

Routing Protocol

- Router의 주 목적 중 하나는 **Routing**

- * **Routing** : **Packet**을 수신했을때 **Best Path**(최적 경로)를 찾아서 어느 경로로 전송할지 결정하는 것

- Router는 **Packet**을 **Forwarding**할 때 **Routing table**을 확인하고 전송.

- **Routing table**에 올라온 경로가 **Best Path**이다.

- 경로가 여러 개일 경우 **Best Path**만을 **Routing table**에 올리고 나머지 경로들은 데이터 베이스에 저장한다.

- **Switch**는 목적지를 모르는 **Frame**을 **Flooding**하지만 **Router**는 목적지를 모르는 **Packet**을 **Drop**시킨다.

즉, **Routing table**에 목적지의 정보가 없는 **Packet**은 폐기한다.

Routing Protocol

Routing table

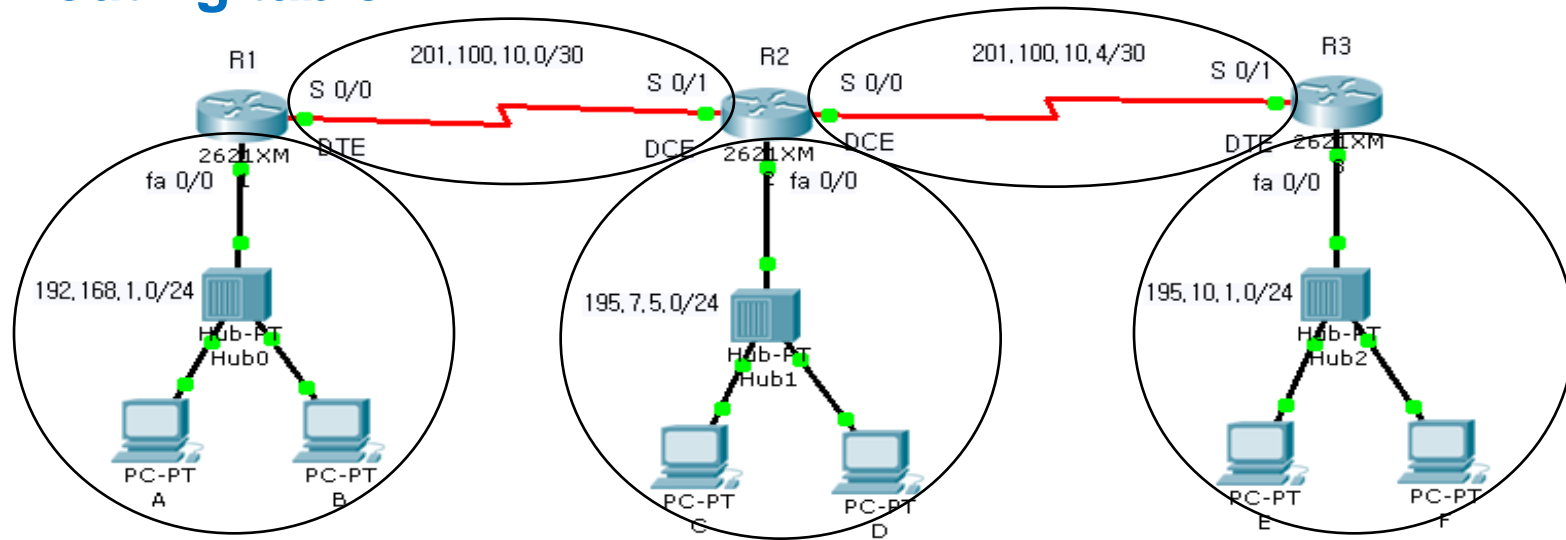
- privileged mode에서 '**show ip route**' 명령어로 확인

```
Router# show ip route
```

- 목적지 네트워크와 해당 네트워크로 가기 위해서 어느 경로 (Interface)로 나가야 하는지의 정보를 가지고 있다.
- Router가 Packet을 목적지로 보내기 위해서는 이렇게 Routing table을 참조한다.
- Best path만을 Routing table에 올린다.
- 즉, Routing table이란 어떤 목적지로 가기 위해서 어떤 경로로 가야 하는지 알 수 있는 네트워크 지도.

Routing Protocol

- Routing table -



```
R1#show ip route
```

```
Codes: C - connected, S - static, I - IGRP, 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, E - EGP
       i - IS-IS, 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    192.168.1.0/24 is directly connected, FastEthernet0/0
    201.100.10.0/30 is subnetted, 1 subnets
C      201.100.10.0 is directly connected, Serial0/0
```

```
R1#
```

Routing Protocol

Routing Protocol

- **Routing Protocol**이란?

→ 목적지 네트워크로 가는 경로를 알아내기 위해 사용하는 **Protocol**.

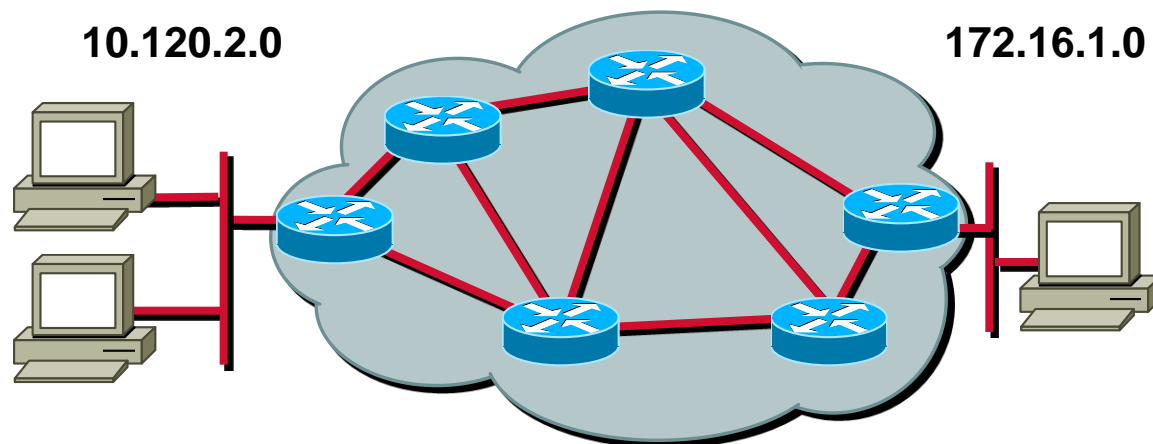
- **Router**는 기본적으로 자신과 연결된 네트워크의 정보만을 **Routing table**에 가지고 있다.

- 때문에 **Routing protocol**을 사용해서 직접 연결되지 않은 네트워크의 정보를 **Routing table**에 추가시킨다.

→ 즉, **Routing Protocol**이 설정되지 않으면 자신과 직접 연결된(**connected**) 주소만 **Routing table**에 보인다.

- 각 **Router**는 서로의 주소를 알아야 통신이 가능.
(한쪽의 **Router**만 상대방 **Router**의 주소를 알고 있을 경우 통신이 불가능)

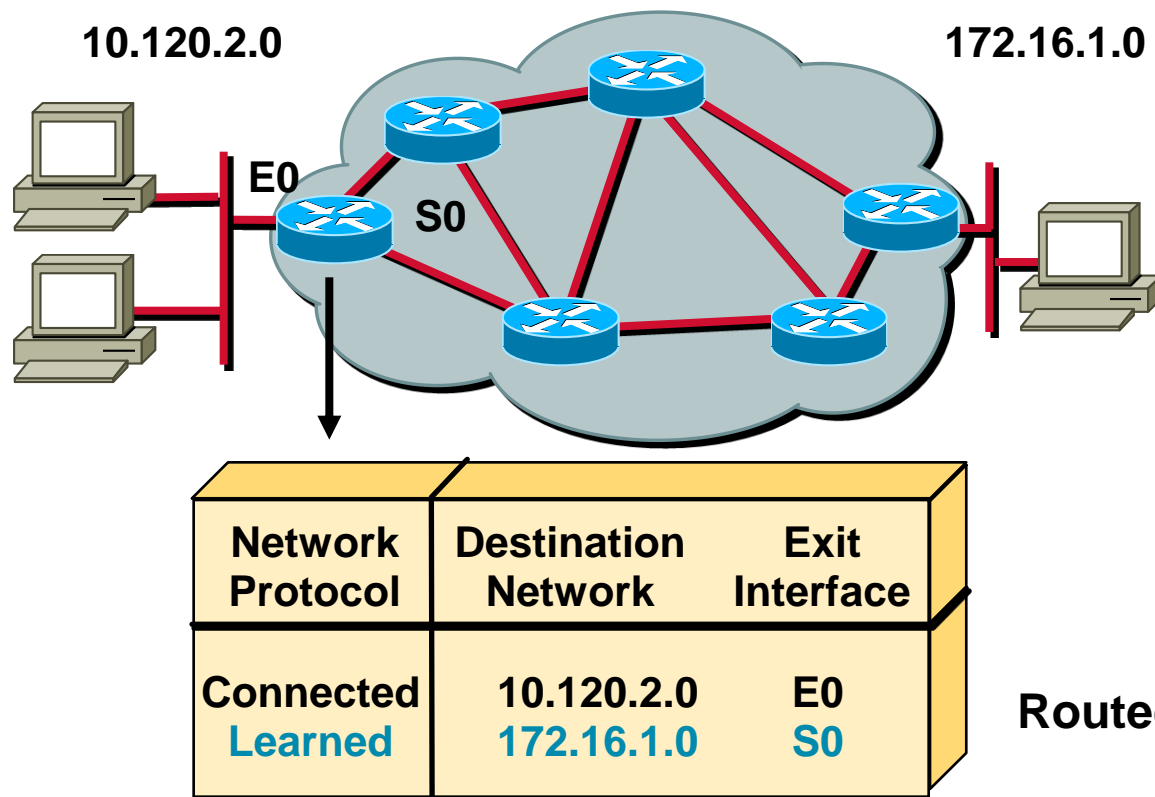
Routing Protocol



To route a router need to know:

- Destination addresses
- Sources it can learn from
- Possible routes
- Best route
- Maintain and verify routing information

Routing Protocol



- Routers must learn destinations that are not directly connected

Routing Protocol

- Routing Protocol의 종류 -

1) Static Routing Protocol

- 관리자가 직접적으로 목적지 네트워크의 정보를 입력하는 프로토콜

2) Dynamic Routing Protocol

- **Router**와 **Router**가 자동으로 서로의 네트워크 정보를 주고 받으며 네트워크 정보를 업데이트하는 프로토콜

ex) RIP, EIGRP, OSPF, IS-IS 등

Routing Protocol

1) Static Routing Protocol (정적 라우팅 프로토콜)

- 관리자가 직접 목적지 네트워크의 정보를 입력하는 **Protocol**.
- **Router**에 관리자(사람)가 일일이 수동으로 경로를 입력.
(즉, 사람이 목적지에 대한 **Best Path**를 찾아서 직접 입력한다.)
- **Static**은 정적인 프로토콜로 한번 정해놓으면 무조건 그 경로로 전송한다.
- **Router**는 단지 그 경로를 따라 **packet**을 전송하면 되기 때문에 **Router** 자체에 별 부담이 없다.

Routing Protocol

1) Static Routing Protocol (정적 라우팅 프로토콜)

- 장점

- CPU가 **Best Path**를 찾기 위한 계산을 하지 않기 때문에 **Routing** 속도도 빨라지고 메모리도 적게 사용하기 때문에 **Router** 성능이 좋아진다.
(Router의 **CPU**와 **MEMORY**를 적게 사용한다.)
- **Dynamic Routing Protocol**처럼 **Routing table**을 교환할 필요가 없어 대역폭을 절약할 수 있다.
- 자신의 정보를 외부로 전송하지 않아 **Dynamic Routing Protocol** 보다 보안에 좋다.

Routing Protocol

1) Static Routing Protocol (정적 라우팅 프로토콜)

- 단점

- 관리자가 직접 네트워크 경로를 일일이 설정해야 하니까 불편하다.
- 입력한 경로에 이상이 생겨도 **Packet**을 계속해서 그 경로로 전송한다.
- 정적, 즉 수동이기 때문에 관리자가 경로 이상을 인지하고 경로 설정을 바꾸기 전까지 계속해서 이상 경로로 전송.
- **Dynamic Routing Protocol**은 경로에 문제가 생길 경우 그때마다 자동으로 경로를 수정
- 대규모 네트워크에서는 사용하기 힘들다.
(**Network** 변화를 감지하기 어렵고 확장성이 낮다.)

Routing Protocol

2) Dynamic Routing Protocol (동적 라우팅 프로토콜)

- 관리자가 직접 경로를 입력할 필요 없이 설정된 **Routing Protocol** 알고리즘이 **Best path**를 찾아서 **Routing table**에 올린다.
- 같은 **Routing Protocol**이 설정된 **Router**가 서로의 네트워크 정보를 교환하면서 **Update**한다

- Dynamic Routing Protocol의 종류

(1) **Distance Vector** 라우팅 프로토콜

→ 순수 물리적으로 **Best path**를 결정 (ex. RIP, IGRP)

(2) **Link-State** 라우팅 프로토콜

→ 링크의 상태로 **Best path**를 결정 (ex. OSPF, IS-IS)

* 그 외에도 **Hybrid** 라우팅 프로토콜 (ex. **EIGRP**)도 있다.

Routing Protocol

2) Dynamic Routing Protocol (동적 라우팅 프로토콜)

- 장점

- 관리자 일일이 경로를 입력할 필요가 없다. 각 **Router**가 설정된 **Routing Protocol**의 알고리즘에 따라 **Best path**를 결정한다.
- 선택된 경로에 문제가 발생하거나 변화가 생기면 스스로 새로운 경로를 찾아서 전송한다.

- 단점

- **Router**에 부담을 준다. (**CPU**와 **MEMORY**를 많이 사용한다.)

Routing Protocol

2) Dynamic Routing Protocol (동적 라우팅 프로토콜)

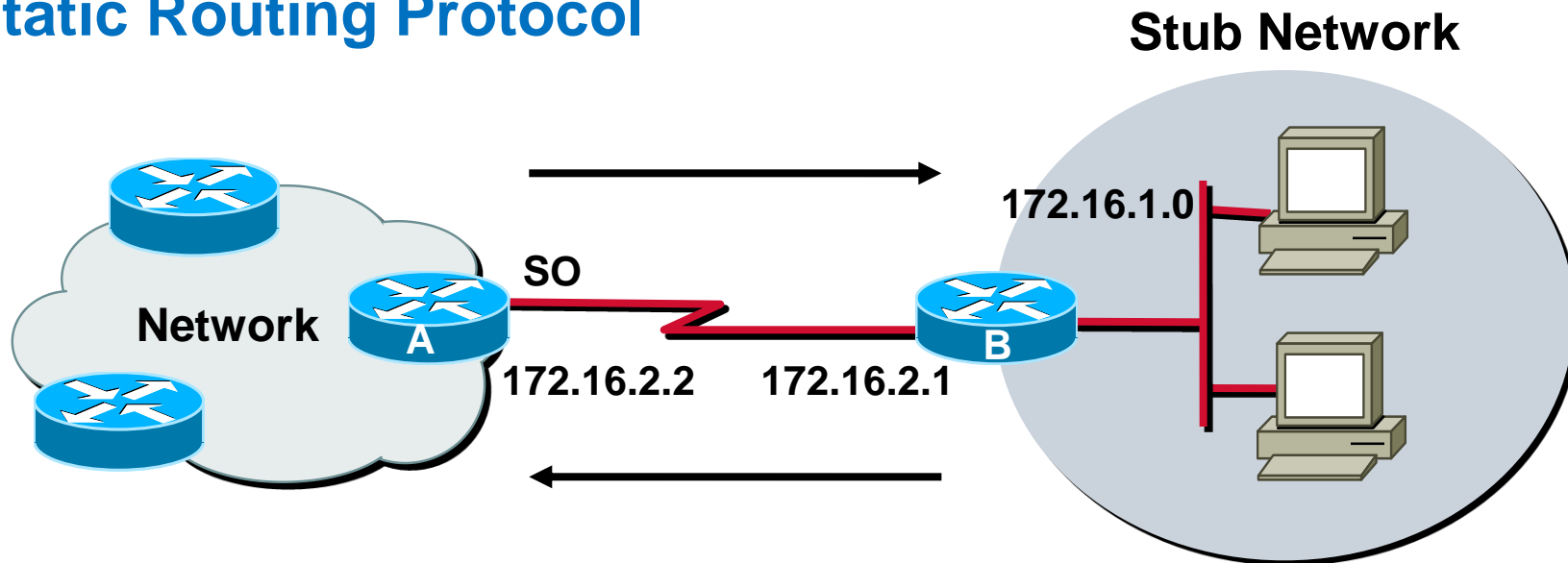
- 각 **Routing protocol**들은 **Best path**를 선택하는 기준이 다르기 때문에 서로 다른 **Routing protocol**은 서로 정보를 교환하지 않는다.

ex) **RIP**은 **Hop-count**, **OSPF**는 **Bandwidth**가 **Best path**를 선택하는 기준이다.

- 목적지까지의 경로가 여러 개 존재할 경우 **Best path**만 **Routing table**에 올라오고 나머지는 데이터 베이스에 저장.
- **Best path**에 문제가 생기면 다음 순의 경로가 **Routing table**에 올라온다.

Routing Protocol

- Static Routing Protocol



Configure unidirectional static routes to and from a stub network to allow communications to occur.

Routing Protocol

- Static Routing Protocol

```
Router(config)#ip route network [subnet mask] {next-Hop address | exit interface} [distance]
```

1 2 3 4 5

1 : 목적지 네트워크 주소

2 : 목적지 네트워크의 서브넷 마스크

3 : 목적지 네트워크로 가기 위한 **next-hop address**

(즉 설정하는 **Router**와 연결된 상대방 **Router** 인터페이스의 주소 (➔ **next-hop 주소**))

'목적지 네트워크로 가려면 **next-hop address**로 가라~'는 의미.

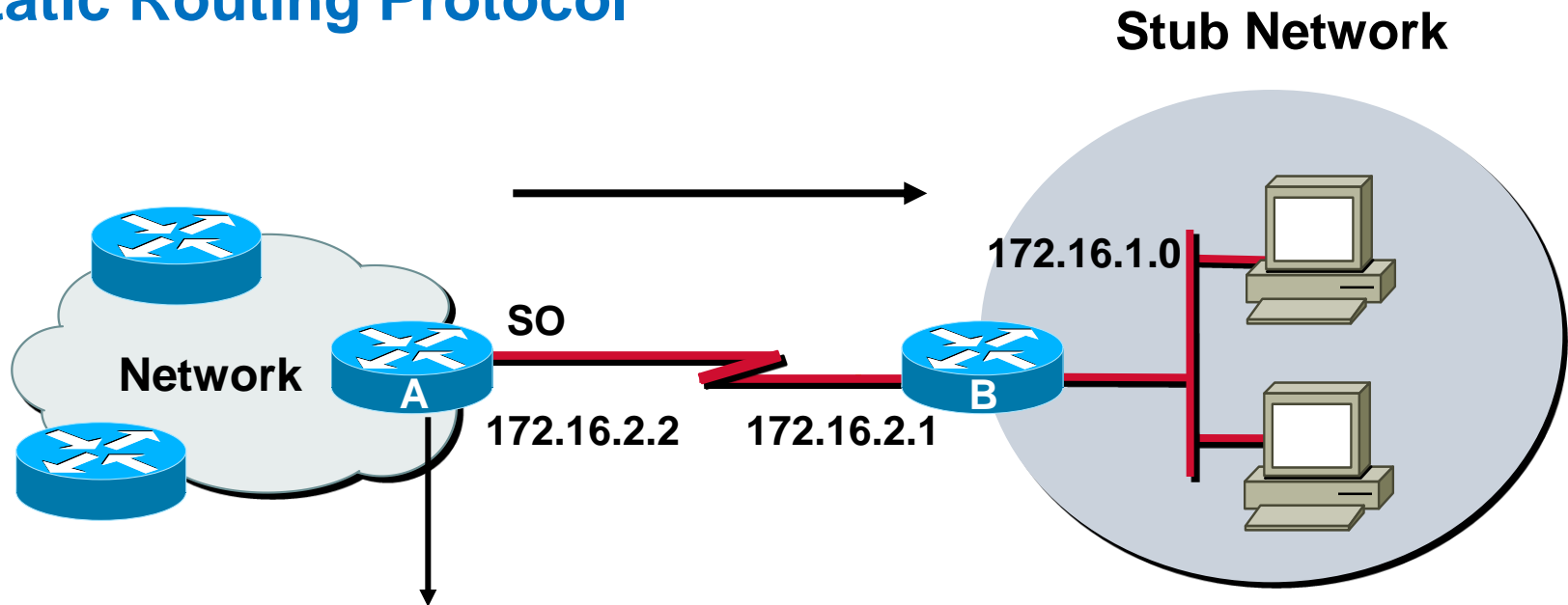
4 : 3번의 **address** 대신 **interface**를 넣을 수도 있다.

➔ 주의점은 **address**는 상대방 **Router**의 주소이지만 **interface**는 자기 **Router**에서 데이터가 나가는 출구 **interface**를 넣어야 한다.

5 : **distance**는 메트릭 값. **distance** 값이 커지면 가치가 떨어진다. (디폴트 값은 1)

Routing Protocol

- Static Routing Protocol

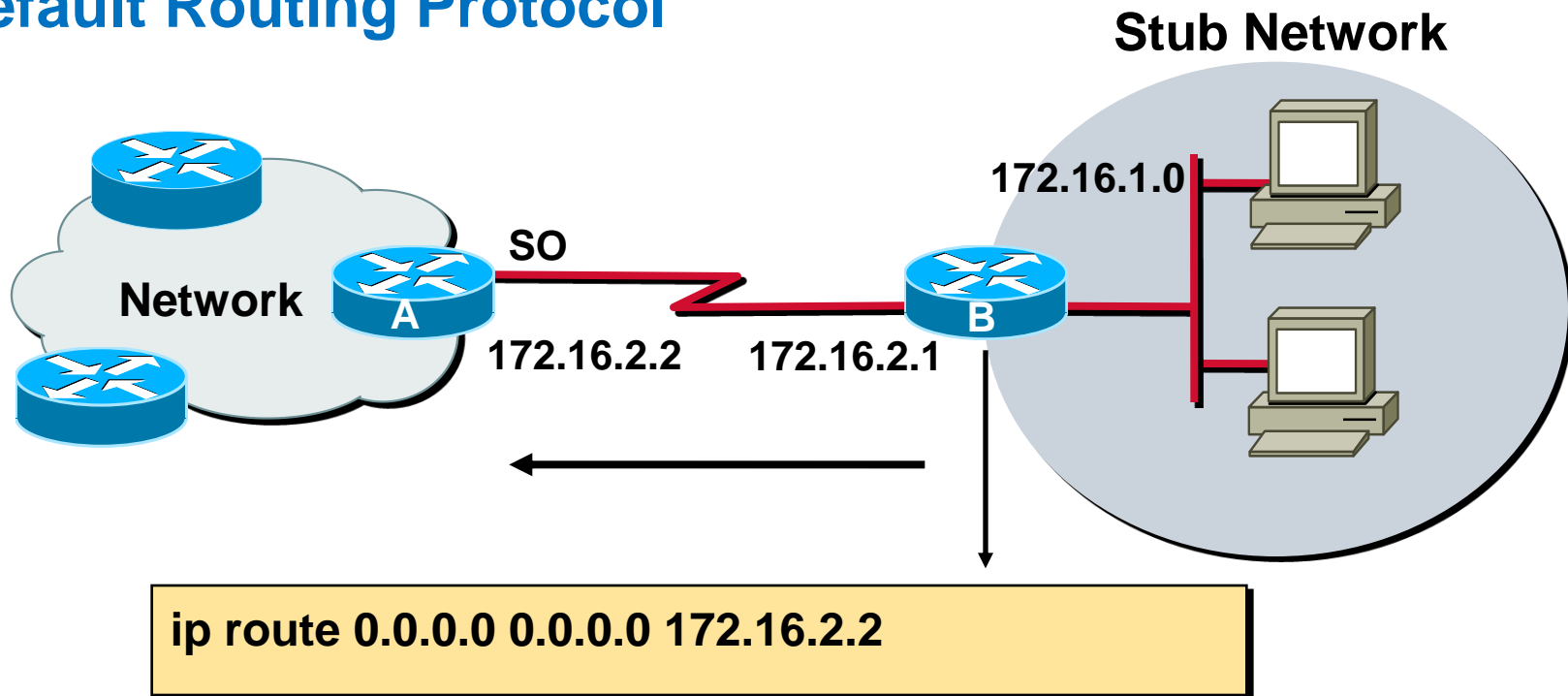


```
ip route 172.16.1.0 255.255.255.0 172.16.2.1
```

This is a unidirectional route. You must have a route configured in the opposite direction.

Routing Protocol

- Default Routing Protocol



This route allows the stub network to reach all known networks beyond router A.