Access Control Lists

Access Control List

- 특정 트래픽의 접근을 허용할지 차단할지 결정하는 리스트 **(Filtering)**

- 보안을 위해서 많이 사용.

- **L3**장비인 **Router**에서 설정하지만 **Application Layer**부분도 관리하기 때문에 **Network Layer**까지라고 단정할 수 없다.

- 하지만 **Application Layer**까지 완벽히 막을 수 없기 때문에 **Firewall** 등의 전문적인 보안 장비를 사용.

- **ACL**은 크게 **Numbered**와 **Named** 두 종류가 있다.
  그리고 다시 **Standard (1~99)**와 **Extended (100~199)**로 구분할 수 있다.

  1) **standard Access list** ➔ **source address**만 참조해서 **filtering** 여부를 결정.

  2) **extended Access list** ➔ **source address**외에도 **destination address, protocol, Port** 번호 등 좀더 자세한 정보를 참조해서 **filtering** 여부를 결정한다.

Access Control Lists

1) Standard ACL (1-99)

- Standard ACL의 경우는 출발지 주소(source address)를 보고 permit, deny 여부를 결정.

- packet의 source address와 ACL에 정의된 source address가 일치하면 ACL의 내용을 수행한다. (permit or deny)

- permit이면 packet을 정해진 경로로 전송하고 deny면 packet의 흐름을 차단

- standard ACL의 사용 list-number는 1-99까지 사용한다.

ex) R1(config)# access-list 1 deny 125.101.1.0 0.0.0.255
R1(config)# access-list 1 permit any

Access Control Lists

1) Standard ACL 설정 (1)

```
R1(config)#access-list <list-number> {permit|deny} source [mask]
                            1            2          3      4
```

1 : **list-number**는 **1-99**까지의 번호를 사용. **(1-99**까지가 **standard ACL**의 번호이다.**)**

2 : 아래 **3**번 조건에 맞는 **packet**을 **permit**할지 **deny**할지 결정.

3 : 조건을 넣는다. **standard ACL**의 조건은 **source address**, 만약 **source address**를 넣지 않고 **any**라고 입력하면 특정한 하나의 출발지 주소가 아닌 모든 주소에 **2**번에서 정의한 수행 내용을 적용.

ex) **R1(config)# access-list 1 deny 125.101.1.0 0.0.0.255**
R1(config)# access-list 1 permit any

Access Control Lists

1) Standard ACL 설정 (2)

- Interface 적용 -

```
R1(config)#interface serial 0/0
R1(config-if)#ip access-group <access-list-number> {in | out}
                                        1              2
```

1 : 앞에서 정의한 **ACL**을 불러와서 **filtering** 내용을 인터페이스에 적용한다.

2 : **inbound**와 **outbound** 설정.

in은 라우터의 인터페이스로 **packet**이 들어오는 경우
out은 **packet**이 라우터의 인터페이스에서 나가는 경우

*** standard ACL**은 항상 **destination** 라우터 쪽에 설정되어야 한다. 중간 라우터에 설정하면 다른 라우터들까지 **ACL**의 영향을 받아서 정상적으로 패킷 전송이 이루어지지 않을 수 있다.

ex) **R1(config)#interface serial 0/0**
R1(config-if)#ip access-group 10 in

Access Control Lists

ACL의 동작방식

1) inbound 설정

- **packet**이 **Router** 내부로 들어올 때 **filtering** 여부를 결정

- **Router** 인터페이스로 **packet**이 들어올 경우 수신 인터페이스에 **ACL**이 설정되어 있는지 확인하고 설정이 되어있지 않으면 그냥 통과.

- 만약 **ACL**이 설정돼 있다면 들어온 **packet** 의 정보와 **ACL**에 설정 내용을 비교해서 통과 여부를 결정. **(조건과 일치하고 permit이면 통과, deny면 통과 X)**

2) outbound 설정

- **packet**이 **Router** 외부로 나갈 때 **filtering** 여부를 결정한다

- **Router** 인터페이스에서 **packet** 이 나갈 경우 인터페이스에 **ACL**이 설정되어 있는지 확인하고 설정이 되어있지 않으면 그냥 보낸다.

- 만약 **ACL**이 설정돼 있다면 나가는 **packet**의 정보와 **ACL**에 설정 내용을 비교해서 통과 여부를 결정. **(조건과 일치하고 permit이면 통과, deny면 통과 X)**

Access Control Lists

ACL 규칙

1) ACL은 윗줄부터 순서대로 수행. 때문에 **ACL**은 좁은 범위 설정이 먼저 되어야 한다.
만약 다음처럼 넓은 범위를 먼저 설정하게 되면 모든 **Packet**이 허용.
(Fitering 효과가 없다.)

```
R1(config)# access-list 1 permit any
R1(config)# access-list 1 deny 125.101.1.0 0.0.0.255
```

2) ACL의 마지막은 **deny any**가 생략되어 있다. 즉, 마지막에 **permit any**가 없을 경우 **ACL** 조건에 없는 모든 **address**는 **deny** 된다.

3) numbered ACL은 순서대로 입력되기 때문에 중간 삽입이나 중간 삭제가 불가능하다. **(중간에 List**가 틀려도 삽입, 수정 , 중간 삭제 불가능)

***** 예외 **named ACL**의 경우는 중간 삭제 및 추가 삽입이 가능한다.

즉, 새로 추가하는 모든 조건은 마지막에 더해진다. **(**순서가 하향식 계산이다.**)**

Access Control Lists

2) Extended ACL (100-199)

- **standard ACL**은 **source address**만 조건으로 보고 **filtering**을 수행한다. 하지만 **extended ACL**은 출발지와 목적지 주소**(destination address)** 모두를 조건으로 보고 제어한다.

- 또한 **standard ACL**은 **TCP/IP**에 대해 제어만을 하지만 **extended ACL**은 **ip, tcp, udp, icmp** 등의 상세 프로토콜을 선택해서 설정할 수 있다.

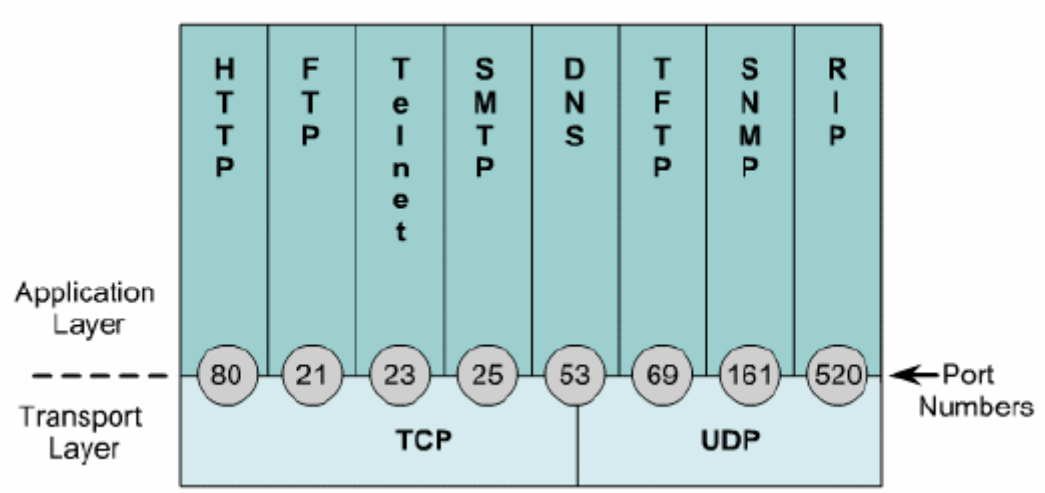
- **extended ACL**의 사용 **list-number**는 **100-199**까지 사용한다.

ex) **R1(config)#access-list 101 deny ip 192.100.51.0 0.0.0.255 210.150.6.0 0.0.0.255**

R2(config)#access-list 110 deny tcp 200.101.52.0 0.0.0.255 129.29.31.0 0.0.0.255 eq 80

Access Control Lists

2) Extended ACL (100-199)



*** Well Known Port (지정포트)**

1) TCP : FTP(20, 21), Telnet(23), SMTP(25), HTTP(80), HTTPs(443)

2) UDP : DNS(53), TFTP(69), DHCP(67, 68)

Access Control Lists

2) Extended ACL (100-199)

```
R1(config)#access-list <list-number> {permit|deny} <protocol>
                                1            2          3
source [mask] destination [mask] [operator port]
      4                5                6
```

1 : list-number는 100-199까지의 번호를 사용한다.
(100-199까지가 extended ACL의 번호이다.)

2 : 조건에 맞는 트래픽을 permit할지 deny할지 결정한다.

3 : filtering을 할 프로토콜을 정의한다. (TCP, UDP, IP 등)

4 : source address를 지정한다.

5 : destination address를 지정한다.

6 : 목적지 TCP/UDP 포트 이름 및 번호를 지정한다.

ex) R1(config)#access-list 101 deny ip 192.100.51.0 0.0.0.255 210.150.6.0 0.0.0.255

R2(config)#access-list 110 deny tcp 200.101.52.0 0.0.0.255 129.29.31.0 0.0.0.255 eq 80

Access Control Lists

2) Extended ACL (100-199)

- Interface 적용 -

```
R1(config)#interface serial 0/0
R1(config-if)#ip access-group <access-list-number> {in | out}
                                        1              2
```

1 : 앞에서 정의한 **ACL**을 불러와서 **filtering** 내용을 인터페이스에 적용한다.

2 : **inbound**와 **outbound** 설정.

in은 라우터의 인터페이스로 **packet**이 들어오는 경우
out은 **packet**이 라우터의 인터페이스에서 나가는 경우

*** standard ACL**은 항상 **destination** 라우터 쪽에 설정되어야 한다. 중간 라우터에 설정하면 다른 라우터들까지 **ACL**의 영향을 받아서 정상적으로 패킷 전송이 이루어지지 않을 수 있다.

ex) **R1(config)#interface serial 0/0**
R1(config-if)#ip access-group 101 in