Dynamic Routing and Access Control List

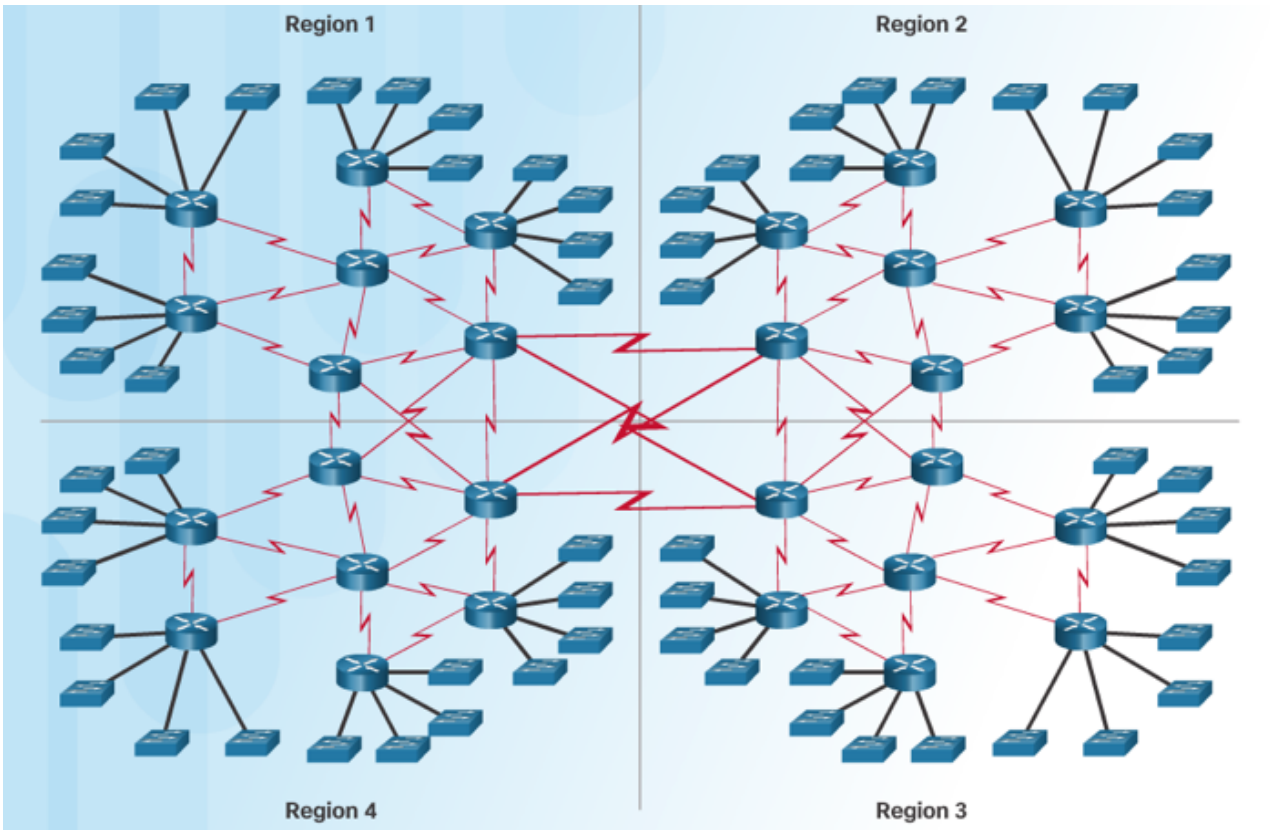
1. Định tuyến động

a) Giới thiệu tổng quan

Định tuyến được chia thành 2 loại:

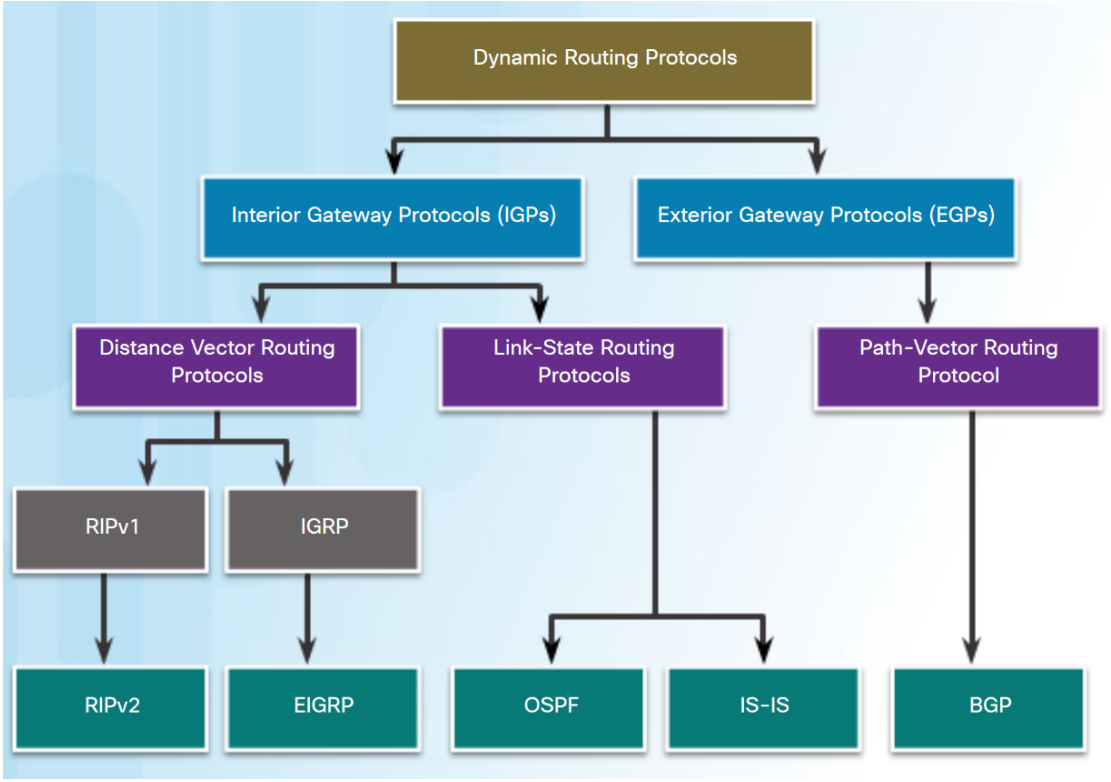
- **Định tuyến tĩnh** (Static Routing) *(đã thực hiện tại Lab 1)*: Người quản trị tự xây dựng bảng định tuyến cho mỗi Router trong mô hình.
  - *Ưu điểm:* Router không mất chi phí (CPU, RAM, bandwidth) để tính toán và xây dựng bảng định tuyến.
  - *Hạn chế*: Tính chính xác phụ thuộc hoàn toàn và hiểu biết và khả năng của người quản trị. Khi có sự thay đổi trong mạng (Ví dụ thêm 1 Router vào mô hình) thì người quản trị phải thực hiện định tuyến trên tất cả các Router trong mô hình.

- **Định tuyến động** (Dynamic Routing): Người quản trị cho Router biết cách học hỏi các đường đi trong mạng, sau đó các Router học hỏi lẫn nhau để xây dựng nên bảng định tuyến.



Hình 1. Định tuyến động sẽ là giải pháp hiệu quả với các mô hình mạng lớn

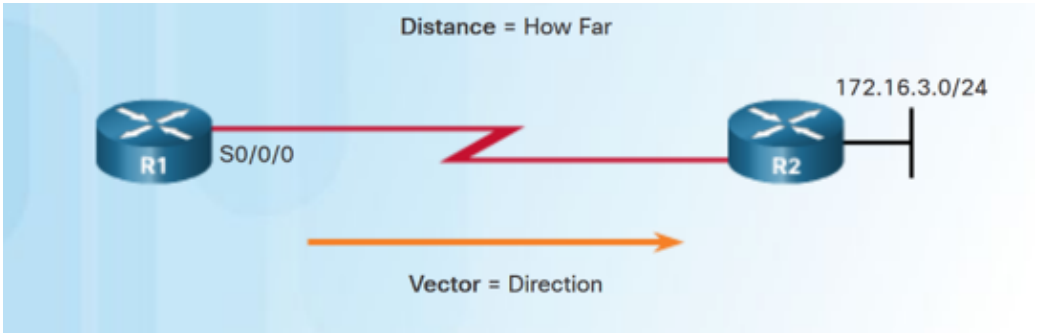
  - *Ưu điểm*: Khi có sự thay đổi trong mạng (VD: thêm 1 Router vào mô hình) thì người quản trị chỉ cần khởi động giao thức định tuyến trên Router vừa thêm vào.
  - *Hạn chế*: Router mất chi phí cho việc học hỏi và duy trì bảng định tuyến.



Hình 2. Các loại định tuyến động tiêu biểu

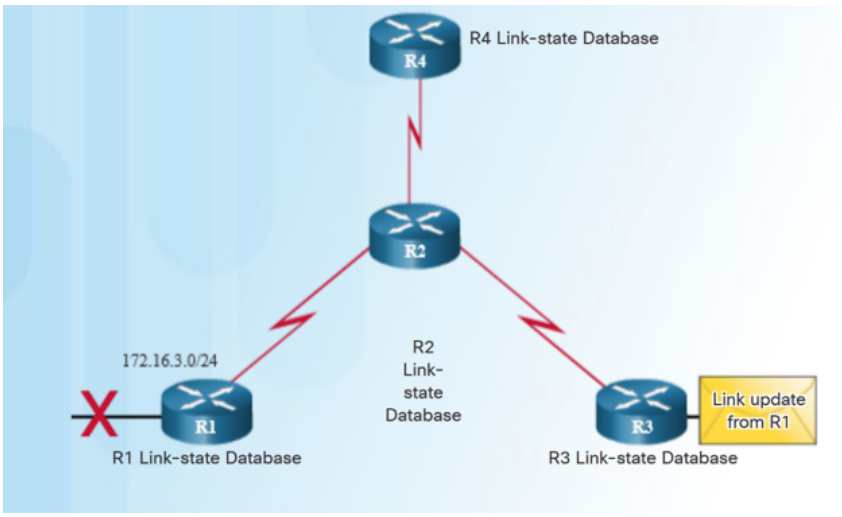
Định tuyến động cũng được chia làm 2 trường phái chính:

- **Distance vector:** hoạt động dựa trên sự "tin tưởng" lẫn nhau giữa các Router. Các Router chia sẻ thông tin định tuyến với nhau và sau một số chu kỳ thì các Router sẽ biết được thông tin các mạng trong toàn bộ mô hình (lúc này mạng được xem là hội tụ). Chi phí trong distance vector được tính bằng **hop count**, là số Router từ nguồn đến đích. Giao thức distance vector thường dùng nhất là RIP. Hiện tại RIP có 2 phiên bản 1 và 2. Phiên bản 1 dùng cho mạng Classfull và phiên bản 2 dùng cho mạng Classless.



Hình 3. 2 đặc điểm của Distance Vector

- **Link State**: Khác với distance vector, mỗi Router chạy giao thức định tuyến link state thu thập thông tin về tất cả các mạng trong mô hình và chứa trong database. Sau đó rút trích những đường đi tốt nhất đến mỗi mạng từ database này để xây dựng nên bảng định tuyến. Giao thức link state thường dùng nhất là **OSPF** phiên bản 2 dành cho IPv4 và phiên bản 3 dành cho IPv6.



Hình 4. Update sẽ được gửi khi có thay đổi trạng thái của 1 link

b) Định tuyến RIPv2 (Routing Information Protocol version 2)

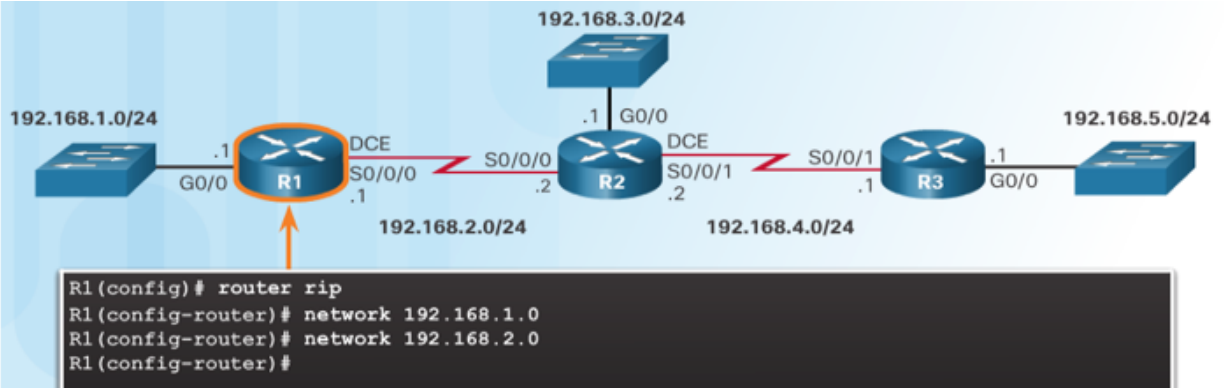
- Hỗ trợ IPv4, mạng Classless
- **Metric** = Hop count (tối đa 15). *Khi packet ra khỏi 1 router thì metric tăng 1*
- **Nguyên tắc hoạt động:** Các Router kết nối trực tiếp với nhau sẽ trao đổi với nhau bằng gói tin *Hello* chứa bảng định tuyến của nó với chu kỳ 30s.

Để thực hiện định tuyến RIPv2, tại Router cần định tuyến, thực hiện các lệnh sau:

```
Router(config)# router rip
Router(config-router)# version 2
Router(config-router)# network <major-network address>
Router(config-router)# no auto-summary
```

Trong đó:

- major-network address là địa chỉ mạng gắn với interface của router được tham gia định tuyến RIP. Ví dụ:



- Tại các interface không cần định kỳ cập nhật bảng định tuyến (ví dụ interface nối với mạng LAN các PC), chúng ta có thể tắt việc cập nhật bảng định tuyến mỗi 30s qua nó, chỉ nhận vào bảng định tuyến (nếu có) để tăng hiệu suất cho mạng bằng cách sử dụng lệnh:

```
Router(config-router)# passive-interface <interface name>
```

c) Định tuyến OSPF - Single Area (Open Shortest Path First)

- Hỗ trợ IPv4, mạng Classless
- **Admin Distance (AD)** = 110
- **Metric** phụ thuộc vào bandwidth
- **Nguyên tắc hoạt động:** Các router sử dụng thuật toán Dijkstra để tìm đường đi tối ưu nhất trong mạng.

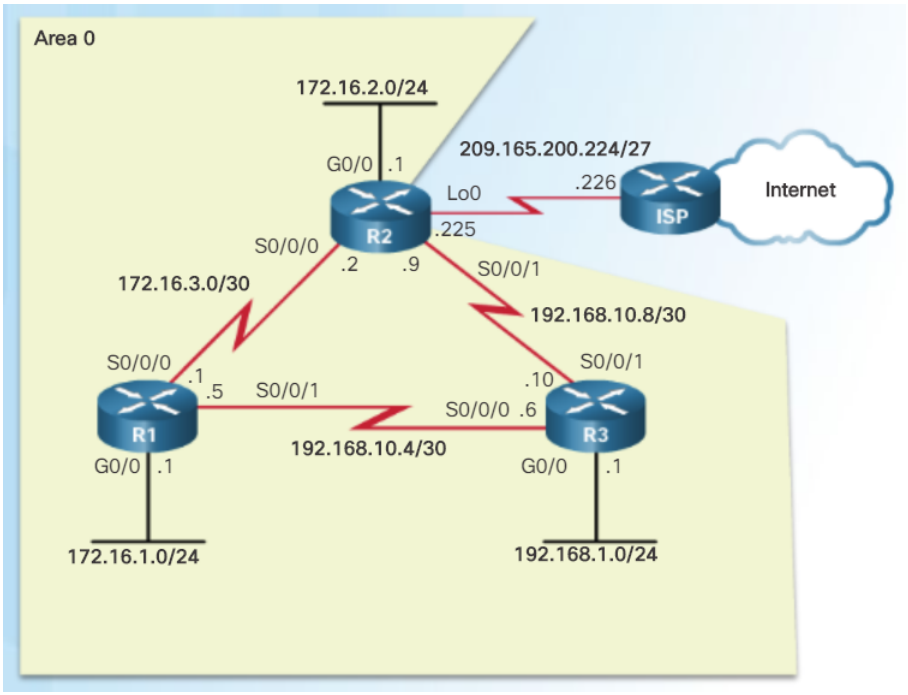
Tại Router cần định tuyến OSPF, thực hiện các lệnh sau:

```
Router(config)# router ospf <Process-ID>
Router(config-router)# network <IP-Address> <wildcard_mask> area <area-id>
```

Trong đó:

- Process-ID là định danh cho mỗi tiến trình OSPF trên Router *(mỗi Router có thể chạy nhiều tiến trình OSPF)*, có giá trị <1-65535>
- Có 2 cách sử dụng lệnh network:

  Ví dụ với mô hình sau:



  - Cách 1 (truyền thống): quảng bá địa chỉ mạng với wildcard mask

```
R1(config)# router ospf 10
R1(config-router)# network 172.16.1.0 0.0.0.255 area 0
R1(config-router)# network 172.16.3.0 0.0.0.3 area 0
R1(config-router)# network 192.168.10.4 0.0.0.3 area 0
```

  - Cách 2: quảng bá IP interface với wildcard 0.0.0.0

```
R1(config)# router ospf 10
R1(config-router)# network 172.16.1.1 0.0.0.0 area 0
R1(config-router)# network 172.16.3.1 0.0.0.0 area 0
R1(config-router)# network 192.168.10.5 0.0.0.0 area 0
```

- wildcard_mask xác định một IP hay một network nào được tham gia định tuyến OSPF, có thể xác định bằng cách lấy 255.255.255.255 – subnetmask

  Ví dụ:

Trong wildcard_mask, vị trí bit 0 là bit cố định (match), bit 1 là tùy ý (ignore).

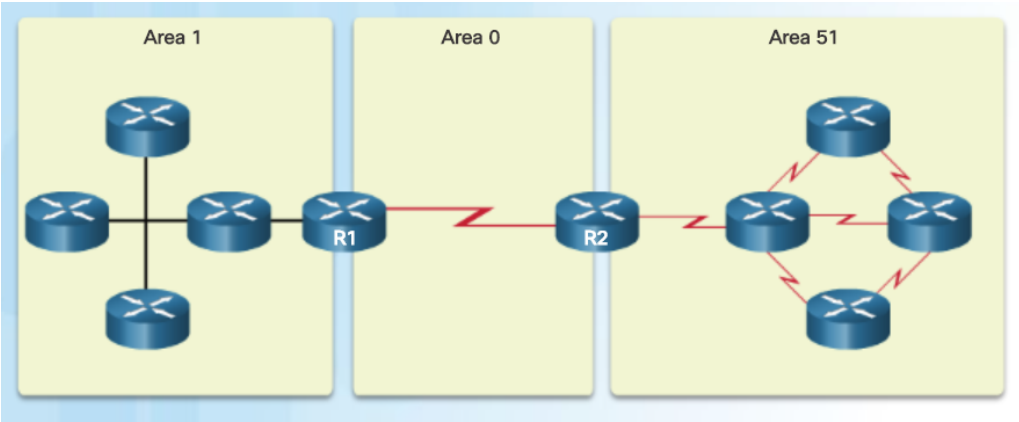
Ví dụ:

10.10.1.1 0.0.0.0 – cố định cả 32 bit – hay chỉ định chính xác IP 10.10.1.1

10.10.1.0 0.0.0.255 – cố định 24 bit đầu, thể hiện cho địa chỉ mạng 10.10.1.0/24

- area number là định danh cho vùng cần chạy OSPF, tất cả các Router chạy định tuyến OSPF trong vùng đều phải có cùng area number, area number có giá trị <0-4294967295>.

  Lưu ý: *Luôn có ít nhất 1 area 0 (backbone area), tất cả các area khác nếu có phải kết nối trực tiếp với area 0 này.*



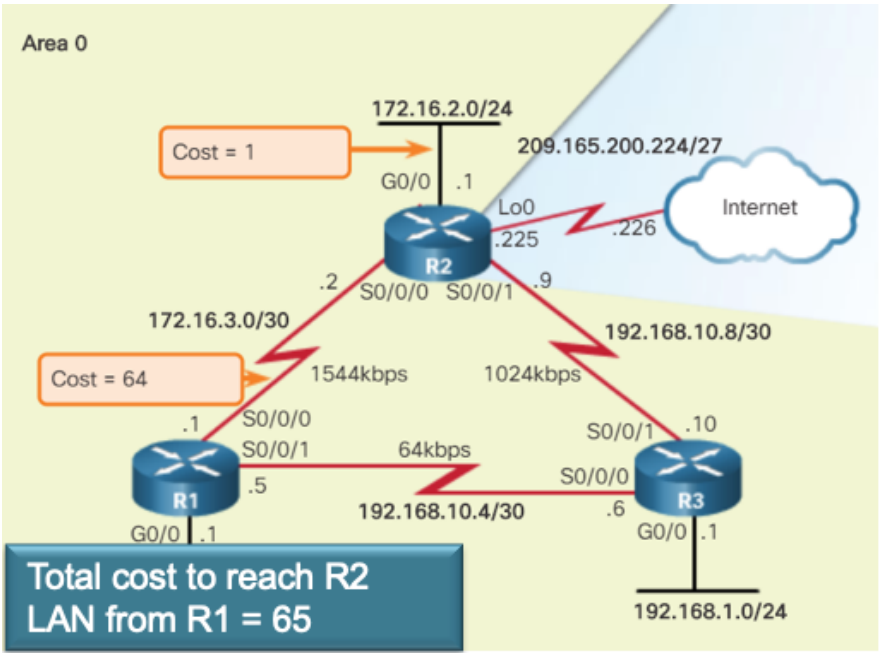
Trong bài thực hành chỉ yêu cầu định tuyến OSPF single area, do đó sử dụng area 0.

- Việc cấu hình Passive interface (nếu có) tương tự RIPv2.
- **Tính metric trong OSPF**: OSPF sử dụng cost để xác định đường đi tốt nhất để đến mạng đích. Cost được tính khi đi vào 1 interface, đi ra không tính.

  Cost = reference bandwidth / interface bandwidth

Interface Type	Reference Bandwidth in bps		Default Bandwidth in bps	Cost
10 Gbps Ethernet	100,000,000	÷	10,000,000,000	1
1 Gbps Ethernet	100,000,000	÷	1,000,000,000	1
100 Mbps Ethernet	100,000,000	÷	100,000,000	1
10 Mbps Ethernet	100,000,000	÷	10,000,000	10
1.544 Mbps Serial	100,000,000	÷	1,544,000	64
128 kbps Serial	100,000,000	÷	128,000	781
64 kbps Serial	100,000,000	÷	64,000	1562

Hình 5. Bảng cost tham khảo cho các loại interface phổ biến



Hình 6. Ví dụ về tính cost từ R1 đến R2 LAN (172.16.2.0/24)

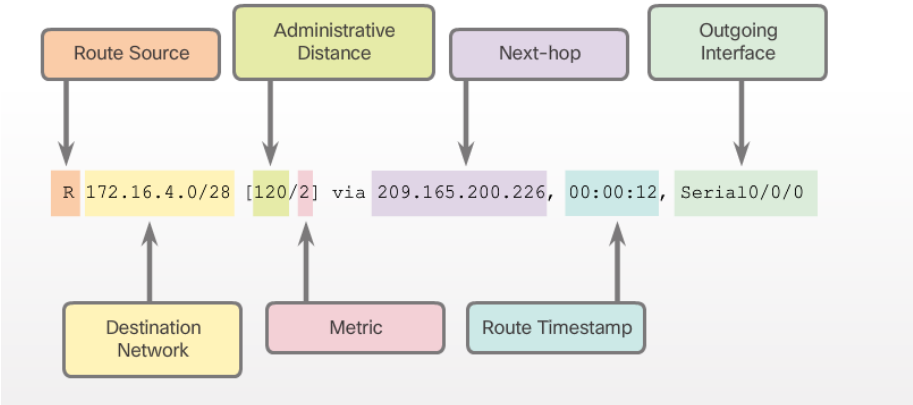
d) Kiểm tra kết quả định tuyến

Để thực hiện kiểm tra kết quả thông tin định tuyến, ta có thể sử dụng các lệnh sau:
show ip protocols - *Xem cấu hình định tuyến IPv4 trên Router*

```
R1# show ip protocols
*** IP Routing is NSF aware ***

Routing Protocol is "rip"
  Outgoing update filter list for all interfaces is not set
  Incoming update filter list for all interfaces is not set
  Sending updates every 30 seconds, next due in 16 seconds
  Invalid after 180 seconds, hold down 180, flushed after 240
  Redistributing: rip

  Default version control: send version 1, receive any version
    Interface             Send  Recv  Triggered RIP  Key-chain
    GigabitEthernet0/0    1     1 2
    Serial0/0/0           1     1 2

  Automatic network summarization is in effect
  Maximum path: 4
  Routing for Networks:
    192.168.1.0
    192.168.2.0

  Routing Information Sources:
    Gateway         Distance      Last Update
    192.168.2.2           120      00:00:15
  Distance: (default is 120)
```

`show ip route` – *Xem bảng định tuyến trên Router*

Ý nghĩa các thông số trong mỗi dòng trong bảng định tuyến như sau:



Ví dụ:

```
R1# show ip route | begin Gateway
Gateway of last resort is not set

      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, GigabitEthernet0/0
L        192.168.1.1/32 is directly connected, GigabitEthernet0/0
      192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.2.0/24 is directly connected, Serial0/0/0
L        192.168.2.1/32 is directly connected, Serial0/0/0
R     192.168.3.0/24 [120/1] via 192.168.2.2, 00:00:24, Serial0/0/0
R     192.168.4.0/24 [120/1] via 192.168.2.2, 00:00:24, Serial0/0/0
R     192.168.5.0/24 [120/2] via 192.168.2.2, 00:00:24, Serial0/0/0
R1#
```

2. Access Control List

ACLs (Access control lists) hay còn gọi là access lists, là một danh sách tuần tự các câu lệnh hay còn gọi là ACEs (Access control entries), được áp dụng trên một Interface nào đó theo chiều vào hoặc ra. Danh sách này cho biết loại gói tin nào được chấp nhận (permit) hay từ chối (deny) khi Router xử lý.

- Nguyên tắc hoạt động của ACLs là duyệt từ trên xuống dưới các entry trong ACLs, nếu dòng nào khớp thì sẽ được áp dụng xử lý và bỏ qua tất cả các dòng còn lại.

- Cuối access list, mặc định sẽ có entry deny any (cấm tất cả traffic)

ACLs gồm 2 loại:

- ***Standard ACL (ACL cơ bản):*** thực hiện kiểm tra địa chỉ nguồn của gói tin. Đặt ở gần mạng đích nhất có thể *(Router quản lý lớp mạng đích, theo chiều ra của Interface)*.

  Cú pháp:

```
Router(config)# access-list access-list-number {deny | permit | remark}
source [source-wildcard]
```

  Ví dụ:

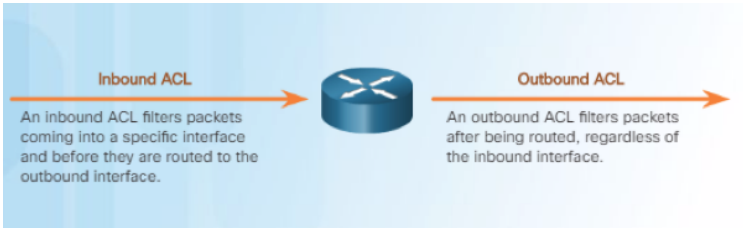
```
access-list 10 permit 192.168.30.0 0.0.0.255
```

  Trong đó:

  - access-list-number trong khoảng từ **1 - 99** và **1300 - 1999**
  - wildcard-mask**:** thể hiện một hoặc một dãy liên tục hoặc không liên tục các IP có cùng một tính chất nào đó. Tham khảo phần OSPF
  - Có thể dùng từ khóa
    - **any**: thể hiện cho tất cả địa chỉ IP (tương đương 0.0.0.0 255.255.255.255)
    - **host**: thể hiện cho 1 địa chỉ IP (tương đương <IP> 0.0.0.0)

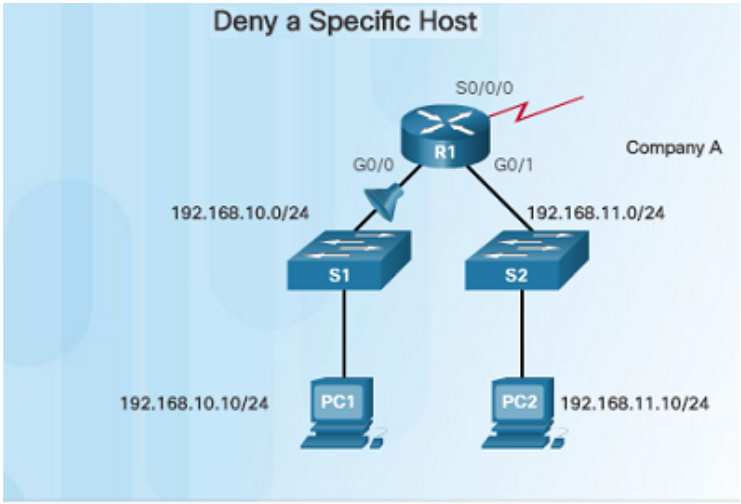
Sau đó, cần gắn wildcard mask vào một interface cụ thể theo một chiều nhất định thì wildcard mask đó mới có tác dụng.

```
Router(config-if)# ip access-group { access-list-number | access-list-name } { in | out }
```



Hình 7. Xác định chiều in/out của ACLs tương ứng với Interface

Ví dụ: Standard ACL sau được cấu hình trên Interface G0/0 của Router R1, cấm tất cả traffic từ host 192.168.10.10/24




```
R1(config)# no access-list 1
R1(config)# access-list 1 deny host 192.168.10.10
R1(config)# access-list 1 permit any
R1(config)# interface g0/0
R1(config-if)# ip access-group 1 in
```

- ***Extended ACL (ACL mở rộng):*** thực hiện kiểm tra địa chỉ nguồn, địa chỉ đích, giao thức và port của gói tin. Đặt ở gần mạng nguồn nhất có thể *(Router quản lý lớp mạng nguồn, theo chiều vào của Interface).* Cú pháp:

```
access-list ACL-# {deny | permit | remark} protocol {source source-wildcard][operator [port-number | port-name]] {destination destination-wildcard][operator [port-number | port-name]]
```

Ví dụ:

```
access-list 103 permit tcp 192.168.30.0 0.0.0.255 any eq 80
```

Trong đó:

- Access-list-number trong khoảng từ 100-199 và 2000-2699
- Source port và Destination port phải khai báo theo dạng
    - eq: bằng
    - gt: lớn hơn
    - lt: nhỏ hơn

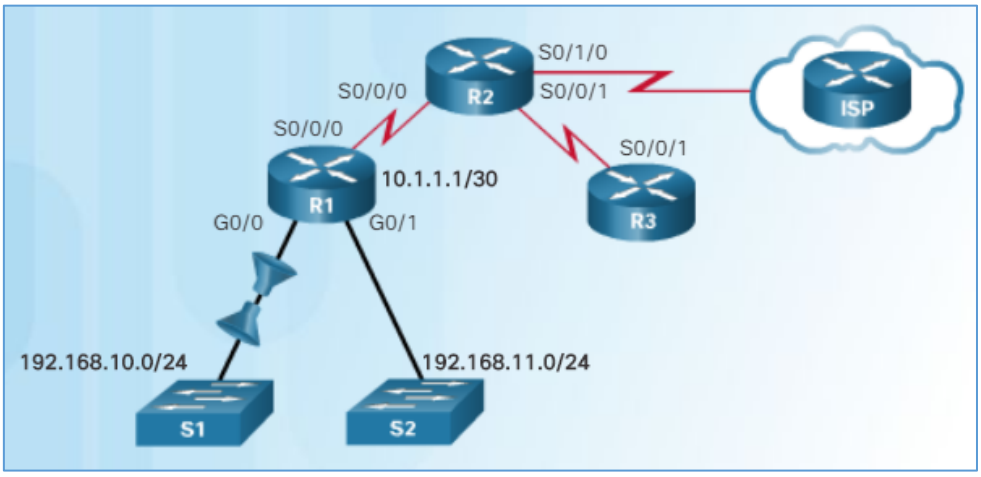
Port Number	Protocol	Application	Acronym
20	TCP	File Transfer Protocol (data)	FTP
21	TCP	File Transfer Protocol (control)	FTP
22	TCP	Secure Shell	SSH
23	TCP	Telnet	–
25	TCP	Simple Mail Transfer Protocol	SMTP
53	UDP, TCP	Domain Name Service	DNS
67	UDP	Dynamic Host Configuration Protocol (server)	DHCP
68	UDP	Dynamic Host Configuration Protocol (client)	DHCP
69	UDP	Trivial File Transfer Protocol	TFTP
80	TCP	Hypertext Transfer Protocol	HTTP
110	TCP	Post Office Protocol version 3	POP3
143	TCP	Internet Message Access Protocol	IMAP
161	UDP	Simple Network Management Protocol	SNMP
443	TCP	Hypertext Transfer Protocol Secure	HTTPS

Hình 8. Một số Port các giao thức thông dụng

Ví dụ: Các extended ACL sau được cấu hình tại Interface G0/0 của Router R1 gồm:

- Cho cho phép các gói tin sử dụng giao thức HTTP/HTTPS từ mạng 192.168.10.0/24 đi ra ngoài.

- Cho phép các gói tin sử dụng giao thức TCP từ ngoài vào mạng 192.168.10.0/24.




```
R1(config)# access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 80
R1(config)# access-list 103 permit tcp 192.168.10.0 0.0.0.255 any eq 443
R1(config)# access-list 104 permit tcp any 192.168.10.0 0.0.0.255 established
R1(config)# interface g0/0
R1(config-if)# ip access-group 103 in
R1(config-if)# ip access-group 104 out
```

Thông số **established** thể hiện cho việc chỉ chấp nhận phản hồi (responses) với các traffic đã xuất phát từ mạng 192.168.10.0/24 trở về mạng này.

Mở rộng:

- Cặp IP và Wildcard mask : 0.0.0.0 255.255.255.0 có ý nghĩa là gì?

- Tính wildcard mask cho dãy IP: 192.168.1.15 → 192.168.1.75

- Viết một ACLs cấm tất cả IP chẵn của mạng 192.168.10.0/24 không được sử dụng giao thức HTTP/HTTPS ra ngoài Internet.

- Kiểm tra ACLs:

Sử dụng lệnh show ip interface <tên interface> và show access-lists

```
R1# show ip interface s0/0/0
Serial0/0/0 is up, line protocol is up
  Internet address is 10.1.1.1/30
<output omitted>
  Outgoing access list is 1
  Inbound access list is not set
<output omitted>

R1# show ip interface g0/0
GigabitEthernet0/0 is up, line protocol is up
  Internet address is 192.168.10.1/24
<output omitted>
  Outgoing access list is NO_ACCESS
  Inbound access list is not set
<output omitted>
```

```
R1# show access-lists
Standard IP access list 1
  10 deny 192.168.10.10
  20 permit 192.168.0.0, wildcard bits 0.0.255.255
Standard IP access list NO_ACCESS
  15 deny 192.168.11.11
  10 deny 192.168.11.10
  20 permit 192.168.11.0, wildcard bits 0.0.0.255
R1#
```

Ngoài ra, còn có cách cấu hình **Name ACLs**.
Cú pháp:

```
Router(config)#ip access-list {standard | extended} <tên ACL>
```

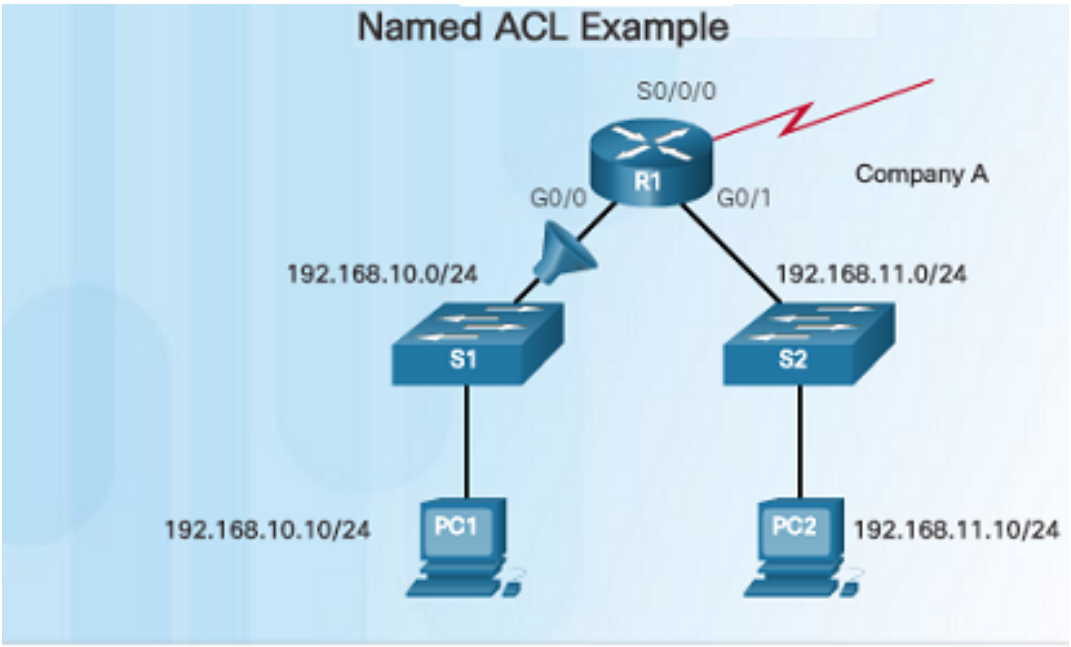
Sau đó, ACLs sẽ chuyển vào mode Standard hay Extended tương ứng. Chèn các entry như sau:

```
Router(config-{std | ext} -nacl}#{sequence number} {permit | deny} ...
```

Router(config-{std | ext} -nacl}#{sequence number} {permit | deny} ...

- Sequence number là số thứ tự của ACE trong Access lists.
- Tùy theo ACLs cơ bản hay mở rộng mà thực hiện phần cấu hình còn lại tương ứng.
- Muốn xóa dòng ACE nào, chỉ cần thêm no phía trước dòng đó. Đây là ưu điểm hơn so với khi muốn xóa một ACLs cấu hình với ACL number.

Ví dụ:




```
R1(config)# ip access-list standard NO_ACCESS
R1(config-std-nacl)# deny host 192.168.11.10
R1(config-std-nacl)# permit any
R1(config-std-nacl)# exit
R1(config)# interface g0/0
R1(config-if)# ip access-group NO_ACCESS out
```