An Overview Report of Running Malware Analysis using a piece of Malware

Malware Analysis Report

Submitted by:Sunny Thakur

1

Table of Contents

Executive Summary

This report presents a comprehensive malware analysis conducted on a recently discovered real-world malware sample. The analysis focused on both static and dynamic malware behaviors using a variety of tools within controlled virtual environments. The primary objectives were to analyze how the malware affects the target system, dissect its inner workings, and document its behavioral attributes, providing insights for further cybersecurity research.

1.1 Introduction

The project entailed detailed malware analysis, conducted as part of the "ITIS 6330 - Malware Analysis" course. This hands-on project aimed to provide an in-depth understanding of malware behaviors, particularly focusing on its interactions with target systems. The tools used included both basic and advanced techniques to reverse-engineer and interpret the malware's functioning.

2. Project Overview

2.1 Motivation

The project was driven by a desire to demonstrate the complexities of malware behavior and to educate individuals on reverse-engineering techniques. The goal was to explore the practical aspects of how malware operates and to improve cybersecurity awareness among researchers.

2.2 Scope

The scope of the project was extensive, covering static and dynamic analysis of a malware sample. VirtualBox was used to set up Windows XP, Windows 7, and Ubuntu virtual environments for testing. The malware's multi-phase operations, including anti-debugging, sandbox evasion, and process spawning, were key areas of focus. The report also touches on encryption methods, data exfiltration, and the malware's communication with external domains.

. 3. Malware Analysis

3.1 Tools and Technologies

The following tools were employed:

- **Static Analysis**: CFF Explorer, PEID, PEView, Resource Hacker, Dependency Walker, VirusTotal
- **Dynamic Analysis**: Process Explorer, Process Monitor, ApateDNS, Wireshark, RegShot
- **Advanced Static Analysis**: IDA Pro, Ghidra, OllyDbg

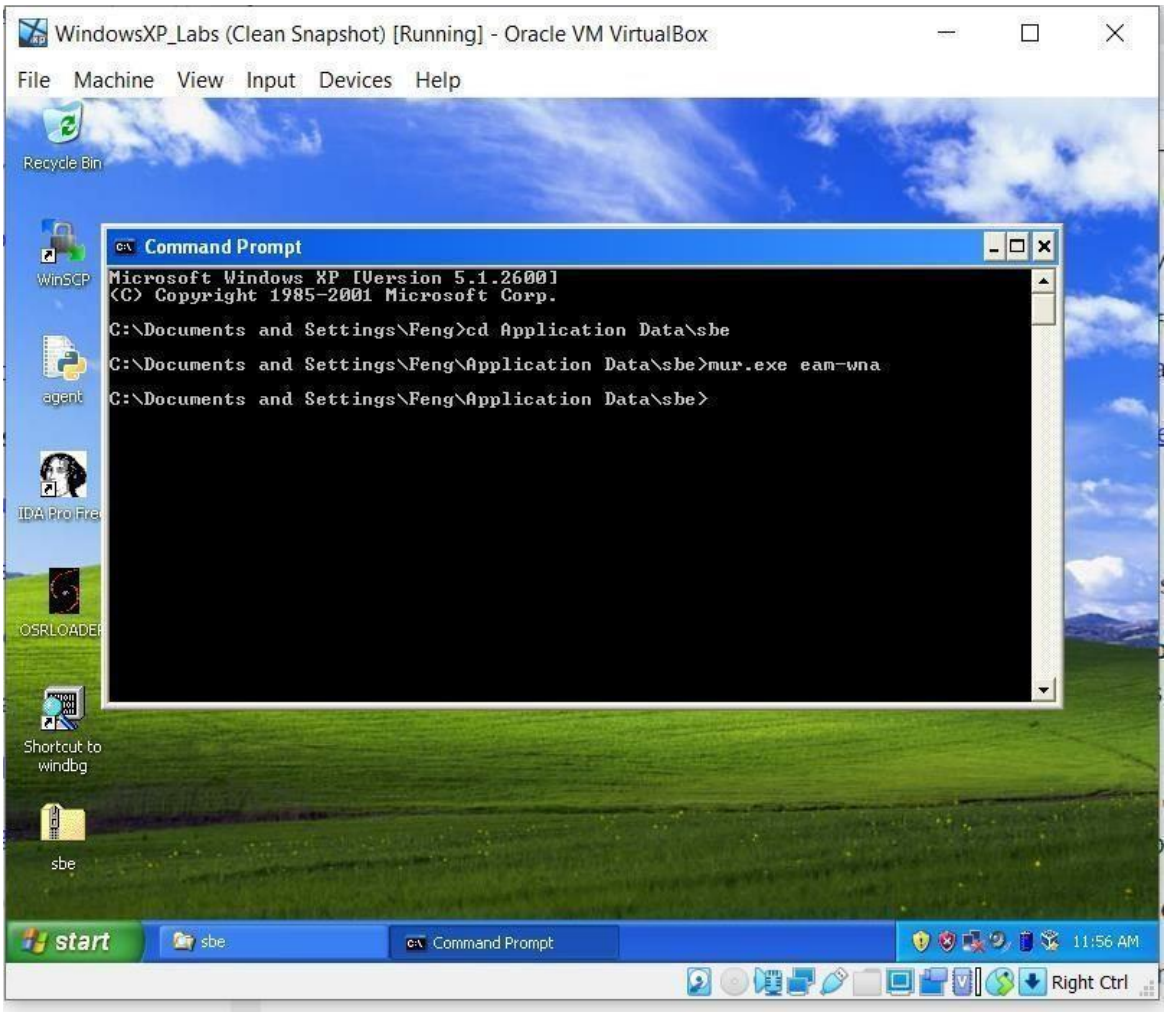
3.2 Environment Setup

The malware analysis was conducted in an isolated environment, utilizing Oracle VirtualBox with virtual machines running Windows XP, Windows 7, and Ubuntu. This controlled environment prevented unintended malware propagation while allowing for comprehensive analysis.

- .

3.3 Malware Behavior Analysis :

I have downloaded the file sbe.zip in order to do malware analysis on it. Then, I simply extracted it by entering the password "infected," and then moved the file to C:\Documents and Settings\Feng\Application Data directory on Windows XP to carryout the analysis. The next step is performing the analysis part by running the malware by placing in the folder Application Data and then **mur.exe eam-wna** is executed in the command prompt and then malware just started running (Note: It may shutdown the VM but you should take snapshot of it and then run again from snapshot).

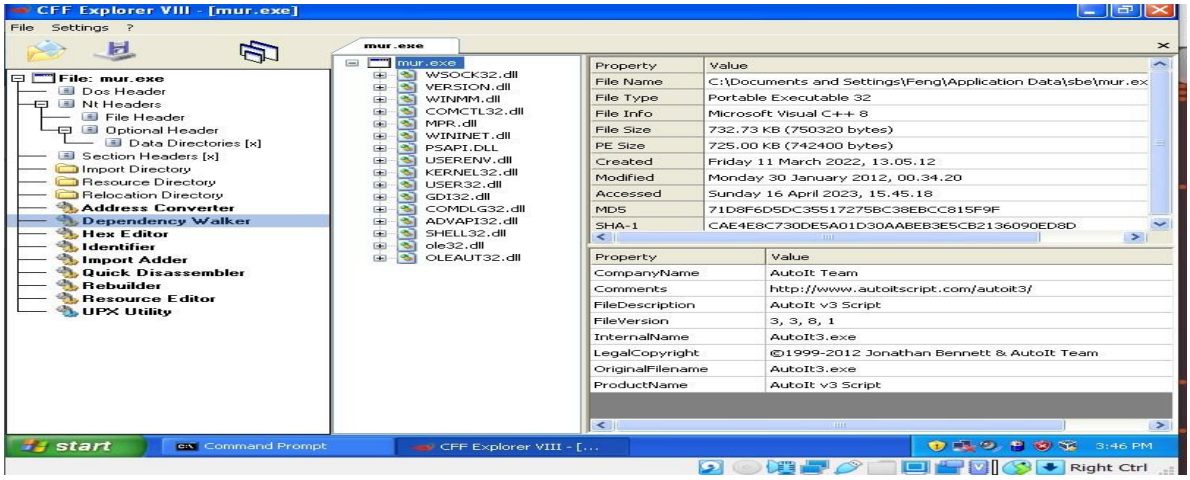


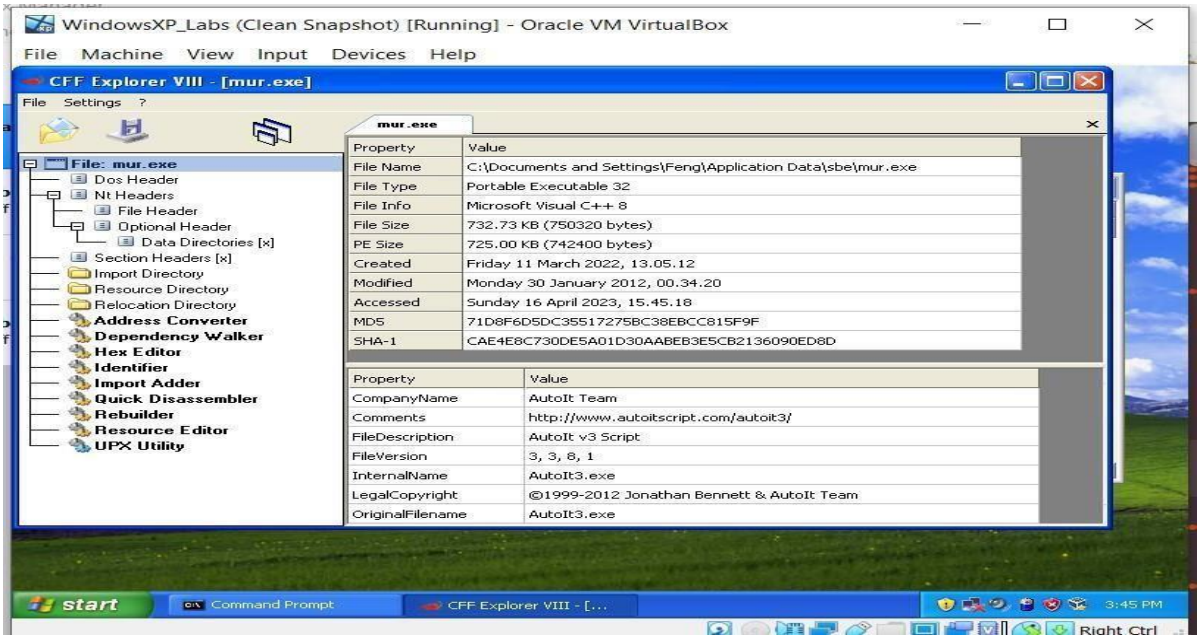
We then run Basic Static Analysis of the malware behavior :

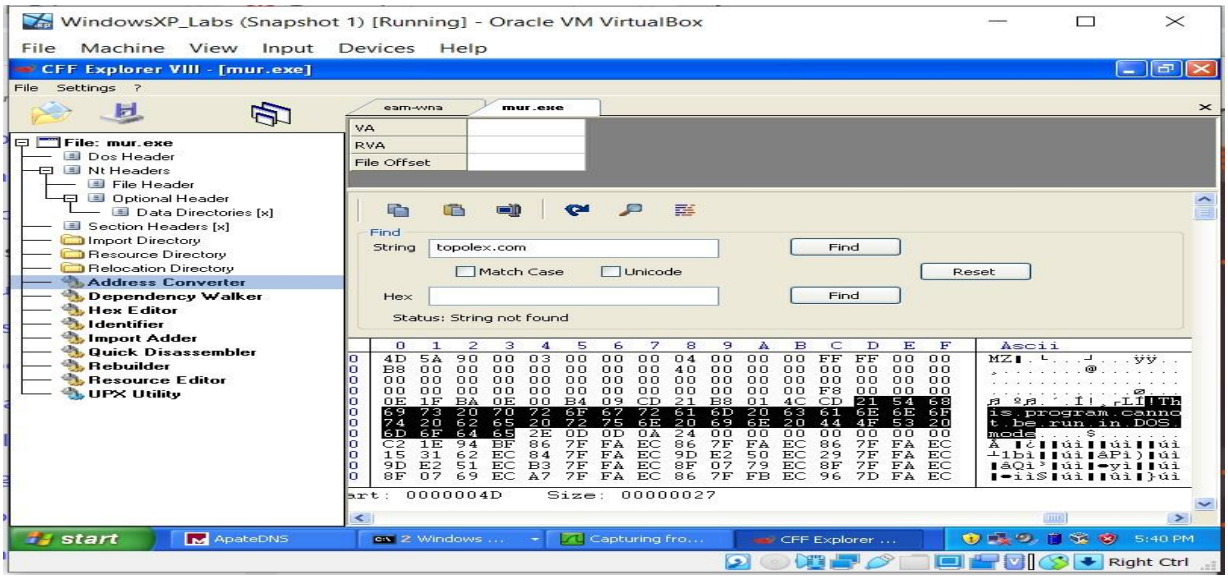
We used the static analysis tools such as: CFF Explorer, PEID etc., as you can check from the following steps :

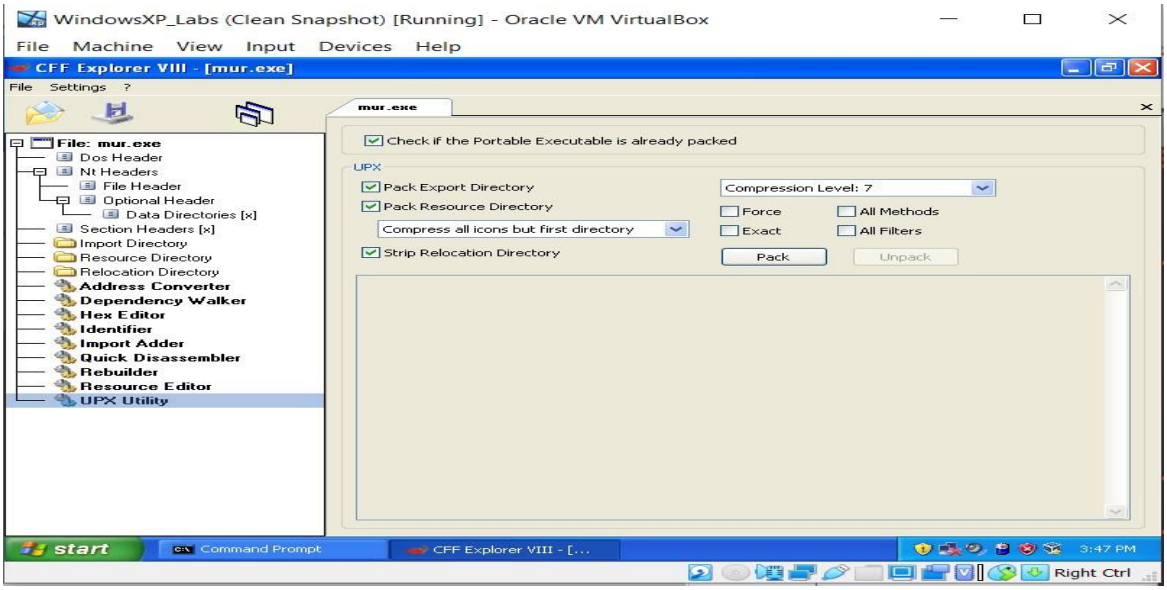
Using CFF Explorer :

With the hep of CFF Explorer, we have seen the Imports Directory, check whether malware is packed or not.


CFF Explorer VIII - [mur.exe]
File Settings ?
File: mur.exe
Dos Header
Nt Headers
File Header
Optional Header
Data Directories [x]
Section Headers [x]
Import Directory
Resource Directory
Relocation Directory
Address Converter
Dependency Walker
Hex Editor
Identifier
Import Adder
Quick Disassembler
Rebuilder
Resource Editor
UPX Utility
mur.exe
WSOCK32.dll
VERSION.dll
WINMM.dll
COMCTL32.dll
MPR.dll
WININET.dll
PSAPI.DLL
USERENV.dll
KERNEL32.dll
USER32.dll
GDI32.dll
COMDLG32.dll
ADVAPI32.dll
SHELL32.dll
ole32.dll
OLEAUT32.dll
Property Value
File Name C:\Documents and Settings\Feng\Application Data\sbe\mur.ex
File Type Portable Executable 32
File Info Microsoft Visual C++ 8
File Size 732.73 KB (750320 bytes)
PE Size 725.00 KB (742400 bytes)
Created Friday 11 March 2022, 13.05.12
Modified Monday 30 January 2012, 00.34.20
Accessed Sunday 16 April 2023, 15.45.18
MD5 71D8F6D5DC35517275BC38EBCC815F9F
SHA-1 CAE4E8C730DE5A01D30AABEB3E5CB2136090ED8D
Property Value
CompanyName AutoIt Team
Comments http://www.autoitscript.com/autoit3/
FileDescription AutoIt v3 Script
FileVersion 3, 3, 8, 1
InternalName AutoIt3.exe
LegalCopyright ©1999-2012 Jonathan Bennett & AutoIt Team
OriginalFilename AutoIt3.exe
ProductName AutoIt v3 Script
start
Command Prompt
CFF Explorer VIII - [...
3:46 PM
Right Ctrl


WindowsXP_Labs (Clean Snapshot) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
CFF Explorer VIII - [mur.exe]
File Settings ?
File: mur.exe
Dos Header
Nt Headers
File Header
Optional Header
Data Directories [x]
Section Headers [x]
Import Directory
Resource Directory
Relocation Directory
Address Converter
Dependency Walker
Hex Editor
Identifier
Import Adder
Quick Disassembler
Rebuilder
Resource Editor
UPX Utility
mur.exe
Property Value
File Name C:\Documents and Settings\Feng\Application Data\sbe\mur.exe
File Type Portable Executable 32
File Info Microsoft Visual C++ 8
File Size 732.73 KB (750320 bytes)
PE Size 725.00 KB (742400 bytes)
Created Friday 11 March 2022, 13.05.12
Modified Monday 30 January 2012, 00.34.20
Accessed Sunday 16 April 2023, 15.45.18
MD5 71D8F6D5DC35517275BC38EBCC815F9F
SHA-1 CAE4E8C730DE5A01D30AABEB3E5CB2136090ED8D
Property Value
CompanyName AutoIt Team
Comments http://www.autoitscript.com/autoit3/
FileDescription AutoIt v3 Script
FileVersion 3, 3, 8, 1
InternalName AutoIt3.exe
LegalCopyright ©1999-2012 Jonathan Bennett & AutoIt Team
OriginalFilename AutoIt3.exe
start
Command Prompt
CFF Explorer VIII - [...
3:45 PM
Right Ctrl

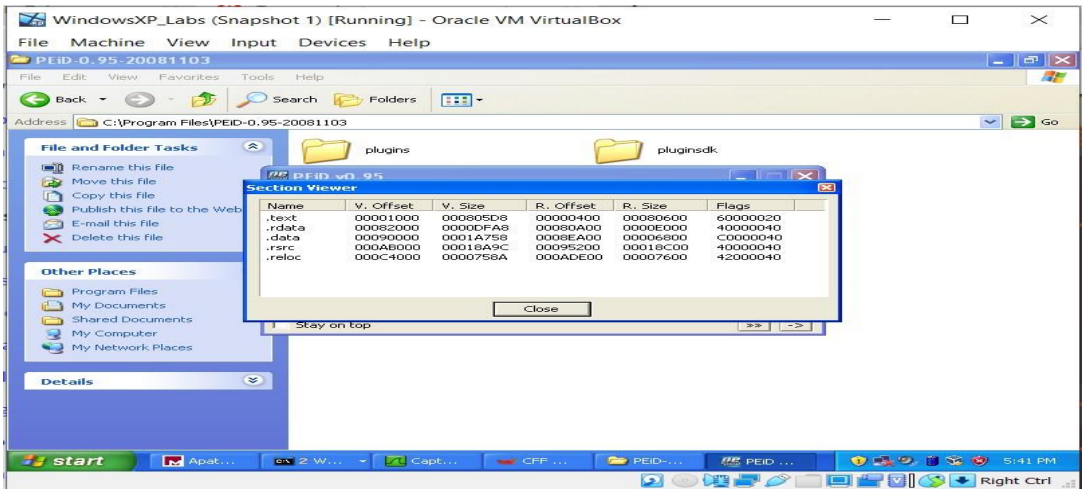


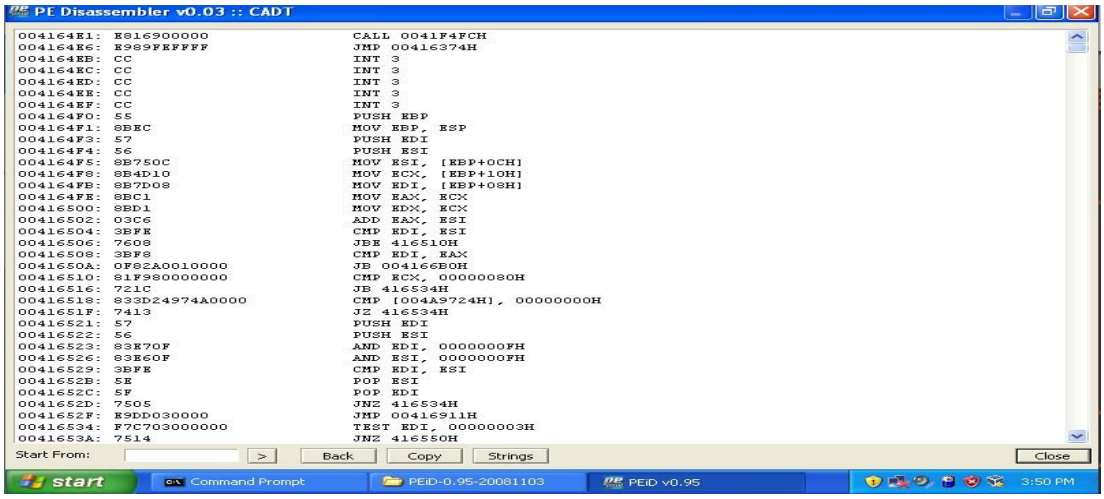


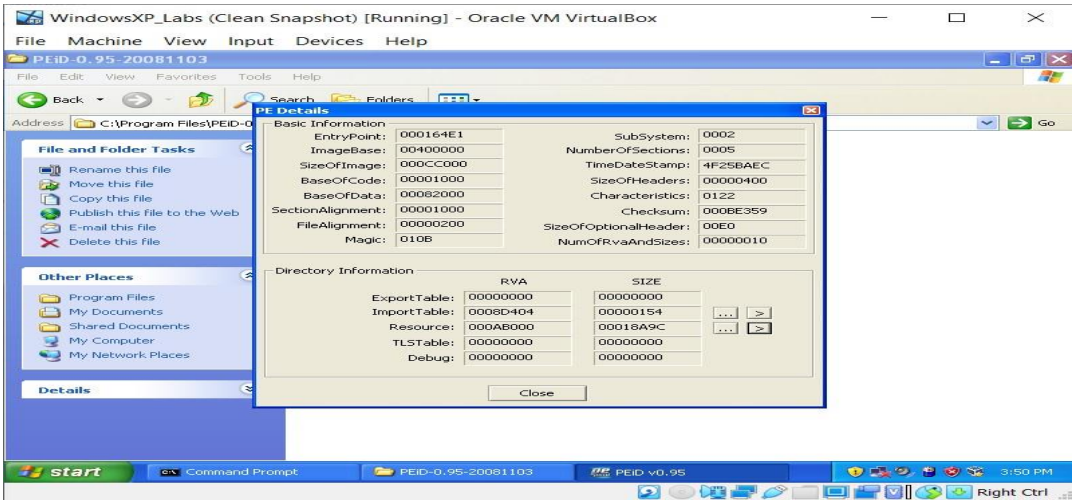
From this we can say malware is unpacked.

Using PEID tool :

It performs static analysis



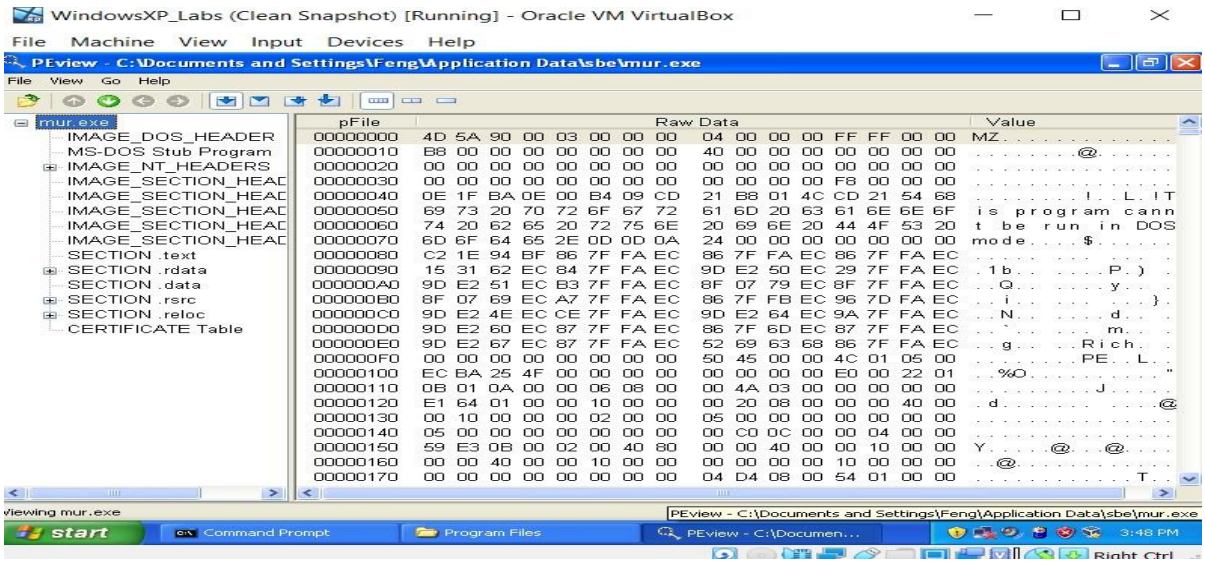




Using PEView Tool :

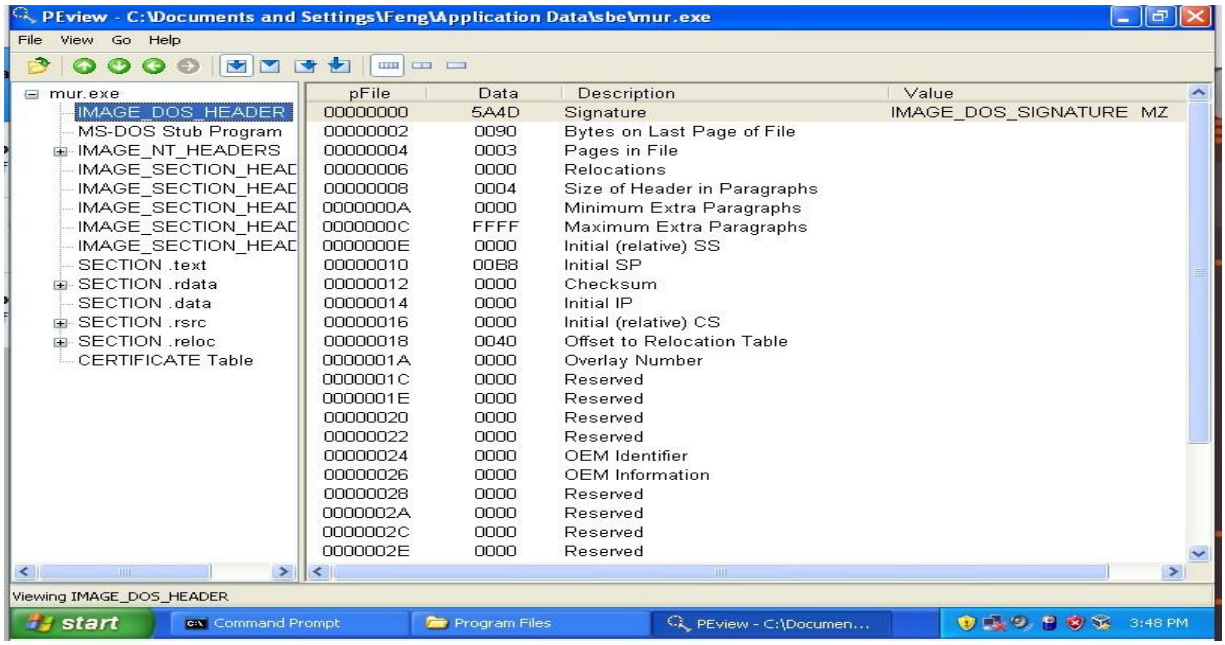
This is the another static analysis tool for the analysis of Malware Behavior and we got the following results after performing the analysis :

We generally interpreted the information in Hex View in the PEView tool:



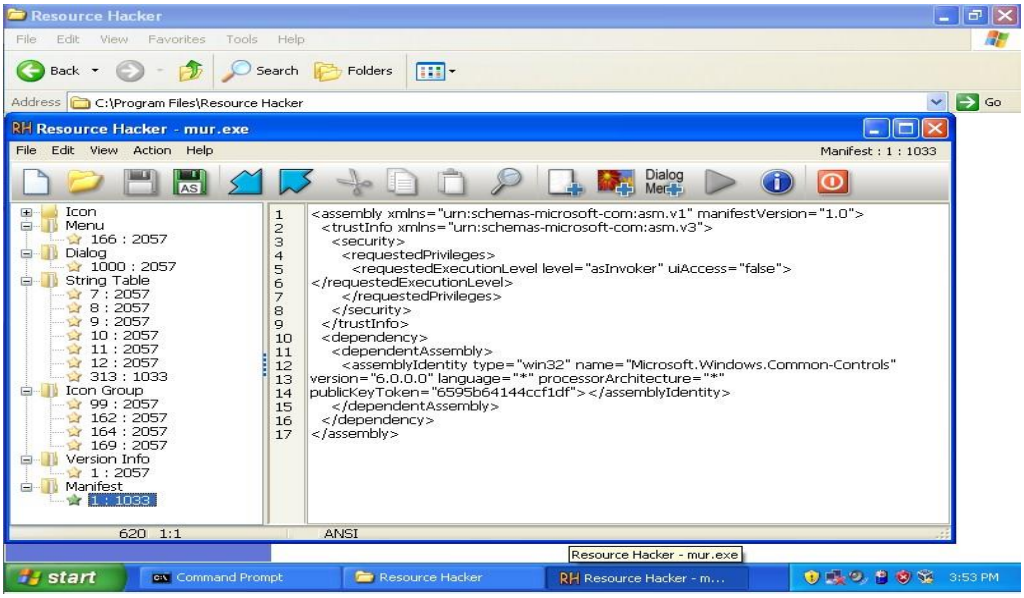
From the following image, we can see that the DOS Header is MZ which is interpreted as 5A4D in Hexadecimal format which represents the signature of the file and we can see the other details as follows in the image:

1. Bytes on Last Page of File
2. Pages in File
3. Size of Headers in Paragraphs
4. CheckSum
5. Offset to Relocation Tables
6. OEM Identifier, Information And also we can see some are Reserved



Use of Resource Hacker tool :

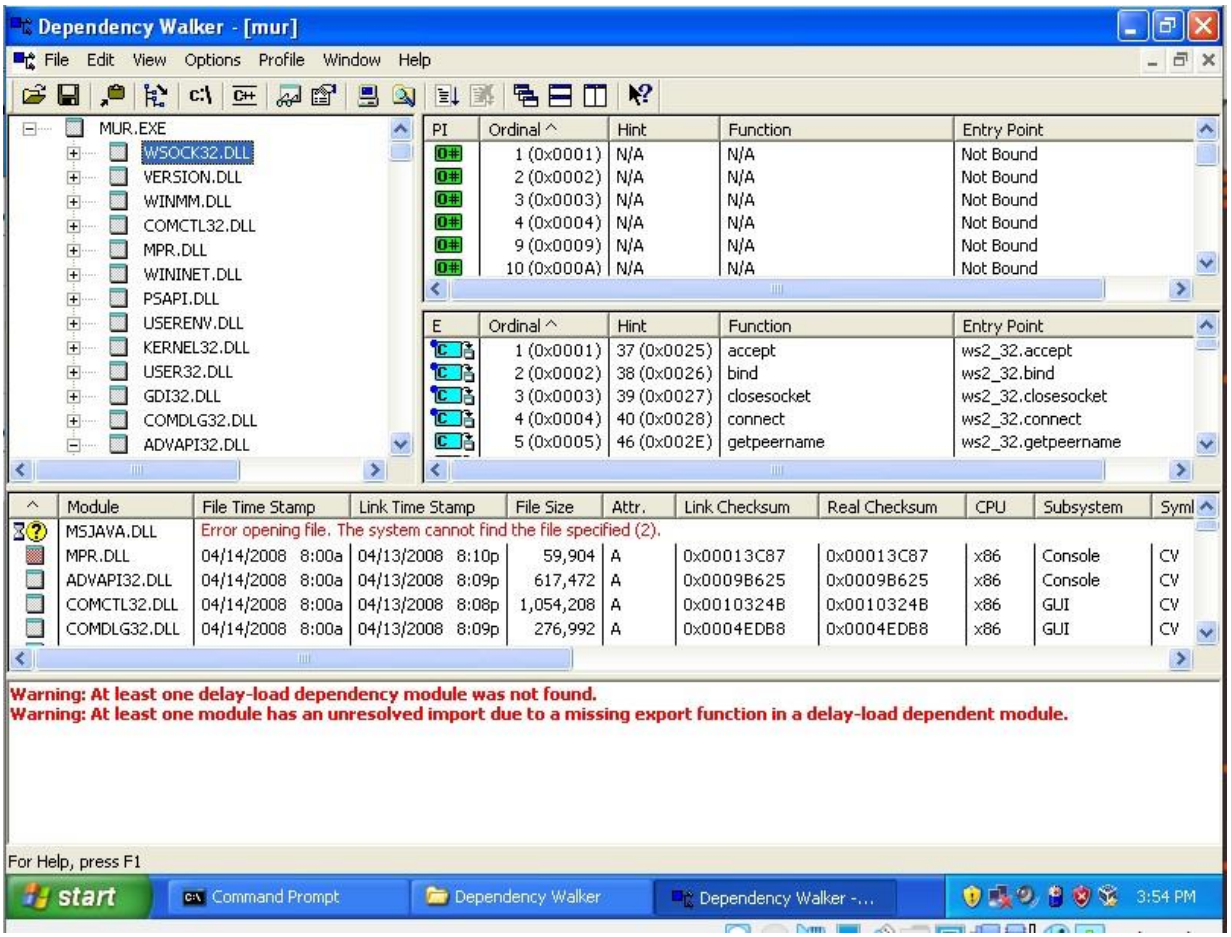
We open the mur.exe in resource hacker and get to know the resources the malware is utilizing to cause problems:



And then we save as .res file which we use to open in dependency walker.

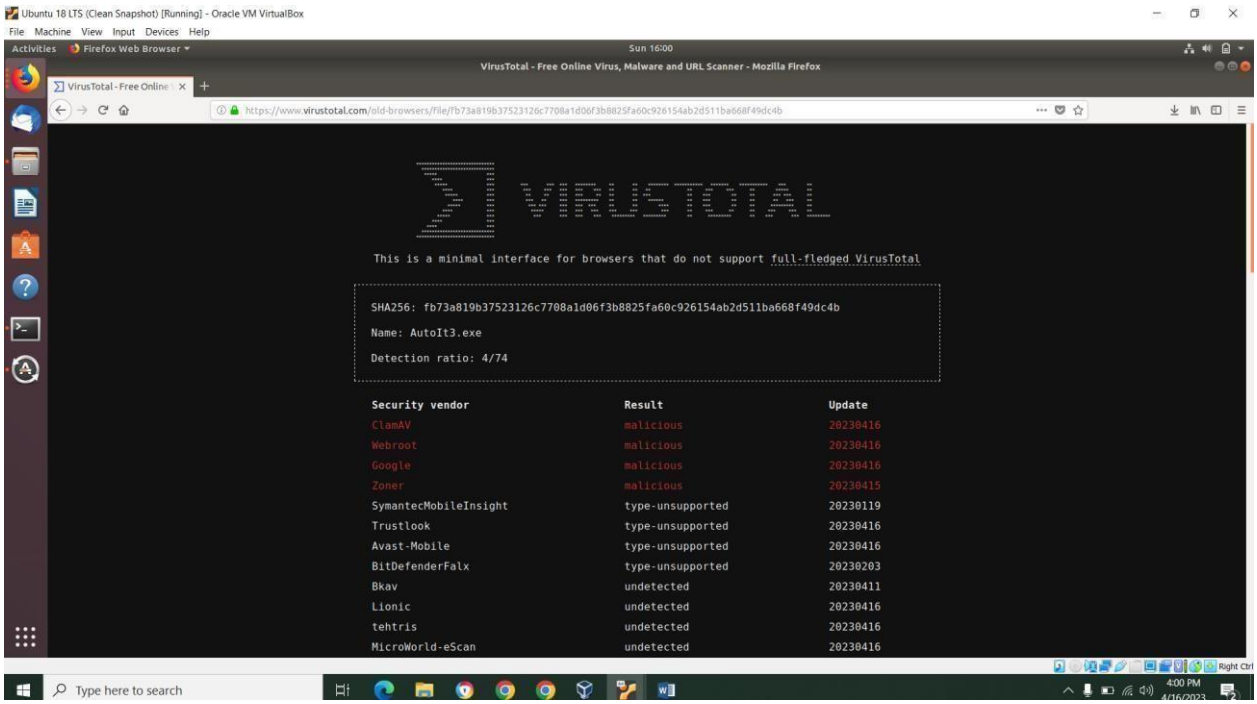
Use of Dependency Walker Tool:

We use that .res file and open in the dependency walker to list the DLL's and check for the bytes and different modules, functions.



Use of VirusTotal.com website:

With the help of the website , we can see that the Detection Ratio is 4/74 and the name as AutoIt3.exe

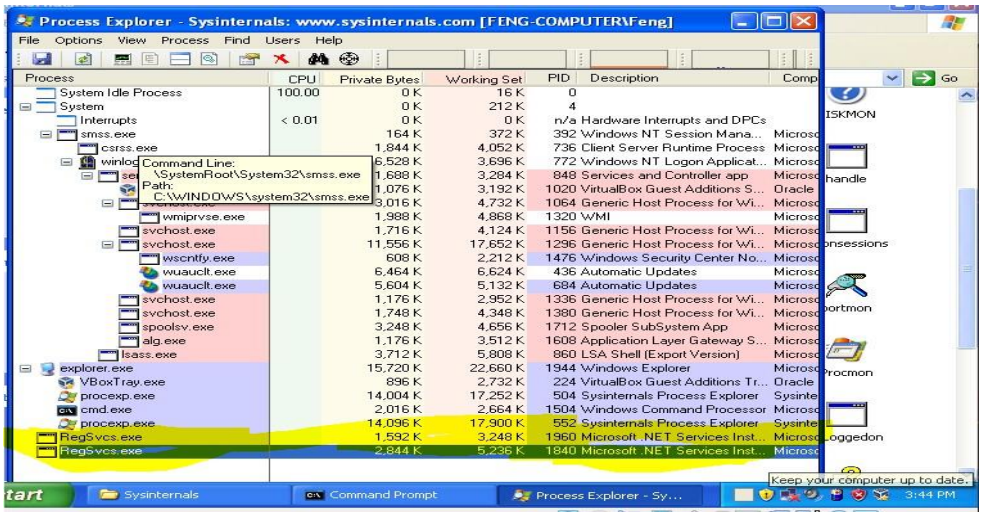


3.4 Running the dynamic analysis of malware:

We use the following tools to perform the dynamic analysis of malware behavior and as follows:

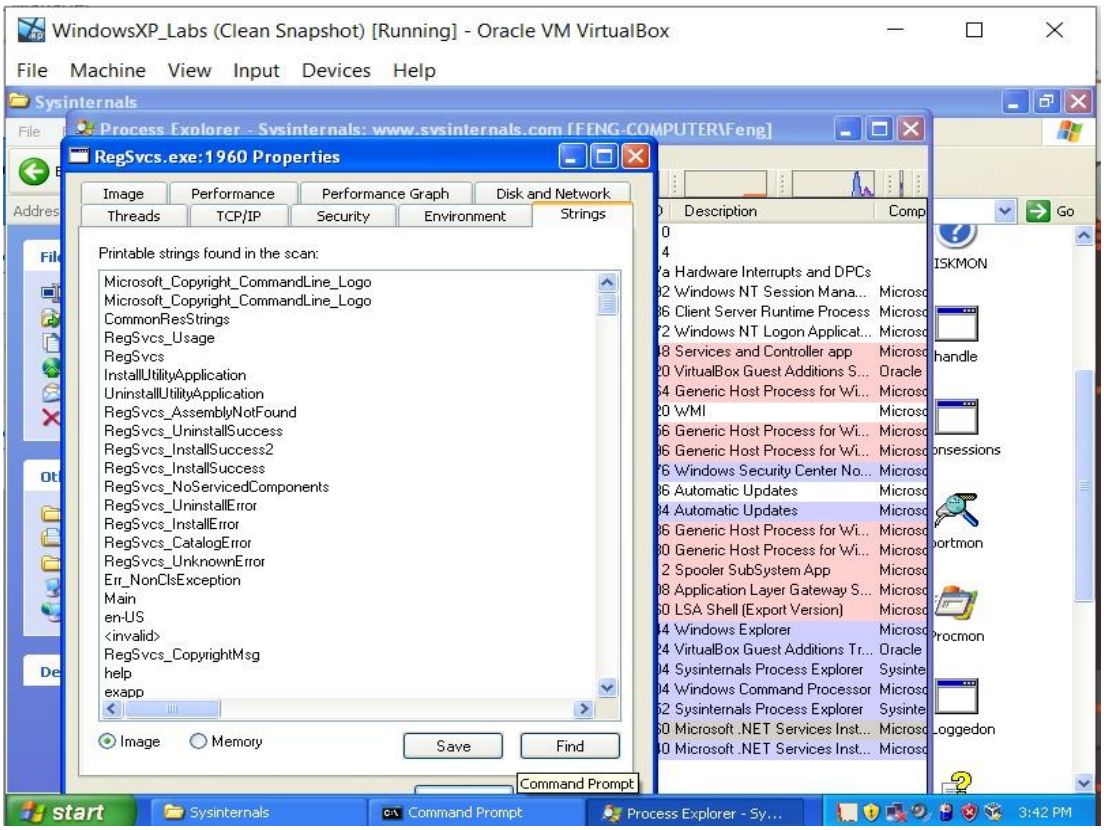
Use of Process Explorer for the analysis :

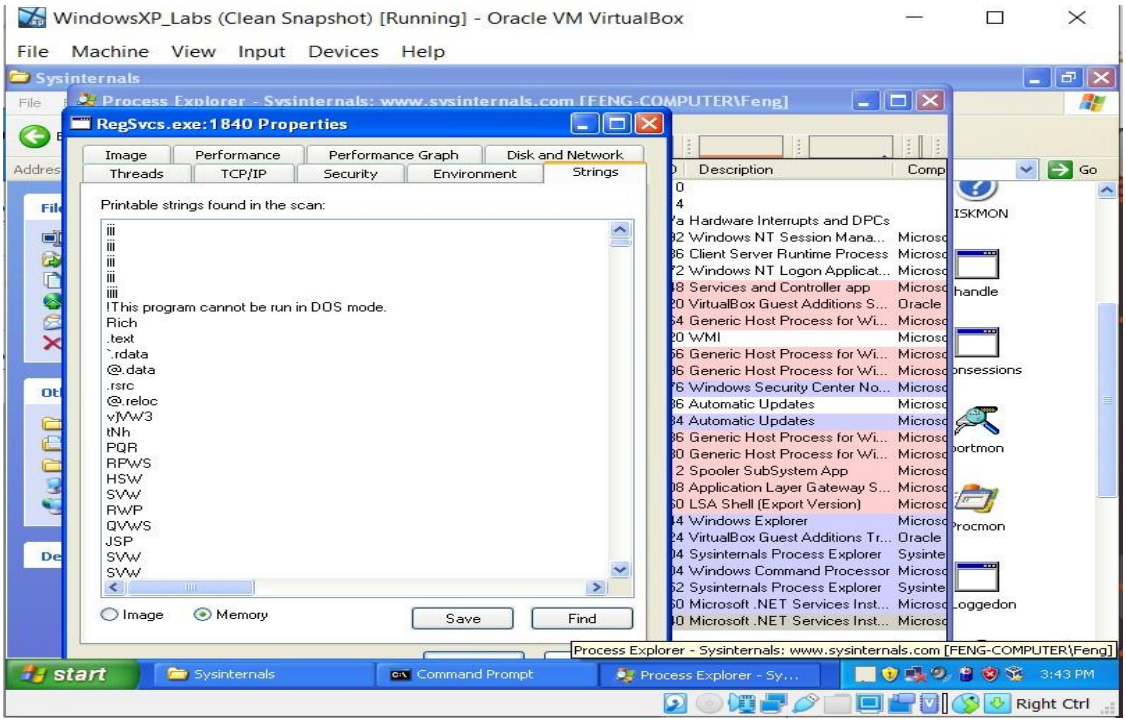
We have used this tool to perform the analysis and get the information of the processes and its child processes , strings from the image and memory and other details as follows from the following the screenshots :



We can see that it propagated the following child processes which are highlighted along with the process ID's 1960 and 1840 respectively.

We see the some properties, strings by double clicking on them both the processes:

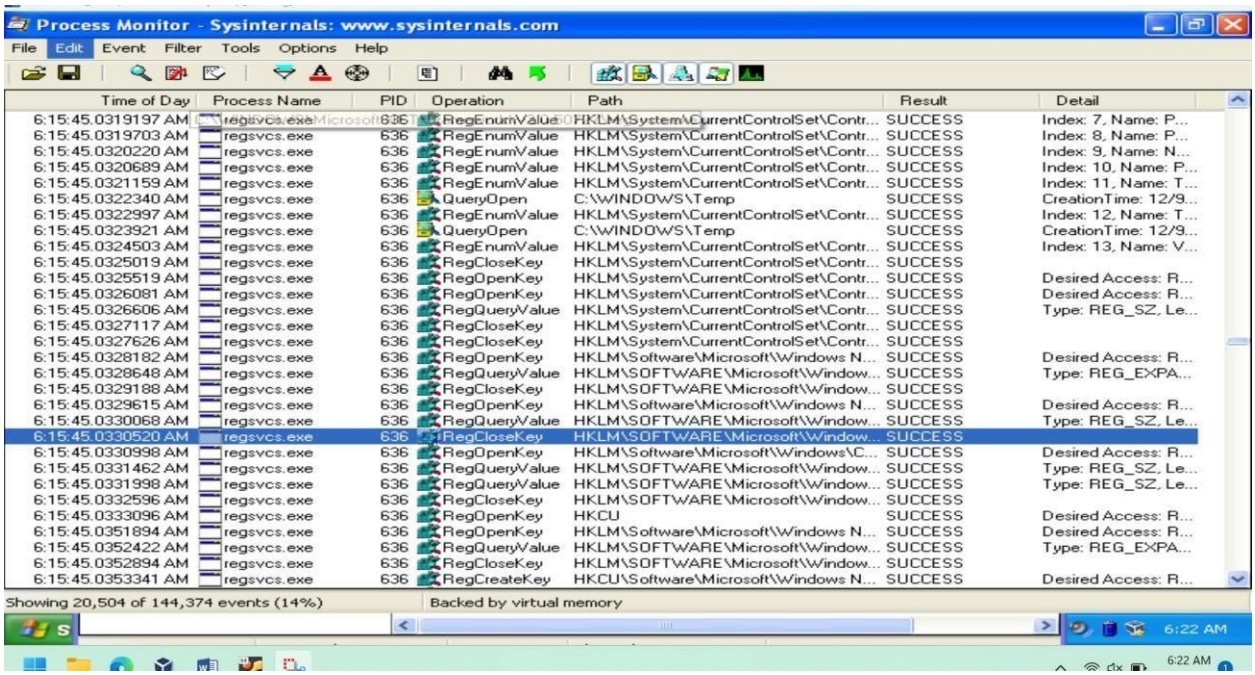


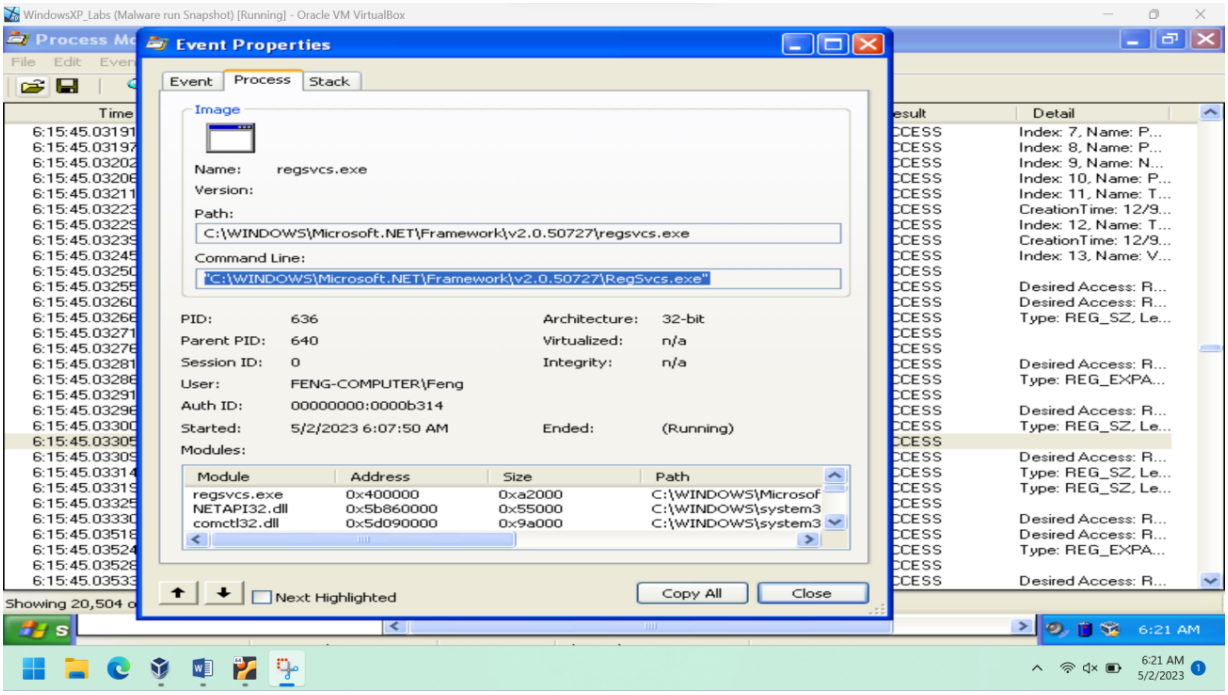


Use of Process Monitor Tool :

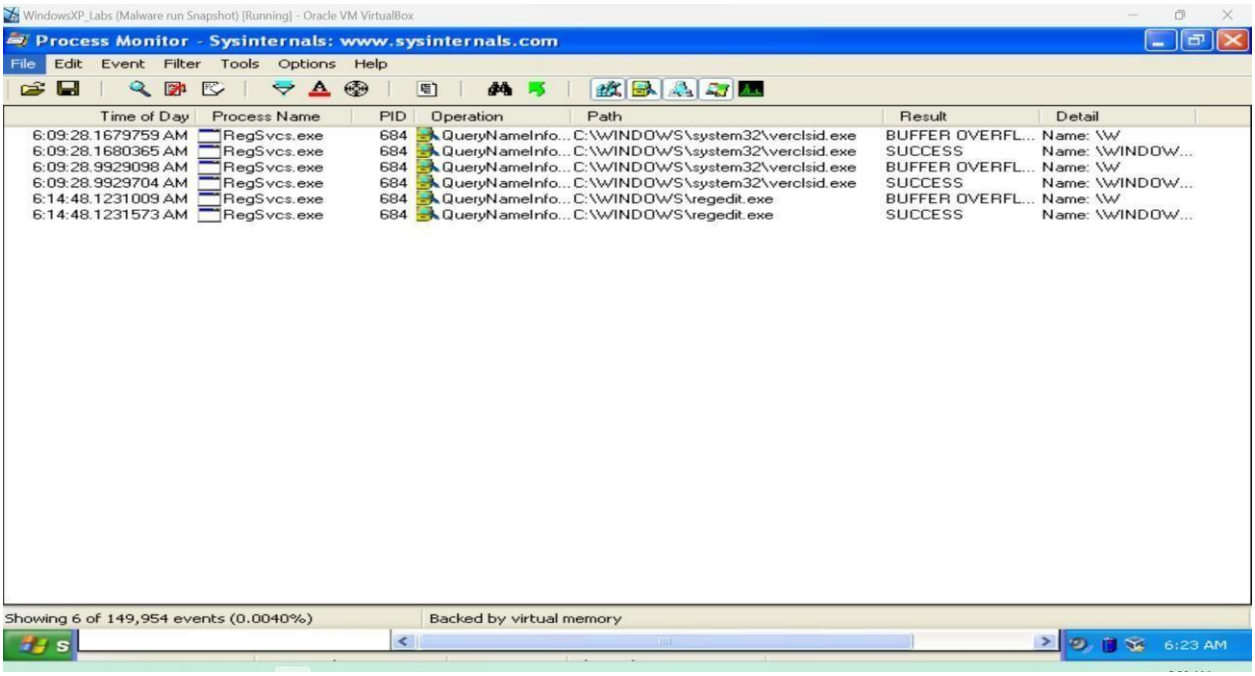
We used this tool to monitor the processes with the help of filtering the results by PID (Process ID).

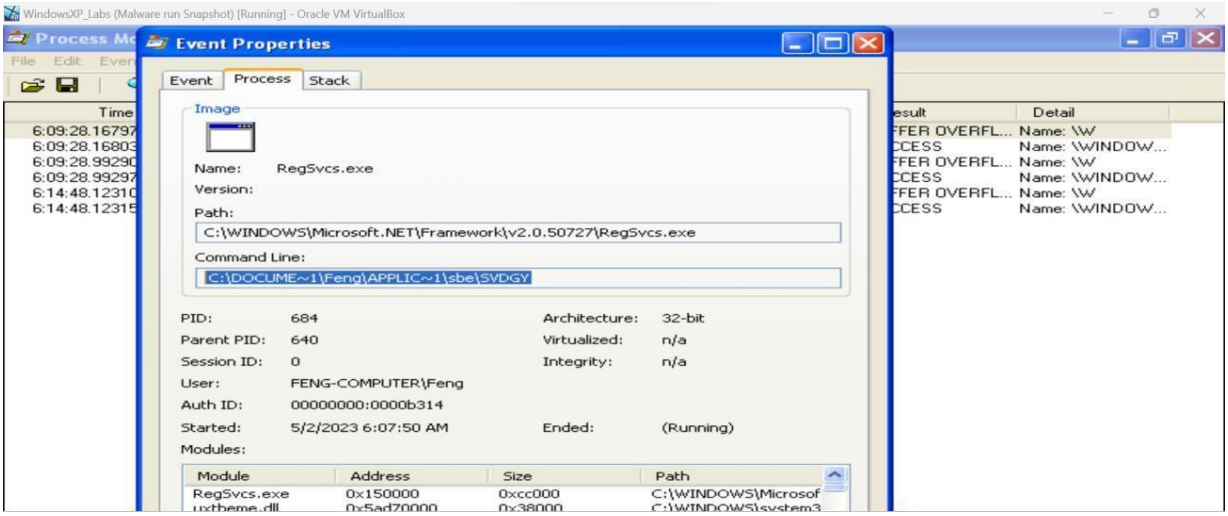
RegSvcs.exe with the PID: 636

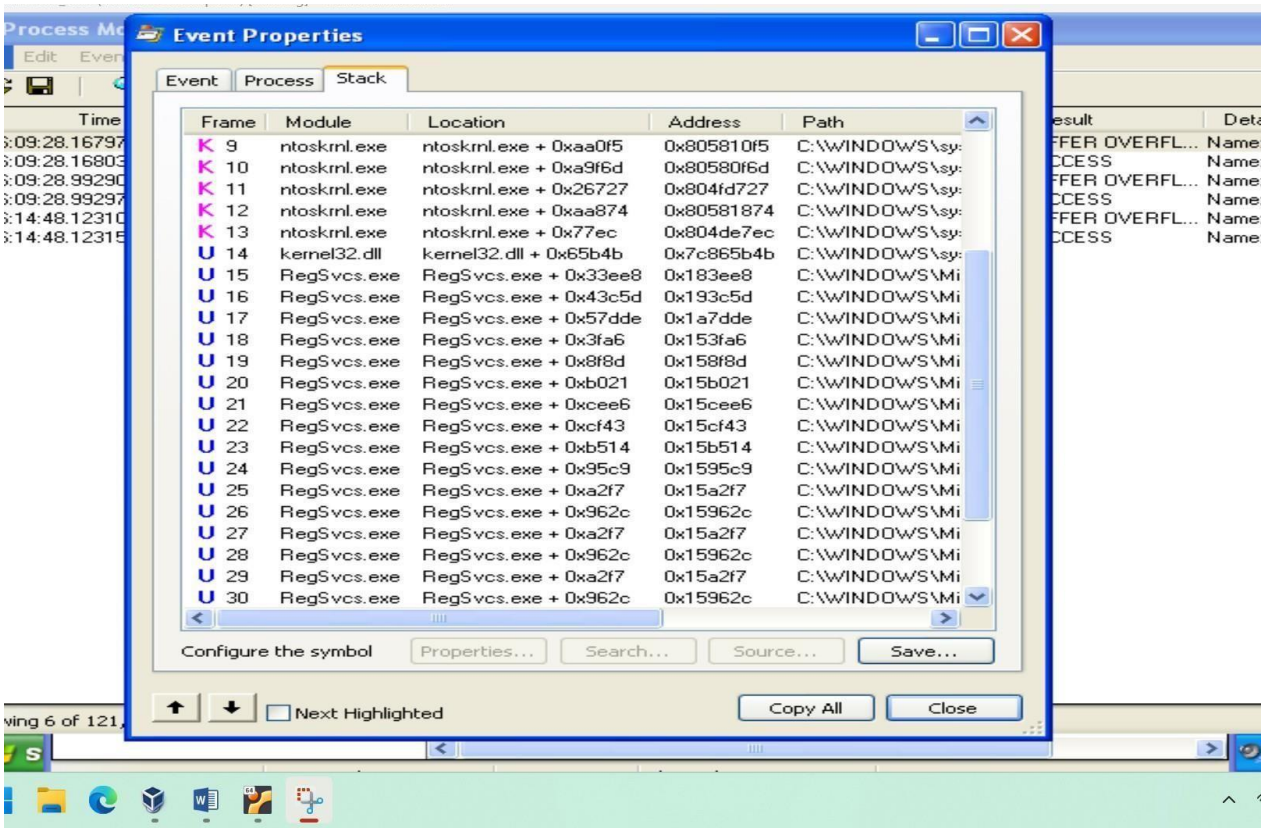




RegSvcs.exe with the PID: 684

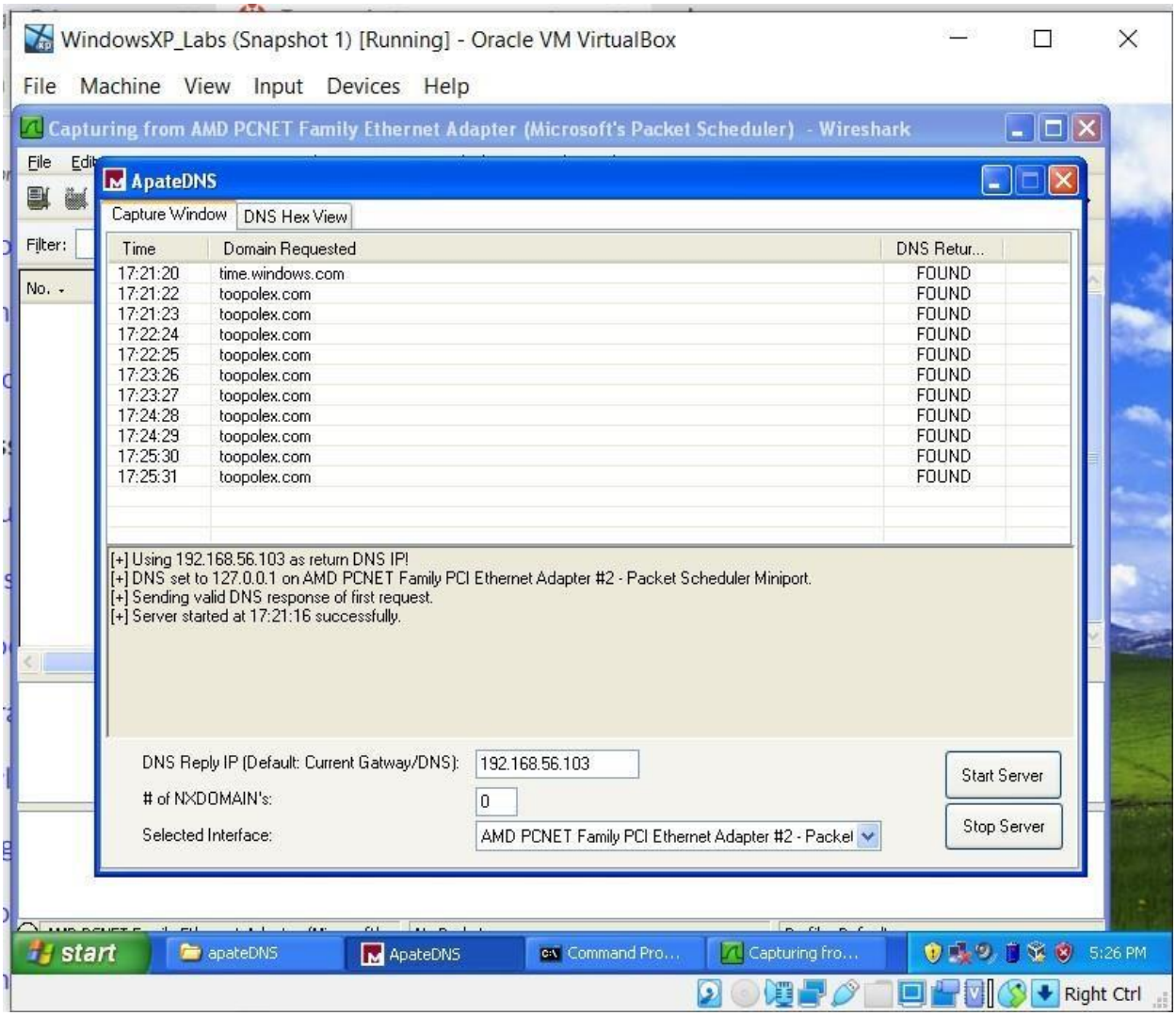

WindowsXP_Labs (Malware run Snapshot) [Running] - Oracle VM VirtualBox
Process Monitor - Sysinternals: www.sysinternals.com
File Edit Event Filter Tools Options Help
Time of Day | Process Name | PID | Operation | Path | Result | Detail
6:09:28.1679759 AM | RegSvcs.exe | 684 | QueryNameInfo... | C:\WINDOWS\system32\verclsid.exe | BUFFER OVERFL... | Name: \W
6:09:28.1680365 AM | RegSvcs.exe | 684 | QueryNameInfo... | C:\WINDOWS\system32\verclsid.exe | SUCCESS | Name: \WINDOW...
6:09:28.9929098 AM | RegSvcs.exe | 684 | QueryNameInfo... | C:\WINDOWS\system32\verclsid.exe | BUFFER OVERFL... | Name: \W
6:09:28.9929704 AM | RegSvcs.exe | 684 | QueryNameInfo... | C:\WINDOWS\system32\verclsid.exe | SUCCESS | Name: \WINDOW...
6:14:48.1231009 AM | RegSvcs.exe | 684 | QueryNameInfo... | C:\WINDOWS\regedit.exe | BUFFER OVERFL... | Name: \W
6:14:48.1231573 AM | RegSvcs.exe | 684 | QueryNameInfo... | C:\WINDOWS\regedit.exe | SUCCESS | Name: \WINDOW...
Showing 6 of 149,954 events (0.0040%)
Backed by virtual memory
6:23 AM


WindowsXP_Labs (Malware run Snapshot) [Running] - Oracle VM VirtualBox
Event Properties
Event Process Stack
Image
Name: RegSvcs.exe
Version:
Path:
C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727\RegSvcs.exe
Command Line:
C:\DOCUME~1\Feng\APPLIC~1\sbe\SVDGY
PID: 684
Parent PID: 640
Session ID: 0
User: FENG-COMPUTER\Feng
Auth ID: 00000000:0000b314
Started: 5/2/2023 6:07:50 AM
Architecture: 32-bit
Virtualized: n/a
Integrity: n/a
Ended: (Running)
Modules:
Module | Address | Size | Path
RegSvcs.exe | 0x150000 | 0xcc000 | C:\WINDOWS\Microsof
uxtheme.dll | 0x5ad70000 | 0x38000 | C:\WINDOWS\system3



Use of ApateDNS tool :

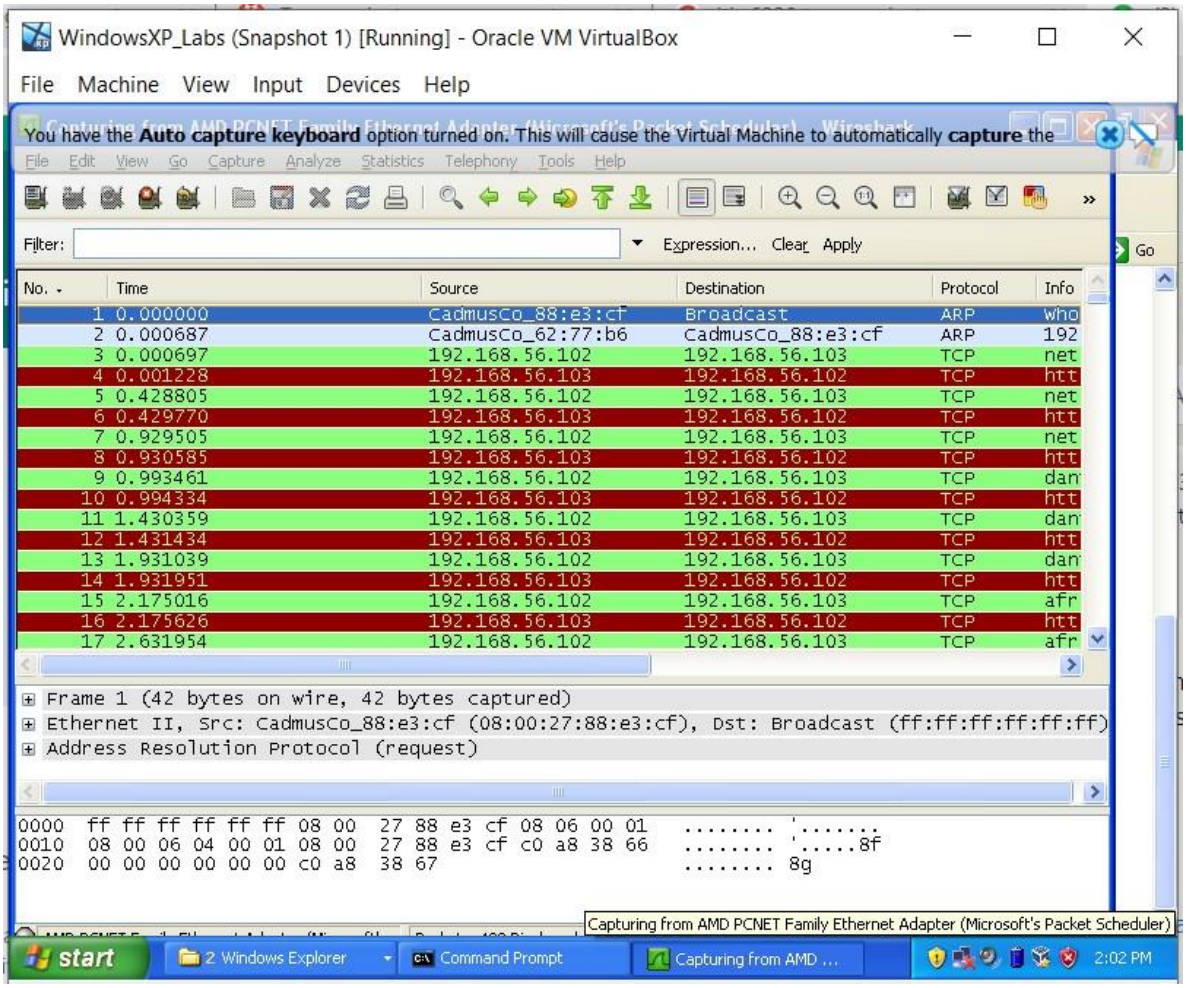
We enter the IP address of Ubuntu VM i.e., 192.168.56.103 and choose Host only Adapter and then start the server after running the malware:



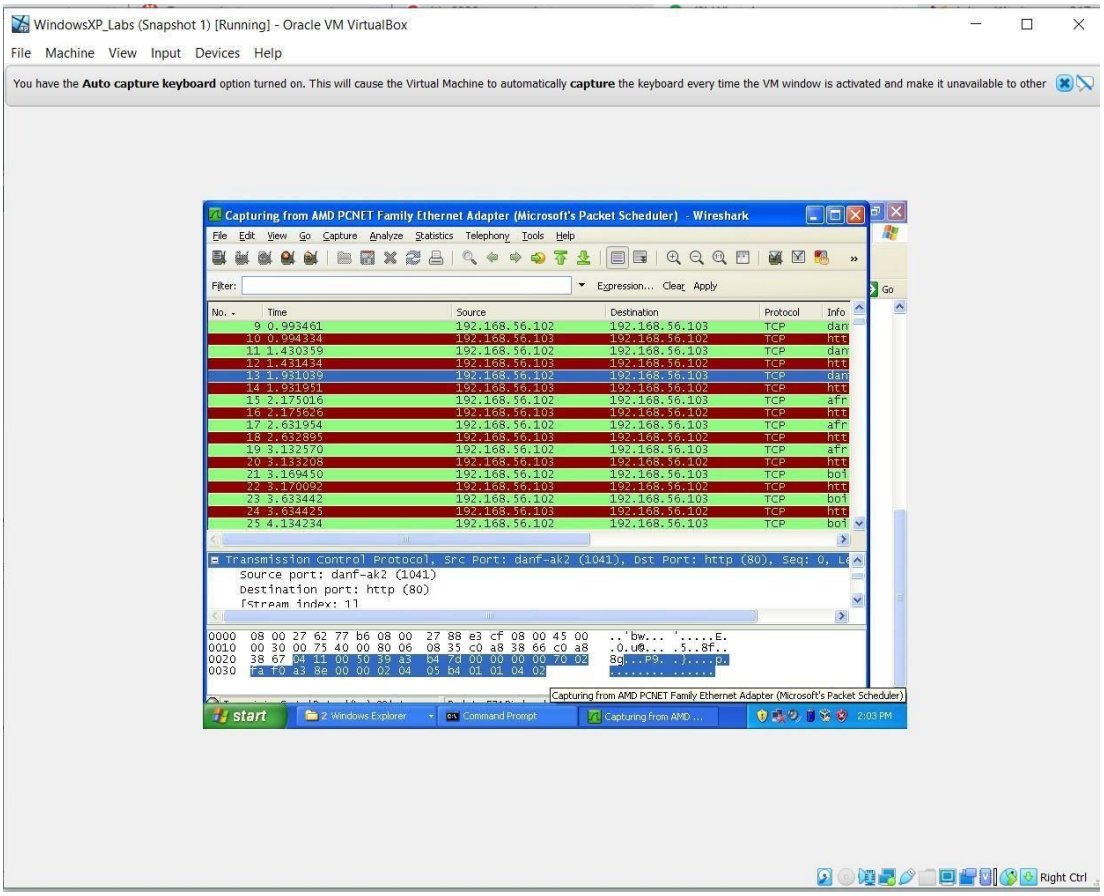
And we can see that malware is trying to reach the website: **toopolex.com**

Use of Wireshark tool :

We use this tool to capture the traffic which malware is trying to reach the internet:



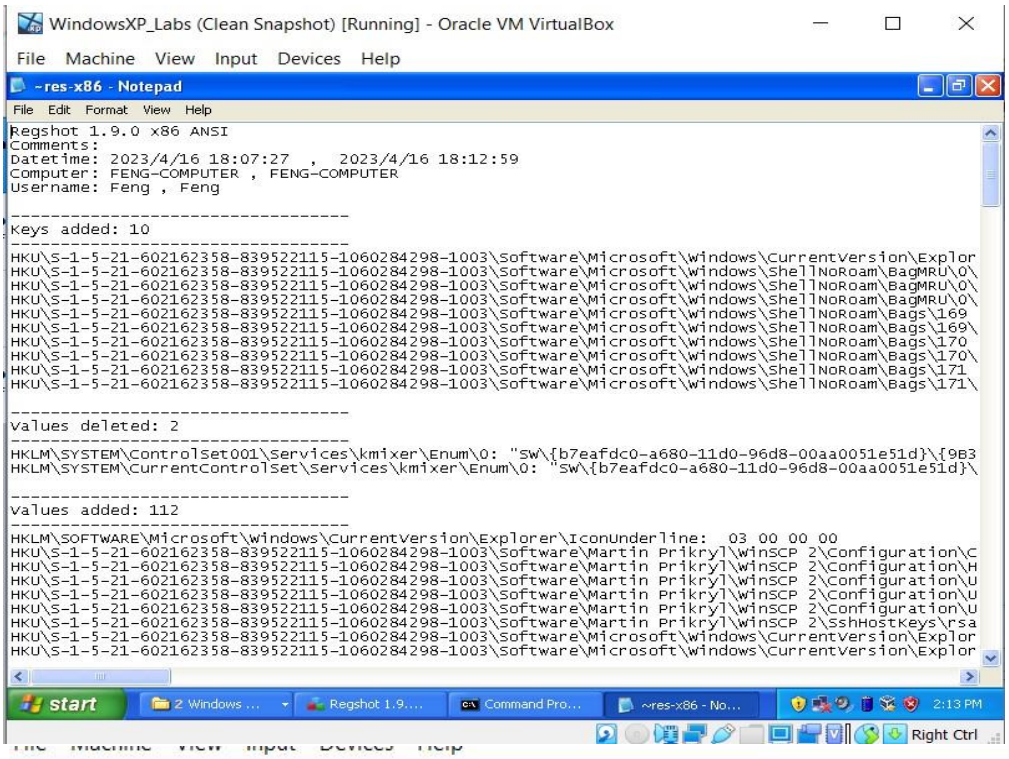
We can see that TCP traffic and the source and destination IP Addresses, this shows that it is trying to reach the internet and communicate through it.

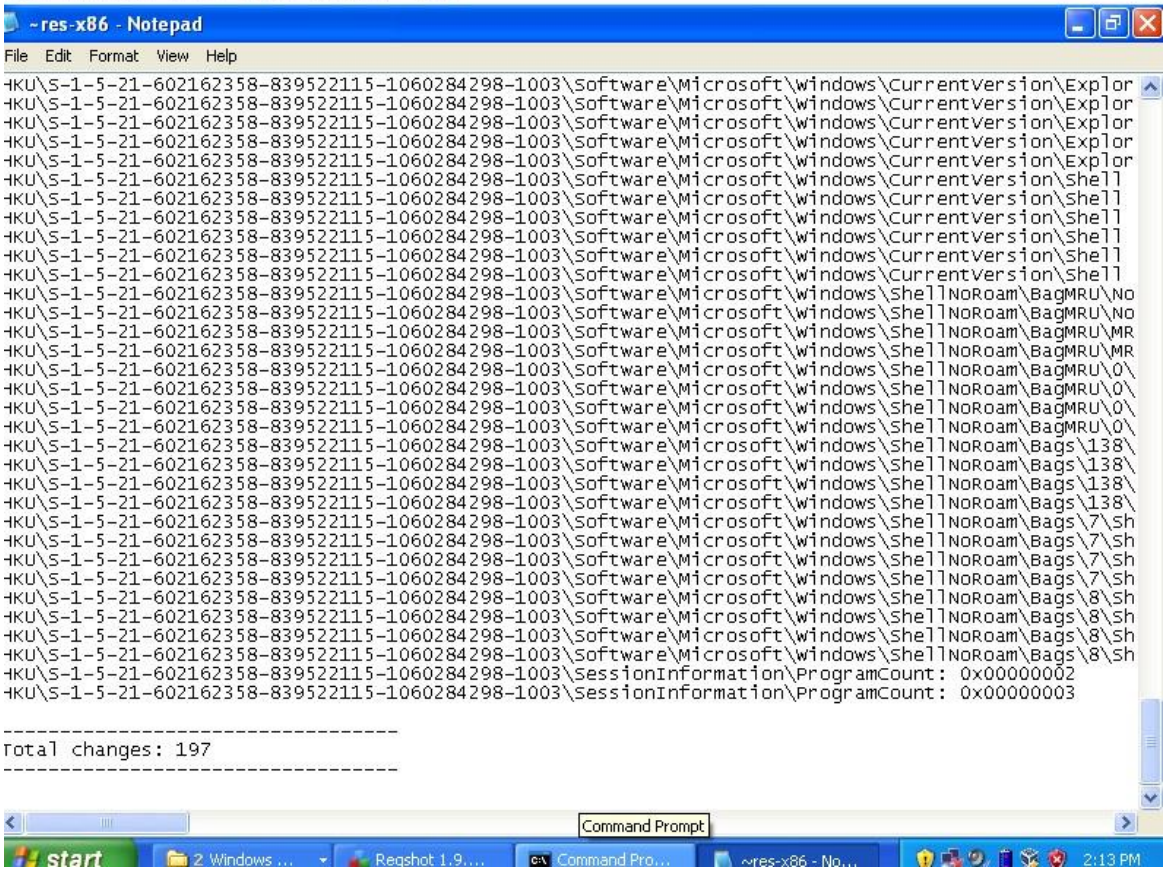


Use of RegShot Tool :

We have performed the RegShot Analysis, in which we take two shots one before running malware and the other after running. We see the total changes made to the Registry files.

Total Changes : 197


WindowsXP_Labs (Clean Snapshot) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
~res-x86 - Notepad
File Edit Format View Help
Regshot 1.9.0 x86 ANSI
Comments:
Datetime: 2023/4/16 18:07:27 , 2023/4/16 18:12:59
Computer: FENG-COMPUTER , FENG-COMPUTER
Username: Feng , Feng
Keys added: 10
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\CurrentVersion\Explor
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\BagMRU\0\
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\BagMRU\0\
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\BagMRU\0\
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\169
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\169\
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\170
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\170\
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\171
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\171\
Values deleted: 2
HKLM\SYSTEM\ControlSet001\Services\kmixer\Enum\0: "SW\{b7eafdc0-a680-11d0-96d8-00aa0051e51d}\{9B3
HKLM\SYSTEM\CurrentControlSet\Services\kmixer\Enum\0: "SW\{b7eafdc0-a680-11d0-96d8-00aa0051e51d}\
Values added: 112
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\IconUnderline: 03 00 00 00
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Martin Prikryl\WinSCP 2\Configuration\C
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Martin Prikryl\WinSCP 2\Configuration\H
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Martin Prikryl\WinSCP 2\Configuration\U
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Martin Prikryl\WinSCP 2\Configuration\U
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Martin Prikryl\WinSCP 2\Configuration\U
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Martin Prikryl\WinSCP 2\SshHostKeys\rsa
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\CurrentVersion\Explor
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\CurrentVersion\Explor
start
2 Windows ...
Regshot 1.9....
Command Pro...
~res-x86 - No...
2:13 PM
Right Ctrl


~res-x86 - Notepad
File Edit Format View Help
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\CurrentVersion\Explor
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\CurrentVersion\Explor
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\CurrentVersion\Explor
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\CurrentVersion\Explor
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\CurrentVersion\Explor
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\CurrentVersion\Shell
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\CurrentVersion\Shell
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\CurrentVersion\Shell
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\CurrentVersion\Shell
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\CurrentVersion\Shell
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\BagMRU\No
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\BagMRU\No
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\BagMRU\MR
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\BagMRU\MR
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\BagMRU\0\
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\BagMRU\0\
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\BagMRU\0\
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\BagMRU\0\
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\138\
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\138\
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\138\
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\138\
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\7\Sh
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\7\Sh
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\7\Sh
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\7\Sh
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\8\Sh
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\8\Sh
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\8\Sh
HKU\S-1-5-21-602162358-839522115-1060284298-1003\Software\Microsoft\Windows\ShellNoRoam\Bags\8\Sh
HKU\S-1-5-21-602162358-839522115-1060284298-1003\SessionInformation\ProgramCount: 0x00000002
HKU\S-1-5-21-602162358-839522115-1060284298-1003\SessionInformation\ProgramCount: 0x00000003
Total changes: 197
Command Prompt
start
2 Windows ...
Regshot 1.9....
Command Pro...
~res-x86 - No...
2:13 PM

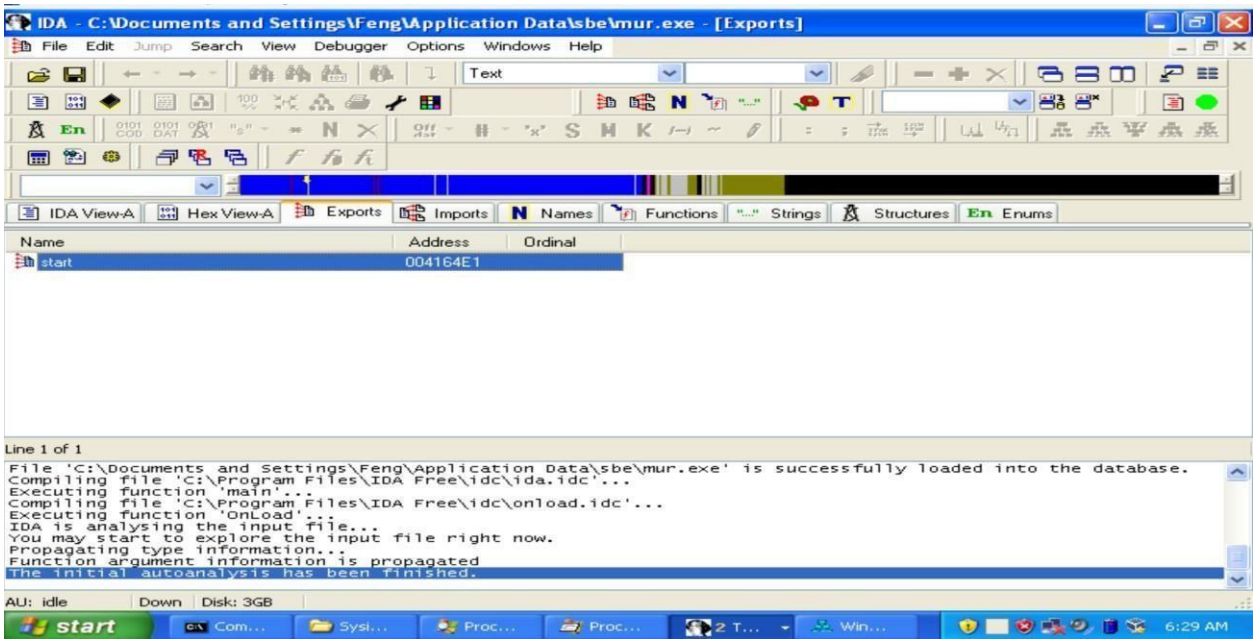
Now we have to perform advanced analysis of both the behavior analysis in static as well as dynamic analysis.

3.5 Advanced Static Analysis on malware behavior :

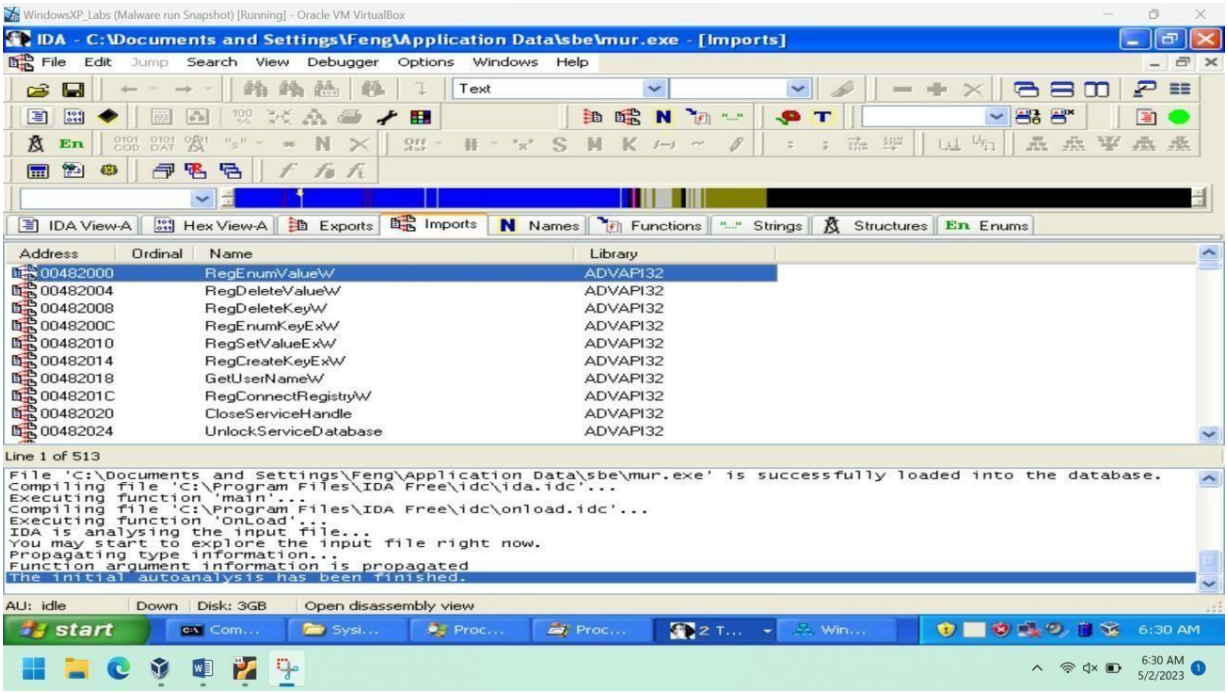
In this section, we use some advanced tools for the static analysis such as IDA Pro and Ghidra software tools. These tools are used for disassembling the malware either in assembly language or C language or both.

Use of IDA Pro tool :

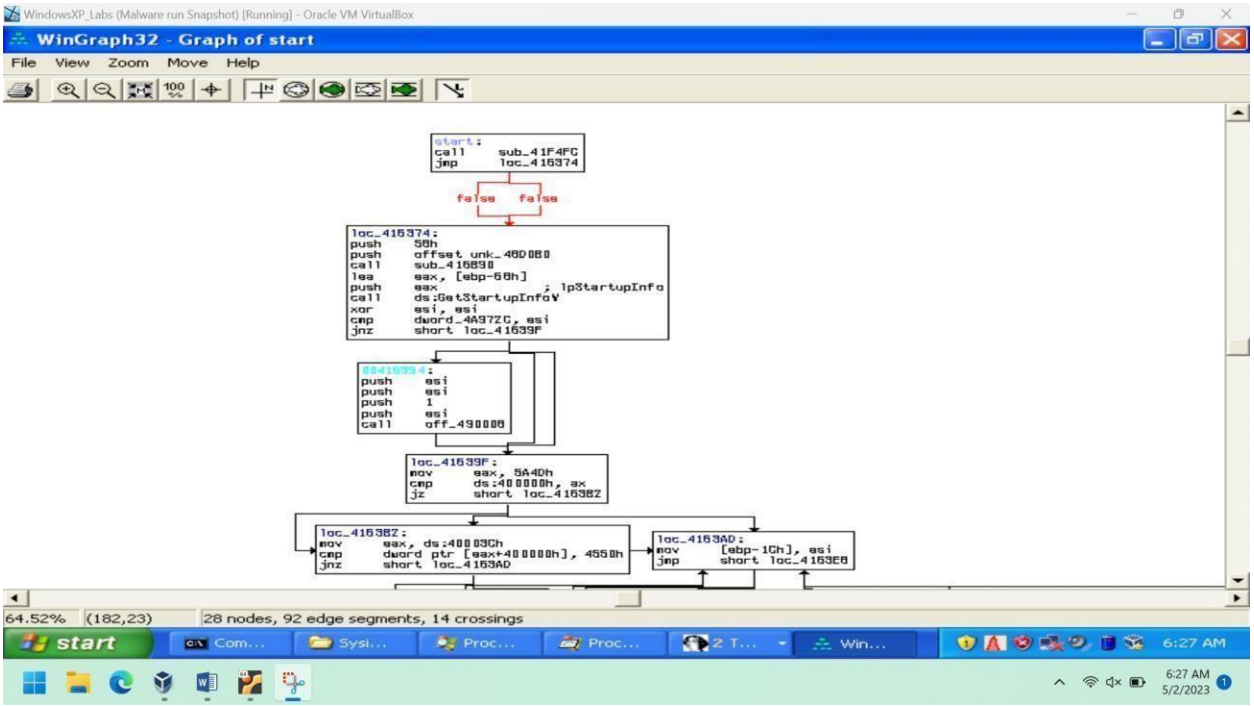
Now we see list of Exports :

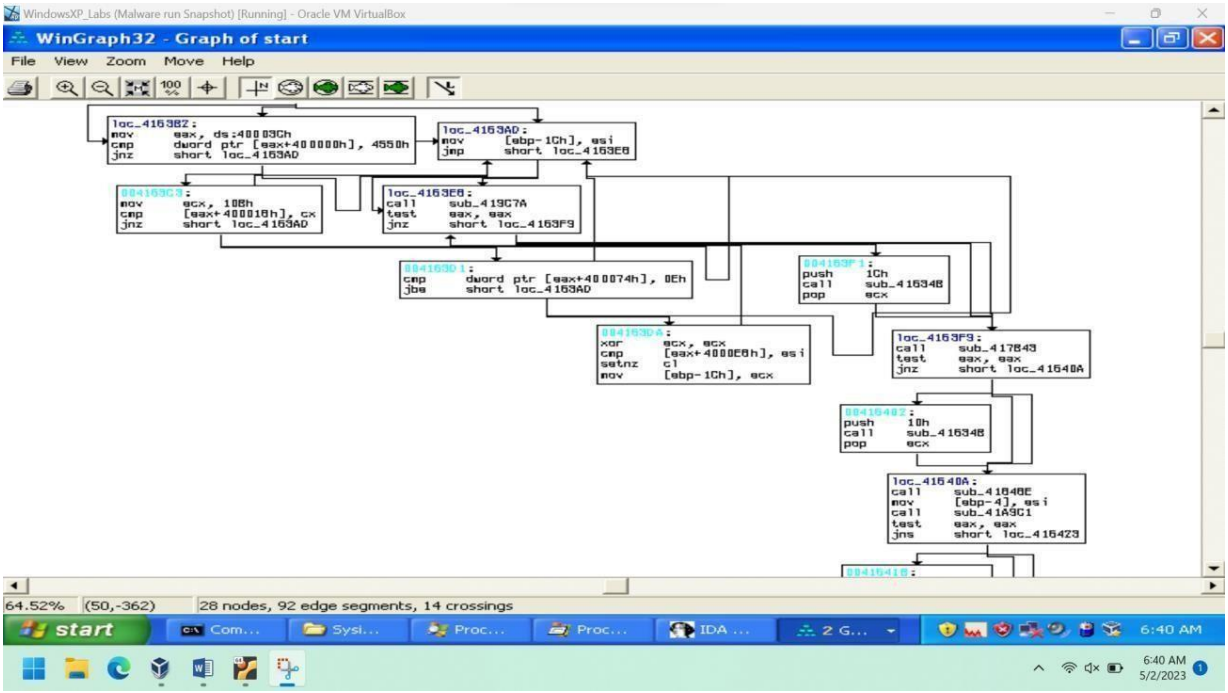


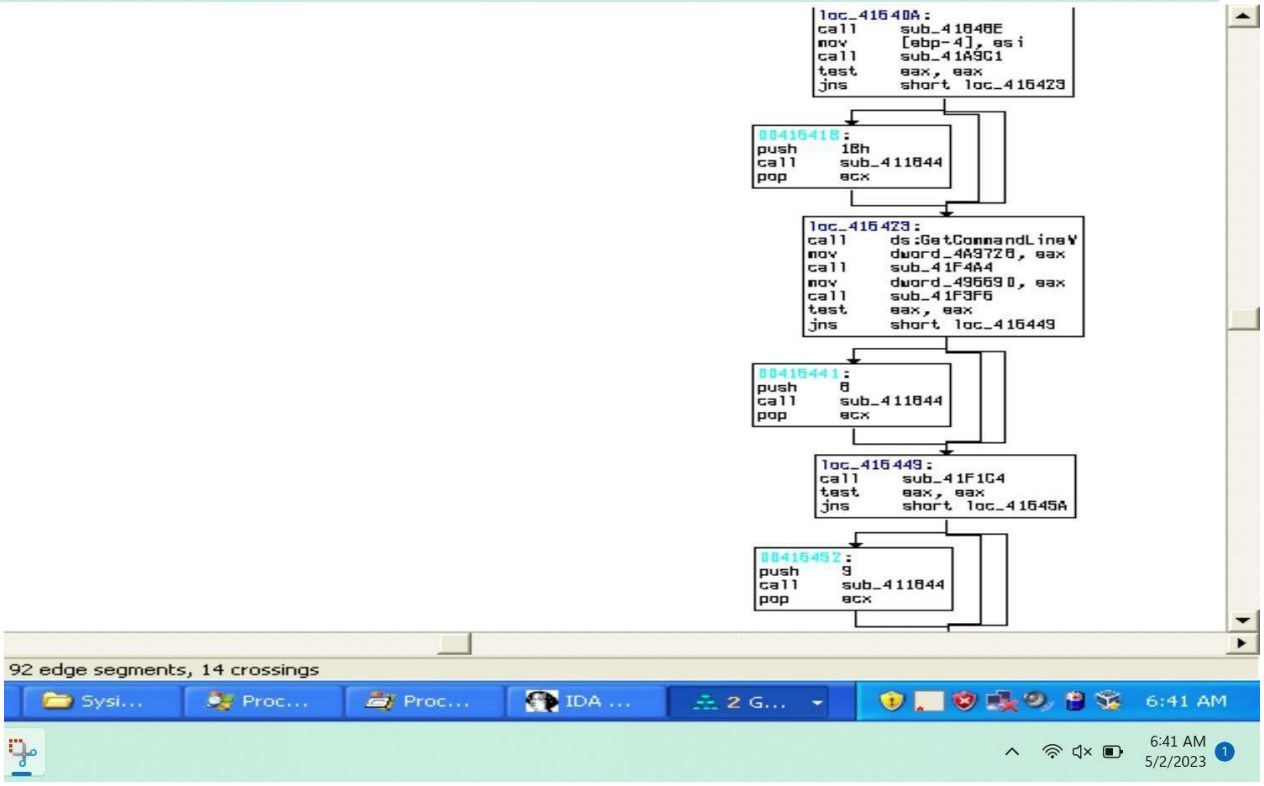
Now the list of imports does the malware in assessing it's behavior :

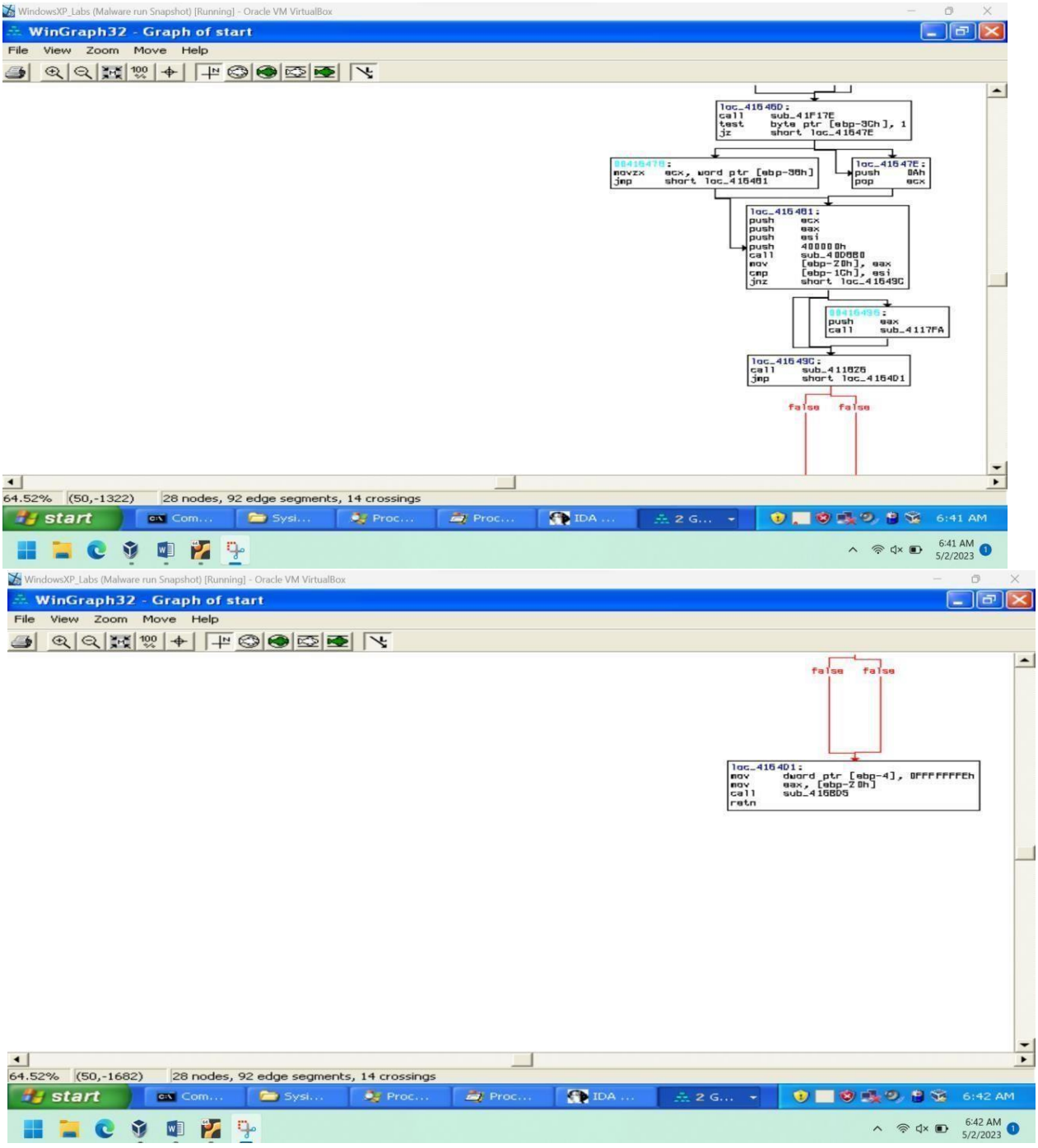


Taking WinGraph32 – Graph of start

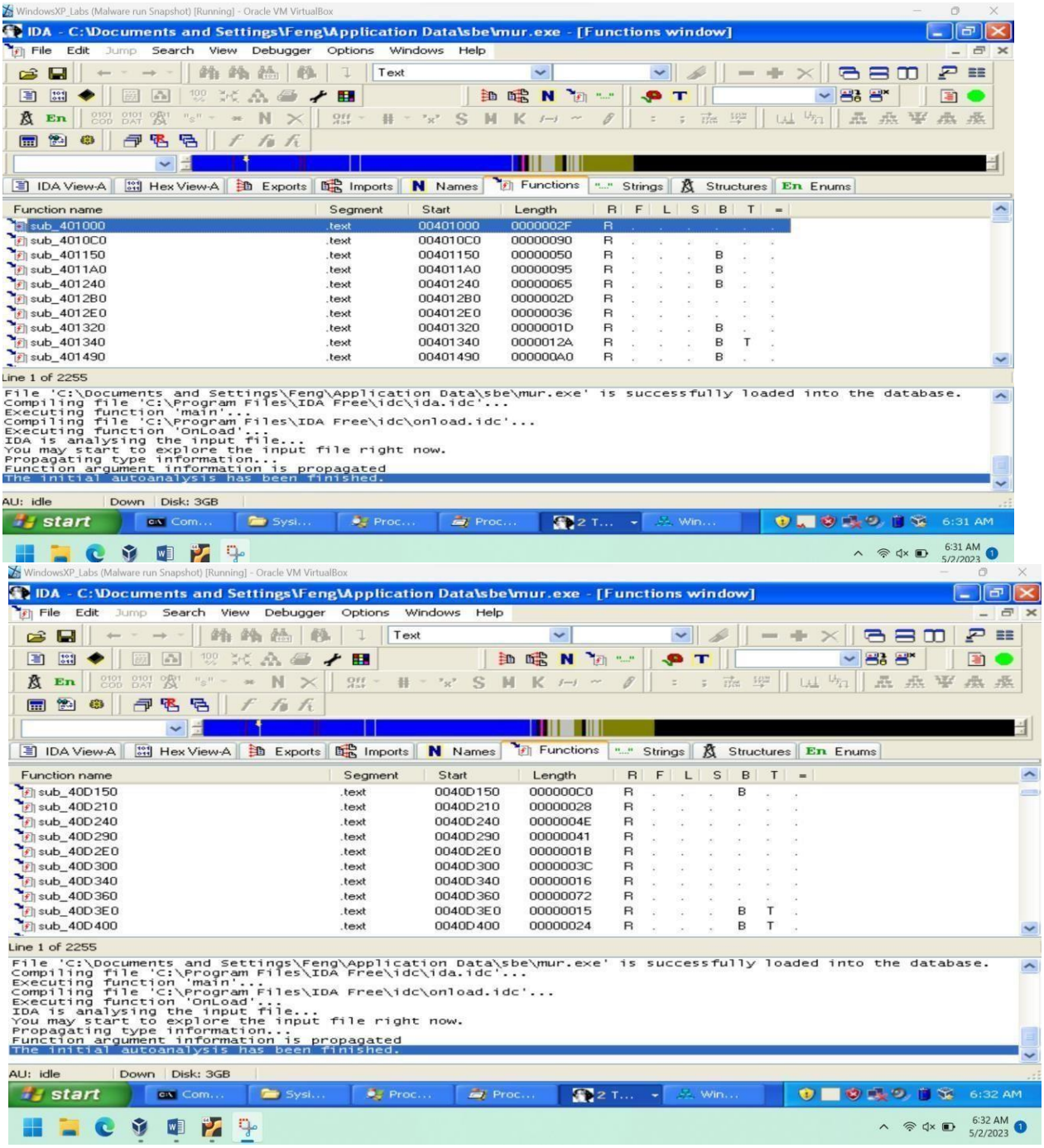



WindowsXP_Labs (Malware run Snapshot) [Running] - Oracle VM VirtualBox
WinGraph32 - Graph of start
File View Zoom Move Help
loc_4163B2:
mov eax, ds:400B3Ch
cmp dword ptr [eax+400000h], 4550h
jnz short loc_4163AD
loc_4163AD:
mov [ebp-1Ch], esi
jmp short loc_4163E8
004163C9:
mov ecx, 10Bh
cmp [eax+400018h], cx
jnz short loc_4163AD
loc_4163E8:
call sub_419C7A
test eax, eax
jnz short loc_4163F9
004163D1:
cmp dword ptr [eax+400074h], 0Eh
jbe short loc_4163AD
004163F1:
push 1Ch
call sub_416348
pop ecx
004163DA:
xor ecx, ecx
cmp [eax+4000E8h], esi
setnz cl
mov [ebp-1Ch], ecx
loc_4163F9:
call sub_417B43
test eax, eax
jnz short loc_41640A
00416402:
push 10h
call sub_416348
pop ecx
loc_41640A:
call sub_41640E
mov [ebp-4], esi
call sub_41A9C1
test eax, eax
jns short loc_416423
64.52% (50,-362) 28 nodes, 92 edge segments, 14 crossings
start
Com...
Sysi...
Proc...
Proc...
IDA ...
2 G...
6:40 AM
6:40 AM
5/2/2023

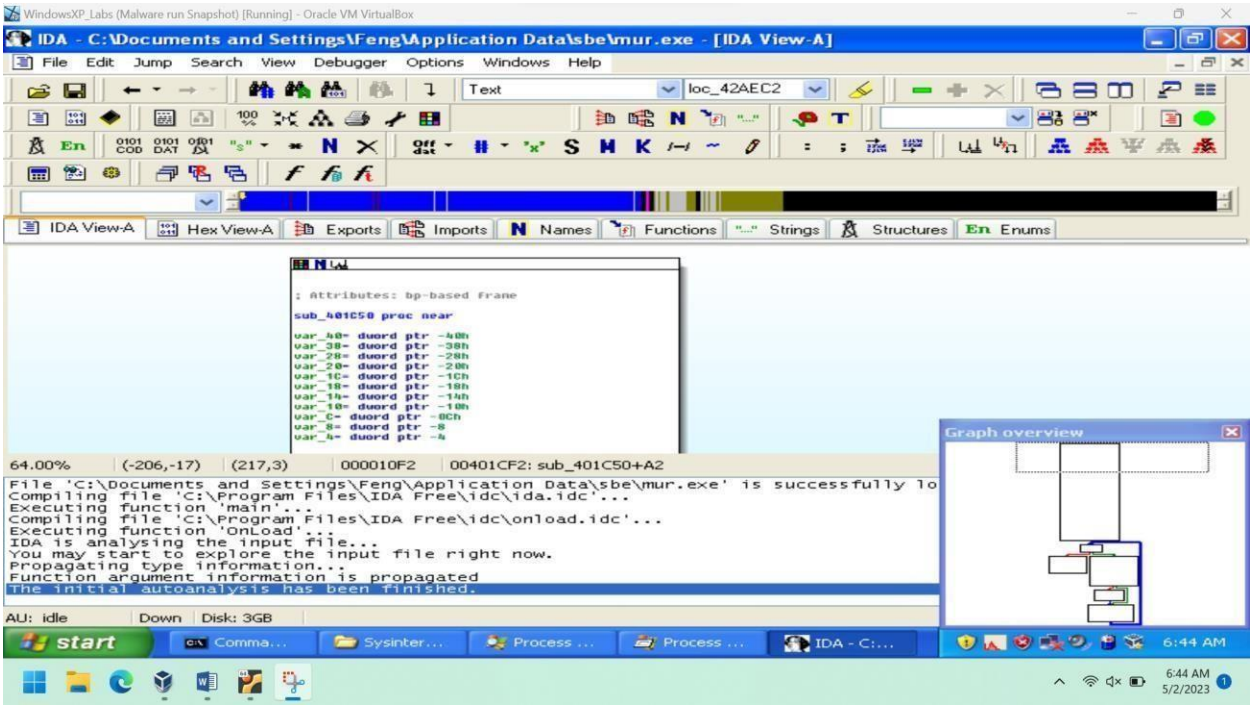

loc_41640A:
call sub_41640E
mov [ebp-4], esi
call sub_41A9C1
test eax, eax
jns short loc_416423
00416418:
push 18h
call sub_411844
pop ecx
loc_416423:
call ds:GetCommandLineW
mov dword_4A97Z8, eax
call sub_41F4A4
mov dword_496690, eax
call sub_41F3F6
test eax, eax
jns short loc_416449
00416441:
push 8
call sub_411844
pop ecx
loc_416449:
call sub_41F1C4
test eax, eax
jns short loc_41645A
00416452:
push 9
call sub_411844
pop ecx
92 edge segments, 14 crossings
Sysi...
Proc...
Proc...
IDA ...
2 G...
6:41 AM
6:41 AM
5/2/2023



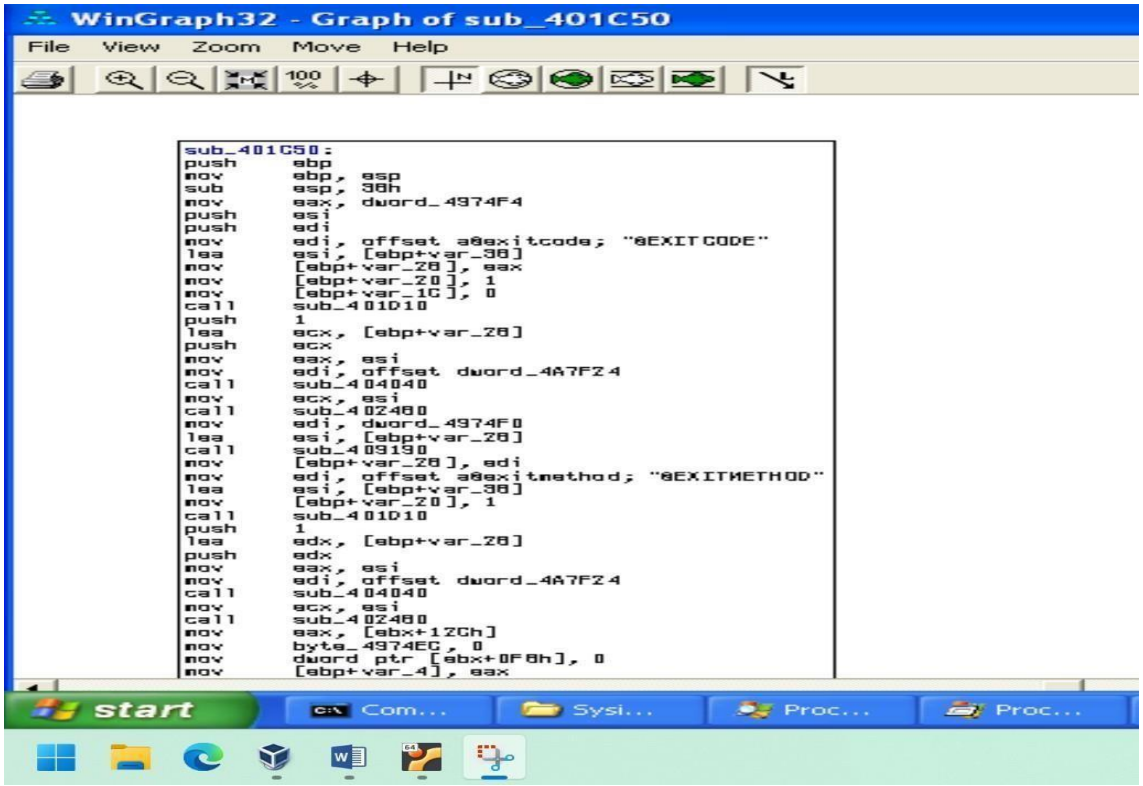
The functions associated in the analysis of mur.exe :

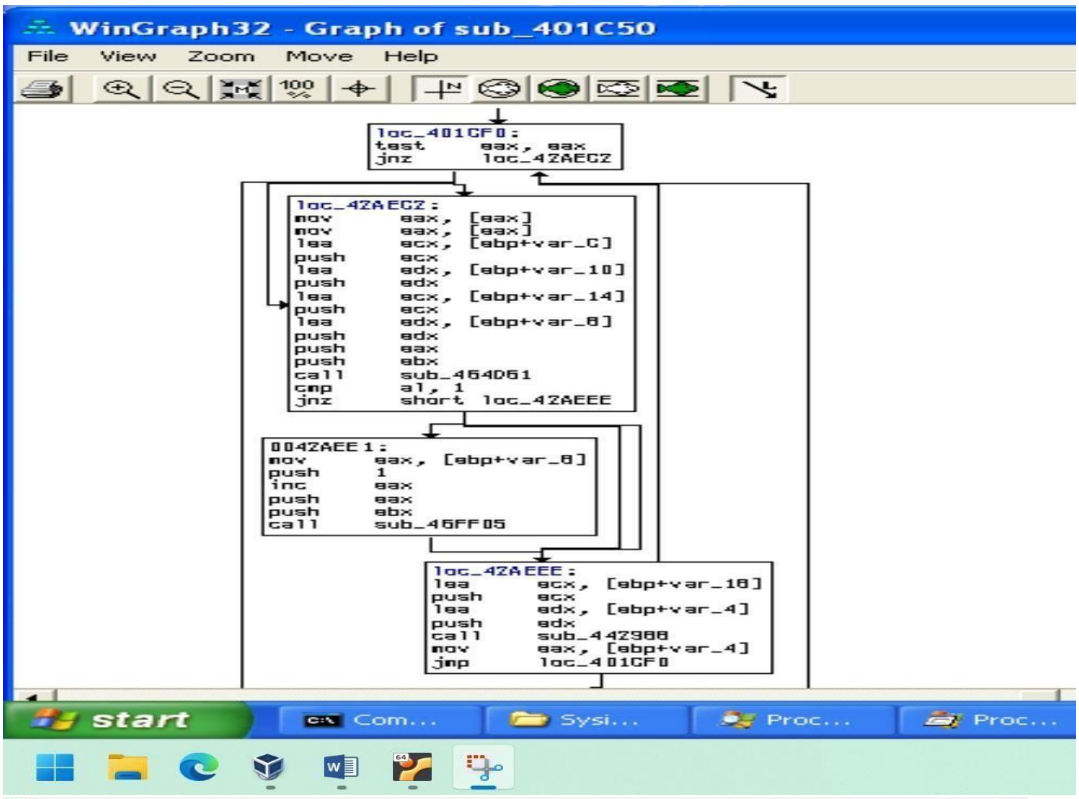


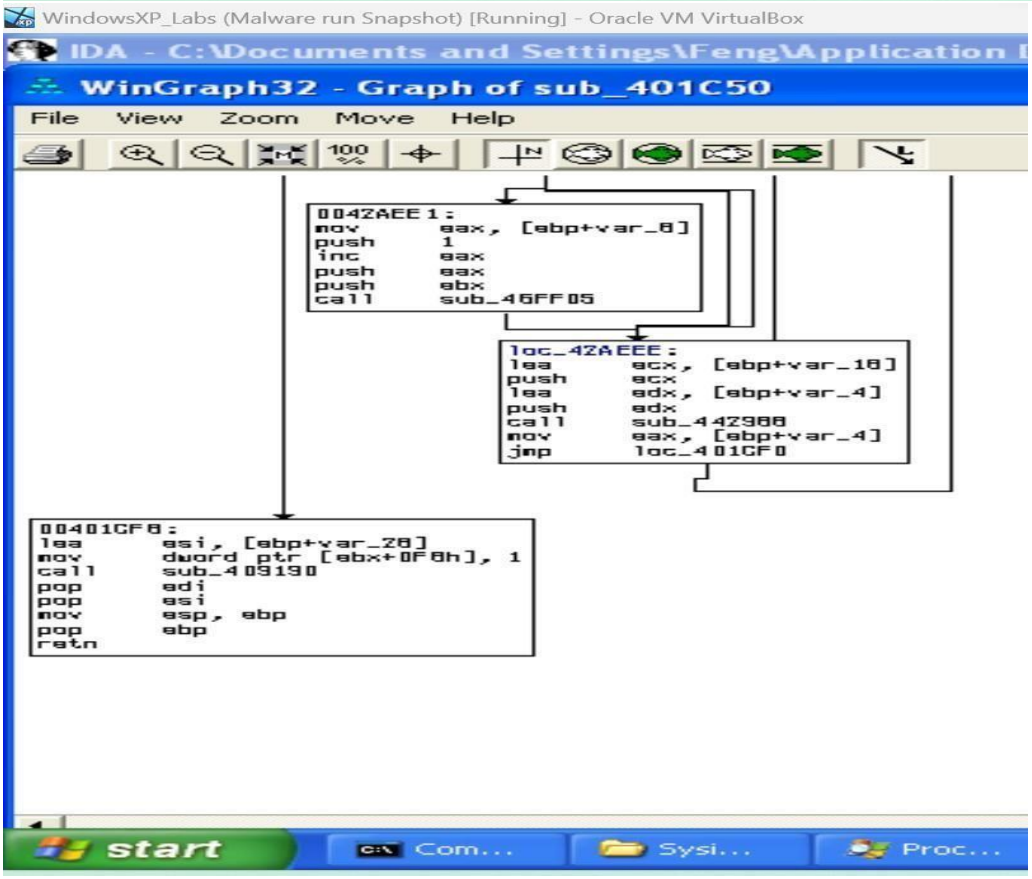
One of the function sub_401C50 :



And it's WinGraph32 :



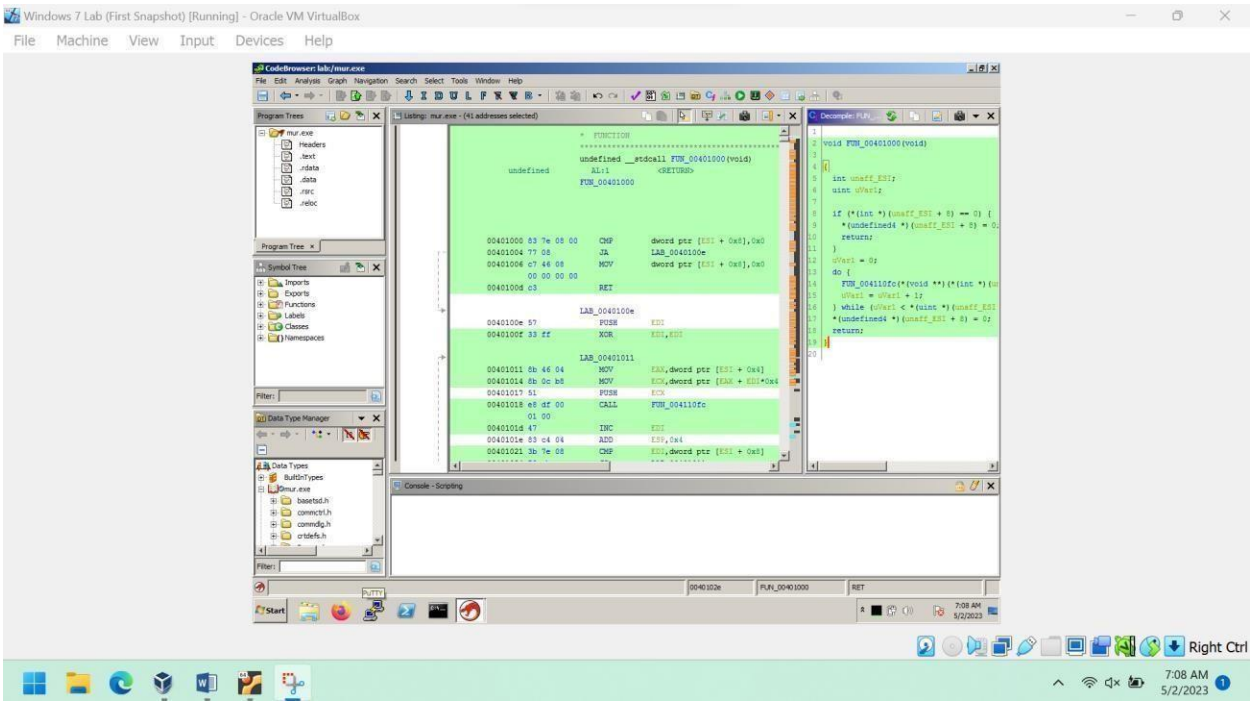

WinGraph32 - Graph of sub_401C50
File View Zoom Move Help
loc_401CF0:
test eax, eax
jnz loc_42AEC2
loc_42AEC2:
mov eax, [eax]
mov eax, [eax]
lea ecx, [ebp+var_C]
push ecx
lea edx, [ebp+var_10]
push edx
lea ecx, [ebp+var_14]
push ecx
lea edx, [ebp+var_8]
push edx
push eax
push ebx
call sub_464D61
cmp al, 1
jnz short loc_42AEEE
0042AEE1:
mov eax, [ebp+var_8]
push 1
inc eax
push eax
push ebx
call sub_46FF05
loc_42AEEE:
lea ecx, [ebp+var_10]
push ecx
lea edx, [ebp+var_4]
push edx
call sub_442980
mov eax, [ebp+var_4]
jmp loc_401CF0
start
Com...
Sysi...
Proc...
Proc...


WindowsXP_Labs (Malware run Snapshot) [Running] - Oracle VM VirtualBox
IDA - C:\Documents and Settings\Feng\Application
WinGraph32 - Graph of sub_401C50
File View Zoom Move Help
0042AEE1:
mov eax, [ebp+var_8]
push 1
inc eax
push eax
push ebx
call sub_46FF05
loc_42AEEE:
lea ecx, [ebp+var_10]
push ecx
lea edx, [ebp+var_4]
push edx
call sub_442980
mov eax, [ebp+var_4]
jmp loc_401CF0
00401CF0:
lea esi, [ebp+var_20]
mov dword ptr [ebx+0F8h], 1
call sub_409190
pop edi
pop esi
mov esp, ebp
pop ebp
retn
start
Com...
Sysi...
Proc...

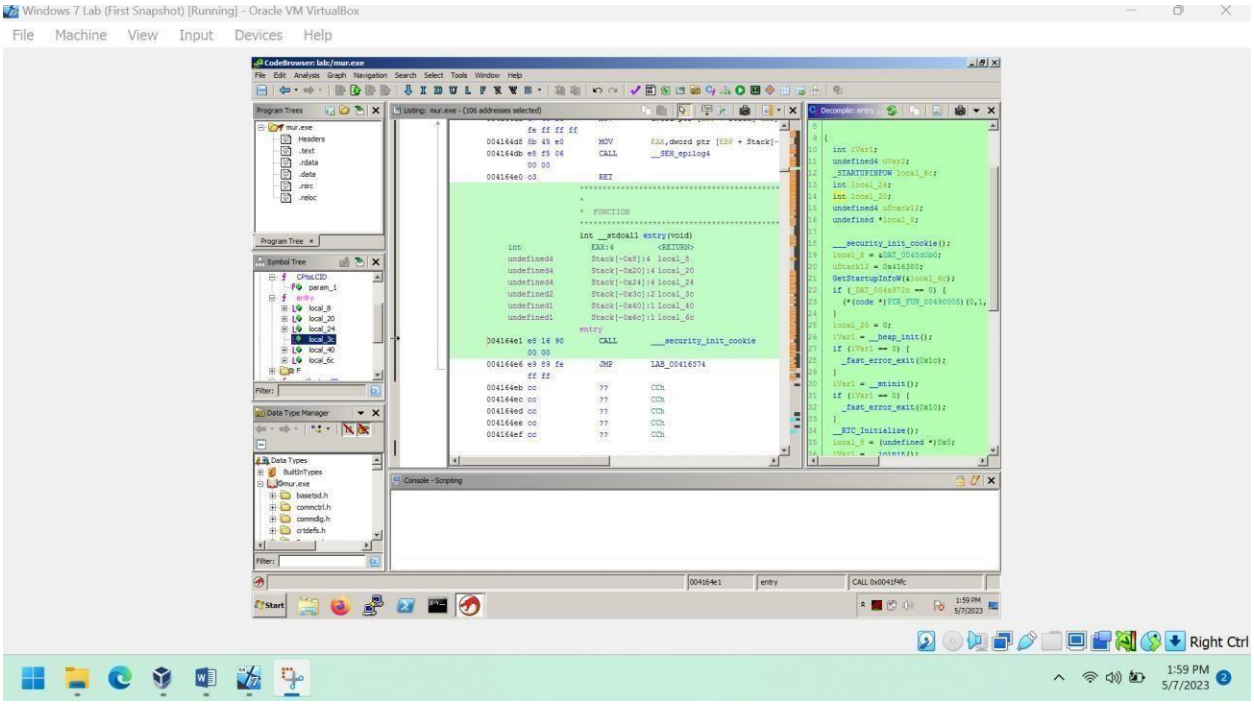
These are the findings I have found so far with the help of IDA Pro tool and now we see the results using Ghidra tool.

Use of Ghidra Software tool :

This tool shows the disassembly in both the assembly language and C language in the CodeViewer browser. As you can see in the following figure it is clearly evident that we can somehow understand in C language rather than Assembly language which is the best feature of Ghidra tool.



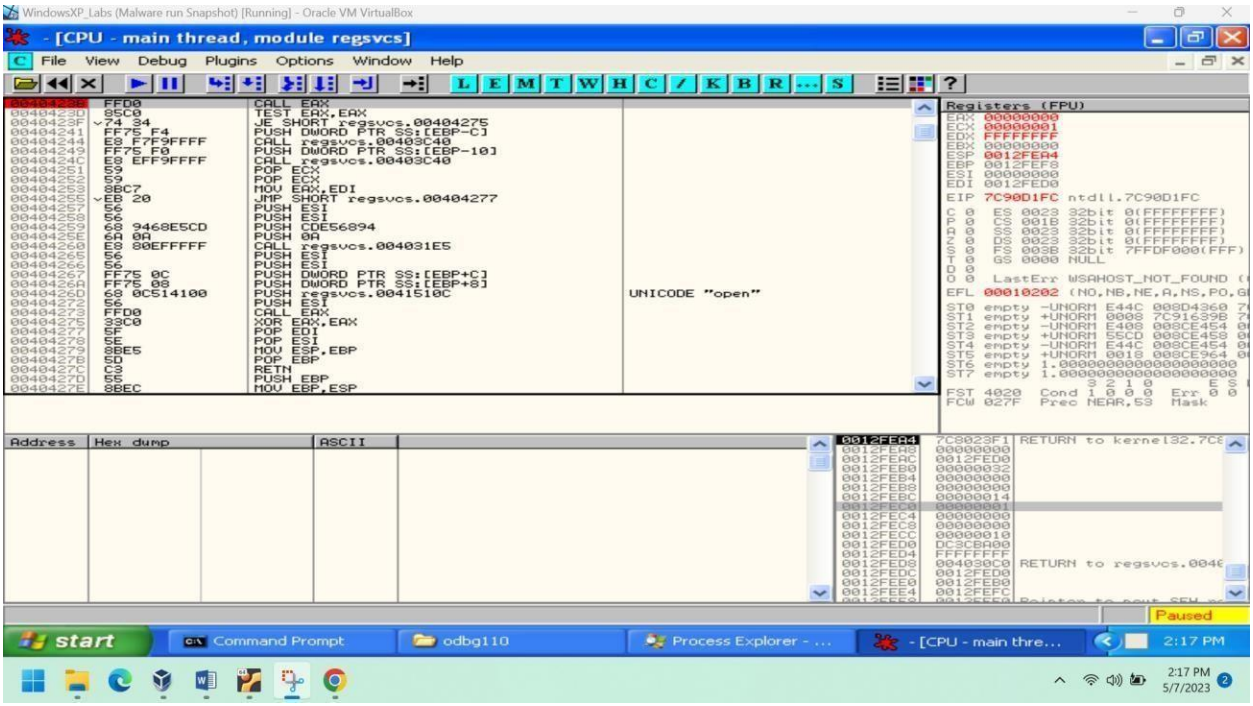
In this you can see C language constructs on the right side in the decompiler section.

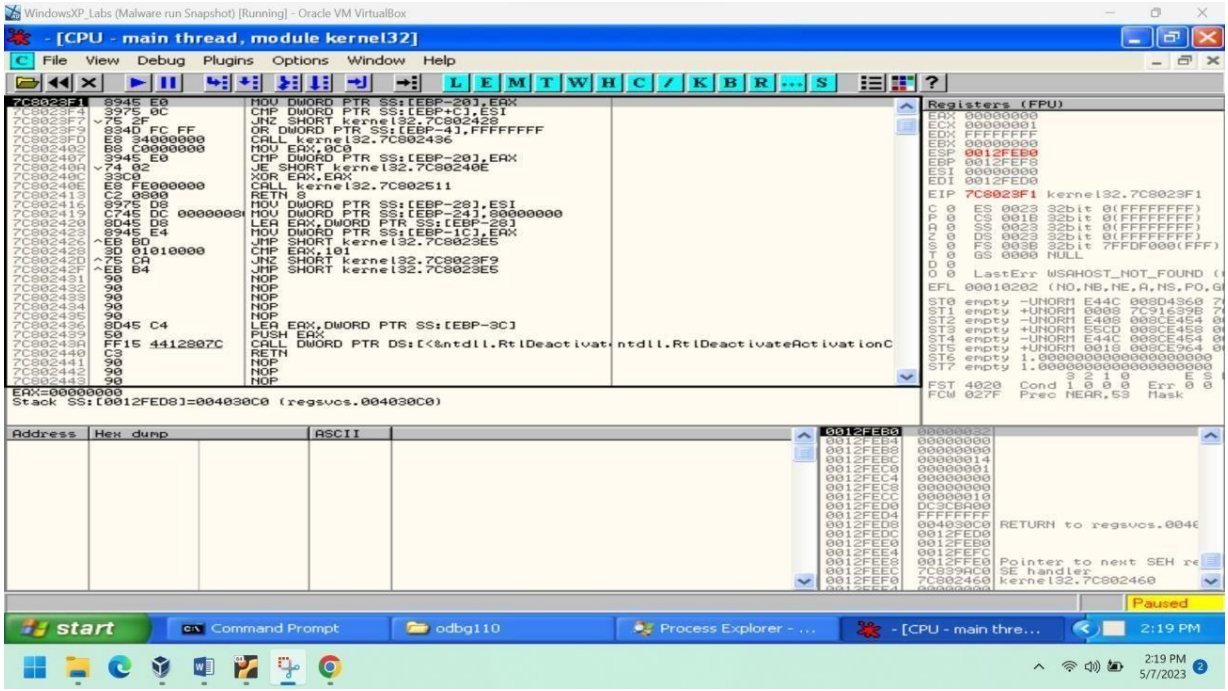


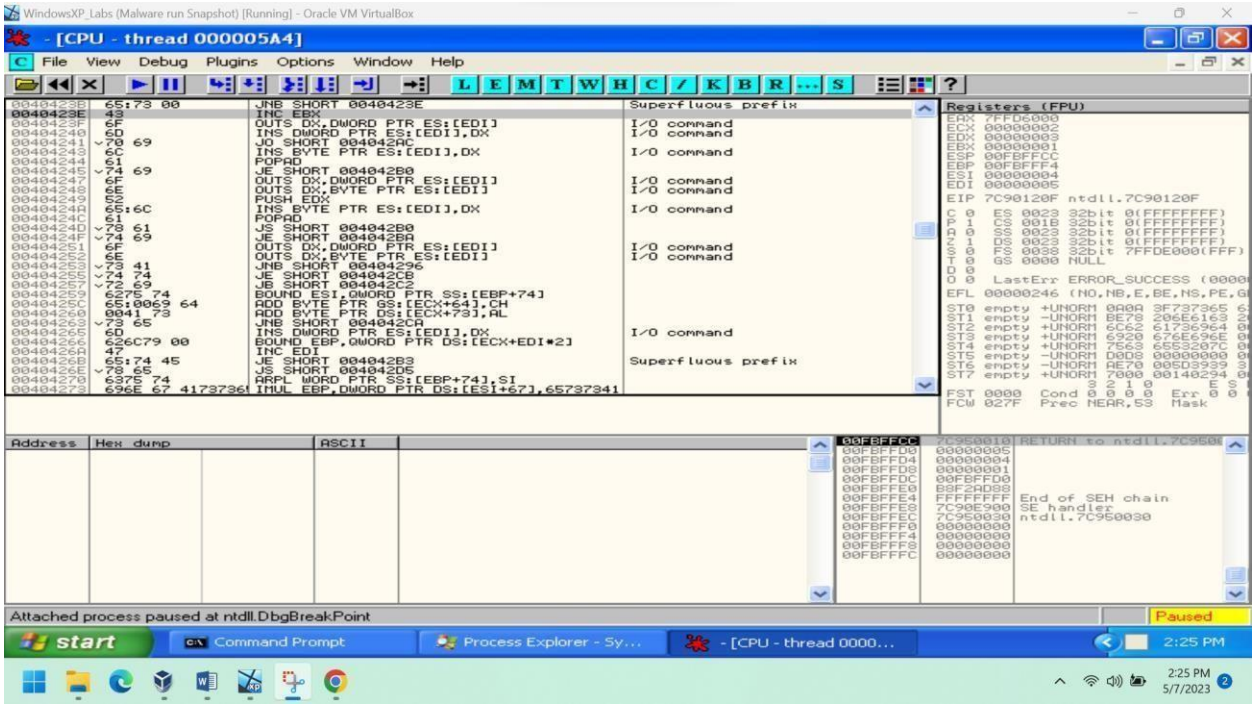
This is another function where we can see that it is working with some cookies related and also we can see some manipulations. There are few other important functions which we can see on the left side in the Functions under symbol tree.

Now we see the analysis performed using OllyDbg 1.10 tool as following as below

Use of Ollydbg 1.10 analysis :




WindowsXP_Labs (Malware run Snapshot) [Running] - Oracle VM VirtualBox
- [CPU - main thread, module kernel32]
File View Debug Plugins Options Window Help
Registers (FPU)
Address Hex dump ASCII
Paused
start
Command Prompt
odbg110
Process Explorer - ...
- [CPU - main thre...
2:19 PM
5/7/2023


WindowsXP_Labs (Malware run Snapshot) [Running] - Oracle VM VirtualBox
- [CPU - thread 000005A4]
File View Debug Plugins Options Window Help
Registers (FPU)
Address Hex dump ASCII
Attached process paused at ntdll.DbgBreakPoint
Paused
start
Command Prompt
Process Explorer - Sy...
- [CPU - thread 0000...
2:25 PM
5/7/2023

4**. Key Findings**

The malware exhibited several advanced techniques, including:

- **Anti-debugging and Evasion**: The malware employed sandbox detection and virtualization evasion tactics to prevent analysis in certain environments.
- **Process Spawning**: Multiple processes were spawned to obscure the primary executable's activities.
- **Network Communication**: The malware attempted to establish communication with an external domain, likely for command and control (C2) purposes.
- **Registry Manipulation**: Significant changes were made to the system registry, indicating persistent behavior on infected systems.
- **Data Exfiltration**: The use of TCP traffic suggested an attempt to exfiltrate sensitive data from the infected system.

5. Challenges

One of the key challenges encountered was dealing with the malware's multi-stage operation. The malware employed sophisticated techniques to hide its true functionality, requiring extensive use of reverse-engineering tools to fully understand its behaviors. Identifying the malware's evasion techniques and anti-debugging strategies required multiple iterations of testing and analysis.

6.Conclusion and Learning

This project provided hands-on experience with real-world malware analysis, greatly enhancing the understanding of malware behaviors and reverse engineering. The application of both static and dynamic analysis techniques allowed for a thorough investigation of the malware’s lifecycle. The findings underscore the importance of staying ahead of malware trends and the evolving sophistication of malicious software.

The experience also demonstrated the necessity of meticulous report writing, emphasizing the importance of clearly communicating complex technical findings. As cybersecurity threats continue to evolve, mastering these analysis techniques will be essential for future security professionals.