MINT 709 – Capstone Project Report

MINT Lab Automation

Submitted By:

Muhammad Idrees Afzal

(MINT 2019-2021)

Mentor:

Shahnawaz Mir

Supervisor:

Dr. Mike MacGregor

This Capstone Project report is submitted in partial fulfillment of the requirements for the degree of

Master of Science
in
Internetworking

University of Alberta

TABLE OF CONTENTS

1 EXECUTIVE SUMMARY

1.1 INTRODUCTION

The focus of this Capstone project is to evaluate and implement a solution to automate MINT-708 (Internet Laboratory) that was required to be delivered remotely for Fall-2020.

1.2 APPROACH AND SOLUTION COMPONENTS

Essential steps performed to evaluate, finalize, and implement a solution to enable remote delivery for MINT-708 are.

1. Standardized Physical Topology
2. Emulating Logical Lab Topologies
3. Align and Update Logical Lab topologies
4. Perform Labs to verify solutions
5. Re-write and update Lab Manuals
6. Replace Previous Pre-Lab Exercises with Pre-Lab Quizzes in e-class
7. Write Quizzes and Exams in e-class to support the Remote Delivery Model.

1.3 DELIVERABLES

Major Deliverables for this Capstone include

1. Physical and Logical Topology Diagrams
2. Configurations and Scripts to Emulate Logical Topology
3. Updated Lab Manuals
4. Solution Verification for Each Lab
5. Pre Lab Quizzes, Class Quizzes, and Exams in e-class

1.4 IMPLEMENTATION RESULTS

Automated labs designed and implemented; during this capstone project; ensured the successful completion of MINT-708 online delivery. Deliverables and artifacts have been uploaded in the allocated google drive, e-class portal, or on both places as relevant.

2 THE STANDARDIZED PHYSICAL TOPOLOGY

The first challenge was to finalize a physical topology that could support automated labs without requiring regular physical intervention. Finalization of technology for lab emulation was a prerequisite to conclude on physical topology.

2.1 PREVIOUS WORK

In the previous year's MINT-708 lab manual for Routing Information Protocol (RIP), the concept of 'router on the stick' and 'inter-VLAN routing' was implemented to simulate routing topology. The physical layer used a star topology was used, VLANs corresponding to each logical link were designed at Layer 2, and inter-VLAN routing emulated Layer 3 logical topology. Area 1 implementation in the OSPF lab followed the same concept. The following diagrams from the previous Lab Manual for RIP illustrate this approach.

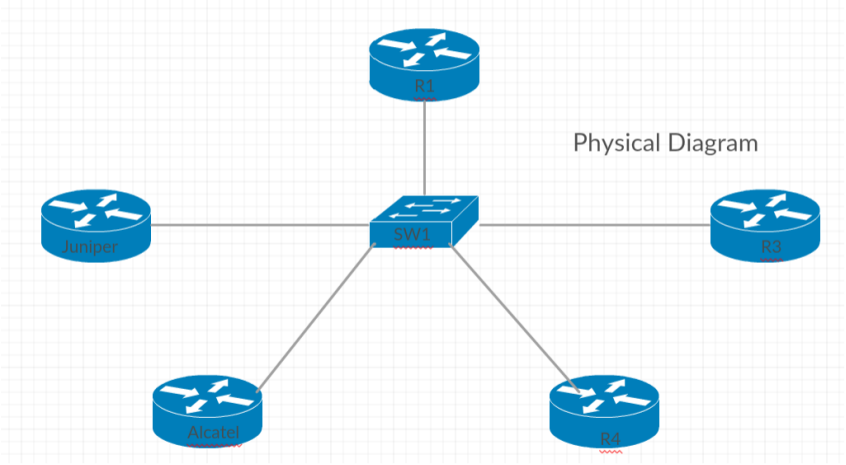


Figure 2.1 (MINT-708, UoA, Ch. 6: RIP, 2019)

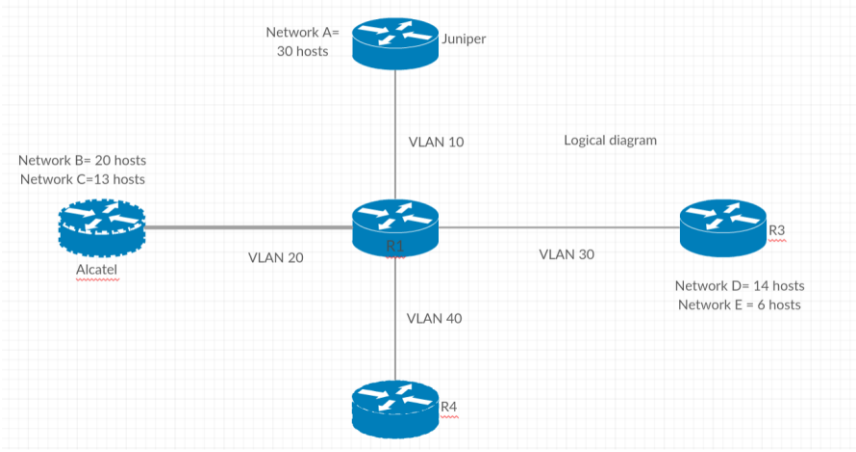


Figure 2.2 (MINT-708, UoA, Ch. 6: RIP, 2019)

2.2 FINALIZING PHYSICAL TOPOLOGY

The conclusion was to implement inter VLAN routing approach to automate all MINT-708 Labs. To deploy inter-VLAN technology for MINT labs has associated requirements to define VLAN structure for these labs. However, the first step was to finalize a standardized physical topology across all racks (or lab stations).

Standard physical topology was finalized possible logical topologies and available equipment in each rack, as shown below in Figure 2.3

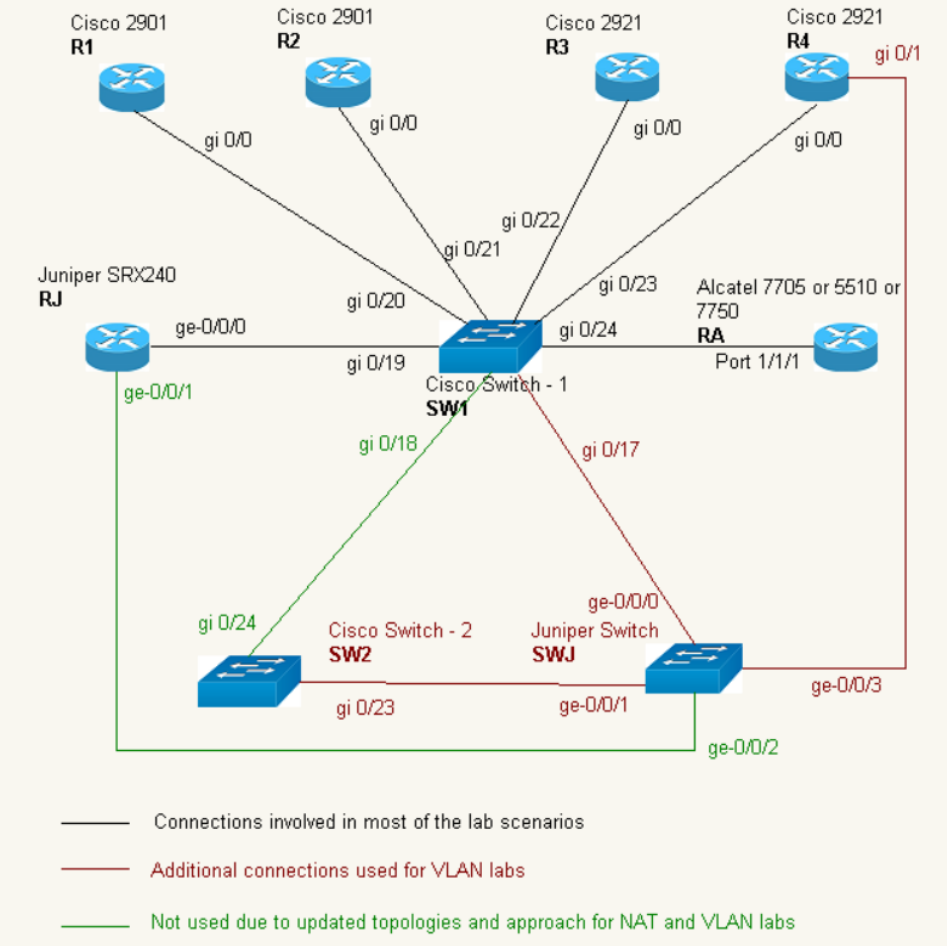


Figure 2.3 Standardized Physical Topology for MINT-708 Labs

The physical topology shown in figure 2.3 was standardized for each rack. Initially, prototype implementation used a limited number of racks. Physical devices on all the racks were inter-connected as per standardized topology before the commencement of Fall-2020 classes.

2.3 STANDARDIZING NAMING CONVENTIONS

The topology and the device names were standardized across all racks to reflect the same terminology and same physical device in all lab manuals and reports. The following table states the standardized naming convention.

Sr.	Device	Name
1	Cisco Switch	SW1
2	Cisco Switch	SW2
3	Cisco Router	R1
4	Cisco Router	R2
5	Cisco Router	R3
6	Cisco Router	R4
7	Juniper Router	RJ

Table 2.1 Standardized Naming Convention for each rack

3 LOGICAL TOPOLOGIES AND VLAN DESIGN

Based on each lab's logical topology, the underlying VLAN design was essential to set up labs for Static Routing, NAT, RIP, OSPF, IS-IS, and BGP. In addition to automation, basic lab topology and scenarios are updated for Static Routing, NAT, RIP, and OSPF and discussed in Chapter 7.

3.1 STATIC ROUTING LAB – LOGICAL TOPOLOGY AND VLANS

Logical topology was changed from the previous lab manual eliminating two Layer 3 switches and using loopbacks at routers. This approach enabled to set up lab scenario within the constraints of the standardized physical topology.

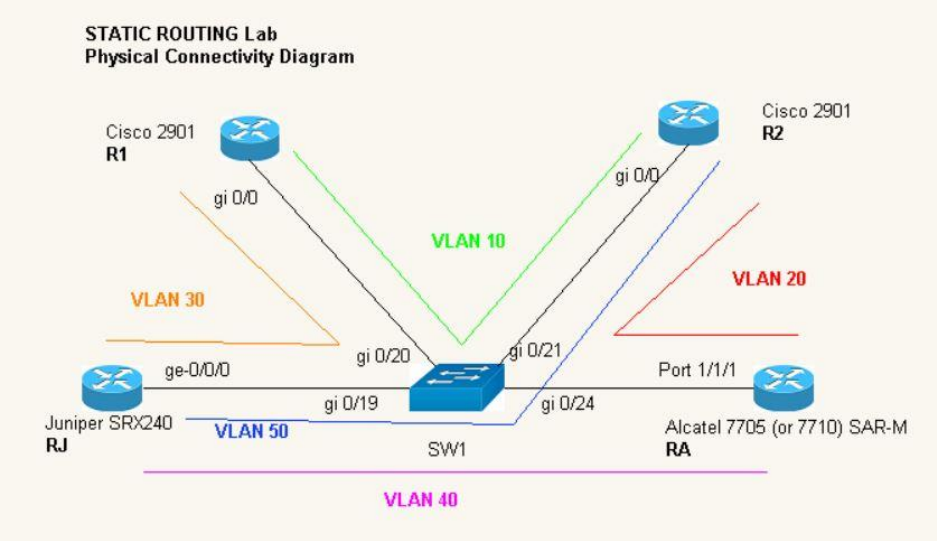


Figure 3.1-a VLAN Design for Automating Static Routing Lab

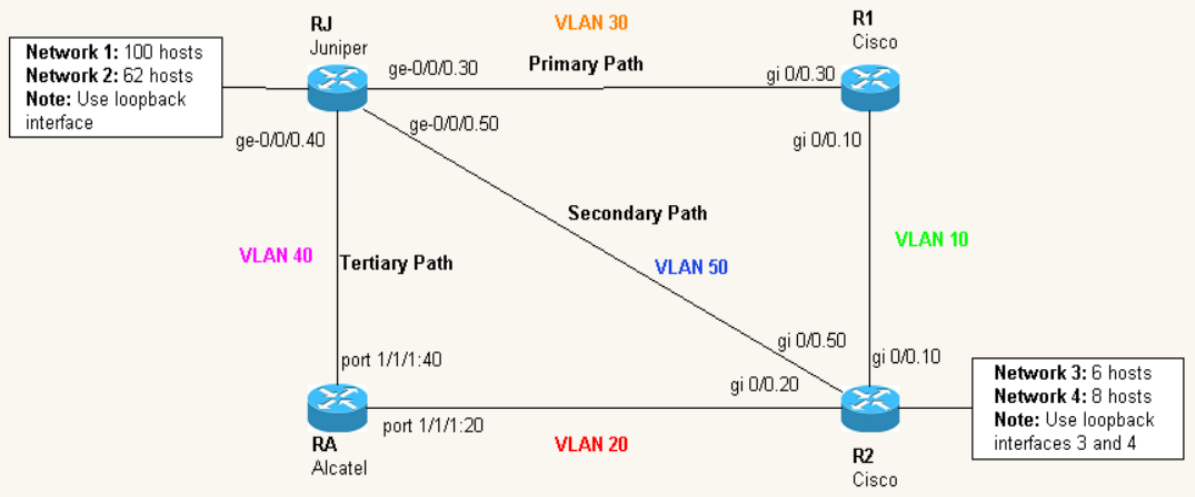


Figure 3.1-b [1]Logical Topology Simulated for Static Routing Lab

[1] Basic logical topology is taken from MINT Lab Manual from the year 2019 and aligned with the automated labs' physical topology constraints by eliminating two switches.

3.2 NAT/PAT LAB – LOGICAL TOPOLOGY AND VLANS

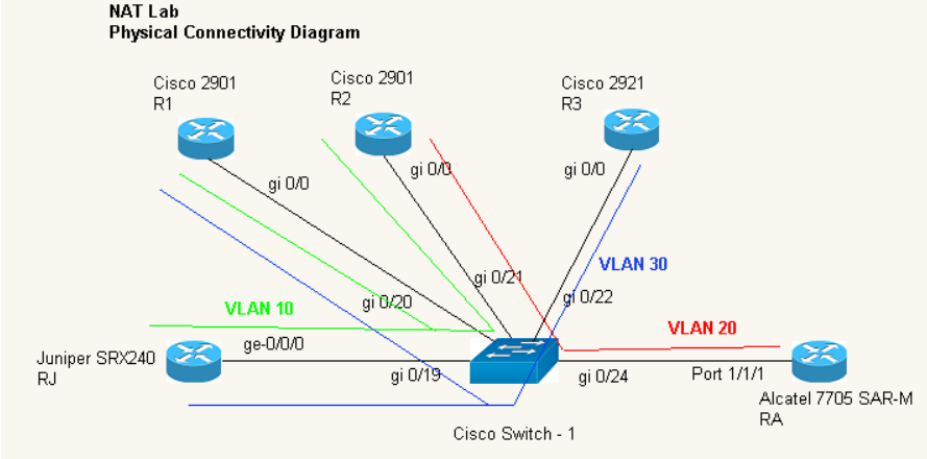


Figure 3.2-a VLAN Design for Automating NAT Lab

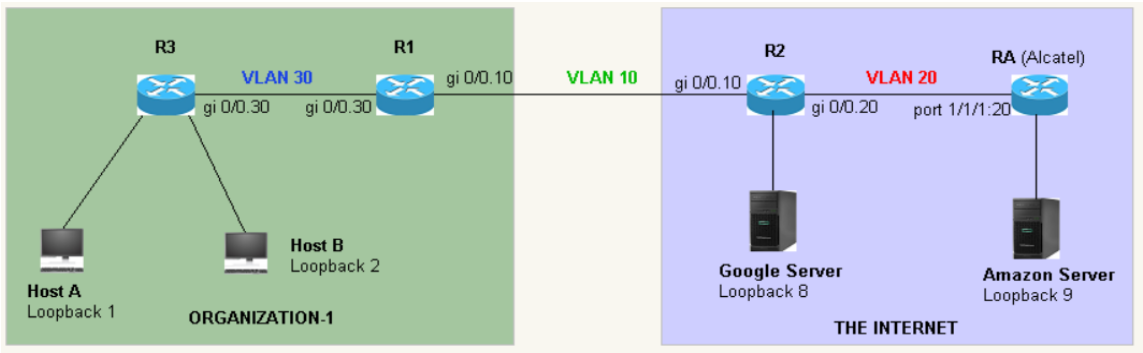


Figure 3.2-b [2]Logical Topology Simulated for Static NAT, Dynamic NAT, and PAT Lab

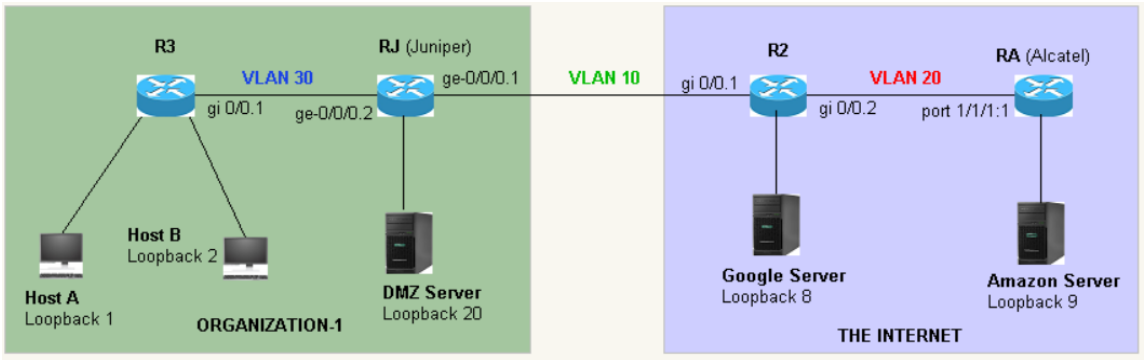


Figure 3.2-c [3]Logical Topology Simulated for Source and Destination NAT Lab

[2] Logical topology reused from 2019 MINT Lab Manuals and updated for Automated Labs.
[3] Logical topology reused from 2019 MINT Lab Manuals and updated for Automated Labs.

3.3 RIP LAB – LOGICAL TOPOLOGY AND VLANS

RIP Lab's scenario now has ring topology rather than star topology previously. The ring topology provides a mechanism for automatic failover demonstration by breaking any (logical) links.

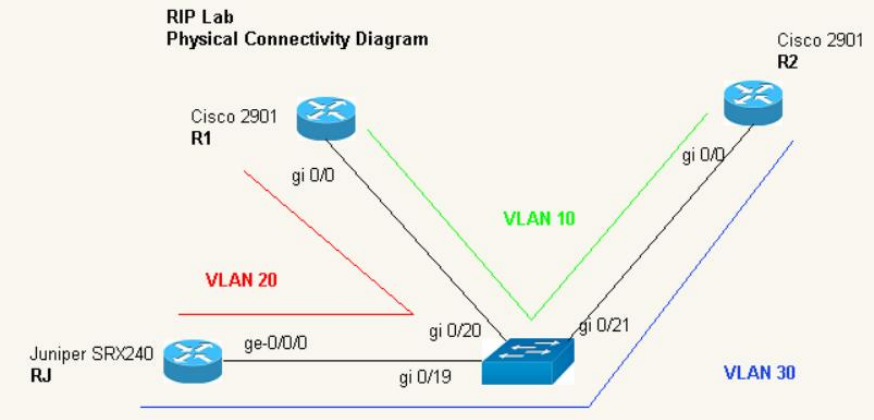


Figure 3.3-a VLAN Design for Automating RIP Lab

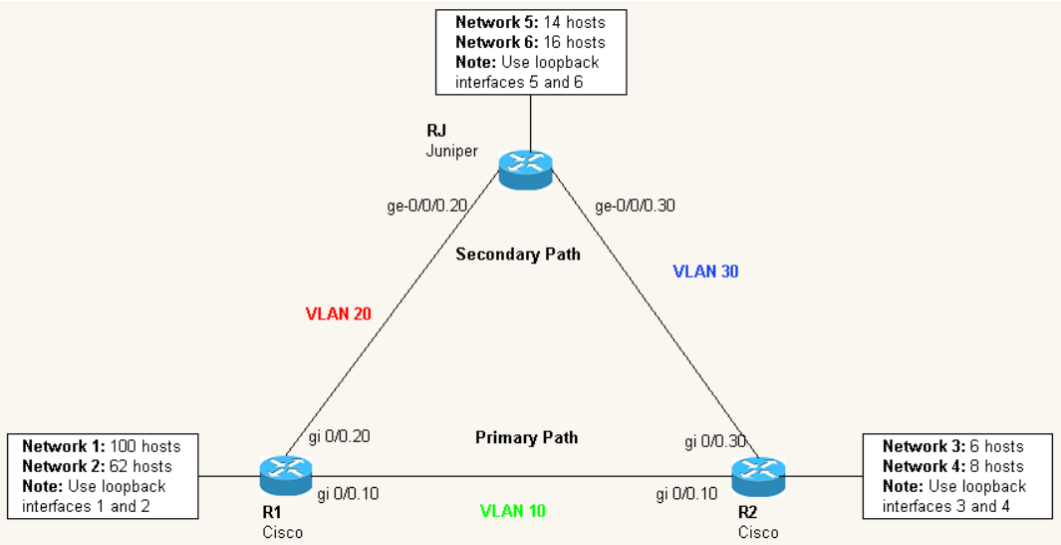


Figure 3.3-b Logical Topology Simulated for RIP Lab

3.4 OSPF LAB – LOGICAL TOPOLOGY AND VLANS

OSPF lab's scenario now has a ring topology in its Area 0, which is a change compared to previous year's scenarios. The new topology has facilitated introducing the option for redundancy in area 0 and the possibility of defining a preferred path in the backbone area. All of the other tasks in the previous topology, including a virtual link, nssa, summary routes, external route redistribution, are still part of the lab exercise and manual.

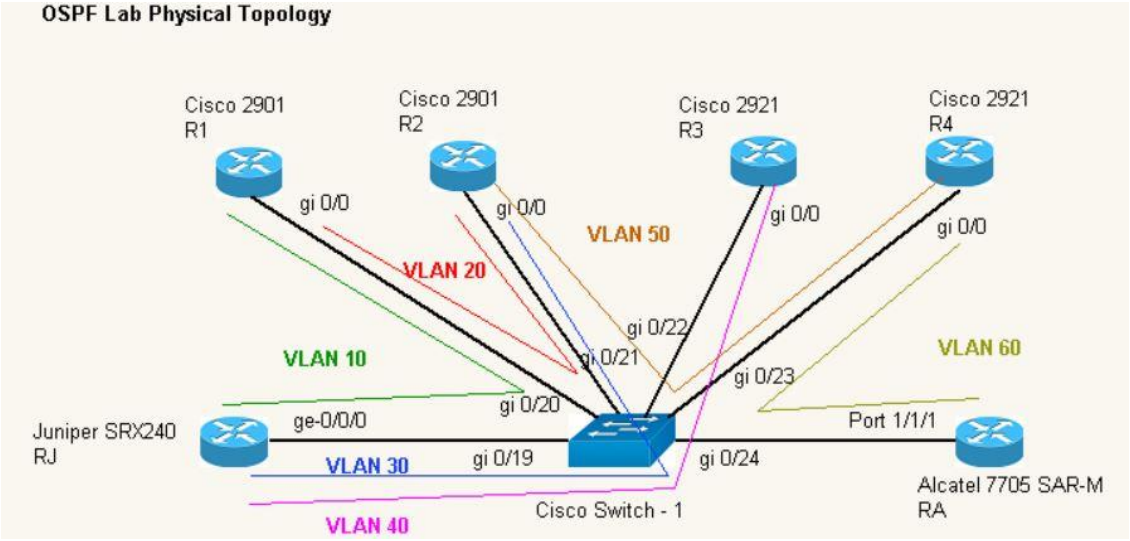


Figure 3.4-a VLAN Design for Automating OSPF Lab

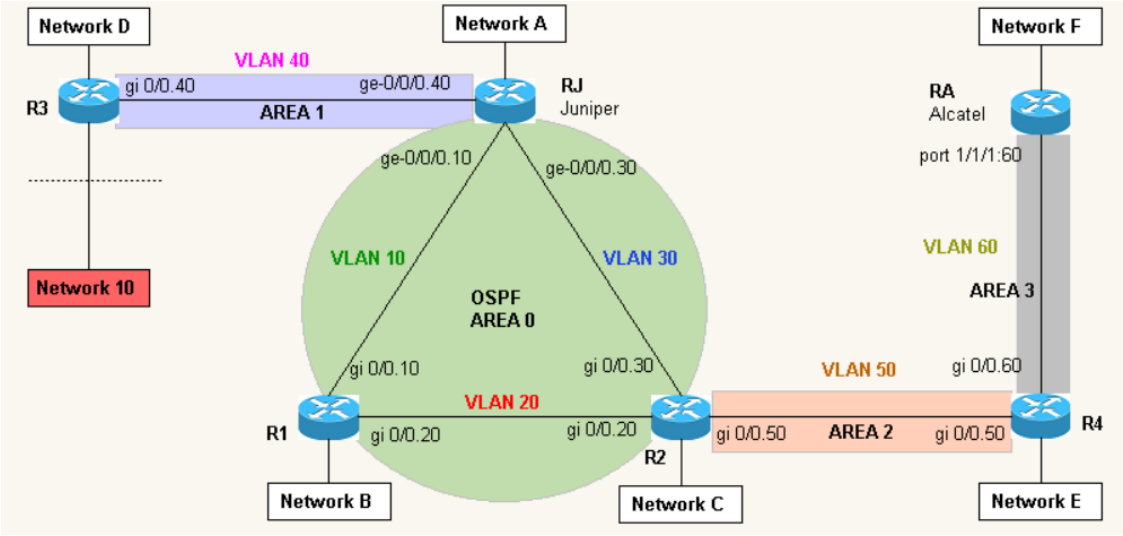


Figure 3.4-b Logical Topology Simulated for OSPF Lab.

3.5 ISIS LAB – LOGICAL TOPOLOGY AND VLANS

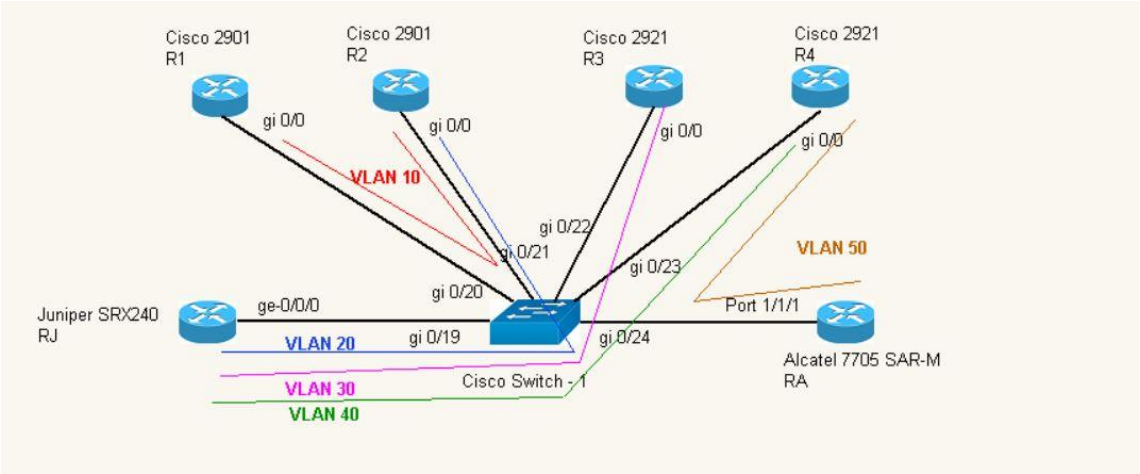


Figure 3.5-a VLAN Design for Automating ISIS Lab

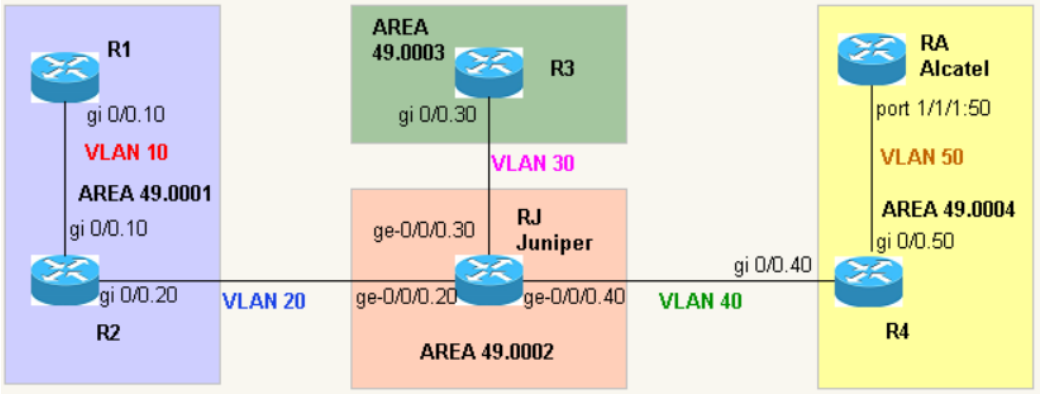


Figure 3.5-b [4]Logical Topology Simulated for ISIS Lab

[4] Logical topology reused from 2019 MINT Lab Manuals and updated for Automated Labs.

3.6 BGP LAB – LOGICAL TOPOLOGY AND VLANS

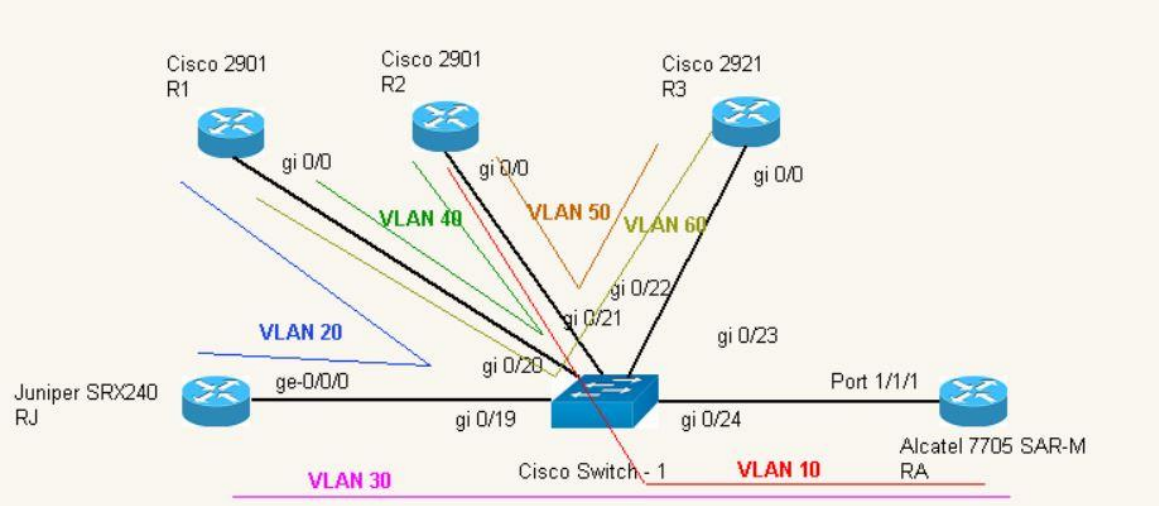


Figure 3.6-a VLAN Design for Automating BGP Lab

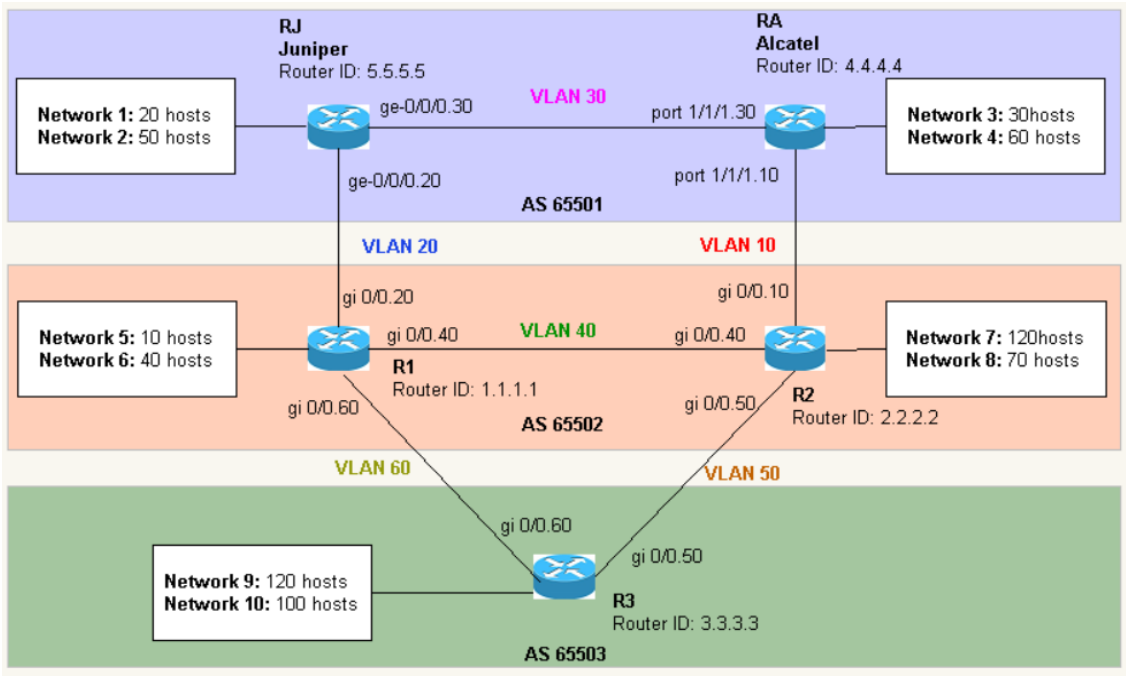


Figure 3.6-b [5]Logical Topology Simulated for BGP Lab

[5] Logical topology reused from 2019 MINT Lab Manuals and updated for Automated Labs.

3.7 VLAN LAB – TOPOLOGY

Though VLAN Lab is not an automated Lab, there is a significant topology change compared to the previous lab manual's topology. Topology is simplified now using one router on the stick, putting a clear emphasis on inter-switch 802.1Q trunk and inter VLAN routing. Cisco, Juniper, and Alcatel routers are used alternatively as a router on the stick. One of the challenges was to accommodate VLAN Lab topology within the standardized physical topology. This challenge was another contributor to initiate VLAN lab redesign.

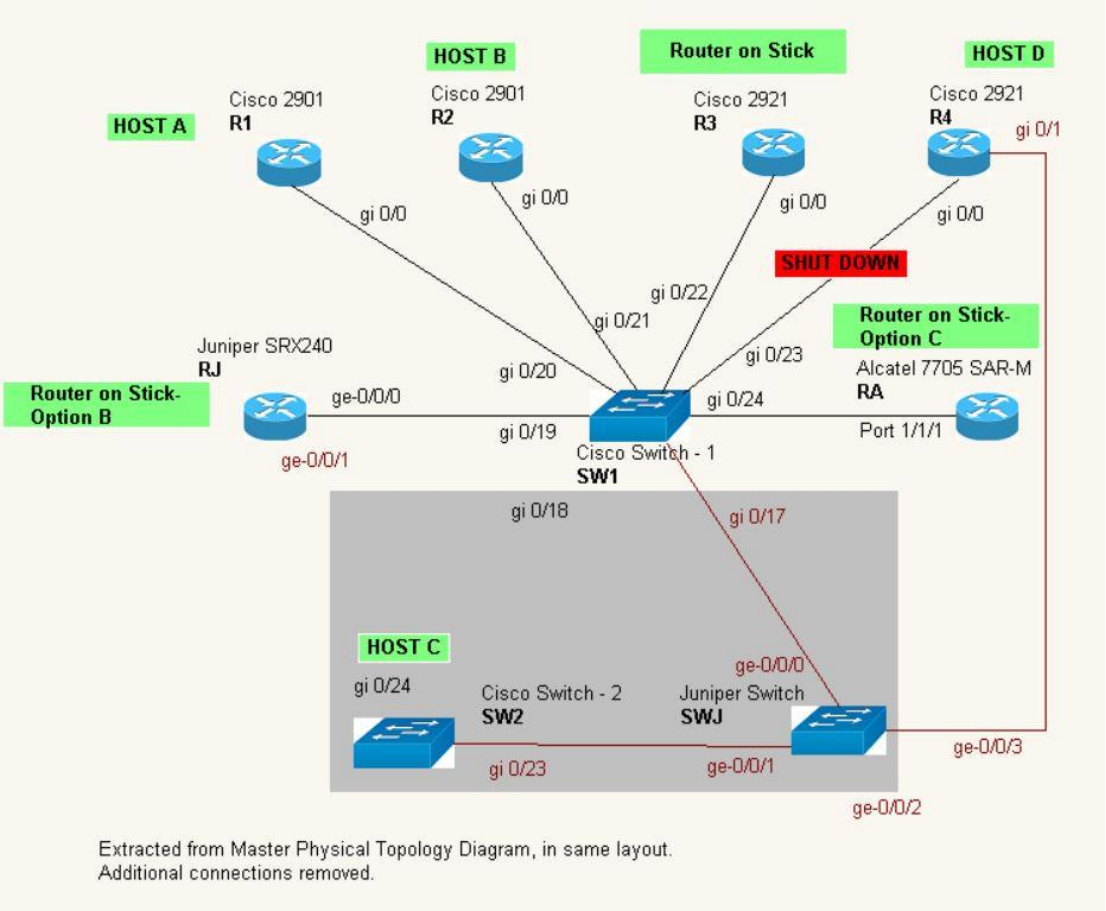


Figure 3.7-a Physical Topology Illustrated for VLAN Lab

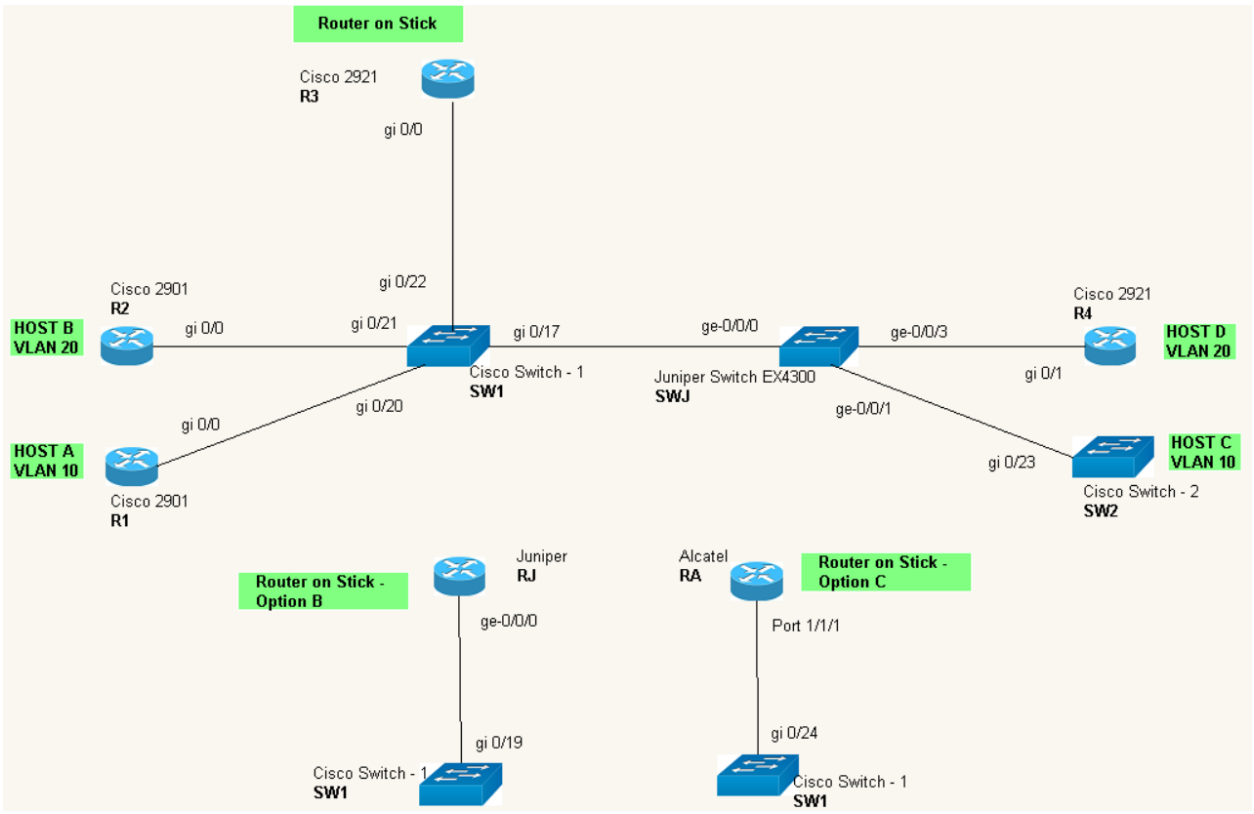


Figure 3.7-b Lab Diagram for VLAN Lab Manual (Re-drawn Figure 3.7-a)

4 SCRIPTS FOR SETTING UP LABS

The main requirement was to rollout configuration for Cisco switch-1 (SW-1) to create required VLANs and configure switch ports to simulate logical topology for each lab.

Application	
Presentation	
Session	
Transport	
Network	Simulated Logical Topology for Labs
Datalink	VLANs and Inter VLAN routing
Physical	Standardized Physical Topology and Cabling

Table 4.1 Automated Labs Implementation and OSI Model

4.1 SELECTING SCRIPTING LANGUAGE

The methodology selected is Python scripting to set up lab scenarios by configuring Cisco SW-1and later extended to remove these configurations. These Python scripts can be amended to push any other configurations on many (similar) devices and erase some configurations. The main reason to select Python is its increasing use in network automation and DevNet scenarios.

"Telnetlib" is the main Python library used to telnet the devices, send commands, and read outputs. (https://docs.python.org/3/library/telnetlib.html). Other study resources consulted are

1- Network Automation Tutorials by David Bombal (https://www.youtube.com/watch?v=s6SIVc7C5U0, n.d.)
2- Python Full Course for Beginners by Giraffe Academy (https://www.youtube.com/watch?v=rfscVS0vtbw, n.d.)
3- Python Documentation and Resources (https://www.python.org/, n.d.)

4.2 NETWORK AUTOMATION SERVER

A Network Automation Server executed these scripts during the test and implementation phases. The following diagram suggests the high-level placement of the network automation server in the lab network.

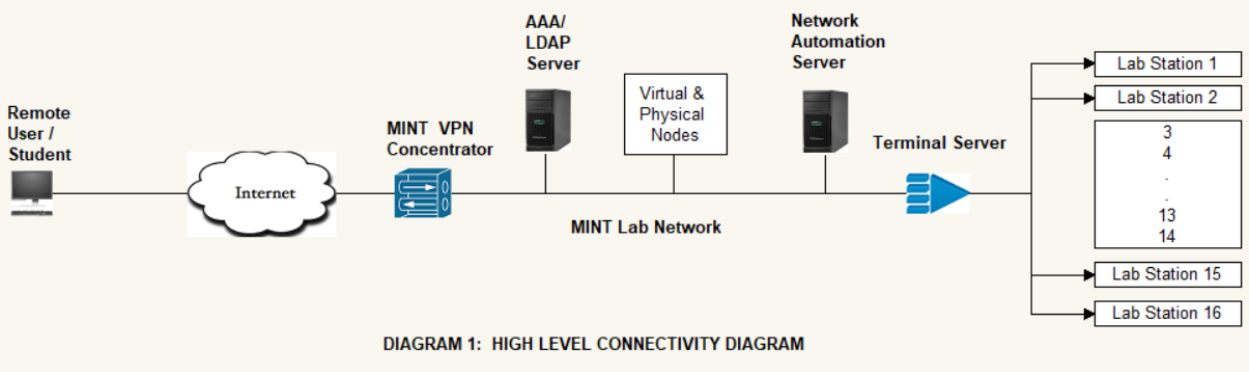


Figure 4.1 [6]High-Level Placement of Network Automation Server in Current Implementation

4.3 THE MECHANISM TO ROLLOUT ON MULTIPLE DEVICES

Each script developed parses IP address entries in file "switches", and for every entry, it repeats a set of actions using a loop. We have to ensure that the file "switches" have correct target IP addresses before running the Python script.

4.4 POSSIBLE ALTERNATE OPTIONS

Possible alternate options to this solution maybe.

1. Push configurations using TFTP server
2. Implement an Out-of-Band management network using dedicated ethernet ports.
3. Use other scripting languages, including Bash scripting.
4. Use traditional telnet scripts by copying the configuration from a text file and pasting it into the console.

4.5 IMPLEMENTATION RESULTS

Overall most of the rollouts through Python scripts worked well; however, the following list documents associated challenges during execution.

1. Console server allows one telnet session at a time; for the script to work smoothly, disconnect all target devices from the console. In case a device still has an active telnet connection in the console server (or someone connects while running script), the script cannot establish a telnet session and goes into a blocking state.
2. All target devices must support the same set of commands, same password where applicable, and in most cases, every device should be in the same mode. Scripts may not execute some or all commands on the specific devices that do not meet these prerequisites.

[6] Figure Reused from MINT Capstone Proposal v02 - mafzal1

3. The script may skip executing one or two commands intermittently and randomly on random device or devices. A general observation is that it happened with the scripts pushing a large number of commands. There is no evidence, but one of the hypotheses is that baud rate limitation on the console may have contributed to this situation and need further investigation and testing.

4.6 POSSIBLE FUTURE IMPROVEMENTS

Though not addressed as part of this Capstone Project scope; possible further improvement and next steps to enhance these developed scripts may:

1. Include error detection and exception handling in the script when a telnet session fails.
2. Include error detection and correction when the script cannot execute one or more commands on a device or a set of devices.
3. Include a mechanism to detect error due to the device being in another configuration mode and correct error by moving the device to the relevant mode and executing the command set again.
4. Consider further analysis on baud rate impact in the current deployment model and possible optimizations.
5. Evaluate out of band ethernet management option to push configuration scripts over ethernet.
6. GUI interface development for automated labs rollout.

4.7 IMMEDIATE BREAK-FIX APPROACH USED

Following break-fix steps and approaches were used to fix the issues faced as mentioned above.

1. Disconnect the previous session from the console and re-run the script in case of already established connections.
2. Password protected the Switches 'SW-1' for the duration of automated labs.
3. Use an ad-hoc script for missed commands due to manual error or malfunctioning. Use the 'adhoc.py' template for this purpose.
4. Use telnet script when running on a limited number of hosts.

5 SWITCH CONFIGURATION AND SCRIPT – OSPF EXAMPLE

5.1 SUMMARY OF UPDATED LAB MANUALS

Uploaded artifacts on google drive include all the scripts to set up labs by configuring Cisco Switch-1 and scripts to delete configurations. This section provides a reference setup for the OSPF lab.

5.2 SWITCH (SW-1) CONFIGURATION 7 -OSPF LAB

Below is the switch configuration (in the form of telnet script [8]) required to set up VLANs to simulate automated lab for OSPF

```
Conf t
hostname MINT708-Lab07-ospf
vlan 10
name VLAN-10
vlan 20
name VLAN-20
vlan 30
name VLAN-30
vlan 40
name VLAN-40
vlan 50
name VLAN-50
vlan 60
name VLAN-60

interface GigabitEthernet0/19
 description To-RJ-Juniper
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 10,30,40
 switchport mode trunk
 no shutdown

interface GigabitEthernet0/20
 description To-R1
 switchport trunk encapsulation dot1q
```

[7] Read this configuration along with Figure 3.4-a VLAN Design for Automating OSPF Lab.

[8] Telnet Script in its text form can be copied and pasted into the console to configure that device

```
 switchport trunk allowed vlan 10,20
 switchport mode trunk
 no shutdown

interface GigabitEthernet0/21
 description To-R2
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 20,30,50
 switchport mode trunk
 no shutdown

interface GigabitEthernet0/22
 description To-R3
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 40
 switchport mode trunk
 no shutdown

interface GigabitEthernet0/23
 description To-R4
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 50,60
 switchport mode trunk
 no shutdown

interface GigabitEthernet0/24
 description To-RA-Alcatel
 switchport trunk encapsulation dot1q
 switchport trunk allowed vlan 60
 switchport mode trunk
 no shutdown

exit
exit
wr mem
```

5.3 PYTHON SCRIPT TO SETUP OSPF LAB 9 -OSPF LAB

Below is a python script to roll out the 'SW-1' configuration to all the switches simultaneously. This script parses file "switches" containing the IP addresses of all target devices

```
import telnetlib
pass1 = 'DELETED'
pass2 = 'DELETED'
s = open ('switches')
for IP in s:
   IP = IP.strip()
   print("Now configuring switch "+ (IP))
   tn = telnetlib.Telnet(IP)
   tn.set_debuglevel(8)
   tn.read_until(b"Password: ")
   tn.write(pass1.encode('ascii') + b"\n")
   tn.write(b"\r")

   tn.write(b"enable\n")
   tn.read_until(b"Password: ")
   tn.write(pass2.encode('ascii') + b"\n")
   tn.write(b"\r")

   tn.write(b"conf t\n")
   tn.write(b"hostname MINT708-Lab07-ospf\n")

   tn.write(b"vlan 10\n")
   tn.write(b"name VLAN-10\n")
   tn.write(b"vlan 20\n")
   tn.write(b"name VLAN-20\n")
   tn.write(b"vlan 30\n")
   tn.write(b"name VLAN-30\n")
   tn.write(b"vlan 40\n")
   tn.write(b"name VLAN-40\n")
   tn.write(b"vlan 50\n")
   tn.write(b"name VLAN-50\n")
   tn.write(b"vlan 60\n")
   tn.write(b"name VLAN-60|n")

   tn.write(b"interface giga 0/19\n")
```

[9] This script will configure SW-1 on multiple racks according to Figure 3.4-a VLAN Design for Automating OSPF Lab.

```
tn.write(b"description To-RJ-Juniper\n")
tn.write(b"switchport trunk encapsulation dot1q\n")
tn.write(b"switchport trunk allowed vlan 10,30,40\n")
tn.write(b"switchport mode trunk\n")
tn.write(b"no shutdown\n")
tn.write(b"exit\n")

tn.write(b"interface giga 0/20\n")
tn.write(b"description To-R1\n")
tn.write(b"switchport trunk encapsulation dot1q\n")
tn.write(b"switchport trunk allowed vlan 10,20\n")
tn.write(b"switchport mode trunk\n")
tn.write(b"no shutdown\n")
tn.write(b"exit\n")

tn.write(b"interface giga 0/21\n")
tn.write(b"description To-R2\n")
tn.write(b"switchport trunk encapsulation dot1q\n")
tn.write(b"switchport trunk allowed vlan 20,30,50\n")
tn.write(b"switchport mode trunk\n")
tn.write(b"no shutdown\n")
tn.write(b"exit\n")

tn.write(b"interface giga 0/22\n")
tn.write(b"description To-R3\n")
tn.write(b"switchport trunk encapsulation dot1q\n")
tn.write(b"switchport trunk allowed vlan 40\n")
tn.write(b"switchport mode trunk\n")
tn.write(b"no shutdown\n")
tn.write(b"exit\n")

tn.write(b"interface giga 0/23\n")
tn.write(b"description To-R4\n")
tn.write(b"switchport trunk encapsulation dot1q\n")
tn.write(b"switchport trunk allowed vlan 50,60\n")
tn.write(b"switchport mode trunk\n")
tn.write(b"no shutdown\n")
tn.write(b"exit\n")
tn.write(b"interface giga 0/24\n")
tn.write(b"description To-RA-Alcatel\n")
tn.write(b"switchport trunk encapsulation dot1q\n")
tn.write(b"switchport trunk allowed vlan 60\n")
tn.write(b"switchport mode trunk\n")
```

```
tn.write(b"no shutdown\n")
tn.write(b"exit\n")

tn.write (b"system mtu routing 1500\n")
tn.write (b"system mtu jumpbo 1500\n")

tn.write(b"end\n")
tn.write(b"exit\n")
print(tn.read_some().decode('ascii'))
```

5.4 OTHER SCRIPTS

Script to remove the configuration given in section 5.2 is also developed and used; for all the automated labs.

Script to delete configuration from multiple devices (cisco routers and switches) is given below; this will parse file “switches” for IPs of target devices. This script also uses “time” library to introduce wait time while configuration is being deleted and reload prompt is awaited.

```
import telnetlib
import time
pass1 = 'DELETED'
s = open ('switches')
for IP in s:
   IP = IP.strip()
   print("Now erasing configuration and reloading "+ (IP))
   tn = telnetlib.Telnet(IP)
   tn.set_debuglevel(8)
   tn.read_until(b"Password: ")
   tn.write(pass1.encode('ascii') + b"\n")
   tn.write(b"\r")
   tn.write(b"enable\n")
   tn.write(b"erase nvram:\n")
   time.sleep(1)
   tn.write(b"yes\n")
   tn.write(b"reload\n")
   time.sleep(1)
   tn.write(b"\n")
   tn.write(b"\n")
   tn.close()
   print(tn.read_some().decode('ascii'))
```

Update the above scripts to push any configuration or commands to any number of devices.

6 EXECUTING PYTHON NETWORK AUTOMATION SCRIPTS

This chapter outlines the process to execute Python network automation scripts and setup MINT labs. The process can push any configuration to multiple devices with some modifications. For MINT-708 automated lab[10] scripts, targeted devices are SW-1 switches in multiple racks;

6.1 NETWORK AUTOMATION SERVER SETUP

A network automation server (preferably with Linux OS) will host and execute these python scripts. Following are the prerequisites of the network automation server:

1. Should have Python 3.x installed in it.
2. All targeted devices must be accessible from the network automation server on telnet protocol.

6.2 DEFINING TARGET DEVICES

For target devices, each script parses file 'switches' on the network automation server. Follow the steps below to define and update this file.

1. Using nano editor, create a file 'switches' in the same directory where automation scripts are stored.
2. In this file ('switches'), write IP addresses of all targeted devices, one IP address per line. Remove any blank lines, specifically checking at the end of the file.
3. File 'switches' may need an update before executing every script based on targeted devices.
4. The name and location of the file ('switches') may change; however, edit the script to reflect the new file name and its path as required.

6.3 EXECUTING THE SCRIPT

Once targeted devices are defined, run the script using standard Python 3.x syntax. For example, to execute OSPF script 'lab07-ospf.py'; use syntax' python3 lab07-ospf.py'.

6.4 PRE REQUISITES TO EXECUTE THE SCRIPT

Ensure following prerequisites requirements before executing the script.

1. Targeted devices must permit telnet sessions for the script. In the current MINT lab setup, the console server must not have an active session for the targeted device.
2. 'Console password' of target devices must be correctly defined in the script. All target devices must share the same 'console password'.

[10] Read these scripts in conjunction with VLAN design, physical diagram, and logical topology for the specific lab.

3. All target devices must be in user mode.
4. Define 'enable password' of targeted devices correctly in the script, where applicable, to avoid blocking state. All targeted devices must share the same 'enable password'. Scripts to setup Static, NAT, RIP, OSPF, ISIS, BGP use predefined enable password.

Note the configuration steps to follow for manual configuration through telnet. The script performs the same steps similarly in the same predefined sequence and requires the same environment to connect and execute.

6.5 UNDERSTANDING NETWORK AUTOMATION SCRIPTS

These scripts use a loop, and each iteration of the loop corresponds to one targeted device. Every loop of these scripts mainly comprises three parts.

1. The first part establishes a telnet socket connection and exchanges authentication parameters until it accesses the device in its 'privileged exec mode'.
2. The second part sends standard configuration commands to the targeted device. Here, edit the script to send any legitimate command(s) to targeted devices if required.
3. 3rd part gracefully closes the session without entering into a blocking state.

6.6 EMBEDDED PASSWORDS

In these network automation scripts, 'console password' has been defined as variable 'pass1'; and 'enable password' as variable 'pass2'. Set up all targeted devices with passwords already mentioned in the scripts or update them to reflect the relevant passwords. If no 'enable password' is used, comment on the two lines after sending the 'enable' command. Use sign # to comment on these lines.

Following code lines illustrate embedded passwords and their usage. Relevant lines are displayed as bold text.

```
pass1 = 'console-password'
pass2 = 'enable-password'
# edit above lines to reflect actual console and enable password
s = open ('switches')
for IP in s:
   IP = IP.strip()
   print("Now configuring switch "+ (IP))
   tn = telnetlib.Telnet(IP)
   tn.set_debuglevel(8)
   tn.read_until(b"Password: ")
   tn.write(pass1.encode('ascii') + b"\n")
   tn.write(b"\r")
```

```
    tn.write(b"enable\n")
    tn.read_until(b"Password: ")
    tn.write(pass2.encode('ascii') + b"\n")
# If no 'enable password' is used, comment above two lines by adding sign # at the beginning
    tn.write(b"\r")
```

6.7 EXECUTING SCRIPTS FOR END-TO-END COURSE DELIVERY

Based on Fall-2020 course delivery, below is the recommended sequence of lab setup activities throughout the course.

1. Cable the racks as per the physical standard connectivity diagram. Having all ports enabled on connected devices, the command 'show interfaces status' can be used on 'SW-1' to verify connectivity.
2. Erase configuration of all cisco devices using script 'reset-to-default.py'. Add IP addresses of all required Cisco routers and switches to the file 'switches'.
3. Once students complete VLAN Lab, set up automated labs for the rest of the course. Before the beginning of automated labs, restore 'SW-1' to default and set 'enable password' as described in the next two steps.
4. Erase configuration of 'SW-1' switches using script 'reset-to-default.py'. This script will delete the configuration of 'SW-1' switches by erasing 'NVRAM' and reload them. At this stage, manually login to the switch and exit the configuration dialog.
5. Edit and use 'adhoc.py' to set enable secret for all 'SW-1' switches. Use the following lines in the script after 'enable' to set enable password to 'IDREES.'

```
tn.write(b"service password-encryption\n")
tn.write(b"enable secret IDREES\n")
tn.write(b"exit\n")
tn.write(b"wr mem\n")
tn.write(b"exit\n")
```

6. Use the relevant script to set up every lab, e.g., use 'lab07-ospf.py' to set up OSPF lab.
7. Remove configuration after every lab using corresponding remove script, e.g. 'remove-lab07-ospf.py' for OSPF lab. The script to remove the lab may not be a mandatory step because the new interface configuration will override the previous configuration at the next lab setup.
8. Edit and use 'adhoc.py' to configure 'system mtu routing 1500' for Alcatel MTU alignment during OSPF lab.
9. Edit and use 'adhoc.py' when the script misses and does not execute some command due to any reason.

10. Telnet scripts are also available on google drive for use in a specific scenario, e.g., to set up any lab on single 'SW-1' instead of editing file 'switches'.

6.8 TROUBLESHOOTING TIPS

1. All Python scripts are configured with debug level 8, and debug messages on the network automation server will provide information that will help in troubleshooting.
2. Verify all prerequisites.
3. Blank lines in file 'switches' specifically at the end will return an error message.
4. Edit and use 'adhoc.py' to push any partial or missed configuration.
5. Consult Telnet scripts as reference checks or even as a manual fallback option.

7 UPDATED LAB MANUALS

7.1 SUMMARY OF UPDATED LAB MANUALS

All the above changes required that lab manuals should be. This section summarizes the changes made in updated lab manuals. In most of the cases, there are updates in IP addressing schemes as compared to last year manuals (MINT-708 Lab Manuals, 2019)

	Automation	Topology Changes	Comments
Device Access and Basic Configuration	No	Topology Reused.	Password Recovery Exercise Removed (Tutorial & short questions only)
IP Addressing	NA	NA	New Assignment
VLANs	No	New Topology	
Static Routing	Yes	Changes made to the previous topology	The core topology part remains the same but with a reduced number of devices.
NAT	Yes	Topology Reused from the previous manual.	
RIP	Yes	New Topology	
OSPF	Yes	New Topology	
ISIS	Yes	Topology Reused from the previous manual.	
BGP	Yes	Topology Reused from the previous manual.	

Table 7.1 Summary View of Updated Lab Manuals

7.2 PRE-LAB QUIZZES [11]

Pre Lab exercises have been replaced with Pre-lab quizzes in e-class for every lab. Following are the key highlights regarding pre-lab quizzes:

1- Descriptive questions are replaced by questioned automatically marked by the system providing immediate feedback to the students.
2- Change of question types also required to write feedback information for post-attempt review.

[11] Several questions for pre-lab quizzes have been reused or rephrased in addition to new questions created.

3- Several non-descriptive questions for pre-lab quizzes are reused or rephrased from previous manuals, in addition to newly created questions.

Below are some examples of Pre-Lab questions having review feedback.

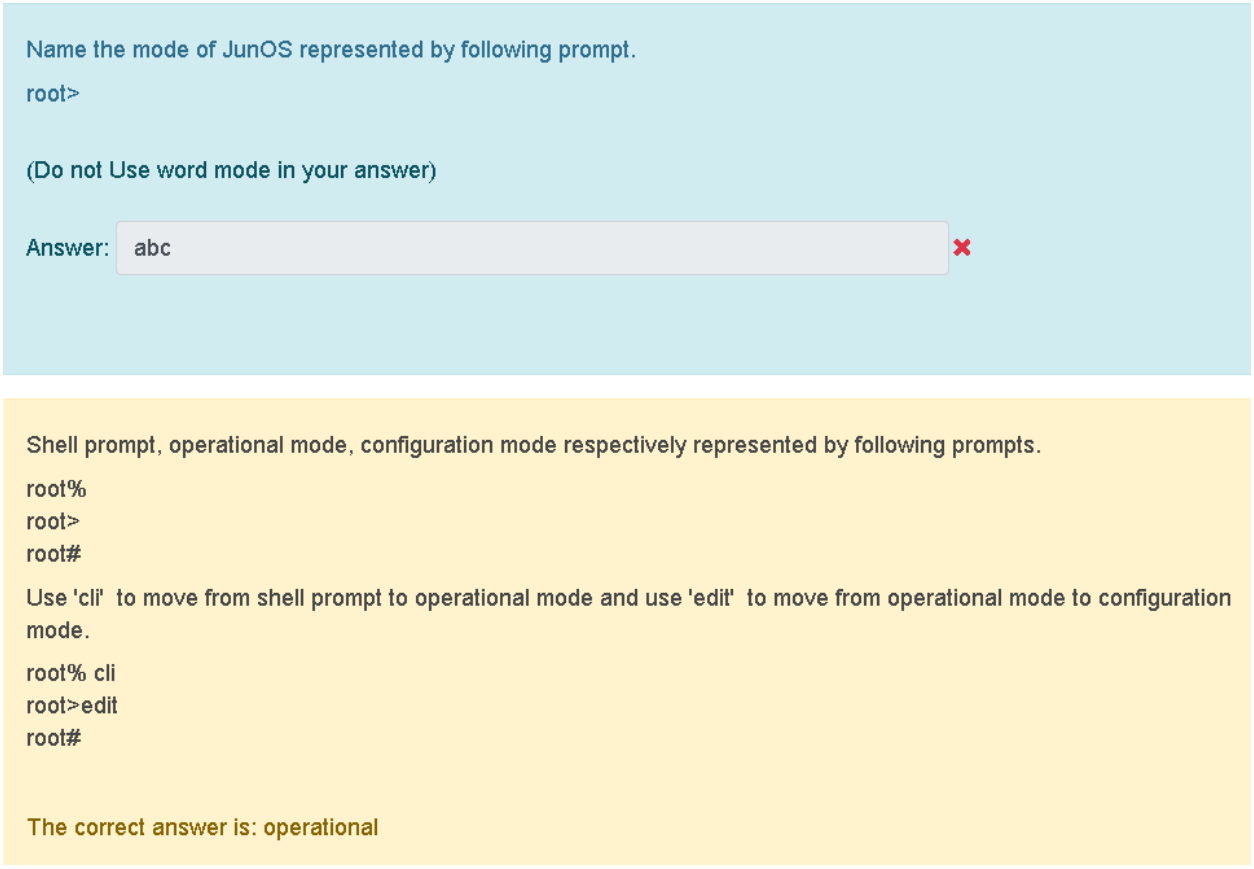


Figure 7.1 Example Pre Lab1' Short Answer' question with review feedback

Figure 7.2 Example Pre Lab 9; drag and drop' question with review feedback

7.3 LAB MANUALS

Appendix-1 includes full and latest versions (A1 Rev02) of lab manuals for automated labs. Shared google drive has these manuals and an amended version (A3 or Fall 2020) in e-class and shared google drive.

Writing this manual was a 3 step approach.

1. Writing Draft Manuals.
2. Updating Lab Manuals. (Rev 01) after performing lab before the lecture.
3. Updating Lab Manuals. (Rev 02) based on all feedback.

Following Documentations and Resources were consulted during the writing and updating of the manuals, performing labs, and taking outputs.

1. Previous Year Lab Manuals (MINT-708 Lab Manuals, 2019)
2. Cisco Systems Technical Documentation (https://www.cisco.com/, n.d.)
3. Juniper Networks TechLibrary (https://www.juniper.net/documentation/, n.d.)
4. Nokia Technical Documentation (https://documentation.nokia.com/cgi-bin/doc_list.pl, n.d.)

8 SOLUTION DOCUMENTS AND CONFIGURATIONS

Before every lecture, relevant lab execution ensured the operationality of the scenario. A collection of artifacts from this execution includes device configurations, required outputs, updated lab manual, and solution document. For BGP and NAT labs, the Fall-2020 solution document constructs on version (A3 ver 01), whereas full versions for all other labs. Shared google drive has all the solution documents (or artifacts) and configurations.

8.1 OSPF SOLUTION DOCUMENT – EXEMPLAR OVERVIEW

This section only presents the first two pages of the OSPF solution document; however, shared google drive has a complete set of solution documents.

(On the Next Page)

UNIVERSITY OF ALBERTA

MINT-708 INTERNET LABORATORY – FALL 2020

MASTER OF SCIENCE IN INTERNETWORKING

Module 7 OSPF
SOLUTION DOCUMENT

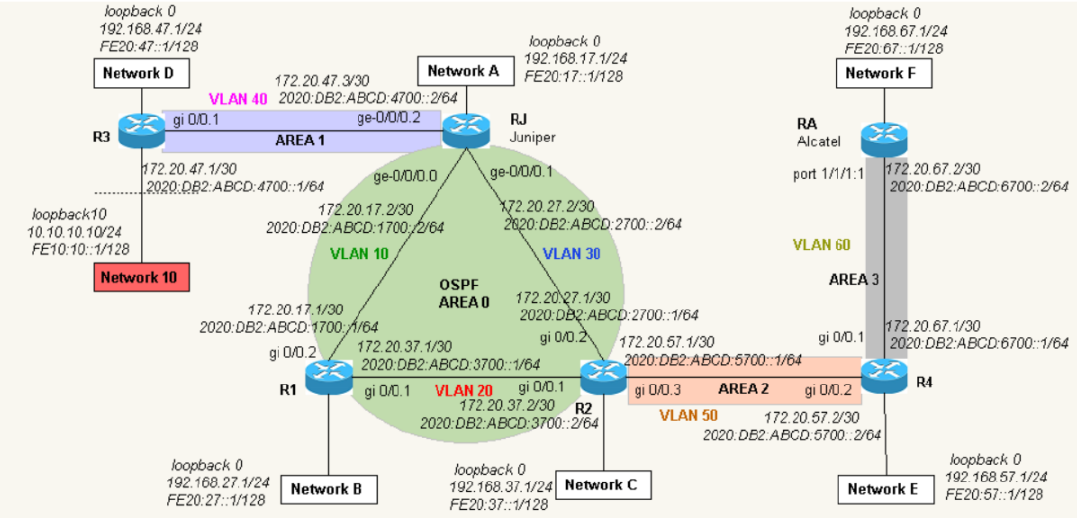
Performed and documented by: Muhammad Idrees Afzal (MINT-2019)

Section 1: Network Diagram, IP Addressing and Configurations:

Network Diagram:

Draw Network Diagram specifying interfaces and IP Addresses. Mention the rack ID which was used to perform this exercise.

Rack ID: 07



Configurations:

Refer to configuration files **'device name – OSPF.txt'** uploaded on share folder

Note : SW-1 file is for reference only and not part of students deliverables

Section 2: Device Outputs – OSPF routing

1. Output of ***"show ip route ospf"*** *and* ***"show ipv6 route ospf"*** on all Cisco Routers '**R1**, **R2, R3, R4**'

```
R1-Rack07#show ip route ospf
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override

Gateway of last resort is not set

      10.0.0.0/24 is subnetted, 1 subnets
O E1     10.10.10.0 [110/23] via 172.20.17.2, 06:05:35, GigabitEthernet0/0.2
      172.20.0.0/16 is variably subnetted, 8 subnets, 2 masks
O        172.20.27.0/30
           [110/2] via 172.20.37.2, 03:29:00, GigabitEthernet0/0.1
O IA     172.20.47.0/30
           [110/2] via 172.20.17.2, 18:38:37, GigabitEthernet0/0.2
O IA     172.20.57.0/30
           [110/2] via 172.20.37.2, 03:29:00, GigabitEthernet0/0.1
O IA     172.20.67.0/30
           [110/3] via 172.20.37.2, 03:28:55, GigabitEthernet0/0.1
      192.168.17.0/24 is variably subnetted, 2 subnets, 2 masks
O        192.168.17.0/24
           [110/1] via 172.20.17.2, 18:38:37, GigabitEthernet0/0.2
O        192.168.17.1/32
           [110/1] via 172.20.17.2, 18:38:37, GigabitEthernet0/0.2
      192.168.37.0/32 is subnetted, 1 subnets
O        192.168.37.1 [110/2] via 172.20.37.2, 03:29:00, GigabitEthernet0/0.1
      192.168.47.0/32 is subnetted, 1 subnets
O IA     192.168.47.1 [110/3] via 172.20.17.2, 18:26:29, GigabitEthernet0/0.2
      192.168.57.0/32 is subnetted, 1 subnets
O IA     192.168.57.1 [110/3] via 172.20.37.2, 03:28:55, GigabitEthernet0/0.1
O IA  192.168.67.0/24 [110/3] via 172.20.37.2, 03:28:55, GigabitEthernet0/0.1
R1-Rack07#
```

9 QUIZZES AND EXAM QUESTIONS IN E-CLASS

Remote delivery also involves remote and online assessments, including class quizzes and exams. The related component was to develop questions in e-class to facilitate online assessment,

Three class quizzes, each having ten questions, including MCQs, short answers, matching questions, drag and drop, and descriptive questions, were developed in e-class. The final exam consists of thirty questions of multiple types created for e-class considering immediate and automated assessment. Below are some examples snapped from e-class for reference

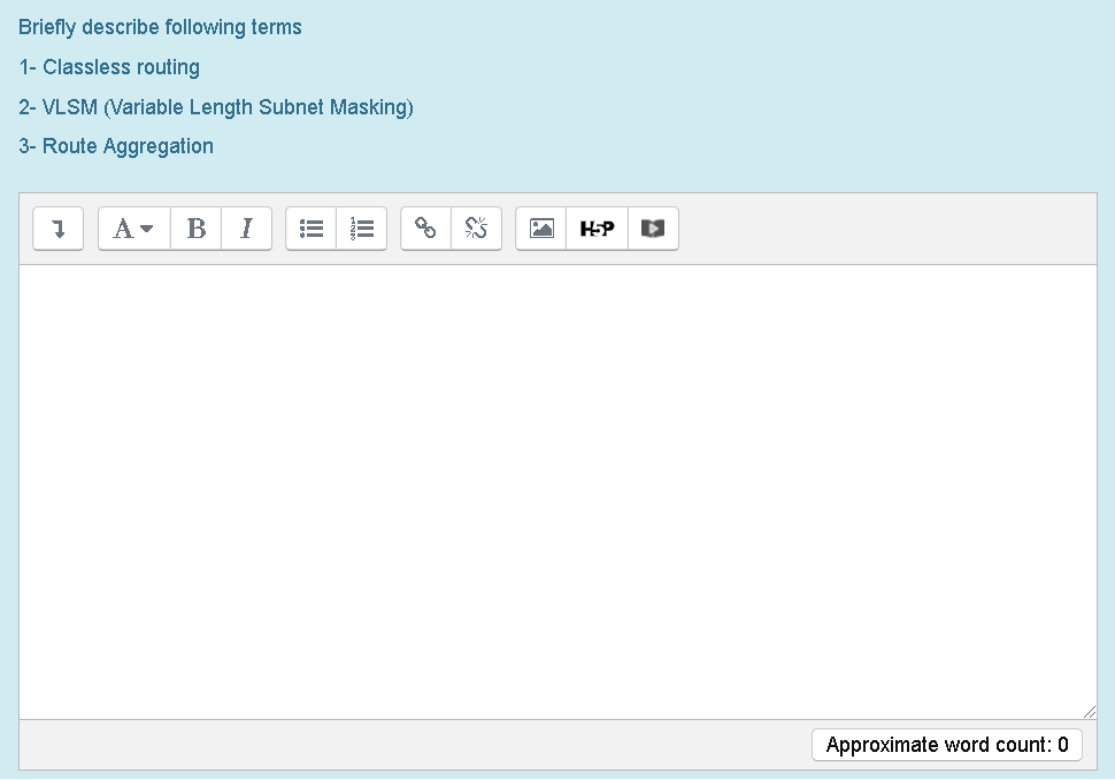


Figure 9.1 Example Descriptive Question – Quiz 1

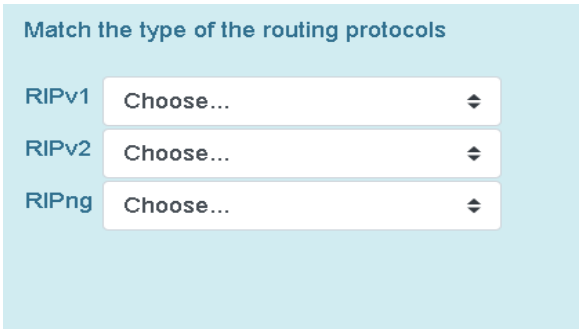


Figure 9.2 Example Matching Question – Quiz 2

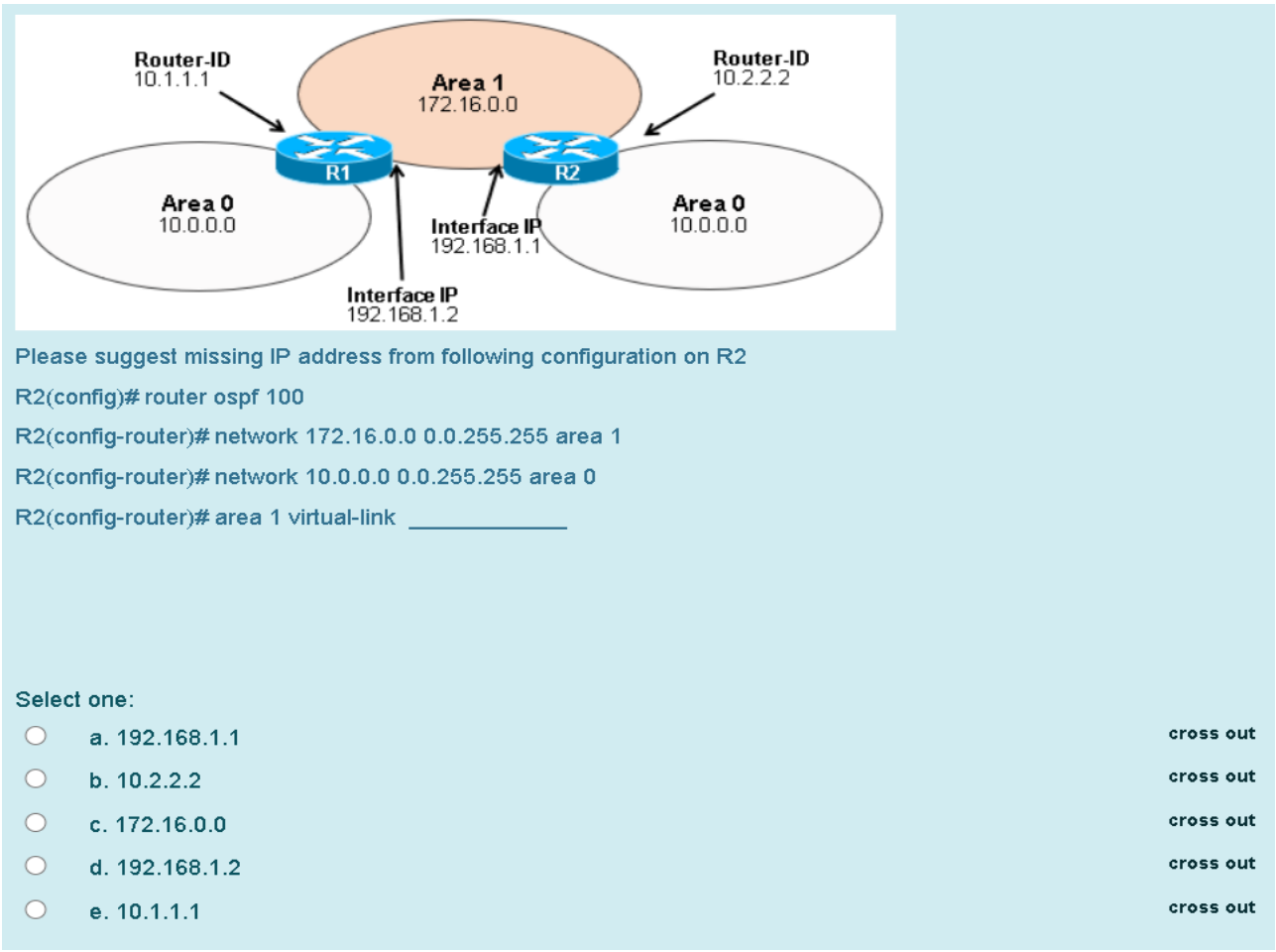
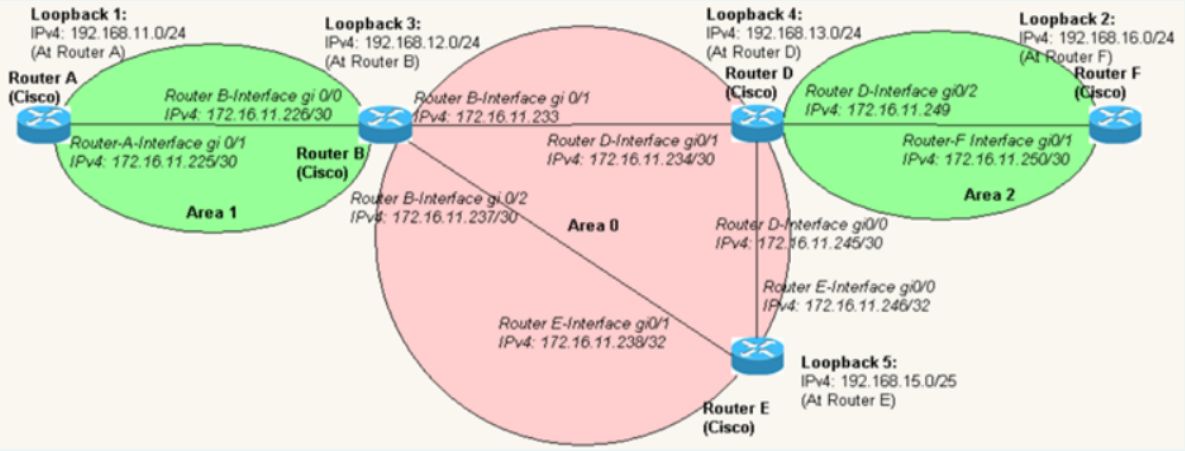


Figure 9.3 Example Single Answer MCQ Question – Quiz 3



For above OSPF network, which statements are true about router B partial configuration given below

Router-B(config)#router ospf 1

Router-B(config-router)#network 192.168.12.0 0.0.0.3 area 0

Router-B(config-router)#network 172.16.11.224 0.0.0.3 area 1

Router-B(config-router)#network 172.16.11.232 0.0.0.3 area 1

Router-B(config-router)#network 172.16.11.236 0.0.0.3 area 0

NOTE: Choose 3 correct answers. Selection of wrong option will have negative marks

Select all that apply:

- [ ] a. "network 172.16.11.236 0.0.0.3 area 0" should have area 1 instead of area 0. cross out
- [ ] b. Area 1 should be defined as stub area. cross out
- [ ] c. Wildcard mask for network 192.168.12.0 /24 is not correct. cross out
- [ ] d. Wildcard for network 172.16.11.232/30 is not correct. cross out
- [ ] e. Area for network 172.16.11.232/30 is not correct. cross out

Figure 9.4 Example Troubleshooting Question in Multiple Choice MCQs - Exam

APPENDIX 1: STUDENTS LAB MANUALS

Appendix 1 is a separate document having complete lab manuals for all the labs.

1- Module 1: Device Access and Basic Configurations
2- Module 2: IP Addressing Assignment
3- Module 3: VLANs and Inter-VLAN Routing
4- Module 4: Static Routing
5- Module 5: Network / Port Address Translation
6- Module 6: RIP and RIPng
7- Module 7: OSPFv2 and OSPFv3
8- Module 8: ISIS
9- Module 9: BGP and BP-BGP

ANNEX A: LIST OF LAB MANUALS UPLOADED ON SHARED FOLDER

The latest versions uploaded on google drive are A1-Rev-02 (Full Version) and A3-Rev-01 (Fall 2020 version), both in PDF and editable Word format.

10- Module 1: Device Access and Basic Configurations
11- Module 2: IP Addressing Assignment
12- Module 3: VLANs and Inter-VLAN Routing
13- Module 4: Static Routing
14- Module 5: Network / Port Address Translation
15- Module 6: RIP and RIPng
16- Module 7: OSPFv2 and OSPFv3
17- Module 8: ISIS
18- Module 9: BGP and BP-BGP

ANNEX B: LIST PYTHON SCRIPTS UPLOADED ON SHARED FOLDER

1- README (File)
2- switches (Sample File)
3- lab01-ports-noshut.py
4- lab04-static.py
5- lab05-nat.py
6- lab06-rip.py
7- lab07-ospf.py
8- lab08-isis.py
9- lab09-bgp.py
10- remove-lab04-static.py
11- remove-lab05-nat.py
12- remove-lab06-rip.py
13- remove-lab07-ospf.py
14- remove-lab08-isis.py
15- reset-to-default.py (to delete nvram and reboot – generally for any Cisco device)
16- adhoc.py (A template script to fill/amend and execute for any command)

ANNEX C: SWITCH-A CONFIGURATIONS (TELNET SCRIPTS) UPLOADED FOR AUTOMATED LABS

1- SW1-lab04-static
2- SW1-lab05-nat
3- SW1-lab06-rip
4- SW1-lab07-ospf
5- SW1-lab08-isis
6- SW1-lab09-bgp

ANNEX D: LIST OF SOLUTION DOCUMENTS UPLOADED ON SHARED DRIVE (IP ADDRESSING, NETWORK DIAGRAM, AND DEVICE OUTPUTS)

.

1- **Module 1:** Device Access and Basic Configurations (Complete Solution Document)
2- **Module 2:** IP Addressing Assignment (Complete Assignment Solution)
3- **Module 3:** VLANs and Inter-VLAN Routing (Complete Solution Document)
4- **Module 4:** Static Routing (Complete Solution Document)
5- **Module 5:** NAT / PAT (Fall 2020-A3 Solution Document)
6- **Module 6:** RIP and RIPng (Complete Solution Document)
7- **Module 7:** OSPFv2 and OSPFv3 (Complete Solution Document)
8- **Module 8:** ISIS (Complete Solution Document)
9- **Module 9:** BGP and BP-BGP (Fall 2020-A3 Solution Document)

ANNEX E: DEVICE CONFIGURATION UPLOADED ON SHARED DRIVE AFTER PERFORMING EACH LAB

Shared google drive has uploaded configurations for all the devices for each of the following labs.

1- **Module 1:** Device Access and Basic Configurations (Complete A1 version)
2- **Module 2:** IP Addressing Assignment (Not Applicable)
3- **Module 3:** VLANs and Inter-VLAN Routing (Complete A1 version)
4- **Module 4:** Static Routing (Complete A1 version)
5- **Module 5:** NAT / PAT (Fall 2020-A3 version)
6- **Module 6:** RIP and RIPng (Complete A1 version)
7- **Module 7:** OSPFv2 and OSPFv3 (Complete A1 version)
8- **Module 8:** ISIS (Complete A1 version)
9- **Module 9:** BGP and BP-BGP (Fall 2020-A3 version)

ANNEX F: LIST OF PHYSICAL AND VLAN CONNECTIVITY DIAGRAMS UPLOADED ON SHARED DRIVE

Shared google drive has physical and VLAN connectivity diagrams to set up each of the following labs:

1- **Overall Physical Connectivity Diagram** (To set up racks connectivity)
2- **Module 3:** VLANs and Inter-VLAN Routing (Physical Connectivity Diagram)
3- **Module 4:** Static Routing (Physical Connectivity / VLAN Diagram)
4- **Module 5:** NAT / PAT (Physical Connectivity / VLAN Diagram)
5- **Module 6:** RIP and RIPng (Physical Connectivity / VLAN Diagram)
6- **Module 7:** OSPFv2 and OSPFv3 (Physical Connectivity / VLAN Diagram)
7- **Module 8:** ISIS (Physical Connectivity / VLAN Diagram)
8- **Module 9:** BGP and BP-BGP (Physical Connectivity / VLAN Diagram)

ANNEX G: LIST OF PRE-LAB QUIZZES, CLASS QUIZZES, AND EXAMS IN E-CLASS

1- **Pre Lab Quiz 1:** Device Access and Basic Configurations
2- **Pre Lab Quiz 2:** IP Addressing Assignment
3- **Pre Lab Quiz 3:** VLANs and Inter-VLAN Routing
4- **Pre Lab Quiz 4:** Static Routing
5- **Pre Lab Quiz 5:** NAT / PAT
6- **Pre Lab Quiz 6:** RIP and RIPng
7- **Pre Lab Quiz 7:** OSPFv2 and OSPFv3
8- **Pre Lab Quiz 8:** ISIS
9- **Pre Lab Quiz 9:** BGP and BP-BGP
10- **Class Quiz 1:** 10 Questions
11- **Class Quiz 2:** 10 Questions
12- **Class Quiz 3:** 10 Questions
13- **Exam:** 30 Questions

WORKS CITED

https://docs.python.org/3/library/telnetlib.html. (n.d.). *Python Documentation*.

https://documentation.nokia.com/cgi-bin/doc_list.pl. (n.d.). *Nokia Technical Documentation*.

https://www.cisco.com/. (n.d.). *Cisco Systems*.

https://www.juniper.net/documentation/. (n.d.). *Juniper Networks TechLibrary*.

https://www.python.org/. (n.d.). *Python Resources*.

https://www.youtube.com/watch?v=rfscVS0vtbw. (n.d.). *Python Full Course for Beginners by Giraffe Academy*.

https://www.youtube.com/watch?v=s6SIVc7C5U0. (n.d.). *Network Automation Tutorial by David Bombal*.

MINT-708 Lab Manuals. (2019).

MINT-708, UoA, Ch. 6: RIP. (2019). Chapter 6: Routing Information Protocol.