# Building BT Network

by
Zhihao DAI
Yunsong ZHANG
Huijing LEI
Changrong CHEN
Yan HUANG

**19COP502 Building Secure Networks**
**Lab Report**

Loughborough University



Nov. 2019

# Abstract

This report describes the building process of BT Network, a small version of Tier-2 Internet Service Provider. To first define the architecture of the network, a network diagram is drawn, in which physical connections are established and IP addresses are assigned. Then, connectivity within the network and with outside networks are made possible through internal routing protocol Intermediate System to Intermediate System (IS-IS) and external Border Gateway Protocol (BGP). In addition, services such as World Wide Web (WWW), Domain Name System (DNS) and Email are provided in the network. At last, main conclusions and further improvements are discussed.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

In this lab, our team sets out to build BT Network, a small version of Tier-2 Internet Service Provider (ISP) located at Haslegrave Building, from scratch. Despite its limitations in terms of size and Internet access, we can proudly attest that BT Network is one of the leading providers at Haslegrave Building.

BT network is an Autonomous System (AS) as a whole and the AS Number is `2030`. Its domain name is `bt.lboro` .

## 1.1 Network Services

Our network provides the following services to each of our individual customers.

- IP addressing with a guaranteed range of 14 host addresses allocated from `23.0.0.0/8` (IPv4) and `2001:2300::/32` (IPv6) blocks.
- Intra-domain Internet connection with Intermediate System to Intermediate System (IS-IS) routing protocol.
- Inter-domain Internet connection with Border Gateway Protocol (BGP).
- A reliable Domain name System (DNS) service with duplicated servers under domain `bt.lboro`.
- A World Wide Web (WWW) service located at http://bt.lboro/ .
- An Email service at `bt.lboro`.

For neighbouring ISPs who is a customer in our business relationships (see Section 1.2.3), we provide the following services.

- Internet connection to our domain as well as all the others'.

In addition, we provide secure remote access to our routers through Secure Shell (SSH) protocol on one of our laptops for administrative purposes.

## 1.2 Business Relationships with Neighbouring ISPs

BT Network has three immediate neighbouring ISPs and it's important to form business relationships with all three of them in order to gain economic benefits. The external routing policies of BGP protocol for each outside network are determined by its business relationship with us (see Section 3.2 for details). Business relationships with neighbouring ISPs are shown in Figure 1.1 and elaborated in the following.

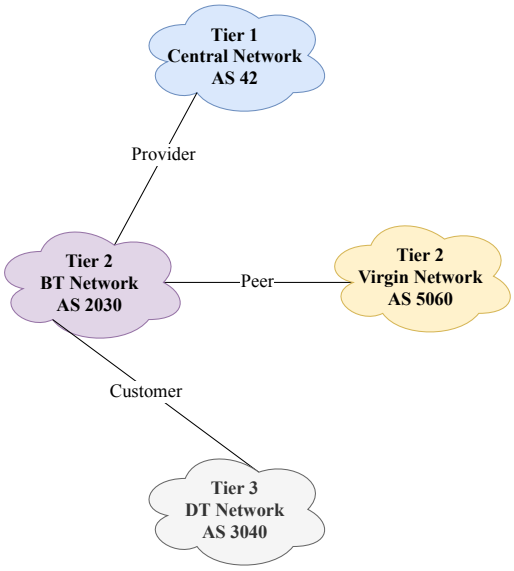



Figure 1.1: Business Relationships of BT Network with Neighbouring ISPs.

### 1.2.1 Provider: Central Network

Since BT Network is a Tier-2 ISP, it need to be connected to a Tier-1 ISP to gain broader Internet connection. Therefore, BT is connected to Central Network, a Tier-1 ISP, as a customer which makes it a network provider for BT.

### 1.2.2 Peer: Virgin Network

BT Network forms a Peer relationship with Virgin Network. This allows Virgin Network to connect to BT Network at zero cost and vice versa.

### 1.2.3 Customer: DT Network

BT Network forms a Provider-Customer relationship with DT Network, in which BT is the provider and DT is the customer. In other words, DT gains access to the broader Internet through BT at a cost.

## 1.3 Roles of Network Components

There are 6 physical components in our network in total, of which 3 are Cisco routers and the other 3 are TOSHIBA laptops. Each component plays an important role in the network as shown in Figure 1.2.

### 1.3.1 Routers

In terms of connection, each router is attached with one customer subnet and thus providing Internet service to one customer. Router 1 (BT-R001) is not physically conncted to any outside network. Router 2 (BT-R002) is connected to DT Network and Router 3 (BT-R003) is connected to Virgin Network and Central Network through cables.

In terms of routing, all routers are Level-1 routers in intra-domain IS-IS routing protocol. In BGP routing protocol, Router 1 (BT-R001) acts as an Internal BGP (IBGP) router while Router 2 (BT-R002) and Router 3 (BT-R003) act as External BGP (EBGP) routers.

### 1.3.2 Laptops

All laptops are running a Ubuntu 16.04 system. Each of them is connected to a customer subnet thourgh a cable. In terms of services, Laptop 1 (BT001) provides DNS service for `bt.lboro` as a secondary DNS server and WWW service at `http://bt.lboro`. It also acts as a secure SSH access point to routers for administrative purposes. Laptop 2 (BT002) doesn't provide any service and thus acts act an normal user in the network. Laptop 3 (BT003) provides DNS service for `bt.lboro` as a primary DNS server. In addition, it hosts Email service at `bt.lboro`.

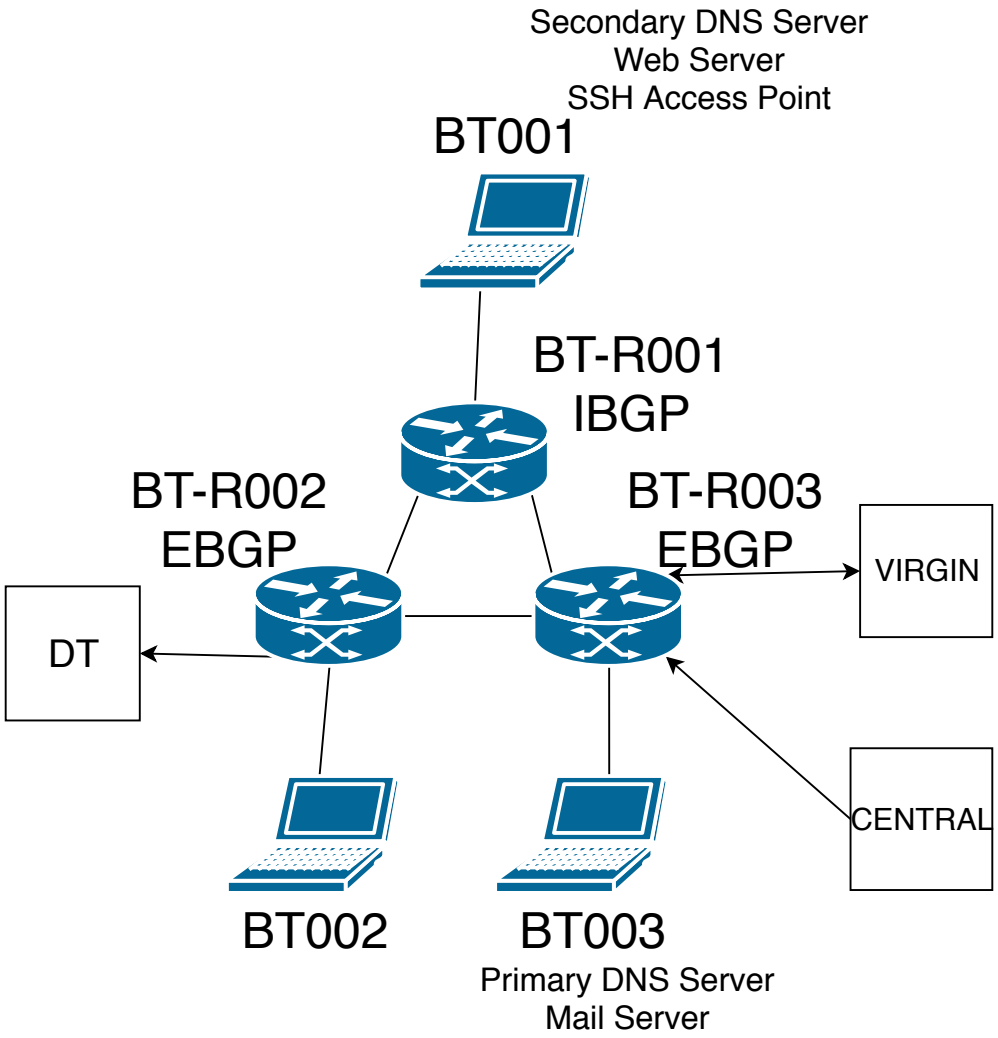



Figure 1.2: Roles of Main Components in BT Network.

## 1.4 Organisation of the Report

The report is organised as follows. The architecture of BT Network on the network layer is described in Chapter 2. A Network Diagram involving all physical components and connections is drawn in the chapter. Then, IP addresses for interfaces in the network are carefully allocated and configured.

In order to allow packets to be forwarded within and outside the network, proper intra-domain and inter-domain routing protocols are implemented and tested in Chapter 3.

In Chapter 4, we move up to the application layer and set up various services in the

network as listed in Section 1.1.

Main conclusions drawn from the building process and possible further improvements are discussed in Chapter 5. At last, a summary of contributions for each team member is presented in Chapter 6.

Although the report does not necessarily reflect the actural order of steps in our building process (eg. remote SSH access was set up before BGP), readers can be assured that all results presented can be reproduced by following the natural order of the report.

To ensure readability, rationale behind important decisions made, problems we encountered and their respective solutions, alternative ways of configurations (if any) as well as reflective commentary are documented along with each step of implementation in the report.

# Chapter 2

# Network Architecture

## 2.1 Network Diagram

### 2.1.1 Description

A full diagram of BT network is shown in Figure 2.1. There are 3 routers in the network, whose names are BT-R001, BT-R002 and BT-R003 resepectively, connected to each other. Each connection forms a Router-Router subnet with only 2 interfaces.

On the other hand, each router is connected with a laptop separately named as BT001, BT002 and BT003 and thus forms a Router-Laptop subnet. A customer of BT Network is assigned with a Router-Laptop subnet and has a minimum of 10 host IP addresses.

To connect to neighbouring ISPs, a Router-Neighbour subnet is formed for each connection. Concretely, Router 2 (BT-R002) is connected to one of DT Network's routers while Router 3 (BT-R003) is connected to one of Virgin Network's and Central Network's routers separately.

All of the above connections are through physical cables.

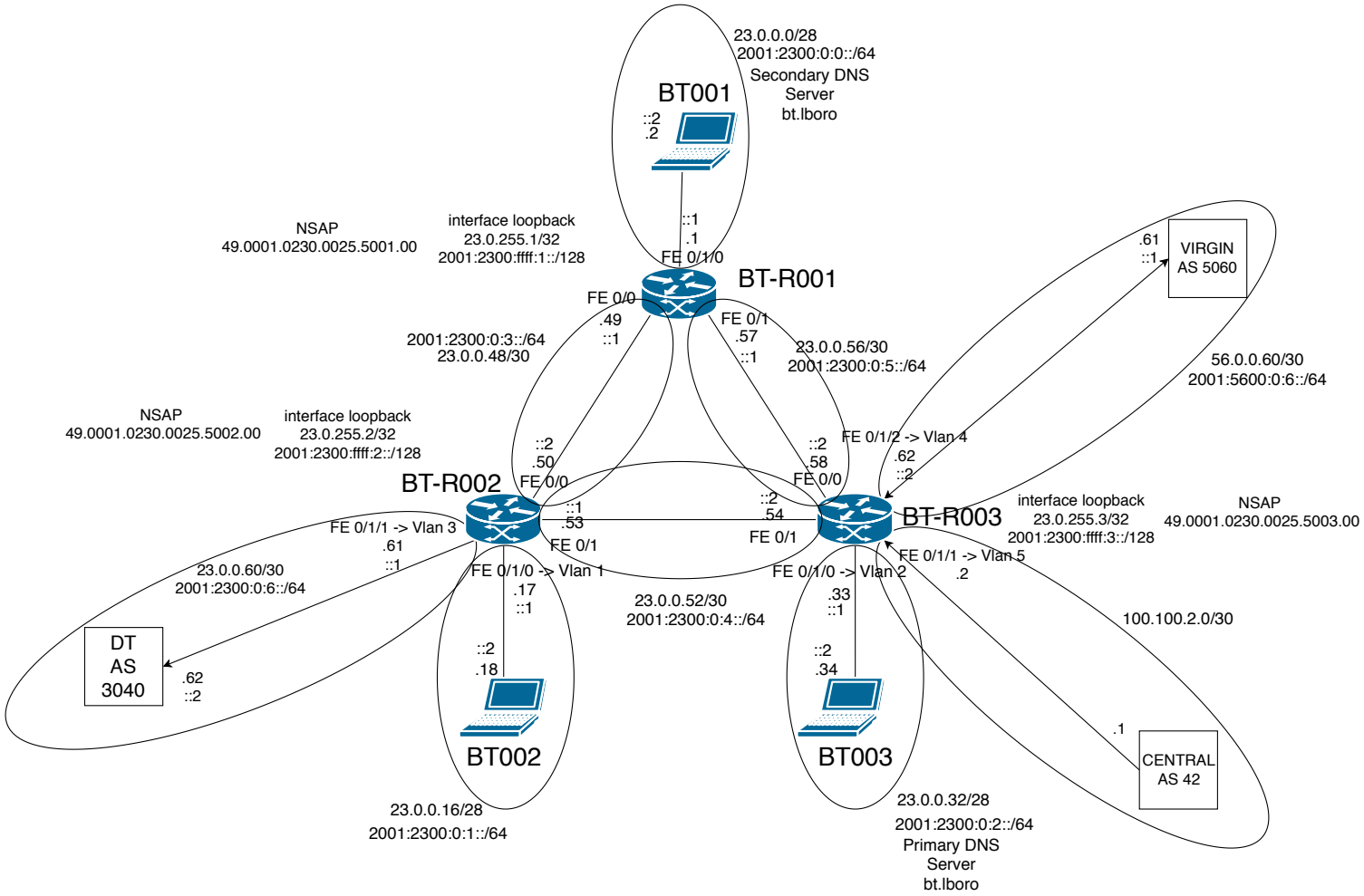



Figure 2.1: Full Network Diagram of BT Network.

### 2.1.2 IP Addresses and Interfaces

An IPv4 address range of `23.0.0.0/8` and IPv6 address range of `2001:2300::/32` are allocated to BT Network, which are further divided into sub-ranges to be allocated to each subnet.

For IPv4 addressing, a prefix of $n$ is needed for a subnet that demands $X$ host addresses, where $n$ is an integer that satisfies $2^{32-n} - 2 \geq X$ and $n \leq 32$. For our lab, the maximum value for prefix is used in order to minimize the size of each subnet and reserve address space for future customers. **However, it's also possible to use a larger value for each Router-Laptop subnet in order to maximize the size of the subnet, given that the number of customers (in this case, 3) is fixed.**

In BT Network, the prefix for each Router-Router and Router-Neighbour subnet is 30 while the prefix for each Router-Laptop subnet is 28. In other words, each Router-Router and Router-Neighbour subnet has 2 guaranteed IPv4 host addresses while each Router-Laptop subnet has 14 guaranteed IPv4 host addresses. During address block allocation, larger subnet is being considered before smaller one reduce the number of block segments.

For IPv6 addressing, however, each subnet has a fixed prefix of 64 to ensure that each interface in the subnet has a unique address. The full details of IP address allocation is shown in Table 2.1.

Figure 2.2: Physical Connections within BT Network.

In terms of interfaces, there are 3 Ethernet interfaces (`FastEthernet0/0`, `FastEthernet0/1` and `FastEthernet0/1/0`) on Router 1, each of which can be assigned with an IP address. On Router 2 and 3, however, there are 6 Ethernet interfaces each and only 2 of them (`FastEthernet0/0` and `FastEthernet0/1`) can be directly assigned with IP addresses. The remaining 4 interfaces are link layer interfaces and thus does not possess any IP address. To be assigned with an IP address, such an interface need to be assigned to an Virtual LAN (VLAN) to which the address is actually assigned.

Router-Router connections are established through either `FastEthernet0/0` or `FastEthernet0/1` interfaces on both ends while Router-Laptop and Router-Neighbour are through one of the remaining interfaces on the router end. Since both interfaces are on the left-hand side of each router and such arrangement helps distinguishing between Router-Router connections and others easily as shown in Figure 2.2. Interfaces of both ends for each connection as well as their corresponding IP addresses are detailed in Table 2.2.

| Subnet | IPv4 Address / Prefix | IPv4 Address Range | IPv6 Address / Prefix | IPv6 Address Range |
|---|---|---|---|---|
| BT-R001 - BT001 | 23.0.0.0/28 | 23.0.0.1 - 23.0.0.14 | 2001:2300:0:0::/64 | 2001:2300:0:0::1 - 2001:2300:0:0:ffff:ffff:ffff:fffe |
| BT-R002 - BT002 | 23.0.0.16/28 | 23.0.0.17 - 23.0.0.30 | 2001:2300:0:1::/64 | 2001:2300:0:1::1 - 2001:2300:0:1:ffff:ffff:ffff:fffe |
| BT-R003 - BT003 | 23.0.0.32/28 | 23.0.0.33 - 23.0.0.62 | 2001:2300:0:2::/64 | 2001:2300:0:2::1 - 2001:2300:0:2:ffff:ffff:ffff:fffe |
| BT-R001 - BT-R002 | 23.0.0.48/30 | 23.0.0.49 - 23.0.0.50 | 2001:2300:0:3::/64 | 2001:2300:0:3::1 - 2001:2300:0:3:ffff:ffff:ffff:fffe |
| BT-R002 - BT-R003 | 23.0.0.52/30 | 23.0.0.53 - 23.0.0.54 | 2001:2300:0:4::/64 | 2001:2300:0:4::1 - 2001:2300:0:4:ffff:ffff:ffff:fffe |
| BT-R001 - BT-R003 | 23.0.0.56/30 | 23.0.0.57 - 23.0.0.58 | 2001:2300:0:5::/64 | 2001:2300:0:5::1 - 2001:2300:0:5:ffff:ffff:ffff:fffe |
| BT-R002 - DT | 23.0.0.60/30 | 23.0.0.61 - 23.0.0.62 | 2001:2300:0:6::/64 | 2001:2300:0:6::1 - 2001:2300:0:6:ffff:ffff:ffff:fffe |
| BT-R003 - Virgin | 56.0.0.60/30 | 56.0.0.61 - 56.0.0.62 | 2001:5600:0:6::/64 | 2001:5600:0:6::1 - 2001:5600:0:6:ffff:ffff:ffff:fffe |
| BT-R003 - Central | 100.100.2.0/30 | 100.100.2.1 - 100.100.2.2 | | |

Table 2.1: Allocation of IPv4 and IPv6 Addresses to Subnets in BT Network.

| Connection | Interface 1 | IPv4 Address | IPv6 Address | Interface 2 | IPv4 Address | IPv6 Address |
|---|---|---|---|---|---|---|
| BT-R001 - BT001 | BT-R001: FastEthernet0/1/0 | 23.0.0.1 | 2001:2300:0:0::1 | BT001: eth0 | 23.0.0.2 | 2001:2300:0:0::2 |
| BT-R002 - BT002 | BT-R002: FastEthernet0/1/0 -> Vlan 1 | 23.0.0.17 | 2001:2300:0:1::1 | BT002: eth0 | 23.0.0.18 | 2001:2300:0:1::2 |
| BT-R003 - BT003 | BT-R003: FastEthernet0/1/0 -> Vlan 2 | 23.0.0.33 | 2001:2300:0:2::1 | BT003: eth0 | 23.0.0.34 | 2001:2300:0:2::2 |
| BT-R001 - BT-R002 | BT-R001: FastEthernet0/0 | 23.0.0.49 | 2001:2300:0:3::1 | BT-R002: FastEthernet0/0 | 23.0.0.50 | 2001:2300:0:3::2 |
| BT-R002 - BT-R003 | BT-R002: FastEthernet0/1 | 23.0.0.53 | 2001:2300:0:4::1 | BT-R003: FastEthernet0/1 | 23.0.0.54 | 2001:2300:0:4::2 |
| BT-R001 - BT-R003 | BT-R001: FastEthernet0/1 | 23.0.0.57 | 2001:2300:0:5::1 | BT-R003: FastEthernet0/0 | 23.0.0.58 | 2001:2300:0:5::2 |
| BT-R002 - DT | BT-R002: FastEthernet0/1/1 -> Vlan 3 | 23.0.0.61 | 2001:2300:0:6::1 | DT | 23.0.0.62 | 2001:2300:0:6::2 |
| BT-R003 - Virgin | BT-R003: FastEthernet0/1/2 -> Vlan 4 | 56.0.0.62 | 2001:5600:0:6::2 | Virgin | 56.0.0.61 | 2001:5600:0:6::1 |
| BT-R003 - Central | BT-R003: FastEthernet0/1/1 -> Vlan 5 | 100.100.2.2 | | Central | 100.100.2.1 | |

Table 2.2: Interfaces for Each Physical Connection and Corresponding IPv4 and IPv6 Addresses.

## 2.2 IP Addresses of Interfaces

Assigning IP addresses[1][2] to interfaces should be the first step in building BT Network since all network services listed in 1.1 cannot operate without IP addresses. The assignment of IP addresses in Table 2.1 and 2.2 is implemented.

### 2.2.1 Implementation

#### 2.2.1.1 Routers

For Router 1 (BT-R001), IP addresses are assigned directly to physical interfaces as all interfaces are network layer interfaces.

```
int fa0/0
ip address 23.0.0.49 255.255.255.252
ipv6 address 2001:2300:0:3::1/64
no shutdown

int fa0/1
ip address 23.0.0.57 255.255.255.252
ipv6 address 2001:2300:0:5::1/64
no shutdown

int fa0/1/0
ip address 23.0.0.1 255.255.255.240
ipv6 address 2001:2300:0:0::1/64
no shutdown
```

For Router 2 (BT-R002), however, only 2 interfaces (`FastEhternet0/0` and `FastEhternet0/1`) each are network layer interfaces. The remaining 4 interfaces are link layer interfaces and need to be assigned to an VLAN separately se where an IP address can be assigned.

```
int fa0/0
ip address 23.0.0.50 255.255.255.252
ipv6 address 2001:2300:0:3::2/64
no shutdown

int fa0/1
ip address 23.0.0.53 255.255.255.252
ipv6 address 2001:2300:0:4::1/64
no shutdown

vlan 1
int fa0/1/0
switchport mode access
switchport access vlan 1
```

```
int vlan 1
ip address 23.0.0.17 255.255.255.240
ipv6 address 2001:2300:0:1::1/64
no shutdown

vlan 3
int fa0/1/1
switchport mode access
switchport access vlan 3
int vlan 3
ip address 23.0.0.61 255.255.255.252
ipv6 address 2001:2300:0:6::1/64
no shutdown
```

Similarly, IP addresses are assigned to Router 3 (BT-R003).

```
int fa0/0
ip address 23.0.0.58 255.255.255.252
ipv6 address 2001:2300:0:5::2/64
no shutdown

int fa0/1
ip address 23.0.0.54 255.255.255.252
ipv6 address 2001:2300:4::2/64
no shutdown

vlan 2
int fa0/1/0
switchport mode access
switchport access vlan 2
int vlan 2
ip address 23.0.0.33 255.255.255.240
ipv6 address 2001:2300:0:2::1/64
no shutdown

vlan 4
int fa0/1/2
switchport mode access
switchport access vlan 4
int vlan 4
ip address 56.0.0.62 255.255.255.252
ipv6 address 2001:5600:0:6::2/64
no shutdown

vlan 5
int fa0/1/1
switchport mode access
switchport access vlan 5
```

```
int vlan 5
ip address 100.100.2.2 255.255.255.252
no shutdown
```

#### 2.2.1.2 Laptops

Unlike routers, the IP assignment of laptops' interfaces are more aligned since all laptops are of the same model. On Laptop 1 (BT001), the file `/etc/network/interface` is edited as follows before rebooting to apply changes. Similarly, IP address can are assigned to Laptop 2's (BT002) and Laptop 3's (BT003) interfaces respectively. Full configuration for all laptops are detailed in Appendix C.

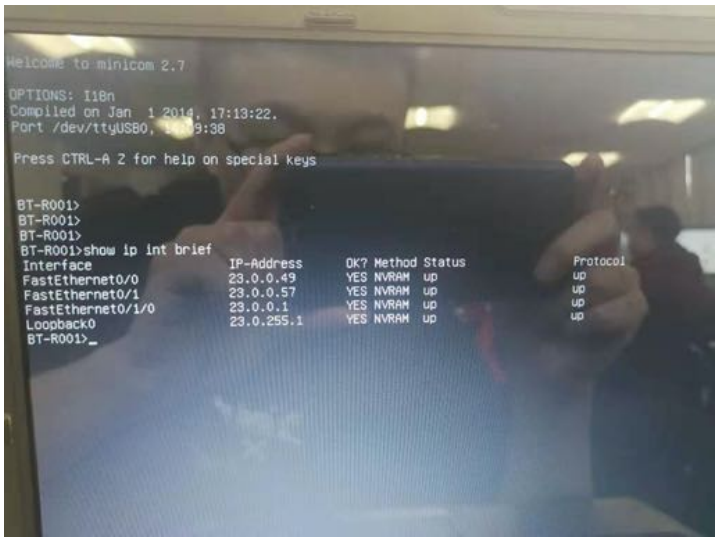
```
auto lo
iface lo inet loopback
auto eth0
ifacce eth0 inet static
address 23.0.0.2
netmask 255.255.255.240
gateway 23.0.0.1

ifacce eth0 inet6 static
address 2001:2300::2
netmask 64
gateway 2001:2300::1
```

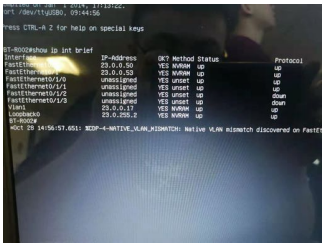
### 2.2.2 Evaluation

On all 3 routers, the implementation of IP assignments is evaluated using `show ip int brief` and `show ipv6 int brief`. These 2 commands show IPv4 and IPv6 addresses of interfaces on routers respectively. Sucessful assignment of IP addresses to routers' interfaces is evident from Figure 2.3 and 2.4.

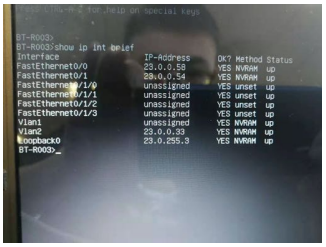
On the laptops' side, the implementation of IP assignments is evaluated using `ifconfig` command. Sucessful assignment of IP addresses to laptops' interfaces is evident from Figure 2.5.




(a) Router 1 (BT-R001)

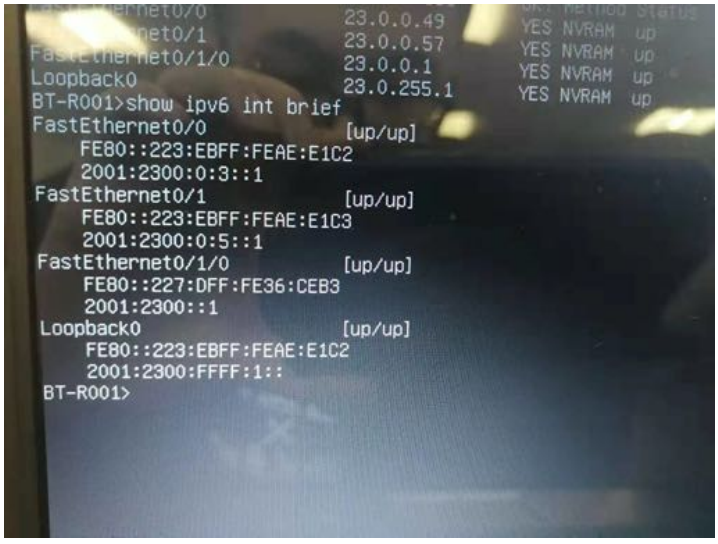


(b) Router 2 (BT-R002)

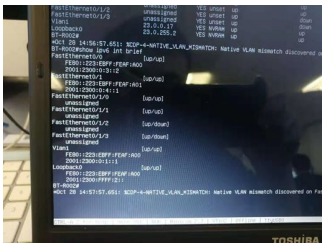



(c) Router 3 (BT-R003)

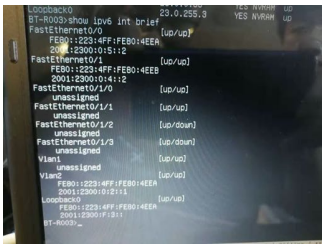
Figure 2.3: Sucessful Assignment of IPv4 Addresses to Routers' Interfaces.




(a) Router 1 (BT-R001)

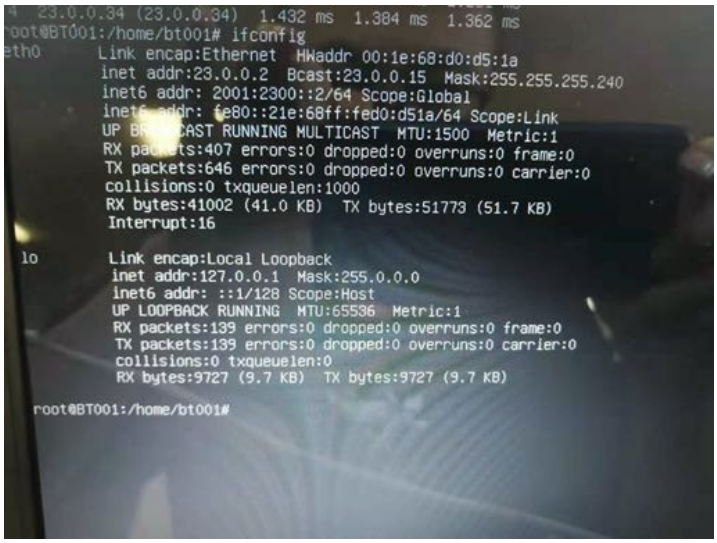


(b) Router 2 (BT-R002)




(c) Router 3 (BT-R003)

Figure 2.4: Sucessful Assignment of IPv6 Addresses to Routers' Interfaces.




(a) Laptop 1 (BT001)

(b) Laptop 2 (BT002)

(c) Laptop 3 (BT003)

Figure 2.5: Sucessful Assignment of IPv4 and IPv6 Addresses to Laptops' Interfaces.

# Chapter 3

# Routing Protocols in the Network

## 3.1 Intra-domain Routing Protocol: IS-IS

### 3.1.1 Design

Due to the limitation of the static routing, routers cannot find alternative paths if a set path is broken and thus a new path need to be set manually. In contrast, dynamic routing always finds the least cost path even when the previous least cost path is broken. Popular dynamic routing protocols include distance vector based protocols like Routing Information Protocol (RIP)[3] and link state based protocols like Intermediate System to Intermediate System (IS-IS)[4] and Open Shortest Path First (OSPF)[5].

In this lab, the IS-IS is used as the Interior Gateway Protocol (IGP), which provides faster convergence and larger scalability compared to distance vector based protocols. IS-IS is short for Intermediate System to Intermediate System Routing Protocol. By using this protocol, each router maintains a database which has a map of the whole topology and all routers have the same information. The best path to every destination is computed by all routers. Figure 3.1 shows the design of IS-IS protocol in BT Network. As the figure shows, the network only has level-1 routers for internal routing.

### 3.1.2 Loopback Addresses and NSAP for Routers

To set up IS-IS, an unique loopback address is needed for each router. An IPv4 address block of `23.0.255.0/24` and IPv6 address block of `2001:2300:ffff::/48` is allocated for loopback addresses.

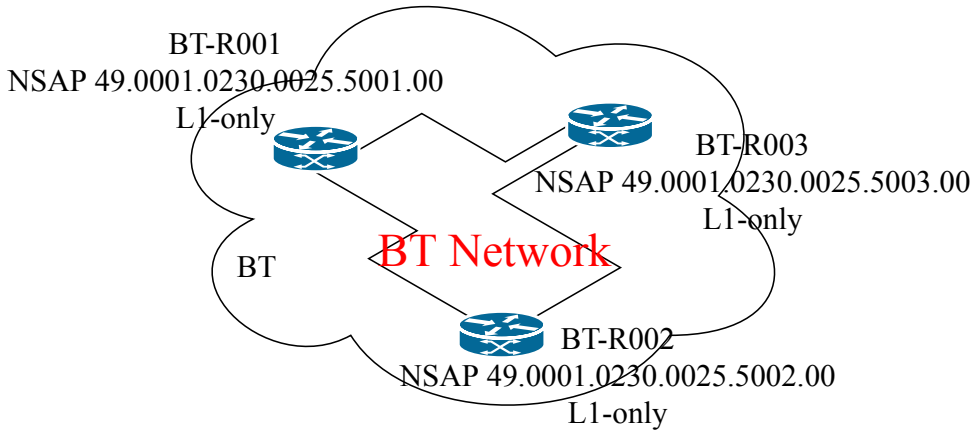



Figure 3.1: Design of IS-IS Protocol in BT Network.

Following the CLNS addressing convention, each router is then assigned with a NSAP. A NSAP has 3 main components.

According to the convention, the leading Area ID is composed of AFI (49) and Area Address (0001). The System ID followed is set to the IPv4 loopback address of the router. If the loopback address is $ABC.DEF.GHI.JKL$, then System ID should be $ABCD.EFGH.IJKL$. The last main component is N-Selector (NSEL) and set to 00.

The assignment of addresses and NSAP to routers are detailed in Table 3.1.

| Router | IPv4 Loopback Address | IPv6 Loopback Address | NSAP |
|---|---|---|---|
| BT-R001 | 23.0.255.1 | 2001:2300:ffff:1:: | 49.0001.0230.0025.5001.00 |
| BT-R002 | 23.0.255.2 | 2001:2300:ffff:2:: | 49.0001.0230.0025.5002.00 |
| BT-R003 | 23.0.255.3 | 2001:2300:ffff:3:: | 49.0001.0230.0025.5003.00 |

Table 3.1: IP Loopback Addresses and NSAP for Routers in BT Network.

### 3.1.3 Implementation

IS-IS is set up on Router 1 (BT-R001) using the following commands.

```
interface Loopback0
ip address 23.0.255.1 255.255.255.255
ipv6 address 2001:2300:FFFF:1::/128

router isis
net 49.0001.0230.0025.5001.00
is-type level-1
```

Then, IS-IS is turned on on all interfaces to internal routers.

```
interface FastEthernet0/0
ip router isis
ipv6 router isis

interface FastEthernet0/1
ip router isis
ipv6 router isis

interface FastEthernet0/1/0
ip router isis
ipv6 router isis
```

However, IS-IS routes should not be broadcasted nor received through the loopback interface (`Loopback0`) while the route to corresponding subnet should be broadcasted to other internal routers. Therefore, the loopback interface should be a passive interface in IS-IS protocol.

```
router isis
passive-interface Loopback0
```

For Router 2 (BT-R002) and Router 3 (BT-R002), IS-IS is set up similarly using the above commands. The main difference is that interfaces to external routers (VLAN 3 for Router 2 and VLAN 4 & 5 for Router 3) should be passive interfaces as well. Below is the configuration for Router 2.

```
interface Loopback0
ip address 23.0.255.2 255.255.255.255
ipv6 address 2001:2300:FFFF:2::/128

router isis
net 49.0001.0230.0025.5002.00
is-type level-1
passive-interface Vlan3
passive-interface Loopback0
```

```
10
11 interface FastEthernet0/0
12 ip router isis
13 ipv6 router isis
14
15 interface FastEthernet0/1
16 ip router isis
17 ipv6 router isis
18
19 interface Vlan1
20 ip router isis
21 ipv6 router isis
```

Below is the configuration for Router 3.

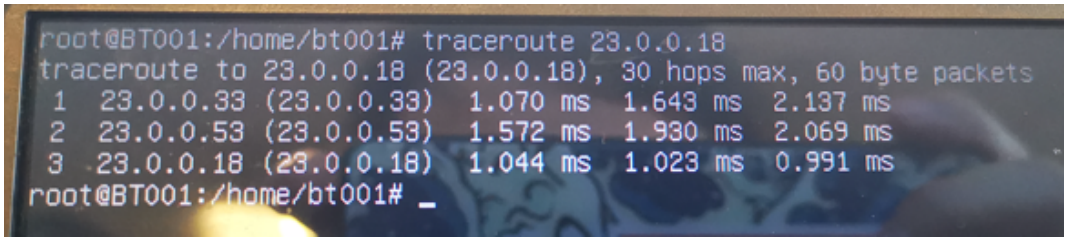
```
1 interface Loopback0
2 ip address 23.0.255.3 255.255.255.255
3 ipv6 address 2001:2300:FFFF:3::/128
4
5 router isis
6 net 49.0001.0230.0025.5003.00
7 is-type level-1
8 passive-interface Vlan4
9 passive-interface Vlan5
10 passive-interface Loopback0
11
12 interface FastEthernet0/0
13 ip router isis
14 ipv6 router isis
15
16 interface FastEthernet0/1
17 ip router isis
18 ipv6 router isis
19
20 interface Vlan2
21 ip router isis
22 ipv6 router isis
```

### 3.1.4 Evaluation

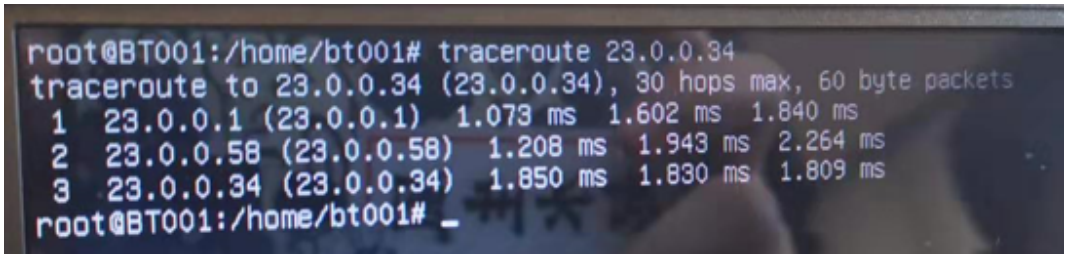
Once the IS-IS is set up, use `traceroute` command to check the IPv4 path from Laptop 1 (BT001, IPv4 Address: `23.0.0.2`) to Laptop 2 (BT002, IPv4 Address: `23.0.0.18`) in the network.

```
1 traceroute 23.0.0.34
```

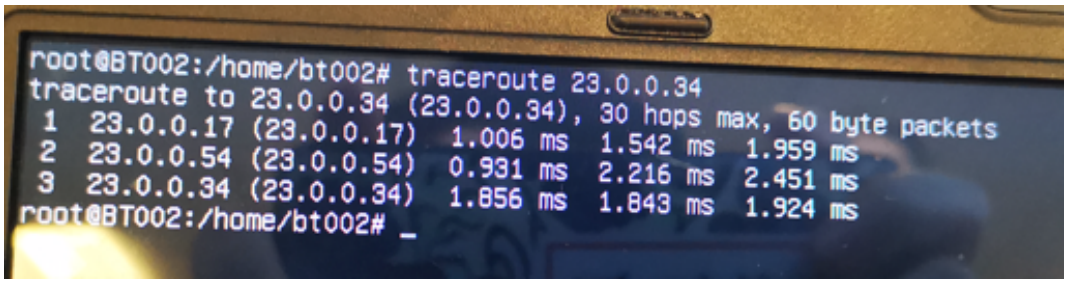
Figure 3.2 shows the route taken is `Laptop 1 (BT001, IPv4 Address: 23.0.0.2) -> Router 1 (BT-R001, IPv4 Address: 23.0.0.1) -> Router 2 (BT-R002, IPv4 Address: 23.0.0.50) -> Laptop 2 (BT002, IPv4 Address: 23.0.0.18)`, which is a correct route. Routes from Laptop 1 to Laptop 3 as well as from Laptop 2 to Laptop 3 are tested and shown in the figure as well.




(a) `traceroute` from Laptop 1 (BT001) to Laptop 2 (BT002)




(b) `traceroute` from Laptop 1 (BT001) to Laptop 3 (BT003)




(c) `traceroute` from Laptop 2 (BT002) to Laptop 3 (BT003)

Figure 3.2: Tracing IPv4 Routes between Laptops using `traceroute`.

Similiar results can be observed for IPv6 routes in Figure 3.3. The IPv6 Route taken from Laptop 1 to Laptop 2 is `Laptop 1 (BT001, IPv6 Address: 2001:2300::2) -> Router 1 (BT-R001, IPv6 Address: 2001:2300::1) -> Router 2 (BT-R002, IPv6 Address: 2001:2300:3:2::2) -> Laptop 2 (BT002, IPv6 Address: 2001:2300:0:1::2),` , which is also a correct route. Routes from Laptop 1 to Laptop 3 as well as from Laptop 2 to Laptop 3 are tested and shown in the figure as well.


(a) `traceroute` from Laptop 1 (BT001) to Laptop 2 (BT002)


(b) `traceroute` from Laptop 1 (BT001) to Laptop 3 (BT003)


(c) `traceroute` from Laptop 2 (BT002) to Laptop 3 (BT003)

Figure 3.3: Tracing IPv6 Routes between Laptops using `traceroute`.

To further evaluate the correctness of our implementation, we check the path from Laptop 1 to Laptop 2 under the condition that the physical connection between Router 1 and Router 2 is broken. The IS-IS protocol on Router 1 should be able to find route to Router 2 through Router 3.

Figure 3.4 shows the IPv4 route taken is `Laptop 1 (BT001, IPv6 Address: 23.0.0.2) -> Router 1 (BT-R001, IPv4 Address: 23.0.0.1) -> Router 3 (BT-R003, IPv4 Address: 23.0.0.58) -> Router 2 (BT-R002, IPv4 Address: 23.0.0.53) -> Laptop 2 (BT002, IPv4 Address: 23.0.0.18)`, which is a correct route.

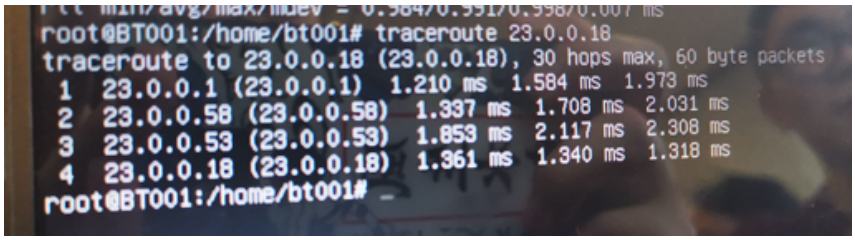


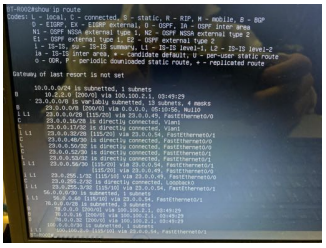

Figure 3.4: Tracing IPv4 Routes from Laptop 2 (BT002) to Laptop 3 (BT003) when Physical Connection between Router 1 (BT-R001) and Router 2 (BT-R002) is broken.

Figure 3.5 shows the IPv6 route taken is `Laptop 1 (BT001, IPv6 Address: 2001:2300::2) -> Router 1 (BT-R001, IPv6 Address: 2001:2300::1) -> Router 3 (BT-R003, IPv6 Address: 2001:2300:0:5::2) -> Router 2 (BT-R002, IPv6 Address: 2001:2300:0:4::1) -> Laptop 2 (BT002, IPv6 Address: 2001:2300:0:1::2)`, which is also a correct route.


Figure 3.5: Tracing IPv6 Routes from Laptop 2 (BT002) to Laptop 3 (BT003) when Physical Connection between Router 1 (BT-R001) and Router 2 (BT-R002) is broken.

On the router's side, `show ip route` and `show ipv6 route` is used to inspect the routes discovered by IS-IS protocol. In Figure 3.6 and 3.7, such routes to other subnets inside BT Network on each router are shown.

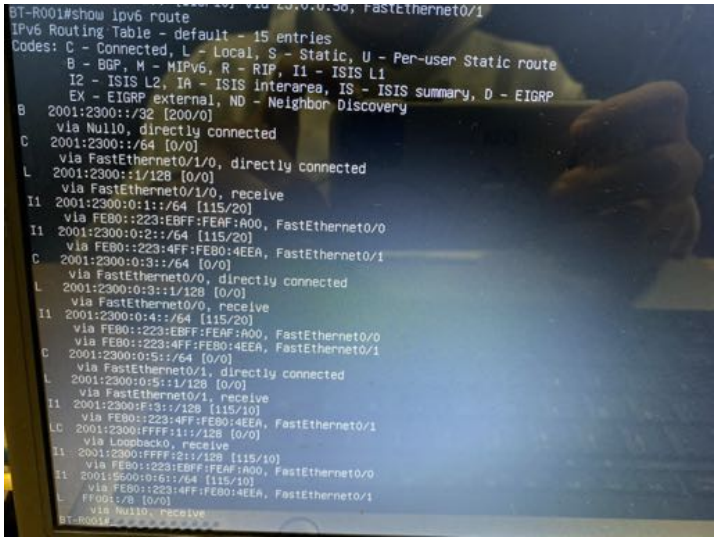

(a) Router 1 (BT-R001)



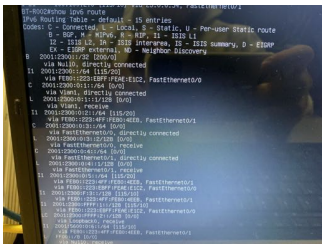
(b) Router 2 (BT-R002)

(c) Router 3 (BT-R003)

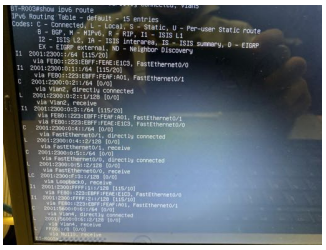
Figure 3.6: IPv4 Routes to Other Subnets on All 3 Routers Respectively using `show ip route`.




(a) Router 1 (BT-R001)



(b) Router 2 (BT-R002)



(c) Router 3 (BT-R003)

Figure 3.7: IPv6 Routes to Other Subnets on All 3 Routers Respectively using `show ipv6 route`.

Noticably, when the physical connection between Router 1 and Router 2 is broken, the route from Router 1 to Router 2 goes through Router 3 instead, as evident in Figure 3.8.

### 3.1.5 Commentary

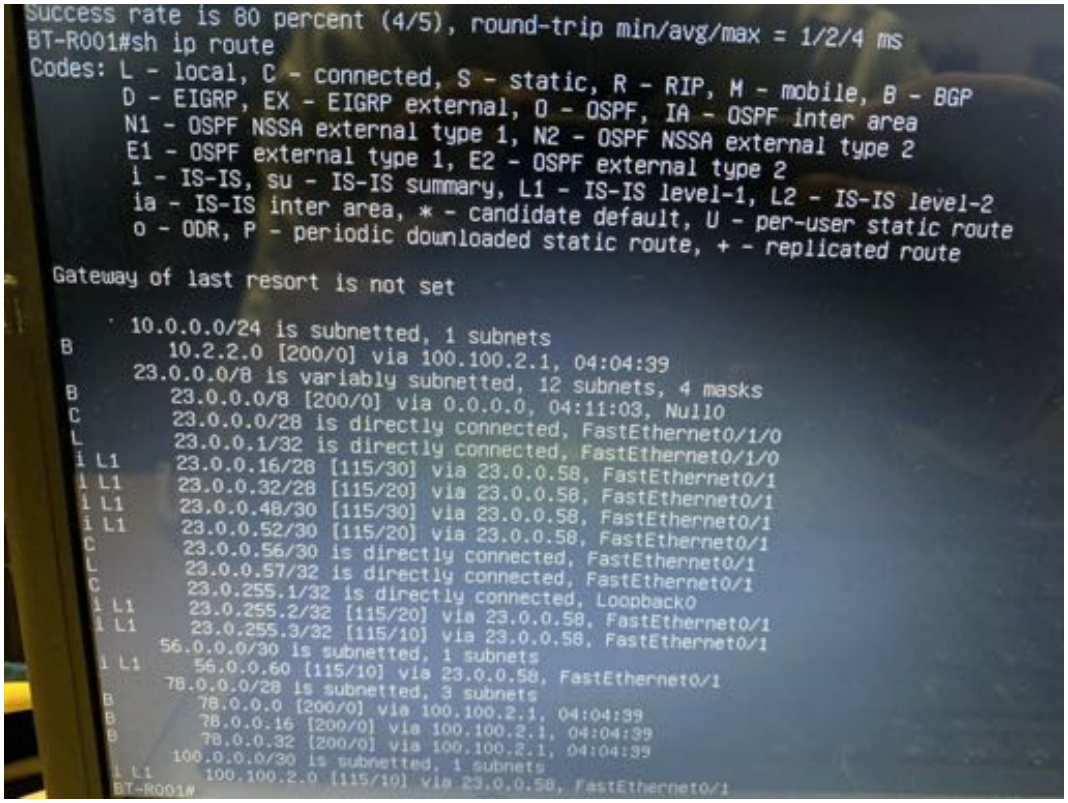
#### 3.1.5.1 Problem: IS-IS Not Set Up for Laptop-Router Interface

When initially setting up IS-IS on the interfaces, only the interfaces between routers have been turned on. This leads to laptop's failure to reach a router not directly connected. To solve this problem, IS-IS is set up on the interface between a laptop and a router.

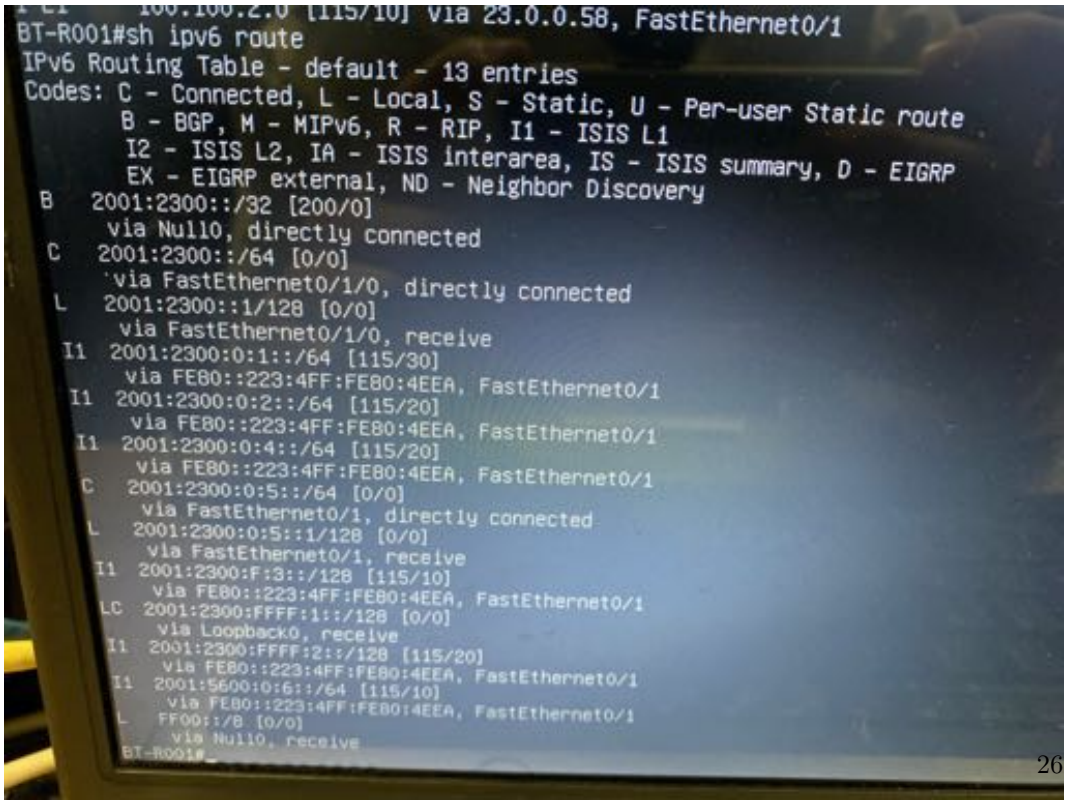
#### 3.1.5.2 Alternative Solution to Passive Interfaces

While turning interfaces connected to outside network into passive interfaces does enable other routers in the network to connect to outside networks through those interfaces, it demands extra computing resource for computing routes to external subnets.

**Alternatively, one can disable IS-IS on such interfaces and replace the external next-hop for each out-going route in BGP with the internal router to which the next-hop is directly connected to. That prevents Router-Neighbour subnet from participating in IS-IS routing and eliminates the extra computing consumption while preserving the connectivity between internal routers and outside networks.**




(a) IPv4 Routes to Other Subnets on Router 1 using `show ip route`.




(b) IPv6 Routes to Other Subnets on Router 1 using `show ip route`.

Figure 3.8: IP Routes to Other Subnets on Router 1 when Physical Connection between Router 1 (BT-R001) and Router 2 (BT-R002) is broken.

## 3.2 Inter-domain Routing Protocol: BGP

### 3.2.1 Design

For inter-domain routing, Border Gateway Protocol (BGP)[6] is applied in BT Network since it's the one and only External Gateway Protocol (EGP) in today's global Internet. BGP provides scaliability to large networks, clear definations of administrative boudnary as well as flexiable policy control, which allow business relaitonships with neighbouring ISPs to be expressed in terms of routing policies.

Figure 3.9 shows the design of BGP protocol in our network. The AS Number (ASN) of BT Network is 2030 while ASN of Central, Virgin, DT are 42, 5060, 3040 respectively.

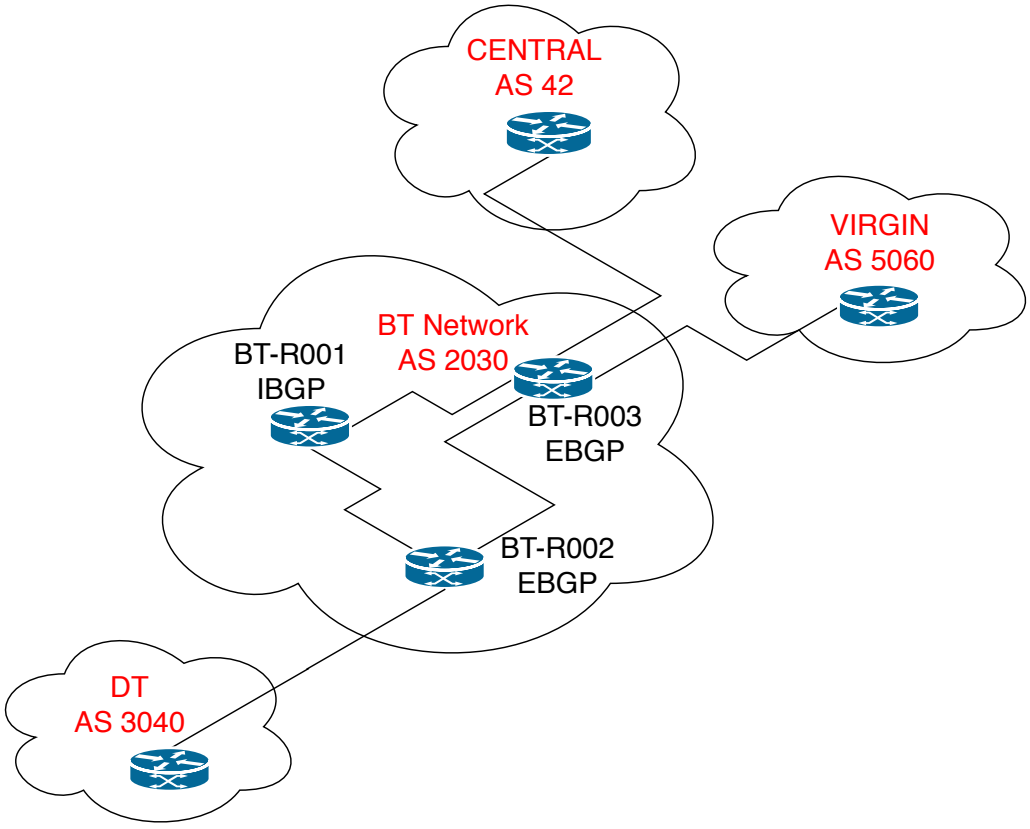



Figure 3.9: Design of BGP Protocol in BT Network.

Router 2 (BT-R002) and Router 3 (BT-R003) act as External BGP routers since they are directly connected to neighbouring ISPs. They receive routes announced by neighbouring ISPs' routers and announce routes originated from BT Network. Router 1 (BT-R001) acts

as an Internal BGP (IBGP) router only since it is not directly connected to any neighbouring ISP and only forwards and receives routes announced by EGBP routers.

### 3.2.2 Routing Policies

To enact business relationships with neighbouring ISPs, proper routing polices should be implemented for each different relationship in BGP protocol. Specifically, for customer ISPs, all routes announced by such ISP should be accepted and all routes received should be annouced to it as well. This allows customers to connect to BT Network as well as connect to other networks through BT Network.

For non-customer neighbouring ISPs (peers and providers), all routes announced by such ISP should be accepted in order for BT Network to connect to it. Meanwhile, only routes originated from either BT Network or BT's customers are announced to such ISP.

### 3.2.3 Implementation

On Router 1 (BT-R001, IPv4 Loopback Address: `23.0.255.1`, IPv6 Loopback Address: `2001:2300:FFFF:1::`), both Router 2 (BT-R002, IPv4 Loopback Address: `23.0.255.2`, IPv6 Loopback Address: `2001:2300:FFFF:2::`) and Router 3 (BT-R003, IPv4 Loopback Address: `23.0.255.3`, IPv6 Loopback Address: `2001:2300:FFFF:3::`) are taken as neighbouring routers in the same AS. In addition, the source of BGP messages are set to be the loopback address of Router 1 to prevent physical disconnection to the 2 routers. Router 1 announces the subnet `BT-R001 - BT001` (IPv4: `23.0.0.0/28`, IPv6: `2001:2300:0:0::/64`) to other routers.

```
router bgp 2030
network 23.0.0.0 mask 255.255.255.240
neighbor 23.0.255.2 remote-as 2030
neighbor 23.0.255.2 update-source Loopback0
neighbor 23.0.255.3 remote-as 2030
neighbor 23.0.255.3 update-source Loopback0
neighbor 2001:2300:FFFF:2:: remote-as 2030
neighbor 2001:2300:FFFF:2:: update-source Loopback0
neighbor 2001:2300:FFFF:3:: remote-as 2030
neighbor 2001:2300:FFFF:3:: update-source Loopback0

address-family ipv6
network 2001:2300::/64
neighbor 2001:2300:FFFF:2:: activate
neighbor 2001:2300:FFFF:3:: activate
```

On Router 2 (BT-R002), both Router 1 and Router 3 are taken as neighbouring routers in the same AS while the router from DT is taken as router from AS 3040. Router 2 announces

the subnet BT-R002 - BT002 (IPv4: 23.0.0.16/28, IPv6: 2001:2300:0:1::/64) to other routers. Since Router 2 is direclty connected to customer DT Network, it applies no filter on inbound and outbound routes.

```
router bgp 2030
network 23.0.0.16 mask 255.255.255.240
neighbor 23.0.0.62 remote-as 3040
neighbor 23.0.255.1 remote-as 2030
neighbor 23.0.255.1 update-source Loopback0
neighbor 23.0.255.3 remote-as 2030
neighbor 23.0.255.3 update-source Loopback0
neighbor 2001:2300:0:6::2 remote-as 3040
neighbor 2001:2300:FFFF:1:: remote-as 2030
neighbor 2001:2300:FFFF:1:: update-source Loopback0
neighbor 2001:2300:FFFF:3:: remote-as 2030
neighbor 2001:2300:FFFF:3:: update-source Loopback0

address-family ipv6
network 2001:2300:0:1::/64
neighbor 2001:2300:0:6::2 activate
neighbor 2001:2300:FFFF:1:: activate
neighbor 2001:2300:FFFF:3:: activate
```

On Router 3 (BT-R003), both Router 1 and Router 2 are taken as neighbouring routers in the same AS while the routers from Central and Virgin are taken as router from AS 42 and AS 5060 respectively. Router 3 announces the subnet BT-R003 - BT003 (IPv4: 23.0.0.32/28, IPv6: 2001:2300:0:2::/64) to other routers.

```
router bgp 2030
network 23.0.0.32 mask 255.255.255.240
neighbor 23.0.255.1 remote-as 2030
neighbor 23.0.255.1 update-source Loopback0
neighbor 23.0.255.2 remote-as 2030
neighbor 23.0.255.2 update-source Loopback0
neighbor 2001:2300:FFFF:1:: remote-as 2030
neighbor 2001:2300:FFFF:1:: update-source Loopback0
neighbor 2001:2300:FFFF:2:: remote-as 2030
neighbor 2001:2300:FFFF:2:: update-source Loopback0
neighbor 2001:5600:0:6::1 remote-as 5060
neighbor 56.0.0.61 remote-as 5060
neighbor 100.100.2.1 remote-as 42

address-family ipv6
network 2001:2300:0:2::/64
aggregate-address 2001:2300::/32 summary-only
neighbor 2001:2300:FFFF:1:: activate
neighbor 2001:2300:FFFF:2:: activate
```

Since Router 3 is direclty connected to non-customer ISPs, it applies a filter on outbound routes, which denies all routes that pass through either Central Network (ASN: 42) or Virgin Network (ASN: 5060).

```
ip as-path access-list 1 deny _42_
ip as-path access-list 1 deny _5060_
ip as-path access-list 1 permit .*
router bgp 2030
neighbor 56.0.0.61 filter-list 1 out
neighbor 100.100.2.1 filter-list 1 out
```

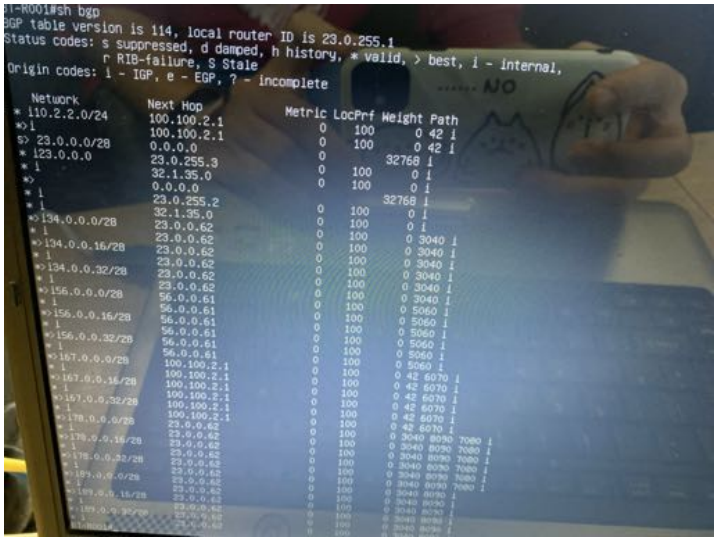
Address aggregation for `23.0.0.0/8` and `2001:2300::/32` is set up on all 3 routers, which aggregates routes destined for all addresses inside BT Network range into a single route.

```
aggregate-address 23.0.0.0 255.0.0.0 summary-only
address-family ipv6
aggregate-address 2001:2300::/32 summary-only
```

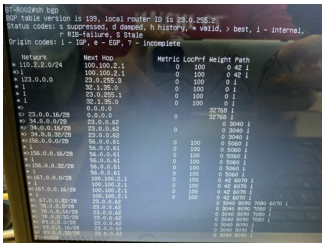
### 3.2.4 Evaluation

#### 3.2.4.1 BGP Routes

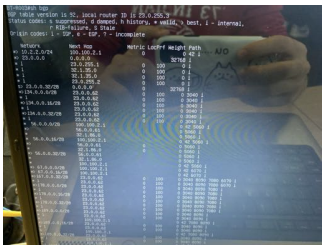
Routes collected through BGP protocol on all 3 routers are shown in Figure 3.10 and **??** using commands `show bgp` and `show bgp ipv6`. Routes to DT Network (IPv4: `34.0.0.0/8`, IPv6: `2001:3400::/32`), Virgin Network (IPv4: `56.0.0.0/8`, IPv6: `2001:5600::/32`), Central Network (IPv4: `10.2.2.0/24`) and other networks can be observed in the figure.




(a) Router 1 (BT-R001)



(b) Router 2 (BT-R002)

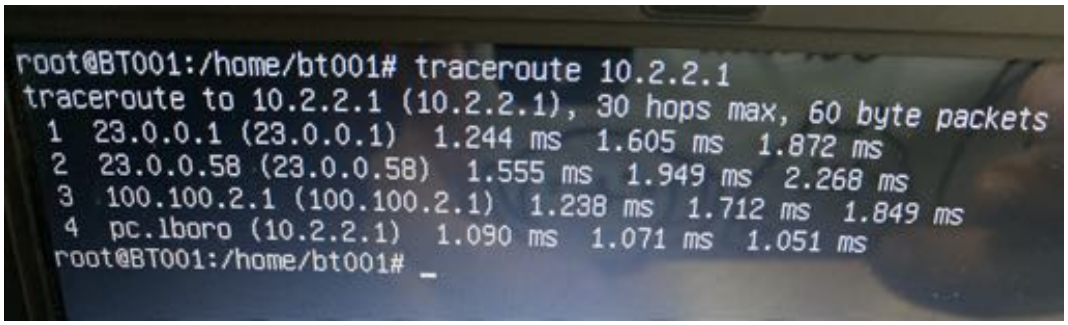


(c) Router 3 (BT-R003)

Figure 3.10: IPv4 Routes Collected through BGP Protocols on All 3 Routers using `show bgp`.

#### 3.2.4.2 Connectivity to Provider Central Network

The connectivity to provider Central Network using BGP protocol is tested and evaluated by tracing routes to IP addresses `10.2.2.1` on Laptop 1 (BT001). As shown in Figure 3.11, connection to Central Network is successfully established through BGP routes.

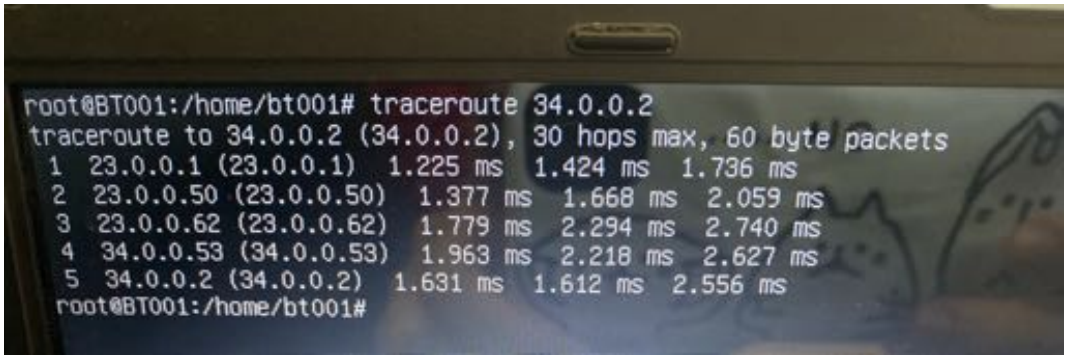



(a) `10.2.2.1`

Figure 3.11: Tracing IPv4 Routes to Central Network on Laptop 1 (BT001) using `traceroute`.

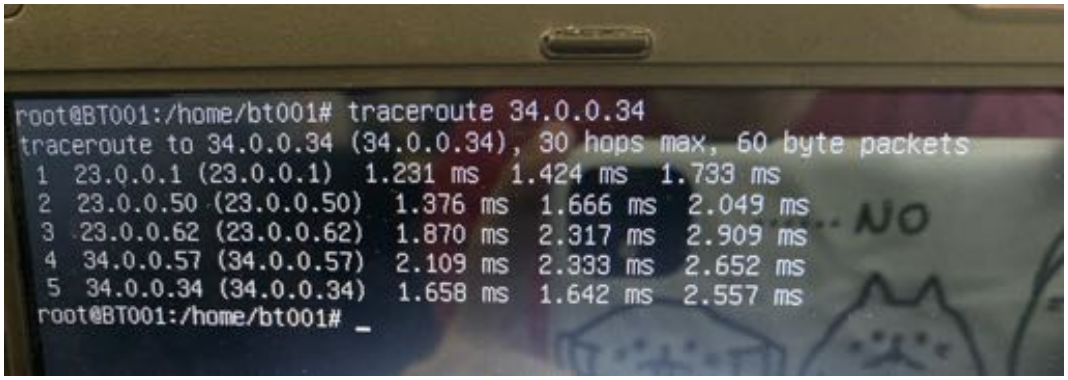
#### 3.2.4.3 Connectivity to Customer DT Network

The connectivity to customer DT Network using BGP protocol is tested and evaluated by tracing routes to IP addresses `34.0.0.2`, `34.0.0.18` and `34.0.0.34` on Laptop 1 (BT001). As shown in Figure 3.12, connection to DT Network is successfully established through BGP routes.




(a) `34.0.0.2`


(b) `34.0.0.18`

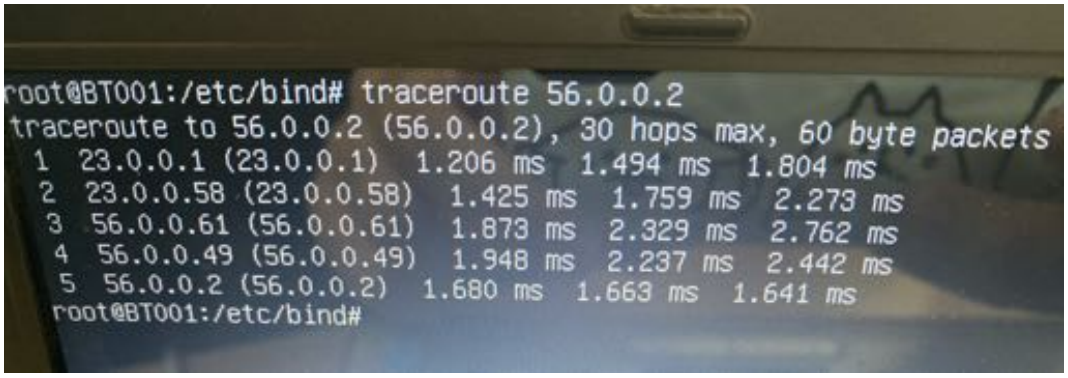



(c) `34.0.0.34`

Figure 3.12: Tracing IPv4 Routes to DT Network on Laptop 1 (BT001) using `traceroute`.

#### 3.2.4.4 Connectivity to Peer Virgin Network

The connectivity to peer Virgin Network using BGP protocol is tested and evaluated by tracing routes to IP addresses `56.0.0.2`, `56.0.0.18` and `56.0.0.34` on Laptop 1 (BT001). As shown in Figure 3.14, connection to Virgin Network is successfully established through BGP routes.




(a) `56.0.0.2`

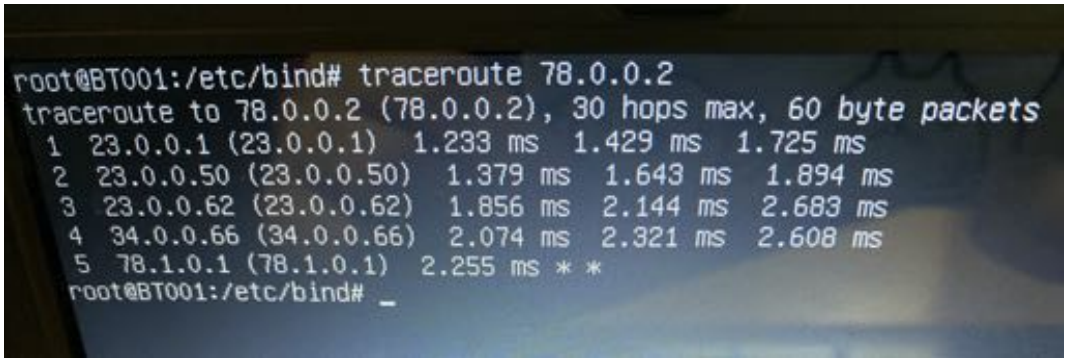

(b) `56.0.0.18`


(c) `56.0.0.34`

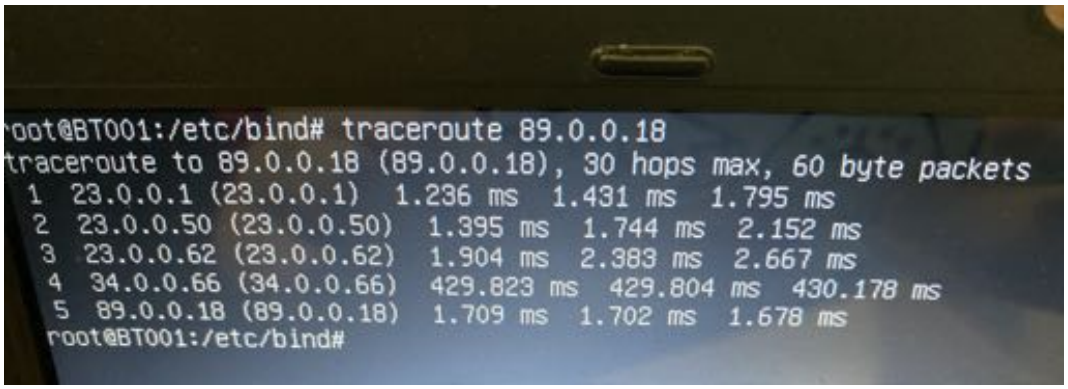
Figure 3.13: Tracing IPv4 Routes to Virgin Network on Laptop 1 (BT001) using `traceroute`.

#### 3.2.4.5 Connectivity to Other Networks

The connectivity to other networks using BGP protocol is tested and evaluated by tracing routes to IP addresses `78.0.0.2` (Sonara Network) and `89.0.0.18` (NTT Network) on Laptop 1 (BT001). As shown in Figure 3.14, connection to Virgin Network is successfully established through BGP routes.




(a) `78.0.0.2`




(b) `89.0.0.18`

Figure 3.14: Tracing IPv4 Routes to Other Networks on Laptop 1 (BT001) using `traceroute`.

#### 3.2.4.6 Connectivity to Virgin when Direct Physical Connection Is Down

In addition, the connectivity to peer Virgin Network under the unfortunate condition that the direct physical connection is down is also tested. Traced routes to IP addresses `56.0.0.2`, `56.0.0.18` and `56.0.0.34` on Laptop 1 (BT001) are shown in Figure 3.15.

Alternative connection to Virgin Network is successfully established through Central Network (ASN: 42).

### 3.2.5 Commentary

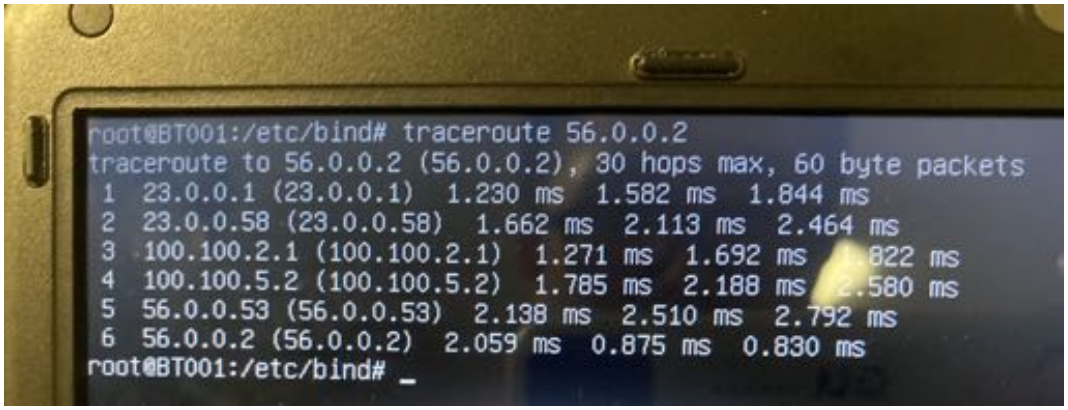
#### 3.2.5.1 Problem: Filter List Not Working for Self-Originated Routes

The initial filter list for outbound routes on Router 3 only permits routes originated from BT Network (ASN 2030) and customer DT Network (ASN 3040).

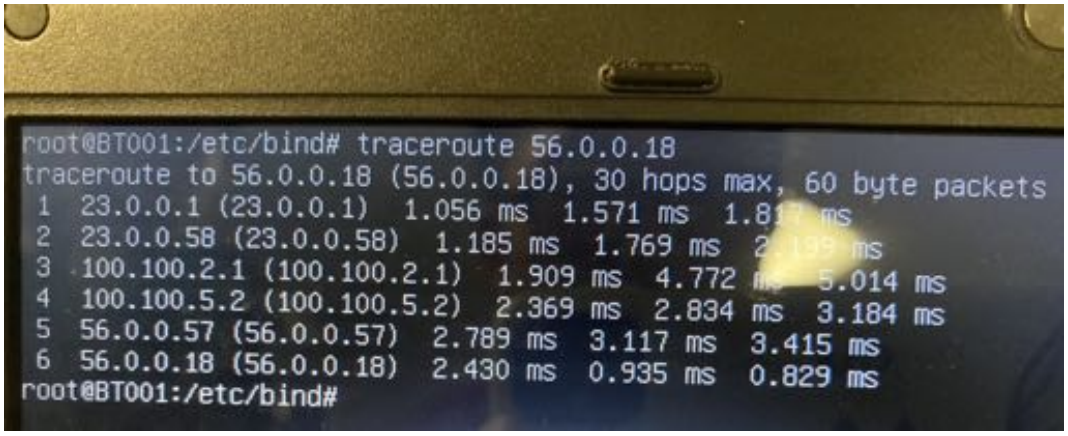
```
ip as-path access-list 1 permit _2030$
ip as-path access-list 1 permit _3040$
```

However, the filter list blocks all routes except those originated from customer DT Network to be announced. To solve this problem, a filter list where all routes except those go through provider Central Network (ASN 42) and peer Virgin Network (ASN 5060) are allowed. The new list should have the same effects as the previous list and indeed works as intended.

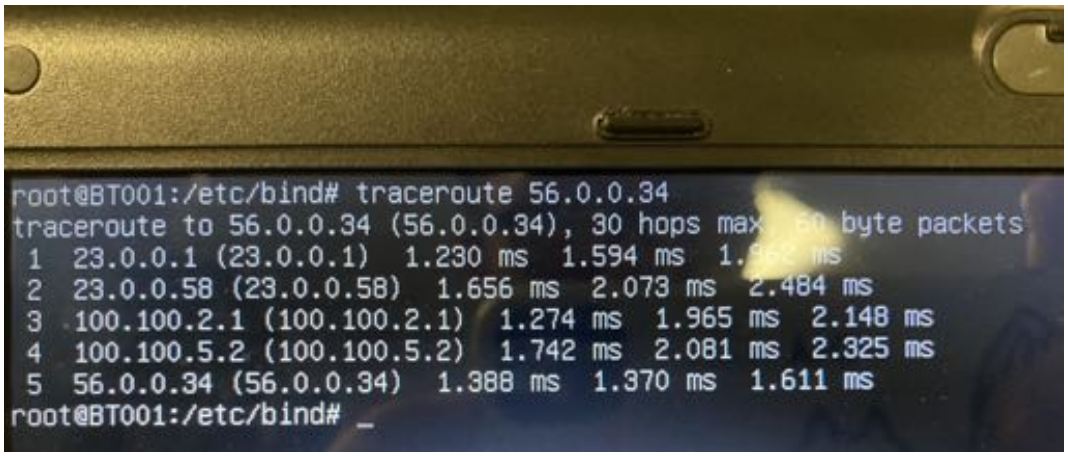
```
ip as-path access-list 1 deny _42_
ip as-path access-list 1 deny _5060_
ip as-path access-list 1 permit .*
```




(a) `56.0.0.2`




(b) `56.0.0.18`




(c) `56.0.0.34`

Figure 3.15: Tracing IPv4 Routes to Virgin Network on Laptop 1 (BT001) using `traceroute` When Direct Physical Connection Is Broken .

# Chapter 4

# Applications in the Network

## 4.1 Secure Remote Access to Routers through SSH

### 4.1.1 Design

Accessing the routers through the physical "console" port is inconvenient and dangerous. Thus, remote acess through Secure Shell (SSH) protocol[7] to routers is needed.

In BT network, remote SSH access is enabled on all 3 routers. Seperate combinations of username and password on each router are used to ensure the independence of security of each router.

In addition, SSH public key authentication is set up on Laptop 1 (BT001), which allows the root user on the laptop to login in to all routers without entering passwords.

### 4.1.2 Implementation

We first set up Remote SSH access was first set up as instructed in Reference Guide on all 3 routers. Below is the configuration commands for Router 1 (BT-R001).

```
hostname BT-R001
ip domain name bt.lboro
username r001 priv 15 secret <secret>
line vty 0 4
transport input ssh telnet
login local

ip ssh version 2
crypto key generate rsa general-keys
ip ssh dh min size 4096
```

We then generate a pair of public and private keys on Laptop 1 (BT001).

```
ssh-keygen
```

After that, the pair of keys is written into files `~/.ssh/id_rsa` and `~/.ssh/id_rsa.pub` . We use the generated public key ( `id_rsa.pub` ) to set up SSH public key authentication on all 3 routers.

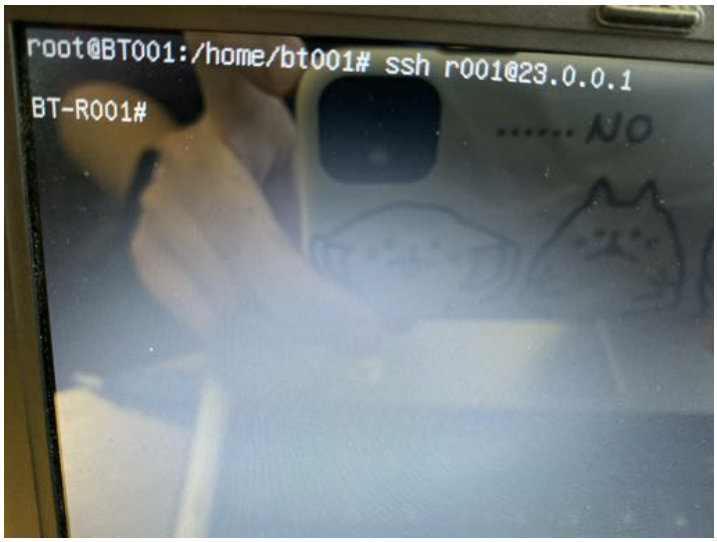
```
ip ssh pubkey-chain
username r001
key-string
```

### 4.1.3 Evaluation

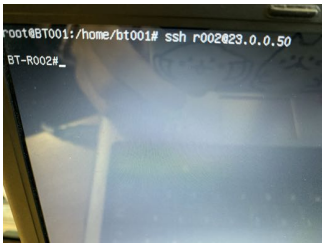
Once remote SSH access is set up on 3 routers, one should be able to access them on Laptop 1 (BT001) without entering the password using the following commands.

```
# access Router 1
ssh r001@23.0.0.1
# access Router 2
ssh r002@23.0.0.50
# access Router 3
ssh r003@23.0.0.33
```

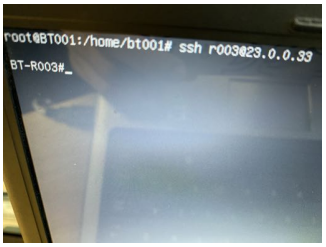
Screenshots of successful remote access to all 3 routers are shown in Figure 4.1.




(a) Router 1 (BT-R001)




(b) Router 2 (BT-R002)




(c) Router 3 (BT-R003)

Figure 4.1: Sucessful remote SSH access to all 3 routers from Laptop 1 (BT001).

### 4.1.4 Commentary

#### 4.1.4.1 Problem: Maximum Limit of Characters per Line

When we tried to set up SSH public key authentication on routers, we failed at our initial attempt. It turned out that Cisco router has maximum limit of characters for each command line. Thus, a public key in a single long line was not accepted by the router.

To solve this problem, `fold` command is used to split the public key into multiple lines before re-uploading the key and SSH public key authentication was successfully set up on the router.

## 4.2 Domain Name System Service

### 4.2.1 Design

Domain Name System (DNS)[8] translates IP addresses to domain names and vice versa. In contrast to hard-to-remember IP addresses, short and meaningful domain names (eg. `lboro.ac.uk` for Loughborough University in UK) are more convient for Internet users. Additionally, the service providers can change the IP addresses of servers without re-notifying their customers.

In BT Network, there are two DNS servers, one primary master server and the other secondary. Primary master server is deployed at Laptop 3 (BT003) and secondary master server is at Laptop 1 (BT001). The rationale is that when the primary becomes unavailable, the secondary can be the backup domain name server.

Both are authoritative of domain `bt.lboro`. **Each laptop in the network has a corresponding domain name. For example, the domain name of Laptop 1 (BT001) is `bt001.bt.lboro`. In addition, both `A` and `AAAA` records of `bt.lboro` point to Laptop 1 while the MX record points to Laptop 3.**

The two DNS servers are also connected to the central DNS server, which is authoritative of domain `lboro`.

### 4.2.2 Implementation

Install `bind9` package and `dnsutils` package on Laptop 1 (BT001) and Laptop 3 (BT001) using the following commands. All DNS configurations are all stored in folder `/etc/bind`.

```
1 sudo apt-get install bind9
2 sudo apt-get install dnsutils
```

On the primary DNS server, forward unknown DNS requests to central DNS server by adding this line to file `named.conf.options`.

```
1 forwarders { 10.2.2.1; };
```

The following lines are added to file `/etc/bind/named.conf.local`.

The zone section defines the type of the DNS server and it is stored in a file mentioned in the 'file' field. The 'allow-transfer' field defines a match list which has IP addresses that are allowed to do transfer and copy operations to the zone information with the server. The 'allow-notify' field defines an IP addresses match list that is allowed to notify this server and implicitly update the zone. In this case, both fields should be the IP address of the secondary DNS server (Laptop 1).

The first zone is a forward zone which translate the domain name to IP address. The zone name of it is the selected domain name `bt.lboro`. The second zone is the reverse zone which translate the IP address to domain name. The zone name of it should be the fixed IP prefix part and host part. In this case, the name is `23.in-addr.arpa`.

```
zone   bt.lboro  {
  type master;
  file    /etc/bind/db.bt.lboro ;
  allow-transfer { 23.0.0.2; };
  also-notify { 23.0.0.2; };
};

zone   23.in-addr.arpa  {
  type master;
  file    /etc/bind/db.23   ;
  allow-transfer { 23.0.0.2; };
  also-notify { 23.0.0.2; };
};
```

Then, the files related to forward zone and reverse zone are edited. The file `db.bt.lboro` whose contents are shown in Listing 4.1 defines the forward DNS configuration.

Listing 4.1: Contents of Forward DNS Configuration File Located at `/etc/bind/db.bt.lboro` on Primary DNS Server (Laptop 3).

```
;
; BIND data file for BT.LBORO
;
$TTL  604800
@ IN  SOA bt.lboro. root.bt.lboro. (
            13  ; Serial
       604800   ; Refresh
        86400   ; Retry
      2419200   ; Expire
       604800 ) ; Negative Cache TTL
;
@ IN  NS  ns.bt.lboro.
@ IN  NS  ns2.bt.lboro.
www IN  CNAME bt.lboro.
ns  IN  A 23.0.0.34
ns  IN  AAAA  2001:2300:0:2::2
ns2 IN  A 23.0.0.2
ns2 IN  AAAA  2001:2300:0:0::2
@ IN  A 23.0.0.2
@ IN  AAAA  2001:2300:0:0::2
@ IN  MX  10  bt003.bt.lboro.
bt001 IN  A 23.0.0.2
```

```
bt001 IN  AAAA  2001:2300:0:0::2
bt002 IN  A 23.0.0.18
bt002 IN  AAAA  2001:2300:0:1::2
bt003 IN  A 23.0.0.34
bt003 IN  AAAA  2001:2300:0:2::2
```

Serial number is the version number of this file and it should be increased after changing the file. For each line of records, the prefix (eg. `bt001`), domain type (eg. `A`) and value (eg. `23.0.0.2`) are specified.

Listing 4.2: Contents of Reverse DNS Configuration File Located at `/etc/bind/db.23` on Primary DNS Server (Laptop 3).

```
;
; BIND reverse data file for 23.xxx.xxx.xxx net
;
$TTL  604800
@ IN  SOA ns.bt.lboro. root.bt.lboro. (
            7   ; Serial
       604800   ; Refresh
        86400   ; Retry
      2419200   ; Expire
       604800 ) ; Negative Cache TTL
;
@ IN  NS  ns.
2.0.0 IN  PTR bt.lboro.
34.0.0  IN  PTR ns.bt.lboro.
2.0.0 IN  PTR ns2.bt.lboro.
2.0.0 IN  PTR bt001.bt.lboro.
18.0.0  IN  PTR bt002.bt.lboro.
34.0.0  IN  PTR bt003.bt.lboro.
```

Listing 4.2 shows the configuration for the reverse zone. The serial number needs to be increased on each change as well. For each line of records, the first column of match list should be reverse host part of IP address. For example, the IP address of Laptop 1 (BT001) is `23.0.0.2` and the host part of it is `0.0.2`. Therefore, the reverse host part of it should be `2.0.0`.

For each A record in forward configurations, a PTR records should be added to the reverse configuration.

When all configurations for DNS server are completed, use following commands to restart `bind9` service to take effect.

```
service bind9 restart
```

As for secondary master, Laptop 1 (BT001) only needs to add the following lines to the file `/etc/bind/named.conf.local`, which specify both forward and reverse DNS zones and

Laptop 3 as their master server.

```
zone   bt.lboro  {
      type slave;
      file   db.bt.lboro ;
      masters { 23.0.0.34; };
};

zone   23.in-addr.arpa  {
      type slave;
      file   db.23   ;
      masters { 23.0.0.34; }
};
```

Restart `bind9` service for configurations to take effect.

```
service bind9 restart
```

Now, Laptop 1 should be able to receive DNS records from the master server and act as a secondary DNS server.

On client's side, Laptop 3 and 2 are set as nameservers in file `/etc/network/interfaces` on all 3 laptops.

```
dns-nameservers 23.0.0.34 23.0.0.2
```

Full configurations for both primary and secondary DNS server are detailed in Appendix D.

### 4.2.3 Evaluation

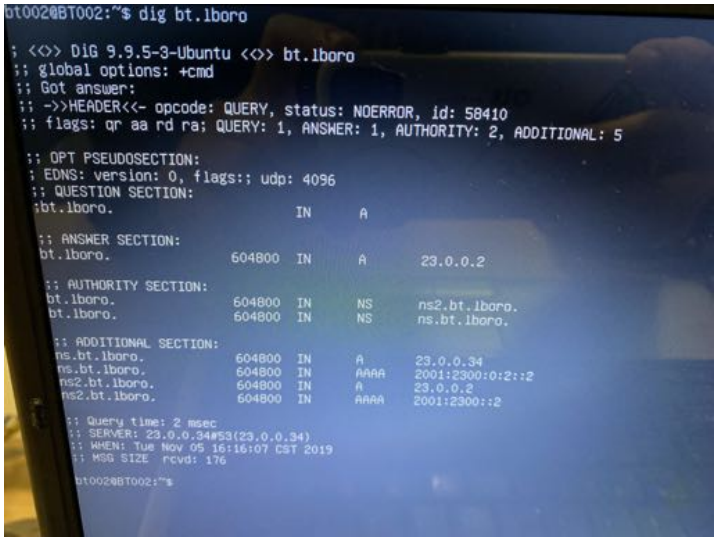
The command `dig` is used to query DNS records on Laptop 2 for evaluation. In Figure 4.2, DNS records of `bt.lboro` and `www.bt.lboro` are "digged" and correct answers are returned by DNS servers.

In addition, DNS records of `bt001.bt.lboro`, `bt002.bt.lboro` and `bt003.bt.lboro` are "digged" on Laptop 2 as well. In Figure 4.3, correct answers are returned by DNS servers.

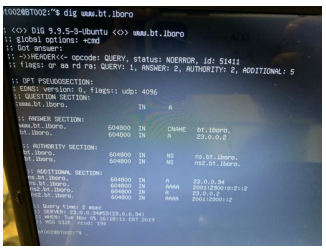
In terms of IPv6 addresses, DNS `AAAA` records of `bt001.bt.lboro`, `bt002.bt.lboro` and `bt003.bt.lboro` are "digged" on Laptop 2 as well. In Figure 4.4, correct answers are returned by DNS servers.

For reverse DNS, records of Laptop 1 (`23.0.0.2`), Laptop 2 (`23.0.0.18`) and Laptop 3 (`23.0.0.34`) are "digged" on Laptop 2 as well. In Figure 4.5, correct answers are returned by DNS servers.

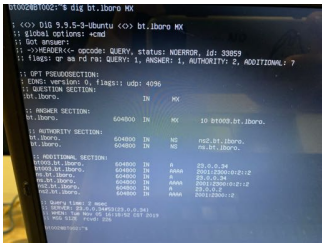
In terms of connectivity, all three laptops are able to directly `ping` the domain name `bt.lboro`, whose DNS `A` record points to Laptop 1 (BT001, IPv4 Address: `23.0.0.2`), as shown in Figure 4.6.




(a) `bt.lboro`

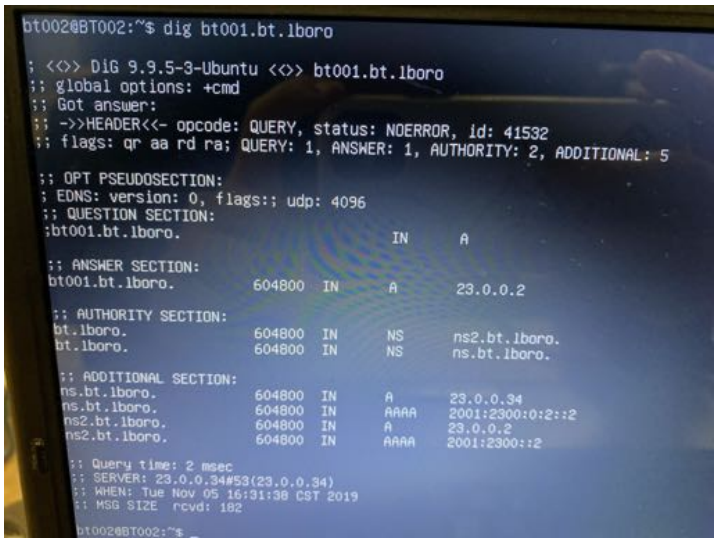


(b) `www.bt.lboro`

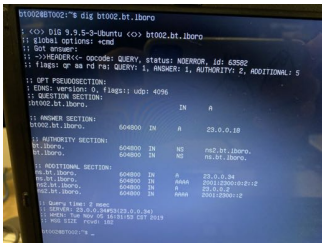


(c) `MX` Records of `bt.lboro`

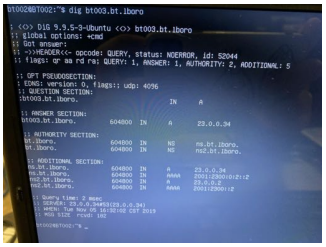
Figure 4.2: Querying DNS Records for `bt.lboro` and `www.bt.lboro` on Laptop 2.




(a) `bt001.bt.lboro`

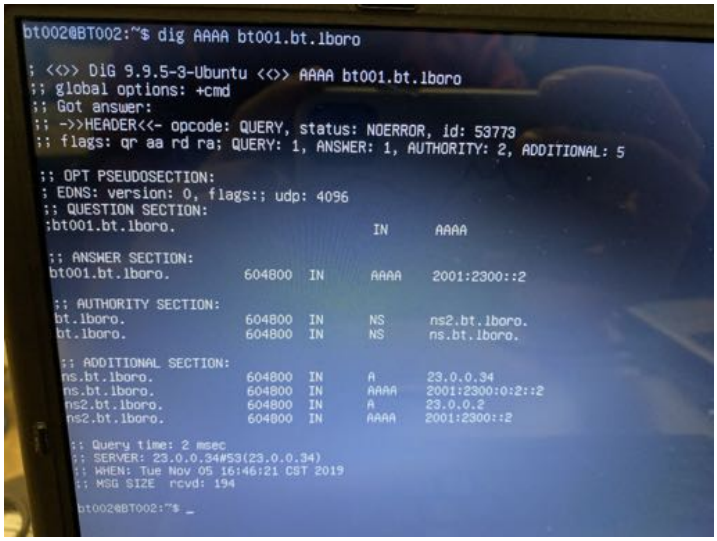


(b) `bt002.bt.lboro`

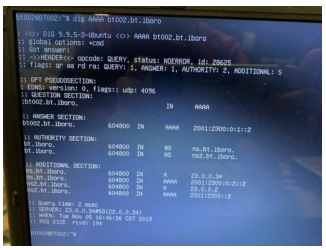


(c) `bt003.bt.lboro`

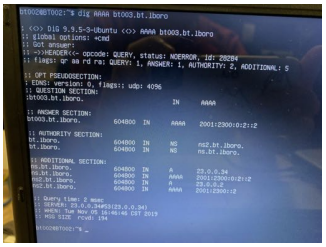
Figure 4.3: Querying DNS Records for sub-domains of `bt.lboro` on Laptop 2.




(a) `bt001.bt.lboro`

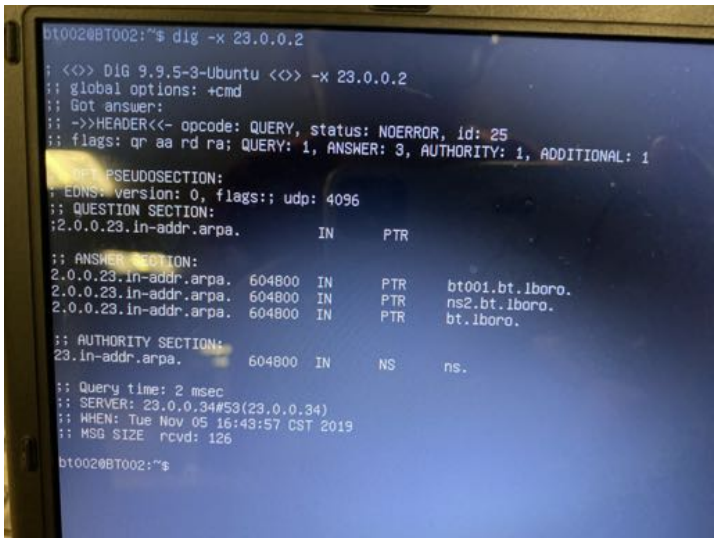


(b) `bt002.bt.lboro`

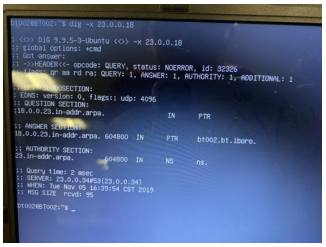


(c) `bt003.bt.lboro`

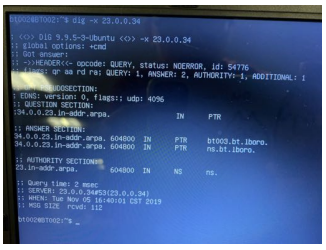
Figure 4.4: Querying DNS `AAAA` Records for sub-domains of `bt.lboro` on Laptop 2.




(a) `23.0.0.2`

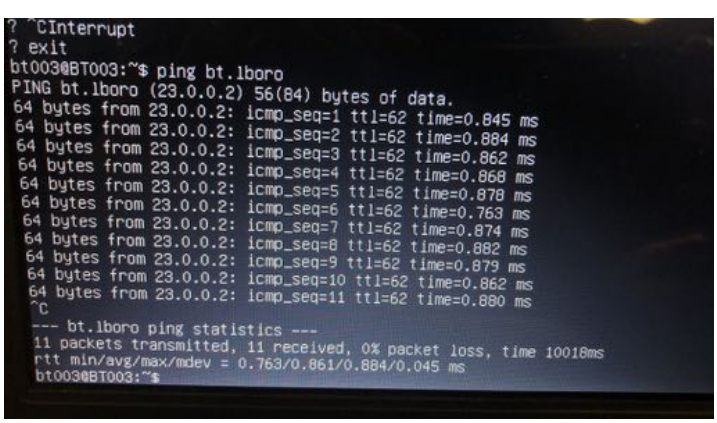


(b) `23.0.0.18`

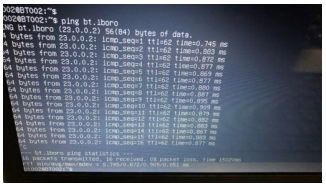


(c) `23.0.0.34`

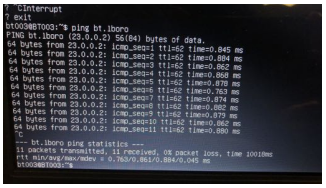
Figure 4.5: Querying Reverse DNS Records for All 3 Laptops on Laptop 2.




(a) Laptop 1 (BT001)



(b) Laptop 2 (BT002)



(c) Laptop 3 (BT003)

Figure 4.6: Connectivity to `bt.lboro` using `ping` on all 3 Laptops.

Under the unfortunate condition that the primary server (Laptop 3) is disconnected from the Internet, availability of DNS service in BT is also tested. As shown in Figure 4.7, DNS records are successfully retrived from the secondary DNS server (Laptop 2, IPv4 Address: `23.0.0.2`) despite the fact that the primary server is down.

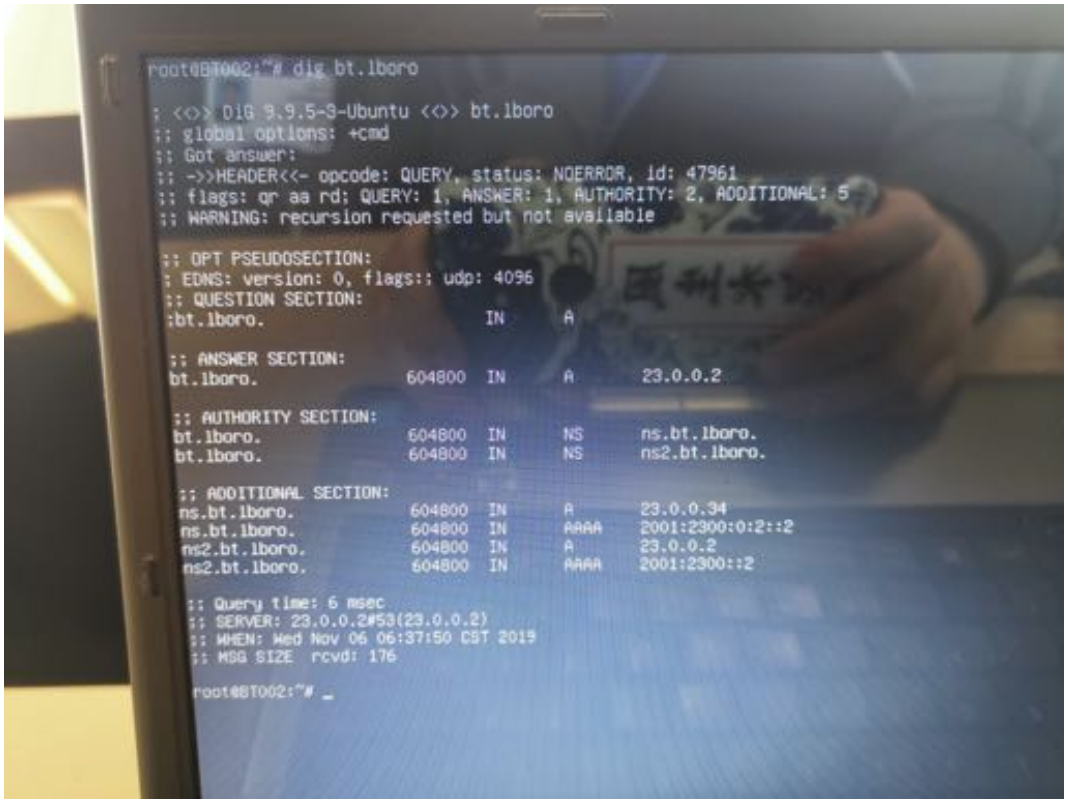



Figure 4.7: Querying DNS Records for `bt.lboro` on Laptop 2 When Primary DNS Server (Laptop 3) Is Down.

### 4.2.4 Commentary

#### 4.2.4.1 Problem: Querying IPv6 DNS Records

Using `dig` directly does not return IPv6 addresses for domains. It turns out that `dig` queries `A` records by default. To query IPv6 DNS records, `dig AAAA domain-name` should be used.

#### 4.2.4.2 Problem: Reverse DNS Not Working

Using `dig -x 23.0.0.2` does not return any answer initially. It's later realized that the reverse DNS zone name has been wrongly set to `0.0.23.in-addr.arpa`.

To set up reverse DNS properly, the DNS zone should be `3.in-addr.arpa`. In addition, the first column for each DNS record inside file `db.23` should be the reverse host name (eg. `2.0.0` for IP address `23.0.0.2`).

## 4.3 World Wide Web Service

### 4.3.1 Design

A World Wide Web (WWW) service[9][10] in the network allows any terminal devices to access the deployed webpages. The service in BT Network is established after routers and three laptops have been configurated and DNS service been set up.

For this lab, `apache2` package is chosen as the tool to establish the web server on Laptop 1 (BT001). Using the default settings of this package is enough and it has a specific folder static web pages are stored.

### 4.3.2 Implementation

Install `apache2` package on Laptop 1 (BT001, IPv4 Address: 23.0.0.2) using following the command.

```
sudo apt-get install apache2
```

A HTML file named `index.html` is created as a test webpage, as detailed in Figure 4.8.

```
<html>
<header><title>BT Network</title></header>
<body>
<h1>Welcome to BT Network |</h1>
A Trustworthy Internet Service Provider at Loughborough University.
</body>
</html>
```

Figure 4.8: Contents of HTML File Named `index.html`

Then, the HTML file is copied to the folder `/var/www/html`. This folder is used to deploy webpages on the server. And the meaning of -r is to cover the same name file.

```
cp -r index.html /var/www/html
```

Then, install `links` package which is a command-line Web browser in Linux by using following command.

```
sudo apt-get install links
```

Then, `links` is started and the web page is accessed through URL with the local IP address.

```
links http://23.0.0.2/
```

Finally, add the following lines to the forward DNS file `/etc/bind/db.bt.lboro` on Primary DNS Server at Laptop 3 (BT003). This enables the clients to browse the webpages by domain name `bt.lboro`.

```
@    IN     A    23.0.0.2
@    IN     AAAA   2001:2300:0:0::2
```

### 4.3.3 Evaluation

The web service is tested using the following command on Laptop 2, which is neither a DNS Server not a Web Server.

```
links http://bt.lboro/
```

As shown in Figure 4.9, the service has been successfully set up and is accessible in the network.


Figure 4.9: Web Service Provided at `http://bt.lboro` .

## 4.4 Email Service

### 4.4.1 Design

Email service[11] allows users to communicate with each other based on texts. In BT Network, it is set up using `exim4` package on Laptop 3 (BT003).

### 4.4.2 Implementation

On Laptop 3 (BT003), the package `exim4` is installed and configured. Important steps in the configuration are shown in 4.10.

```
apt-get install exim4
dpkg-reconfigure exim4-config
```

The full configuration of Email service is detailed in Appendix E.

### 4.4.3 Evaluation

For evaluation, the following command are used to send a mail to DT Network.

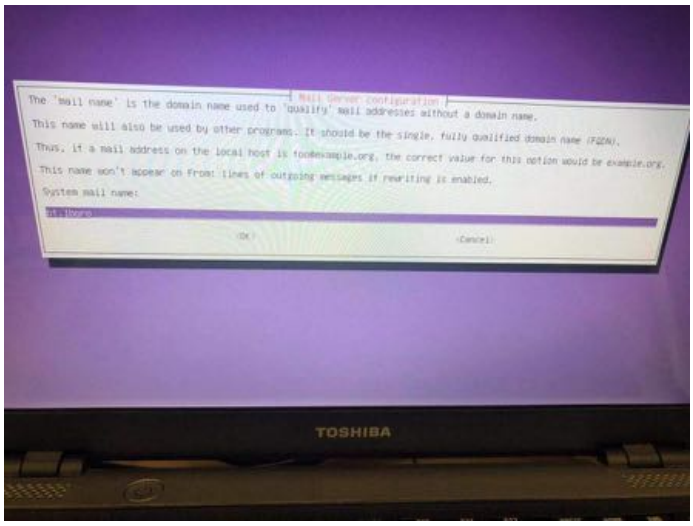
```
echo "mua." | sendmail -v mail@dt3.lboro
```

The mail can be seen arriving the destination in Figure 4.11.

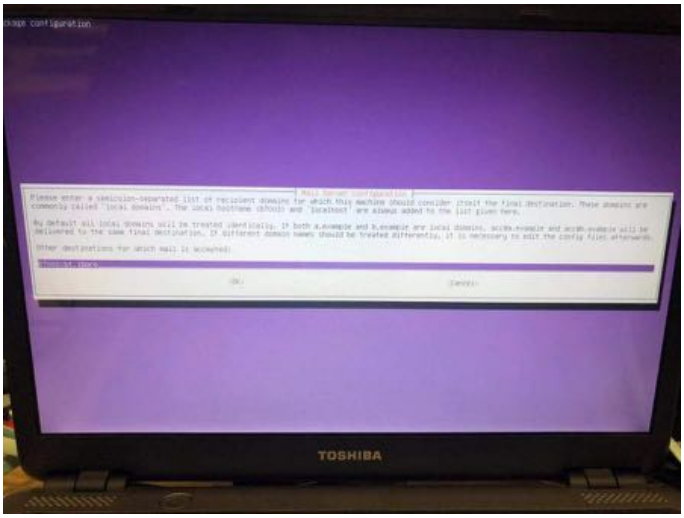
### 4.4.4 Commentary

#### 4.4.4.1 Problem: Setting Up Domain Name

At the very beginning, when we were configurating `exim4`, we just skipped the step and forgot to set up the domain name. Later, we found that the mail cannot be sent to other groups. The domain name is reset as `bt.lboro` to solve the problem.




(a) Setting Up Mail Name.




(b) Setting Up Domain Name.

Figure 4.10: Important Configuration Steps for `exim4`.


(a) SMTP Sucess Message Is Returned from DT Network on Laptop 3 (BT003).


(b) Email Is Received on DT Network's Side.

Figure 4.11: Email Can Be Seen Arrived on Both Sides.

# Chapter 5

# Discussion

## 5.1 Conclusions

Several conclusions can be drawn from this lab.

1. BT Network, a small Tier-2 ISP, has been built and well tested.
2. BT Network provides both intra-domain and inter-domain Internet connection to its users. It serves common Internet applications including Web, DNS and Email as well.
3. BT Network forms and implements business relationships with neighbouring ISPs.
4. Both IS-IS and BGP routing protocols can provide alternative route(s) to the destination when one of the physicial links is down.

## 5.2 Further Work

For the future, the following improvements are being considered.

1. Implement the alternative **next-hop solution** instead of "passive interface" as in Section 3.1.5.2.
2. Fully test the implementation of BGP routing in IPv6. We are unable to test it as no neighbouring ISP has set up IPv6 BGP routing as far as we know.
3. Provide other Internet services such as Dynamic Host Configuration Protocol (DHCP)[12] and File Transfer Protocol (FTP)[13].

# Chapter 6

# Contributions

Our team is composed of 5 members from Loughborough University.

## 6.1 Group Leader: Zhihao DAI

In this lab, Zhihao DAI contributes to designing the architecture of the network, assigning IP addresses to interfaces, setting up BGP routing protocol as well as securing access to routers. Additionally, DAI is responsible for organising and formatting the whole report.

## 6.2 Technical Director: Yunsong ZHANG

Yunsong ZHANG contributes to designing the architecture of the network, setting up IS-IS routing protocol as well as DNS services.

## 6.3 Network Engineer: Huijing LEI

Huijing LEI contributes to setting up Web services. Additionally, LEI is responsible for References section in the report.

## 6.4 Network Engineer: Changrong CHEN

Changrong CHEN contributes to assigning IP addresses to interfaces and setting up Email services.

## 6.5 Network Engineer: Yan HUANG

Yan HUANG contributes to setting up Email services. Additionally, HUANG is responsible for Discussion section in the report.

# References

[1] Internet Protocol. RFC 791, September 1981.

[2] Bob Hinden and Dr. Steve E. Deering. Internet Protocol, Version 6 (IPv6) Specification. RFC 2460, December 1998.

[3] Gary S. Malkin. RIP Version 2. RFC 2453, November 1998.

[4] OSI IS-IS Intra-domain Routing Protocol. RFC 1142, February 1990.

[5] John Moy. OSPF Version 2. RFC 2328, April 1998.

[6] Yakov Rekhter, Susan Hares, and Tony Li. A Border Gateway Protocol 4 (BGP-4). RFC 4271, January 2006.

[7] Chris M. Lonvick and Tatu Ylonen. The Secure Shell (SSH) Transport Layer Protocol. RFC 4253, January 2006.

[8] Domain names - implementation and specification. RFC 1035, November 1987.

[9] Tim Berners-Lee. Universal Resource Identifiers in WWW: A Unifying Syntax for the Expression of Names and Addresses of Objects on the Network as used in the World-Wide Web. RFC 1630, June 1994.

[10] Henrik Frystyk Nielsen, Jeffrey Mogul, Larry M Masinter, Roy T. Fielding, Jim Gettys, Paul J. Leach, and Tim Berners-Lee. Hypertext Transfer Protocol – HTTP/1.1. RFC 2616, June 1999.

[11] Pete Resnick. Internet Message Format. RFC 5322, October 2008.

[12] Ralph Droms. Dynamic Host Configuration Protocol. RFC 2131, March 1997.

[13] File Transfer Protocol. RFC 959, October 1985.

# Appendix A

# Login Details

## A.1 Laptops

```
Laptop 1 (BT001):
  Username: bt001
  Password: Bt9049.4581

Laptop 2 (BT002):
  Username: bt002
  Password: Bt8717.0801

Laptop 3 (BT003):
  Username: bt003
  Password: Bt6941.6657
```

## A.2 Routers

Routers can be remotely accessed without entering the password using public-key authentication (root required) on Laptop 1 (BT001). The password for each router is one of `a^2+2ab+b^2`, `a^2-ab+b^2` or `a^2+b^2`.

```
Router 1 (BT-R001):
  Username: r001

Router 2 (BT-R002):
  Username: r002

Router 3 (BT-R003):
  Username: r003
```

# Appendix B

# Routers Configuration

## B.1 Router 1 Configuration

Listing B.1: Contents of Configuration on Router 1 (BT-R001).

```
1
2 !
3 ! Last configuration change at 14:43:27 UTC Mon Nov 4 2019 by r001
4 !
5 version 15.0
6 service timestamps debug datetime msec
7 service timestamps log datetime msec
8 no service password-encryption
9 !
10 hostname BT-R001
11 !
12 boot-start-marker
13 boot-end-marker
14 !
15 !
16 no aaa new-model
17 dot11 syslog
18 ip source-route
19 !
20 !
21 !
22 !
23 ip cef
24 no ip domain lookup
25 ip domain name bt.lboro
26 ipv6 unicast-routing
```

```
ipv6 cef
multilink bundle-name authenticated
!
!
!
!
!
!
!
!
!
voice-card 0
!
!
!
!
!
license udi pid CISCO2801 sn FCZ1339C10B
username r001 privilege 15 secret 5 $1$zzE2$UaRCPrx0iQCcbQ1Jg1Ys21
!
!
ip ssh version 2
ip ssh dh min size 4096
ip ssh pubkey-chain
 username r001
  key-hash ssh-rsa AFC27BDF03A4FB6173D4D0482B4C084A
  key-hash ssh-rsa 601E98BF47BA6CC0089AEC1177F6DEE5 root@BT001
!
!
!
!
!
interface Loopback0
 ip address 23.0.255.1 255.255.255.255
 ipv6 address 2001:2300:FFFF:1::/128
!
interface FastEthernet0/0
 ip address 23.0.0.49 255.255.255.252
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:2300:0:3::1/64
 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::1/64
 ipv6 router isis
!
interface FastEthernet0/1/0
 ip address 23.0.0.1 255.255.255.240
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:2300::1/64
 ipv6 router isis
!
router isis
 net 49.0001.0230.0025.5001.00
 is-type level-1
 passive-interface Loopback0
!
router bgp 2030
 no synchronization
 bgp log-neighbor-changes
 network 23.0.0.0 mask 255.255.255.240
 aggregate-address 23.0.0.0 255.0.0.0 summary-only
 neighbor 23.0.255.2 remote-as 2030
 neighbor 23.0.255.2 update-source Loopback0
 neighbor 23.0.255.3 remote-as 2030
 neighbor 23.0.255.3 update-source Loopback0
 neighbor 2001:2300:FFFF:2:: remote-as 2030
 neighbor 2001:2300:FFFF:2:: update-source Loopback0
 neighbor 2001:2300:FFFF:3:: remote-as 2030
 neighbor 2001:2300:FFFF:3:: update-source Loopback0
 no auto-summary
 !
 address-family ipv6
  network 2001:2300::/64
  aggregate-address 2001:2300::/32 summary-only
  neighbor 2001:2300:FFFF:2:: activate
  neighbor 2001:2300:FFFF:3:: activate
 exit-address-family
!
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
```

## B.2 Router 2 Configuration

Listing B.2: Contents of Configuration on Router 2 (BT-R002).

```

!
! Last configuration change at 14:35:53 UTC Mon Nov 4 2019 by r002
!
version 15.0
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname BT-R002
!
boot-start-marker
boot-end-marker
!
!
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 r002 privilege 15 secret 5 $1$..EF$yfYESFK6llTyMnP.f1ABh0
!
!
ip ssh version 2
ip ssh dh min size 4096
ip ssh pubkey-chain
 username r002
  key-hash ssh-rsa 601E98BF47BA6CC0089AEC1177F6DEE5 root@BT001
!
!
!
!
!
interface Loopback0
 ip address 23.0.255.2 255.255.255.255
 ipv6 address 2001:2300:FFFF:2::/128
!
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::2/64
 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::1/64
 ipv6 router isis
!
interface FastEthernet0/1/0
!
interface FastEthernet0/1/1
 switchport access vlan 3
!
interface FastEthernet0/1/2
!
interface FastEthernet0/1/3
!
interface Vlan1
 ip address 23.0.0.17 255.255.255.240
 ip router isis
```

```
 ipv6 address 2001:2300:0:1::1/64
 ipv6 router isis
!
interface Vlan3
 ip address 23.0.0.61 255.255.255.252
 ipv6 address 2001:2300:0:6::1/64
!
router isis
 net 49.0001.0230.0025.5002.00
 is-type level-1
 passive-interface Vlan3
 passive-interface Loopback0
!
router bgp 2030
 bgp log-neighbor-changes
 neighbor 23.0.0.62 remote-as 3040
 neighbor 23.0.255.1 remote-as 2030
 neighbor 23.0.255.1 update-source Loopback0
 neighbor 23.0.255.3 remote-as 2030
 neighbor 23.0.255.3 update-source Loopback0
 neighbor 2001:2300:0:6::2 remote-as 3040
 neighbor 2001:2300:FFFF:1:: remote-as 2030
 neighbor 2001:2300:FFFF:1:: update-source Loopback0
 neighbor 2001:2300:FFFF:3:: remote-as 2030
 neighbor 2001:2300:FFFF:3:: update-source Loopback0
 !
 address-family ipv4
  no synchronization
  network 23.0.0.16 mask 255.255.255.240
  aggregate-address 23.0.0.0 255.0.0.0 summary-only
  neighbor 23.0.0.62 activate
  neighbor 23.0.255.1 activate
  neighbor 23.0.255.3 activate
  neighbor 2001:2300:0:6::2 activate
  neighbor 2001:2300:FFFF:1:: activate
  neighbor 2001:2300:FFFF:3:: activate
  no auto-summary
 exit-address-family
 !
 address-family ipv6
  network 2001:2300:0:1::/64
  aggregate-address 2001:2300::/32 summary-only
  neighbor 2001:2300:0:6::2 activate
  neighbor 2001:2300:FFFF:1:: activate
  neighbor 2001:2300:FFFF:3:: activate
 exit-address-family
!
```

```
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
```

## B.3 Router 3 Configuration

Listing B.3: Contents of Configuration on Router 3 (BT-R003).

```

!
! Last configuration change at 06:40:58 UTC Sun Nov 3 2019 by r003
!
version 15.0
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname BT-R003
!
boot-start-marker
boot-end-marker
!
!
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 r003 privilege 15 secret 5 $1$Jn3f$lSSIZA5XOcOJxuXDkz8rE/
!
!
ip ssh version 2
ip ssh dh min size 4096
ip ssh pubkey-chain
 username r003
  key-hash ssh-rsa 601E98BF47BA6CC0089AEC1177F6DEE5 root@BT001
!
!
!
!
!
interface Loopback0
 ip address 23.0.255.3 255.255.255.255
 ipv6 address 2001:2300:FFFF:3::/128
!
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::2/64
 ipv6 router isis
!
interface FastEthernet0/1
 ip address 23.0.0.54 255.255.255.252
 ip router isis
 duplex auto
 speed auto
 ipv6 address 2001:2300:0:4::2/64
 ipv6 router isis
!
interface FastEthernet0/1/0
 switchport access vlan 2
!
interface FastEthernet0/1/1
 switchport access vlan 5
!
interface FastEthernet0/1/2
 switchport access vlan 4
!
interface FastEthernet0/1/3
!
interface Vlan1
```

```
 no ip address
!
interface Vlan2
 ip address 23.0.0.33 255.255.255.240
 ip router isis
 ipv6 address 2001:2300:0:2::1/64
 ipv6 router isis
!
interface Vlan4
 ip address 56.0.0.62 255.255.255.252
 ipv6 address 2001:5600:0:6::2/64
!
interface Vlan5
 ip address 100.100.2.2 255.255.255.252
!
router isis
 net 49.0001.0230.0025.5003.00
 is-type level-1
 passive-interface Vlan4
 passive-interface Vlan5
 passive-interface Loopback0
!
router bgp 2030
 no synchronization
 bgp log-neighbor-changes
 network 23.0.0.32 mask 255.255.255.240
 aggregate-address 23.0.0.0 255.0.0.0 summary-only
 neighbor 23.0.255.1 remote-as 2030
 neighbor 23.0.255.1 update-source Loopback0
 neighbor 23.0.255.2 remote-as 2030
 neighbor 23.0.255.2 update-source Loopback0
 neighbor 2001:2300:FFFF:1:: remote-as 2030
 neighbor 2001:2300:FFFF:1:: update-source Loopback0
 neighbor 2001:2300:FFFF:2:: remote-as 2030
 neighbor 2001:2300:FFFF:2:: update-source Loopback0
 neighbor 2001:5600:0:6::1 remote-as 5060
 neighbor 56.0.0.61 remote-as 5060
 neighbor 56.0.0.61 filter-list 1 out
 neighbor 100.100.2.1 remote-as 42
 neighbor 100.100.2.1 filter-list 1 out
 no auto-summary
 !
 address-family ipv6
  network 2001:2300:0:2::/64
  aggregate-address 2001:2300::/32 summary-only
  neighbor 2001:2300:FFFF:1:: activate
  neighbor 2001:2300:FFFF:2:: activate
```

```
 exit-address-family
!
ip forward-protocol nd
!
ip as-path access-list 1 deny _42_
ip as-path access-list 1 deny _5060_
ip as-path access-list 1 permit .*
!
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
```

# Appendix C

# Laptops Configuration

## C.1 Laptop 1 Configuration

Listing C.1: Contents of Interfaces Network Configuration File Located at `/etc/network/interfaces` on Laptop 1 (BT001).

```
# 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 23.0.0.2
  netmask 255.255.255.240
  gateway 23.0.0.1
  dns-nameservers 23.0.0.34 23.0.0.2
# up flush-mail


iface eth0 inet6 static
  address 2001:2300::2
  netmask 64
  gateway 2001:2300::1
```

## C.2 Laptop 2 Configuration

Listing C.2: Contents of Interfaces Network Configuration File Located at `/etc/network/interfaces` on Laptop 2 (BT002).

```
# 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 23.0.0.18
        netmask 255.255.255.240
  gateway 23.0.0.17
  dns-nameservers 23.0.0.34 23.0.0.2
#        up flush-mail

iface eth0 inet6 static
  address 2001:2300:0:1::2
  netmask 64
  gateway 2001:2300:0:1::1
```

## C.3 Laptop 3 Configuration

Listing C.3: Contents of Interfaces Network Configuration File Located at `/etc/network/interfaces` on Laptop 3 (BT003).

```
# 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 23.0.0.2
  netmask 255.255.255.240
  gateway 23.0.0.1
  dns-nameservers 23.0.0.34 23.0.0.2
# up flush-mail


iface eth0 inet6 static
  address 2001:2300::2
  netmask 64
  gateway 2001:2300::1
```

# Appendix D

# DNS Configuration

## D.1 Primary DNS Server (Laptop 3)

Listing D.1: Contents of DNS Configuration File Located at `/etc/bind/named.conf.options` on Primary DNS Server (Laptop 3).

```
options {
  directory "/var/cache/bind";

  // If there is a firewall between you and nameservers you want
  // to talk to, you may need to fix the firewall to allow multiple
  // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

  // If your ISP provided one or more IP addresses for stable
  // nameservers, you probably want to use them as forwarders.
  // Uncomment the following block, and insert the addresses replacing
  // the all-0's placeholder.

  // forwarders {
  //  0.0.0.0;
  // };

  //========================================================================
  // If BIND logs error messages about the root key being expired,
  // you will need to update your keys.  See https://www.isc.org/bind-keys
  //========================================================================
  // dnssec-validation auto;

  auth-nxdomain no;    # conform to RFC1035
  listen-on-v6 { any; };
```

```
25   forwarders { 10.2.2.1; };
26   recursion yes;
27   dnssec-enable no;
28   dnssec-validation no;
29   empty-zones-enable no;
30   allow-query {any;};
31 };
```

Listing D.2: Contents of Local DNS Configuration File Located at `/etc/bind/named.conf.local` on Primary DNS Server (Laptop 3).

```
1  //
2  // Do any local configuration here
3  //
4
5  // Consider adding the 1918 zones here, if they are not used in your
6  // organization
7  //include "/etc/bind/zones.rfc1918";
8
9  zone "bt.lboro" {
10   type master;
11   file "/etc/bind/db.bt.lboro";
12   allow-transfer { 23.0.0.2; };
13   also-notify { 23.0.0.2; };
14 };
15
16 zone "23.in-addr.arpa" {
17   type master;
18   file "/etc/bind/db.23";
19   allow-transfer { 23.0.0.2; };
20   also-notify { 23.0.0.2; };
21 };
```

## D.2 Secondary DNS Server (Laptop 1)

Listing D.3: Contents of DNS Configuration File Located at `/etc/bind/named.conf.options` on Secondary DNS Server (Laptop 1).

```
options {
  directory "/var/cache/bind";

  // If there is a firewall between you and nameservers you want
  // to talk to, you may need to fix the firewall to allow multiple
  // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

  // If your ISP provided one or more IP addresses for stable
  // nameservers, you probably want to use them as forwarders.
  // Uncomment the following block, and insert the addresses replacing
  // the all-0's placeholder.

  // forwarders {
  //  0.0.0.0;
  // };

  //========================================================================
  // If BIND logs error messages about the root key being expired,
  // you will need to update your keys.  See https://www.isc.org/bind-keys
  //========================================================================
  dnssec-validation auto;

  auth-nxdomain no;    # conform to RFC1035
  listen-on-v6 { any; };
};
```

Listing D.4: Contents of Local DNS Configuration File Located at `/etc/bind/named.conf.local` on Secondary DNS Server (Laptop 1).

```
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "bt.lboro" {
  type slave;
  file "db.bt.lboro";
  masters { 23.0.0.34; };
```

```
13 };
14
15 zone "0.0.23.in-addr.arpa" {
16   type slave;
17   file "db.23";
18   masters { 23.0.0.34; };
19 };
```

# Appendix E

# Email Configuration

Listing E.1: Contents of Generated Email Configuration File Located at `/var/lib/exim4/config.autogenerated` on Email Server (Laptop 3).

```
#########
# WARNING WARNING WARNING
# WARNING WARNING WARNING
# WARNING WARNING WARNING
# WARNING WARNING WARNING
# WARNING WARNING WARNING
# This file was generated dynamically from
# non-split config (/etc/exim4/exim4.conf.localmacros
# and /etc/exim4/exim4.conf.template).
# The config files are supplemented with package installation/configuration
# settings managed by debconf. This data is stored in
# /etc/exim4/update-exim4.conf.conf
# Any changes you make here will be lost.
# See /usr/share/doc/exim4-base/README.Debian.gz and update-exim4.conf(8)
# for instructions of customization.
# WARNING WARNING WARNING
# WARNING WARNING WARNING
# WARNING WARNING WARNING
# WARNING WARNING WARNING
# WARNING WARNING WARNING
#########

exim_path = /usr/sbin/exim4

.ifndef CONFDIR
CONFDIR = /etc/exim4
.endif
```

```
28
29 UPEX4CmacrosUPEX4C = 1
30 ##################################################
31 # the following macro definitions were created
32 # dynamically by /usr/sbin/update-exim4.conf
33 .ifndef MAIN_PACKAGE_VERSION
34 MAIN_PACKAGE_VERSION=4.82-3ubuntu2
35 .endif
36 .ifndef MAIN_LOCAL_DOMAINS
37 MAIN_LOCAL_DOMAINS=@:localhost:bt003:bt.lboro
38 .endif
39 .ifndef MAIN_RELAY_TO_DOMAINS
40 MAIN_RELAY_TO_DOMAINS=empty
41 .endif
42 .ifndef ETC_MAILNAME
43 ETC_MAILNAME=bt.lboro
44 .endif
45 .ifndef LOCAL_DELIVERY
46 LOCAL_DELIVERY=mail_spool
47 .endif
48 .ifndef MAIN_RELAY_NETS
49 MAIN_RELAY_NETS=: 127.0.0.1 : ::::1
50 .endif
51 .ifndef DCreadhost
52 DCreadhost=empty
53 .endif
54 .ifndef DCsmarthost
55 DCsmarthost=empty
56 .endif
57 .ifndef DC_eximconfig_configtype
58 DC_eximconfig_configtype=internet
59 .endif
60 .ifndef DCconfig_internet
61 DCconfig_internet=1
62 .endif
63 ##################################################
64
65
66 domainlist local_domains = MAIN_LOCAL_DOMAINS
67
68 domainlist relay_to_domains = MAIN_RELAY_TO_DOMAINS
69
70 hostlist relay_from_hosts = MAIN_RELAY_NETS
71
72 .ifndef MAIN_PRIMARY_HOSTNAME_AS_QUALIFY_DOMAIN
73 .ifndef MAIN_QUALIFY_DOMAIN
74 qualify_domain = ETC_MAILNAME
```

```
.else
qualify_domain = MAIN_QUALIFY_DOMAIN
.endif
.endif

.ifdef MAIN_LOCAL_INTERFACES
local_interfaces = MAIN_LOCAL_INTERFACES
.endif

.ifndef LOCAL_DELIVERY
LOCAL_DELIVERY=mail_spool
.endif

gecos_pattern = ^([^,:]*)
gecos_name = $1

.ifndef CHECK_RCPT_LOCAL_LOCALPARTS
CHECK_RCPT_LOCAL_LOCALPARTS = ^[.] : ^.*[@%!/|`#&?]
.endif

.ifndef CHECK_RCPT_REMOTE_LOCALPARTS
CHECK_RCPT_REMOTE_LOCALPARTS = ^[./|] : ^.*[@%!`#&?] : ^.*/\\.\\./
.endif

.ifndef MAIN_LOG_SELECTOR
MAIN_LOG_SELECTOR = +tls_peerdn
.endif

.ifndef MAIN_ACL_CHECK_MAIL
MAIN_ACL_CHECK_MAIL = acl_check_mail
.endif
acl_smtp_mail = MAIN_ACL_CHECK_MAIL

.ifndef MAIN_ACL_CHECK_RCPT
MAIN_ACL_CHECK_RCPT = acl_check_rcpt
.endif
acl_smtp_rcpt = MAIN_ACL_CHECK_RCPT

.ifndef MAIN_ACL_CHECK_DATA
MAIN_ACL_CHECK_DATA = acl_check_data
.endif
acl_smtp_data = MAIN_ACL_CHECK_DATA

.ifdef MESSAGE_SIZE_LIMIT
message_size_limit = MESSAGE_SIZE_LIMIT
.endif

```

```
.ifdef MAIN_ALLOW_DOMAIN_LITERALS
allow_domain_literals
.endif

.ifndef DC_minimaldns
.ifndef MAIN_HOST_LOOKUP
MAIN_HOST_LOOKUP = *
.endif
host_lookup = MAIN_HOST_LOOKUP
.endif

.ifdef MAIN_HARDCODE_PRIMARY_HOSTNAME
primary_hostname = MAIN_HARDCODE_PRIMARY_HOSTNAME
.endif

.ifdef MAIN_SMTP_ACCEPT_MAX_NOMAIL_HOSTS
smtp_accept_max_nonmail_hosts = MAIN_SMTP_ACCEPT_MAX_NOMAIL_HOSTS
.endif

.ifndef MAIN_FORCE_SENDER
local_from_check = false
local_sender_retain = true
untrusted_set_sender = *
.endif

.ifndef MAIN_IGNORE_BOUNCE_ERRORS_AFTER
MAIN_IGNORE_BOUNCE_ERRORS_AFTER = 2d
.endif
ignore_bounce_errors_after = MAIN_IGNORE_BOUNCE_ERRORS_AFTER

.ifndef MAIN_TIMEOUT_FROZEN_AFTER
MAIN_TIMEOUT_FROZEN_AFTER = 7d
.endif
timeout_frozen_after = MAIN_TIMEOUT_FROZEN_AFTER

.ifndef MAIN_FREEZE_TELL
MAIN_FREEZE_TELL = postmaster
.endif
freeze_tell = MAIN_FREEZE_TELL

.ifndef SPOOLDIR
SPOOLDIR = /var/spool/exim4
.endif
spool_directory = SPOOLDIR

.ifndef MAIN_TRUSTED_USERS
MAIN_TRUSTED_USERS = uucp
```

```
.endif
trusted_users = MAIN_TRUSTED_USERS
.ifdef MAIN_TRUSTED_GROUPS
trusted_groups = MAIN_TRUSTED_GROUPS
.endif

.ifdef MAIN_TLS_ENABLE
.ifndef MAIN_TLS_ADVERTISE_HOSTS
MAIN_TLS_ADVERTISE_HOSTS = *
.endif
tls_advertise_hosts = MAIN_TLS_ADVERTISE_HOSTS

.ifdef MAIN_TLS_CERTKEY
tls_certificate = MAIN_TLS_CERTKEY
.else
.ifndef MAIN_TLS_CERTIFICATE
MAIN_TLS_CERTIFICATE = CONFDIR/exim.crt
.endif
tls_certificate = MAIN_TLS_CERTIFICATE

.ifndef MAIN_TLS_PRIVATEKEY
MAIN_TLS_PRIVATEKEY = CONFDIR/exim.key
.endif
tls_privatekey = MAIN_TLS_PRIVATEKEY
.endif

.ifndef MAIN_TLS_VERIFY_CERTIFICATES
MAIN_TLS_VERIFY_CERTIFICATES = ${if exists{/etc/ssl/certs/ca-certificates.crt
    }\
                                    {/etc/ssl/certs/ca-certificates.crt}\
            {/dev/null}}
.endif
tls_verify_certificates = MAIN_TLS_VERIFY_CERTIFICATES

.ifdef MAIN_TLS_VERIFY_HOSTS
tls_verify_hosts = MAIN_TLS_VERIFY_HOSTS
.endif

.ifdef MAIN_TLS_TRY_VERIFY_HOSTS
tls_try_verify_hosts = MAIN_TLS_TRY_VERIFY_HOSTS
.endif

.endif

.ifdef MAIN_LOG_SELECTOR
log_selector = MAIN_LOG_SELECTOR
.endif
```

```
215
216 begin acl
217
218 acl_local_deny_exceptions:
219   accept
220     hosts = ${if exists{CONFDIR/host_local_deny_exceptions}\
221                  {CONFDIR/host_local_deny_exceptions}\
222                  {}}
223   accept
224     senders = ${if exists{CONFDIR/sender_local_deny_exceptions}\
225                    {CONFDIR/sender_local_deny_exceptions}\
226                    {}}
227   accept
228     hosts = ${if exists{CONFDIR/local_host_whitelist}\
229                  {CONFDIR/local_host_whitelist}\
230                  {}}
231   accept
232     senders = ${if exists{CONFDIR/local_sender_whitelist}\
233                    {CONFDIR/local_sender_whitelist}\
234                    {}}
235
236   .ifdef LOCAL_DENY_EXCEPTIONS_LOCAL_ACL_FILE
237   .include LOCAL_DENY_EXCEPTIONS_LOCAL_ACL_FILE
238   .endif
239
240   .ifdef WHITELIST_LOCAL_DENY_LOCAL_ACL_FILE
241   .include WHITELIST_LOCAL_DENY_LOCAL_ACL_FILE
242   .endif
243
244 acl_check_mail:
245   .ifdef CHECK_MAIL_HELO_ISSUED
246   deny
247     message = no HELO given before MAIL command
248     condition = ${if def:sender_helo_name {no}{yes}}
249   .endif
250
251   accept
252
253 acl_check_rcpt:
254
255   accept
256     hosts = :
257     control = dkim_disable_verify
258
259 .ifdef DC_minimaldns
260   warn
261     control = dkim_disable_verify
```

```
.else
.ifdef DISABLE_DKIM_VERIFY
  warn
    control = dkim_disable_verify
.endif
.endif

  .ifdef CHECK_RCPT_LOCAL_LOCALPARTS
  deny
    domains = +local_domains
    local_parts = CHECK_RCPT_LOCAL_LOCALPARTS
    message = restricted characters in address
  .endif



  .ifdef CHECK_RCPT_REMOTE_LOCALPARTS
  deny
    domains = !+local_domains
    local_parts = CHECK_RCPT_REMOTE_LOCALPARTS
    message = restricted characters in address
  .endif

  accept
    .ifndef CHECK_RCPT_POSTMASTER
    local_parts = postmaster
    .else
    local_parts = CHECK_RCPT_POSTMASTER
    .endif
    domains = +local_domains : +relay_to_domains

  .ifdef CHECK_RCPT_VERIFY_SENDER
  deny
    message = Sender verification failed
    !acl = acl_local_deny_exceptions
    !verify = sender
  .endif

  deny
    !acl = acl_local_deny_exceptions
    senders = ${if exists{CONFDIR/local_sender_callout}\
                         {CONFDIR/local_sender_callout}\
                  {}}
    !verify = sender/callout

  accept
    hosts = +relay_from_hosts
```

```
309     control = submission/sender_retain
310     control = dkim_disable_verify
311
312   accept
313     authenticated = *
314     control = submission/sender_retain
315     control = dkim_disable_verify
316
317   require
318     message = relay not permitted
319     domains = +local_domains : +relay_to_domains
320
321   require
322     verify = recipient
323
324   deny
325     !acl = acl_local_deny_exceptions
326     recipients = ${if exists{CONFDIR/local_rcpt_callout}\
327                             {CONFDIR/local_rcpt_callout}\
328                     {}}
329     !verify = recipient/callout
330
331   deny
332     message = sender envelope address $sender_address is locally blacklisted
      here. If you think this is wrong, get in touch with postmaster
333     !acl = acl_local_deny_exceptions
334     senders = ${if exists{CONFDIR/local_sender_blacklist}\
335                   {CONFDIR/local_sender_blacklist}\
336                   {}}
337
338   deny
339     message = sender IP address $sender_host_address is locally blacklisted
      here. If you think this is wrong, get in touch with postmaster
340     !acl = acl_local_deny_exceptions
341     hosts = ${if exists{CONFDIR/local_host_blacklist}\
342                 {CONFDIR/local_host_blacklist}\
343                 {}}
344
345   .ifdef CHECK_RCPT_REVERSE_DNS
346   warn
347     condition = ${if and{{def:sender_host_address}{!def:sender_host_name}}\
348                       {yes}{no}}
349     add_header = X-Host-Lookup-Failed: Reverse DNS lookup failed for
      $sender_host_address (${if eq{$host_lookup_failed}{1}{failed}{deferred}})
350   .endif
351
352   .ifdef CHECK_RCPT_SPF
```

```
353  deny
354    message = [SPF] $sender_host_address is not allowed to send mail from \
355              ${if def:sender_address_domain {$sender_address_domain}{
     $sender_helo_name}}.  \
356              Please see \
357          http://www.openspf.org/Why?scope=${if def:sender_address_domain \
358              {mfrom}{helo}};identity=${if def:sender_address_domain \
359              {$sender_address}{$sender_helo_name}};ip=$sender_host_address
360    log_message = SPF check failed.
361    !acl = acl_local_deny_exceptions
362    condition = ${run{/usr/bin/spfquery.mail-spf-perl --ip \
363                    ${quote:$sender_host_address} --identity \
364                    ${if def:sender_address_domain \
365                        {--scope mfrom  --identity ${quote:$sender_address}}\
366                        {--scope helo --identity ${quote:$sender_helo_name}}}}\
367                    {no}{${if eq {$runrc}{1}{yes}{no}}}}
368
369  defer
370    message = Temporary DNS error while checking SPF record.  Try again later.
371    !acl = acl_local_deny_exceptions
372    condition = ${if eq {$runrc}{5}{yes}{no}}
373
374  warn
375    condition = ${if <={$runrc}{6}{yes}{no}}
376    add_header = Received-SPF: ${if eq {$runrc}{0}{pass}\
377                                 {${if eq {$runrc}{2}{softfail}\
378                                  {${if eq {$runrc}{3}{neutral}\
379           {${if eq {$runrc}{4}{permerror}\
380            {${if eq {$runrc}{6}{none}{error}}}}}}}}}\
381         } client-ip=$sender_host_address; \
382         ${if def:sender_address_domain \
383            {envelope-from=${sender_address}; }{}}\
384         helo=$sender_helo_name
385
386  warn
387    log_message = Unexpected error in SPF check.
388    condition = ${if >{$runrc}{6}{yes}{no}}
389  .endif
390
391  .ifdef CHECK_RCPT_IP_DNSBLS
392  warn
393    dnslists = CHECK_RCPT_IP_DNSBLS
394    add_header = X-Warning: $sender_host_address is listed at $dnslist_domain
     ($dnslist_value: $dnslist_text)
395    log_message = $sender_host_address is listed at $dnslist_domain (
     $dnslist_value: $dnslist_text)
396  .endif
```

```
397
398   .ifdef CHECK_RCPT_DOMAIN_DNSBLS
399   warn
400     !senders = ${if exists{CONFDIR/local_domain_dnsbl_whitelist}\
401                    {CONFDIR/local_domain_dnsbl_whitelist}\
402                    {}}
403     dnslists = CHECK_RCPT_DOMAIN_DNSBLS
404     add_header = X-Warning: $sender_address_domain is listed at
      $dnslist_domain ($dnslist_value: $dnslist_text)
405     log_message = $sender_address_domain is listed at $dnslist_domain (
      $dnslist_value: $dnslist_text)
406   .endif
407
408   .ifdef CHECK_RCPT_LOCAL_ACL_FILE
409   .include CHECK_RCPT_LOCAL_ACL_FILE
410   .endif
411
412   accept
413     domains = +relay_to_domains
414     endpass
415     verify = recipient
416
417   accept
418
419 acl_check_data:
420
421   .ifdef CHECK_DATA_VERIFY_HEADER_SYNTAX
422   deny
423     message = Message headers fail syntax check
424     !acl = acl_local_deny_exceptions
425     !verify = header_syntax
426   .endif
427
428   .ifdef CHECK_DATA_VERIFY_HEADER_SENDER
429   deny
430     message = No verifiable sender address in message headers
431     !acl = acl_local_deny_exceptions
432     !verify = header_sender
433   .endif
434
435   .ifdef CHECK_DATA_LOCAL_ACL_FILE
436   .include CHECK_DATA_LOCAL_ACL_FILE
437   .endif
438
439   accept
440
441 begin routers
```

```
442
443 .ifdef MAIN_ALLOW_DOMAIN_LITERALS
444 domain_literal:
445   debug_print = "R: domain_literal for $local_part@$domain"
446   driver = ipliteral
447   domains = ! +local_domains
448   transport = remote_smtp
449 .endif
450
451 hubbed_hosts:
452   debug_print = "R: hubbed_hosts for $domain"
453   driver = manualroute
454   domains = "${if exists{CONFDIR/hubbed_hosts}\
455                    {partial-lsearch;CONFDIR/hubbed_hosts}\
456                fail}"
457   same_domain_copy_routing = yes
458   route_data = ${lookup{$domain}partial-lsearch{CONFDIR/hubbed_hosts}}
459   transport = remote_smtp
460
461 .ifdef DCconfig_internet
462
463 dnslookup_relay_to_domains:
464   debug_print = "R: dnslookup_relay_to_domains for $local_part@$domain"
465   driver = dnslookup
466   domains = ! +local_domains : +relay_to_domains
467   transport = remote_smtp
468   same_domain_copy_routing = yes
469   no_more
470
471 dnslookup:
472   debug_print = "R: dnslookup for $local_part@$domain"
473   driver = dnslookup
474   domains = ! +local_domains
475   transport = remote_smtp
476   same_domain_copy_routing = yes
477   ignore_target_hosts = 0.0.0.0 : 127.0.0.0/8 : 192.168.0.0/16 :\
478                         172.16.0.0/12 : 10.0.0.0/8 : 169.254.0.0/16 :\
479       255.255.255.255
480   no_more
481
482 .endif
483
484 .ifdef DCconfig_local
485 nonlocal:
486   debug_print = "R: nonlocal for $local_part@$domain"
487   driver = redirect
488   domains = ! +local_domains
```

```
489   allow_fail
490   data = :fail: Mailing to remote domains not supported
491   no_more
492
493 .endif
494
495 .ifdef DCconfig_smarthost DCconfig_satellite
496
497 smarthost:
498   debug_print = "R: smarthost for $local_part@$domain"
499   driver = manualroute
500   domains = ! +local_domains
501   transport = remote_smtp_smarthost
502   route_list = * DCsmarthost byname
503   host_find_failed = ignore
504   same_domain_copy_routing = yes
505   no_more
506
507 .endif
508
509 COND_LOCAL_SUBMITTER = "\
510                ${if match_ip{$sender_host_address}{:@[]}\
511                     {1}{0}\
512     }"
513
514 real_local:
515   debug_print = "R: real_local for $local_part@$domain"
516   driver = accept
517   domains = +local_domains
518   condition = COND_LOCAL_SUBMITTER
519   local_part_prefix = real-
520   check_local_user
521   transport = LOCAL_DELIVERY
522
523 system_aliases:
524   debug_print = "R: system_aliases for $local_part@$domain"
525   driver = redirect
526   domains = +local_domains
527   allow_fail
528   allow_defer
529   data = ${lookup{$local_part}lsearch{/etc/aliases}}
530   .ifdef SYSTEM_ALIASES_USER
531   user = SYSTEM_ALIASES_USER
532   .endif
533   .ifdef SYSTEM_ALIASES_GROUP
534   group = SYSTEM_ALIASES_GROUP
535   .endif
```

```
536   .ifdef SYSTEM_ALIASES_FILE_TRANSPORT
537   file_transport = SYSTEM_ALIASES_FILE_TRANSPORT
538   .endif
539   .ifdef SYSTEM_ALIASES_PIPE_TRANSPORT
540   pipe_transport = SYSTEM_ALIASES_PIPE_TRANSPORT
541   .endif
542   .ifdef SYSTEM_ALIASES_DIRECTORY_TRANSPORT
543   directory_transport = SYSTEM_ALIASES_DIRECTORY_TRANSPORT
544   .endif
545
546 .ifdef DCconfig_satellite
547 hub_user:
548   debug_print = "R: hub_user for $local_part@$domain"
549   driver = redirect
550   domains = +local_domains
551   data = ${local_part}@DCreadhost
552   check_local_user
553
554 hub_user_smarthost:
555   debug_print = "R: hub_user_smarthost for $local_part@$domain"
556   driver = manualroute
557   domains = DCreadhost
558   transport = remote_smtp_smarthost
559   route_list = * DCsmarthost byname
560   host_find_failed = ignore
561   same_domain_copy_routing = yes
562   check_local_user
563 .endif
564
565 userforward:
566   debug_print = "R: userforward for $local_part@$domain"
567   driver = redirect
568   domains = +local_domains
569   check_local_user
570   file = $home/.forward
571   require_files = $local_part:$home/.forward
572   no_verify
573   no_expn
574   check_ancestor
575   allow_filter
576   forbid_smtp_code = true
577   directory_transport = address_directory
578   file_transport = address_file
579   pipe_transport = address_pipe
580   reply_transport = address_reply
581   skip_syntax_errors
582   syntax_errors_to = real-$local_part@$domain
```

```
583   syntax_errors_text = \
584     This is an automatically generated message. An error has\n\
585     been found in your .forward file. Details of the error are\n\
586     reported below. While this error persists, you will receive\n\
587     a copy of this message for every message that is addressed\n\
588     to you. If your .forward file is a filter file, or if it is\n\
589     a non-filter file containing no valid forwarding addresses,\n\
590     a copy of each incoming message will be put in your normal\n\
591     mailbox. If a non-filter file contains at least one valid\n\
592     forwarding address, forwarding to the valid addresses will\n\
593     happen, and those will be the only deliveries that occur.
594
595 procmail:
596   debug_print = "R: procmail for $local_part@$domain"
597   driver = accept
598   domains = +local_domains
599   check_local_user
600   transport = procmail_pipe
601   require_files = ${local_part}:\
602                   ${if exists{/etc/procmailrc}\
603                     {/etc/procmailrc}{${home}/.procmailrc}}:\
604                   +/usr/bin/procmail
605   no_verify
606   no_expn
607
608 maildrop:
609   debug_print = "R: maildrop for $local_part@$domain"
610   driver = accept
611   domains = +local_domains
612   check_local_user
613   transport = maildrop_pipe
614   require_files = ${local_part}:${home}/.mailfilter:+/usr/bin/maildrop
615   no_verify
616   no_expn
617
618 .ifndef FIRST_USER_ACCOUNT_UID
619 FIRST_USER_ACCOUNT_UID = 0
620 .endif
621
622 .ifndef DEFAULT_SYSTEM_ACCOUNT_ALIAS
623 DEFAULT_SYSTEM_ACCOUNT_ALIAS = :fail: no mail to system accounts
624 .endif
625
626 COND_SYSTEM_USER_AND_REMOTE_SUBMITTER = "\
627                ${if and{{! match_ip{$sender_host_address}{:@[]}}\
628                         {<{$local_user_uid}{FIRST_USER_ACCOUNT_UID}}}\
629                     {1}{0}\
```

```
630     }"
631
632 lowuid_aliases:
633   debug_print = "R: lowuid_aliases for $local_part@$domain (UID
        $local_user_uid)"
634   check_local_user
635   driver = redirect
636   allow_fail
637   domains = +local_domains
638   condition = COND_SYSTEM_USER_AND_REMOTE_SUBMITTER
639   data = ${if exists{CONFDIR/lowuid-aliases}\
640               {${lookup{$local_part}lsearch{CONFDIR/lowuid-aliases}\
641               {$value}{DEFAULT_SYSTEM_ACCOUNT_ALIAS}}}\
642               {DEFAULT_SYSTEM_ACCOUNT_ALIAS}}
643
644 local_user:
645   debug_print = "R: local_user for $local_part@$domain"
646   driver = accept
647   domains = +local_domains
648   check_local_user
649   local_parts = ! root
650   transport = LOCAL_DELIVERY
651   cannot_route_message = Unknown user
652
653 mail4root:
654   debug_print = "R: mail4root for $local_part@$domain"
655   driver = redirect
656   domains = +local_domains
657   data = /var/mail/mail
658   file_transport = address_file
659   local_parts = root
660   user = mail
661   group = mail
662
663 begin transports
664
665 .ifdef HIDE_MAILNAME
666 REMOTE_SMTP_HEADERS_REWRITE=*@+local_domains $1@DCreadhost frs : *
        @ETC_MAILNAME $1@DCreadhost frs
667 REMOTE_SMTP_RETURN_PATH=${if match_domain{$sender_address_domain}{+
        local_domains}{${sender_address_local_part}@DCreadhost}{${if match_domain{
        $sender_address_domain}{ETC_MAILNAME}{${sender_address_local_part}
        @DCreadhost}fail}}}
668 .endif
669
670 .ifdef REMOTE_SMTP_HELO_FROM_DNS
671 .ifdef REMOTE_SMTP_HELO_DATA
```

```
672 REMOTE_SMTP_HELO_DATA==${lookup dnsdb {ptr=$sending_ip_address}{$value}{
        $primary_hostname}}
673 .else
674 REMOTE_SMTP_HELO_DATA=${lookup dnsdb {ptr=$sending_ip_address}{$value}{
        $primary_hostname}}
675 .endif
676 .endif
677
678 address_file:
679   debug_print = "T: address_file for $local_part@$domain"
680   driver = appendfile
681   delivery_date_add
682   envelope_to_add
683   return_path_add
684
685 address_pipe:
686   debug_print = "T: address_pipe for $local_part@$domain"
687   driver = pipe
688   return_fail_output
689
690 address_reply:
691   debug_print = "T: autoreply for $local_part@$domain"
692   driver = autoreply
693
694 mail_spool:
695   debug_print = "T: appendfile for $local_part@$domain"
696   driver = appendfile
697   file = /var/mail/$local_part
698   delivery_date_add
699   envelope_to_add
700   return_path_add
701   group = mail
702   mode = 0660
703   mode_fail_narrower = false
704
705 maildir_home:
706   debug_print = "T: maildir_home for $local_part@$domain"
707   driver = appendfile
708   .ifdef MAILDIR_HOME_MAILDIR_LOCATION
709   directory = MAILDIR_HOME_MAILDIR_LOCATION
710   .else
711   directory = $home/Maildir
712   .endif
713   .ifdef MAILDIR_HOME_CREATE_DIRECTORY
714   create_directory
715   .endif
716   .ifdef MAILDIR_HOME_CREATE_FILE
```

```
717   create_file = MAILDIR_HOME_CREATE_FILE
718   .endif
719   delivery_date_add
720   envelope_to_add
721   return_path_add
722   maildir_format
723   .ifdef MAILDIR_HOME_DIRECTORY_MODE
724   directory_mode = MAILDIR_HOME_DIRECTORY_MODE
725   .else
726   directory_mode = 0700
727   .endif
728   .ifdef MAILDIR_HOME_MODE
729   mode = MAILDIR_HOME_MODE
730   .else
731   mode = 0600
732   .endif
733   mode_fail_narrower = false
734
735 maildrop_pipe:
736   debug_print = "T: maildrop_pipe for $local_part@$domain"
737   driver = pipe
738   path = "/bin:/usr/bin:/usr/local/bin"
739   command = "/usr/bin/maildrop"
740   return_path_add
741   delivery_date_add
742   envelope_to_add
743
744 procmail_pipe:
745   debug_print = "T: procmail_pipe for $local_part@$domain"
746   driver = pipe
747   path = "/bin:/usr/bin:/usr/local/bin"
748   command = "/usr/bin/procmail"
749   return_path_add
750   delivery_date_add
751   envelope_to_add
752
753 remote_smtp:
754   debug_print = "T: remote_smtp for $local_part@$domain"
755   driver = smtp
756 .ifdef REMOTE_SMTP_HOSTS_AVOID_TLS
757   hosts_avoid_tls = REMOTE_SMTP_HOSTS_AVOID_TLS
758 .endif
759 .ifdef REMOTE_SMTP_HEADERS_REWRITE
760   headers_rewrite = REMOTE_SMTP_HEADERS_REWRITE
761 .endif
762 .ifdef REMOTE_SMTP_RETURN_PATH
763   return_path = REMOTE_SMTP_RETURN_PATH
```

```
.endif
.ifdef REMOTE_SMTP_HELO_DATA
  helo_data=REMOTE_SMTP_HELO_DATA
.endif
.ifdef DKIM_DOMAIN
dkim_domain = DKIM_DOMAIN
.endif
.ifdef DKIM_SELECTOR
dkim_selector = DKIM_SELECTOR
.endif
.ifdef DKIM_PRIVATE_KEY
dkim_private_key = DKIM_PRIVATE_KEY
.endif
.ifdef DKIM_CANON
dkim_canon = DKIM_CANON
.endif
.ifdef DKIM_STRICT
dkim_strict = DKIM_STRICT
.endif
.ifdef DKIM_SIGN_HEADERS
dkim_sign_headers = DKIM_SIGN_HEADERS
.endif
.ifdef TLS_DH_MIN_BITS
tls_dh_min_bits = TLS_DH_MIN_BITS
.endif
.ifdef REMOTE_SMTP_TLS_CERTIFICATE
tls_certificate = REMOTE_SMTP_TLS_CERTIFICATE
.endif
.ifdef REMOTE_SMTP_PRIVATEKEY
tls_privatekey = REMOTE_SMTP_PRIVATEKEY
.endif

remote_smtp_smarthost:
  debug_print = "T: remote_smtp_smarthost for $local_part@$domain"
  driver = smtp
  hosts_try_auth = <; ${if exists{CONFDIR/passwd.client} \
        {\
        ${lookup{$host}nwildlsearch{CONFDIR/passwd.client}{$host_address}}\
        }\
        {} \
      }
.ifdef REMOTE_SMTP_SMARTHOST_HOSTS_AVOID_TLS
  hosts_avoid_tls = REMOTE_SMTP_SMARTHOST_HOSTS_AVOID_TLS
.endif
.ifdef REMOTE_SMTP_HEADERS_REWRITE
  headers_rewrite = REMOTE_SMTP_HEADERS_REWRITE
.endif
```

```
.ifdef REMOTE_SMTP_RETURN_PATH
  return_path = REMOTE_SMTP_RETURN_PATH
.endif
.ifdef REMOTE_SMTP_HELO_DATA
  helo_data=REMOTE_SMTP_HELO_DATA
.endif
.ifdef TLS_DH_MIN_BITS
tls_dh_min_bits = TLS_DH_MIN_BITS
.endif
.ifdef REMOTE_SMTP_SMARTHOST_TLS_CERTIFICATE
tls_certificate = REMOTE_SMTP_SMARTHOST_TLS_CERTIFICATE
.endif
.ifdef REMOTE_SMTP_SMARTHOST_PRIVATEKEY
tls_privatekey = REMOTE_SMTP_SMARTHOST_PRIVATEKEY
.endif

address_directory:
  debug_print = "T: address_directory for $local_part@$domain"
  driver = appendfile
  delivery_date_add
  envelope_to_add
  return_path_add
  check_string = ""
  escape_string = ""
  maildir_format

begin retry

*                      *           F,2h,15m; G,16h,1h,1.5; F,4d,6h

begin rewrite

.ifndef NO_EAA_REWRITE_REWRITE
*@+local_domains "${lookup{${local_part}}lsearch{/etc/email-addresses}\
                   {$value}fail}" Ffrs
*@ETC_MAILNAME "${lookup{${local_part}}lsearch{/etc/email-addresses}\
                   {$value}fail}" Ffrs
.endif

begin authenticators

cram_md5:
  driver = cram_md5
  public_name = CRAM-MD5
  client_name = ${extract{1}{:}{${lookup{$host}nwildlsearch{CONFDIR/passwd.
    client}{$value}fail}}}
  client_secret = ${extract{2}{:}{${lookup{$host}nwildlsearch{CONFDIR/passwd.
```

```
    client}{$value}fail}}}
857
858 PASSWDLINE=${sg{\
859                 ${lookup{$host}nwildlsearch{CONFDIR/passwd.client}{$value}fail
    }\
860           }\
861           {\\N[\\^]\\N}\
862           {^^}\
863       }
864
865 plain:
866   driver = plaintext
867   public_name = PLAIN
868 .ifndef AUTH_CLIENT_ALLOW_NOTLS_PASSWORDS
869   client_send = "<; ${if !eq{$tls_out_cipher}{}\
870                       {^${extract{1}{:}{PASSWDLINE}}\
871          ^${sg{PASSWDLINE}{\\N([^:]+:)(.*)\\N}{\\$2}}\
872        }fail}"
873 .else
874   client_send = "<; ^${extract{1}{:}{PASSWDLINE}}\
875         ^${sg{PASSWDLINE}{\\N([^:]+:)(.*)\\N}{\\$2}}"
876 .endif
877
878 login:
879   driver = plaintext
880   public_name = LOGIN
881 .ifndef AUTH_CLIENT_ALLOW_NOTLS_PASSWORDS
882   client_send = "<; ${if and{\
883                            {!eq{$tls_out_cipher}{}}\
884                            {!eq{PASSWDLINE}{}}\
885                           }\
886                       {}fail}\
887                  ; ${extract{1}{::}{PASSWDLINE}}\
888      ; ${sg{PASSWDLINE}{\\N([^:]+:)(.*)\\N}{\\$2}}"
889 .else
890   client_send = "<; ${if !eq{PASSWDLINE}{}\
891                       {}fail}\
892                  ; ${extract{1}{::}{PASSWDLINE}}\
893      ; ${sg{PASSWDLINE}{\\N([^:]+:)(.*)\\N}{\\$2}}"
894 .endif
```