Packet Tracer Cheat Sheet

Table of contents

Basic Configuration

Show configurations

```
 sh run
sh start
sh ip int br
```

Device name and password settings

```
 en
conf t
hostname <name>
```

Setting line password

To secure user exec mode, give password to line console 0

```
 en
conf t
line console 0
password <password>
login
end
```

Securing privileged mode

To secure privileged mode give password to privileged mode

```
 en
conf t
enable secret <password>
exit
```

Securing vty line (virtual lines that are required to telnet/SSH)

```
 en
conf t
line vty 0 15
password <password>
login
end
```

`note:` vty 0 15 means that vty lines from 0 to 15 are secured. Can be used any range of vty lines.

Encrypting passwords

```
 en
conf t
service password-encryption
exit
```

Setting banner message

```
 en
conf t
banner motd #<message>#
```

Saving configurations

```
 copy run start
```

Erasing start configurations

```
 erase start
reload
```

Back to top

Enabling Telnet/SSH on layer 2 switch

To access the switch remotely, an IP address and a subnet mask must be configured on the SVI (Switch virtual interface). For SVI use VLAN 1. And then line vty needs to be secured

```
 en
conf t
int vlan 1
ip add <id-address> <subnet-mask>
no shut
exit
line vty 0 4
password <password>
login
```

Enabling Telnet/SSH on router

It's the same process as enabling telnet/SSH on layer 2 switch except it doesn't require any ip address on vlan 1.

```
 en
conf t
line vty 0 4
password <password>
login

int vlan 1
no shut
exit
```

Back to top

Configuring VLANs

Show VLAN configuration

```
sh vlan br
```

To check the VLAN configuration of a specific port. (Does not work on router)

```
sh int <line name> switchport
```

Step 1: VLAN creation

```
 en
conf t
vlan <vlan-id>
name <vlan-name>
end
```

Step 2: VLAN port assignment

```
 en
conf t
int <line name>
switchport mode access
switchport access vlan <vlan-id>
end
```

Configuring trunk VLAN

```
 en
conf t
int <line name>
sw mode trunk
sw trunk native vlan <vlan-id>
sw trunk allowed vlan <vlan-list separated with comma>
end
```

Deleting VLAN

- Deleting one VLAN:

  ```
  no vlan <vlan-id>
  ```

- Deleting all VLANs:

  ```
  delete flash:vlan.dat
  ```

  Then reload

`Caution:` Before deleting a VLAN, reassign all member ports to a different VLAN

Back to top

Configuring router subinterface

```
 en
conf t
int <subinterface-id> (eg: g0/0.10)
encapsulation dot1q <vlan-id>
ip add <ip-address> <subnet-mask>
int <line name> (eg: g0/0 in this case)
no shut
```

Back to top

Configuring NAT

Show NAT translations

```
sh ip nat trans
```

Steps to connect a network to the internet or to another network using router

1. Add ports (WIC-2T) to the router. (Turn the router off before inserting port)
2. connect the routers in serial0/0/0
3. set ip addresses on the lines of the routers.
4. set clock rate (64000) of the network side router: `ios en conf t int s0/0/0 ip add <ip> <subnet> clock rate 64000` `Note:` Write the following if no route is set to the network side router or the network is a stubbed network:

```
ip route 0.0.0.0 0.0.0.0 <interface-number> (e.g.: s0/0/0)
```

Dynamic NAT examples

Step 1: Configuring the inside and outside interfaces

```
 int g0/0
ip nat inside
int g0/1
ip nat inside
int s0/0/0
ip nat outside
exit
```

`Note:` The inside interfaces are the interfaces that are connected to the private network. The outside interface is the interface that is connected to the internet or another network.

Step 2: Setting the NAT pool

```
 ip nat pool BUET-pool 209.165.200.8 209.165.200.11 netmask 255.255.255.224
access-list 1 permit 192.168.10.0  0.0.0.255 (wildcard mask - reverse of subnet)
access-list 1 permit 192.168.20.0  0.0.0.255
ip nat inside source list 1 pool BUET-pool
```

`Note:` Here 209.165.200.8 209.165.200.11 is the range of public ip addresses that will be assigned to the private ip addresses.

Static NAT examples

```
 int g0/0
ip nat inside
int g0/1
ip nat inside
int s0/0/0
ip nat outside
exit

ip nat inside source static <source-pc-ip> <outside-line-ip>
(do this for all the PCs)
```

PAT examples

Everything is the same as Dynamic NAT except an "overload"

```
 int g0/0
ip nat inside
int g0/1
ip nat inside
int s0/0/0
ip nat outside
exit

ip nat pool BUET-pool 209.165.200.8 209.165.200.8 netmask 255.255.255.224
access-list 1 permit 192.168.10.0  0.0.0.255
access-list 1 permit 192.168.20.0  0.0.0.255
ip nat inside source list 1 pool BUET-pool overload
```

Back to top

Configuring ACL

ACL has to be applied to an interface (inbound or outbound)

Show ACL configuration

```
 sh access-list
```

Clear access-list 1 before configuring

```
 conf t
no access-list 1
```

Standard ACL examples:

- Only specifies the source address in standard ACL.
- 0-99 are standard ACLs.

`placement:` Closest to the destination. That means, set the ACL on the interface that is connected to the destination.

Here is an example:

```
access-list 10 remark ACE permits only host 192.168.10.10 (this is just a comment)
ip access-list 10 permit 192.168.10.10  0.0.0.0 (wildcard mask)

int s0/0/0
ip access-group 10 out
```

`Note:` writing `host 192.168.10.10` and `192.168.10.10 0.0.0.0` are the same.

** If we want to deny all but `192.168.10.10` of 10: **

```
ip access-list host 192.168.10.10
ip access-list deny any 192.168.10.0  0.0.0.255
```

Clearing ACL example

```
no access-list 10
```

Named standard ACL example

```
ip access-list standard <name>
permit host 192.168.10.10

int s0/0/0
ip access-group <name> out
```

Back to top

Extended ACL examples

- Extended ACLs are used to specify both source and destination addresses.
- 100-199 are extended ACLs. `placement:` closest to the source. That means, set the ACL on the interface that is closer to the source.

Here is an example of named extended ACL:

```
ip access-list extended FTP-FILTER
permit tcp 192.168.10.0  0.0.0.255 host 100.100.100.3 eq ftp
permit tcp 192.168.10.0  0.0.0.255 host 100.100.100.3 eq ftp-data

int s0/0/0
ip access-group FTP-FILTER out
```

To clear the above FILTER:

```
 no ip access-list extended FTP-FILTER
no ip access-group FTP-FILTER out
```

`Note:` To permit or deny all other, use `any` keyword.

```
 permit ip any any
deny ip any any
```

Some important configuration steps

Communication between devices under same switch

1. set ip address and subnet of the devices' line.

Telnet configuration on switch

** host part of ip should have to match **

1. enable password
2. conf a vlan interface on the switch with an ip
3. configure a telnet password (line vty 0 15)

Telnet configuration on Router

** Network part of ip should have to match ** 1. enable password 2. configure ip address of the line that is connected to the switch/Device 3. set ip address, subnet mask and default gateway on the pc 4. up vlan 1 (no ip required) 5. set telnet password (line vty)

Vlan configuration

1. create vlans on individual switches
2. assign vlans on individual ports in access mode
3. assign native and other vlans to trunk port (line connecting to other switch)
4. configure ip addresses of the PCs

Legacy inter-VLAN routing

1. create vlans on individual switches
2. assign vlans on individual ports in access mode
3. connect the router with the switch. (one line for every vlan. Max 2 is possible)
4. set ip of the router lines from the router side
5. set default gateways of the PCs

Router on a stick inter-VLAN routing

1. create vlans on individual switches
2. assign vlans on individual ports in access mode
3. connect the router with the switch (use only one port)
4. set the port as trunk on the switch side
5. configure subinterfaces in router (no. of subinterfaces = no. of vlans)
6. set default gateways of the PCs

Back to top