# Building Secure Networks

By Group 7 - Sonara

- Helin Liu - B516464
- Sayali Chavan - B812081
- Hitesh Rathi - B813146
- Tom Coombs - B831314

# 1. Introduction

The aim of this module was for each group to build an Internet Service Provider (ISP) with the ultimate goal of building an internet which will enable us to communicate with each host in laboratory. As Group 7 we were assigned with name ‘Sonara’. Each group will have to configure the functionality of an ISP that will be carried out using IPv4 and IPv6 with various protocols.

Groups will configure routing, DNS (Domain Name System), Web Servers, etc and then will interconnect these networks to form an internet using IPv4 and IPv6. Functionality to be provided will be:

- IPv4 and IPv6 on all devices
- An IGP using IS-IS
- IBGP and EBGP interacting with other groups
- A webserver
- TFTP and USB backups to store router configuration

# 2. Lab Exercises

This part explains the procedures which we had followed to complete the below tasks. Each task contains the requirements and how we designed it and the reason behind each task.

<u>Softwares used</u>

We had installed the Ubuntu command line only operating system.

Additional packages:

1. Apache2 -
2. openbsd - inetd
3. tftpd (e.g. tftpd-hpa) - for retrieving router configurations and the backups
4. bind9 -
5. Minicom - Used for Router configurations
6. Traceroute - To find the path of packets
7. Elinks - Its text based browser to view the Webpage

To set up a serial connection we had established the following settings:

- The communications rate - 9600 baud
- data bits- 8 data bits
- The parity – No
- The stop bits – 1
- The flow control – No

Note: For connecting two Routers use a Crossover cable, because routers will not automatically crossover the incoming and outgoing packets. A router needs this cable to crossover the packets internally between routers otherwise it will not show any packages received.

### 2.1.a Fully interconnected LAN using the 3 laptops as clients and the Cisco network switch

Requirement

Interconnection of three hosts using one switch.

Implementation and Reason

Initially we needed to create a network which is fully connected to each other and functioning. So we set static ip addresses to each laptop in the file interfaces through the path /etc/network/interfaces. Then every host was connected to the switch. Once they were configured we were able to ping each laptop.

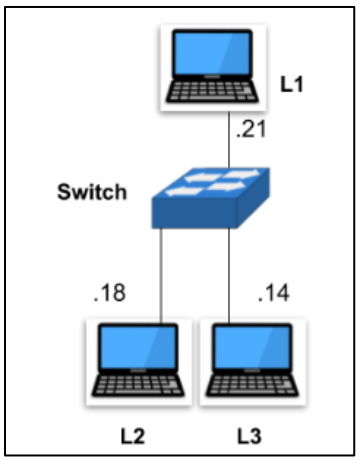

Figure 0. 3 hosts connection using 1 switch

### 2.1.b Introduction to a Web Server

Requirement

Create the Web-page on a host and access that page on other clients.

Implementation and Reason

We had installed apache2-server on server-host (client3).

To access the webpage we needed a web-browser as we were using Ubuntu command line, it doesn't have any graphical interface installed, so we used the text-based browser named - 'elinks' to view the webpage.

Root path: /var/www/html

We had created the webpage named test.html and written HTML program which will display "Hello World !!" message on webpage.

To retrieve that page on host-1 (client) or host-2 (client2) we used elink :

Sudo elinks will pop-up: go to URL tab.

We used the path: 78.1.0.2/group7/test.html where the webpage is stored on server. It retrieves the data from server and displays the message on the client.

Test and Result

We tried to access the Webserver page from another AS's host network and it downloaded the webpage on that desktop and we were able to view that we used 'wget' text web browser on that host laptop. And successfully viewed the webpage.

### 2.2.a Two Networks interconnected by a single Router

Requirement

This task required us to connect two hosts through a router, and for each LAN there should be at least 10 spare IP addresses available for future enhancement. Implementation needed for both IPv4 and IPv6.

Implementation and Reason

The network diagrams of IPv4 and IPv6 for this task are shown below using figure 1 and figure 2.

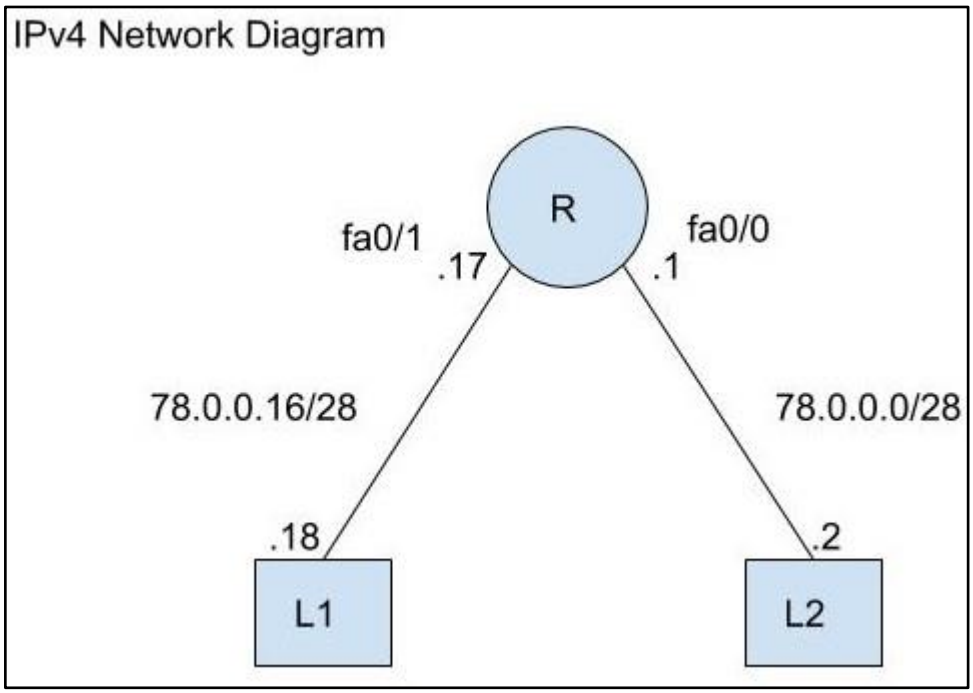


Figure 1. IPv4 Network diagram of 2 hosts & 1 router

For IPv4 (see figure 1.), in order to meet the requirements, due to the limited available addresses in IPv4 addressing system, we did the following calculation:

Task demanded up to 10 hosts on each subnet, Therefore:

$2^n - 2 >= 11$, (10 hosts and 1 IP for the router interface)

Therefore n = 4,

So the CIDR notation (/number) is /32-4 = /28

The 4th octet :

Subnet 1: 0000/0000 => 78.0.0.0/28 (available IPs from .1 to .14)

Subnet 2: 0001/0000 => 78.0.0.16/28 (available IPs from .17 to .30)

Subnet mask 255.255.255.240

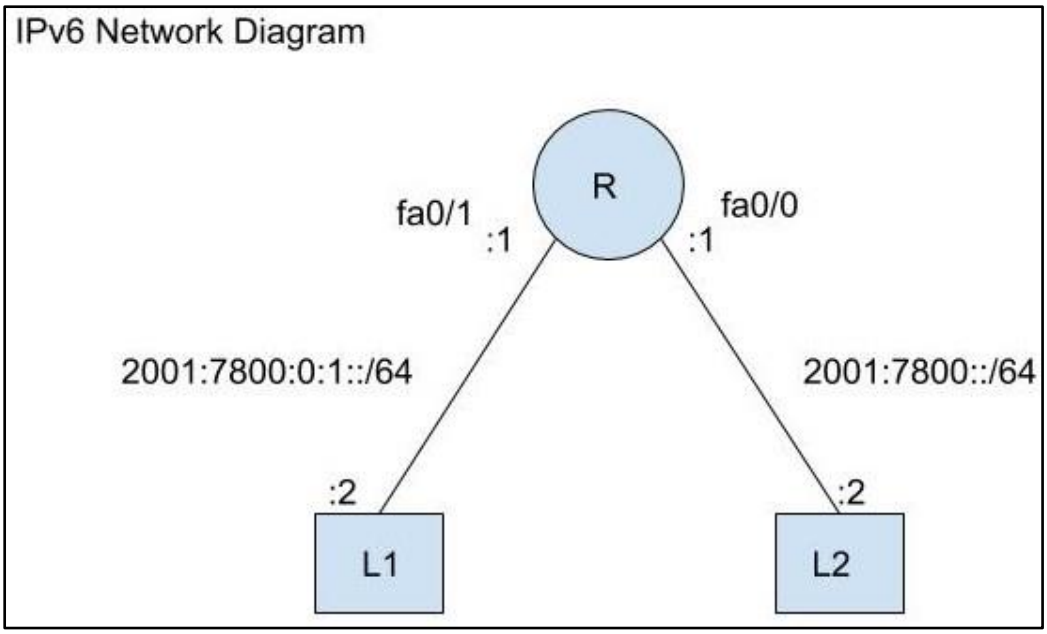


Figure 2. IPv6 Network diagram of 2 hosts & 1 router

For IPv6 (see figure 2.), as there are huge amount of addresses available, we made the subnet simple.

The 4th section in network part:

Subnet 1:

0000 0000 0000 0000 (binary)

=> 0000 (hex)

=> 2001:7800::/64 (18,446,744,073,709,551,616 available IPs)

Subnet 2:

0000 0000 0000 0001 (binary)

=>0001(hex)

=>2001:7800:0:0001::/64 =>2001:7800:0:1::/64

(18,446,744,073,709,551,616 available IPs)

A static route was not required because in this task as the router had interfaces connected to both subnets, therefore there was no need for static routing.

### 2.2.b Three Fully Interconnected Routers

Requirement

This task required 3 laptops and 3 routers. Each laptop would connect to a router and there was point-to-point connection between routers. Implementation needed for both IPv4 and IPv6.

Implementation and Reason

The network diagrams of IPv4 and IPv6 for this task are showing on figure 3 and figure 4.

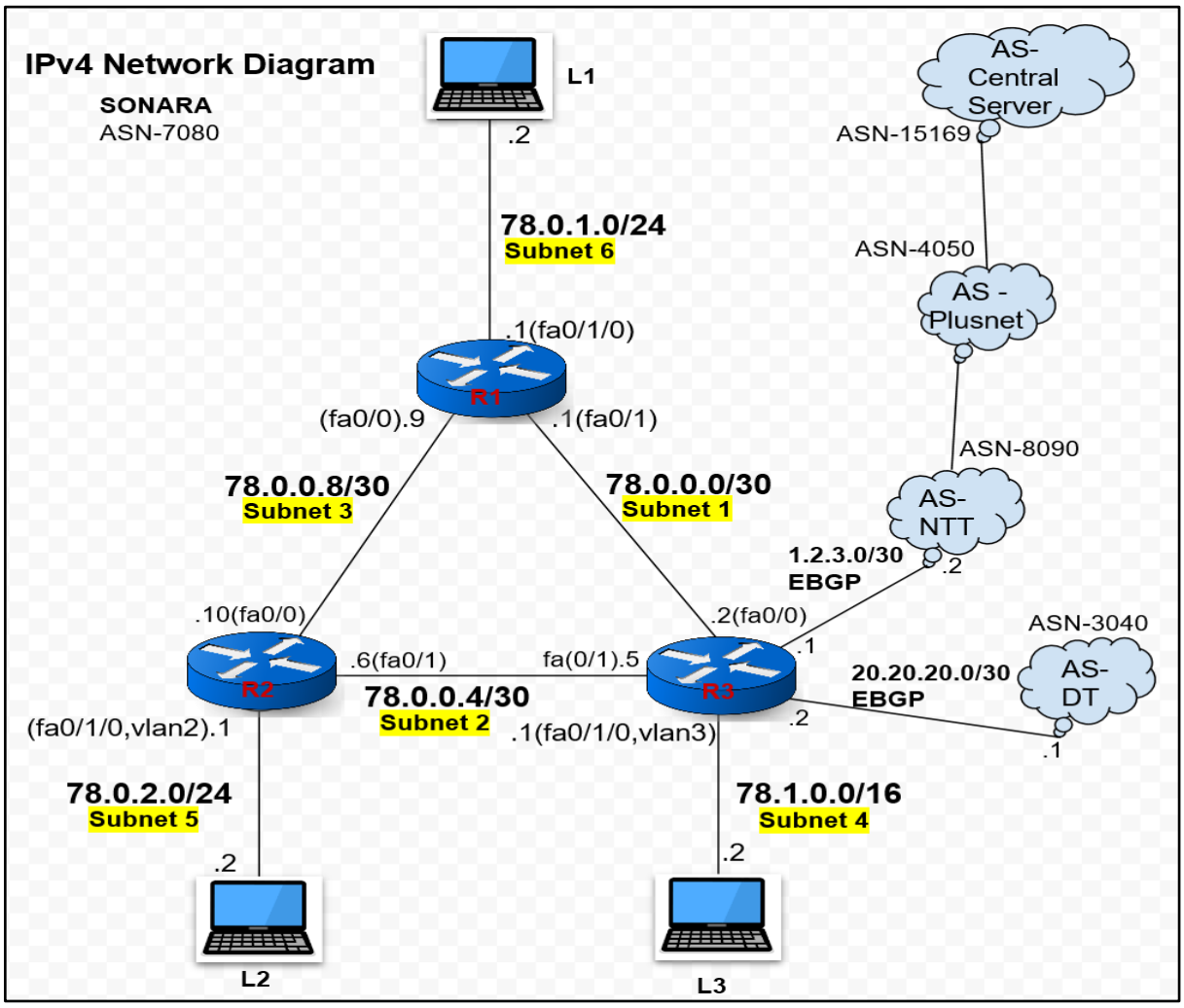


Figure 3. IPv4 Network Diagram 3 hosts & 3 routers

In this task for IPv4. In the subnets which connected the routers, only 2 IPs were required for each of the subnet. Therefore:

$2^n - 2 >= 2$, n = 2,

So the CIDR notation (/number) is /32-2 = /30,

The 4th octet:

Subnet 1: 000000/00 => 78.0.0.0/30 (available IPs from .1 to .2)

Subnet 2: 000001/00 => 78.0.0.4/30 (available IPs from .5 to .6)

Subnet 3: 000010/00 => 78.0.0.8/30 (available IPs from .9 to .10)

Subnet 4: 78.1.0.0/16 (65534 IPs available)

Subnet 5: 78.0.2.0/24 (254 IPs available)

Subnet 6: 78.0.1.0/24 (254 IPs available)

-Reason of IP assignment

As for the subnets that connected to the hosts, in order to fit the realistic situation, we use CIDR notation /24 and /16 to increase the scalability. So for each subnet whose CIDR notations are /24 and /16 would have more available IPs to assign. This would allow expansion for each subnet when additional IPs (mobile hosts, desktop hosts) are required. For subnet 4, the CIDR notation /16 would have 65534 IPs which would be more suitable for clients like large companies, and the CIDR notation /24 was more suitable for clients like small companies. The basic rule is that the bigger the CIDR notation is, the more subnets, the less number of hosts there are in each subnet.

-VLAN

In figure 3, router 1 had a 1-port HWICs. Therefore we assigned the 3 interfaces including fa0/0, fa0/1, fa0/1/0 in a corresponding way. However, router 2 and router 3, instead of having a 1-port HWICs, they had HWICs. The HWICs was a switch-router interface which had 4 small ports. In order to use them, we were required to create vlans to gain access to these port in HWICs. In this case, we assigned a “vlan2” for router 2 and a “vlan3” for router 3, vlan2 was then configured to be able to gain access to the port fa0/1/0 in router 2, and we did the same configuration to vlan3 to gain access to the port fa0/1/0 in router 3. And we kept the vlan configuration for all the following tasks.

-Static routes

In this task as we did not implement IS-IS yet, in that case we needed to assign static routes to all of the routers. Refer to figure 3, the table below shows our configuration of static routing. ( As the way of assigning static routes for each router was similar, here only use the table of static route for router 1 as an example. See table 1.)

| Router 1 | Destination | Destination IP | Netmask | Gateway IP | cost |
|---|---|---|---|---|---|
| | L2 | 78.0.2.0 | 255.255.255.0 | 78.0.0.10 | 2 |
| | L3 | 78.1.0.0 | 255.255.0.0 | 78.0.0.2 | 2 |
| Subnet R2-R3 | | 78.0.0.4 | 255.255.255.252 | 78.0.0.10 | 1 |
| | L2 backup | 78.0.2.0 | 255.255.255.0 | 78.0.0.2 | 3 |
| | L3 backup | 78.1.0.0 | 255.255.0.0 | 78.0.0.10 | 3 |
| Subnet R2-R3 backup | | 78.0.0.4 | 255.255.255.252 | 78.0.0.2 | 2 |

Table 1. Static Routing table of Router 1

-Reason of static routing assignment (refer to table 1.)

The destination IP and netmask column are easy to understand, the key things here are gateway IP (or next hop point) and cost (or distance). The gateway IP is the next point where the router would send its current holding package to. In this case, if the destination is L2. Therefore from Router 1, the next hop point would be the interface of Router 2 which connects Router 1 and Router 2. Normally, the cost parameter does not matter if there was only one route to a destination. However, when a backup route is required, the cost matters. According to Table 1, when there is more than one routes to a destination, the router will take the route that has the least cost. If there are two routes to one destination which have the same cost, the router does not know which route to execute. The value of the cost parameter is used only as comparison so that the router can distinguish between normal routes and backup routes.

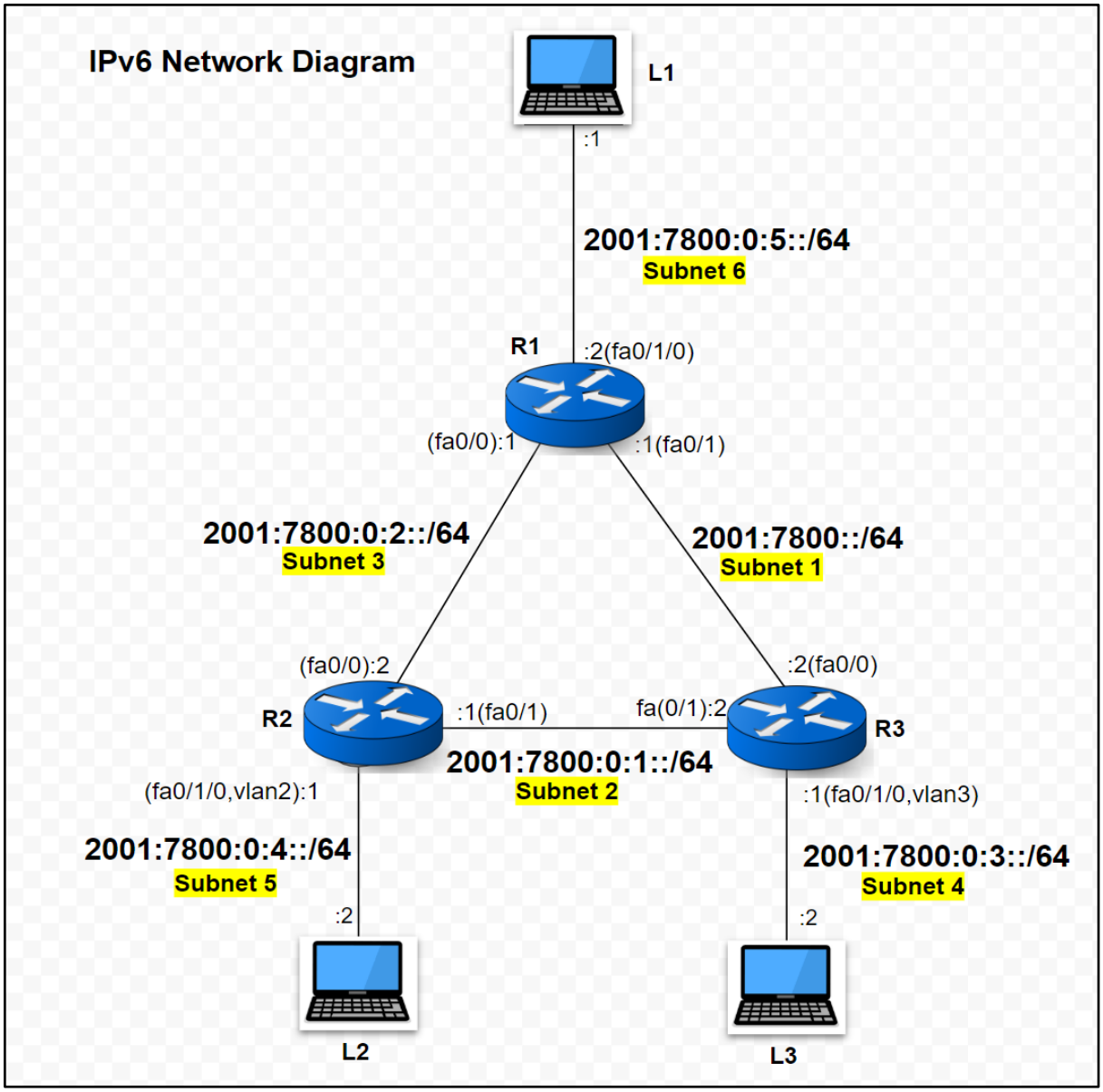


Figure 4. IPv6 Network Diagram - 3 hosts & 3 routers

In this task for IPv4. In each subnet, only 2 IPs were required. However, as stated before from the v6 addressing in 2.2a. IPv6 addressing system has huge and enough amount of IPs that can be assigned to. So we defined the IPv6 subnets in this task in a simple way.

The 4th section in network part:

Subnet 1:

0000 0000 0000 0000 (binary)

=> 0000 (hex)

=> 2001:7800::/64 (18,446,744,073,709,551,616 available IPs)

Subnet 2:

0000 0000 0000 0001 (binary)

=> 0001 (hex)

=> 2001:7800:0:0001::/64

=> 2001:7800:0:1::/64 (18,446,744,073,709,551,616 available IPs)

Subnet 3:

0000 0000 0000 0002 (binary)

=> 0002 (hex)

=> 2001:7800:0:0002::/64

=> 2001:7800:0:2::/64 (18,446,744,073,709,551,616 available IPs)

Subnet 4:

0000 0000 0000 0003 (binary)

=> 0003 (hex)

=> 2001:7800:0:0003::/64

=> 2001:7800:0:3::/64 (18,446,744,073,709,551,616 available IPs)

Subnet 5:

0000 0000 0000 0004 (binary)

=> 0004 (hex)

=> 2001:7800:0:0004::/64

=> 2001:7800:0:4::/64 (18,446,744,073,709,551,616 available IPs)

Subnet 6:

0000 0000 0000 0005 (binary)

=> 0005 (hex)

=> 2001:7800:0:0005::/64

=> 2001:7800:0:5::/64 (18,446,744,073,709,551,616 available IPs)

### 2.2.c Three Fully Interconnected Routers with Dynamic Routing

Requirement

This task required 3 laptops and 3 routers. Each laptop would connect to a router and there should be point-to-point connection between routers once again. Implementation needed for both IPv4 and IPv6. The routing should be carried out dynamically, the protocol chosen was IS-IS.

Implementation and Reason

The first step was to remove all the current static routes. To do this we had to reset each router to its default settings. To reset the router the command “write erase” was used. Then we can use command “reload” or power off and on the router. Once the commands had finished executing successfully the router was then restarted. The static routes had now been removed. The next step was to configure the chosen dynamic protocol IS-IS. Although the static routes had been removed, the static IP addresses that were previously chosen were kept the same. Then we set up the IS-IS.

After configuring IS-IS for IPv4 we tested that the protocol was working. We were able to ping from different hosts at multiple points in the network. To test IS-IS further a route was removed by disconnecting the interface. When configuring IS-IS for IPv6 we encountered a few problems. The solution was to enable IPv6 (through IPv6 unicasting),enable IPv6 IS-IS for each interface, and also to enable the IPv6 cef.

### 2.2.d Securing access to your routers

Requirement

Set up password and remote access for router.

Implementation and reason

We assigned each router a password in a corresponding way, after the password had been set up, login to the router would then need a password. So its more secure.

### 2.2.e Interdomain Routing

Requirement

Set up BGP and routing with another group.

Implementation and Reason

First we assigned a new vlan (vlan 4) to the fa0/1/1, then we configured the vlan 4 which we gave it an IPv4 address. In our network, as we only had one router with level 2 and the other 2 router were both level ½. Therefore there was no need for us

to implement Internal BGP, package inside our network would be managed by IS-IS. For External BGP, we first enabled the BGP for router 3 and assigned a new vlan (vlan 5) for the BGP connection then made neighbors with our pairing group. Then we faced a new problem, from our pairing group, the only point that they can ping was the host which the router 3 (router that connected the outside AS) was connected to. After some troubleshooting, we discovered that we needed a new line of command inside router IS-IS, and the command was “redistribute BGP <asn>”, this command allowed IS-IS and BGP to work corporately. Problem solved. Another point to mention was that we advertised all of our host-router subnets to our pairing neighbours but not any router-router subnets, the reason of doing so was there was no need to advertise our router-router subnets.

### 2.2.f TFTP backups

Requirement

Use TFTP to store the configuration file from each router to the laptop storage.

Implementation and Reason

Trivial File Transfer Protocol (TFTP) was used in order to save the config file of the router to an external device. The first step was to create a TFTP tp server on one of the laptops. The first step was to install the package TFTP packages. The next step was to edit the configuration file stored under the path “/etc/default/tftpd-hpa”. The next step was to create an outgoing directory owned by root mode 755. The directory “var/lib/tftpboot” was directed to and then the command “ sudo chmod 755 $(sudo mktemp -d XXXXXXXXXX --suffix=-outgoing)” was run. The next step was to create an incoming directory owned by tftp mode 700. The command “sudo chown tftp:tftp $(sudo mktemp -d XXXXXXXXXX --suffix=-incoming)” was written. The last step was to restart the TFTP server. With the server configured the next step is to create an empty file on the laptop that the server can write to, using the touch command. This was done using the “chmod 777” command setting the file accessible to all. Then within the server the command copy running-config was used to copy the config from the server to the destination file on the laptop. With the file on the laptop the file can then be copied to a USB. To do this the USB must first be mounted. Once mounted

the cp command was used with the source and destination file paths to copy the file to the USB. The config files have been successfully backed up.

### 2.2.g Policies

Requirement

Create appropriate policies and apply them on the BGP, so there are some restrictions on the traffic that come in and out through our network.

Implementation and Reason

For this task, we tried two lines of command to filter the traffic that went through our network, the commands were combined of two parts, the first was inside “router bgp”, we tried; neighbor 1.2.3.2 prefix-list peer-out out, and then outside the router bgp, we configured the restriction on the prefix list using the following command: ip prefix-list peer-out deny 78.0.2.0/24 le 32 and ip prefix-list peer-out permit 0.0.0.0/0 le 32. To explain what we did and why, the first point is the command inside router bgp, the reason why we only used peer out was that we only wanted to block one of our host subnet. Therefore, we concluded that we only needed to deny our subnet (78.0.2.0/24) traffic to exit from our network and gave permit to every other traffic and that was the reason why we only used peer-out. And it was working properly which means the host in the subnet we blocked could not ping the outside network but the other two hosts can.

# 3. Conclusion and Further work

From above experiment, we can conclude that we have created the three fully Interconnected Routers and with BGP we managed to connect with another AS’s to reach the central server.When we tried to ping the one host from one network to another host of another network, by using traceroute we saw that router is taking the best possible path with less hop count. ISIS and BGP both protocols are working. We tried incorporating DNS server and Email functionality but due to time constraint we were not able to finish it.However, in future we are planning to complete these functionality along with security policies like SSL and Prefix-list policies.

**Appendix**

- (Server) Router 3 Configurations

```
!
version 15.0
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router3
!
boot-start-marker
boot-end-marker
!
enable password router3
!
no aaa new-model
dot11 syslog
ip source-route
!
!
!
!
ip cef
ipv6 unicast-routing
ipv6 cef
multilink bundle-name authenticated
!
!
!
!
!
!
```

```
!
!
!
voice-card 0
!
!
!
!
!
license udi pid CISCO2801 sn FCZ124112JK
username router3 password 0 router3
!
!
!
!
!
!
!
!
interface Loopback0
 ip address 10.10.1.4 255.255.255.255
 ipv6 address 3:3:3:3:3:3:3:3/128
!
interface FastEthernet0/0
 ip address 78.0.0.2 255.255.255.252
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:7800::2/64
 ipv6 enable
 ipv6 router isis
!
interface FastEthernet0/1
 ip address 78.0.0.5 255.255.255.252
 ip router isis
```

```
 duplex auto
 speed auto
 ipv6 address 2001:7800:0:1::2/64
 ipv6 enable
 ipv6 router isis
!
interface FastEthernet0/1/0
 switchport access vlan 3
!
interface FastEthernet0/1/1
 switchport access vlan 4
!
interface FastEthernet0/1/2
 switchport access vlan 5
!
interface FastEthernet0/1/3
 switchport access vlan 6
!
interface Vlan1
 no ip address
!
interface Vlan3
 ip address 78.1.0.1 255.255.0.0
 ip router isis
 ipv6 address 2001:7800:0:3::1/64
 ipv6 enable
 ipv6 router isis
!
interface Vlan4
 ip address 30.20.10.1 255.255.255.252
!
interface Vlan5
 ip address 1.2.3.1 255.255.255.252
!
```

```
interface Vlan6
 ip address 20.20.20.2 255.255.255.252
!
router isis
 net 49.0001.0100.1000.1004.00
 is-type level-2-only
 redistribute bgp 7080
!
router bgp 7080
 no synchronization
 bgp log-neighbor-changes
 network 78.0.0.0
 network 78.0.1.0 mask 255.255.255.0
 network 78.0.2.0 mask 255.255.255.0
 network 78.1.0.0 mask 255.255.0.0
 neighbor 1.2.3.2 remote-as 8090
neighbor 1.2.3.2 prefix-list peer-in in
neighbor 1.2.3.2 prefix-list peer-out out
 neighbor 20.20.20.1 remote-as 3040
 neighbor 30.20.10.2 remote-as 6070
 no auto-summary
!
ip forward-protocol nd
!
!
no ip http server
no ip http secure-server
!
!
ip prefix-list peer-out seq 5 deny 78.0.2.0/24 le 32
ip prefix-list peer-out seq 10 permit 0.0.0.0/0 le 32
!
!
!
```

control-plane
!
!
!
mgcp fax t38 ecm
mgcp behavior g729-variants static-pt
!
!
!
!
line con 0
line aux 0
line vty 0 4
 login
!
scheduler allocate 20000 1000
end

- (Client 2) Router 2 Configurations

!
version 15.0
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router2
!
boot-start-marker
boot-end-marker
!
enable password router2
!
no aaa new-model

```
dot11 syslog
ip source-route
!
!
!
!
ip cef
ipv6 unicast-routing
ipv6 cef
multilink bundle-name authenticated
!
!
!
!
!
!
!
!
!
!
voice-card 0
!
!
!
!
!
license udi pid CISCO2801 sn FCZ1339C100
username router2 password 0 router2
!
!
!
!
!
!
!
```

```
interface Loopback0
 ip address 10.10.1.3 255.255.255.255
 ipv6 address 2:2:2:2:2:2:2:2/128
!
interface Loopback1
 no ip address
 ipv6 address 2001:DB8::1/128
!
interface FastEthernet0/0
 ip address 78.0.0.10 255.255.255.252
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:7800:0:2::2/64
 ipv6 enable
 ipv6 router isis
!
interface FastEthernet0/1
 ip address 78.0.0.6 255.255.255.252
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:7800:0:1::1/64
 ipv6 enable
 ipv6 router isis
!
interface FastEthernet0/1/0
 switchport access vlan 2
!
interface FastEthernet0/1/1
!
interface FastEthernet0/1/2
!
interface FastEthernet0/1/3
```

```
!
interface Vlan1
 no ip address
!
interface Vlan2
 ip address 78.0.2.1 255.255.255.0
 ip router isis
 ipv6 address 2001:7800:0:4::1/64
 ipv6 enable
 ipv6 router isis
!
router isis
 net 49.0001.0100.1000.1003.00
!
ip forward-protocol nd
!
!
no ip http server
no ip http secure-server
!
!
!
!
control-plane
!
!
!
mgcp fax t38 ecm
mgcp behavior g729-variants static-pt
!
!
!
!
line con 0
```

line aux 0
line vty 0 4
 login
!
scheduler allocate 20000 1000
end

- (Client 1) Router 1 Configurations

!
version 15.0
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router1
!
boot-start-marker
boot-end-marker
!
enable password router1
!
no aaa new-model
dot11 syslog
ip source-route
!
!
!
!
ip cef
ipv6 unicast-routing
ipv6 cef
multilink bundle-name authenticated
!

```
!
!
!
!
!
!
!
!
voice-card 0
!
!
!
!
!
license udi pid CISCO2801 sn FCZ1339C10B
username router1 password 0 router1
!
!
!
!
!
!
!
!
interface Loopback0
 ip address 10.10.1.5 255.255.255.255
 ipv6 address 1:1:1:1:1:1:1:1/128
!
interface FastEthernet0/0
 ip address 78.0.0.9 255.255.255.252
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:7800:0:2::1/64
 ipv6 enable
```

```
 ipv6 router isis
!
interface FastEthernet0/1
 ip address 78.0.0.1 255.255.255.252
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:7800::1/64
 ipv6 enable
 ipv6 router isis
!
interface FastEthernet0/1/0
 ip address 78.0.1.1 255.255.255.0
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:7800:0:5::2/64
 ipv6 enable
 ipv6 router isis
!
router isis
 net 49.0001.0100.1000.1005.00
!
ip forward-protocol nd
!
!
no ip http server
no ip http secure-server
!
!
!
!
control-plane
!
```

```
!
!
mgcp fax t38 ecm
mgcp behavior g729-variants static-pt
!
!
!
!
line con 0
line aux 0
line vty 0 4
 login
!
scheduler allocate 20000 1000
end
```

- WebServer HTML Program

```
<html>
<head>
        <meta charset="utf-8">
        <title>Building Secure Network</title>

</head>
<body>
    <div style="margin-top:250px;" align="center">
        <h1>This page is retrived from Server/LAPTOP-3</h1>
   </div>
</body>
</html>
```