Cisco Packet Tracer Commands Summary

Basic Router Configuration

Command line modes

- **User EXEC mode:** is the default when you open.
- **Priviliged EXEC mode:** `enable`
- **Global configuration mode:** `configure terminal`

    1. **Interface mode:** Used to configure the device's interface/port
    2. **Line mode:** Used to configure the device's access lines
    3. **Router mode:** Used to configure the routing information

---

Configuring router's interfaces in different ways

- GUI of the router
- CLI directly on the router
- Through an end-device using **Console Cable** connected to the router

```
// Console port: is the port used to directly access the router,
// and configure it to authenticate the access of this port.
Router(config)# line console 0
Router(config-line)# password yourpassword
Router(config-line)# login
```

- Through an end-device using **Remote Access Control**

```
// Virtual Teletype (VTY) port: is the port used to remotely access the r
outer
// and configure it to authenticate the access of this port.
Router(config)# line vty 0 4
Router(config-line)# password yourpassword
Router(config-line)# login
```

```
// to access it, open the command prompt on any end-device
// and do the following command
telnel ip-address-of-router
```

Basic commands

1. Configuring a router Host Name:

```
Router(config)# hostname NES413
```

2. Login banners: A login banner is a message that is displayed at login.

```
Router(config)# banner motd #your statment#
```

3. Prevent the translation of incorrectly entered commands as though they were hostnames.

```
Router(config)# no ip domain-lookup
// or use Ctrl+Shift+6
```

4. Configuring router passwords: Passwords restrict access to routers. The following two commands can be used to establish authentication
   before accessing **privileged EXEC mode**:

   - Enable Password: set a password to the privileged mode. The password is displayed as clear text in the router's configuration file.

     ```
     Router(config)# enable password yourpassword
     ```

   - Enable Secret: encrypted secret password to privileged mode. The password is displayed as encrypted text in the router's configuration file.

     ```
     Router(config)# enable secret yourpassword
     ```

5. The service password-encryption global configuration command to prevent passwords from displaying as plain text in the configuration file.

   ```
   Router(config)# service password-encryption
   ```

6. Configuring Ethernet interfaces:

   ```
   Router(config)#interface type-and-number
   Router(config-if)# description descriptive-text
   Router(config-if)#ip address ipaddress subnetmask
   Router(config-if)#no shutdown
   ```

7. To verify the IPv4 addresses for all interfaces:

   ```
   Router# show ip interface brief
   ```

8. To show the routing table of the router use the following command in the privileged EXEC mode:

   ```
   Router# show ip route
   ```

A Cisco network device contains two configuration files

1. **The running configuration file (RAM).**
2. **The startup configuration file (NVRAM).**

- To save the current configuration of the router to the startup configuration file:

```
Router# copy running-config startup-config
```

- To show the overall configurations that you make:

```
Router# show running-config
```

- If you need to restore the previous configurations:

```
Router# reload
```

- If the undesired changes were saved to the startup-config file, it may be necessary to clear all the configurations:

```
Router# erase stratup-config
Router# reload
```

Static Routing

1. To configure **static routes** with a *next-hop IP address* or *exit interface* specified:

```
Router(config)# ip route network-address subnet-mask next-hop-address
```

```
Router(config)# ip route network-address subnet-mask exit-interface
```

2. To configure a **default static route** with a *next-hop IP address* or *exit interface* specified:

```
Router(config)# ip route 0.0.0.0 0.0.0.0 next-hop-address
```

```
Router(config)# ip route 0.0.0.0 0.0.0.0 exit-interface
```

Dynamic Routing

To view information about the routing processes:

```
Router# show ip protocols
```

1) Using RIP (Routing Information Protocol)

```
Router(config)# router rip
Router(config-router)# version 2
Router(config-router)# no auto-summary

// for each directly connected network
Router(config-router)# network network-address

// set passive interface
Router(config-router)# passive-interface interface-type-and-number
```

2) Using OSPF (Open Shortest Path First Protocol)

```
Router(config)# router ospf process_ID


// for each directly connected network
// wildcard_mask = 255.255.255.255 - subnet mask
Router(config-router)# network network_address wildcard_mask area 0


// set passive interface
Router(config-router)# passive-interface interface-type-and-number
```

DHCP, FTP and Standard ACL Configuration

1) DHCP (Dynamic Host Configuration Protocol)

- Exclude statically assigned IP addresses:

```
Router(config)# ip dhcp excluded-address FirstIP LastIP
```

- Configure the DHCP pool:

```
Router(config)# ip dhcp pool POOLNAME
Router(dhcp-config)# network NETWORKID MASK
Router(dhcp-config)# dns-server IP Address
Router(dhcp-config)# default-router IP Address
```

- To display a list of all IPv4 address to MAC address bindings that have been provided by the DHCP server:

```
Router# show ip dhcp binding
```

---

2) FTP (File Transfer Protocol)

- Enable FTP service on the server from the **Services tab**.

- Create user accounts in the **User Setup** and specify the permissions you want for each user: *(Write, Read, Delete, Rename and List)*.

- Now, on end-devices, on the command prompt:

```
// to connect to your FTP server
ftp ip_address_of_Server

// to list the contents of the directory
dir

// to upload a file into the server
put myfile.txt

// to download a file from the server
get myfile.txt

// to delete a file from directory
delete myfile.txt
```

3) ACL (Access Control List)

ACL is a set of IOS commands applied to a router's interface and used to **filter packets** based on the information found in the packet header. **There are two types of ACL:**

- **Standard ACL** uses the source IP address of the packet to control whether a packet is permitted or denied.
- **Extended ACL** allows the router to filter the packets based on the source and/or destination IP addresses.

We will cover Standard ACL only:

1. **Numbered Standard ACL:**

```
Router(config)# access-list access-list-number {deny | permit | remark te
xt} source [source-wildcard] [log]

// wildcard mask identifes which source address should be filtered
// there are two keywords for the most common use of wildcard mask:

// 1) Host: this keword is equivalent to the 0.0.0.0 wildcard
// (filtering only single IP address)
Router(config)# access-list 1 deny host 10.0.0.2

// 2) Any: this keyword is equivalent to the 255.255.255.255 wildcard
// (any IP address is accepted)
Router(config)# access-list 1 permit any
```

2. **Named Standard ACL:**

```
Router(config)# ip access-list standard access-list-name

// After an ACL is created it must be linked to a router's interface
Router(config)# interface interface-type-and-number

```

```
Router(config-if)# ip access-group {access-list-number | accesslist-name}
{in | out}

// in: packets are filtered before being routed.
// out: packets are filtered after being routed.
```

- In Priviledged mode, you can view the **ACL configurations** using:

```
Router# show access-lists
```

Basic IPv6 Configuration

- Configure IPv6 addressing on interfaces:

```
// to enable the router to forward IPv6 packets.
R1(config)# ipv6 unicast-routing

// Configure the IPv6 Global Unicast Address (GUA)
R1(config)#interface type-and-number
R1(config-if)# ipv6 address 2001:DB8:1:1::1/64

// Configure the link-local IPv6 address
// sed to communicate with other devices that are on the same link
R1(config-if)# ipv6 address FE80::1 link-local

// Don't forget to activate the interface
R1(config-if)# no shutdown
```

- To verify the IPv6 addresses for all interfaces:

```
R1# show ipv6 interface brief
```

- OSPFv3 routing:

```
// enable OSPFv3 routing
R1(config)# ipv6 router ospf process_id

// inside the ospf router subconfiguration mode
R1(config-rtr)# router-id router_id
R1(config-rtr)# passive-interface interface_name

// access the activated and configured interfaces
R1(config-if)# ipv6 ospf process_id area 0
```

- OSPFv3 validation:

```
R1# show ipv6 route

R1# show ipv6 protocol
```

VLANs Configuration in Switched Networks

- Creating the VLANs:

```
Switch(config)# vlan vlan-id
Switch(config)# name vlan-name
Switch(config)# end
```

- Assigning switch ports to specific VLANs (either access or trunk).

```
Switch(config)# interface interface-id

Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan vlan-id

Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan 10,20,30
```

- Show commands:

```
Switch# show vlan
Switch# show interfaces interface_id switchport
Switch# show interfaces trunk
```

- Inter-VLAN routing:

```
R1(config)#interface interface-id
R1(config-subif)# encapsulation dot1Q 10
R1(config-subif)# ip address ip-address subnet-mask
```

- To manage the switch (and the VLANs) remotely:

```
// Set up the SVI (switch virtual interface)
Switch1(config)# interface vlan vlan-id
Switch1(config-if)# ip address ip-address subnet-mask
Switch1(config-if)# no shutdown
Switch1(config-if)# exit
Switch1(config)# ip default-gateway ip-address
```

```
// Configure the VTY lines on the switch
Switch1 (config)# line vty 0 4
Switch1 (config-line)# password your-password
Switch1 (config-line)# login

// Access the switch remotely using telnet
// Using any end-device, on the command prompt
telnet switch-ip-address
```

NAT Configuration

1. **Static NAT:**
   Static NAT is a **one-to-one mapping** between an inside local address (private address) and an inside global address (public address) **configured by the network administrator** that remain constant.

```
// Create a mapping b/w the inside local address & the inside global addr
ess
Edge(config)# ip nat inside source static Private-Address Public-Address

// Configure the participating interfaces in the translation as inside or
outside
Edge(config)# interface interface-id
Edge(config-if)# ip nat inside

Edge(config)# interface interface-id
Edge(config-if)# ip nat outside
```

- **Verify NAT operations:**

```
Edge# show ip nat translations

Edge# show ip nat statistics
```

2. **Dynamic NAT:**
   Dynamic NAT **automatically** maps inside local addresses to inside global addresses (many-to-many mapping), it uses a **pool of inside global addresses.**

```
// 1. Define the pool of addresses that will be used for translation
Edge(config)# ip nat pool pool-name First-IP Last-IP netmask Subnet-mask

// 2. Configure a standard ACL to permit only those addresses that are to
be translated
Edge(config)# access-list list-id-number permit IP-address wild-card

// Note: in the wildcard, 0 bit means exact match in IP bit (host)
// and 1 bit means any value in IP bit (any).

// 3. Bind the ACL to the pool
Edge(config)# ip nat inside source list list-id-number pool pool-name

// 4. Identify which interfaces are inside & which are outside
Edge(config)# interface interface-id
Edge(config-if)# ip nat inside

Edge(config)# interface interface-id
Edge(config-if)# ip nat outside
```

- It is best to clear statistics from any past translations:

```
Edge# clear ip nat translation *
```

PAT Configuration

PAT, also known as **NAT overload**, maps **multiple private IPv4 addresses to a single public IPv4 address** or a few addresses (many-to-one mapping).

1. **Configure PAT to use a single IP address:**

```
// 1. Configure a standard ACL to permit only those addresses that are to
be translated
Edge(config)# access-list list-id-number permit IP-address wild-card

// 2. Bind the ACL to the single IP address that will be used
Edge(config)# ip nat inside source list list-id-number interface interfac
e-name overload

// 3. Identify which interfaces are inside & which are outside
Edge(config)# interface interface-id
Edge(config-if)# ip nat inside

Edge(config)# interface interface-id
Edge(config-if)# ip nat outside
```

1. **Configure PAT to use an Address Pool:**
   To configure PAT to use an address pool, you will use the same steps used in the dynamic NAT except that you **add the keyword `overload` to the ip nat inside source command**.

```
Edge(config)# ip nat inside source list list-id-number pool pool-name ove
rload
```

Tips

1. Use the `?` in the command prompt to remember any command you might've forgetten.

2. Check the connectivity using `ping` on the command prompt of any end-device.

3. Check the up/down indicators (the green triangles) on Cisco Packet Tracer interfaces as they represent the **link status** of the network interface.

```
// or using this command
Router# show ip interface brief
```