Assignment 1: Command Line Kung Foo

Description

There is no luxurious GUI to guide you, just you, a keyboard and a command prompt. Enjoy! You may want to get Lab 1 going as well, since it will guide you through setting up VMS where the following commands can be used.

Learning Objectives

- Students will learn and practice basic command line skills required for the course.
- Students will become apt at the ancient art of ‘command line kung foo’.
- Students will gain experience with command line networking tools.
- Students will complete a series of networking steps on both Windows and Linux.
- Students will reinforce scripting and automation knowledge by creating simple scripts.

Method

For this lab you need to achieve the following:

Part 1 – Windows without Windows

Done?	Task
☐	1. Display the system variables currently active on the machine of interest.
☐	a. What is the systemroot variable and what is it used for?
☐	b. What is the appdata variable and what is it used for?
☐	c. What is the path variable and how does it work?
☐	2. What is the processor family? Speed?
☐	3. What physical devices are attached?
☐	4. Display the list of currently active processes.
☐	5. Using the dir command with appropriate switches find the hosts file then display its contents. (Pay particular attention to /b and /s, what do they do?)
☐	6. Use the findstr to locate a text file on the C:\ drive.
☐	7. List the users of the machine.
☐	8. Add a user and remove the user.
☐	9. Create a group and add a user to it. Now delete them.
☐	10. Display the firewall settings. (Consider the netsh command).
☐	a. Disable the firewall.
☐	b. Reenable the firewall.
☐	11. Display the arp table.
☐	12. Display a registry key. How would you change the value? Why is this very dangerous?
☐	13. What do “ipconfig /displaydns” and “ipconfig /flushdns” do? What else does ipconfig do?
☐	14. Mount a network drive and display a directory of its root.
☐	15. List the services running on the machine (different than processes check out sc cmd).
☐	16. Check out the FOR command.

□	a. What does the /L switch do?
□	b. What does the /F switch do?
□	c. What does “for /R %i in (1,0,2) do echo Hello!” do? How many times will ‘Hello’ be displayed?
□	17. What does “echo >stuff this is some stuff” do?
□	18. What does “find “stuff” s*” do?
□	19. Create 10 files names file1..file10 containing the numbers 1 through 10 using a single command line.
□	20. What does the AT command do?
□	21. What does the SCHTASKS command do?
□	22. What are the differences?
□	23. Enable remote desktop (and set firewall rules to permit remote desktop) from the command line.

Part 2 – The Linux Muscle

You may do this on any Linux based machine

Done?	Task
□	1. Display the system variables currently active on the machine of interest.
□	2. What is the processor family? Speed?
□	3. What physical devices are attached?
□	4. Display the list of currently active processes.
□	5. Find the hosts file and display its contents.
□	6. List the users of the machine.
□	7. Add a user and remove the user.
□	8. Create a group and add a user to it. Now delete them.
□	9. Display the firewall settings. (look up iptables)
□	a. Disable the firewall.
□	b. Re-enable the firewall.
□	10. Where are all the configuration files stored on your particular *nix version?
□	11. Where are all the log files stored?
□	12. Does it use initd or xinitd?
□	a. What do the above daemons do?
□	b. Why are they used?
□	c. What is the difference between how they operate?
□	13. List the services on the machine. Which ones are actually active?
□	14. Mount a remote *nix file system and display a directory of its root.
□	15. Mount a remote windows file system and display a directory of its root.
□	16. Create 10 files names file1..file10 containing the numbers 1 through 10 using a single command line.
□	17. Schedule a command that displays the time every minute somewhere where you can see it. (Note: default cron behavior uses email).

Part 3 – Networking

You should perform these steps the following machines:

1. Linux Virtual Machine
2. A Windows Virtual Machine or host

Done?	Task
☐	1. Display the network settings for all of the NIC's installed.
☐	a. What are the associated IP addresses?
☐	b. What are the associated MAC addresses?
☐	2. Display the ARP table. Add a static entry, then remove it.
☐	3. Display the routing table. Add a static entry, then remove it.
☐	4. Display the open ports on the machine along with the associated processes.
☐	a. Pick an open port, and display the detailed information about the associated process (especially the executable code file).
☐	5. Display some network traffic from one interface using (tcpdump/windump)
☐	6. Since both dumps are in the same output format, pick one that has a TCP initialization sequence and describe the setup in detail through analysis of the packet exchange that initializes the session. Make sure you understand the flags and packet content for a normal TCP setup. (Do this for ONE system only, not all 4)
☐	7. Why do Windows systems frequently report bad TCP checksum? (Although this may or may not occur on your hardware, you should research and explain it).
☐	8. Do a ping sweet of the security lab subnet (192.168.185.0). Be sure to use a /24 network filter. You should do this using a single command.
☐	9. Find and use 3 network enumeration tools. Make sure your scope is unobtrusive and limited to port scanning only. You may test the lab workstation 192.168.185.0/24 network. You should use 5 TOTAL tools between all OS's (not 5 per OS).

Part 4 – Scripting

You should pick two out of the following scripting languages:

- Powershell
- Python
- Ruby

In each language you should perform the following activities. You may use multiple scripts, or a single script.

Done?	Task
☐	1. Search all files of a given extension for a specified string. (eg: *search [drive] [extension] [string]*)
☐	2. Scrape network information from the local host to include:
☐	a. All network interfaces IP addresses, subnets, gateways and DNS servers.
☐	b. The public IP address for NAT'ed host. (eg: if ip == private, use webservice to identify public IP).
☐	3. Send a 'GET' command to a webserver with a user-specified variable and display the result on screen. (eg. *myscript index.html www.byu.edu*)

Submission

Submit the commands/scripts you used, their syntax and a **summary of the output** received. You may submit the occasional screenshot to help clarify when appropriate. Walls of text or walls of screenshot's will lose marks.

Grading Rubric

20%	P1: Windows – Process Followed
20%	P2: Linux – Process Followed
20%	P3: Networking – Process Followed/Correct Tools/Syntax
20%	P4: Scripting – Process followed, results
10%	Results summarized
10%	Style/Clarity
2%	Extra credit for usability considerations on part 4.

Expectations

Students are expected to be familiar with the command line. Some personal research is expected depending on the students past experience. Students are expected to understand these commands at a technical and conceptual level.

Students are expected to research tools for the above steps. Students may use their own scripts where appropriate. Students should not use GUI tools, or scripts written by others (such as forensic scripts).

Students may use any command line or shell they wish. For example, Windows Command Line, VBScript, PowerShell are all appropriate on Windows.

Resources

- http://commandwindows.com
- http://ss64.com
- http://technet.microsoft.com (you might like the command line reference)
- http://www.computerhope.com/unix.htm
- https://www.google.com/?gws_rd=ssl#q=whats+my+ip+web+service