Access Groups

- ACLs are applied at the interface level with the Access-Group command
- ACLs can be applied in the inbound or outbound direction
- You can have a maximum of one ACL per interface per direction
- You can have both an inbound and an outbound ACL on the same interface, but not 2 inbound or outbound ACLs
- An interface can have no ACL applied, an inbound ACL only, an outbound ACL only, or ACLs in both directions

Access-Group Configuration

```
R1(config)# interface GigabitEthernet0/1
R1(config-if)# ip access-group 100 out
R1(config-if)# ip access-group 101 in
```

Access-Group Configuration – show ip interface

```
R3#show ip interface f1/0 | include access list
  Outgoing access list is 100
  Inbound  access list is 101
```

('not set' if ACL is not applied)

Access Control Entry Order

- The ACL is read by the router from top to bottom
- As soon as a rule matches the packet, the permit or deny action is applied and the ACL is not processed any further
- The order of rules is important

Access Control Entry Order

- This will deny 10.10.10.10 but permit the rest of the 10.10.10.0/24 subnet

```
R1(config)# access-list 1 deny host 10.10.10.10
R1(config)# access-list 1 permit 10.10.10.0 0.0.0.255
```

- This will permit all of the 10.10.10.0/24 subnet including 10.10.10.10

```
R1(config)# access-list 1 permit 10.10.10.0 0.0.0.255
R1(config)# access-list 1 deny host 10.10.10.10
```

Injecting ACEs in an Existing ACL

- ACEs are automatically numbered in increments of 10

```
R1#sh access-lists 110
Extended IP access list 110
    10 deny tcp host 10.10.10.10 host 10.10.50.10 eq telnet
    20 permit tcp 10.10.10.0 0.0.0.255 host 10.10.50.10 eq telnet
    30 deny tcp host 10.10.20.10 host 10.10.50.10 eq telnet
    40 permit tcp 10.20.10.0 0.0.0.255 host 10.10.50.10 eq telnet
```

Injecting ACEs in an Existing ACL

- Support for injecting ACEs in an existing ACL started in Named ACLs but is also supported in Numbered ACLs now

```
R1(config)#ip access-list extended 110
R1(config-ext-nacl)#15 deny tcp host 10.10.10.11 host 10.10.50.10 eq telnet

R1#sh access-lists 110
Extended IP access list 110
    10 deny tcp host 10.10.10.10 host 10.10.50.10 eq telnet
    15 deny tcp host 10.10.10.11 host 10.10.50.10 eq telnet
    20 permit tcp 10.10.10.0 0.0.0.255 host 10.10.50.10 eq telnet
    30 deny tcp host 10.10.20.10 host 10.10.50.10 eq telnet
    40 permit tcp 10.20.10.0 0.0.0.255 host 10.10.50.10 eq telnet
```

Implicit Deny All

- There is an implicit ‘deny any any’ rule at the bottom of ACLs
- If an ACL is not applied to an interface, all traffic is allowed
- If an ACL is applied, all traffic is denied except what is explicitly allowed

- Traffic from 10.10.10.0/24 will be permitted, everything else is denied

```
R1(config)# access-list 1 permit 10.10.10.0 0.0.0.255
```

Explicit Deny All

- Many organisations include an explicit deny all at the end of ACLs to log illegal traffic

```
R1(config)# access-list 1 permit 10.10.10.0 0.0.0.255
R1(config)# access-list 1 deny any log
```

Explicit Permit All

- If an ACL is applied, all traffic is denied except what is explicitly allowed
- If you want to reverse this so that all traffic is permitted except what is explicitly denied, add a permit all statement to the end of the ACL

- Traffic from 10.10.10.0/24 is denied, everything else is permitted

```
R1(config)# access-list 1 deny 10.10.10.0 0.0.0.255
R1(config)# access-list 1 permit any
```

Traffic Sourced from Router

- ACL's applied to an interface do not apply to traffic which originates from the router itself
- The hosts in the 10.1.1.0/24 subnet cannot Telnet to R2
- An administrator can Telnet to R2 from the CLI on R1

```
R1(config)# access-list 100 deny tcp any any eq 23
R1(config)# interface f1/0
R1(config)# ip access-group 100 out
```