

In this security exercise, students will work individually with their notebook computers to examine two different operating systems. They will also become familiar with a “Virtual Machine” environment that has become *de rigueur* in the modern information technology world.

Equipment: Students will each log into a Windows operating system using a common login (provided by the instructor). They will then use a virtual machine environment to open a second operating system, Ubuntu Linux. The virtual machine is an environment that allows a host computer (in this case the Windows notebook) to run a completely separate “computer” with a different operating system within a window of the host. The second virtual computer is known as the “Guest.” The advantage is that the host and guest do not affect each other, and the guest runs as if it were an actual physical computer. The virtual machine host software “fools” it into thinking it is a real computer.

In addition, students will experience two different user interfaces to the operating system. Of course, humans don’t speak binary, so when one interacts with a computer, it is actually at a very high level and translated into something that a human can understand. There’s actually quite a lot going on below the surface to make the translation between machine language to human language. We will work with two interfaces: The Graphical User Interface (GUI) – pronounced “gooey,” and the command line interface (CLI). The latter is often referred to as a “Shell,” but in reality a shell is a specific form of CLI. There are several “shells” that are in popular use in the Linux / Unix world, but the one most often used is the “Bourne Again Shell (bash), pronounced “bash.” It is named after the guy who first created the “Bourne shell” (which was “sh”). The bash shell is an improved version, thus the “Bourne again” part. Computer geeks have a unique sense of humor, you will discover.

Unless you’ve lived in a cave most of your life, you are probably very familiar with the GUI. But, long long ago, there was no such thing. Humans interacted with the computer through a textual interface, the CLI. There are many types. We already discussed the most popular shells in Linux and Unix. In Windows, you may be familiar with the “DOS” shell, which was based upon “CPM.” Even today, with all the fancy GUI’s that exist, those who really know computers almost always prefer the CLI to the GUI. This is not just nostalgia or stubbornness. There are very good reasons. Can you suppose some of them? I’ll give you a hint: the GUI is just a fancy graphical way to execute the CLI commands. It’s just a pretty picture...

From the Windows Start menu, type into the search box "cmd" and hit enter. A Windows "Command Prompt" CLI window should appear. On your desktops you should find a link to the Vsphere client. Double-click to open it. After logging in, an Ubuntu GUI will magically appear. Better yet, within that GUI window, a Linux command line interface should open (bash shell). You now should have four separate computer interfaces available to you:

- The Windows host system GUI
- A Windows host system CLI
- An Ubuntu GUI running inside a VM window
- An Ubuntu bash CLI running inside the Ubuntu GUI.

We will now do some routine operating system functions using all four of these interfaces. Note that the differences are all really language differences, what computer types call "syntax." Think of it like two dialects of English. The British use some funny words and phrases (to us), but it still gets the same job done.

Complete the following tasks in all four interfaces, making notes of your observations along the way. Use the Internet to look up more details of the commands you are given to use. A good command line references is here:

<http://ss64.com/>

Click 'bash' for Linux commands, and 'CMD' for Windows commands.

- 1) Browse the computer hard drive directory structure:
  - a. Windows GUI – use the "Windows Explorer" interface
  - b. Windows CLI – type the command "`cd /`" and then "`dir`"
  - c. Linux GUI – Find the equivalent interface to Windows Explorer (what is it?)
  - d. Linux CLI – type the command "`cd /`" and then "`ls`"
  
- 2) Look up the "`cd`" "`dir`" and "`ls`" commands. Do they apply to both Linux and Windows? Is the syntax the same? How do you specify options? Use these commands to move about in the directory structure of both CLI's. Do the same thing using the GUI.

- 3) Navigate to the '/tmp' directory in the Linux CLI. Type the following command:

```
touch tempfile
```

What happened? **Hint:** use the 'ls' command.

Now use the `ls` command to print out detailed information about the files in that directory (stuck? Ask your instructor about the *man page*). Can you find the 'tempfile' in the GUI? Can you find its file properties? What are they?

See if you can do the same thing using the Windows CLI and GUI directly in the C:\ directory.

**Extra Credit:** Write the full command which accomplishes this in the CLI.

- 4) Now (from Ubuntu), type the command:

```
nano /tmp/tempfile
```

Can you figure out how to change the contents of that file to say "Beat Army"? Verify your success by using the GUI.

Now try to do this in Windows; does 'nano' work? Can you edit the file in the GUI instead so that it also says "Go Navy"?

- 5) In the Windows CLI, navigate to the "C:\\" directory. Type the command "

```
md midshipmen
```

What happened?

Can you do the same thing in Ubuntu? What command did you use? Need a hint? (Try typing 'help md' in Windows and see if there are any other command that accomplish the same goal)

- 6) If you didn't figure it out in the previous step, in the Ubuntu CLI (type 'pwd' to verify you are still in your '/tmp' directory), type:

```
mkdir midshipmen
```

then (regardless) type:

```
mv tempfile /tmp/midshipmen
```

What happened (Use the GUI to see what's going on under the covers)? Try to do this in Windows, too.

- 7) Now enter (in the Linux CLI):

```
cp -a /tmp/midshipmen/* /tmp
```

What happened? Why did I have you type the "-a"?

What command will do this in Windows?

- 8) Back to Ubuntu (Linux) CLI. From the /tmp directory, type

```
rmdir midshipmen
```

What happened?

Try to do the same thing in Windows now. Did it work?

- 9) From the Linux CLI, what happens when you type the command:

```
rm -rf /tmp/*
```

Look up the `rm` command and write down what the options `-rf` tell the `rm` command to do. Is this a good idea in general?

10) From the Linux CLI, navigate to the “/” directory (spoken “root”). Windows likes to use drive letters like “C:” to define hard drives, but Linux and Unix don’t do this. The “root” of any drive is just the very top level directory. Root also has another meaning in Linux. What is it?

11) From the root directory in Ubuntu CLI, type:

```
ls -lha
```

Write down the line for the “/etc” directory. Now figure out who the owner and group for that directory are. (Once again, the *man page* is your friend)

12) Look up the Linux “**sudo**” command and write a brief description here:

13) Type this command:

```
cat /etc/shadow
```

What output do you see? Why?

Do you have better luck typing the following?

```
sudo cat /etc/shadow
```

**Extra Credit:** What is the significance of this file? Why can’t anyone access it?

14) Now try this command in the Linux CLI:

```
sudo rm -rf /*
```

**Note:** this is a very bad thing to do if you’re a good guy. What happened? Did the system try to stop you? Why or why not?

**Conclusion and Results:**

Your typed lab report will consist of two paragraphs, in the first paragraph:

- Briefly describe what you did in the lab in your own words.
- Discuss something new that you learned.

In the second paragraph, answer the questions:

- How could an adversary use this knowledge or these tools for malicious purposes?
- How could you use your new understanding to protect your systems and personnel from attack?

**Staple** the completed report to the back of your original lab and turn it in to your instructor at the beginning of the next class.