

Practical Laboratory Assignment 1

Introduction to Information Security, INTROSEC. Autumn Term 2019

Group 22

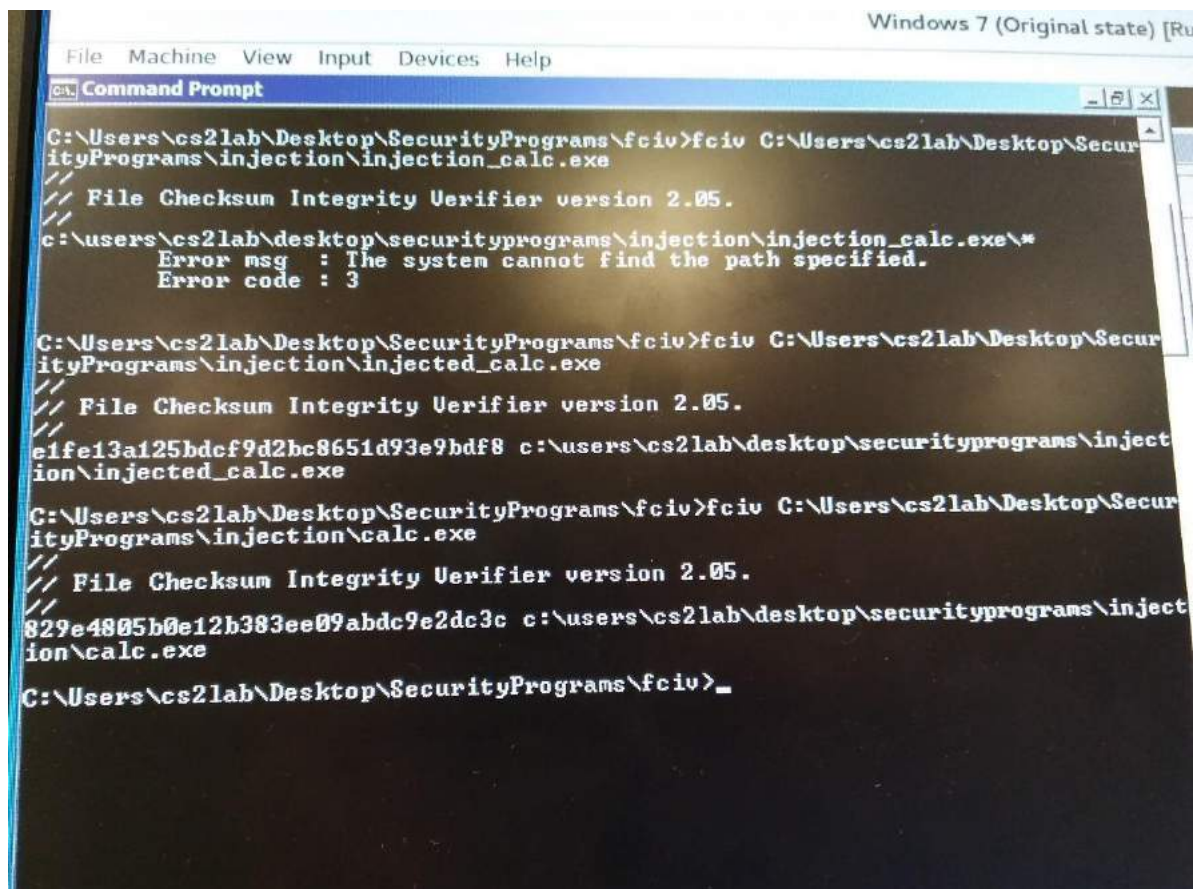
Zaki Naveed & Qijie YE

Part 1. Code Injection:

In this task, we examine and compare the original windows calculator application “**calc.exe**” with the injected one “**Injected_calc.exe.**”

The purpose of code injection is to change the course of execution. Code injection is a hotbed of a computer worms. Attackers use modified an executable to modify values in database, or install malware or execute malevolent code[1].

One does not properly tests every input supplied by a user or application may be exposed to such threats. I think victims can hardly detect that one of their executable has been modified when they have been injected unless they use FCIV frequently to compare hash values, but most important thing is always taking precaution before being attack. One can protect oneself by using vulnerability scanners to ensure applications are safe from various types of attacks, including code injection.



```
Windows 7 (Original state) [Ru]
File Machine View Input Devices Help
C:\Users\cs2lab\Desktop\SecurityPrograms\fciv>fciv C:\Users\cs2lab\Desktop\SecurityPrograms\injection\injection_calc.exe
// File Checksum Integrity Verifier version 2.05.
c:\users\cs2lab\desktop\securityprograms\injection\injection_calc.exe\*
Error msg : The system cannot find the path specified.
Error code : 3

C:\Users\cs2lab\Desktop\SecurityPrograms\fciv>fciv C:\Users\cs2lab\Desktop\SecurityPrograms\injection\injected_calc.exe
// File Checksum Integrity Verifier version 2.05.
e1fe13a125bdcf9d2bc8651d93e9bdf8 c:\users\cs2lab\desktop\securityprograms\injection\injected_calc.exe

C:\Users\cs2lab\Desktop\SecurityPrograms\fciv>fciv C:\Users\cs2lab\Desktop\SecurityPrograms\injection\calc.exe
// File Checksum Integrity Verifier version 2.05.
829e4805b0e12b383ee09abdc9e2dc3c c:\users\cs2lab\desktop\securityprograms\injection\calc.exe

C:\Users\cs2lab\Desktop\SecurityPrograms\fciv>_
```

Figure 1 Hash Codes

- Secure input and output handling.
- Runtime image hash validation – capture a hash of a part or complete image of the executable loaded into memory and compare it with stored and expected hash.

Part 2. Windows Registry

Registry Exercise A:

Windows registry is a special database to store setting information of system and applications. As I mentioned before if an attacker has done code injection, modifying values in database is easy. By going through all the exercise 2, I found that the last user name can not be shown(it is good for privacy); hard drives and RUN button in the start menu can not be visible. These changes are all harmless, but what if an attacker simply changes the setting of windows registry? The functionality of the calculator will mess up. Everytime When we try to login in a Windows Operating System, we can see the usernames of the previous users. According to the security perspective, this is not a good practice. We fixed this issue in a windows OS in this task.

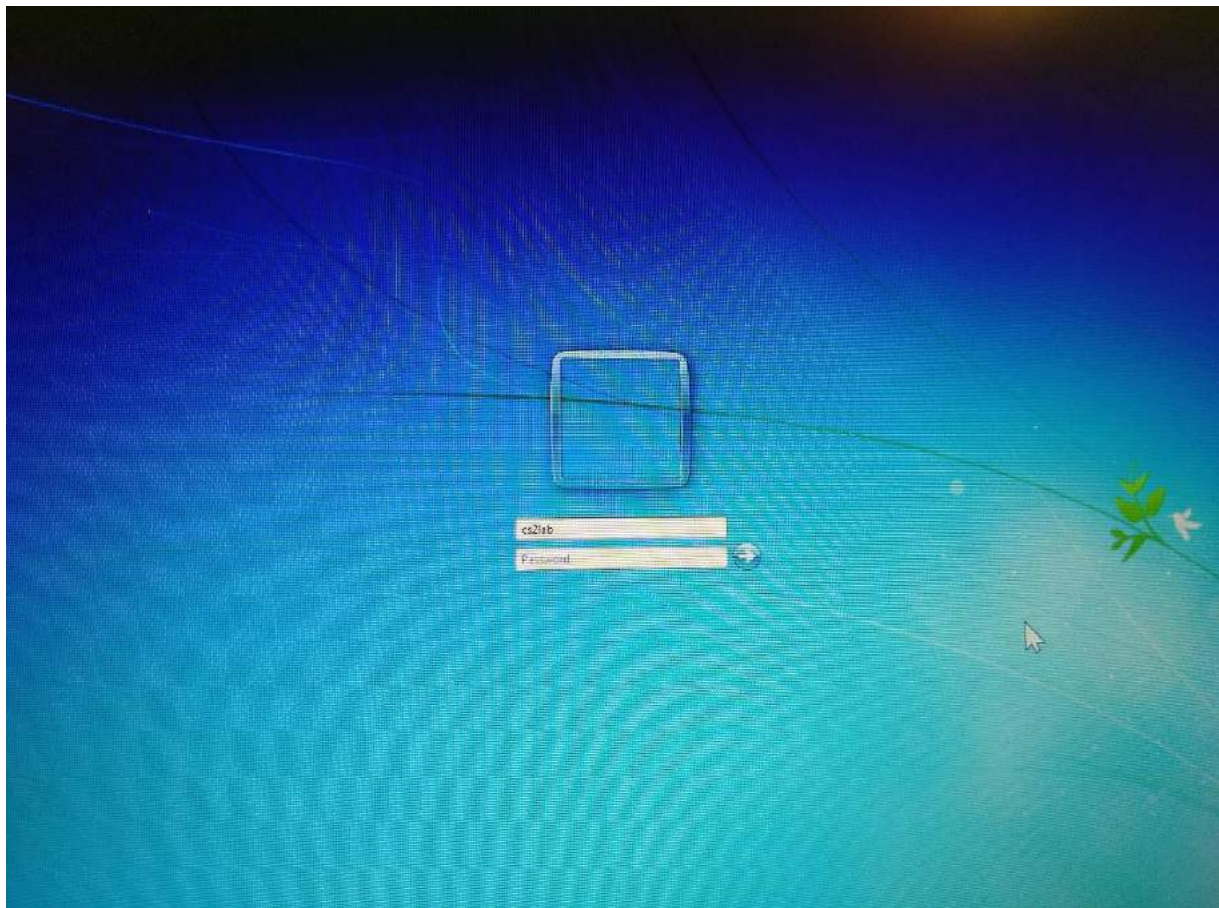


Figure 2 Login Window

When we adding the “DontDisplayLastUserName” file in the windows registry folder, and setting the file value=1. It fixes the problem and hides the previous user’s details at the login screen as shown in the figure 2 above.

Registry Exercise B:

We can stop a user from accessing the particular drive in the windows system by hiding it. In this exercise task, we use the file “NoDrives” and check the working of this file.

We adding the file “NoDrives” in the windows registry folder and setting the value=1 of this File, After this, we can see in figure 3 that it hides the Drive-C from the system.

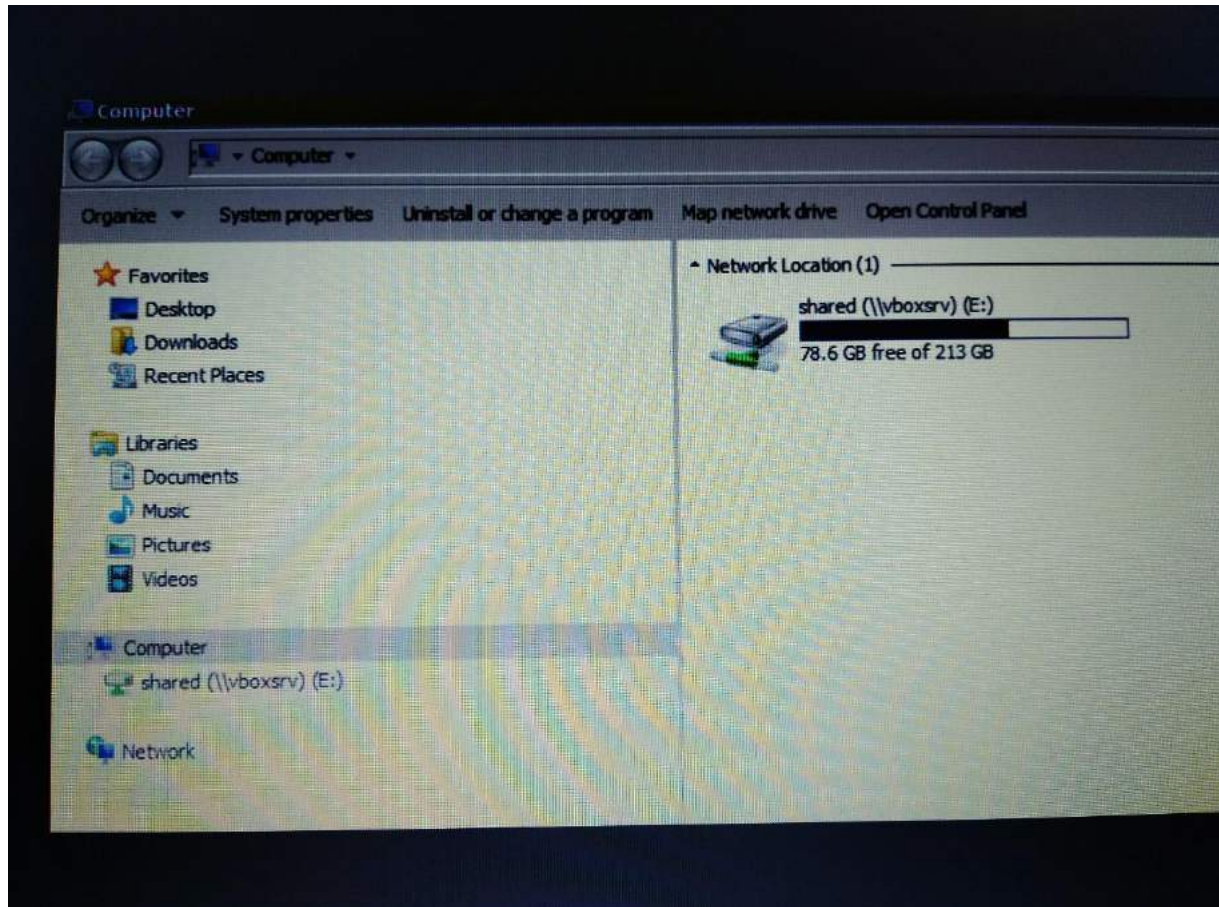


Figure 3 Hidden Drive

Registry Exercise C:

In this exercise, we hide the “Run” Program from the start menu bar by adding the file in the registry with name “NoRun”. When we did this ,the user cannot have access from the start menu.

By editing registry through a command prompt or C:\Windows\System32. We can unhide the drive by changing its value to 0 or deleting the “NoRun” file from the registry .

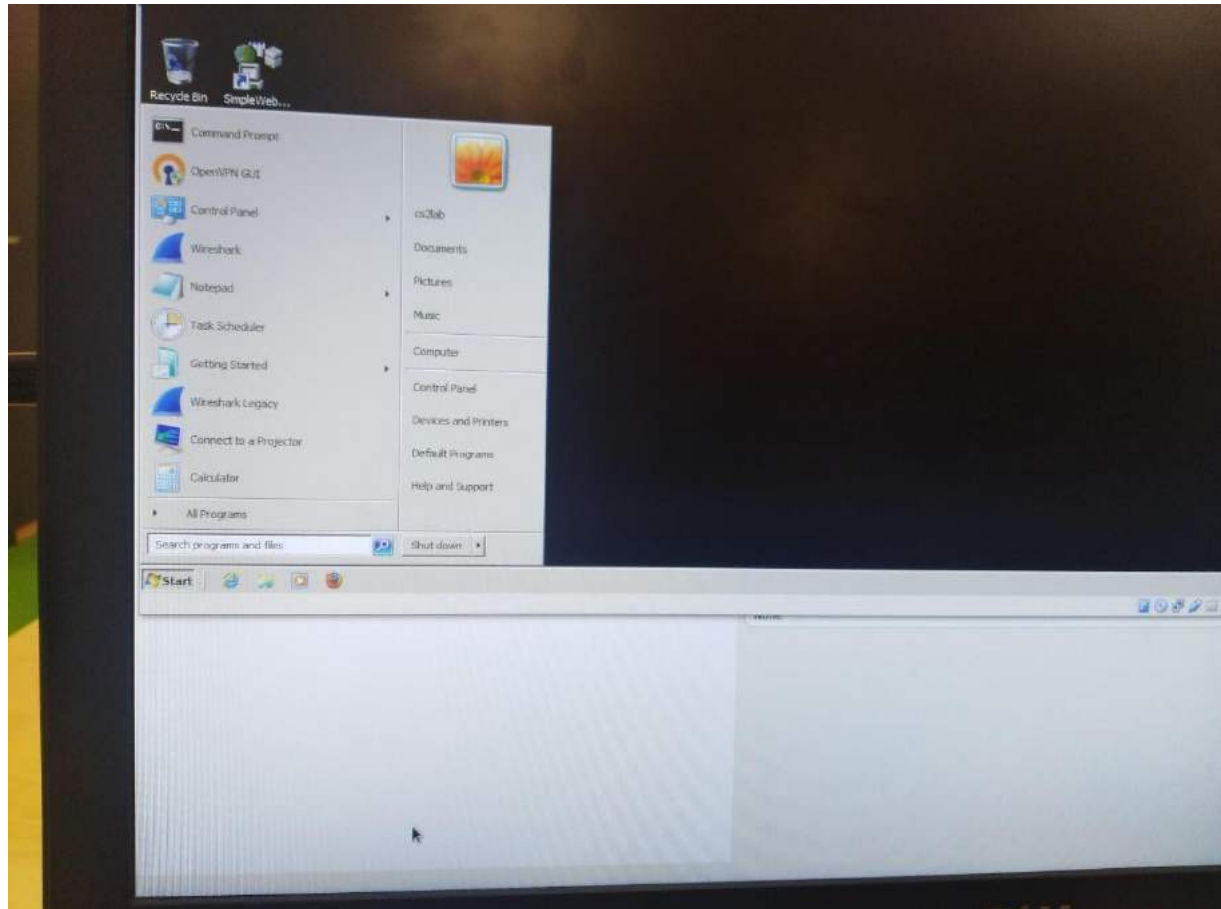


Figure 4 Hiding “Run” Program

Part 3. Spoofing – Bypassing the Login Screen:

By making a copy of the cmd.exe application which is the Windows command line application and then Rename the duplicate “cmd” application into “sethc” and only press the shift button five times. Then the command line presented. What if I replace sethc to any other exe? Maybe I can not only access to command shell but also the explorer or other else. The initial use of some auxiliary functions like sticky keys are harmless, but sometimes they can be a flaw to computer security.

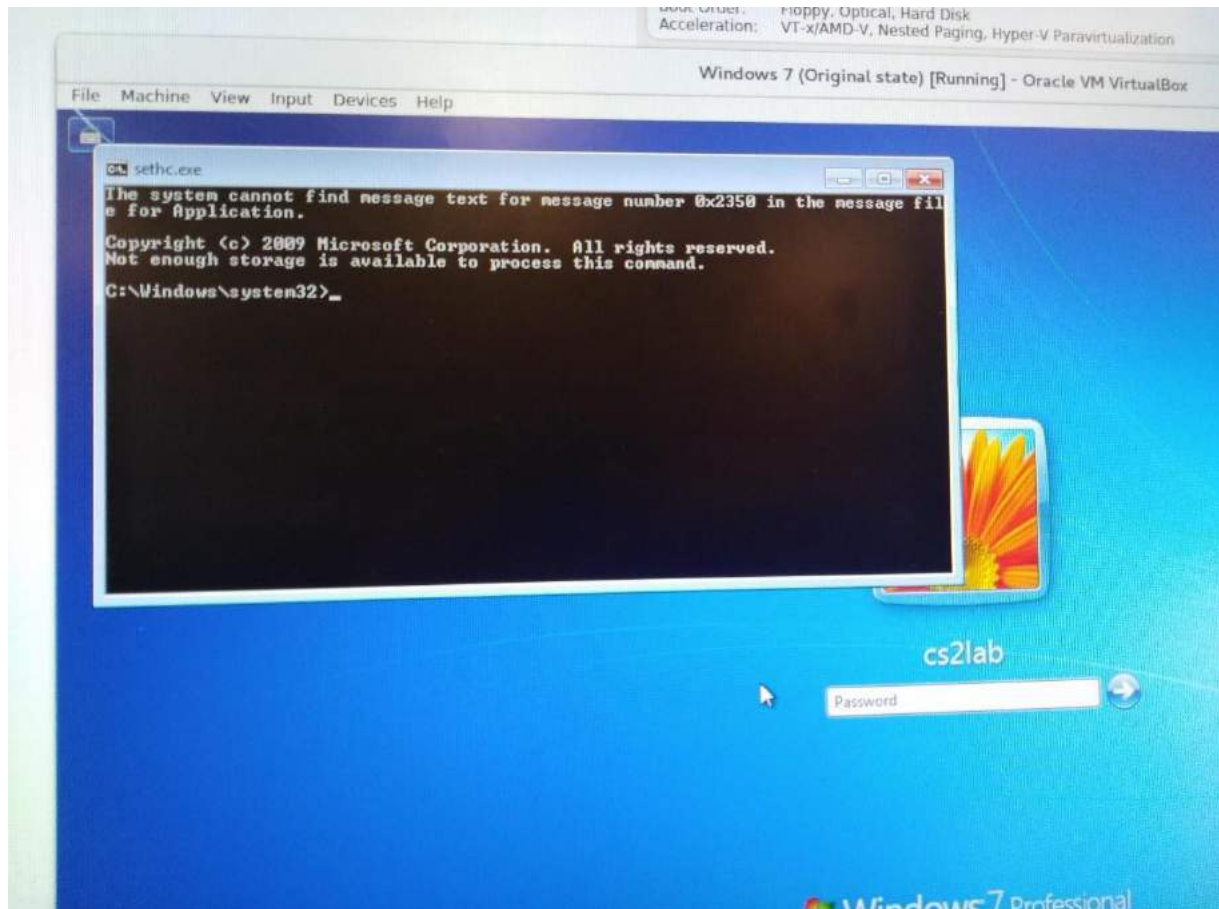


Figure 5 Bypassing the login screen

The users get access to a command prompt with Administrator privileges after pressing the sticky keys combination at the login screen

We can be stop by disabling the sticky keys prompt from running when you press shift five times or by Locking the BIOS with a password can also be helpful.

Part 4. Monitoring keyboard strokes:

In this exercise, Keylogger record all the typing.

Keylogger recorders may also be used by employers to observe employees' computer activities, parents to supervise their children's internet usage, users to track possible unauthorized activity on their devices or law enforcement agencies to analyze incidents involving computer use. These uses are considered ethical or appropriate in varying degrees[2].

Keylogger seems to have a lot of uses. Surprisingly it has proper use--to track possible unauthorized activity on their devices or law enforcement agencies to analyze incidents involving computer use. It seems like some application can be use to detect computer security but simultaneously destroy computer security, vise versa.

In this task, we execute the “**klogger.exe**” Program, and it logged all the keyboard strokes in the “**klogger.txt**” file as shown in figure below.

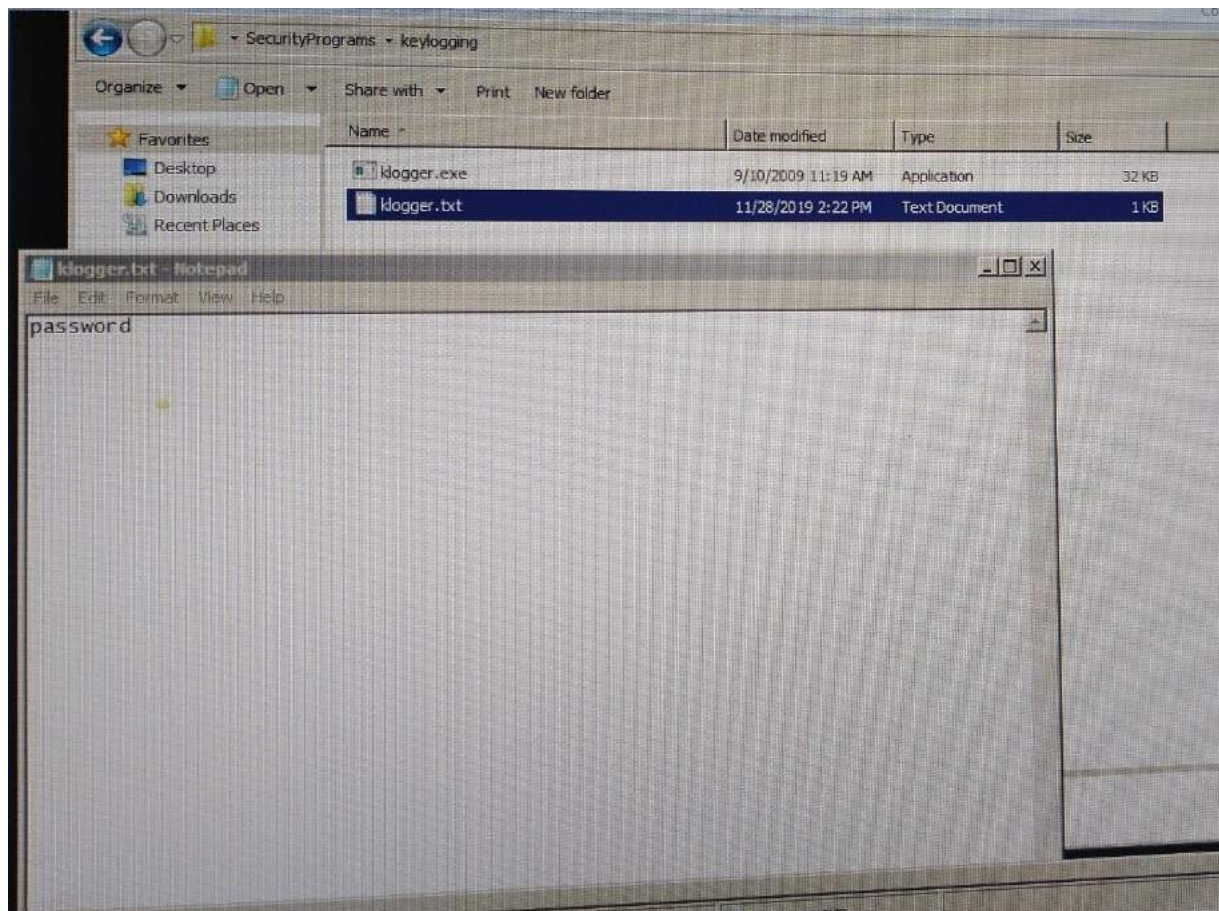


Figure 6 Keylogger

Part 5. Disclosing Masked Passwords:

There are a lot of common tricks to hide passwords by showing stars, bombs or other characters in password fields. Instead of the characters that are actually written. However, the real characters will still be there, this trick can not work on Firefox and Google, but it works on Internet Explorer, which is a low security browser. We try to unmask the password on Nelly website login webpage with the password revealer tool “ASnadBoy’s Revelation”.

By dragging the + cursor at the password field, it shows the password “123456789” as shown in the figure below.

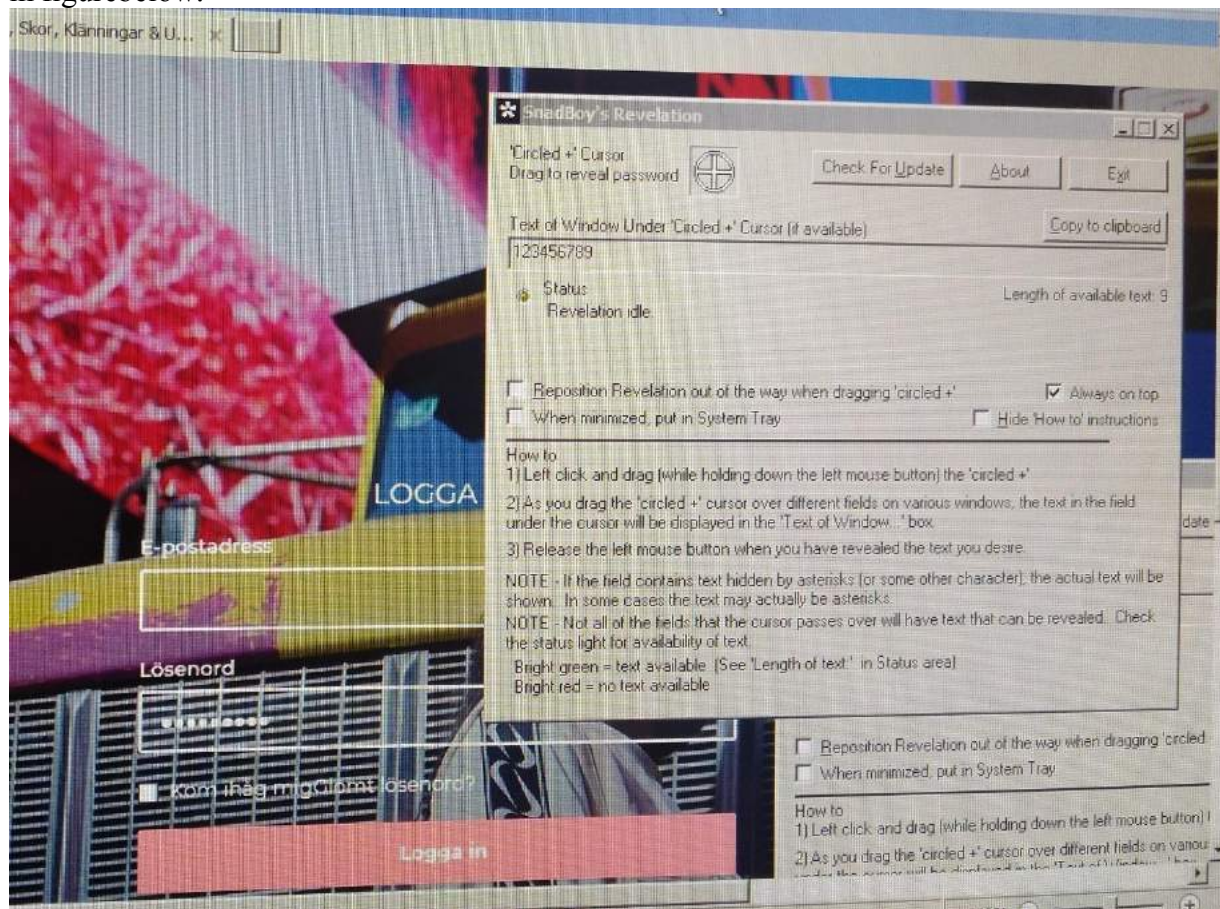


Figure 7 Password Revelation

Part 6. NetBus – Take control over another computer:

NetBus has a scanning-function that is used to find active servers on a local net. In this task, we test NetBus by engaging another group's computer in the Lab and experiment with the different functions of NetBus by using their IP-address as shown in figure 8.

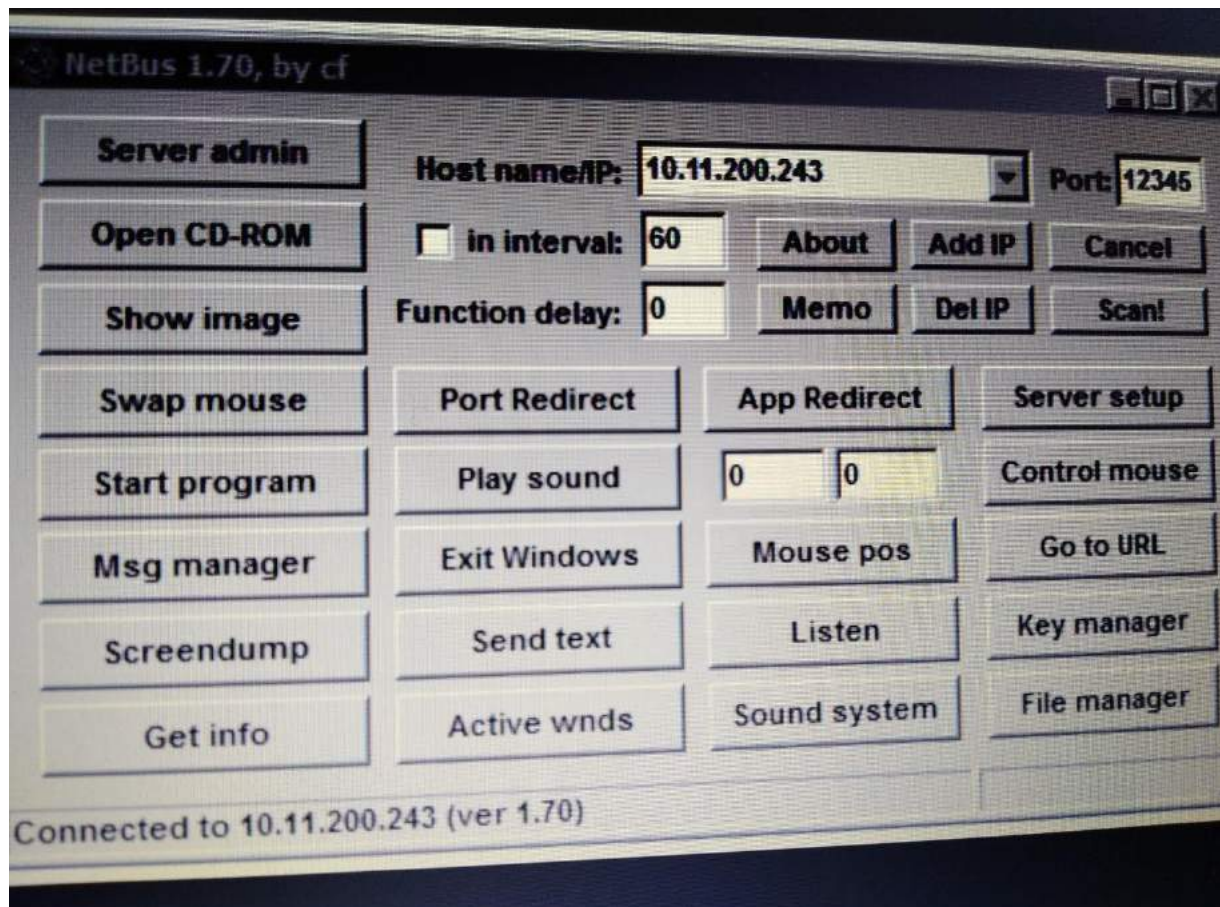


Figure 8 Connecting NetBus with an IP

This Figure show that we have access to the other group's computer. Now we can do anything on the victim's computer.

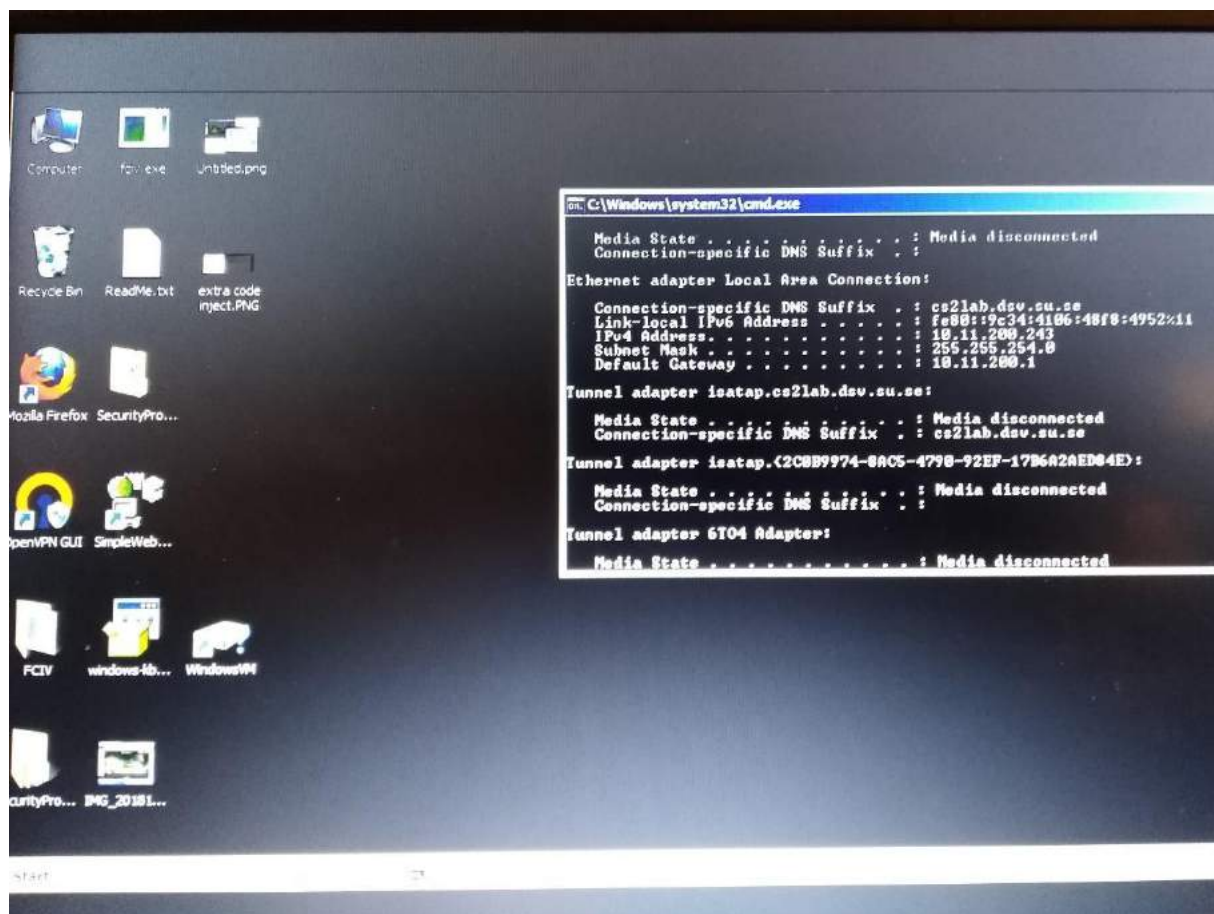


Figure 9 Access victim's computer

Laboratory Assignments for Kali Linux

Exercise 1. Utilizing port- and vulnerability scanners

In this task we used Nmap, In order to start Nmap Open a terminal window and write “zenmap” for GUI based Nmap or we can also use terminal based Nmap.

For this task we have used zenmap to run different tests. Nmap can perform many scans with very useful information.

- Writing the “zenmap” command
- Choose Ping scan or Intense scan
- enter the address “10.11.200.0/24” in the Target field, and click “Scan”

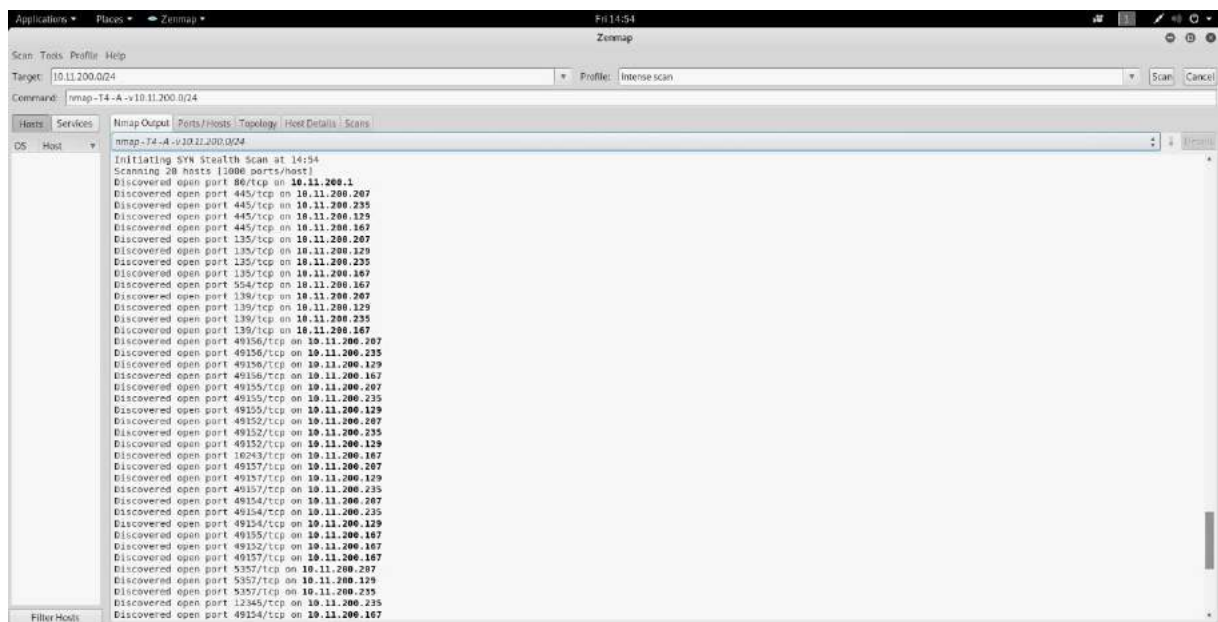


Figure-k1

Enter the address “10.11.200.129” (windows IP) in the target field, and click “Scan”

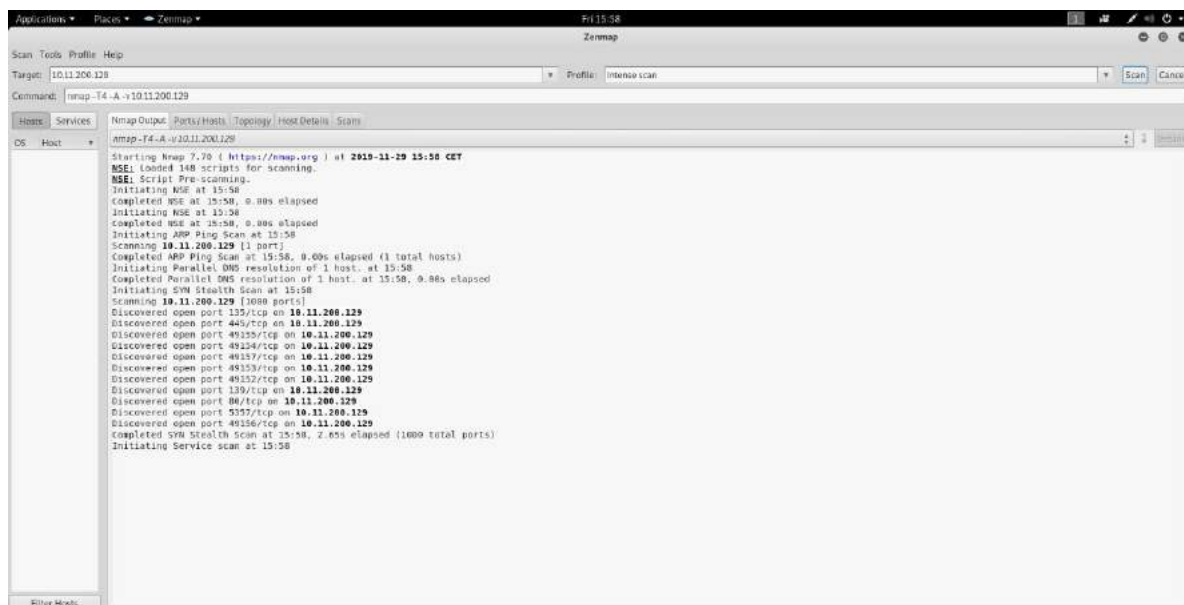


Figure-k2

Open the “nc.exe” backdoor, then enter the address “10.11.200.129” (windows IP) in the target field, and click “Scan”

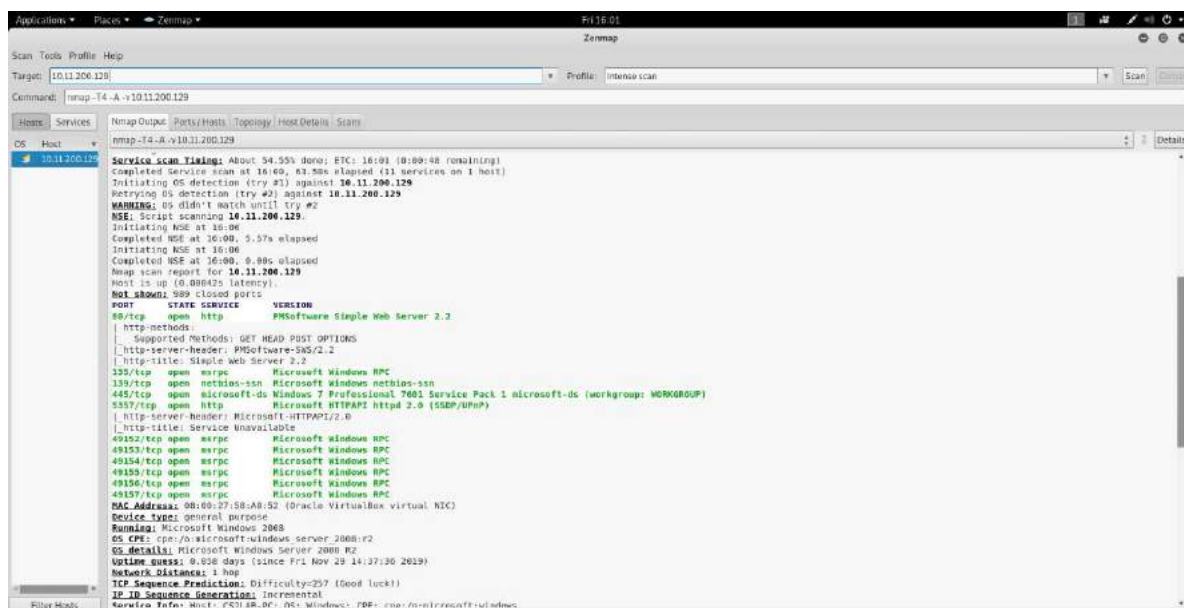


Figure-k3

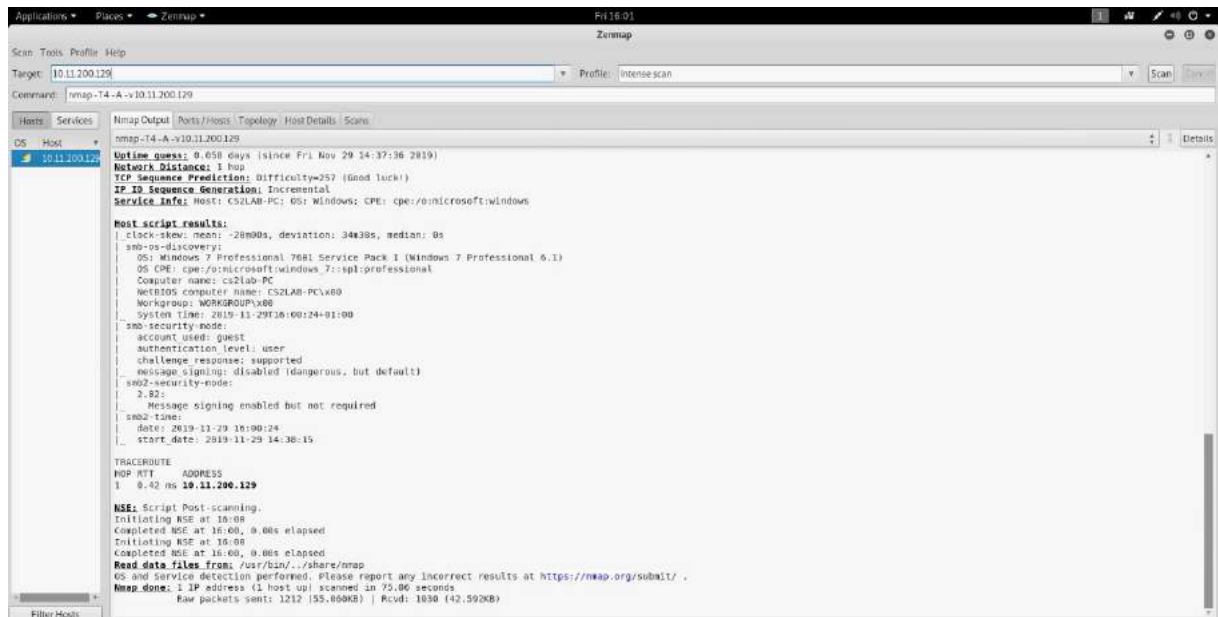


Figure-k4

Questions:

(1) which ports are open?

Open ports are 80, 445, 135, 554, 139, 49156, 49155 and so on.

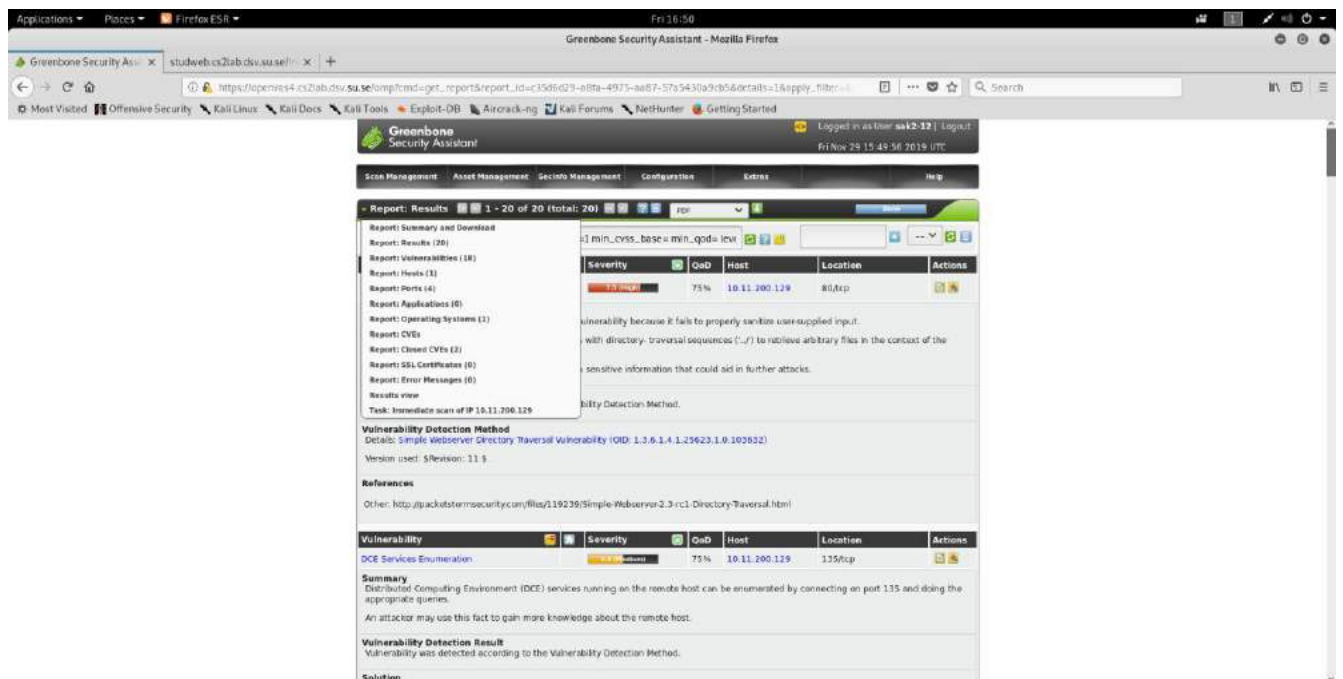
(2) Is there any difference in scanning Windows or Linux computer?

- By scanning windows, there are a little more information presented like the accurate time to complete scan(Figure-k2);
- Scanning Linux computer only presents discovered open ports(Figure-k1);
- If opening a backdoor, there are much more information presenting(Figure-k3, Figure-k4), like port, state service, version and MAC address.

OpenVAS:

In this Exercise task we will use the OpenVas vulnerability scanner and we will try to scan different Operating systems for possible vulnerabilities.

- Login to the OpenVas web server with the credentials obtained from the provided link.
- Create new targets, i.e. Windows, Kali or Metasploitable server.
- We obtained the IP-addresses of the target windows OS with “IPCONFIG” and for Kali and metasploitable we used “IFCONFIG” in a command shell.
- Now add new task with a specific target OS and initiate the scan.
- When the task finishes do the same steps for other Operating systems as well.
- The following is the result of scanning windows:



- Simple Webserver is prone to a directory-traversal vulnerability because it fails to properly sanitize user-supplied input. Exploiting this issue may allow an attacker to obtain sensitive information that could aid in further attacks.
- Distributed Computing Environment (DCE) services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries. An attacker may use this fact to gain more knowledge about the remote host.
- Calibre is prone to a cross-site scripting vulnerability and a directory- traversal vulnerability because it fails to sufficiently sanitize user- supplied input. This may

let the attacker steal cookie-based authentication credentials and other harvested information may aid in launching further attacks.

Exercise 2. Utilizing sniffer tools

- we tried to run the webspay and Dsniff in kali at eth0 interface along with firefox in host
- kali and target windows. It did not work for us
- Enter the command "ettercap -G".
- Then run "man ettercap_curses"



Figure K-6

Exercise 3. Analyzing network traffic

Tcpdump:

In this Task we will use TcpDump which is used for network monitoring, protocol debugging and data acquisition.

- In a terminal in kali, write "tcpdump -i ethN"
- Tcpdump will start and will be monitoring at interface for network traffic
- Let it run for some time, after some time stop the process by pressing Control + Z.
- In order to view the traffic that tcpdump intercepted write **tcpdump -i ethN -w filename**
- This will save the information in a text file.

```

File Edit Format View Help
tcpdump: verbose output suppressed, use -v or -vv for full protocol decoding
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
13:20:759731 IP kali.52748 > fra07s63-in-f141.1e100.net.https: Flags [P.], seq 613450726:613450772, ack 4070519991, win 3065, options [nop,nop,TS val 1009891824 ecr 4291189767], length 46
13:31:20.761711 IP fra07s63-in-f141.1e100.net.https > kali.52748: Flags [P.], seq 1:47, ack 46, win 307, options [nop,nop,TS val 4291247782 ecr 1009891824], length 4613:31:20.761751 IP kali.52748 > fra07s63-in-f141.1e100.net.https: Flags [.], ack 47, win 3065, options [nop,nop,TS val 1009891826 ecr 4291247782], length 013:31:20.807853 IP kali.51298 > dns-resolver.it.su.se.domain: 30091+ PTR? 141.21.217.172.in-addr.arpa. (45)13:31:20.809899 IP dns-resolver.it.su.se.domain > kali.51298: 30091 4/0/0 PTR fra07s63-in-f141.1e100.net., PTR fra07s63-in-f141.1e100.net., PTR arn11s02-in-f13.1e100.net., PTR arn11s02-in-f13.1e100.net. (143)13:31:20.810313 IP kali.46439 > dns-resolver.it.su.se.domain: 21018+ PTR? 162.200.11.10.in-addr.arpa. (44)13:31:20.815270 IP dns-resolver.it.su.se.domain > kali.46439: 21018 NXDomain 0/1/0 (104)13:31:20.816579 IP kali.53108 > dns-resolver.it.su.se.domain: 41235+ PTR? 200.30.11.193.in-addr.arpa. (44)13:31:20.818265 IP dns-resolver.it.su.se.domain > kali.53108: 41235 1/3/0 PTR dns-resolver.it.su.se. (133)13:31:22.026665 STP 802.1w, Rapid STP, Flags [Learn, Forward, Agreement], bridge-id 8000.88:f0:77:da:84:8c.803e, length 4313:31:22.144970 LLDP, length 52: cupboardswitch13:31:24.026535 STP 802.1w, Rapid STP, Flags [Learn, Forward, Agreement], bridge-id 8000.88:f0:77:da:84:8c.803e, length 4313:31:24.753134 IP kali.53428 > fra07s63-in-f131.1e100.net.https: Flags [P.], seq 3844214245:3844214291, ack 2626734334, win 296, options [nop,nop,TS val 2115880398 ecr 4055026777], length 4613:31:24.753531 IP kali.43510 > dns-resolver.it.su.se.domain: 23076+ PTR? 131.21.217.172.in-addr.arpa. (45)13:31:24.753849 IP kali.55730 > arn11s02-in-f5.1e100.net.https: Flags [P.], seq 721363908:721363954, ack 2660859541, win 257, options [nop,nop,TS val 3370461772 ecr 3792479865], length 4613:31:24.755028 IP dns-resolver.it.su.se.domain > kali.43510: 23076 4/0/0 PTR fra07s63-in-f131.1e100.net., PTR arn11s02-in-f3.1e100.net., PTR arn11s02-in-f3.1e100.net., PTR fra07s63-in-f131.1e100.net. (142)13:31:24.755396 IP kali.37713 > dns-resolver.it.su.se.domain: 47541+ PTR? 133.21.217.172.in-addr.arpa. (45)13:31:24.757199 IP dns-resolver.it.su.se.domain > kali.37713: 47541 4/0/0 PTR arn11s02-in-f5.1e100.net., PTR fra07s63-in-f133.1e100.net., PTR fra07s63-in-f133.1e100.net., PTR arn11s02-in-f5.1e100.net. (142)13:31:24.793886 IP fra07s63-in-f131.1e100.net.https > kali.53428: Flags [.], ack 46, win 244, options [nop,nop,TS val 4055032812 ecr 2115880398], length 013:31:24.795976 IP arn11s02-in-f5.1e100.net.https > kali.55730: Flags [.], ack 46, win 248, options [nop,nop,TS val 3792485902 ecr

```

Figure 7: Tcp log file.

- Tcpdump logged the info in the file which we specified earlier.
- By going through the logs we can see that it has timestamps of the network operation with microsecond resolution.
- With the logs info we can determine the network latency packet by packet.
- These logs can be imported in Wireshark etc. for post analysis
- As we can see with the operations of tcpdump it will be a major overhead in the system and the performance can vary with a busy interface. It would be ideal to use it with some filter expressions to reduce the overhead.

Wireshark:

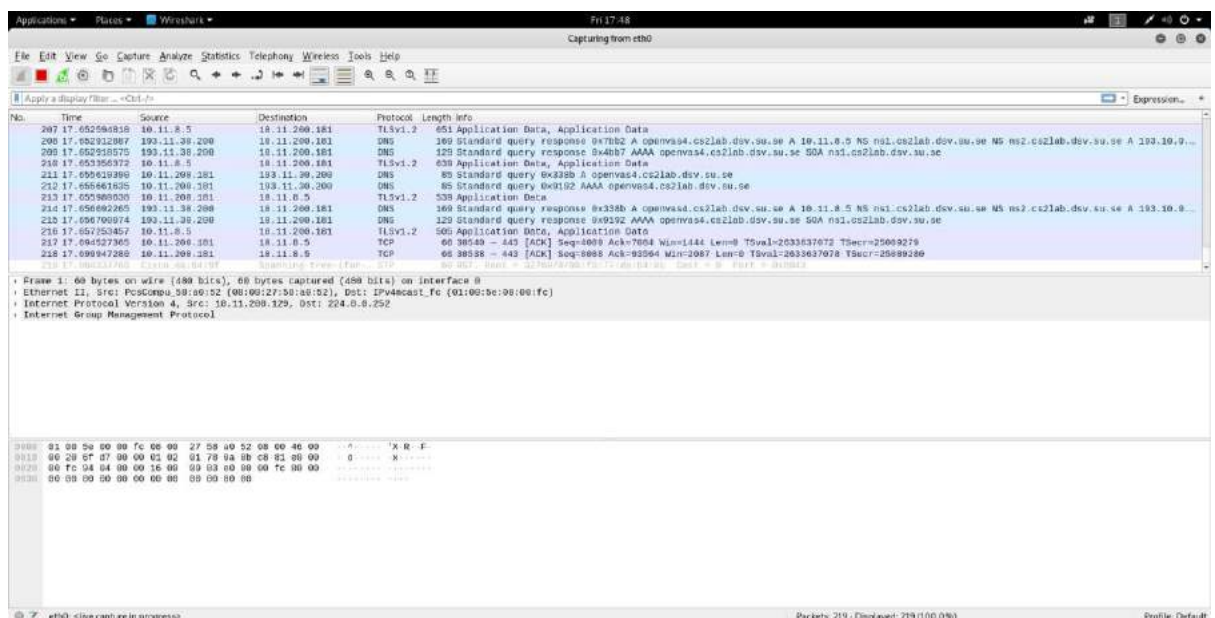


Figure 8: wireshark

In this task we will use Wireshark to capture some network traffic and get some useful information. Wireshark is very useful in terms of browsing the captured data, viewing the summary and detailed information.

- Start by writing Wireshark in terminal.
- Start the capture of live traffic at eth0 interface from the capture settings.
- We browsed the internet.
- Logged into google account.
- The corresponding tcp/ip connection can be seen in Wireshark interface.
- We can see the TCP SYN and ACK request when the login was in progress.
- Stop the capture and analyze the log in Wireshark.
- Wire shark will also flag some useful information which makes it notable to analyze

Snort:

To analyze the network and monitor the traffic we will use Snort for this task. When snort is started it starts to log the network packet information on screen which can be clearly seen in the command shell.

- Start snort by writing “snort -v -i ethN”.
- As soon as we hit enter snort will start to log the traffic information on screen.
- We will be able to see the logged information of live network traffic.
- We can save this information in a txt file using command “**snort -dvi ethN >snortlab.txt**”.
- We captured the network traffic for some time then stopped the capture and saved the data in the text file for further analysis.

Result:(After comparing these three Tcpdump,Wireshark and Snort).

- Tcpdump is a packet sniffer to capture the packets and saves the raw data in a dump file for further analysis[3].
- Wireshark can load the dump file and provide a user-friendly interface with sorting and filtering features for analysis which can capture packets in both from live network and from a saved capture file. It clearly presents sender IP address, receiver IP address, which protocol, length of the entire IP packet and information.
- Snort is similar with tcpdump but it is more focused on security end of packet sniffing. as we can use snort for IDS (Intrusion Detection System).

Exercise 4. Metasploit – h4Xing made easy:

In order to perform this task turn on windows and kali machines. On windows VM turn on the SimpleWebServer and keep it running in the background.

On kali VM perform these steps:

- Type the command “msfdb init” and then starting the GUI with the command “armitage” from a shell;
- Enter “sws” in the search field;
- Set “Targets” to 0=> SimpleWebServer2.2-rc2 / Windows XP SP3 / Windows ..
- Enter the Windows victim's IP(10.11.200.129) in the RHOST field;
- Click Launch.
- Run a command like `cd \Users\cs2lab\Desktop echo You're a beauty! > hack.txt`
- we checked the windows machine and the corresponding changes were clearly present.

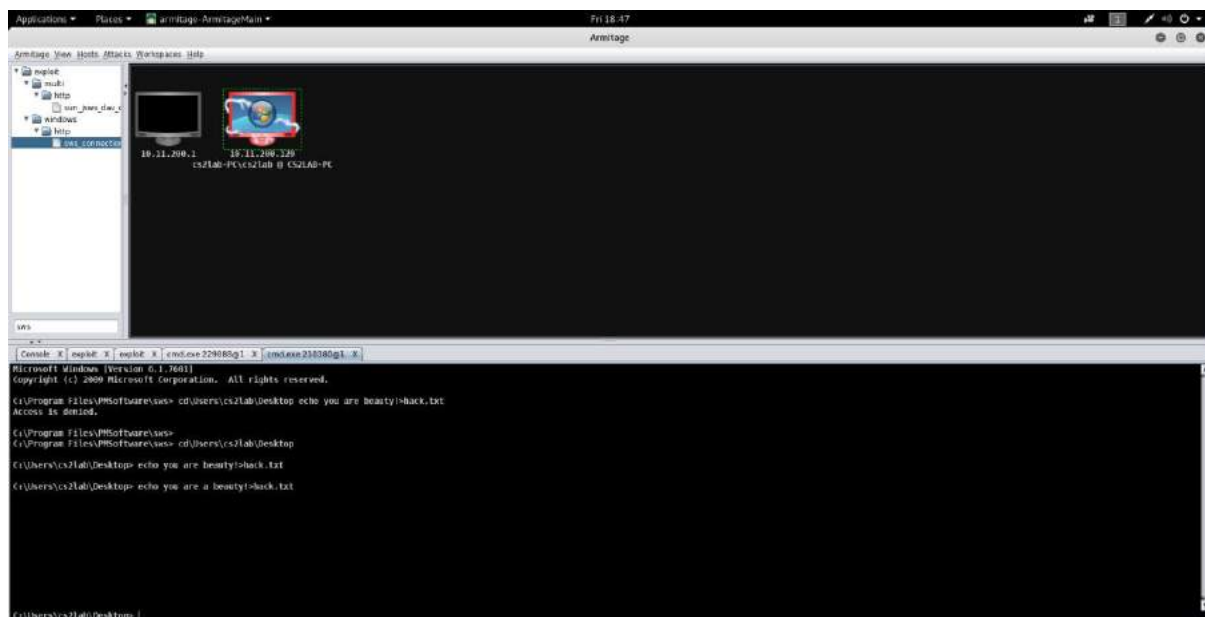


Figure 9: Armitage GUI interface

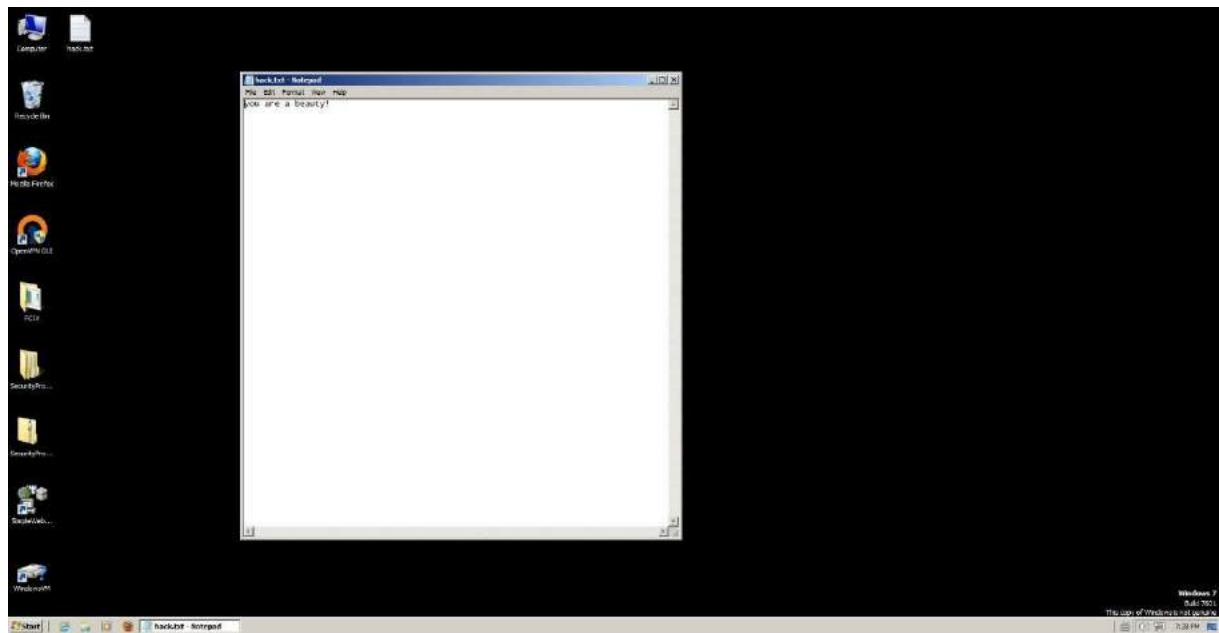


Figure 10

Reference:

[1] En.wikipedia.org. (2019). Code injection. [online] Available at:

https://en.wikipedia.org/wiki/Code_injection [Accessed 30 Nov. 2019]. [2] SearchSecurity. (2019). What is keylogger (keystroke logger or system monitor)? - Definition from WhatIs.com. [online] Available at:

<https://searchsecurity.techtarget.com/definition/keylogger> [Accessed 1 Dec. 2019]. [3] Sites.google.com. (2019). Traffic analysis (Snort/Tcpdump) - IAS on CS. [online]

Available at:

<https://sites.google.com/site/iasoncs/home/network-security/traffic-analysis-snort-tcpdump> [Accessed 1 Dec. 2019].