Internet Security – COS80013 Lab - 4 Report

Student ID: 104837257
Student Name: Arun Ragavendhar Arunachalam Palaniyappan
Lab Name: COS80013 Lab 4 – Malware
Lab Date: 28 /03/2025
Tutor: Yasas Akurudda Liyanage Don

Title and Introduction

This lab's learning was about how malware infects systems, hides in memory, and communicates with external servers. It covered spyware (Vundo), a Trojan (Arucer), and a Linux backdoor shell. Tools like Wireshark, Malwarebytes, netstat, Process Explorer, and Registry Editor were used to monitor, detect, and analyse these infections.

Methodology

The XP VM was started and Wireshark was launched to capture network traffic. After visiting a malicious site and running 1001Passwords.exe, new .dll files appeared in the System32 folder. Wireshark showed DNS/NetBIOS queries to suspicious domains like SEARCHMEUP.BIZ.

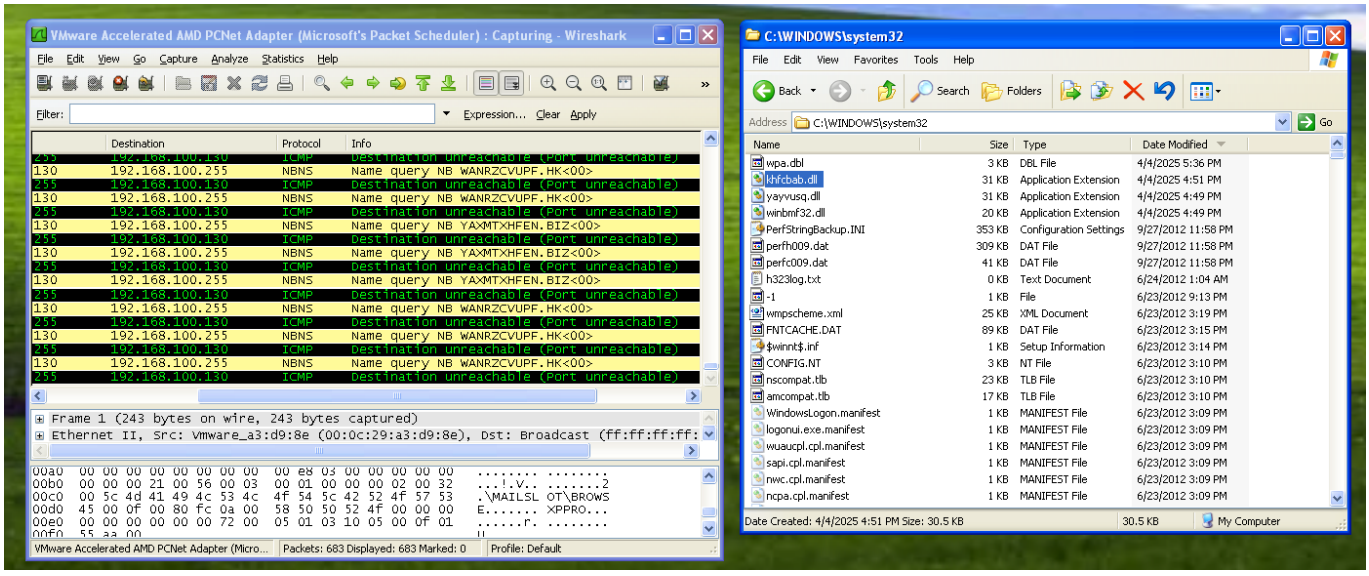
Malwarebytes was used to scan the system. It detected 6 Trojans, though some files couldn't be deleted due to being active in memory. The scan results were reviewed and removal was attempted.

Next, EnergisterDuoSetup.exe was used to simulate the Arucer Trojan. netstat -ao revealed port 7777 in use due to the trojan. Using Process Explorer and the PID (992), the active malicious process was located. Registry Editor confirmed the presence of Arucer via registry keys.

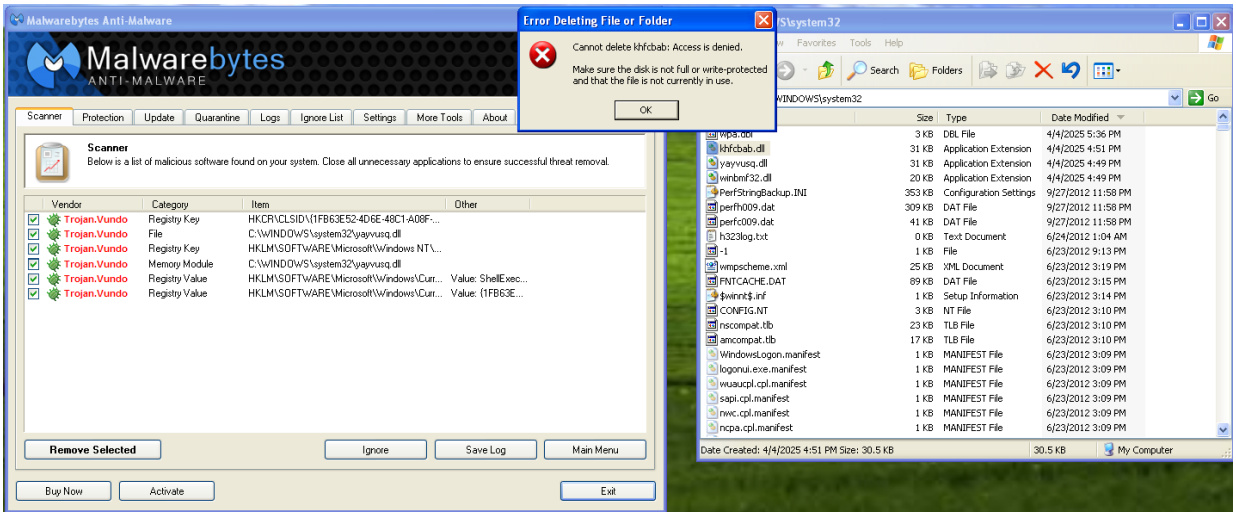
In Linux, an assembly file (hello1.asm) was compiled and run. A socket program was launched in the background to open a shell. From the XP VM, Wintcpclient.exe connected to Linux, confirming backdoor access through remote commands like ls and ps –al.

Data Recording and Screenshots

- Vundu Trojan -Wireshark packet captures showing name queries to suspicious sites and .dll files created in SYSTEM32 folder. The trojans are trying to contact their masters.



- Malwarebytes scan results and unable to delete .dll files created as its already running in memory.



- Once the EnergisterDuoSetup.exe file is run and installed, Arusar opens up Port 7777 and also creates a rundll32.exe file.

- Terminal output from Linux shell and remote access

```
dec     bl
mov     al,1
int     0x80
string:
call    code
db      'Hello World!',0x0a

[student@server student]$
[student@server student]$ cat fixasm
#!/bin/sh
nano $1.asm
nasm -f elf -o $1.o $1.asm
ld -o $1 $1.o
nasm -o $1.s $1.asm
./sproc -p $1.s
echo Running $1...
./$1
[student@server student]$ nasm -f elf -o hello1.o hello1.asm
[student@server student]$ ld -o hello1 hello1.o
[student@server student]$ chmod +x hello1
[student@server student]$ ./hello1
Hello World!
[student@server student]$ ./socket &
[1] 14743
[student@server student]$ _
```

The hello1.asm contain simple NASM assembly code. After running hello1, a socket was executed, a backdoor program that runs silently in the background and opens a listening port. This simulates a real-world backdoor, allowing remote access from another machine (like the XP VM) using Wintcpclient.exe—without any login.

Discussion and Learnings

Learning 1
I learned how spyware like Vundo can silently infect a system, add hidden files to system directories, and attempt to contact unknown domains using DNS and NetBIOS queries. Using Wireshark helped me capture and analyse this background traffic, showing how malware communicates without user awareness.

Real-World Cybersecurity Application 1
This technique is often used in threat hunting. Security analysts rely on tools like Wireshark or network monitors to detect unusual outbound DNS requests that might indicate malware try to reach to a command-and-control (C2) server.

Learning 2
I understood how Trojans like Arucer can open a hidden port (7777 in this case) to leak data. By using netstat, Process Explorer, and checking registry keys, I could trace and verify the malware's presence and persistence on the system.

Real-World Cybersecurity Application 2
This approach shows how security teams investigate suspicious network activity. Port scans and process monitoring are standard methods in detecting remote access Trojans (RATs) and identifying malicious processes in a compromised system.

Learning 3
I learned how simple shellcode in Linux, when executed, can silently open a backdoor shell for remote access. The Windows XP VM was then able to access the Linux system without credentials through the backdoor.

Real-World Cybersecurity Application 3
This is similar to how attackers use lightweight payloads to bypass authentication and gain shell access. Understanding this helps defenders recognise and block these techniques by hardening endpoints and monitoring for unusual background processes or open ports.

Limitation

Some malware couldn’t be fully removed without a shutdown. Outdated OS environments may behave differently from modern systems.