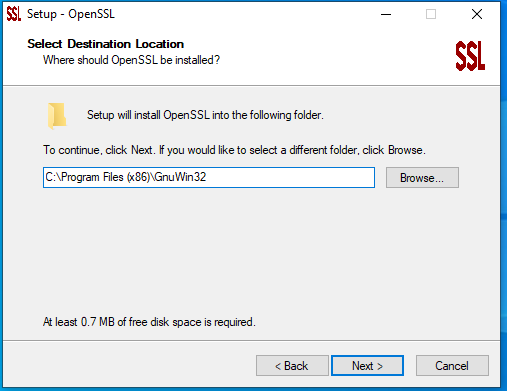
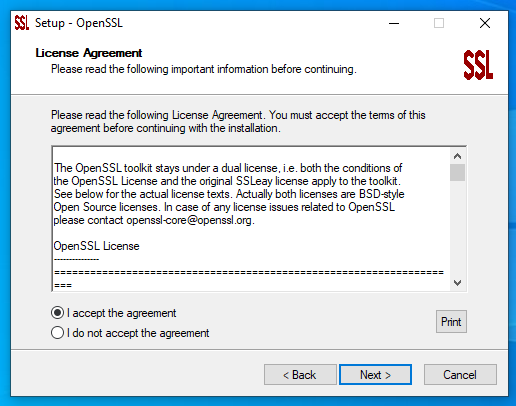
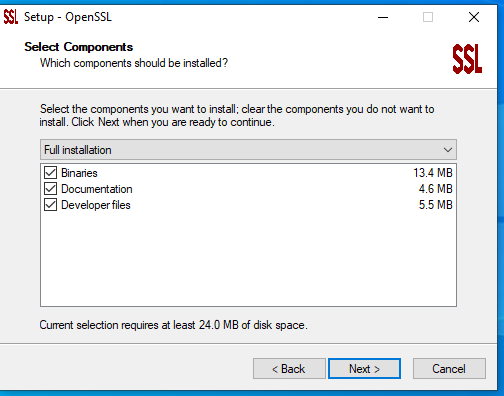
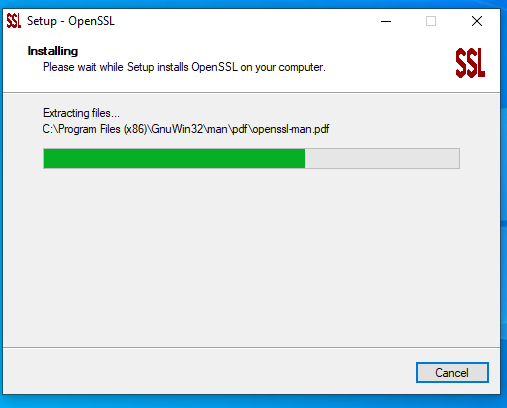
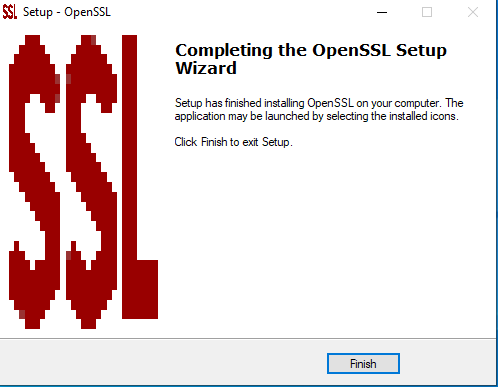
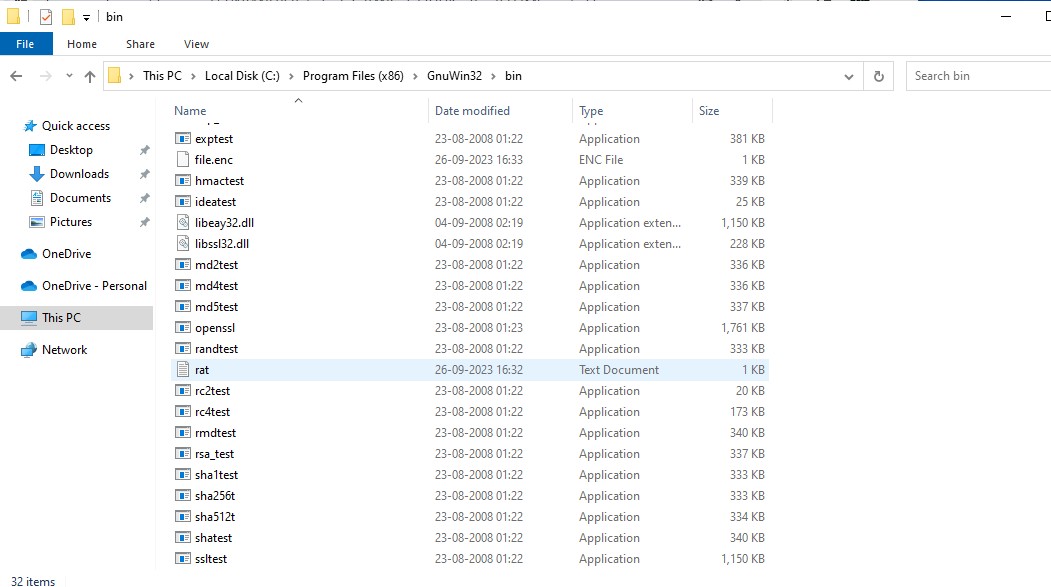
**Practical 1**

**Aim:** File encryption using SSL. (single private key)

Installation of SSL -





Go in Bin folder Of Gnu And create a text file that needs to be encrypted for example here we have created rat.txt

Open Cmd and open the location by using cd command the path where bin folder is located Command : cd C:\Program Files (x86)\GnuWin32\bin

For Encryption:

Then We have to write command :

**openssl enc -aes-256-cbc -salt -in rat.txt -out file.enc -k password**

(openssl) -

(enc) - Encryption (aes-256) - Algorithm

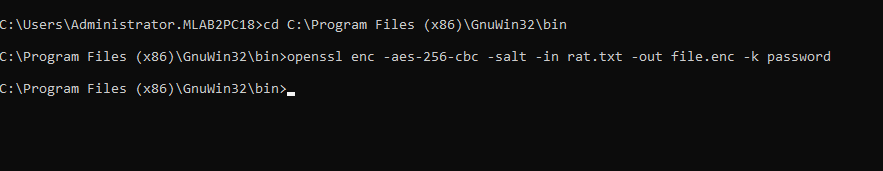
(salt) - Adding some random value (in) - input

(rat.txt) - file to be encrypted

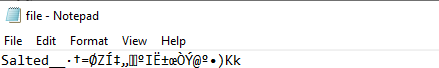
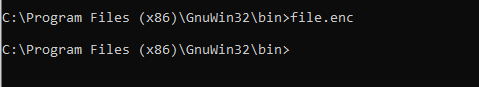
(out) - output

(file.enc) - Encrypted file

(-k)- option to set password (password) - Password which is set

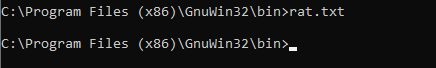


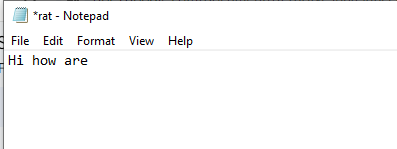
To view the encrypted file , just write encrypted file name : file.enc



For Decryption :

openssl enc -d -aes-256-cbc -in file.enc -out rat.txt -k password



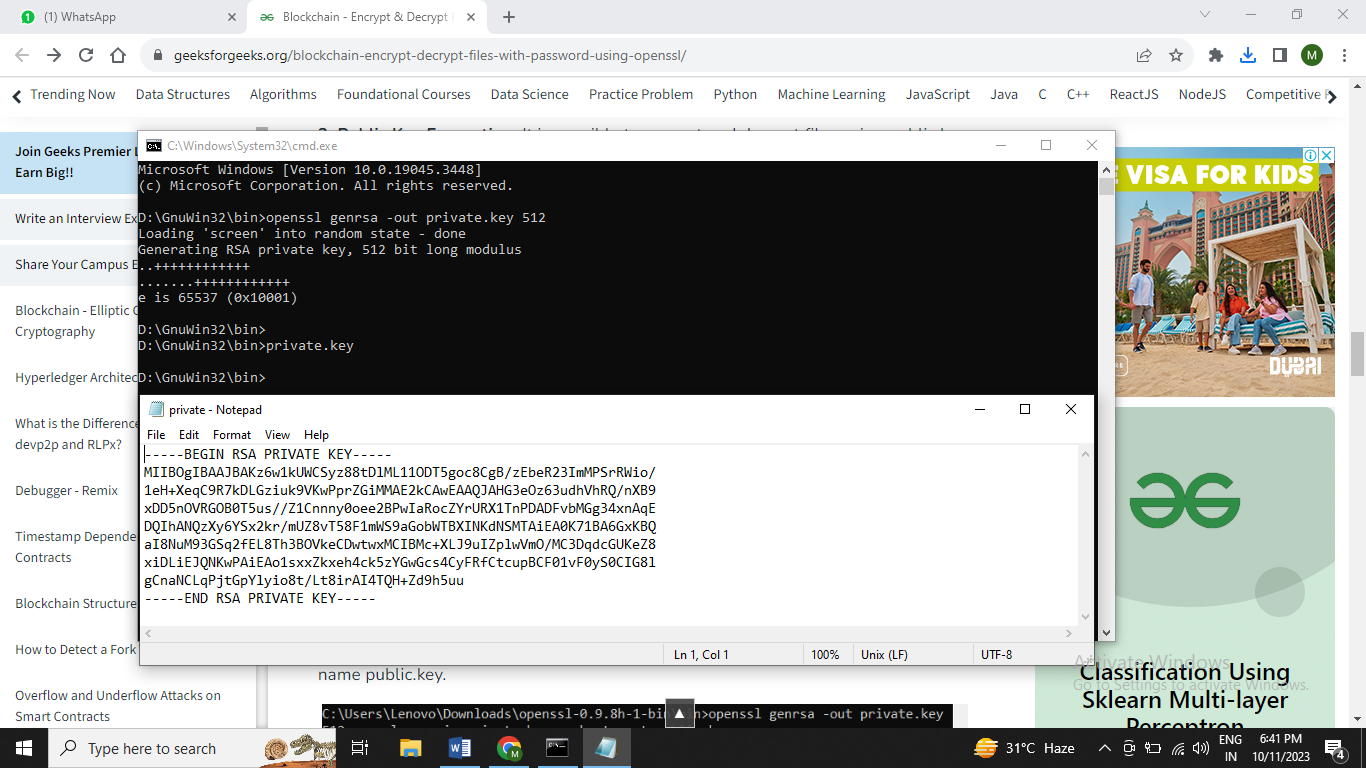


**Practical 2**

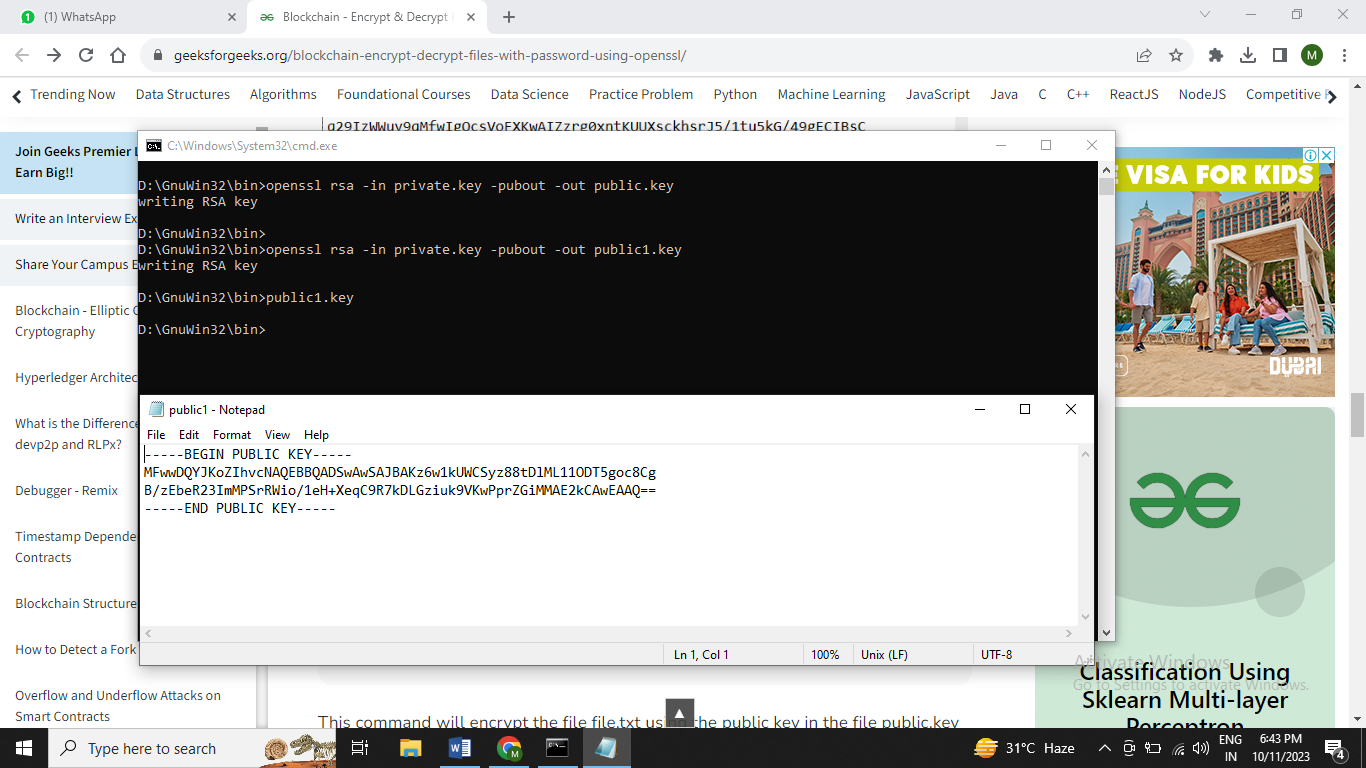
**Aim:** File encryption using SSL. (double private and public key)

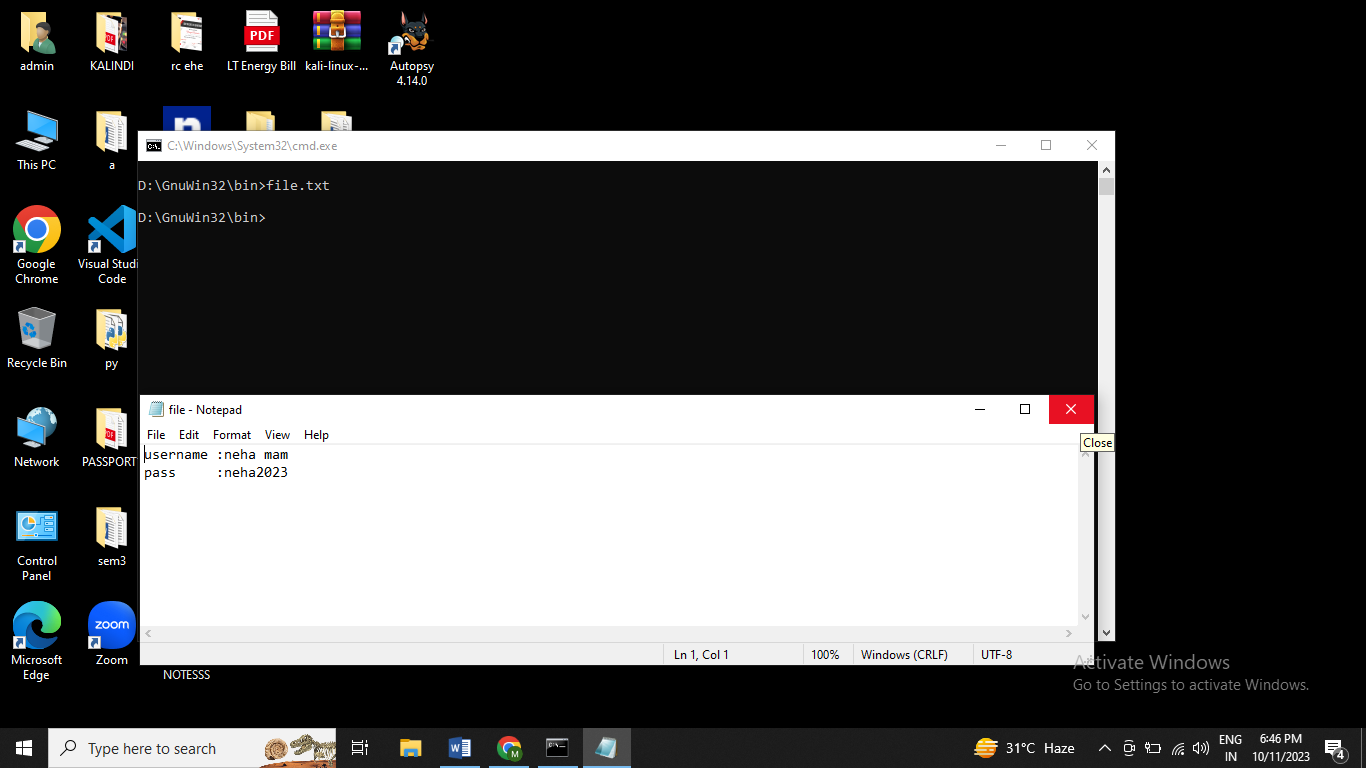
*openssl genrsa -out private.key 512*

.

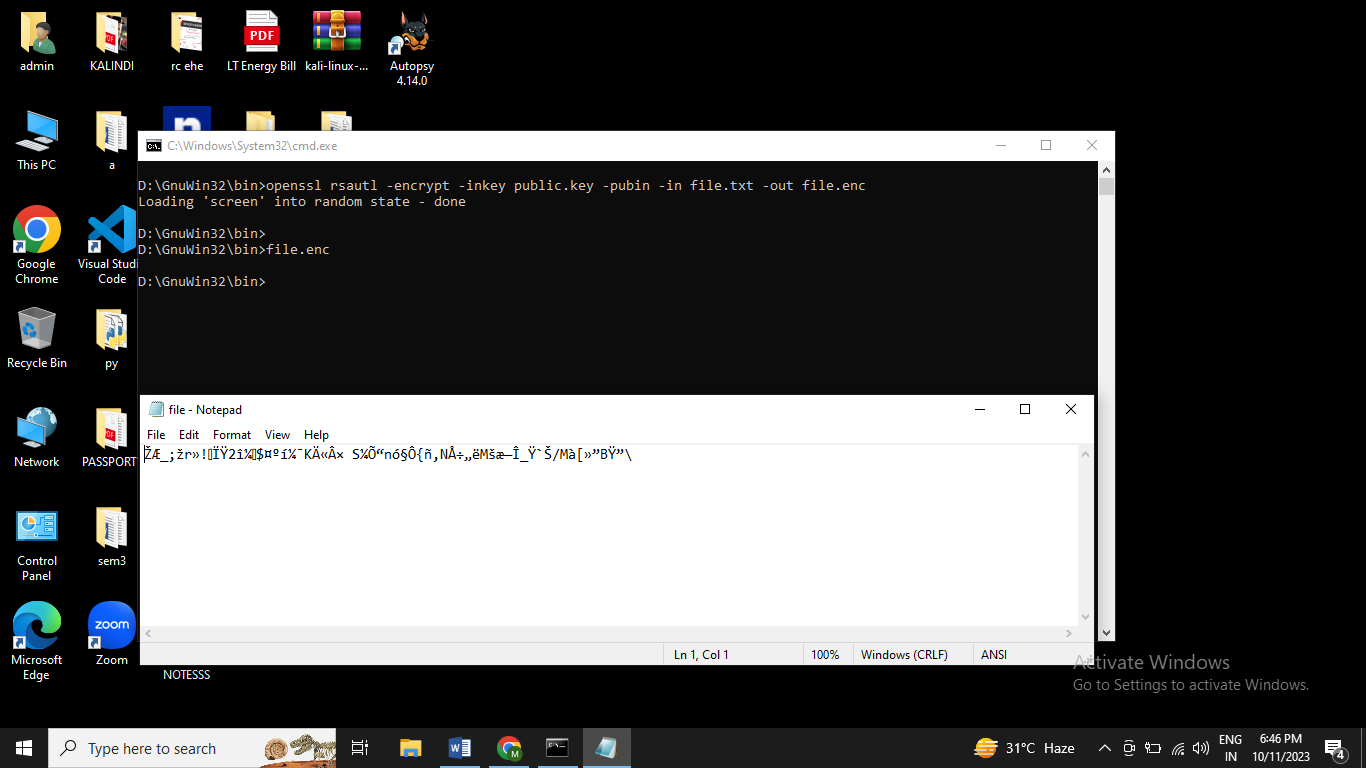


*openssl rsa -in private.key -pubout -out public.key*

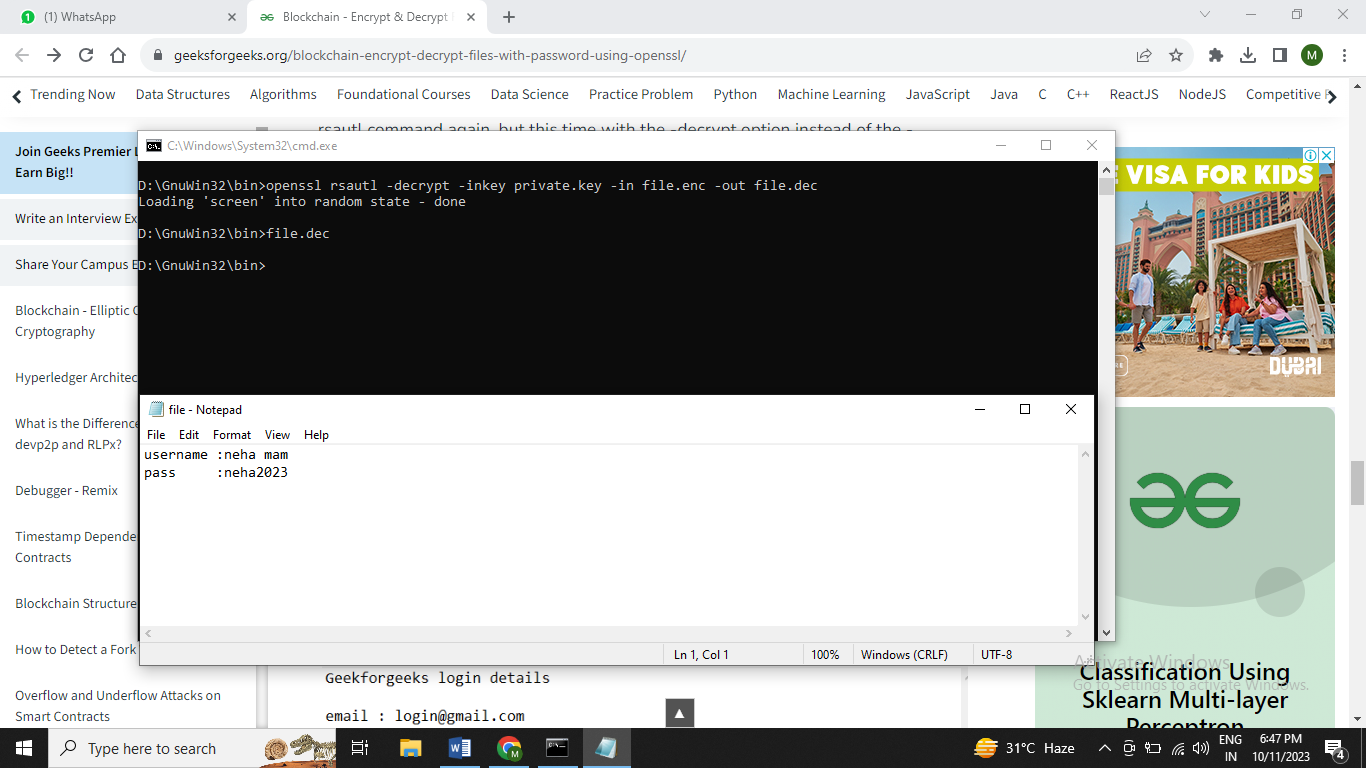




*openssl rsautl -encrypt -inkey public.key -pubin -in file.txt -out file.enc*

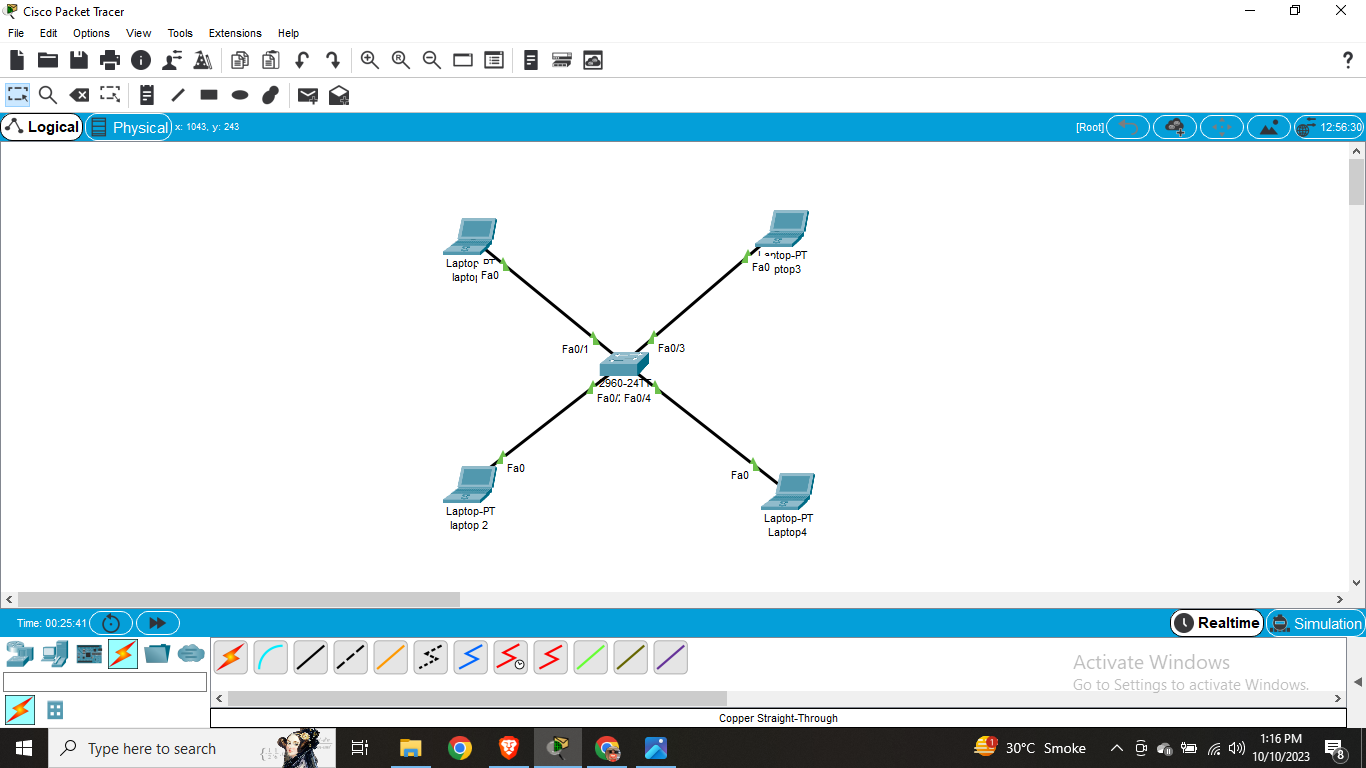


*openssl rsautl -decrypt -inkey private.key -in file.enc -out file.dec*



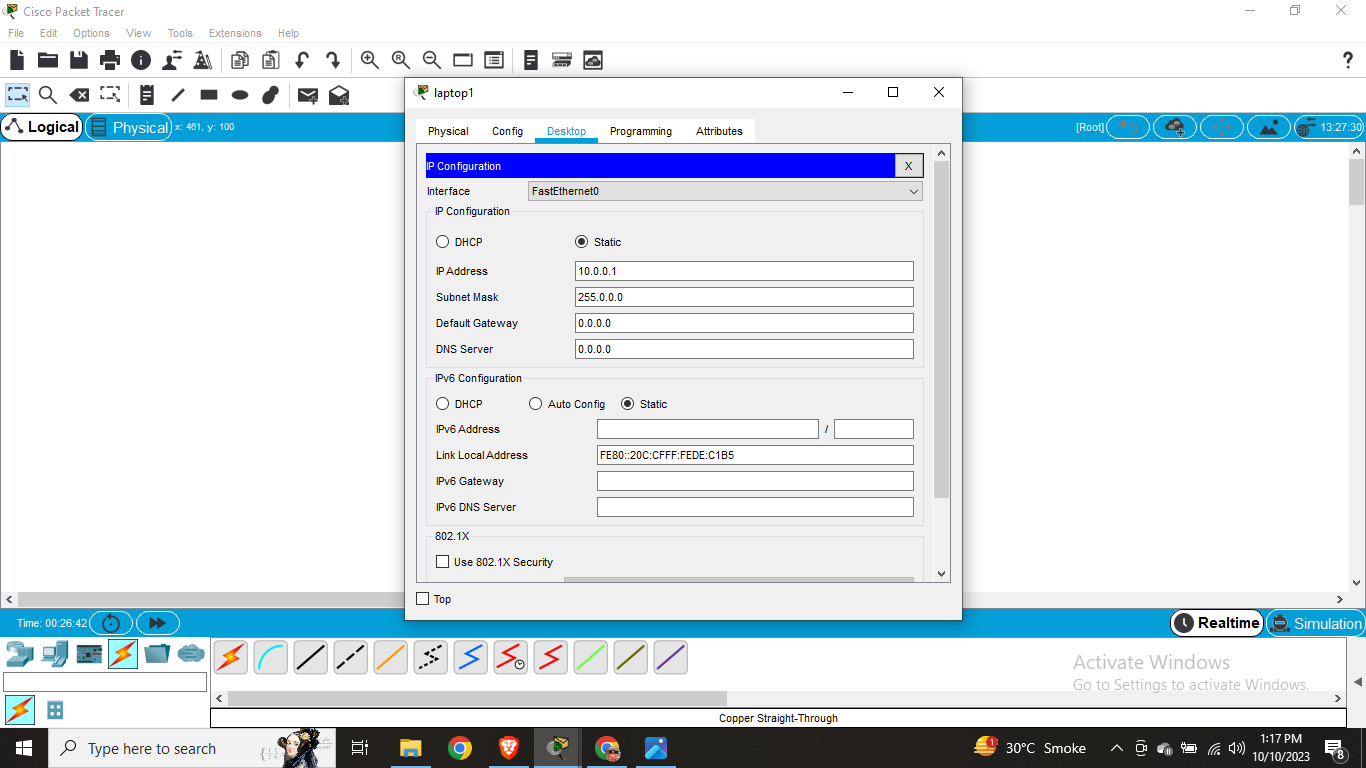
**Practical 3**

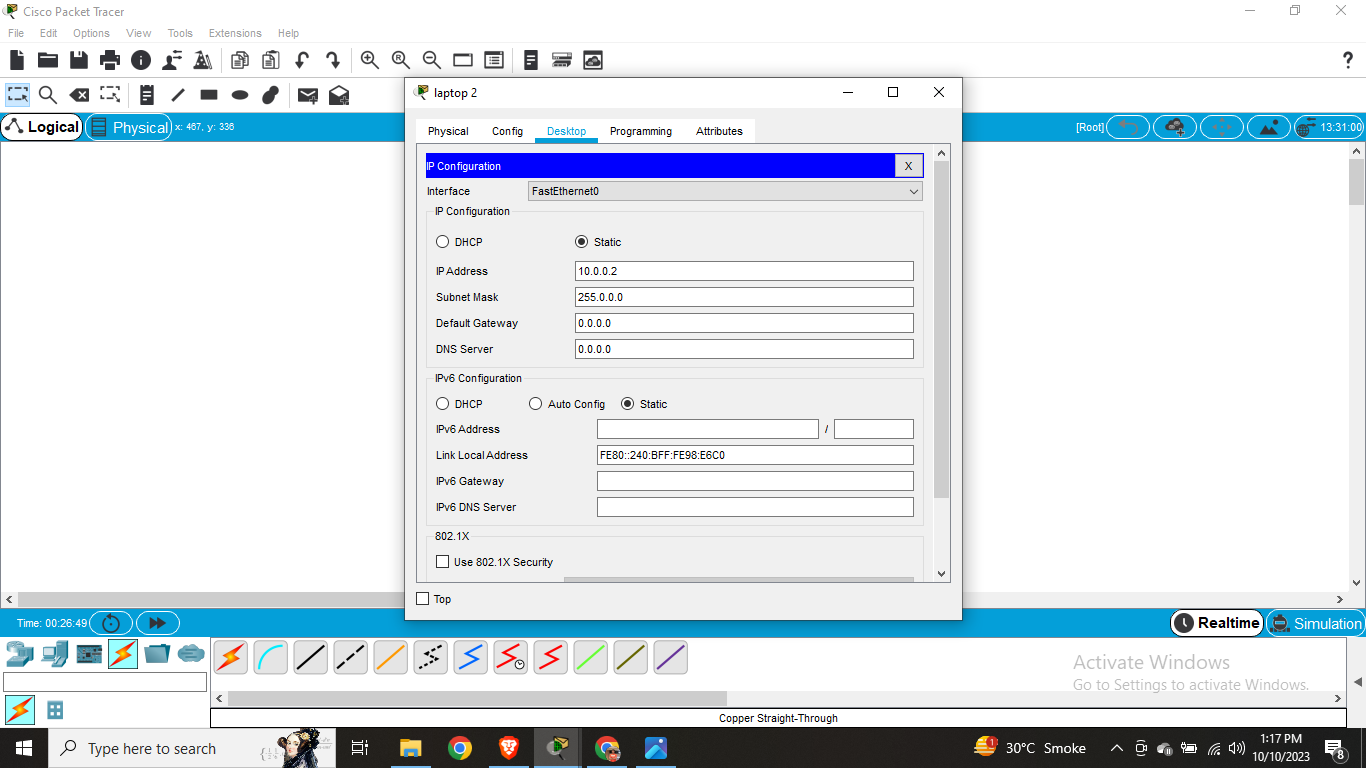
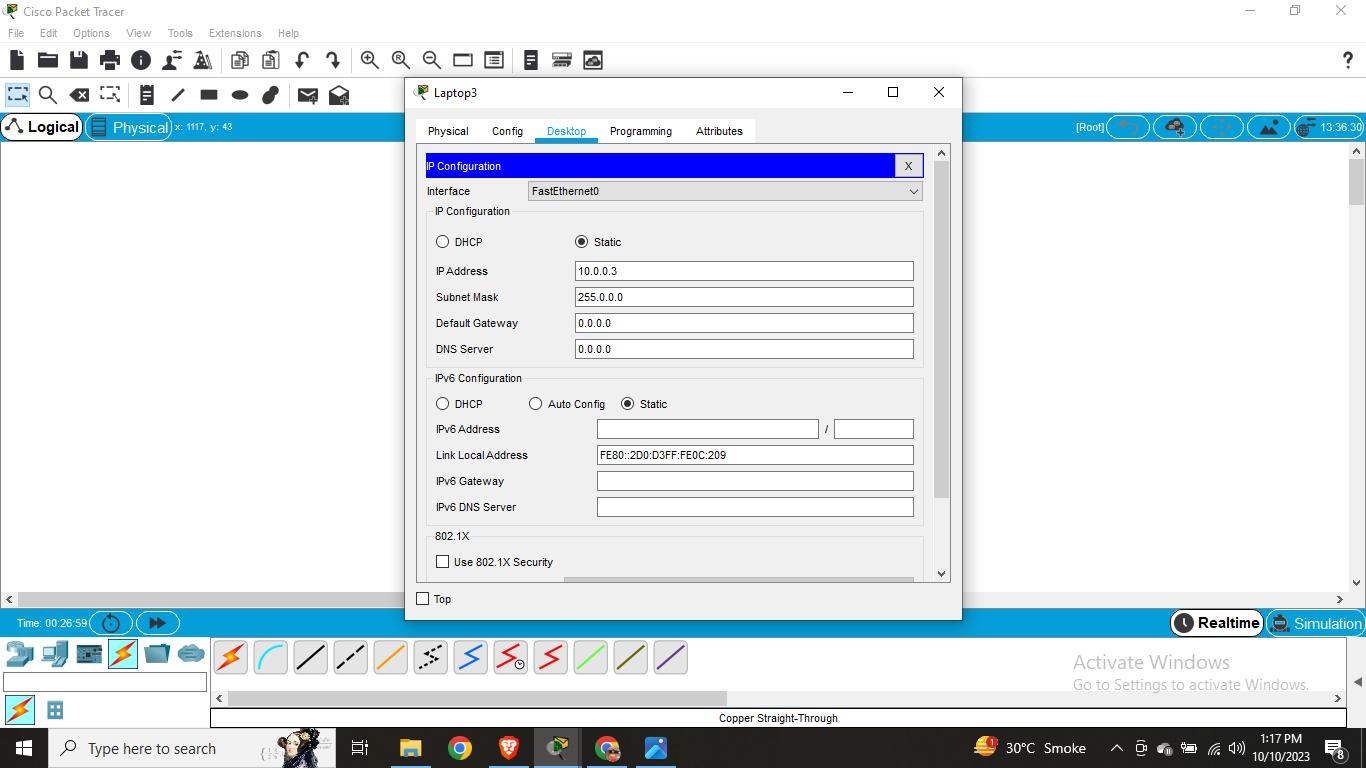
**Aim:** Creating vlan with single switch

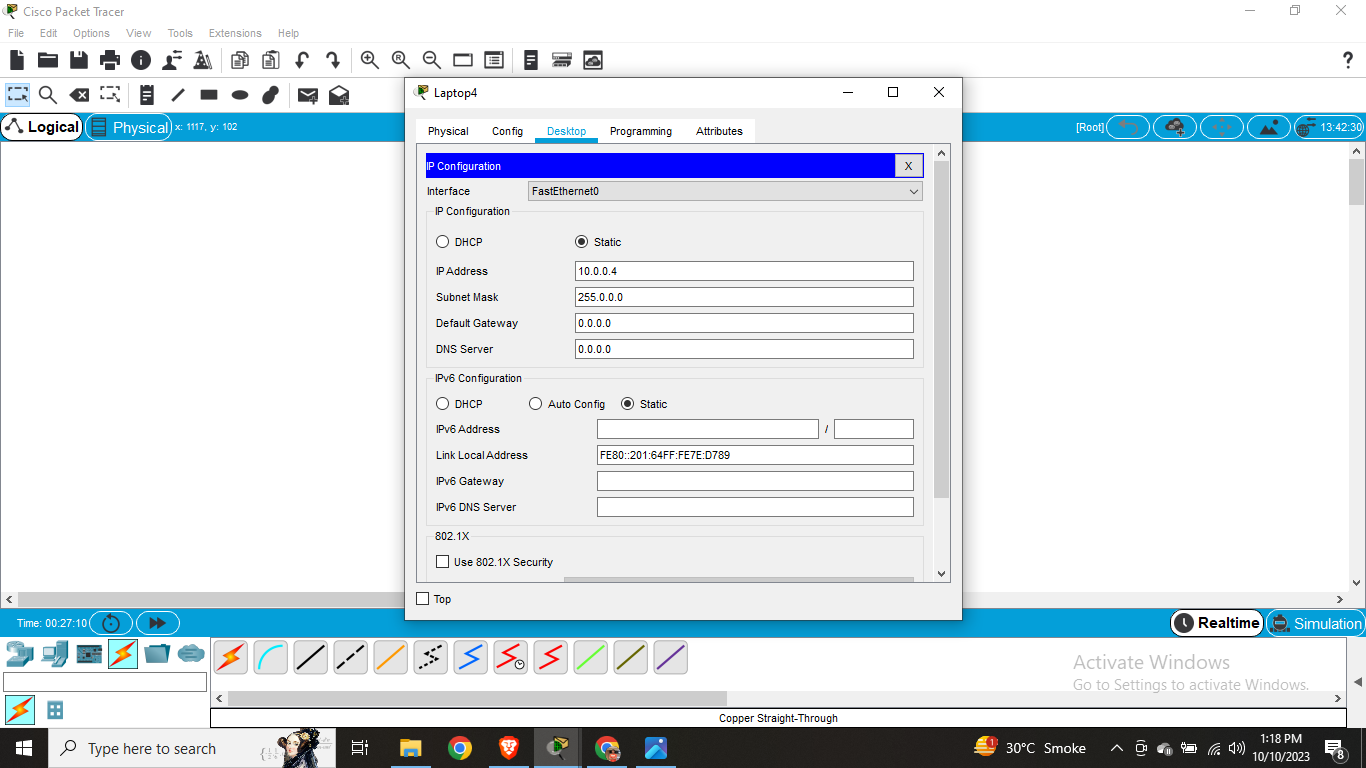


Step – 1

First we have to set ip to all laptops







Step - 2

Configuration commands of switch

Switch>ENable

Switch#CONF T

Enter configuration commands, one per line. End with CNTL/Z.

Switch(config)#vlan 10

Switch(config-vlan)#name upper

Switch(config-vlan)#exit

Switch(config)#vlan 20

Switch(config-vlan)#name lower

Switch(config-vlan)#exit

Switch>ENable

Switch#CONF T

Enter configuration commands, one per line. End with CNTL/Z.

Switch(config)#vlan 10

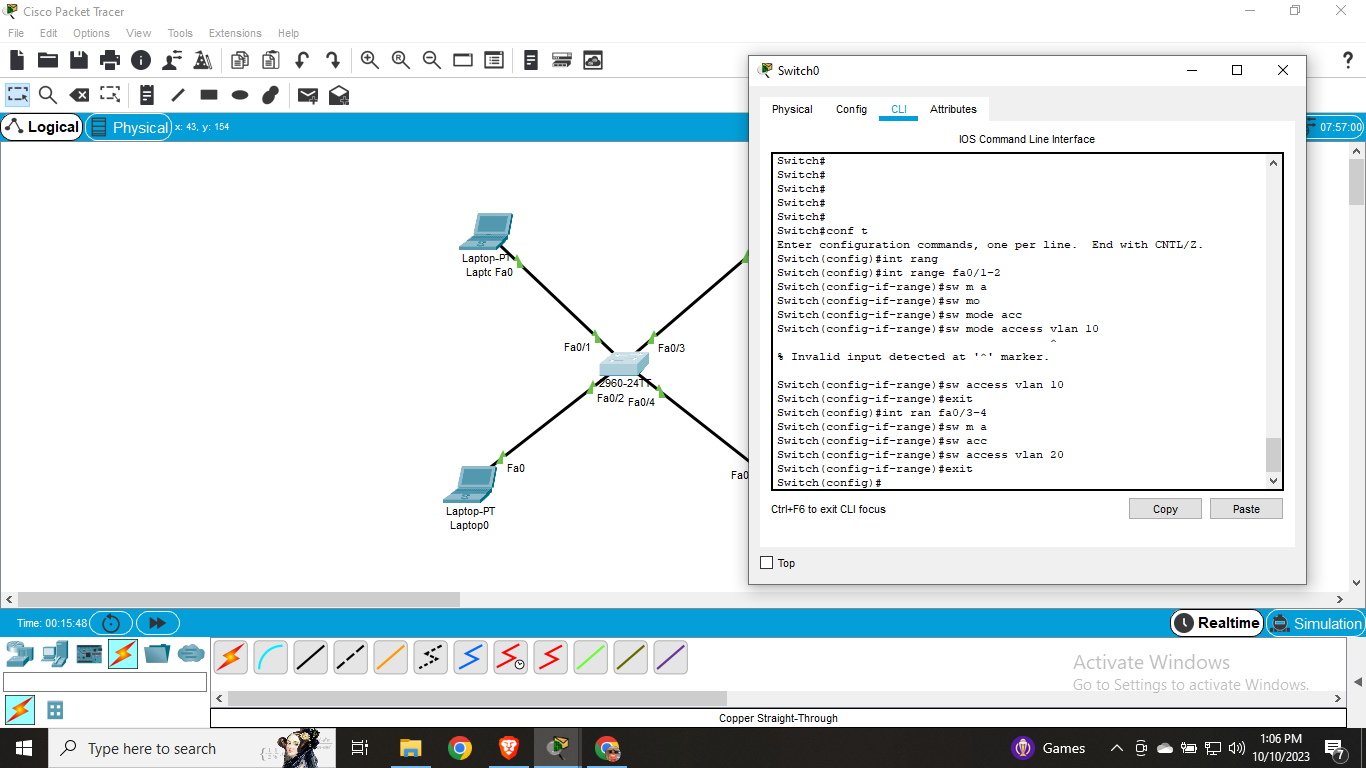
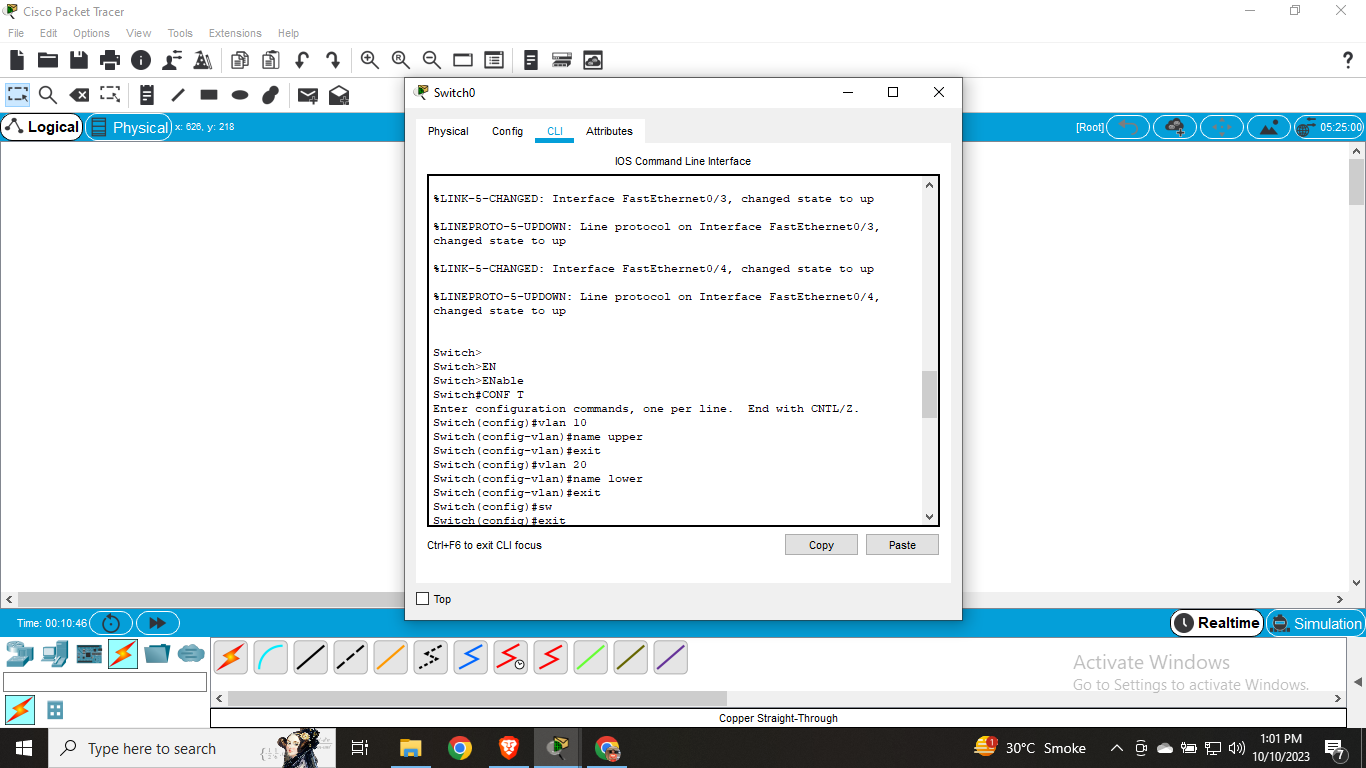
Switch(config-vlan)#name upper

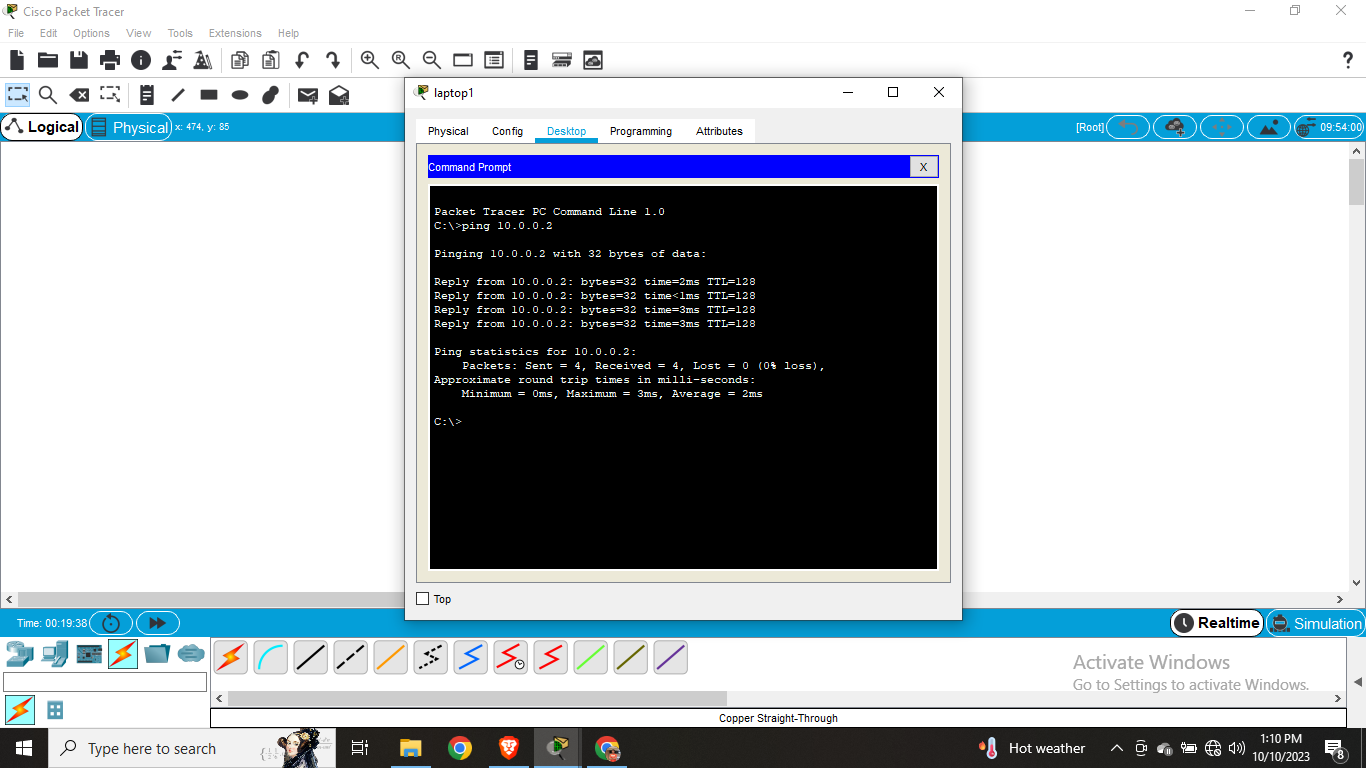
Switch(config-vlan)#exit

Switch(config)#vlan 20

Switch(config-vlan)#name lower

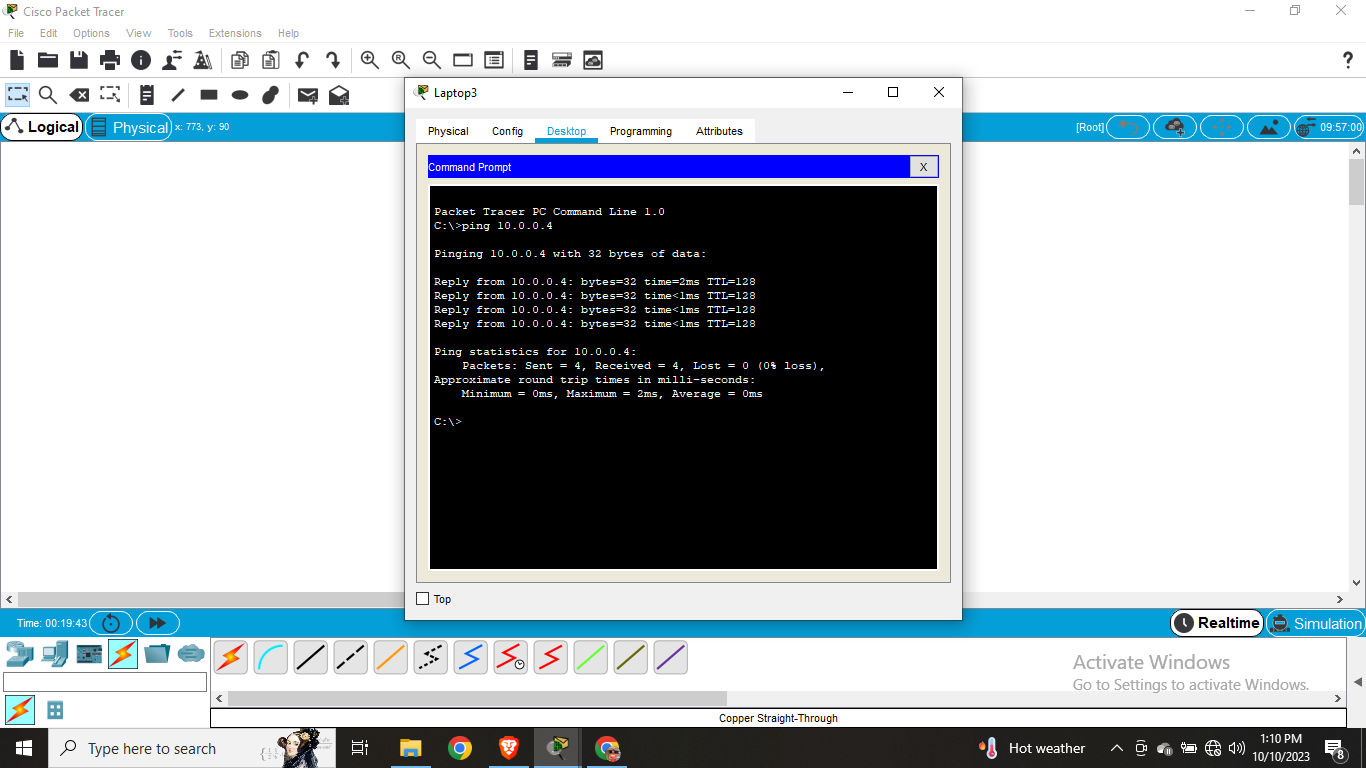
Switch(config-vlan)#exit



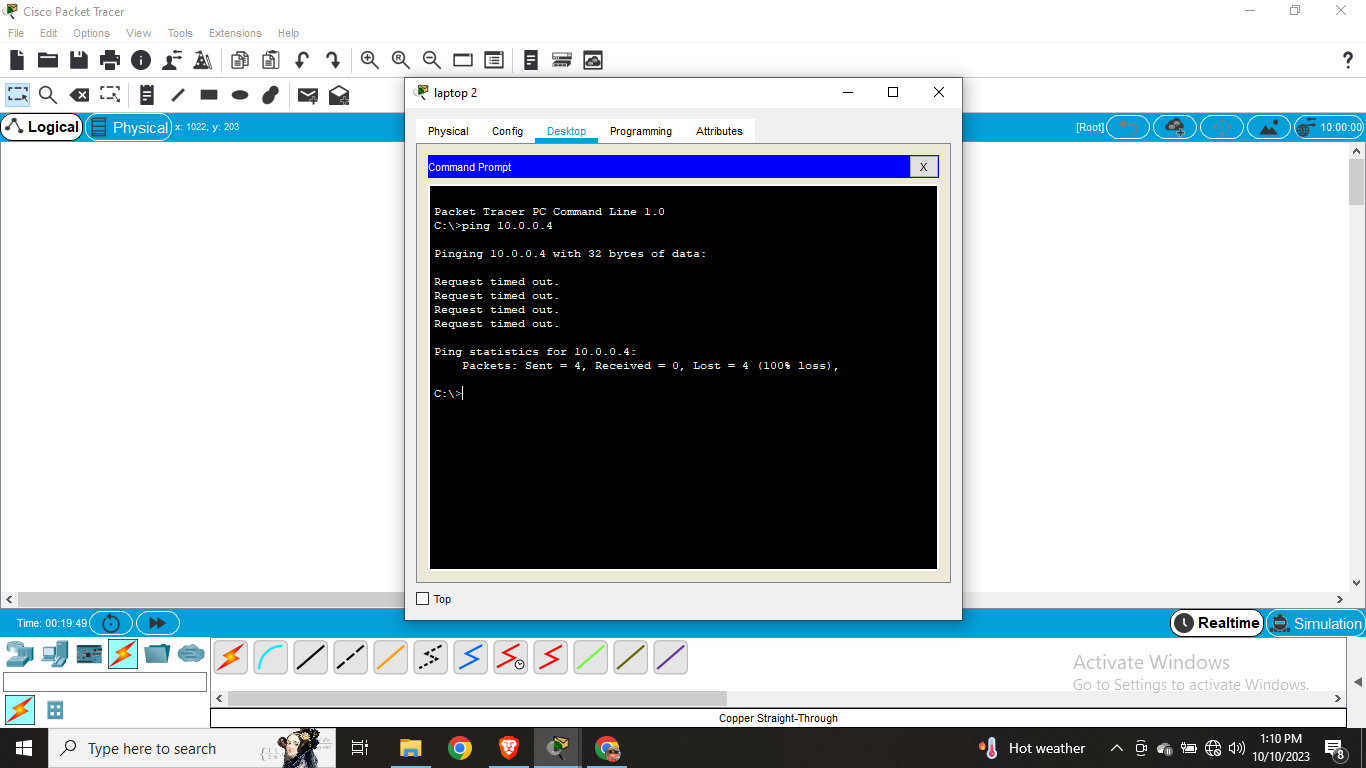


Ping pc1 to pc3

Ping pc2 to pc4

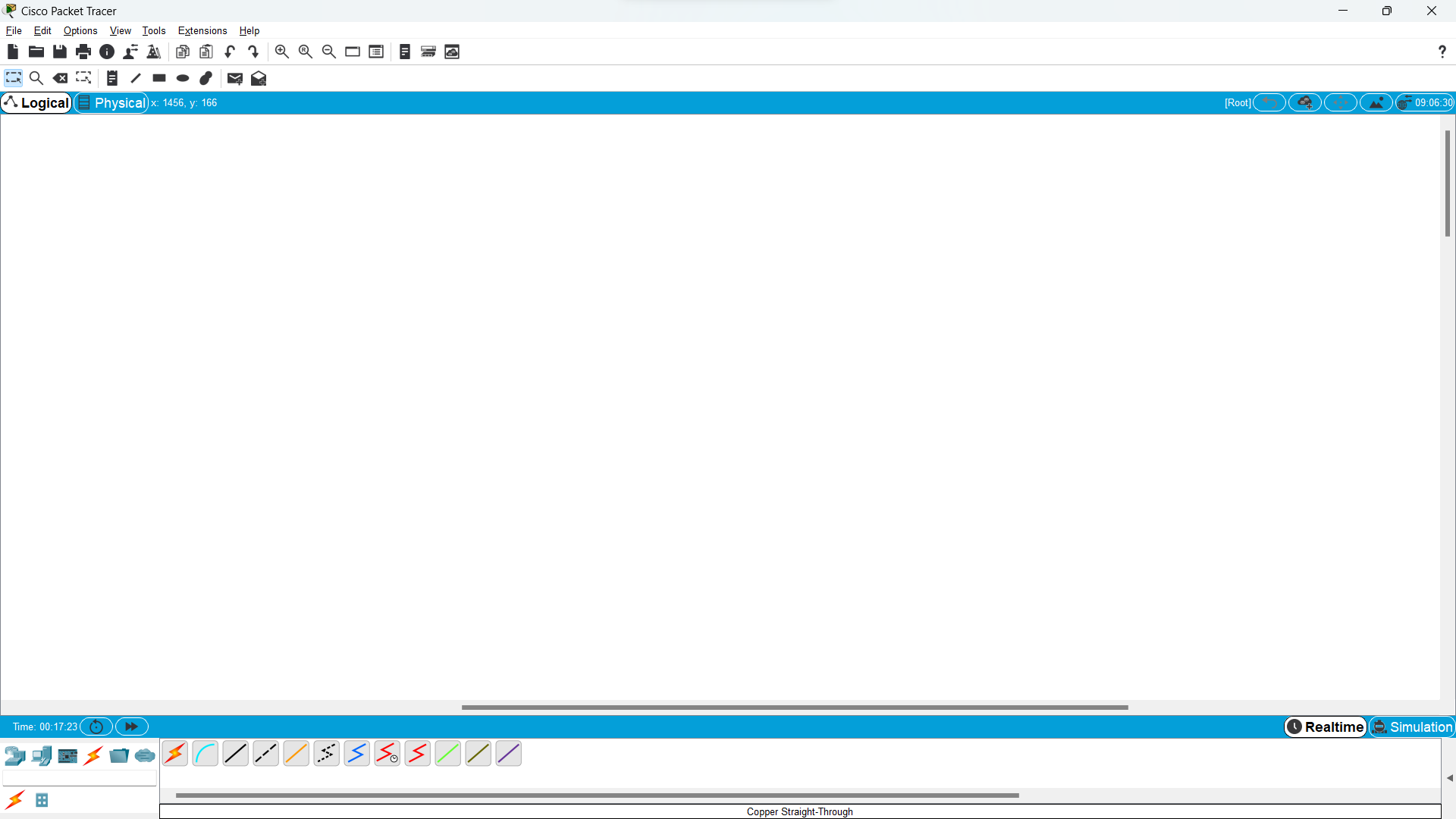


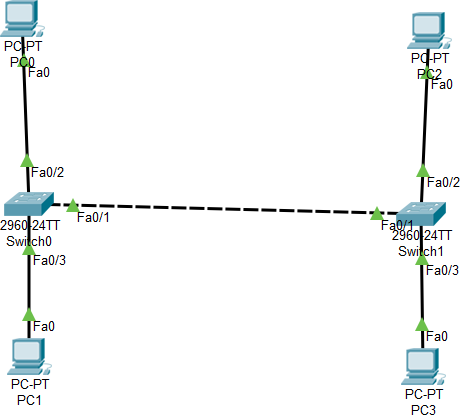
Ping pc1 to pc4

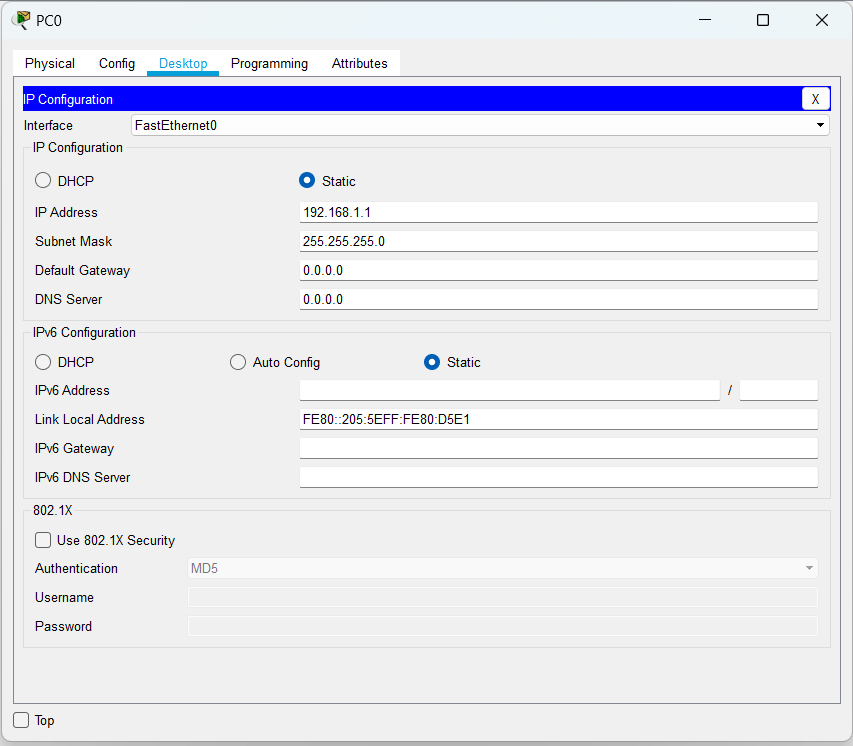


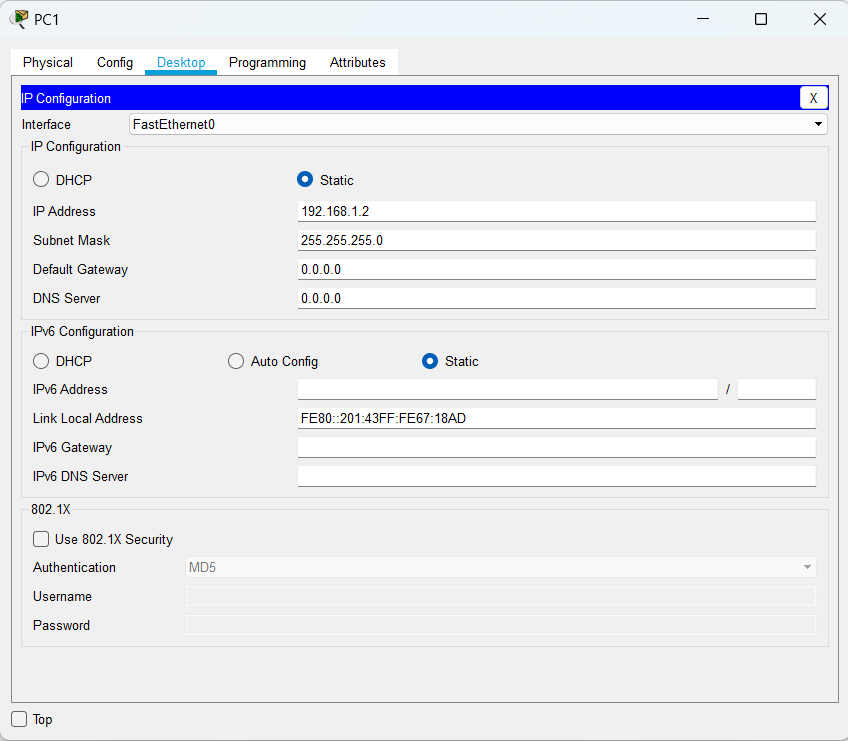
**Practical 4**

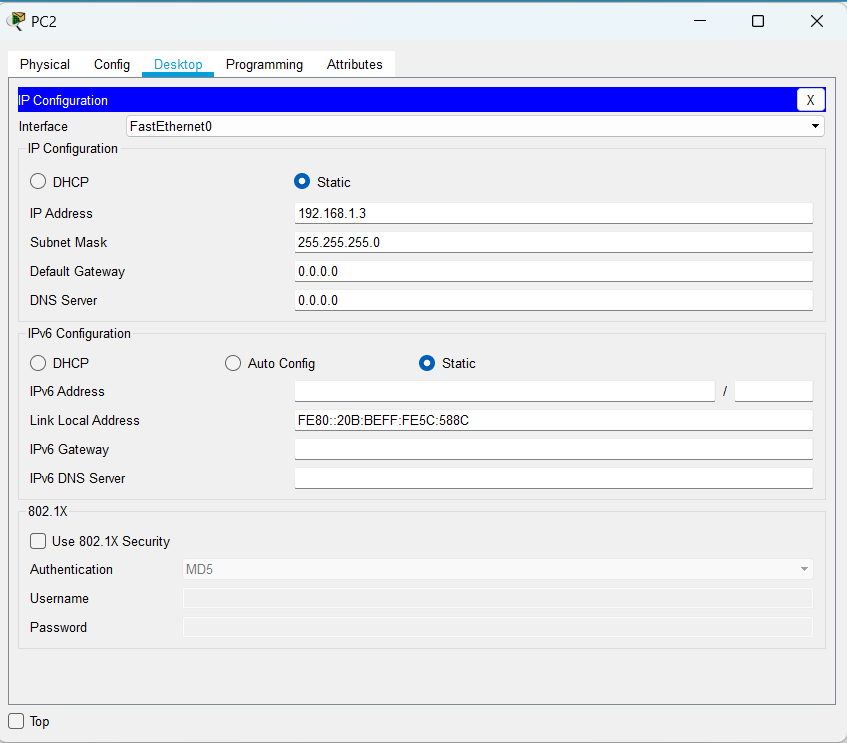
**Aim:** Creating vlan with double switch

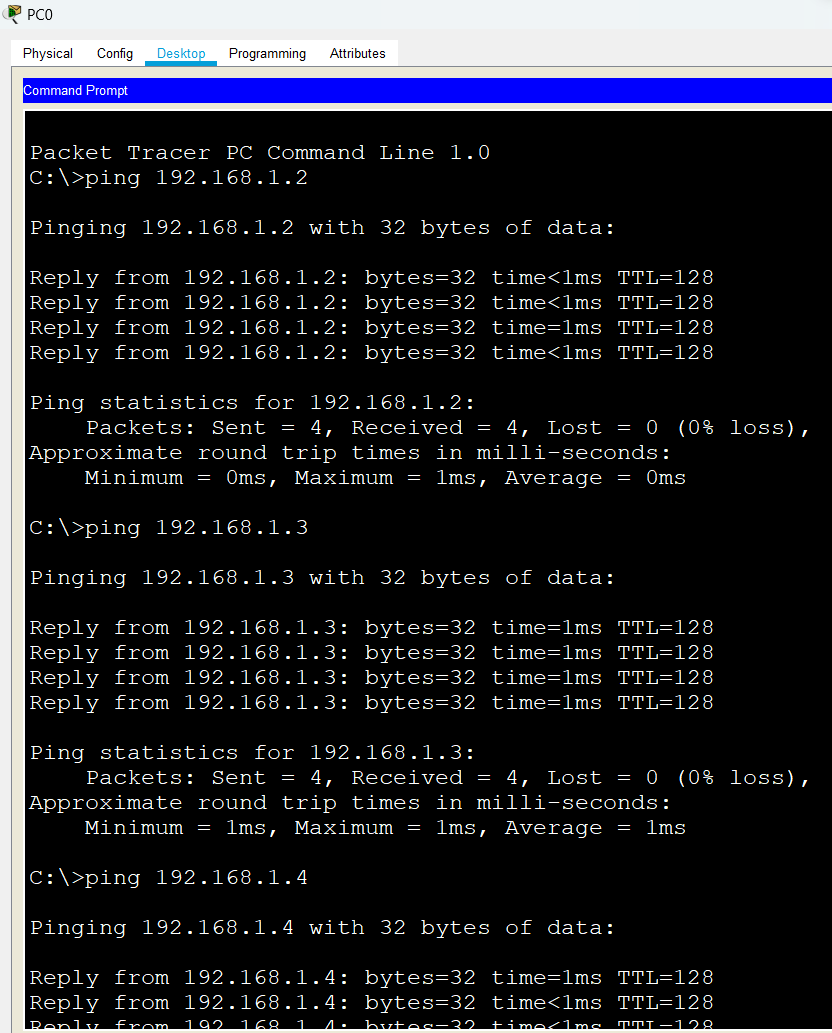












FOR SWITCH 00

Switch>en Switch#conf t

Enter configuration commands, one per line. End with CNTL/Z. Switch(config)#vlan 10

Switch(config-vlan)#name upper Switch(config-vlan)#vlan 20 Switch(config-vlan)#name lower Switch(config-vlan)#int fa0/2 Switch(config-if)#switchport mode access

Switch(config-if)#switchport access vlan 10 Switch(config-if)#int fa0/3

Switch(config-if)#switchport mode access Switch(config-if)#switchport access vlan 20 Switch(config-if)#int fa0/1

Switch(config-if)#switchport mode trunk Switch(config-if)#

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up Switch(config-if)#

Switch(config-if)#exit Switch(config)#interface FastEthernet0/1 Switch(config-if)#

Switch(config-if)#end Switch#vlan database

% Warning: It is recommended to configure VLAN from config mode, as VLAN database mode is being deprecated. Please consult user documentation for configuring VTP/VLAN in config mode.

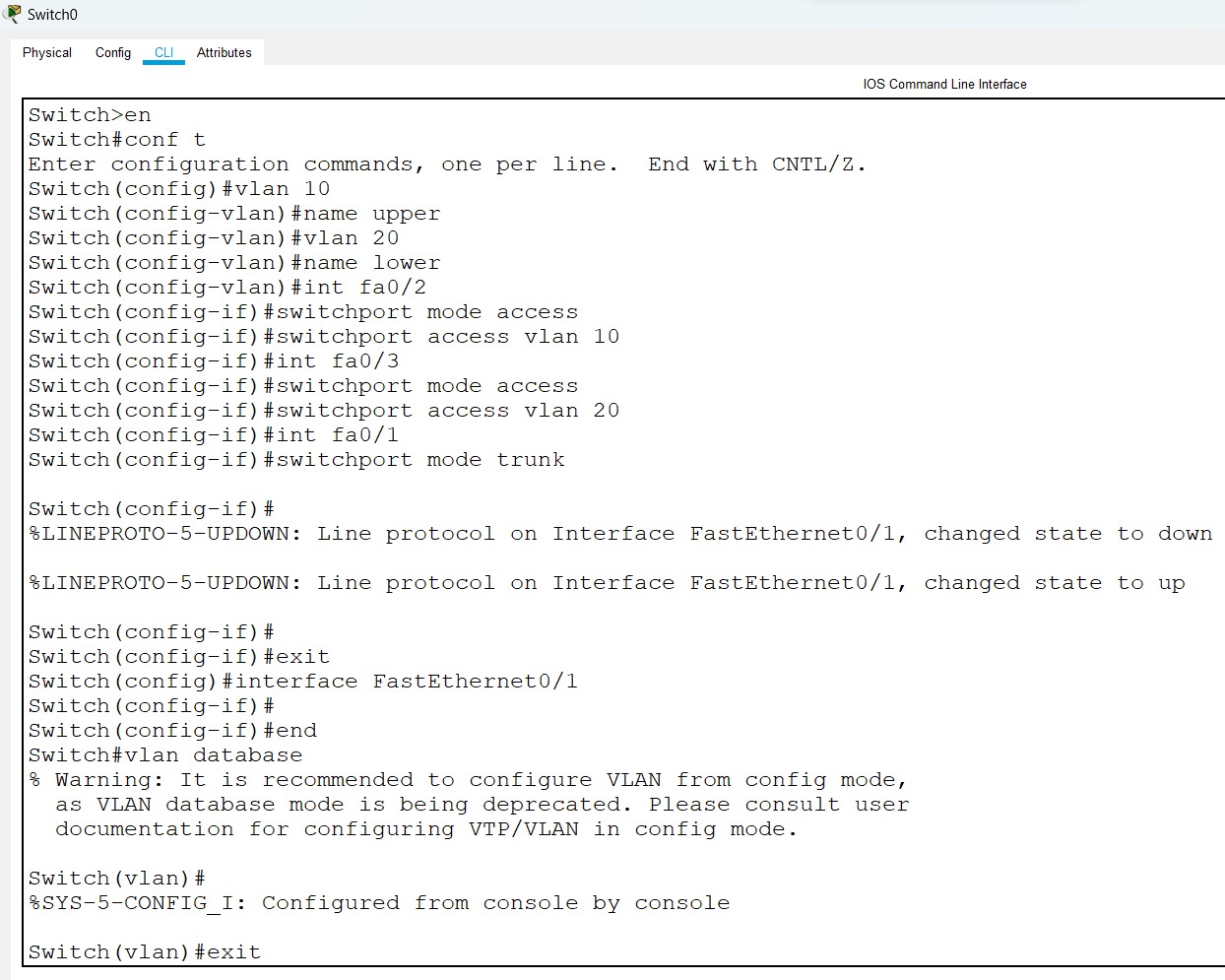
Switch(vlan)#

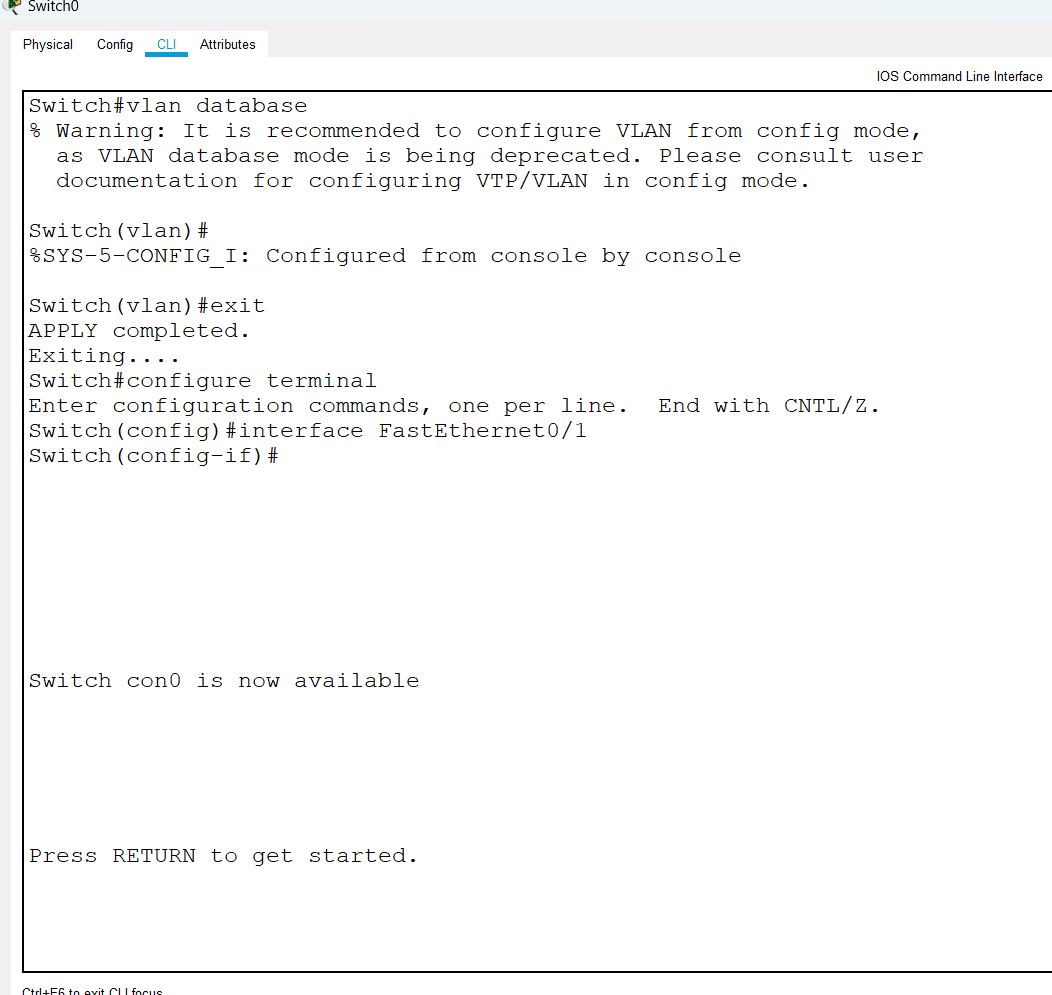
%SYS-5-CONFIG\_I: Configured from console by console Switch(vlan)#exit

APPLY completed. Exiting....

Switch#configure terminal

Enter configuration commands, one per line. End with CNTL/Z. Switch(config)#interface FastEthernet0/1





FOR SWITCH 01

Switch>en Switch#conf t

Enter configuration commands, one per line. End with CNTL/Z. Switch(config)#vlan 10

Switch(config-vlan)#name upper Switch(config-vlan)#

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to down

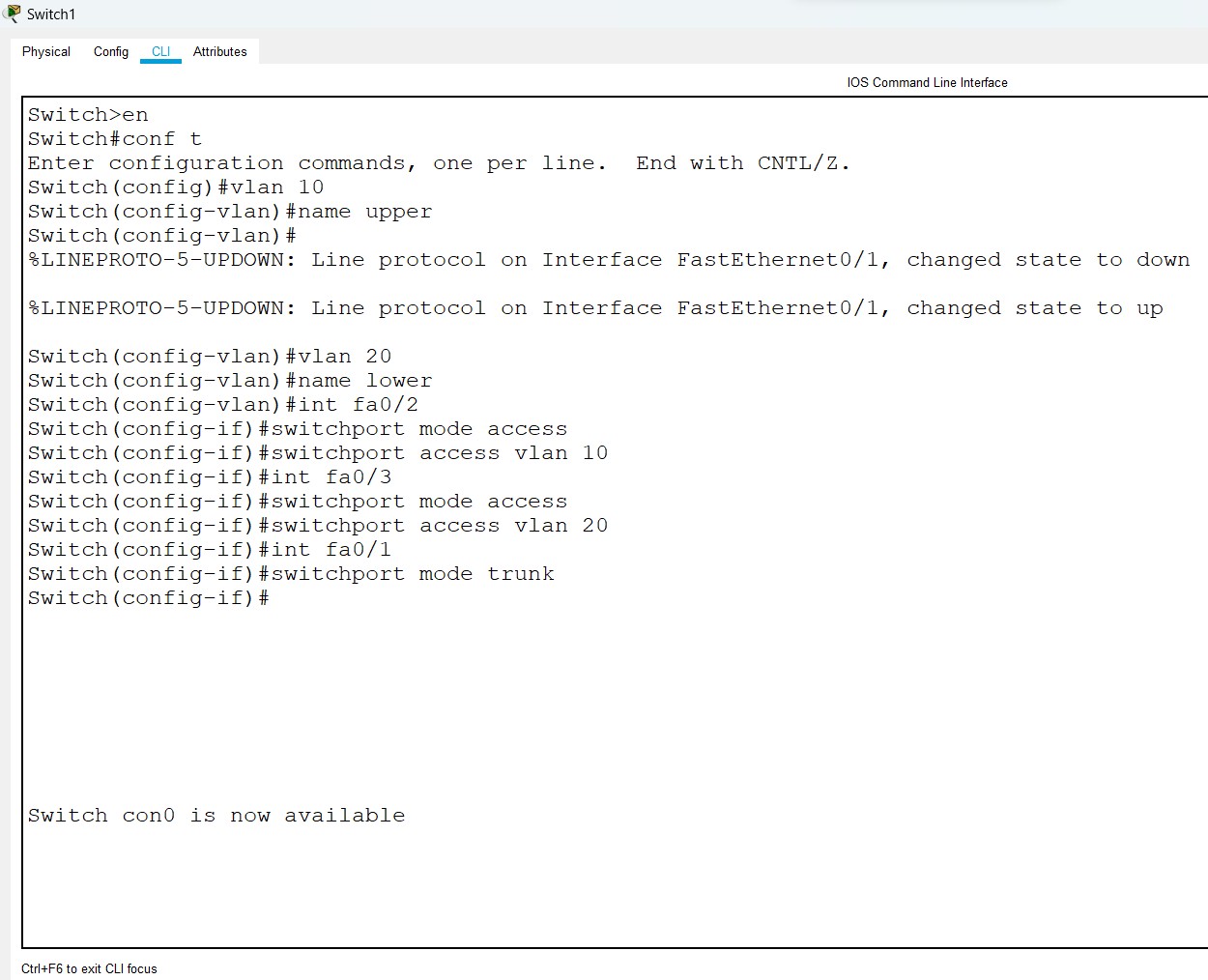
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up Switch(config-vlan)#vlan 20

Switch(config-vlan)#name lower Switch(config-vlan)#int fa0/2 Switch(config-if)#switchport mode access

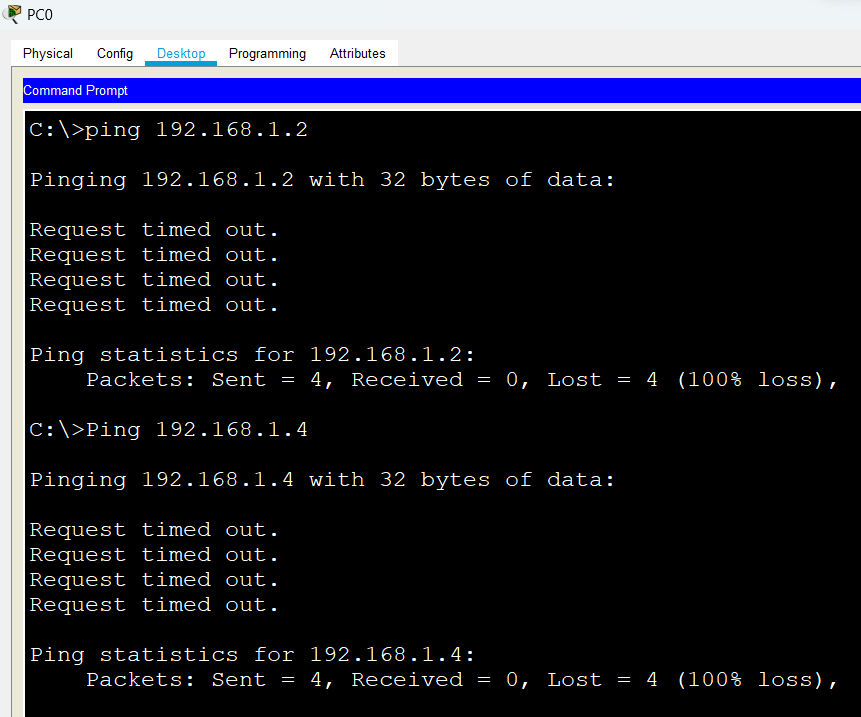
Switch(config-if)#switchport access vlan 10 Switch(config-if)#int fa0/3

Switch(config-if)#switchport mode access Switch(config-if)#switchport access vlan 20 Switch(config-if)#int fa0/1

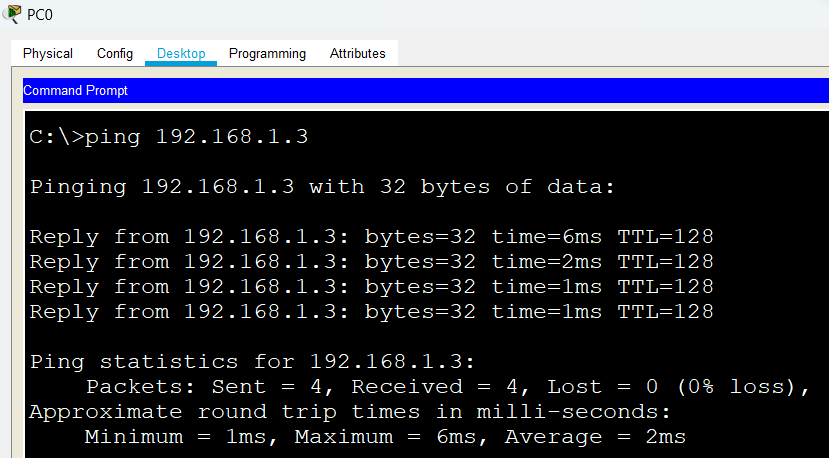
Switch(config-if)#switchport mode trunk



Trying to send packets from pc0 to pc1 and pc3 which are in vlan 20

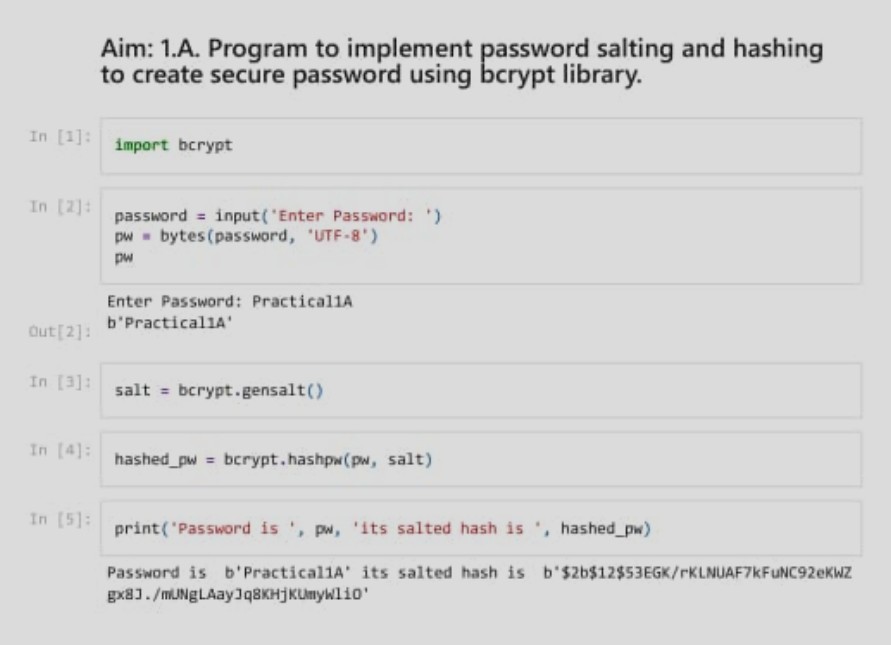


Sending packets to pc0 to pc2 which is vlan 10



**Practical 5**

Aim - Program to implement password salting and hashing to create secure password using bcrypt library.



Code :

!pip install bcrypt import bcrypt

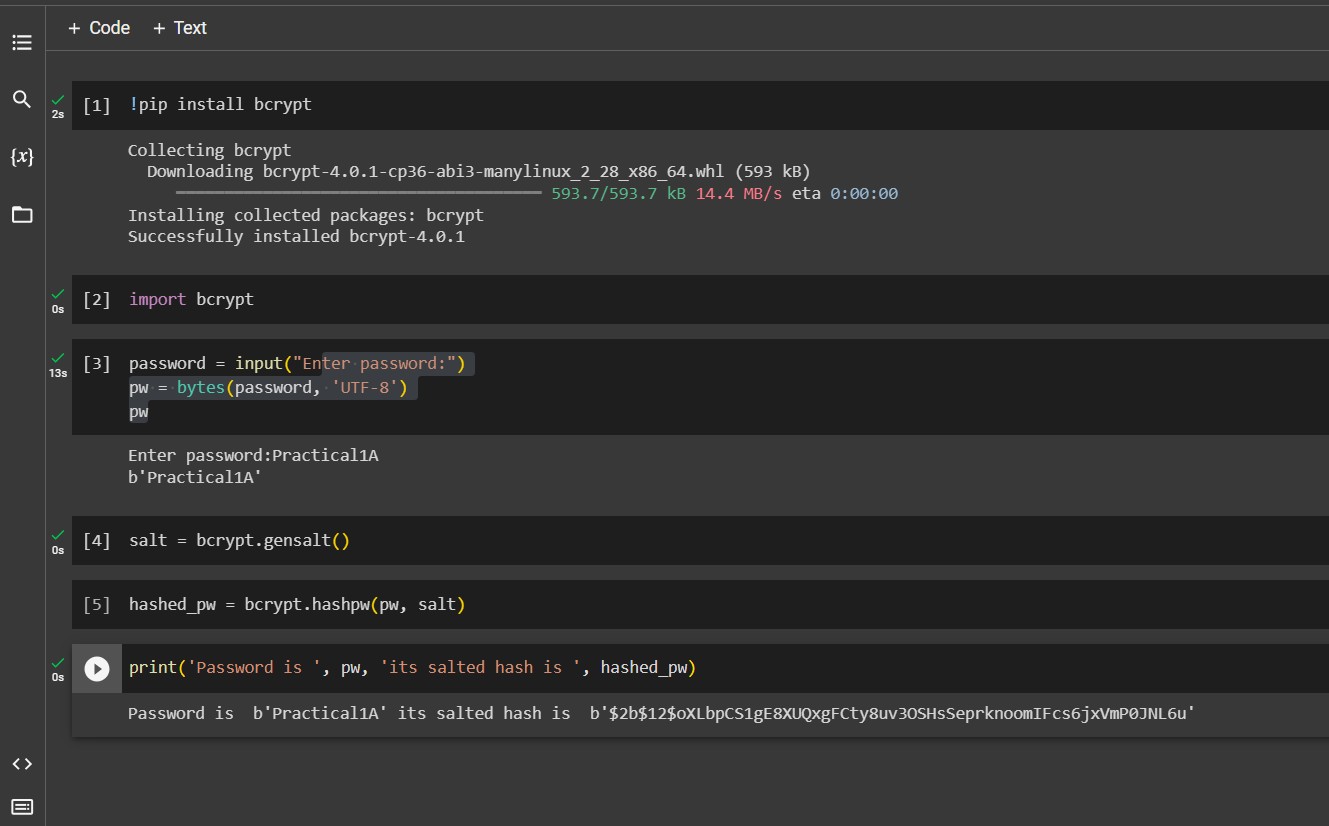
password = input("Enter password:") pw = bytes(password, 'UTF-8')

pw

salt = bcrypt.gensalt()

hashed\_pw = bcrypt.hashpw(pw, salt)

print('Password is ', pw, 'its salted hash is ', hashed\_pw)

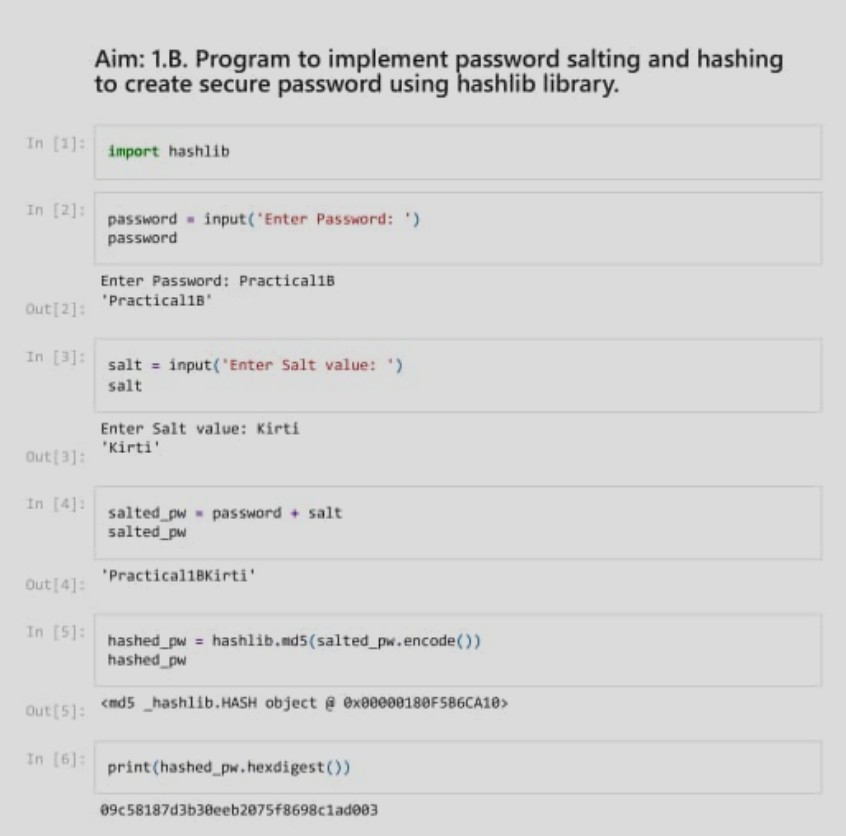
Output :

# Password is b'Practical1A' its salted hash is

**b'$2b$12$oXLbpCS1gE8XUQxgFCty8uv3OSHsSeprknoomIFcs6jxVmP0JNL6u'**

**Practical 6**

Aim: Program to implement password salting and hashing to create secure password using hashlib library.



Code :

import hashlib

password = input('Enter Password: ')

Password

salt = input('Enter Salt value: ')

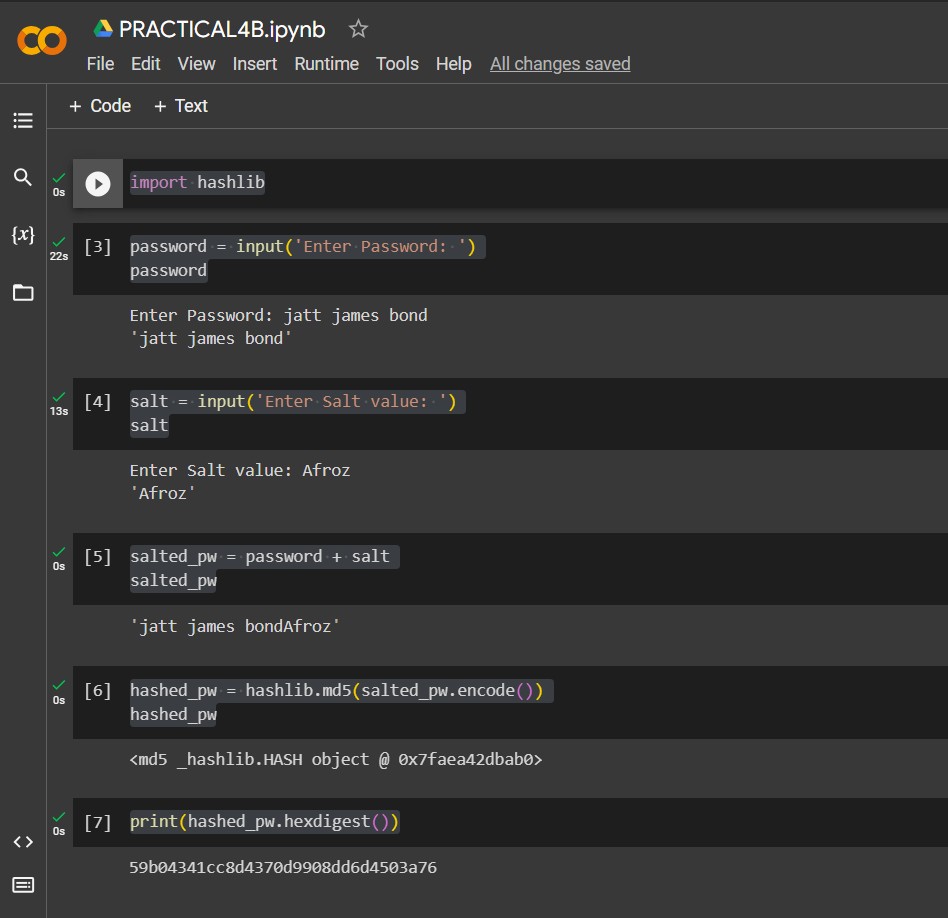
Salt

salted\_pw = password + salt Salted\_pw

hashed\_pw = hashlib.md5(salted\_pw.encode())

Hashed\_pw

print(hashed\_pw.hexdigest())

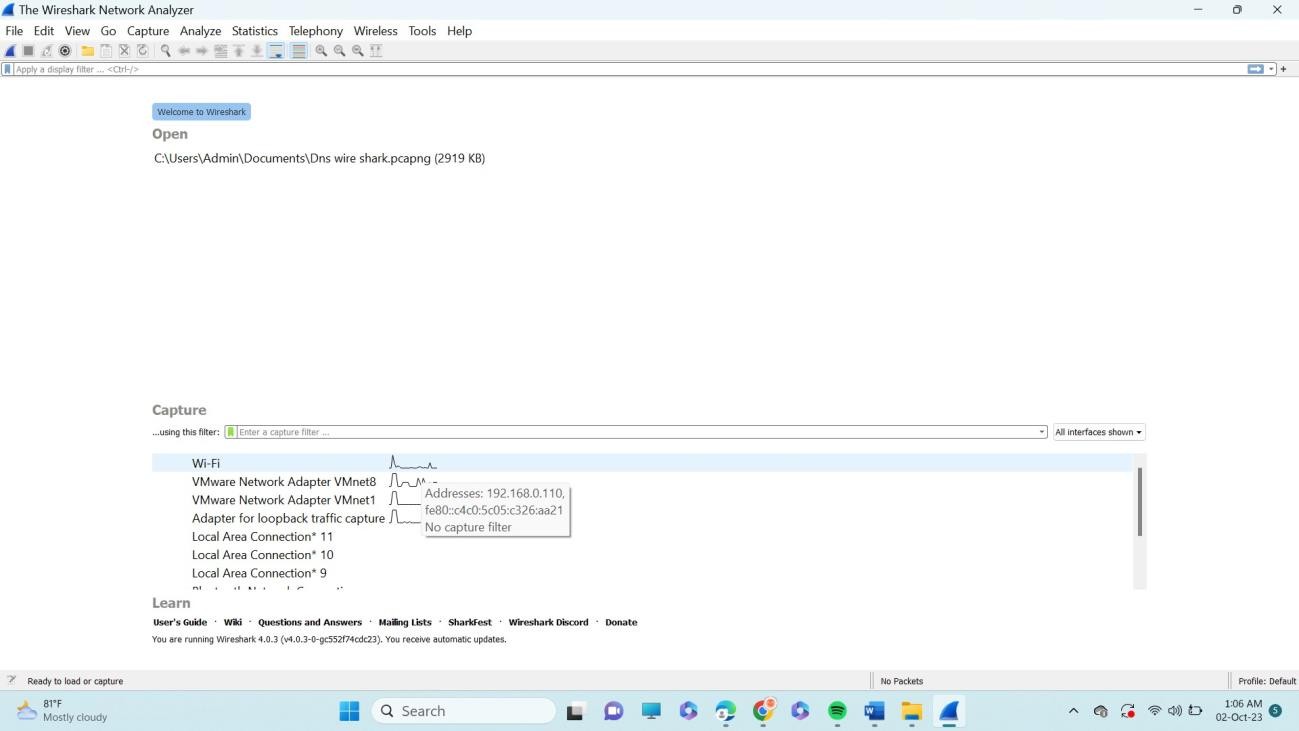
Output:

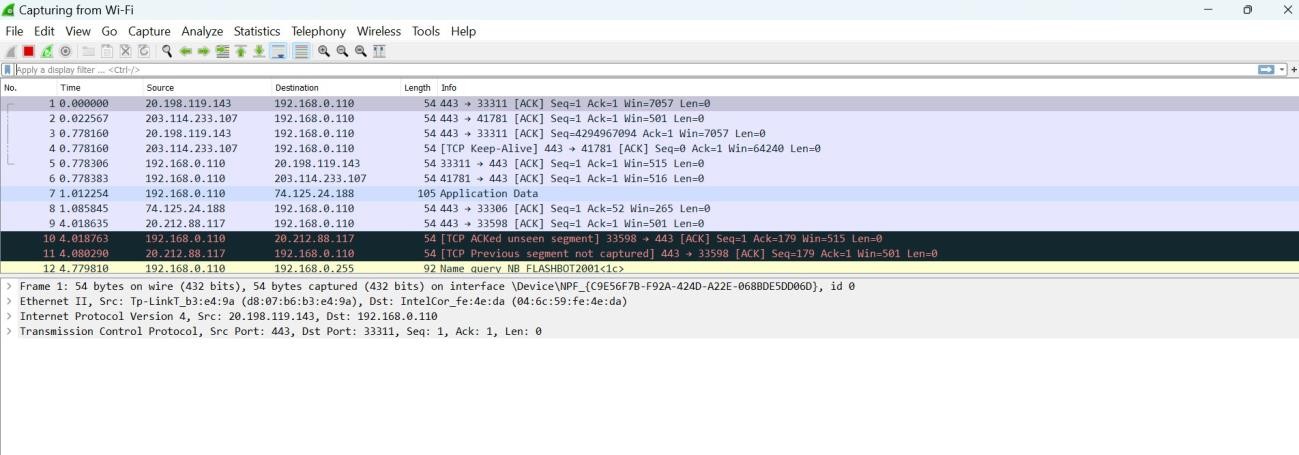
# 59b04341cc8d4370d9908dd6d4503a76

**Practical 7**

Aim : Analyzing DNS with wireshark.

1.) Open wire shark , click at wifi then you will see the data packets start capturing.



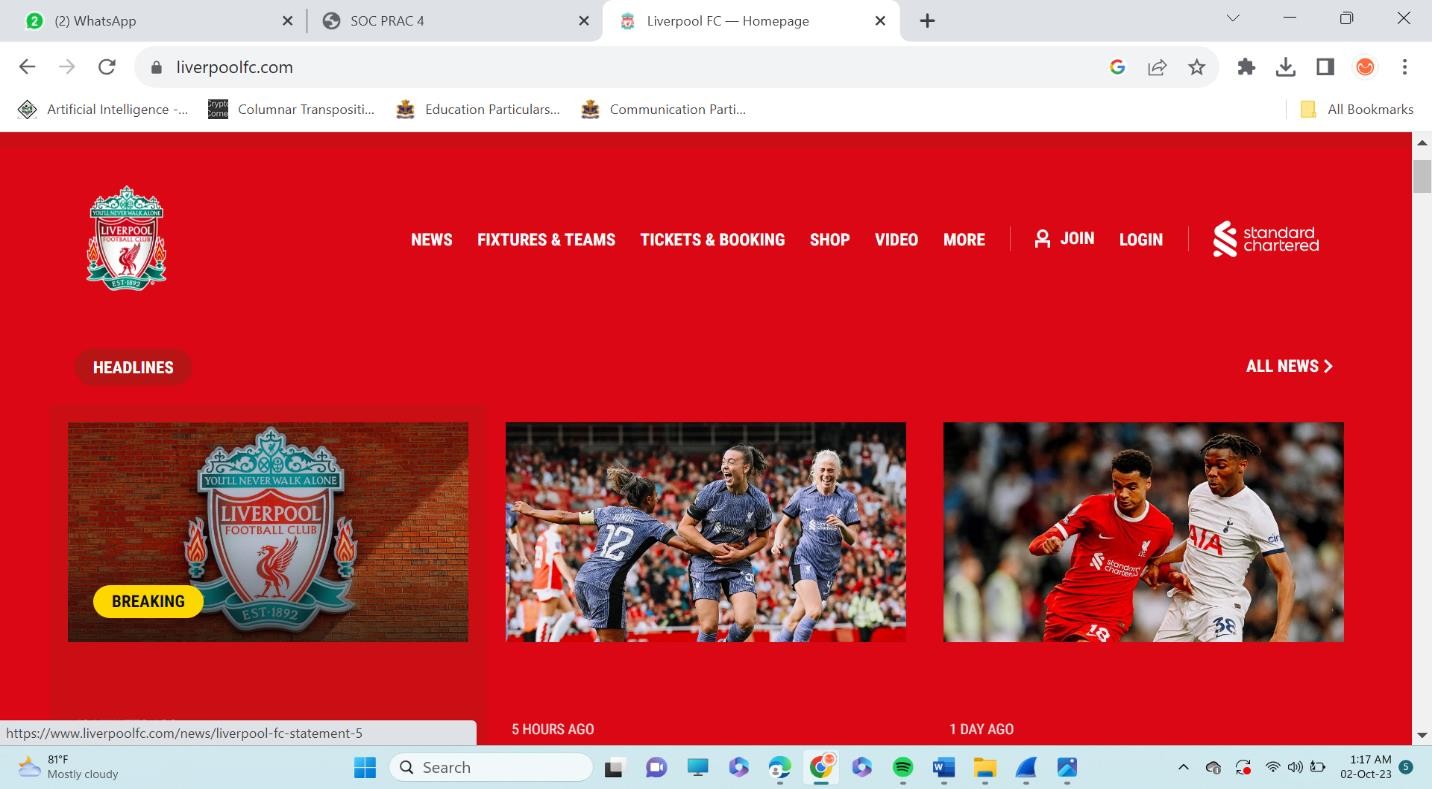


2.) Select capture  Capture filter  Change the port into port 53 at capture filters.

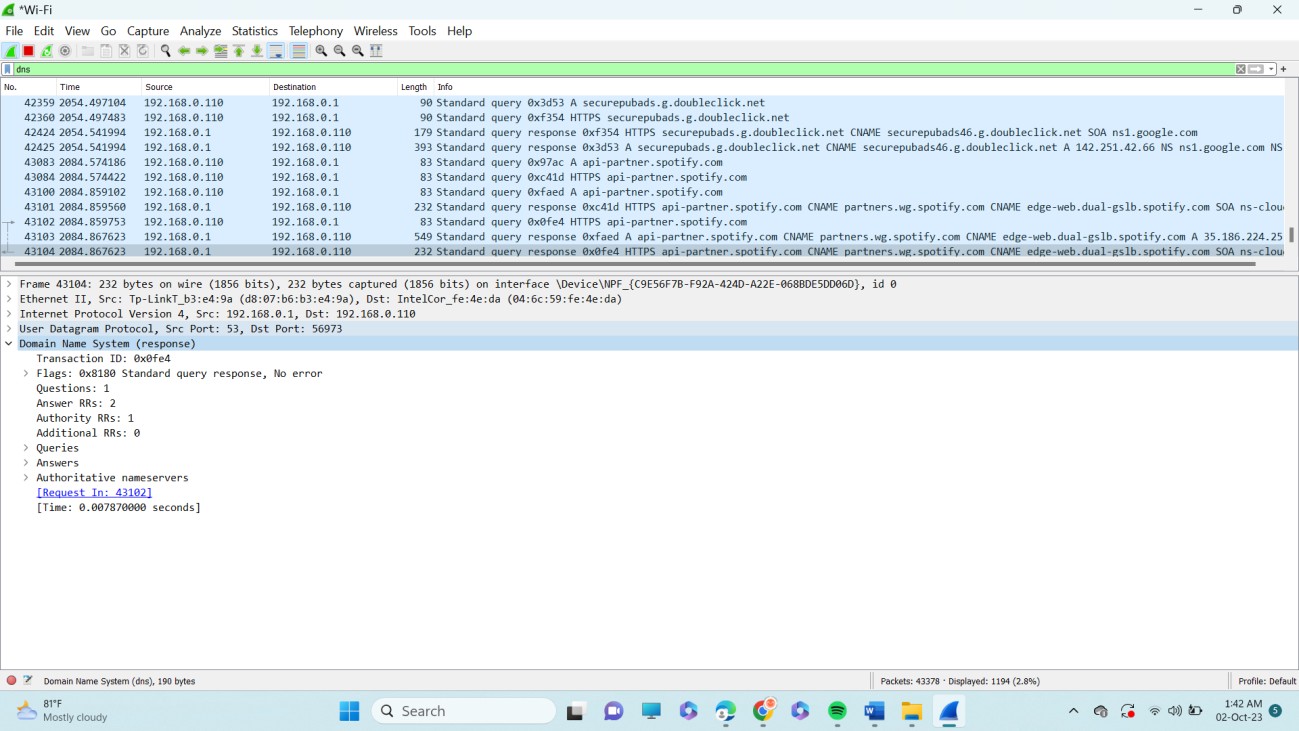


3.) Then search any topic in your chrome. Let’s see here we have searched at a site. While data packets are still

capturing.



4.) Apply display filter and type dns you will see the output at domain name system .

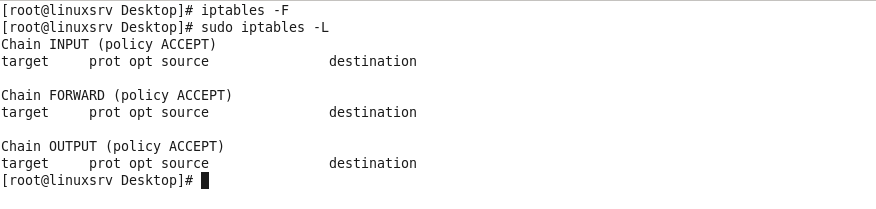


**Practical 8**

**Aim:** Using IP TABLES on linux and setting filtering rules

1. Open redhat terminal run a command to check terminal details.

**Command 1:** sudo iptables -L

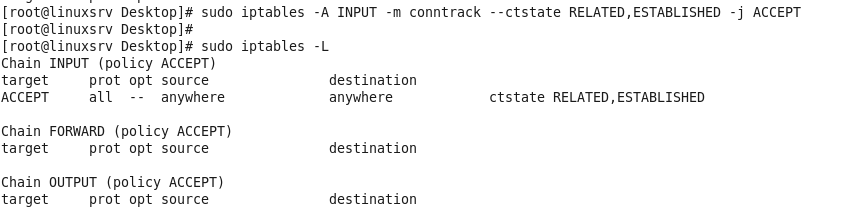


**Note:** If all conncetions are done already flush it using commnad "iptables -F".

2. Allowing Established session.

**Command 2:** sudo iptables -A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT

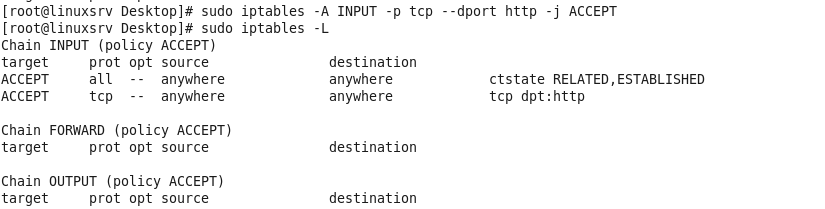
**To check output:** sudo iptables -L



3. Allowing incoming http and ssh requests:

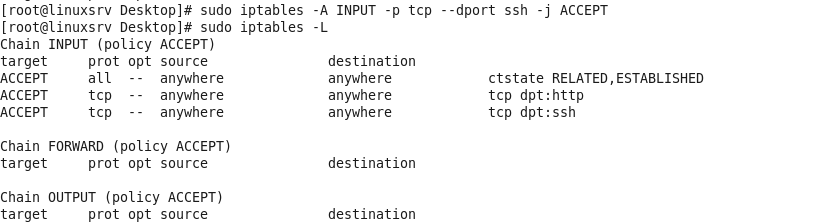
**Command 3:** sudo iptables -A INPUT -p tcp --dport http -j ACCEPT

**To check output:** sudo iptables -L



**Command 4:** sudo iptables -A INPUT -p tcp --dport ssh -j ACCEPT

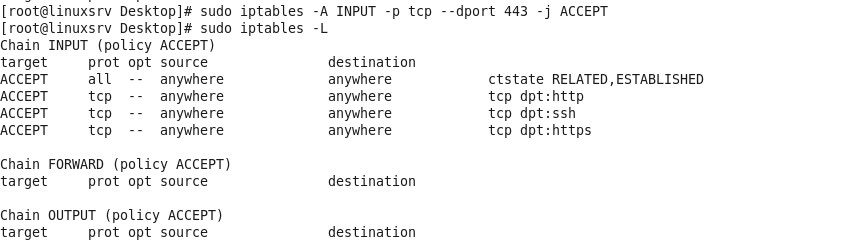
**To check output:** sudo iptables -L



4. Allowing incoming traffic on sprcific port 443

**Command 5**: sudo iptables -A INPUT -p tcp --dport 443 -j ACCEPT

**To check output:** sudo iptables -L



5. Blocking traffic for machine

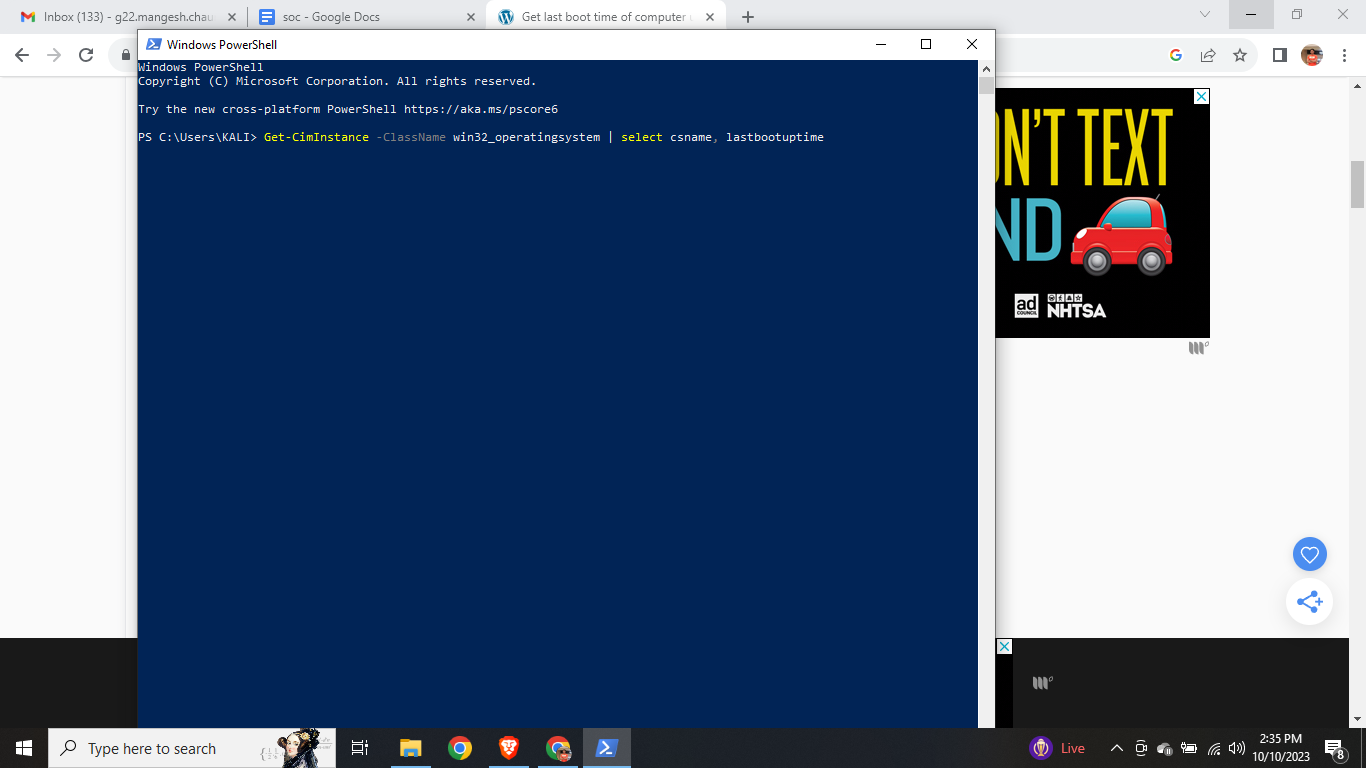
**Command6:** sudo iptables -A INPUT -j DROP

**To check output:** sudo iptables -L



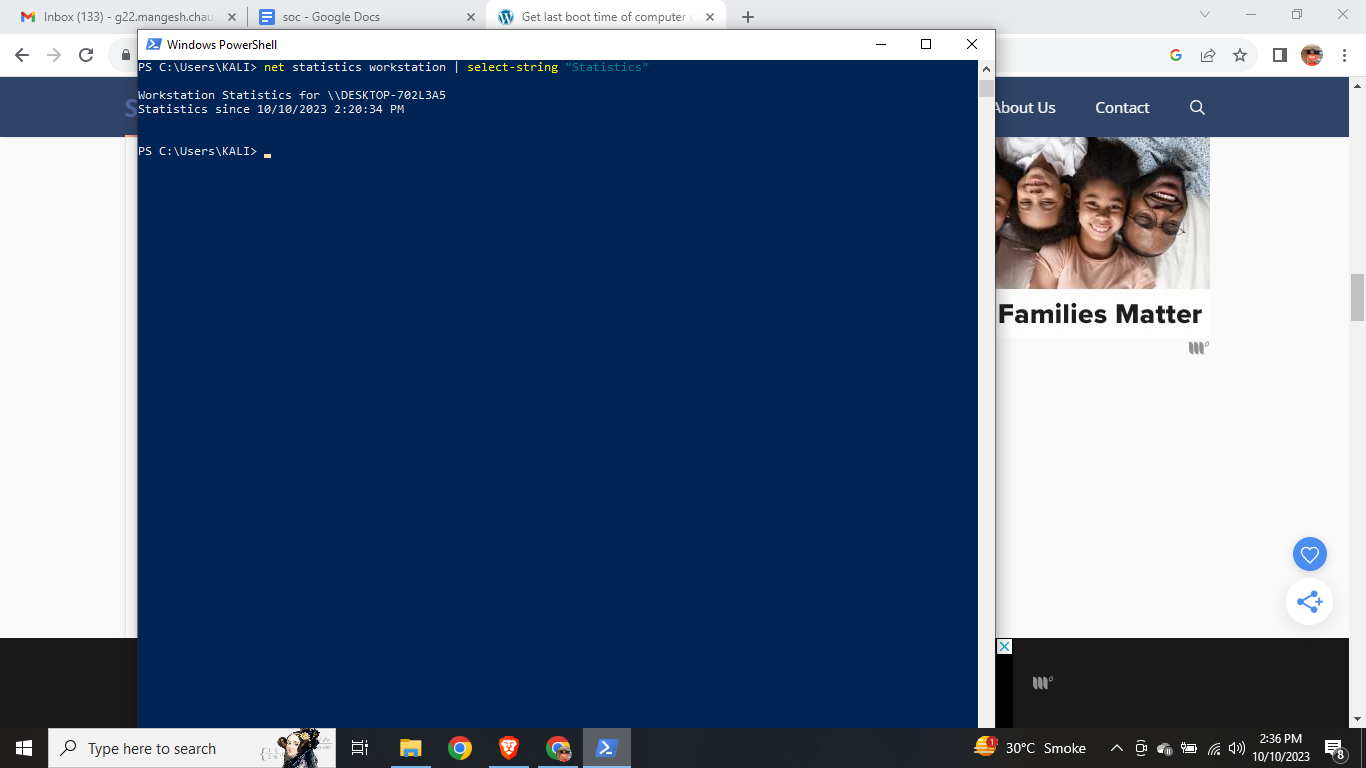
**Practical 9**

Get-CimInstance -ClassName win32\_operatingsystem | select csname, lastbootuptime

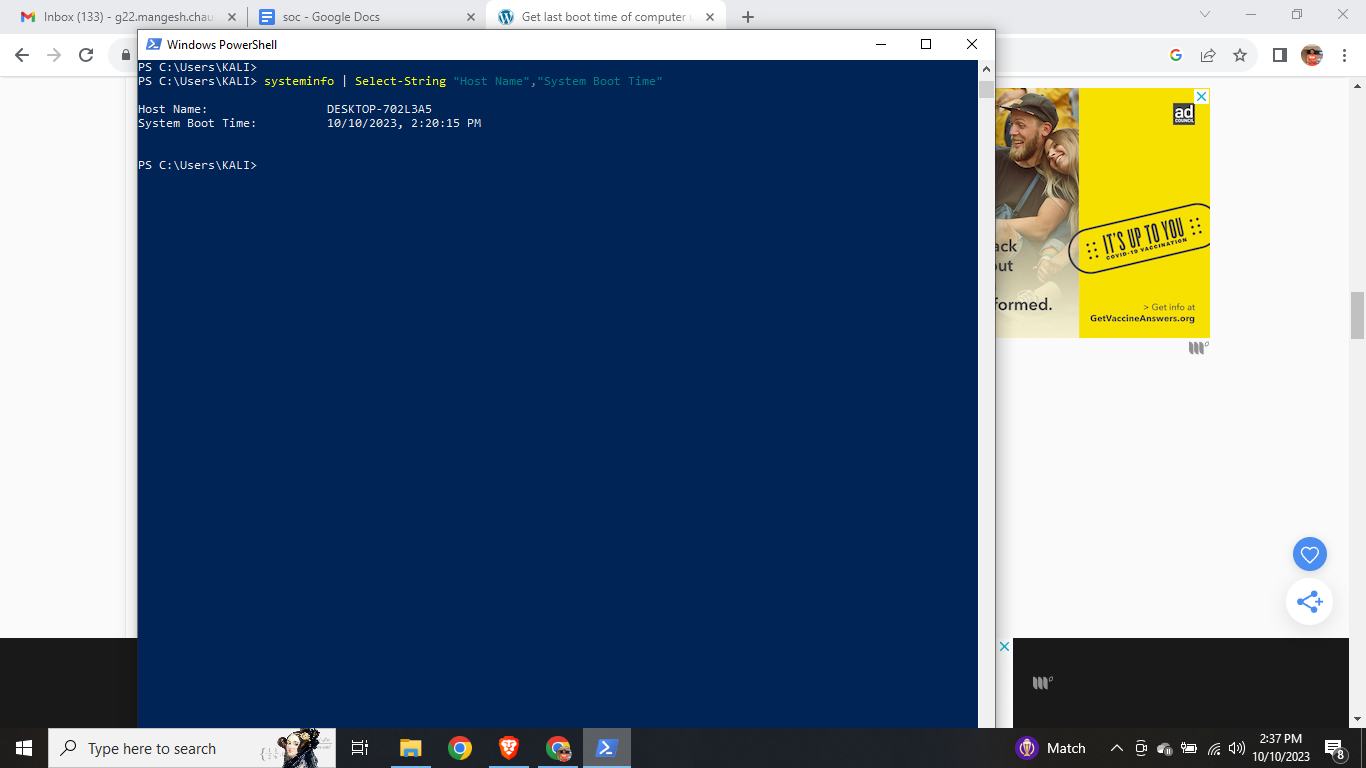
****



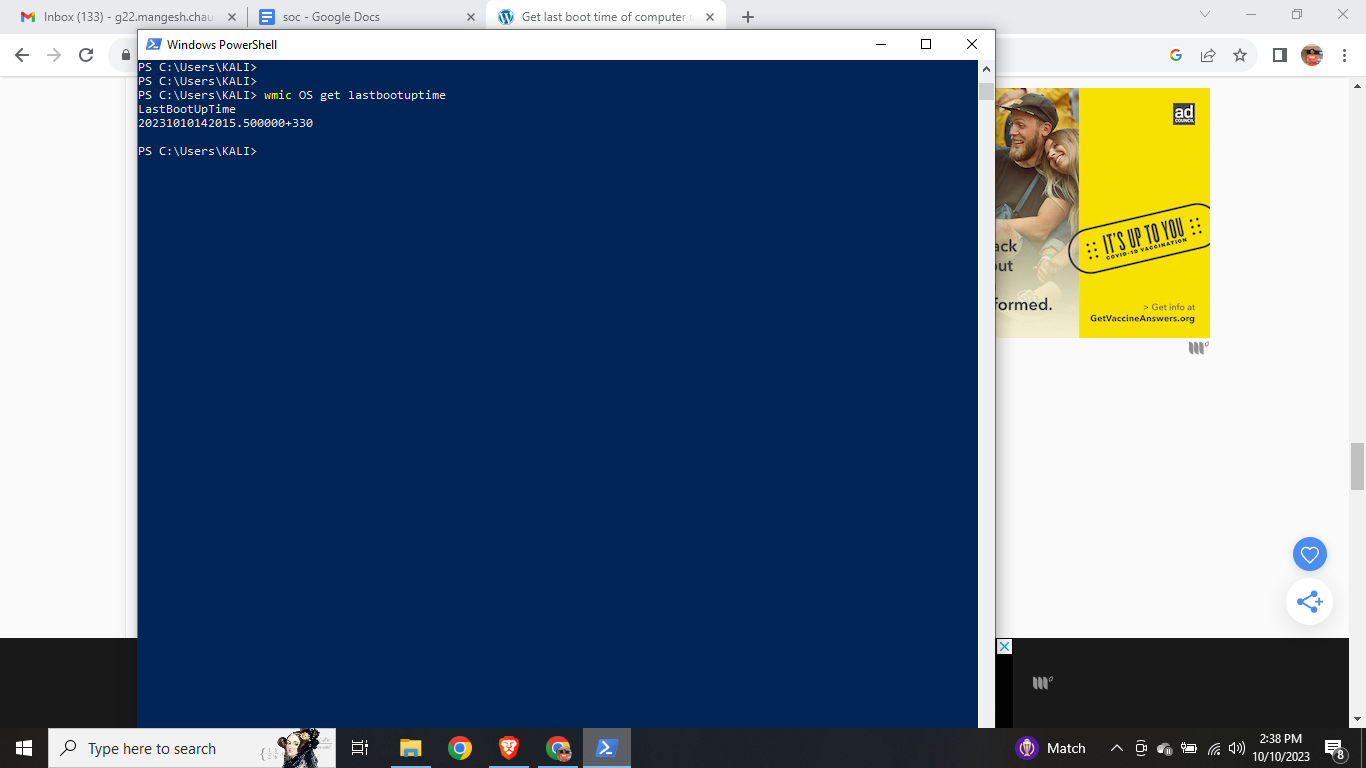
net statistics workstation | select-string "Statistics"



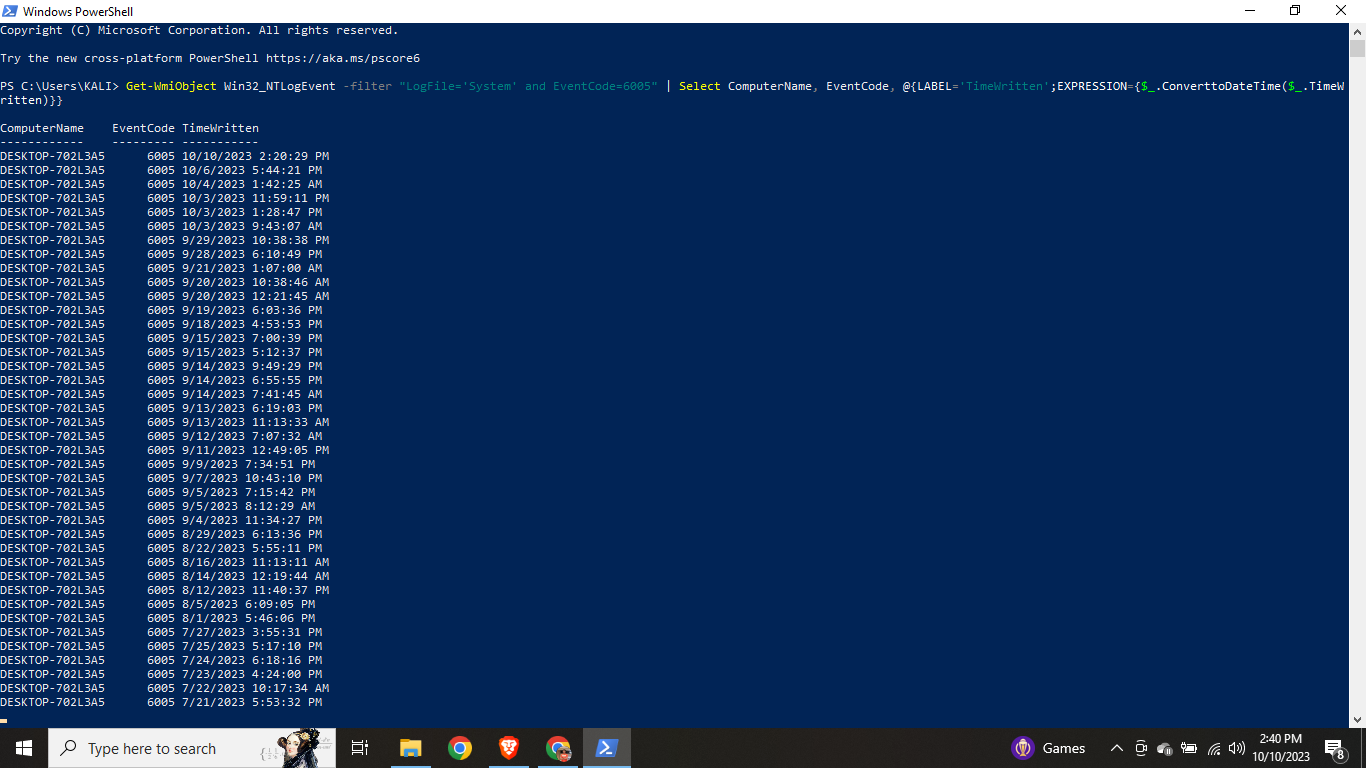
systeminfo | Select-String "Host Name","System Boot Time"



wmic OS get lastbootuptime

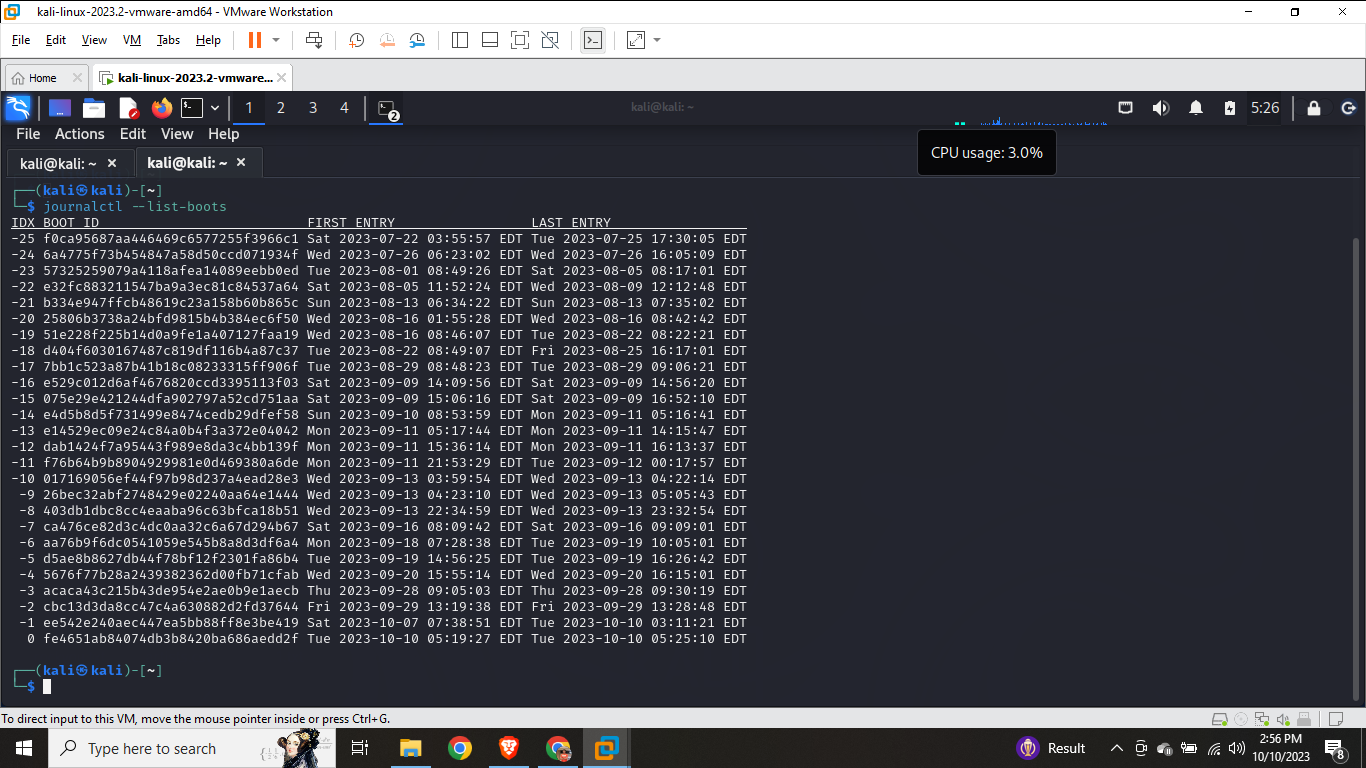


Get-WmiObject win32\_operatingsystem | select csname, @{LABEL='LastBootUpTime';EXPRESSION={$\_.ConverttoDateTime($\_.lastbootuptime)}}



**Practical 10**

**Journalctl --list-boots**

****