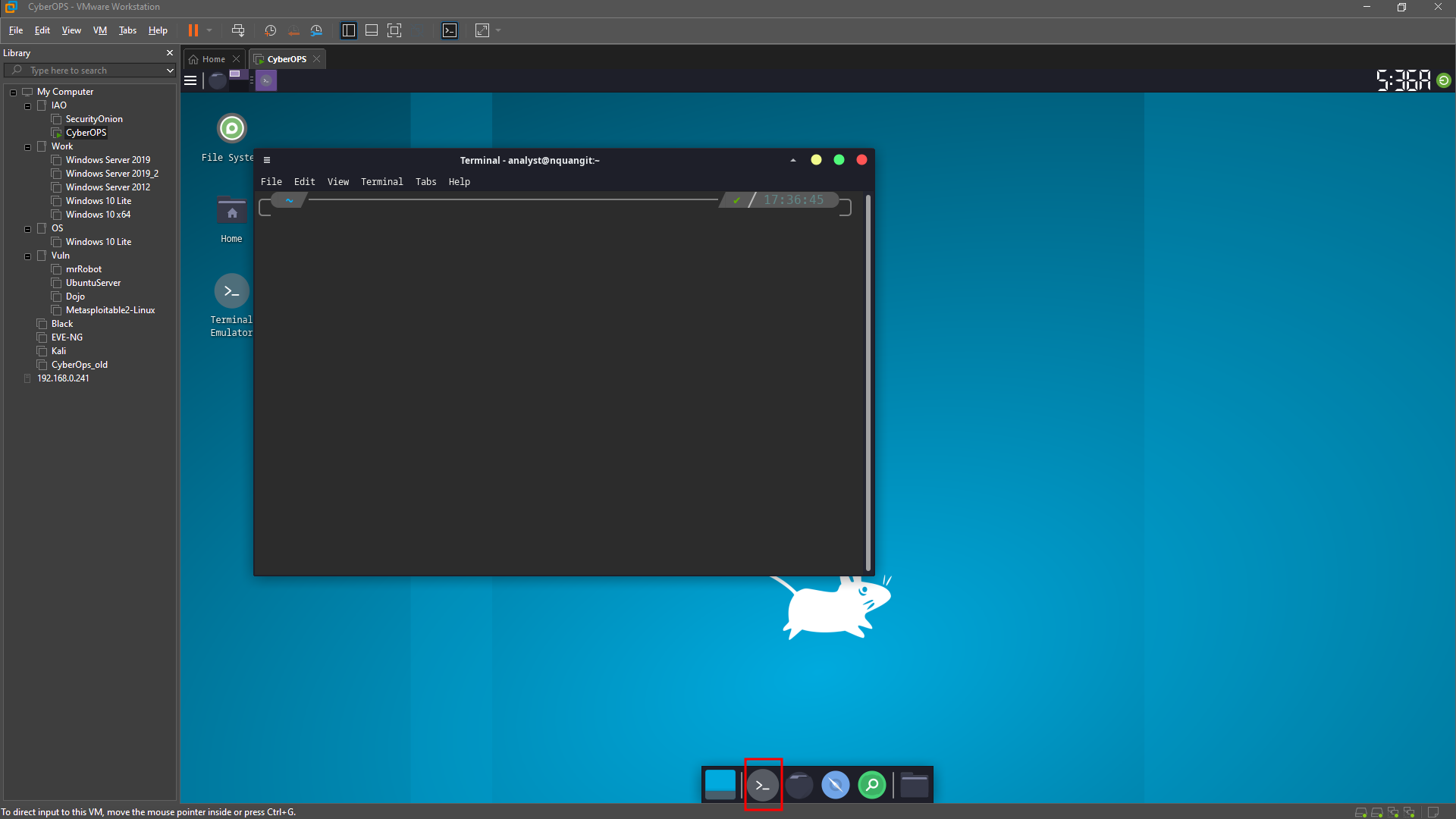
**Lab - Linux Servers**

# Instructions

## Servers

### Access the command line.

* + - 1. Log on to the CyberOps Workstation VM as the **analyst**,using the password **cyberops**. The account **analyst** is used as the example user account throughout this lab.
      2. To access the command line, click the **terminal** icon located in the Dock, at the bottom of VM screen. The terminal emulator opens.



### Display the services currently running.

* + - 1. Use the **ps** command to display all the programs running in the background:

A screenshot of a computer

Description automatically generated

Why was it necessary to run ps as root (prefacing the command with sudo)?

Because some processes do not belong the the current user and can not be show if running ps as a normal user.

* + - 1. In Linux, programs can also call other programs. The **ps** command can also be used to display such process hierarchy. Use **–ejH** options to display the currently running process tree after starting the nginx webserver with elevated privileges.

A screenshot of a computer

Description automatically generated

How is the process hierarchy represented by ps?

Uses indentation.

A screenshot of a computer

Description automatically generated

* + - 1. As mentioned before, servers are essentially programs, often started by the system itself at boot time. The task performed by a server is called a *service.* In such fashion, a web server provides web services.

A screenshot of a computer

Description automatically generated

* + - 1. Use **netstat** with the **–tunap** options to adjust the output of **netstat**.Notice that **netstat** allows multiple options to be grouped together under the same “**-**“ sign.

A screenshot of a computer

Description automatically generated

What is the meaning of the –t, -u, –n, –a and –p options in netstat? (use man netstat to answer)

A screenshot of a computer

Description automatically generated

-t: list tcp services/processes/connections

-u: list udp services/processes/connections

-n: Show numerical addresses instead of trying to determine symbolic host, port or user names.

-a: Show both listening and non-listening sockets. With the --interfaces option, show interfaces that are not up

-p: Show the PID and name of the program to which each socket belongs. A hyphen is shown if the socket belongs to the kernel (e.g. a kernel service, or the process has exited but the socket hasn't finished closing yet).

Is the order of the options important to **netstat**?

No, the option order is irrelevant.

Based on the **netstat** output shown in item (d), what is the Layer 4 protocol, connection status, and PID of the process running on port 80?

While port numbers are just a convention, can you guess what kind of service is running on port 80 TCP?

* + - 1. Sometimes it is useful to cross the information provided by **netstat** with **ps**. Based on the output of item (d), it is known that a process with **PID 395** is bound to TCP port 80. Port 395 is used in this example. Use **ps** and **grep** to list all lines of the **ps** output that contain **PID 395**. Replace 395 with the PID number for your particular running instance of nginx:

The process PID 395 is nginx. How could that be concluded from the output above?

What is **nginx**? What is its function? (Use google to learn about nginx)

The second line shows that process 396 is owned by a user named http and has process number 395 as its parent process. What does that mean? Is this common behavior?

Why is the last line showing grep 395?

## Using Telnet to Test TCP Services

* + - 1. In Part 1, **nginx** was found to be running and assigned to port 80 TCP. Although a quick internet search revealed that **nginx** is a lightweight web server, how would an analyst be sure of that? What if an attacker changed the name of a malware program to **nginx**, just to make it look like the popular webserver? Use **telnet** to connect to the local host on port 80 TCP:
      2. Press a few letters on the keyboard. Any key will work. After a few keys are pressed, press ENTER. Below is the full output, including the Telnet connection establishment and the random keys pressed (fdsafsdaf, this case):

Why was the error sent as a web page?

* + - 1. Looking at the **netstat** output presented earlier, it is possible to see a process attached to port 22. Use Telnet to connect to it.

Use Telnet to connect to port 68. What happens? Explain.

# Reflection Questions

* 1. What are the advantages of using netstat?
  2. What are the advantages of using Telnet? Is it safe?