screenshots:

No.	Time	Source	Destination	Protocol	Length	Info
24	31.996249	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
25	33.998969	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
26	36.001083	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
27	38.003221	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
28	40.005912	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
29	41.038696	10.0.0.41	224.0.0.9	RIPv2	190 Response	
30	41.915928	Extreme-EEP	Extreme-EDP	EDP	346 EDP: Info Display [0x15]	
31	41.916192	Extreme-EEP	Extreme-EDP	EDP	94 EDP: VL40	
32	42.008294	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
33	44.010598	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
34	45.063309	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
35	46.190599	10.0.0.42	224.0.0.9	RIPv2	190 Response	
36	46.283227	0:c:b1:cd:59:00:00	LLDP_Multicast	LLDP	150 MA/0:c:b1:cd:59:00:00 IN/2 120 SysN=EXOS-VM SysD=ExtremeXOS (EXOS-VM) version 31.1.1.3 31.1.1.3-p4	

No.	Time	Source	Destination	Protocol	Length	Info
24	31.996249	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
25	33.998969	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
26	36.001083	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
27	38.003221	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
28	40.005912	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
29	41.038696	10.0.0.41	224.0.0.9	RIPv2	190 Response	
30	41.915928	Extreme-EEP	Extreme-EDP	EDP	346 EDP: Info Display [0x15]	
31	41.916192	Extreme-EEP	Extreme-EDP	EDP	94 EDP: VL40	
32	42.008294	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
33	44.010598	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
34	45.063309	0:c:b1:cd:59:00:00	Spanning-tree-(for-> STP	119 MST. Root = 32768/0:0:c:b1:cd:59:00:00 Cost = 0 Port = 0x8002		
35	46.190599	10.0.0.42	224.0.0.9	RIPv2	190 Response	
36	46.283227	0:c:b1:cd:59:00:00	LLDP_Multicast	LLDP	150 MA/0:c:b1:cd:59:00:00 IN/2 120 SysN=EXOS-VM SysD=ExtremeXOS (EXOS-VM) version 31.1.1.3-p4	

Test Case #8: Basic or Advanced Networking

Custom Test Case

Define a **custom test case** to be run within your network project aligned to your specific organizational need or opportunity identified in Task 1. The custom test case should be equivalent in scope and requirements to the predefined test cases and pertain to basic or advanced networking.

"Your solution should have OSPF configured to get link state metrics to ensure that your connection to the critical service is fast and



WESTERN GOVERNORS UNIVERSITY

stable. You should also enable logging for link state adjacency changes on the edge router."

Functionality

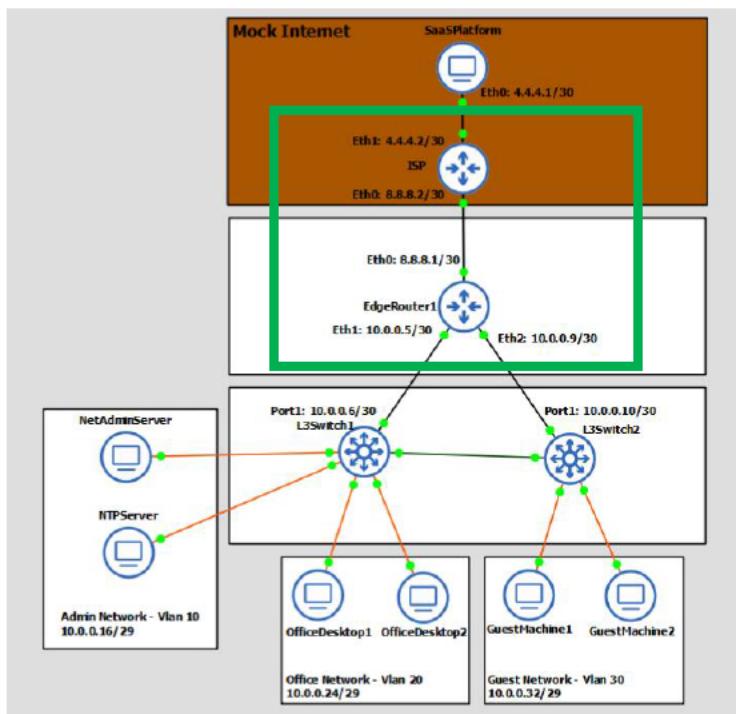
*Describe the **functionality** of the test case in relation to your network project. Identify the relevant tools (devices, subnets, etc.) used in this test case and their specific interactions.*

EdgeRouter1 and ISP have OSPF configured and are neighbors. They are both in area 1. EdgeRouter1 advertises network 8.8.8.0/30 and ISP advertises network 4.4.4.0/30.

Network Diagram or Segment

*Provide a **network diagram or segment** visualizing the topology and devices used in this test case.*

Devices that are directly applicable to the testing method are enclosed in green boxes.



Legend

SaaS	Software as a Service
Eth*	Ethernet + port number
ISP	Internet Service Provider
L3	Layer 3
NetAdminServer	Network Administration Server
Admin	Administration
VLAN	Virtual Local Area Network



Testing Method

Summarize the **testing method** used to verify functionality of the network project within the virtual lab environment, including any metrics of success.

To verify the configuration, I will show that ISP and EdgeRouter1 are OSPF neighbors, and they each have the expected routes in their routing tables. I will also verify that the correct interfaces are configured. Lastly, I will take a packet capture on the link between ISP eth0 and EdgeRouter1 eth0 to verify that OSPF packets are being exchanged.

Process List

Provide a comprehensive **process list** of the steps taken within the network project to run the testing method. Include screenshots to illustrate the process and ensure clarity for others attempting to replicate the test.

1. Login to EdgeRouter1.
2. Run "show ip ospf neighbor". This will show that the directly connected 8.8.8.2 interface on ISP is a neighbor.



```
vyos@vyos:~$ show ip ospf neighbor

Neighbor ID      Pri State          Up Time      Dead Time Address      Interface
terface          RXmtL RqstL DBsmL
8.8.8.2          1 Full/DR        2h10m01s    37.539s  8.8.8.2      et
h0:8.8.8.1

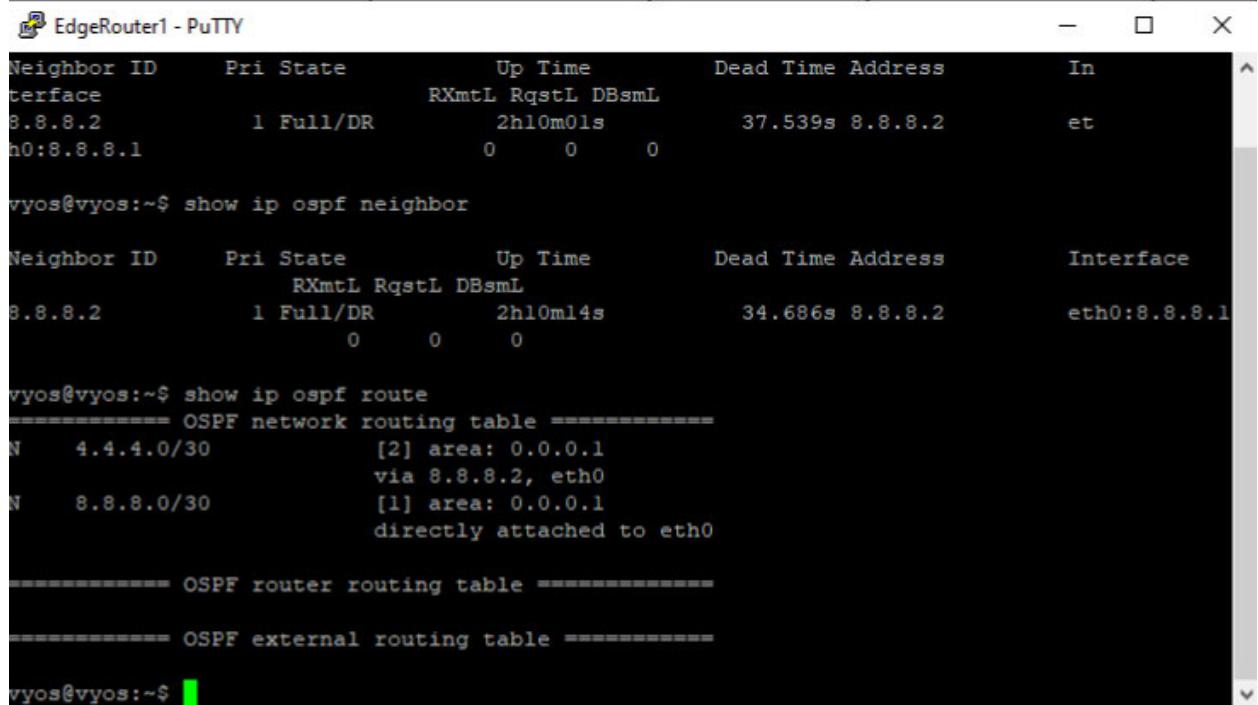
vyos@vyos:~$ show ip ospf neighbor

Neighbor ID      Pri State          Up Time      Dead Time Address      Interface
terface          RXmtL RqstL DBsmL
8.8.8.2          1 Full/DR        2h10m14s    34.686s  8.8.8.2      eth0:8.8.8.1

vyos@vyos:~$
```



3. Run "show ip ospf route" to verify that ISP is broadcasting the 4.4.4.0/30 network. We see that 4.4.4.0/30 is an OSPF route via ISP 8.8.8.2.



```

EdgeRouter1 - PuTTY

Neighbor ID      Pri State          Up Time      Dead Time Address      Interface
terface          RXmtL RqstL DBsmL
8.8.8.2          1 Full/DR        2h10m01s    37.539s  8.8.8.2      eth0:8.8.8.1

vyos@vyos:~$ show ip ospf neighbor

Neighbor ID      Pri State          Up Time      Dead Time Address      Interface
terface          RXmtL RqstL DBsmL
8.8.8.2          1 Full/DR        2h10m14s    34.686s  8.8.8.2      eth0:8.8.8.1

vyos@vyos:~$ show ip ospf route
===== OSPF network routing table =====
N   4.4.4.0/30           [2] area: 0.0.0.1
                                via 8.8.8.2, eth0
N   8.8.8.0/30           [1] area: 0.0.0.1
                                directly attached to eth0

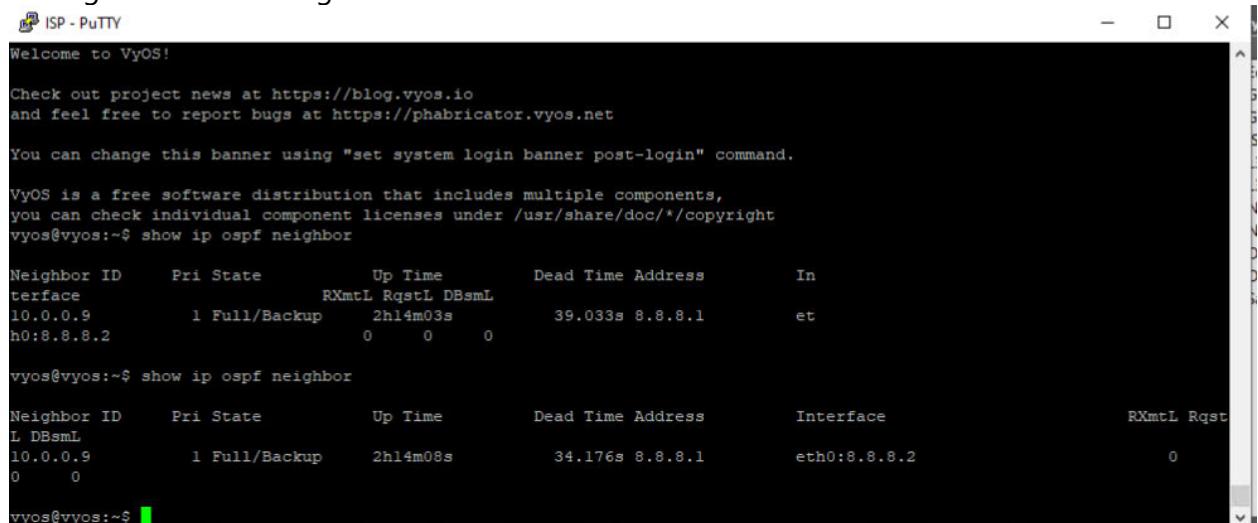
===== OSPF router routing table =====

===== OSPF external routing table =====

vyos@vyos:~$ 

```

4. Login to ISP.
 5. Run "show ip ospf neighbor". This will show that the directly connected 8.8.8.1 interface on EdgeRouter1 is a neighbor.



```

ISP - PuTTY

Welcome to VyOS!

Check out project news at https://blog.vyos.io
and feel free to report bugs at https://phabricator.vyos.net

You can change this banner using "set system login banner post-login" command.

VyOS is a free software distribution that includes multiple components,
you can check individual component licenses under /usr/share/doc/*copyright
vyos@vyos:~$ show ip ospf neighbor

Neighbor ID      Pri State          Up Time      Dead Time Address      Interface
terface          RXmtL RqstL DBsmL
10.0.0.9         1 Full/Backup    2h14m03s    39.033s  8.8.8.1      eth0:8.8.8.2

vyos@vyos:~$ show ip ospf neighbor

Neighbor ID      Pri State          Up Time      Dead Time Address      Interface
terface          RXmtL RqstL DBsmL
10.0.0.9         1 Full/Backup    2h14m08s    34.176s  8.8.8.1      eth0:8.8.8.2

vyos@vyos:~$ 

```

6. ISP is already aware of the 8.8.8.0/30 and 4.4.4.0/30 networks so it does not need to receive any routes from 8.8.8.1. We can see from running "show ip ospf route" that both



networks are directly connected.

```
ISP - PuTTY
vyos@vyos:~$ show ip ospf neighbor
Neighbor ID      Pri State          Up Time      Dead Time Address      Interface
10.0.0.9         1 Full/Backup    2h14m03s     39.033s  8.8.8.1      eth0:8.8.8.2
0:0:0:0:0:0:0:0

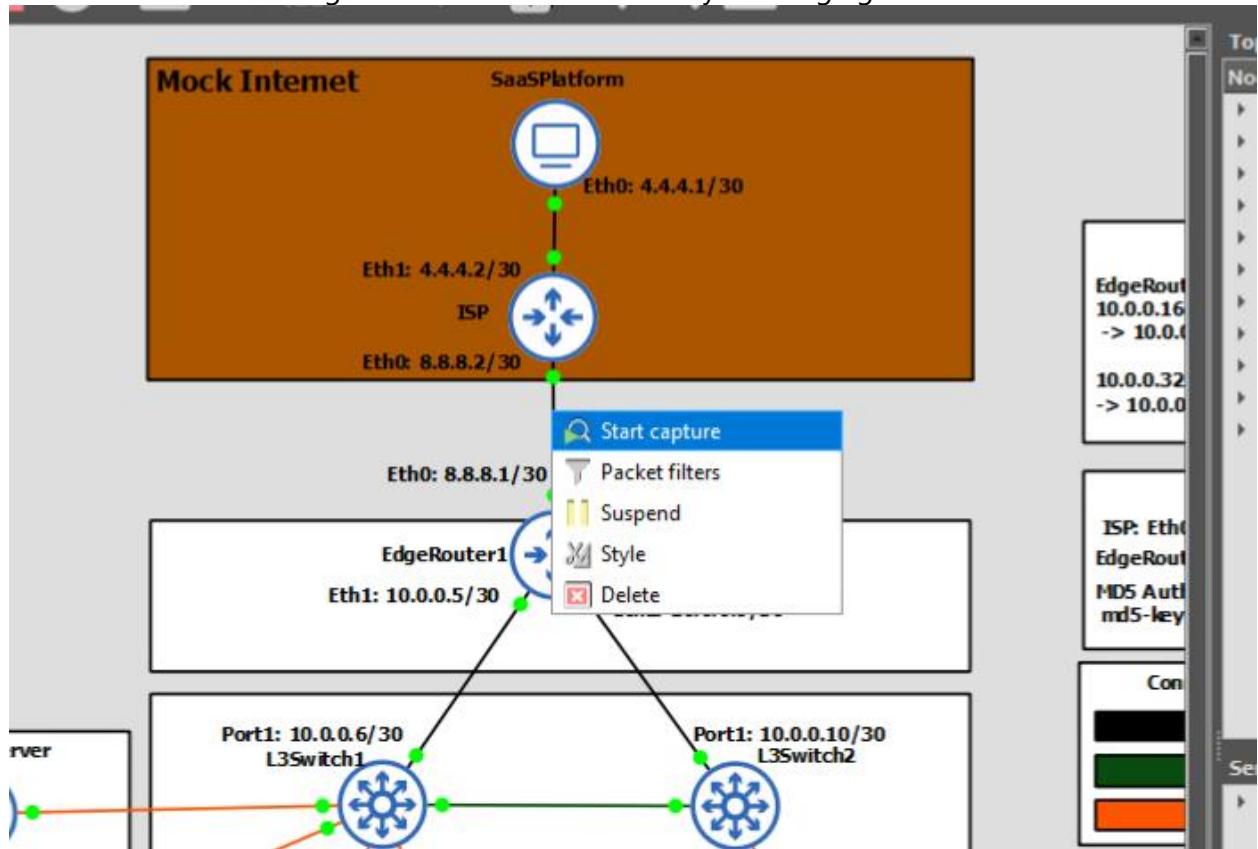
vyos@vyos:~$ show ip ospf neighbor
Neighbor ID      Pri State          Up Time      Dead Time Address      Interface
10.0.0.9         1 Full/Backup    2h14m08s     34.176s  8.8.8.1      eth0:8.8.8.2
0:0:0:0:0:0:0:0

vyos@vyos:~$ show ip ospf route
===== OSPF network routing table =====
N   4.4.4.0/30           [1] area: 0.0.0.1
                                directly attached to eth1
N   8.8.8.0/30           [1] area: 0.0.0.1
                                directly attached to eth0

===== OSPF router routing table =====
===== OSPF external routing table =====

vyos@vyos:~$
```

- Start a Wireshark packet capture on the link between EdgeRouter1 eth0 and ISP eth0. We can see that both EdgeRouter1 and ISP are actively exchanging OSPF info.



Capturing from - [EdgeRouter1 Ethernet0 to ISP Ethernet0]

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	8.8.8.2	224.0.0.5	OSPF	98	Hello Packet
2	3.455552	8.8.8.1	224.0.0.5	OSPF	98	Hello Packet
3	10.000329	8.8.8.2	224.0.0.5	OSPF	98	Hello Packet
4	13.455671	8.8.8.1	224.0.0.5	OSPF	98	Hello Packet

```

Checksum: 0x0000 (None)
Auth Type: Cryptographic (2)
Auth Crypt Key id: 1
Auth Crypt Data Length: 16
Auth Crypt Sequence Number: 1676259030
Auth Crypt Data: 18dbb4fd94a8b6803fcfc5f8c6d7a492b
(OSPF Hello Packet)
  Network Mask: 255.255.255.252
  Hello Interval [sec]: 10
  Options: 0x02, (E) External Routing
  Router Priority: 1
  Router Dead Interval [sec]: 40
  Designated Router: 8.8.8.2
  Backup Designated Router: 8.8.8.1
  Active Neighbor: 10.0.0.9

```

Test Case #9: Network Automation

Custom Test Case

Define a **custom test case** to be run within your network project aligned to your specific organizational need or opportunity identified in Task 1. The custom test case should be equivalent in scope and requirements to the predefined test cases and pertain to network automation.

"To save backups of network device configurations, a script should be configured on a server to retrieve the full network device configuration for each network device and store it on the server executing the script"

Functionality

Describe the **functionality** of the test case in relation to your network project. Identify the relevant tools (devices, subnets, etc.) used in this test case and their specific interactions.

NetAdminServer has a script located at "/home/student/NetAutomation/backup.py". The script uses netmiko to login to EdgeRouter1, L3Switch1 and L3Switch2 via SSH, show the full device configurations, and copy it to a folder located at "/home/student/NetAutomation/Backups" using the backup file name configured in the script.

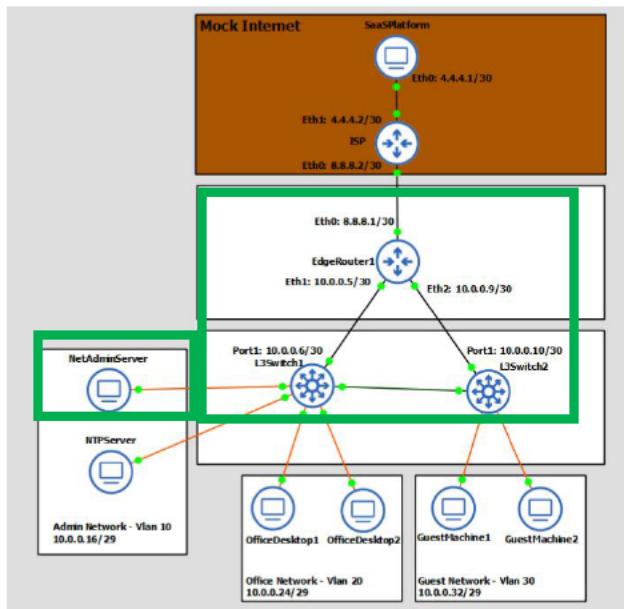
Network Diagram or Segment

Provide a **network diagram or segment** visualizing the topology and devices used in this test case.



WESTERN GOVERNORS UNIVERSITY®

Devices that are directly applicable to the testing method are enclosed in green boxes.



Legend

SaaS	Software as a Service
Eth*	Ethernet + port number
ISP	Internet Service Provider
L3	Layer 3
NetAdminServer	Network Administration Server
Admin	Administration
VLAN	Virtual Local Area Network

Testing Method

Summarize the **testing method** used to verify functionality of the network project within the virtual lab environment, including any metrics of success.

To verify the functionality of the script, I will remove any existing backups in the folder, execute the script, view the backups stored in the folder, and verify that they match the configurations on the network devices.

Process List

Provide a comprehensive **process list** of the steps taken within the network project to run the testing method. Include screenshots to illustrate the process and ensure clarity for others attempting to replicate the test.

1. Login to NetAdminServer.
2. Open a terminal and run "cd /home/student/NetAutomation/Backups"
3. Run "rm -f *" to remove all previous backups for testing.



4. Run "cd .. && python3 backup.py" to execute the script to get backups. If it errors, re-run the script with "python3 backup.py" since your current working directory will be where it is located now.

```

student@student-Standard-PC-i440FX-PIIX-1996:~/NetAutomation$ python3 backup.py
student@student-Standard-PC-i440FX-PIIX-1996:~/NetAutomation$ cd .. && python3 backup.py
python3: can't open file 'backup.py': [Errno 2] No such file or directory
student@student-Standard-PC-i440FX-PIIX-1996:~$ ls
Desktop Documents Downloads Music NetAutomation Pictures Public Templates Videos
student@student-Standard-PC-i440FX-PIIX-1996:~$ cd NetAutomation/
student@student-Standard-PC-i440FX-PIIX-1996:~/NetAutomation$ cd Backups/
student@student-Standard-PC-i440FX-PIIX-1996:~/NetAutomation/Backups$ ls
router1.bak switch1.bak
student@student-Standard-PC-i440FX-PIIX-1996:~/NetAutomation/Backups$ rm *
student@student-Standard-PC-i440FX-PIIX-1996:~/NetAutomation/Backups$ cd ..
student@student-Standard-PC-i440FX-PIIX-1996:~/NetAutomation$ python3 backup.py
Getting backup of 10.0.0.5's configuration
Getting backup of 10.0.0.6's configuration
Getting backup of 10.0.0.10's configuration
student@student-Standard-PC-i440FX-PIIX-1996:~/NetAutomation$ 
```

5. Run "cd Backups && ls" to view the new backup files.

```

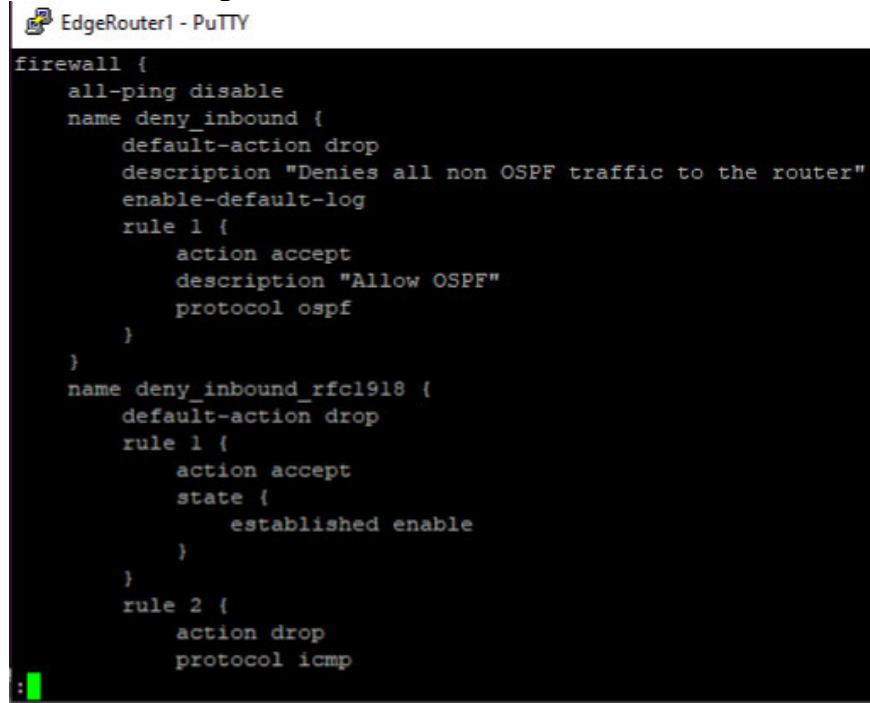
student@student-Standard-PC-i440FX-PIIX-1996:~/NetAutomation$ cd Backups/ && ls
EdgeRouter1.bak L3Switch1.bak L3Switch2.bak
student@student-Standard-PC-i440FX-PIIX-1996:~/NetAutomation$ 
```

6. For each file, Login to the respective network device and run "show config" on the network device. Compare the output of the "show config" with the backup taken on the NetAdminServer by opening the respective ".bak" text file in your favorite text editor and checking for any discrepancies.

- Example:
 - Login to EdgeRouter1.



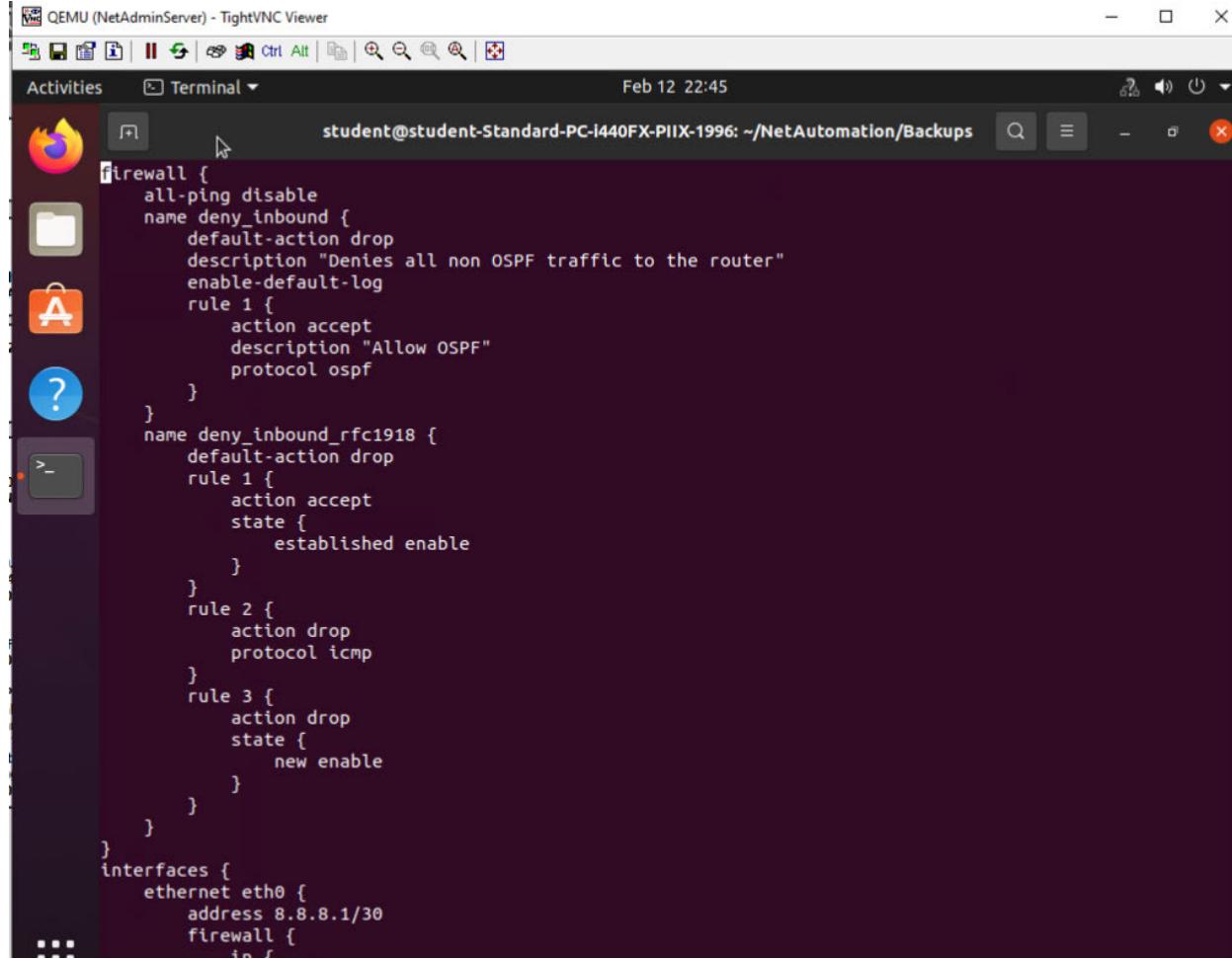
- ii. Run "show config"



```
firewall {
    all-ping disable
    name deny_inbound {
        default-action drop
        description "Denies all non OSPF traffic to the router"
        enable-default-log
        rule 1 {
            action accept
            description "Allow OSPF"
            protocol ospf
        }
    }
    name deny_inbound_rfc1918 {
        default-action drop
        rule 1 {
            action accept
            state {
                established enable
            }
        }
        rule 2 {
            action drop
            protocol icmp
        }
}
```

- iii. On NetAdminServer, open EdgeRouter1.bak with vim "vim EdgeRouter1.bak". We can initially see that these outputs are identical, and can conduct a manual review if needed.





```
student@student-Standard-PC-i440FX-PIIX-1996: ~/NetAutomation/Backups
```

```
firewall {
    all-ping disable
    name deny_inbound {
        default-action drop
        description "Denies all non OSPF traffic to the router"
        enable-default-log
        rule 1 {
            action accept
            description "Allow OSPF"
            protocol ospf
        }
        name deny_inbound_rfc1918 {
            default-action drop
            rule 1 {
                action accept
                state {
                    established enable
                }
            }
            rule 2 {
                action drop
                protocol icmp
            }
            rule 3 {
                action drop
                state {
                    new enable
                }
            }
        }
    }
    interfaces {
        ethernet eth0 {
            address 8.8.8.1/30
            firewall {
                in f
```

Test Case #10: Network Security

Custom Test Case

Define a **custom test case** to be run within your network project aligned to your specific organizational need or opportunity identified in Task 1. The custom test case should be equivalent in scope and requirements to the predefined test cases and pertain to network security.

"Configure MD5 OSPF authentication to ensure that only authorized users will receive routes to the SaaS and company network."

Functionality

Describe the **functionality** of the test case in relation to your network project. Identify the relevant tools (devices, subnets, etc.) used in this test case and their specific interactions.



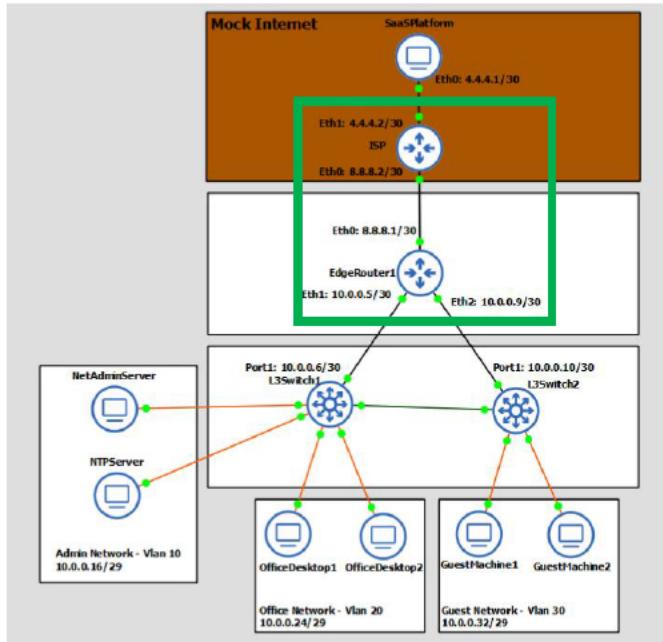
WESTERN GOVERNORS UNIVERSITY[®]

As discussed, both EdgeRouter1 and ISP have OSPF configured to advertise and receive routes. MD5 authentication has been configured on ISP eth0 and EdgeRouter1 eth0 using key-id1 1 with the md5-key "secretkey1"

Network Diagram or Segment

Provide a network diagram or segment visualizing the topology and devices used in this test case.

Devices that are directly applicable to the testing method are enclosed in green boxes.



Legend

SaaS	Software as a Service
Eth*	Ethernet + port number
ISP	Internet Service Provider
L3	Layer 3
NetAdminServer	Network Administration Server
Admin	Administration
VLAN	Virtual Local Area Network

Testing Method

Summarize the testing method used to verify functionality of the network project within the virtual lab environment, including any metrics of success.

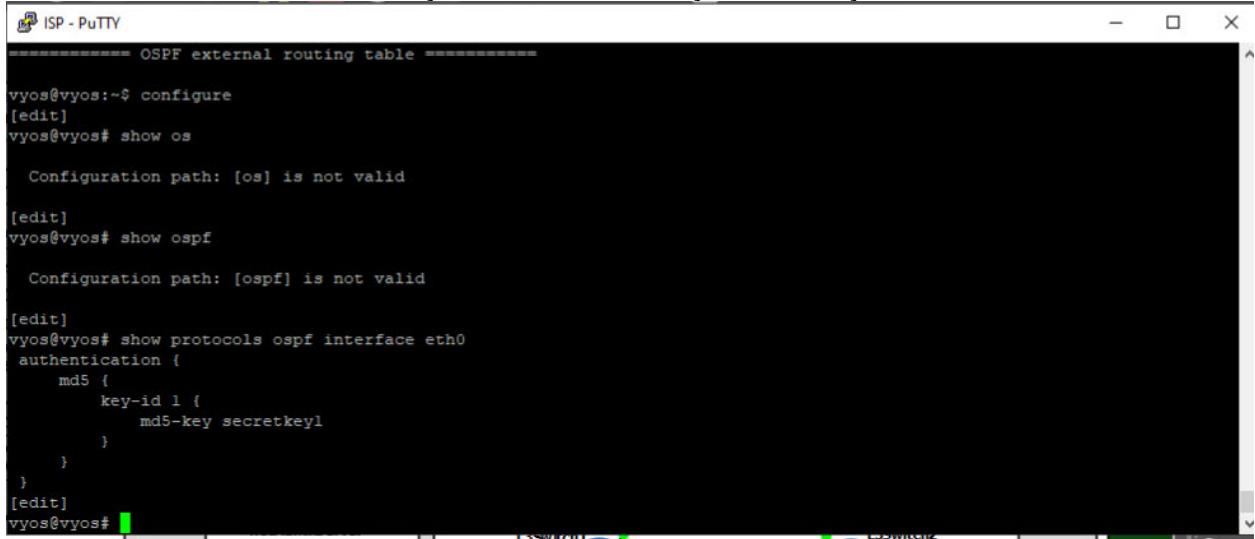
To verify the configuration, I will show the interface OSPF configuration, and inspect the OSPF packets on the link between EdgeRouter1 eth0 and ISP eth0 to verify that authentication parameters are contained in the packets.



Process List

Provide a comprehensive **process list** of the steps taken within the network project to run the testing method. Include screenshots to illustrate the process and ensure clarity for others attempting to replicate the test.

1. Login to ISP.
2. Run "config" to enter configuration mode.
3. Run "show protocols ospf interface eth0" to view the MD5 authentication settings for the interface. We can see that the key-id is 1 and md5-key is secretkey1



The screenshot shows a PuTTY terminal window with the title 'ISP - PuTTY'. The terminal displays the following command-line session:

```
ISP - PuTTY
=====
OSPF external routing table =====

vyos@vyos:~$ configure
[edit]
vyos@vyos# show os

    Configuration path: [os] is not valid

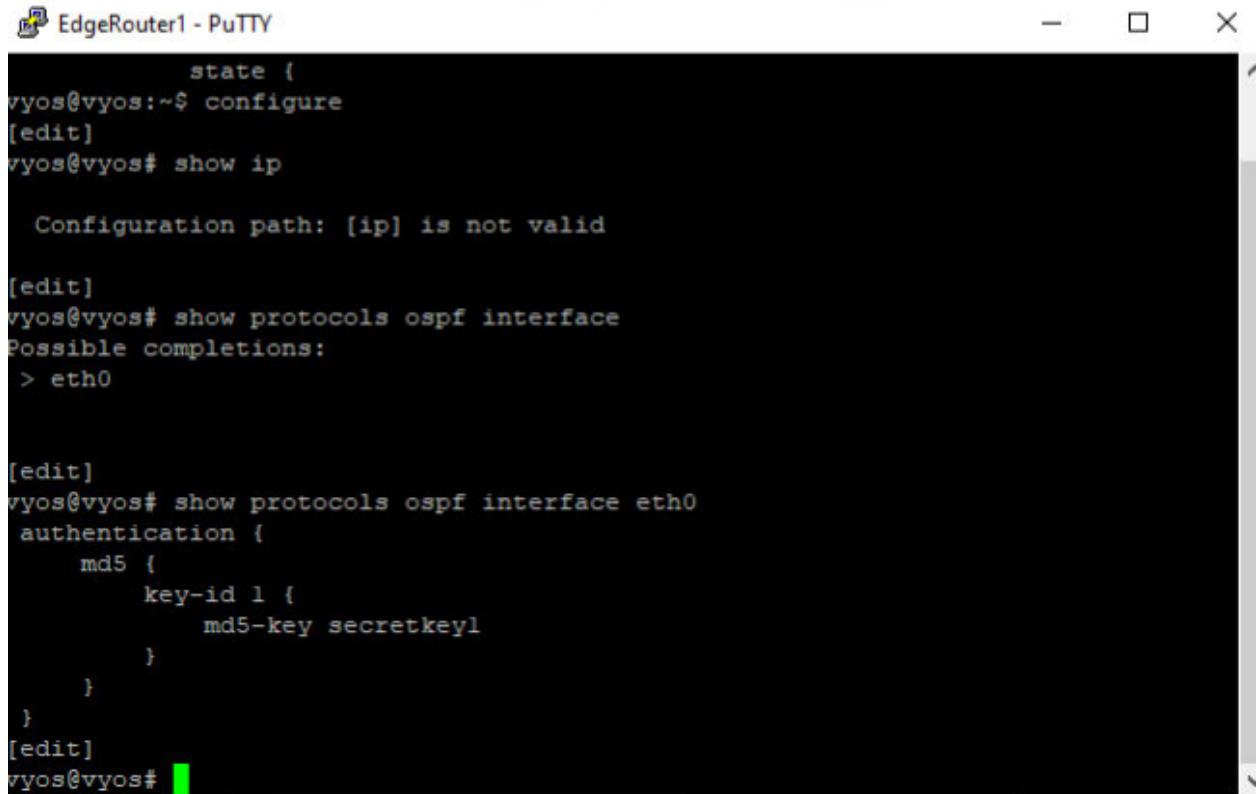
[edit]
vyos@vyos# show ospf

    Configuration path: [ospf] is not valid

[edit]
vyos@vyos# show protocols ospf interface eth0
authentication {
    md5 {
        key-id 1 {
            md5-key secretkey1
        }
    }
}
[edit]
vyos@vyos#
```

4. Login to EdgeRouter1.
5. Run "config" to enter configuration mode"
6. Run "show protocols ospf interface eth0" to view the MD5 authentication settings for the interface. We can see that the key-id is 1 and md5-key is secretkey1





```
EdgeRouter1 - PuTTY
state {
vyos@vyos:~$ configure
[edit]
vyos@vyos# show ip

Configuration path: [ip] is not valid

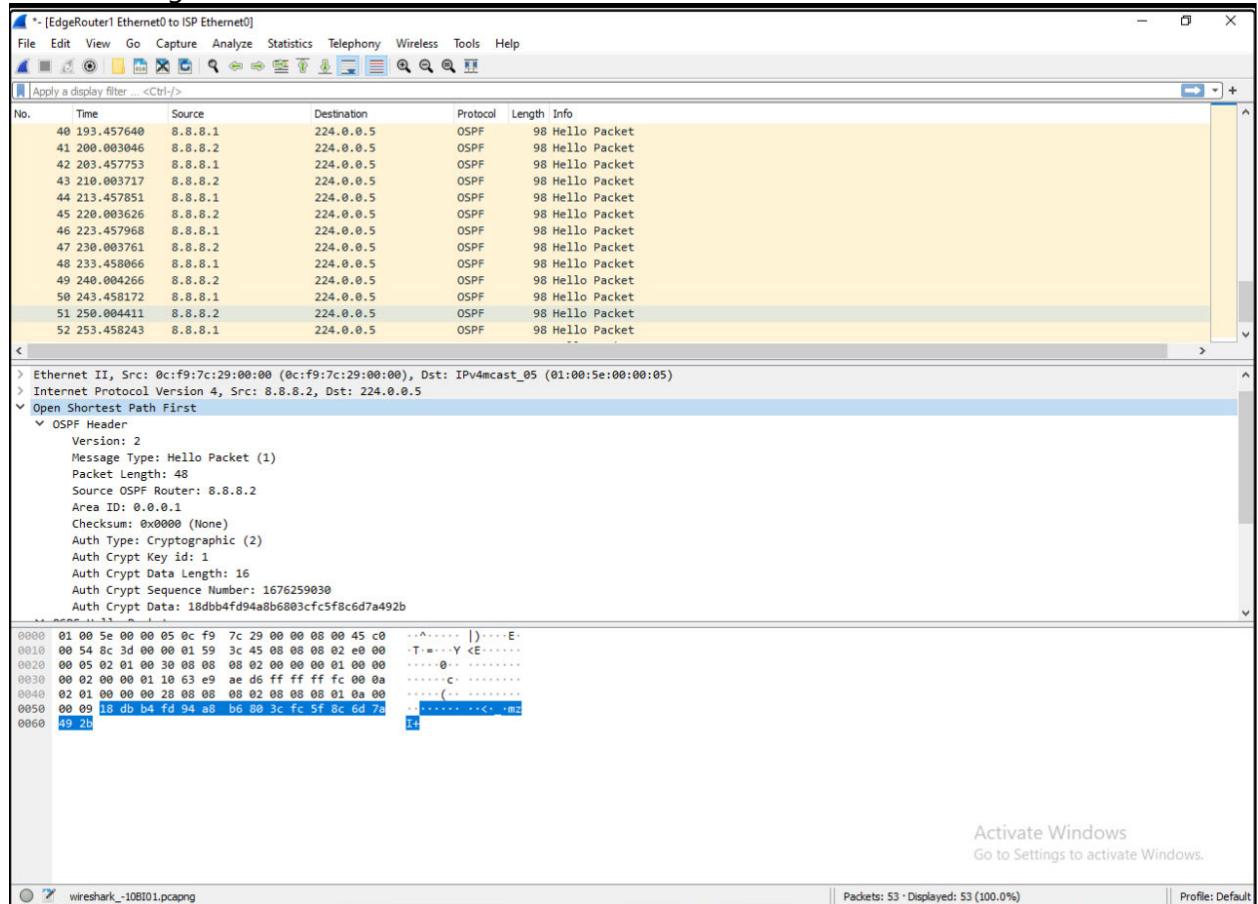
[edit]
vyos@vyos# show protocols ospf interface
Possible completions:
> eth0

[edit]
vyos@vyos# show protocols ospf interface eth0
authentication {
    md5 {
        key-id 1 {
            md5-key secretkey1
        }
    }
}
[edit]
vyos@vyos#
```

7. Take a Wireshark packet capture to inspect the OSPF packets being exchanged. As we can see in the packet, Auth Type field is Cryptographic, the Auth Crypt Key id field is set



to 1 as configured on each router.



Network Troubleshooting

Discuss how you analyzed the network to identify, troubleshoot, and resolve issues during development or when ensuring functionality of the test cases.

During the development of the network, there were various types of issues I encountered. Fortunately, GNS3 and the software on network devices and endpoints made troubleshooting possible. Sometimes restarting GNS3 and the VDI resolved issues as well. The most helpful troubleshooting method was inspecting packets through Wireshark. Wireshark allowed me to view both IP packet headers, and detailed information about most protocols. This was helpful in troubleshooting both network service, routing, and switching issues.

On network devices, I most commonly used traceroute, ping and "show" commands. I utilized traceroute to view that the appropriate path was being taken for IR routing, ping to verify basic connectivity, and "show" commands to view configurations on each network device. I also referenced the relevant network device documentation to gain an understanding of their features.



WESTERN GOVERNORS UNIVERSITY

On endpoints, I used several commands. On Ubuntu desktops, I used “dhclient” to release DHCP releases and get new ones. I also used “nmtui” to configure network interfaces on the host. On Windows, I used “ipconfig” to view IP address information, and manage host DHCP leases.



WESTERN GOVERNORS UNIVERSITY.

References

- Charles, K. (2018, July 22). *Warning banner sample for systems and network devices*. SecurityOrb.com | The Information Security, Digital Privacy & Internet Safety Website... Retrieved February 12, 2023, from <https://securityorb.com/general-security/warning-banner-sample-for-systems-and-network-devices/>



WESTERN GOVERNORS UNIVERSITY®