



# Module 04 : Enabling EIGRP and OSPF

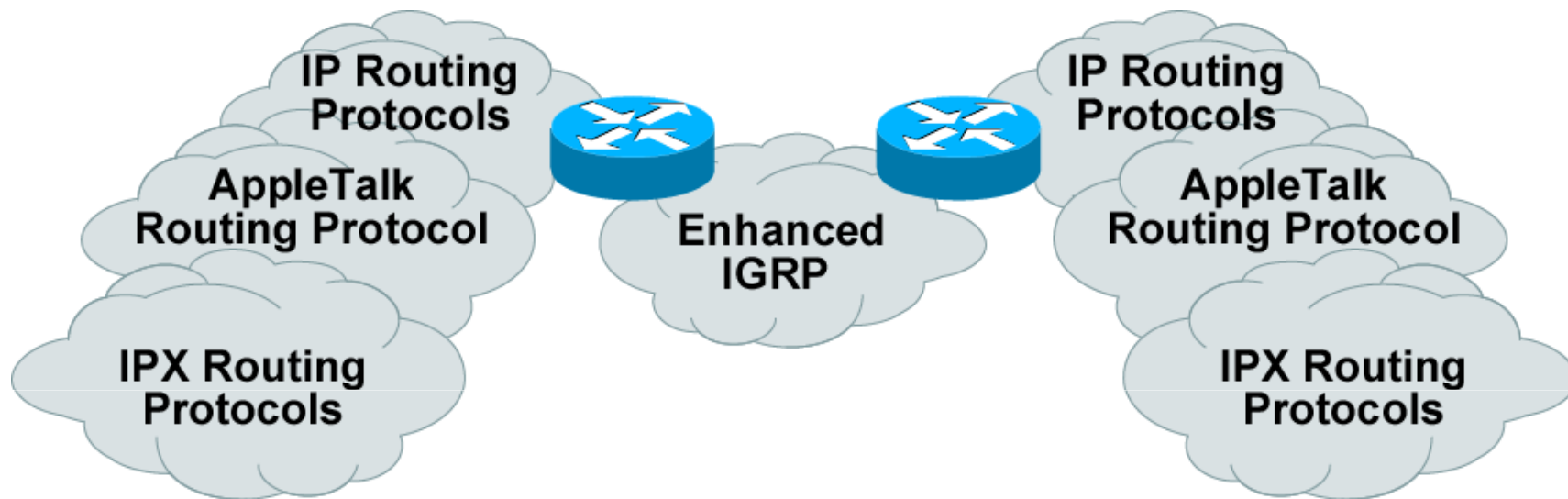
---

**Enabling EIGRP**

**Enabling OSPF**



# Introducing EIGRP



- **EIGRP Support :**

- **Network Topology** 변화에 수렴 시간이 빠르다.
- **Multiple Routed Protocol (IP, IPX, Apple talk)**을 지원함.
- **EIGRP**는 **Auto Summary** 및 **Manual Summary**를 지원한다.
- 정상 운용 중에 적은 **Network** 자원을 이용해 **Routing Table**을 유지한다.
- **Classless Routing Protocol** 지원



# EIGRP Packets

---

- **Hello:**
  - **Establish neighbor relationships**
  - **Ack 번호 0을 가지고 multicast (224.0.0.10)**
- **Update:**
  - **Send routing updates**
  - **New neighbor 발견 시 topology table 동기화 : unicast**
  - **Topology Change 발견 시 : multicast**
- **Query:**
  - **Ask neighbors about routing information : always multicast**
- **Reply:**
  - **Response to query about routing information : unicast**
- **ACK:**
  - **Acknowledgement of a reliable packet (update, query, reply)**
  - **Nonzero ack 번호를 가진 unicast**



# EIGRP Terminology

---

Neighbor Table - IP	
Next-Hop Router	Interface



Topology Table - IP	
Destination 1	Successor
Destination 2	Feasible Successor



Routing Table - IP	
Destination 1	Successor



# EIGRP Terminology

---

- **Neighbor Table** : EIGRP router는 인접 Router(직접 연결된 neighbor router)에 대한 table을 보유하여 인접 router간의 양방향 통신을 확립한다. 지원하는 프로토콜(IP, IPX, AppleTalk)별로 각각의 Neighbor Table을 유지한다.
- **Topology Table** : EIGRP router는 routing 정보교환에 의하여 알게 된 모든 네트워크에 대한 Topology Table을 유지한다. 지원하는 프로토콜 별로 각각의 Topology Table을 유지한다.
- **Routing Table** : EIGRP router는 Topology Table을 기초로 하여, Destination에 대한 최적의 경로를 routing table에 보유한다. 지원하는 프로토콜 별로 각각의 Routing Table을 유지한다.
- **Successor** : Destination에 대한 Primary Route이다. Routing table에 유지된다.
- **Feasible Successor** : Destination에 대한 Backup Route이다. Feasible Successor는 Successor와 동시에 선택되는데 Topology Table에 보유 된다. Destination에 대해서 여러 개의 Feasible Successor를 보유할 수 있다.



## Comparing EIGRP and IGRP

---

- **Metric** 운영은 비슷한 방법을 사용한다.
- **load balancing** 지원은 동일하다.
- **Convergence time**은 개선 되었다.
- **Network overhead**는 많이 줄어 들었다.



## Configuring EIGRP

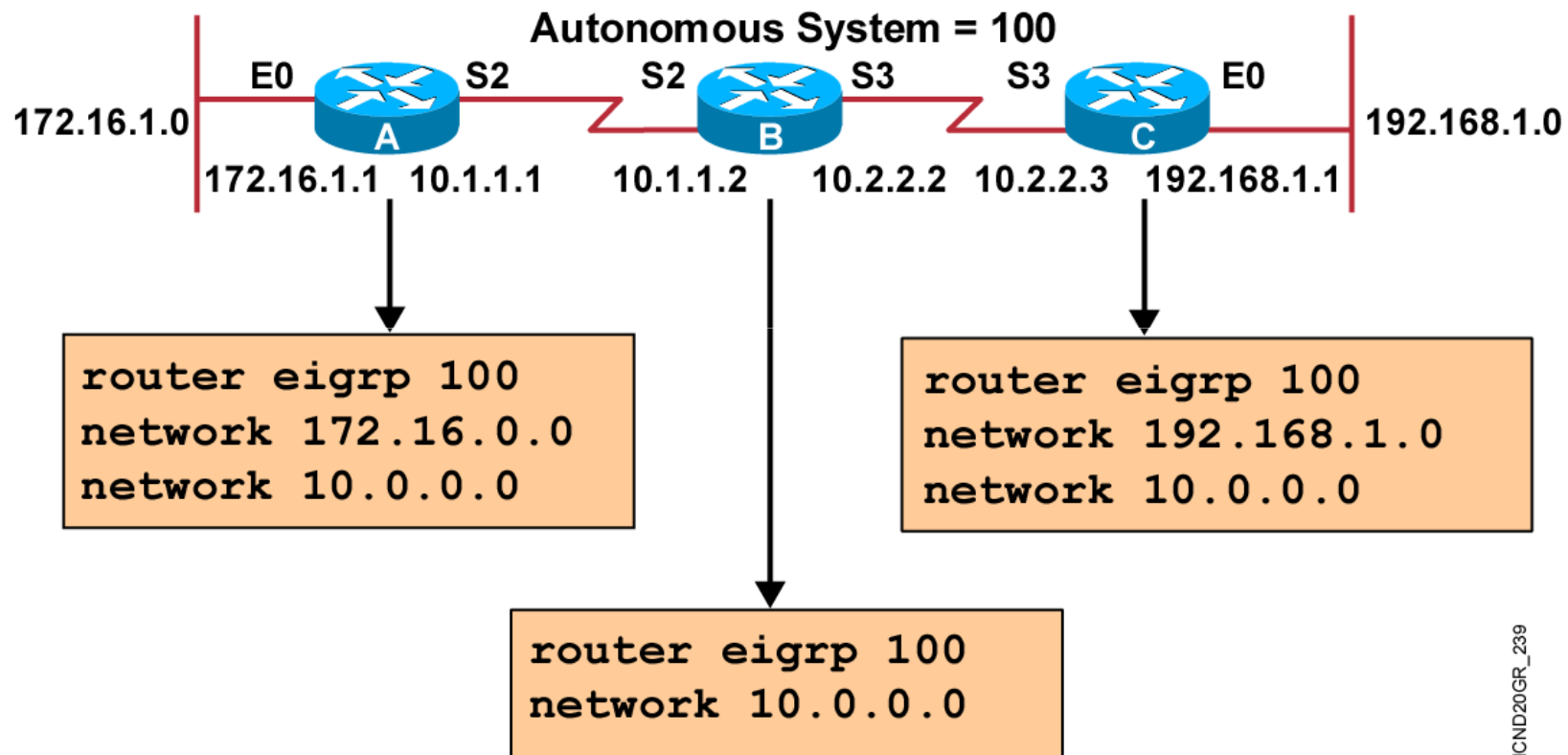
```
Router(config)#router eigrp autonomous-system
```

- IP Routing Protocol을 위한 **EIGRP** 설정하기

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

- 연결된 **Network** 설정하기 (**Classful Network Address**를 할당한다.)

# EIGRP Configuration Example



ICND20GR\_239





# Verifying the EIGRP Configuration

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

- IP EIGRP Neighbor를 표시한다.

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

- IP EIGRP Topology를 표시한다.

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


- Routing Table에 EIGRP Route 정보를 표시한다.

```
Router#show ip protocols
```

- 현재 Router에 구성된 Active Routing Protocol에 상세 정보를 표시한다.

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

- IP EIGRP 송수신 Packet에 수를 표시한다.



# Verifying EIGRP Operation

---

! 송수신 되는 EIGRP Packet을 보여준다.

**Router# debug eigrp packet**

! EIGRP Advertisement 및 EIGRP가 Routing Table에 만드는 변경을 보여준다.

**Router# debug ip eigrp route**

! EIGRP Routing Activity의 간략한 정보를 보여준다.

**Router# debug ip eigrp summary**

! Route Calculation을 포함한 다양한 EIGRP Activity에 대한 간략한 정보를 보여 준다.

**Router# show ip eigrp events**



## debug ip eigrp Command

```
Router#debug ip eigrp
IP-EIGRP: Processing incoming UPDATE packet
IP-EIGRP: Ext 192.168.3.0 255.255.255.0 M 386560 - 256000 130560 SM
360960 - 256000 104960
IP-EIGRP: Ext 192.168.0.0 255.255.255.0 M 386560 - 256000 130560 SM
360960 - 256000 104960
IP-EIGRP: Ext 192.168.3.0 255.255.255.0 M 386560 - 256000 130560 SM
360960 - 256000 104960
IP-EIGRP: 172.69.43.0 255.255.255.0, - do advertise out Ethernet0/1
IP-EIGRP: Ext 172.69.43.0 255.255.255.0 metric 371200 - 256000 115200
IP-EIGRP: 192.135.246.0 255.255.255.0, - do advertise out Ethernet0/1
IP-EIGRP: Ext 192.135.246.0 255.255.255.0 metric 46310656 - 45714176
596480
IP-EIGRP: 172.69.40.0 255.255.255.0, - do advertise out Ethernet0/1
IP-EIGRP: Ext 172.69.40.0 255.255.255.0 metric 2272256 - 1657856 614400
IP-EIGRP: 192.135.245.0 255.255.255.0, - do advertise out Ethernet0/1
IP-EIGRP: Ext 192.135.245.0 255.255.255.0 metric 40622080 - 40000000
622080
IP-EIGRP: 192.135.244.0 255.255.255.0, - do advertise out Ethernet0/1
```



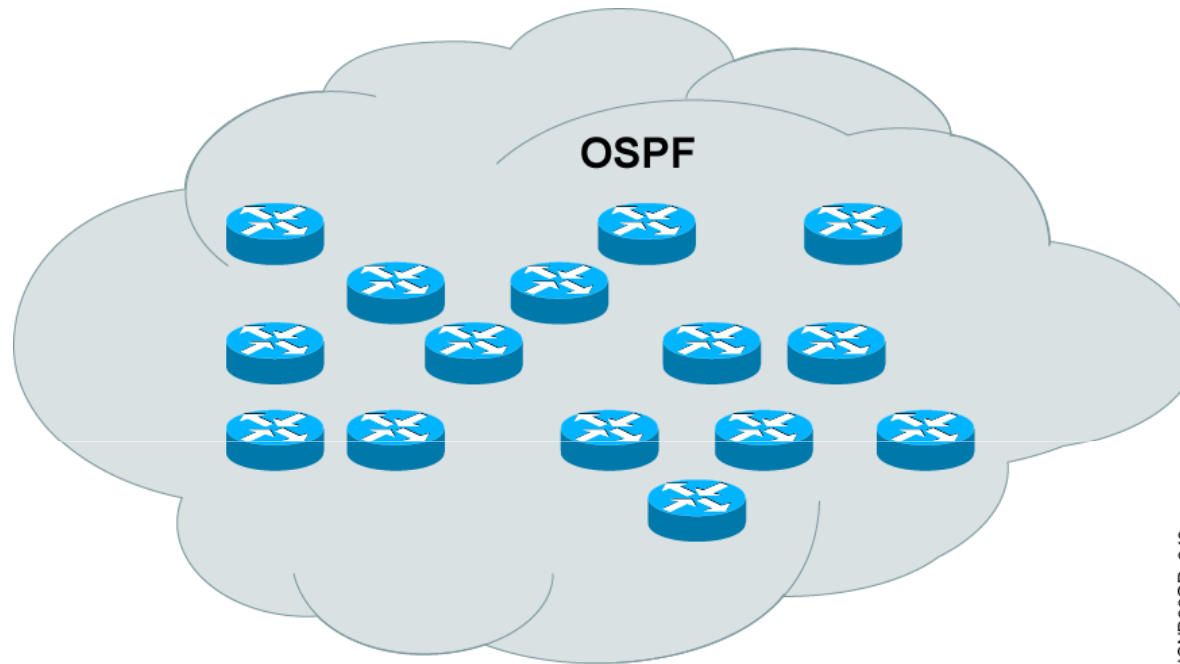
# **EIGRP Enable LAB**



# Enabling OSPF



# Introducing OSPF



ICND20GR\_240

- **OSPF**는 **IETF** 표준이다.
- **Shortest Path First (SPF)** 알고리즘을 사용한다.
- **Link-state Routing Protocol** 이다.

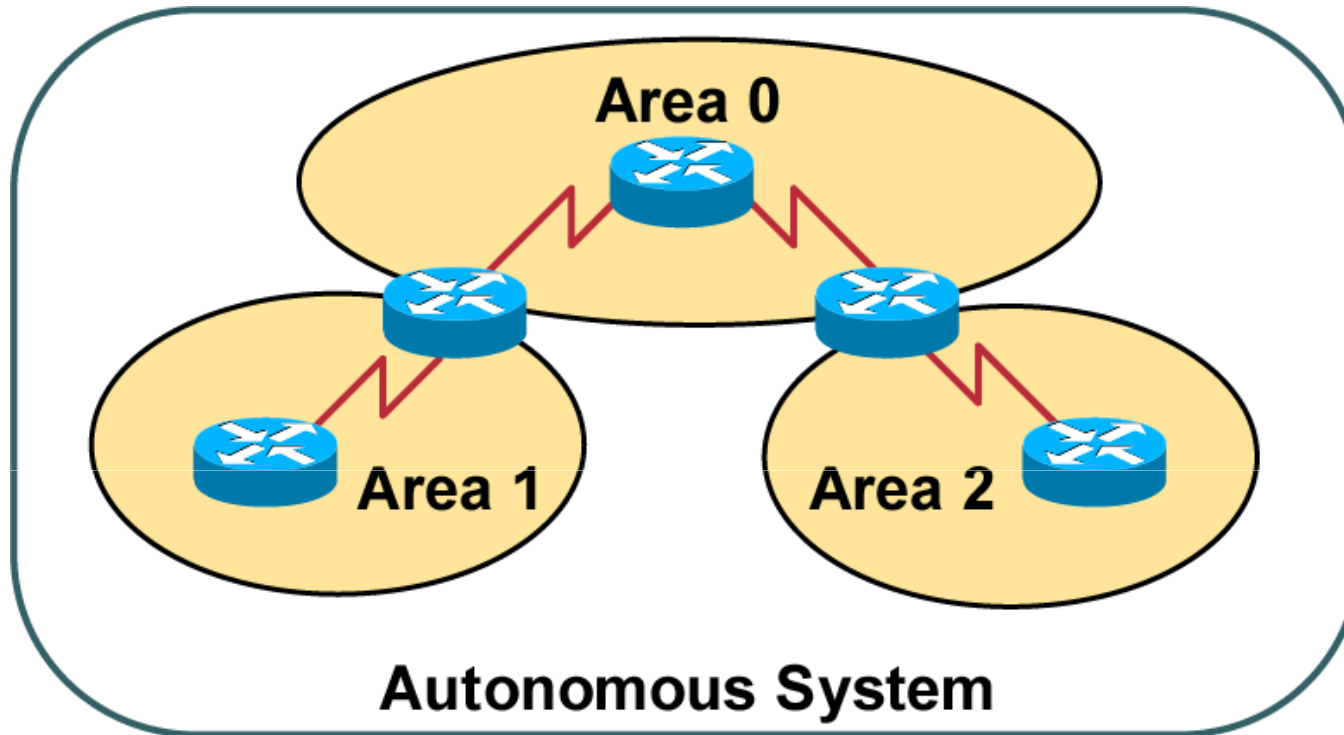


## OSPF as a Link-State Protocol

---

- **OSPF**는 **Routing Table**을 **Update**하지 않고 **Link-state** 정보를 광고한다.
- **OSPF**는 동일한 **Area**내의 다른 모든 **Router**에게 **LSA (Link State Advertisement)**의 전송을 요청한다.
- **OSPF**는 **Link-state Database**에 상태가 변경되면 즉시 **LSAs**를 전달하여 다른 **Router**에게 알린다.
- **OSPF**는 목적지 경로에 최적 정보를 **SFP** 알고리즘이 계산한다.
  - Link = router interface
  - State = Interface와 Neighbor 연결관계를 설명한다.

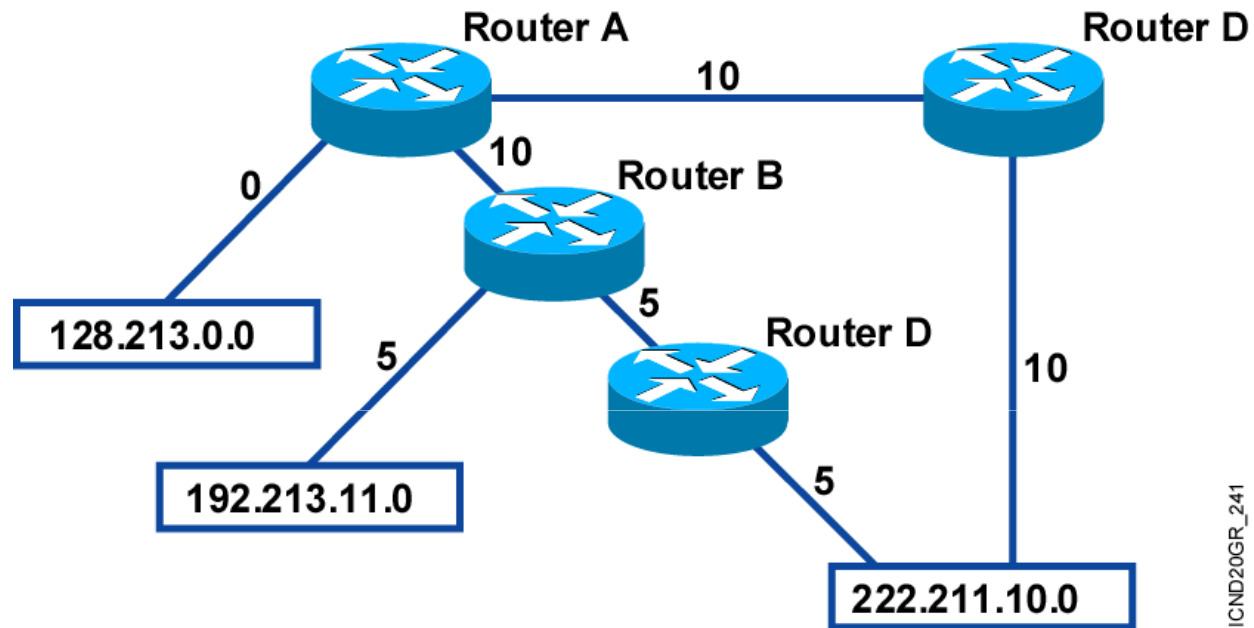
## OSPF Hierarchical Routing



- **SPF** 계산의 빈도가 줄어든다.
- **Routing Table**이 더 작다.
- **Link-state Update** 과부하가 줄어든다.



# Shortest Path First Algorithm



ICND20GR\_241

- **SPF**알고리즘에서는 **Tree**의 **Root**에 각 **Router**를 두고 수신지에 도달하는데 필요한 누적 **Cost**를 기반으로 각 **Node**로 가는 최단 경로를 계산한다.
- **Cost** =  $10^8 / \text{bandwidth (bps)}$



## Configuring Single Area OSPF

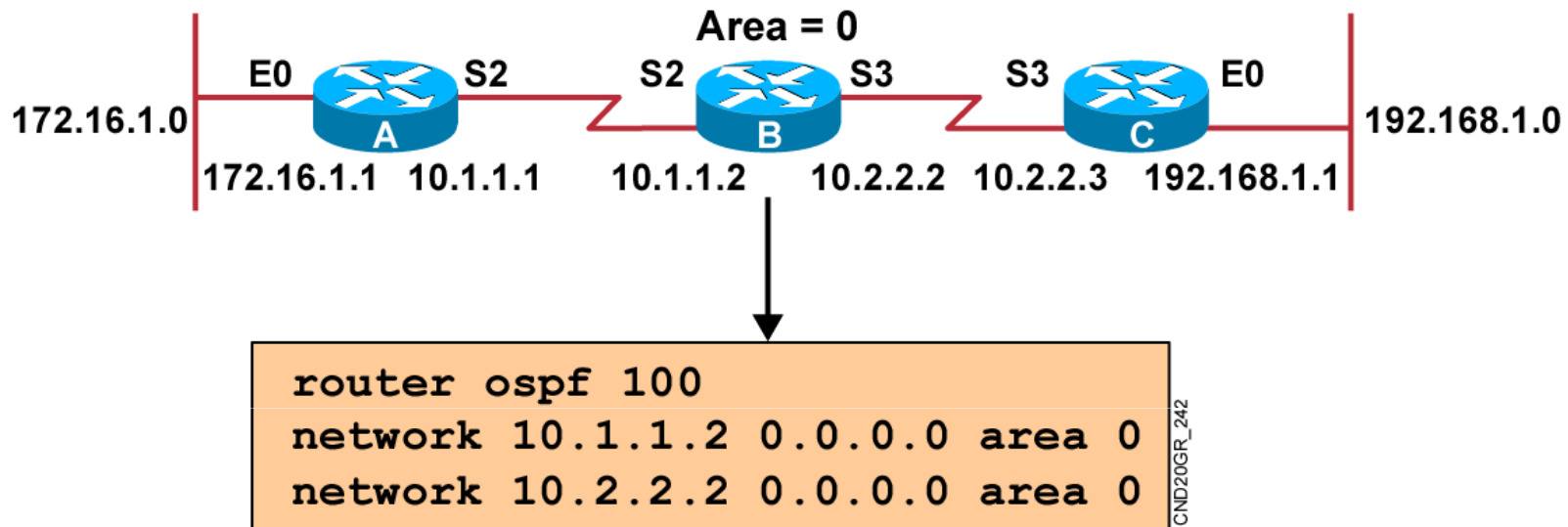
---

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

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

- **Router ospf** 명령을 사용하여 **OSPF Routing Process**를 시작하고, **Network** 명령을 사용하여 **Address**들을 연결시킨다.

# OSPF Configuration Example



- “router ospf 100”은 OSPF Routing Process ID 100으로 활성화한다.
- “network 10.1.1.2 0.0.0.0 area 0” 은 Interface Serial 2에서 OSPF Process를 활성화 하고, ID 100의 OSPF Process를 사용하여 Interface Serial 2와 연계된 Network, Sub network, Mask를 광고한다.
- “network 10.2.2.2 0.0.0.0 area 0”은 Interface Serial 3에서 OSPF Process를 활성화하고, ID 100의 OSPF Process를 사용하여 Interface Serial 3과 연계된 Network, Sub network, Mask를 광고한다.



# Verifying the OSPF Configuration

```
Router#show ip protocols
```

- 전체 **Router**에 대한 **Timer, Filter, Metric, Network**등에 매개 변수들을 표시.

```
Router#show ip route
```

- **Router**에 알려진 경로와 경로가 학습된 방법을 표시

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

- **Area-ID** 와 **Neighbor** 인접성을 표시

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

- **Interface**별로 **OSPF Neighbor** 정보를 표시



# OSPF debug commands

---

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

```
OSPF:hello with invalid timers on interface Ethernet0  
hello interval received 10 configured 10  
net mask received 255.255.255.0 configured 255.255.255.0  
dead interval received 40 configured 30  
Router# debug ip ospf packet
```

```
OSPF: rcv. v:2 t:1 l:48 rid:200.0.0.117  
aid:0.0.0.0 chk:6AB2 aut:0 auk:
```

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

```
OSPF: rcv. v:2 t:1 l:48 rid:200.0.0.116  
aid:0.0.0.0 chk:0 aut:2 keyid:1 seq:0x0
```



## OSPF LAB

---

# Enabling OSPF LAB