

CCNA Study Group Week 3

Preview

- ACLs
- Security Concepts
- Securing Cisco Devices
- Layer 2 security features
- VPNs
- Wireless security protocols
- Automation
- Controller-based networking vs. Traditional networking
- Configuration Management Tools
- Data serialization and scripting

Subnet Mask vs. Wildcard Mask

Subnet Mask are used when you are identifying *networks*.

Wildcard Mask are used when you are comparing *individual IPs*.

Configuring a route: subnet mask.

You are identifying a network.

Configuring an ACL: wildcard mask.

You are comparing which IPs should be allowed or not allowed to a rule.

Subnet Mask vs. Wildcard Mask Cont.

When configuring OSPF, you need to activate individual interfaces.

So you use a wildcard mask to pick which interfaces should be activated.

For router 2, 10.1.1.1 would match the *network* command.

10.1.1.254 would also match.

We are picking **IPs**. Not networks.


PIVIT

```
R1(config)# router ospf 1
R1(config-router)# router-id 1.1.1.1
R1(config-router)# network 10.1.1.0 0.0.0.255 area 0
R1(config-router)# network 192.168.1.0 0.0.0.255 area 0
```


```
R2(config)# router ospf 1
R2(config-router)# router-id 2.2.2.2
R2(config-router)# network 10.1.1.0 0.0.0.255 area 0
R2(config-router)# network 172.16.1.0 0.0.0.255 area 0
```

Creating Wildcard Masks

172.20.0.0 0.0.127.255



The IP must have 172.20
as its first two octets.



0.0.127.255 means that the
last 12 bits can be anything.

Will match 172.20.0.2 because the first 20 bits are the same.

Won't match 172.20.254.2 because the first 20 bits are not the same

Creating Wildcard Masks Cont.

The device will automatically change your configuration if you configure an IP with the wrong length.

172.20.4.1 0.0.255.255 becomes
172.20.0.0 0.0.255.255.

This is because the wildcard bits
overlap with the provided IP.

```
R1(config)#router ospf 1
R1(config-router)#
R1(config-router)#router-id 1.1.1.1
R1(config-router)#network 172.20.4.1 0.0.255.255
R1(config-router)#
R1(config-router)#
R1(config-router)#do show run | s router ospf 1
router ospf 1
  router-id 1.1.1.1
  network 172.20.0.0 0.0.255.255
```

Creating Wildcard Masks Cont.

Consider this wildcard: 192.168.0.0 0.0.255.255

192.168.245.239 would match because the last two octets are wildcarded.

192.167.255.244 would not match because the second octet isn't in the wildcard bits.

Access Control Lists (ACLs)

ACL mechanics

1. Find the first matching rule, apply the action.
2. If there are no matching rules, deny by default.

If the packet matches a permit, the device will allow the packet to pass.

IT WILL NOT KEEP GOING DOWN THE ACL

Standard vs. Extended ACLs

Standard ACLs only look at the source IP.

Should be placed near the destination.

Extended ACLs can look at:

1. Source IP
2. Destination IP
3. Protocol Type
4. Time (you can set an ACL to deny over the weekend)

Should be placed near the source.

Numbered ACL

Typically seen on older configurations.

Configured using multiple global commands.

Subsequent commands will add a new rule to the end of the ACL.

Standard ACL: *access-list <ACL Number> <permit|deny> <IP>*

Extended ACL: *access-list <ACL Number> <permit|deny> <ip|udp|tcp> <source IP> <destination IP>*

Named ACL

The modern way to do ACLs.

The all the ACL rules are under a global command, vs. everything being global.

Starts with 'ip access-list', instead of just 'access-list'.

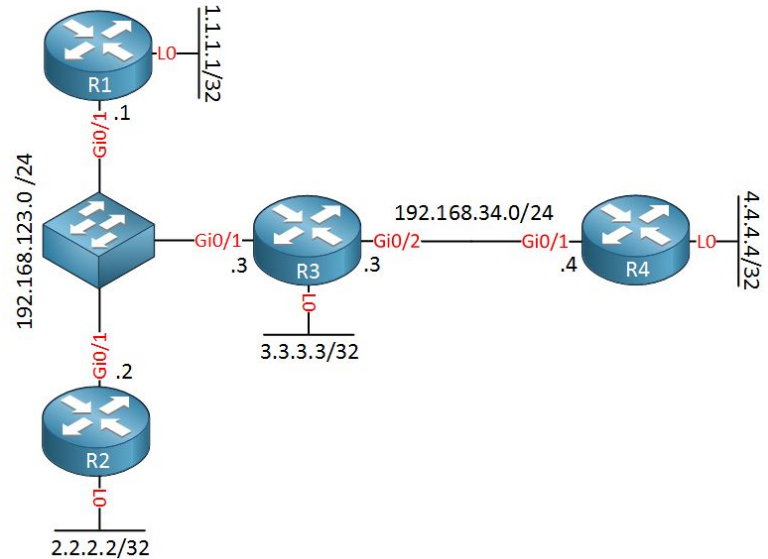
```
ip access-list extended TEST
  permit tcp 172.20.0.0 0.0.127.255 192.168.0.1 0.0.0.0 http
  deny udp 172.21.0.0 0.0.255.255 172.20.0.0 0.0.0.0 eq 9999
  deny udp host 10.0.0.1 gt 12345 host 172.20.4.2 le 98654
!
ip access-list standard TEST2
  permit 192.168.0.0 0.0.0.255
  deny 172.20.0.0 0.0.127.255
```

Standard ACL placement

Desired effect:

Stop 4.4.4.4 from talking to 2.2.2.2

Allow 4.4.4.4 to talk to anybody else



We are given this ACL: *access-list deny host 4.4.4.4*

If we are trying to stop 4.4.4.4 from talking to 2.2.2.2, we should apply it on R2
Lo0, not R3 Gi0/2. **PLACE STANDARD ACLS CLOSER TO THE DESTINATION.**

Extended ACLs

Placement of Extended ACLs will not affect how the rule is applied.

You can place it along the path between any two networks and the effect will be the same.

But, extended ACLs take up a lot of RAM (TCAM) on the router, so you should put them all closest to the source.

10 source routers each having 1 ACL vs. 1 destination router with 10 ACLs.

Spread out the work so one router doesn't have to do everything.

Security Concepts

Vulnerability: Something that theoretically could give an attacker some control.

Exploit: Takes advantage of the vulnerability.

Threat: The entity who would use the exploit to take advantage of the threat.

Attack: The entity actually using the exploit.

Types of Attacks

Spoofing/Impersonation: the attacker attempting to seem like someone else

An attacker could statically assign themselves same IP as someone else

Man-in-the-Middle (MITM): The attacker using impersonation and interception

Attacker acts like the client and server at the same time

Denial of Service: anything to disrupt service

DDoS is overwhelming a system using multiple clients, but other kinds exist

Types of Attacks Cont.

Reflection: When an attacker tries to get a system to send a response to the victim

Like if you sent a fake return address and it got returned to someone else

Amplification: When reflection attacks become high volume.

Securing local logins

Use *username* <username> *secret* <secret> over *password*

Password is an old command that is kept for compatibility reasons.

Always use *service password-encryption*

You can create an enable secret with a specific encryption by doing:

enable algorithm-type <md5|sha256|scrypt> *secret* <secret>

Securing Login Lines

Lines on Cisco devices dictate who can login.

Lines are like phone lines. When someone is using a phone line, nobody else can use it.

You can secure lines by creating an ACL and applying it to the line.

Remember *access-class* is for lines.

I remembered that we lined up for class in elementary.

access-group is for interfaces.

```
line vty 0 5
password cisco
access-class TEST_ACL in
```

Port Security

switchport port-security maximum <max number>

switchport port-security violation <protect|restrict|shutdown>

switchport port-security mac-address <mac address|sticky>

DHCP Snooping

Attacker can try and respond to DHCP requests themselves, becoming a MITM for hosts.

DHCP Snooping only allows DHCP responses from trusted ports.

You have to manually designate a trusted port (usually trunk).

Any DHCP responses from untrusted ports will be flagged.

Configuring DHCP Snooping

Enable the trusted port (configure this under the trusted port)

```
ip dhcp snooping trust
```

Enable DHCP Snooping

```
ip dhcp snooping
```

Enable DHCP Snooping for the specific VLAN

```
ip dhcp snooping vlan 1,2,3
```

YOU NEED BOTH THE GLOBAL AND SPECIFIC VLAN COMMANDS

Virtual Private Networks

Allows for two network devices to seem right next to or close to each other.

Remote Access VPN: NordVPN, Cisco AnyConnect

Allows a single host to be in a network remotely

Site to Site VPN: GRE Tunnel, DMVPN, GETVPN, FlexVPN

Allows routers to seem directly plugged to each other

Use case for Site to Site VPN

OSPF requires that you be able to send multicast over a link.

The routers have to be connected/reachable over layer 2.

GRE Tunnels allow those two routers to seem plugged into each other

You can wrap the OSPF hellos and other multicast packets in a GRE Tunnel

Now you can share the network topology of two sites without having to buy a direct circuit to that other location.

Securing Wireless

WEP was the first authentication and encryption protocol but its really insecure

Now we have WPA, WPA2, and WPA3

2 Options for authentication and authorization for the new WPAs

- PSK (preshared keys)

- 802.1X (Usually called WPA Enterprise)

Securing Wireless Cont.

TKIP: Temporal Key Integrity Protocol - Weak, only WPA

CCMP: Counter/CBC-MAC Protocol - Strong, WPA & WPA 2

GCMP: Galios/Counter Mode Protocol - Strongest, WPA 3

Networking planes

The planes represent functions and behaviors that allowed designers to focus better.

Forwarding plane: actually sending the packets around

Control plane: acquiring and sharing routes

Management plane: keeping track of time and SSH sessions

Controller Based Network

Instead of routers having to individually share their own routing tables, there is one controller that calculates the routing table and distributes it to every router.

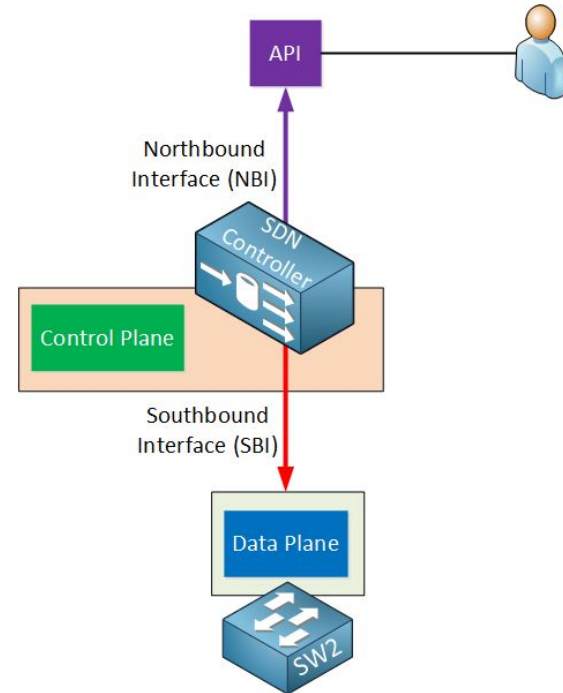
The control and management plane are now centralized in the controller.

Now the devices only have to worry about actually sending the packets, not doing OSPF or EIGRP.

Northbound and Southbound APIs

The controller is between the user (north) and the devices (south)

An API is a way to communicate with something using code.



Configuration Management Tools

We have tools to manage our network.

So a little bit of a step down from complete controller-based SDN network, but we are still automating stuff.

To actually communicate with these tools you need a language or serialization format.

Ansible

Package of actions: Playbook

Playbooks hold all the stuff that will run on each device

Protocol: SSH/NETCONF

Uses SSH to login and do the changes

Agentless/Agent Model: Agentless

Nothing has to be installed on the devices to be managed

Puppet

Package of actions: Manifest

Manifests hold all the stuff that will run on each device

Protocol: REST over HTTP

Uses HTTP to manage devices

Agentless/Agent Model: Agent/Agentless

Usually requires an agent to be installed on the device to be managed.

Can provide workaround for devices that can't have software installed normally

Chef

Package of actions: Recipe/Runlist

Recipes hold all the stuff that will run on each device

Protocol: REST over HTTP

Uses HTTP to manage devices.

Agentless/Agent Model: Agent

Agent must be installed on the devices to be managed

Data serialization

Serialization means to turn data into a portable format.

Serialization formats include:

- JSON
- YAML
- XML
- CSV (Spreadsheets)