CHƯƠNG 5

BUỔI THỰC HÀNH SỐ 5

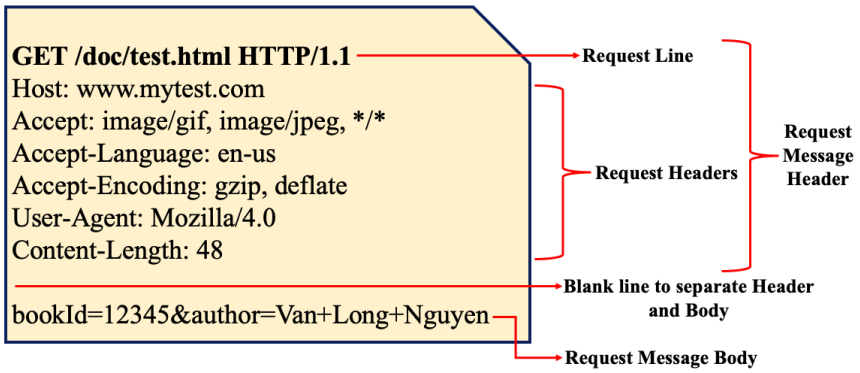
Chương này sẽ làm rõ hoạt động của một số giao thức và dịch vụ phổ biến trên tầng ứng dụng như: giao thức HTTP, giao thức DNS, giao thức SMTP, giao thức POP3 và IMAP. Các nội dung được giới thiệu trong chương gồm có: giới thiệu sơ lược về giao thức HTTP và mô hình Client – Server; giới thiệu sơ lược về giao thức DNS và minh họa triển khai đơn giản DNS trong một tổ chức ; giới thiệu sơ lược về giao thức SMTP, POP3 và IMAP cũng như các thành phần tham gia vào hệ thống thư điện tử. Cuối chương là các bài tập thực hành khảo sát những giao thức đã trình bày trên mạng ảo Kathará nhằm giúp sinh viên hiểu chi tiết về hoạt động của các giao thức đấy. Ngoài ra, trong các bài tập thực hành cũng đặt các câu hỏi giúp sinh viên ôn tập lại về giao thức TCP và UDP trên tầng vận chuyển.

5.1 CÁC GIAO THỨC TRÊN TẦNG ỨNG DỤNG

5.1.1 Giao thức HTTP

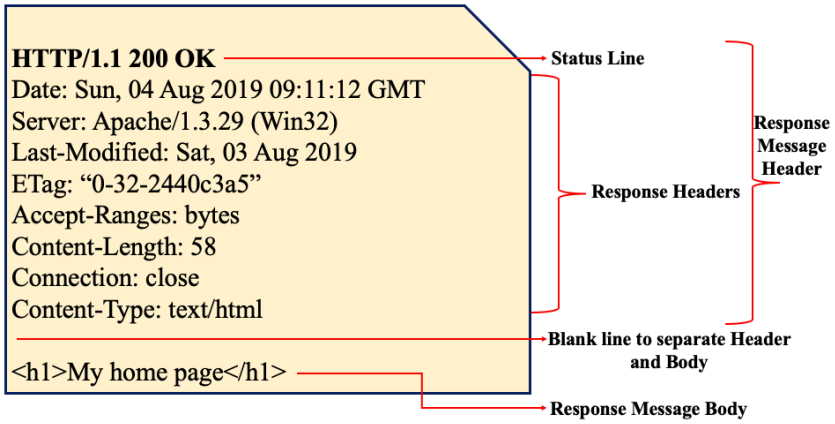
Hyper Text Transfer Protocol – HTTP là một giao thức được sử dụng trong triển khai dịch vụ World Wide Web (www) của Internet. Giao thức HTTP sử dụng kết nối TCP hình thành kênh giao tiếp giữa client và server. Trong mô hình Client – Server:

– Phía client, trình duyệt web (Google Chorme, Firefox...) sẽ truyền đi các yêu cầu dưới dạng thông điệp HTTP. Một thông điệp HTTP yêu cầu từ phía client sẽ bao gồm những thông tin cơ bản sau: thao tác nội dung (*GET, POST...*), đường dẫn URL, phiên bản HTTP sử dụng, các thông tin liên quan đến trình duyệt...



Hình 5.1 Cấu trúc thông điệp request kiểu GET được gửi đi của slient

– Phía server, các máy chủ phục vụ (web server) sẽ trả về kết quả dưới dạng thông điệp HTTP. Một thông điệp HTTP trả lời từ phía server sẽ bao gồm những thông tin cơ bản sau: phiên bản HTTP sử dụng, Mã trạng thái trả lời (200, 404...), các thông tin liên quan đến web server, nội dung client muốn...



Hình 5.2 Cấu trúc thông điệp response được gửi đi của server

5.1.2 Giao thức DNS

Hệ thống phân giải tên miền *Domain Name System – DNS* làm nhiệm vụ duy trì một tập ánh xạ các cặp `Tên luận lý` và `Địa chỉ IP` của một dịch vụ mạng được cung cấp tại một địa chỉ nào đó. Lợi ích dễ nhận thấy nhất của DNS đó là giúp cho người dùng dễ sử dụng các dịch vụ mạng hơn thông qua tên dịch vụ so với địa chỉ IP dạng nhị phân rắc rối, khó nhớ.

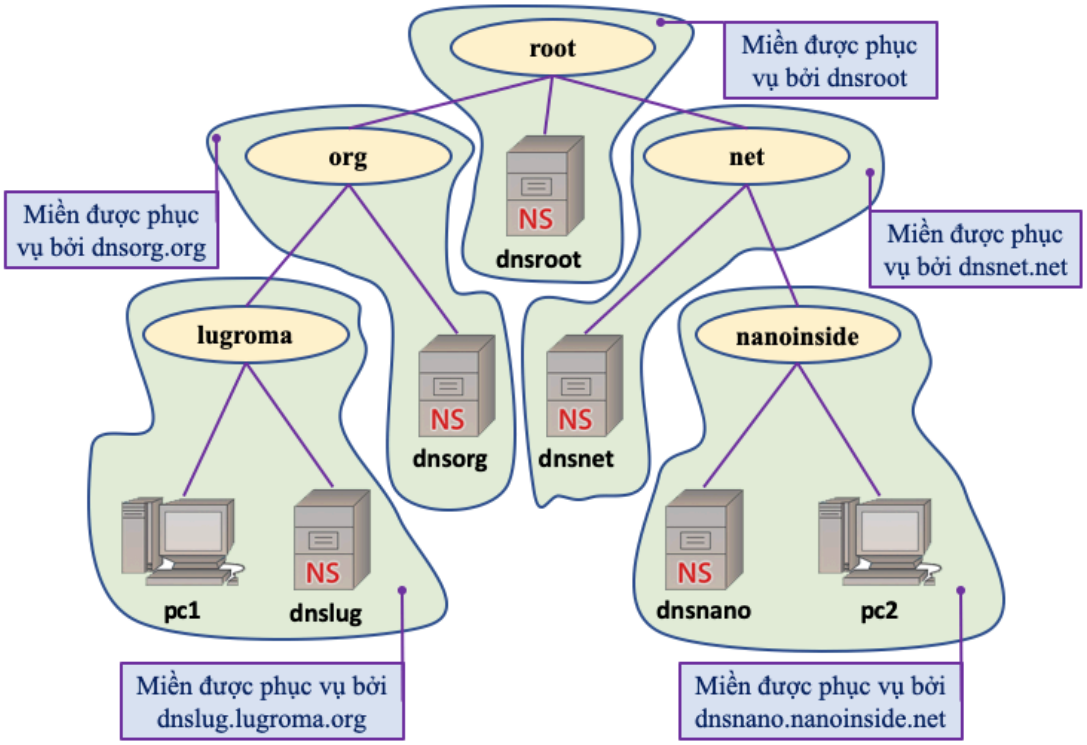
Thông thường, hệ thống phân giải tên miền của tổ chức, của toàn cầu được tổ chức dưới dạng *Miền phân cấp*. Trong đó, Miền phân cấp sẽ được đại diện bởi các máy đặc biệt, gọi là server phục vụ tên (Domain Naming Server).

Để cho thuận lợi, trong các bài tập thực hành của Chương 5 này sẽ sử dụng mô hình DNS minh họa đại diện cho một tổ chức, tạm gọi là tổ chức X. Bảng 5.1 trình bày sự phân chia miền và liên hệ của các miền trong X dựa theo cấu trúc minh họa của hình 5.3.

Tên miền	Domain Name Server của miền	Miền cha của miền	Miền con của miền	Máy tính trong miền
root	dnsroot	-	-	-
org	dnsorg.org	root	lugroma	-
net	dnsnet.net	root	nanoinside	-
lugroma	dnslug.lugroma.org	org	-	pc1

nanoinside	dnsnano.nanoinsde.net	net	-	pc2

Bảng 5.1 Bảng mô tả chức năng các thành phần trong hệ thống DNS của tổ chức



Hình 5.3 Minh họa triển khai hệ thống DNS của tổ chức X

5.1.3 Giao thức SMTP và IMAP

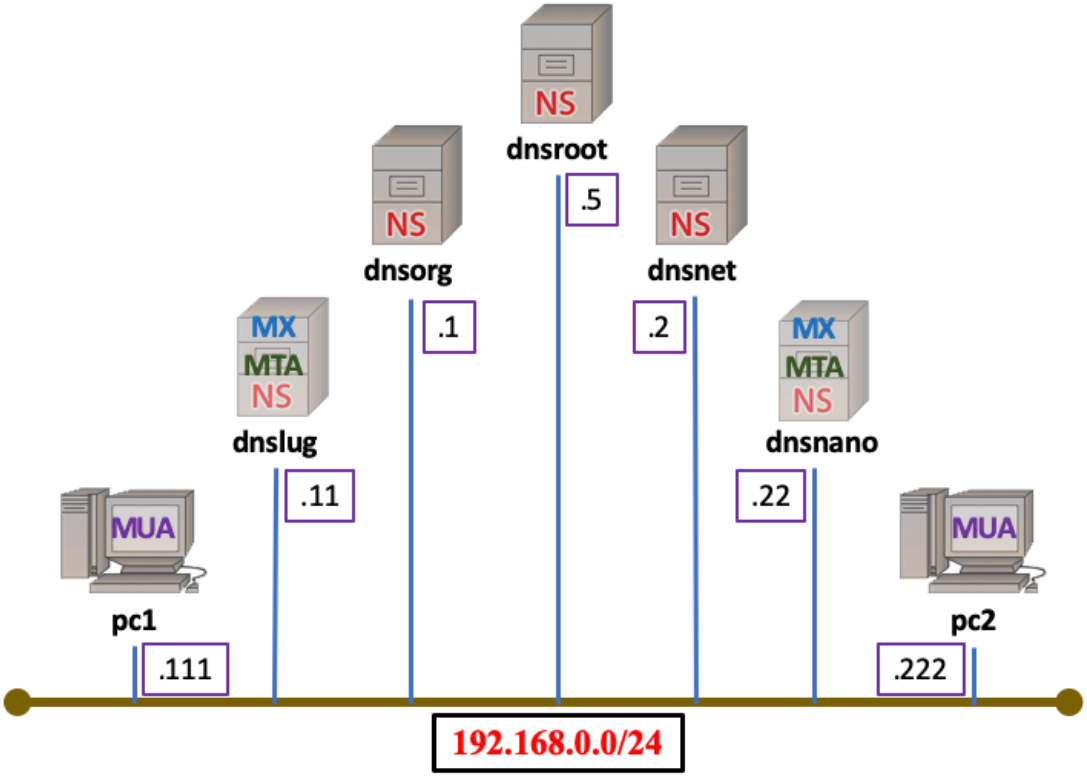
Một hệ thống thư điện tử trong một miền thuộc một tổ chức có thể được triển khai bằng các thành phần sau:

- *Mail Exchanger – MX* hay còn gọi là *Incoming Mail Server* làm nhiệm vụ tập hợp và lưu trữ các mail chuyển đến trong miền quản lý qua giao thức `POP3 (port 110)` và `IMAP (port 143)`
- *Mail Transfer Agent – MTA* hay còn gọi là *Outcoming Mail Server* làm nhiệm vụ giúp người dùng trong miền chuyển thư đến các địa chỉ mong muốn thông qua giao thức `SMTP (port 25)`
- *Mail User Agent – MUA* là 1 phần mềm cài đặt trên máy người dùng cho phép kết nối đến mail server và quản lý hộp thư đến, hộp thư đi.

Hệ thống email cho 2 miền đó là `lugroma` và `nanoinsde` của tổ chức X có thể được xây dựng như trong hình 5.4 dưới đây. Trong đó, hệ thống email này sử dụng DNS đã trình bày trong hình 5.3.

– Nhằm đơn giản mô hình DNS của tổ chức X nên tất cả các thành phần mạng của tổ chức X đều nằm trong cùng một mạng LAN 192.168.0.0/24 hay chung một vùng đụng độ.

– *MX* và *MTA* hay *name sever* có thể được triển khai trên cùng một máy tính. *MUA* sẽ được triển khai trên các pc.



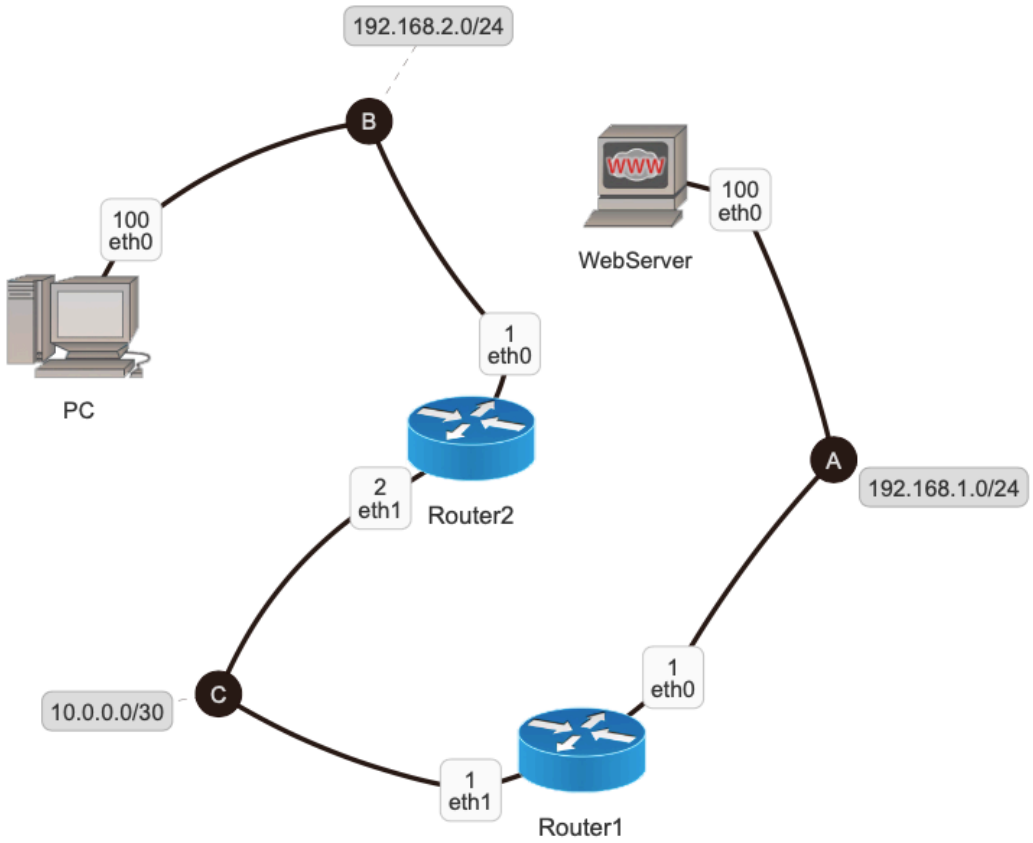
Hình 5.4 Hệ thống thư điện tử của tổ chức X trên cùng một vùng đụng độ

5.2 BÀI TẬP THỰC HÀNH

5.2.1 Bài tập 15

Mục tiêu: Xây dựng mô hình mạng triển khai giao thức HTTP theo kiến trúc Client – Server. Các bước thực hiện *Bài tập 15* được trình bày chi tiết như sau:

1) Quan sát mô hình mạng cần xây dựng. Nhận diện các thiết bị (PC, Router...), giao diện (eth0, eth1...) với các địa chỉ IP được gán.



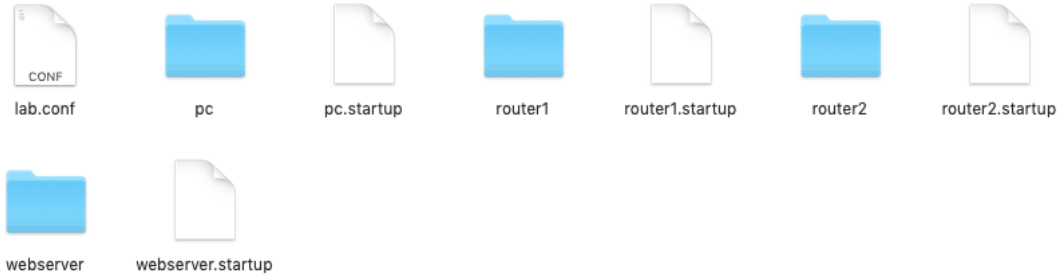
Hình 5.5 Mô hình mạng sử dụng trong Bài tập 15

2) Tạo thư mục *BaiTap15* trong workspace của sinh viên. Thư mục sẽ này chứa các thư mục con và các file cấu hình (`.startup, lab.conf`) theo cấu trúc quy định của *Kathará*.

   Trên máy thực, di chuyển đến thư mục *BaiTap15* bằng lệnh:

   ```
   cd /home/student/your_workspace/BaiTap15
   ```

   Cấu trúc thư mục *BaiTap15* được miêu tả như hình 5.6



Hình 5.6 Các thư mục con và tệp tin trong thư mục BaiTap15

3) Trên file *`Lab.conf`*, soạn thảo nội dung mô tả hình thái mạng theo thiết kế

4) Trên các file `.startup` của các router, soạn thảo nội dung cấu hình cho giao diện mạng và thêm thông tin vạch đường tĩnh. Nội dung file router1.startup có thể tham khảo:

```
ifconfig eth0 192.168.1.1/24 up
ifconfig eth1 10.0.0.1/30 up
route add -net 192.168.2.0/24 gw 10.0.0.2
```

5) Trên các file .startup của pc và webserver, soạn thảo nội dung cấu hình cho giao diện mạng và thêm vào thông tin vạch đường mặc nhiên. Nội dung file pc.startup có thể tham khảo như sau:

```
ifconfig eth0 192.168.2.100/24 up
route add default gw 192.168.2.1
```

6) Khởi động mạng ảo *BaiTap15*.

7) Để `webserver` có thể phục vụ và cung cấp các trang web cho `pc` và các máy tính khác truy cập tới thì trên `webserver` phải khởi động một phần mềm đặc biệt là `apache2` bằng lệnh:

```
/etc/init.d/apache2 start
```

– `apache2` là một trong nhiều công cụ xây dựng `webserver`, giấy phép thuộc về *Apache Licenses*[1].

– Khi `apache2` khởi động xong, `webserver` đã sẵn sàng cung cấp 1 trang web cho `pc` truy cập đến. Nội dung mặc nhiên của trang web là "`It works!`"

– Nội dung và cấu trúc trang web có thể thay đổi theo nhu cầu triển khai `webserver` của người dùng.

– Nội dung trang web có thể được tìm thấy trong tập tin `/var/www/index.html`

8) Trên `pc`, mở trình duyệt web đơn giản bằng lệnh: `links`

Trình duyệt web này nhỏ gọn, không có giao diện đồ họa, phù hợp với kích thước máy ảo *Kathará*[2]. Giao diện của `links` trong máy ảo *Kathará* khi khởi động xong là màn hình đen.

[1] https://httpd.apache.org

[2] http://links.twibright.com

9) Trên `webserver`, dùng lệnh `tcpdump` để lắng nghe các gói tin sẽ gửi đến từ máy ảo `pc`:

```
tcpdump -s 1536 -w /hostlab/BT15_webserver.pcap
```

10) Trong `links` của `pc`, thực hiện các thao tác sau để truy cập đến trang web mặc nhiên đang có trên `webserver`:

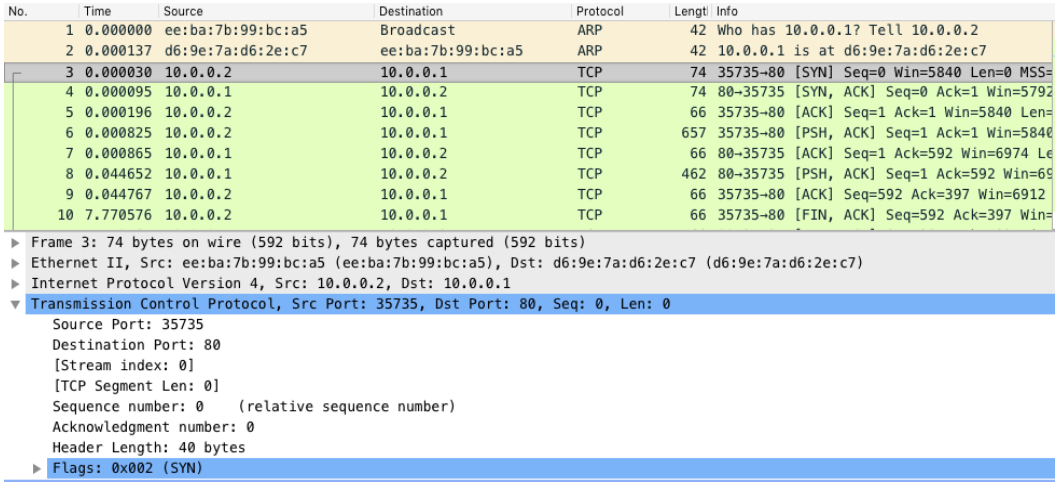
– Sử dụng `F10` để chuyển tới `Menu Bar`

– Chọn tiếp `Go to URL` và nhập vào `http://192.168.1.100/`. Đây là địa chỉ IP của `web server`. Do *Bài tập 15* không xây dựng DNS nên không thể nhập địa chỉ dưới dạng tên luận lý mà phải nhập trực tiếp địa chỉ IP.

– Kết quả hiển thị là trang chủ của `webserver`.

11) Đóng trình duyệt `links` trên pc. Dừng lệnh `tcpdump` trên `webserver` lại.

12) Trên máy thực, dùng *WireShark* mở tập tin `BT15_webserver.pcap` đã ghi nhận được.



Hình 5.7 Thông tin file BT15_webserver.pcap hiển thị trên Wireshark

– Chọn khung vật lý số 3 và mở *Tranmisson Control Protocol Header* trong khung này:

  ✓ Trình duyệt `links` đang hoạt động ở cổng bao nhiêu?

  ✓ `apache2` của `webserver` đang hoạt động ở cổng bao nhiêu?

  ✓ Cờ SYN đang được bật lên (bit SYN có giá trị bằng 1). Nêu nhiệm vụ của gói tin SYN trong giao thức bắt tay 3 chiều của TCP.

– Chọn khung vật lý số 4 và mở *Tranmisson Control Protocol Header* trong khung này:

- ✓ Cờ SYN và cờ ACK đang được bật lên. Nêu nhiệm vụ của gói tin (SYN, ACK) trong giao thức bắt tay 3 chiều của TCP.

– Chọn khung vật lý số 5 và mở *Tranmisson Control Protocol Header* trong khung này:

- ✓ Cờ ACK đang được bật lên. Nêu nhiệm vụ của gói tin ACK trong giao thức bắt tay 3 chiều của TCP.

– Chọn khung vật lý số 6:

- ✓ Mở *Tranmisson Control Protocol Header*: cờ PUSH có được bật lên không? Tại sao? Từ đó giải thích ý nghĩa của cờ PUSH trong giao thức TCP.
- ✓ Mở *HTTP Header*: hãy cho biết thông điệp HTTP gửi đi có dạng là GET, POST, DELETE hay dạng gì? Sinh viên tự tìm hiểu thêm thông tin về các trường *User-Agent, Accept-Encoding*, *Accept-Charset*, *Accept-Language*.

– Chọn khung vật lý là phản hồi đầu tiên cho khung vật lý số 6:

- ✓ Mở *HTTP Header*: hãy cho biết thông điệp HTTP trả lời có mã là bao nhiêu (200, 404, 502..)? Sinh viên tự tìm hiểu thêm thông tin về các trường *Date*, *Server*, *Content-Encoding*, *Content-Length*, *Connection-Type* và *Connection*.

– Chọn khung vật lý được gửi từ `webserver` và cờ FIN được bật lên:

- ✓ Hãy cho biết nhiệm vụ của gói tin FIN này trong cơ chế giải phóng kết nối 3 chiều của giao thức TCP
- ✓ Hãy chỉ ra số thứ tự của các khung vật lý còn lại tham gia vào quá trình giải phóng kết nối 3 chiều này.

13) Hủy mạng ảo bằng lệnh `lwipe` sau khi đã thực hiện xong *Bài tập 15*

5.2.2 Bài tập 16

Mục tiêu: Quan sát hoạt động của giao thức DNS trong mô hình tên miền phân cấp của tổ chức X. Do việc cấu hình DNS tương đối phức tạp nên sinh viên sẽ sử dụng thư mục mạng ảo *Kathará* về DNS được cung cấp sẵn kèm theo. Các bước thực hiện *Bài tập 16* được trình bày chi tiết như sau

1) Quan sát mô hình mạng cần xây dựng. Nhận diện các thiết bị (PC, Router...), giao diện (eth0, eth1...) với các địa chỉ IP được gán.

– Sử dụng lại mô hình DNS của tổ chức X đã giới thiệu ở hình 5.3 và 5.4.

– Địa chỉ IP được gán theo hướng dẫn trên mô hình mạng của hình 5.4

2) Sao chép thư mục *BaiTap16* được cung cấp vào workspace của sinh viên. Thư mục sẽ này chứa các thư mục con và các file cấu hình (`.startup`, `lab.conf`) theo cấu trúc quy định của *Kathará*.

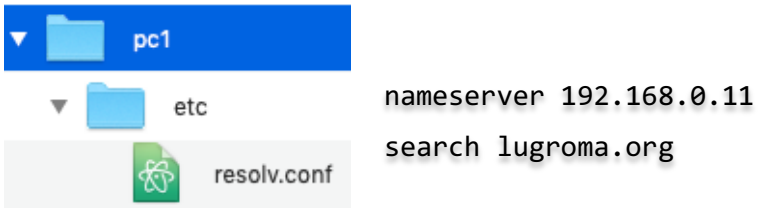
   Trên máy thực, di chuyển đến thư mục *BaiTap16* bằng lệnh:

   ```
   cd /home/student/your_workspace/BaiTap16
   ```

   Cấu trúc thư mục *BaiTap16* được miêu tả như hình 5.8

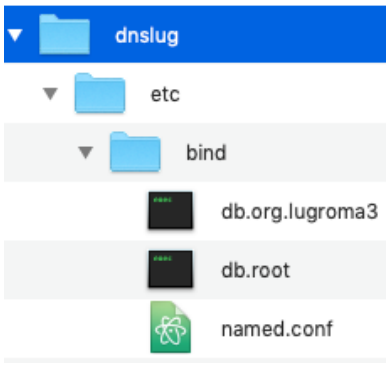
Hình 5.8 Các thư mục con và tệp tin trong thư mục BaiTap16

3) Quan sát tổ chức của thư mục `pc1` như trong hình 5.9 dưới đây. File `resolv.conf` chứa nội dung khai báo Domain Naming Server của miền mà `pc1` phụ thuộc.



Hình 5.9 Cấu trúc thư mục máy ảo pc1 và nội dung file resolv.conf

4) Quan sát tổ chức của thư mục `dnslug` như trong hình 5.10 dưới đây. Thư mục `/etc/bind/` đại diện cho các thiết lập của dịch vụ `bind`, một dịch vụ hỗ trợ triển khai DNS trên Domain Naming Server.



Hình 5.10 Các thư mục con và tệp tin trong thư mục /etc/bind của dịch vụ bind

Các file trong thư mục /etc/bind lần lượt có các chức năng sau:

- File named.conf: đóng vai trò tạo liên kết (association) giữa Domain Naming Server dnslug và vùng mà nó quản lý là lugroma.org
- File db.root: chứa thông tin liên quan đến Root Name Server dnsroot
- File db.org.lugroma: chứa các thông tin liên quan đến Domain Naming Server của miền phân cấp cha là dns.org (thuộc miền org). Các thông tin được thể hiện thường là:
  - ✓ Thông tin thẩm quyền (autoritative info) từ dnslug đến dnsorg
  - ✓ Thông tin ánh xạ giữa địa chỉ IP và tên luận lý tương ứng. Chẳng hạn:

```
@               IN      NS      dnslug.lugroma.org.
dnslug          IN      A       192.168.0.11
pc1             IN      A       192.168.0.111
```

5) Khởi động mạng ảo *BaiTap16* bằng lệnh lstart
6) Trên máy ảo pc2, dùng lệnh:

```
tcpdump -n -t port domain -w /hosthome/BT16_pc2.pcap
```

7) Trên máy ảo pc1, gửi dữ liệu đến pc2 bằng lệnh:

```
ping pc2.nanoinside.net
```

Nhờ DNS được triển khai trong tổ chức X mà lệnh ping có thể sử dụng tên luận lý của máy tính (pc2.nanoinside.net) để gửi dữ liệu đến thay vì sử dụng địa chỉ IP (192.168.0.222) như lúc trước.

8) Dừng lệnh tcpdump đang thực hiện ở pc2 lại. Trên máy thực, mở file BT16_pc2_dns.pcap bằng *Wireshark* và trả lời các câu hỏi sau đây:

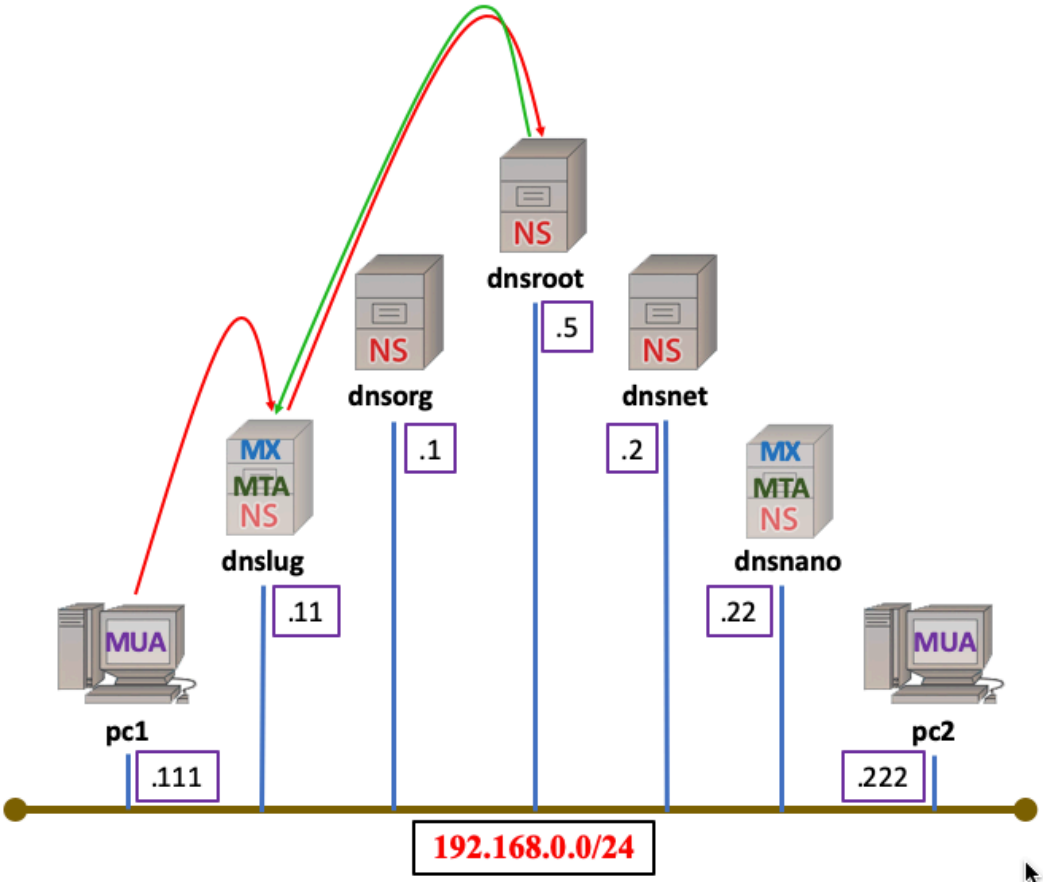
- Trước khi pc1 gửi dữ liệu đến pc2 thì Domain Naming Server dnslug của miền lugroma.org có thông tin gì về miền nanoinside.net hay không?
- Chọn gói tin DNS số 1:
  - ✓ Gói tin này được gửi từ máy nào đến máy nào (địa chỉ IP)?
  - ✓ Giao thức mà gói tin truyền đi trên tầng vận chuyển là gì?
  - ✓ Dựa vào *User Datagram Protocol Header*, cho biết cổng (port) hoạt động của DNS?
  - ✓ Gói tin này là truy vấn thông tin (query) hay trả lời (response)?
  - ✓ Nếu gói tin là query, hãy cho biết thông tin muốn truy vấn là gì? Chẳng hạn: dnslug truy vấn root địa chỉ IP của pc2.nanoinside.net là bao nhiêu?

✓ Nếu là gói tin `response`, hãy cho biết thông tin trả lời là gì? Chẳng hạn: trả lời rằng không biết địa chỉ IP của `pc2.nanoinside.net` nhưng biết được địa chỉ IP và Domain Naming Server của miền chứa máy tính đó là `dnsnet` trong miền `nanoinside.net`

– Trả lời các câu hỏi tương tự trên các gói tin DNS số 2, 4, 6, 7, 8 và 9.

– Hãy cho biết kết quả đạt được trên Domain Naming Server `dnslug` của miền `lugroma.org` sau quá trình trao đổi các gói tin DNS ở trên. Chẳng hạn: Domain Naming Server `dnslug` của miền `lugroma.org` đã biết được ai là Domain Naming Server của miền `.net`, của miền `nanoinside.net`, địa chỉ IP của máy tính `pc2.nanoinside.net`...

9) Vẽ Sequence Diagram thể hiện cho hoạt động của hệ thống DNS trong tổ chức X qua việc trao đổi các gói tin DNS 1, 2, 4, 6, 7, 8 và 9 đã khảo sát. Hình 5.11 dưới đây thể hiện việc trao đổi gói tin DNS số 1 và 2. Trong đó, màu đỏ là các gói tin `query`; màu xanh (sinh viên tự vẽ thêm) là các gói tin `response`.



Hình 5.11 Sequence diagram mô tả hoạt động của dịch vụ DNS trong tổ chức X

10) Khảo sát tính năng của *Named Server cache*:

– Trên `pc1` thực hiện lại lệnh ping:

```
ping pc2.nanoinside.net
```

– Hãy mô tả loại hoạt động của Domain Naming Server `dnslug` trong trường hợp này.

  ✓ Gợi ý: lúc này `dnslug` có cần phải gửi các gói tin `query` nữa hay không? Các gói tin 1, 2, 4, 6, 7, 8 và 9 có xuất hiện lại khi bắt gói tin trên `pc2` bằng *Wireshark* hay không?

– Vẽ lại Sequence Diagram để mô tả cho hoạt động của DNS.

11) Một số khảo sát khác mà sinh viên có thể tự thực hiện để hiểu rõ hơn về giao thức DNS:

– Khảo sát *Named Server cache* khi khởi động lại dịch vụ `bind` trên `dnslug` bằng lệnh: `/etc/init.d/bind restart`

– Khảo sát *Named Server Negative cache* khi thực hiện gửi dữ liệu đến địa chỉ không tìm thấy hoặc không tồn tại. Ví dụ, trên `pc1` thực hiện:

```
ping notexist.nanoinside.net
```

12) Hủy mạng ảo bằng lệnh `lwipe` sau khi đã thực hiện xong *Bài tập 16.*

5.2.3 Bài tập 17

Mục tiêu: Quan sát hoạt động của SMTP và IMAP trong dịch vụ thư điện tử. Các bước thực hiện *Bài tập 17* được trình bày chi tiết như sau:

1) Quan sát mô hình mạng cần xây dựng. Nhận diện các thiết bị (PC, Router...), giao diện (eth0, eth1...) với các địa chỉ IP được gán.

– Sử dụng lại mô hình DNS của tổ chức X đã giới thiệu ở hình 5.3 và 5.4.

– Địa chỉ IP được gán theo hướng dẫn trên mô hình mạng của hình 5.4

2) Sao chép thư mục *BaiTap17* được cung cấp vào workspace của sinh viên. Thư mục sẽ này chứa các thư mục con và các file cấu hình (`.startup`, `lab.conf`) theo cấu trúc quy định của *Kathará*.

   Trên máy thực, di chuyển đến thư mục *BaiTap17* bằng lệnh:

```
cd /home/student/your_workspace/BaiTap17
```

   Cấu trúc thư mục *BaiTap17* được miêu tả như hình 5.12

Hình 5.12 Các thư mục con và tệp tin trong thư mục BaiTap17

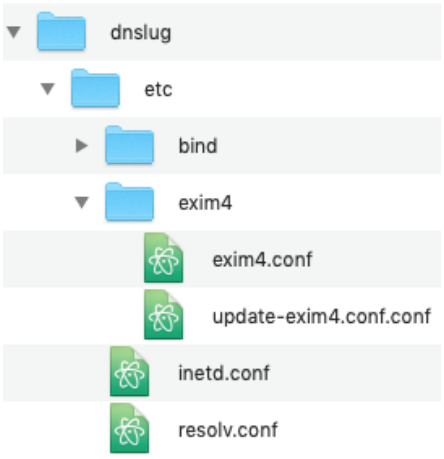
3) Các thiết lập đã thực hiện cho mô hình DNS ở *Bài tập 16* được sử dụng lại và mở rộng thêm ở một số điểm như: khai báo MTA, khai báo MX...

4) Quan sát tổ chức của thư mục `dnslug` như trong hình 5.14 dưới đây. Thư mục này chứa một số thành phần đặc biệt như sau:

– Thư mục `exim4` đại diện cho các thiết lập của dịch vụ `exim4`, một dịch vụ hỗ trợ triển khai *Mail Transfer Agent MTA* với giao thức SMTP.

– File `inetd.conf` chứa các thiết lập cần thiết cho 2 dịch vụ là `imapd` và `ipop3d`. Đây là 2 dịch vụ tương ứng với 2 giao thức IMAP và POP3 cần triển khai trên một *Mail Exchanger MX*.

– Như vậy, ta có thể kết luận rằng: `dnslug` vừa là Domain Naming Server, vừa là MTA và là MX đúng với thiết kế được yêu cầu trong hình 5.4.



Hình 5.13 Các thư mục con và tệp tin trong thư mục dnslug/etc/

5) Quan sát tổ chức của thư mục `shared` như trong hình 5.15 dưới đây. Thư mục `shared` cho phép tạo ra các thiết lập chung và áp dụng các thiết lập đó trên những máy ảo *Kathará* được chỉ định. Cụ thể trong Bài tập này `shared` được áp dụng cho `pc1` và `pc2`. Thư mục `shared` chứa một số thành phần đặc biệt như sau:

Hình 5.14 Các thư mục con và tệp tin trong thư mục shared/usr/

– File `pine` là phần mềm *pine* được cài đặt vào bên trong `pc`.

– File `imapd` và `ipop3d` lần lượt là dịch vụ *imapd* và *ipop3d* được cài đặt vào bên trong `pc`. Việc cài đặt này giúp cho các `pc`, với vai trò là các *Mail User Agent MUA*, có thể trao đổi với các MX bằng các thao tác như Duyệt hộp thư, Gửi thư...qua giao thức POP3 hoặc IMAP.

6) Khởi động mạng ảo *BaiTap17* bằng lệnh `lstart`.

7) Trên `dnsroot` dùng lệnh:

```
tcpdump -i eth0 -t -q port domain or port smtp -w /hosthome/BT17_dnsroot.pcap
```

8) Trên `pc1`, mở phần mềm *pine* bằng lệnh: `pine`

Nhập vào mật khẩu là `guest` (nếu có yêu cầu). Hình 5.15 dưới đây mô tả giao diện chính của phần mềm *pine*.

```
PINE 4.64   MAIN MENU                          Folder: INBOX  No Messages

    ?     HELP              -  Get help using Pine
    C     COMPOSE MESSAGE   -  Compose and send a message
    I     MESSAGE INDEX     -  View messages in current folder
    L     FOLDER LIST       -  Select a folder to view
    A     ADDRESS BOOK      -  Update address book
    S     SETUP             -  Configure Pine Options
    Q     QUIT              -  Leave the Pine program
```

Hình 5.15 Giao diện Main Menu của phần mềm pine

9) Chọn `COMPOSE MESSAGE` để để soạn 1 email có nội dung đơn giản như sau: "*Xin chao, toi la pc1. Tam biet!!*"

– Trường `TO` của email thì nhập vào địa chỉ: `guest@nanoinside.net`. Đây là địa chỉ hộp thư điện tử của người dùng `guest` trên `pc2`.

– Dùng tổ hợp phím `Ctrl + X` để gửi mail đi.

10) Trên `dnsroot`, dừng lệnh `tcpdump` lại. Mở file `BaiTap16_dnsroot_smtp.pcap` bằng *Wireshark* và trả lời các câu hỏi sau nhằm làm rõ hoạt động có sự kết hợp giữa giao thức DNS và SMTP.

– *Hoạt động 1*: `pc1` tìm kiếm thông tin về MTA của miền chứa nó (`lugroma.org`)

  ✓ `pc1` hỏi ai về thông tin của MTA trong miền `lugroma.org`? Chỉ ra gói tin số mấy thể hiện hoạt động này.

  ✓ Câu trả lời dành cho truy vấn của `pc1` nằm ở gói tin số máy? Câu trả lời này cho biết ai là MTA của `pc1`?

– *Hoạt động 2*: `pc1` kết nối đến `MTA` trước khi có thể gửi mail đi

  ✓ Việc thực hiện kết nối của `pc1` đến `MTA` diễn ra theo hình thức nào trên tầng vận chuyển? Chỉ ra các gói tin thể hiện hình thức kết nối này.

– *Hoạt động 3*: `pc1` chuyển mail đến MTA và trông cậy vào việc MTA sẽ chuyển được mail này đến MTA của địa chỉ nhận mail.

  ✓ Giao thức trên tầng ứng dụng mà `pc1` sử dụng sử dụng để chuyển mail đến MTA của nó là gì?

  ✓ Cho thông điệp mẫu giữa một máy tính và MTA như hình 5.16 dưới đây. Hãy chỉ những gói tin tương ứng với hoạt động của `pc1` và MTA căn cứ theo thông điệp mẫu. Lưu ý: thông điệp mẫu này chỉ mới thể hiện việc nhận `Header` của thư, tức là địa chỉ gửi, địa chỉ nhận chứ chưa thể hiện việc nhận nội dung của thư.

```
S: 220 smtp.example.com ESMTP Postfix
C: HELO relay.example.com
S: 250 smtp.example.com, I am glad to meet you
C: MAIL FROM:<bob@example.com>
S: 250 Ok
C: RCPT TO:<alice@example.com>
S: 250 Ok
```

Hình 5.16 Thông điệp mẫu giao tiếp giữa pc1 (C) và MTA (S). Nguồn: Wikipedia

- ✓ Gợi ý: các thông điệp sẽ chứa từ khóa theo khuôn mẫu của SMTP như: `HELO`, `MAIL FROM`, `RCPT TO`...
- ✓ Ví dụ: Thông điệp `MAIL FROM` từ `pc1` đến MTA nằm trong gói tin 10

– *Hoạt động 4:* Sau khi nhận xong `Header` của thư từ `pc1`, MTA (đồng thời là Domain Name Server) thấy rằng đích đến của mail là: `guest@nanoinside.net`. MTA sẽ gửi đi các DNS `query` đến các Domain Naming Server thuộc các miền khác cho đến khi MTA biết được MTA đích nằm ở đâu.

- ✓ Hãy chỉ ra các gói tin DNS mà MTA dùng để truy vấn các Domain Naming Server đó. Gợi ý: xem lại 8) và 9) *Bài Tập 16*
- ✓ MTA đích là ai?

– *Hoạt động 5:* Nếu như MTA tìm thấy MTA của đích đến thì nó sẽ trả về cho `pc1` thông điệp là `250 Accepted`. Lúc này `pc1` có thể gửi nội dung thư cho MTA.

- ✓ Hãy chỉ ra gói tin số mấy thể hiện cho mã lệnh `250 Accepted`
- ✓ Hãy chỉ ra các gói tin thể hiện cho việc gửi nội dung thư từ `pc1` đến MTA. Gợi ý: các gói tin đó sẽ chứa thông điệp `FETCH`

– *Hoạt động 6*: `pc1` và MTA giải phóng kết nối trên tầng vận chuyển.

- ✓ Hãy chỉ ra các gói tin thể hiện hoạt động này

– *Hoạt động 7*: Lúc này MTA của `pc1` đã biết được địa chỉ của MTA đích nhờ vào kết quả của Hoạt động 4. Vì vậy, MTA của `pc1` khởi tạo kết nối TCP đến MTA đích.

- ✓ Hãy chỉ ra các gói tin thể hiện cho hoạt động này

– *Hoạt động 8*: 2 MTA này sử dụng giao thức SMTP để trao đổi thư. Khi không còn nội dung nào cần trao đổi nữa thì giao dịch SMTP giữa 2 MTA được kết thúc bằng cú pháp `QUIT`.

- ✓ Hãy chỉ ra các gói tin thể hiện cho việc gửi nội dung thư từ MTA nguồn sang MTA đích
- ✓ Hãy chỉ ra gói tin thể hiện cho mã lệnh `QUIT`.

– *Hoạt động 9*: 2 MTA giải phóng kết nối TCP.

- ✓ Hãy chỉ ra các gói tin thể hiện cho thao tác giải phóng kết nối này.

11) Vẽ Sequence Diagram miêu tả lại các hoạt động đã thực hiện khảo sát về DNS và SMTP. Gợi ý: tham khảo lại Sequence Diagram đã vẽ minh họa cho giao thức DNS trong *Bài tập 16*.

12) Trên `dnsroot` dùng lệnh:

```
tcpdump -i eth0 -t -q port domain or port smtp -w /hosthome/BT17_dnsroot_imap.pcap
```

13) Trên `pc2` thực hiện các thao tác sau:

- Đăng nhập vào `pine` bằng tài khoản `guest`.
- Chọn `FOLDER LIST` rồi chọn `INBOX` để xem danh sách các thư điện tử của `pc2` đang được lưu trữ tại MX của nó.
- Chọn thư điện tử cần đọc (chỉ có 1 thư lúc nãy mà `pc1` đã gửi đi)
- Kiểm tra nội dung có trùng khớp với nội dung đã soạn thảo ở 9) không?

14) Trên dnsroot dừng lệnh `tcpdump` lại và mở file `BaiTap16_dnsroot_imap.pcap` bằng *Wireshark* và trả lời các câu hỏi sau nhằm làm rõ hoạt động có sự kết hợp giữa giao thức DNS và IMAP.

- *Hoạt động 1*: `pc2` tìm kiếm thông tin về MX của miền chứa nó (`nanoinside.net`).
  - ✓ `pc2` hỏi ai về thông tin của MX trong miền `nanoinside.net`? Chỉ ra gói tin số mấy thể hiện hoạt động này.
  - ✓ Câu trả lời dành cho truy vấn của `pc2` nằm ở gói tin nào? Câu trả lời này cho biết ai là MX của `pc2`?
- *Hoạt động 2*: `pc2` tạo một phiên kết nối đến MX
  - ✓ Việc thực hiện kết nối của `pc2` đến MX diễn ra theo hình thức nào trên tầng vận chuyển? Chỉ ra các gói tin thể hiện cho hình thức kết nối này.
- *Hoạt động 3*: `pc2` lấy về danh sách thư có trong `INBOX` của nó trên MX
  - ✓ Giao thức (tầng ứng dụng) sử dụng để truy cập vào `INBOX` và về lấy danh sách thư điện tử trên MX là gì?
  - ✓ Cho thông điệp mẫu giữa một máy tính và MX như hình 5.18. Hãy chỉ ra các gói tin tương ứng với hoạt động của `pc2` và MX căn cứ theo thông điệp mẫu.
  - ✓ Các gói tin thể hiện quá trình chứng thực (đăng nhập) của `pc2` vào hộp thư trên MX

- ✓ Gói tin yêu cầu mở hộp thư điện tử của `pc2`. ***Gợi ý:*** thông điệp có từ khóa `SELECT`
- ✓ Gói tin lấy về số lượng thư trong hộp thư của `pc2`. ***Gợi ý:*** thông điệp có từ khóa `EXISTS`
- ✓ Gói tin lấy về phần nội dung thư cần đọc của `pc2`. Gợi ý: thông điệp có từ khóa `FETCH … BODY`

15) Vẽ Sequence Diagram miêu tả lại các hoạt động đã thực hiện khảo sát về DNS và IMAP. Gợi ý: tham khảo lại Sequence Diagram đã vẽ minh họa cho giao thức DNS trong *Bài tập 16*.

16) Kết luận về hoạt động của dịch vụ thư điện tử kết hợp với mô hình DNS của tổ chức X.

17) Hủy mạng ảo bằng lệnh `lwipe` sau khi đã thực hiện xong *Bài tập 17*.

```
C: <open connection>
S:   * OK IMAP4rev1 Service Ready
C:   a001 login mrc secret
S:   a001 OK LOGIN completed
C:   a002 select inbox
S:   * 18 EXISTS
S:   * FLAGS (\Answered \Flagged \Deleted \Seen \Draft)
S:   * 2 RECENT
S:   * OK [UNSEEN 17] Message 17 is the first unseen message
S:   * OK [UIDVALIDITY 3857529045] UIDs valid
S:   a002 OK [READ-WRITE] SELECT completed
C:   a003 fetch 12 full
S:   * 12 FETCH (FLAGS (\Seen) INTERNALDATE "17-Jul-1996 02:44:25 -0700"
      RFC822.SIZE 4286 ENVELOPE ("Wed, 17 Jul 1996 02:23:25 -0700 (PDT)"
      "IMAP4rev1 WG mtg summary and minutes"
      (("Terry Gray" NIL "gray" "cac.washington.edu"))
      (("Terry Gray" NIL "gray" "cac.washington.edu"))
S:   * 12 FETCH (BODY[HEADER] {342}
S:   Date: Wed, 17 Jul 1996 02:23:25 -0700 (PDT)
S:   From: Terry Gray <gray@cac.washington.edu>
S:   Subject: IMAP4rev1 WG mtg summary and minutes
S:   To: imap@cac.washington.edu
S:   a004 OK FETCH completed
C    a005 store 12 +flags \deleted
S:   * 12 FETCH (FLAGS (\Seen \Deleted))
S:   a005 OK +FLAGS completed
C:   a006 logout
S:   * BYE IMAP4rev1 server terminating connection
S:   a006 OK LOGOUT completed
```

Hình 5.17 Thông điệp mẫu giao tiếp giữa pc2 (C) và MX (S). Nguồn: Wikipedia

CHƯƠNG 6
BUỔI THỰC HÀNH SỐ 6

Trong chương cuối cùng, chúng tôi sẽ giới thiệu về Định tuyến liên miền và vai trò của định tuyến liên miền trong hình thành mạng Internet. Chương này bao gồm các nội dung chính: giới thiệu về định tuyến liên miền; một số những đặc trưng khác biệt giữa định tuyến liên miền và nội miền; sử dụng giao thức BGP trên dịch vụ Quagga để triển khai định tuyến liên miền theo mô hình Stub đơn giản. Ngoài ra, trong chương này cũng giới thiệu các bài tập thực hành nhằm mục đích ôn tập và mở rộng các kiến thức thực hành đã được giới thiệu trong những buổi trước.

6.1 ĐỊNH TUYẾN LIÊN MIỀN – INTERDOMAIN ROUTING

Mạng Internet là một mạng diện rộng (wide area network – WAN) được hình thành từ những miền tự trị AS có các cơ chế, chính sách quản lý và mục tiêu vận hành khác nhau. Mỗi một AS sẽ lựa chọn và sử dụng một hoặc nhiều loại giải thuật vạch đường nội miền phù hợp (RIP, OSPF...) để áp dụng cho các khu vực trong miền của mình. Các giải thuật định tuyến liên miền (Exterior Gateway Protocol – EGP) chính là giải pháp để giúp các AS có thể kết nối lại với nhau và kết nối vào mạng Internet.

Thông thường, mỗi một AS sẽ lựa chọn một Router làm đại diện và áp đặt EGP lên Router đó chứ không phải tất cả các Router hoạt động trong miền. Điều này sẽ giúp làm giảm sự phức tạp khi tạo kết nối giữa các AS với nhau. Trong số các giải thuật EGP đã được công nhận thì *Border Gateway Protocol – BGP* là một giải thuật được sử dụng rất phổ biến và sẽ được sử dụng để minh họa cho phần thực hành Mạng máy tính CT112.

Một số những đặc trưng khác biệt giữa Định tuyến nội miền IGP và Định tuyến liên miền EGP được trình bày trong bảng 6.1 dưới đây:

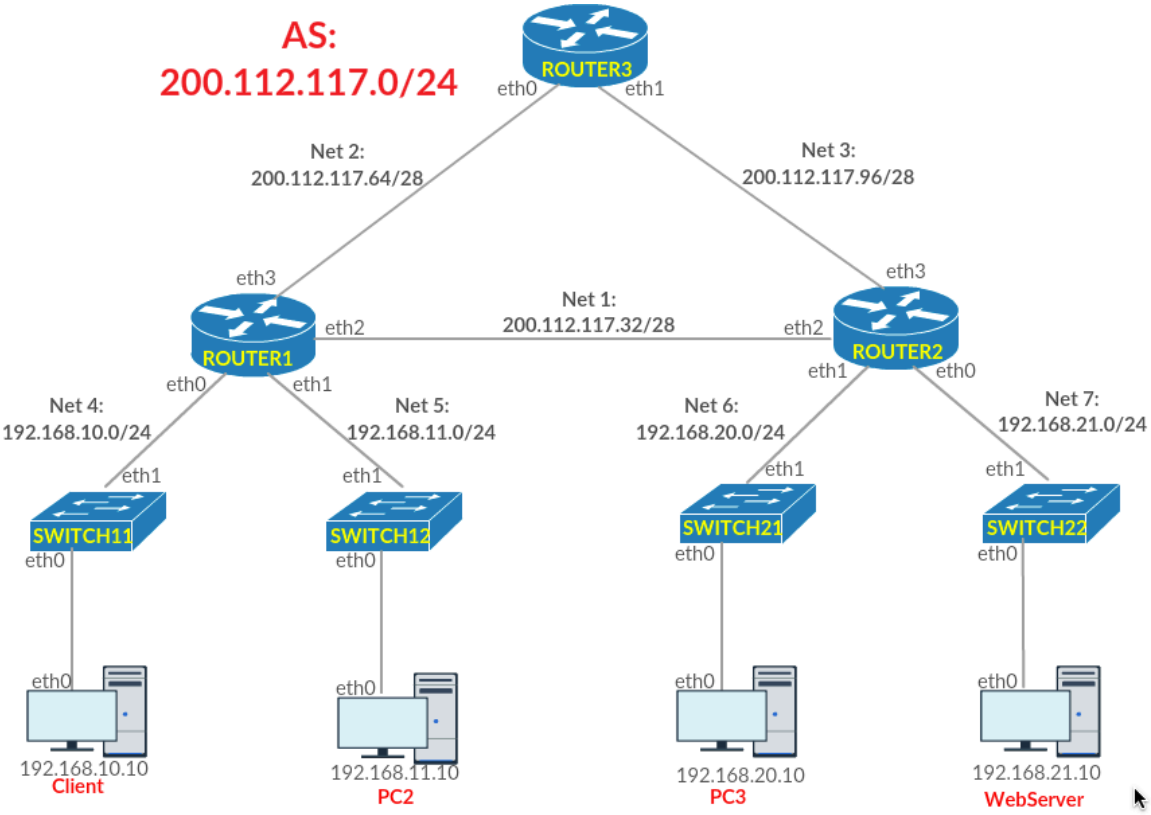
Số thứ tự	Định tuyến nội miền	Định tuyến liên miền
1	Giao thức vạch đường chỉ hoạt động trong phạm vi của miền	Giao thức vạch đường hoạt động trong phạm vi của miền hoặc giữa các miền với nhau

2	Các Router trong miền chỉ cần biết đến các Router khác cũng hoạt động trong cùng miền	Router làm đại diện của miền biết đến các Router trong cùng miền và các Router đại diện của các miền kết nối đến
3	Giao thức vạch đường không quan tâm đến kết nối bên ngoài (Internet) của miền	Giao thức vạch đường quan tâm đến đến kết nối bên ngoài (Internet) của miền vì đây là cơ sở của mạng diện rộng toàn cầu.
4	Giao thức vạch đường tiêu biểu là OSPF, RIP, IS-IS	Giao thức vạch đường tiêu biểu là BGP

Bảng 6.1 Bảng so sánh các đặc trưng của Định tuyến nội miền và liên miền

6.2 BÀI TẬP THỰC HÀNH

6.2.1 Bài tập 18



Hình 6.1 Mô hình mạng sử dụng trong Bài tập 18

Mục tiêu: Xây dựng mô hình mạng của một AS sử dụng giải thuật vạch đường nội miền RIPv2 kết hợp với việc phân chia các đoạn mạng bằng Switch. Các bước thực hiện *Bài tập 18* được gợi ý sơ lược như sau.

1) Gán địa chỉ IP tương ứng cho các giao diện mạng trên các thiết bị. Ví dụ: `eth0` của `router1` có địa chỉ IP là `192.168.10.1/24`.

– Lưu ý: các giao diện của `switch11` không cần đặt địa chỉ IP

2) Xây dựng `router1`, `router2` và `router3`

– Kích hoạt giải thuật RIPv2 trên `quagga` của các `router` này.

– Kết quả mong muốn khi kiểm thử:

  ✓ Bảng vạch đường trên các Router có đầy đủ thông tin đến các mạng (Net) có kết nối đến các Router đó.

  ✓ Việc gửi và nhận dữ liệu là thông suốt (bằng lệnh `ping`) giữa các giao diện của các Router.

3) Xây dựng `switch11`, `client` và liên kết đến `router1`

– Kết quả mong muốn khi kiểm thử: `client` gửi và nhận dữ liệu thành công với tất cả các giao diện mạng của các `router`.

– Lưu ý:

  ✓ `client` thiết lập vạch đường mặc nhiên đến `router1`.

  ✓ `switch11` tạo cầu nối br0 để chuyển tiếp dữ liệu từ `client` đến `router1`.

4) Xây dựng `switch12`, `pc2` và liên kết đến `router2`

– Kết quả mong muốn khi kiểm thử:

  ✓ `pc21` gửi và nhận dữ liệu thành công với tất cả các giao diện mạng của các router.

  ✓ `pc21` gửi và nhận dữ liệu thành công với `client`.

5) Áp dụng lại các thao tác của 3) và 4) cho cấu hình trên `Net6` và `Net7`

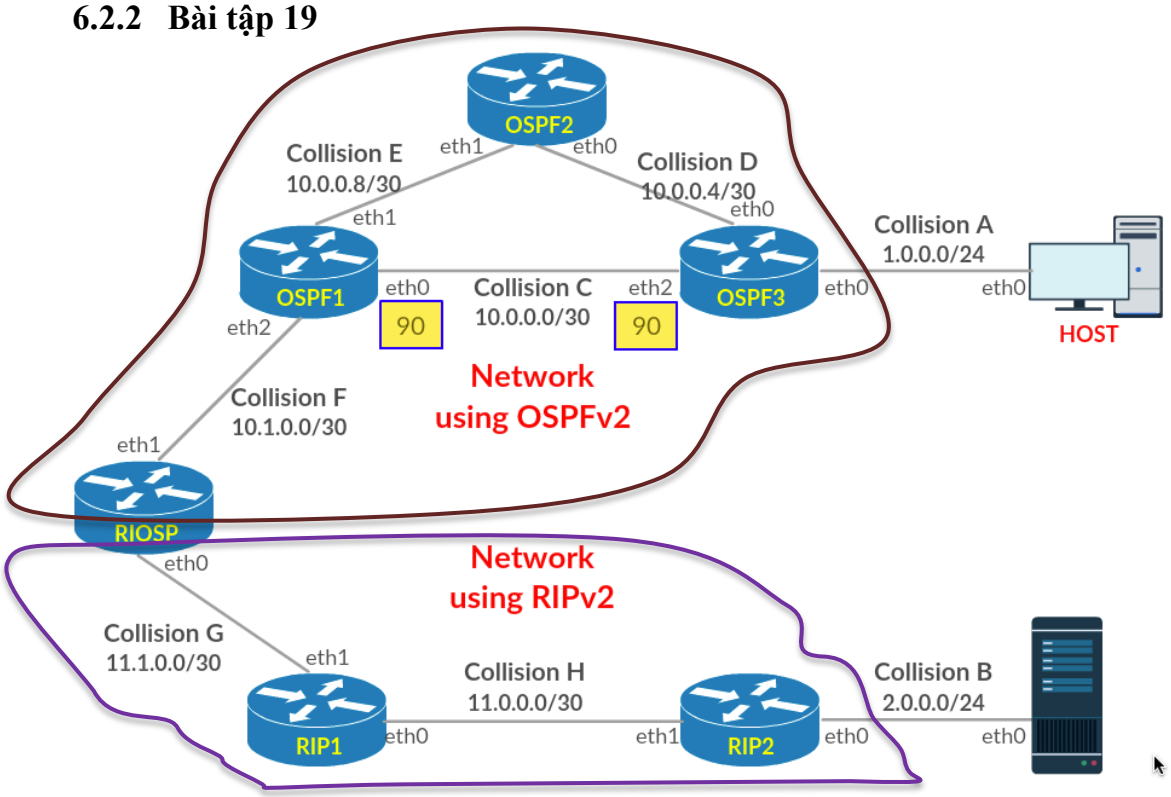
6) Trên `webserver`, khởi động `apache2` với trang chủ có nội dung tùy ý.

7) *Bài tập 18* hoàn thành khi:

– Tất cả việc truyền tải dữ liệu đều thông suốt.

– `client` hiển thị thành công nội dung của trang web từ `webserver`.

6.2.2 Bài tập 19



Hình 6.2 Mô hình mạng sử dụng trong Bài tập 19

Mục tiêu: Xây dựng mô hình mạng của một AS sử dụng giải thuật vạch đường nội miền RIPv2 và OSPFv2.

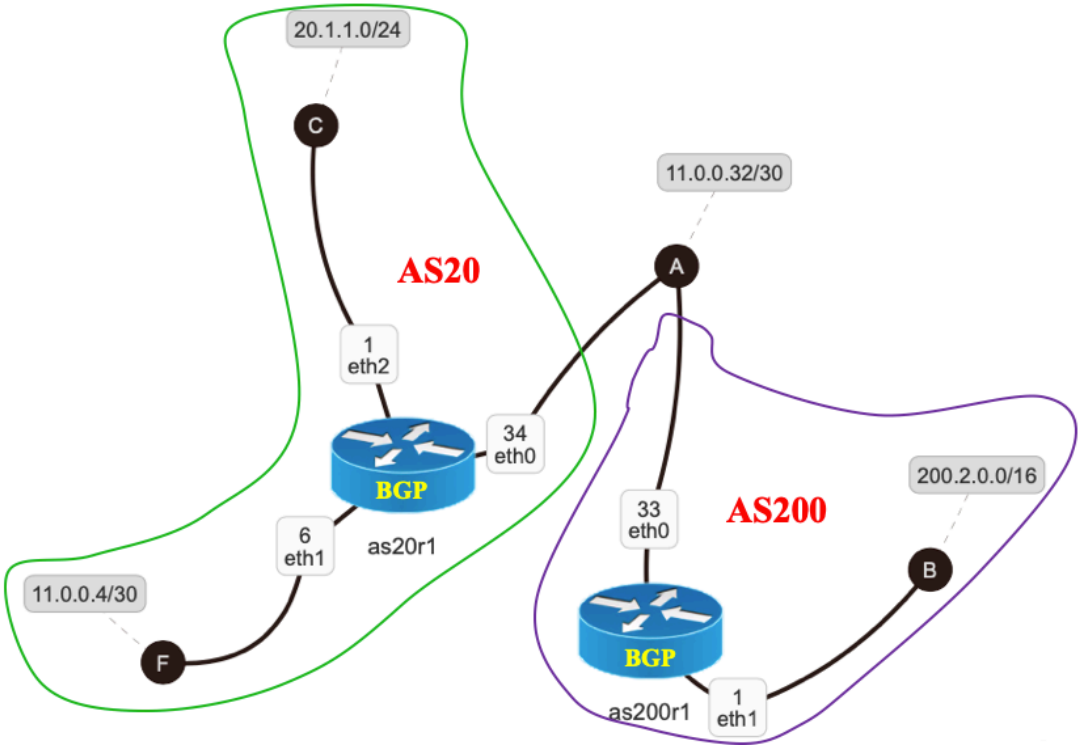
Lưu ý: *Bài tập 19* sẽ xuất hiện một số kiến thức mới, sinh viên theo dõi hướng dẫn chi tiết của giảng viên phụ trách thực hành.

Miêu tả mô hình mạng Bài tập 19:

- AS được chia làm 2 khu vực (area). Khu vực 1 sử dụng RIPv2 và một khu vực 2 sử dụng OSPFv2.
- Router `riosp` chạy cả RIPv2 và OSPFv2.
  - ✓ Trên `br`, thông tin vạch đường của khu vực 1 (dưới dạng RIPv2) được phân phối lại để chuyển đến khu vực 2 (dưới dạng OSPFv2).
  - ✓ Ngược lại, thông tin vạch đường của khu vực 2 (dưới dạng OSPFv2) được phân phối lại để chuyển đến khu vực 1 (dưới dạng RIPv2).
- Với khu vực sử dụng OSPFv2:
  - ✓ ID của khu vực này là `0.0.0.0`
  - ✓ Chi phí mặc nhiên được gán cho một giao diện của Router cài đặt OPSFv2 là `10` trong trường hợp mô hình mạng không miêu tả chi phí.

Kết quả mong muốn: `client` mở được trang web có nội dung bất kỳ tại `server`

6.2.3 Bài tập 20



Hình 6.3 Mô hình mạng sử dụng trong Bài tập 20

Mục tiêu: Khảo sát mô hình mạng minh họa đơn giản cho định tuyến liên miền bằng giao thức BGP.

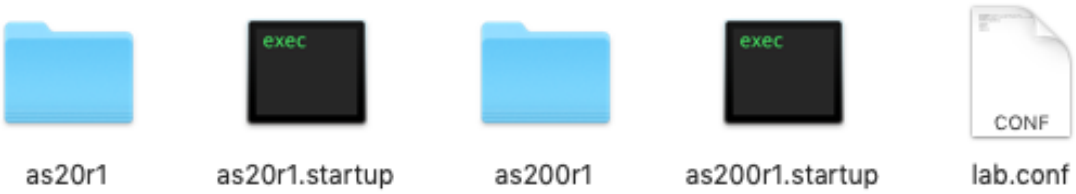
Lưu ý:

- Đây là bài tập để sinh viên tham khảo, không bắt buộc phải hoàn thành.
- Source nguồn của bài tập sẽ được giảng viên hướng dẫn thực hành cung cấp để sinh viên tham khảo.

Mô tả mô hình mạng Bài tập 20:

- `AS20` và `AS200` là định danh của 2 miền cần thực hiện kết nối.
- Giữa 2 miền chỉ hình thành 1 đường nối kết duy nhất, không có đường nối kết dự phòng (`backup`) hoặc chia sẻ tải (`load sharing`) → Mô hình này là một dạng của `Stub Network`.
- `as20r1` và `as2000r1` là các Router đóng vai trò đại diện cho 2 miền này.
- `as20r1` và `as200r1` sẽ được thiết lập và thực thi giao thức `BGP` để tất cả các phân nhánh LAN trong 2 AS có thể truyền dữ liệu đến nhau.

– Cấu trúc thư mục tham khảo của *Bài tập 20* như trong hình 6.3 dưới đây:



Hình 6.4 Các thư mục con và tệp tin trong thư mục BaiTap20

Kết quả mong muốn:

– Bảng vạch đường của 2 Router đại diện có đủ thông tin đến các mạng.

– Các nhánh mạng LAN trong 2 miền đều có thể truyền dữ liệu cho nhau.

– `client` mở được trang web có nội dung mặc định trên `webserver`.

Thiết lập tham khảo dành cho dịch vụ giao thức BGP của dịch vụ quagga:

– Nội dung của file `as20r1/etc/quagga/bgpd.conf`

```
!
hostname bgpd
password zebra
enable password zebra
!
router bgp 20
network 20.1.1.0/24
network 0.0.0.0/0
neighbor 11.0.0.33 remote-as 200
neighbor 11.0.0.33 description Router as200r1
neighbor 11.0.0.33 default-originate
neighbor 11.0.0.33 prefix-list customerIn in
neighbor 11.0.0.33 prefix-list defaultOut out
!
ip prefix-list customerIn permit 200.2.0.0/16
ip prefix-list defaultOut permit 0.0.0.0/0
!
log file /var/log/zebra/bgpd.log
!
```

```
debug bgp
debug bgp events
debug bgp filters
debug bgp fsm
debug bgp keepalives
debug bgp updates
!
```

– Nội dung của file as200r1/etc/quagga/bgpd.conf

```
!
hostname bgpd
password zebra
enable password zebra
!
router bgp 200
network 200.2.0.0/16
neighbor 11.0.0.34 remote-as 20
neighbor 11.0.0.34 description Router as20r1
!
log file /var/log/zebra/bgpd.log
!
debug bgp
debug bgp events
debug bgp filters
debug bgp fsm
debug bgp keepalives
debug bgp updates
!
```