CN - LAB MID

LAB 04:

- errors starting from 4 → client side errors
  " " " 5 → server " "

① Make a topology (server, end devices, switches) of the network.

② The server should host a particular website so when a PC/laptop requests a website, it can be fetched

③ Give IPs starting from server and then to other devices

④ Turn on services of HTTP & HTTPS on server in services tab
edit the website in index.html option with your website name.

⑤ Go to any PC webbrowser → type IP address of website (hosted on webbrowser) and search.

⑥ You can transfer ✉ from PC to server using simulation mode → click on HTTP after request is recieved to open HTTP header and see details.

⑦ to add HTTPS, add 's' in webbrowser search bar for the PC ~~on which~~ from which you requested website and again simulate the message and ~~open~~ open HTTPS header.

DNS Server Addition

⑧ to all the connected servers hosting diff websites ON DNS in services tab
assign name to website (www.myweb.com) and its own IP address → click ADD.

⑨ Add a DNS server and give it IP
add IP's of all other webservers along with their website names to DNS under services tab.
ADD IP of ~~&~~ DNS server in ~~default gateway~~ DNS Server of each PC/Laptop under config bar

Router's IP assigning

→ using GUI;
Fa 0/0 → assign IP → turn port ON
add that IP as default gateway to all devices connected in that network to pass through router

→ using CLI:
Router> en
Router# config t
Router(config)# int fa0/1
Router(config-if)# ip address 192.168.2.1 255.25
// # no ~~shut~~ shut.

LAB 05:

SMTP

① Make topology and assign IPs accordingly.

② Go to Server: → Services tabs:

a. ~~On~~Enable HTTP and HTTS

b. Enable SMTP and Pop 3 ~~se~~ in email tab.

c. Set a domain name: ie: nu.edu.pk → click set.

i.e: gmail.com → click set

~~d. set users in email tabs:~~ add

d. add username and password against that domain name in email tab:

ie: User: user1 password: abc → [+]

user: user2 password def → [+]

③ Go to user's PC → Desktop → Email:

Configure Mail:

Your name: Ali

Email adder: user1@nu.edu.pk

Incoming Mail serv:

Outgoingg " " :

Username: user1

Password: abc.

click: [Save]

④ Now send email from one user to another and recieve ~~to~~ on that serves.

* use C class IPs in paper

→ When you are adding multiple domains and want to sent email to cross domains: each domain name should have a ~~dt~~ separate server allocated.

⑤ Add a DNS server:

a. configure it

b. ~~add~~ enable DNS in services tab.

c. add domain names of all server:

ie:

Name: nu.edu.pk Address: of respective server

Name: gmail.com Address: " " "

d. Add IP of DNS server in all other user PCs/laptops DNS Server bar.

⑥ Now again go to any user → Desktop → Email → send email cross domain. and check if it is being recieved.

→ If you add a router and now your DNS server is on another side of network and you want to send cross domain email: ADD ip address of DNS to all servers as well in DNS server tab.

FTP: file transfer Protocol.

Using the same topology and continuing from there:

① setup FTP for any particular server (for example setting it for nu.edu.pk server):

a. services → FTP → on (by default)

b. services → FTP → Username: umer
Password: 123
check permission boxes as applied

add more files/users as per requirment.

② Go to any user Desktop → cmd:

C:> ftp (server_ip)

Username: umer

Password: 123

loggged - in

ftp> quit

C:>

③ a. Make any file on user's Desktop → Text editor. and write something
give extension while saving along with filename.

. CN.txt ; lab.py

b. to verify file creation:

Desktop → cmd:

C:> dir

: : - (will list all files on this user)

④ Now if you want to put this created file on server:

(a) C:> ftp (server_ip)

Username: User

Password: 123

Logged-in.

ftp> put filename.

(Transfer complete - 15 bytes)

(b) to verify if file reached:

ftp> dir

Listing /ftp directory from (server_ip)

0 1 CN.Txt 15

if you want to delete from server.

(c) ftp> delete CN.txt

(d) if you want to get file from server and download it on your PC

ftp> get CN.txt

ftp> quit.

LAB 06

Telnet:

(1)

→ if switch is attached: ip configuration:

~~or~~

switch > en

switch # config t

switch (config) # int vlan 1

sw (config-if) # ip address (ip of switch) (subnet mask)

" " # no shut.

(2) Assign IPs to PCs

~~(2) Assign IPs to PCs.~~

→ Apply authentication to switch:

switch > en

switch # config t

switch (config) # line vty 0 15

switch (config-line) # password cisco

" " " # login

" " " # end

switch #

→ Apply password or enable mode:

switch # config t

switch (config) # enable password cs

" " # exit

switch #

SSH configuration:

* do same for routers:

switch # config t
switch (config) # hostname huzaila
huzaila(") # ip domain name cn-lab
" (") # crypto key generate rsa

1024

huzaila (config) # ip ssh version 2
" " # line vty 0 15
" (config-line) # transport input ssh
" " " # exit.

→ if you want to change username:

huzaila (config) # username huzaila-asif secret abc
" " # line vty 0 15
" " (config-line) # login local
" " " # exit

now checking ssh on any PC/laptop:
and on PC:
C:\> ssh -l huzaila-asif (ip-switch)
password: abc

CN Lab 8

Static NAT : 1:1 mapping
translating private IP address to public IP address.

① Inside local: private IP address assigned to inside local network to end devices.

② Inside global: The public IP address to which your private local ip address would be translated. (itself it would be public)

③ Outside local: The private ip address that is of the local devices from another network that is destination.

④ Outside global: The public IP address to which the outside local/privat destinate IP address is mapped.

* Outside local and outside global are SAME if the destination you are trying to reach itself has a public IP assigned

Dynamic NAT: mapping multiple private IPs to one public IP
create a pool

PAT (port address translation)

0 - 65535 ports.

CONFIGURATIONS:

Static NAT: → giving/mapping one PC permission

router> en

router# config t

router(config)# ip nat inside source static
private (PC) → public
192.168.1.2 7.62.1.10

router(config)# int fa0/0

router(config-if)# ip nat inside

" " " # exit

router(config)# int fa0/1

router(config-if)# ip nat outside.

" " " exit

router(config)# ip route 192.168.2.0 255.255.255.0
dest network add
7.62.1.2 → next hop(router) ip add

Dynamic NAT: When we configure dynamic NAT, we have to define ACL to permit only those addresses that are allowed to be translated.

router(config)# ip nat pool pool_name 7.62.1.100 ip
start pool
end pool ip
7.62.1.115 netmask 255.0.0.0
net mask

router(config)# access-list 1 permit 192.168.2.0
network to be permitted
wildcard
0.0.0.255

router(config)# ip nat inside source list 1 pool pool_name

router(config)# int fa 0/0
router(config-if)# ip nat inside
" " " # exit
router(config)# int fa 0/1
router(config-if)# ip nat outside
" " " # exit.

PAT configuration (NAT overload)

① first create access list and add a network of ip addresses you want to allow OR if any specific you want to allow

② Then create a NAT overload (PAT) pool

router > en
" # config t
" (config)# access-list 1 permit 192.168.20.0 0.0.0.255
" " # ip nat inside source list 1 interface fa 0/1 overload
" " # int fa 0/0
router(config-if)# ip nat inside
router(config-if)# exit.
router(config)# int fa 0/1
router(config-if)# ip nat outside.

LAB - 09 (Subnetting) - in Lab

CLASS C: example 1:

dividing a network into two networks.

192.168.1.0 — 192.168.1.255

$\frac{256}{2} = 128 \Rightarrow$ 0-127, 128-255

- range of N1: 192.168.1.0 – 192.168.1.127
- range of N2: 192.168.1.128 – 192.168.1.255

and coz you changed network by dividing it into two halves, you have to keep subnet mask in such a way that it knows that network is divided into 2 segments. (& can't keep default subnet as it will then consider it one single network)

→ subnetting: 255.255.255.128

rule to check correct subnetting:

① The network address ~~should~~ before segmentation should be same as the first address of the first network after segmentation.

② The broadcast address before segmentation should be same as the last address of the last network segment after segmentation

example. 2: dividing into 8 segments.

$\frac{256}{8} = 32$ → each segment will have 8 ips.

2^3 → for subnetting take ~~last~~ most significant 3 bits of network octet and add it

- network address (192.168.0.0 and broadcast address (192.168.1.255) are NOT usable.

0 - 31
32 - 63
64 - 95
96 - 127
128 - 159
160 - 191
192 - 223
224 - 255

Subnetting: take 2^n for no. of segment you are dividing into!

255.255.255.224

How did 224 come:

8 bits = 2^3 → last 3 bits

$2^7\ 2^6\ 2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0$

0 0 0 0 0 0 0 0

Subnetting: $2^7 + 2^6 + 2^5 = 224$

$2^3 + 2^4 + 2^5 + 2^6 + 2^7$

default

- Network bits:

200.254.254.0/24 → 3 network octets = 24 bits

- Network bits after borrowing because in segmentation/subnetting, we borrow 2^n → n bits to network

for this case:

$\because 8 = 2^3 =$ 3 bits borrowed.

200.254.254.0/27 → 24+3 = 27

- to find number of networks/segments in which we are dividing:

no. of bits borrowing = n

$$\frac{256}{2^n}$$

class A: 1-127
class B: 128-191
class C: 192-223

N H H H
N N H H
N N N H

Subnetting Class A
subnet 255.0.0.0 = 11111111.00000000.00000000.00000000.
example:

10.0.0.0 255.0.0.0
borrowing one bit => $2^1 = 2$ → creating 2 subnets.
255.(0)0000000.00000000.00000000
↳ 2nd octet: 10000000 = 128
↳ subnet mask: 255.128.0.0/9

Now after breaking into two networks (changed 2nd octet)
range: 10.0.0.0 – 10.127.255.255.
10.128.255.255 – 10.255.255.255.

- valid hosts: (count all bits in 2nd, 3rd and 4th octet except borrowed bits) = 23 bits.
valid hosts: $2^{23} = 8,388,608 - 2 = 8,388,606$

Class B subnetting:
subnet: 255.255.0.0
11111111.11111111.00000000.00000000
example:
172.16.0.0 255.255.0.0
borrowing 3 bits from 3rd octet:
255.255.(000)00000.00000000
↳ 3rd octet: 11100000 = 224
↳ subnet mask: 255.255.224.0/19
↳ How many subnets created: $2^3 = 8$

↳ ~~How~~ what will be size of each network (How many hosts per subnetwork):

2^{13} → 13 bits left in host portion after borrowing 3 bits.

$2^{13} = 8192$

- usable hosts = 8192 − 2 = **8190**

↳ How to figure out the range:

255 . 255 . [224] . 0

subtract this with 256 and you'll get what to add to get next range:

256 − 224 = 32

range: ① 172.16.0.0 – 172.16.31.255

② 172.16.32.0 – 172.16.63.255

③ 172.16.64.0 – 172.16.95.255

④ 172.16.96.0 – 172.16.127.255

and so on.

RIP: (Distance vector) not

- version 1: subnetting ↑ supported
- version 2: subnetting supported.

* While establishing RIP, you need to know the networks that are directly connected to that router

configuration

router > en

router# config t

router(config)# router rip

// // # network 192.168.0.0 → directly connected networks

network 192.168.1.0

version 2.

LAB 10:

OSPF:

(Link State routing - shortest-path-first)

How to calculate Wildcard Mask

① No subnetting: (default subnet mask):

255.255.255.0 → 0.0.0.255

* Revert it (subnet mask)

② If subnetting is done:

example: subnet mask after subnetting:

255.255.255.192

* substract this with default subnet mask:

255.255.255.255

255.255.255.192

0.0.0.63 → wildcard mask

* While establishing OSPF, write network addresses AND wildcard masks of the networks directly connected to that router: (on any router)

configurations:

router>en
router# config t
router(config)# router ospf 1
router(config-router)# network 192.168.1.0
0.0.0.255 area 0
// // # network 192.168.4.0
0.0.0.255 area 0
// // # network 192.168.2.0
0.0.0.255 area 0

BGP – Border Gateway Protocol
(Path vector)

Autonomous System (AS): is a group of networks operated by one organizations under a single routing policy.

* While establishing BGP in any router, write
① IP address AND AS number of the its neighbouring router.
② and if there is directly connected network: mention its IP address of that network AND its subnet mask.

configurations:

```
router>en
router # config t
router(config) # router bgp 1        ← AS number for that particular router
router(config-router)# neighbor 192.168.1.2
                      remote-as 71
//      //      //    # network 192.168.0.0 mask
                      255.255.255.0
                    # exit
router(config) # do write → to save configuration
```

Lab 11.

vlan on switch:

switch(config)# vlan 10
(config-vlan)# name EE
" " # vlan 20
" " # name CS
" " # exit
switch(config)# int fa 0/1
for access mode: (config-if)# switchport mode access
#switchport access vlan 10

for trunk mode: switch(config)# int fa 0/1
switch(config-if)# switchport mode trunk

configuring routers ace to vlan.
Router>en
#config t
int fa0/0
no shut
int fa0/0.10
router(config-subif)# encapsulation dot1q 10
#ip address 10.0.0.1 255.0.0.0
#int fa0/0.20
#encapsulation dot1q 20
#ip address 20.0.0.1 255.0.0.0
#int fa0/0.30
encapsulation dot1q 20
ip address 30.0.0.1 255.0.0.0

Which ports to add to increase ports on router?

① Fast Ethernet: PT-ROUTER-NM-1CFE

② Serial: PT-ROUTER-NM-1S