Công ty An ninh mạng Viettel

Báo cáo về Switching

Sinh viên: Nguyễn Đan Trường

*MỤC LỤC*

[1. Encapsulation IP packet 3](#_Toc154497904)

[2. Router Funtion 4](#_Toc154497905)

[3. Path Determination 5](#_Toc154497906)

[4. Định tuyến và định tuyến tính cho router 7](#_Toc154497907)

[5. Linkstate Routing Protocol 7](#_Toc154497908)

[6. Distance vector routing protocol 20](#_Toc154497909)

[7. Địa chỉ Ipv4 22](#_Toc154497910)

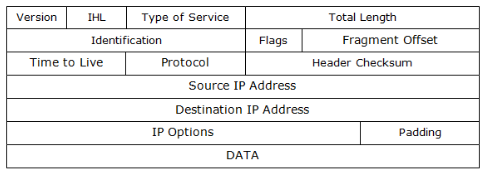
[8. NAT – Network Address Translation 24](#_Toc154497911)

[9. ACL – Access Control List 25](#_Toc154497912)

## Encapsulation IP packet

Encapsulation IP packet là quá trình thêm các thông tin bổ sung vào một gói tin IP trước khi truyền đi. Các thông tin bổ sung này thường bao gồm địa chỉ IP nguồn và đích, các thông số quản lý mạng, và dữ liệu từ các tầng trên.

IP header Version 4 gồm 12 trường bắt buộc với tổng chiều dài 20 byte (không tính các trường options và data). Cấu trúc của IP header Version 4 như sau:



* Version (4 bit): Chỉ ra phiên bản IP đang được dùng là IPv4 (0100). Nếu trường này khác với phiên bản IP của thiết bị nhận, thiết bị nhận sẽ từ chối và loại bỏ các gói tin này. Bằng cách nhìn vào số phiên bản, Router có thể xác định phần còn lại của IP Datagram.
* IP Header Length (IHL) (4 bit): Chỉ ra chiều dài của header, mỗi đơn vị là 1 word, mỗi word = 32 bit = 4 byte. Ở đây trường IP Header Length có 4 bit nên có 2^4 = 16 word = 16 x 4byte = 64 byte nên chiều dài header tối đa là 64 byte. Bình thường Hearder dài 20 byte. Đây là chiều dài của tất cả các thông tin Header.
* Type Of Services (TOS) (8 bit): Chỉ ra cách thức xử lý gói dữ liệu, có độ ưu tiên hay không, độ trễ cho phép của gói dữ liệu. Trường này thường được dùng để thực hiện quản lý chất lượng dịch vụ mạng.
* Total Length (16 bit): Chỉ ra chiều dài của toàn bộ IP Datagram tính theo byte, bao gồm data và phần header. Do có 16 bit nên tối đa là 2^16 = 65536 byte = 64 Kb nên chiều dài tối đa của 1 IP Datagram là 64 Kb.
* Identification (16 bit): Chỉ mã số của 1 IP Datagram , giúp bên nhận có thể ghép các mảnh của 1 IP Datagram lại với nhau vì IP Datagram phân thành các mảnh và các mảnh thuộc cùng 1 IP Datagram sẽ có cùng Identification.
* Flag (3 bit): Bit 0: không dùng, Bit 1: cho biết gói có phân mảnh hay không, Bit 2: nếu gói IP Datagram bị phân mảnh thì mảnh này cho biết mảnh này có phải là mảnh cuối không. Bao gồm 6 cờ: URG – cờ cho trường Urgent pointer, ACK – cờ cho trường Acknowledgement, PSH – hàm Push, RST – thiết lập lại đường truyền, SYN – đồng bộ lại số thứ tự, FIN – không gửi thêm dữ liệu.
* Fragment Offset (13 bit): Báo bên nhận vị trí offset của các mảnh so với gói IP datagram gốc để có thể ghép lại thành IP datagram gốc.
* Time To Live (TTL) (8 bit): Chỉ ra số bước nhảy (hop) mà một gói có thể đi qua. Con số này sẽ giảm đi 1, khi gói tin đi qua 1 router. Khi router nào nhận gói tin thấy TTL đạt tới 0 gói này sẽ bị loại. Đây là giải pháp nhằm ngăn chặn tình trạng lặp vòng vô hạn của gói tin trên mạng.
* Protocol (8 bit): Chỉ ra giao thức nào của tầng trên (tầng Transport) sẽ nhận phần data sau khi công đoạn xử lí IP diagram ở tầng Network hoàn tất hoặc chỉ ra giao thức nào của tầng trên gởi segment xuống cho tầng Network đóng gói thành IP Diagram, mỗi giao thức có 1 mã (06: TCP, 17: UDP, 01: ICMP…).
* Header CheckSum (16 bit): Hỗ trợ cho Router phát hiện lỗi bit trong khi nhận IP datagram. Giúp bảo đảm sự toàn vẹn của IP Header.
* Source IP Address (32 bit): Chỉ ra địa chỉ của thiết bị truyền IP diagram (Xem cấu trúc của địa chỉ IPv4).
* Destination IP Address (32 bit): Chỉ ra địa chỉ IP của thiết bị sẽ nhận IP diagram (Xem cấu trúc của địa chỉ IPv4).
* IP Option: kích thước không cố định, chứa các thông tin tùy chọn như: Time stamp – thời điểm đã đi qua Router, Security – cho phép Router nhận gói dữ liệu không, nếu không thì gói sẽ bị hủy, Record Router – lưu danh sách địa chỉ IP của Router mà gói phải đi qua, Source Route – bắt buộc đi qua Router nào đó. Lúc này sẽ không cần dùng bảng định tuyến ở mỗi Router nữa.
* Padding: Các số 0 được bổ sung vào trường này để đảm bảo IP Header luôn là bội số của 32 bit.

## Router Funtion

Router là thiết bị mạng có chức năng định tuyến và chuyển tiếp gói tin giữa các mạng máy tính. Router có thể được kết nối với nhiều mạng khác nhau, chẳng hạn như mạng LAN, …

Chức năng chính của Router là định tuyến (routing). Định tuyến là quá trình tìm đường đi ngắn nhất để chuyển tiếp một gói dữ liệu từ nguồn đến đích. Router sử dụng bảng định tuyến (routing table) để lưu trữ thông tin về các mạng và đường đi giữa chúng.

Ngoài chức năng định tuyến, router còn có thể thực hiện một số chức năng sau:

* Chuyển tiếp gói dữ liệu: Router có thể chuyển tiếp gói dữ liệu giữa các mạng khác nhau.
* Bảo mật: Router có thể được sử dụng để bảo vệ mạng khỏi các mối đe dọa bảo mật, chẳng hạn như tấn công từ chối dịch vụ (DoS).
* Quản lý: Router có thể được sử dụng để quản lý mạng, chẳng hạn như giám sát lưu lượng mạng và phân tích lỗi.

## Path Determination

Path determination là quá trình xác định đường đi tốt nhất để chuyển tiếp một gói dữ liệu từ nguồn đến đích. Router là thiết bị mạng thực hiện quá trình này.

Việc tìm kiếm đường đi tốt nhất sẽ phụ thuộc vào giao thức định tuyến sử dụng:

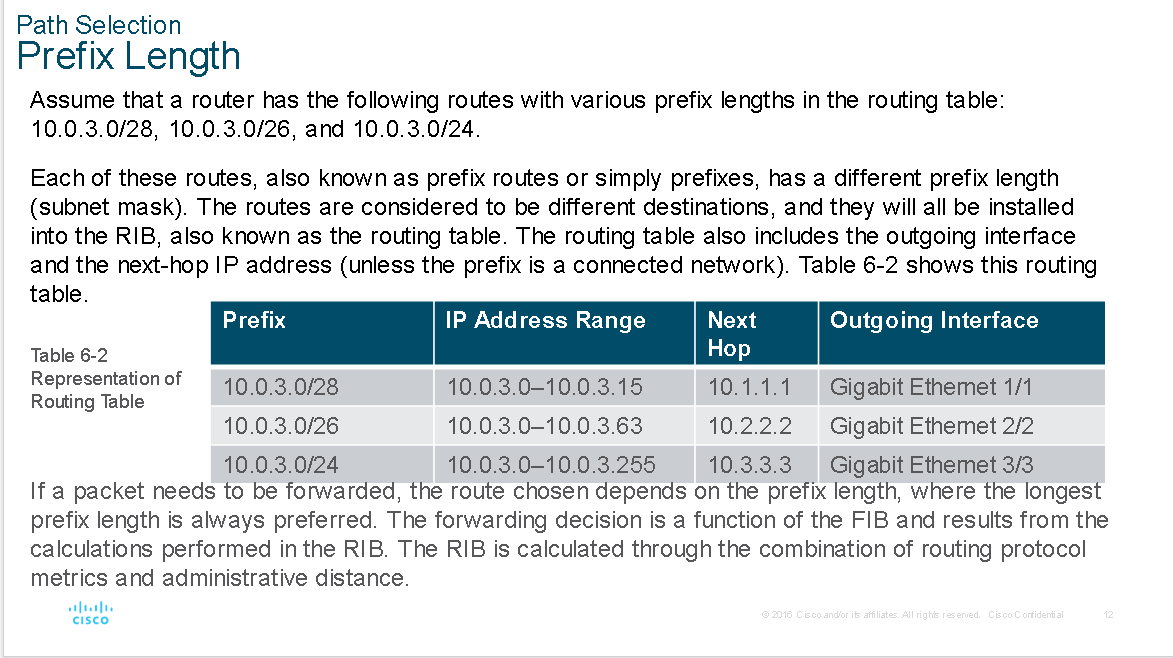
* Distance vector: Căn cứ vào số lượng router đi qua nhiều hay ít
* Link state: OSPF, thu thập thông tin trên toàn mạng sau đó tự xây dựng bản đồ và tìm được đường đi ngắn nhất.
* Path vector: BGP, căn cứ vào nhiều thuộc tính khác nhau để chọn đường đi

**Một số khái niệm:**

* Metric: chi phí cho một gói tin đi từ nguồn tới đích. Với một giao thức định tuyến, router sẽ chọn ra đường đi có metric thấp nhất là đường đi tốt nhất.
* Hop: khi một packet đi qua 1 router tức là đi qua 1 hop
* Next-hop: router tiếp theo mà packet đi tới (interface đối diện)

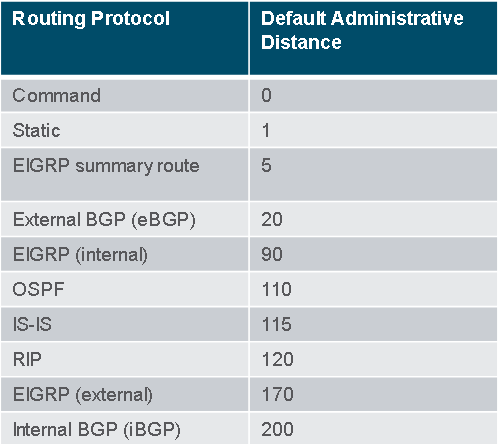
**Độ ưu tin chọn route**

* Prefix length:



Longest match: khớp với phần mạng nhiều nhất thì đường đi đó sẽ được chọn.

* Chọn 10.0.3.0/28 nếu gói tin có IP dest từ 10.0.3.0-10.0.3.15
* Adminitrative Distance (AD): là giá trị thể hiện độ ưu tiên của giao thức routing. Giá trị càng thấp thì độ ưu tiên càng cao



* Metric: thể hiện giá của một gói tin đi từ nguồn tới đích. Ưu tiên đường đi có metric nhỏ hơn. Giá trị metric này sẽ được tính toán khác nhau với từng giao thức định tuyến.

## Định tuyến và định tuyến tính cho router

Định tuyến là quá trình tìm ra đường đi tốt nhất để chuyển tiếp một gói dữ liệu từ nguồn đến đích. Router là thiết bị mạng thực hiện quá trình này.

Định tuyến tĩnh là một phương pháp định tuyến trong đó các tuyến đường được cấu hình thủ công bởi quản trị viên mạng. Trong định tuyến tĩnh, các tuyến đường không được học hỏi từ các router khác mà được cấu hình trực tiếp vào bảng định tuyến của router.

**Ưu điểm của định tuyến tĩnh:**

* Đơn giản và dễ quản lý: Các tuyến đường được cấu hình thủ công nên việc quản lý và cập nhật bảng định tuyến trở nên đơn giản.
* Có thể kiểm soát được đường đi: Quản trị viên mạng có thể kiểm soát được đường đi của các gói dữ liệu bằng cách cấu hình các tuyến đường tĩnh. Điều này có thể giúp cải thiện hiệu suất hoặc bảo mật của mạng.

**Nhược điểm của định tuyến tĩnh:**

* Không linh hoạt: Các tuyến đường tĩnh không thể được cập nhật tự động khi cấu trúc mạng thay đổi.
* Không hiệu quả: Định tuyến tĩnh có thể không hiệu quả trong các mạng lớn hoặc phức tạp.

**Cấu hình định tuyến tĩnh:**

*ip route ip\_dest subnetmask next-hop [AD]*

*ip route 0.0.0.0 0.0.0.0 next-hop (add default route)*

## Linkstate Routing Protocol

Linkstate Routing Protocol là một loại giao thức định tuyến động trong đó các router trao đổi thông tin về trạng thái của các đường link mà chúng kết nối. Thông tin này được sử dụng để xây dựng một bản đồ mạng toàn bộ, từ đó các router có thể xác định đường đi tốt nhất để chuyển tiếp các gói dữ liệu.

Các giao thức Linkstate Routing Protocol hoạt động theo các bước sau:

Tạo bản tin trạng thái đường link (LSA): Mỗi router sẽ tạo một bản tin LSA cho mỗi đường link mà nó kết nối. Bản tin LSA chứa các thông tin sau:

* + Địa chỉ IP của router
  + Địa chỉ IP của đường link
  + Trạng thái của đường link
  + Chi phí của đường link

Trao đổi LSA

* + Các router sẽ trao đổi LSA với nhau thông qua các giao thức định tuyến.
  + Các router sẽ lưu trữ các LSA mà chúng nhận được trong một cơ sở dữ liệu.

Xây dựng bản đồ mạng

* + Mỗi router sẽ sử dụng các LSA mà nó lưu trữ để xây dựng một bản đồ mạng toàn bộ.
  + Bản đồ mạng này chứa thông tin về tất cả các mạng và đường link trong hệ thống mạng.

Xác định đường đi

* + Mỗi router sẽ sử dụng bản đồ mạng để xác định đường đi tốt nhất để chuyển tiếp các gói dữ liệu

Ưu điểm của Linkstate Routing Protocol:

* Tính hội tụ nhanh: Các giao thức Linkstate Routing Protocol có tính hội tụ nhanh, nghĩa là chúng có thể nhanh chóng cập nhật bảng định tuyến khi có sự thay đổi trong mạng.
* Tính chính xác: Các giao thức Linkstate Routing Protocol có tính chính xác cao, nghĩa là chúng có thể xác định đường đi tốt nhất để chuyển tiếp các gói dữ liệu.

Nhược điểm của Linkstate Routing Protocol:

* Tốn tài nguyên: Các giao thức Linkstate Routing Protocol tiêu tốn nhiều tài nguyên hơn các giao thức Distance-Vector Routing Protocol.
* Phức tạp: Các giao thức Linkstate Routing Protocol phức tạp hơn các giao thức Distance-Vector Routing Protocol.

**Giao thức OSPF**

OSPF (Open Shortest Path First): OSPF là một giao thức Linkstate Routing Protocol phổ biến nhất. OSPF được sử dụng trong các mạng lớn và phức tạp.

OSPF là một giao thức định tuyến nội miền. Giao thức sẽ hoạt động theo 3 bước sau:

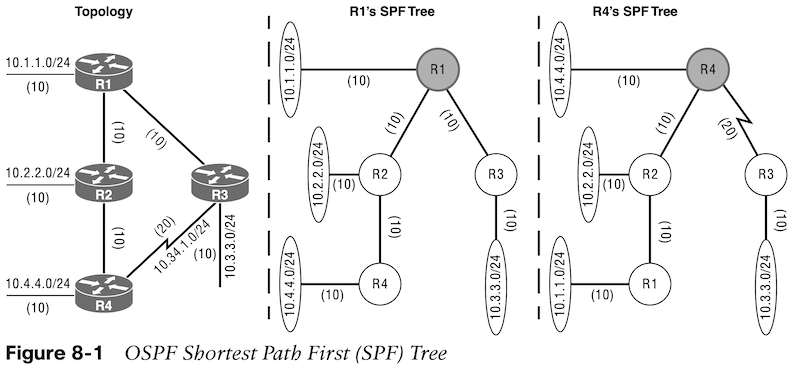
* Tạo dựng mối quan hệ hàng xóm (neighbor)
* Tìm đường đi tới các mạng
* Xây dựng bản đồ toàn mạng cùng chạy OSPF, từ đó tính toán đường đi ngắn nhất tới từng mạng.

**LSA, LSDB, SPT**

* LSA – Link state advertisement: bản tin gửi thông tin trạng thái đường link
* LSDB – Link state database: sau khi nhận được các bản tin LSA router sẽ lưu thông vào database
* SPT: sau khi chạy thuật toán shortest path first thì router xây dựng được một tree gọi là SPT

Bản chất thì tree của từng router sẽ giống nhau vì chúng cùng chung một bản đồ. Cái khác nhau giữa tree của các router là điểm gốc, cành và ngọn sẽ khác nhau.

Vd: đừng từ 2 địa điểm khác nhau trên Hà Nội thì ta vẫn chỉ có 1 bản đồ cho cả Thành Phố, nhưng để đi tới Lăng Bác thì từ 2 điểm ta sẽ tính toán và đưa ra được 2 tree khác nhau dựa trên những thông tin đã biết.



Ta thấy được với mỗi router làm gốc sẽ xây dựng lên 1 tree, đường đến 1 mạng đích sẽ chỉ có 1 đường duy nhất mặc dù trên topo có nhiều hơn 1 đường

VD: R1 đến mạng 10.4.4.0 sẽ phải đẩy gói tin qua R2

R2 đến mẹng 10.4.4.0 không cần phải đây qua router nào khác nữa

=> Tree của từng cây có thể khác nhau, nhưng chúng cùng thống nhất với nhau đường đi tới 1 mạng đích chỉ có 1 đường đi cho tất cả các tree

=> R1 chỉ có nhiệm vụ đẩy gói tin ra exit-interface đã tính toán

R2 nhận được cũng đẩy ra exit-interface đúng như ý của R1 muốn

Các router sẽ không quan tâm tới việc gói tin có đi được đến đích hay không. Vì vậy, đường đi từ 1 mạng nguồn tới mạng đích trong miền OSPF là duy nhất.

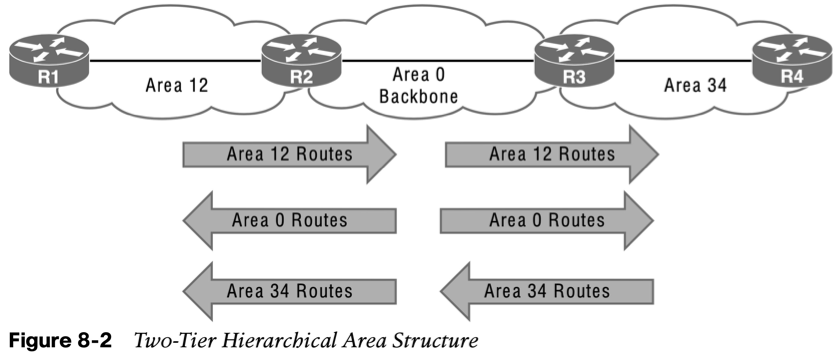
**Database-tables**:

* Adjacency database: bảng lưu trữ thông tin hàng xóm
* Topology database: Lưu thông tin link state của tất cả các router trong mạng
* Forwarding database:(Routing table) Lưu trữ tuyến đường tốt nhất cho từng đường mạng, được tính toán dựa trên topo database

**Area-Id**

Nếu vùng mạng OSPF có quá nhiều router cùng chạy thì việc xây dựng database cho vùng sẽ rất mất thời gian và bảng định tuyến của Router cũng sẽ rất dài. Nếu có thay đổi gì trong mạng thì toàn bộ router đều sẽ được cật nhật.

* Ta sẽ chia thành từng vùng mạng chạy OSPF và phân biệt với nhau bởi **area-id**



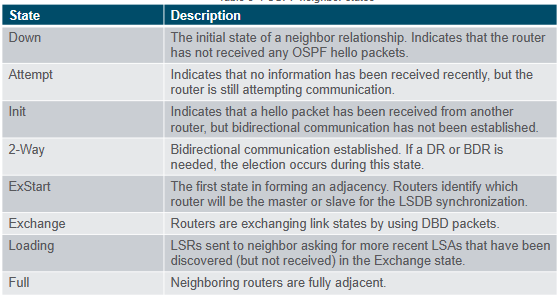
Có 2 loại area:

* **Backbone area: (area 0)** tất cả các kết nối của các area #0 đều phải thông qua area 0
* **Area khác 0**

**Các bản tin trong OSPF**

|  |  |  |
| --- | --- | --- |
| **Type** | **Packet Name** | **Functional Overview** |
| 1 | Hello | Discover and maintain neighbors. Packets are sent periodically on all OSPF interfaces to discover neighbors and ensure that other adjacent neighbors are still online. |
| 2 | Database description (DBD) or (DDP) | Summarize database contents. Packets are exchanged when an OSPF adjacency is formed. They describe the LSDB contents. |
| 3 | Link-state request (LSR) | Download databases. If a router thinks that part of its LSDB is stale, it requests part of a neighbor’s DB using this packet type. |
| 4 | Link-state update (LSU) | Update databases. This is an explicit LSA for a specific network link and normally is sent in direct response to an LSR. |
| 5 | Link-state ack | Flood acknowledgments. These are sent in response to LSA flooding, making flooding a reliable transport feature. |

**Các trạng thái của router**

****

Full: 2 router đã thiết lập neighbor thành công

Init: Bắt đầu gửi ra bản tin hello, nhưng chưa nhận được hello từ router khác

2- way: đã gửi và nhận được hello

Exstart: chuẩn bị gửi bản tin database

Exchange: đã trao đổi bản tin Database

Loading: gửi bản tin LSR và LSU để đồng bộ Database

**Cấu hình:**

**Cách 1:**

*#router ospf 1*

*#network ip-add wildcard-mask area \_\_\_*

**Cách 2:**

*#interface e0/0*

*#ip ospf 1 area \_\_\_*

**Ý nghĩa wildcard-mask:** Xác định ip liên tục hoặc không liên tục

**VD:**

Với 0.0.0.1 => ta thu được các địa chỉ chẵn

Ở vị trí bit 0 – bit ở IP tham chiếu giữ nguyên

Ở vị trí bit 1 – bit ở IP tham chiếu sẽ chạy kiểm tra 0-1

Khi 1 IP tham chiếu đi kèm 1 Sub mask hặc wildcard Mask thì nó sẽ tính ra 1 dải địa chỉ IP đi kèm

Subnet mask: 192.168.1.10 255.255.255.0

192.168.1.1 - 192.168.1.254 (256 kết quả trong cùng 1 lớp mạng)

wildcard Mask:

|  |  |  |  |
| --- | --- | --- | --- |
| **192= 11000000** | **168** | **1** | **0= 00000000** |
| **0** | **0** | **0** | **255=11111111** |
| **192** | **168** | **1** | **00000001=1**  **00000010=2**  **00000011=3**  **….**  **11111111=255** |

Vd: 192.168.1.1 0.0.0.254

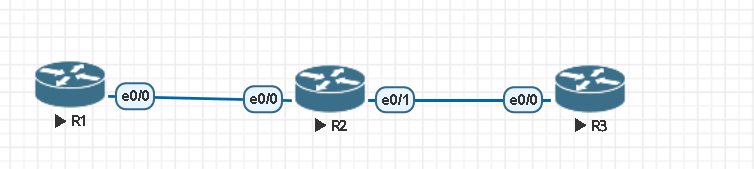
11000000.10101000.00000010. 0000000 1

00000000.00000000.00000000. 1111111 0

Phần đỏ do các bit wildcard =1 nên bit của IP sẽ chạy 0-1 để có thể list ra 1 dải IP

**#network ip-add wildcard-mask area \_\_**

Mục đích của câu lệnh là enable ospf trên interface. Nên hoàn toàn ta có thể dùng cách 2 enable trực tiếp trên cổng



VD: #network 192.168.12.0 0.0.0.255 area 0

#network 192.168.12.1 0.0.0.0 area 0 (cách khác)





Ta thấy được mặc dù router 2 quảng bá mạng 2.2.2.0/24 đi nhưng trên R2 chỉ có cổng int l0: 2.2.2.2/32 nên nó sẽ quảng bá đi địa chỉ int l0

* với OSPF, router sẽ quảng bá **IP và Subnet** của interface match với pool được khai báo



Bản chất câu lệnh này chỉ là việc ta enable int 23.2 vào mạng OSPF



Nhưng vì mạng 192.168.23.0/24 đã có trong bảng định tuyến của R2 trước đó (mạng chứa cổng e0/1) thông qua cổng e0/1 (là cổng enable OSPF), nên R2 sẽ quảng bá cả mạng vào miền

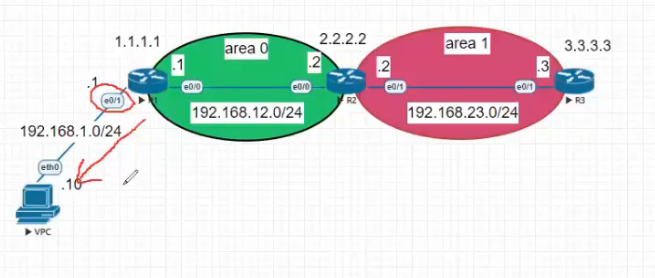
**Router-id:** dùng để định danh cho từng router sử dụng trong mạng OSPF

* Tự động chọn router-id: chọn IP cao nhất của loopback, nếu không có loopback thì sẽ chọn IP cao nhất trên các cổng
* Cấu hình tĩnh:  
  #router ospf 1

#router-id 2.2.2.2

# clear ospf process

**Passive-interface:**



Nếu router 2 muốn biết được mạng 192.168.1.0 thì r1 phải quảng bá nó vào miền ospf. Tuy nhiên nếu enable cổng e0/1 của r1 vào ospf thì trên cổng này sẽ gửi ra các bản tin LSA. Rõ ràng thì VPC không cần nhận những bản tin này

=> Passive-interface sẽ giải quyết việc này. Áp dụng cho các interface looback và mạng stub.

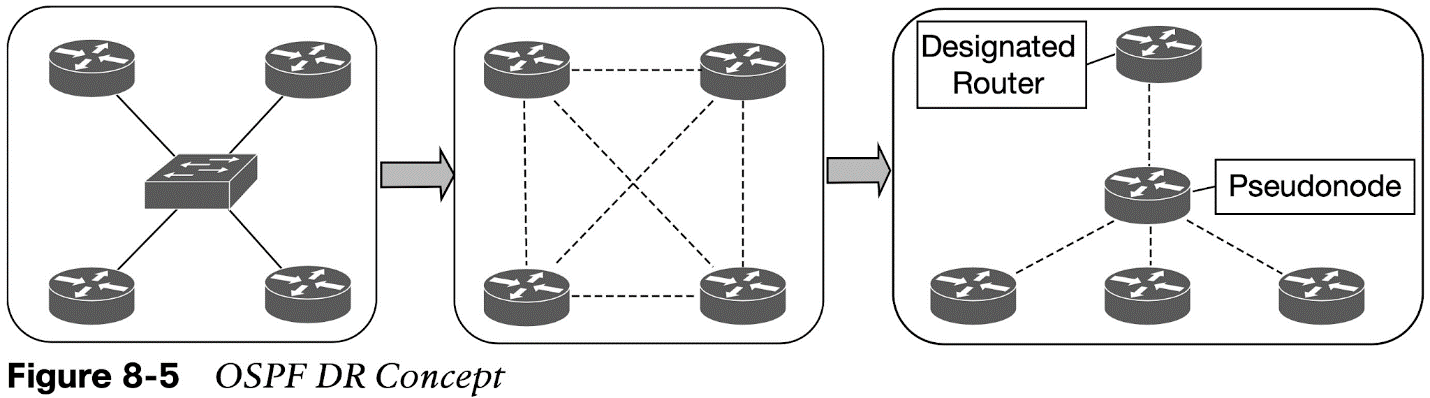
#router ospf 1

#passive-interface e0/1

**Điều kiện thiết lập Neighbor:**

* Router-id phải là duy nhất
* Interface quảng bá 2 đầu phải chung subnet mask
* MTU trên 2 đầu phải match
* area ID match nhay
* Hello time, dead time giống nhau ….

**Bình bầu DR và BDR:**



Như hình ta sẽ có n(n-1)/2 mỗi quan hệ được thiết lập

Sẽ cần phải bình bầu ra DR và BDR để giảm bớt các mối quan hệ

Tối ưu việc cật nhật thông tin link sate

* Khi 1 router có sự thay đổi thì bản tin sẽ được gửi lên cho DR chứ không gửi cho tất cả các router
* Nếu DR chết thì BDR sẽ lên làm DR

Nguyên tắc bình bầu:

* Priority lớn nhất là DR
* Router-ID cao nhất

*#interface e0/0*

*#ip ospf priority \_\_\_\_ (0-255)*

*#clear ip ospf process*

Router sẽ căn cứ vào bản tin hello để biết rằng DR còn sống hay chết

Trường thông tin **hello interval** và **dead interval** sẽ cho thấy điều này

DR và BDR sẽ được gom vào 1 nhóm, các DROTHER khi gửi bản tin tới nhóm này sẽ có địa chỉ IP đích (multicast) là 224.0.0.6

Địa chỉ của các DROTHER sẽ là 224.0.0.5

Khi Router mới join vào mạng thì trường thông tin DR/BDR =0.0.0.0 để lắng nghe trong mạng có DR/BDR chưa (trong bản tin hello). Nếu không router sẽ tự tuyên bố là DR

**Lưu ý:**

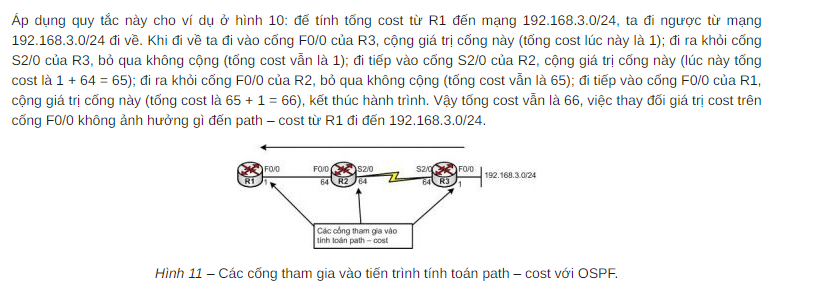
* Trong 1 mạng multiaccess thì router nào được cấu hình OSPF và khởi động lên đầu tiên với đủ thời gian thì sẽ được trở thành DR
* Nếu trong mạng mới có 2 DR cùng join vào mạng thì quá trình bình bầu diễn ra

2 router là DROTHER thì khi nhìn thấy nhau sẽ ở trạng thái 2WAY, không cần lên FULL vì DROTHER sẽ không phải cật nhật bản tin linkstate

**Link- Cost:**

Cost = 10^8 / BW(bps)

**Nguyên tắc: tính ngược từ đích về nguồn, đi vào thì cộng, đi ra thì không cộng**



Điểu chỉnh cost

#router os 1

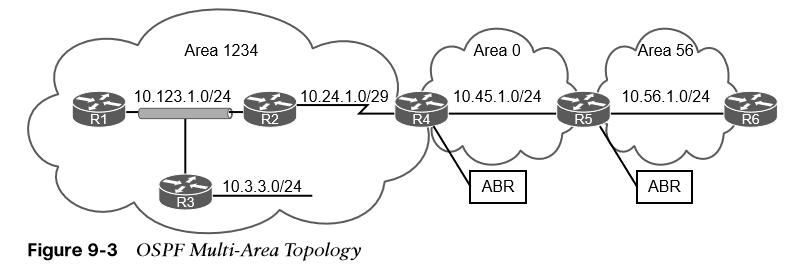
**#auto-cost reference-bandwidth** *bandwidth-in-mbps (tất cả interface)*

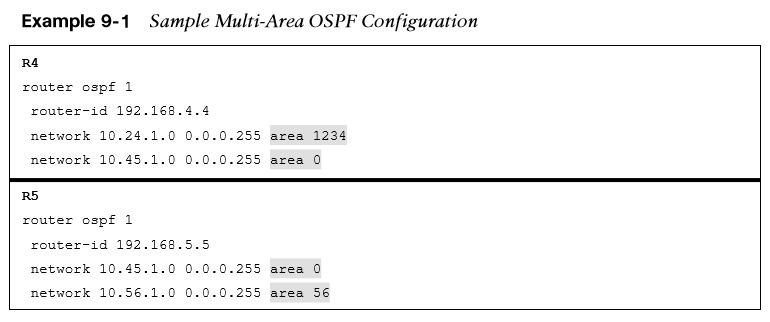
*Hoặc*

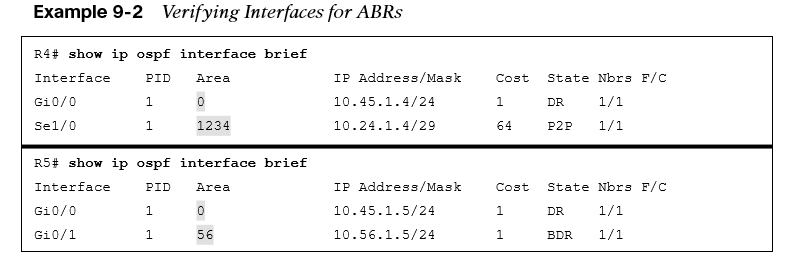
*#int e0/0*

*# ip ospf cost 1-65535*

**Multi area**







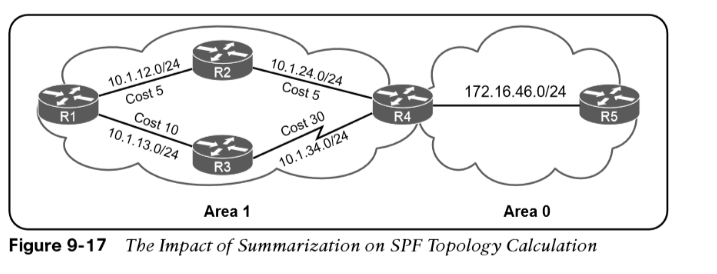
ABR là router kết nối giữa Area 0 và các area khác 0. ABR có trách nhiệm quảng bá các route giữa các AREA

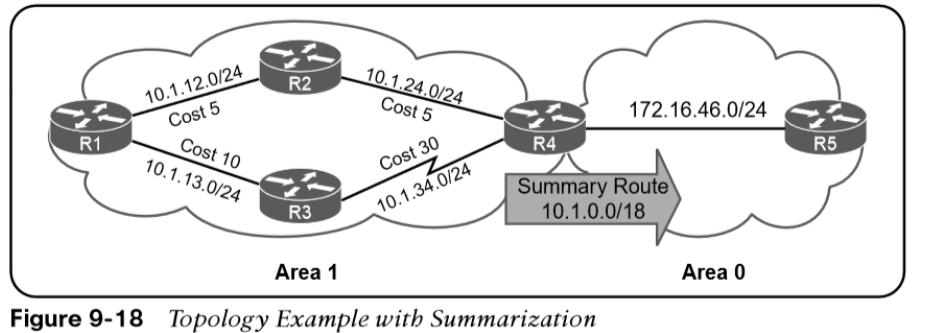
ASBR là router kết nối giữa các miền định tuyến 2 giao thức khác nhau

Intra-area: là định tuyến cùng 1 area (O)

Inter-area: định tuyến khác area (O IA)

**Summary route:**





*R4(config)#router os 1*

*#area 1 range 10.1.0.0 255.255.192.0 (metric)*

Vì R123 không cùng area với R5 nên khi học được các mạng của area 1 thì R5 thực chất chỉ biết rằng muốn đến các mạng này thì đẩy tới R4 chứ không biết được là đường đi ở area 1 như thế nào

=> vì vậy ta có thể gom các mạng area 1 thành 1 mạng cha rồi quảng bá nó vào area 0

**Các loại LSA**

|  |  |  |
| --- | --- | --- |
| **Kiểu LSA** | **Tên chung** | **Mô tả** |
| 1 | Router | Mỗi router có một LSA loại này, liệt kê tất cả các RID và địa chỉ IP của tất cả các cổng. Đại diện cho mạng stub. |
| 2 | Mạng | Mỗi mạng trung gian sẽ có một LSA. Được tạo ra bởi DR và tượng trưng cho toàn bộ mạng và các cổng kết nối vào mạng. |
| 3 | Tóm tắt mạng | Được tạo ra bởi ABR để tượng trưng cho các LSA kiểu 1 và kiểu 2 khi được quảng bá sang vùng khác. Định nghĩa các kết nối, các liên kết (subnets) trong vùng ban đầu và chi phí để đi về kết nối đó nhưng không có thông tin về sơ đồ mạng. |
| 4 | Tóm tắt ASBR | Cũng giống như LSA loại 3 ngoại trừ kiểu LSA này quảng bá các tuyến dạng host được dùng để đến một ASBR. |
| 5 | AS bên ngoài | Được tạo ra bởi ASBR cho những tuyến ngoại được đưa vào trong OSPF. |
| 6 | Nhóm thành viên | Định nghĩa cho MOSPF, không được Cisco IOS hỗ trợ kiểu này. |
| 7 | NSSA bên ngoài | Tạo bởi ASBR bên trong một vùng NSSA, ngoại trừ một LSA loại 5. |

LSA Type 1 (Router LSA): Router LSA được tạo ra bởi tất cả các router OSPF và chứa thông tin về các đường link trực tiếp được kết nối với router đó.

LSA Type 2 (Network LSA): Network LSA được tạo ra bởi DR (Designated Router) trong các mạng multi-access và chứa thông tin về tất cả các router trực tiếp được kết nối với mạng đó.

LSA Type 3 (Summary LSA): Summary LSA được tạo ra bởi các router ngoại vi (ABR - Area Border Router) và chứa thông tin về các mạng không nằm trong khu vực của chúng.

LSA Type 4 (ASBR Summary LSA): ASBR Summary LSA được tạo ra bởi các ASBR (Autonomous System Border Router) và chứa thông tin về các mạng nằm bên ngoài AS (Autonomous System) của chúng.

LSA Type 5 (Opaque LSA): Opaque LSA là một loại LSA mở rộng có thể được sử dụng để quảng bá bất kỳ thông tin nào mà người quản trị mạng muốn

Một số cấu hình quảng bá route vào trong miền OSPF:

Ví dụ: để quảng bá tuyến đường mặc định tĩnh vào OSPF, bạn có thể sử dụng lệnh sau:

*router ospf 1*

*redistribute static*

Ví dụ: để tạo tuyến đường mặc định động trong OSPF, bạn có thể sử dụng lệnh sau:

router ospf 1

default-information originate

## Distance vector routing protocol

Distance Vector Routing Protocol (Định tuyến vector khoảng cách) là một loại giao thức định tuyến động trong đó các router trao đổi thông tin về khoảng cách đến các mạng đích. Thông tin này được sử dụng để xây dựng một bảng định tuyến, từ đó các router có thể xác định đường đi tốt nhất để chuyển tiếp các gói dữ liệu.

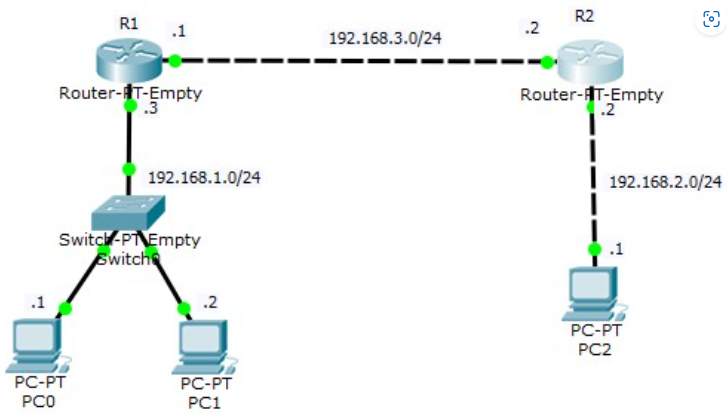
Các giao thức Distance Vector Routing Protocol hoạt động theo các bước sau:

* Tính toán khoảng cách: Mỗi router sẽ tính toán khoảng cách đến các mạng đích dựa trên thông tin mà nó đã nhận được từ các router khác. Khoảng cách thường được biểu thị bằng một số nguyên, chẳng hạn như 1, 2, 3, v.v.
* Trao đổi thông tin: Các router sẽ trao đổi thông tin về khoảng cách đến các mạng đích với các router hàng xóm của chúng.
* Cập nhật bảng định tuyến: Mỗi router sẽ sử dụng thông tin khoảng cách mà nó đã nhận được từ các router hàng xóm để cập nhật bảng định tuyến của mình.

**Giao thức RIP**

Routing Information Protocol (RIP) là giao thức định tuyến vector khoảng cách. Mỗi router sẽ gửi toàn bộ bảng định tuyến của nó cho router láng giềng theo định kỳ 30s/lần. Thông tin này lại tiếp tục được láng giềng lan truyền tiếp cho các láng giềng khác và cứ thế lan truyền ra mọi router trên toàn mạng. RIP chỉ sử dụng metric là hop-count để tính ra tuyến đường tốt nhất tới mạng đích. Vì sử dụng tiêu chí định tuyến là hop-count và bị giới hạn ở số hop là 15 nên giao thức này chỉ được sử dụng trong các mạng nhỏ dưới 15 hop (15 router)

**Cấu hình RIP**

****

Với sơ đồ ví dụ như hình trên, tiến hành cấu hình định tuyến RIP cho các router như sau: Cấu hình router R1: sử dụng RIP version 2

*R1(config)#router rip*

*R1(config-router)#version 2*

*R1(config-router)#network 192.168.1.0*

*R1(config-router)#network 192.168.3.0*

Cấu hình router R2: sử dụng RIP version 2

*R2(config)#router rip*

*R2(config-router)#version 2*

*R2(config-router)#network 192.168.2.0*

*R2(config-router)#network 192.168.3.0*

Tóm lại, để cấu hình RIP cho router thì sử dụng các câu lệnh cơ bản sau:

*Router (config) # router rip*

*Router (config-router) # version 2*

*Router (config-router) # network mang\_can\_quang\_ba*

Ngoài ra còn có các option sau:

- Auto-summary (gộp các subnet lại thành một network chung)

- Default-information originate (quảng bá tuyến default route của nó cho các router cùng chạy RIP bên trong)

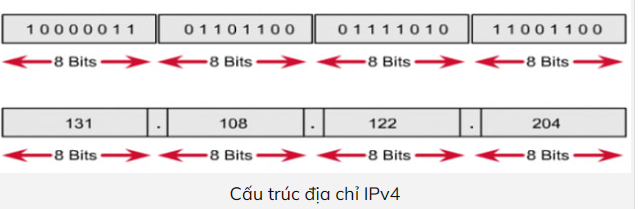
- Redistribute static (quảng bá những static route của nó cho các router cùng chạy RIP bên trong)

- Distance (set giá trị AD)

- Passive-interface (không cho gửi thông tin RIP đến các cổng connected với host để giảm traffic vô ích)

## Địa chỉ Ipv4

Địa chỉ IPv4 sẽ có 32 bit và được biểu diễn thành một dãy số nhị phân và chia thành 4 cụm. Mỗi cụm như vậy sẽ gọi là octet. Mỗi octet sẽ là 8 bit và chúng được ngăn cách bằng dấu chấm (.)



Subnet Mask là một con số 32-bit tương tự, cũng được biểu diễn dạng bốn số thập phân phân cách nhau bằng dấu chấm, dùng để tách địa chỉ IP thành hai phần: mạng (network) và host (host).

Ví dụ:

Giả sử có địa chỉ IP 192.168.1.10 và Subnet Mask 255.255.255.0:

* + - * Địa chỉ IP nhị phân: 11000000.10101000.00000001.00001010
      * Subnet Mask nhị phân: 11111111.11111111.11111111.00000000
      * Mạng nhị phân: 11000000.10101000.00000001.00000000
      * Địa chỉ mạng thập phân: 192.168.1.0

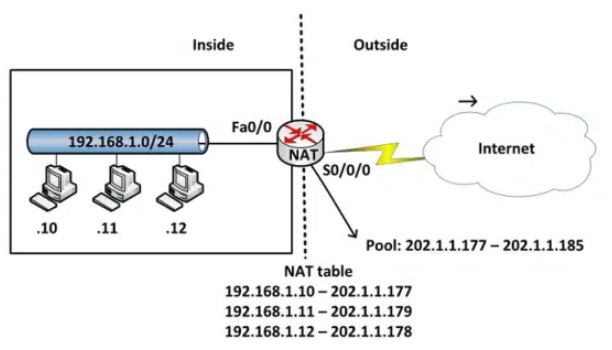
**Phân lớp địa chỉ ipv4**

* + - * **Lớp A**: 1.0.0.0 đến 126.0.0.0/8
      * **Lớp B**: 128.0.0.0 đến 191.255.0.0/16
      * **Lớp C**: 192.0.0.0 -> 223.255.255.0/24
      * **Lớp D:** 224.0.0.0 -> 239.255.255.255

| **ĐỊA CHỈ IP PRIVATE** | **ĐỊA CHỈ IP PUBLIC** |
| --- | --- |
| Phạm vi cục bộ | Phạm vi toàn cầu |
| Được sử dụng để giao tiếp trong mạng | Được sử dụng để giao tiếp bên ngoài mạng |
| Địa chỉ IP private của các hệ thống được kết nối trong mạng sẽ khác nhau, nhưng vẫn theo một quy luật thống nhất. | Địa chỉ IP public có thể khác nhau theo một quy luật đồng nhất hoặc không đồng nhất. |
| Chỉ hoạt động trong [mạng LAN](https://quantrimang.com/cong-nghe/mang-cuc-bo-lan-phan-i-304) | Được sử dụng để truy cập dịch vụ [Internet](https://quantrimang.com/cong-nghe/internet-va-www-khac-nhau-nhu-the-nao-127708) |
| Được sử dụng để load hệ điều hành mạng | Được kiểm soát bởi ISP (nhà cung cấp dịch vụ Internet). |
| Có sẵn miễn phí | Không miễn phí |
| Có thể tìm địa chỉ IP private bằng cách nhập **“ipconfig”** vào [Command Prompt](https://quantrimang.com/cong-nghe/huong-dan-su-dung-command-prompt-85301) | Tìm địa chỉ IP public bằng cách gõ **“what is my ip”** vào [Google](https://quantrimang.com/cong-nghe/google-la-gi-163417) |
| Phạm vi:   * 10.0.0.0 – 10.255.255.255 * 172.16.0.0 – 172.31.255.255 * 192.168.0.0 – 192.168.255.255 | Phạm vi:  Ngoại trừ các địa chỉ IP private, toàn bộ phần còn lại đều là địa chỉ IP public |
| Ví dụ: 192.168.1.10 | Ví dụ: 17.5.7.8 |

## NAT – Network Address Translation

NAT viết tắt từ Network Address Translation. Đây là một kỹ thuật cho phép một hoặc nhiều địa chỉ IP nội miền chuyển đổi sang một hoặc nhiều địa chỉ IP ngoại miền. Thường thì 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.



Có 2 loại NAT:

**Static NAT**

NAT static là một phương pháp ánh xạ một địa chỉ IP riêng (private IP) sang một địa chỉ IP công cộng (public IP) một cách cố định.

Cấu hình:

router(config)#ip nat inside source static

192.168.1.10 10.1.1.10

**NAT Overload**

NAT Overload là một phương pháp ánh xạ nhiều địa chỉ IP riêng sang một địa chỉ IP công cộng.

Cấu hình:

router(config)#ip nat inside source list 1 overload

router(config)#access-list 1 permit 192.168.1.0 0.0.0.255

router#show ip nat translations

Total active translations: 2

Inside source translation prefix destination mask age

---------------------------------------------------------------------------------------

192.168.1.10 10.1.1.10 0.0.0.0 255.255.255.255 00:00:00

192.168.1.11 10.1.1.11 0.0.0.0 255.255.255.255 00:00:00

Cơ chế NAT hoạt động như sau:

* Khi một thiết bị trong mạng nội bộ gửi lưu lượng truy cập đến internet, router sẽ chuyển đổi địa chỉ IP của thiết bị đó thành địa chỉ IP public.
* Router sẽ chuyển tiếp lưu lượng truy cập có địa chỉ IP public đến máy chủ đích trên internet.
* Khi máy chủ đích gửi phản hồi, router sẽ chuyển đổi địa chỉ IP public của máy chủ đích thành địa chỉ IP của máy chủ đích trên internet.

Để lưu lượng truy cập trả về biết được trả về đâu khi sử dụng NAT, router cần lưu trữ thông tin về các bản dịch NAT đang hoạt động. Thông tin này được lưu trữ trong bảng NAT. Bảng NAT bao gồm các mục sau:

* Nguồn IP: Địa chỉ IP private của thiết bị trong mạng nội bộ đã gửi yêu cầu.
* Đích IP: Địa chỉ IP public của máy chủ đích trên internet.
* Cổng: Cổng của yêu cầu.

Khi một thiết bị trong mạng nội bộ gửi yêu cầu đến internet, router sẽ thêm một cổng ngẫu nhiên vào yêu cầu. Máy chủ đích sẽ trả lời yêu cầu với cổng đó. Router sẽ sử dụng cổng này để xác định thiết bị trong mạng nội bộ cần nhận phản hồi.

## ACL – Access Control List

ACL là danh sách tuần tự các dòng rule giúp kiểm soát truy cập, thực hiện phân loại cho phép hoặc từ chối gói tin dựa trên các điểu kiện đã thiết lập

Việc phân loại gói tin bắt đầu từ trên xuống dưới cho đến khi xác định được rule. Khi tìm thấy một rule trùng khớp, hành động thích hợp (permit/deny) sẽ được thực hiện.

Ở cuối mỗi ACL là một rule từ chối ngầm – từ chối tất cả các gói tin không khớp trước đó.

Trong ACL được chia thành các loại sau:

* Numbered standard ACLs: các ACL này xác định các gói tin chỉ dựa trên ip nguồn và chúng được đánh số 1-99 và 1300-1999
* Numbered extended ACLs: các ACL này xác định gói tin dựa trên IP nguồn, giao thức, port hoặc sự kết hợp của nhiều thuộc tính. Chúng được đánh số 100-199 và 2000-2699
* Named ACLs: cho phép cả standard và extended ACL nhưng được đặt tên chứ không phải số.
* Port ACLs (PACL): có thể sử dụng standard, extended, named và thêm chức năng filter MAC ACL tại Layer 2
* VLAN ACLs (VACL): là PACL và có thể filter traffic từ các vlan

ACL sẽ không có hiệu lực cho đến khi chúng được apply lên một interface. Nên để sử dụng một ACL ta cần trải qua 2 bước:

* Định nghĩa ra ACL
* Apply ACL lên một interface

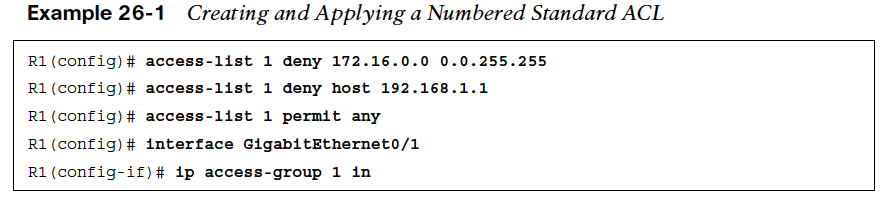
**Numbered ACL:**

Bước 1: *access-list-number { deny | permit } source [source-wildcard] [log]*

Bước 2: *ip access-group {acl-number} {in|out}*

Từ khóa any có thể hiểu là 0.0.0.0 255.255.255.255

Từ khóa host có nghĩa là chỉ định một host duy nhất (wildcard mask 0.0.0.0)

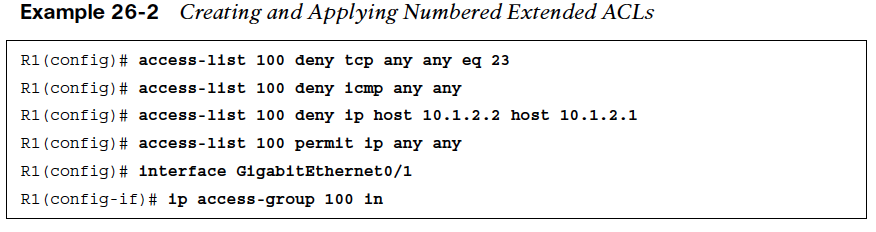


|  |  |
| --- | --- |
| **ACE Entry** | **Networks** |
| Permit any | Permits all networks |
| permit 172.16.0.0 0.0.255.255 | Permits all networks in the 172.16.0.0/16 range |
| permit host 192.168.1.1 | Permits only the 192.168.1.1/32 network |

**Numbered Extended ACL**

**Step 1. access-list** *acl-number* {**deny**|**permit**} *protocol source source-wildcard destination destination-wildcard* [*protocol-options*] [**log** | **log-input**]

**Step 2. ip access-group** {*acl-number*} {**in**|**out**}

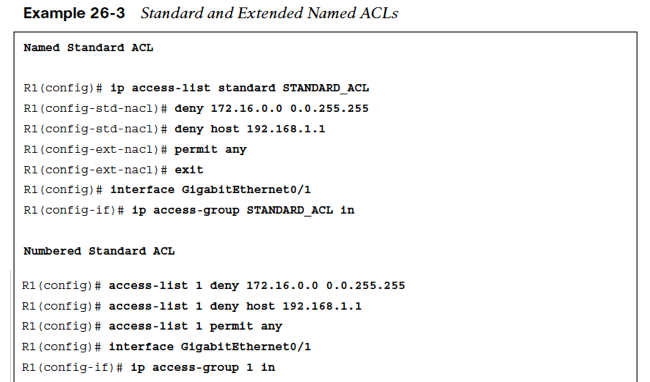


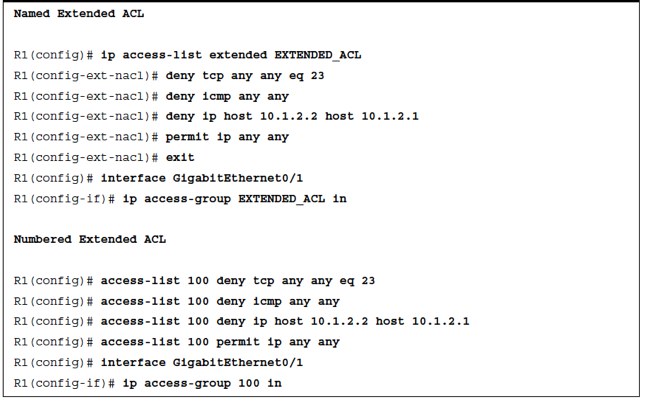
**Named ACL**

**Step 1. ip access-list standard|extended** {acl-number | acl-name}.

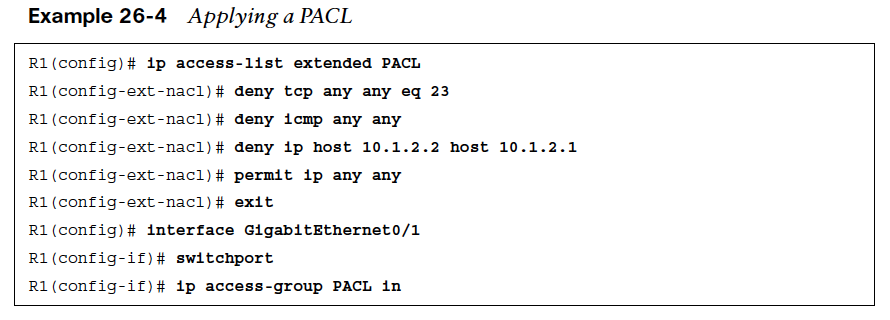
**Step 2. [sequence] {permit | deny}** *source source-wildcard*.

**Step 3. ip access-group** { acl-number | acl-name } {in|out}





**Port ACL:**



**VLAN ACL:**

**Step 1.** Define a VLAN access map by using the command **vlan access-map** *name sequence*.

**Step 2.** Configure the match statement by using the command **match** { **ip address** { *acl-number* | *acl-name* } | **mac address** *acl-name* }.

**Step 3.** Configure the action statement by using the command **action forward**|**drop** [**log**]. The action statement specifies the action to be taken when a match occurs.

**Step 4.** Apply the VACL by using the command **vlan filter** *vlan-access-map-name* *vlan-list*. *vlan-list* can be a single VLAN, a range of VLANs (such as 5–30), or a comma-separated list of multiple VLANs (such as 1,2–4,6)

