

MPLS VPN

1. MPLS 란?

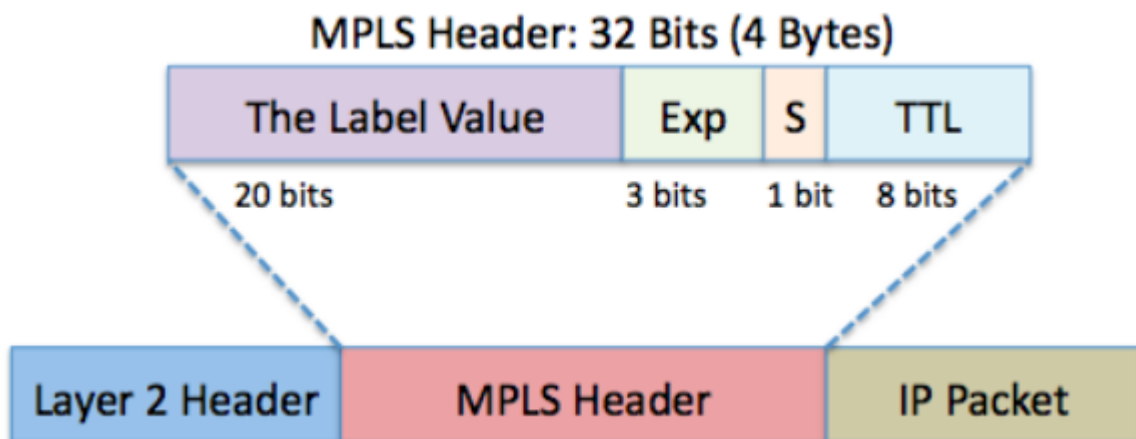
○ IPv4/IPv6 패킷 또는 L2 프레임에 라벨을 첨부하여 전송하는 기술

○ MPLS 사용 이유

- MPLS VPN 구성
 - ISP 입장: 대규모의 VPN을 쉽게 구성
 - 고객 입장: VPN 내부와 외부 트래픽이 완전히 분리되어 비교적 안전한 망을 구성
 - MPLS VPN 자체는 인증이나 암호화 기능을 제공하지 않는다. 고도의 보안을 위해서는 MPLS VPN과 더불어 IPSec VPN 등을 함께 사용하는 것이 좋다.
- Traffic Engineering
 - 특정 트래픽에 대해 대역폭을 보장하고, IGP가 사용하지 않는 링크까지 사용할 수 있어 네트워크 자원을 최대한 보장

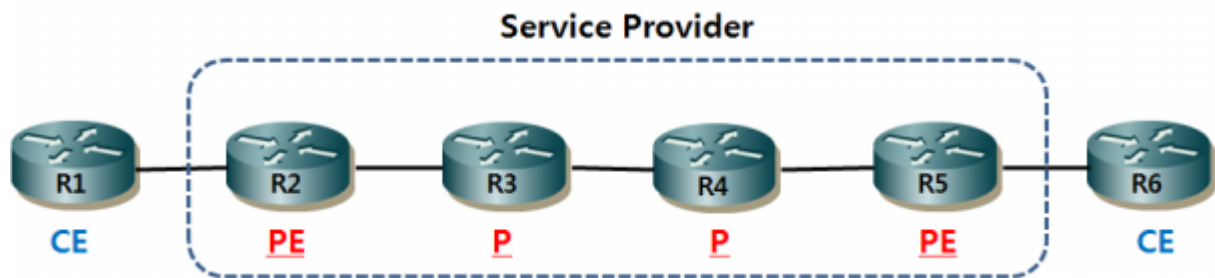
2. MPLS 헤더의 구조

[그림 MPLS 헤더]



- **Label**
 - MPLS에서 사용하는 라벨을 표시하는 필드
 - 라벨값은 0 - 1,048,757 사이의 값
- **EXP**
 - Experimental 비트는 QoS 값을 표시할 때 사용
 - 0 - 7 사이의 값
- **BoS**
 - Bottom of Stack 비트 값이 1이면 마지막 라벨임을 나타낸다.
 - 이 값이 0이면 현재의 라벨 다음에 또 다른 라벨이 있음을 의미
- **TTL**
 - Time to live. 패킷의 루프를 방지하기 위한 필드

3. MPLS VPN 라우터의 종류



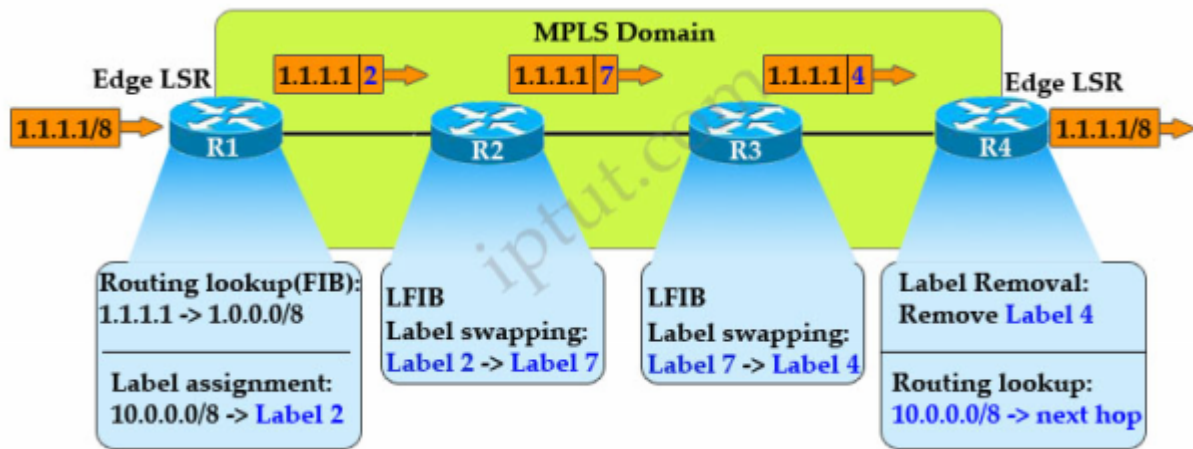
- MPLS VPN 용어
 - **PE(Provider Edge)** 라우터
 - 고객 라우터와 직접 연결되는 MPLS 라우터
 - **P(Provider)** 라우터
 - 고객 라우터와 직접 연결되지 않는 MPLS 라우터
 - **CE(Customer Edge)** 라우터
 - MPLS 망과 직접 연결되는 고객사의 라우터
 - MPLS 관련 설정이 없다.
- MPLS 용어
 - **LER(Label Edge Router)**
 - 사용자 및 가입자 네트워크가 연결된 라우터
 - **LSR(Label Swap Router)**
 - MPLS Provider 네트워크의 백본에 연결되어 있는 라우터
 - **LSP(Label Switch Path)**
 - MPLS를 구성한 라우터와 라우터 상호간에 생기는 경로

4. FEC(Forwarding Equivalence Class)

- **FEC**: 동일한 MPLS 경로를 이용하여 전송되는 패킷의 그룹
 - 동일 FEC에 속하는 패킷들은 동일한 라벨값이 부여된다.
 - Ingress(입력) LSR
 - 패킷을 처음 수신한 MPLS 라우터
- 동일 FEC 소속
 - 목적지 네트워크가 같은 패킷
 - 동일 그룹에 소속된 멀티캐스트 패킷
 - BGP 넥스트 홉 주소가 동일한 패킷
- **Binding**: FEC 별로 라벨을 붙여주는 것
 - 로컬 바인딩: 라우팅 테이블에 있는 각각의 IGP 네트워크 별로 하나씩의 라벨을 부여하는 것
 - 리모트 바인딩: 인접 LSR로부터 광고받은 것
- **LIB(Label Information Base)**
 - 로컬바인딩과 리모트바인딩을 보관하는 데이터베이스 = 모든 경로
 - 확인: **sh mpls ldp bind (LIB)**
- **LFIB(Label Forwarding Information Base)**
 - LIB와 라우팅테이블을 이용하여 라벨을 결정한 다음 저장되는 곳 = 최적 경로
 - 확인: **sh mpls forward (LFIB)**

5. MPLS 패킷 전송 과정

[그림 MPLS 패킷 전송 과정]



- **Push = Imposition**
 - 처음 라벨을 부여하는 동작
- **Swap**
 - 라벨을 변경하는 동작
- **POP = Disposition**
 - 최종 LSR 직전의 라우터에서 제거(PHP: penultimate hop popping)
 - 최종 LSR이 MPLS 라벨과 라우팅 테이블을 동시에 참조하는 이중행위를 방지하기 위해서

6. MPLS VPN 서비스망 구축

○ 구축 순서

1. MPLS VPN 서비스망 **IGP** 설정
2. MPLS 라우터간 **LDP** 설정
3. PE 라우터간 **MBGP** 설정
4. **VRF** 구성
5. **PE-CE간 Routing** 설정

7. VRF(Virtual Routing Forwarding)

○ **VRF**

- 각 고객별 라우팅테이블, CEF 테이블 및 PE 라우터의 관련된 라우팅 프로토콜의 집합
- PE 라우터는 연결된 하나의 고객에 대해 하나의 VRF를 가진다.
 - RD, RT 포함

○ **RD(Route Distinguisher)**

- 서로 다른 VPN 고객을 구분해주는 식별자
- AS:NN 또는 IP주소:NN 형태
 - 10:100, 172.16.1.1:100
 - RD 형태 48비트 = 앞 16비트: 뒤 32비트

○ **RT(Route Target)**

- 특정 경로를 어느 VRF로 전달해야 할 지를 결정
- BGP의 확장 커뮤니티, 모든 VPN 경로에 표시
 - **export**: VPN망 방향 광고 = PE가 인접한 CE 네트워크의 RT 값을 지정하는 것

- **import:** 고객망 방향 광고 = 다른 VRF에서 수신 네트워크 중 특정 RT를 가진 것을 인접 CE에 광고해주는 것

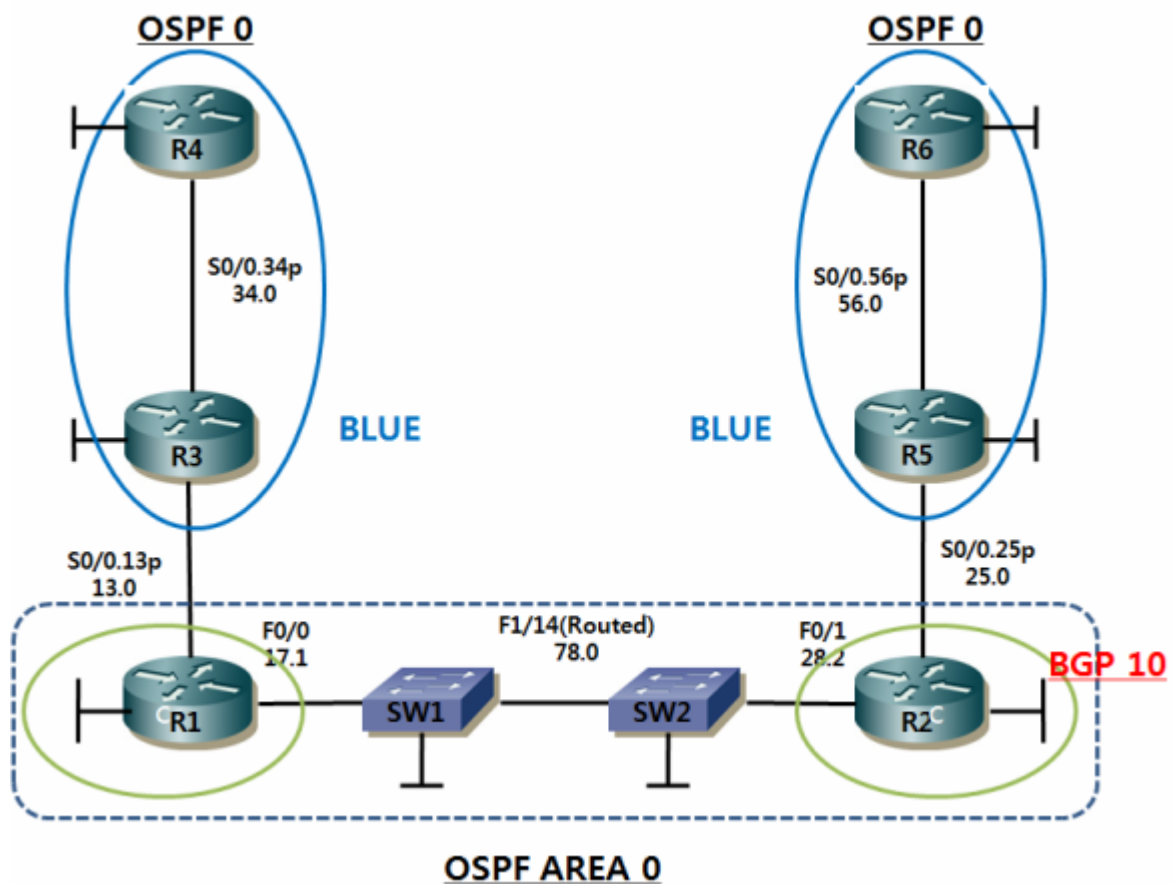
○ VRF 설정 예

```
ip vrf 이름
rd ...
route-target export ...
route-target import ...
```

```
ip vrf BLUE
rd 10:100
route-target export 10:100
route-target import 10:100
```

8. MPLS 망 구축 구체적 순서

[그림 MPLS VPN]



8-1. IGP 설정: 망내 모든 라우터

```
R(config)#router ospf 1
...
```

- 확인: `show ip route`

8-2. LDP 설정: 망내 모든 라우터

```
R(config)# mpls label protocol ldp
R(config)# mpls ldp route-id lo0
```

```
R(config)# int ..
R(config-if)# mpls ip
```

- 망내 라우터 연결 인터페이스만

- 확인
 - **show mpls int**
 - **show mpls ldp neighbor**
 - **show mpls ldp discovery**

8-3. MBGP 설정: PE만

```
R(config)# router bgp 10
R(config-router)# b router-id ...
R(config-router)# neighbor ... remote-as 10
R(config-router)# neighbor ... up lo0
R(config-router)# address-family vpn4
R(config-router-af)# nei ... activate
R(config-router-af)# nei ... send-community extended
```

- 확인: **show ip bgp summary**

8-4. VRF 구성: PE만

```
R(config)# ip vrf BLUE
R(config-vrf)# rd 10:100
R(config-vrf)# route-target both 10:100
```

```
R(config)# int ...
R(config-if)# ip vrf forwarding BLUE
```

- CE쪽 인터페이스만
- 주소를 재입력해야만 한다.

- 확인
 - **show ip vrf**
 - **show ip vrf int**

9. PE-CE간 라우팅: PE만

9-1. OSPF

```
R(config)# router ospf 번호 vrf VRF이름
R(config-router)# network PE-CE간 0.0.0.0 area 0
R(config-router)# redistribute bgp 10 subnets

R(config)#router bgp 10
R(config-router)# address-family ipv4 vrf VRF이름
R(config-router)# redistribute ospf 10 match internal external 1 external 2
```

- 확인
 - **sh ip bgp vpn rd 10:100**
 - **sh ip route vrf BLUE**
 - **sh ip bgp vpn all**
 - **sh ip bgp vpn vrf BLUE sum**
 - **ping vrf BLUE 1.1.13.3**

9-2. Static

```
R(config)# ip route vrf BLUE 3.3.3.3 255.255.255.255 1.1.13.3

R(config)#router bgp 10
R(config-router)# address-family ipv4 vrf VRF이름
```

```
R(config-router-af)# redistribute static
R(config-router-af)# redistribute connected
```

9-3. RIPv2

```
R(config)# router rip
R(config-router)# address-family ipv4 vrf ...
R(config-router)# net ...
R(config-router)# redistribute bgp 10 metric 2
```

```
R(config)# router bgp 10
R(config-router)# address-family ipv4 vrf BLUE
R(config-router)# redistribute rip
```

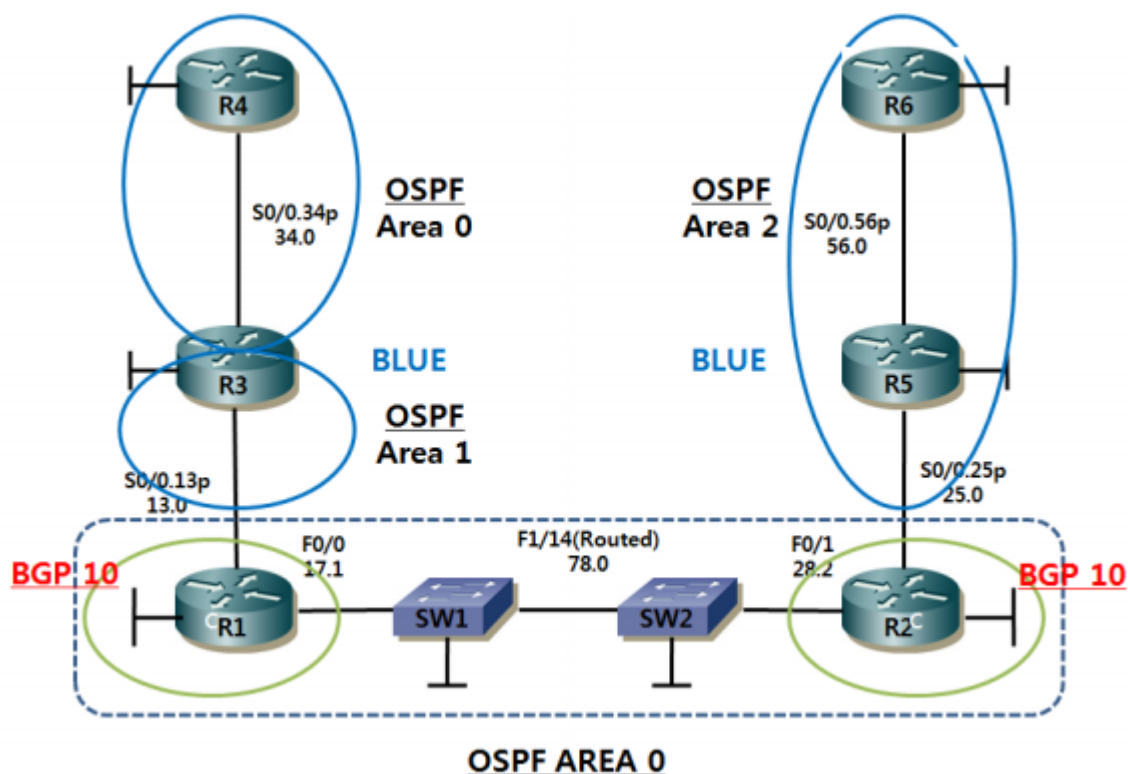
9-4. EIGRP

```
R(config)# router eigrp 1
R(config-router)# address-family ipv4 vrf ...
R(config-router)# autonomous-system 1
R(config-router)# net ...
R(config-router)# redistribute bgp 10 metric 1 1 1 1 1
```

```
R(config)# router bgp 10
R(config-router)# address-family ipv4 vrf BLUE
R(config-router)# redistribute eigrp 1
```

9-5. OSPF 슈퍼 백본 에어리어

[그림 OSPF 슈퍼 백본 에어리어]



- MPLS VPN 망으로 연결된 원격지의 네트워크는 다른 에어리어 (IA)에 있는 것으로 인식된다.
- MPLS VPN은 OSPF에 대해 추가적인 최상위 에어리어인 Super Backbone Area로 동작하기 때문.

R1# show ip ospf

...

Connected to MPLS VPN Superbackbone, VRF BLUE

It is an area border and autonomous system boundary router

9-6. BGP

R(config)# router bgp 10

R(config-router)# **address-family ipv4 vrf BLUE**

R(config-router)# nei ... remote-as 34

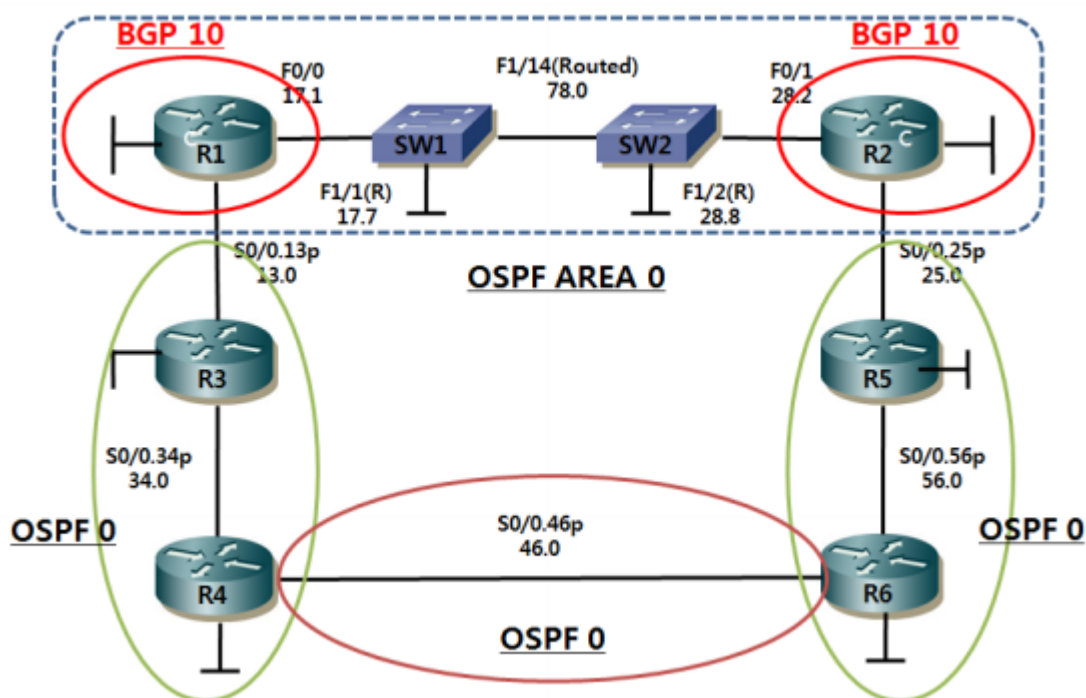
R(config-router)# nei ... activate

R(config-router)# **nei ... as-override**

- * AS번호가 동일할 때

9-7. 샴 링크

[그림 Sham-link]



- MPLS VPN 망을 통해 OSPF 경로를 광고할 때 LSA Type 3이 아닌 Type 1로 광고하여 동일 에어리어에 소속된 경로로 보이게 하는 것
- MPLS VPN에서 PE 라우터간에 샴링크를 만들기 전 라우터에 /32 주소를 만든다.
 - VRF에 소속되어야 한다.
 - OSPF에 포함시키지 말아야 한다.
 - BGP에 광고되어야 한다.

(1) 인터페이스 생성(VRF 소속)

R(config)# **int lo100**

R(config-if)# **ip vrf for GREEN**

R(config-if)# **ip add 1.1.100.1 255.255.255.255**

(2) BGP address-family ipv4 vrf ... 광고

R(config)#router bgp 10

R(config-router)# **address-family ipv4 vrf GREEN**

R(config-router)# **network 1.1.100.1 mask 255.255.255.255**

(3) sham-link 설정

R(config)# router ospf 100 vrf GREEN

R(config-router)# **area 0 sham-link** 1.1.100.1 1.1.100.2 (cost 1)

- 확인: **sh ip ospf sham-link**