Building Secure Networks (COP502)

Lab Work Assessment
Group 2 - BT

Andy Toward
Dan Forrester
Matt Beechey
Rawa Resul
Simon Yip

Introduction

In this report we discuss our work undertaken to design and implement our own Internet Service Provider (ISP). This ISP (BT) includes IPv4 with range 23.0.0.0/8 and IPv6 with range 2001:2300::/32. For BGP our ASN is 3356. We have configured internal and external routing, Domain Name System (DNS), Web Servers, TFTP, SSH and IS-IS. We will discuss our work completed and work that was incomplete, but also how we would want to improve our network in the future if we had more time, and how we may improve the process of development if we had another attempt at the project.

Startup Configuration

Laptops

For each laptop, we installed a command-line only version of Ubuntu. To do this, we connected the laptop to the central server via ethernet cable, booted the laptop and held down F12 to get to the boot menu where we chose to install over the network. Once we had installed ubuntu, we then opened the file with the command '*sudo nano /etc/network/interfaces'* and edited the following lines:

auto etho

iface th0 inet static

This will change the configuration to static. Finally, we closed the file and rebooted the computer.

In order to ensure that the machines had the required functionality for the tasks, a number of packages needed to first be downloaded from the central server. We achieved this by using the command '*sudo get-apt install <package>'* to install the following:
apache2, w3m, tftpd, minicom, bind9, traceroute and exim4.

Startup Discussion

Setup

Firstly, the middle and bottom routers are each connected to a laptop via a standard Category 5 ethernet cable and the router ports that connect to the laptop are HWIC ports, meaning they each have a VLAN established on them. The top router however does not have a VLAN established as there is only one port on that router, meaning separation of networks in such a manner is not necessary. The group's network is connected to the main server via a straight ethernet Category 5 cable. As this is on a router with more than one HWIC port, it is assigned as VLAN4 on port Fa0/1/3 with the address 1.2.2.2. The connection to the "Sky" group is via VLAN5 on an HWIC port from the "bottom" router, connected via a straight Category 5 ethernet cable. The IP address for that network is 12.0.0.62.

Each router (being "top", "middle" and "bottom") is connected to the other two routers respectively with crossover Category 5 ethernet cables, via the router's FastEthernet ports. For example, "top" is connected to "middle" via FE0/0 on the "top" router to FE0/1 on the "middle" router. This system allows for each laptop to connect to any router with SSH (and also if required, all laptops could potentially connect to the same router).

IPv4 On All Devices

Before we setup the IP addresses on the routers and laptops, we drew a diagram of our network to ensure our subnet and broadcast address were correct. Once we confirmed that we had chosen suitable IPv4 address for the port on the routers and laptop, we implemented this into the devices.

To set ipv4 address on the laptops, we opened the /etc/network/interfaces file with the command

sudo nano /etc/network/interfaces

and added the following lines to the file:

auto etho
iface th0 inet static
address 23.0.0.1
netmask 255.255.255.240
broadcast 23.0.0.15
gateway 23.0.0.14

After we setup IPv4 address on all three laptops, we began configuring the IPv4 addresses on each of the routers. As previously mentioned, we have connected all three routers with ethernet cables, with each router connected to a laptop. First, we used minicom to access the router terminal (e.g. router: bottom), and went into privileged Exec mode using the command *enable*. To see the interfaces in the router, we used the command *show interface summary.* This allows us to see which interfaces are connected. To assign IPv4 addresses to an interface (port), we went into configuration mode of the router using the command *config t* (To configure from the terminal). Then, we chose an interface to configure with the command *int <InterfaceName>* (e.g. int fa0/0). After this, we entered the ip address that we wanted to assign to that interface (port) using the following syntax: "*IP address <ipv4 address> <mask>*", e.g. "*IP address 23.0.0.14 255.255.255.252".* We have then assigned an IPv4 address to a port. To finish, we enter *no shutdown* to prevent the interface from shutting down.

For the routers 'bottom' and 'middle', we have HWICs which are small 4-port (on 1-port) plugs on the router which are basically a switch. To assign an IPv4 address on one of the ports on a HWIC, we need to configure a VLAN on the router. In the router's configuration mode, we created a VLAN using this command *interface vlan <num>*, where <num> is the number you want to assign for this VLAN. Next, we add a port from the HWIC to the created VLAN using these three commands:

router(config) interface fa0/1/<port>
router(config-if) switchport mode access
router(config-if) switchport access vlan <num>

To add an IPv4 address to the VLAN, we selected the vlan interface using the command *interface vlan <num>*, and enter the ip address we want to assign using the syntax *ip address <ipv4 address> <mask>*.

IPv6 On All Devices

Setting up IPv6 addresses is the same as setting up IPv4 addresses, except we use the syntax *IPv6 <ipv6 address>/<mask>.*

To configure a IPv6 address for the laptops, we added the following lines in the /etc/network/interfaces file:

```
iface th0 inet6 static
    address 2001:2300::1
    netmask 64
    gateway 2001:2300::14
```

Finally, we saved these changes on the router to NVRAM using the command *copy running-config startup-config.*

TFTP Backups

We implemented a TFTP server on Laptop 3, 23.0.0.33. To do this we installed the TFTPD-HPA application. It proved to be slightly tricky to set up as the application does not provide very good error description and so we had an issue where “no such file or directory exists”. This was because we found that when using the “copy running-config tftp://23.0.0.33/routerx” command TFTPD-HPA cannot create a file, only edit it. We therefore used the “touch” command to create empty files for the application to then save the router configurations to. This also meant that we had to use “chmod” to modify permissions.

As we set up TFTP relatively early on compared to other groups, we were able to provide it as a service to other ASes by allowing them to perform backups to our TFTP server. This does have issues around how the files are stored and access rights but this was still a very useful experience for us to see how much of the work in this module could be profitable.

SSH/Telnet setup

Firstly, the group needed to set up the router’s hostname, they were assigned the names “top”, “middle” and “bottom”, depending on where in the stack of three routers they were located. After this, the domain name had to be allocated (despite the fact that at the time DNS had not been completed on the routers yet), to do this it was necessary to access the router’s configuration file, (using “config t”) and then entering “ip domain name bt.lboro”. Then, to continue to authentication steps provided, setting up a username and password with “username bt1 password lsu” (for the “bottom” router, as the laptop connected to it was also called “bt1”). It was then required to enable the password requirement with “enable password lsu”. This concluded the authentication steps for enabling and activating SSH.

The group could then move on to actually enabling SSH. Firstly, it was necessary to enter “vty” configuration mode, using the command “line vty 0 4”, this allows for five simultaneous connections to each router from a laptop (given there is only three, this means that all laptops are able to connect at once). After this, we enabled SSH and Telnet with “transport input ssh telnet”, and then this allowed for assigning the locally defined username and password to the SSH/Telnet sessions with “login local”.

Afterwards, we decided to add an extra layer of security by generating private and public keys for SSH version 2. This was achieved with “crypto key generate rsa general - keys”, and then we increased the minimum keys size to 4096 (otherwise modern SSH clients such as PuTTY for Microsoft Windows) does not work. To achieve this end, we used the command “ip ssh dh min size 4096”.

With all of the steps completed, it was then possible to disconnect the ethernet-to-USB cables from the console port and solely utilise SSH to connect to any of the routers, with the command “ssh -l bt1 23.23.23.1”, with “bt1” being replaced by the username provided earlier on for the other routers, and replacing “23.23.23.1” with the loopback addresses of the other router that a connection is desired with.

We had some issues with connecting to routers from laptops that weren’t directly connected to a specific router, the issue it turned out was simply that we had not added the router’s loopback address to the IS-IS configuration. Once we had done so, the issue was resolved and we could then SSH into any router from any laptop.

Network Structure

IS-IS

For our network, an interior gateway protocol (IGP) was required to exchange routing information between the three routers in our AS, which could then be used to route our IPv4 and IPv6 protocols. The IGP we elected to used was IS-IS, standing for Intermediate System to Intermediate System. IS-IS is one of the three network-layer protocols which work in tandem to deliver the CLNS defined by ISO.

To set up IS-IS routing, each router was first given its own loopback address, which needed to be unique so that it can be independently identified and referenced. This was achieved by logging into the router's configuration mode, and accessing “interface loopback 0”. From here, the IPv4/v6 addresses were entered, along with their respective netmasks. /32 was used for IPv4 and /128 for IPv6.

As well as its own routable loopback address, each router required a system ID, which forms part of its network service access point (NSAP). These are based on the router's loopback address, which ensures a distinct ID for each. By converting the structure of the router's IPv4 loopback address from ABC.DEF.GHI.JKL (including leading zeros) to ABCD.EFGH.IJKL, a system ID is obtained.

The NSAP of the router is then set by entering an AFI (in this case 49), followed by an area ID (0001), then the system ID before being ended with ‘00’. To do this on the laptop, ‘router isis’ is entered from the router's configuration menu, then ‘net’ followed by its NSAP. For example:

```
top(config)# router isis
top(config-router)# net 49.0001.0230.2302.3003.00
```

We stayed with single level IS-IS, and this was specified by entering ‘is-type level-1’, again in the router's IS-IS configuration. The final step was to indicate the interfaces which needed to take part in IS-IS, and which version of IP they needed to be configured for, by accessing the appropriate interface(s) of the router and entering ‘ip/ipv6 router isis’. Once this was done for all three routers, IS-IS was running successfully across our network, allowing our IP protocols to be routed.

Initially, we overlooked the fact that IS-IS needed to be enabled on the router's interfaces, instead attempting to enable it on only the router itself. This meant that when searching for the local network neighbours, via ‘sh isis neighbors’, the other two laptops in the AS were not being displayed. This meant that it was impossible to ping between any two end systems. Eventually, this issue was realised and rectified, enabling IS-IS on the Fa0/0 and 0/1 interfaces of each router, as well as the VLANs which were set up for the HWIC ports.

BGP

Unlike IGPs, where a number of different options were both available and viable for our AS, there is only one recognised and used exterior routing protocol in the Internet: Border Gateway Protocol (BGP). BGP acts as a routing protocol between different ASes, allowing them to connect and behave as peers. It allows end systems to connect to multiple higher-tier ISPs, reducing costs and potentially increasing end-to-end performance.

Setting up BGP on our network began by configuring our interior BGP (iBGP) first. Using the AS number given to us by our provider (3356), each router had the other two added as neighbours, using the same remote AS number as that of our local network. In the router's configuration mode, the following example was entered to add an iBGP neighbour to the router 'top':

```
top(config)# router bgp 3356
top(config-router)# neighbor 23.0.0.54 remote-as 3356
```

In addition, each neighbour's source (and by extension, destination) address for BGP messages were defaulted to be loopback addresses, by entering the following:

```
top(config-router)# neighbour 23.0.0.54 update-source loopback0
```

Using the loopback address as the source of messages ensures that the BGP session isn't affected by any hardware complications, and will continue to function correctly if a physical interface affecting multiple BGP paths were to go down. This is only required for an iBGP setup, and once we expanded our network to incorporate eBGP, the need for this loopback fallback could be removed.

Once a full mesh was achieved across our local network, with both other laptops appearing in the BGP table of each machine. We use the bottom router as a connection point for the higher-tier ISP, allowing for eBGP to be implemented and our ISP competitor's systems to be seen and interacted with. The only different in adding neighbours as eBGP, is that instead of using our local AS number, we use the AS number of the network we wish to connect to. For example, In the router's configuration mode, the following example was entered to add an iBGP neighbour to the router 'bottom:

```
top(config)# router bgp 3356
top(config-router)# neighbor 1.2.2.1 remote-as 15169
```

Initially with the eBGP, all laptop and routers except the router physically connected to the other AS weren't able to send data to that AS. We eventually resolved this issues by removing all the 'update-source loopback0' statement in the router configuration mode for all routers.

```
top(config-router)# no neighbour <ip address> update-source loopback0
```

The reason for removing this statement is because the loopback from iBGP is preventing the datagram being sent out to the different ASes. After removing the statements, all other laptops were able to connect and send data to another AS.

Additional Functionality

Web Server

The Apache2 server (our web server) is responsible for hosting our website on one of our laptops (we chose BT1 for this). Installing Apache was very simple with the “sudo apt-get install apache” command being used to install the package. After the package was installed, we could then access the web server directory (on “/var/www/html”) and add our own index.html file with basic information stating which group the server belonged to (Index.html is the default page for apache2). From here (after w3m was installed) we could then access the web page from each laptop in our local AS.

Domain Name Server

The Domain Name System (DNS) is responsible for translating hostnames into IP addresses. This is due to the fact that it is difficult for humans to remember all of the IP addresses and to make it easier to access a host by simply typing in human-readable names. For this purpose, we have set up a DNS server on one of our hosts, namely, BT2 to serve as the main DNS.

Since we are using Ubuntu we opted to use the package Bind9 (Berkley Internet Naming Daemon) to implement the DNS as it is the common package to use. For our network, we only implemented the DNS as a primary server on BT2. We did not implement a (backup) secondary server as this would be unnecessary since our network size is small.

We first started by installing the package by using the command “sudo apt-get install bind9”.
We then changed the local configuration file stored in “/etc/bind/named.conf.local” to add a zone to make the DNS act as a primary master server. The name for the zone file was simply “db.bt.lboro” and of type master.

After the configuration, we copied the “db.local” file and named it “db.bt.lboro” to act as a template instead of writing everything from scratch by using the command “sudo cp /etc/bind/db.local /etc/bind/db.bt.lboro”.

Inside “db.bt.lboro” we changed the localhost to “bt.lboro” and the email address to “root.bt.lboro”. Firstly, we included the name server by adding

- IN NS BT2.bt.lboro.

followed by our own (A) address records, namely,

- BT1.bt.lboro. IN A 23.0.0.1
- BT2.bt.lboro. IN A 23.0.0.17
- BT3.bt.lboro. IN A 23.0.0.33

To test our configuration, we saved the “db.bt.lboro” file and then run the command “sudo service bind9 restart” to restart the bind9 service. We then made use of the nslookup function to check for the hostnames we created by using “nslookup BTx.bt.lboro” where x stands for the number of the host.

Reverse DNS

Reverse DNS is the exact opposite of DNS whereby we try to map an IP address to a domain name. However, as one would assume 23.0.0.1 does not point to BT1.bt.lboro in reverse DNS as they are stored on different data sets but 1.0.0.23 will.

In order for the reverse DNS to function, we have created a new file called “db.23.0.0” and added it to the “named.conf.local” file as a new zone. For the name, we wrote “0.0.23.in-addr.arpa” and with the type master. Noticeably, the IP address is written in reverse excluding the last octet and we added “in-addr.arpa”. This is a special type of PTR that we are going to use for the reverse DNS and our PTR records in “db.23.0.0”.

In “db.23.0.0” we simply copy everything from “db.bt.lboro” except for the (A) address records. Instead, we add our PTR records

- 1 IN PTR BT1.bt.lboro.
- 17 IN PTR BT2.bt.lboro.
- 33 IN PTR BT3.bt.lboro.

The numbers 1, 17, and 33 are our host addresses from our IP address.

Again, to test our new configuration we saved the file, restarted the bind9 service and used “nslookup + IP address of the host” to check for the domain name (see DNS appendices).

Conclusion/Summary

For this project, we feel that we have completed the majority of its objectives and functionality and everyone has participated in the practical and report writing. While we have encountered a few difficulties with the practicals, we managed to overcome a large number of them and obtain valuable knowledge to configure Cisco routers. A reflective commentary is included throughout the report, but we will also address some specific points here.

The naming of our routers and end systems followed a logical process, so that each member of the group could easily identify which component of the network they were interacting with/configuring at any time. The routers were named ‘top’, ‘middle’ and ‘bottom’, corresponding with their position in the rack mount. Additionally, this meant that when dealing with the physical components of the network (e.g. cables), we could alter configurations and setups on the correct ports. Finally, when it came to creating our network diagrams, our naming system made addressing far easier, as it was easy to visualise the network topology.

We have configured three routers in full mesh and used the first two ports (Fa0/0 and Fa0/1) to interconnect to each router. We used the first two ports in each router because it is easier to use direct ports than the HWICs on the router to connect. This is because to utilise the HWIC ports, we would need to create more VLANs and therefore more subnetworks. This would have added to the complexity of the network significantly for absolutely no gain in efficiency or ease-of-use.

We have drawn a network diagram and allocated continuous IPv4 addresses so there are no unused IPv4 addresses between the subnets. As we were required to set up the network to account for up to 10 end systems in each network segment, we elected to give each router a /28 address, as this provides space for up to 14 devices (2^4 - 2). A /29 address would only allow for up to 6, and /27 leaves a lot of wasted address space.

We have chosen the specific loopback address 23.23.23.1, 23.23.23.2 and 23.23.23.3 because it repeats the first number of the ip address we have been allocated for this project (23.0.0.0/8). This makes it clear that these gateways belong to our network, and is highly memorable, increasing our efficiency when working. IPv6 addresses for each subnet are given a prefix of /64 as we have an excess number of addresses, and so decided to use a commonly used prefix for all subnets in our network.

We have used IS-IS for the internet network as it uses Dijkstra’s algorithm for computing the best path through a network. This negates the risk of a “count to infinity” problem, if one were to arise, and also allows for convergence at a faster rate. As IS-IS is more lightweight, it scales up to larger networks easier, so if we had more time, using IS-IS would mean that we could significantly increase the size of the network and still have efficient routing.

We set up SSH on our router to remove the need to physically connect from the laptop to the router via the console port. We opted to do this because we were needlessly utilising ports on the laptop and routers (using Category 5 ethernet cables and USB-to-ethernet cables). Adding SSH to our laptops and allowing them to connect to the routers meant we could simplify our network and remove some of the less necessary cabling.

When creating the web server, we opted to use Apache (as opposed to something like NGINX) as Apache has vastly more supportive documentation readily available online and generally is easier to create and support the server on the version of Ubuntu provided. Furthermore, although NGINX is faster in some circumstances, the website used for displaying via W3M was so small that the difference would be negligible. However, should we have been allotted more time, we may have added to the website and therefore it may have been worth it to configure an NGINX web server, rather than Apache.

We attempted to configure exim4 as our email server, but ran into multiple issues that no documentation found was able to fix. For example, even though we had our network fully configured (and were able to ping every part of the network from every laptop or router), the email server had issues with routing between laptops through the network we had established. The lack of error messaging in exim4's main log file meant that unfortunately we weren't able to locate the specific issue that was causing the routing problem and fix it, and thus were unable to complete this section of the specification. If we were to try the project again, we would certainly like to allocate more time to the email server and try to get exim4 working fully with our network, as we had done with the rest of the specification.

If we were given more time or were to give this project a second attempt, we would also like to have completed the email functionality and perhaps perform an AS prefix hijack of other groups. We have configured the email server using the exim4 module but unfortunately due an unforeseen error, we were unable to receive emails from the laptops. We have also implemented prefix-filters to control inbound and outbound traffic from different ASes. However, we were unable to test this due to time constraints.

Overall, we can conclude this as a fairly successful project.

Appendices

IPv4 Address Table

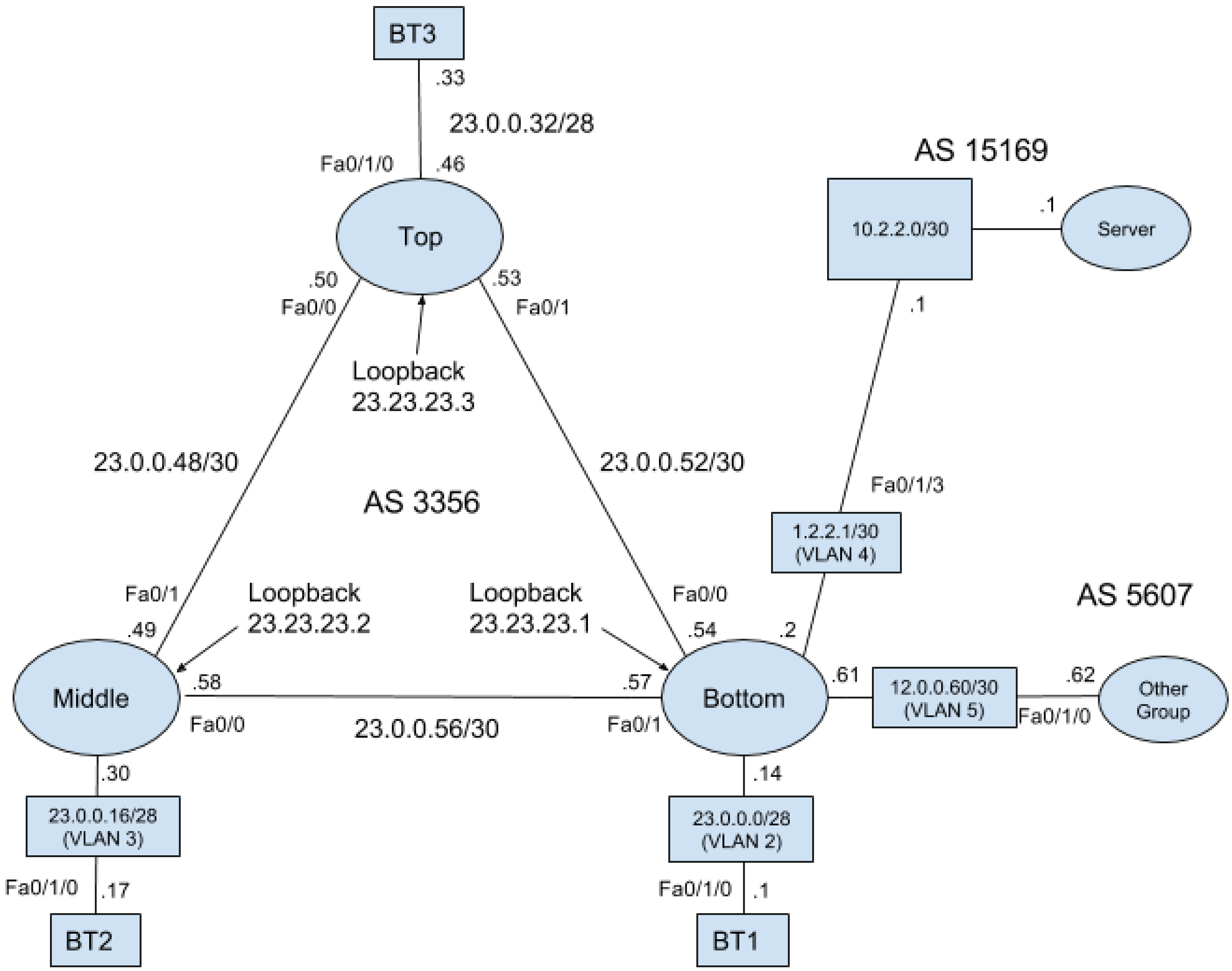
Device Name	Own IP	Netmask	Gateway	Fa0/0	Fa0/1	VLAN Fa0/1/0
BT1	23.0.0.1	255.255.255.240	23.0.0.14	-	-	-
BT2	23.0.0.17	255.255.255.240	23.0.0.30	-	-	-
BT3	23.0.0.33	255.255.255.240	23.0.0.46	-	-	-
top	23.23.23.3	255.255.255.255	-	23.0.0.50	23.0.0.53	23.0.0.46
middle	23.23.23.2	255.255.255.255	-	23.0.0.58	23.0.0.49	23.0.0.30
bottom	23.23.23.1	255.255.255.255	-	23.0.0.64	23.0.0.57	23.0.0.14

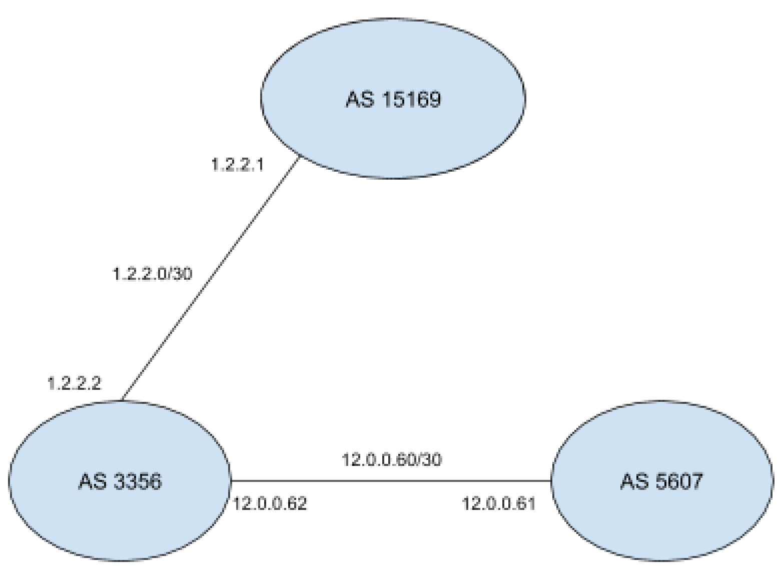
Device Name	VLAN Fa0/1/1	VLAN Fa0/1/3
bottom	12.0.0.61	1.2.2.1

IPv4 Link Table

	BT1	BT2	BT3	top	middle	bottom
BT1						23.0.0.14
BT2					23.0.0.30	
BT3				23.0.0.46		
top			23.0.0.33		23.0.0.49	23.0.0.64
middle		23.0.0.17		23.0.0.49		23.0.0.57
bottom	23.0.0.1			23.0.0.53	23.0.0.53	

IPv4 Diagrams





IPv6 Address Table

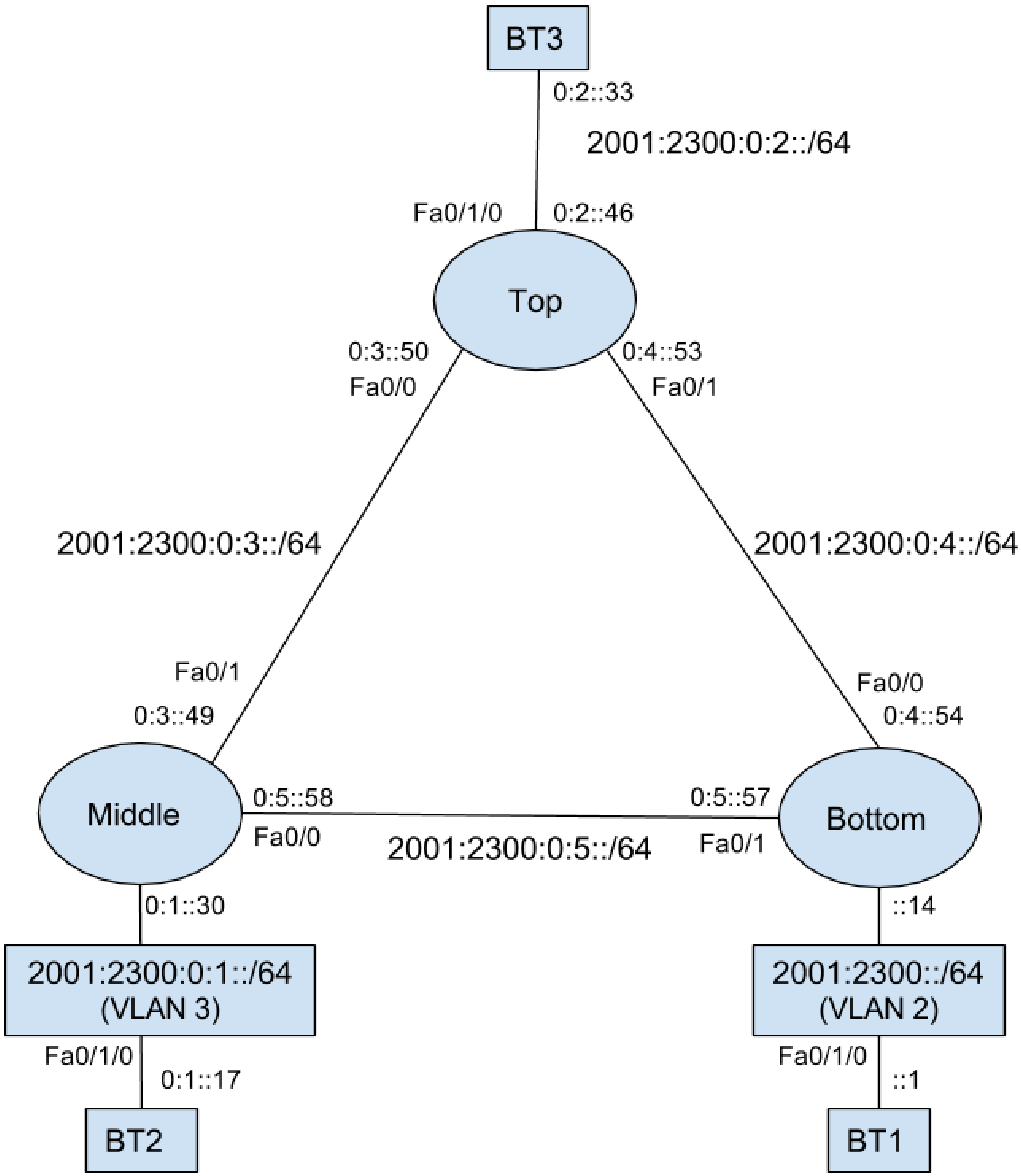
Device Name	Own IP	Netmask	Gateway	Fa0/0	Fa0/1	VLAN Fa0/1/0
BT1	2001:2300::1	/64	2001:2300::14	-	-	-
BT2	2001:2300:0:1::17	/64	2001:2300:0:1::30	-	-	-
BT3	2001:2300:0:2::33	/64	2001:2300:0:2::46	-	-	-
top	2001:2300:23::3	/128	-	:0:3::50	:0:4::53	:0:2::46
middle	2001:2300:23::2	/128	-	:0:5::58	:0:3::49	:0:1::30
bottom	2001:2300:23::1	/128	-	:0:4::54	:0:5::57	::14

IPv6 Link Table

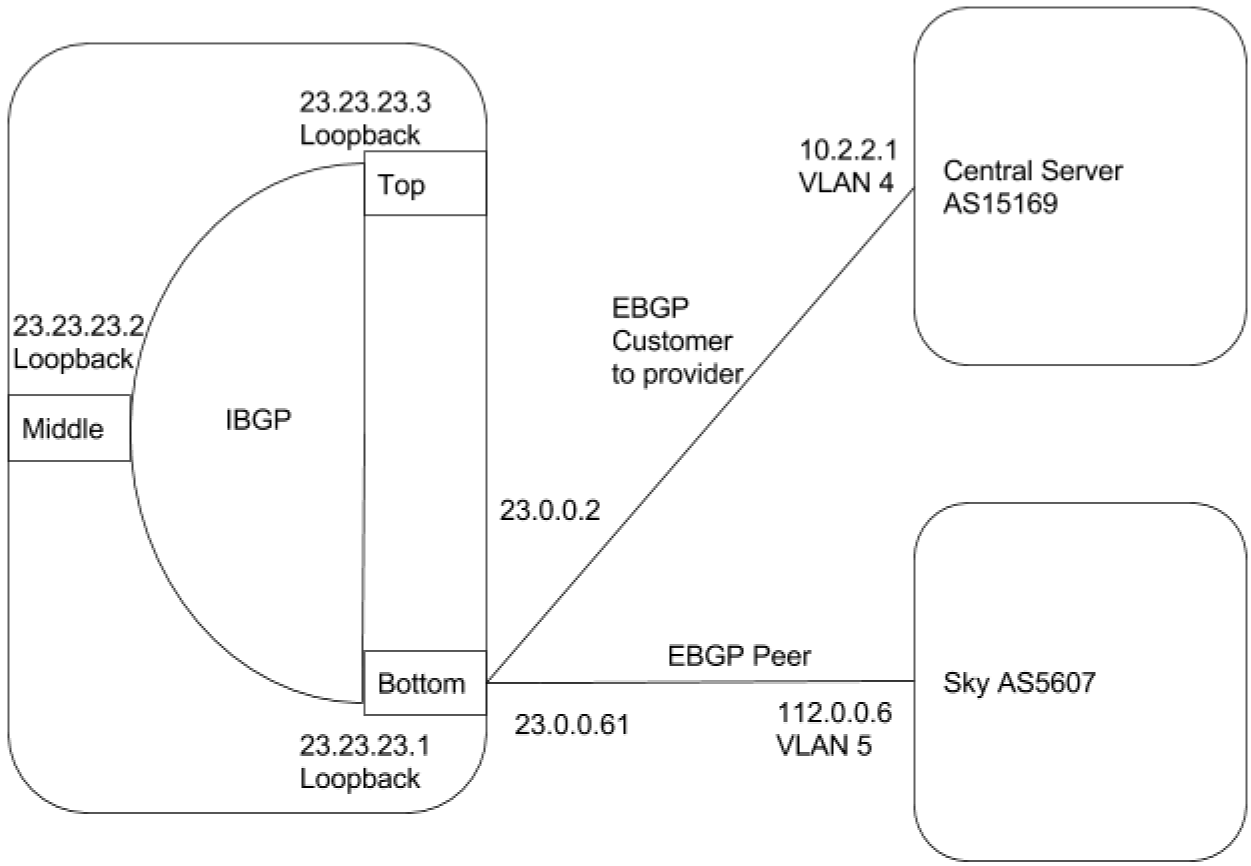
(2001:2300:...)

	BT1	BT2	BT3	top	middle	bottom
BT1						::14
BT2					0:1::30	
BT3				0:2::46		
top			0:2::33		0:3::49	0:4::54
middle		0:1::17		0:3::50		0:5::57
bottom	::1			0:4::53	0:5::58	

IPv6 Diagram



BGP Diagram


23.23.23.3
Loopback
Top
23.23.23.2
Loopback
Middle
IBGP
Bottom
23.23.23.1
Loopback
10.2.2.1
VLAN 4
Central Server
AS15169
EBGP
Customer
to provider
23.0.0.2
EBGP Peer
23.0.0.61
112.0.0.6
VLAN 5
Sky AS5607

Top Router Configuration

```
!
! Last configuration change at 14:52:33 UTC Tue Nov 28 2017 by bt3
!
version 15.0
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname top
!
boot-start-marker
boot-end-marker
!
enable password lsu
!
no aaa new-model
dot11 syslog
ip source-route
!
!
!
!
ip cef
no ip domain lookup
ip domain name bt.lboro
ipv6 unicast-routing
ipv6 cef
multilink bundle-name authenticated
!
!
!
!
!
!
!
!
!
voice-card 0
!
!
!
!
!
```

```
license udi pid CISCO2801 sn FCZ1339C10B
username bt3 password 0 lsu
!
!
ip ssh version 2
ip ssh dh min size 4096
!
!
!
!
!
interface Loopback0
 ip address 23.23.23.3 255.255.255.255
 ip router isis
 ipv6 address 2001:2300:23::3/128
 ipv6 router isis
!
interface FastEthernet0/0
 ip address 23.0.0.50 255.255.255.252
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:2300:0:3::50/64
 ipv6 router isis
!
interface FastEthernet0/1
 ip address 23.0.0.53 255.255.255.252
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:2300:0:4::53/64
 ipv6 router isis
!
interface FastEthernet0/1/0
 ip address 23.0.0.46 255.255.255.240
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:2300:0:2::46/64
 ipv6 router isis
!
router isis
 net 49.0001.0230.2302.3003.00
 is-type level-1
!
router bgp 3356
 no synchronization
```

```
 bgp log-neighbor-changes
 network 23.0.0.48 mask 255.255.255.252
 network 23.0.0.52 mask 255.255.255.252
 neighbor 23.0.0.49 remote-as 3356
 neighbor 23.0.0.49 update-source Loopback0
 neighbor 23.0.0.54 remote-as 3356
 neighbor 23.0.0.54 update-source Loopback0
 no auto-summary
!
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 local
 transport input telnet ssh
!
scheduler allocate 20000 1000
end
```

Middle Router Configuration

```
!
version 15.0
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname middle
!
boot-start-marker
boot-end-marker
!
enable password lsu
!
no aaa new-model
dot11 syslog
ip source-route
!
!
!
!
ip cef
no ip domain lookup
ip domain name bt.lboro
ipv6 unicast-routing
ipv6 cef
multilink bundle-name authenticated
!
!
!
!
!
!
!
!
!
voice-card 0
!
!
!
!
!
license udi pid CISCO2801 sn FCZ1339C100
```

```
username bt2 password 0 lsu
!
!
ip ssh version 2
ip ssh dh min size 4096
!
!
!
!
!
interface Loopback0
 ip address 23.23.23.2 255.255.255.255
 ip router isis
 ipv6 address 2001:2300:23::2/128
 ipv6 router isis
!
interface FastEthernet0/0
 ip address 23.0.0.58 255.255.255.252
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:2300:0:5::58/64
 ipv6 router isis
!
interface FastEthernet0/1
 ip address 23.0.0.49 255.255.255.252
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:2300:0:3::49/64
 ipv6 router isis
!
interface FastEthernet0/1/0
 switchport access vlan 3
!
interface FastEthernet0/1/1
 switchport access vlan 30
!
interface FastEthernet0/1/2
!
interface FastEthernet0/1/3
!
interface Vlan1
 no ip address
!
interface Vlan3
 ip address 23.0.0.30 255.255.255.240
```

```
 ip router isis
 ipv6 address 2001:2300:0:1::30/64
 ipv6 enable
 ipv6 router isis
!
router isis
 net 49.0001.0230.2302.3002.00
 is-type level-1
!
router bgp 3356
 no synchronization
 bgp log-neighbor-changes
 network 23.0.0.16 mask 255.255.255.240
 network 23.0.0.48 mask 255.255.255.252
 network 23.0.0.56 mask 255.255.255.252
 neighbor 23.0.0.50 remote-as 3356
 neighbor 23.0.0.50 update-source Loopback0
 neighbor 23.0.0.57 remote-as 3356
 neighbor 23.0.0.57 update-source Loopback0
 no auto-summary
!
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 local
 transport input telnet ssh
!
scheduler allocate 20000 1000
```

end

Bottom Router Configuration

```
!
version 15.0
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname bottom
!
boot-start-marker
boot-end-marker
!
enable password lsu
!
no aaa new-model
dot11 syslog
ip source-route
!
!
!
!
ip cef
no ip domain lookup
ip domain name bt.lboro
ipv6 unicast-routing
ipv6 cef
multilink bundle-name authenticated
!
!
!
!
!
!
!
!
!
voice-card 0
!
!
!
!
!
license udi pid CISCO2801 sn FCZ124112JK
```

```
username bt1 password 0 lsu
!
!
ip ssh version 2
ip ssh dh min size 4096
!
!
!
!
!
interface Loopback0
 ip address 23.23.23.1 255.255.255.255
 ip router isis
 ipv6 address 2001:2300:23::1/128
 ipv6 router isis
!
interface FastEthernet0/0
 ip address 23.0.0.54 255.255.255.252
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:2300:0:4::54/64
 ipv6 router isis
!
interface FastEthernet0/1
 ip address 23.0.0.57 255.255.255.252
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:2300:0:5::57/64
 ipv6 router isis
!
interface FastEthernet0/1/0
 switchport access vlan 2
!
interface FastEthernet0/1/1
 switchport access vlan 5
!
interface FastEthernet0/1/2
!
interface FastEthernet0/1/3
 switchport access vlan 4
!
interface Vlan1
 no ip address
!
interface Vlan2
```

```
 ip address 23.0.0.14 255.255.255.240
 ip router isis
 ipv6 address 2001:2300::14/64
 ipv6 router isis
!
interface Vlan4
 ip address 1.2.2.2 255.255.255.252
!
interface Vlan5
 ip address 12.0.0.62 255.255.255.252
!
router isis
 net 49.0001.0230.2302.3001.00
 is-type level-1
 passive-interface Vlan4
 passive-interface Vlan5
!
router bgp 3356
 no synchronization
 bgp log-neighbor-changes
 network 1.2.2.0 mask 255.255.255.252
 network 12.0.0.60 mask 255.255.255.252
 network 23.0.0.0 mask 255.255.255.240
 network 23.0.0.16 mask 255.255.255.240
 network 23.0.0.32 mask 255.255.255.240
 network 23.0.0.48 mask 255.255.255.252
 network 23.0.0.52 mask 255.255.255.252
 network 23.0.0.56 mask 255.255.255.252
 neighbor 1.2.2.1 remote-as 15169
 neighbor 12.0.0.61 remote-as 5607
 neighbor 12.0.0.61 prefix-list PEER-IN in
 neighbor 12.0.0.61 prefix-list PEER-OUT out
 neighbor 23.0.0.53 remote-as 3356
 neighbor 23.0.0.53 next-hop-self
 neighbor 23.0.0.58 remote-as 3356
 neighbor 23.0.0.58 next-hop-self
 no auto-summary
!
ip forward-protocol nd
!
!
no ip http server
no ip http secure-server
!
!
ip prefix-list PEER-IN seq 5 deny 23.0.0.0/8
ip prefix-list PEER-IN seq 10 permit 0.0.0.0/0 le 32
```

```
!
ip prefix-list PEER-OUT seq 5 permit 23.0.0.0/28
ip prefix-list PEER-OUT seq 10 permit 23.0.0.16/28
ip prefix-list PEER-OUT seq 15 permit 23.0.0.32/28
ip prefix-list PEER-OUT seq 20 deny 0.0.0.0/0 le 32
access-list 55 permit 23.0.0.0
!
route-map LOCALPREF permit 10
 set local-preference 500
!
!
!
control-plane
!
!
!
mgcp fax t38 ecm
mgcp behavior g729-variants static-pt
!
!
!
!
line con 0
line aux 0
line vty 0 4
 login local
 transport input telnet ssh
!
scheduler allocate 20000 1000
end
```

Laptop 1 Configuration

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static

        address 23.0.0.1
        netmask 255.255.255.240
    broadcast 23.0.0.15
    gateway 23.0.0.14
        dns-search bt.lboro
        dns-nameserver 23.0.0.17

iface eth0 inet6 static

    address 2001:2300::1
    netmask 64
    gateway 2001:2300::14
```

Laptop 2 Configuration

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
        address 23.0.0.33
        netmask 255.255.255.240
        broadcast 23.0.0.47
        gateway 23.0.0.46
        dns-search bt.lboro
        dns-nameservers 23.0.0.17

iface eth0 inet6 static
     address 2001:2300:0:2::33
     netmask 64
     gateway 2001:2300:0:2::46
```

Laptop 3 Configuration

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
        address 23.0.0.33
        netmask 255.255.255.240
        broadcast 23.0.0.47
        gateway 23.0.0.46
        dns-search bt.lboro
        dns-nameservers 23.0.0.17

iface eth0 inet6 static
     address 2001:2300:0:2::33
     netmask 64
     gateway 2001:2300:0:2::46
```

Laptop login details

	BT1	BT2	BT3
Username	bt1	bt2	bt3
Password	lsu	lsu	lsu

DNS db.bt.lboro

```
;
; BIND data file for local loopback interface
;
;
; BIND data file for local loopback interface
;
$TTL    604800
@       IN      SOA     bt.lboro. root.bt.lboro. (
                            3           ; Serial
                         604800              ; Refresh
                          86400              ; Retry
                        2419200              ; Expire
                         604800 )       ; Negative Cache TTL
;
;Name Server
        IN      NS      BT2.bt.lboro.

;Area Codes

;BT1
BT1.bt.lboro.   IN      A       23.0.0.1

;BT2
BT2.bt.lboro.   IN      A       23.0.0.17

;BT3
BT3.bt.lboro.   IN      A       23.0.0.33
```

DNS db.23.0.0

```
;
; BIND data file for local loopback interface
;

$TTL    604800
@       IN      SOA     bt.lboro. root.bt.lboro. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
;Name Server
        IN      NS      BT2.bt.lboro.

;PTR Records

;BT1
1       IN      PTR     BT1.bt.lboro.   ;23.0.0.1

;BT2
17      IN      PTR     BT2.bt.lboro.   ;23.0.0.17

;BT3
33      IN      PTR     BT3.bt.lboro.   ;23.0.0.33
```