

TRUNG TÂM ĐÀO TẠO NEWSTAR

Biên soạn: Huỳnh Nguyễn Chính

CCNA

Routing & Switching



*Dựa theo nội dung chương trình đào tạo chứng chỉ CCNA
Routing & Switching của Cisco*

TRUNG TÂM ĐÀO TẠO NEWSTAR



CCNA

Routing & Switching



Biên soạn: Huỳnh Nguyễn Chính

7-2016

LỜI NÓI ĐẦU

Tài liệu được biên soạn dựa theo nội dung chương trình đào tạo chứng chỉ CCNA Routing & Switching của Cisco. Các nội dung được tóm tắt một cách ngắn gọn, mạng tính hệ thống, giúp người đọc có thể tiếp cận nhanh chóng với các kiến thức trong chương trình.

Các bài thực hành trong từng chương trình bày từ mô hình, yêu cầu và các hướng dẫn cấu hình từng bước, các lệnh kiểm tra cấu hình, các thông số hệ thống giúp người đọc có thể dễ dàng thực hiện và kiểm tra kết quả nhanh chóng, trong đó có phần giải thích ý nghĩa của một số thông số quan trọng giúp người đọc nắm bắt được các nội dung trọng tâm trong từng bài thực hiện cũng như các kiến thức liên quan.

Thông qua sách này, tác giả mong muốn góp phần giúp các bạn đọc dễ dàng tiếp cận kiến thức và là bước đệm để đọc thêm các tài liệu tiếng Anh liên quan. Tác giả cũng mong muốn nhận được nhiều góp ý của các bạn đọc để có thể điều chỉnh, bổ sung làm cho cuốn sách ngày một giúp ích nhiều hơn cho các bạn đọc.

Trân trọng cảm ơn và chúc các bạn thành công.

MỤC LỤC

Chương 1. TỔNG QUAN VỀ MẠNG MÁY TÍNH	6
1. Mạng máy tính	6
1.1. Khái niệm	6
1.2. Các thành phần cơ bản:	6
1.3. Các ứng dụng trên mạng	6
1.4. Phân loại mạng	7
1.5. Đặc trưng của một mạng:	7
2. Mô hình OSI và TCP/IP	8
2.1. Mô hình tham chiếu OSI	8
2.2. Mô hình TCP/IP	9
3. Quá trình vận chuyển dữ liệu qua mạng	10
3.1. Quá trình đóng gói dữ liệu	10
3.2. Quá trình mở gói dữ liệu	10
3.3. Mối liên quan giữa tầng ứng dụng và tầng vận chuyển	11
4. Tổng kết chương	12
5. Câu hỏi và bài tập	12
6. Lab. Cấu hình cơ bản trên Router/Switch	13
Lab 1-1. Cấu hình cơ bản	15
Lab 1-2. Password recovery	23
Lab 1-3. Backup & restore cisco IOS	26
Lab 1-4. Cisco discovery protocol	28
Chương 2. ĐỊA CHỈ IPv4	30
1. Giới thiệu	30
2. Phân lớp địa chỉ IPv4	30
3. IP public và IP private	32
4. Subnet Mask	32
5. Kỹ thuật chia mạng con	32
6. Kỹ thuật VLSM	32
7. Một số dạng bài tập IP	33
8. Bài tập	34
Chương 3. ĐỊNH TUYẾN	36
1. Giới thiệu	36

2. Phân loại định tuyến	36
2.1. Định tuyến tĩnh – static routing	37
2.2. Định tuyến động	38
3. Cấu hình định tuyến động – distance vector	40
3.1. RIP	40
3.2. OSPF	43
3.3. EIGRP	47
3.4. Redistribution giữa các giao thức định tuyến	50
4. DHCP	53
5. Tính sẵn sàng và dự phòng	54
6. Tổng kết chương	54
7. Câu hỏi và bài tập	55
8. Lab: Định tuyến	57
Lab 3-1. Static routing	57
Lab 3-2. Dynamic routing – RIP	59
Lab 3-3. Dynamic routing – RIPv2	64
Lab 3-4. RIPv2 Plain Text Authentication	68
Lab 3-5. RIPv2 MD5 Authentication	69
Lab 3-6. Dynamic routing – EIGRP	70
Lab 3-7. EIGRP Authentication	73
Lab 3-8. Dynamic routing – OSPF	75
Lab 3-9. OSPF Authentication	78
Lab 3-10. Redistribute giữa RIP & EIGRP	80
Lab 3-11. Redistribute giữa RIP & OSPF	83
Lab 3-12. DHCP Server	86
Lab 3-13. DHCP – Helper Address	88
Lab 3.14. HSRP	90
Chương 4. SWITCH	93
1. Giới thiệu	93
2. VLAN	93
3. Phân loại	95
4. Cấu hình VLAN	95
5. Đường Trunk	97
6. VLAN Trunking Protocol (VTP)	99
7. Định tuyến giữa các VLAN	102
8. Giao thức STP	106
9. EtherChannel	108
10. Tổng kết chương	110
11. Câu hỏi và bài tập	110
12. Lab. SWITCH	113
Lab 4-1. VLAN	114

Lab 4-2. VLAN TRUNKING	117
Lab 4-3. Traditional Spanning Tree Protocol - 802.1D	120
Lab 4-4. ĐỊNH TUYẾN GIỮA CÁC VLAN	121
Lab 4-5. Inter-VLAN routing (MultiLayer Switch)	124
Lab 4-6. Port security	126
Lab 4-7. EtherChannel	127
Lab 4-8. Port-based authentication - 802.1x	128
CHƯƠNG 5. ACL	129
1. Giới thiệu	129
2. Phân loại và hoạt động của ACL	129
3. Cấu hình ACL	130
4. Standard ACL	131
5. Extended ACL	134
6. Named ACL	135
7. Tổng kết chương	137
8. Câu hỏi và bài tập	137
9. Lab. ACL	140
Lab 5-1. STANDARD ACL	141
Lab 5-2. EXTENDED ACL	142
Lab 5-3. ACL (tt)	144
CHƯƠNG 6. NAT	145
1. Giới thiệu	145
2. Static NAT	146
3. Dynamic NAT	147
4. NAT overload	148
5. Tổng kết chương	150
6. Câu hỏi và bài tập	150
7. Lab NAT	154
Lab 6-1. STATIC NAT	155
Lab 6-2. DYNAMIC NAT	157
Lab 6-3. DYNAMIC NAT WITH OVERLOAD	159
CHƯƠNG 7. IPv6	161
1. Giới thiệu	161
2. Những hạn chế của IPv4	161
3. Khái quát IPv6	162
4. Cấu trúc địa chỉ IPv6	165
4. Các giải pháp triển khai IPv6 trên nền IPv4	171
4.1. Dual Stack (Dual IP Layer)	171
4.2 Tunneling	172
4.3 NAT-PT	172
5. Lab. Ipv6	172

Lab 7-1. IPv6 CĂN BẢN	173
Chương 8. IPv6 ROUTING	174
1. Tổng quan	174
2. Định tuyến tĩnh	174
3. RIPng	176
4. OSPF cho Ipv6	177
5. EIGRP for Ipv6	180
6. Lab. Routing for IPv6	182
Lab. 8-1. STATIC ROUTING CHO IPv6	182
Lab. 8-2. CẤU HÌNH RIPng	183
CHƯƠNG 9. WAN	186
1. Giới thiệu	186
2. Kết nối serial point-to-point	186
3. VPN	191
3. Tổng kết chương	195
4. Câu hỏi và bài tập	195
5. Lab. WAN	199
Lab. 9-1. PPP PAP	199
Lab. 9-2. PPP CHAP – Dạng 1	200
Lab. 9-3. PPP CHAP – Dạng 2	201
Lab. 9-4. Cấu hình “remote access VPN” (IPSec)	202
Lab. 9-5.	203
Lab. 9-6.	205
Lab. 9-7. VPN kết hợp với NAT	206
Chương 10. GIÁM SÁT MẠNG	207
1. Giới thiệu	207
2. Các thành phần của hệ thống giám sát	207
3. Các công cụ nguồn mở hỗ trợ trong việc giám sát mạng	209
3.1. Cacti	209
3.2. Nagios	209
Final Lab 1.	211
Final Lab 2.	213

Chương 1. TỔNG QUAN VỀ MẠNG MÁY TÍNH

Chương này trình bày một số khái niệm về mạng máy tính, phân loại mạng máy tính, đặc điểm của mô hình tham chiếu OSI và mô hình TCP/IP, quá trình trao đổi dữ liệu giữa các máy tính qua mạng. Học xong chương này, người học có khả năng:

- Phân biệt được đặc điểm của mô hình tham chiếu OSI và TCP/IP
- Phân biệt được các loại mạng máy tính
- Trình bày được các bước cơ bản của quá trình trao đổi dữ liệu của các máy qua mạng
- Phân tích được các thành phần cơ bản của các gói tin gửi qua mạng bằng phần mềm bắt gói

1. Mạng máy tính

1.1. Khái niệm

Mạng máy tính là một hệ thống gồm các máy tính kết nối với nhau để trao đổi dữ liệu với nhau thông qua môi trường kết nối.

Trong thời đại ngày nay, có nhiều thiết bị kết nối vào môi trường mạng máy tính như máy in, camera, điện thoại,... gọi chung là thiết bị đầu cuối. Môi trường kết nối gồm môi trường có dây và không dây; các thiết bị mạng thường dùng để kết nối các thiết bị đầu cuối như: Switch, router, firewall,... Các giao thức được sử dụng để các thiết bị đầu cuối có thể giao tiếp được với nhau.

1.2. Các thành phần cơ bản:

Các thành phần cơ bản của mạng máy tính bao gồm

- Máy tính: đóng vai trò là thiết bị đầu cuối, làm việc trực tiếp với người dùng

- Thiết bị mạng: Switch là thiết bị tập trung, kết nối các máy tính trong mạng có dây, Access Point là thiết bị tập trung kết nối các máy tính trong mạng không dây, Router là thiết bị định tuyến dùng để kết nối các mạng với nhau.
- Các thiết bị kết nối: gồm card mạng, đầu nối
- Môi trường kết nối: môi trường có dây và không dây

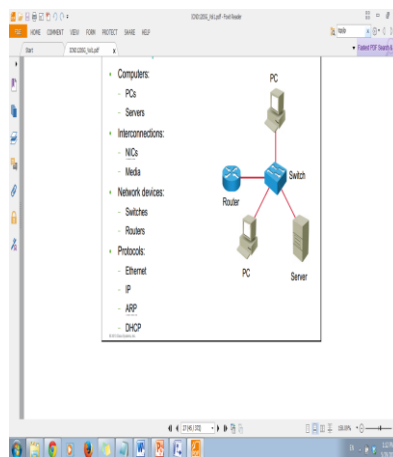
1.3. Các ứng dụng trên mạng

Các ứng dụng trên mạng phổ biến gồm các ứng dụng sau:

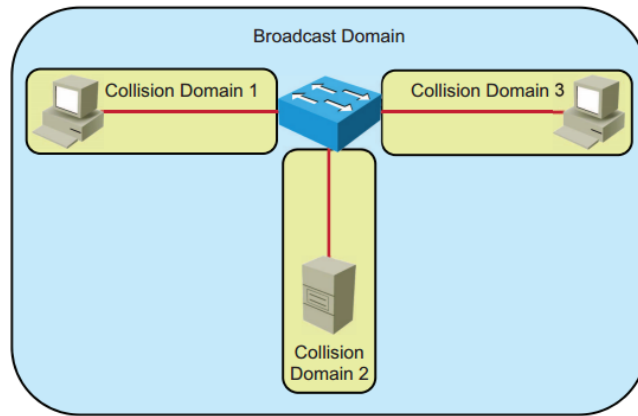
- Email
- Web
- Trao đổi trực tuyến
- Cộng tác: whiteboard, Netmeeting, WebEx
- Cơ sở dữ liệu

1.4. Phân loại mạng

- LAN: Mạng LAN (Local Area Network) là mạng cục bộ, được tổ chức cho một đơn vị trong một không gian địa lý nhỏ. Các thiết bị trong LAN có kết nối trực tiếp với nhau, tốc độ cao. Công nghệ mạng được sử dụng trong LAN phổ biến là Ethernet.



- Các thành phần trong mạng LAN: PC, server, Switch, router
- Vai trò của Switch:

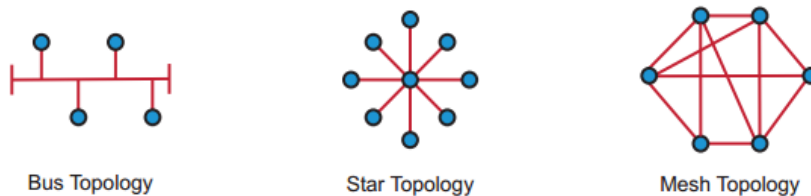


- **WAN:** Mạng WAN (Wide Area Network) là mạng diện rộng, là mạng của một tổ chức có nhiều chi nhánh kết nối với nhau thông qua môi trường Internet. Các công nghệ được sử dụng trong WAN phổ biến là: MPLS, VPN,...
- **MAN:** mạng MAN (Metropolitan Area Network) là mạng đô thị, các thành phố lớn thường tổ chức hệ thống mạng đường trục tốc độ cao để phục vụ cho các đơn vị trong thành phố đó.
- **SAN:** Mạng SAN (Storage Area Network) là mạng lưu trữ, nhằm thực hiện chức năng lưu trữ cho lượng dữ liệu lớn trong đơn vị.
- **INTERNET:** Mạng Internet là mạng của các mạng, là hệ thống mạng toàn cầu.

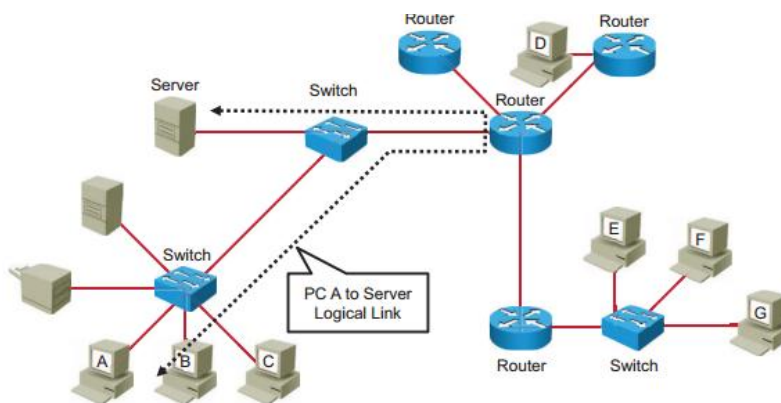
1.5. Đặc trưng của một mạng:

- Topology: trong mạng có 2 loại topology được nhắc tới là “physical topology” và “logical topology”.

Physical topology:



Logical topology: thể hiện các đường đi luận lý mà tín hiệu sử dụng để chuyển từ một điểm trong mạng đến một điểm khác.



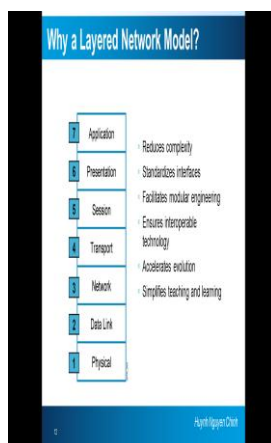
- **Tốc độ:** là thước đo của tốc độ truyền dữ liệu trên đường truyền.
- **Chi phí:** mức độ đầu tư cho các thành phần mạng, cài đặt và bảo trì của một hệ thống mạng.
- **Bảo mật:** sự bảo mật chỉ ra cách thức bảo vệ một mạng
- Sự sẵn sàng
- Khả năng mở rộng
- Sự tin cậy

2. Mô hình OSI và TCP/IP

Mô hình tham chiếu OSI và TCP/IP là hai mô hình mạng cơ bản trong mạng máy tính.

2.1. Mô hình tham chiếu OSI

Mô hình tham chiếu OSI gồm 7 tầng (layer).



Tầng 1 - Physical: Tầng vật lý liên quan các vấn đề về điện tử, cơ khí; xử lý dữ liệu dạng bit; thiết bị mạng hoạt động ở tầng này là Hub.

Tầng 2 – Data link: Tầng liên kết dữ liệu liên quan đến việc định dạng dữ liệu theo các chuẩn, điều khiển cách thức truy xuất đến môi trường vật lý; xử lý dữ liệu dạng khung (frame); liên quan đến địa chỉ vật lý (phổ biến là địa chỉ MAC); thiết bị mạng hoạt động ở tầng này là Switch.

Tầng 3 - Network: Tầng mạng thực hiện chức năng định tuyến cho các gói tin; xử lý dữ liệu dạng gói (packet); liên quan đến địa chỉ luận lý (phổ biến là địa chỉ IP,...); thiết bị hoạt động ở tầng này là Router.

Tầng 4 - Transport: Tầng vận chuyển thực hiện chức năng đảm bảo việc vận chuyển dữ liệu từ nguồn đến đích thông qua hệ thống mạng. Thực hiện việc chia nhỏ dữ liệu phù hợp với kích thước tối đa của kênh truyền ở bên gửi và tái lập ở bên nhận.

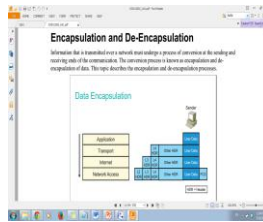
Tầng 5 - Session: Tầng phiên thực hiện việc thiết lập, quản lý và kết thúc các phiên làm việc của các chương trình ứng dụng.

Tầng 6 - Presentation: Tầng trình bày thực hiện việc đảm bảo dữ liệu đọc được ở tầng ứng dụng. Các chức năng của tầng này liên quan đến định dạng dữ liệu, cấu trúc dữ liệu, nén dữ liệu, mã hóa dữ liệu.

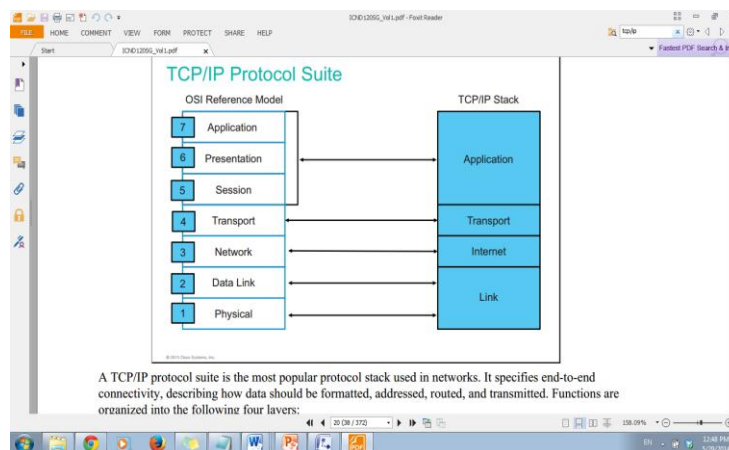
Tầng 7 - Application: Tầng ứng dụng là tầng cao nhất trong mô hình OSI, liên quan đến các chương trình ứng dụng làm việc trực tiếp với người dùng (như Email, FTP, Web,...) hoặc các dịch vụ hỗ trợ khác

2.2. Mô hình TCP/IP

Mô hình TCP/IP gồm có 4 tầng. Là mô hình được sử dụng phổ biến. Trong đó, hai giao thức quan trọng nhất được nhắc tới là TCP và IP.



Mối liên quan giữa 2 mô hình:



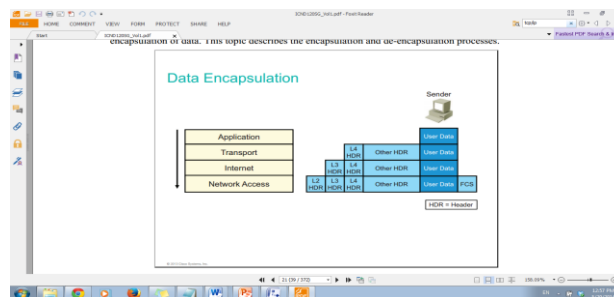
- **Tầng 1 - Network access (link):** đặc điểm của tầng này bao gồm đặc điểm của 2 tầng thấp nhất của mô hình OSI là tầng vật lý và tầng liên kết dữ liệu. Tầng này mô tả về các đặc điểm vật lý của các kết nối, điều khiển truy cập và định dạng dữ liệu để truyền tải.
- **Tầng 2 - Internet:** cung cấp tính năng định tuyến cho dữ liệu từ nguồn đến đích trong các gói tin và thông tin về địa chỉ, di chuyển dữ liệu giữa tầng Link và tầng transport
- **Tầng 3 - Transport:** là tầng quan trọng của kiến trúc TCP/IP. Tầng này cung cấp các dịch vụ truyền thông trực tiếp đến quá trình xử lý của ứng dụng đang chạy trên mạng.
- **Tầng 4 – Application:** cung cấp các ứng dụng cho việc truyền tập tin, xử lý sự cố và các hoạt động Internet

3. Quá trình vận chuyển dữ liệu qua mạng

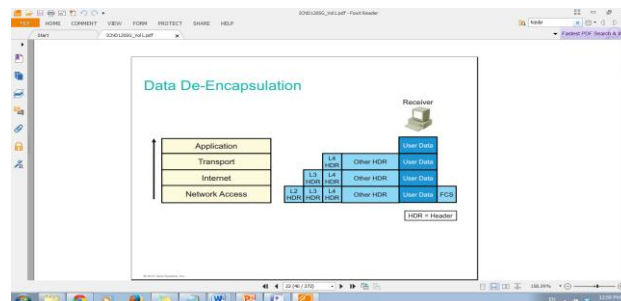
3.1. Quá trình đóng gói dữ liệu

Quá trình đóng gói dữ liệu diễn ra bên máy gửi. Dữ liệu xuất phát từ tầng ứng dụng được đóng gói và chuyển xuống các tầng kế tiếp, đến mỗi tầng dữ liệu được gắn thêm thông tin mô tả của tầng tương ứng gọi là header.

Khi dữ liệu đến tầng “transport”, tại đây diễn ra quá trình chia nhỏ gói tin nếu kích thước dữ liệu lớn hơn so với kích thước truyền tối đa cho phép. Dữ liệu đến tầng “network”, mỗi gói tin sẽ gắn thêm thông tin tương ứng ở tầng này gọi là “IP header”, trong đó có chứa thông tin quan trọng là địa chỉ IP nguồn và IP đích được sử dụng trong quá trình định tuyến. Dữ liệu đến tầng “Data-Link” sẽ gắn thêm thông tin mô tả tầng này gọi là “Frame header”, trong đó có chứa thông tin về địa chỉ MAC nguồn và MAC đích. Trường hợp địa chỉ MAC đích không biết, máy tính sẽ dùng giao thức ARP để tìm. Sau đó dữ liệu chuyển xuống tầng “Physical”, chuyển thành các tín hiệu nhị phân để truyền đi.



3.2. Quá trình mở gói dữ liệu



Quá trình mở gói dữ liệu diễn ra bên máy nhận. Nguyên tắc chung là các “header” sẽ được mở ở các tầng tương ứng. Khi máy đích nhận được một dãy các bit, dữ liệu được xử lý bởi quá trình mở gói như sau:

- (1) Tầng link kiểm tra trailer (FCS) để xem dữ liệu có bị lỗi hay không. Frame có thể bị loại bỏ hoặc yêu cầu để được truyền lại
- (2) Nếu dữ liệu không bị lỗi, tầng link đọc và thông dịch thông tin điều khiển trong tầng 2.
- (3) Tầng link gỡ bỏ “header” và “trailer”, sau đó gửi phần dữ liệu còn lại lên tầng Internet

3.3. Mối liên quan giữa tầng ứng dụng và tầng vận chuyển

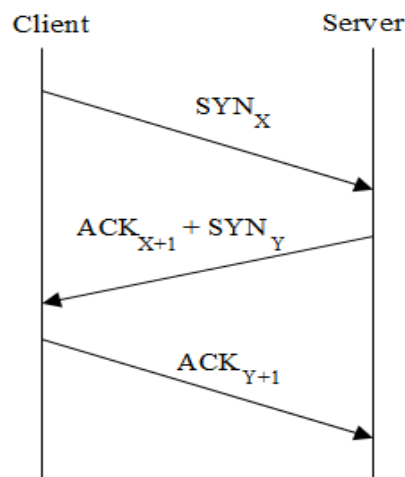
- Mối liên quan giữa tầng ứng dụng và tầng vận chuyển thể hiện thông qua các công ở các ứng dụng và giao thức truyền ở tầng vận chuyển.

Tầng Transport:

- Chức năng: đảm bảo việc vận chuyển dữ liệu từ nguồn đến đích thông qua hệ thống mạng.
- Đơn vị dữ liệu: segment
- 2 cơ chế truyền: tin cậy (*reliable*) và tốt nhất có thể (*best effort*)

Cơ chế truyền	Tin cậy (<i>Reliable</i>)	Tốt nhất có thể (<i>Best Effort</i>)
Kiểu kết nối	Hướng kết nối (<i>connection-oriented</i>)	Phi kết nối (<i>connectionless</i>)
Giao thức	TCP	UDP
Gửi có báo nhận	Có	Không
Một số ứng dụng	<ul style="list-style-type: none"> Email File sharing Downloading 	<ul style="list-style-type: none"> Void streaming Video streaming

- Hướng kết nối** (connection-oriented): là kiểu truyền theo cách thiết lập kênh truyền trước khi gửi dữ liệu đi. Thiết lập bằng cơ chế 3 bước bắt tay (three-way handshake).



4. Tổng kết chương

Trong chương này trình bày một số vấn đề cơ bản và mạng máy tính, phân loại các mạng máy tính phổ biến.

Hai mô hình mạng quan trọng được trình bày là OSI và TCP/IP. Hai mô hình này có đặc điểm chung là phân chia thành các tầng, mỗi tầng đảm nhiệm các chức năng khác nhau.

Quá trình đóng gói dữ liệu diễn ra bên máy gửi và quá trình mở gói diễn ra bên máy nhận. Trong quá trình đóng gói, dữ liệu từ tầng ứng dụng được chuyển xuống các tầng thấp hơn và thông tin ở mỗi tầng đó được thêm vào. Quá trình mở gói ngược lại với quá trình đóng gói.

Đơn vị dữ liệu ở tầng ứng dụng gọi là “data”, ở tầng vận chuyển gọi là “segment”, ở tầng mạng gọi là “packet” và ở tầng liên kết gọi là “frame”.

5. Câu hỏi và bài tập

5.1. Phân biệt Hub, Switch và Router?

5.2. Trình bày quá trình trao đổi dữ liệu qua mạng trong trường hợp hai máy kết nối bằng cáp trực tiếp hay qua Hub?

5.3. Trình bày quá trình trao đổi dữ liệu qua mạng trong trường hợp hai máy kết nối qua Switch?

5.4. Trình bày quá trình trao đổi dữ liệu qua mạng trong trường hợp hai máy kết nối bằng cáp trực tiếp hay qua Router?

5.5. Tầng nào trong mô hình OSI cung cấp chức năng phân tách và tái hợp gói tin trong quá trình truyền dữ liệu?

5.6. Tầng nào trong mô hình OSI cung cấp chức năng lựa chọn đường đi cho các gói tin?

5.7. Phân biệt địa chỉ MAC và địa chỉ IP?

5.8. Phân biệt mạng LAN và WAN?

5.9. Một trường Đại học có nhiều tòa nhà, mỗi tòa nhà có nhiều phòng ban. Hệ thống mạng được thiết kế cho trường này được coi là mạng LAN hay WAN?

5.10. Phân biệt Hub và Access Point?

5.11. Dịch vụ HTTP ở tầng ứng dụng (Application) sử dụng cơ chế truyền nào ở tầng vận chuyển (Transport)

- A. Reliable
- B. Best-effort
- C. Half-duplex
- D. Full-duplex

5.12. Địa chỉ MAC có bao nhiêu bit?

- A. 32 bit
- B. 48 bit
- C. 56 bit
- D. 64 bit

5.13. Các thiết bị sau đây hoạt động chính ở tầng (layer) nào trong mô hình OSI ?

- A. Router : Hoạt động chính ở layer ?
- B. Hub: Hoạt động chính ở layer ?
- C. Switch: Hoạt động chính ở layer ?

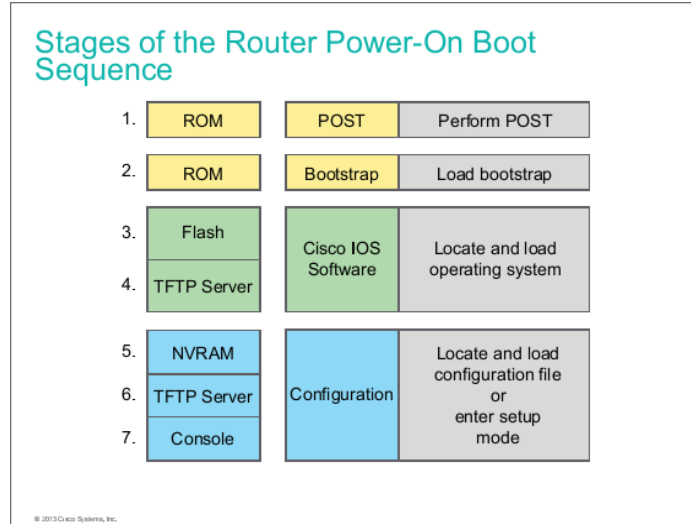
- 5.14. Email và FTP hoạt động ở layer nào trong mô hình OSI?
- 5.15. Các thiết bị mạng nào sau đây hoạt động ở layer Data Link
- A. Hub
 - B. Switch
 - C. Router
- 5.16. Dùng lệnh nào sau đây để biết được địa chỉ MAC trên máy tính?
- A. ipconfig
 - B. Ping
 - C. ipconfig /all
 - D. tracert
- 5.17. Layer nào trong mô hình OSI đảm nhận vai trò nén (encryption) và giải nén (decryption) dữ liệu
- A. network
 - B. presentation
 - C. session
 - D. physical
- 5.18. Layer nào trong mô hình OSI đảm nhận vai trò thiết lập các kết nối tin cậy
- A. network
 - B. session
 - C. transport
 - D. data link
- 5.19. Những thiết bị nào sau đây hoạt động ở layer Network trong mô hình OSI
- A. Router
 - B. Repeater
 - C. Hub
 - D. Switch
- 5.20. HTTPS hoạt động ở cổng nào?
- A. 80
 - B. 81
 - C. 443
 - D. 8080

6. Lab. Cấu hình cơ bản trên Router/Switch

Các thành phần của Router

- RAM/DRAM (Random Access Memory)
- ROM (Read Only Memory)

- FLASH: lưu trữ hệ điều hành (IOS) của router
- NVRAM: lưu tập tin cấu hình (configuration file) của router
- Interfaces: các cổng của router: console, serial, fastEthernet, aux, ...
- Trình tự khởi động của Router:



Lab 1-1. CẤU HÌNH CƠ BẢN

Sơ đồ kết nối:

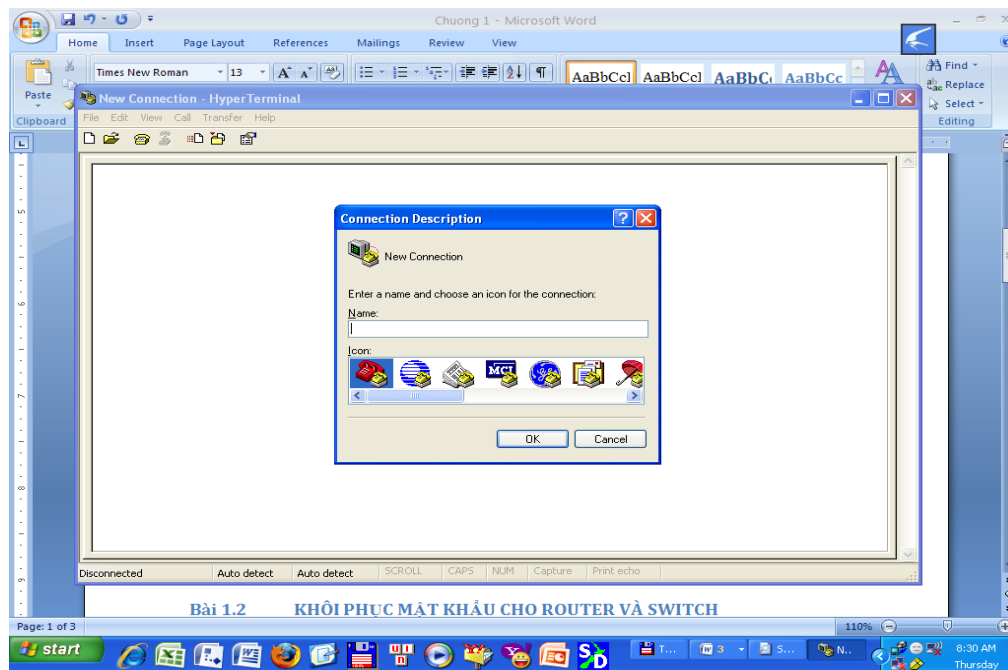
Kết nối cổng COM của PC với cổng console của Router.

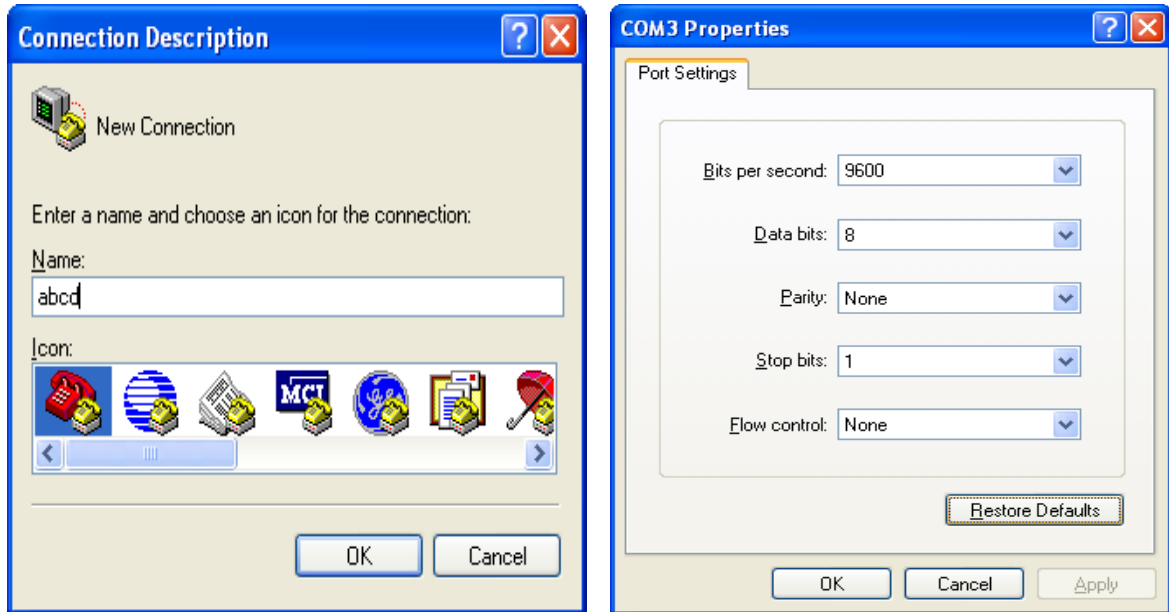
❖ Yêu cầu

- Đăng nhập bằng chế độ dòng lệnh (CLI)
- Đặt mật khẩu cho các mode
- Đặt địa chỉ cho các interface
- Mở đường telnet cho phép truy cập router từ xa

❖ Các bước thực hiện

- Gắn cáp console giữa PC và Router
- Trên PC dùng chương trình Hyper Terminal hoặc Putty



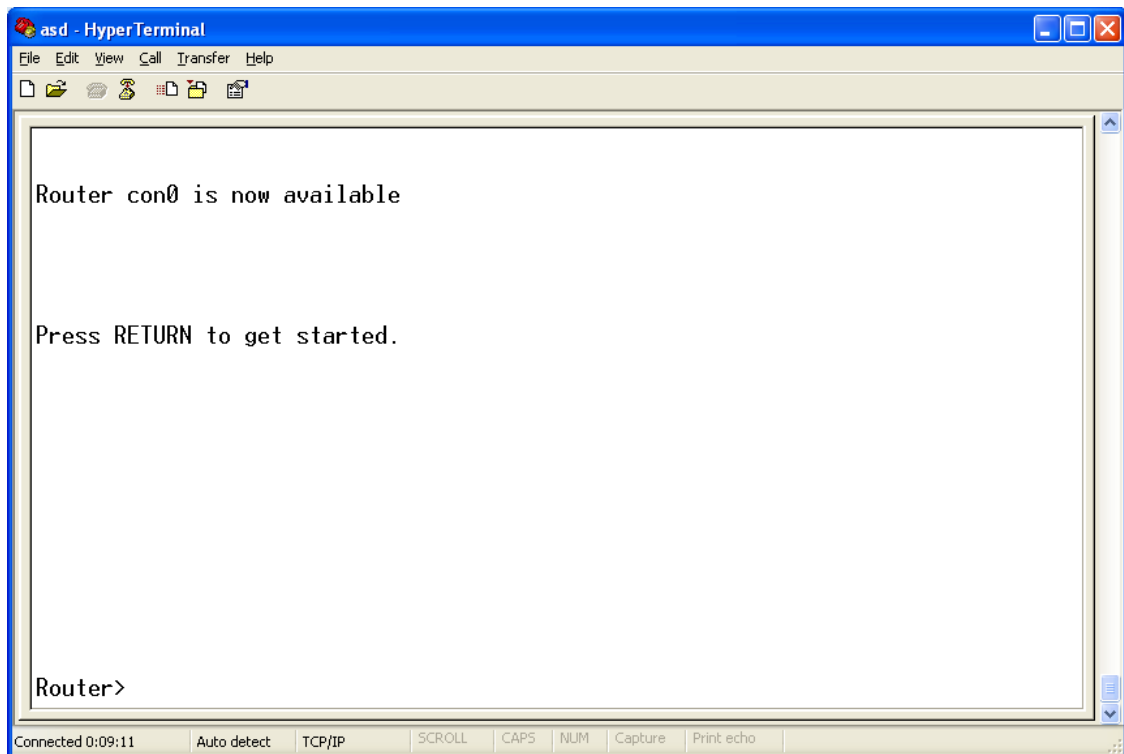


- Khởi động Router (bật nguồn): xem các thông tin hiển thị trên màn hình Hyper Terminal.

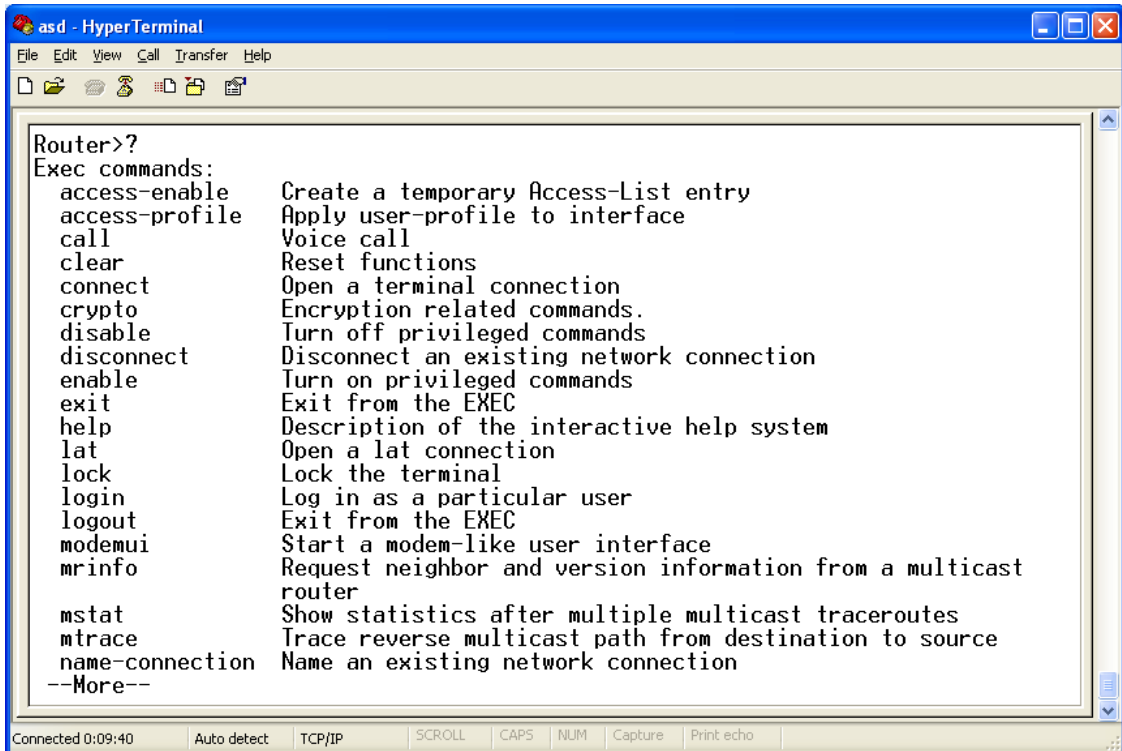
❖ Các mode của router

Dấu nhắc lệnh đầu tiên xuất hiện có dạng

Router> → đang ở *user mode*



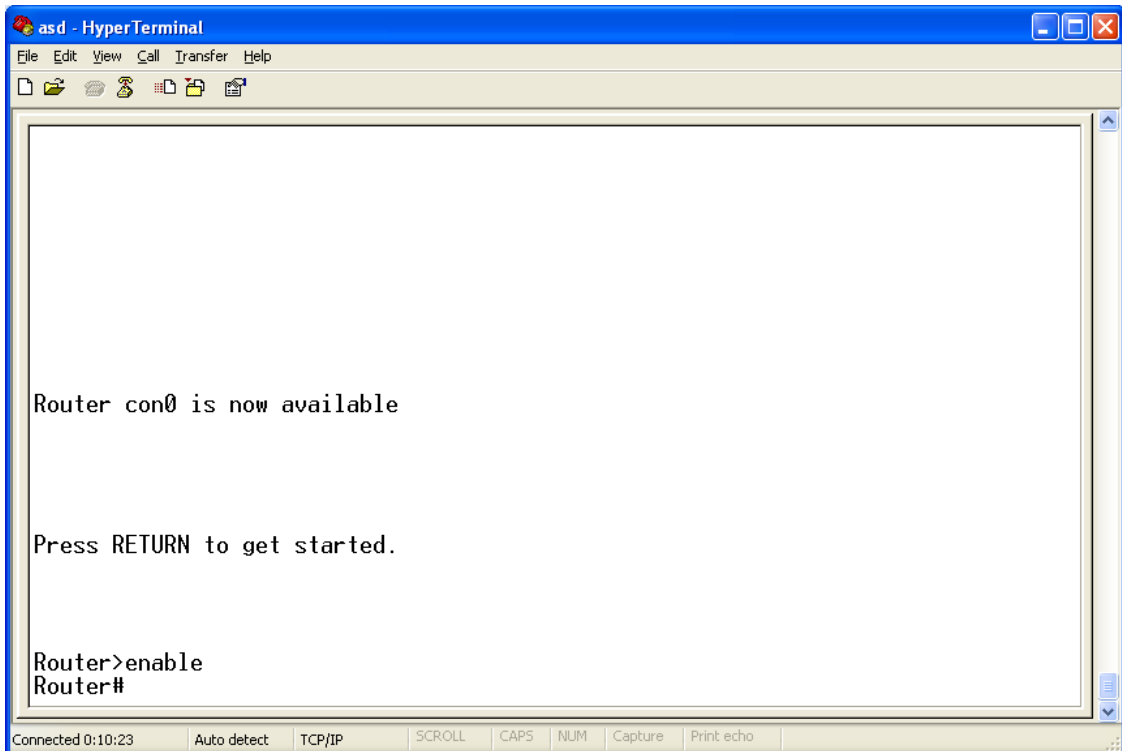
- Để quan sát các lệnh được phép sử dụng ở user mode, ta gõ dấu chấm hỏi (?) và enter Router>?



```

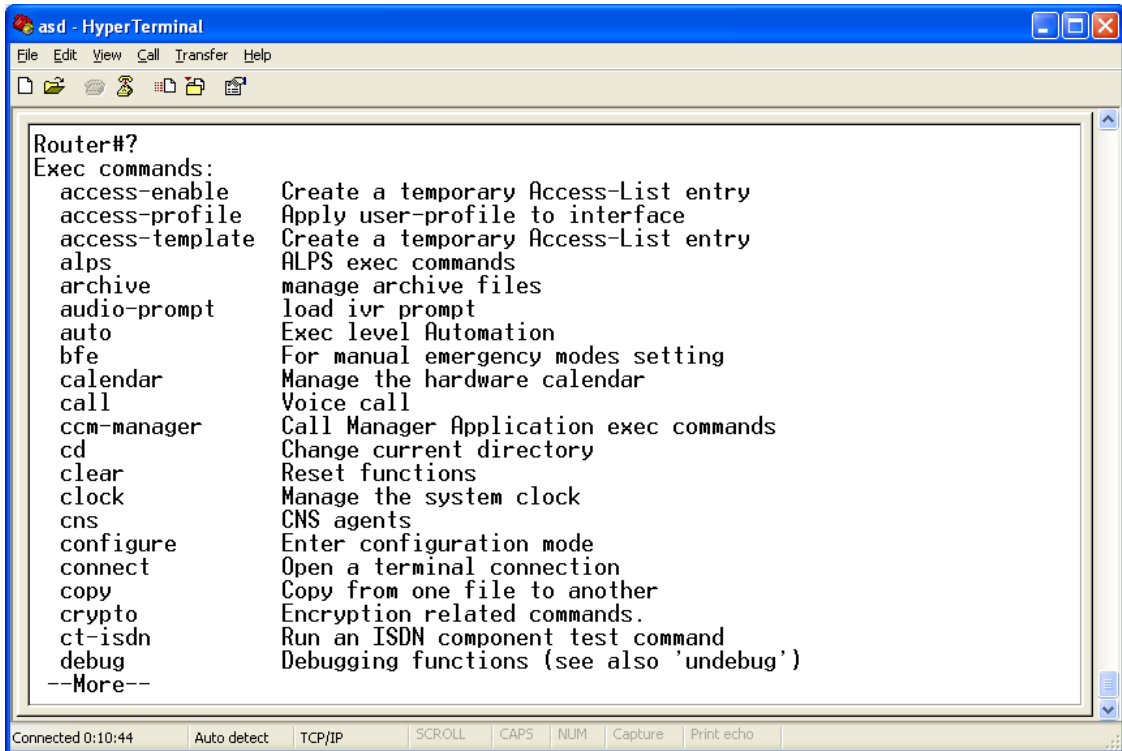
asd - HyperTerminal
File Edit View Call Transfer Help
Router>?
Exec commands:
  access-enable      Create a temporary Access-List entry
  access-profile    Apply user-profile to interface
  call              Voice call
  clear            Reset functions
  connect         Open a terminal connection
  crypto         Encryption related commands.
  disable        Turn off privileged commands
  disconnect     Disconnect an existing network connection
  enable         Turn on privileged commands
  exit          Exit from the EXEC
  help         Description of the interactive help system
  lat         Open a lat connection
  lock       Lock the terminal
  login      Log in as a particular user
  logout    Exit from the EXEC
  modemui   Start a modem-like user interface
  mrinfo    Request neighbor and version information from a multicast
            router
  mstat     Show statistics after multiple multicast traceroutes
  mtrace    Trace reverse multicast path from destination to source
  name-connection Name an existing network connection
  --More--
  
```

- Để vào **priviledge mode**, ta dùng lệnh enable
Router>enable
Router# → đang ở *priviledge mode*
- Để quan sát các lệnh được phép sử dụng ở *priviledge mode*, ta gõ dấu chấm hỏi (?) và enter
Router#?



```

asd - HyperTerminal
File Edit View Call Transfer Help
Router con0 is now available
Press RETURN to get started.
Router>enable
Router#
  
```



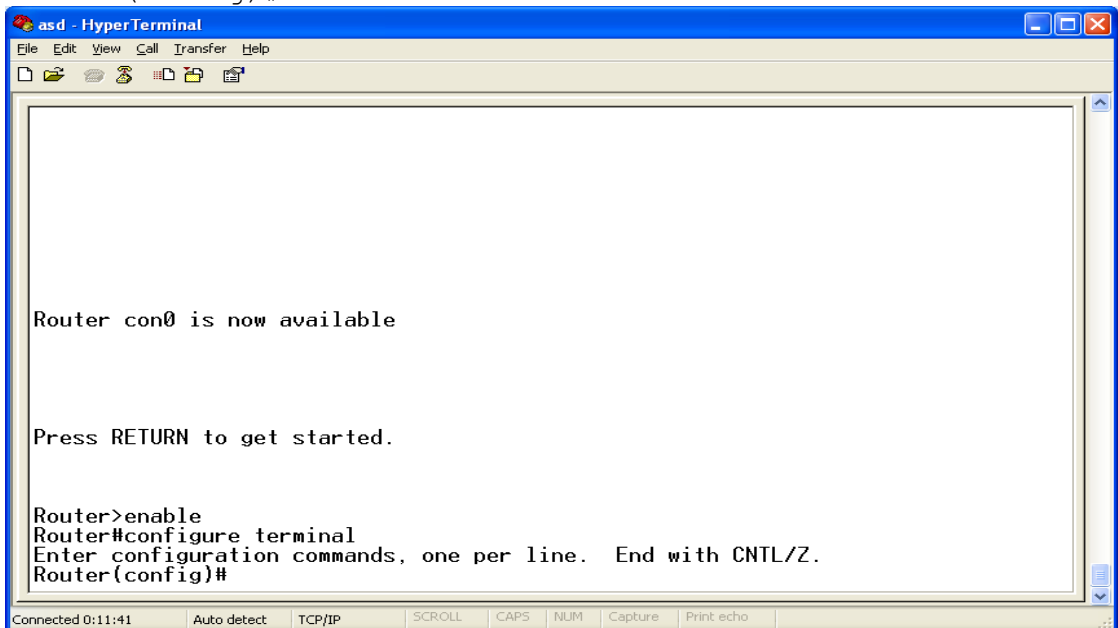
```

Router#?
Exec commands:
  access-enable      Create a temporary Access-List entry
  access-profile     Apply user-profile to interface
  access-template    Create a temporary Access-List entry
  alps               ALPS exec commands
  archive            manage archive files
  audio-prompt       load ivr prompt
  auto               Exec level Automation
  bfe                For manual emergency modes setting
  calendar           Manage the hardware calendar
  call               Voice call
  ccm-manager        Call Manager Application exec commands
  cd                 Change current directory
  clear              Reset functions
  clock              Manage the system clock
  cns                CNS agents
  configure          Enter configuration mode
  connect            Open a terminal connection
  copy               Copy from one file to another
  crypto             Encryption related commands.
  ct-isdn            Run an ISDN component test command
  debug              Debugging functions (see also 'undebug')
--More--
  
```

- Để vào **configuration mode**, ta dùng lệnh *configure terminal* (có thể gõ tắt là *config t*)

```

Router#configure terminal
Router(config)#
  
```



```

Router con0 is now available

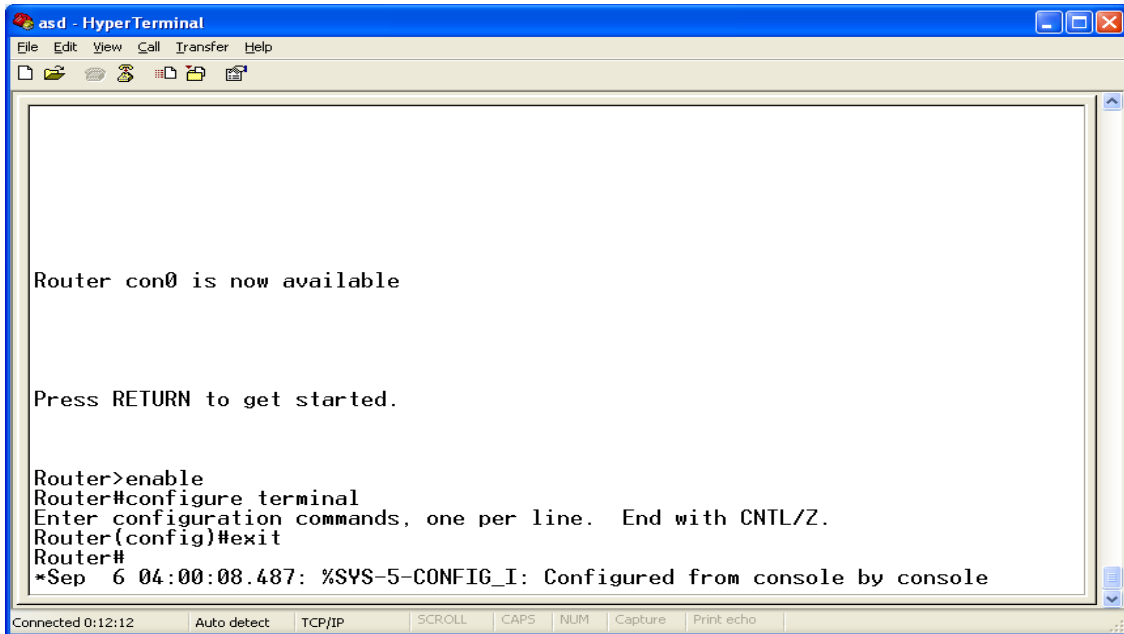
Press RETURN to get started.

Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#
  
```

- Để trở lại mode trước đó, ta dùng lệnh *exit*

```

Router(config)#exit
Router#exit
Router>
  
```



```
as d - HyperTerminal
File Edit View Call Transfer Help
Router con0 is now available

Press RETURN to get started.

Router>enable
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#exit
Router#
*Sep 6 04:00:08.487: %SYS-5-CONFIG_I: Configured from console by console

Connected 0:12:12 Auto detect TCP/IP SCROLL CAPS NUM Capture Print echo
```

❖ **Đặt password cho router**

Vì lý do bảo mật, ta sẽ đặt mật khẩu cho các mode cấu hình. Điều này có nghĩa là mỗi khi đăng nhập vào một mode thì IOS sẽ yêu cầu chúng ta nhập vào password. nếu password đúng thì mới có thể vào mode này.

Trong phần này, chúng ta sẽ thực thi việc đặt password cho

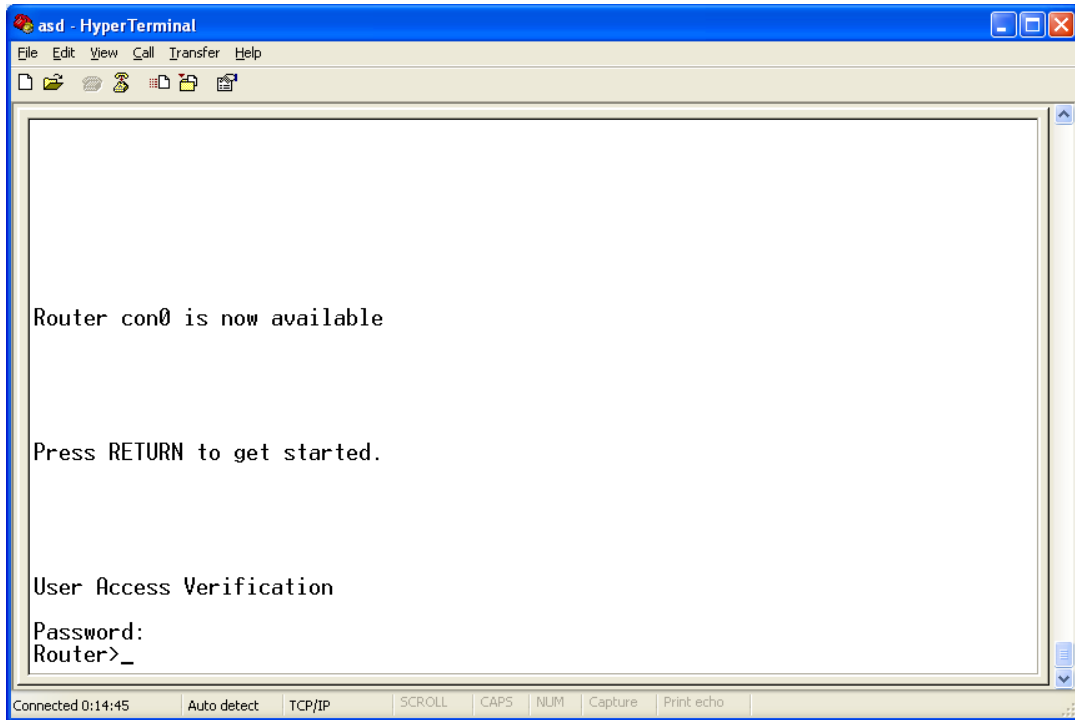
- Console
- Enable mode

● **Đặt password cho cổng console (console password)**

Ý nghĩa: trước khi vào user mode, IOS sẽ yêu cầu nhập password để kiểm tra.

Thực hiện:

```
Router>enable
Router#configure terminal
Router(config)#line console 0
Router(config-line)#password cisco → password là cisco
Router(config-line)#login
```

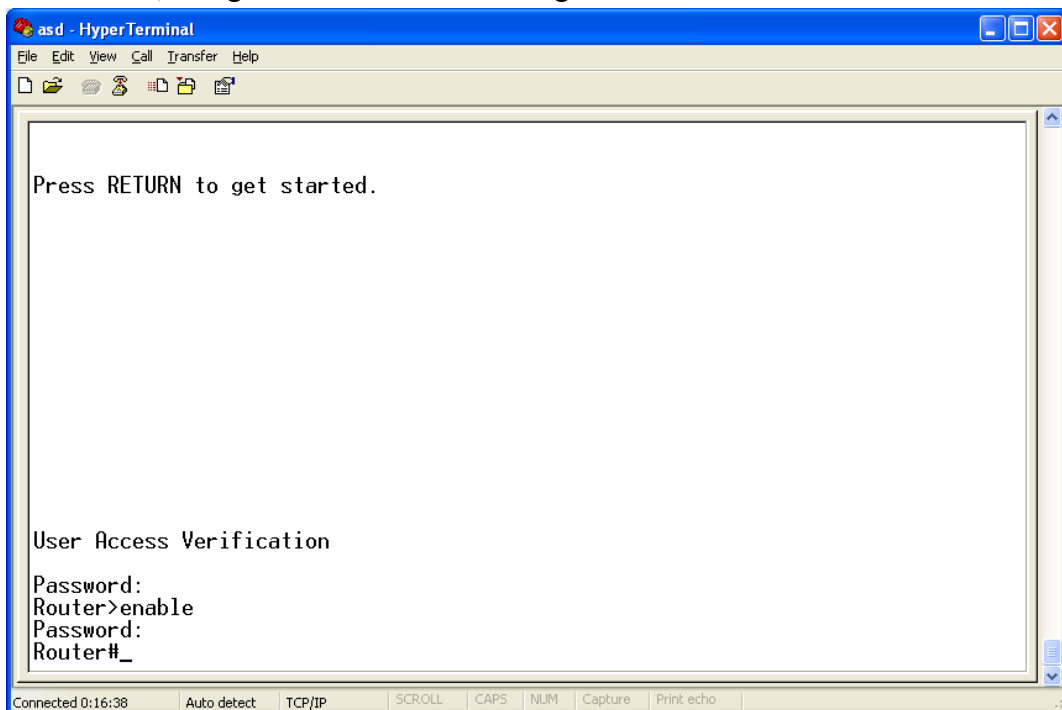
- **Đặt password cho enable mode**

Router(config)#enable password *password*

hoặc

Router(config)#enable secret *password*

Nếu sử dụng cả hai câu lệnh này, thì password của câu lệnh thứ hai (enable secret) mạnh hơn, có nghĩa là nó sẽ được sử dụng.



- **Đặt password cho truy cập router từ xa (virtual terminal password)**

Router(config)#line vty 0 4

Router(config-line)#password *cisco*

Lab 1-2. PASSWORD RECOVERY

❖ Khôi phục mật khẩu cho Router

Quá trình khởi động của router đã được định trước. Sau quá trình POST và nạp hệ điều hành IOS, router sẽ nạp cấu hình hoạt động trong NVRAM. Các cấu hình này không chỉ chứa thông tin giao thức định tuyến, địa chỉ mà còn chứa mật khẩu của router.

Mật khẩu được phục hồi bằng cách bỏ qua nội dung file cấu hình (configuration file) trong NVRAM trong quá trình khởi động. Việc bỏ qua cấu hình được thực hiện bằng cách sửa đổi nội dung thanh ghi cấu hình (configuration register) của router. Lúc này router sẽ không cấu hình chứa mật khẩu cần phục hồi. Khi đã vào được router, người dùng có thể xem mật khẩu trong NVRAM và có thể sử dụng, xoá hay thay đổi chúng.

Việc phục hồi mật khẩu (password recovery) khác nhau đối với các dòng router, tổng quát cách thực hiện như sau:

- Nối console vào router
- Tắt router và bật lại.
- Phải ngắt quá trình khởi động của router và đưa nó về monitor mode.
- Tại monitor mode, cấu hình router sao cho router khởi động mà không đọc nội dung file cấu hình trong NVRAM.
- Khởi động lại router.
- Sau khi router khởi động lại, mật khẩu sẽ không còn tác dụng. Vào privileged mode, xem, thay đổi, hay xoá NVRAM mật khẩu.
- Vào *global configuration mode* và đặt lại thanh ghi để router khởi động từ NVRAM.
- Nạp lại router. Lúc này mật khẩu đã biết.

Lưu ý: Khôi phục mật khẩu chỉ có thể thực hiện trên terminal (PC) gắn trực tiếp với cổng console của router.

❖ Cách thực hiện khôi phục mật khẩu cho Cisco Router:

- **Tắt router và bật lại**

Nhấn ngắt (Ctrl+Break) tại dòng xuất hiện dung lượng bộ nhớ RAM của Router. (Lưu ý rằng đối với các chương trình terminal tổ hợp phím ngắt khác nhau. Phổ biến là HyperTerminal dùng tổ hợp phím Ctrl+Break để ngắt.)

- **Lúc này router vào monitor mode, thực hiện lệnh confreg 0x2142**

```
rommon 1>confreg 0x2142
```

- **Router sẽ nhắc phải reset lại để thay đổi có tác dụng, đánh lệnh reset tại dấu nhắc:**

```
rommon 2>reset
```

Router sẽ nạp lại. Lúc này router đã bỏ qua cấu hình trong NVRAM.

- Dùng lệnh show running-configuration để xem cấu hình hiện tại. Chú ý rằng cấu hình không chứa loại mật khẩu nào. Đây là cấu hình mặc định của router khi bỏ qua file cấu hình trong NVRAM:

```
Current configuration:
!
version 11.2
...
!
line con 0 ← không có mật khẩu
line aux 0
line vty 0 4
login
!
end
```

Nếu muốn xem mật khẩu hiện tại và tiếp tục sử dụng nó, dùng lệnh show startup-configuration. Từ cấu hình bạn có thể thấy enable mật khẩu và console mật khẩu là “cisco”. Nếu mật khẩu đã mã hoá bạn sẽ phải thay đổi hay xoá nó.

```
Router#sh start
Using 355 out of 30712 bytes
!
hostname Cisco3600
!
enable pass cisco ← enable mật khẩu
!
no ip classless
!
line console 0
mật khẩu cisco ← Login mật khẩu
login
line aux 0
line vty 0 4
login
!
end
```

- ***Chép cấu hình NVRAM vào running configuration bằng lệnh***

```
Router#copy start run
```

Vào mode config bằng lệnh

```
Router#config terminal
```

Đặt lại mật khẩu mới. Nhấn Ctrl-Z để thoát khỏi configuration mode khi hoàn tất.

Đánh lệnh write memory hoặc copy run start để lưu cấu hình.

- ***Bước cuối cùng thay đổi thanh ghi cấu hình (configuration register) để router nạp từ NVRAM.***

Xem cấu hình thanh ghi hiện tại:

```
Router#show version
...
Configuration register is 0x2142
```

- **Thay đổi thanh ghi**

```
Router#config terminal
Router(config)#config-register 0x2102
Router(config)#exit
```

- **Dùng lệnh show version để xem ảnh hưởng của cài đặt mới**

```
Router#show version
...
Configuration register is 0x2142 (will be 0x2102 at next reload)
```

- **Đánh lệnh reload để nạp lại router và làm cho giá trị thanh ghi mới có tác dụng. Bạn không cần phải lưu thay đổi.**

```
Router#reload
```

• **Router sẽ khởi động lại. Nó sẽ lấy cấu hình từ NVRAM, mật khẩu của router bây giờ đã biết và bạn có thể truy cập vào privileged mode của router**

❖ **Khôi phục mật khẩu cho switch (switch password recovery)**

-
- Tắt Switch, bật lại
 - Press mode button (3s)
 - Thực hiện các lệnh sau


```
Switch:flash_init
Switch:load_helper
Switch:dir flash:
Switch:rename flash:config.text flash:config.old
Switch:boot
Switch#rename flash:config.old flash:config.text
Switch#copy flash:config.text running-config
Dùng lệnh show run để xem password hoặc đổi password mới.
Switch#copy run start
```

❖ **Xóa cấu hình switch**

```
Switch#delete vlan.dat
Switch#erase startup-config
Switch#reload
```

Lab 1-3. BACKUP & RESTORE CISCO IOS

TFTP Server

Backup IOS Router vào TFTP server: Router#copy flash tftp

Nạp IOS từ TFTP vào Router
192.168.1.254/2

❖ Yêu cầu

PC chạy TFTP server nối với router trong cùng mạng LAN. IOS image mới sẽ chứa trong PC và sẽ truyền qua Cisco router bằng giao thức truyền TFTP. PC hoạt động như TFTP server, router sẽ là TFTP client

❖ Cấu hình

- Cấu hình IP cho TFTP server và router như mô hình
- Dùng lệnh show version để tìm phiên bản của IOS hiện có. Lệnh này cung cấp một số thông tin như dung lượng bộ nhớ và image của router
- Xem nội dung bộ nhớ flash dùng lệnh show flash
- Để chắc chắn có thể truy cập được TFTP server ở địa chỉ 192.168.1.254, dùng lệnh ping để kiểm tra.
- Khi đã kiểm tra kết nối đến TFTP server, ta bắt đầu nạp IOS mới vào router bằng lệnh copy tftp flash
RouterA#copy tftp flash
- Sau khi quá trình nạp hoàn tất, ta kiểm tra nội dung của bộ nhớ flash bằng lệnh show flash
- Sau khi nạp xong hệ điều hành cho router, nên kiểm tra lại tình trạng các file hiện có trong flash bằng show flash hoặc dir flash:

❖ Nạp IOS cho router mất IOS:

Kiểm tra kết nối đến TFTP server.

Thực hiện các lệnh sau trên Router:

```
Rommon>IP_ADDRESS=192.168.1.1
Rommon>IP_SUBNET_MASK=255.255.255.0
Rommon>DEFAULT_GATEWAY=192.168.1.254
Rommon>TFTP_SERVER=192.168.1.254
Rommon>TFTP_FILE= c2960-lanbase-mz.122-35.SE5.bin
Rommon>sync
Rommon>tftpdnld
```

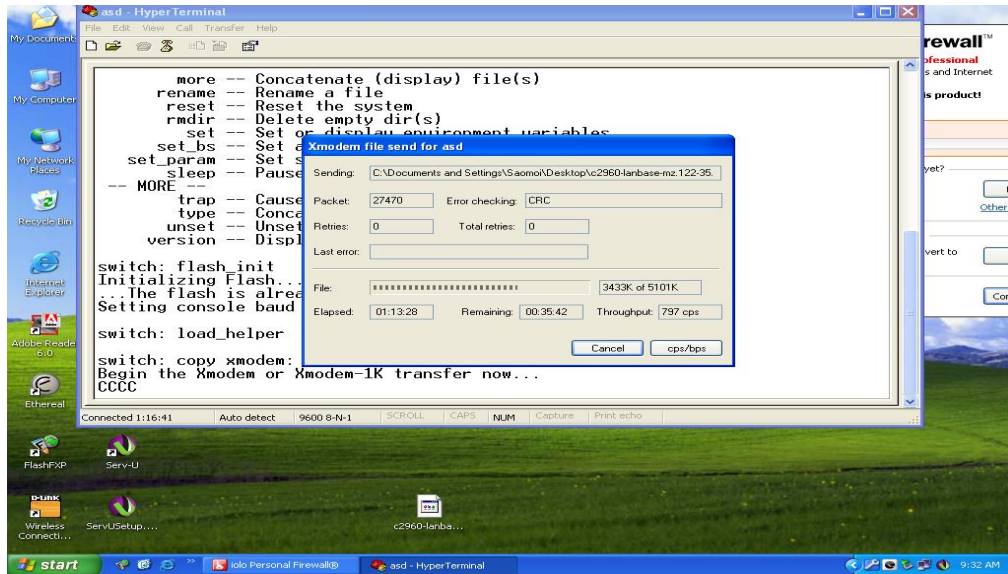
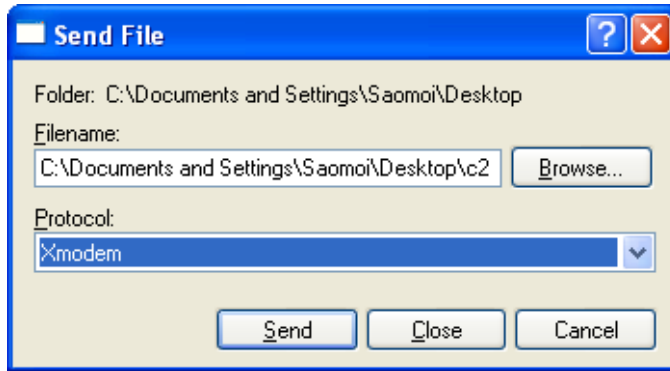
❖ Nạp IOS cho Switch mất IOS:

Kết nối với switch qua cổng console

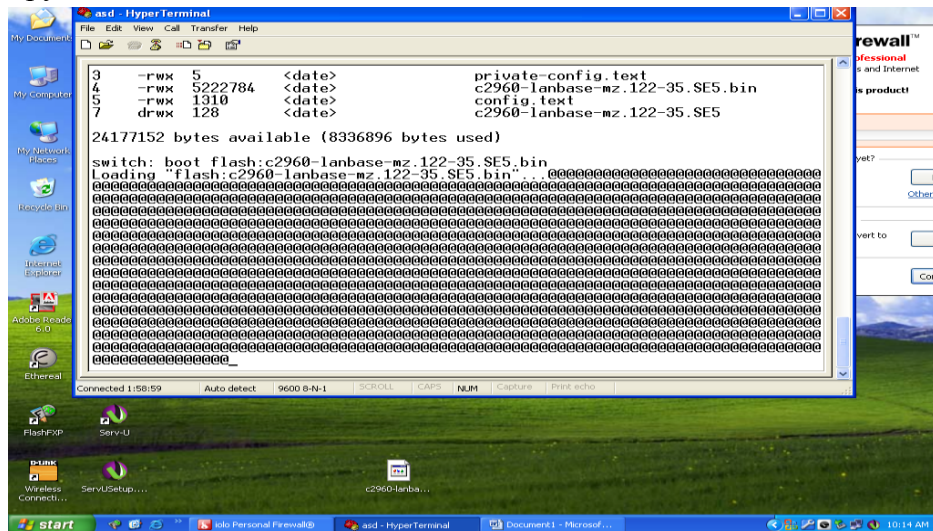
Switch: copy xmodem: flash:tenfile

ở cửa sổ hyper Terminal chọn

- Transfer/Send File...
- Chọn IOS và giao thức xmodem



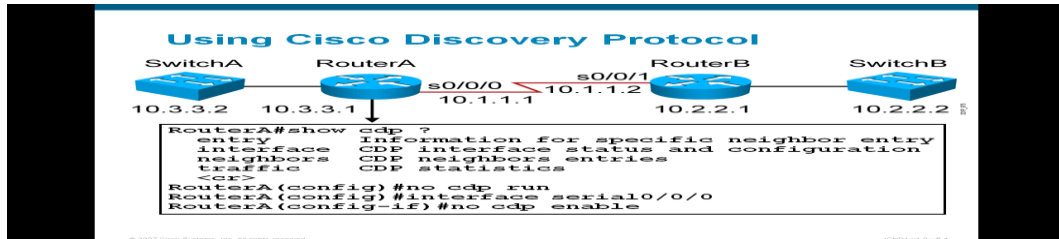
- Khi quá trình copy đã hoàn thành
- Sử dụng lệnh switch:boot flash:tenfile để switch khởi động lại với hệ điều hành vừa được copy vào switch



Lab 1-4

CISCO DISCOVERY PROTOCOL

Cisco Discovery Protocol (CDP) là giao thức riêng của Cisco. Nó được dùng để thu thập thông tin về các thiết bị lân cận. Khi sử dụng giao thức CDP, ta có thể biết được thông tin phần cứng, phần mềm của các thiết bị láng giềng. Thông tin này rất hữu ích trong quá trình xử lý sự cố hay kiểm soát các thiết bị trong một hệ thống mạng. Giao thức CDP mặc định được bật trên các thiết bị của Cisco.



- Để tắt chức năng CDP trên toàn bộ các cổng của Router:


```
Router#config terminal
Router(config)#no cdp run
```
- Để bật chức năng CDP trên toàn bộ các cổng của Router:


```
Router#config terminal
Router(config)#cdp run
```
- Để tắt chức năng CDP trên một interface của Router, ta vào interface đó và sử dụng lệnh `no cdp enable`
 Ví dụ:


```
Router#config terminal
Router(config)#interface fastethernet 0/0
Router(config-if)#no cdp enable
```
- Để bật chức năng CDP trên một interface của Router, ta vào interface đó và sử dụng lệnh `cdp enable`.
 Ví dụ:


```
Router#config terminal
Router(config)#interface fastethernet 0/0
Router(config-if)#cdp enable
```
- Các câu lệnh hiển thị thông tin cdp


```
Router#show cdp
Router#show cdp interfaces
Router#show cdp traffic
Router#show cdp neighbor
Router#show cdp neighbor detail
Router#show cdp entry *
```
- ❖ Kiểm tra kết nối với các lệnh **Telnet, Ping, Traceroute**
 - Lệnh Telnet: Kiểm tra kết nối tầng application(application layer)
 - Lệnh ping: Kiểm tra kết nối tầng Network (network layer)

ý nghĩa của một số kết quả hiển thị của lệnh ping

```
Router#ping
```

!	Successful echo reply
.	Timed out waiting for reply
U	Destination unreachable

C	Network congestion
I	Interrupted
?	Unknown packet type
&	Packet TTL exceeded

- Lệnh traceroute: Kiểm tra sự hoạt động tầng Network (network layer)

Using the show cdp neighbors Command

```

RouterA#show cdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater
Device ID        Local Intrfce   Holdtme    Capability   Platform  Port ID
SwitchA          fa0/0           122        S I          WS-C2960- fa0/2
RouterB          s0/0/0          177        R S I
    
```

Click to turn off Vietnamese mode

Chương 2.

ĐỊA CHỈ IPv4

Chương này đề cập đến địa chỉ IP, cấu trúc, phân lớp địa chỉ, kỹ thuật chia mạng con. Học xong chương này, người học có khả năng:

- Trình bày được vai trò của địa chỉ IP trong mạng
- Trình bày được cấu trúc địa chỉ IP, phân lớp địa chỉ IPv4
- Trình bày và vận dụng được các kỹ thuật chia mạng con
- Hoạch định được địa chỉ IP cho một sơ đồ mạng

1. Giới thiệu

Địa chỉ IP là địa chỉ được dùng để định danh cho một đối tượng trên mạng. Các đối tượng này có thể là máy tính, máy in, camera, điện thoại... gọi là các thiết bị người dùng cuối “end-user devices” hay là các “host”.

Cấu trúc địa chỉ IP gồm 2 phần:

Network.Host

Có khi người ta gọi là Net_ID và Host_ID, nghĩa là phần định danh cho phần Network và phần định danh cho Host. Các địa chỉ IP có cùng phần Network gọi là cùng mạng. Các địa chỉ IP trên một mạng là duy nhất.

2. Phân lớp địa chỉ IPv4

Địa chỉ IPv4 có 32 bit, chia làm 4 phần (octet), ngăn cách nhau bởi dấu “.”, được biểu diễn dưới dạng thập phân hoặc nhị phân. Địa chỉ IPv4 được chia thành 5 lớp: A, B, C, D, E

Trong đó:

- Các lớp A, B, C được dùng để gán cho các host
- Lớp D là lớp địa chỉ multicast
- Lớp E không dùng

❖ Đặc điểm của các lớp

- **Lớp A (class A):**

0 x x x x x x x . Host . Host . Host

- Dành 1 octet đầu tiên làm phần Network, 3 octet còn lại làm phần host
- Bit đầu tiên của octet đầu tiên phải là bit 0
- Một mạng lớp A có thể đánh cho $2^{24} - 2 = 16.777.214$ (*) host

Như vậy: octet đầu tiên có giá trị:

- 00000000 đến 01111111 (viết dưới dạng nhị phân)
- Hay từ 0 đến 127 (viết dưới dạng thập phân)

Lưu ý:

- Giá trị đầu tiên 00000000 không dùng
- Giá trị cuối: 01111111 (127) được dùng làm địa chỉ loopback

Kết luận: Địa chỉ lớp A có octet đầu tiên mang giá trị 00000001 đến 01111110 (hay từ 1 đến 126)

Ví dụ: 10.10.10.1

00001010.00001010.00001010.00000001

- **Lớp B (class B):**



- Dành 2 octet đầu tiên làm phần Network, 2 octet còn lại làm phần host
- 2 Bit đầu tiên của octet đầu tiên phải là bit 10
- Một mạng lớp A có thể đánh cho $2^{16} - 2 = 65.534$ (*) host

Như vậy: octet đầu tiên có giá trị:

- 10000000 đến 10111111 (viết dưới dạng nhị phân)
- Hay từ 128 đến 191 (viết dưới dạng thập phân)

Ví dụ: 172.16.10.1

10101100.00010000.00001010.00000001

- **Lớp C (class C):**



- Dành 3 octet đầu tiên làm phần Network, 1 octet còn lại làm phần host
- 3 Bit đầu tiên của octet đầu tiên phải là bit 110
- Một mạng lớp A có thể đánh cho $2^8 - 2 = 254$ (*) host

Như vậy: octet đầu tiên có giá trị:

- 11000000 đến 11011111 (viết dưới dạng nhị phân)
- Hay từ 192 đến 223 (viết dưới dạng thập phân)

Ví dụ: 192.168.1.1

11000000.10101000.00000001.00000001

- **Lớp D (class D): Địa chỉ lớp D là địa chỉ Multicast**

- 4 Bit đầu tiên của octet đầu tiên phải là bit 1110

Như vậy: octet đầu tiên có giá trị:

- 11100000 đến 11101111 (viết dưới dạng nhị phân)
- Hay từ 224 đến 239 (viết dưới dạng thập phân)

Ví dụ: 224.0.0.5

11100000.00000000.00000000.00000101

- **Lớp E (class E): Chưa sử dụng**

5 Bit đầu tiên của octet đầu tiên phải là 11110

(*): Trong mỗi mạng: có 2 địa chỉ không dùng để gán cho các host

- **Địa chỉ mạng (network address):** là địa chỉ mà tất cả các bit ở phần Host đều là bit 0
- **Địa chỉ Broadcast (broadcast address):** là địa chỉ mà tất cả các bit ở phần Host đều là bit 1

3. IP public và IP private

Dãy địa chỉ IP private (RFC 1918)

- o Class A: **10.0.0.0** → **10.255.255.255**
- o Class B: **172.16.0.0** → **172.31.255.255**
- o Class C: **192.168.0.0** → **192.168.255.255**

4. Subnet Mask

Subnet Mask có chiều dài bit bằng với địa chỉ IP được dùng để chỉ ra trong một địa chỉ IP những bit nào thuộc phần *Network* và những bit nào thuộc phần *Host*. Trong đó, các bit 1 chỉ ra tương ứng các bit thuộc phần *Network* và các bit 0 chỉ ra tương ứng các bit thuộc phần *Host*.

Subnet mask được biểu diễn dưới dạng: (1) 4 octet giống như địa chỉ IP hoặc (2) /n với n là số bit làm phần Network.

Subnet Mask mặc định (Default Subnet Mask) cho các lớp IP như sau:

- Class A: 255.0.0.0 hoặc /8
- Class B: 255.255.0.0 hoặc /16
- Class C: 255.255.255.0 hoặc /24

5. Kỹ thuật chia mạng con

- Subnetting là một kỹ thuật cho phép tạo ra nhiều mạng con (subnetworks) từ một mạng lớn (major network). Với kỹ thuật này cho phép tạo ra nhiều mạng con (subnetwork) với số lượng host ít hơn, phù hợp cho nhu cầu sử dụng và tối ưu cho hệ thống.
- Để thực hiện điều này, người ta sử dụng một số bit ở phần **Host-ID** tham gia vào phần **Network-ID**.

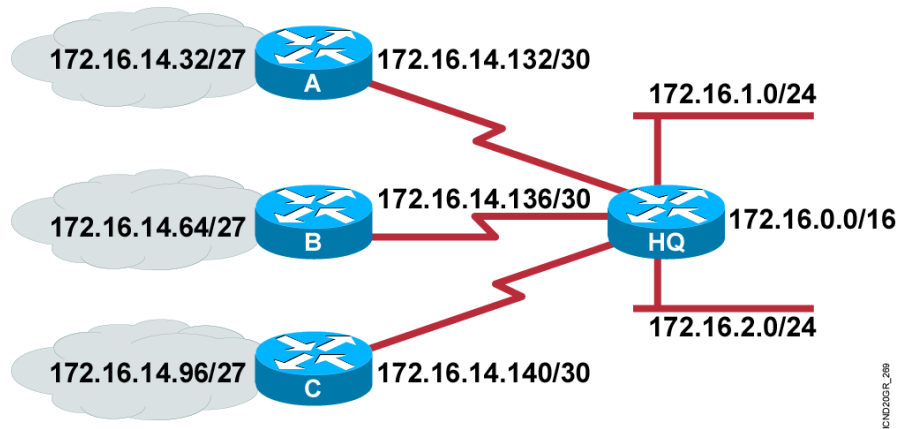


Ta có một số tính chất cần lưu ý như sau:

- Nếu gọi n là số bit mượn ở phần Host để chia subnetting thì số mạng con (subnetwork) có thể chia là 2^n
- Gọi m là số bit còn lại còn lại của phần host thì số host cho mỗi mạng con là $2^m - 2$
- $n + m =$ số bit phần host của mạng ban đầu.

6. Kỹ thuật VLSM

VLSM (Variable Length Subnet Mask) là kỹ thuật chia mạng con trong đó các mạng con (subnet) của cùng một mạng ban đầu (major network) sau khi chia có chiều dài Subnet mask (số bit thuộc phần network) khác nhau.



Trong mô hình trên sử dụng subnet 172.16.14.0/24 chia thành các subnet nhỏ hơn (sub-subnet) với các subnet-mask /27 và /30.

VLSM được sử dụng trong các giao thức định tuyến:

- OSPF
- IS-IS
- EIGRP
- RIP v2
- Static route

7. Một số dạng bài tập IP

a) **Dạng 1.** Xác định địa chỉ mạng của một địa chỉ IP cho trước

❖ **Phương pháp 1:** Áp dụng công thức

$$\text{Network Address} = \text{IP Address AND Subnet_Mask}$$

Ta có: Subnet Mask = /28 = 255.255.255.240

Bây giờ thực hiện phép AND giữa IP và Subnet Mask:

- Nhận xét:

$$1 \text{ AND } x = x$$

$$0 \text{ AND } x = 0$$

Trong đó: $x=0$ hoặc $x=1$

- Ta có: 255 (thập phân) đổi ra hệ nhị phân là: 11111111 (8 bit 1)
- Do vậy: **192.168.12.158 AND 255.255.255.240**

3 octet đầu ta được kết quả là **192.168.12**. (do 255.255.255 → tất cả là bit 1)

Bây giờ chỉ cần thực hiện phép AND giữa 158 và 240. Ta đổi sang nhị phân và thực hiện phép AND

158 (thập phân) = 1 0 1 0 1 1 1 0 (nhị phân)

240 (thập phân) = 1 1 1 1 0 0 0 0 (nhị phân)

1 0 1 0 0 0 0 0 (nhị phân) = **144** (thập phân)

Kết luận: 192.168.12.144/28 là địa chỉ mạng cần tìm.

❖ Phương pháp 2: dùng tính chất: **địa chỉ mạng (network address) là địa chỉ mà tất cả các bit phần Host (Host_id) đều là bit 0.**

Bước 1: Xác định đường ranh giới giữa phần Net_ID và Host_ID trong địa chỉ IP dựa vào **Subnet Mask**.

Bước 2: Xác định giá trị của octet chứa đường ranh giới

Bước 3: Cho tất cả các bit phần Host_ID = 0 → Xác định địa chỉ chỉ mạng.

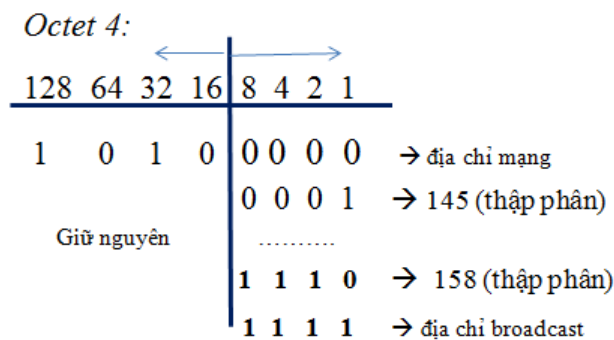
b) Dạng 2. Xác định dãy địa chỉ IP của một mạng

Bước 1: Xác định đường ranh giới giữa phần Network và Host dựa vào Subnet Mask

Bước 2: Xác định giá trị của octet chứa đường ranh giới

Bước 3: Xác định IP đầu và IP cuối của dựa vào các bit phần Host.

Ví dụ: Tìm dãy IP của mạng có chứa của địa chỉ IP: 192.168.12.158/28



Dãy địa chỉ IP cần tìm là: 192.168.12.**145/28** → 192.168.12.**158/28**

8. Bài tập

1. Địa chỉ IP nào sau đây là địa chỉ thuộc lớp B (class B)?
 - A. 10.10.10.1
 - B. 100.128.254.1
 - C. 190.162.41.1
 - D. 192.168.12.1
2. Địa chỉ IP nào sau đây là địa chỉ dạng Private?
 - A. 11.11.11.11
 - B. 172.30.150.1

- b) PC-1 đang ping tới FTP-Server. Xác định các địa chỉ MAC nguồn, MAC đích và IP nguồn và IP đích trong frame mà FTP-Server nhận được?

Chương 3.

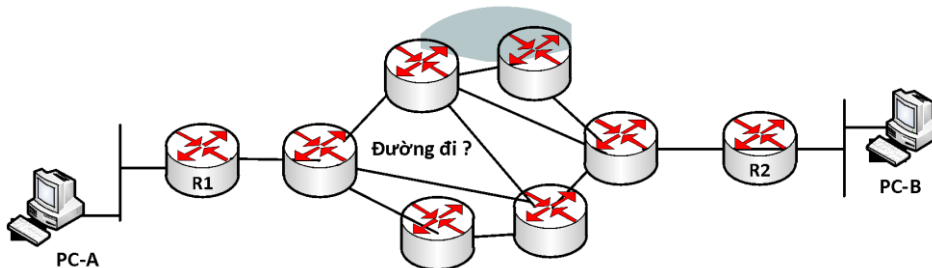
ĐỊNH TUYẾN

Chương này trình bày một số vấn đề cơ bản về định tuyến, phân loại định tuyến, đặc điểm của một số giao thức định tuyến phổ biến và cách cấu hình trên thiết bị của Cisco. Học xong chương này, người học có khả năng:

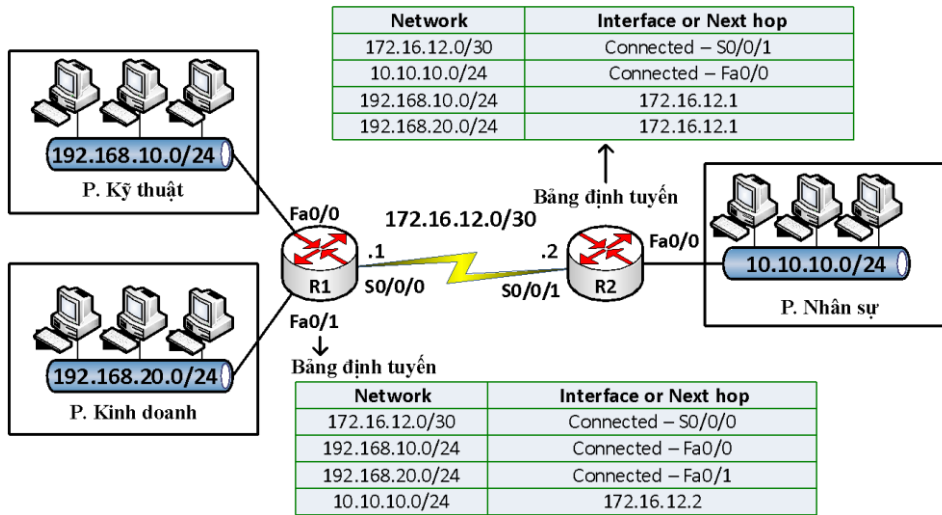
- Phân biệt được định tuyến tĩnh và định tuyến động
- Phân biệt được giao thức định tuyến dạng distance-vector, link-state, classful và classless
- Trình bày được đặc điểm của các giao thức RIP, OSPF, EIGRP
- Cấu hình định tuyến tĩnh, định tuyến động bằng các giao thức RIP, OSPF, EIGRP

1. Giới thiệu

Định tuyến là chức năng của router giúp xác định quá trình tìm đường đi cho các gói tin từ nguồn tới đích thông qua hệ thống mạng



Router dựa vào địa chỉ IP đích (destination IP) trong các gói tin và sử dụng bảng định tuyến (routing table) để xác định đường đi cho chúng.



Trong bảng định tuyến, mỗi mạng mà router có thể chuyển đi (mạng đích) thể hiện bằng một dòng. Mỗi mạng này có được có thể do chúng đang kết nối trực tiếp với router đang xét hay router học được thông qua việc cấu hình định tuyến.

2. Phân loại định tuyến

Có hai loại định tuyến là : định tuyến tĩnh và định tuyến động

2.1. Định tuyến tĩnh – static routing

Định tuyến tĩnh là loại định tuyến mà trong đó router sử dụng các tuyến đường đi tĩnh để vận chuyển dữ liệu đi. Các tuyến đường đi tĩnh này có được do người quản trị cấu hình thủ công vào các router.

Cấu hình:

R(config)#ip route <destination-net> <subnet-mask> <NextHop|OutPort>

Trong đó:

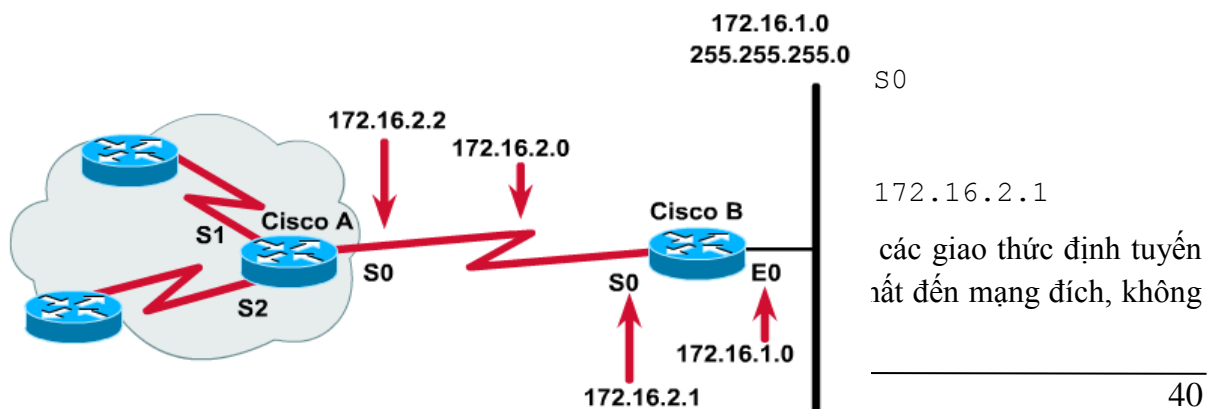
destination-network: là địa chỉ mạng cần đi tới

subnet-mask: subnet mask của destination-network

next-hop: địa chỉ IP của router kế tiếp kết nối trực tiếp với router đang xét

OutPort: cổng của router mà packet sẽ đi ra

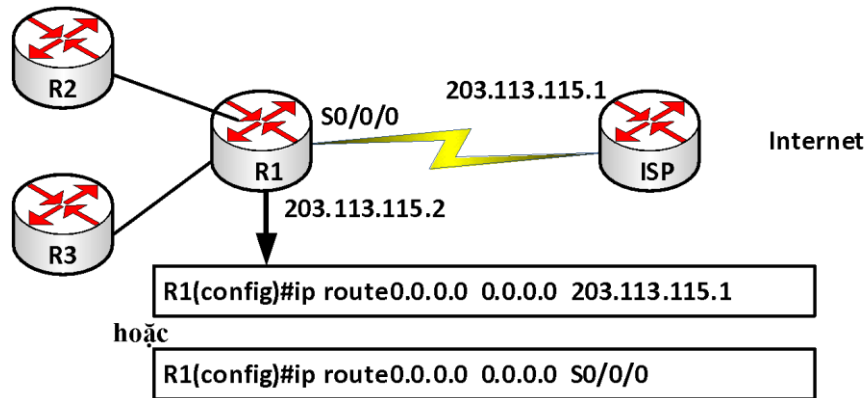
Ví dụ: Cấu hình trên router Cisco A để học mạng 172.16.1.0/24



còn đường nào khác phải chọn lựa. Khi đó, ta sẽ cấu hình đường **default route** cho hệ thống mạng.

❖ Default route

Default route nằm ở cuối bảng định tuyến và được sử dụng để gửi các gói tin đi trong trường hợp mạng đích không tìm thấy trong bảng định tuyến. Nó rất hữu dụng trong các mạng dạng “*stub network*” như kết nối từ mạng nội bộ ra ngoài Internet.



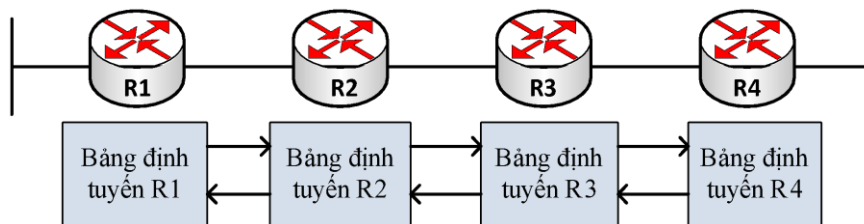
2.2. Định tuyến động

Định tuyến động là loại định tuyến mà trong đó router sử dụng các tuyến đường đi động để vận chuyển dữ liệu đi. Các tuyến đường đi động này có được do các router sử dụng các giao thức định tuyến động trao đổi thông tin định tuyến với nhau tạo ra.

Một số giao thức định tuyến động phổ biến: RIP, OSPF, BGP,...

Giao thức định tuyến động chia làm hai loại là *distance-vector* và *link-state*

• Distance vector



Giao thức định tuyến thuộc loại này như RIP,...

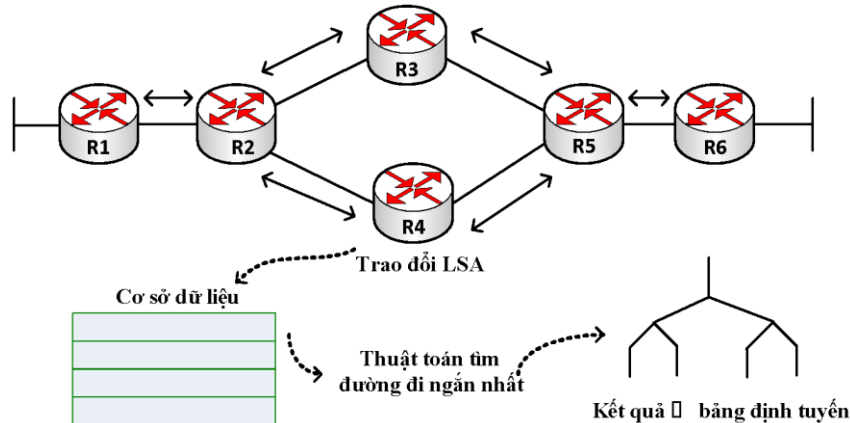
Các router định tuyến theo **Distance vector** thực hiện gửi định kỳ toàn bộ bảng định tuyến của mình và chỉ gửi cho các router láng giềng kết nối trực tiếp với mình.

Các router định tuyến theo *Distance vector* không biết được đường đi đến đích một cách cụ thể, không biết về các router trung gian trên đường đi và cấu trúc kết nối giữa chúng.

Bảng định tuyến là nơi lưu kết quả chọn đường tốt nhất của mỗi router. Do đó, khi chúng trao đổi bảng định tuyến với nhau, các router chọn đường dựa trên kết quả đã chọn của router láng giềng. Mỗi router nhìn hệ thống mạng theo sự chi phối của các router láng giềng.

Các router định tuyến theo *distance vector* thực hiện cập nhật thông tin định tuyến theo định kỳ nên tốn nhiều băng thông đường truyền. Khi có sự thay đổi xảy ra, router nào nhận biết sự thay đổi đầu tiên sẽ cập nhật bảng định tuyến của mình trước rồi chuyển bảng định tuyến cập nhật cho các router láng giềng.

- **Link state**



Các giao thức định tuyến thuộc loại này như OSPF, IS-IS

Trong các giao thức định tuyến link-state, các router sẽ trao đổi các LSA (link state advertisement) với nhau để xây dựng và duy trì cơ sở dữ liệu về trạng thái các đường liên kết hay còn gọi là cơ sở dữ liệu về cấu trúc mạng (topology database). Các thông tin trao đổi được gửi dưới dạng multicast.

Như vậy mỗi router đều có một cái nhìn đầy đủ và cụ thể về cấu trúc của hệ thống mạng. Từ đó mỗi router sẽ dùng thuật toán SPF để tính toán chọn đường đi tốt nhất đến từng mạng đích.

Khi các router định tuyến theo **link-state** đã hội tụ xong, nó không thực hiện cập nhật định tuyến định kỳ mà chỉ cập nhật khi nào có sự thay đổi xảy ra. Do đó thời gian hội tụ nhanh và ít tốn băng thông.

Giao thức định tuyến theo **link-state** có hỗ trợ CIDR, VLSM nên chúng là một chọn lựa tốt cho các mạng lớn và phức tạp. Nhưng đồng thời nó đòi hỏi dung lượng bộ nhớ lớn và khả năng xử lý mạnh của CPU của router.

Để đảm bảo là các database luôn cập nhật thông tin mới, trong các LSA này được đánh thêm chỉ số sequence. Chỉ số sequence được bắt đầu từ giá trị *initial* đến giá trị *Max-age*. Khi một router nào đó tạo ra một LSA, nó sẽ đặt giá trị sequence bằng *initial*. Mỗi khi router gửi ra một phiên bản LSA update khác, nó sẽ tăng giá trị đó lên 1. Như vậy, giá trị *sequence* càng cao thì LSA update càng mới.

Nếu giá trị *sequence* này đạt đến *max-age*, router sẽ flood LSA ra cho tất cả các router còn lại, sau đó router đó sẽ set giá trị *sequence* về *initial*.

- ❖ Ngoài cách phân chia các giao thức định tuyến động theo hai loại : **distance vector** và **link-state** như chúng ta đã tìm hiểu bên trên, các giao thức định tuyến còn được phân thành hai loại, đó là **classfull routing protocol** và **classless routing protocol**.

- **Classfull routing protocol**

Các giao thức định tuyến nhóm classfull không quảng bá *subnet-mask* cùng với địa chỉ đích trong các gói tin cập nhật định tuyến (routing update). Do đó, khi router nhận được các update này, router phải lấy giá trị *network-mask* mặc định có cùng với địa chỉ lớp mạng của địa chỉ đích.

Nếu địa chỉ đích được kết nối trực tiếp với router, *network-mask* được lấy cùng với *mask* được cấu hình trên interface kết nối đến mạng đó. Nếu địa chỉ đích không nối trực tiếp (*disconnected*), router sẽ lấy địa chỉ *subnetmask default* của địa chỉ đích.

- ***Classless routing protocol***

Các giao thức định tuyến thuộc nhóm classless sẽ quảng bá subnet –mask cùng với địa chỉ đích trong các gói tin cập nhật định tuyến.

- ❖ Hai tham số quan trọng trong định tuyến: Metric và AD

- ✓ ***Metric***

Là tham số được sử dụng để chọn đường tốt nhất cho việc định tuyến. Đây là giá trị mà bất kỳ giao thức định tuyến nào cũng phải dùng để tính toán đường đi đến mạng đích.

Trong trường hợp có nhiều đường đi đến một mạng đích thì đường đi nào có *metric* thấp nhất sẽ được lựa chọn để đưa vào bảng định tuyến. Mỗi giao thức định tuyến có một kiểu *metric* khác nhau.

- ✓ ***AD***

AD (Administrative Distance) là giá trị quy ước dùng để chỉ độ tin cậy của các giao thức định tuyến, giao thức nào có AD nhỏ hơn sẽ được xem là đáng tin cậy hơn. Trong trường hợp router học được một mạng đích thông qua nhiều giao thức định tuyến khác nhau, thì tuyến của giao thức định tuyến nào có AD nhỏ nhất thì sẽ được lựa chọn và đưa vào bảng định tuyến.

3. Cấu hình định tuyến động – distance vector

3.1. RIP

RIP là một giao thức định tuyến theo kiểu *distance-vector*. Hop count được sử dụng làm *metric* cho việc chọn đường. Nếu có nhiều đường đến cùng một đích thì RIP sẽ chọn đường nào có số *hop-count* (số router) ít nhất.

Nếu *hop-count* lớn hơn 15 thì packet bị loại bỏ. Mặc định thời gian update là 30 giây. *Administrative Distance* là 120.

RIP có hai phiên bản là RIPv1 và RIPv2.

RIPv1:

RIPv1 là một giao thức định tuyến theo kiểu *distance-vector* và là một giao thức định tuyến theo lớp (classfull routing protocol). *Metric* của RIP là hop-count. Cập nhật định tuyến theo chu kỳ mặc định là 30 giây. Hop-count tối đa để chuyển gói là 15.

RIPv1 không hỗ trợ VLSM và mạng không liên tục (discontiguous network).

Các câu lệnh cấu hình

```
Router(config)#router rip
```

```
Router(config-router)#network network_number
```

RIPv2:

RIPv2 là một phiên bản cải tiến của RIPv1. RIPv2 là giao thức định tuyến dạng classless, nghĩa là có gửi thông tin subnet-mask qua cập nhật định tuyến. Nó hỗ trợ VLSM, hỗ trợ chứng thực trong các cập nhật định tuyến.

RIPv2 cập nhật định tuyến dạng multicast, sử dụng địa chỉ lớp D 224.0.0.9.

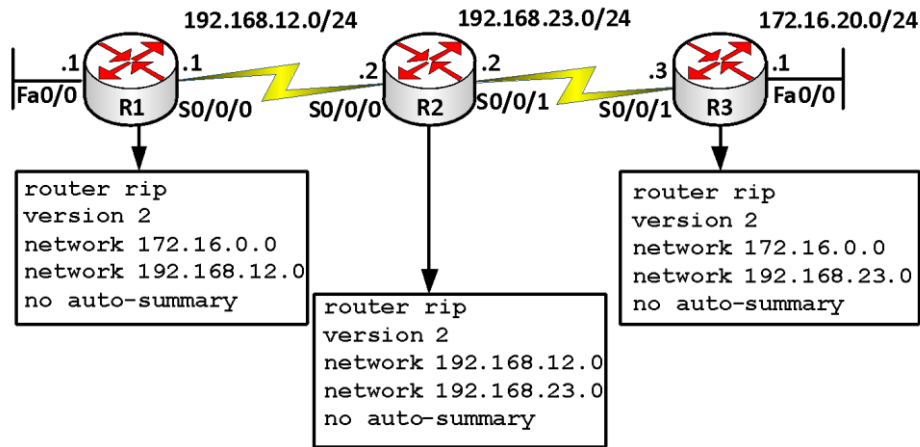
- Metric của RIPv2

Giống như RIPv1, RIPv2 sử dụng metric là hop-count.

- Cấu hình RIPv2

```
Router (config) #router rip
Router (config-router) #version 2
Router (config-router) #network network-number
```

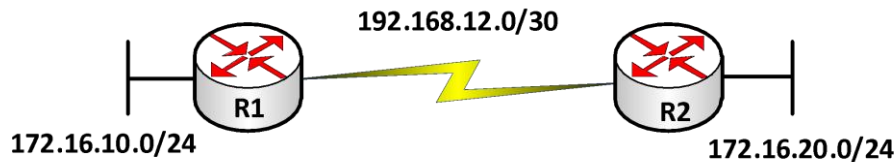
Ví dụ:



Bảng so sánh giữa RIPv1 và RIPv2

Đặc điểm	RIPv1	RIPv2
Loại định tuyến	Classful	Classless
Hỗ trợ VLSM và mạng không liên tục	Không	Có
Gửi kèm Subnet-mask trong bản tin cập nhật định tuyến	Không	Có
Quảng bá thông tin định tuyến	Broadcast	Multicast
Hỗ trợ tóm tắt các tuyến thủ công	Không	Có
Hỗ trợ chứng thực	Không	Có
Định nghĩa trong RFC	RFC 1058	RFC 1721, 1722, 2453

- **Mạng không liên tục (discontiguous network):** là mạng mà trong đó các mạng con (subnet) của cùng một mạng lớn (major network: là mạng theo đúng lớp) bị ngăn cách bởi “major-network” khác.



❖ Chứng thực trong RIPv2

Chứng thực trong định tuyến là cách thức bảo mật trong việc trao đổi thông tin định tuyến giữa các router. Nếu có cấu hình chứng thực thì các router phải vượt qua quá trình này trước khi các thông tin trao đổi định tuyến được thực hiện. RIPv2 hỗ trợ hai kiểu chứng thực là: “Plain text” và “MD5”

- Chứng thực dạng “Plain Text”: còn gọi là “Clear text”

Quá trình chứng thực chỉ đơn giản là các router được cấu hình một khóa (password) và trao đổi chúng để so khớp. Các khóa này được gửi dưới dạng không mã hóa trên đường truyền.

Các bước cấu hình:

Bước 1. Tạo bộ khóa

```
Router(config)#key chain <name>
```

Bước 2. Tạo các khóa

```
Router(config-keychain)#key <key-id>
```

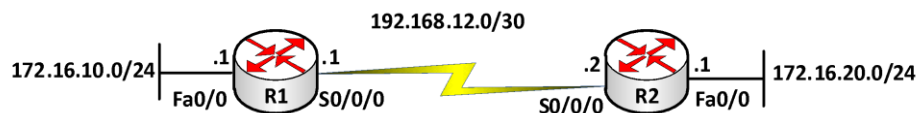
```
Router(config-keychain-key)#key-string <password>
```

Bước 3. Áp đặt vào cổng gửi chứng thực

```
Router(config)#interface <interface>
```

```
Router(config-if)#ip rip authentication key-chain <name>
```

Ví dụ: Cấu hình chứng thực trong định tuyến RIPv2 dạng “Plain Text”



```
R1(config)#key chain newstar
```

```
R1(config-keychain)#key 1
```

```
R1(config-keychain-key)#key-string ccna
```

```
R1(config)#interface S0/0/0
```

```
R1(config-if)#ip rip authentication key-chain newstar
```

```
R2(config)#key chain newstar2
```

```
R2(config-keychain)#key 1
```

```
R2(config-keychain-key)#key-string ccna
```

```
R2(config)#interface S0/0/0
```

```
R2(config-if)#ip rip authentication key-chain newstar2
```

- **Chứng thực dạng MD5**

Dạng chứng thực này sẽ gửi thông tin về khóa đã được mã hóa giúp các thông tin trao đổi được an toàn hơn. Các bước cấu hình tương tự như dạng “Plain Text”, chỉ có khác ở bước 3 phải thêm 1 lệnh sau:

```
Router(config-if)#ip rip authentication mode md5
```

Ví dụ: Sử dụng lại mô hình mạng trong ví dụ chứng thực dạng “Plain Text”, chúng ta sẽ cấu hình chứng thực định tuyến RIPv2 bằng MD5 với tên bộ khóa là “*spkt*” và mật khẩu là “123456” trên R1 và tên bộ khóa là “*cntt*” và mật khẩu là “123456” trên R2

```
R1(config)#key chain spkt
```

```
R1(config-keychain)#key 1
```

```
R1(config-keychain-key)#key-string 123456
```

```
R1(config)#interface S0/0/0
```

```
R1(config-if)#ip rip authentication mode md5
```

```
R1(config-if)#ip rip authentication key-chain spkt
```

```
R2(config)#key chain cntt
```

```
R2(config-keychain)#key 1
```

```
R2(config-keychain-key)#key-string 123456
```

```
R2(config)#interface S0/0/0
```

```
R2(config-if)#ip rip authentication mode md5
```

```
R2(config-if)#ip rip authentication key-chain cntt
```

- ❖ **Các lệnh kiểm tra cấu hình**

```
R#debug ip rip
```

```
R#show ip route
```

3.2. OSPF

OSPF (*Open Shortest Path First*) là một giao thức định tuyến dạng *link-state*, sử dụng thuật toán Dijkstra “Shortest Path First (SPF)” để xây dựng bảng định tuyến.

OSPF mang những đặc điểm của giao thức link-state. Nó có ưu điểm là hội tụ nhanh, hỗ trợ được mạng có kích thước lớn và không xảy ra *routing loop*. Là giao thức định tuyến dạng *classless* nên hỗ trợ VLSM và mạng không liên tục (discontiguous network). OSPF sử dụng địa chỉ multicast 224.0.0.5 và 224.0.0.6 (DR và BDR router) để gửi các thông điệp *hello* và *update*.

Bên cạnh đó OSPF còn sử dụng *area* để giảm yêu cầu về CPU, memory của OSPF router cũng như lưu lượng định tuyến. OSPF còn có khả năng hỗ trợ chứng thực dạng plain-text và dạng MD5.

- ❖ **Metric của OSPF**

OSPF sử dụng *metric* là *cost*. Cost của toàn tuyến được tính theo cách cộng dồn cost dọc theo tuyến đường đi của packet. Cách tính cost được IETF đưa ra trong RFC 2328.

Cost được tính dựa trên băng thông sao cho tốc độ kết nối của đường link càng cao thì cost càng thấp dựa trên công thức $10^8/\text{bandwidth}$ với giá trị *bandwidth* được cấu hình trên mỗi interface và đơn vị tính là *bps*.

Tuy nhiên, chúng ta có thể thay đổi giá trị cost. Nếu router có nhiều đường đến đích mà cost bằng nhau thì router sẽ cân bằng tải trên các đường đó (tối đa là 16 đường). Những tham số bắt buộc phải giống nhau trong các router chạy OSPF trong một hệ thống mạng đó là *Hello/dead interval*, *Area – ID*, *authentication password* (nếu có), *stub area flag*.

❖ Các loại môi trường OSPF

- Multiple access (ethernet)
- Point-to-point
- NBMA (Non-Broadcast Multiple Access)

❖ Quá trình xây dựng bảng định tuyến của OSPF

- Các OSPF gửi các gói *hello* định kỳ để thiết lập quan hệ láng giềng (*neighbor*). Gói tin *hello* mang các thông tin thương lượng với các router *neighbor* trước khi thiết lập quan hệ adjacency. Trong mạng đa truy cập, giao thức *hello* sẽ bầu ra DR và BDR. DR và BDR sẽ thiết lập mối quan hệ adjacency với tất cả các router khác và những router này chỉ trao đổi thông tin với DR và BDR. Trong mạng point-to-point không cần chọn DR và BDR.
- Mỗi router nhận một LSA từ *neighbor* với cơ sở dữ liệu về trạng thái các đường liên kết (*link-state database*) của *neighbor* đó và gửi một copy của LSA tới tất cả *neighbor* khác của nó.
- Bằng cách flooding các LSA cho toàn bộ một area, tất cả router sẽ xây dựng chính xác link state database. Khi database được hoàn tất, mỗi router sử dụng thuật toán SPF để xây dựng nên cây SPF.
- Mỗi router sẽ xây dựng nên bảng định tuyến từ cây SPF. Kết quả là mỗi router sẽ có thông tin về đường đến tất cả các mạng đích trong hệ thống mạng.

❖ Quá trình bầu chọn DR và BDR

Quá trình bầu chọn liên quan đến 2 tham số: độ ưu tiên (*priority*) và *router ID*. Tham số *priority* được chọn trước tiên, giá trị *priority* nằm trong khoảng từ 0 đến 255. Nếu *priority* đặt là 0 thì router này sẽ không tham gia vào quá trình bầu chọn DR/BDR. Router nào có độ ưu tiên cao nhất sẽ được chọn là DR, cao thứ hai sẽ là BDR. Mặc định giá trị *priority* OSPF là 1. Khi giá trị *priority* đều bằng nhau thì OSPF sẽ bầu chọn DR dựa vào tham số thứ hai là *router ID*.

Trong hệ thống mạng dùng OSPF không cấu hình cổng *interface loopback* thì giá trị *router ID* được chọn là giá trị địa chỉ IP lớn nhất của các interface đang hoạt động (*active*

interface) của router. Nếu có cổng loopback thì cổng loopback được chọn, trường hợp có nhiều cổng loopback thì chọn cổng loopback nào có địa chỉ IP cao nhất.

❖ Cấu hình OSPF

- Khởi tạo tiến trình định tuyến OSPF

```
Router(config)#router ospf <process-id>
```

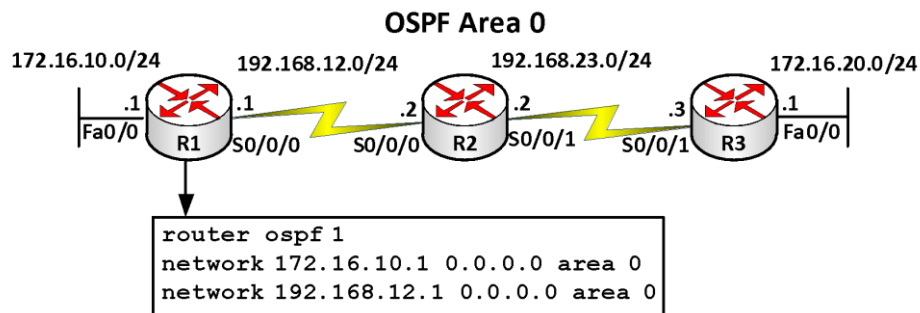
- Chọn cổng tham gia vào quá trình trao đổi thông tin định tuyến

```
Router(config-router)#network <address> <wildcard-mask> area <area-id>
```

Trong đó:

- *Process-id*: chỉ số tiến trình của OSPF, mang tính chất cục bộ, có giá trị 1 đến 65535.
- *Address*: địa chỉ cổng tham gia định tuyến
- *Wildcard mask*: điều kiện kiểm tra giữa địa chỉ cấu hình trong address và địa chỉ các cổng trên router, tương ứng bit 0 – phải so khớp, bit 1 – không cần kiểm tra.
- *Area-id*: vùng mà cổng tương ứng thuộc về trong kiến trúc OSPF.

Ví dụ:



❖ Các câu lệnh kiểm tra cấu hình OSPF

```
Router#show ip protocol
```

```
Router#show ip route
```

```
Router#show ip ospf interface
```

```
Router#show ip ospf neighbor
```

```
Router#debug ip ospf events
```

```
Router#debug ip ospf packet
```

❖ Chứng thực trong OSPF

Giao thức OSPF hỗ trợ hai dạng chứng thực là: “Plain Text” và MD5

- Chứng thực bằng “Plain Text”

Cấu hình giữa hai cổng của 2 router nối trực tiếp với nhau để chứng thực giữa chúng trước khi trao đổi thông tin định tuyến. Mật khẩu gửi chứng thực không được mã hóa.

```
R(config)#interface <interface>
```

```
R(config-if)#ip ospf authentication
```

```
R(config-if)#ip ospf authentication-key <password>
```

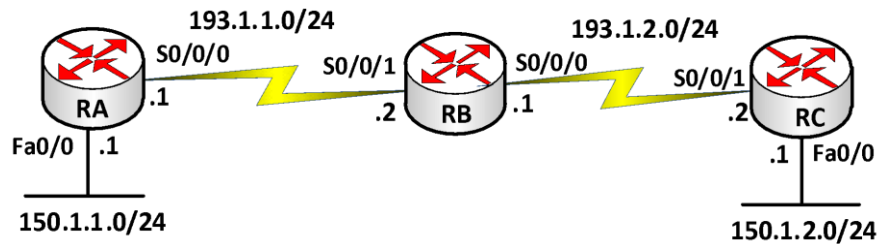
- Chứng thực bằng MD5

Trên cổng của router gửi thông tin chứng thực cấu hình lệnh sau:

```
R(config)#interface <interface>
R(config-if)#ip ospf authentication message-digest
R(config-if)#ip ospf messages-digest-key 1 md5 <password>
```

Ví dụ 1: Cho mô hình mạng sau.

Yêu cầu: Cấu hình OSPF cho các router RA, RB và RC (Area 0) trong mô hình mạng sau để quảng bá các thông tin định tuyến. Cấu hình chứng thực dạng “Plain Text” và MD5 giữa 2 router: RA và RB với mật khẩu là “cisco”.



Hướng dẫn cấu hình:

Bước 1: Cấu hình cơ bản (đặt hostname, địa chỉ IP cho các cổng: Serial, FastEthernet)

Bước 2: Cấu hình giao thức định tuyến OSPF trên mỗi router

```
RA(config)#router ospf 1
RA(config-router)#network 150.1.1.0 0.0.0.255 area 0
RA(config-router)#network 193.1.1.0 0.0.0.255 area 0

RB(config)#router ospf 1
RB(config-router)#network 193.1.1.0 0.0.0.255 area 0
RB(config-router)#network 193.1.2.0 0.0.0.255 area 0

RC(config)#router ospf 1
RC(config-router)#network 150.1.2.0 0.0.0.255 area 0
RC(config-router)#network 193.1.2.0 0.0.0.255 area 0
```

Bước 3.1. Cấu hình chứng thực dạng “Plain Text” giữa 2 router: RA và RB

```
RA(config)#int S0/0/0
RA(config-if)#ip ospf authentication
RA(config-if)#ip ospf authentication-key cisco

RB(config)#int S0/0/1
RB(config-if)#ip ospf authentication
RB(config-if)#ip ospf authentication-key cisco
```

Bước 3.2 Cấu hình chứng thực dạng MD5 giữa 2 router: RA và RB

```
RA(config)#int S0/0/0
RA(config-if)#ip ospf authentication message-digest
RA(config-if)#ip ospf messages-digest-key 1 md5 cisco

RB(config)#int S0/0/1
RB(config-if)#ip ospf authentication message-digest
RB(config-if)#ip ospf messages-digest-key 1 md5 cisco
```

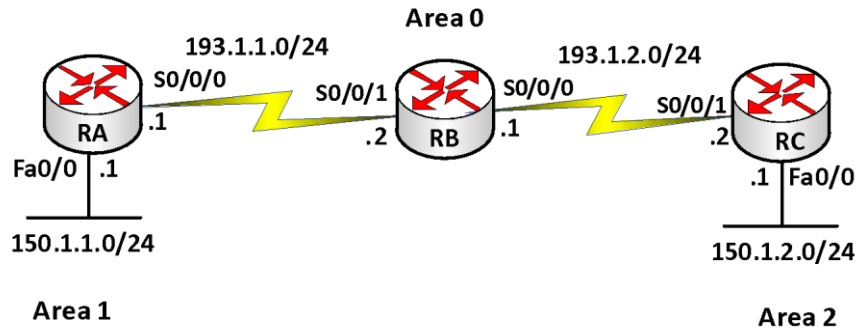
Bước 4. Kiểm tra cấu hình

Thực hiện các câu lệnh sau để kiểm tra cấu hình

show ip route: xem bảng định tuyến

debug ip ospf event: xem quá trình cập nhật định tuyến của OSPF

Ví dụ 2: Định tuyến động – OSPF



❖ Mô tả

- RA, RB, RC sử dụng OSPF để quảng bá thông tin định tuyến
- Các router cấu hình OSPF và quảng bá tất cả các mạng nối trực tiếp. Từ Router RA, RB và RC ta ping được hết các địa chỉ trong mạng.

❖ Các bước thực hiện

- Đặt hostname, địa chỉ IP cho các cổng trên router.
- Cấu hình giao thức định tuyến OSPF trên mỗi router

```

RA(config)#router ospf 1
RA(config-router)#network 150.1.1.0 0.0.0.255 area 1
RA(config-router)#network 193.1.1.0 0.0.0.255 area 0

RB(config)#router ospf 1
RB(config-router)#network 193.1.1.0 0.0.0.255 area 0
RB(config-router)#network 193.1.2.0 0.0.0.255 area 0

RC(config)#router ospf 1
RC(config-router)#network 150.1.2.0 0.0.0.255 area 2
RC(config-router)#network 193.1.2.0 0.0.0.255 area 0

```

❖ Kiểm tra cấu hình

Thực hiện các câu lệnh sau để kiểm tra cấu hình

Router#show ip route: xem bảng định tuyến

Router#ping: kiểm tra kết nối

3.3. EIGRP

EIGRP là giao thức định tuyến do Cisco tạo ra, chỉ hoạt động trên các thiết bị của Cisco. EIGRP là một giao thức định tuyến lai, nó vừa mang những đặc điểm của “*distance vector*” vừa mang một số đặc điểm của “*link-state*”. EIGRP là dạng định tuyến “*classless*”.

EIGRP hỗ trợ VLSM và CIDR nên sử dụng hiệu quả không gian địa chỉ, sử dụng địa chỉ multicast (224.0.0.10) để trao đổi thông tin cập nhật định tuyến.

❖ Cách tính metric của EIGRP

$$metric_{EIGRP} = \left[K1 * BW + \frac{K2 * BW}{(256 - load)} + K3 * Delay \right] * \frac{K5}{(reliability + K4)}$$

Với K1, K2, K3, K4, K5 là hằng số

Mặc định: K1=1, K2=0, K3=1, K4=0, K5=0

Do đó, ta có:

$$metric = bandwidth + delay$$

Những xử lý cơ bản của EIGRP trong việc học các mạng đích:

- Các router phát hiện các láng giềng của nó, danh sách các láng giềng được lưu giữ trong “*neighbor table*”.
- Mỗi router sẽ trao đổi các thông tin về cấu trúc mạng với các láng giềng của nó.
- Router đặt những thông tin về cấu trúc hệ thống mạng học được vào cơ sở dữ liệu về cấu trúc mạng (*topology table*).
- Router chạy thuật toán DUAL với cơ sở dữ liệu đã thu thập được ở bước trên để tính toán tìm ra đường đi tốt nhất đến mỗi một mạng trong cơ sở dữ liệu.
- Router đặt các đường đi tốt nhất đến mỗi mạng đích vào bảng định tuyến.
- Trong EIGRP có hai tuyến ta cần quan tâm là “*successor route*” và “*fessible successor route*”.
 - ✓ **Successor route**: là tuyến đường đi chính được sử dụng để chuyển dữ liệu đến đích, được lưu trong bảng định tuyến. EIGRP cho phép chia tải tối đa trên 16 đường (mặc định là 4 đường) đến mỗi mạng đích.
 - ✓ **Fessible successor route**: là đường đi dự phòng cho đường đi chính và được lưu trong bảng cấu trúc mạng (*topology table*).

❖ EIGRP chống “*routing loop*”

“*Routing loop*” là một trở ngại rất lớn trong các giao thức định tuyến dạng “*distance vector*”. Các giao thức định tuyến dạng “*link-state*” vượt qua vấn đề này bằng cách mỗi router đều nắm giữ toàn bộ cấu trúc mạng. Trong giao thức EIGRP, khi tuyến đường đi chính gặp sự cố, router có thể kịp thời đặt đường đi dự phòng vào bảng định tuyến đóng vai trò như đường đi chính.

Trường hợp không có đường đi dự phòng, EIGRP sử dụng thuật toán DUAL cho phép router gửi các yêu cầu và tính toán lại các đường đi đến đích.

❖ Cấu hình EIGRP

- Bước 1. Kích hoạt giao thức định tuyến EIGRP

```
Router(config)#router eigrp <autonomous-system>
```

Trong đó: *autonomous-system*: có giá trị từ 1 đến 65535, giá trị này phải giống nhau ở tất cả các router trong hệ thống chạy EIGRP

- Bước 2. Chọn công tham gia vào quá trình trao đổi thông tin định tuyến

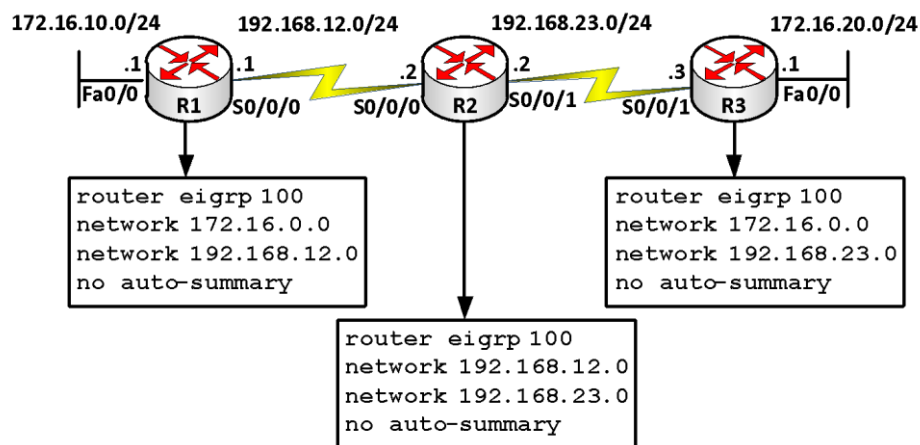
```
Router(config-router)#network <network-number>
```

Trong đó: *network-number* là địa chỉ công theo đúng lớp mạng của nó.

Để quảng bá các mạng con và hỗ trợ mạng không liên tục, chúng ta phải sử dụng lệnh sau:

```
Router(config-router)#no auto-summary
```

Ví dụ: Cấu hình định tuyến EIGRP cho mô hình mạng sau



❖ Các câu lệnh kiểm tra cấu hình EIGRP

```
Router#show ip eigrp neighbors
```

```
Router#show ip eigrp topology
```

```
Router#show ip route eigrp
```

```
Router#show ip protocols
```

```
Router#show ip eigrp traffic
```

❖ Chứng thực trong EIGRP

EIGRP chỉ hỗ trợ một dạng chứng thực là MD5.

Trên công của router gửi thông tin chứng thực cấu hình lệnh sau:

```
R(config)#key chain <keychain>
```

```
R(config-keychain)#key <key-id>
```

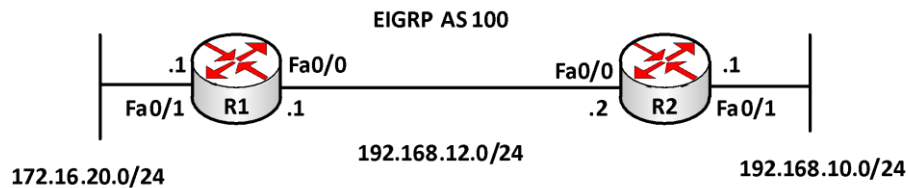
```
R(config-keychain-key)#key-string <password>
```

```
R(config)#interface <interface>
```

```
R(config-if)#ip authentication mode eigrp <AS> md5
```

```
R(config-if)#ip authentication key-chain eigrp <AS> <keychain>
```

Ví dụ: Cấu hình chứng thực cho giao thức định tuyến EIGRP giữa hai router R1 và R2.



- **Hướng dẫn cấu hình**

- Cấu hình cơ bản: hostname, địa chỉ IP cho các cổng trên các router.

- Cấu hình định tuyến EIGRP AS 100

```
R1(config)#router eigrp 100
R1(config-if)#network 192.168.12.0
R1(config-if)#network 172.16.0.0
R1(config-if)#no auto-summary
```

```
R2(config)#router eigrp 100
R2(config-if)#network 192.168.12.0
R2(config-if)#network 192.168.10.0
R2(config-if)#no auto-summary
```

- Cấu hình chứng thực

```
R1(config)#key chain my_keychain1
R1(config-keychain)#key 1
R1(config-keychain-key)#key-string cisco
```

```
R1(config)#interface fa0/0
R1(config-if)#ip authentication mode eigrp 100 md5
R1(config-if)#ip authentication key-chain eigrp 100 my_keychain1
```

```
R2(config)#key chain my_keychain2
R2(config-keychain)#key 1
R2(config-keychain-key)#key-string cisco
```

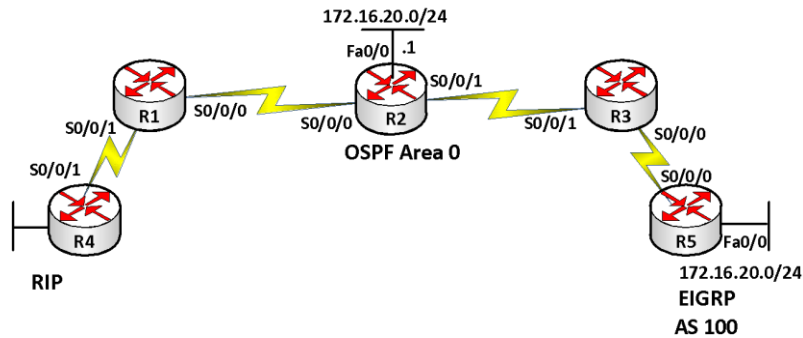
```
R2(config)#interface fa0/0
R2(config-if)#ip authentication mode eigrp 100 md5
R2(config-if)#ip authentication key-chain eigrp 100 my_keychain2
```

- **Kiểm tra cấu hình:** Dùng các lệnh sau

```
show ip eigrp neighbors
show ip eigrp interfaces details
show key chain
```

3.4. Redistribution giữa các giao thức định tuyến

Nếu một hệ thống mạng chạy nhiều hơn một giao thức định tuyến, người quản trị cần một vài phương thức để phân phối các đường đi của một giao thức này vào một giao thức khác. Quá trình đó gọi là phân phối giữa các giao thức định tuyến (*redistribution*).



Phân phối định tuyến định nghĩa cách thức trao đổi thông tin định tuyến giữa các giao thức định tuyến. Mỗi giao thức định tuyến có cách tính toán “metric” khác nhau, do đó khi thực hiện quá trình phân phối thì dạng ”metric” sẽ được chuyển đổi sao cho phù hợp với giao thức định tuyến đó để các giao thức đó có thể quảng bá các đường đi cho nhau.

- Phân phối định tuyến giữa RIP và OSPF
- Quảng bá các tuyến học được từ OSPF vào RIP

```
Router(config)#router rip
```

```
Router(config-router)#redistribute ospf 1 metric <number>
```

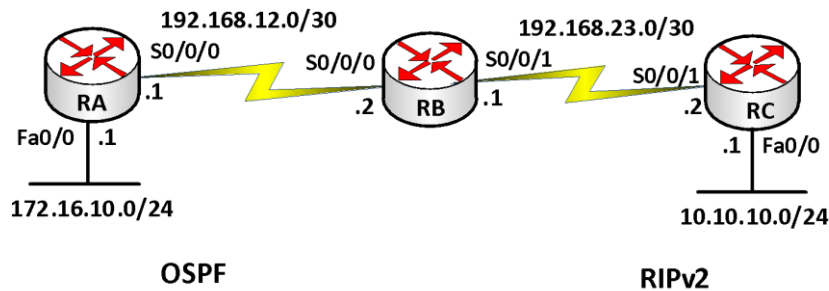
Lưu ý: Do RIP sử dụng *metric* có giá trị tối đa là 15 nên giá trị <number> trong lệnh trên cũng phải nhỏ hơn 15.

- Quảng bá các tuyến học được từ RIP vào OSPF

```
Router(config)#router ospf <process-id>
```

```
Router(config-router)#redistribute rip metric <metric> subnets
```

Ví dụ: Cho mô hình mạng sau



Mô tả yêu cầu:

- ✓ RA, RB sử dụng OSPF để quảng cáo thông tin định tuyến
- ✓ RB, RC sử dụng RIP để quảng cáo thông tin định tuyến
- ✓ Từ RA, RB, RC ping được hết các địa chỉ trong mạng

Các bước thực hiện:

- ✓ Đặt hostname, địa chỉ IP cho các cổng trên router.

✓ Cấu hình giao thức định tuyến OSPF trên mỗi RA và RB

```
RA(config)#router ospf 1
RA(config-router)#network 172.16.10.0 0.0.0.255 area 0
RA(config-router)#network 192.168.12.0 0.0.0.3 area 0
RB(config)#router ospf 1
RB(config-router)#network 192.168.12.0 0.0.0.3 area 0
RB(config)#router rip
RB(config-router)#version 2
RB(config-router)#network 192.168.23.0
RB(config-router)#no auto-summary
RC(config)#router rip
RC(config-router)#version 2
RC(config-router)#network 192.168.23.0
RC(config-router)#network 10.0.0.0
RC(config-router)#no auto-summary
```

✓ Cấu hình phân phối định tuyến

Để RC thấy được RA, ta thực hiện các lệnh sau:

```
RB(config)#router rip
RB(config-router)#redistribute ospf 1 metric 3
```

Tương tự: để RA thấy RC

```
RB(config)#router ospf 1
RB(config-router)#redistribute rip metric 100 subnets
```

✓ Kiểm tra cấu hình

Thực hiện các câu lệnh sau để kiểm tra cấu hình

```
Router#show ip route: xem bảng định tuyến
```

```
Router#ping: kiểm tra kết nối
```

● Phân phối định tuyến giữa RIP và EIGRP

- Quảng bá các tuyến học được từ EIGRP vào RIP

```
RB(config)#router rip
RB(config-router)#redistribute eigrp <AS> metric <number>
```

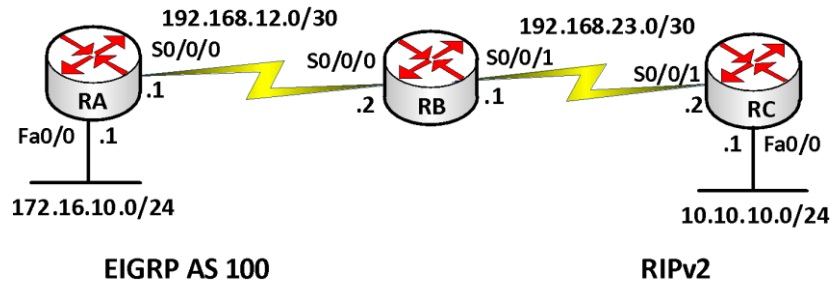
- Quảng bá các tuyến học được từ RIP vào EIGRP

```
RB(config)#router eigrp <AS>
RB(config-router)#redistribute rip metric BW DL L R MTU
```

Trong đó: BW, DL, L, R, MTU tương ứng với các thông số trong metric của EIGRP (trừ MTU). Tương tự, chúng ta có thể suy luận ra phương pháp để phân phối các tuyến học

được từ một giao thức này sang một giao thức khác là phải tuân theo thông số về *metric* của giao thức mà ta sẽ phân phối vào.

Ví dụ:



Mô tả

- ✓ RA, RB sử dụng EIGRP để quảng cáo thông tin định tuyến
- ✓ RB, RC sử dụng RIP để quảng cáo thông tin định tuyến
- ✓ Từ RA, RB, RC *ping* được hết các địa chỉ trong mạng

Các bước thực hiện

- ✓ Đặt Hostname, địa chỉ IP cho các cổng trên router.
- ✓ Cấu hình giao thức định tuyến EIGRP trên mỗi RA và RB

```

RA(config)#router eigrp 100
RA(config-router)#network 172.16.0.0
RA(config-router)#network 192.168.12.0
RA(config-router)#no auto-summary
RB(config)#router eigrp 100
RB(config-router)#network 192.168.12.0
RB(config-router)#no auto-summary
RB(config)#router rip
RB(config-router)#version 2
RB(config-router)#network 192.168.23.0
RB(config-router)#passive interface S0/0/0
RC(config)#router rip
RC(config-router)#version 2
RC(config-router)#network 10.0.0.0
RC(config-router)#network 192.168.23.0
  
```

Để RC thấy được RA, ta thực hiện các lệnh phân phối định tuyến:

```

RB(config)#router rip
RB(config-router)#redistribute eigrp 100 metric 3
  
```


Tương tự: để RA thấy RC

```
RB(config)#router eigrp 100
```

```
RB(config-router)#redistribute rip metric 100 1 255 255 1500
```

❖ Kiểm tra

Thực hiện các câu lệnh sau để kiểm tra cấu hình

```
show ip route: xem bảng định tuyến
```

```
ping: kiểm tra kết nối
```

4. DHCP

Dịch vụ DHCP cung cấp địa chỉ IP tự động cho các thiết bị trong mạng. Đây là một dịch vụ được sử dụng phổ biến đơn giản hóa trong việc cấu hình, quản lý địa chỉ trong mạng.

DHCP hoạt động theo dạng Client/Server. Máy chủ đóng vai trò DHCP server được cấu hình các tham số để cấp phát. Các tham số gồm: Tên, địa chỉ mạng, dãy địa chỉ cấp phát, default-gateway, địa chỉ DNS, thời gian người dùng sử dụng địa chỉ IP này.

❖ Nguyên tắc hoạt động:

Quá trình trao đổi thông tin giữa DHCP server và DHCP client trong trường hợp chúng nằm cùng miền quảng bá diễn ra như sau:

- o Bước 1: DHCP client gửi gói tin DHCPDISCOVER dạng broadcast đến DHCP Server
- o Bước 2: DHCP Server gửi lại gói DHCP OFFER dạng broadcast cho DHCP client
- o Bước 3: DHCP Client gửi gói DHCPREQUEST dạng broadcast cho DHCP server
- o Bước 4: DHCP Server gửi gói DHCPACK dạng broadcast cho DHCP client

Trong trường hợp DHCP server và DHCP client nằm khác miền quảng bá, thì cần thiết phải sử dụng một thiết bị trung gian để chuyển tiếp yêu cầu từ DHCP client đến DHCP server. Bởi vì, trong trường hợp này các gói tin (local) broadcast từ client bị router chặn nên sẽ không đến được DHCP server.

5. Tính sẵn sàng và dự phòng

● Load balancing

Kỹ thuật Load Balancing (cân bằng tải) là một kỹ thuật phân phối lưu lượng công việc qua hai hay nhiều máy tính, liên kết mạng, CPU, ổ cứng hoặc các tài nguyên tác để có thể sử dụng được tối ưu các tài nguyên hệ thống, giảm thiểu thời gian phản ứng, tránh quá tải, làm tăng độ tin cậy trong các hoạt động của hệ thống. Các thành phần được sử dụng trong kỹ thuật cân bằng tải có thể là phần cứng hay phần mềm

● High Availability (HA)

HA (độ sẵn sàng) trong hệ thống mạng là một kỹ thuật tạo ra tính năng dự phòng (backup) trong hoạt động truyền thông trên mạng.

Sự kết hợp của hai kỹ thuật HA và Load Balancing tạo nên một hệ thống mạng có tính dự phòng hệ thống trên cả các thiết bị phần cứng (Router, Switch, Firewall,...) hay thiết bị phần mềm (Server Clustering,...) đồng thời thực hiện chức năng chia tải, nâng cao hiệu quả trong việc sử dụng tài nguyên hệ thống và công việc quản trị mạng.

- **HSRP**

Mô hình HSRP gồm 2 hay nhiều Router. Trong đó một Router luôn ở trạng thái hoạt động gọi là Activer Router và các router dự phòng gọi là Standby Router.

Mỗi router sử dụng một độ ưu tiên (priority), mặc định là 100. Dựa vào tham số Priority, Router nào có chỉ số Priority cao nhất sẽ trở thành Active Router và những Router còn lại là Standby Router.

- **VRRP**

Cũng tương tự như HSRP, VRRP cũng sử dụng nhiều Router. Trong đó, có một Router được chọn làm Master và các Router còn lại gọi là Backup. Mô hình sử dụng nhiều nhóm VRRP sẽ hỗ trợ thêm tính năng tận dụng các Router chưa làm việc và cân bằng tải cho hệ thống.

- **GLBP**

Là giao thức của Cisco. Giống như HSRP, trong nhóm GLBP cũng bầu chọn một Active Router gọi là Active Virtual Gateway (AVG) và các router còn lại gọi là Backup Router cho AVG. AVG gán từng Virtual MAC address đến mỗi Router. Mỗi router sau khi được gán địa chỉ MAC sẽ đảm nhận nhiệm vụ định hướng gói tin, được gọi là Active Virtual Forwarder (AVF)

6. Tổng kết chương

Trong chương này đề cập đến một số vấn đề cơ bản và định tuyến. Định tuyến là quá trình tìm đường đi cho các gói tin từ nơi gửi đến nơi nhận. Quá trình định tuyến được phân làm 2 loại là định tuyến tĩnh và định tuyến động.

Định tuyến tĩnh là quá trình định tuyến sử dụng các tuyến được cấu hình thủ công bởi người quản trị mạng, còn định tuyến động là quá trình định tuyến được thực hiện bằng các giao thức định tuyến động.

Trong định tuyến động người ta phân ra làm 2 dạng: *distance-vector* và *link-state*. Ngoài ra, định tuyến còn được chia thành 2 kiểu là *classful* và *classless*.

RIPv1 là giao thức định tuyến động theo dạng *distance-vector* và là giao thức định tuyến theo kiểu *classful*, có nghĩa là không mang theo thông tin *subnet-mask* trong các thông tin cập nhật định tuyến và mỗi router nhìn hệ thống mạng theo sự chi phối của các router láng giềng. RIPv2 là một giao thức định tuyến cải tiến từ RIPv1, mang các đặc điểm của loại giao thức *distance-vector*. Tuy nhiên, RIPv2 thuộc nhóm giao thức *classless*. Do đó, nó còn mang một số tính chất như hỗ trợ VLSM và mạng không liên tục.

OSPF là một giao thức định tuyến loại *link-state*, thuộc nhóm giao thức *classless*. OSPF được sử dụng trên các mạng lớn, phức tạp. EIGRP là một giao thức lai giữa *distance-vector* và *link-state*. EIGRP thuộc nhóm giao thức *classless*.

7. Câu hỏi và bài tập

7.1. Các câu nào sau đây nói về đặc điểm của giao thức định tuyến loại "distance vector" và "link-state"?

- A. Các giao thức định tuyến loại "distance vector" gửi toàn bộ bảng định tuyến cho các router láng giềng có kết nối trực tiếp với nó.
- B. Các giao thức định tuyến loại "distance vector" gửi bản cập nhật thay đổi đến các mạng được liệt kê trong bảng định tuyến
- C. Các giao thức định tuyến loại "link-state" gửi toàn bộ bảng định tuyến cho các router khác trong mạng
- D. Các giao thức định tuyến loại "link-state" gửi các cập nhật về trạng thái kết nối cho các router khác

7.2. Giao thức định tuyến nào mặc định sử dụng hai tham số "bandwidth" và "delay" làm metric?

- A. RIP
- B. OSPF
- C. EIGRP
- D. Định tuyến tĩnh

7.3. Các câu nào sau đây là đúng cho các giao thức định tuyến thuộc loại "classless"?

- A. Không chạy được trong mạng không liên tục (discontiguous network)
- B. Hỗ trợ mạng VLSM
- C. RIPv1 là giao thức thuộc "classless"
- D. RIPv2 là giao thức thuộc loại "classless"

7.4. "Default route" dùng để làm gì?

- A. Nó được sử dụng khi các giao thức định tuyến không sử dụng được.
- B. Nó được nhà cung cấp dịch vụ (ISP) cấu hình để gửi dữ liệu cho các đối tác qua mạng.
- C. Nó được sử dụng khi gói tin gửi tới mạng đích không có trong bảng định tuyến.
- D. Nó được cấu hình thủ công đến các mạng khác mà các giao thức định tuyến chưa cấu hình.
Nó được dùng để gửi các gói tin đến các "stub network".

7.5. Các câu nào sau đây mô tả đúng về giao thức định tuyến RIP?

- A. RIPv1 không hỗ trợ chứng thực trong cập nhật thông tin định tuyến. RIPv2 hỗ trợ chứng thực trong cập nhật thông tin định tuyến.
- B. RIPv1 không hỗ trợ quảng bá các đường đi qua mạng WAN. RIPv2 hỗ trợ quảng bá các đường đi qua LAN và WAN.
- C. RIPv1 không gửi kèm thông tin "subnet-mask" trong bản tin cập nhật định tuyến. RIPv2 gửi kèm thông tin "subnet-mask" trong bản tin cập nhật định tuyến.
- D. RIPv1 định thời gửi thông tin cập nhật định tuyến qua địa chỉ IP multicast:224.0.0.10. RIPv2 định thời gửi thông tin cập nhật định tuyến qua địa chỉ IP multicast: 224.0.0.9.

- E. RIPv1 sử dụng "hold-down timer" và "split horizon" để chống tình trạng "routing loop". RIPv2 không yêu cầu "hold-down timer" hay "split horizon" để chống "routing loop".
- 7.6. Một router học được đường đi đến một mạng đích 131.107.4.0/24 bằng RIP và OSPF. Bạn cũng cấu hình thêm bằng định tuyến tĩnh trên router này đến mạng 131.107.4.0/24. Router sẽ chọn đường đi như thế nào để chuyển dữ liệu đi?
- A. Đường đi học được từ RIP
 - B. Đường đi học được từ OSPF
 - C. Đường đi học được từ định tuyến tĩnh
 - D. Chia tải đi trên cả 3 đường này
- 7.7. Giá trị AD (Administrative Distance) mặc định của OSPF là
- A. 1
 - B. 90
 - C. 110
 - D. 120
- 7.8. Các đặc điểm nào dưới đây là của các giao thức định tuyến thuộc loại *link-state*?
- A. Cung cấp thông tin toàn diện về hệ thống mạng
 - B. Trao đổi bảng định tuyến với các láng giềng.
 - C. Tính toán đường đi ngắn nhất
 - D. Có tính năng "trigger update"
 - E. Có tính năng cập nhật định thời
- 7.9. Router R1 đang chạy giao thức định tuyến RIP, thông tin bảng định tuyến được hiển thị như sau:

```
Gateway of last resort is 10.1.2.2 to network 0.0.0.0
  10.0.0.0/24 is subnetted, 2 subnets
R    10.1.3.0 [120/1] via 10.1.2.2, 00:00:00, Serial0/0
C    10.1.2.0 is directly connected, Serial0/0
C    10.1.5.0 is directly connected, Serial0/1
C    10.1.6.0 is directly connected, FastEthernet0/0
R*   0.0.0.0/0 [120/1] via 10.1.5.5, 00:00:00, Serial0/1
```

- Dựa vào thông tin trên, nếu ta thực hiện lệnh *ping* đến địa chỉ 10.1.8.5 từ máy tính có địa chỉ 10.1.6.100, thì router R1 xử lý các gói tin ICMP này như thế nào?
- A. Các gói tin sẽ bị loại bỏ
 - B. Các gói tin sẽ được chuyển ra cổng S0/0
 - C. Các gói tin sẽ được chuyển ra cổng S0/1
 - D. Các gói tin sẽ được chuyển ra cổng Fa0/0

E. Các gói tin sẽ được chuyển đến gateway 10.1.2.2

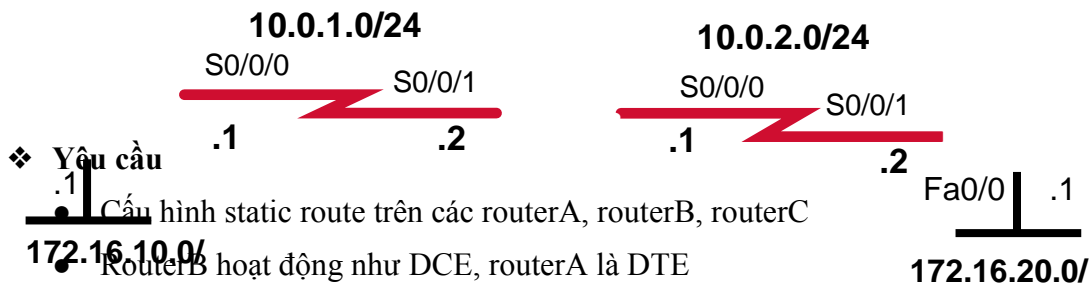
7.10. Giao thức định tuyến OSPF dùng khái niệm khu vực (area). Đặc điểm của OSPF area là gì?

- A. Mỗi OSPF area cần cấu hình cổng loopback
- B. Các area có thể được gán giá trị từ 0 đến 65535
- C. Area 0 còn gọi là area backbone
- D. Kiến trúc phân cấp OSPF không yêu cầu nhiều area
- E. Các area phải kết nối về area 0
- F. OSPF đơn vùng phải cấu hình là area 1

8. Lab: Định tuyến

Lab 3-1

STATIC ROUTING



- Từ các router, ta phải có thể ping được tất cả các địa chỉ trong mạng.

❖ Cấu hình

Bước 1: Cấu hình cơ bản (cấu hình hostname, địa chỉ IP cho các interface, ...)

• Cấu hình routerA

```
Router(config)#hostname routerA
routerA(config)#interface serial 0/0/0
routerA(config-if)#ip address 10.0.1.1 255.255.255.0
routerA(config-if)#no shutdown
routerA(config-if)#exit
routerA(config)#
```

• Cấu hình routerB

```
Router(config)#hostname routerB
routerB(config)#interface serial 0/0/0
routerB(config-if)#ip address 10.0.2.1 255.255.255.0
routerB(config-if)#no shutdown

routerB(config-if)#interface serial 0/0/1
```

```

routerB(config-if)#ip address 10.0.1.2 255.255.255.0
routerB(config-if)#clock rate 64000
routerB(config-if)#no shutdown
routerB(config-if)#exit
routerB(config)#

```

- **Cấu hình routerC**

```

Router>enable
Router#config terminal
Router(config)#hostname routerC
routerC(config)#interface S0/0/1
routerC(config-if)#ip address 10.0.2.2 255.255.255.0
routerC(config-if)#no shutdown
routerC(config-if)#exit

```

- **Kiểm tra cấu hình**

Sử dụng lệnh **ping** để kiểm tra cấu hình

- Kiểm tra kết quả ping giữa routerA với routerB
- Kiểm tra kết quả ping giữa routerB với routerA, routerC
- Kiểm tra kết quả ping giữa routerC với routerA, routerB

Bước 2: Cấu hình static route

- **RouterA**

```

RouterA(config)#ip route 10.0.2.0 255.255.255.0 10.0.1.2
RouterA(config)#ip route 172.16.20.0 255.255.255.0 10.0.1.2

```

- **Router B**

```

RouterB(config)#ip route 172.16.10.0 255.255.255.0 10.0.1.1
RouterB(config)#ip route 172.16.20.0 255.255.255.0 10.0.2.2

```

- **RouterC**

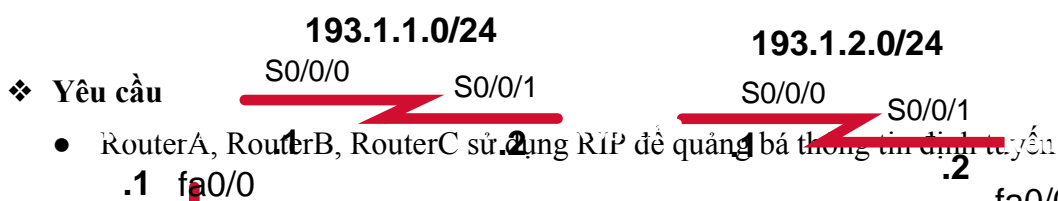
```

RouterC(config)#ip route 10.0.1.0 255.255.255.0 10.0.2.1
RouterC(config)#ip route 172.16.10.0 255.255.255.0 10.0.2.1

```

Lab 3-2

DYNAMIC ROUTING – RIP



- Router B hoạt động như DCE cung cấp xung clock cho RouterA, RouterC
- Các router cấu hình RIP và quảng bá tất cả các mạng nội trực tiếp. Từ router A, B và C ta ping được hết các địa chỉ trong mạng.

❖ Cấu hình

Bước 1: Cấu hình cơ bản (đặt hostname, địa chỉ IP cho các cổng loopback, serial, fastethernet, ...)

- **Đối với router A**

```
Router>enable
Router#config terminal
Router(config)#hostname RouterA
RouterA(config)#interface fa0/0
RouterA(config-if)#ip address 152.1.1.1 255.255.255.0
RouterA(config-if)#no shutdown
RouterA(Config-if)#exit

RouterA(config)#interface Serial 0/0/0
RouterA(config-if)#ip address 193.1.1.1 255.255.255.0
RouterA(config-if)#clock rate 64000
RouterA(config-if)#no shutdown
RouterA(config-if)#exit
```

- **Đối với router B**

```
Router>enable
Router#config terminal
Router(config)#hostname RouterB
RouterB(config)#interface S0/0/1
RouterB(config-if)#ip address 193.1.1.2 255.255.255.0
RouterB(Config-if)#no shut
RouterB(Config-if)#exit

RouterB(config)#int S0/0/0
RouterB(config-if)#ip address 193.1.2.1 255.255.255.0
RouterB(config-if)#clock rate 64000
RouterB(config-if)#no shutdown
RouterB(config-if)#exit
```

- **Đối với router C**

```
Router>enable
Router#config terminal
Router(config)#hostname RouterC
```

```

RouterC(config)#interface fa0/0
RouterC(config-if)#ip address 148.1.1.1 255.255.255.0
RouterC(config-if)#no shutdown
RouterC(Config-if)#exit

RouterC(config)#interface s0/0/1
RouterC(config-if)#ip address 193.1.2.2 255.255.255.0
RouterC(config-if)#no shutdown
RouterC(config-if)#exit

```

Bước 2: Cấu hình giao thức định tuyến RIP trên mỗi router

```

routerA(config)#router rip
routerA(config-router)#network 152.1.0.0
routerA(config-router)#network 193.1.1.0

routerB(config)#router rip
routerB(config-router)#network 193.1.1.0
routerB(config-router)#network 193.1.2.0

RouterC(config)#router rip
RouterC(config-router)#network 148.1.0.0
RouterC(config-router)#network 193.1.2.0

```

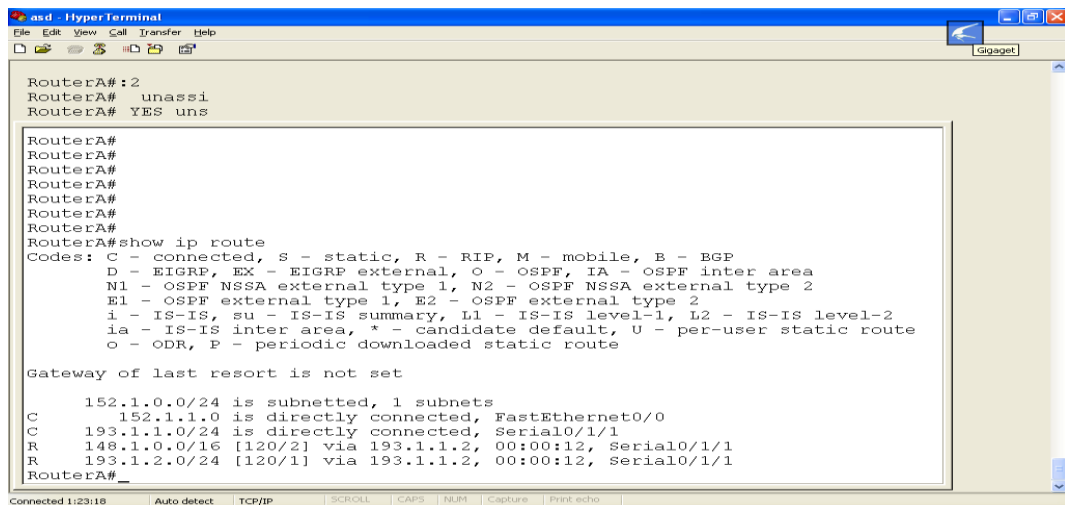
❖ Kiểm tra:

Thực hiện các câu lệnh sau để kiểm tra cấu hình

Router#show ip route : xem bảng định tuyến

Router#debug ip rip : xem quá trình cập nhật định tuyến của RIP

Router#undebug all : dừng quá trình debug



```

RouterA# :2
RouterA# unassi
RouterA# YES uns

RouterA#
RouterA#
RouterA#
RouterA#
RouterA#
RouterA#
RouterA#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

152.1.0.0/24 is subnetted, 1 subnets
C       152.1.1.0 is directly connected, FastEthernet0/0
C       193.1.1.0/24 is directly connected, Serial0/1/1
R       148.1.0.0/16 [120/2] via 193.1.1.2, 00:00:12, Serial0/1/1
R       193.1.2.0/24 [120/1] via 193.1.1.2, 00:00:12, Serial0/1/1
RouterA#_

```



```

asd - HyperTerminal
File Edit View Call Transfer Help
RouterB#0 keepal
RouterB#rmatio
R
RouterB#n329 32

RouterB#
RouterB#
RouterB#
RouterB#
RouterB#
RouterB#
RouterB#
RouterB#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

R    152.1.0.0/16 [120/1] via 193.1.1.1, 00:00:01, Serial0/0/0
C    193.1.1.0/24 is directly connected, Serial0/0/0
R    148.1.0.0/16 [120/1] via 193.1.2.2, 00:00:26, Serial0/0/1
C    193.1.2.0/24 is directly connected, Serial0/0/1
RouterB#_

```

```

asd - HyperTerminal
File Edit View Call Transfer Help
RouterC# dow
RouterC#d -
RouterC#5-3 (192
RouterC#0, 2079)

RouterC#
RouterC#
RouterC#
RouterC#
RouterC#
RouterC#
RouterC#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

R    152.1.0.0/16 [120/2] via 193.1.2.1, 00:00:25, Serial0/0/0
R    193.1.1.0/24 [120/1] via 193.1.2.1, 00:00:25, Serial0/0/0
C    148.1.0.0/24 is subnetted, 1 subnets
C    148.1.1.0 is directly connected, FastEthernet0/0
C    193.1.2.0/24 is directly connected, Serial0/0/0
RouterC#

```

```

RouterA#
Building configuration...
Current configuration : 1426 bytes
!
hostname RouterA
!
interface FastEthernet0/0
 ip address 152.1.1.1 255.255.255.0
 duplex auto
 speed auto

```

```
!  
interface Serial0/1/1  
  ip address 193.1.1.1 255.255.255.0  
!  
!  
router rip  
  network 152.1.0.0  
  network 193.1.1.0  
!  
ip classless  
!  
scheduler allocate 20000 1000  
!  
end
```

```
RouterB#  
Building configuration...  
interface Serial0/0/0  
  ip address 193.1.1.2 255.255.255.0  
  clock rate 64000  
!  
interface Serial0/0/1  
  ip address 193.1.2.1 255.255.255.0  
  --More--          !  
router rip  
  network 193.1.1.0  
  network 193.1.2.0  
!  
ip http server  
no ip http secure-server  
!  
control-plane  
!  
!  
  scheduler allocate 20000 1000  
End  
RouterC#  
Building configuration...  
Current configuration : 778 bytes  
!  
interface FastEthernet0/0  
  ip address 148.1.1.1 255.255.255.0
```

```

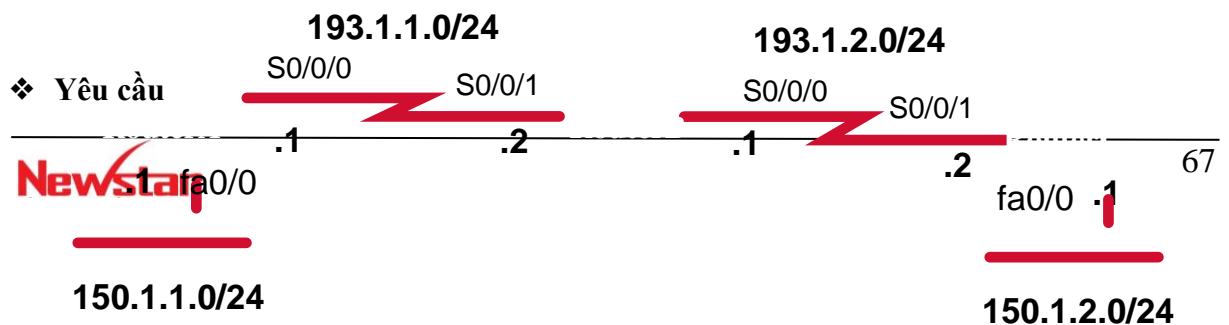
duplex auto
speed auto

!
interface Serial0/0/0
 ip address 193.1.2.2 255.255.255.0
 clock rate 64000
!
!
router rip
 network 148.1.0.0
 network 193.1.2.0
!
ip http server
no ip http secure-server
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 password cisco
 login
!
scheduler allocate 20000 1000
end
RouterC#

```

Lab 3-3.

DYNAMIC ROUTING - RIPv2



- RouterA, RouterB, RouterC sử dụng RIPv2 để quảng bá thông tin định tuyến
- Các router cấu hình RIPv2 và quảng bá tất cả các mạng nối trực tiếp. Từ router A, B và C ta ping được tất cả các địa chỉ trong mạng.

❖ Cấu hình

Bước 1: Cấu hình cơ bản (đặt hostname, địa chỉ IP cho các cổng loopback, serial, FastEthernet, ...)

- Đối với router A

```
Router>enable
Router#config terminal
Router(config)#hostname routerA
routerA(config)#int f0/0
routerA(config-if)#ip address 150.1.1.1 255.255.255.0
routerA(config-if)#no shutdown
routerA(Config-if)#exit

routerA(config)#int s0/0/0
routerA(config-if)#ip address 193.1.1.1 255.255.255.0
routerA(config-if)#clock rate 64000
routerA(config-if)#no shutdown
routerA(config-if)#exit
```

- Đối với router B

```
Router>enable
Router#configure terminal
Router(config)#hostname routerB
routerB(config)#interface serial 0/0/1
routerB(config-if)#ip address 193.1.1.2 255.255.255.0
routerB(Config-if)#no shutdown
routerB(Config-if)#exit

routerB(config)#interface serial 0/0/0
routerB(config-if)#ip address 193.1.2.1 255.255.255.0
routerB(config-if)#clock rate 64000
routerB(config-if)#no shutdown
routerB(config-if)#exit
```

- Đối với router C

```
Router>enable
```

```
Router#configure terminal
Router(config)#hostname RouterC
RouterC(config)#interface fastEthernet 0/0
RouterC(config-if)#ip address 150.1.2.1 255.255.255.0
RouterC(config-if)#no shutdown
RouterC(Config-if)#exit

RouterC(config)#int s0/0/1
RouterC(config-if)#ip address 193.1.2.2 255.255.255.0
RouterC(config-if)#no shutdown
RouterC(config-if)#exit
```

Bước 2: Cấu hình giao thức định tuyến RIP trên mỗi router

```
routerA(config)#router rip
routerA(config-router)#version 2
routerA(config-router)#network 150.1.0.0
routerA(config-router)#network 193.1.1.0
routerA(config-router)#no auto-summary

routerB(config)#router rip
routerB(config-router)#version 2
routerB(config-router)#network 193.1.1.0
routerB(config-router)#network 193.1.2.0
routerB(config-router)#no auto-summary

RouterC(config)#router rip
RouterC(config-router)#version 2
RouterC(config-router)#network 150.1.0.0
RouterC(config-router)#network 193.1.2.0
RouterC(config-router)#no auto-summary
```

❖ Kiểm tra cấu hình

Thực hiện các câu lệnh sau để kiểm tra cấu hình

```
show ip route : xem bảng định tuyến
debug ip rip : xem quá trình cập nhật định tuyến của RIP
undebug all : dừng quá trình debug
```

```

asd - HyperTerminal
File Edit View Call Transfer Help
RouterA#clear ip rout *P Echos to 193.1.1.2, t
RouterA#

Serial

*Sep  6 05:39:29.003: RIP: sending request on FastEthernet0/0 to 224.0.0.9
*Sep  6 05:39:29.003: RIP: sending request on Serial0/1/1 to 224.0.0.9
*Sep  6 05:39:29.019: RIP: received v2 update from 193.1.1.2 on Serial0/1/1
*Sep  6 05:39:29.019:   150.1.2.0/24 via 0.0.0.0 in 2 hops
*Sep  6 05:39:29.019:   193.1.2.0/24 via 0.0.0.0 in 1 hops
*Sep  6 05:39:29.031: RIP: received v2 update from 193.1.1.2 on Serial0/1/1
*Sep  6 05:39:29.031:   150.1.2.0/24 via 0.0.0.0 in 2 hops
*Sep  6 05:39:29.031:   193.1.2.0/24 via 0.0.0.0 in 1 hops
*Sep  6 05:39:29.039: RIP: received v2 update from 193.1.1.2 on Serial0/1/1
*Sep  6 05:39:29.039:   150.1.2.0/24 via 0.0.0.0 in 2 hops
*Sep  6 05:39:29.039:   193.1.2.0/24 via 0.0.0.0 in 1 hops
*Sep  6 05:39:30.267: RIP: received v2 update from 193.1.1.2 on Serial0/1/1
*Sep  6 05:39:30.267:   150.1.2.0/24 via 0.0.0.0 in 2 hops
*Sep  6 05:39:30.267:   193.1.2.0/24 via 0.0.0.0 in 1 hops
*Sep  6 05:39:31.003: RIP: sending v2 flash update to 224.0.0.9 via FastEthernet
0/0 (150.1.1.1)
*Sep  6 05:39:31.003: RIP: build flash update entries
*Sep  6 05:39:31.003:   150.1.2.0/24 via 0.0.0.0, metric 3, tag 0
*Sep  6 05:39:31.003:   193.1.1.0/24 via 0.0.0.0, metric 1, tag 0
*Sep  6 05:39:31.003:   193.1.2.0/24 via 0.0.0.0, metric 2, tag 0
*Sep  6 05:39:31.003: RIP: sending v2 flash update to 224.0.0.9 via Serial0/1/1
(193.1.1.1)
*Sep  6 05:39:31.003: RIP: build flash update entries
*Sep  6 05:39:31.003:   150.1.1.0/24 via 0.0.0.0, metric 1, tag 0

Connected 1:51:33      Auto detect      TCP/IP      SCROLL      CAPS      NUM      Capture      Print echo

```

```

asd - HyperTerminal
File Edit View Call Transfer Help
RouterA#ely down
RouterA#L/Z.Rout
RouterA#

RouterA#
RouterA#
RouterA#
RouterA#
RouterA#
RouterA#
RouterA#
RouterA#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    193.1.1.0/24 is directly connected, Serial0/1/1
R    193.1.2.0/24 [120/1] via 193.1.1.2, 00:00:12, Serial0/1/1
     150.1.0.0/24 is subnetted, 2 subnets
R      150.1.2.0 [120/2] via 193.1.1.2, 00:00:12, Serial0/1/1
C    150.1.1.0 is directly connected, FastEthernet0/0
RouterA#_

Connected 1:52:13      Auto detect      TCP/IP      SCROLL      CAPS      NUM      Capture      Print echo

```

```

asd - HyperTerminal
File Edit View Call Transfer Help
Enter co
ii
lr
ii

% Unknown command or computer name, or unable to find computer address
RouterB#
RouterB#
RouterB#
RouterB#
RouterB#
RouterB#
RouterB#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    193.1.1.0/24 is directly connected, Serial0/0/0
C    193.1.2.0/24 is directly connected, Serial0/0/1
C    150.1.0.0/24 is subnetted, 2 subnets
R    150.1.2.0 [120/1] via 193.1.2.2, 00:00:03, Serial0/0/1
R    150.1.1.0 [120/1] via 193.1.1.1, 00:00:03, Serial0/0/0
RouterB#_
    
```

```

asd - HyperTerminal
File Edit View Call Transfer Help
Enter co
ii
lr
ii

% Unknown command or computer name, or unable to find computer address
RouterB#
RouterB#
RouterB#
RouterB#
RouterB#
RouterB#
RouterB#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    193.1.1.0/24 is directly connected, Serial0/0/0
C    193.1.2.0/24 is directly connected, Serial0/0/1
C    150.1.0.0/24 is subnetted, 2 subnets
R    150.1.2.0 [120/1] via 193.1.2.2, 00:00:03, Serial0/0/1
R    150.1.1.0 [120/1] via 193.1.1.1, 00:00:03, Serial0/0/0
RouterB#_
    
```

```

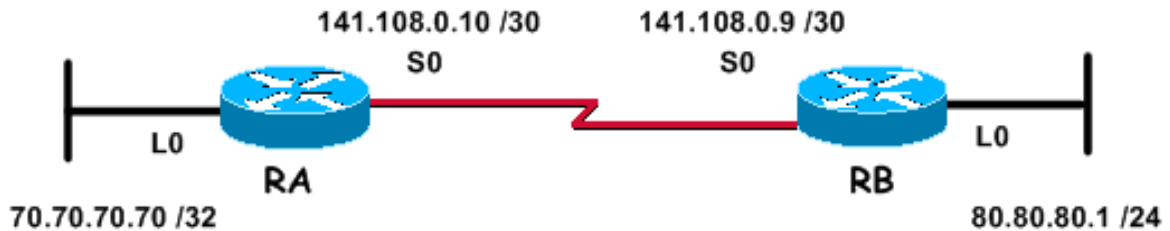
asd - HyperTerminal
File Edit View Call Transfer Help
RouterC#S agents
RouterC#    E1
RouterC#xternal
RouterC#2 - OSPF

RouterC#
RouterC#
RouterC#
RouterC#
RouterC#
RouterC#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

R    193.1.1.0/24 [120/1] via 193.1.2.1, 00:00:20, Serial0/0/0
C    193.1.2.0/24 is directly connected, Serial0/0/0
C    150.1.0.0/16 is variably subnetted, 3 subnets, 2 masks
C    150.1.2.0/24 is directly connected, FastEthernet0/0
R    150.1.1.0/24 [120/2] via 193.1.2.1, 00:00:20, Serial0/0/0
R    150.1.0.0/16 [120/2] via 193.1.2.1, 00:02:30, Serial0/0/0
RouterC#_
    
```

Lab 3-4. RIPv2 Plain Text Authentication



❖ Yêu cầu

Cấu hình chứng thực cho RIPv2 dạng Plain Text

❖ Các bước thực hiện

Bước 1: Cấu hình Hostname, IP cho các interface theo sơ đồ mạng

Bước 2: Cấu hình Routing RIPv2

```

RA(config)#router rip
RA(config-router)#version 2
RA(config-router)#network 70.0.0.0
RA(config-router)#network 142.108.0.0
RA(config-router)#no auto-summary

RB(config)#router rip
RB(config-router)#version 2
RB(config-router)#network 80.0.0.0
RB(config-router)#network 142.108.0.0
RB(config-router)#no auto-summary

```

Bước 3: Cấu hình Plain Text Authentication

```

RA(config)#key chain newstar
RA(config-keychain)#key 1
RA(config-keychain-key)#key-string ccna
RA(config)#interface S0
RA(config-if)#ip rip authentication key-chain newstar

RB(config)#key chain newstar2
RB(config-keychain)#key 1

```

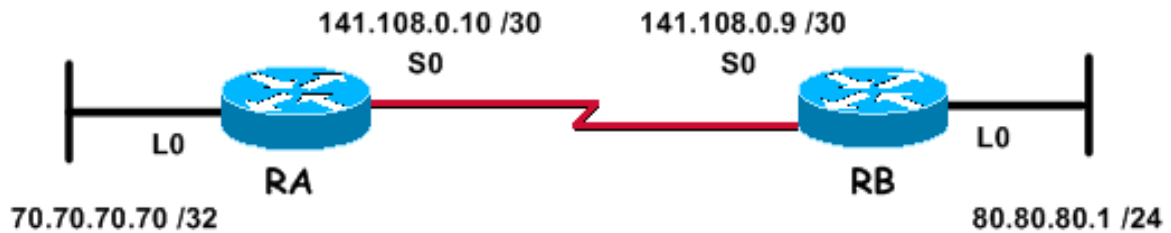


```

RB(config-keychain-key)#key-string ccna
RB(config)#interface S0
RB(config-if)#ip rip authentication key-chain newstar2

```

Lab 3-5. RIPv2 MD5 Authentication



❖ Yêu cầu

Cấu hình RIPv2 MD5 authentication

❖ Các bước thực hiện

Bước 1: Cấu hình Hostname, IP

Bước 2: Cấu hình Routing RIPv2

Bước 3: Cấu hình **MD5 Authentication**

```

RA(config)#key chain newstar
RA(config-keychain)#key 1
RA(config-keychain-key)#key-string ccna
RA(config)#interface S0
RA(config-if)#ip rip authentication mode md5
RA(config-if)#ip rip authentication key-chain newstar

```

```

RB(config)#key chain newstar2
RB(config-keychain)#key 1
RB(config-keychain-key)#key-string ccna
RB(config)#interface S0
RB(config-if)#ip rip authentication mode md5
RB(config-if)#ip rip authentication key-chain newstar2

```

❖ Các lệnh kiểm tra cấu hình

```

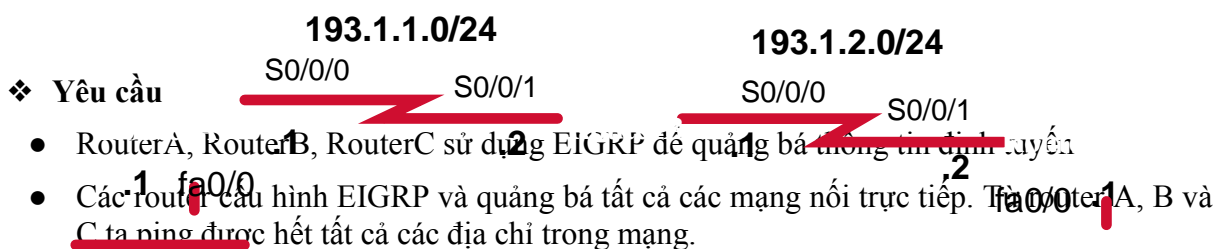
R#debug ip rip
R#show ip route

```

- ❖ **Lưu ý:** khi thực hiện kiểm tra cấu hình nên thử lại với trường hợp hai bên chứng thực không khớp thông tin với nhau và kiểm tra lại kết quả hiển thị qua các lệnh trên.

Lab 3-6

DYNAMIC ROUTING – EIGRP

- Yêu cầu
- 
- Router A, Router B, Router C sử dụng EIGRP để quảng bá thông tin định tuyến
 - Các router cấu hình EIGRP và quảng bá tất cả các mạng nối trực tiếp. Router A, B và C ta ping được hết tất cả các địa chỉ trong mạng.

❖ Các bước thực hiện

Bước 1: Cấu hình cơ bản (đặt hostname địa chỉ IP cho các cổng loopback, serial, fastEthernet, ...)

• Đối với router A

```

Router>enable
Router#config terminal
Router(config)#hostname routerA
routerA(config)#interface fa0/0
routerA(config-if)#ip address 150.1.1.1 255.255.255.0
routerA(config-if)#no shutdown
routerA(Config-if)#exit

routerA(config)#interface S0/0/0
routerA(config-if)#ip address 193.1.1.1 255.255.255.0
routerA(config-if)#clock rate 64000
routerA(config-if)#no shutdown
routerA(config-if)#exit
  
```

• Đối với router B

```

Router>enable
Router#config terminal
Router(config)#hostname routerB
  
```

```
routerB(config)#interface S0/0/1
routerB(config-if)#ip address 193.1.1.2 255.255.255.0
routerB(Config-if)#no shut
routerB(Config-if)#exit

routerB(config)#interface S0/0/0
routerB(config-if)#ip address 193.1.2.1 255.255.255.0
routerB(config-if)#clock rate 64000
routerB(config-if)#no shutdown
routerB(config-if)#exit
```

- **Đối với router C**

```
Router>enable
Router#config terminal
Router(config)#hostname RouterC
RouterC(config)#interface fastethernet 0/0
RouterC(config-if)#ip address 150.1.2.1 255.255.255.0
RouterC(config-if)#no shutdown
RouterC(Config-if)#exit

RouterC(config)#interface S0/0/1
RouterC(config-if)#ip address 193.1.2.2 255.255.255.0
RouterC(config-if)#no shutdown
RouterC(config-if)#exit
```

Bước 2: Cấu hình giao thức định tuyến EIGRP trên mỗi router

```
RouterA(config)#router eigrp 10
RouterA(config-router)#network 150.1.0.0
RouterA(config-router)#network 193.1.1.0
RouterA(config-router)#no auto-summary

routerB(config)#router eigrp 10
routerB(config-router)#network 193.1.1.0
routerB(config-router)#network 193.1.2.0
routerB(config-router)# no auto-summary

RouterC(config)#router eigrp 10
RouterC(config-router)#network 150.1.0.0
RouterC(config-router)#network 193.1.2.0
```

```
RouterC(config-router)#no auto-summary
```

❖ Kiểm tra cấu hình

Thực hiện các câu lệnh sau để kiểm tra cấu hình

Router#show ip route : xem bảng định tuyến

```

asd - HyperTerminal
File Edit View Call Transfer Help
Rout
RouterA#5-2
RouterA# 192.16
RouterA#a - IS-1

RouterA#
RouterA#
RouterA#
RouterA#
RouterA#
RouterA#
RouterA#sh ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    193.1.1.0/24 is directly connected, Serial0/1/1
D    193.1.2.0/24 [90/21024000] via 193.1.1.2, 00:01:02, Serial0/1/1
     150.1.0.0/24 is subnetted, 2 subnets
D    150.1.2.0 [90/21026560] via 193.1.1.2, 00:01:02, Serial0/1/1
C    150.1.1.0 is directly connected, FastEthernet0/0
RouterA#
  
```

```

asd - HyperTerminal
File Edit View Call Transfer Help
Gateway of last resort is not setived v2 update from 193.1.2.2 on
C    193.1.1.0/24 is directly connected, Serial0/1/1nistratively down downace Fa
ial0/0/0) is resync: peer graceful-restart
RouterB#
RouterB#
RouterB#
RouterB#
RouterB#
RouterB#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

C    193.1.1.0/24 is directly connected, Serial0/0/0
C    193.1.2.0/24 is directly connected, Serial0/0/1
     150.1.0.0/24 is subnetted, 2 subnets
D    150.1.2.0 [90/20514560] via 193.1.2.2, 00:03:08, Serial0/0/1
D    150.1.1.0 [90/20514560] via 193.1.1.1, 00:03:22, Serial0/0/0
RouterB#
  
```

```

asd - HyperTerminal
File Edit View Call Transfer Help
ia - IS-IS inter area, * - candidate default, U - per-user static route
!
!m
!n

RouterB#0
Termserver#3
[Resuming connection 3 to r5-3 ... ]

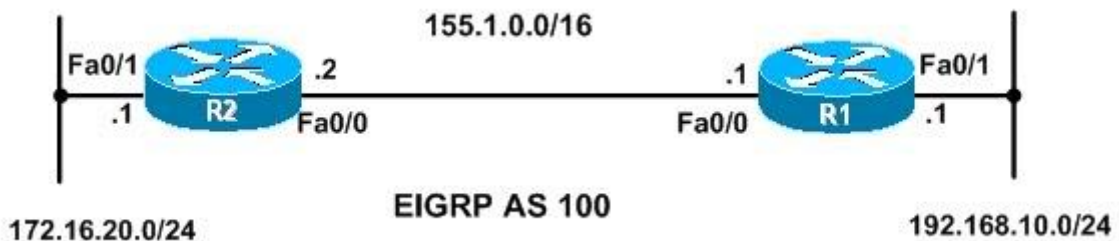
RouterC#
RouterC#
RouterC#
RouterC#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

D   193.1.1.0/24 [90/21024000] via 193.1.2.1, 00:05:12, Serial0/0/0
C   193.1.2.0/24 is directly connected, Serial0/0/0
   150.1.0.0/24 is subnetted, 2 subnets
C       150.1.2.0 is directly connected, FastEthernet0/0
D       150.1.1.0 [90/21026560] via 193.1.2.1, 00:03:42, Serial0/0/0
RouterC#_

```

Lab 3-7. EIGRP Authentication



- **Yêu cầu**

Cấu hình EIGRP authentication giữa hai router R1 và R2.

- **Cấu hình**

- Cấu hình cơ bản

```

R1(config)#interface fa0/0
R1(config-if)#ip address 155.1.0.1 255.255.0.0
R1(config-if)#no shutdown

R1(config)#interface fa0/1
R1(config-if)#ip address 192.168.10.1 255.255.255.0
R1(config-if)#no shutdown

R1(config)#router eigrp 100

```

```
R1(config-if)#network 192.168.10.0
R1(config-if)#network 155.1.0.0
R1(config-if)#no auto-summary

R2(config)#interface fa0/0
R2(config-if)#ip address 155.1.0.2 255.255.0.0
R2(config-if)#no shutdown

R2(config)#interface fa0/1
R2(config-if)#ip address 172.16.20.1 255.255.255.0
R2(config-if)#no shutdown

R2(config)#router eigrp 100
R2(config-if)#network 172.16.0.0
R2(config-if)#network 155.1.0.0
R2(config-if)#no auto-summary
```

- Cấu hình authentication

```
R1(config)#key chain my_keychain1
R1(config-keychain)#key 1
R1(config-keychain-key)#key-string cisco

R1(config)#interface fa0/0
R1(config-if)#ip authentication mode eigrp 100 md5
R1(config-if)#ip authentication key-chain eigrp 100 my_keychain1

R2(config)#key chain my_keychain2
R2(config-keychain)#key 1
R2(config-keychain-key)#key-string cisco

R2(config)#interface fa0/0
R2(config-if)#ip authentication mode eigrp 100 md5
R2(config-if)#ip authentication key-chain eigrp 100 my_keychain2
```

• Kiểm tra cấu hình

Dùng các lệnh sau:

```
show ip eigrp neighbors
show ip eigrp interfaces details
show key chain
```

Lab 3-8. DYNAMIC ROUTING – OSPF

❖ **Mô tả**

- RouterA, RouterB, RouterC sử dụng OSPF để quảng bá thông tin định tuyến
- Các router cấu hình OSPF để quảng bá tất cả các mạng nối tiếp. Từ Router A, B và C ta ping thử tất cả các địa chỉ trong mạng.

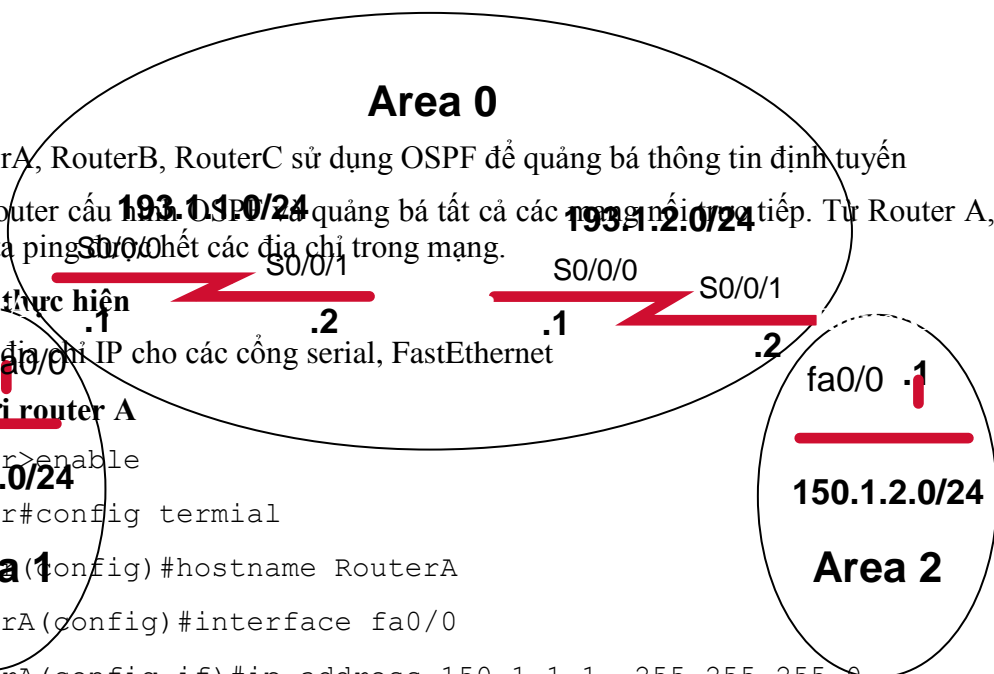
❖ **Các bước thực hiện**

Đặt hostname và chỉ IP cho các cổng serial, FastEthernet

• **Đối với router A**

```

Router>enable
Router#config terminal
Router(config)#hostname RouterA
RouterA(config)#interface fa0/0
RouterA(config-if)#ip address 150.1.1.1 255.255.255.0
RouterA(config-if)#no shutdown
RouterA(Config-if)#exit
    
```



```
RouterA(config)#interface s0/0/0
RouterA(config-if)#ip address 193.1.1.1 255.255.255.0
RouterA(config-if)#clock rate 64000
RouterA(config-if)#no shutdown
RouterA(config-if)#exit
```

- **Đối với router B**

```
Router>enable
Router#config terminal
Router(config)#hostname RouterB
RouterB(config)#interface S0/0/1
RouterB(config-if)#ip address 193.1.1.2 255.255.255.0
RouterB(config-if)#no shutdown
RouterB(config-if)#exit

RouterB(config)#interface S0/0/0
RouterB(config-if)#ip address 193.1.2.1 255.255.255.0
RouterB(config-if)#clock rate 64000
RouterB(config-if)#no shutdown
RouterB(config-if)#exit
```

- **Đối với router C**

```
Router>enable
Router#config terminal
Router(config)#hostname RouterC
RouterC(config)#interface fa0/0
RouterC(config-if)#ip address 150.1.2.1 255.255.255.0
RouterC(config-if)#no shutdown
RouterC(Config-if)#exit

RouterC(config)#interface S0/0/1
RouterC(config-if)#ip address 193.1.2.2 255.255.255.0
RouterC(config-if)#no shutdown
RouterC(config-if)#exit
```

- **Cấu hình giao thức định tuyến OSPF trên mỗi router**

```
RouterA(config)#router ospf 1
RouterA(config-router)#network 150.1.1.0 0.0.0.255 area 1
```



```

RouterA(config-router)#network 193.1.1.0 0.0.0.255 area 0

RouterB(config)#router ospf 1

RouterB(config-router)#network 193.1.1.0 0.0.0.255 area 0

RouterB(config-router)#network 193.1.2.0 0.0.0.255 area 0

RouterC(config)#router ospf 1

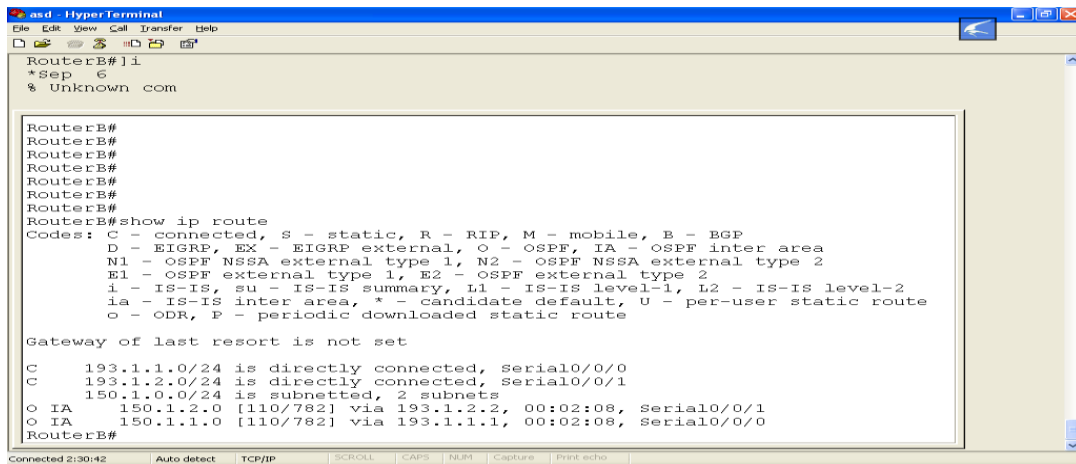
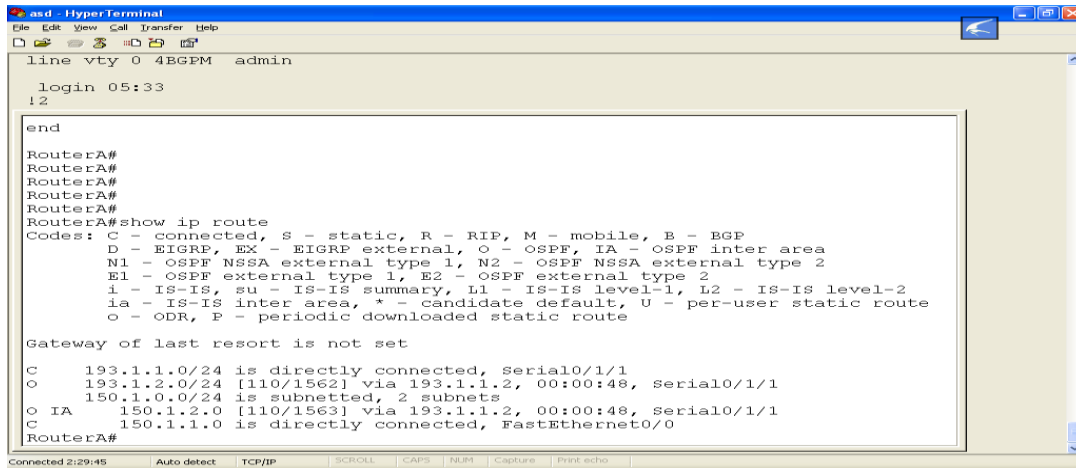
RouterC(config-router)#network 150.1.2.0 0.0.0.255 area 2

RouterC(config-router)#network 193.1.2.0 0.0.0.255 area 0
    
```

❖ **Kiểm tra cấu hình**

Thực hiện các câu lệnh sau để kiểm tra cấu hình

- Router#show ip route : xem bảng định tuyến
- Router#ping : kiểm tra kết nối



```

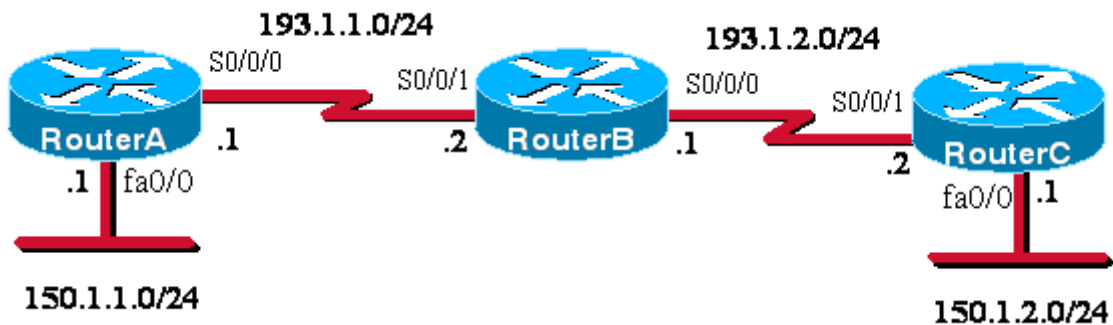
asd - HyperTerminal
File Edit View Call Transfer Help
Gateway
NTP/Z.
O IA 150.1.1.0 [110/782] via 193.1.1.1, 00:02:08, Serial0/0/0
RouterB#0
Termserver#3
[Resuming connection 3 to r5-3 ... ]
RouterC#
RouterC#
RouterC#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
I - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

O 193.1.1.0/24 [110/1562] via 193.1.2.1, 00:02:24, Serial0/0/0
O 193.1.2.0/24 [110/1562] is directly connected, Serial0/0/0
O 150.1.0.0/24 [110/1562] is subnetted, 2 subnets
C 150.1.1.0 is directly connected, FastEthernet0/0
O IA 150.1.1.0 [110/1563] via 193.1.2.1, 00:02:24, Serial0/0/0
RouterC#_
    
```

Lab 3-9. OSPF Authentication

❖ Topology



❖ Yêu cầu

Cấu hình OSPF cho các router RouterA, RouterB và RouterC (Single Area - Area 0) trong mô hình mạng trên để quảng bá các thông tin định tuyến. Cấu hình chứng thực dạng Plain text và MD5 giữa 2 router: **RouterA** và **RouterB**

❖ Hướng dẫn cấu hình

Bước 1: Cấu hình cơ bản (đặt hostname, địa chỉ IP cho các interface: loopback, serial, FastEthernet)

- Đối với router A

```

Router>enable
Router#config terminal
Router(config)#hostname routerA
routerA(config)#int f0/0
routerA(config-if)#ip address 150.1.1.1 255.255.255.0
routerA(config-if)#no shutdown
routerA(Config-if)#exit

routerA(config)#int s0/0/0
    
```

```
routerA(config-if)#ip address 193.1.1.1 255.255.255.0
routerA(config-if)#clock rate 64000
routerA(config-if)#no shutdown
routerA(config-if)#exit
```

- Đối với router B

```
Router>enable
Router#configure terminal
Router(config)#hostname routerB
routerB(config)#interface serial 0/0/1
routerB(config-if)#ip address 193.1.1.2 255.255.255.0
routerB(Config-if)#no shutdown
routerB(Config-if)#exit
```

```
routerB(config)#interface serial 0/0/0
routerB(config-if)#ip address 193.1.2.1 255.255.255.0
routerB(config-if)#clock rate 64000
routerB(config-if)#no shutdown
routerB(config-if)#exit
```

- Đối với router C

```
Router>enable
Router#configure terminal
Router(config)#hostname RouterC
RouterC(config)#interface fastEthernet 0/0
RouterC(config-if)#ip address 150.1.2.1 255.255.255.0
RouterC(config-if)#no shutdown
RouterC(Config-if)#exit
```

```
RouterC(config)#int s0/0/1
RouterC(config-if)#ip address 193.1.2.2 255.255.255.0
RouterC(config-if)#no shutdown
RouterC(config-if)#exit
```

Bước 2: Cấu hình giao thức định tuyến OSPF trên mỗi router

```
routerA(config)#router ospf 1
routerA(config-router)#network 150.1.1.0 0.0.0.255 area 0
routerA(config-router)#network 193.1.1.0 0.0.0.255 area 0
```

```
routerB(config)#router ospf 1
routerB(config-router)#network 193.1.1.0 0.0.0.255 area 0
routerB(config-router)#network 193.1.2.0 0.0.0.255 area 0
```

```
RouterC(config)#router ospf 1
RouterC(config-router)#network 150.1.2.0 0.0.0.255 area 0
RouterC(config-router)#network 193.1.2.0 0.0.0.255 area 0
```

Bước 3.1. Cấu hình chứng thực dạng Plain text giữa 2 router: RouterA và RouterB

```
routerA(config)#int S0/0/0
routerA(config-if)#ip ospf authentication
routerA(config-if)#ip ospf authentication-key cisco
```

```
RouterB(config)#int S0/0/1
routerB(config-if)#ip ospf authentication
routerB(config-if)#ip ospf authentication-key cisco
```

Bước 3.2 Cấu hình chứng thực dạng MD5 giữa 2 router: RouterA và RouterB

```
routerA(config)#int S0/0/0
routerA(config-if)#ip ospf authentication message-digest
routerA(config-if)#ip ospf messages-digest-key 1 md5 cisco
```

```
RouterB(config)#int S0/0/1
routerB(config-if)#ip ospf authentication message-digest
routerB(config-if)#ip ospf messages-digest-key 1 md5 cisco
```

Bước 4. Kiểm tra cấu hình

Thực hiện các câu lệnh sau để kiểm tra cấu hình

show ip route: xem bảng định tuyến

debug ip ospf event : xem quá trình cập nhật định tuyến của OSPF

Lab 3-10. REDISTRIBUTE GIỮA RIP & EIGRP

Mô tả

- RouterA, RouterB sử dụng EIGRP để quảng cáo thông tin định tuyến
- RouterB, RouterC sử dụng RIP để quảng cáo thông tin định tuyến
- Từ RouterA, RouterB, RouterC ping được hết các địa chỉ trong mạng

Các bước thực hiện

Đặt hostname địa chỉ IP cho các cổng Loopback, Serial, FastEthernet

EIGRP AS 100

```
Router(config)#hostname RouterA
RouterA(config)#interface fa0/0
RouterA(config-if)#ip addresss 172.16.1.1 255.255.255.0
RouterA(config-if)#no shutdown
RouterA(Config-if)#exit

RouterA(config)#interface s0/0/0
RouterA(config-if)#ip address 193.1.1.1 255.255.255.0
```

RIP

193.1.1.0/24 S0/0/0

193.1.2.0/24 S0/0/0

150.1.2.0/24

```
RouterA(config-if)#clock rate 64000
RouterA(config-if)#no shutdown
RouterA(config-if)#exit
```

- **Đối với router B**

```
Router(config)#hostname RouterB
RouterB(config)#interface S0/0/1
RouterB(config-if)#ip address 193.1.1.2 255.255.255.0
RouterB(Config-if)#no shut
RouterB(Config-if)#exit

RouterB(config)#interface S0/0/0
RouterB(config-if)#ip address 193.1.2.1 255.255.255.0
RouterB(config-if)#clock rate 64000
RouterB(config-if)#no shutdown
RouterB(config-if)#exit
```

- **Đối với router C**

```
Router(config)#hostname RouterC
RouterC(config)#interface fa0/0
RouterC(config-if)#ip address 150.1.2.1 255.255.255.0
RouterC(config-if)#no shutdown
RouterC(config-if)#exit

RouterC(config)#interface S0/0/1
RouterC(config-if)#ip address 193.1.2.2 255.255.255.0
RouterC(config-if)#no shutdown
RouterC(config-if)#exit
```

- **Cấu hình giao thức định tuyến EIGRP trên mỗi RouterA và RouterB**

```
RouterA(config)#router eigrp 100
RouterA(config-router)#network 172.16.0.0
RouterA(config-router)#network 193.1.1.0 0
RouterA(config-router)#no auto-summary

RouterB(config)#router eigrp 100
RouterB(config-router)#network 193.1.1.0
RouterB(config-router)#no auto-summary

RouterB(config)#router rip
RouterB(config-router)#network 193.1.2.0
RouterB(config-router)#passive interface S0/0/1
```

```

RouterC (config) #router rip
RouterC (config-router) #network 150.1.0.0
RouterC (config-router) #network 193.1.2.0

```

Để RouterC thấy được RouterA, ta thực hiện redistribute

```

RouterB (config) #router rip
RouterB (config-router) #redistribute eigrp 100 metric 3

```

Tương tự : để RouterA thấy RouterC

```

RouterB (config) #router eigrp 100
RouterB (config-router) #redistribute rip metric 100 1 255 255 1500

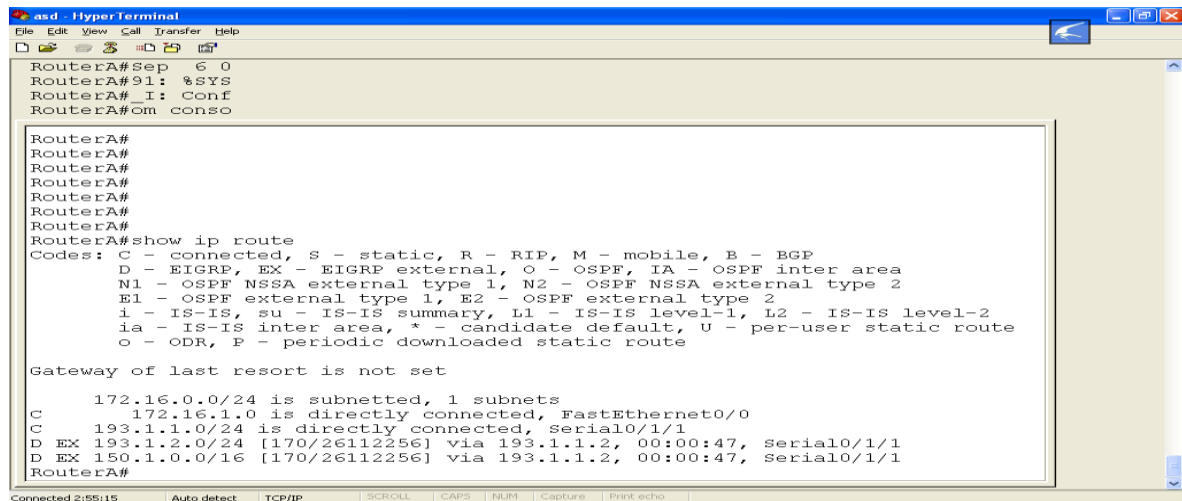
```

❖ Kiểm tra

Thực hiện các câu lệnh sau để kiểm tra cấu hình

show ip route : xem bảng định tuyến

ping : kiểm tra kết nối



```

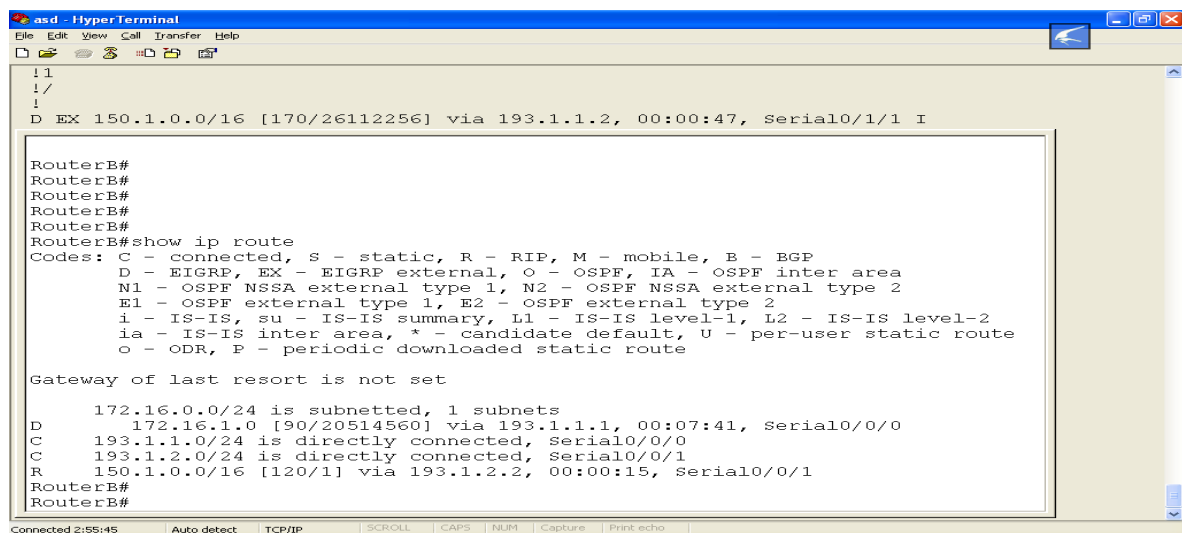
asd - HyperTerminal
File Edit View Call Transfer Help
RouterA#Sep 6 0
RouterA#91: %SYS
RouterA#_I: Conf
RouterA#Om conso

RouterA#
RouterA#
RouterA#
RouterA#
RouterA#
RouterA#
RouterA#
RouterA#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

    172.16.0.0/24 is subnetted, 1 subnets
C       172.16.1.0 is directly connected, FastEthernet0/0
C       193.1.1.0/24 is directly connected, Serial0/1/1
D EX    193.1.2.0/24 [170/26112256] via 193.1.1.2, 00:00:47, Serial0/1/1
D EX    150.1.0.0/16 [170/26112256] via 193.1.1.2, 00:00:47, Serial0/1/1
RouterA#

```



```

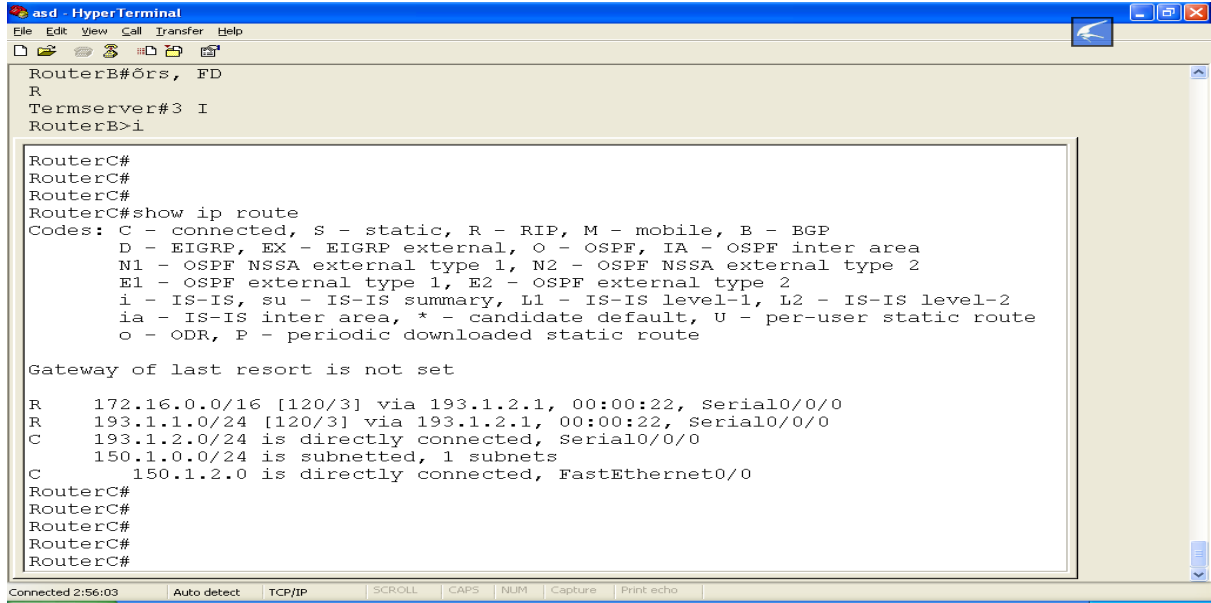
asd - HyperTerminal
File Edit View Call Transfer Help
!
!
!
D EX    150.1.0.0/16 [170/26112256] via 193.1.1.2, 00:00:47, Serial0/1/1 I

RouterB#
RouterB#
RouterB#
RouterB#
RouterB#
RouterB#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

    172.16.0.0/24 is subnetted, 1 subnets
D       172.16.1.0 [90/20514560] via 193.1.1.1, 00:07:41, Serial0/0/0
C       193.1.1.0/24 is directly connected, Serial0/0/0
C       193.1.2.0/24 is directly connected, Serial0/0/1
R       150.1.0.0/16 [120/1] via 193.1.2.2, 00:00:15, Serial0/0/1
RouterB#
RouterB#

```



Lab 3-11 REDISTRIBUTE GIỮA RIP & OSPF

Mô tả

- RouterA, RouterB sử dụng OSPF để quảng cáo thông tin định tuyến
- RouterB, RouterC sử dụng RIP để quảng cáo thông tin định tuyến
- Từ RouterA, RouterB, RouterC ping được hết các địa chỉ trong mạng

Các bước thực hiện

Đặt hostname địa chỉ IP cho các cổng Loopback, Serial, FastEthernet

OSPF

RIP

```

Router>enable
Router#config terminal
Router(config)#hostname RouterA
RouterA(config)#interface fa0/0
RouterA(config-if)#ip address 172.16.1.1 255.255.255.0
RouterA(config-if)#no shutdown
RouterA(Config-if)#exit

RouterA(config)#interface s0/0/0
RouterA(config-if)#ip address 193.1.1.1 255.255.255.0
RouterA(config-if)#clock rate 64000
RouterA(config-if)#no shutdown
    
```

```
RouterA(config-if)#exit
```

- Đối với router B

```
Router>enable
```

```
Router#config terminal
```

```
Router(config)#hostname RouterB
```

```
RouterB(config)#interface S0/0/1
```

```
RouterB(config-if)#ip address 193.1.1.2 255.255.255.0
```

```
RouterB(Config-if)#no shutdown
```

```
RouterB(Config-if)#exit
```

```
RouterB(config)#interface S0/0/0
```

```
RouterB(config-if)#ip address 193.1.2.1 255.255.255.0
```

```
RouterB(config-if)#clock rate 64000
```

```
RouterB(config-if)#no shutdown
```

```
RouterB(config-if)#exit
```

- Đối với router C

```
Router>enable
```

```
Router#config terminal
```

```
Router(config)#hostname RouterC
```

```
RouterC(config)#interface fa0/0
```

```
RouterC(config-if)#ip address 150.1.2.1 255.255.255.0
```

```
RouterC(config-if)#no shutdown
```

```
RouterC(Config-if)#exit
```

```
RouterC(config)#interface S0/0/1
```

```
RouterC(config-if)#ip address 193.1.2.2 255.255.255.0
```

```
RouterC(config-if)#no shutdown
```

```
RouterC(config-if)#exit
```

- Cấu hình giao thức định tuyến OSPF trên mỗi RouterA và RouterB

```
RouterA(config)#router ospf 1
```

```
RouterA(config-router)#network 172.16.1.0 0.0.0.255 area 0
```

```
RouterA(config-router)#network 193.1.1.0 0.0.0.255 area 0
```

```
RouterB(config)#router ospf 1
```

```
RouterB(config-router)#network 193.1.1.0 0.0.0.255 area 0
```

```
RouterB(config)#router rip
```

```
RouterB(config-router)#network 193.1.2.0
```



```
RouterC(config)#router rip
RouterC(config-router)#network 150.1.0.0
RouterC(config-router)#network 193.1.2.0
```

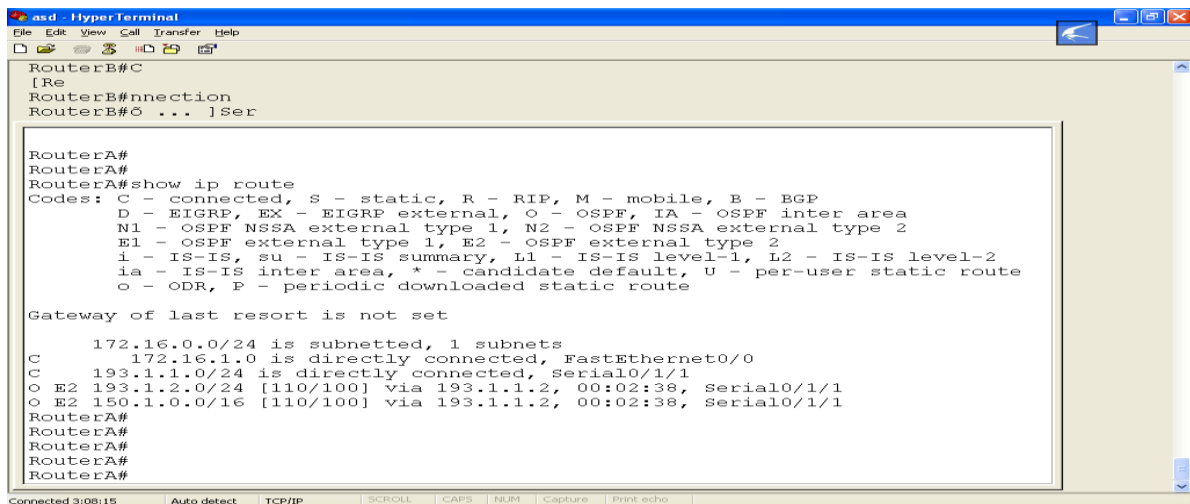
- Cấu hình redistribute

Để RouterC thấy được RouterA, ta thực hiện redistribute

```
RouterB(config)#router rip
RouterB(config-router)#redistribute ospf 1 metric 3
```

Tương tự : để RouterA thấy RouterC

```
RouterB(config)#router ospf 1
RouterB(config-router)#redistribute rip metric 100 subnets
```



```

RouterB#C
[Re
RouterB#nnection
RouterB#O ... ]ser

RouterA#
RouterA#
RouterA#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

      172.16.0.0/24 is subnetted, 1 subnets
C       172.16.1.0 is directly connected, FastEthernet0/0
C       193.1.1.0/24 is directly connected, Serial0/1/1
O E2 193.1.2.0/24 [110/100] via 193.1.1.2, 00:02:38, Serial0/1/1
O E2 150.1.0.0/16 [110/100] via 193.1.1.2, 00:02:38, Serial0/1/1
RouterA#
RouterA#
RouterA#
RouterA#
RouterA#

```



```

RouterA#, Serial
RouterA#, * - ca
RouterA#efault,
RouterA#Oser static

RouterB#
RouterB#
RouterB#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

      172.16.0.0/24 is subnetted, 1 subnets
O       172.16.1.0 [110/782] via 193.1.1.1, 00:02:56, Serial0/0/0
C       193.1.1.0/24 is directly connected, Serial0/0/0
C       193.1.2.0/24 is directly connected, Serial0/0/1
R       150.1.0.0/16 [120/1] via 193.1.2.2, 00:00:04, Serial0/0/1
RouterB#
RouterB#
RouterB#
RouterB#
RouterB#
RouterB#

```

```

RouterB#G.255.255.
Termserver#30.0/24

Rout

RouterC#
RouterC#
RouterC#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2
        i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, * - candidate default, U - per-user static route
        o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

R    172.16.0.0/16 [120/3] via 193.1.2.1, 00:00:26, Serial0/0/0
R    193.1.1.0/24 [120/3] via 193.1.2.1, 00:00:26, Serial0/0/0
C    193.1.2.0/24 is directly connected, Serial0/0/0
C    150.1.0.0/24 is subnetted, 1 subnets
C      150.1.2.0 is directly connected, FastEthernet0/0
RouterC#
RouterC#
RouterC#
RouterC#
RouterC#
RouterC#_
    
```

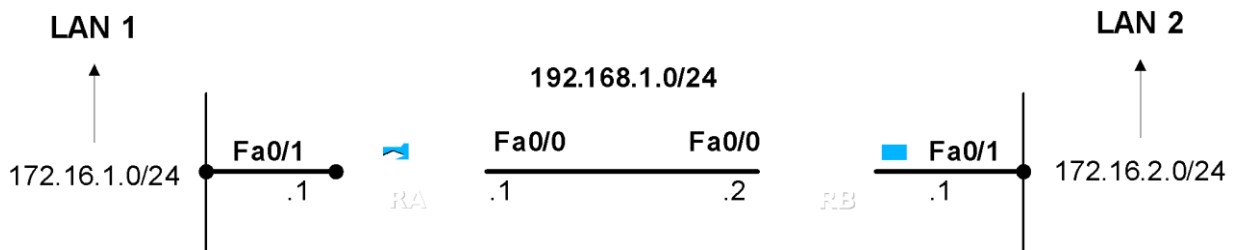
❖ Kiểm tra cấu hình

Thực hiện các câu lệnh sau để kiểm tra cấu hình

Router#show ip route : xem bảng định tuyến

Router#ping : kiểm tra kết nối

Lab 3-12. DHCP Server



❖ Yêu cầu

Cấu hình theo mô hình mạng trên sao cho router RA và RB đóng vai trò là DHCP server cấp IP cho các host thuộc LAN của mình.

- LAN 1
 - IP Pool: 172.16.1.10 -172.16.1.100
 - Default Gateway: 172.16.1.1
 - DNS: 1.1.1.1
- LAN 2
 - IP Pool: 172.16.2.200 -172.16.2.250
 - Default Gateway: 172.16.2.1
 - DNS: 2.2.2.2

❖ Cấu hình

Router RA

- **Cấu hình địa chỉ IP**

```
RA(config)#interface fa0/0
RA(config-if)#ip address 192.168.1.1 255.255.255.0
RA(config-if)#no shutdown

RA(config)#interface fa0/1
RA(config-if)#ip address 172.16.1.1 255.255.255.0
RA(config-if)#no shutdown
```

- **Cấu hình định tuyến tĩnh**

```
RA(config)#ip route 172.16.2.0 255.255.255.0 192.168.1.2
```

- **Cấu hình DHCP**

```
RA(config)#ip dhcp pool LAN_1
RA(dhcp-config)#network 172.16.1.0 255.255.255.0
RA(dhcp-config)#default-router 172.16.1.1
RA(dhcp-config)#dns-server 1.1.1.1
RA(dhcp-config)#exit

RA(config)#ip dhcp excluded-address 172.16.1.1 172.16.1.9
RA(config)#ip dhcp excluded-address 172.16.1.101 172.16.1.254
```

Router RB

- **Cấu hình địa chỉ IP**

```
RB(config)#interface fa0/0
RB(config-if)#ip address 192.168.1.2 255.255.255.0
RB(config-if)#no shutdown

RB(config)#interface fa0/1
RB(config-if)#ip address 172.16.2.1 255.255.255.0
RB(config-if)#no shutdown
```

- **Cấu hình định tuyến tĩnh:**

```
RB(config)#ip route 172.16.1.0 255.255.255.0 192.168.1.1
```

- **Cấu hình DHCP:**

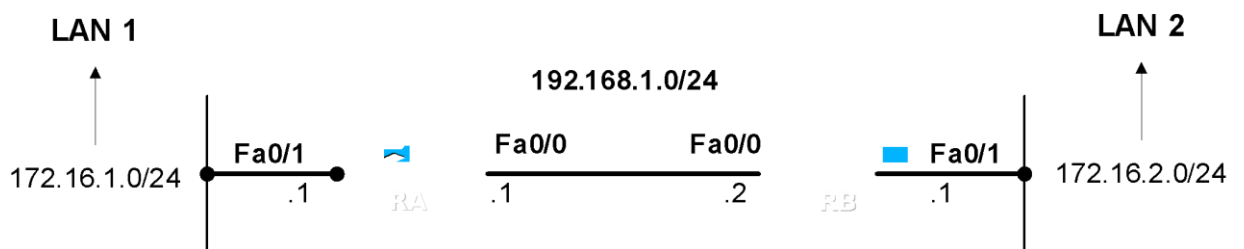
```
RB(config)#ip dhcp pool LAN_2
RB(dhcp-config)#network 172.16.2.0 255.255.255.0
RB(dhcp-config)#default-router 172.16.2.1
RB(dhcp-config)#dns-server 2.2.2.2
RB(dhcp-config)#exit

RB(config)#ip dhcp excluded-address 172.16.2.1 172.16.1.199
```

```
RB(config)#ip dhcp excluded-address 172.16.2.251 172.16.1.254
```

Lab 3-13. DHCP - Helper Address

❖ Topology



❖ Yêu cầu

Cấu hình mô hình mạng trên sao cho RA làm DHCP server cấp IP cho LAN 1 và LAN 2.

- LAN1
IP Pool: 172.16.1.10 -172.16.1.100
DG: 172.16.1.1
DNS: 1.1.1.1
- LAN 2

IP Pool: 172.16.2.200 -172.16.2.250

DG: 172.16.2.1

DNS: 2.2.2.2

❖ Cấu hình

Router RA

- Cấu hình địa chỉ IP

```
RA(config)#interface fa0/0
```

```
RA(config-if)#ip address 192.168.1.1 255.255.255.0
```

```
RA(config-if)#no shutdown
```

```
RA(config)#interface fa0/1
```

```
RA(config-if)#ip address 172.16.1.1 255.255.255.0
```

```
RA(config-if)#no shutdown
```

- Cấu hình định tuyến tĩnh:

```
RA(config)#ip route 172.16.2.0 255.255.255.0 192.168.1.2
```

- Cấu hình DHCP:

```
RA(config)#ip dhcp pool LAN_1
```

```
RA(dhcp-config)#network 172.16.1.0 255.255.255.0
```

```
RA(dhcp-config)#default-router 172.16.1.1
```

```
RA(dhcp-config)#dns-server 1.1.1.1
```

```
RA(dhcp-config)#exit
```

```
RA(config)#ip dhcp excluded-address 172.16.1.1 172.16.1.9
```

```
RA(config)#ip dhcp excluded-address 172.16.1.101 172.16.1.254
```

```
RA(config)#ip dhcp pool LAN_2
```

```
RA(dhcp-config)#network 172.16.2.0 255.255.255.0
```

```
RA(dhcp-config)#default-router 172.16.2.1
```

```
RA(dhcp-config)#dns-server 2.2.2.2
```

```
RA(dhcp-config)#exit
```

```
RA(config)#ip dhcp excluded-address 172.16.2.1 172.16.1.199
```

```
RA(config)#ip dhcp excluded-address 172.16.2.251 172.16.1.254
```

Router RB

- Cấu hình địa chỉ IP

```
RB(config)#interface fa0/0
```

```
RB(config-if)#ip address 192.168.1.2 255.255.255.0
```

```
RB(config-if)#no shutdown
```

```
RB(config)#interface fa0/1
```

```
RB(config-if)#ip address 172.16.2.1 255.255.255.0
```

```
RB(config-if)#no shutdown
```

- Cấu hình định tuyến tĩnh:

```
RB(config)#ip route 172.16.1.0 255.255.255.0 192.168.1.1
```

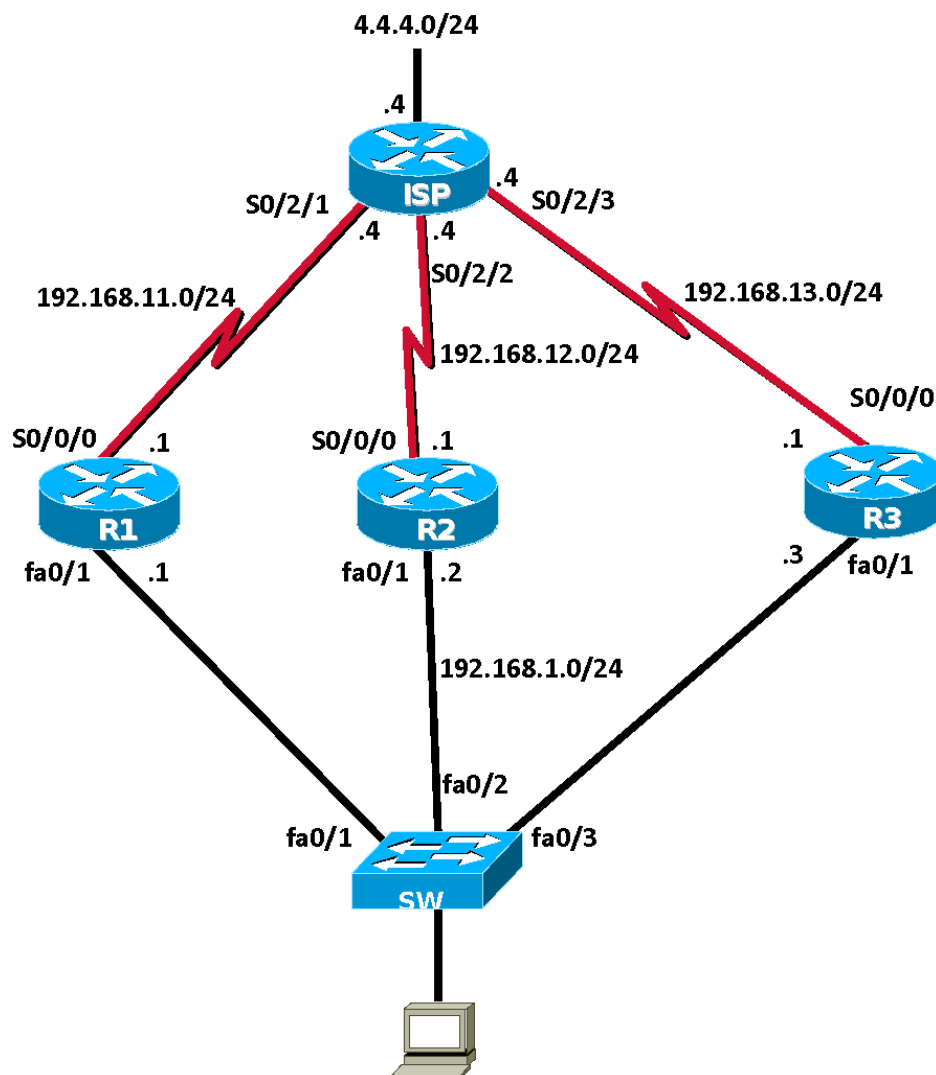
- Cấu hình DHCP:

```
RB(config)#interface fa0/1
```

```
RB(config-if)#ip helper-address 192.168.1.1
```

Lab 3.14. HSRP

Topology



Yêu cầu

1. Cấu hình cho routing cho mạng trên
2. Cấu hình HSRP cho 3 router R1, R2 và R3
 - a. Virtual IP 192.168.1.254

- b. R1 là active router, R2 là standby router
- c. R1, R2, R3 có khả năng trở thành active router nếu có độ ưu tiên cao hơn
- d. Cấu hình loadbalance giữa 3 router

Group 1: 192.168.1.254

Group 2: 192.168.1.253

Group 3: 192.168.1.252

Cấu hình

1. Cấu hình cho routing cho mạng sử dụng RIPv2

```
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#network 192.168.1.0
R1(config-router)#network 192.168.11.0
R1(config-router)#no auto-summary
```

```
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#network 192.168.1.0
R2(config-router)#network 192.168.12.0
R2(config-router)#no auto-summary
```

```
R3(config)#router rip
R3(config-router)#version 2
R3(config-router)#network 192.168.1.0
R3(config-router)#network 192.168.13.0
R3(config-router)#no auto-summary
```

```
ISP(config)#router rip
ISP(config-router)#version 2
ISP(config-router)#network 192.168.11.0
ISP(config-router)#network 192.168.12.0
ISP(config-router)#network 192.168.13.0
ISP(config-router)#network 4.0.0.0
ISP(config-router)#no auto-summary
```

2. Cấu hình HSRP

a. Virtual IP 192.168.1.254

```
R1(config)#interface fa0/1
```

```
R1(config-if)#standby 1 ip 192.168.1.254
R2(config)#interface fa0/1
R2(config-if)#standby 1 ip 192.168.1.254
R3(config)#interface fa0/1
R3(config-if)#standby 1 ip 192.168.1.254
```

b. R1 là active router, R2 là standby router

```
R1(config)#interface f0/1
R1(config-if)#standby 1 priority 150
R2(config)#interface f0/1
R2(config-if)#standby 1 priority 120
```

c. R1, R2, R3 có khả năng trở thành active router nếu có độ ưu tiên cao hơn

```
R1(config)#interface f0/1
R1(config-if)#standby 1 preempt
R2(config)#interface f0/1
R2(config-if)#standby 1 preempt
R3(config)#interface f0/1
R3(config-if)#standby 1 preempt
```

d. Cấu hình 3 group HSRP để loadbalance giữa 3 router

R1 là active của Group 1

R2 là active của Group 2

R3 là active của Group 3

```
R1(config)#interface f0/1
R1(config-if)#standby 2 ip 192.168.1.253
R1(config-if)#standby 3 ip 192.168.1.252

R2(config)#interface f0/1
R2(config-if)#standby 2 ip 192.168.1.253
R2(config-if)#standby 2 preempt
R2(config-if)#standby 2 priority 150
R2(config-if)#standby 3 ip 192.168.1.252

R3(config)#interface f0/1
R3(config-if)#standby 2 ip 192.168.1.253
R3(config-if)#standby 3 ip 192.168.1.252
R3(config-if)#standby 3 preempt
R3(config-if)#standby 3 priority 150
```


Kiểm tra cấu hình

show standby

show track

Chương 4.

SWITCH

Chương này đề cập đến một số kỹ thuật được triển khai trên Switch như VLAN, VTP, STP. Học xong chương này, người học có khả năng:

- Phân biệt giữa miền đưng độ và miền quảng bá
- Trình bày được khái niệm và đặc điểm của VLAN, VTP, STP
- Cấu hình VLAN, VTP, STP trên switch
- Cấu hình định tuyến giữa các VLAN

1. Giới thiệu

- **Collision domain:** miền đưng độ

Đưng độ xảy ra khi có hai hay nhiều máy truyền dữ liệu đồng thời trong một mạng chia sẻ. Khi đưng độ xảy ra, các gói tin đang được truyền đều bị phá hủy, các máy đang truyền sẽ ngưng việc truyền dữ liệu và chờ một khoảng thời gian ngẫu nhiên theo quy luật của CSMA/CD. Nếu đưng độ xảy ra quá nhiều mạng có thể không hoạt động được.

Miền đưng độ là khu vực mà dữ liệu được phát ra có thể bị đưng độ. Tất cả các môi trường mạng chia sẻ là các miền đưng độ.

- **Broadcast domain:** miền quảng bá

Các thông tin liên lạc trong mạng được thực hiện theo ba cách: unicast, multicast và broadcast.

- Unicast: gửi trực tiếp từ một máy đến một máy.
- Multicast: được thực hiện khi một máy muốn gửi gói tin cho một nhóm máy.
- Broadcast: được thực hiện khi một máy muốn gửi cho tất cả các máy khác trong mạng.

Khi một thiết bị muốn gửi một gói quảng bá thì địa chỉ MAC đích của gói tin đó sẽ là FF:FF:FF:FF:FF:FF. Với địa chỉ như vậy, mọi thiết bị đều nhận và xử lý gói quảng bá.

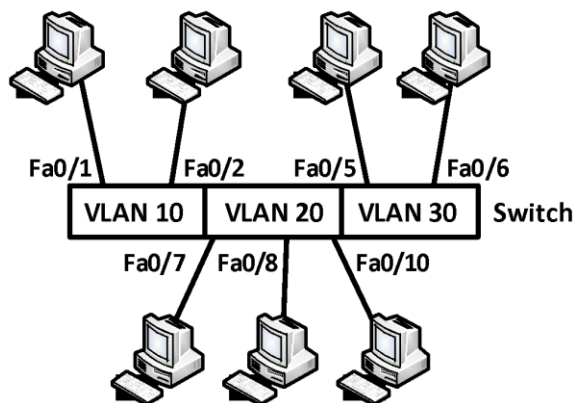
Miền quảng bá là miền bao gồm tất cả các thiết bị có thể nhận được gói tin quảng bá từ một thiết bị nào đó trong LAN.

Switch là thiết bị hoạt động ở tầng liên kết dữ liệu, khi Switch nhận được gói quảng bá thì nó sẽ gửi ra tất cả các cổng của nó trừ cổng nhận gói tin vào. Mỗi thiết bị nhận được gói quảng bá đều phải xử lý thông tin nằm trong đó.

Router là thiết bị hoạt động ở tầng mạng, router không chuyển các gói quảng bá. Router được sử dụng để chia mạng thành nhiều miền đưng độ và nhiều miền quảng bá.

2. VLAN

VLAN (Virtual LAN) là kỹ thuật được sử dụng trên Switch, dùng để chia một Switch vật lý thành nhiều Switch luận lý. Mỗi một Switch luận lý gọi là một VLAN hoặc có thể hiểu VLAN là một tập hợp của các cổng trên Switch nằm trong cùng một miền quảng bá. Các cổng trên Switch có thể được nhóm vào các VLAN khác nhau trên một Switch hoặc được triển khai trên nhiều Switch.



Hình 2.1 Chia VLAN trên switch

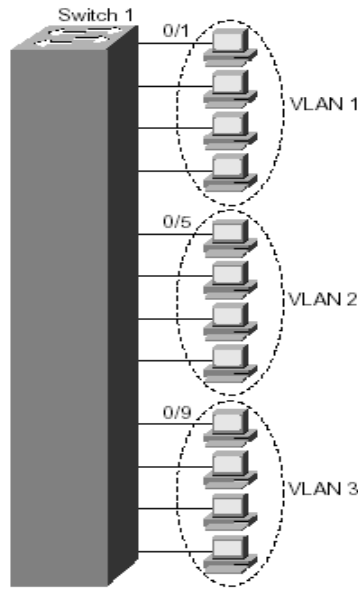
Khi có một gói tin quảng bá được gửi bởi một thiết bị nằm trong một VLAN sẽ được chuyển đến các thiết bị khác nằm trong cùng VLAN đó, gói tin quảng bá sẽ không được chuyển tiếp đến các thiết bị thuộc VLAN khác.

VLAN cho phép người quản trị tổ chức mạng theo luận lý chứ không theo vật lý. Sử dụng VLAN có ưu điểm là:

- ✓ Tăng khả năng bảo mật
- ✓ Thay đổi cấu hình LAN dễ dàng
- ✓ Di chuyển máy trạm trong LAN dễ dàng
- ✓ Thêm máy trạm vào LAN dễ dàng.

VLAN = broadcast domain = logical network
--

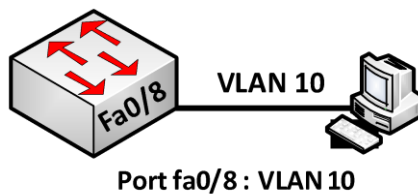
Một VLAN là một tập hợp của các switchport nằm trong cùng một broadcast domain. Các cổng trên switch có thể được nhóm vào các VLAN khác nhau trên từng switch hoặc trên nhiều switch.



Khi có một gói tin broadcast được gửi bởi một thiết bị nằm trong một VLAN sẽ được chuyển đến các thiết bị khác nằm trong cùng VLAN đó, broadcast sẽ không được forward đến các thiết bị trong vlan khác.

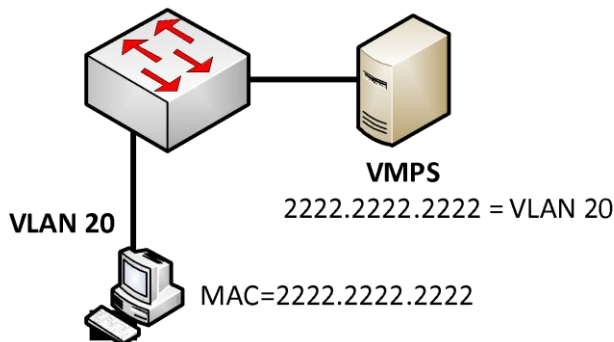
3. Phân loại

- VLAN tĩnh (Static VLAN)



Đối với loại này, các cổng của Switch được cấu hình thuộc về một VLAN nào đó, các thiết bị gắn vào cổng đó sẽ thuộc về VLAN đã định trước. Đây là loại VLAN dùng phổ biến.

- VLAN động (dynamic VLAN)



Loại VLAN này sử dụng một server lưu trữ địa chỉ MAC của các thiết bị và qui định VLAN mà thiết bị đó thuộc về, khi một thiết bị gắn vào Switch, Switch sẽ lấy địa chỉ MAC của thiết bị và gửi cho server kiểm tra và cho vào VLAN định trước.

4. Cấu hình VLAN

Bước 1. Tạo VLAN

```
Switch(config)#vlan <vlan-id>
Switch(config-vlan)#name <vlan-name>
```

Ví dụ:

```
Switch(config)#vlan 10
Switch(config-if)#name P.KyThuat
```

Bước 2. Gán các cổng cho VLAN

- Gán 1 cổng vào LAN

```
Switch(config)#interface <interface>
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan <vlan-id>
```

Ví dụ:

```
Switch(config)#interface fa0/5
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 10
```

- Gán 1 dãy các cổng liên tiếp

```
Switch(config)#interface range <start>-<end-intf>
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan <vlan-id>
```

Ví dụ:

```
Switch(config)#interface fa0/10 - 20
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 10
```

- Gán nhiều cổng không liên tiếp

```
Switch(config)#interface range <interfacel, interface2,...>
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan <vlan-id>
```

Ví dụ:

```
Switch(config)#interface fa0/7, fa0/9, fa0/2
Switch(config-if-range)#switchport mode access
Switch(config-if-range)#switchport access vlan 10
```

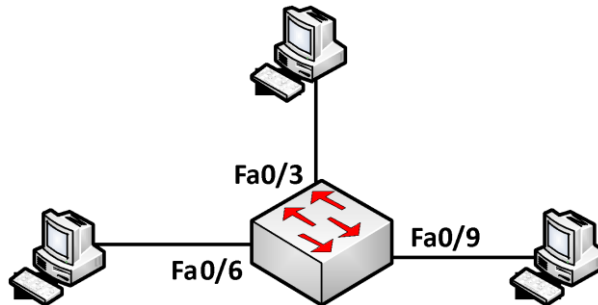
❖ **Xóa VLAN:** Xóa một VLAN trên switch bằng cách sử dụng lệnh “no” trước câu lệnh tạo VLAN.

❖ **Lệnh kiểm tra cấu hình VLAN**

```
Switch#show vlan
```

Lệnh này cho phép hiển thị các VLAN-ID (số hiệu VLAN), tên VLAN, trạng thái VLAN và các cổng được gán cho VLAN trên switch.

Ví dụ:



Mô tả yêu cầu:

- Cấu hình VLAN trên Switch
- Tạo 3 VLAN: VLAN 10, VLAN 20, VLAN 30
- Fa0/1 – Fa0/6: VLAN 10, Fa0/7 – Fa0/9: VLAN 20, Fa0/10 – Fa0/12: VLAN 30

Các bước thực hiện:

✓ **Tạo vlan:**

```
Switch(config)#vlan 10
```

```
Switch(config)#vlan 20
```

```
Switch(config)#vlan 30
```

✓ **Gán các cổng vào VLAN**

```
Switch(config)#interface range f0/1 - 6
```

```
Switch(config-if-range)#switchport mode access
```

```
Switch(config-if-range)#switchport access vlan 10
```

```
Switch(config)#interface range f0/7 - 9
```

```
Switch(config-if-range)#switchport mode access
```

```
Switch(config-if-range)#switchport access vlan 20
```

```
Switch(config)#interface range f0/10 - 12
```

```
Switch(config-if-range)#switchport mode access
```

```
Switch(config-if-range)#switchport access vlan 30
```

✓ **Kiểm tra cấu hình:**

Thực hiện các câu lệnh sau để kiểm tra cấu hình

```
Switch#show run
```

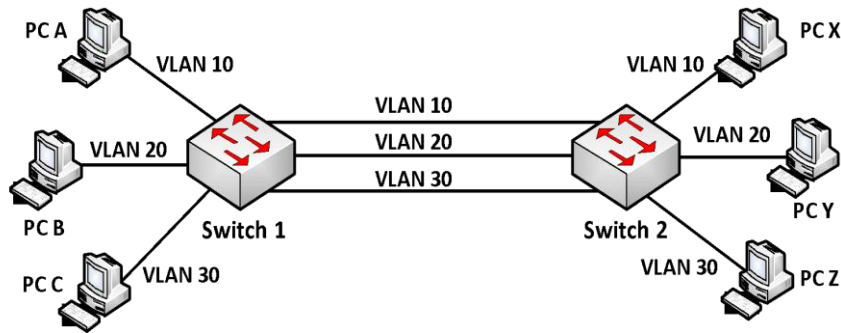
```
Switch#show vlan
```

Gắn PC vào các cổng như trên sơ đồ, đặt IP cho các PC và dùng lệnh “ping” để kiểm tra kết nối.

5. Đường Trunk

VLAN tổ chức trên nhiều switch như vậy làm sao các thiết bị thuộc cùng một VLAN nhưng nằm ở những switch khác nhau có thể liên lạc với nhau? Chúng ta có hai cách để giải quyết vấn đề này:

- Dùng mỗi kết nối cho từng VLAN

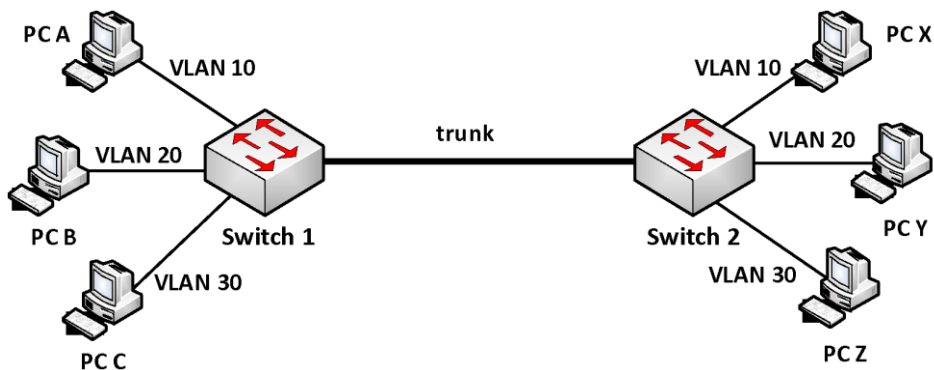


Có nghĩa là mỗi VLAN ở trên các switch sẽ được kết nối lại bằng một đường kết nối riêng. Theo mô hình trên ta thấy: nếu PC A trong VLAN 10 ở Switch 1 muốn liên lạc với PC X trong VLAN 10 ở Switch 2, ta phải có một kết nối vật lý nối Switch 1 với Switch 2 và hai cổng kết nối này phải thuộc cùng VLAN 10.

Tương tự đối với VLAN 2 và VLAN3, ta cần hai kết nối vật lý. Như vậy, với n VLAN được tạo ra tổng cộng ta phải dùng đến n dây nối để các thành viên trong cùng VLAN có thể giao tiếp được với nhau. Điều này gây ra lãng phí.

- Kết nối trunk (đường trunk)

Một kỹ thuật khác để giải quyết vấn đề trên là dùng chỉ một kết nối cho phép dữ liệu của các VLAN có thể cùng lưu thông qua đường này. Người ta gọi kết nối này là đường *trunk*.



Theo như mô hình trên chúng ta chỉ dùng một dây nối Switch 1 với Switch 2, các thành viên trong cùng VLAN ở các Switch khác nhau vẫn có thể giao tiếp với nhau. Đường dây như thế gọi là liên kết trunk lớp 2.

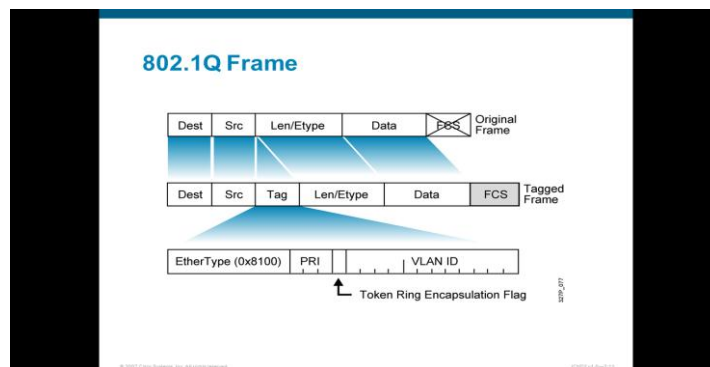
Mỗi thành viên trong cùng VLAN chỉ có thể thấy thành viên khác trong cùng VLAN với nó. Để PC A có thể giao tiếp với PC B hoặc C (không thuộc cùng VLAN), cần phải sử dụng thiết bị ở lớp 3 như router hay switch lớp 3 (Multilayer Switch hay Switch layer 3).

Kết nối “*trunk*” là liên kết Point-to-Point giữa các cổng trên switch với router hoặc với switch khác. Kết nối “*trunk*” sẽ vận chuyển dữ liệu của nhiều VLAN thông qua một liên kết đơn và cho phép mở rộng VLAN trên hệ thống mạng.

Vi kỹ thuật này cho phép dùng chung một kết nối vật lý cho dữ liệu của các VLAN đi qua nên để phân biệt được chúng là dữ liệu của VLAN nào, người ta gắn vào các gói tin một dấu hiệu gọi là “*tagging*”. Hay nói cách khác là dùng một kiểu đóng gói riêng cho các gói tin đi chuyển qua đường “*trunk*” này. Giao thức được sử dụng là 802.1Q (dot1q).

- **Giao thức 802.1Q**

Đây là giao thức chuẩn của IEEE để dành cho việc nhận dạng các VLAN bằng cách thêm vào “frame header” đặc điểm của một VLAN. Phương thức này còn được gọi là gắn thẻ cho VLAN (frame tagging).



- **Cấu hình VLAN trunking:**

Để cấu hình đường “trunk”, chúng ta cấu hình 2 cổng “trunk” như sau:

```
switch(config)#interface <interface>
switch(config-if)#switchport mode trunk
switch(config-if)#switchport trunk encapsulation dot1q
```

Lệnh cuối cùng là mặc định ở một số dòng switch

6. VLAN Trunking Protocol (VTP)

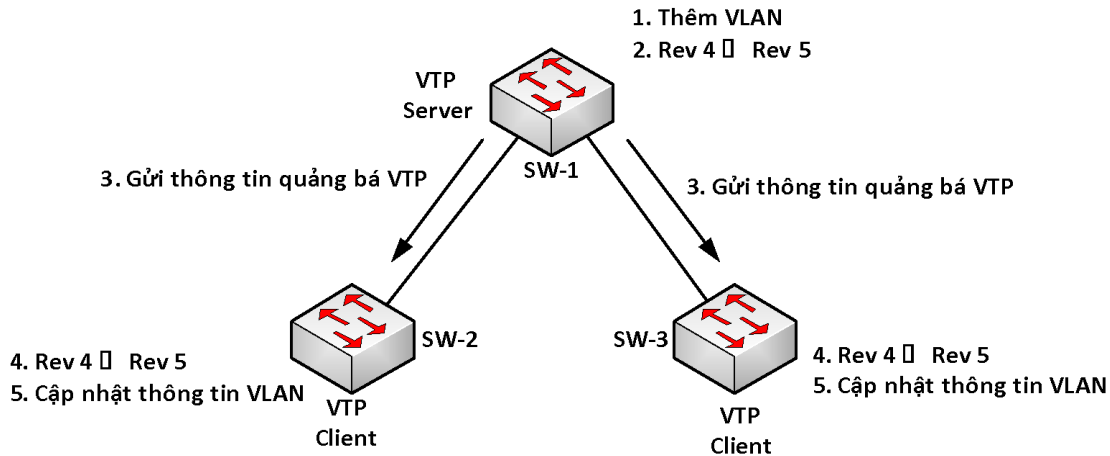
VTP là giao thức hoạt động ở tầng liên kết dữ liệu trong mô hình OSI. VTP giúp cho việc cấu hình VLAN luôn đồng nhất khi thêm, xóa, sửa thông tin về VLAN trong hệ thống mạng.

- **Hoạt động của VTP**

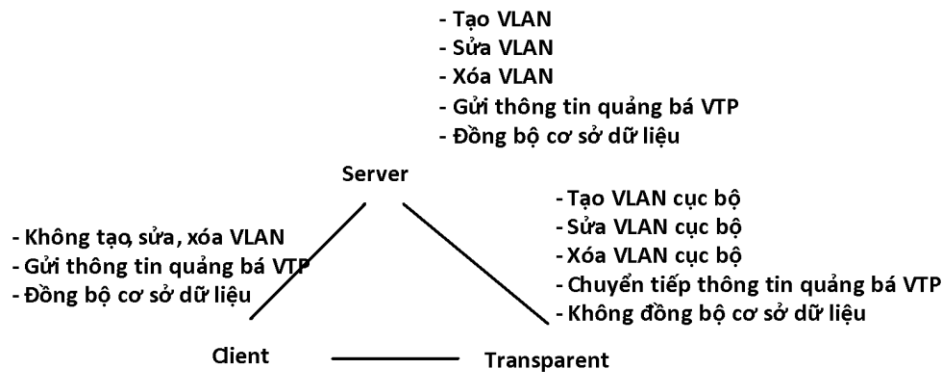
VTP gửi thông điệp quảng bá qua “VTP domain” mỗi 5 phút một lần, hoặc khi có sự thay đổi xảy ra trong cấu hình VLAN. Một thông điệp VTP bao gồm “*revision-number*”, tên VLAN (VLAN name), số hiệu VLAN (VLAN number), và thông tin về các switch có cổng

gắn với mỗi VLAN. Bằng sự cấu hình *VTP Server* và việc quảng bá thông tin VTP, tất cả các switch đều đồng bộ về tên VLAN và số hiệu VLAN của tất cả các VLAN.

Một trong những thành phần quan trọng trong các thông tin quảng bá VTP là tham số “*revision number*”. Mỗi lần *VTP server* điều chỉnh thông tin VLAN, nó tăng “*revision-number*” lên 1, rồi sau đó *VTP Server* mới gửi thông tin quảng bá VTP đi. Khi một switch nhận một thông điệp VTP với “*revision-number*” lớn hơn, nó sẽ cập nhật cấu hình VLAN.



• VTP hoạt động ở một trong ba chế độ:



Switch ở chế độ *VTP server* có thể tạo, chỉnh sửa và xóa VLAN. *VTP server* lưu cấu hình VLAN trong NVRAM của nó. *VTP Server* gửi thông điệp ra tất cả các cổng “trunk”.

Switch ở chế độ *VTP client* không tạo, sửa và xóa thông tin VLAN. *VTP Client* có chức năng đáp ứng theo mọi sự thay đổi của VLAN từ *Server* và gửi thông điệp ra tất cả các cổng “trunk” của nó. *VTP Client* đồng bộ cấu hình VLAN trong hệ thống.

Switch ở chế độ *transparent* sẽ nhận và chuyển tiếp các thông điệp quảng bá VTP do các switch khác gửi đến mà không quan tâm đến nội dung của các thông điệp này. Nếu “*transparent switch*” nhận được thông tin cập nhật VTP nó cũng không cập nhật vào cơ sở dữ liệu của nó; đồng thời nếu cấu hình VLAN của nó có gì thay đổi, nó cũng không gửi thông tin cập nhật cho các switch khác. Trên “*transparent switch*” chỉ có một việc duy nhất là chuyển tiếp thông điệp VTP. Switch hoạt động ở “*transparent-mode*” chỉ có thể tạo ra các VLAN cục bộ. Các VLAN này sẽ không được quảng bá đến các switch khác.

• Cấu hình VTP

- Cấu hình VTP domain

```
Switch(config)#vtp domain <domain_name>
```

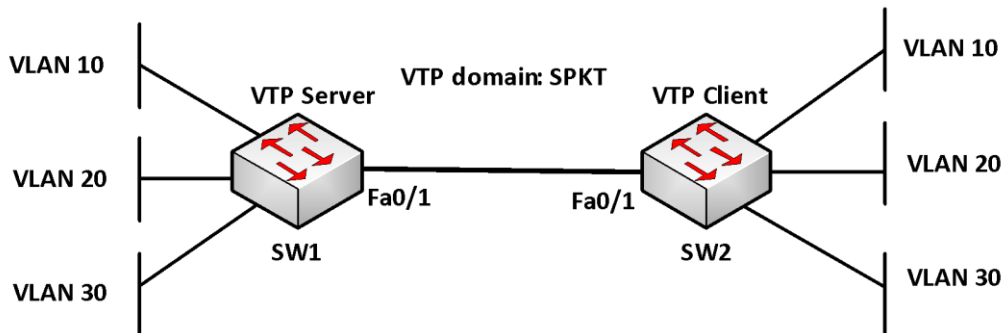
- Cấu hình VTP mode

```
Switch(config)#vtp [client| transparent| server]
```

- Lệnh xem cấu hình VTP

```
Switch#show vtp status
```

Ví dụ: Cho sơ đồ mạng



Mô tả

- ✓ Hai switch kết nối với nhau qua đường “trunk”.
- ✓ Tạo 3 vlan: VLAN 10, VLAN 20, VLAN 30 trên SW1
- ✓ Cấu hình VTP để các thông tin các VLAN trên SW1 cập nhật cho SW2
- ✓ Trên SW1: VLAN 10 (Fa0/2 – Fa0/4), VLAN 20 (Fa0/5 – Fa0/7), VLAN 30 (Fa0/8 – Fa0/10)
- ✓ Trên SW2: VLAN 10 (Fa0/4 – Fa0/6), VLAN 20 (Fa0/7 – Fa0/9), VLAN 30 (Fa0/10 – Fa0/12)

Các bước cấu hình

Cấu hình Sw1 làm VTP Server:

- ✓ Thiết lập VTP domain: SPKT, VTP mode Server, và tạo các VLAN

```
sw1#config terminal
sw1(config)#vtp mode server
sw1(config)#vtp domain SPKT
sw1(config)#vlan 10 name CNTT
sw1(config)#vlan 20 name TTTH
sw1(config)#vlan 30 name TTCLC
```

- ✓ Cấu hình đường trunk và cho phép tất cả các VLAN qua đường trunk

```
sw1(config)#interface f0/1
sw1(config-if)#switchport mode trunk
sw1(config-if)#switchport trunk encapsulation dot1q
```


7. Định tuyến giữa các VLAN

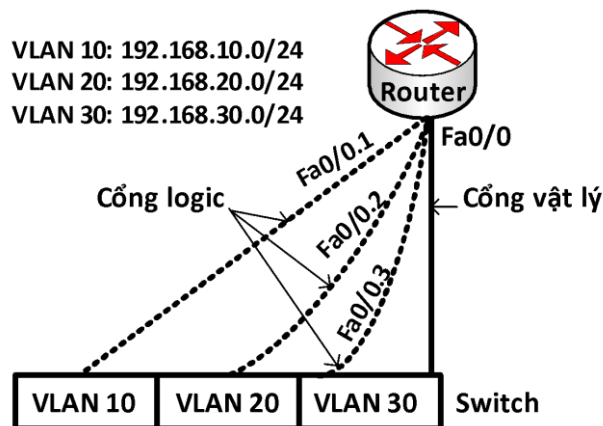
Mỗi VLAN là một miền quảng bá. Do đó, mỗi thiết bị trong VLAN chỉ liên lạc được với các thiết bị khác trong cùng một VLAN. Nếu một máy tính trong một VLAN muốn liên lạc với một máy tính thuộc một VLAN khác thì nó phải thông qua thiết bị định tuyến như là router.

Router trong cấu trúc VLAN thực hiện ngăn chặn quảng bá, bảo mật và quản lý các lưu lượng mạng. Switch layer 2 không thể chuyển dữ liệu giữa các VLAN với nhau. Dữ liệu trao đổi giữa các VLAN phải được định tuyến qua thiết bị hoạt động ở tầng mạng như router.

Giả sử trên switch tạo 3 VLAN, nếu ta dùng 3 cổng của router để định tuyến cho 3 VLAN này thì quá cồng kềnh và không tiết kiệm. Ta chỉ cần sử dụng 1 cổng trên router kết nối với một cổng trên switch và cấu hình đường này làm đường *trunk* (trunk layer 3) để định tuyến cho các VLAN.

Đường kết nối cho phép mang lưu lượng của nhiều VLAN gọi là kết nối *trunk lớp 3*. Nó không phải là của riêng VLAN nào. Ta có thể cấu hình một đường *trunk* để vận chuyển lưu thông cho tất cả VLAN hoặc một số VLAN cụ thể nào đó được chỉ ra trong cấu hình. *Trunking layer 3* đòi hỏi cổng trên VLAN phải có thể hoạt động ở tốc độ FastEthernet trở lên.

❖ Cổng vật lý và cổng logic



Đường “*trunk*” có ưu điểm là làm giảm số lượng cổng cần sử dụng của router và switch. Điều này không chỉ tiết kiệm chi phí mà còn giúp cho cấu hình bớt phức tạp. Kết nối “*trunk*” trên router có khả năng mở rộng với số lượng lớn VLAN. Nếu mỗi VLAN phải có một kết nối vật lý thì không thể đáp ứng được khi số lượng VLAN lớn.

Một cổng vật lý có thể được chia thành nhiều cổng luận lý. Mỗi cổng luận lý tương ứng với một VLAN và được đặt một địa chỉ IP của vlan đó. Mỗi VLAN là một mạng riêng, do đó cổng luận lý thuộc VLAN nào thì có địa chỉ IP thuộc mạng của VLAN đó.

❖ Cấu hình định tuyến cho các VLAN dùng Router

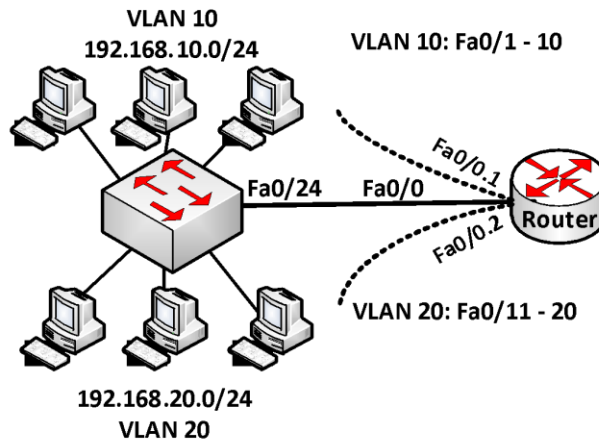
Sử dụng các cổng luận lý được chia từ một cổng vật lý để cấu hình định tuyến giữa các VLAN, các câu lệnh được sử dụng như sau:

```
R(config)#interface <interface.subintf-number>
```

```
R(config-if)#encapsulation dot1q <vlan-id>
```

```
R(config-if)#ip address <address> <subnet-mask>
```

Ví dụ: Cấu hình định tuyến giữa các VLAN



Yêu cầu

- Tạo 2 vlan: **VLAN 10** (P.KinhDoanh) và **VLAN 20** (P.KeToan)
- Các cổng Fa0/1–Fa0/10 thuộc VLAN 10, các cổng Fa0/11–Fa0/20 thuộc VLAN 20
- Cấu hình định tuyến cho phép hai VLAN này có thể liên lạc được với nhau.

Các bước thực hiện

• Cấu hình trên switch

✓ Tạo vlan

```
switch(config)#vlan 10
switch(config-vlan)#name P.KinhDoanh
switch(vlan)#vlan 20
switch(config-vlan)#name P.KeToan
```

✓ Gán các port vào vlan

```
switch(config)#interface range fa0/1 - 10
switch(config-if-range)#switchport mode access
switch(config-if-range)#switchport access vlan 10
switch(config)#int fa0/11 - 20
switch(config-if-range)#switchport mode access
switch(config-if-range)#switchport access vlan 20
```

✓ Cấu hình đường trunk

```
switch(config)#int fa0/24
switch(config-if)#switchport mode trunk
switch(config-if)#switchport trunk encapsulation dot1q
```

Lưu ý: Lệnh cuối là mặc định trên một số dòng switch.

• Cấu hình trên router

- ✓ Chọn cổng fa0/0 để cấu hình trunk

```
router(config)#interface fa0/0
router(config-if)#no shutdown
```

- ✓ Kích hoạt *trunk* trên subinterface fa0/0.1 và đóng gói bằng dot1q

```
router(config)#int fa0/0.1
router(config-if)#encapsulation dot1q 10
```

- ✓ Cấu hình thông tin lớp 3 cho sub-interface **fa0/0.1**

```
router(config-subif)#ip address 192.168.1.1 255.255.255.0
```

- ✓ Kích hoạt “trunk” trên sub-interface **fa0/0.2** và đóng gói bằng **dot1q**

```
router(config)#int fa0/0.2
router(config-subif)#encapsulation dot1q 20
```

- ✓ Cấu hình thông tin lớp 3 cho sub-interface **fa0/0.2**

```
router(config-subif)#ip address 192.168.2.1 255.255.255.0
```

- ✓ Lưu cấu hình

```
router#copy run start
```

- **Kiểm tra cấu hình**

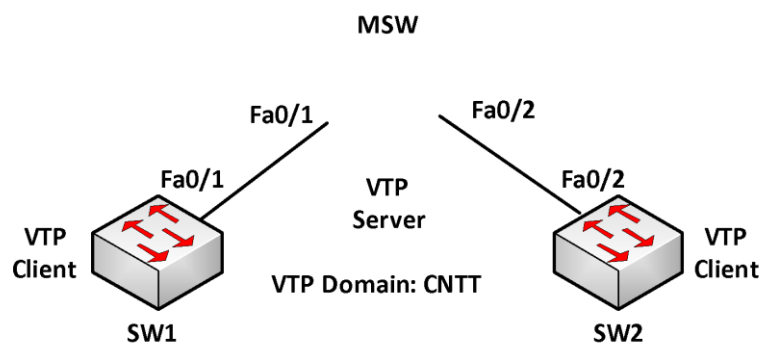
Trên switch dùng các lệnh sau:

```
Switch#show interface interface
Switch#show vlan
Switch#show vtp status
```

Trên router dùng các lệnh sau

```
Router#show vlan: thông tin layer 2 và layer 3 cấu hình cho mỗi VLAN.
Router#show interfaces <interface>
```

- ❖ Định tuyến cho các VLAN dùng switch layer 3 (MSW)



VLAN10: 192.168.10.0/24

VLAN20: 192.168.20.0/24

VLAN30: 192.168.30.0/24

VLAN40: 192.168.40.0/24

Yêu cầu

- Cấu hình đường “trunk”
- Cấu hình VTP, VLAN
VTP domain: CNTT; MSW: VTP Server; SW1, SW2: VTP Client
- Cấu hình MSW để định tuyến cho 4 VLAN

Hướng dẫn cấu hình

❖ Cấu hình trunk

```
SW1(config)#interface fa0/1
SW1(config-if)#switchport mode trunk
SW1(config-if)#switchport trunk encapsulation dot1q
SW2(config)#interface fa0/2
SW2(config-if)#switchport mode trunk
SW2(config-if)#switchport trunk encapsulation dot1q
MSW(config)#interface fa0/1
MSW(config-if)#switchport mode trunk
MSW(config-if)#switchport trunk encapsulation dot1q
MSW(config)#interface fa0/2
MSW(config-if)#switchport mode trunk
MSW(config-if)#switchport trunk encapsulation dot1q
```

❖ Cấu hình VTP, VLAN

```
MSW(config)#vtp domain CNTT
MSW(config)#vtp mode server
SW1(config)#vtp domain CNTT
SW1(config)#vtp mode client
SW2(config)#vtp domain CNTT
SW2(config)#vtp mode client
MSW(config)#vlan 10
MSW(config)#vlan 20
MSW(config)#vlan 30
MSW(config)#vlan 40
```

❖ Cấu hình MSW để routing giữa 4 VLAN

```
MSW(config)#ip routing
MSW(config)#interface vlan 10
MSW(config-if)#ip address 192.168.10.1 255.255.255.0
```

```
MSW(config)#interface vlan 20
MSW(config-if)#ip address 192.168.20.1 255.255.255.0
MSW(config)#interface vlan 30
MSW(config-if)#ip address 192.168.30.1 255.255.255.0
MSW(config)#interface vlan 40
MSW(config-if)#ip address 192.168.40.1 255.255.255.0
```

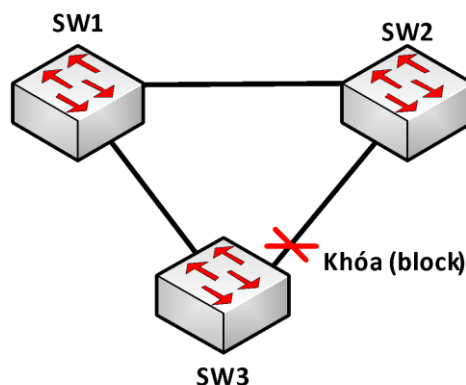
❖ Kiểm tra cấu hình

```
show interface trunk
show vtp status
show vlan brief
show ip route
```

8. Giao thức STP

Trong thiết kế mạng, việc tạo ra các kết nối dư thừa là cần thiết nhằm tạo khả năng dự phòng cho hệ thống. Tuy nhiên, khi thiết kế dự phòng trên Switch thì có 3 vấn đề cần xem xét là: bão quảng bá, nhiều gói tin được nhận giống nhau và bảng địa chỉ MAC trên các Switch không ổn định. Có thể gọi chung trường hợp này là “switching loop”.

Giao thức STP được sử dụng để giải quyết vấn đề này bằng cách khóa tạm thời một hoặc một số cổng để tránh tình trạng như trên.



❖ Hoạt động của STP qua các bước sau:

- o Bầu chọn 1 switch làm “Root switch” còn gọi là “Root bridge”
- o Chọn “Root port” trên các switch còn lại
- o Chọn “Designated port” trên mỗi phân đoạn (segment) mạng
- o Cổng còn lại gọi là “Nondesignated port” sẽ bị khóa

❖ Quá trình bầu chọn “root switch”

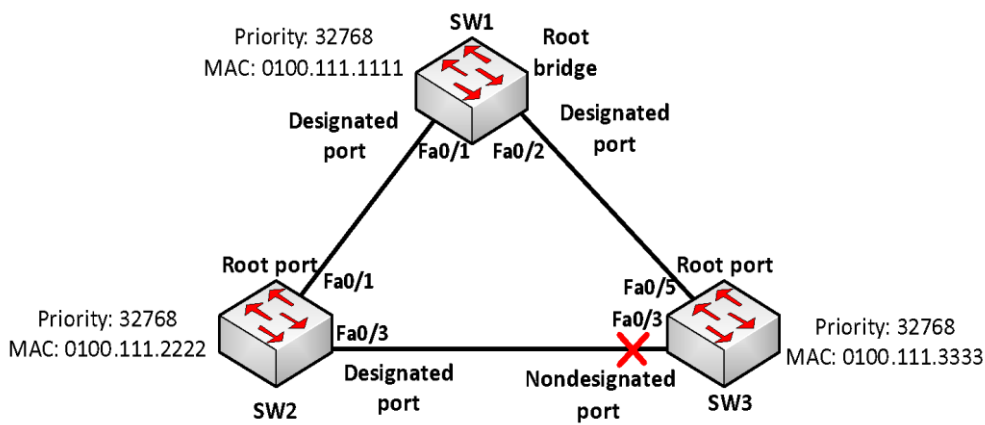
Mỗi switch có một giá trị “Bridge-ID” gồm 2 trường là “Bridge priority” và “MAC address” và được đặt vào trong BPDU và gửi quảng bá cho các switch khác mỗi 2 giây. Switch được chọn làm “root switch” là switch có giá trị “Bridge-ID” nhỏ nhất. Để so sánh,

giá trị “*Bridge priority*” được dùng để so sánh trước, nếu tất cả các switch đều có giá trị này bằng nhau thì tham số thứ 2 là “*MAC address*” sẽ được dùng để so sánh.

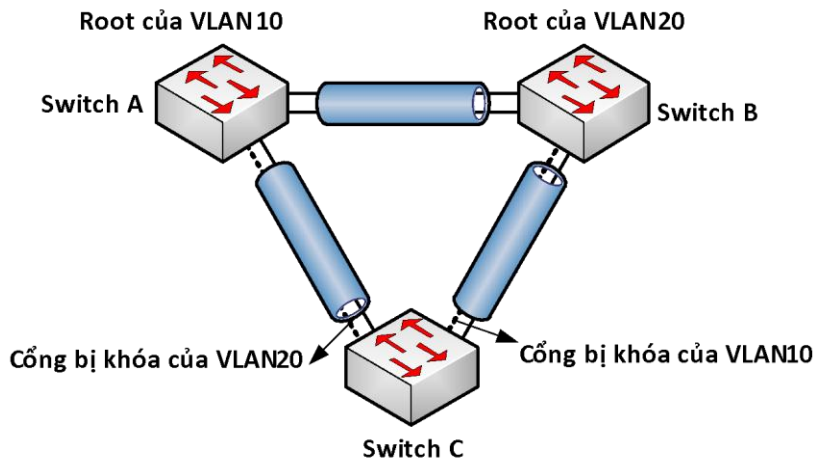
Các loại cổng khác “*root port*”, “*designated port*” sẽ lần lượt được bầu chọn dựa vào chi phí nhỏ nhất tính từ nó đến “*root switch*”. Dựa vào bảng sau để tính chi phí cho mỗi chặn.

Tốc độ kết nối	Chi phí (Cost)
10 Gb/s	2
1 Gb/s	4
100 Mb/s	19
10 Mb/s	100

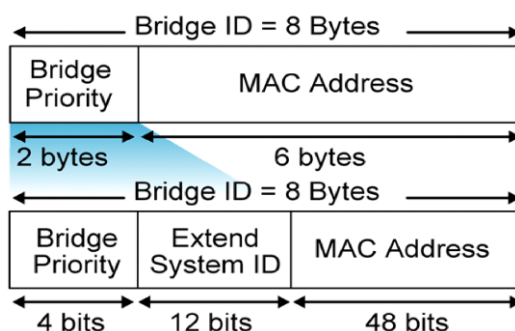
Ví dụ:



Một số dạng STP được cải tiến như: PVSTP+ (Per VLAN Spanning Tree Plus) dùng tạo cho mỗi VLAN một STP riêng.



Trong PVSTP+, *Bridge-ID* có thêm trường *System-ID* (VLAN-ID) để phân biệt cho từng VLAN.



Một số cải tiến khác như RSTP (Rapid Spanning Tree Protocol), MSTP.

Một số lệnh cấu hình để điều chỉnh giá trị “Bridge priority” mặc định của switch. Chọn switch làm “root switch” bằng lệnh sau:

```
Switch(config)#spanning-tree vlan <vlan-id> root primary
```

Hoặc

```
Switch(config)#spanning-tree vlan <vlan-id> priority <priority>
```

9. EtherChannel

Công nghệ EtherChannel của Cisco cho phép kết hợp các kết nối Ethernet thành một bó (bundle) để tăng băng thông. Mỗi bundle có thể bao gồm từ hai đến tám kết nối Fast Ethernet hay Gigabit Ethernet, tạo thành một kết nối luận lý gọi là FastEtherChannel hay Gigabit EtherChannel. Kết nối này cung cấp một băng thông lên đến 1600Mbps hoặc 16 Gbps.

Công nghệ này được xem là một cách đơn giản để nâng cấp kết nối giữa các switch mà không cần phải mua phần cứng mới. Ví dụ, một kết nối Fast Ethernet (có throughput là 200Mbps) có thể mở rộng lên đến 8 kết nối FE (1600Mbps) để trở thành một kết nối FastEtherChannel. Nếu lưu lượng lưu lượng tăng quá mức này, quá trình nâng cấp có thể lại bắt đầu với một kết nối Gigabit Ethernet. Sau đó, ta có thể lại tiếp tục mở rộng kết nối này lên thành GigabitEtherChannel. Quá trình này có thể được lặp lại với việc tiếp tục chuyển sang kết nối 10Gbps. Bình thường, việc có nhiều kết nối giữa các switch tạo ra khả năng bị bridging loops. EtherChannel sẽ tránh tình huống này bằng cách xem cả một bundle như là một kết nối đơn duy nhất, hoặc là access, hoặc là trunk.

Kết hợp cổng bên trong EtherChannel

EtherChannel có thể bao gồm tối đa tám kết nối vật lý của cùng kiểu phần cứng và cùng tốc độ. Một vài ràng buộc phải được đáp ứng sao cho chỉ có những kết nối tương tự là được kết hợp. Thông thường, tất cả các cổng phải thuộc về cùng một vlan. Nếu được dùng như một kết nối trunk, tất cả các cổng phải ở trong trunking, có cùng native vlan và truyền cùng một tập hợp của vlan. Mỗi cổng phải có cùng tốc độ, duplex và có cùng cấu hình spanning tree.

Các giao thức của EtherChannel: PagP và LACP.

Cấu hình EtherChannel

Các lệnh cơ bản để cấu hình Etherchannel. Cấu hình PAGP Ethechannel:

```
Switch(config-if)#channel-protocol pagp
Switch(config-if)#channel-group number mode {on | auto |
desirable }
```

Các chế độ này có ý nghĩa như sau:

ON: ở mode này thì Switch tự động enable etherchannel tuy nhiên nó lại không gửi hay nhận bất kỳ gói PAGP nào, do đó mà phải cấu hình on mode ở hai đầu

Auto: Switch sẽ tự động enable ethechannel nếu nó nhận được PAGP packet.

Desirable: Switch sẽ tự động cố gắng yêu cầu đầu kia chuyển kết nối sang thành EtherChannel.

Cấu hình LACP

```
Switch(config)#lacp system-priority priority
Switch(config-if)#channel-protocol lacp
Switch(config-if)#channel-group number mode {on|passive|active}
Switch(config-if)# lacp port-priority priority
```

Lệnh đầu tiên để xác định system priority để xác định Switch nào làm Switch điều khiển Ethechannel, hoặc nếu Priority bằng nhau thì Switch nào có địa chỉ mac nhỏ hơn sẽ được chọn. Ta còn xác định priority của cổng để xác định xem cổng nào là active và cổng nào ở trạng thái standby. Cổng có priority nhỏ sẽ active và, lớn sẽ ở trạng thái standby. Các mode trong lệnh channel-group On, Passive, active tuân tự tương tự như On, Auto , Desirable trong PAGP

Khi các cổng được cấu hình như là thành viên của EtherChannel, switch sẽ tự động tạo ra các cổng EtherChanel. Interface này sẽ đại diện cho cả bundle

```
Switch(config)# interface type mod/num
Switch(config-if)# channel-protocol pagp
Switch(config-if)# channel-group number mode {on | {auto | desirable}}
```

10. Tổng kết chương

Trong môi trường Ethernet LAN, tập hợp các thiết bị cùng nhận một gói quảng bá bởi bất kỳ một thiết bị còn lại được gọi là một “*broadcast domain*”. Trên các switch không hỗ trợ VLAN, switch sẽ gửi tất cả các gói tin quảng bá ra tất cả các cổng, ngoại trừ cổng mà nó nhận gói tin vào. Kết quả là trên các cổng của loại switch này là cùng một “*broadcast domain*”. Nếu switch này kết nối đến các switch và các hub khác, các cổng trên switch này sẽ cùng “*broadcast domain*”.

VLAN cho phép kết hợp các cổng trên switch thành các nhóm để giảm lưu lượng broadcast. VLAN là một LAN theo logic dựa trên chức năng, ứng dụng của một tổ chức chứ

không phụ thuộc vào vị trí vật lý hay kết nối vật lý trong mạng. Một VLAN là một miền quảng bá được tạo nên bởi một hay nhiều switch.

Giao thức VTP có vai trò duy trì cấu hình của VLAN và đồng nhất trên toàn mạng. VTP là giao thức sử dụng đường trunk để quản lý sự thêm, xóa, sửa các VLAN trên toàn mạng từ switch trung tâm được đặt trong *Server mode*. VTP hoạt động chủ yếu là đồng nhất các thông tin VLAN trong cùng một VTP domain giúp giảm đi sự cấu hình giống nhau trong các switch.

Kết nối *trunk* là liên kết Point-to-Point giữa các cổng trên switch với router hoặc với switch khác. Kết nối *trunk* sẽ vận chuyển thông tin của nhiều VLAN thông qua một liên kết đơn và cho phép mở rộng VLAN trên hệ thống mạng. Các VLAN được định tuyến sử dụng thiết bị ở tầng 3 như router hay “Switch layer 3”.

Giao thức STP được dùng trong trường hợp hệ thống mạng thiết kế các kết nối dự phòng trên Switch. STP chống tình trạng “switching loop” bằng cách khóa tạm một số cổng trong mạng. Một số phiên bản cải tiến từ STP truyền thống như PVSTP+, RSTP,...

11. Câu hỏi và bài tập

11.1. Một VLAN là một tập các thiết bị nằm cùng miền _____.

- A. Autonomous system
- B. Broadcast domain
- C. Bandwidth domain
- D. Collision domain

11.2. Thiết bị nào sau đây được dùng để kết nối các VLAN?

- A. Switch
- B. Bridge
- C. Router
- D. Hub

11.3. Giao thức nào sau đây được dùng để phân phối thông tin về cấu hình VLAN đến các Switch khác trong mạng?

- A. STP
- B. VTP
- C. EIGRP
- D. SNMP
- E. CDP

11.4. Giao thức STP (Spanning-Tree Protocol) dùng để làm gì?

- A. Dùng để cập nhật định tuyến trong môi trường Switch.
- B. Dùng để chống "routing loop" trong mạng
- C. Dùng để tránh "switching loop" trong mạng
- D. Dùng để quản lý việc thêm, xóa, sửa thông tin VLAN trong hệ thống có nhiều Switch.

E. Dùng để phân hoạch mạng thành nhiều miền đưng độ

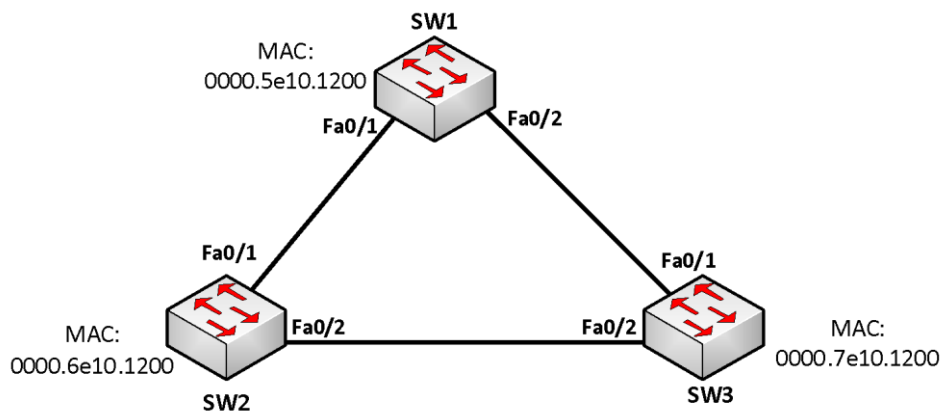
11.5. Để kiểm tra interface fa0/5 có được gán cho VLAN Sales không, thì ta sử dụng lệnh nào sau đây?

- A. show vlan
- B. show mac-address-table
- C. show vtp status
- D. show spanning-tree root
- E. show ip interface brief

11.6. Tại sao Switch không bao giờ học một địa chỉ "broadcast"?

- A. Frame broadcast không bao giờ được gửi tới Switch
- B. Địa chỉ broadcast sử dụng định dạng không đúng trong bảng chuyển mạch trên Switch
- C. Địa chỉ broadcast không bao giờ là địa chỉ nguồn trong một frame.
- D. Địa chỉ broadcast chỉ dùng trong layer 3
- E. Switch không bao giờ chuyển tiếp các gói tin broadcast

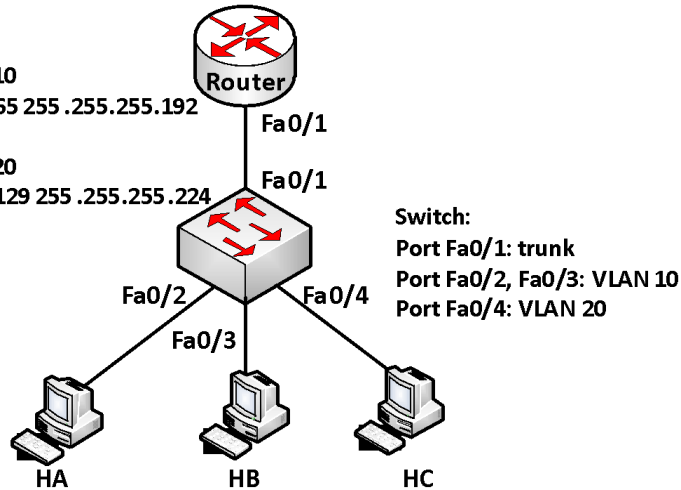
11.7. Cho mô hình mạng:



Tất cả các switch được cấu hình STP mặc định và tất cả các kết nối qua port FastEthernet. Port nào sẽ chuyển vào trạng thái "blocking"?

- A. Switch SW1 - Port Fa0/1
- B. Switch SW1 - Port Fa0/2
- C. Switch SW2 - Port Fa0/2
- D. Switch SW2 - Port Fa0/1
- E. Switch SW3 - Port Fa0/1
- F. Switch SW3 - Port Fa0/2

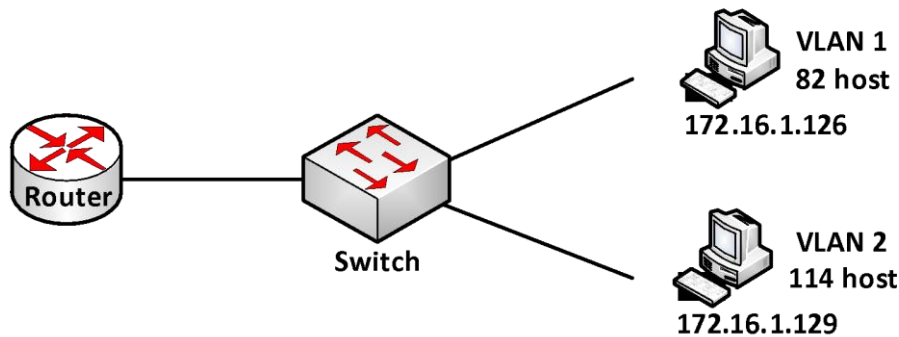
11.8. Cho mô hình mạng:

Router:**Interface fa0/1.1****encapsulation dot1q 10****ip address 192.168.1.65 255.255.192****Interface fa0/1.2****encapsulation dot1q 20****ip address 192.168.1.129 255.255.255.224**

Những thông tin cấu hình nào sau đây là đúng cho các host trong mô hình trên?

- A. Địa chỉ IP của HA: 192.1.1.65
- B. Subnet mask của HA: 255.255.255.224
- C. Địa chỉ IP của HB: 192.1.1.125
- D. Default gateway của HB: 192.1.1.65
- E. Địa chỉ IP của HC: 192.1.1.66
- F. Subnet mask của HC: 255.255.255.224

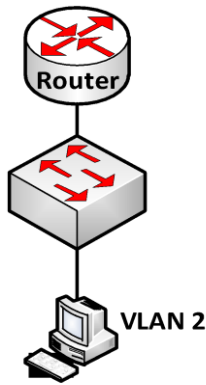
11.9. Cho mô hình mạng:



Những phát biểu nào sau đây là đúng trong mô hình mạng trên?

- A. Subnet mask được sử dụng là 255.255.255.192
- B. Subnet mask được sử dụng là 255.255.255.128
- C. Địa chỉ IP 172.16.1.25 có thể được gán cho các host thuộc VLAN1
- D. Địa chỉ IP 172.16.1.205 có thể được gán cho các host thuộc VLAN1
- E. Cổng LAN trên router được cấu hình với một địa chỉ IP
- F. Cổng LAN trên router được cấu hình với nhiều địa chỉ IP

11.10. Cho mô hình mạng:



```

R(config)#interface fastethernet 0/1.1
R(config-if)#encapsulation dot1q 1
R(config-if)#ip address 192.168.1.1 255.255.255.0
R(config)#interface fastethernet 0/1.2
R(config-if)#encapsulation dot1q 2
R(config-if)#Ip address 192.168.2.1 255.255.255.0
R(config)#interface fastethernet 0/1.3
R(config-if)#encapsulation dot1q 3
R(config-if)# Ip address 192.168.3.1 255.255.255.0
  
```

Router trong mô hình mạng được cấu hình như trên. Switch kết nối với router qua đường *trunk*. Trên Switch cấu hình 3 VLAN: VLAN1, VLAN2, and VLAN3. Một máy tính A kết nối vào VLAN2. Hỏi địa chỉ **default gateway** phải đặt cho máy tính này là địa chỉ nào sau đây?

- A. 192.168.1.1
- B. 192.168.1.2
- C. 192.168.2.1
- D. 192.168.2.2
- E. 192.168.3.1
- F. 192.168.3.2

11.11. Hai tham số được STP sử dụng để bầu chọn “root bridge”?

- A. Bridge priority
- B. Địa chỉ IP
- C. Địa chỉ MAC
- D. Phiên bản IOS
- E. Dung lượng RAM
- F. Tốc độ kết nối

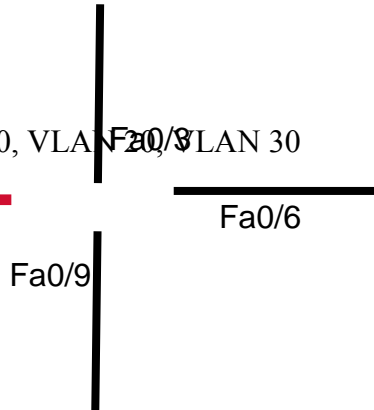
12. Lab. SWITCH

Lab 4-1.

VLAN

❖ Mô tả

- Cấu hình VLAN trên Switch
- Cấu hình 3 VLAN: VLAN 10, VLAN 20, VLAN 30
- F0/1 – f0/6 :vlan 10
- F0/7 – f0/9 :vlan 20
- F0/10 – f0/12 : vlan 30



❖ Các bước thực hiện:

• Tạo vlan:

```
Switch>enable
Switch#config terminal
Switch(config)#vlan 10
Switch(config)#vlan 20
Switch(config)#vlan 30
```

• Gán các port vào vlan

```
Switch(config)#interface f0/1
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 10
Switch(config-if)#exit
```

```
Switch(config)#interface f0/2
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 10
Switch(config-if)#exit
```

```
Switch(config)#interface f0/3
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 10
Switch(config-if)#exit
```

```
Switch(config)#interface f0/4
```

```
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 10
Switch(config-if)#exit
```

```
Switch(config)#interface f0/5
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 10
Switch(config-if)#exit
```

```
Switch(config)#interface f0/6
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 10
Switch(config-if)#exit
```

```
Switch(config)#interface f0/7
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 20
Switch(config-if)#exit
```

```
Switch(config)#interface f0/8
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 20
Switch(config-if)#exit
```

```
Switch(config)#interface f0/9
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 20
Switch(config-if)#exit
```

```
Switch(config)#interface f0/10
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 30
Switch(config-if)#exit
```

```
Switch(config)#interface f0/11
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 30
```



```
Switch(config-if)#exit
```

```
Switch(config)#interface f0/12
```

```
Switch(config-if)#switchport mode access
```

```
Switch(config-if)#switchport access vlan 30
```

```
Switch(config-if)#exit
```

❖ Kiểm tra

Thực hiện các câu lệnh sau để kiểm tra cấu hình

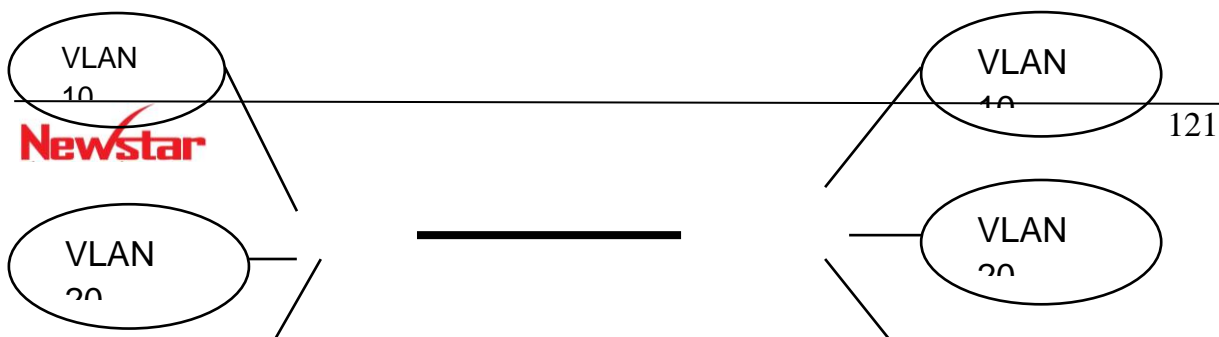
```
Switch#show run
```

```
Switch#show vlan
```

Gắn PC vào các port như trên sơ đồ. Đặt IP cho các PC và dùng lệnh ping : kiểm tra kết nối.

Lab 4-2.

VLAN TRUNKING



❖ **Mô tả**

- Hai switch kết nối với nhau qua đường trunk.
- Tạo 3 vlan: VLAN 10, VLAN 20, VLAN 30

❖ **Các bước cấu hình**

Cấu hình Sw1 làm VTP Server:

- ✓ **Đặt hostname, mật khẩu trên Sw1**

```
switch>enable
switch#config terminal
switch(config)#hostname sw1
sw1(config)#enable password cisco
```

- ✓ **Thiết lập VTP domain: SPKT, VTP mode server, và tạo các vlan**

```
sw1#config terminal
sw1(config)#vtp mode server
sw1(config)#vtp domain SPKT
sw1(config)#vlan 10 name CNTT
sw1(config)#vlan 20 name TTTH
sw1(config)#vlan 30 name TTCLC
```

- ✓ **Cấu hình đường trunk và cho phép tất cả các vlan qua đường trunk**

```
sw1#config terminal
sw1(config)#interface f0/1
sw1(config-if)#switchport mode trunk
sw1(config-if)#switchport trunk encapsulation dot1q
sw1(config-if)#switchport trunk allowed vlan all (mặc định)
sw1(config-if)#exit
sw1(config)#
```

- ✓ **Gán các port vào các vlan**

```
sw1(config)#int range f0/2 - 4
sw1(config-...)#switchport mode access
sw1(config-...)#switchport access vlan 10
sw1(config-if)#int range f0/5 - 7
sw1(config-...)#switchport mode access
sw1(config-...)#switchport access vlan 20
sw1(config-if)#int range f0/8 - 10
sw1(config-...)#switchport mode access
sw1(config-...)#switchport access vlan 30
```

- ✓ **Kiểm tra cấu hình**

Sử dụng các lệnh : switch#show vlan
 switch# show vtp status

Cấu hình Sw2 làm VTP client:

- ✓ **Cấu hình hostname, password**

```
switch#config terminal
switch(config)#hostname SW2
SW2(config)#enable password cisco
```

✓ **Cấu hình vtp domain: SPKT, vtp mode: client**

```
SW2#config terminal
SW2(config)#vtp domain SPKT
SW2(config)#vtp mode client
SW2(config)#exit
```

✓ **Cấu hình trunking trên cổng f0/1 của SW2**

```
SW2#config terminal
SW2(config)#int f0/1
SW2(config-if)#switchport mode trunk
SW2(config-if)#switchport trunk encapsulation dot1q
SW2(config-if)#switchport trunk allowed vlan all
SW2(config-if)#exit
```

✓ **Gán các port vào các vlan**

```
sw2(config)#int f0/4
sw2(config-if)#switchport mode access
sw2(config-if)#switchport access vlan 10
sw2(config-if)#int f0/5
sw2(config-if)#switchport mode access
sw2(config-if)#switchport access vlan 10
sw2(config-if)#int f0/6
sw2(config-if)#switchport mode access
sw2(config-if)#switchport access vlan 10
sw2(config)#int f0/7
sw2(config-if)#switchport mode access
sw2(config-if)#switchport access vlan 20
sw2(config-if)#int f0/8
sw2(config-if)#switchport mode access
sw2(config-if)#switchport access vlan 20
sw2(config-if)#int f0/9
sw2(config-if)#switchport mode access
sw2(config-if)#switchport access vlan 20
sw2(config)#int f0/10
sw2(config-if)#switchport mode access
sw2(config-if)#switchport access vlan 30
sw2(config-if)#int f0/11
sw2(config-if)#switchport mode access
sw2(config-if)#switchport access vlan 30
sw2(config-if)#int f0/12
```

```
sw2(config-if)#switchport mode access  
sw2(config-if)#switchport access vlan 30
```

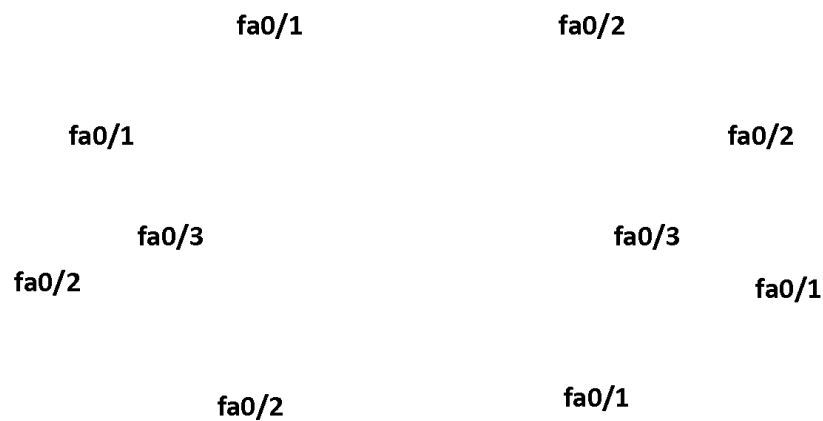
✓ Kiểm tra

Sử dụng các câu lệnh sau

```
switch#show vlan  
switch#show int interface  
switch#show vtp status  
switch#show vtp counters : kiểm tra số lần gửi và nhận thông tin trunking
```

Lab 4-3. Traditional Spanning Tree Protocol - 802.1D

Topology



Yêu cầu

- Xác định Root Bridge, Root port, Designated port, Non DP
- Cấu hình SW1: priority 4096
SW2: 8192
SW3: 28672
SW4: 36864
- Xác định Root Bridge, Root port, Designated port, Non DP
- Cấu hình portfast

Cấu hình

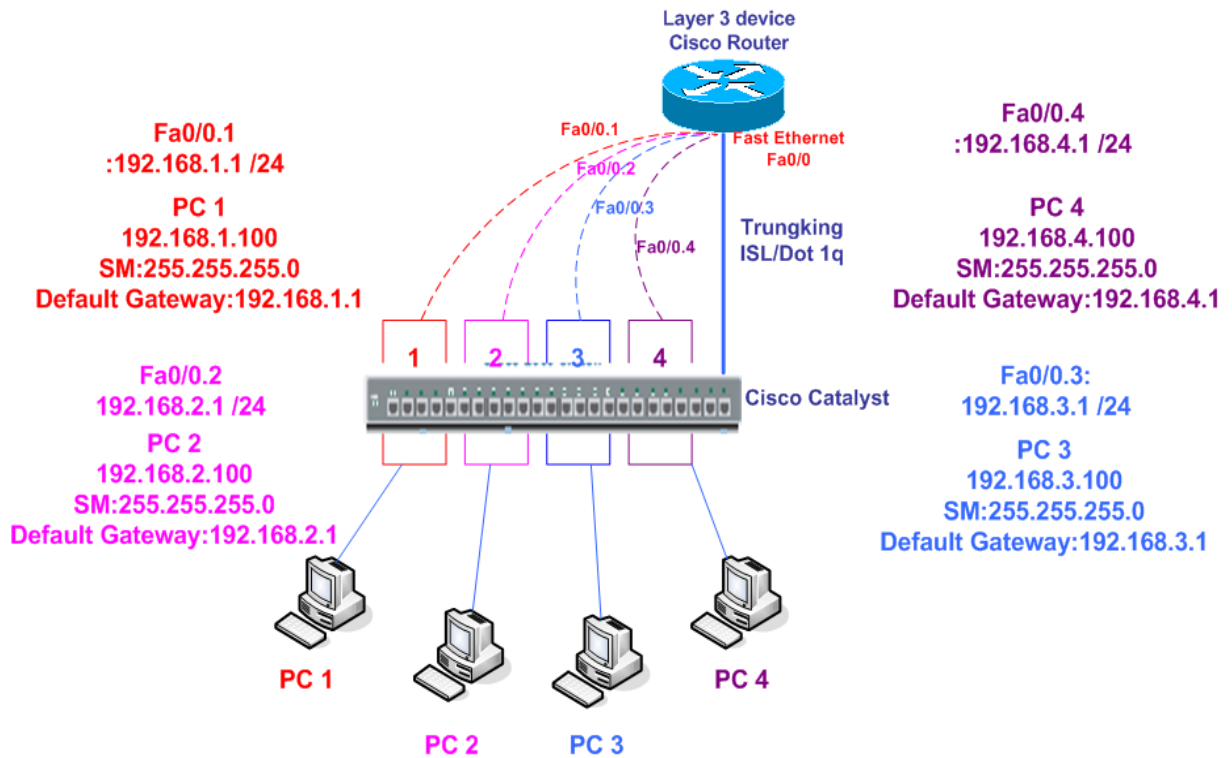
- Cấu hình priority:
SW1 (config) #spanning-tree vlan 1 priority 4096
SW2 (config) #spanning-tree vlan 1 priority 8192
SW3 (config) #spanning-tree vlan 1 priority 28672
SW4 (config) #spanning-tree vlan 1 priority 36864
- Cấu hình portfast:
SW1 (config) #interface range fa0/1 - 24
SW1 (config-range-if) #spanning-tree portfast

Kiểm tra cấu hình

```
show spanning-tree vlan 1
```

Lab 4-4 ĐỊNH TUYẾN GIỮA CÁC VLAN

(Inter-VLANs routing)



❖ Yêu cầu

- Tạo 4 VLAN : VLA, VLB, VLC, VLD
- Gán các port vào các vlan như sau: **VLA** (Fa0/1 – Fa0/3), **VLB** (Fa0/4 – Fa0/6), **VLC** (Fa0/7 – Fa0/9) , **VLD** (Fa0/10 – Fa0/12)
- PC1 thuộc **VLA**, PC2 thuộc **VLB**, PC3 thuộc **VLC**, và PC4 thuộc **VLD**.
- Cấu hình trunking cho phép các host thuộc các VLAN khác nhau có thể liên lạc được với nhau.

❖ Các bước thực hiện

- Cấu hình trên switch

✓ Tạo vlan

```
Switch#config terminal
Switch(config)#vlan 10
Switch(config-vlan)#name VLA
Switch(config-vlan)#exit
Switch(config)#vlan 20
Switch(config-vlan)#name VLB
Switch(config-vlan)#exit
Switch(config)#vlan 30
Switch(config-vlan)#name VLC
Switch(config-vlan)#exit
Switch(config)#vlan 40
```

```
Switch(config-vlan)#name VLD
Switch(config-vlan)#exit
```

✓ Kiểm tra cấu hình VLAN

```
Switch#show vlan
```

✓ Gán các port cho VLAN tương ứng

```
Switch(config)#interface fa0/2
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 10
Switch(config)#interface fa0/3
Switch(config-if)#switchport access vlan 10
Switch(config)#interface fa0/4
Switch(config-if)#switchport access vlan 20
Switch(config)#interface fa0/5
Switch(config-if)#switchport access vlan 20
Switch(config)#interface fa0/6
Switch(config-if)#switchport access vlan 20
Switch(config)#interface fa0/7
Switch(config-if)#switchport access vlan 30
Switch(config)#interface fa0/8
Switch(config-if)#switchport access vlan 30
Switch(config)#interface fa0/9
Switch(config-if)#switchport access vlan 30
Switch(config)#interface fa0/10
Switch(config-if)#switchport access vlan 40
Switch(config)#interface fa0/11
Switch(config-if)#switchport access vlan 40
Switch(config)#interface fa0/12
Switch(config-if)#switchport access vlan 40
```

Kích hoạt trunking trên cổng fa0/1, encapsulation trunking bằng dot1q, cấu hình cho phép các vlan lưu thông qua kết nối trunk.

```
Switch(config)#int fa0/1
Switch(config-if)#switchport mode trunk
Switch(config-if)#switchport trunk encapsulation dot1q
--> (mặc định trên sw 2950, 2960)
Switch(config-if)#switchport trunk allowed vlan all
```

--> (mặc định)

Lưu cấu hình

```
switch#copy running-config startup-config
```

- **Cấu hình trên router** (cấu hình sub-interface và trunking)

```
Router(config)#interface fa0/0
```

```
Router(config-if)#no shutdown
```

```
Router(config)#int fastethernet 0/0.1
```

```
Router(config-subif)#encapsulation dot1q 10
```

```
Router(config-subif)#ip address 192.168.1.1 255.255.255.0
```

```
Router(config-subif)#exit
```

```
Router(config)#int fastethernet 0/0.2
```

```
Router(config-subif)#encapsulation dot1q 20
```

```
Router(config-subif)#ip address 192.168.2.1 255.255.255.0
```

```
Router(config-subif)#exit
```

```
Router(config)#int fastethernet 0/0.3
```

```
Router(config-subif)#encapsulation dot1q 30
```

```
Router(config-subif)#ip address 192.168.3.1 255.255.255.0
```

```
Router(config-subif)#exit
```

```
Router(config)#interface fastethernet 0/0.4
```

```
Router(config-subif)#encapsulation dot1q 40
```

```
Router(config-subif)#ip address 192.168.4.1 255.255.255.0
```

```
Router(config-subif)#exit
```

- ✓ Lưu cấu hình

```
Router#copy run start
```

- **Kiểm tra**

- ✓ Xem thông tin VLAN: lệnh `show vlan`

- ✓ Xem trạng thái các cổng: lệnh `Switch#show interfaces interface`

- ✓ Kiểm tra sự liên lạc giữa các VLAN: sử dụng lệnh `ping` giữa các PC

Lab 4-5. Inter-VLAN routing (MultiLayer Switch)

Topology

fa0/1 fa0/1 fa0/2 fa0/2

VLAN10: 192.168.10.0/24

VLAN20: 192.168.20.0/24

VLAN30: 192.168.30.0/24

VLAN40: 192.168.40.0/24

Yêu cầu:

- Cấu hình trunk
- Cấu hình VTP, VLAN
Vtp domain: CNTT
MSW: vtp server
SW1, SW2: vtp client
- Cấu hình MSW để routing giữa 4 VLAN

Cấu hình:**❖ Cấu hình trunk**

```
SW1(config)#interface fa0/1
SW1(config-if)#switchport mode trunk
SW1(config-if)#switchport trunk encapsulation dot1q

SW2(config)#interface fa0/2
SW2(config-if)#switchport mode trunk
SW2(config-if)#switchport trunk encapsulation dot1q

MSW(config)#interface fa0/1
MSW(config-if)#switchport mode trunk
MSW(config-if)#switchport trunk encapsulation dot1q
MSW(config)#interface fa0/2
MSW(config-if)#switchport mode trunk
MSW(config-if)#switchport trunk encapsulation dot1q
```

❖ Cấu hình VTP, VLAN

```
MSW(config)#vtp domain CNTT
MSW(config)#vtp mode server
```

```
SW1(config)#vtp domain CNTT
SW1(config)#vtp mode client
```

```
SW2(config)#vtp domain CNTT
SW2(config)#vtp mode client
```

```
MSW(config)#vlan 10
MSW(config)#vlan 20
MSW(config)#vlan 30
MSW(config)#vlan 40
```

❖ Cấu hình MSW để routing giữa 4 VLAN

```
MSW(config)#ip routing
MSW(config)#interface vlan 10
MSW(config-if)#ip address 192.168.10.1 255.255.255.0
MSW(config)#interface vlan 20
MSW(config-if)#ip address 192.168.20.1 255.255.255.0
MSW(config)#interface vlan 30
MSW(config-if)#ip address 192.168.30.1 255.255.255.0
MSW(config)#interface vlan 40
MSW(config-if)#ip address 192.168.40.1 255.255.255.0
```

1. Kiểm tra cấu hình

```
show interface trunk
show vtp status
show vlan brief
show ip route
```

fa0/1

00-40-45-19-71-83

❖ Yêu cầu

- Chỉ có client với địa chỉ MAC: 00-40-45-19-71-83 được sử dụng port fa0/1 trên Switch.
- Các client khác gắn vào port fa0/1, port fa0/1 sẽ bị shutdown
- Port fa0/1 sẽ khôi phục lại sau 30 giây.

❖ Cấu hình

1. Cấu hình port security. Chỉ có client với địa chỉ MAC: 00-40-45-19-71-83 được sử dụng port fa0/1 trên Switch.

```
Switch(config)#interface fa0/1
Switch(config-if)#switchport mode access
Switch(config-if)#switchport port-security
Switch(config-if)#switchport port-security mac-address 0040.4519.7183
```

2. Các client khác gắn vào port fa0/1, port fa0/1 sẽ bị shutdown

```
Switch(config)#interface fa0/1
Switch(config-if)#switchport port-security violation shutdown
```

3. port fa0/1 sẽ khôi phục lại sau 30 giây.

```
Switch(config)#errdisable recovery cause all
Switch(config)#errdisable recovery interval 30
```

Kiểm tra cấu hình

```
show interface switchport
show port-security interface
```

Lab 4-7. EtherChannel

Topology



Fa0/1 - 4

Fa0/1 - 4

Yêu cầu

- Cấu hình EtherChannel dùng PAGP
- Cấu hình EtherChannel dùng LACP
- Cấu hình EtherChannel dùng mode ON
- Cấu hình Loadbalance dựa vào Source MAC, Destination MAC

Cấu hình

- ❖ Cấu hình EtherChannel dùng PAGP (desirable - desirable, desirable - auto)

```
Switch(config)#interface range fa0/1 - 4
Switch(config-if-range)#channel-protocol PAGP
Switch(config-if-range)#channel-group 1 mode {Desirable | Auto}
```

- ❖ Cấu hình EtherChannel dùng LACP (Active - Active, Active - Passive)

```
Switch(config)#interface range fa0/1 - 4
Switch(config-if-range)#channel-protocol LACP
Switch(config-if-range)#channel-group 1 mode {Active| Passive}
```

- ❖ Cấu hình EtherChannel dùng mode ON

```
Switch(config)#interface range fa0/1 - 4
Switch(config-if-range)#channel-group 1 mode ON
```

Kiểm tra cấu hình

Sử dụng các lệnh sau để kiểm tra cấu hình:

```
show etherchannel 1 detail
show etherchannel summary
show pagp 1 neighbor
```

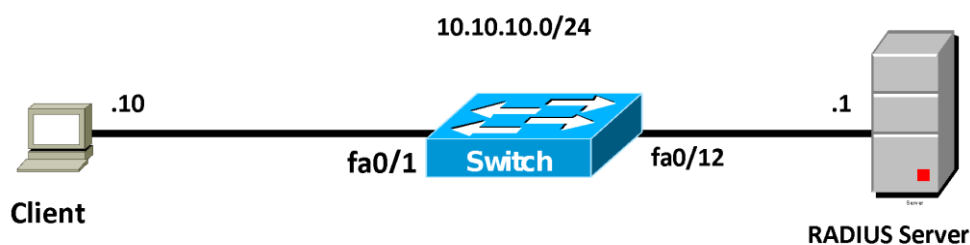
```
show pagp 1 counters
show pagp 1 internal
show lacp 1 neighbor
show lacp 1 counters
show lacp 1 internal
```

Ghi chú: Nếu là Switch Layer 3 bạn có thể cấu hình như sau để EtherChannel hoạt động ở Layer 3 :

```
Switch(config)# ip routing
Switch(config)# interface port-channel {number}
Switch(config-if)# no switchport
```

Lab 4-8. Port-based authentication - 802.1x

Topology



Yêu cầu

1. Cấu hình IP như mô hình trên
2. Cấu hình dot1x trên port fa0/1 cho phép client truy cập vào mạng với
Username: newstar
Password: cisco

Cấu hình

1. Cấu hình IP như mô hình trên
2. Cấu hình dot1x

```
Switch(config)#aaa new-model
Switch(config)#radius-server host 10.10.10.1
Switch(config)#radius-server key cisco

Switch(config)#aaa authentication dot1x default group radius
Switch(config)#dot1x system-auth-control
Switch(config)#interface fa0/1
Switch(config-if)#switchport mode access
Switch(config-if)#dot1x port-control auto
```

Tạo account (username: cisco, password: cisco) trên RADIUS Server.

CHƯƠNG 5.

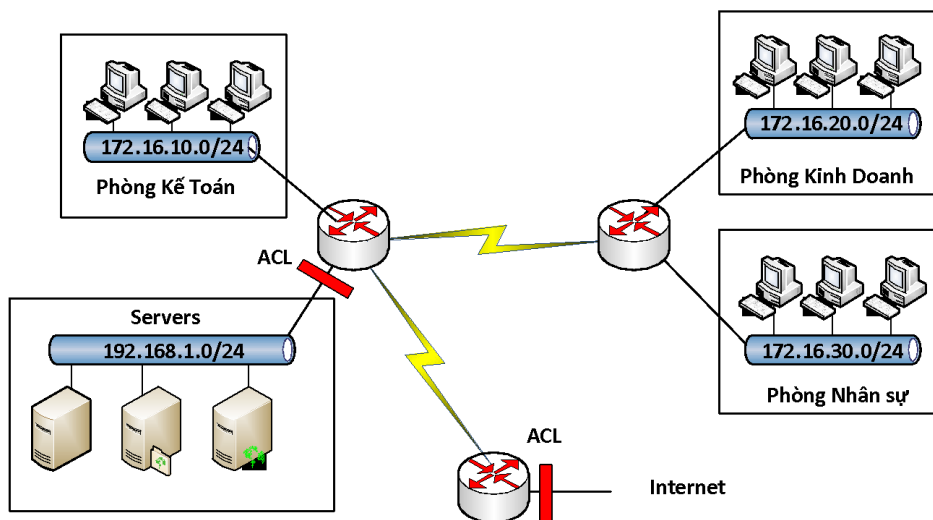
ACL

Chương này trình bày chức năng và đặc điểm của việc sử dụng ACL trong hệ thống mạng để điều khiển các truy cập, đặc điểm của các loại ACL và cách thức cấu hình trên thiết bị Cisco. Học xong chương này, người học có khả năng:

- Xác định được vai trò của ACL trong hệ thống mạng
- Phân biệt và cấu hình được “Standard ACL” và “Extended ACL” sử dụng hai phương pháp cấu hình là Numbered ACL và Named ACL
- Vận dụng ACL trong các bài toán cụ thể

1. Giới thiệu

ACL là một danh sách các điều kiện được áp đặt vào các cổng của router để lọc các gói tin đi qua nó. Danh sách này chỉ ra cho router biết loại dữ liệu nào được cho phép (allow) và loại dữ liệu nào bị hủy bỏ (deny). Sự cho phép và hủy bỏ này có thể được kiểm tra dựa vào địa chỉ nguồn, địa chỉ đích, giao thức hoặc chỉ số cổng.



Sử dụng ACL để quản lý các lưu lượng mạng, hỗ trợ ở mức độ cơ bản về bảo mật cho các truy cập mạng, thể hiện ở tính năng lọc các gói tin qua router.

2. Phân loại và hoạt động của ACL

- ❖ ACL được chia thành 2 loại:
 - Standard ACL

- Extended ACL

❖ Hoạt động của ACL

ACL thực hiện việc kiểm tra theo trình tự của các điều kiện trong danh sách cấu hình. Nếu có một điều kiện được so khớp trong danh sách thì nó sẽ thực hiện hành động tương ứng trong điều kiện đó, và các điều kiện còn lại sẽ không được kiểm tra nữa. Trường hợp tất cả các điều kiện trong danh sách đều không khớp thì một câu lệnh mặc định “deny any” được thực hiện, có nghĩa là điều kiện cuối cùng ngầm định trong một ACL mặc định sẽ là cấm tất cả. Vì vậy, trong cấu hình ACL cần phải có ít nhất một câu lệnh có hành động là “permit”.

Khi gói tin đi vào một cổng, router sẽ kiểm tra xem có ACL nào được đặt trên cổng để kiểm tra hay không, nếu có thì các gói tin sẽ được kiểm tra với những điều kiện trong danh sách. Nếu gói tin đó được cho phép bởi ACL, nó sẽ tiếp tục được kiểm tra trong bảng định tuyến để quyết định chọn cổng ra để đi đến đích.

Tiếp đó, router sẽ kiểm tra xem trên cổng dữ liệu chuyển ra có đặt ACL hay không. Nếu không thì gói tin đó có thể sẽ được gửi tới mạng đích. Nếu có ACL thì nó sẽ kiểm tra với những điều kiện trong danh sách ACL đó.

3. Cấu hình ACL

Có 2 phương pháp cấu hình ACL:

- Dựa vào số (numbered ACL)
- Dựa vào tên (named ACL)

Tổng quát: để cài đặt một ACL, ta thực hiện các bước sau:

Bước 1: Tạo ACL

- ✓ Xác định loại ACL dựa vào số hiệu ACL (numbered ACL) hoặc tên (named ACL)
- ✓ Lựa chọn hành động cho từng điều kiện “permit” hay “deny” theo yêu cầu cụ thể

Bước 2: Gán ACL vào cổng của router

- ✓ Các ACL được gán vào một hoặc nhiều cổng và có thể được lọc theo chiều các gói tin đi vào hay đi ra.
- ✓ Một router với một ACL được đặt ở cổng dữ liệu vào phải kiểm tra mỗi gói tin để tìm xem nó có khớp các điều kiện trong danh sách ACL trước khi chuyển gói tin đó đến một cổng ra.

❖ Một số thuật ngữ

● Wildcard mask

“Wildcard mask” có 32 bit, chia thành 4 phần, mỗi phần có 8 bit, là tham số được dùng xác định các bit nào sẽ được bỏ qua hay buộc phải so trùng trong việc kiểm tra điều kiện. Bit ‘1’ trong “wildcard mask” có nghĩa là bỏ qua vị trí bit đó khi so sánh, và bit ‘0’ xác định vị trí bit đó phải giống nhau.

Với Standard ACL, nếu không thêm “wildcard-mask” trong câu lệnh tạo ACL thì mặc định “wildcard-mask” sẽ là 0.0.0.0

Mặc dù “Wildcard mask” có cấu trúc 32 bit giống với “Subnet mask” nhưng chúng hoạt động khác nhau. Các bit 0 và 1 trong một “Subnet mask” xác định phần “Network” và phần “Host” trong một địa chỉ IP. Các bit 0 và 1 trong một “wildcard-mask” xác định bit nào sẽ được kiểm tra hay bỏ qua cho mục đích điều khiển truy cập.

- **Wildcard “host”**

- ✓ “Wildcard mask” dùng cho một thiết bị hay còn gọi là “wildcard-host” có dạng: 0.0.0.0 (kiểm tra tất cả các bit)

Ví dụ: 172.30.16.29 0.0.0.0

- ✓ Ý nghĩa: khi kiểm tra ACL, nó sẽ kiểm tra tất cả các bit trong địa chỉ dùng để so khớp.

- ✓ “Wildcard mask” cho một thiết bị có thể được đại diện bằng từ khóa “host”

Ví dụ: host 172.30.16.29

Câu lệnh ACL cho phép một thiết bị như sau:

```
R(config)#access-list 1 permit 172.30.16.29 0.0.0.0
```

hoặc:

```
R(config)#access-list 1 permit host 172.30.16.29
```

- **Wildcard “any”**

- ✓ Wildcard mask cho tất cả các thiết bị được gọi là wildcard “any” có dạng: 255.255.255.255 (không kiểm tra tất cả các bit)

- ✓ Ý nghĩa: chấp nhận tất cả các địa chỉ

- ✓ “Wildcard mask” dùng cho tất cả các thiết bị có thể đại diện bằng từ khóa “any”

Ví dụ:

```
R(config)#access-list 1 permit 0.0.0.0 255.255.255.255
```

hoặc:

```
R(config)#access-list 1 permit any
```

- **Inbound và outbound**

Khi áp dụng ACL trên một cổng, phải xác định ACL đó được dùng cho luồng dữ liệu vào (inbound) hay ra (outbound). Chiều của luồng dữ liệu được xác định trên cổng của router.

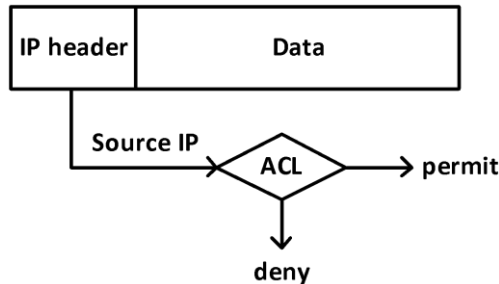


4. Standard ACL

Sử dụng “Standard ACL” khi ta muốn cấm hay cho phép tất cả các luồng dữ liệu từ một thiết bị hay một mạng xác định trên toàn bộ giao thức.

“Standard ACL” kiểm tra điều kiện dựa vào địa chỉ nguồn trong các gói tin và thực hiện hành động cấm hoặc cho phép tất cả các lưu lượng từ một thiết bị hay một mạng xác định nào đó.

Kiểm tra gói tin với “Standard ACL”:



❖ Cấu hình Standard ACL

- Router (config) # **access-list** <ACL-number> {**permit|deny**} **source** [*wildcast-mask*]

Trong đó: *ACL-number*: có giá trị từ 1 đến 99, hoặc 1300-1999

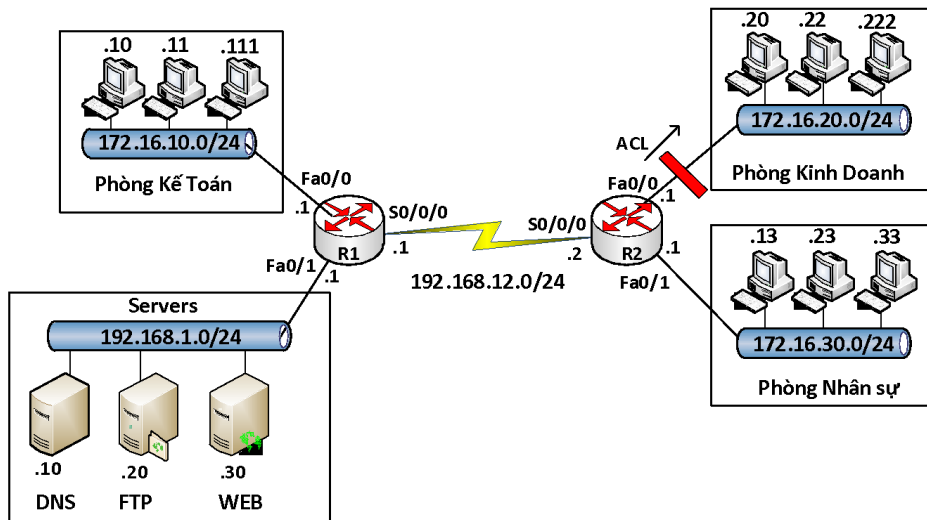
Wildcast-mask: nếu không được cấu hình sẽ lấy giá trị mặc định là:0.0.0.0

- Router (config-if) #**ip access-group** <ACL-number> {**in|out**}

Câu lệnh này có tác dụng gán ACL vào một cổng và đặt chế độ kiểm tra cho luồng dữ liệu đi vào hay đi ra khỏi cổng của router.

Dùng lệnh **no ip access-group** <ACL-number> để không áp đặt ACL vào cổng. Có nghĩa là huỷ bỏ câu lệnh trên.

Ví dụ 1: Cấm các máy tính thuộc mạng 172.16.10.0/24 truy cập tới mạng 172.16.20.0/24.

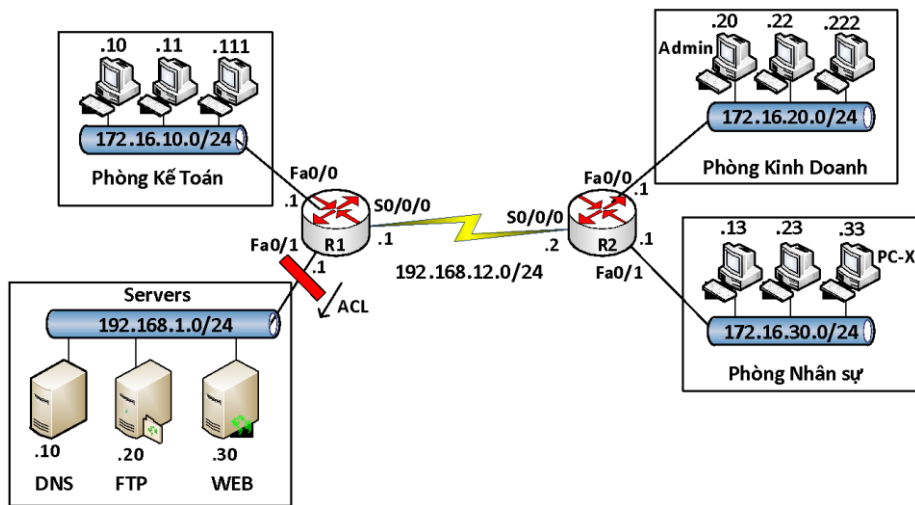


```
R2 (config)#access-list 1 deny 172.16.10.0 0.0.0.255
R2 (config)#access-list 1 permit any
```

```
R2(config)#interface fa0/0
```

```
R2(config-if)#ip access-group 1 out
```

Ví dụ 2: Cấm PC-X có địa chỉ 172.16.30.33/24 truy cập vào mạng 192.168.1.0/24



```
R1(config)# access-list 10 deny host 172.16.30.33
```

```
R1(config)# access-list 10 permit any
```

```
R1(config)#interface fa0/1
```

```
R1(config-if)#ip access-group 10 out
```

Ví dụ 3. Sử dụng lại mô hình trong ví dụ 2, viết ACL chỉ cho phép máy Admin có IP 172.16.20.20 telnet vào các router R1, R2.

Hướng dẫn cấu hình: trước tiên, cấu hình mở telnet trên R1 và R2.

ACL thực hiện yêu cầu đầu bài: trên R1 và R2 sử dụng ACL sau

```
R(config)#access-list 20 permit host 172.16.20.20
```

```
R(config)#line vty 0 4
```

```
R(config-line)#access-class 20 in
```

❖ Dùng “Standard ACL” để điều khiển telnet

Trên router có các “virtual terminal port” được dùng để cấu hình cho mục đích cho phép telnet vào router. Telnet cũng là một cách thức cho phép người quản trị cấu hình hay theo dõi thiết bị từ xa. Ta có thể lọc các địa chỉ truy xuất vào các cổng này bằng “Standard ACL”.

Cấu hình: thực hiện hai bước chính sau

- Chọn các thiết bị hoặc mạng được phép telnet vào các thiết bị dùng *Standard ACL*
 - Gán ACL đã được cài đặt ở trên vào cổng telnet.
- Các câu lệnh cấu hình:

```
Router(config)#line vty {vty-number|vty-range}
```

```
Router(config-line)#access-class <access-list-number> {in|out}
```

Trong đó:

vty-number: có giá trị 0 đến 4 (mặc định trên Router), có giá trị 0 đến 15 (mặc định trên Switch)

vty-range: là một dãy liên tiếp các port vty được sử dụng. Trong cấu hình ta sẽ cấu hình như sau: line vty start-number end-number

access-list-number: ACL gán vào các cổng **vty** để điều khiển truy cập

Ví dụ:

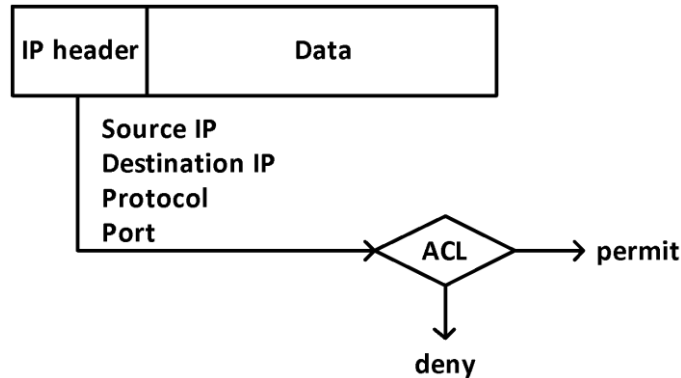
```
access-list 12 permit 192.168.1.0 0.0.0.255
(implicit deny all)
!
line vty 0 4
access-class 12 in
```

Các câu lệnh cấu hình trên có nghĩa là: chỉ cho phép các thiết bị thuộc mạng 192.168.1.0/24 có thể kết nối vào router thông qua telnet.

5. Extended ACL

“Extended ACL” cung cấp sự điều khiển linh hoạt hơn “Standard ACL”. Nó kiểm tra cả địa chỉ nguồn, địa chỉ đích, giao thức, chỉ số cổng ứng dụng. “Extended ACL” thực hiện hành động cấm hay cho phép ở một số ứng dụng xác định.

Kiểm tra các gói tin với “Extended ACL”:



❖ Cấu hình Extended ACL

- Router (config) # **access-list** <access-list-number> {**permit|deny**} <protocol> <source-address> <source-wildcard> <destination-address> <destination-wildcard> <operation> <operand>

Trong đó: *access-list-number*: có giá trị từ 100 – 199 hoặc 2000 - 2699

protocol: là ip, udp, tcp, icmp,...

operator: thường dùng là **eq**

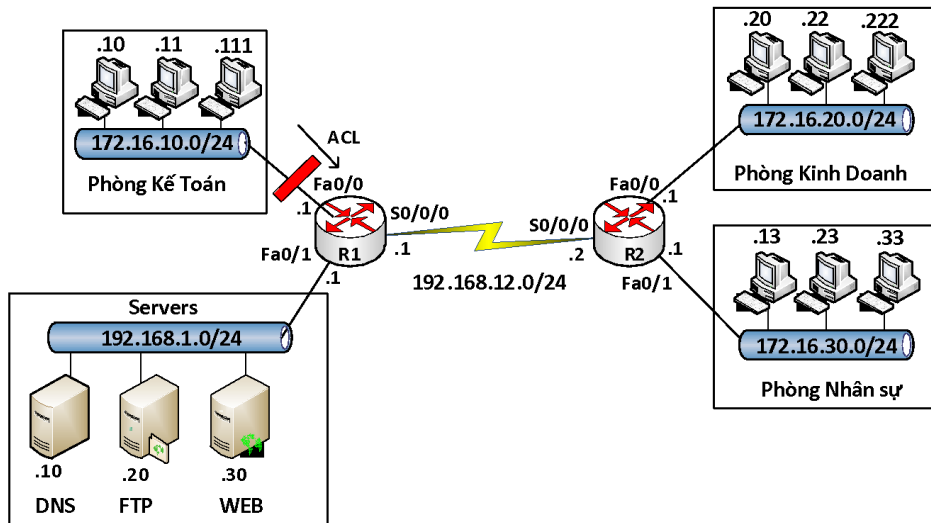
operand: là chỉ số port của dịch vụ hay tên của dịch vụ. Ví dụ: ta có thể dùng chỉ số port **23** hay có thể dùng tên dịch vụ là **telnet**

Câu lệnh trên được dùng để tạo một điều kiện (ACL entry) trong một ACL *access-list-number*

- Router (config-if) # **ip access-group** *access-list-number* {in|out}

Trong đó, *access-list-number* là số hiệu (có giá trị 100 – 199 hoặc 2000 - 2699) chỉ danh sách ACL ta đã tạo. Câu lệnh này có ý nghĩa là gán danh sách ACL vào interface và chọn hướng (*inbound hoặc outbound*) các traffic sẽ được kiểm tra

Ví dụ 1: Cấu hình trên router trong mô hình mạng dưới đây để cấm các FTP traffic từ các host thuộc subnet 172.16.10.0 đến FTP server có IP 192.168.1.20/24, cho phép tất cả các traffic còn lại hoạt động bình thường.



```
R1(config)#access-list 100 deny tcp 172.16.10.0 0.0.0.255 host
192.168.1.20 eq 20
R1(config)#access-list 100 deny tcp 172.16.10.0 0.0.0.255 host
192.168.1.20 eq 21
R1(config)#access-list 100 permit ip any any
R1(config)#interface fa0/0
R1(config-if)#ip access-group 100 in
```

- **Vị trí đặt ACL**

Nên đặt *extended ACL* gần nguồn của traffic muốn cấm và nên đặt *Standard ACL* gần đích đến của traffic.

6. Named ACL

Named-ACL cho phép *Standard* và *Extended ACL* được định danh bởi một tên thay vì đại diện bởi một con số. Loại ACL này có thể cho phép xóa một số dòng (điều kiện) trong một danh sách đã được cấu hình.

Named-ACL không tương thích với các Cisco IOS phiên bản trước 11.2 và không thể sử dụng cùng một tên cho nhiều ACL. ACL của các loại giao thức khác nhau không thể có cùng một tên.

- **Các câu lệnh cấu hình Name ACL**

```
Router(config)#ip access-list {standard | extended} name
```

```
Router(config{std-|ext-}nacl)#[sequence-number] {permit|deny} {ip
access list test conditions}
```

```
Router(config-if)#ip access-group name {in | out}
```

❖ Một số lệnh kiểm tra cấu hình ACL

```
Router#show access-list {access-list-number | name}
```

Sau đây một ví dụ về kết quả hiển thị của lệnh *show access-lists*

```
Router#show access-lists
```

```
Standard IP access list 1
```

```
permit 10.2.2.1
```

```
permit 10.3.3.1
```

```
permit 10.4.4.1
```

```
permit 10.5.5.1
```

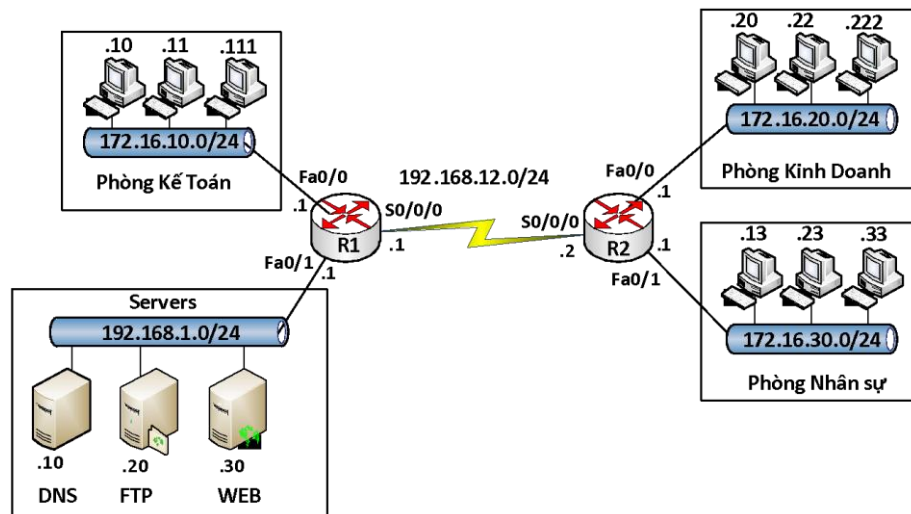
```
Extended IP access list 101
```

```
permit tcp host 10.22.22.1 any eq telnet
```

```
permit tcp host 10.33.33.1 any eq ftp
```

```
permit tcp host 10.44.44.1 any eq ftp-data
```

Ví dụ:



❖ Yêu cầu:

- (1) Cấu hình standard ACL cấm các máy tính thuộc phòng Kinh doanh truy cập tới phòng Kế toán
- (2) Cấm các máy tính thuộc phòng Kế toán truy cập tới Web server bằng dịch vụ www
- (3) Cấm các máy tính thuộc phòng Nhân sự ping tới DNS server

❖ Hướng dẫn cấu hình

Bước 1: Cấu hình hostname, địa chỉ IP cho các cổng trên các thiết bị, cấu hình định tuyến cho hệ thống mạng trên với giao thức định tuyến tùy chọn.

Bước 2: Cấu hình ACL theo yêu cầu

(1) Có thể dùng standard ACL và extended ACL cho yêu cầu này

- Dùng “Standard ACL”

```
R1(config)#ip access-list standard abc
R1(config-std-nacl)# deny 172.16.20.0 0.0.0.255
R1(config-std-nacl)# permit any
R1(config)#interface fa0/0
R1(config-if)#ip access-group abc out
```

- Dùng “Extended ACL” (có thể cấu hình trên R1 hoặc R2)

```
R2(config)#ip access-list extended xyz
R2(config-ext-nacl)#deny ip 172.16.20.0 0.0.0.255 172.16.10.0
0.0.0.255
R2(config-ext-nacl)#permit ip any any
R2(config)#interface fa0/0
R2(config-if)#ip access-group xyz in
```

(2) Cấm các máy tính thuộc phòng Kế toán truy cập tới Web server bằng dịch vụ www

```
R1(config)#ip access-list extended spkt
R1(config-ext-nacl)#deny tcp 172.16.10.0 0.0.0.255 host
192.168.1.30 eq 80
R1(config-ext-nacl)#permit ip any any
R1(config)#interface fa0/1
R1(config-if)#ip access-group spkt out
```

(3) Cấm các máy tính thuộc phòng Nhân sự ping tới DNS server

```
R2(config)#ip access-list extended cntt
R2(config-ext-nacl)#deny icmp 172.16.30.0 0.0.0.255 host
192.168.1.10
R2(config-ext-nacl)#permit ip any any
R2(config)#interface fa0/1
R2(config-if)#ip access-group cntt in
```

❖ Kiểm tra

Dùng lệnh *ping*, trình duyệt Web để kiểm tra kết quả, dùng các câu lệnh show trên router để kiểm tra cấu hình

```
show run
show ip route
show access-lists
```

7. Tổng kết chương

ACL có thể xem như là một tường lửa nhỏ định ra một tập luật để chặn các truy cập bất hợp pháp được cấu hình trên các router.

ACL được chia làm hai loại: *standard ACL* và *Extended ACL*. Trong đó, *standard ACL* thường được đặt ở gần đích, còn *Extended ACL* thường đặt ở gần nguồn cần cấm luồng dữ liệu.

ACL hoạt động theo trình tự cấu hình được thiết lập, khi một điều kiện được so khớp thì các câu lệnh còn lại sẽ không được kiểm tra nữa và cuối danh sách luôn có câu lệnh mặc định là “*deny all*”.

8. Câu hỏi và bài tập

8.1. ACL sau đây được áp đặt vào cổng fa0/0 theo chiều outbound:

```
access-list 123 deny tcp 192.168.1.8 0.0.0.7 eq 20 any
access-list 123 deny tcp 192.168.1.9 0.0.0.7 eq 21 any
```

Cho biết ý nghĩa của ACL trên?

- A. Tất cả các gói tin sẽ được cho phép đi qua cổng fa0/0 trừ các gói tin FTP.
- B. Cấm các gói tin FTP xuất phát từ 192.168.1.22 đến bất kỳ đâu
- C. Cấm các gói tin FTP xuất phát từ 92.168.1.9 đến bất kỳ đâu
- D. Tất cả các gói tin đi qua cổng fa0/9 đều bị cấm.
- E. Cấm các gói tin FTP từ bất kỳ đâu đến mạng 192.168.1.8/29

8.2. Standard ACL lọc các gói tin dựa vào thành phần nào trong gói tin?

- A. Dựa vào địa chỉ IP nguồn và IP đích
- B. Dựa vào chỉ số port đích
- C. Dựa vào địa chỉ IP nguồn
- D. Tất cả các câu trên

8.3. ACL nào sau đây được sử dụng để cấm telnet xuất phát từ mạng 210.93.105.0/24 đến mạng 223.8.151.0/24?.

- A.

```
access-list one deny 210.93.105.0 0.0.0.0 any eq 23
access-list one permit any
```
- B.

```
access-list 100 deny tcp 210.93.105.0 0.0.0.255 223.8.151.0
0.0.0.255 eq 23
```
- C.

```
access-list 100 deny ip 223.8.151.0 0.0.0.255 any 23
access-list 100 permit ip any any
```
- D.

```
access-list 100 deny tcp 210.93.105.0 0.0.0.255 223.8.151.0
0.0.0.255 eq telnet
access-list 100 permit ip any any
```

8.4. Câu nào sau đây là “Standard ACL”?.

- A. `access-list 10 permit 192.168.1.0 0.0.0.255`
- B. `access-list 100 deny host 192.168.1.100`
- C. `access-list 101 permit ip any 192.168.1.0 0.0.0.255`
- D. `access-list 10 permit tcp 192.168.1.0 0.0.0.255 any`

8.5. Công ty XYZ sử dụng *Subnet mask /29*. *Wildcard mask* được sử dụng để cấu hình ACL để *permit* hay *deny* truy cập cho mạng này?

- A. 255.255.255.224
- B. 255.255.255.248
- C. 0.0.0.224
- D. 0.0.0.8
- E. 0.0.0.7
- F. 0.0.0.3

8.6. Một ACL được cấu hình như sau:

```
access-list 10 permit 172.29.16.0 0.0.0.255
access-list 10 permit 172.29.17.0 0.0.0.255
access-list 10 permit 172.29.18.0 0.0.0.255
access-list 10 permit 172.29.19.0 0.0.0.255
```

Lệnh nào sau đây có thể thay thế cho tất cả các lệnh trên?

- A. `Access-list 10 permit 172.29.16.0 0.0.0.255`
- B. `Access-list 10 permit 172.29.16.0 0.0.1.255`
- C. `Access-list 10 permit 172.29.16.0 0.0.3.255`
- D. `Access-list 10 permit 172.29.16.0 0.0.15.255`
- E. `Access-list 10 permit 172.29.0.0 0.0.255.255`

8.7. ACL nào sau đây là ví dụ dùng để cấm các gói tin xuất phát từ một host cụ thể?

- A. `router(config)#access list 1 deny 172.31.212.74`
- B. `router(config)#access list 1 deny 10.6.111.48 host`
- C. `router(config)#access list 1 deny 172.16.4.13 0.0.0.0`
- D. `router(config)#access list 1 deny 192.168.14.132 255.255.255.0`
- E. `router(config)#access list 1 deny 192.168.166.127 255.255.255.255`

8.8. ACL nào sau đây được dùng để cấm tất cả các gói tin telnet đến mạng 10.10.1.0/24?

- A. `access-list 15 deny telnet any 10.10.1.0 0.0.0.255 eq 23`

- B. access0list 115 deny udp any 10.10.1.0 eq telnet
- C. access-list 15 deny tcp 10.10.1.0 255.255.255.0 eq telnet
- D. access-list 115 deny tcp any 10.10.1.0 0.0.0.255 eq 23
- E. access-list 15 deny udp any 10.10.1.0 255.255.255.0 eq 23

8.9. ACL được cấu hình trong router như sau:

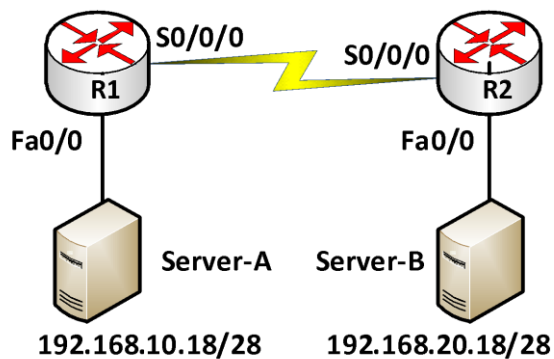
```

router#show access-lists
Extended IP access list 110
10 deny tcp 172.16.0.0 0.0.255.255 any eq telnet
20 deny tcp 172.16.0.0 0.0.255.255 any eq smtp
30 deny tcp 172.16.0.0 0.0.255.255 any eq http
40 permit tcp 172.16.0.0 0.0.255.255 any
  
```

Hãy cho biết router sẽ thực hiện hành động gì khi các gói tin HTTP từ Internet đến 172.16.12.10 nếu các gói HTTP này được ACL kiểm tra.

- A. Các gói tin này sẽ bị hủy bởi so khớp với điều kiện có số thứ tự 30
- B. Các gói tin này sẽ cho phép đi qua bởi so khớp với điều kiện có số thứ tự 40
- C. Các gói tin này sẽ bị hủy bởi vì lệnh ngầm định cấm tất cả ở cuối ACL
- D. Các gói tin này sẽ cho phép đi qua bởi vì địa chỉ nguồn không thuộc trong ACL

8.10. Cho mô hình mạng sau



Để điều khiển truy cập trong mạng, người quản trị tạo ACL như sau:

```

access-list 101 permit tcp 192.168.10.16 0.0.0.15 192.168.20.16
0.0.0.15 eq 23
  
```

Cho biết ý nghĩa của ACL trên và nên đặt ACL này trên router nào, cổng nào và theo hướng nào.

- A. Cho phép các gói tin Telnet từ 192.168.1.16/28 đến 192.168.2.16/28.

- B. Cho phép các gói tin SMTP từ 192.168.1.16/28 đến 192.168.2.16/28.
- C. ACL cho phép các gói tin từ một host này đến một host khác.
- D. ACL nên đặt vào cổng fa0/0 trên Router R1 theo hướng inbound.
- E. ACL nên đặt vào cổng fa0/0 trên router R1 theo hướng outbound.

9. Lab. ACL

Lab 5-1. STANDARD ACL

❖ Yêu cầu

Lọc các packet sử dụng standard ACL, thực hiện cấm tất cả các traffic từ PC2 đến các PC trong mạng 172.16.0.0/16

❖ Các bước thực hiện

Bước 1: Cấu hình trên RouterA

```

Router(config)#hostname RouterA
RouterA(config)#int f0/0
RouterA(config-if)#ip add 192.168.1.1 255.255.255.0
RouterA(config-if)#no shutdown
RouterA(config)#int s0/0/0
RouterA(config-if)#ip add 203.162.1.1 255.255.255.0
RouterA(config-if)#clock rate 64000
RouterA(config-if)#no shutdown
  
```

Bước 2: Cấu hình trên RouterB

```

router(config)#hostname RouterB
RouterB(config)#int s0/0/1
RouterB(config-if)#ip add 203.162.1.2 255.255.255.0
RouterB(config-if)#no shutdown
RouterB(config)#access-list 1 deny 192.168.1.3 0.0.0.0
(hoặc RouterB(config)#access-list 1 deny host 192.168.1.3)
RouterB(config)#access-list 1 permit ip any
RouterB(config)#interface f0/0
RouterB(config-if)#ip add 172.16.1.1 255.255.255.0
RouterB(config-if)#ip access-group 1 out
RouterB(config-if)#no shut
RouterB(config-if)#exit

```

❖ Kiểm tra

Dùng lệnh ping để theo dõi kết quả hiển thị

- Ping từ PC1 đến PC2, PC3
- Ping từ PC3 đến PC1, PC2
- Ping từ PC2 đến PC1, PC3

Dùng các câu lệnh show trên router để kiểm tra cấu hình

- Router#show run
- Router#show ip route
- Router#show access-lists

Lab 5-2. EXTENDED ACL

S0/0/0



❖ Yêu cầu

203.162.1.2/24

- Lọc các gói dữ liệu sử dụng extended access-list. Router RouterA cho phép tất cả các lưu lượng từ PC3 tới PC1 và từ chối tất cả các lưu lượng từ PC3 tới PC2.
- Router RouterA và RouterB nối bằng đường serial và đặt địa chỉ IP theo như hình trên.
- Access-list được dùng để lọc ngõ vào trên cổng serial của RouterA, cho phép các gói từ PC3 tới PC1 và không cho phép các gói từ PC3 tới PC2.

❖ Các bước cấu hình

- 192.168.1.2 192.168.1.3, các interface

172.16.1.2/1
6

- ✓ Cấu hình RouterA

Router>enable

```
Router#config terminal
Router(config)#hostname RouterA
RouterA(config)#interface fa0/0
RouterA(config-if)#ip address 192.168.1.1 255.255.255.0
RouterA(config-if)#no shut
RouterA(config-if)#interface S0/0/0
RouterA(config-if)#ip address 203.162.1.1 255.255.255.0
RouterA(config-if)#clock rate 64000
RouterA(config-if)#no shutdown
RouterA(config-if)#exit
RouterA(config)#
```

✓ Cấu hình RouterB

```
Router>enable
Router#config terminal
Router(config)#hostname RouterB
RouterB(config)#interface fa0/0
RouterB(config-if)#ip address 172.16.1.1 255.255.255.0
RouterB(config-if)#no shutdown
RouterB(config-if)#interface s0/0/1
RouterB(config-if)#ip address 203.162.1.2 255.255.255.0
RouterB(config-if)#no shutdown
RouterB(config-if)#exit
RouterB(config)#
```

● Cấu hình định tuyến tĩnh

```
RouterA(config)#ip route 172.16.1.0 255.255.255.0 S0/0/0
RouterB(config)#ip route 192.168.1.0 255.255.255.0 203.162.1.1
```

● Cấu hình ACL

```
RouterA(config)#access-list 100 permit ip host 172.16.1.1 host 192.168.1.2
RouterA(config)#access-list 100 deny ip host 172.16.1.1 host 192.168.1.3
```

● Gán ACL vào cổng serial của RouterA

```
RouterA(config)#interface Serial 0/0/0
RouterA(config-if)#ip access-group 100 in
```

● Kiểm tra cấu hình

```
Từ PC3 ping PC2
Từ PC3 ping PC1
```

Trên RouterA dùng lệnh `show ip access-list` để xem số lượng các gói thoả mãn điều kiện ACL

Lab 5-3. ACL (tt)

Access-list có thể dùng để kiểm soát các kết nối vty tới router. Access-list cho phép xác định trạm nào được telnet vào router dựa trên địa chỉ IP.

❖ **Yêu cầu**

- Chỉ cho PC1 telnet vào RouterB.
- Chỉ cho PC2 truy cập vào R2 qua giao diện web.

❖ **Các bước thực hiện**

- Tạo các access list:

```
RouterA(config)#access-list 1 permit 192.168.1.2/24
RouterB(config)#access-list 2 permit host 192.168.1.2
```

hoặc: **PC1** **PC2**

```
RouterB(config)#access-list 2 permit host 192.168.1.2
```

- Áp access-list 1 vào các line vty để hạn chế truy cập vào RouterB qua telnet

```
RouterB(config)#line vty 0 4
RouterB(config-line)#access-class 1 in
```

- Bật web server và áp access-list 3 vào http server để hạn chế truy cập vào giao diện web.

```
RouterB(config)#ip http server
RouterB(config)#ip http access-class 2
```

❖ **Kiểm tra**

- ✓ Telnet từ PC3 vào RouterB
- ✓ Telnet từ PC1 vào RouterB
- ✓ Từ PC2 có thể truy cập vào giao diện web: từ *Internet Explorer* gõ <http://203.162.1.2> vào thanh *address* → nhập *username* là *RouterB* và *mật khẩu* là *cisco*.

CHƯƠNG 6.

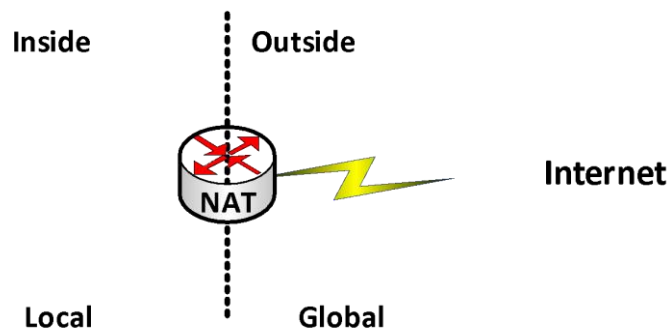
NAT

Chương này trình bày một số đặc điểm của NAT, phân loại và cấu hình trên thiết bị Cisco. Học xong chương này, người học có khả năng:

- Trình bày được một số khái niệm dùng trong kỹ thuật NAT
- Phân loại và trình bày được đặc điểm của mỗi loại NAT
- Cấu hình NAT

1. Giới thiệu

NAT (*Network Address Translation*) là một kỹ thuật cho phép chuyển đổi từ một địa chỉ IP này thành một địa chỉ IP khác. Thông thường, NAT được dùng phổ biến trong mạng sử dụng địa chỉ cục bộ, cần truy cập đến mạng công cộng (Internet). Vị trí thực hiện NAT là router biên kết nối giữa hai mạng.

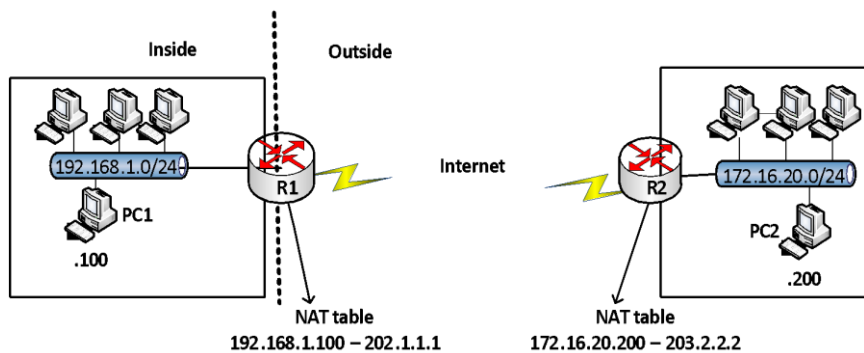


Hình 4.1 Mô hình thực hiện NAT

❖ Địa chỉ *private* và địa chỉ *public*

- Địa chỉ *private*: được định nghĩa trong RFC 1918
 - ✓ 10.0.0.0 – 10.255.255.255
 - ✓ 172.16.0.0 – 172.31.255.255
 - ✓ 192.168.0.0 – 192.168.255.255
- Địa chỉ *public*: các địa chỉ còn lại. Các địa chỉ *public* là các địa chỉ được cung cấp bởi các tổ chức có thẩm quyền.

❖ Một số thuật ngữ



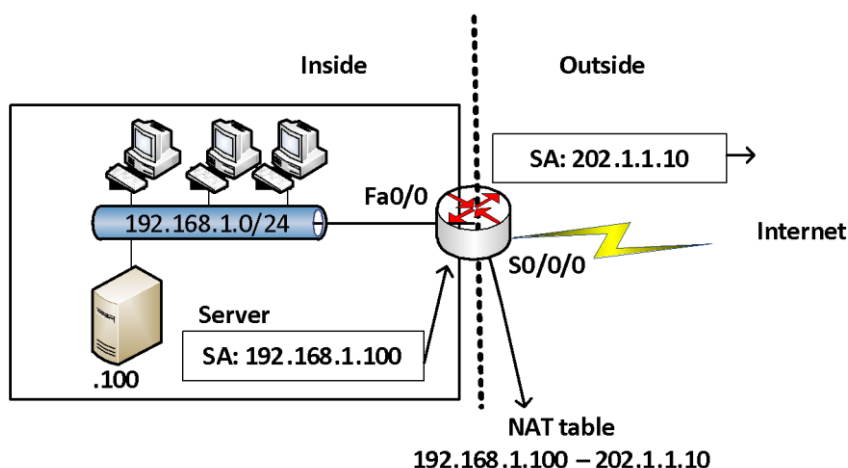
Hình 4.2 Địa chỉ *inside* và *outside*

- Địa chỉ **inside local**: là địa chỉ IP gán cho một thiết bị ở mạng bên trong. Địa chỉ này hầu như không phải địa chỉ được cấp bởi NIC (Network Information Center) hay nhà cung cấp dịch vụ.
- Địa chỉ **inside global**: là địa chỉ đã được đăng ký với NIC, dùng để thay thế một hay nhiều địa chỉ IP *inside local*.
- Địa chỉ **outside local**: là địa chỉ IP của một thiết bị bên ngoài khi nó xuất hiện bên trong mạng. Địa chỉ này không nhất thiết là địa chỉ được đăng ký, nó được lấy từ không gian địa chỉ bên trong.
- Địa chỉ **outside global**: là địa chỉ IP gán cho một thiết bị ở mạng bên ngoài. Địa chỉ này được lấy từ địa chỉ có thể dùng để định tuyến toàn cầu hay từ không gian địa chỉ mạng.

2. Static NAT

Static NAT được dùng để chuyển đổi một địa chỉ IP này sang một địa chỉ khác một cách cố định, thông thường là từ một địa chỉ cục bộ sang một địa chỉ công cộng và quá trình này được cài đặt thủ công, nghĩa là địa chỉ ánh xạ và địa chỉ được ánh xạ được chỉ định rõ ràng tương ứng duy nhất.

Static NAT rất hữu ích trong trường hợp những thiết bị cần phải có địa chỉ cố định để có thể truy cập từ bên ngoài Internet. Những thiết bị này phổ biến là những Server như Web, Mail,...



Hình 4.3 Chuyển dịch địa chỉ dạng tĩnh

❖ Cấu hình Static -NAT

- ✓ Thiết lập mối quan hệ chuyển đổi giữa địa chỉ nội bộ bên trong và địa chỉ đại diện bên ngoài.

```
Router(config)#ip nat inside source static local-ip global-ip
```

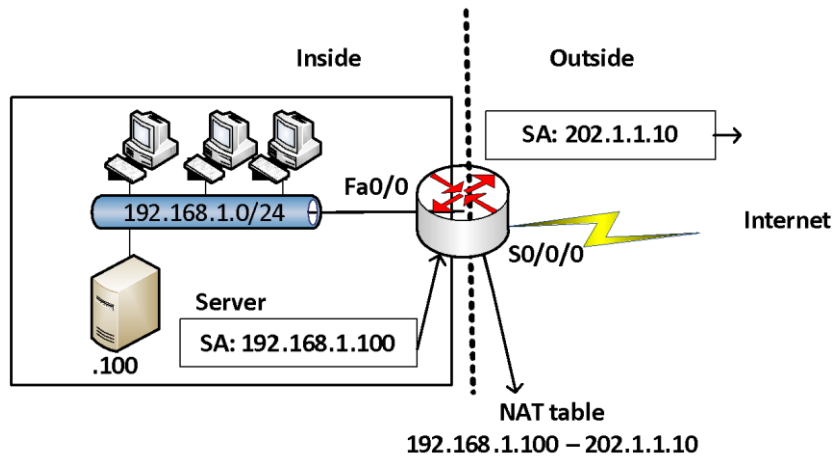
- ✓ Xác định các cổng kết nối vào mạng bên trong và thực hiện lệnh

```
Router(config-if)#ip nat inside
```

- ✓ Xác định các cổng kết nối ra mạng công cộng bên ngoài và thực hiện lệnh

```
Router(config-fi) #ip nat outside
```

Ví dụ:



```
Router(config)#ip nat inside source static 192.168.1.100 202.1.1.10
```

```
Router(config)#interface fa0/0
```

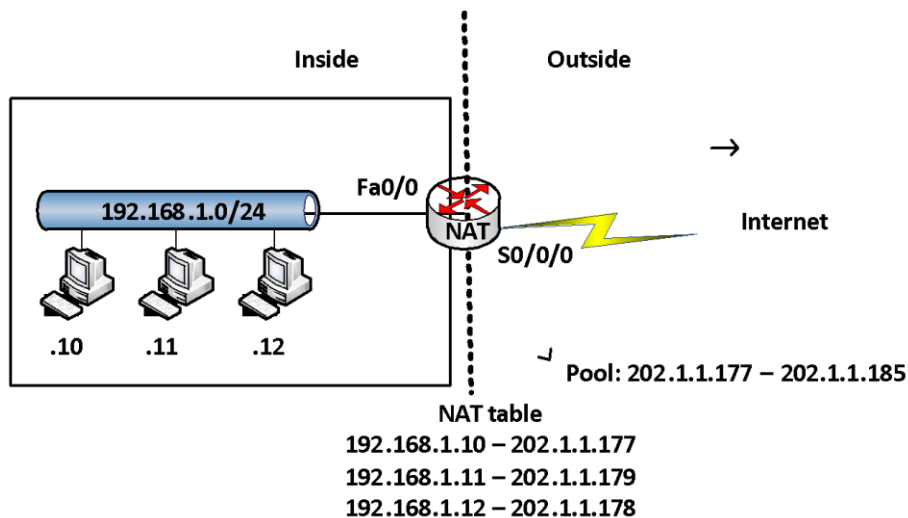
```
Router(config-if)#ip nat inside
```

```
Router(config)#interface S0/0/0
```

```
Router(config-if)#ip nat outside
```

3. Dynamic NAT

Dynamic NAT được dùng để ánh xạ một địa chỉ IP này sang một địa chỉ khác một cách tự động, thông thường là ánh xạ từ một địa chỉ cục bộ sang một địa chỉ được đăng ký. Bất kỳ một địa chỉ IP nào nằm trong dải địa chỉ IP công cộng đã được định trước đều có thể được gán cho một thiết bị bên trong mạng.



❖ Cấu hình Dynamic NAT

- ✓ Xác định dải địa chỉ đại diện bên ngoài (public): các địa chỉ NAT

```
Router(config)#ip nat pool name start-ip end-ip [netmask
netmask/prefix-length prefix-length]
```

- ✓ Thiết lập ACL cho phép những địa chỉ nội bộ bên trong nào được chuyển đổi: các địa chỉ được NAT

```
Router(config)#access-list access-list-number permit source
[source-wildcard]
```

- ✓ Thiết lập mối quan hệ giữa địa chỉ nguồn đã được xác định trong ACL với dải địa chỉ đại diện ra bên ngoài

```
Router(config)#ip nat inside source list <acl-number> pool <name>
```

- ✓ Xác định các cổng kết nối vào mạng nội bộ

```
Router(config-if)# ip nat inside
```

- ✓ Xác định các cổng kết nối ra bên ngoài

```
Router(config-if)#ip nat outside
```

Ví dụ: Cấu hình cho mô hình trong hình trên

```
Router(config)#ip nat pool abc 202.1.1.177 202.1.1.185 netmask
255.255.255.0
```

```
Router(config)#access-list 1 permit 192.168.1.0 0.0.0.255
```

```
Router(config)#ip nat inside source list 1 pool abc
```

```
Router(config)#interface fa0/0
```

```
Router(config-if)#ip nat inside
```

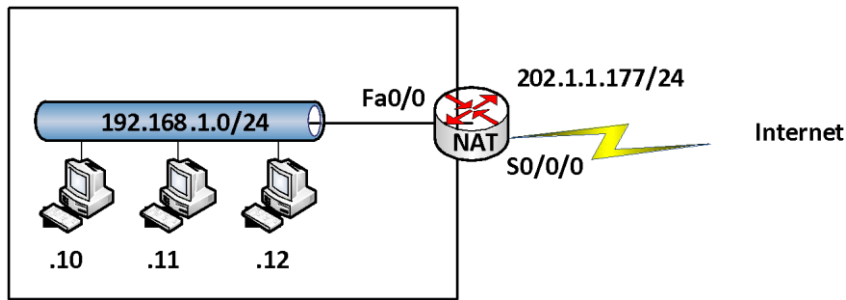
```
Router(config)#interface S0/0/0
```

```
Router(config-if)#ip nat outside
```

4. NAT overload

NAT Overload là một dạng của *Dynamic NAT*, nó thực hiện ánh xạ nhiều địa chỉ IP thành một địa chỉ (many – to – one) và sử dụng các chỉ số cổng khác nhau để phân biệt cho từng chuyển đổi. NAT Overload còn có tên gọi là PAT (*Port Address Translation*).

Chỉ số cổng được mã hóa 16 bit, do đó có tới 65536 địa chỉ nội bộ có thể được chuyển đổi sang một địa chỉ công cộng.



NAT table

```

192.168.1.10 – 202.1.1.177:1030
192.168.1.11 – 202.1.1.177:1031
192.168.1.12 – 202.1.1.177:1032

```

❖ Cấu hình NAT Overload

- ✓ Xác định dãy địa chỉ bên trong cần chuyển dịch ra ngoài (*private ip addresses range*)

```

Router(config)#access-list <ACL-number> permit <source>
<wildcard>

```

- ✓ Cấu hình chuyển đổi địa chỉ IP sang cổng nối ra ngoài

```

Router(config)#ip nat inside source list <ACL-number> interface
<interface> overload

```

- ✓ Xác định các cổng nối vào mạng bên trong và nối ra mạng bên ngoài

Đối với các cổng nối vào mạng bên trong:

```

router(config-if)#ip nat inside

```

Đối với nối ra mạng bên ngoài:

```

router(config-if)#ip nat outside

```

Ví dụ:

Giả sử hệ thống mạng công ty mô tả như sơ đồ trên, công ty thuê một đường kết nối Internet qua cổng S0/0/0 của router. Công ty muốn tất cả các thành viên trong công ty đều có thể truy cập được Internet.

Trong trường hợp này, người quản trị mạng thực hiện cấu hình PAT (NAT Overload) trên router để cho phép người dùng trong công ty có thể truy cập ra ngoài bằng địa chỉ được đăng ký trên cổng S0/0/0 của router.

Các lệnh cấu hình NAT như sau:

```

R(config)#access-list 1 permit 192.168.1.0 0.0.0.255
R(config)#ip nat inside source list 1 interface s0/0/0 overload
R(config)#interface fa0/0
R(config-if)#ip nat inside

```

```
R(config)#interface S0/0/0
```

```
R(config-if)#ip nat outside
```

❖ Các lệnh kiểm tra cấu hình

```
R#show ip nat translation → hiển thị bảng NAT đang hoạt động
```

```
R#show ip nat statistics → hiển thị trạng thái hoạt động của NAT
```

```
R#clear ip nat translation * → xóa bảng NAT
```

R#debug ip nat → kiểm tra hoạt động của NAT, hiển thị các thông tin chuyển đổi NAT bởi router.

5. Tổng kết chương

Cisco IOS NAT cho phép một tổ chức với những địa chỉ không đăng ký (địa chỉ local) có thể kết nối Internet bằng cách chuyển những địa chỉ này thành những địa chỉ đã được đăng ký (public).

NAT có ưu điểm là tiết kiệm địa chỉ đăng ký (public). Tuy nhiên, sử dụng NAT cũng có khuyết điểm là làm tăng thời gian trễ do phải thực hiện việc chuyển đổi địa chỉ trong các gói dữ liệu.

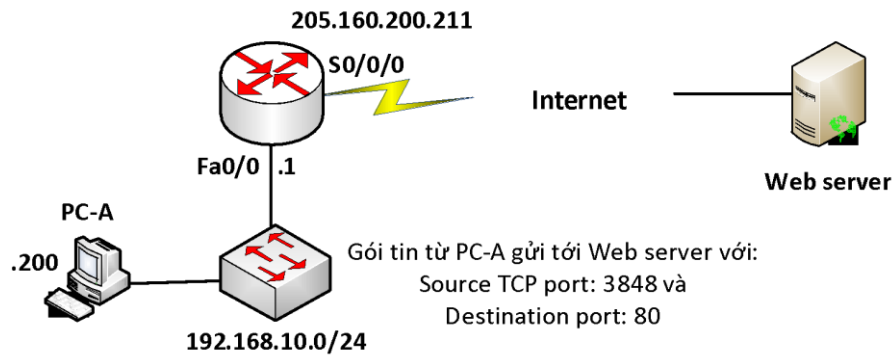
Ba kỹ thuật NAT được dùng là: *Static NAT*, *Dynamic NAT* và *NAT Overload (PAT)*. *Static NAT* được sử dụng để ánh xạ địa chỉ theo kiểu “one-to-one” và được chỉ định bởi người quản trị. *Dynamic NAT* là kiểu chuyển dịch địa chỉ dạng “one-to-one” một cách tự động. *NAT Overload* là kiểu chuyển dịch địa chỉ dạng “many-to-one” một cách tự động, sử dụng các chỉ số cổng (port) để phân biệt cho từng chuyển dịch.

6. Câu hỏi và bài tập

6.1. Địa chỉ "Inside Global" trong cấu hình NAT có ý nghĩa gì?

- A. Là địa chỉ MAC được các máy tính sử dụng để kết nối ra ngoài.
- B. Là địa chỉ tóm tắt đại diện cho tất cả các mạng bên trong.
- C. Là địa chỉ cục bộ gán cho máy tính ở mạng bên trong.
- D. Là địa chỉ được đăng ký (public) đại diện cho các máy tính bên trong khi đi ra mạng bên ngoài.

6.2. Cho mô hình mạng



NAT Overload đã được cấu hình trên router, phát biểu nào sau đây là đúng khi máy tính PC-A giao tiếp với Web server?

- A. Web server sử dụng địa chỉ IP đích là 205.160.200.211 và port đích là 80 khi gửi gói tin đến cho PC-A
- B. Máy tính PC-A sử dụng địa chỉ IP đích là 192.168.10.1 và port nguồn là 80 khi gửi các gói tin đến Web server.
- C. Web server sử dụng địa chỉ IP đích là 205.160.200.211 và port đích là 3848 khi gửi gói tin đến cho PC-A
- D. Máy tính PC-A sử dụng địa chỉ IP đích là 205.160.200.211 và port đích là 3848 khi gửi các gói tin đến Web server.

6.3. Cho mô hình mạng



Lệnh nào sau đây được cấu hình trên cổng S0/0/0 của Router NAT khi cấu hình NAT trên router này?

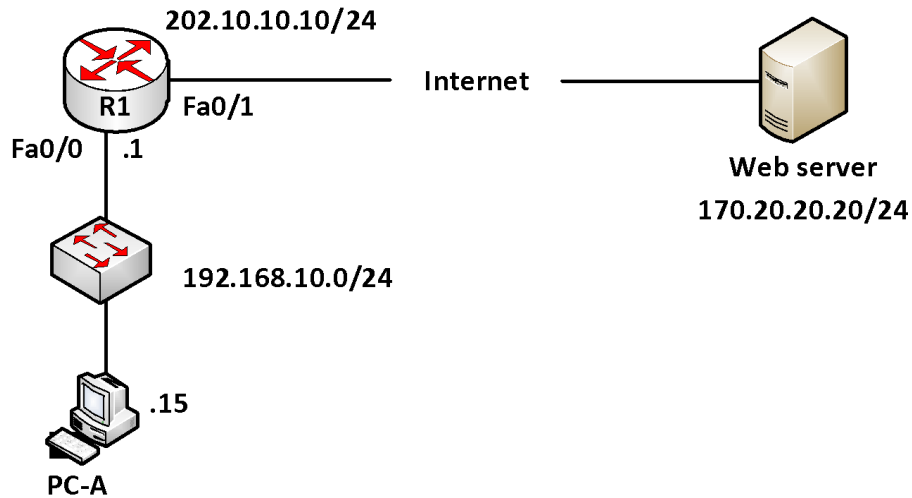
- A. ip nat inside
- B. ip nat outside
- C. ip pat inside
- D. ip pat outside

6.4. Hai phát biểu nào sau đây là đúng cho loại Static NAT

- A. Loại này cho phép từ bên ngoài có thể khởi tạo kết nối vào bên trong
- B. Loại này không yêu cầu phải chỉ ra cổng nào gắn với mạng ngoài và cổng nào gắn với mạng bên trong ở router thực hiện NAT
- C. Loại này có thể dùng ACL để cho phép nhiều kết nối khởi tạo từ mạng bên ngoài

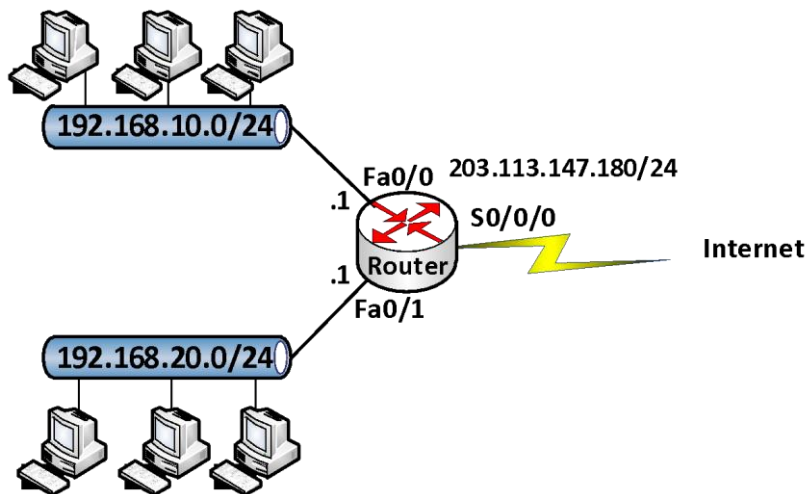
D. Loại này luôn được hiển thị trong bảng NAT

6.5. Cho mô hình mạng



Trong mô hình trên đã cấu hình NAT overload trên router R1. PC-A đang truy cập tới Web server. Hãy cho biết các địa chỉ: *inside local*, *inside global*, *outside local*, *outside global*.

6.6. Cho mô hình mạng



Router được cấu hình như sau:

```
interface FastEthernet0/0
 ip address 192.168.10.1 255.255.255.0
 ip nat outside
 duplex auto
 speed auto
```

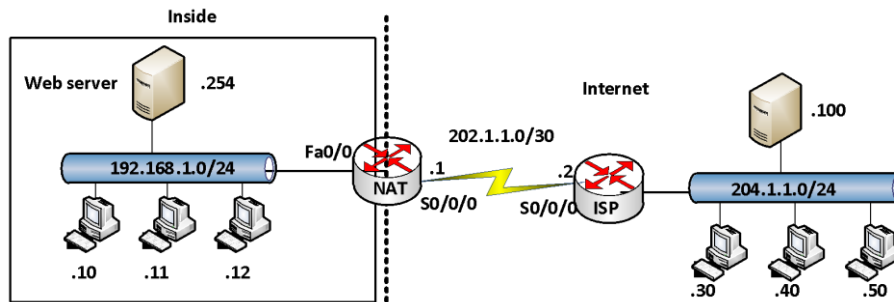
```
!  
interface FastEthernet0/1  
 ip address 192.168.20.1 255.255.255.0  
 ip nat inside  
 duplex auto  
 speed auto  
!  
interface Serial0/0/0  
 ip address 203.113.147.180 255.255.255.0  
 ip nat inside  
 clock rate 64000  
!  
interface Serial0/0/1  
 no ip address  
 shutdown  
!  
interface Vlan1  
 no ip address  
!  
ip nat inside source list 1 interface Serial0/0/0 overload  
ip classless  
ip route 0.0.0.0 0.0.0.0 Serial0/0/0  
!  
access-list 1 permit 192.168.10.0 0.0.0.255  
access-list 1 permit 192.168.20.0 0.0.0.255
```

Trên Router đã cấu hình NAT sai ở đâu?

- A. ACL cấu hình chưa đúng
- B. Cổng S0/0/0 và Fa0/0

- C. Cổng Fa0/1
- D. Lệnh default route cấu hình sai

6.7. Cho sơ đồ mạng



❖ Mô tả yêu cầu

Công ty thuê một IP public để dùng cho local Web server là 202.2.2.254 và đang kết nối ra Internet qua cổng S0/0/0 của Router. Yêu cầu cấu hình để cho các máy bên ngoài Internet có thể truy cập vào Web server và các máy bên trong có thể ra ngoài Internet net.

❖ Hướng dẫn cấu hình

- Các máy tính đặc địa chỉ IP và **default gateway** cho phù hợp
- Giữa router NAT và ISP không cấu hình bất kỳ giao thức định tuyến nào
- Router NAT tạo đường “default route” lên ISP

```
NAT(config)#ip route 0.0.0.0 0.0.0.0 s0/0/0
```

- Cấu hình public cho Web server

```
NAT(config)#ip nat inside source static 192.168.1.254
202.1.1.254
```

- Cấu hình cho phép các máy tính bên trong ra ngoài Internet

```
NAT(config)#access-list 1 permit 192.168.1.0 0.0.0.255
```

```
NAT(config)#ip nat inside source list 1 interface S0/0/0
overload
```

```
NAT(config-if)#int Fa0/0
```

```
NAT(config-if)#ip nat inside
```

```
NAT(config-if)#int S0/0/0
```

```
NAT(config-if)#ip nat outside
```

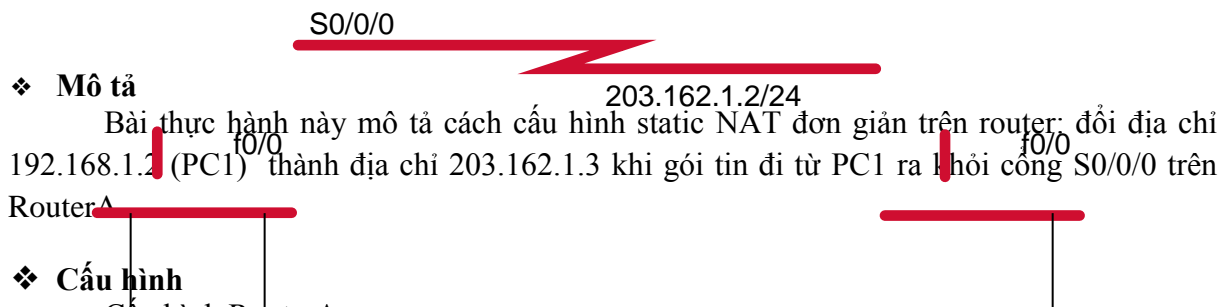
❖ Kiểm tra cấu hình bằng các lệnh đã học

- ✓ Thực hiện ping giữa các máy tính, phân tích các gói truyền nhận bằng lệnh *debug ip packet* trên router trước khi thực hiện lệnh *ping*.

Sử dụng lệnh *debug ip nat* để xem quá trình hoạt động của quá trình NAT

7. Lab NAT

Lab 6-1. STATIC NAT



❖ Mô tả

Bài thực hành này mô tả cách cấu hình static NAT đơn giản trên router: đổi địa chỉ 192.168.1.2 (PC1) thành địa chỉ 203.162.1.3 khi gói tin đi từ PC1 ra khỏi cổng S0/0/0 trên RouterA

❖ Cấu hình

- Cấu hình RouterA

```

router(config)#hostname RouterA
RouterA(config)#interface fa0/0
RouterA(config-if)#ip address 192.168.1.1 255.255.255.0
RouterA(config-if)#no shutdown
RouterA(config-if)#exit
RouterA(config)#interface serial 0/0/0
RouterA(config-if)#ip address 203.162.1.1 255.255.255.0
RouterA(config-if)#clock rate 64000
RouterA(config-if)#no shutdown
RouterA(config-if)#exit
RouterA(config)#ip nat inside source static 192.168.1.2 203.162.1.3
RouterA(config)#interface fa0/0
RouterA(config-if)#ip nat inside
RouterA(config-if)#interface Serial 0/0/0
RouterA(config-if)#ip nat outside
  
```

- Cấu hình RouterB

```

router(config)#hostname RouterB
RouterB(config)#interface fa0/0
  
```



```
RouterB(config-if)#ip address 172.16.1.1 255.255.255.0
```

```
RouterB(config-if)#no shutdown
```

```
RouterB(config-if)#interface serial 0/0/1
```

```
RouterB(config-if)#ip address 203.162.1.2 255.255.255.0
```

```
RouterB(config-if)#no shutdown
```

❖ Kiểm tra

- ✓ Thực hiện ping từ PC1 đến PC3, phân tích các gói truyền nhận bằng lệnh
debug ip packet trên router trước khi ping
- ✓ Sử dụng lệnh debug ip nat để xem quá trình hoạt động của quá trình NAT

S0/0/0

Bài thực hành này trình bày cách chuyển đổi động các địa chỉ inside thành địa chỉ outside. RouterA sẽ chuyển đổi các địa chỉ nguồn trong mạng 192.168.1.0/24 thành các địa chỉ global được xác định trong vùng (pool) địa chỉ "globalpool: 203.162.1.5 - 203.162.1.8".

RouterA được cấu hình NAT động, chuyển đổi các địa chỉ nguồn inside thành các địa chỉ global; các địa chỉ inside được xác định bởi access-list 1.

❖ Cấu hình

- Cấu hình RouterA

```

192.168.1.2 192.168.1.3 172.16.1.2/1
6
PC1 Router (config) #hostname RouterA
RouterA(config) #int f0/0
RouterA(config-if) #ip add 192.168.1.1 255.255.255.0
RouterA(config-if) #no shut
RouterA(config-if) #exit
RouterA(config) #int s0/0/0
RouterA(config-if) #ip add 203.162.1.1 255.255.255.0
RouterA(config-if) #no shut
RouterA(config-if) #exit
RouterA(config) #ip nat pool globalpool 203.162.1.5
203.162.1.8 netmask 255.255.255.0
RouterA(config) #access-list 1 permit 192.168.1.0 0.0.0.255
RouterA(config) #ip nat inside source list 1 pool globalpool
RouterA(config) #int f0/0
RouterA(config-if) #ip nat inside

RouterA(config-if) #int s0/0/0
RouterA(config-if) #ip nat outside
RouterA(config-if) #end
RouterA#
  
```

- Cấu hình RouterB

```

router (config) #hostname RouterB
RouterB(config) #interface s0/0/1
RouterB(config-if) #ip address 203.162.1.2 255.255.255.0
RouterB(config-if) #clock rate 64000
  
```

```

RouterB(config-if)#no shutdown
RouterB(config-if)#exit

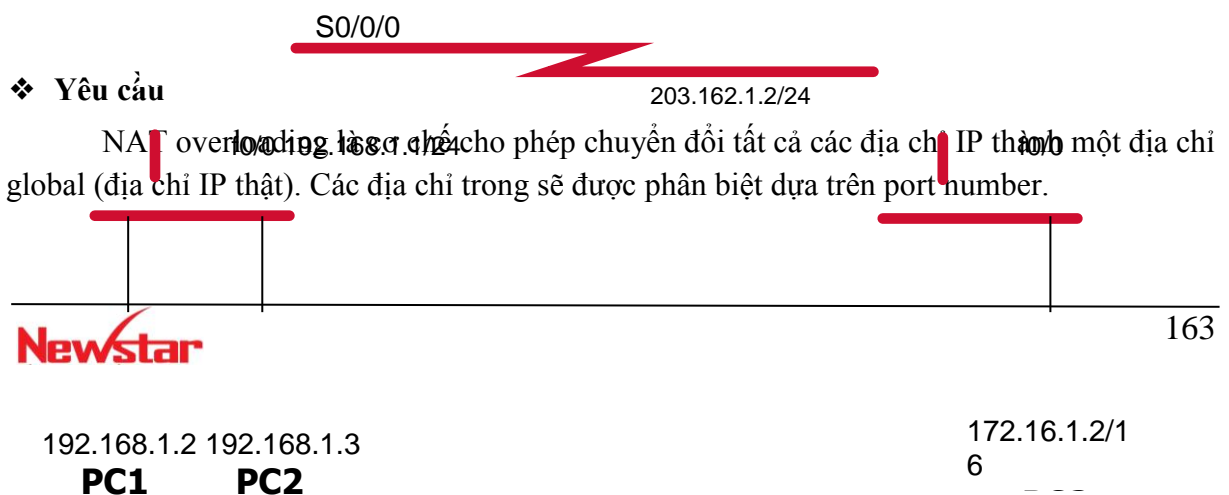
RouterB(config)#interface fa0/0
RouterB(config-if)#ip address 172.16.1.1 255.255.255.0
RouterB(config-if)#no shutdown
RouterB(config-if)#exit
RouterB(config)#

```

- **Kiểm tra**

- ✓ Từ RouterA, thực hiện lệnh ping mở rộng đến RouterB (172.16.1.1), với địa chỉ nguồn lần lượt là 192.168.1.2, 192.168.1.3, 192.168.1.4, ...
- ✓ Kiểm tra NAT đã thực hiện bằng lệnh Router#debug ip nat trên RouterA

Lab 6-3. DYNAMIC NAT WITH OVERLOAD



RouterA được cấu hình NAT và sẽ tự động chuyển dịch bất kỳ địa chỉ trong nào thuộc mạng 192.168.1.0/24 thành 203.162.1.3

❖ Cấu hình

● Cấu hình RouterA

```
Router(config)#hostname RouterA
RouterA(config)#interface fastEthernet 0/0
RouterA(config-if)#ip add 192.168.1.1 255.255.255.0
RouterA(config-if)#no shutdown

RouterA(config)#int S0/0/0
RouterA(config-if)#ip add 203.162.1.1 255.255.255.0
RouterA(config-if)#no shutdown

RouterA(config)#access-list 1 permit 192.168.1.0 0.0.0.255
RouterA(config)#ip nat pool globalpool 203.162.1.3 203.162.1.3
                                                netmask 255.255.255.0
RouterA(config)#ip nat inside source list 1 pool globalpool overload
```

● Cấu hình trên RouterB

```
Router(config)#hostname RouterB
RouterB(config)#interface fastEthernet 0/0
RouterB(config-if)#ip address 172.16.1.1 255.255.255.0
RouterB(config-if)#no shutdown

RouterB(config)#interface serial 0/0/1
RouterB(config-if)#ip address 203.162.1.2
RouterB(config-if)#clock rate 64000
RouterB(config-if)#no shutdown
```

❖ Kiểm tra

Từ RouterA, thực hiện lệnh ping mở rộng đến RouterB, source từ 192.168.1.2 và 192.168.1.3. Kiểm tra chuyển dịch bằng lệnh debug ip nat → các địa chỉ này sẽ được chuyển dịch thành 203.162.1.3

Để xem bảng chuyển đổi NAT trên RouterA dùng lệnh show ip nat translation. Lưu ý port number sau mỗi địa chỉ IP, số thứ tự các port này là chìa khóa để chuyển các gói đúng về địa chỉ IP *inside local*.

CHƯƠNG 7.

IPv6

1. Giới thiệu

Theo đặc tả của giao thức IPv6, tất cả các loại địa chỉ IPv6 được gán cho các Interface, không gán cho các Node. Mỗi địa chỉ IPv6 loại Unicast được gán cho một Interface đơn. Vì mỗi Interface thuộc về một Node đơn, do vậy mỗi địa chỉ Unicast định danh một Interface sẽ định danh cho Node đó.

Mỗi Interface đơn có thể được gán nhiều loại địa chỉ IPv6 (cho phép cả 3 dạng địa chỉ đồng thời Unicast, Anycast, Multicast). Nhưng bắt buộc mỗi Interface phải được gán một địa chỉ Unicast Link-local nhằm phục vụ cho các kết nối Point-to-point.

Một Host có thể được gán các địa chỉ sau:

- Một địa chỉ Link-local cho mỗi Interface gắn với Host đó.

- Một địa chỉ Unicast được cung cấp bởi các nhà cung cấp dịch vụ.
- Một địa chỉ Loopback.
- Một địa chỉ Multicast, mà Host đó là thành viên.

Một router nếu hỗ trợ IPv6 sẽ nhận biết được tất cả các loại địa chỉ mà host chấp nhận kể trên, ngoài ra nó còn được gán các loại địa chỉ sau:

- Tất cả các địa chỉ Multicast được gán trên Router.
- Tất cả các địa chỉ Anycast được cấu hình trên Router.
- Tất cả các địa chỉ Multicast của các nhóm thuộc Router quản lý.

2. Những hạn chế của IPv4

❖ Hạn chế

- Thiếu hụt địa chỉ
- Kích thước của các bảng định tuyến trở nên rất lớn.

❖ Giải pháp

- Giải pháp trước mắt:
 - RFC1918
 - Subnetting (1985)
 - VLSM (1987)
 - CIDR (1993)
 - NAT : là một công cụ cho phép hàng ngàn host truy cập vào Internet với một vài địa chỉ IP hợp lệ.
- Giải pháp lâu dài
 - IPv6 : năm 1994 IETF đề xuất IPv6 trong RFC1752. IPv6 khắc phục một số vấn đề về thiếu hụt địa chỉ, QoS, autoconfiguration, xác thực và bảo mật.

3. Khái quát IPv6

- Không gian địa chỉ lớn
- Tăng khả năng phân cấp địa chỉ
- Tự động cấu hình
- Header IPv6 được xây dựng lại hợp lý hơn
- Built-in security
- Tính di động

2.1 Không gian địa chỉ lớn

- IPv4 có độ dài địa chỉ là 32 bit (4 byte), có khoảng 4.200.000.000 địa chỉ

- IPv6 có độ dài địa chỉ là 128 bit (16 byte), có khoảng 3,4x 10³⁸ địa chỉ

2.1 Tăng sự phân cấp địa chỉ IP

3 bit	13 bit	32 bit	16 bit	64 bit
FP	TLA ID	NLA ID	SLA ID	Interface ID

- Phần tiền tố (format prefix) trong địa chỉ IPv6 sẽ chỉ ra địa chỉ này thuộc dạng nào (unicast, multicast, ...). Điều này cho phép hệ thống định tuyến làm việc hiệu quả hơn.
- TLA ID (Top Level Aggregation Identification) : xác định các nhà cung cấp dịch vụ cấp cao nhất trong hệ thống các nhà cung cấp dịch vụ
- NLA ID (Next Level Aggregation Identification) : xác định nhà cung cấp dịch vụ bậc 2
- SLA ID (Site Level Aggregation Identification) : xác định các Site của khách hàng
- Interface ID: xác định các Interface của các Host kết nối trong một Site

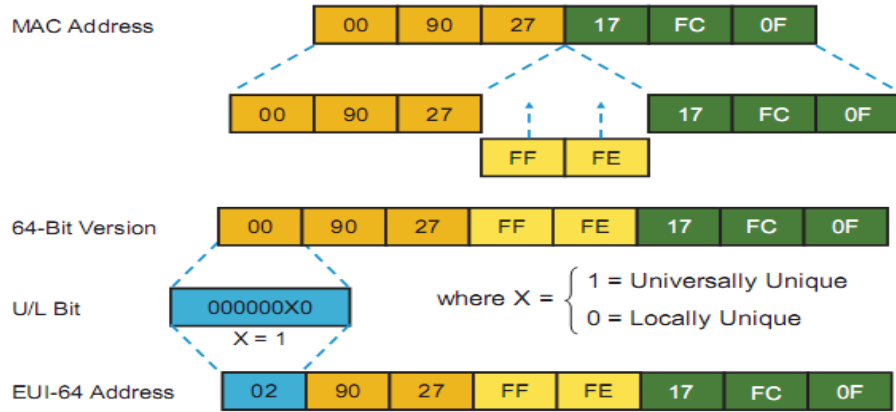
2.2 2.3 Cấu hình tự động địa chỉ IP

Các địa chỉ cục bộ hay các router kết nối trực tiếp gửi prefix ra các kết nối cục bộ và ra tuyến đường mặc định. Các thông tin này được gửi đến tất cả các node trên hệ thống mạng, cho phép các host còn lại tự động cấu hình địa chỉ IPv6. Router cục bộ sẽ cung cấp 48-bit địa chỉ toàn cục và SLA hoặc các thông tin subnet đến các thiết bị đầu cuối. Các thiết bị đầu cuối chỉ cần đơn giản thêm vào địa chỉ lớp 2 của nó. Địa chỉ lớp 2 này, cùng với 16-bit địa chỉ subnet tạo thành một địa chỉ 128-bit. Khả năng gắn một thiết bị vào mà không cần bất cứ một cấu hình nào hoặc dùng DHCP sẽ cho phép các thiết bị mới thêm vào Internet, chẳng hạn như dùng cellphone, dùng các thiết bị wireless và mạng Internet trở thành plug-and-play.

IPv6 có 128 bit, trong đó 64 bit đầu dùng cho Network và 64 bit sau dùng cho host. 64 bit của host ID định dạng theo EUI-64 có thể thu được từ địa chỉ MAC của Network interface bằng cách thành lập như sau(địa chỉ dạng EUI-64):

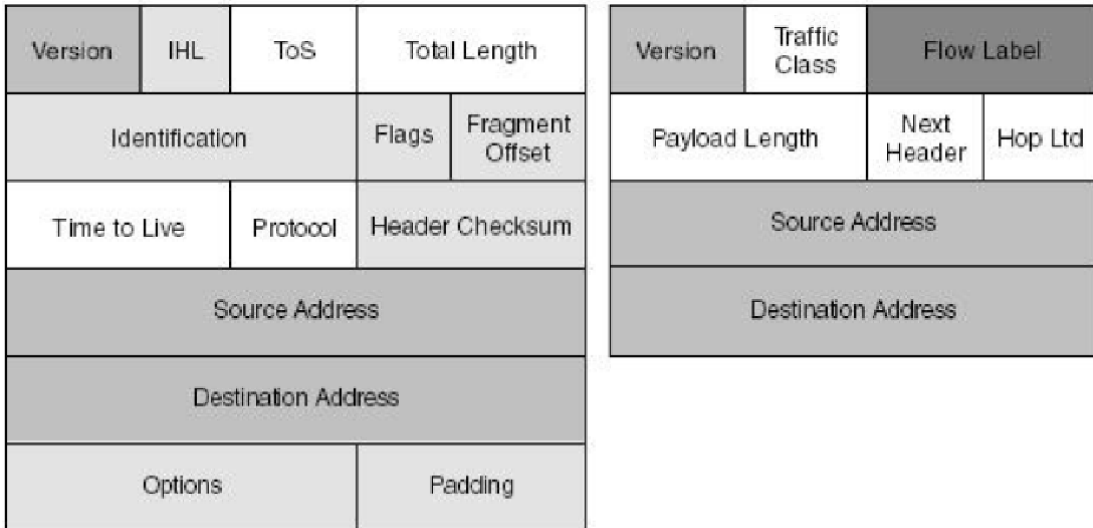
- Chèn FFFE vào giữa byte thứ 3 và thứ 4 của địa chỉ MAC (48 bit)
- Đảo bit thứ 2 trong byte thứ nhất (tính từ phải sang trái) của địa chỉ MAC

EUI-64 Interface ID Assignment



© 2013 Cisco Systems, Inc.

2.4. Header IPv6



- Name Change but Same Functionality in IPv6
- No Change in IPv6
- Removed in IPv6
- New to IPv6

Hình 7. 2. Header IPv4 và IPv6

Các trường (field) trong header IPv6:

- **Version** (4 bit): xác định phiên bản của giao thức (mang giá trị 6)
- **Traffic Class** (8 bit): xác định loại lưu lượng

- **Flow label** (20 bit): cùng với traffic class cung cấp các kiểu QoS
 - **Payload Length** (16 bit): unsigned integer. Xác định kích thước phần dữ liệu (data) theo sau IPv6 Header
 - **Next-Header** (8 bit): giúp xác định Header tiếp theo (next header) trong gói tin
 - **Hop Limited** (8 bit): unsigned integer. Qua mỗi Node, giá trị này giảm 1 đơn vị (giảm đến 0 thì gói tin bị loại bỏ).
 - **Source Address** (128 bit) : mang địa chỉ IPv6 nguồn của gói tin.
 - **Destination Address** (128 bit): mang địa chỉ IPv6 đích của gói tin.
- ❖ Phần header của IPv6 đã được đơn giản hóa để tăng tốc độ xử lý và tăng hiệu quả cho router.
- IPv6 header có kích thước cố định 40 byte và ít field hơn nên thời gian xử lý header nhanh hơn
 - Không có header checksum: Hệ thống mạng trước đây có tốc độ kết nối chậm và không đảm bảo nên việc tính toán checksum tại mỗi hop là cần thiết để đảm bảo tính toàn vẹn dữ liệu. Các kết nối mạng ngày nay nhanh và có tính tin cậy cao hơn, do đó chỉ cần các host tính checksum, không cần trên router.
 - Bỏ các header: **header length, Identification, Flags, Fragment offset,**

header checksum

- Field **Header Length** loại bỏ vì kích thước của IPv6 header là cố định. Trong IPv4 header có kích thước thay đổi từ 20 - 60 byte.
- Các field **Identification, Flags** và **Fragment Offset** được sử dụng trong việc fragment một packet trong IPv4 header. Fragment xảy ra khi một packet có kích thước lớn (large packet) được gửi qua môi trường mạng chỉ hỗ trợ packet có kích thước nhỏ hơn. Trong trường hợp này, router sẽ chia packet ra thành nhiều phần có kích thước nhỏ hơn để chuyển chúng đi. Host đích (destination host) sẽ tập hợp các packet này và lắp ráp chúng lại (reassemble). Trong quá trình truyền, nếu một gói trong chúng bị lỗi (error) hoặc không đến được đích thì toàn bộ phải được gửi lại.
- Trong IPv6, host sẽ học một MTU size. Nếu host trong quá trình gửi muốn fragment packet, nó sẽ dùng **Extension Header** để làm việc này. Các router IPv6 dọc theo đường đi của packet không fragment packet.
- Field **Checksum** loại bỏ để tăng tốc độ xử lý.

2.5 Built-in security

IPv6 tích hợp tính năng bảo mật bằng cách sử dụng 2 header mở rộng: (AH) Authentication header và Encrypted Security payload (ESP).

2.6 Tính di động

Địa chỉ IPv6 được thiết kế với tính di động được tích hợp vào trong Mobile IP cho phép các hệ thống đầu cuối thay đổi vị trí mà không mất các kết nối. Đây là điểm rất cần thiết cho những sản phẩm wireless chẳng hạn như IP phone và các hệ thống GPS, ...

Các header mở rộng trong IPv6

- Hop – by – Hop options header
- Destination options header
- Routing header
- Fragment header
- Authentication header
- Encapsulating Security Payload header

4. Cấu trúc địa chỉ IPv6

Địa chỉ IPv6 có sự khác biệt rất lớn so với địa chỉ IPv4 không chỉ về kích thước mà còn khác nhau về các thể hiện ở dạng thập lục phân.

Địa chỉ IPv6 dài 128 bit được chia thành 8 phần ở dạng thập lục phân được phân cách bởi các dấu hai chấm (:). Mỗi phần của nó có độ dài 16 bit, IPv6 sử dụng dạng hiển thị thập lục phân và không phân biệt chữ hoa hay chữ thường.

Cấu trúc địa chỉ IPv6: X:X:X:X:X:X:X:X

Trong đó: X là dạng hexa 16 bit

Ví dụ: 2031:0000:130F:0000:0000:09C0:876A:130B

❖ Một số quy tắc rút gọn địa chỉ IPv6

Địa chỉ IPv6 có chiều dài 128 bit nên vấn đề nhớ địa chỉ là hết sức khó khăn.

Sau đây là một số quy tắc rút gọn địa chỉ:

- Cho phép bỏ qua những số 0 đứng trước mỗi thành phần hệ 16, có thể viết 0 thay vì viết 0000.

Ví dụ: với block 0008 --> ta có thể viết 8

với block 0800 --> ta có thể viết 800

- Thay thế nhiều nhóm số 0 thành một dấu "::" và dấu ":::" chỉ được dùng duy nhất một lần trong mỗi địa chỉ IPv6.

Ví dụ:

2031:0000:130F:0000:0000:09C0:876A:130B

→ 2031:0:130F::9C0:876A:130B

→2031::130F::9C0:876A:130B (sai)

FF01:0:0:0:0:0:0:1 → **FF01::1**

0:0:0:0:0:0:0:1 → **::1**

0:0:0:0:0:0:0:0 → **::**

❖ Các loại địa chỉ IPv6

- Unicast
- Anycast
- Multicast

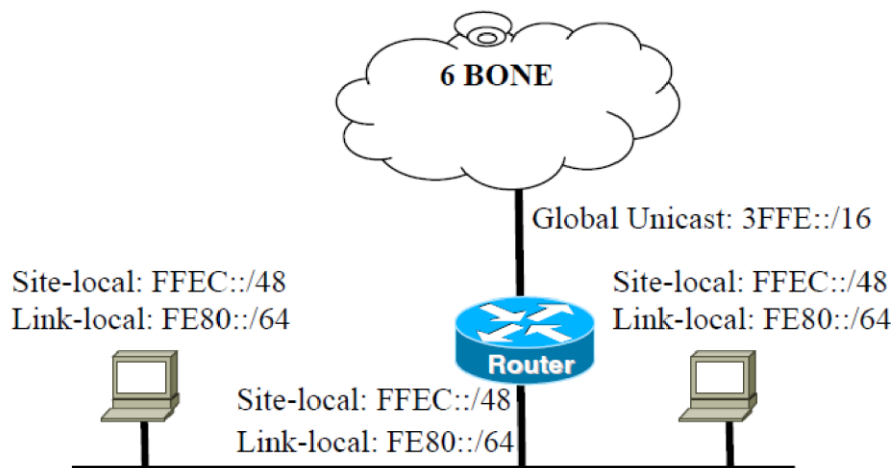
Mỗi interface có thể được gán nhiều loại địa chỉ IPv6 (cho phép cả 3 dạng địa chỉ đồng thời Unicast, Anycast, Multicast). Nhưng bắt buộc mỗi interface phải được gán một địa chỉ Unicast Link-local nhằm phục vụ cho các kết nối Point-to-point.

Theo thiết kế của IPv6, một Host có thể được định danh bởi các địa chỉ sau:

- Địa chỉ Link-local cho mỗi Interface gắn với host đó
- Địa chỉ Unicast được cung cấp bởi các nhà cung cấp dịch vụ
- Địa chỉ Loopback
- Địa chỉ Multicast, mà Host đó là thành viên

Router hỗ trợ IPv6 sẽ nhận biết được tất cả các loại địa chỉ mà host chấp nhận kể trên, ngoài ra nó còn được gán các loại địa chỉ sau:

- Tất cả các địa chỉ Multicast được gán trên Router.
- Tất cả các địa chỉ Anycast được cấu hình trên Router.
- Tất cả các địa chỉ Multicast của các nhóm thuộc Router quản lý.



Hình 7.3. Nhiều địa chỉ được gán cho một interface

Bảng phân bổ địa chỉ IPv6

Phân bố	Prefix	Tỉ trọng trong không gian địa chỉ
Dự trữ	0000 0000	1/256
Chưa sử dụng	0000 0001	1/256
Dự trữ phân bố cho NSAP	0000 001	1/128
Dự trữ phân bố cho IPX	0000 010	1/128
Chưa sử dụng	0000 011	1/128
Chưa sử dụng	0000 1	1/32
Chưa sử dụng	0001	1/16
Dùng cho địa chỉ Global Unicast	001	1/8
Chưa sử dụng	010	1/8
Chưa sử dụng	011	1/8
Chưa sử dụng	100	1/8
Chưa sử dụng	101	1/8
Chưa sử dụng	110	1/8
Chưa sử dụng	1110	1/16
Chưa sử dụng	1111 0	1/32
Chưa sử dụng	1111 10	1/64
Chưa sử dụng	1111 110	1/128
Chưa sử dụng	1111 1110 0	1/512
Địa chỉ Link Local Unicast	1111 1110 10	1/1024
Địa chỉ Site Local Unicast	1111 1110 11	1/1024
Địa chỉ Multicast	1111 1111	1/256

❖ Unicast

- Global Unicast Address

Được sử dụng để định danh các interface, cho phép thực hiện kết nối các host trong mạng Internet IPv6 toàn cầu. Ý nghĩa của nó cũng giống địa chỉ Public IPv4.

3	13 bit	8 bit	24 bit	16 bit	64 bit
FP	TLA ID	8 RES	NLA ID	SLA ID	Interface ID

Trong đó:

FP=001: Format Prefix

TLA: Top Level Aggregate

NLA: Next Level Aggregate

SLA: Site Level Aggregate

- **Link Local**

10 bit	54 bit	64 bit
1111111010	0000....0000	Interface ID

Dùng để các neighbor giao tiếp với nhau trên cùng một liên kết.

Địa chỉ Link-local Unicast luôn bắt đầu bởi Prefix **FE80::/64**, kết thúc là 64 bit Interface-ID dùng để phân biệt các Host trong một Subnet. Những địa chỉ này chỉ được định nghĩa trong phạm vi kết nối point-to-point.

Quy tắc định tuyến đối với loại địa chỉ này là Router không thể chuyển bất kỳ gói tin nào có địa chỉ nguồn hoặc đích là địa chỉ Link-local.

- **Site Local:**

10 bit	38 bit	16 bit	64 bit
1111111011	0000...0000	Subnet ID	Interface ID

Dùng để liên kết các node trong cùng một Site mà không xung đột với các địa chỉ Global. Các gói tin mang loại địa chỉ này trong IP Header, Router sẽ không chuyển ra mạng ngoài.

- Địa chỉ Site-local Unicast luôn bắt đầu bởi Prefix **FEC0::/48** theo sau là 16 bit Subnet_ID, người dùng có thể dùng 16 bit này để phân cấp hệ thống mạng của mình. Cuối cùng là 64 bit Interface_ID dùng để phân biệt các Host trong một Subnet.

Quy tắc định tuyến đối với dạng địa chỉ Site-local:

- Router không thể chuyển các gói tin có địa chỉ nguồn hoặc đích là địa chỉ Site-local Unicast ra ngoài mạng đó.

- Các địa chỉ Site-local không thể được định tuyến trên Internet. Phạm vi của chúng chỉ trong một Site, chỉ dùng để trao đổi dữ liệu giữa các host trong Site đó.

- **Anycast**

n bit	(128 – n) bit
Subnet Prefix	0000.....0000

Địa chỉ Anycast được gán cho một nhóm các interface. Những gói tin có địa chỉ đích là một địa chỉ Anycast sẽ được gửi đến node gần nhất mang địa chỉ này. Khái niệm gần nhất ở đây dựa vào khoảng cách gần nhất được xác định qua giao thức định tuyến được sử dụng.

Sử dụng Anycast có 2 lợi ích :

- Tiết kiệm được thời gian bằng cách giao tiếp với máy gần nhất
- Thứ hai là việc giao tiếp với máy gần nhất giúp tiết kiệm được băng thông

Địa chỉ Anycast không có các tầm địa chỉ được định nghĩa riêng như Multicast, mà nó giống như một địa chỉ Unicast, chỉ có khác là có thể có nhiều máy khác cũng được đánh số với cùng scope trong cùng một khu vực xác định. Anycast được sử dụng trong các ứng dụng như DNS...

- **Multicast**

8 bit	4 bit	4 bit	112 bit
11111111	Flag	scope	Group ID

Một địa chỉ multicast có thể được gán cho một nhóm các interface. Một gói tin khi chuyển đến địa chỉ multicast sẽ được chuyển đến tất cả các node mang địa chỉ multicast này.

- Địa chỉ Multicast luôn bắt đầu bởi một Prefix 8 bit “1111 1111”
- Flag có cấu trúc: **000T**

Trong đó:

3 bit thứ tự cao được dự trữ và được xác lập ở giá trị 0.

T = 0: địa chỉ Multicast “Well-known”, địa chỉ này được phân bổ bởi Global Internet Numbering Authority. Và được phân bổ cố định.

T = 1 : địa chỉ Multicast “transient”. Địa chỉ này không được phân bổ cố định.

- Scope (4 bit): được dùng để xác định phạm vi (scope) của nhóm địa chỉ Multicast. Ý nghĩa của các giá trị trong scope như sau:

- 0 : Chưa sử dụng
- 1 : Node-local
- 2 : Link-local
- 3 : Chưa sử dụng
- 4 : Chưa sử dụng
- 5 : Site-local
- 6 : Chưa sử dụng
- 7 : Chưa sử dụng
- 8 : Organization-local
- 9 : Chưa sử dụng
- A Chưa phân bổ
- B Chưa phân bổ
- C : Chưa sử dụng
- D : Chưa sử dụng
- E : Global
- F : Chưa sử dụng

- Group ID: Xác định nhóm multicast trong phạm vi một Scope. Địa chỉ Multicast cấp phát cố định hoàn toàn độc lập với giá trị được xác lập trong trường Scope.
- Ví dụ: một nhóm NTP Server được cấp group ID 111 (hex). Ta có:
 - FF01:0:0:0:0:0:111 : gửi đến tất cả các NTP trên cùng Node với Node gửi
 - FF02:0:0:0:0:0:111 : gửi đến tất cả các NTP trên cùng Link với Node gửi
 - FF05:0:0:0:0:0:111 : gửi đến tất cả các NTP trên cùng Site với Node gửi
 - FF0E:0:0:0:0:0:111 : gửi đến tất cả các NTP trên Internet

Địa chỉ multicast cấp phát không cố định chỉ có ý nghĩa trong phạm vi một Scope. Ví dụ một địa chỉ Multicast FF15:0:0:0:0:0:111 có thể được dùng trong nhiều Site mà không xung đột lẫn nhau.

• **Một số địa chỉ đặc biệt**

- Địa chỉ không xác định: 0:0:0:0:0:0:0

Địa chỉ này giống với địa chỉ 0.0.0.0 trong IPv4.

- Địa chỉ loopback: 0:0:0:0:0:0:1

Địa chỉ này có ý nghĩa giống như địa chỉ 127.0.0.1 trong IPv4

- Địa chỉ IPv4-embedded IPv6

Loại địa chỉ này được sử dụng trong cơ chế **Automatic Tunneling**, một cơ chế sử dụng trong quá trình chuyển đổi từ IPv4 lên IPv6. Địa chỉ loại này cấu tạo bởi Prefix 96 bit 0, 32 bit còn lại lấy từ một địa chỉ IPv4 hoàn chỉnh. Khi Node IPv6 truyền thông với nhau qua **Automatic Tunneling**, địa chỉ IPv4 của Tunneling sẽ được tách ra từ địa chỉ IPv4-embedded IPv6

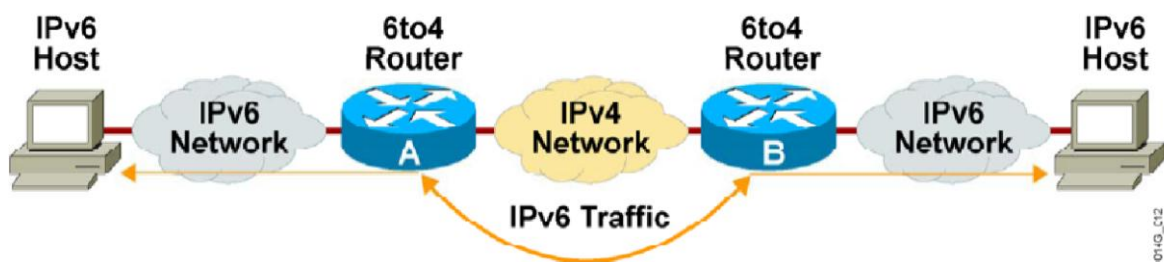
Ví dụ:

Cho địa chỉ IPv4 **10.0.0.5**.

--> Địa chỉ IPv4-embedded IPv6 có dạng **0:0:0:0:0:A00:5**.

Ta có thể giữ nguyên chấm thập phân của phần cuối. Trong trường hợp này, địa chỉ có thể được viết lại dưới dạng **::10.0.0.5**.

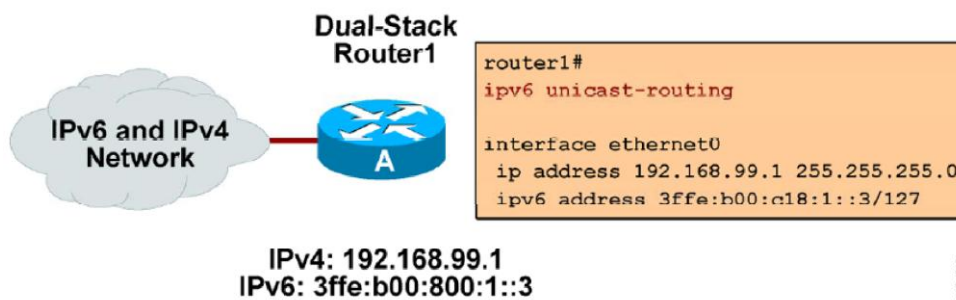
4. Các giải pháp triển khai IPv6 trên nền IPv4



Hiện tại IPv4 đang chiếm lĩnh trong môi trường Internet. Để chuyển sang sử dụng IPv6 cần có những bước trung gian trước khi chuyển hẳn sang sử dụng IPv6. Như vậy, các biện pháp thay thế sẽ diễn ra từ lớp access đến lớp core.

Có 3 giải pháp phổ biến được sử dụng để triển khai IPv6 trên nền IPv4 là: Dual Stack, Tunneling và NAT-PT.

4.1. Dual Stack (Dual IP Layer)

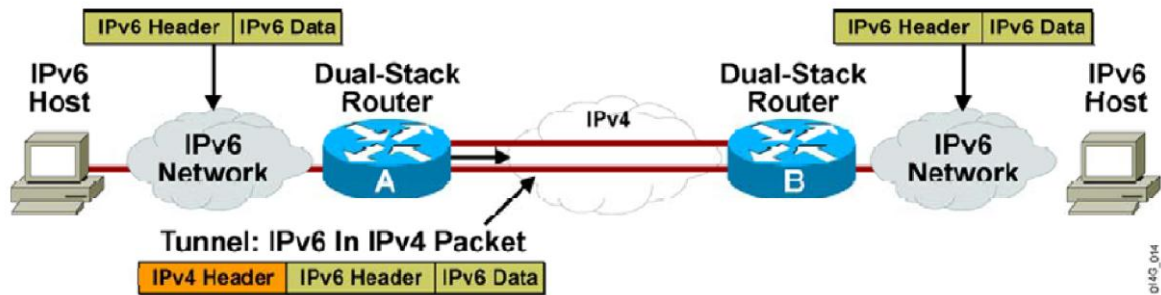


Ý tưởng của giải pháp này là: Ở mỗi host/router cài đặt cả hai giao thức IPv4 và IPv6 cùng hoạt động.

Những node này hỗ trợ cả hai giao thức, có thể làm việc được với node IPv4 thuần túy cũng như node IPv6 thuần túy. Hạn chế của cơ chế này là phải gán thêm một địa chỉ IPv4 với mỗi node IPv6 mới.

Đối với Host/Router dùng kỹ thuật Dual-IP-layer, có thể kết hợp với cơ chế chuyển đổi IPv6-over-IPv4 Tunneling.

4.2 Tunneling

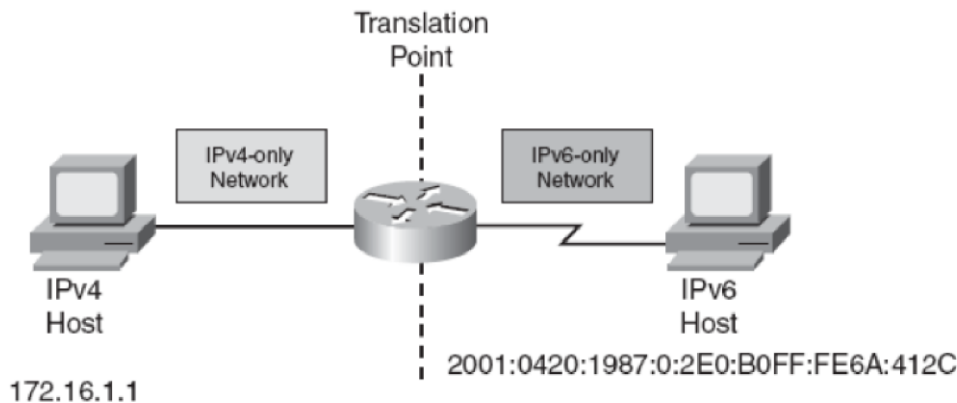


Cơ chế này thực hiện đóng gói tin IPv6 vào một gói theo chuẩn giao thức IPv4 để có thể chuyển gói tin qua mạng IPv4 thuần túy. Trong trường hợp này, mạng xem như đó là một gói tin IPv4 bình thường.

IETF đã giới thiệu hai phương pháp để tạo đường hầm cho các Site IPv6 kết nối với nhau xuyên qua hạ tầng IPv4: Automatic Tunneling và Configured Tunneling.

- Automatic tunneling: địa chỉ cuối cùng trong Tunnel là địa chỉ IPv4-compatible IPv6.
- Configured tunneling: địa chỉ cuối cùng trong Tunnel được xác định nhờ thông tin cấu hình tại các nút thực hiện đóng, mở gói IPv6 thành gói IPV4 và ngược lại.

4.3 NAT-PT

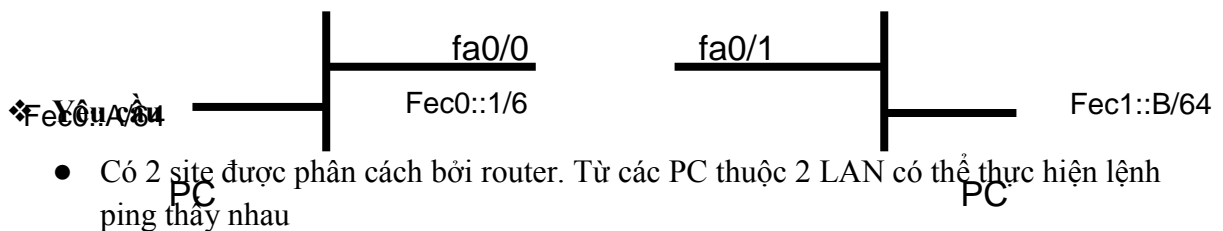


NAT-PT: Network Address Translation - Protocol Translation được mô tả trong RFC2766

NAT-PT cho phép các Host/Router dùng IPv4 thuần túy và các Host/Router dùng IPv6 thuần túy có thể kết nối làm việc với nhau. Dùng NAT-PT, ta có thể ánh xạ qua lại giữa địa chỉ IPv6 và IPv4.

5. Lab. IPv6

Lab 7-1. IPv6 CĂN BẢN



- Có 2 site được phân cách bởi router. Từ các PC thuộc 2 LAN có thể thực hiện lệnh ping thấy nhau

❖ Các bước thực hiện

Cài đặt IPv6 cho các máy tính, gán địa chỉ IPv6 cho chúng theo hình vẽ

- Thiết lập địa chỉ site-local trên PC A
- Thiết lập địa chỉ site-local trên PC B
- Xem lại cấu hình bằng lệnh ipconfig
- Kiểm tra các thông số đã cấu hình cho các PC bằng lệnh ipconfig /all
- Cấu hình router, cho các interface tham gia vào mạng sử dụng địa chỉ IPv6

```
Router#config terminal
```

```
Router(config)#ipv6 unicast-routing --> Bật chức năng định tuyến IPv6 trên Router
```

```
Router(config)#interface FastEthernet 0/0
```

```
Router(config-if)#ipv6 address fec0::1/64
```

```
Router(config-if)#no shutdown
```

```
Router(config-if)#exit
```

```
Router(config)#interface FastEthernet 0/1
```

```
Router(config-if)#ipv6 enable
```

```
Router(config-if)#ipv6 address fec1::1/64
```

```
Router(config-if)#no shutdown
```

```
Router(config-if)#end
```

• Kiểm tra

- Kiểm tra thông số đã cấu hình trên các interface:

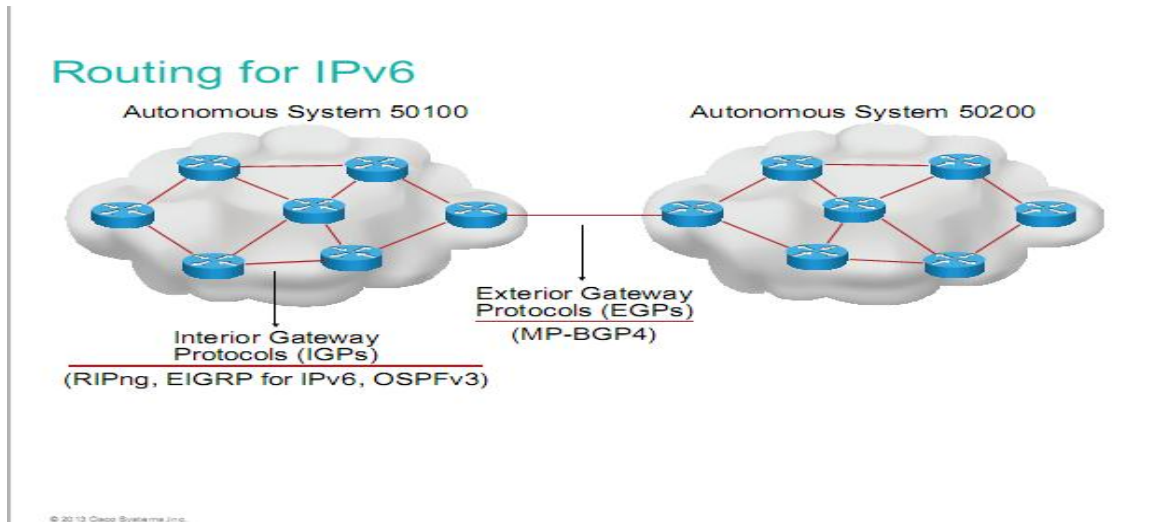
```
Router#show ipv6 interface brief
```

- Kiểm tra lại địa chỉ default gateway bằng lệnh ping
- Thực hiện lệnh ping từ PC A tới PC B bằng địa chỉ site-local.

Chương 8. IPv6 ROUTING

1. Tổng quan

Định tuyến trên nền Ipv6 cũng đã được hỗ trợ cấu hình trên các thiết bị định tuyến từ lâu. Trong chương này sẽ trình bày cách cấu hình định tuyến tĩnh và định tuyến động trên nền Ipv6.

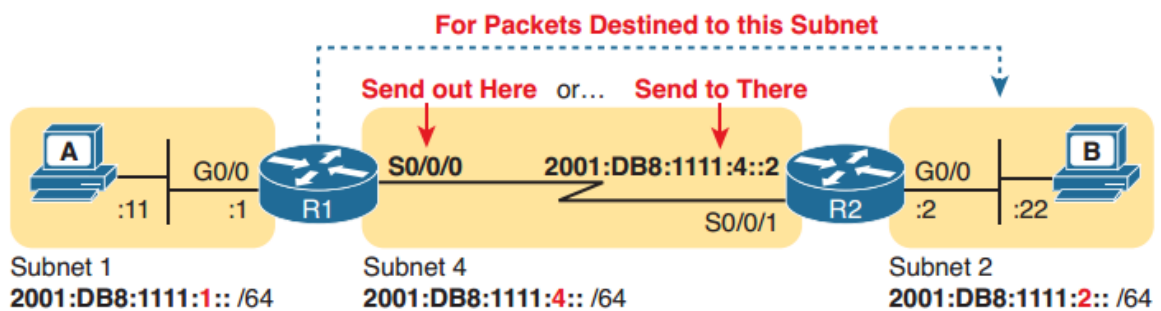


2. Định tuyến tĩnh

Cú pháp:

R(config)#ipv6 router prefix-network/prefix-length [OutGoing Interface | Next-Hop]

Ví dụ 1: Định tuyến tĩnh sử dụng “outgoing interface”



```
! Static route on router R1
R1(config)# ipv6 route 2001:db8:1111:2::/64 s0/0/0
```

```
! Static route on router R2
R2(config)# ipv6 route 2001:db8:1111:1::/64 s0/0/1
```

```
R1# show ipv6 route static
! Legend omitted for brevity
S 2001:DB8:1111:2::/64 [1/0]
via Serial0/0/0, directly connected
```

Ví dụ 2: Định tuyến tĩnh sử dụng “Next-Hop Ipv6 address”

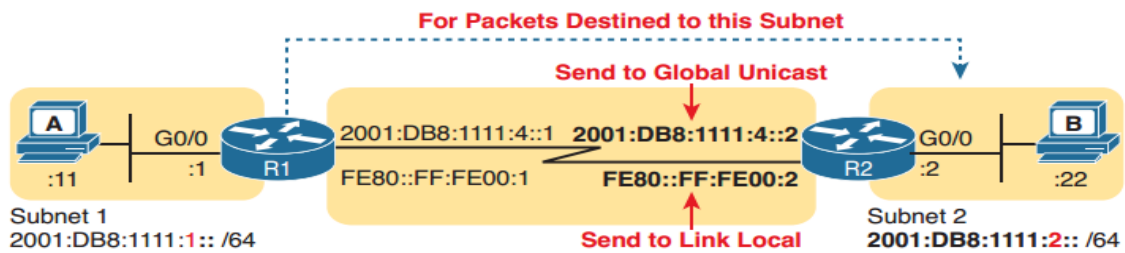


Figure 32-3 Using Unicast or Link-Local as the Next-Hop Address for Static Routes

Sử dụng Unicast hoặc Link-local làm Next-Hop cho các tuyến tĩnh

Trường hợp sử dụng địa chỉ Global Unicast

```
! The first command is on router R1, listing R2's global unicast address
R1(config)# ipv6 route 2001:db8:1111:2::/64 2001:DB8:1111:4::2
! The next command is on router R2, listing R1's global unicast address
R2(config)# ipv6 route 2001:db8:1111:1::/64 2001:db8:1111:4::1
```

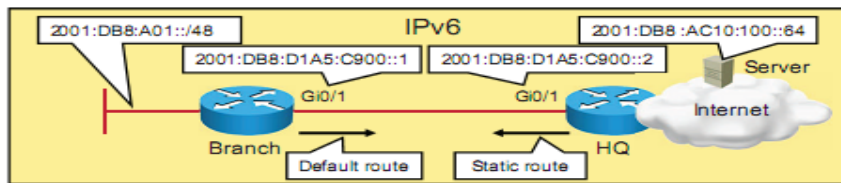
Trường hợp sử dụng Link-Local Neighbor address

Example 32-10 Static IPv6 Routes Using Link-Local Neighbor Addresses

```
! The first command is on router R1, listing R2's link-local address
R1(config)# ipv6 route 2001:db8:1111:2::/64 S0/0/0 FE80::FF:FE00:2
! The next command is on router R2, listing R1's link-local address
R2(config)# ipv6 route 2001:db8:1111:1::/64 S0/0/1 FE80::FF:FE00:1
```

Default route

Static Routing



The static IPv6 route is configured on the HQ router:

```
HQ(config)#ipv6 route 2001:DB8:A01::/48 G10/1 2001:DB8:D1A5:C900::1
```

The default IPv6 route is configured on the Branch router:

```
Branch(config)#ipv6 route ::/0 G10/1 2001:DB8:D1A5:C900::2
```

© 2013 Cisco Systems, Inc.

Xem bảng định tuyến

```
HQ#show ipv6 route static
IPv6 Routing Table - default - 4 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
        B - BGP, R - RIP, I1 - ISIS L1, I2 - ISIS L2
        IA - ISIS interarea, IS - ISIS summary, D - EIGRP, EX - EIGRP external
        ND - Neighbor Discovery, 1 - LISP
        O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
        ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
S 2001:DB8:A01::/48 [1/0]
  via 2001:DB8:D1A5:C900::1, GigabitEthernet0/1
```

Static Routing (Cont.)

Verify the default IPv6 route on the Branch router:

```
Branch#show ipv6 route static
IPv6 Routing Table - default - 4 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
        B - BGP, R - RIP, I1 - ISIS L1, I2 - ISIS L2
        IA - ISIS interarea, IS - ISIS summary, D - EIGRP, EX - EIGRP external
        ND - Neighbor Discovery, 1 - LISP
        O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
        ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
S ::/0 [1/0]
  via 2001:DB8:D1A5:C900::2, GigabitEthernet0/1
```

© 2013 Cisco Systems, Inc.

3. RIPng

Cấu hình:

B1: Chọn giao thức định tuyến

```
R(config)#ipv6 router rip tag
```

Trong đó: tag là một chuỗi định danh, do người cấu hình tự đặt

B2. Chọn cổng tham gia vào quá trình trao đổi thông tin định tuyến

```
R(config-if)#ipv6 rip tag enable
```

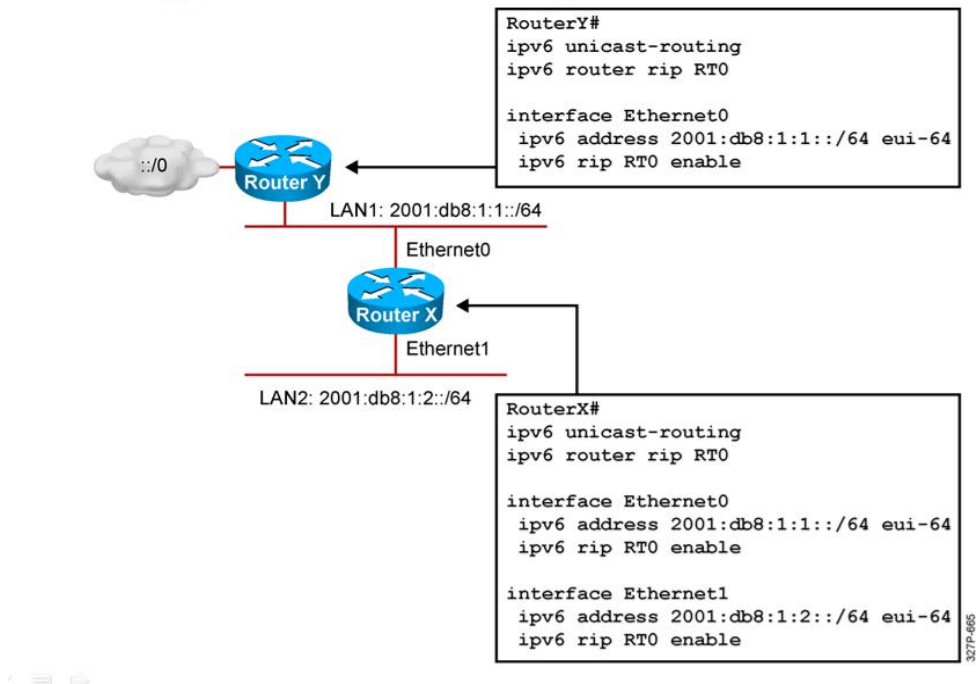
Các lệnh kiểm tra cấu hình:

```
R#show ipv6 rip
```

```
R#show ipv6 route [rip]
```

Ví dụ:

RIPng for IPv6 Configuration Example



4. OSPF cho Ipv6

Các bước cấu hình

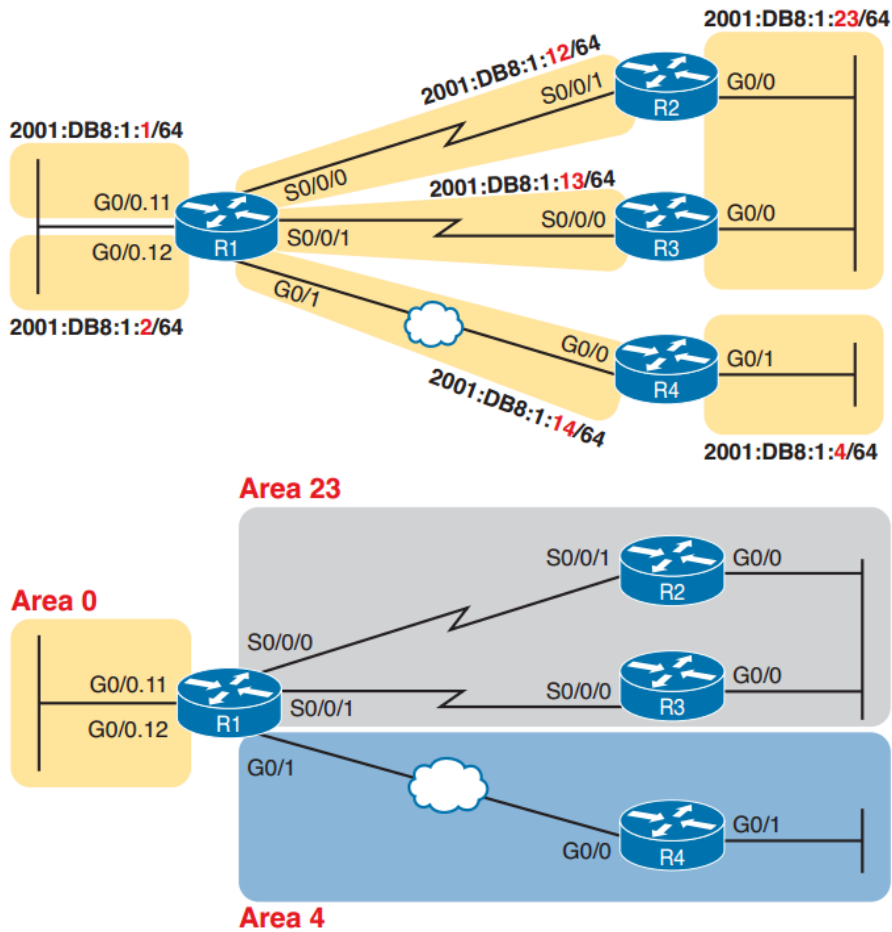
- B0: Bật tính năng định tuyến cho Ipv6

```
R(config)#ipv6 unicast-routing
```
- B1: Chọn giao thức định tuyến

```
R(Config)#ipv6 router ospf <process-id>
R(config-router)#router-id H.H.H.H
```
- B2. Chọn cổng tham gia vào quá trình trao đổi thông tin định tuyến

```
R(Config-if)#ipv6 ospf <process-id> area <area-id>
```

Ví dụ:



```

ipv6 unicast-routing
!
interface GigabitEthernet0/0
  mac-address 0200.0000.0004
  ipv6 address 2001:db8:1:14::4/64
  ipv6 ospf 4 area 4
!
interface GigabitEthernet0/1
  ipv6 address 2001:db8:1:4::4/64
  ipv6 ospf 4 area 4
!
ipv6 router ospf 4
  router-id 4.4.4.4
  passive-interface gigabitethernet0/1
  
```

```

ipv6 unicast-routing
!
interface GigabitEthernet0/0
  mac-address 0200.0000.0001
!
interface GigabitEthernet0/0.11
  encapsulation dot1q 11
  ipv6 address 2001:db8:1:1::1/64
  ipv6 ospf 1 area 0
!
interface GigabitEthernet0/0.12
  
```



```
encapsulation dot1q 12
ipv6 address 2001:db8:1:2::1/64
ipv6 ospf 1 area 0
!
interface GigabitEthernet0/1
ipv6 address 2001:db8:1:14::1/64
ipv6 ospf 1 area 4
!
interface serial 0/0/0
ipv6 address 2001:db8:1:12::1/64
ipv6 ospf 1 area 23
!
interface serial 0/0/1
ipv6 address 2001:db8:1:13::1/64
ipv6 ospf 1 area 23
!
ipv6 router ospf 1
router-id 1.1.1.1
```

Các lệnh kiểm tra cấu hình:

```
R#show ipv6 protocols
R#show ipv6 ospf neighbor
R#show ipv6 ospf database
R#show ipv6 route [ospf]
```

Địa chỉ multicast cập nhật thông tin định tuyến của OSPF cho Ipv6 là FF02::5 cho tất cả các router chạy OSPF và FF02::6 cho các DR và BDR. Để điều chỉnh cost dùng lệnh Router(config-if)#ipv6 ospf cost x (với x có giá trị từ 1- 65.535)

5. EIGRP for Ipv6

Các bước cấu hình:

- B1. Chọn giao thức định tuyến
R(config)#ipv6 router eigrp <AS>
R(config-router)#eigrp router-id H.H.H.H → tham số tùy chọn
- B2. Chọn cổng tham gia vào quá trình trao đổi thông tin định tuyến
R(config-if)#ipv6 eigrp <AS>

Các lệnh kiểm tra cấu hình:

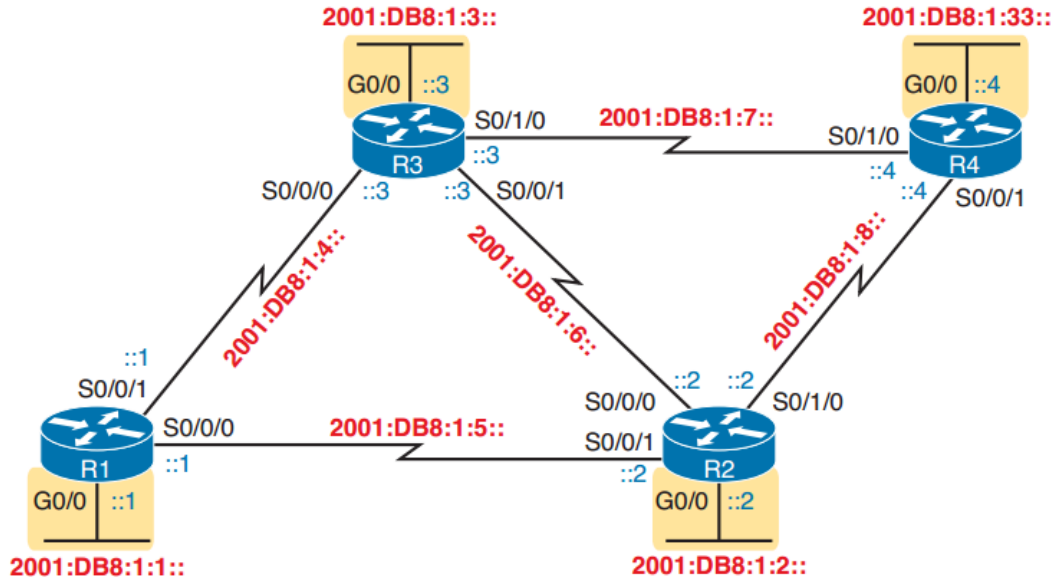
```
R#show ipv6 protocols
```



```
R#show ipv6 eigrp neighbors
R#show ipv6 eigrp topology
R#show ipv6 route [eigrp]
```

EIGRP cho Ipv6 cập nhật định tuyến qua địa chỉ multicast FF02::A

Ví dụ:



```
ipv6 unicast-routing
!
ipv6 router eigrp 1
  eigrp router-id 1.1.1.1
!
interface GigabitEthernet0/0
  ipv6 address 2001:db8:1:1::1/64
  ipv6 eigrp 1
!
interface serial 0/0/0
  description link to R2
  ipv6 address 2001:db8:1:5::1/64
  ipv6 eigrp 1
!
interface serial 0/0/1
  description link to R3
  ipv6 address 2001:db8:1:4::1/64
  ipv6 eigrp 1
```

Example 24-2 *EIGRP for IPv6 Configuration on R2*

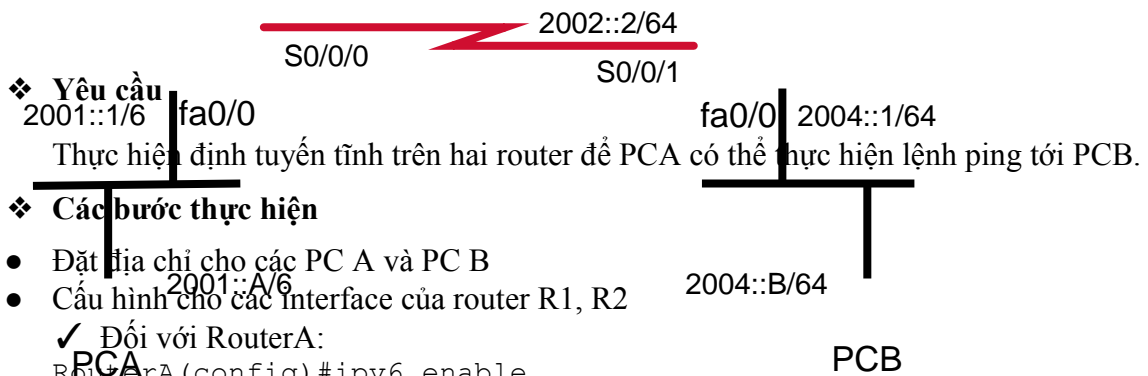
```
ipv6 unicast-routing
!
ipv6 router eigrp 1
  eigrp router-id 2.2.2.2
!
interface GigabitEthernet0/0
  ipv6 address 2001:db8:1:2::2/64
```

```

ipv6 eigrp 1
!
interface serial 0/0/0
description link to R3
ipv6 address 2001:db8:1:6::2/64
ipv6 eigrp 1
!
interface serial 0/0/1
description link to R1
ipv6 address 2001:db8:1:5::2/64
ipv6 eigrp 1
!
interface serial 0/1/0
description link to R4
ipv6 address 2001:db8:1:8::2/64
ipv6 eigrp 1
    
```

6. Lab. Routing for IPv6

Lab. 8-1. STATIC ROUTING CHO IPv6



```

RouterA(config)#ipv6 unicast-routing
--> cho phép router hoạt động định tuyến với IPv6

RouterA(config)#interface f0/0
RouterA(config-if)#ipv6 address 2001::1/64
RouterA(config-if)#no shutdown
RouterA(config-if)#exit

RouterA(config)#interface S0/0/0
RouterA(config-if)#clock rate 64000
    
```

```
RouterA(config-if)#ipv6 address 2002::1/64
RouterA(config-if)#no shutdown
RouterA(config-if)#end
```

✓ Đối với RouterB:

```
RouterB(config)#ipv6 unicast-routing
RouterB(config)#interface F0/0
RouterB(config-if)#ipv6 address 2004::1/64
RouterB(config-if)#no shutdown
RouterB(config-if)#exit
RouterB(config)#interface S0/0/1
RouterB(config-if)#ipv6 address 2002::2/64
RouterB(config-if)#no shutdown
RouterB(config-if)#end
```

● Cấu hình định tuyến tĩnh

✓ Trên RouterA:

```
RouterA(config)#ipv6 route 2004::/64 2002::2
RouterA(config)#exit
```

✓ Trên RouterB:

```
R2(config)#ipv6 route 2001::/64 2002::1
R1(config)#exit
```

Lab. 8-2. CẤU HÌNH RIPng

2002::2/64 2003::1/64 2003::2/64

Giao thức định tuyến ~~RIPng~~ (~~RIP next generation~~) trong IPv6 tương tự như RIP trong IPv4. Các gói tin update sử dụng địa chỉ multicast của tất cả các router chạy IPv6.

❖ **Yêu cầu**

2001::1/64 fa0/0 fa0/0 2004::1/64

S0/0/0 S0/0/1 S0/0/0 S0/0/1

- Router RouterA, RouterB, RouterC sử dụng RIPng để quảng bá thông tin định tuyến
- Các PC ping được toàn bộ các địa chỉ trong mạng

❖ **Các bước thực hiện**

- Đặt địa chỉ IPv6 cho các PC

2001::A/6 2001::B/6 2004::C/64

PC PC PC

- **Cấu hình địa chỉ IPv6 cho các interface của router RouterA, RouterB**

- ✓ **Cấu hình cho RouterA:**

```
RouterA(config)#ipv6 unicast-routing
RouterA(config)#interface fa0/0
RouterA(config-if)#ipv6 address 2001::1/64
RouterA(config-if)#no shutdown
RouterA(config-if)#exit
RouterA(config)#interface S0/0/0
RouterA(config-if)#clock rate 64000
RouterA(config-if)#ipv6 address 2002::1/64
RouterA(config-if)#no shutdown
RouterA(config-if)#end
```

- ✓ **Cấu hình cho RouterB:**

```
RouterB(config)#ipv6 unicast-routing
RouterB(config)#interface S0/0/1
RouterB(config-if)#ipv6 address 2002::2/64
RouterB(config-if)#no shutdown
RouterB(config-if)#exit
RouterB(config)#interface S0/0/0
RouterB(config-if)#ipv6 address 2003::1/64
RouterB(config-if)#no shutdown
```

- ✓ **Cấu hình cho RouterC:**

```
RouterC(config)#ipv6 unicast-routing
RouterC(config)#interface S0/0/1
RouterC(config-if)#ipv6 address 2003::2/64
RouterC(config-if)#no shutdown
RouterC(config-if)#exit
RouterC(config)#interface fa0/0
RouterC(config-if)#ipv6 address 2004::1/64
RouterC(config-if)#no shutdown
```

- **Cấu hình định tuyến RIPng**

- ✓ **Trên RouterA:**

```
RouterA(config)#ipv6 router rip cisco
```

đặt tên cho process RIPng là cisco

Chú ý: tên của process-id chỉ mang ý nghĩa cục bộ trong một router

```
RouterA(config-router)#exit
```

```
RouterA(config)#interface f0/0
```

```
RouterA(config-if)#ipv6 rip cisco enable
```

→ Chạy giao thức RIPng process “cisco” trên f0/0

```
RouterA(config)#interface S0/0/0
```

```
RouterA(config-if)#ipv6 rip cisco enable
```

→ Chạy giao thức RIPng process “cisco” trên S0/0/0

- Cấu hình cho RouterB tương tự như RouterA
- Cấu hình cho RouterC tương tự như RouterA

❖ Kiểm tra cấu hình

- Xem bảng định tuyến bằng lệnh

```
RouterA#show ipv6 route
```

hoặc

```
RouterA#show ipv6 route rip
```

--> xem các đường đi học từ RIPng

- Xem process-id của RIPng, các interface nào chạy RIPng

```
RouterA#show ipv6 rip
```

- Xem cơ sở dữ liệu của RIPng bằng lệnh show ipv6 rip database

```
RouterA#show ipv6 rip database
```

- Từ RouterA, RouterB hoặc RouterC ping được các địa chỉ trên mạng
- Dùng lệnh debug ipv6 packet [detail] để xem quá trình gói tin đi từ RouterA đến PC C

CHƯƠNG 9. WAN

Chương này sẽ đề cập đến một số giao thức hoạt động trên môi trường WAN và một số dịch vụ WAN phổ biến. Học xong chương này, người học có khả năng:

- Trình bày được khái niệm về WAN
- Trình bày được một số đặc điểm cơ bản của giao thức PPP, HDLC
- Phân biệt và cấu hình hai giao thức chứng thực trên PPP (PAP, CHAP)
- Phân biệt và cấu hình một số kỹ thuật WAN: Serial Point-to-Point, Frame Relay

1. Giới thiệu

Một WAN là một mạng trao đổi dữ liệu, nó hoạt động vượt ra ngoài phạm vi vật lý của LAN. WAN hoạt động trên một miền địa lý rộng lớn, kết nối hệ thống máy tính của cùng một đơn vị giữa các tỉnh, các quốc gia hay châu lục...

WAN sử dụng các liên kết dữ liệu như là Frame Relay, ATM, MPLS hỗ trợ các dịch vụ để truy cập băng thông vượt qua vùng địa lý rộng lớn.

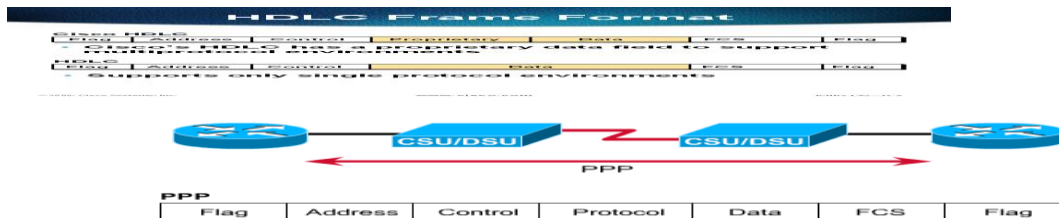
Các tính năng kỹ thuật WAN nằm ở ba tầng cuối cùng của mô hình OSI.

- Các kiểu kết nối WAN (layer 1): đường thuê riêng (leased line), chuyển mạch kênh (circuit switched), chuyển mạch gói (packet-switched).
- Các giao thức đóng gói WAN (Layer 2): HDLC, PPP, ATM, Frame Relay, VPN, MPLS
- Một số kỹ thuật WAN: trước đây một số kỹ thuật được dùng như ISDN, X.25, ATM, các kỹ thuật đang sử dụng nhiều hiện nay như DSL, Leased lined, MPLS,...

Trong chương này, chúng ta sẽ tìm hiểu về 2 kỹ thuật WAN: PPP và Frame-Relay.

2. Kết nối serial point-to-point

Hai giao thức liên kết dữ liệu (data link) WAN sử dụng trong mạng WAN kết nối Serial Point-to-Point được dùng phổ biến là HDLC và PPP.



PPP là một giao thức thường được chọn để triển khai trên một kết nối WAN nối tiếp. PPP có hỗ trợ quá trình xác thực PAP và CHAP.

❖ Quá trình chứng thực trong PPP

PPP tổ chức gồm 2 giao thức sau:

- *Link Control Protocol (LCP)*: sử dụng cho việc thiết lập, cấu hình và kiểm tra kết nối ở tầng liên kết dữ liệu.
- *Network Control Protocol (NCP)*: sử dụng cho việc thiết lập và cấu hình các giao thức tầng mạng khác nhau.

❖ Quá trình thiết lập kết nối PPP

Quá trình thiết lập kết nối PPP qua 4 bước: Thiết lập kết nối và thương lượng cấu hình; quyết định chất lượng kết nối; thương lượng cấu hình giao thức tầng mạng và kết thúc kết nối.

- *Thiết lập kết nối và cấu hình*

Mỗi thiết bị PPP gửi gói tin LCP để cấu hình và thiết lập kết nối ở tầng liên kết dữ liệu. Gói tin LCP chứa các trường: “MTU”, “compression”, và giao thức chứng thực kết nối. LCP đầu tiên mở kết nối và thương lượng các tham số cấu hình. Giai đoạn này hoàn tất khi các gói tin thống nhất cấu hình (ACK) được gửi và nhận.

- *Quyết định chất lượng kết nối*

Liên kết được kiểm tra xem có tốt không để chuyển các giao thức lên tầng mạng hay không. Sau đó *Client* có thể được chứng thực. Việc chứng thực diễn ra trước giai đoạn cấu hình giao thức tầng mạng. PPP hỗ trợ hai giao thức chứng thực là: PAP và CHAP.

- *Thương lượng cấu hình tầng mạng*

Các thiết bị PPP gửi gói tin NCP để chọn và cấu hình một hoặc nhiều giao thức tầng mạng (ví dụ như IP). Khi giao thức tầng mạng được cấu hình, các gói tin từ giao thức tầng mạng có thể được gửi qua liên kết. Nếu LCP kết thúc kết nối, nó cung cấp các giao thức tầng mạng để có thể có những hành động phù hợp.

- *Kết thúc kết nối*

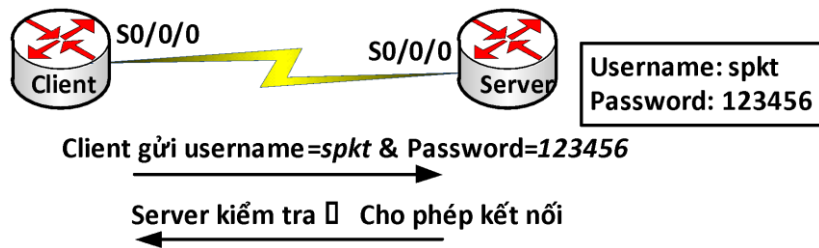
LCP có thể kết thúc kết nối bất cứ lúc nào. Điều này luôn được thực hiện ở yêu cầu của người dùng. Kết thúc kết nối cũng có thể xảy ra do sự cố vật lý, như là đứt kết nối hay vượt quá thời gian qui định (timeout).

❖ Giao thức chứng thực PAP và CHAP

- *Chứng thực PPP bằng PAP*

PAP sử dụng cơ chế bắt tay 2 bước. Đầu tiên *Client* sẽ gửi *username* và *password* cho *Server* để xác thực. *Server* sẽ tiến hành kiểm tra, nếu thành công thì sẽ thiết lập kết nối; ngược lại sẽ không thiết lập kết nối với *Client*.

Password được gửi dưới dạng không được mã hóa (clear – text) và *username/password* được gửi đi kiểm tra một lần khi thiết lập kết nối.



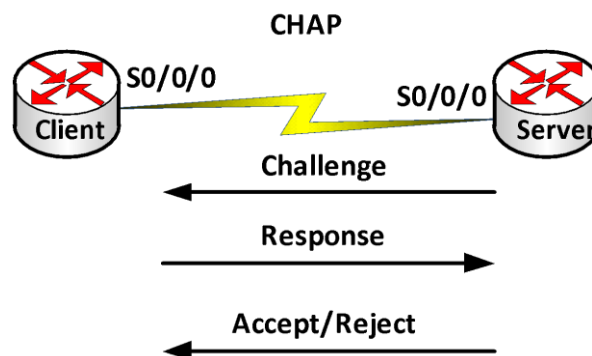
- Chứng thực PPP bằng CHAP

Sử dụng kỹ thuật 3 bước bắt tay (three-way handshake). CHAP được thực hiện ở lúc bắt đầu thiết lập kết nối và luôn được lặp lại trong suốt quá trình kết nối được duy trì.

Client muốn thiết lập kết nối với *Server*, *Server* gửi một thông điệp “challenge” yêu cầu *Client* gửi giá trị để *Server* chứng thực. Thông điệp gửi từ *Server* có chứa một số ngẫu nhiên dùng làm đầu vào cho thuật toán “hash”.

Client nhận được thông điệp yêu cầu của *Server*. Nó sẽ sử dụng thuật toán “hash” với đầu vào là *hostname*, *password* và số ngẫu nhiên vừa nhận được và tính toán ra một giá trị nào đó và gửi giá trị này qua cho *Server*.

Server sẽ kiểm tra danh sách “username” (nếu cấu hình nhiều username) để tìm ra “username” nào giống với *hostname* của *Client*. Sau khi tìm được “username” đó, nó dùng thuật toán “hash” để mã hóa *password* tương ứng và số ngẫu nhiên trong thông điệp “challenge” ban đầu mà nó gửi cho *Client* để tính ra một giá trị nào đó. Và giá trị này sẽ so sánh với giá trị do *Client* gửi qua, nếu giống nhau thì xác thực thành công; nếu không thì kết nối sẽ bị xóa ngay.



Một cách đơn giản, ta cần nắm ý tưởng sau khi cấu hình CHAP: mỗi đầu kết nối phải có khai báo *username* và *password*. *Username* bên R1 phải là *hostname* của R2 và *username* khai báo bên R2 là *hostname* của R1, *password* hai bên phải giống nhau.

❖ Cấu hình PPP

- Cấu hình PPP

```
Router(config)#interface <interface>
```

```
Router(config-if)#encapsulation ppp
```

- Cấu hình chứng thực PPP PAP

Bước 1: Tạo username và password trên Server

```
Router(config)#username <username> password <password>
```

Bước 2: Enable PPP

```
Router(config-if)#encapsulation ppp
```

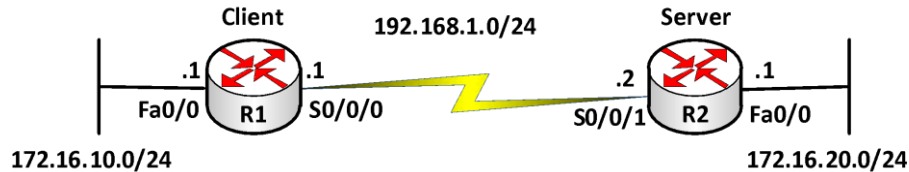
Bước 3: Cấu hình xác thực

```
Router(config-if)#ppp authentication {pap|chap|pap-chap|chap-
pap}
```

Bước 4: PAP phải được enable trên interface bằng lệnh

```
Router(config-if)#ppp pap sent-username <username> password
<password>
```

Ví dụ 1: Cấu hình PPP chứng thực bằng PAP



- **Mô tả**

Router R2 sẽ chứng thực cho router R1 bằng giao thức PAP

- **Hướng dẫn cấu hình**

- **Cấu hình cơ bản**

```
R1(config)#int S0/0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#exit
R2(config)#int S0/0/1
R2(config-if)#ip address 192.168.1.2 255.255.255.0
R2(config-if)#exit
```

- **Cấu hình chứng thực PAP**

```
R1(config)#int S0/0/0
R1(config-if)#encapsulation ppp
R1(config-if)#ppp pap sent-username cisco password cisco
R2(config)#username cisco password cisco
R2(config)#int S0/0/1
R2(config-if)#encapsulation ppp
R2(config-if)#ppp authentication pap
```

- **Cấu hình định tuyến:** tùy chọn giao thức

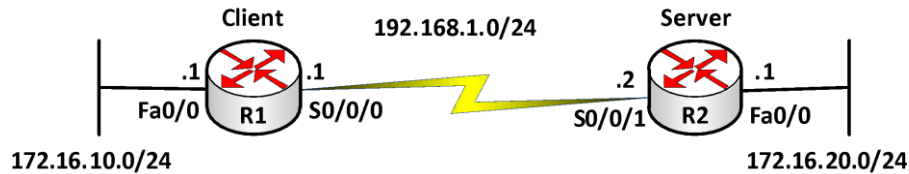
- **Kiểm tra cấu hình**

Sử dụng các lệnh sau:

```
ping
debug ppp authentication
```

- ❖ **Cấu hình chứng thực PPP CHAP**

Trường hợp 1: Các router dùng hostname để chứng thực



- **Mô tả**

Router R2 chứng thực cho router R1 bằng giao thức CHAP. Trường hợp mặc định, router gửi hostname để chứng thực.

- **Các bước cấu hình**

- ✓ **Cấu hình cơ bản**

```
R1(config)#int S0/0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#exit
R2(config)#int S0/0/1
R2(config-if)#ip address 192.168.1.2 255.255.255.0
R2(config-if)#exit
```

- **Cấu hình chứng thực CHAP**

```
R1(config)#username R2 password cisco
R1(config)#int S0/0/0
R1(config-if)#encapsulation ppp
R2(config)#username R1 password cisco
R2(config)#interface serial 0/0/1
R2(config-if)#encapsulation ppp
R2(config-if)#ppp authentication chap
```

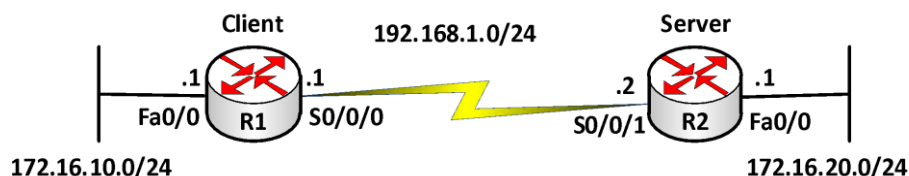
- **Cấu hình định tuyến:** tùy chọn giao thức

- **Kiểm tra cấu hình**

Sử dụng các lệnh sau:

```
Router#ping
Router#debug ppp authentication
```

Trường hợp 2: Các router gửi username & password bất kỳ



- **Yêu cầu**

Router R2 sẽ chứng thực cho router R1 bằng giao thức CHAP trường hợp router gửi hostname và password được chỉ ra.

- **Các bước cấu hình**

- **Cấu hình cơ bản:**

```
R1(config)#interface serial 0/0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#exit
```

```
R2(config)#interface serial 0/0/1
R2(config-if)#ip address 192.168.1.2 255.255.255.0
R2(config-if)#exit
```

- **Cấu hình chứng thực CHAP**

```
R1(config)#int S0/0/0
R1(config-if)#encapsulation ppp
R1(config-if)#ppp chap hostname abc
R1(config-if)#ppp chap password cisco
R2(config)#username abc password cisco
R2(config)#int S0/0/1
R2(config-if)#encapsulation ppp
R2(config-if)#ppp authentication chap
```

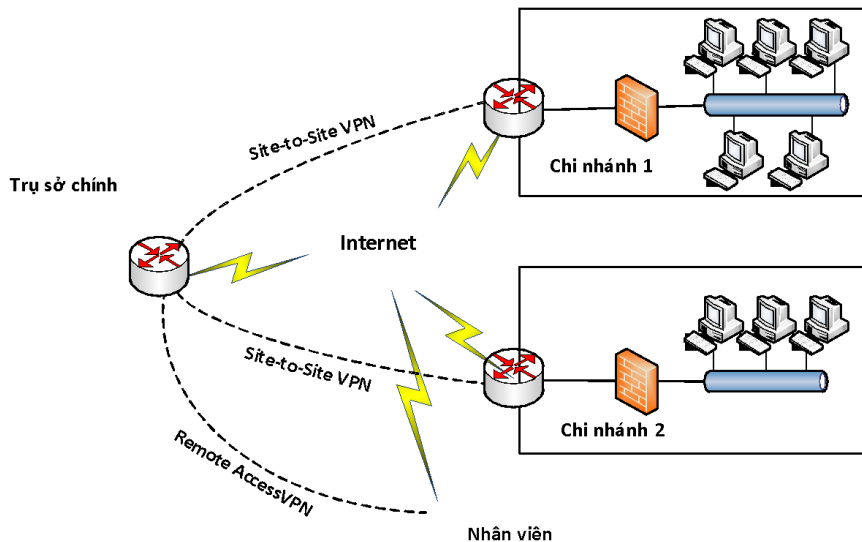
- **Cấu hình định tuyến:** tùy chọn giao thức
- **Kiểm tra cấu hình**

Sử dụng các lệnh sau:

```
Router#ping
Router#debug ppp authentication
```

3. VPN

VPN là sự mở rộng của một mạng riêng (private network) thông qua mạng công cộng (internet), được dùng để kết nối các văn phòng chi nhánh, người từ xa kết nối về văn phòng chính.



VPN có thể được tạo ra bằng cách sử dụng phần cứng, phần mềm hay kết hợp cả hai để tạo ra một kết nối ảo bảo mật giữa hai mạng riêng thông qua mạng công cộng. Lợi ích của công nghệ VPN là đáp ứng nhu cầu trao đổi thông tin, truy cập từ xa và tiết kiệm chi phí.

❖ Các mode kết nối VPN

Có hai chế độ kết nối VPN để chuyển dữ liệu giữa hai thiết bị là

- Tunnel mode
- Transport mode

Cả hai mode này định nghĩa quá trình đóng gói được sử dụng để di chuyển dữ liệu một cách an toàn giữa hai thực thể.

● Transport mode

Một kết nối ở *mode transport* được sử dụng địa chỉ IP nguồn và đích thật sự của các thiết bị trong các gói tin để truyền dữ liệu.

● Tunnel mode

Hạn chế của *transport mode* là không có khả năng mở rộng. Do đó, nếu chúng ta có nhiều thiết bị ở hai vị trí riêng biệt cần nói chuyện với nhau trong chế độ bảo mật, ta nên sử dụng *tunnel mode* thay vì *transport-mode*. Trong *tunnel mode*, các thiết bị nguồn-đích thực thông thường sẽ không bảo vệ dữ liệu, thay vào đó các thiết bị trung gian được sử dụng để bảo vệ luồng dữ liệu. Các thiết bị này được gọi là các *VPN gateway*.

Tunnel mode cung cấp nhiều tính năng ưu việt hơn *transport mode*:

- *Tính mở rộng*: ta có thể chọn một thiết bị phù hợp để thực hiện việc xử lý bảo vệ.
- *Tính linh động*: không cần phải thay đổi gì trong cấu hình VPN khi thêm vào một thiết bị mới sau *VPN Gateway*.
- *Tính ẩn của các giao tiếp*: các lưu lượng được các *VPN Gateway* đại diện trao đổi với nhau, vì vậy sẽ che dấu nguồn và đích thật sự của kết nối.
- *Sử dụng địa chỉ cục bộ*: các thiết bị đích và nguồn thực có thể sử dụng địa chỉ được đăng ký (public) hay cục bộ bởi vì các gói tin được đóng gói bởi các *VPN Gateway*.

- Sử dụng các chính sách bảo mật hiện có: các chính sách bảo mật được thực hiện trên các thiết bị tường lửa và bộ lọc gói tin.

CÁC THÀNH PHẦN CỦA VPN:

❖ Chứng thực

Có 2 loại chứng thực là: Chứng thực thiết bị và chứng thực người dùng

- Chứng thực thiết bị: cho phép hạn chế các truy cập vào hệ thống mạng dựa vào các thông tin cung cấp bởi các thiết bị VPN đầu xa.

Có 2 dạng chứng thực kiểu này là: *Pre-shared key*, *Digital signature* hoặc *certificate*.

Pre-shared key được sử dụng trong các môi trường VPN nhỏ. Một hay nhiều khóa được cấu hình và dùng để chứng thực để nhận dạng một thiết bị. *Digital signature*, *digital certificate* được sử dụng để chứng thực thiết bị trong các môi trường triển khai VPN lớn.

- Chứng thực người dùng: chỉ cho phép người dùng hợp lệ kết nối và truy cập hệ thống VPN. Người dùng phải cung cấp *username* và *password*.

❖ Phương pháp đóng gói

Làm thế nào mà thông tin người dùng, dữ liệu được đóng gói và vận chuyển qua mạng. Các câu hỏi cần đặt ra là: Các trường (field) gì sẽ tồn tại trong VPN header và VPN trailer, thứ tự xuất hiện các trường, kích thước của các trường?

❖ Mã hóa dữ liệu

Mã hóa dữ liệu giải quyết vấn đề dữ liệu bị đánh cắp trên đường truyền. Mã hóa dữ liệu chỉ đơn giản là lấy dữ liệu, một giá trị khóa và chạy thuật toán mã hóa để làm cho dữ liệu trở thành dạng khác với nội dung ban đầu. Chỉ có thiết bị có cùng khóa mới có thể giải mã được thông tin về dạng ban đầu. Một số thuật toán mã hóa như *DES*, *3DES*, *RSA*, *AES*, *RC4*...

❖ Toàn vẹn dữ liệu

Có thể xảy ra tình trạng có các gói tin giả làm tăng sự hoạt động lãng phí của CPU. VPN cung cấp một cơ chế để khắc phục là kiểm tra sự toàn vẹn của dữ liệu, hay còn gọi là “packet authentication”. Với “packet authentication”, một chữ ký (signature) được đóng vào các gói tin. *Signature* được tạo ra bằng cách lấy nội dung từ gói tin, một “share-key” và chạy thông tin này qua một hàm băm và xuất ra một giá trị gọi là *digital signature*. *Signature* này được thêm vào các gói tin và gửi đi đến đích. Ở đích đến sẽ kiểm tra “signature”, và nếu “signature” được kiểm tra là chính xác, nó sẽ giải mã nội dung gói tin.

Hai trong số các hàm băm được sử dụng cho việc kiểm tra toàn vẹn dữ liệu là *SHA* và *MD5*.

❖ Quản lý khóa

Chúng ta đã đề cập đến 3 thành phần VPN có sử dụng khóa là: *chứng thực*, *mã hóa* và *hàm băm*. Việc quản lý khóa trở nên quan trọng trong các kết nối VPN. Ví dụ như: làm thế nào để phân phối các khóa, chúng được cấu hình tĩnh hay phát sinh ngẫu nhiên, các khóa được tạo lại bao nhiêu lần để tăng tính bảo mật?

❖ Non-repudiation

Repudiation là nơi ta không thể chứng thực các giao tiếp xảy ra (như là việc thiết lập kết nối). *Non-repudiation* trái ngược với điều này: ta có thể chứng thực một giao tiếp xảy ra giữa hai bên kết nối.

Ví dụ khi ta vào một cửa hàng online như Amazone.com và mua một quyển sách và thanh toán bằng credit card. Amazone sẽ phải thu gom thông tin cá nhân khi ta điền vào đơn đặt hàng như tên, địa chỉ, số điện thoại, thông tin về credit card... Khi đó, Amazone sẽ kiểm tra những thông tin đó với công ty phân phối *credit card* và lưu giữ lại các thông tin giao dịch như ngày, tháng, ...

❖ Hỗ trợ ứng dụng và giao thức

Khi lựa chọn cài đặt VPN, đầu tiên chúng ta cần phải xác định loại dữ liệu nào cần được bảo vệ. Ví dụ như loại dữ liệu IP hay IPX hoặc cả hai, hoặc là chỉ cần bảo vệ một số loại dữ liệu cho một số chương trình ứng dụng nào đó như Web hay Email,...

❖ Quản lý địa chỉ:

Quản lý địa chỉ là một vấn đề quan trọng trong việc hoạch định địa chỉ cho toàn hệ thống mạng của công ty.

PHÂN LOẠI VPN: Có 2 loại VPN thông dụng

- *Site-to-Site VPN*
- *Remote Access VPN*

❖ Remote Access VPN

Remote access VPN thường được sử dụng cho các kết nối có băng thông thấp giữa một thiết bị của người dùng như là PC, Ipad,... và một thiết bị *Gateway VPN*. *Remote access VPN* thông thường sử dụng *tunnel mode* cho các kết nối.

Người dùng ở xa sử dụng các phần mềm VPN để truy cập vào mạng của công ty thông qua Gateway hoặc VPN concentrator (bản chất là một server), giải pháp này thường được gọi là client/server. Trong giải pháp này, người dùng thường sử dụng các công nghệ truyền thống để tạo lại các tunnel về mạng của họ.

Một phần quan trọng của thiết kế này là việc thiết kế quá trình xác thực ban đầu nhằm đảm bảo là yêu cầu được xuất phát từ một nguồn tin cậy. Thường thì giai đoạn ban đầu này dựa trên cùng một chính sách về bảo mật của công ty.

Trong Remote Access VPN có nhiều kỹ thuật được sử dụng để bảo mật trong việc trao đổi dữ liệu: IPSec, SSL,...

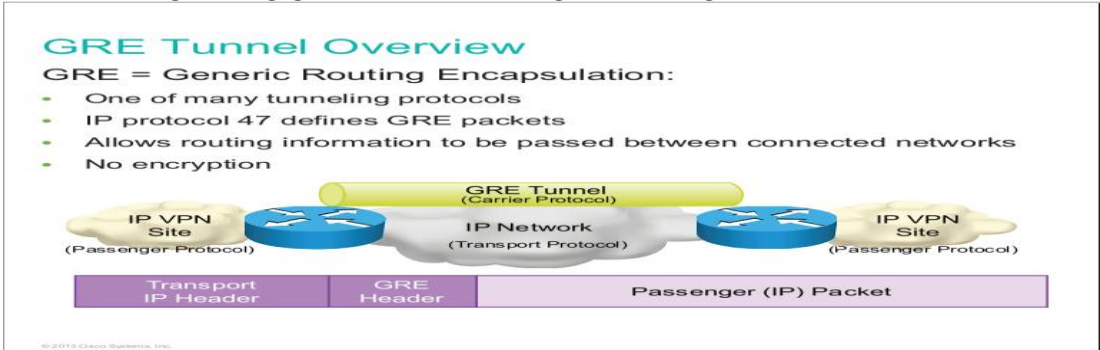
❖ Site-to-Site VPN

Site-to-site VPN (LAN-to-LAN) là kỹ thuật kết nối các hệ thống mạng (site) của cùng một công ty ở các nơi khác nhau tạo thành một hệ thống mạng thống nhất thông qua môi trường mạng công cộng. Trong trường hợp này, quá trình xác thực ban đầu cho những người dùng

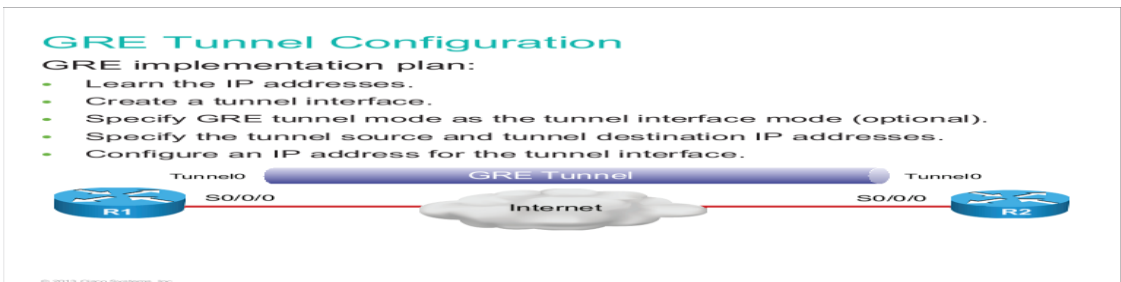
cần phải được kiểm soát chặt chẽ bởi các thiết bị ở các site tương ứng. Các thiết bị này hoạt động như *Gateway*, truyền lưu lượng một cách an toàn cho đầu bên kia.

Giao thức GRE:

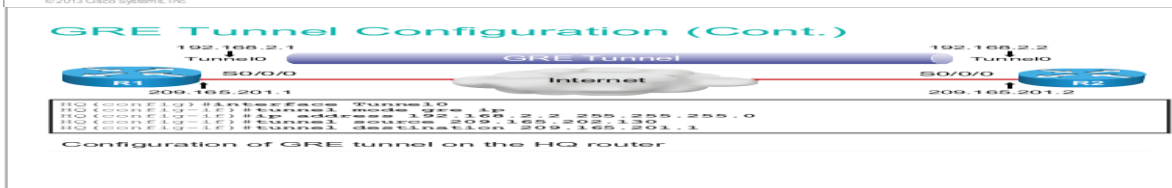
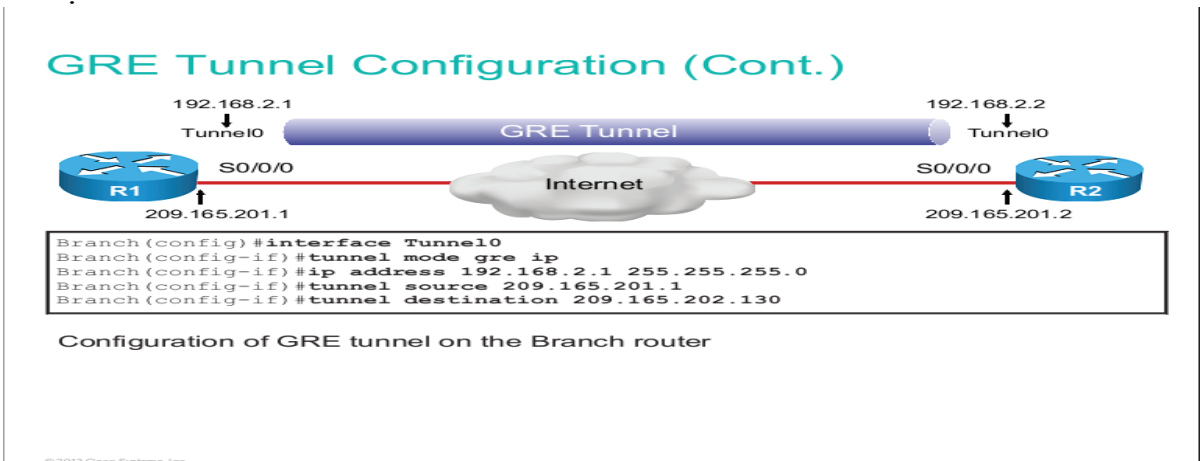
Là một trong những giao thức tạo đường hầm trong VPN



Cấu hình GRE



Ví dụ:



Kiểm tra cấu hình:

GRE Tunnel Verification

```
Branch#show ip interface brief | include Tunnel
Tunnel0      192.168.2.1      YES manual  up      up
```

Verifies that the tunnel interface is up.

```
Branch#show interface Tunnel 0
Tunnel0 is up, line protocol is up
Hardware is Tunnel
Internet address is 192.168.2.1/24
MTU 17916 bytes, BW 100 Kbit/sec, DLY 50000 usec,
  reliability 255/255, txload 1/255, rxload 1/255
Encapsulation TUNNEL, loopback not set
Keepalive not set
Tunnel source 209.165.201.1, Destination 209.165.202.130
Tunnel protocol/transport GRE/IP
<output omitted>
```

Verifies that the tunnel interface is up and shows tunnel IPs, source and destination IPs, and tunnel protocol.

© 2013 Cisco Systems, Inc.

GRE Tunnel Verification (Cont.)

```
Branch#show ip route
<output omitted>
192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
I   192.168.2.0/24 is directly connected, Tunnel0
C   209.165.201.0/24 is variably subnetted, 2 subnets, 2 masks
L   209.165.201.0/27 is directly connected, GigabitEthernet0/1
L   209.165.201.1/32 is directly connected, GigabitEthernet0/1
```

Verifies the tunnel route between the Branch and HQ routers

© 2013 Cisco Systems, Inc.

3. Tổng kết chương

Các kỹ thuật WAN hoạt động trong phạm vi rộng và phức tạp hơn trong các mạng LAN. Các kỹ thuật WAN được sử dụng phổ biến như: ISDN, leased line, X.25, Frame Relay, ATM, DSL, ... Các kiểu đóng gói thường dùng trong mạng WAN là: HDLC, PPP, Frame Relay, ...

Hai giao thức dùng để chứng thực trên PPP trong môi trường WAN là PAP và CHAP. PAP có độ bảo mật kém vì nó gửi *username/password* dưới dạng không mã hóa và việc chứng thực chỉ diễn ra một lần. Đối với CHAP, tham số chứng thực được gửi đi dưới dạng mã hóa và việc chứng thực được lặp lại trong suốt quá trình kết nối.

Công nghệ VPN được sử dụng phổ biến hiện nay. Nó cung cấp kết nối an toàn và hiệu quả để truy cập vào tài nguyên nội bộ từ nhân viên ở bên ngoài thông qua Internet vào hệ thống mạng công ty hay kết nối các chi nhánh của công ty để trao đổi dữ liệu với nhau.

Có nhiều cách phân loại VPN. Trong giáo trình này trình bày 2 loại VPN thông dụng: *Remote access VPN*, *Site-to-Site VPN*.

4. Câu hỏi và bài tập

4.1 CHAP sử dụng thuật toán nào sau đây để tạo giá trị gửi đến cho “remote peer” trong các bước chứng thực.

- A. 3DES
- B. DES
- C. SHA
- D. MD5

4.2 Những kiểu đóng gói ở tầng Data-Link nào sau đây sử dụng trên các cổng WAN?

- A. Ethernet
- B. PPP
- C. Token Ring
- D. HDLC
- E. Frame Relay
- F. POTS

4.3 Các loại LMI nào sau đây sử dụng trong Frame Relay?

- A. Q.931
- B. IEEE
- C. Cisco
- D. IETF
- E. Q933a
- F. ANSI

4.4 Trong mạng Frame Relay; mục đích của “Inverse ARP” là gì?

- A. Nó được để ánh xạ giữa một địa chỉ IP với địa chỉ MAC
- B. Nó được để ánh xạ giữa một DLCI với địa chỉ MAC
- C. Nó được để ánh xạ giữa một địa chỉ MAC với địa chỉ IP
- D. Nó được để ánh xạ giữa một DLCI với địa chỉ IP
- E. Nó được để ánh xạ giữa một địa chỉ MAC với DLCI

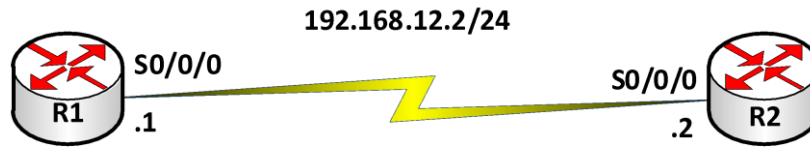
4.5 Sau khi các router được cấu hình Frame Relay, quản trị dùng lệnh *show frame relay map*, kết quả như sau:

```
Router#show frame-relay map
Serial0/0/0 (up): ip 192.168.12.1 dlci 100 (0x64, 0x1840), dynamic
                  Broadcast, status defined, active
```

Hãy cho biết ý nghĩa của từ *dynamic* trong kết quả trên

- A. Cổng Serial 0/0/0 đang gửi các gói tin
- B. DLCI 100 được router tự động sinh ra
- C. Cổng Serial 0/0/0 được DHCP server gán IP là 192.168.12.1
- D. DLCI 100 sẽ tự động thay đổi để đáp ứng yêu cầu thay đổi trên mạng Frame Relay
- E. Một ánh xạ giữa DLCI 100 và “remote router” có IP là 192.168.12.1 được học qua “Inverse ARP”

4.6 Cho mô hình mạng sau:



Hai router đã cấu hình chính xác địa chỉ IP như mô tả trên hình và thông tin trên cổng Serial0/0/0 của các router khi thực hiện lệnh show interface S0/0/0 như sau:

```
R1#show interface s0/0/0
  Serial0/0/0 is up, line protocol is down
  Hardware is HD64570
  Internet address is 192.168.12.1/24
  MTU 1500 bytes, BW 1433 Kbit
  Reliable 255/255
  Encapsulation HDLC, loopback not set
  Keepalive set(10sec)
R2#show interface s0/0/0
  Serial0/0/0 is up, line protocol is down
  Hardware is HD64570
  Internet address is 192.168.12.2/24
  MTU 1500 bytes, BW 1433 Kbit
  Reliable 255/255
  Encapsulation PPP, loopback not set
  Keepalive set(10sec)
```

Hai router trên không liên lạc với nhau được. Bạn hãy cho biết nguyên nhân vì sao?

- A. PCP không ở trạng thái “open”
- B. Subnet mask cấu hình sai
- C. Đóng gói không tương thích
- D. Băng thông được cấu hình quá thấp
- E. Địa chỉ IP cấu hình chưa đúng

4.7 Trong một hệ thống mạng cấu hình giao thức PPP và chứng thực bằng CHAP qua kết nối WAN. Lệnh nào sau đây dùng để hiển thị các chứng thực CHAP trong thời gian thực?

- A. show pp authentication
- B. debug pap authentication
- C. debug ppp authentication
- D. show chap authentication

5.8 Cho mô hình mạng sau:



Hai router này đã được cấu hình chứng thực PPP CHAP như sau:

```
Saigon(config)#username Hanoi password spkt@123
Saigon(config)#interface Serial0/0/0
Saigon(config-if)#encapsulation ppp
Saigon(config-if)#ppp authentication chap
Hanoi(config)#username Saigon password spkt@124
Hanoi(config)#interface Serial0/0/0
Hanoi(config-if)#encapsulation ppp
Hanoi(config-if)#ppp authentication chap
```

Hai router không chứng thực thành công. Hỏi lý do tại sao?

- A. Cấu hình username không đúng trên hai router
- B. Password cấu hình không giống nhau trên hai router
- C. Chứng thực CHAP không thể cấu hình được trên cổng Serial
- D. Các router không thể tạo kết nối được từ cổng Serial 0/0/0 đến Serial 0/0/0
- E. Chứng thực CHAP không cho phép các router chứng thực lẫn nhau

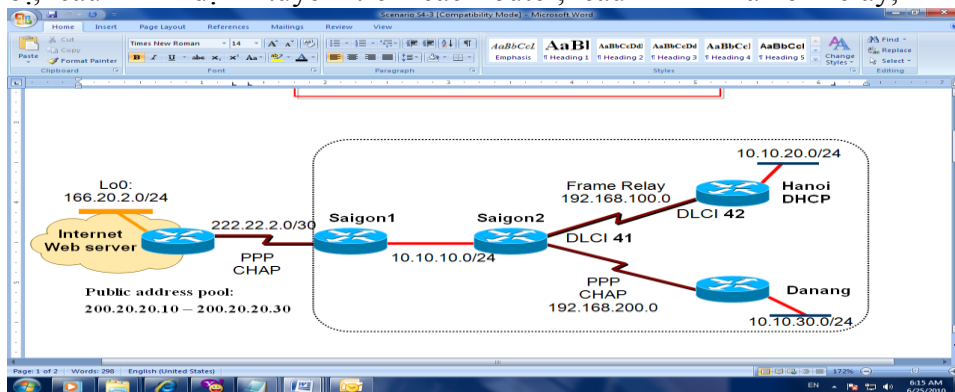
4.9 Mạng frame relay sử dụng DLCI cho mỗi PVC. DLCI có ý nghĩa gì?

- A. Nó được dùng để chọn loại đóng gói trong Frame Relay
- B. Nó dùng để xác định kết nối logic giữa local router và FR switch
- C. Nó đại diện cho địa chỉ vật lý của router.
- D. Nó đại diện cho kết nối của mỗi PVC

4.10 Các lệnh nào sau đây dùng trên cổng WAN nhưng không dùng được trên cổng LAN?

- A. IP address
- B. Encapsulation PPP
- C. No shutdown
- D. PPP authentication CHAP
- E. Speed

4.11 Trong bài tập này, yêu cầu học viên phải hoạch định và gán địa chỉ IP phù hợp các thiết bị, cấu hình định tuyến trên các router, cấu hình Frame Relay, PPP, NAT và DHCP.



Yêu cầu 1: Cấu hình cơ bản trên Router và Switch

- Đặt địa chỉ IP cho các cổng trên Router
- Cấu hình hostname cho các Router
- Cấu hình telnet trên các Router và Switch

Yêu cầu 2: Cấu hình Frame-Relay

- Sử dụng DLCI như trên mô hình
- Xác định kiểu đóng gói Frame Relay là IETF và kiểu LMI là ANSI.
- Giả sử chức năng Inverse-ARP đã bị khóa, cấu hình map tĩnh cho các địa chỉ *remote IP* và *local DLCI*.

Yêu cầu 3: Cấu hình PPP

- Cấu hình đóng gói PPP trên cổng Serial giữa các router như trong mô hình
- Cấu hình chứng thực bằng CHAP giữa 2 router sử dụng password là **spkt**.

Yêu cầu 4: Cấu hình định tuyến

- Cấu hình định tuyến cho hệ thống mạng trên sử dụng giao thức định tuyến tùy chọn.
- Trên Router **Saigon1** quảng bá **default network** cho các router bên trong để cho phép hệ thống mạng bên trong có thể truy cập ra ngoài Internet.
- **Không cấu hình định tuyến trên router ISP**, sử dụng **default route hoặc static route** để gửi các gói tin đến nó.

Yêu cầu 5: Cấu hình NAT

- Cấu hình NAT trên router Saigon1 để chuyển đổi các IP bên trong ra IP bên ngoài.

Yêu cầu 6: Cấu hình DHCP

Cấu hình DHCP trên router Hanoi để cấp IP cho LAN.

5. Lab. WAN

Lab. 9-1. PPP PAP

- | Client | 192.168.1.0/24 | Server |
|---|----------------|--------|
| <ul style="list-style-type: none"> • Mô tả S0/0/ | | |
| Router R2 sẽ chứng thực cho router R1 bằng giao thức PAP | | S0/0/ |
| | | .2 |

• Các bước cấu hình

• Cấu hình cơ bản:

```
R1(config)#int S0/0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#exit
```

```
R2(config)#int S0/0/1
R2(config-if)#ip address 192.168.1.2 255.255.255.0
R2(config-if)#exit
```

- **Cấu hình chứng thực PAP**

```
R1(config)#int S0/0/0
R1(config-if)#encapsulation ppp
R1(config-if)#ppp pap sent-username cisco password cisco
```

```
R2(config)#username cisco password cisco
R2(config)#int S0/0/1
R2(config-if)#encapsulation ppp
R2(config-if)#ppp authentication pap
```

- ✓ **Kiểm tra cấu hình**

Sử dụng các lệnh sau:

- ping
- debug ppp authentication

Lab. 9-2. PPP CHAP - Dạng 1

Client

Server

- **Mô tả** S0/0/1 **192.168.1.0/24**
Router R2 sẽ chứng thực cho Router R1 bằng giao thức CHAP. Mặc định sẽ gửi hostname đi để chứng thực.
1 .2

- **Các bước cấu hình**

- **Cấu hình cơ bản**

```
R1(config)#int S0/0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#exit
```

```
R2(config)#int S0/0/1
R2(config-if)#ip address 192.168.1.2 255.255.255.0
R2(config-if)#exit
```

- **Cấu hình chứng thực CHAP**

```
R1(config)#username R2 password cisco
R1(config)#int S0/0/0
```

```
R1(config-if)#encapsulation ppp

R2(config)#username R1 password cisco
R2(config)#interface serial 0/0/1
R2(config-if)#encapsulation ppp
R2(config-if)#ppp authentication chap
```

✓ Kiểm tra cấu hình

Sử dụng các lệnh sau:

- Router#ping
- Router#debug ppp authentication

Lab. 9-3. PPP CHAP – Dạng 2

Client

Server

- **Yêu cầu** 192.168.1.0/24
Router R2 sẽ chứng thực cho router R1 bằng giao thức CHAP. Gửi hostname và password được chỉ ra. .1 .2

• Các bước cấu hình

• Cấu hình cơ bản:

```
R1(config)#interface serial 0/0/0
R1(config-if)#ip address 192.168.1.1 255.255.255.0
R1(config-if)#exit

R2(config)#interface serial 0/0/1
R2(config-if)#ip address 192.168.1.2 255.255.255.0
R2(config-if)#exit
```

• Cấu hình chứng thực CHAP

```
R1(config)#int S0/0/0
R1(config-if)#encapsulation ppp
R1(config-if)#ppp chap hostname abc
R1(config-if)#ppp chap password cisco

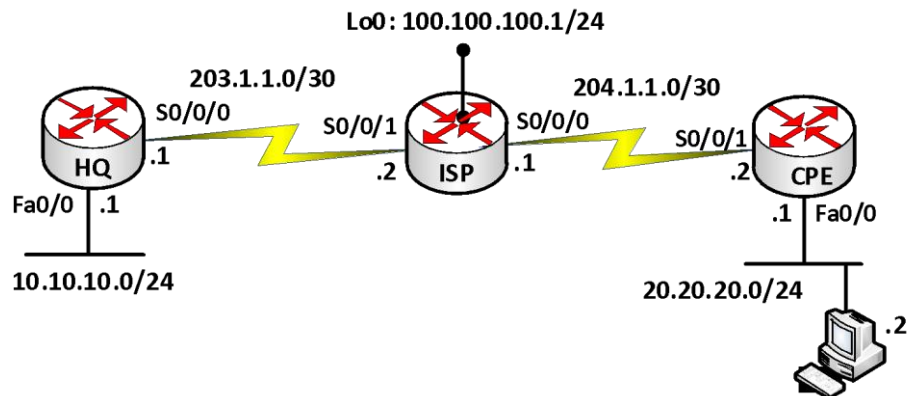
R2(config)#username abc password cisco
R2(config)#int S0/0/1
R2(config-if)#encapsulation ppp
R2(config-if)#ppp authentication chap
```

✓ Kiểm tra cấu hình

Sử dụng các lệnh sau:

- Router#ping
- Router#debug ppp authentication

Lab. 9-4. Cấu hình “remote access VPN” (IPSec)



Hướng dẫn cấu hình

```
HQ(config)#aaa new-model
```

```
HQ(config)#username cisco privilege 15 password cisco
```

```
HQ(config)#aaa authentication login default local
```

```
HQ(config)#aaa authentication login my_authen local
```

```
HQ(config)#aaa authorization network my_autho local
```

- ISAKMP phase 1:

```
HQ(config)#crypto isakmp policy 1
```

```
HQ(config-isakmp)#encryption 3des
```

```
HQ(config-isakmp)#hash sha
```

```
HQ(config-isakmp)#group 2
```

```
HQ(config-isakmp)#authentication pre-shared
```

```
HQ(config)#crypto isakmp client configuration group my_VPN
```

```
HQ(config-isakmp-group)#key cisco
```

```
HQ(config-isakmp-group)#pool VPN_pool
```

```
HQ(config)#ip local pool VPN_pool 10.10.10.2 10.10.10.50
```

- Cấu hình phase 2:

```
HQ(config)#crypto ipsec transform-set my_set esp-3des esp-sha-hmac
```

```
HQ(config)#crypto dynamic-map my_dyn 1
```

```
HQ(config-crypto-map)#set transform-set my_set
```

```
HQ(config-crypto-map)#reverse route
```

- Cấu hình phase 1.5


```

HQ(config)#crypto map my_map isakmp athorization list my_autho
HQ(config)#crypto map my_map client authentication list my_authen
HQ(config)#crypto map my_map client configuration address respond
HQ(config)#crypto map my_map 20 ipsec-isakmp dynamic my_dyn
  
```

Vào cổng kết nối đến ISP:

```

HQ(config-if)#crypto map my_map
  
```

PC:

Dùng phần mềm VPN Client để kết nối với HQ

Khai báo:

Group: my_VPN

Password: cisco (key trong my_VPN)

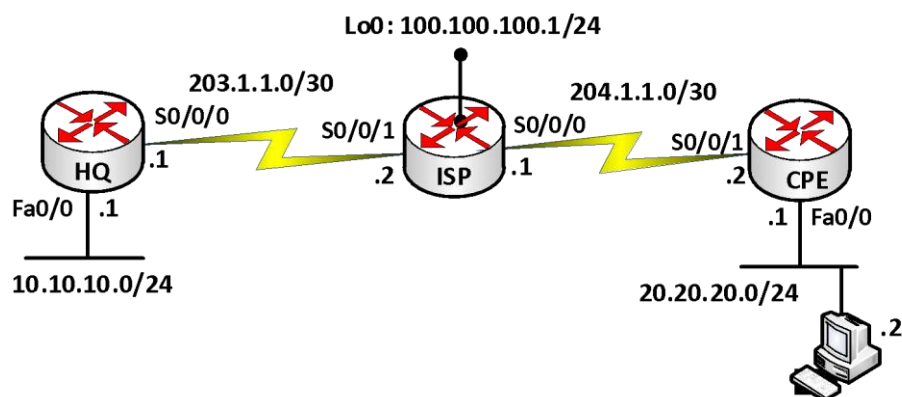
Kết nối tới VPN HQ: xuất hiện popup window để nhập username/password

Username: cisco

Password: cisco

Lab. 9-5.

Ở ví dụ 1, chúng ta dùng phần mềm trên PC để kết nối VPN. Trong bài này, chúng ta sẽ dùng router kết nối VPN về HQ.



- *Hướng dẫn cấu hình*

Cấu hình HQ:

Giống như bài trước, có thêm các câu lệnh sau:

```

HQ(config)#crypto isakmp client configuration group my_VPN
  
```

```

HQ(config-isakmp-group) #save-password
  
```

Cấu hình cho Remote:

```

Remote(config)#crypto ipsec client ezvpn my_remote
  
```

```
Remote(config-ipsec)#group my_VPN key cisco
Remote(config-ipsec)#connect manual
Remote(config-ipsec)#peer 203.1.1.1
Remote(config-ipsec)#mode client
Remote(config-ipsec)#username cisco password cisco
Remote(config-ipsec)#xauth userid mode local
```

- **Áp đặt lên cổng:**

- **Cổng nối ra Internet**

```
Remote(config-if)#crypto ipsec client ezvpn my_remote
```

- **Cổng nối vào LAN**

```
Remote(config-if)#crypto ipsec client ezvpn my_remote inside
```

- **Kiểm tra cấu hình**

- **Thực hiện lệnh sau trên Remote để connect**

```
Remote#crypto ipsec client ezvpn connect
```

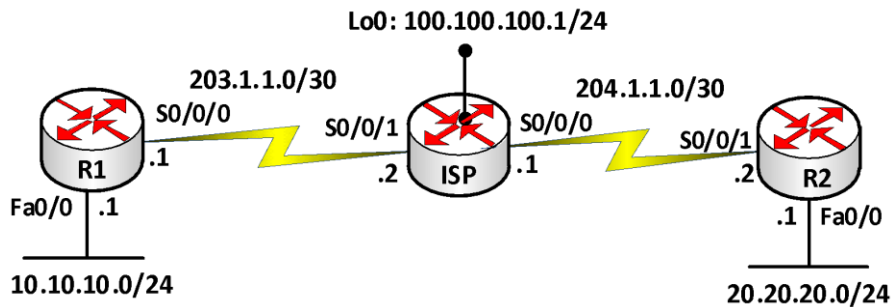
Lưu ý: Trên Remote tự động thực hiện NAT

```
Remote#show ip nat translation
```

- **Disconnect VPN bằng 2 cách:**

```
Remote#clear crypto sa
```

```
Remote#clear crypto session
```

Lab. 9-6. Cấu hình VPN site-to-site giữa R1 và R2 (IPSec).


Cấu hình cơ bản: cấu hình hostname và địa chỉ IP cho các router theo mô hình

```
R1(config)#ip route 0.0.0.0 0.0.0.0 203.1.1.2
```

```
R2(config)#ip route 0.0.0.0 0.0.0.0 204.1.1.1
```

Cấu hình VPN site-to-site (R1 và R2)
Phase 1:

```
Router(config)#crypto isakmp policy 1
```

```
Router(config-isakmp)#encryption 3des
```

```
Router(config-isakmp)#hash sha
```

```
Router(config-isakmp)#authentication pre-shared
```

```
Router(config-isakmp)#group 2
```

```
Router(config)#crypto isakmp key cisco address A.B.C.D → địa chỉ IP của Router bên site kia.
```

Phase 2:

```
Router(config)#crypto ipsec transform-set myset esp-3des esp-sha-hmac
```

```
Router(config-crypto-transform-set)#exit | mode transport //(mode tunnel default)
```

```
Router(config)#access-list 100 permit ip 10.10.10.0 0.0.0.255 20.20.20.0 0.0.0.255
```

```
Router(config)#crypto map mymap 1 ipsec-isakmp
```

```
Router(config-crypto)#match address 100
```

```
Router(config-crypto)#set transform-set myset
```

```
Router(config-crypto)#set peer A.B.C.D
```

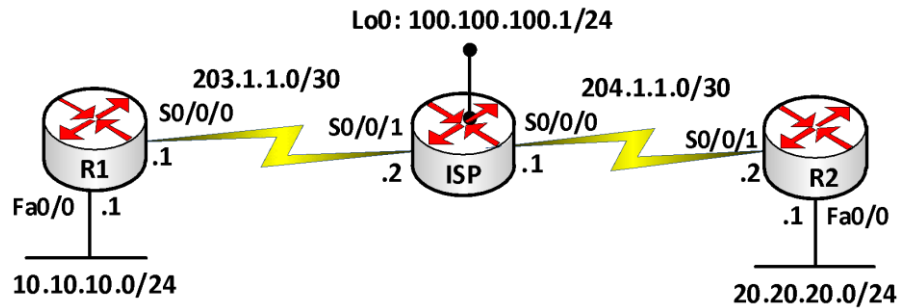
Gắn vào cổng:

```
Router(config-if)#crypto map mymap
```

Kiểm tra cấu hình:

```
Router#show crypto isakmp policy
```

```
Router#show crypto ipsec sa
```

Lab. 9-7. VPN kết hợp với NAT

Yêu cầu: Cấu hình cho 2 mạng riêng liên lạc bằng VPN. Các mạng còn lại sẽ đi bằng NAT overload.

Hướng dẫn cấu hình

```
R1(config)#access-list 101 deny ip 10.10.10.0 0.0.0.255 20.20.20.0 0.0.0.255
```

```
R1(config)#access-list 101 permit ip 10.10.10.0 0.0.0.255 any
```

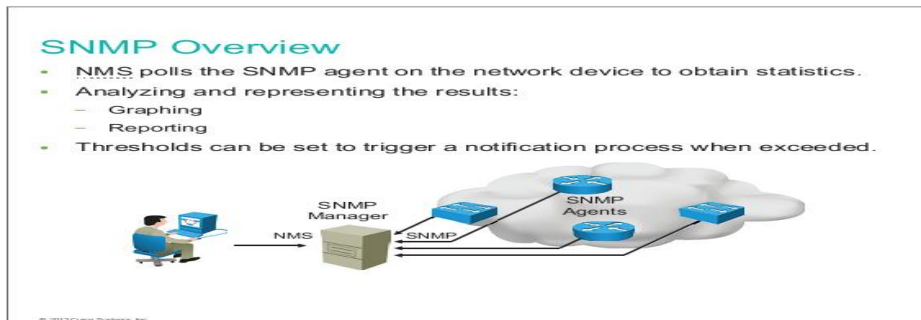
```
R2(config)#access-list 101 deny ip 20.20.20.0 0.0.0.255 10.10.10.0 0.0.0.255
```

```
R2(config)#access-list 101 permit ip 20.20.20.0 0.0.0.255 any
```

Chương 10.

GIÁM SÁT MẠNG

1. Giới thiệu



Trong những hệ thống mạng lớn, người quản trị mạng cần có công cụ để có thể theo dõi hoạt động của hệ thống. Việc theo dõi hoạt động của hệ thống bao gồm theo dõi tình trạng hoạt động của các thiết bị và dịch vụ nhằm hạn chế các rủi ro gây ra. Từ đó giúp người quản trị kịp thời khắc phục, đảm bảo hệ thống hoạt động ổn định.

Bên cạnh việc theo dõi các gói tin để phát hiện những dấu hiệu tấn công mạng đã tìm hiểu ở các phần trên thì việc giám sát những hoạt động khác của hệ thống mạng như giám sát các luồng lưu lượng mạng, hoạt động của CPU, bộ nhớ, trạng thái hoạt động các máy chủ, theo dõi trạng thái hoạt động của các dịch vụ mạng trên nó và trạng thái của các thiết bị mạng (Router, Switch,...) ... là một trong những yêu cầu đặt ra để giám sát hệ thống một cách hiệu quả hơn. Điều này giúp cho người quản trị mạng có thể nắm bắt được tình trạng hoạt động của toàn bộ hệ thống mạng một cách nhanh chóng và tiện lợi.

Một hệ thống IDS/IPS như đã tìm hiểu ở chương trước có thể phát hiện và phòng chống các cuộc tấn công xâm nhập mạng dựa vào các dấu hiệu tấn công được lưu trữ và cập nhật thường xuyên. Tuy nhiên cũng không tránh khỏi những trường hợp có những dạng tấn công mới mà những dấu hiệu chưa được biết tới, tập luật của hệ thống phát hiện chưa được cập nhật.

Với sự kết hợp hệ thống giám sát trực quan, những hoạt động của những thiết bị trong hệ thống được theo dõi và hiển thị thời gian thực các hoạt động của hệ thống một cách trực quan thông qua những đồ thị về lưu lượng mạng, trạng thái hoạt động của CPU, RAM, dịch vụ mạng, ... cho phép người quản trị có những phân tích để đưa ra các giải pháp phù hợp tránh những nguy hại cho hệ thống mạng.

Ngoài ra, người quản trị có thể thiết lập những ngưỡng cảnh báo kết hợp với hệ thống báo động để người quản trị nhanh chóng có được những thông tin về những cuộc tấn công hay phát hiện những bất thường trong hệ thống. Những bất thường ở đây như là một dịch vụ mạng ngưng hoạt động, máy chủ ngưng hoạt động, hay CPU hoạt động quá tải (đặt ngưỡng cảnh báo), ...

2. Các thành phần của hệ thống giám sát

• Giao thức

Để có thông tin cho việc giám sát trạng thái hoạt động của các thiết bị và dịch vụ mạng. Trong hệ thống giám sát sử dụng giao thức SNMP (Simple Network Management Protocol).

SNMP là một giao thức chính được sử dụng cho mục đích theo dõi tình trạng hoạt động của các thiết bị và dịch vụ trong hệ thống mạng. SNMP làm nhiệm vụ thu thập thông tin từ các thiết bị mạng (Router, Switch, Server...) cần giám sát và gửi về cho chương trình giám sát để phân tích và sử dụng để hiển thị ra giao diện quản trị các thông tin cần thiết theo mục đích của chương trình giám sát.

Trong SNMP có 3 vấn đề cần quan tâm: Manager, Agent và MIB (Management Information Base).

- MIB: là cơ sở dữ liệu dùng phục vụ cho Manager và Agent.

- Manager: nằm trên máy chủ giám sát hệ thống mạng

- Thành phần Agent: là một chương trình nằm trên các thiết bị cần giám sát, quản lý. Agent có thể là một chương trình riêng biệt (ví dụ như daemon trên Unix) hay được tích hợp vào Hệ điều hành, ví dụ như trong IOS của các thiết bị Cisco. Nhiệm vụ của các Agent là thông báo các thông tin đến cho thành phần điều khiển được cấu hình nằm trên máy chủ giám sát.

SNMP sử dụng UDP (User Datagram Protocol) như là giao thức truyền tải thông tin giữa các Manager và Agent. Việc sử dụng UDP, thay vì TCP, bởi vì UDP là phương thức truyền mà trong đó hai đầu thông tin không cần thiết lập kết nối trước khi dữ liệu được trao đổi (connectionless), thuộc tính này phù hợp trong điều kiện mạng gặp trục trặc, hư hỏng ...

• Giám sát lưu lượng

Giám sát lưu lượng được áp dụng ở các thiết bị mạng dùng là vai trò quan trọng trong việc chuyển tải các lưu lượng trên đường truyền như ở các Router, Core Switch,...

• Giám sát tình trạng hoạt động

Giám sát tình trạng hoạt động của các thiết bị là theo dõi trạng thái còn hoạt động hay đã ngưng hoạt động. Trong những hệ thống có triển khai các thiết bị dự phòng thì khó nhận ra một thiết bị đang trong tình trạng ngưng hoạt động vì trong khi đó hệ luồng lưu lượng sẽ đi qua thiết bị dự phòng.

• Giám sát dịch vụ

Giám sát dịch vụ thường được triển khai áp dụng ở các máy chủ để theo dõi tình trạng hoạt động của các dịch vụ cần giám sát. Có thể xảy ra trường hợp máy chủ vẫn đang hoạt động như dịch vụ bị tắt đi.

- *Giám sát tài nguyên của thiết bị*

Giám sát các tài nguyên như hoạt động của CPU, RAM, dung lượng đĩa cứng,... giúp theo dõi tình trạng hoạt động, khả năng đáp ứng, từ đó tiến hành các biện pháp nâng cấp, thay thế.

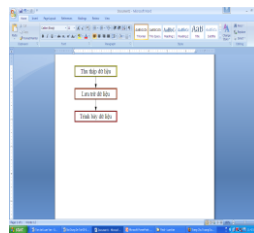
- Giám sát Syslog

3. Các công cụ nguồn mở hỗ trợ trong việc giám sát mạng

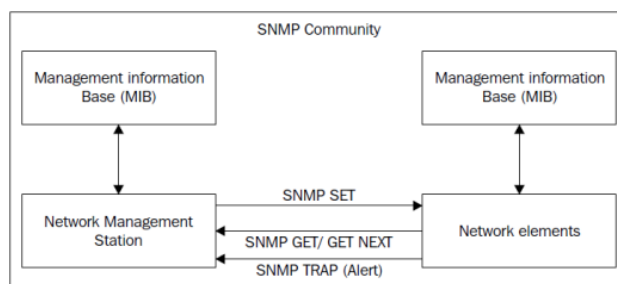
Có rất nhiều công cụ cho phép người quản trị mạng dò tìm những lỗi hỏng trong hệ thống mạng đã nêu ở phần trên... Những công cụ này rất cần thiết để kiểm tra những lỗi hỏng bảo mật trong hệ thống mạng.

3.1. Cacti

Cacti là một phần mềm nguồn mở hàng đầu về việc giám sát các lưu lượng mạng. Cacti trong hệ thống đề xuất được dùng để giám sát lưu lượng qua các Switch trung tâm, Router và các Server trong hệ thống mạng.



Cacti thể hiện lưu lượng qua các đồ thị trực quan. Điều này giúp cho người quản trị theo dõi được sự bất thường trong hệ thống. Những bất thường này có thể là những dấu hiệu của tấn công xâm nhập hoặc sự quá tải của một số thiết bị mạng trong hệ thống. Cacti sử dụng giao thức SNMP để thu thập thông tin từ các thiết bị, lưu trữ thông tin và vẽ hình trên các đồ thị.



Để tạo ra những đồ thị, Cacti cần thu thập dữ liệu từ các thiết bị giám sát thông qua giao thức SNMP. Trong hệ thống mạng lớn với nhiều thiết bị cần giám sát thì dữ liệu thu thập nên được quản lý dựa vào cơ sở dữ liệu. Trong mô hình thực nghiệm của luận văn này, dữ liệu thu thập được từ SNMP được lưu trữ vào cơ sở dữ liệu MySQL.

Cacti có khả năng giám sát lưu lượng vào/ra các cổng của thiết bị cần theo dõi; giám sát tình trạng hoạt động của CPU và bộ nhớ. Cacti còn cho phép thể hiện sơ đồ mạng trực quan để theo dõi lưu lượng trao đổi giữa các thiết bị mạng.

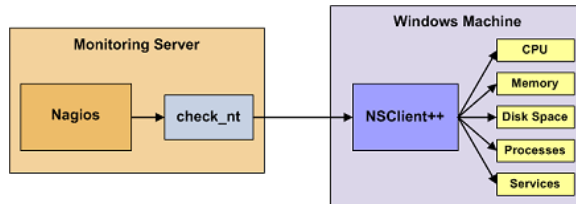
Một đặc điểm quan trọng của Cacti là cho phép tích hợp nhiều nhiều phần mềm khác vào nó. Đây là một đặc điểm quan trọng cho việc cài đặt hệ thống giám sát mạng tích hợp.

3.2. Nagios

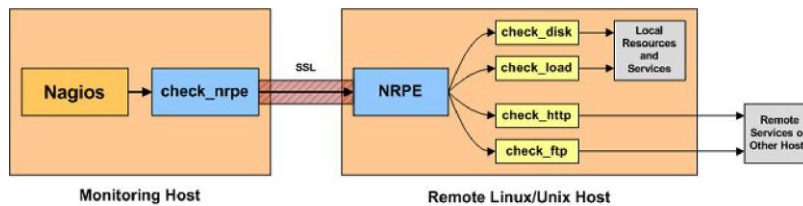
Nagios là một phần mềm mã nguồn mở hàng đầu trong việc giám sát hoạt động của các thiết bị và các dịch vụ trong mạng. Nagios hỗ trợ trong việc giám sát hoạt động một số thiết bị trung tâm trong mạng như Server, Switch trung tâm, Router, Đồng thời, nó kết hợp với bộ phận phát cảnh báo qua SMS, Audio, Web nhằm phát cảnh báo trong trường hợp một thiết bị ngưng hoạt động hoặc một dịch vụ mạng ngưng hoạt động.

Nagios giám sát các thiết bị mạng thông qua các giao thức ICMP, SNMP, ... để theo dõi trạng thái hoạt động của các thiết bị. Đồng thời Nagios còn cho phép thiết lập cơ chế giám sát hoạt động của các dịch vụ mạng trên các thiết bị. Các dịch vụ phổ biến được giám sát như: HTTP, FTP, SMTP, POP3, DHCP, SSH, IMAP...

Để giám sát các host Windows, chúng ta có thể sử dụng trực tiếp thông qua SNMP hoặc sử dụng phần mềm NSClient++ để lấy thêm nhiều thông tin hơn từ máy Windows.



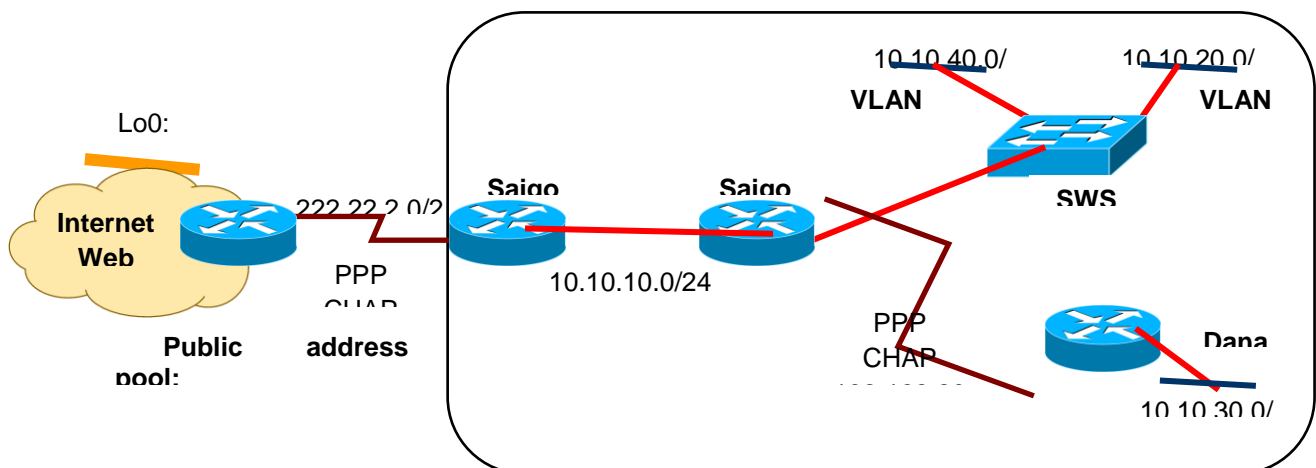
Cũng tương tự như vậy, chúng ta có thể sử dụng SNMP để giám sát các máy Linux qua việc cài đặt gói NRPE để giám sát host và các dịch vụ trên máy Linux.



Final Lab 1.

In this exam you are required to configure all routers and attached workstations so that they enable to route packets. You must decide the plan and assign IP addresses for network equipment, configure routers for IP routing, configure VLAN, configure Inter-VLANs routing, configure PPP, configure NAT.

NewStar Corp. Network Topology



Task 1: Router and Switch basic configuration

- Assign IP address to routers interface, switches and PCs
- Configure hostnames on all routers and switches
- Configure vty and privileged password

- Clock rate on DCE interfaces, and no shutdown commands
- Start www service on routers and switches

Task 2: Configure VLAN & Inter-VLANs routing

- Configure VLAN10 & VLAN20 on switch SWSG
- Configure Inter-VLANs routing so that PCs on VLANs can communication each other

Task 3: Configure PPP

- Configure PPP encapsulation on the serial link between routers as topology
- Configure CHAP authentication on both routers using the password cisco.

Task 4: Configure routing

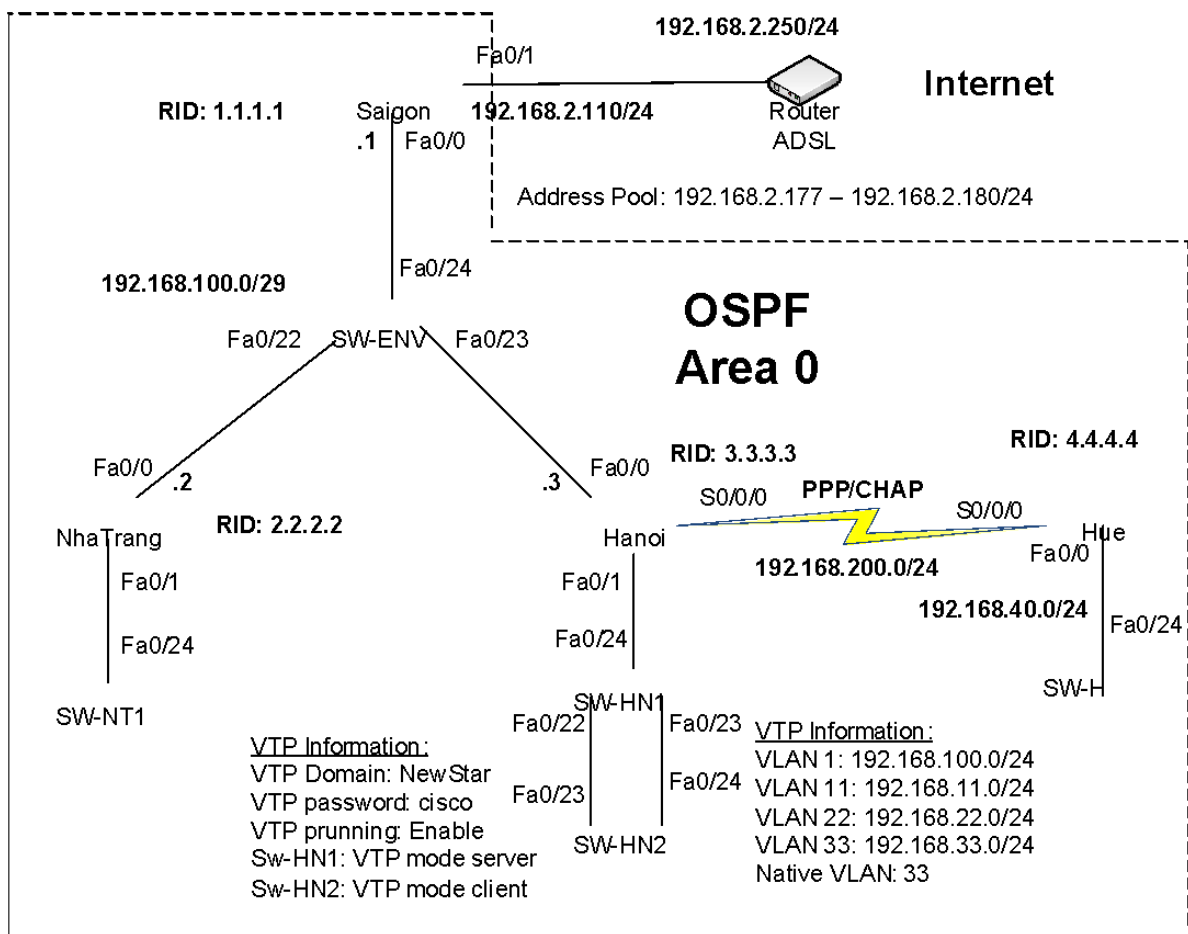
- Using any IP routing protocol for NewStar Corp. network.
- Router **Saigon1** should advertise the **default network** into NewStar Corp. network so that the Internet can be accessed from NewStar Corp. network.
- **No routing protocol** is allowed at **ISP Router**, using a **default route or static route** for routing its packets

Task 5: Configure NAT

- Configure NAT on the Saigon1 router to translate the private (inside) host IP addresses to the public (outside) network address range.
- Or use PAT, so that all addresses are using only some IP addresses of public address range

- END -

Final Lab 2.



A. STARTING – Basic Cisco Devices Configuration

1. Cisco Catalyst Switches Configuration (05 points)

- Configure the switches with host name of **Sw-NT1**, **Sw-SG**, **Sw-HN1**, **Sw-HN2** and **Sw-H** as the diagram.
- Set the management console password to **newstar**
- Set the **telnet** acces (vty 0 15) password to **cisco**
- An enable secret password of **ccna**

- Start **www service** on all switches
 - Disable domain name resolution service on all switches
 - All clear text passwords in the running-config should be encrypted
 - Configure all switches to show banner when you login to the switch (from console port or Telnet)
2. **Cisco Routers Configuration (5 points)**
- Configure the routers with host name of Saigon, Nhatrang, Hanoi, and Hue as the diagram
 - Set the management console password to **newstar**
 - Create a user account on all routers with the name is **netadmin** and the password is **master** with the privilege level of 15
 - Disable domain name resolution service on all routers
 - Configure all routers so that messages from the routers to the console screen will not be appended to the command line
 - All clear text passwords in the **running-config** should be encrypted
 - Configure all routers to show a banner when you login to the router
3. **IP addressing Assignment (05 points)**
- Look at the network diagram for IP addressing for each interface, and assign them to appropriate interfaces. Don't forget look at the netmask.
 - While configuring IP addresses on the interfaces, configure the data link layer (PPP, if appropriate) and place a description on each interface starting the router that they are connected to and which interface they are.

B. **CHALLENGING**

1. **DHCP service (05 points)**

- Configure DHCP service on Hanoi router
 - Pool name VLAN1: 192.168.1.0/24**
 - o DNS server 1: 208.67.222.222
 - o DNS server 2: 208.67.220.220
 - Pool name VLAN11: 192.168.11.0/24**
 - o DNS server 1: 208.67.222.222
 - o DNS server 2: 208.67.220.220
 - Pool name VLAN22: 192.168.22.0/24**
 - o DNS server 1: 208.67.222.222
 - o DNS server 2: 208.67.220.220
 - Pool name VLAN33: 192.168.33.0/24**
 - o DNS server 1: 208.67.222.222
 - o DNS server 2: 208.67.220.220

2. **Switching (20 points)**

- Configure **VTP** and **VLAN** database in **Hanoi** site's switches with the VTP and VLAN information given in the above diagram.
- Configure uplinks between access layer **Sw-HN2** and distribution layer **Sw-HN1** switches. Each of these channels should be 802.1q – compliant trunk links that are capable of transporting all VLAN traffic. The **VLAN 33** should be the **native VLAN**. Enable the pruning of unnecessary traffic from nonresident VLANs.
- On the Catalyst Switch **Sw-HN1**, configure Port 02, 03, 04 into **VLAN11**, Port 12, 13, 14 into **VLAN22** and Port 17, 18, 19 into **VLAN33**.
- On the Catalyst Switch **Sw-HN2**, configure Port 01 09 into **VLAN11**, Port 13, 15, 17, and 19 into **VLAN22** and Port 20, 21 into **VLAN33**.

- Enable spanning tree mode PVRST on all switches. Ensure the **Sw-HN1** should be the **Root Bridge** of all VLAN.

3. Routing (30 points)

- **OSPF**
 - o Configure OSPF routing protocol on each router. Don't forget to configure appropriate **RouterID** on each router. Ensure that users can reach any site of network and the Internet.
 - o Enable **OSPF clear text authentication** feature on the Serial interfaces of **Hanoi** and **Hue** routers with the key of **newstar**.

4. WAN (10 points)

- **PPP**
 - o Configure **PPP** encapsulation on the WAN link between **Hanoi** and **Hue**
 - o Configure **CHAP** authentication with the password is **newstar**

5. Others (20 points)

- Standard Access List: Create a standard out going access list, and apply it on Nhatrang's FastEthernet 0/1 to fulfill the following requirement:
 - o Deny access to users on VLAN11 and VLAN22 to **Nhatrang's** LAN
- **Extended Access list**: Create 02 extended incoming access lists, and apply each of them on appropriate interfaces on Hanoi router to fulfill the following requirements:
 - o Users in VLAN1 cannot **ping Hue's** LAN but can **telnet** to **Sw-H**
 - o Deny **http (www)** request from users from **VLAN33**
 - o Permit anything else
- **NAT**: configure NAT on Saigon to have the following:
 - o Every host resides in every VLAN and LAN can go through the Internet

C. ENDING

- Test the connectivity between any interfaces of any devices in the diagram
- Every host resides in every VLAN and LAN can through the Internet.