CYSE 280 – Windows System Security Lab

Lab Title

Windows TCP/IP Security: Network Configuration, Ports, Processes, and Windows Firewall

Objective

The purpose of this lab is to understand how TCP/IP works on a Windows system and how Windows security tools can be used to monitor and protect network communication.

By completing this lab, you will learn how to:

- Examine TCP/IP configuration.
- Identify active TCP connections and listening ports.
- Identify the Windows processes associated with network connections.
- Examine Windows Firewall settings.
- Create a Windows Firewall rule.
- Verify the firewall rule.
- Remove the test firewall rule.
- Explain why TCP/IP security is important.

Part 1 – Examine TCP/IP Configuration

Step 1

Login to Virginia Cyber Range..

Search for:

Command Prompt

Right-click it and select:

Run as administrator

Step 2

Type:

```
ipconfig /all
```

and press **Enter**.

Step 3

Find the network adapter that you are currently using.

Look for:

- IPv4 Address
- Subnet Mask
- Default Gateway
- DNS Servers

Example

Your result may look similar to:

```
IPv4 Address. . . . . . . . : 192.168.1.25
Subnet Mask . . . . . . . . : 255.255.255.0
Default Gateway . . . . . . : 192.168.1.1
DNS Servers . . . . . . . . : 192.168.1.1
```

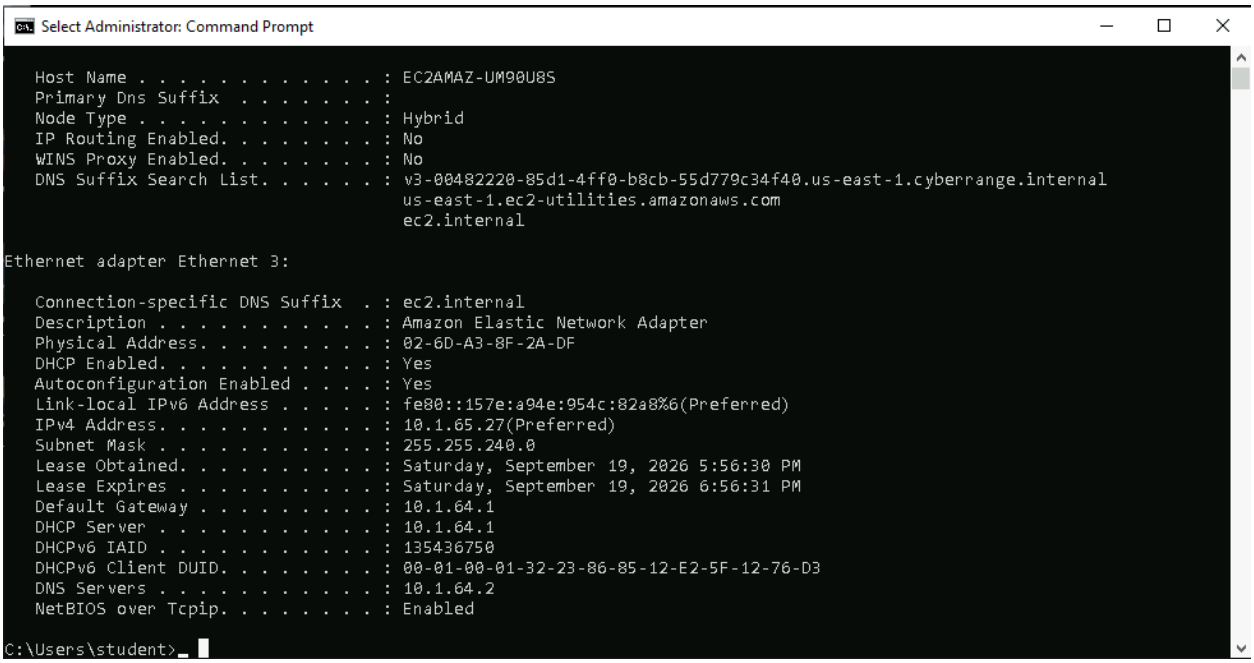
Your values will probably be different.

Screenshot #1

Take a screenshot showing your TCP/IP configuration.

Make sure the following information is visible:

- IPv4 address
- Subnet mask
- Default gateway
- DNS server



Part 2 – Examine TCP Connections and Listening Ports

Step 1

In Command Prompt, run:

```
netstat -ano
```

Step 2

Examine the output.

You may see something similar to:

```
Proto  Local Address      Foreign Address       State           PID
TCP    0.0.0.0:135        0.0.0.0:0             LISTENING       1080
TCP    0.0.0.0:445        0.0.0.0:0             LISTENING       4
TCP    192.168.1.25:51532 142.250.x.x:443       ESTABLISHED     6250
```

Important Terms

Local Address:
The IP address and port being used by your computer.

Foreign Address:
The remote computer/IP address and port.

LISTENING:
A service is waiting for incoming connections.

ESTABLISHED:
A TCP connection has been successfully established.

PID:
Process Identification Number. It identifies the Windows process responsible for the connection.

Step 3

Find at least **5 TCP entries**.

Record them in the table below.

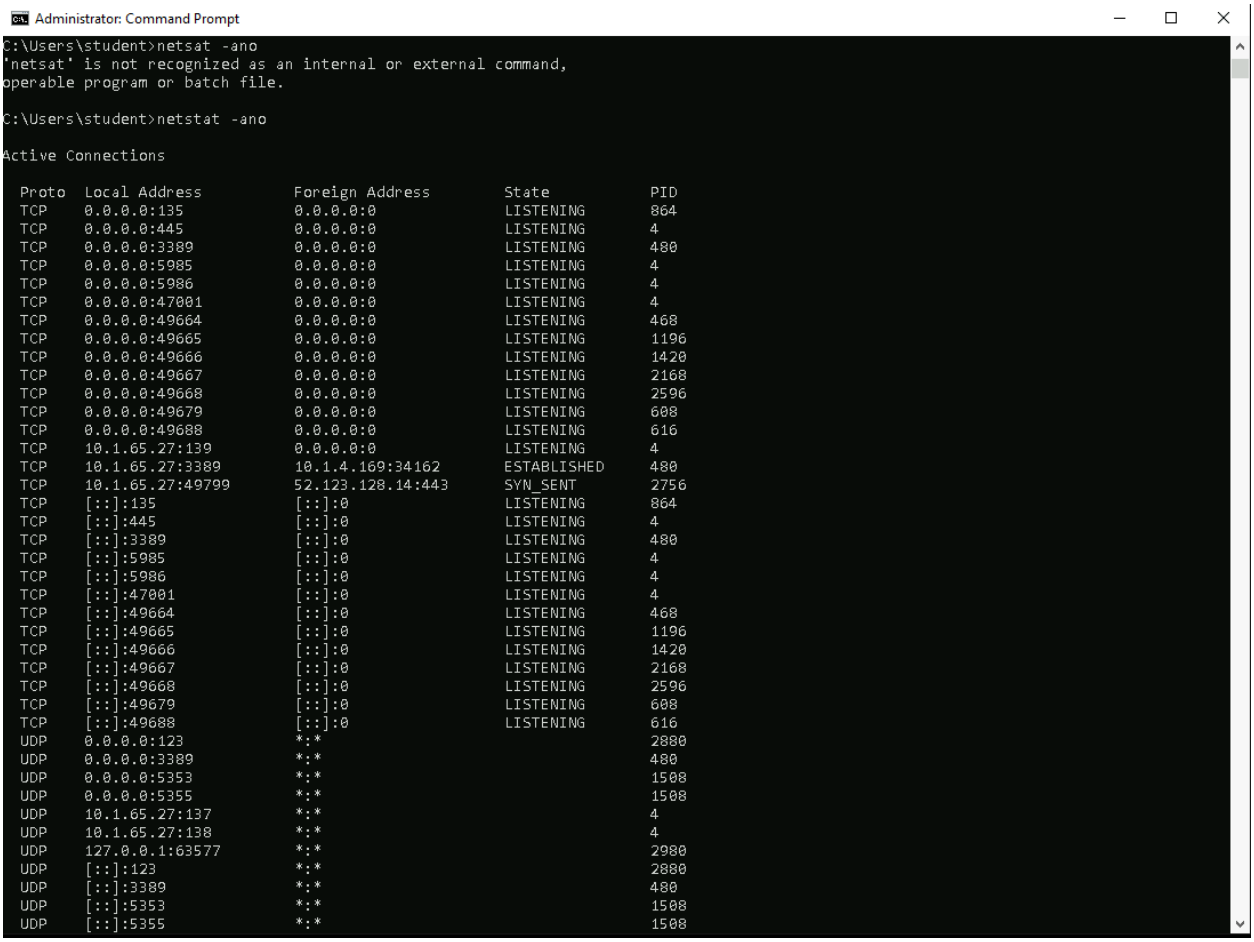
#	Proto	Local Address	Foreign Address	State	PID
1	TCP	0.0.0.0:135	0.0.0.0:0	LISTENING	864
2	TCP	0.0.0.0:445	0.0.0.0:0	LISTENING	4
3	TCP	0.0.0.0:3389	0.0.0.0:0	LISTENING	480
4	TCP	0.0.0.0:5985	0.0.0.0:0	LISTENING	4
5	TCP	0.0.0.0:5986	0.0.0.0:0	LISTENING	4

Screenshot #2

Take a screenshot of your `netstat -ano` results.

Make sure the screenshot clearly shows:

- Local Address
- Foreign Address
- State
- PID



Part 3 – Identify the Windows Process

The PID from `netstat` can be used to identify the Windows process responsible for a network connection.

Step 1

Choose one PID from your `netstat -ano` results.

For example:

```
PID = 1080
```

Your PID will be different.

Step 2

Run:

```
tasklist /FI "PID eq 1080"
```

Replace `1080` with the PID you selected.

For example:

```
tasklist /FI "PID eq XXXX"
```

Step 3

Windows should display the process associated with that PID.

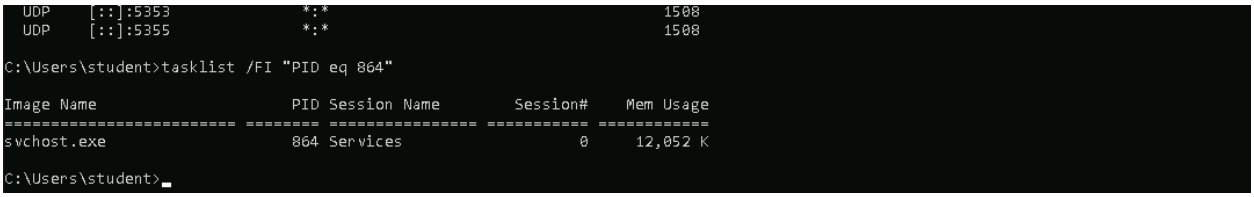
Example:

```
Image Name                     PID
=========================
svchost.exe                  1080
```

Screenshot #3

Take a screenshot showing:

- The command you entered
- The PID
- The process name



Part 4 – Examine Windows Firewall

Windows Firewall helps control network traffic entering and leaving a computer.

Step 1

Open **PowerShell as Administrator**.

Step 2

Run:

```
Get-NetFirewallProfile
```

Step 3

Look at the firewall profiles:

- Domain
- Private
- Public

You should see an `Enabled` value.

Example:

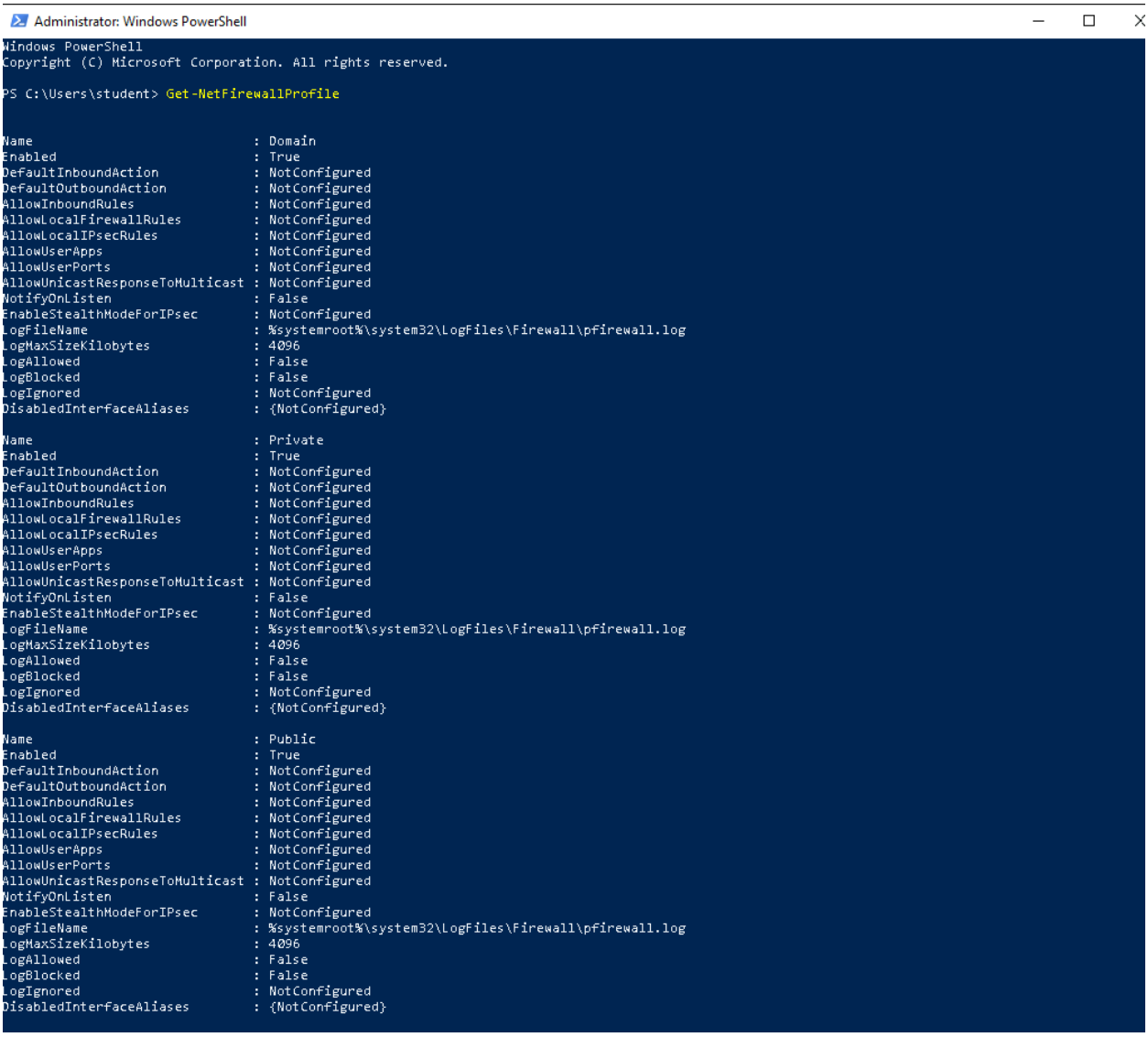
```
Name      Enabled
----      -------
Domain    True
Private   True
Public    True
```

Your results may be different.

Screenshot #4

Take a screenshot showing the output of:

```
Get-NetFirewallProfile
```



Part 5 – Check TCP Port 5555

In this part, you will create a temporary firewall rule for a test port.

We will use **TCP port 5555**.

Step 1

First, check whether anything is currently using port 5555.

Run:

```
Get-NetTCPConnection -LocalPort 5555 -ErrorAction SilentlyContinue
```

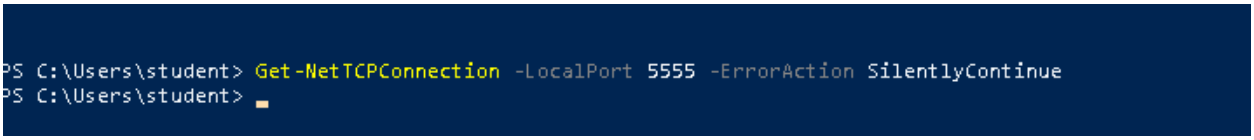
Expected Result

If nothing is using the port, PowerShell should return **no output**.

If you see a result, **do not create the rule yet**. Contact your TA.

Screenshot #5

Take a screenshot showing the command and its result.



Part 6 – Create a Windows Firewall Rule

Now you will create a temporary inbound firewall rule.

Step 1

In PowerShell, run:

```
New-NetFirewallRule -DisplayName "CYSE TCP 5555 Test Block" -Direction
Inbound -Protocol TCP -LocalPort 5555 -Action Block
```

What does this command do?

The command creates a firewall rule that:

- Applies to inbound traffic
- Uses TCP
- Applies to port 5555
- Blocks the traffic

Step 2 – Verify the Rule

Run:

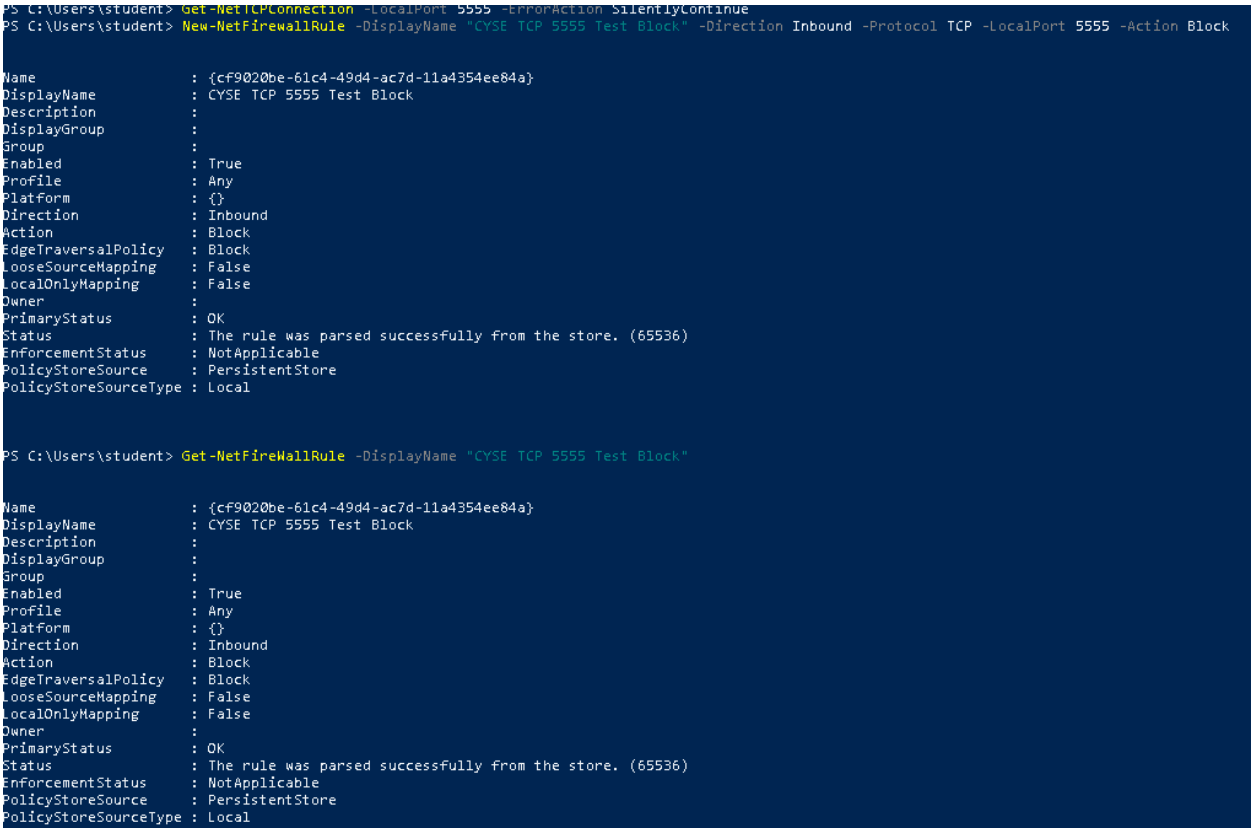
```
Get-NetFirewallRule -DisplayName "CYSE TCP 5555 Test Block"
```

You should see information similar to:

```
DisplayName : CYSE TCP 5555 Test Block
Enabled     : True
Direction   : Inbound
Action      : Block
```

Screenshot #6

Take a screenshot showing the firewall rule.



Part 7 – Remove the Test Firewall Rule

Important: Remove the test rule when you finish the lab.

Step 1

Run:

```
Remove-NetFirewallRule -DisplayName "CYSE TCP 5555 Test Block"
```

Step 2

Verify that the rule has been removed:

```
Get-NetFirewallRule -DisplayName "CYSE TCP 5555 Test Block"
```

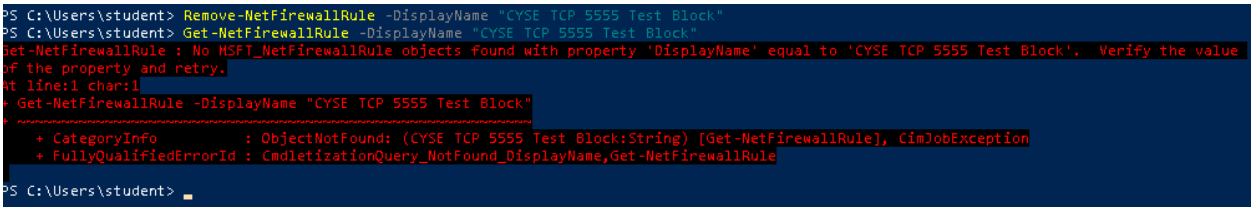
Expected Result

The rule should no longer appear.

There should be no matching firewall rule.

Screenshot #7

Take a screenshot showing that the test firewall rule has been removed.



Part 8– Short Conclusion

Write **5–10 sentences** explaining what you learned from this lab.

Your conclusion should discuss:

- TCP/IP configuration
- TCP connections
- Ports
- PIDs
- Windows processes
- Windows Firewall
- Why network security is important

In this lab, I learned about TCP/IP configurations and Windows security tools, such as firewalls, that enable secure network communications. The ipconfig /all command provided a list of important network settings such as the IPv4 address, subnet mask, gateway, and DNS. Netstat -ano was used to view active TCP connections, ports that are listening, status of connections, and numbers that identify processes. The PID information enabled me to use the tasklist command to identify which Windows process is associated with the numbers provided by the Netstat -ano command. Listening ports show services waiting for inbound communication, while established connections represent active communication. Using PowerShell, I was able to view the firewall profiles and then create and delete a firewall rule. Overall, network security is important, as attackers can potentially cause significant harm to organizations or individuals if a network is not secure.

Learning Outcome

After completing this lab, students should be able to explain the relationship between **TCP/IP networking and Windows system security** and demonstrate how Windows tools such as `ipconfig`, `netstat`, `tasklist`, PowerShell, and Windows Firewall can be used to monitor and protect a Windows system.