Malware Analysis

Malware is any software that does something that causes harm to a user, computer, or network. Malware includes viruses, Trojan horses, worms, rootkits, scareware, ransomware and spyware.
Malware analysis is the process of determining the functionality, origin and potential impact of a given malware sample.
Malware analysis is classified into two types – static and dynamic. Static techniques involve analysis of code while dynamic techniques analyze the behavior of a malware.

<u>The project</u>
My project discusses and presents the importance of the process of malware analysis in the field of cybersecurity. As a security incident occurs as a result of a malware, malware analysis is a crucial step during the incident response phase as in order to take the necessary actions for recovery, understanding of the malware is the starting point.

The steps-

The Environment ---2-7
1- Introduction --2
2- Configuring The Network -------------------------------------3-5
3- Preparing The Machine For Analyze--------------------------5-6
4- Taking Snapshots--6-7

The Malware--8-12
Static Analysis---12-23
Dynamic Analysis--24-36
Summary---37
Source Of Information--38

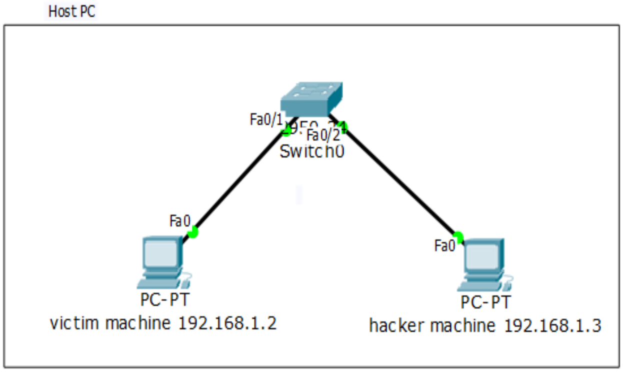
The environment

1) Introduction

My environment is bases on virtual machines - using VMware.
A closed and safe environment based on virtual machines will allow me to securely investigate the malware without putting the other machines that are connected to the same network at risk.

My environment consists of 2 different virtual machines:

1- Win 10 - the machine that is using for attacking
2 - Win 10 - the "victim's" computer, the machine that is going to be infected with the malware for the sake of the analysis.



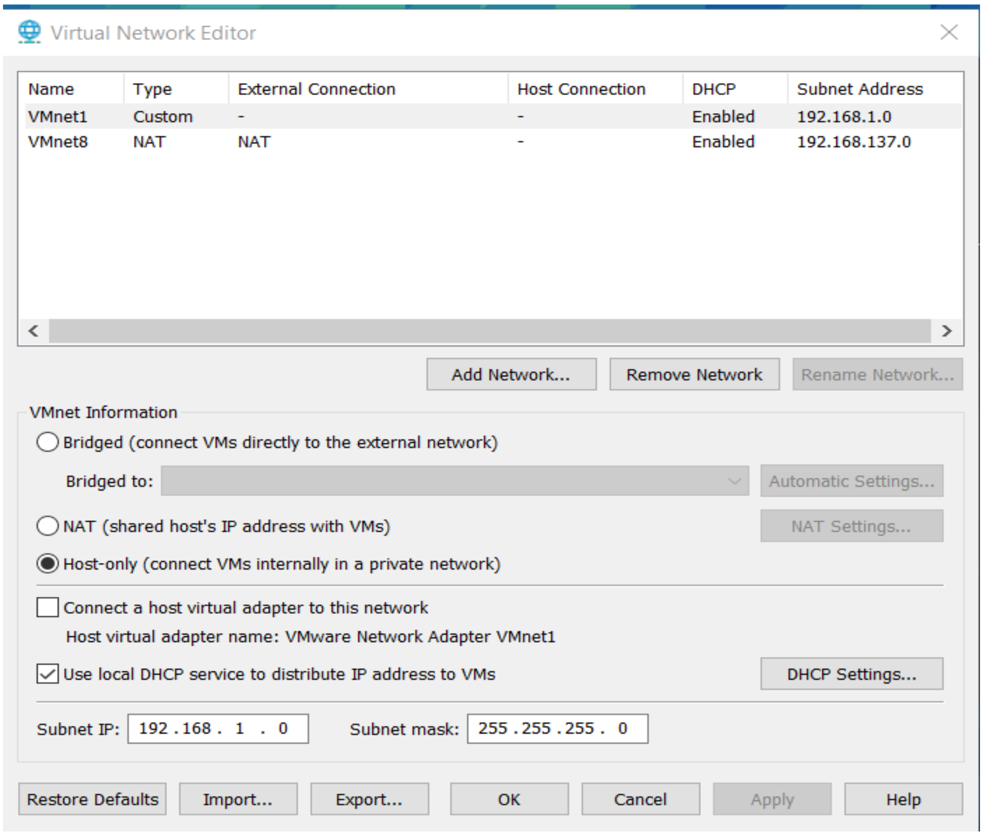
I chose the Windows 10 operating system because of the fact that windows is by far the most common operating system today.

Both of the machines are on the same network so that communication would be enabled.
In order to prevent the malware from spreading to the host, the virtual machines' network adapter is set on "host only". When "host only" is configured, VMware creates a virtual network adapter on the host and on the virtual machines and connect the 2 machines without relying on the host's physical network adapter. The host's physical network adapter is still connected to the internet or any other network. This means that the virtual machines are not connected to the internet or any other network in order to prevent the malware from spreading onto different networks and machines.

Both the attacking machine and the "victim's" machine have got to be on the same network so that the attacking machine would be able to share the malware with the "victim's" computer and by that - infecting it with the malware. Furthermore, since the machines are on the same network, capturing the network traffic between them is possible and by that enabling investigation of the traffic as the malware is running (executed).

2) Configuring The Network

VMware "Virtual Network Editor" setting:
network adapter name- **VMnet1**
Type-"**host only**"
IP addresses + subnet mask- **192.168.1.0/24**



Virtual Network Editor

Name	Type	External Connection	Host Connection	DHCP	Subnet Address
VMnet1	Custom	-	-	Enabled	192.168.1.0
VMnet8	NAT	NAT	-	Enabled	192.168.137.0

VMnet Information

○ Bridged (connect VMs directly to the external network)

Bridged to:

○ NAT (shared host's IP address with VMs)

● Host-only (connect VMs internally in a private network)

□ Connect a host virtual adapter to this network

Host virtual adapter name: VMware Network Adapter VMnet1

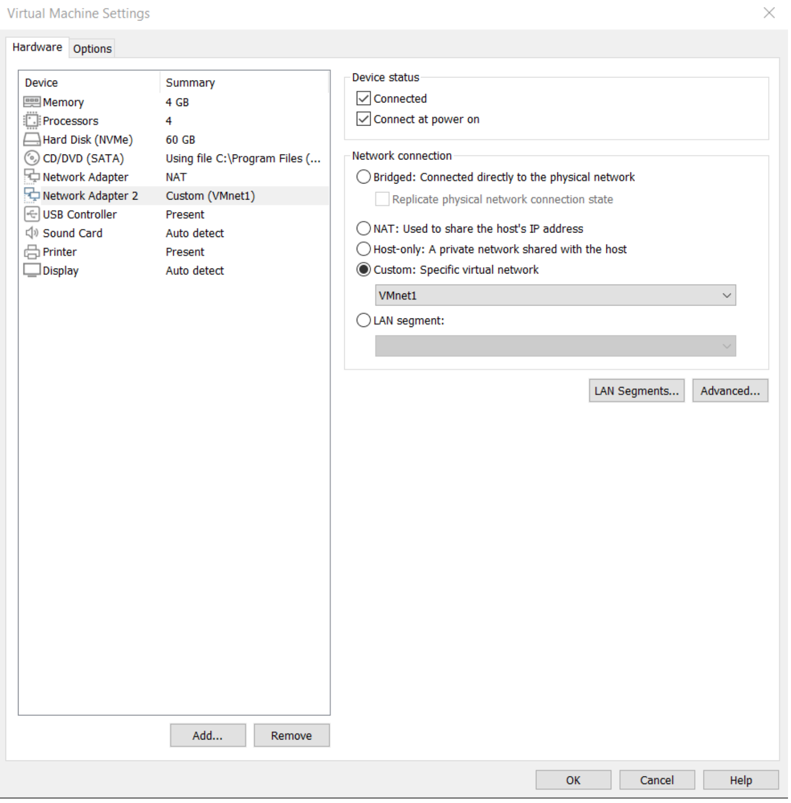
☑ Use local DHCP service to distribute IP address to VMs

Subnet IP: 192 . 168 . 1 . 0 Subnet mask: 255 . 255 . 255 . 0

Configuring the machines network adapters (the same network) – **VMnet1 - host only**

The attacker's machine

The victim's machine



Virtual Machine Settings

Hardware | Options

Device	Summary
Memory	4 GB
Processors	4
Hard Disk (NVMe)	60 GB
CD/DVD (SATA)	Using file C:\Program Files (...
Network Adapter	NAT
Network Adapter 2	Custom (VMnet1)
USB Controller	Present
Sound Card	Auto detect
Printer	Present
Display	Auto detect

Device status

☑ Connected

☑ Connect at power on

Network connection

○ Bridged: Connected directly to the physical network

☐ Replicate physical network connection state

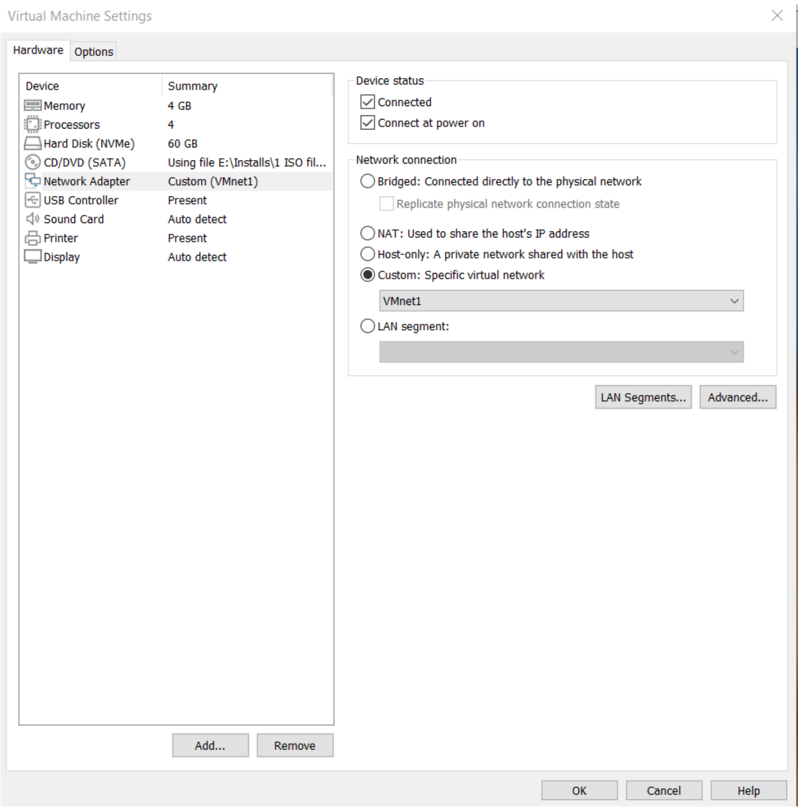
○ NAT: Used to share the host's IP address

○ Host-only: A private network shared with the host

● Custom: Specific virtual network

VMnet1

○ LAN segment:



Virtual Machine Settings

Hardware | Options

Device	Summary
Memory	4 GB
Processors	4
Hard Disk (NVMe)	60 GB
CD/DVD (SATA)	Using file E:\Installs\1 ISO fil...
Network Adapter	Custom (VMnet1)
USB Controller	Present
Sound Card	Auto detect
Printer	Present
Display	Auto detect

Device status

☑ Connected

☑ Connect at power on

Network connection

○ Bridged: Connected directly to the physical network

☐ Replicate physical network connection state

○ NAT: Used to share the host's IP address

○ Host-only: A private network shared with the host

● Custom: Specific virtual network

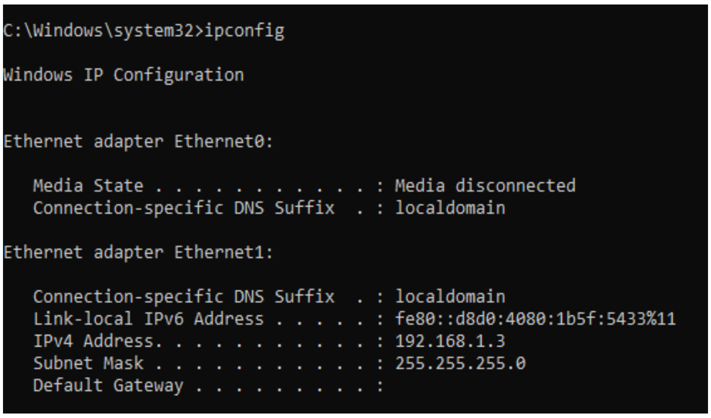
VMnet1

○ LAN segment:

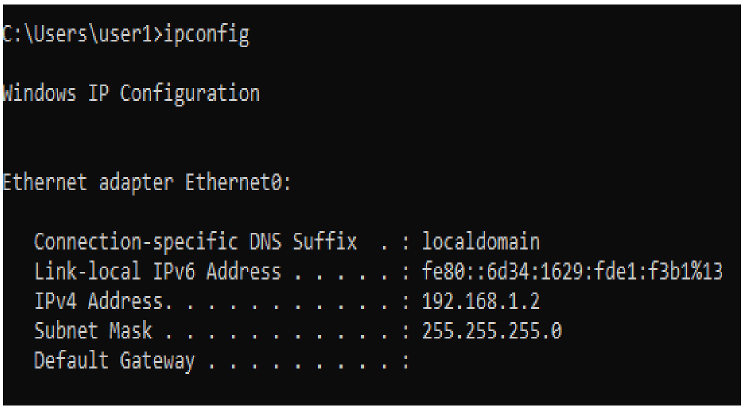
IP Addresses:

The attacker's machine:
IP address-192.168.1.3 /24
the second adapter (Ethernet0) will be specified (later)

The victim's machine:
IP address-192.168.1.2 /24



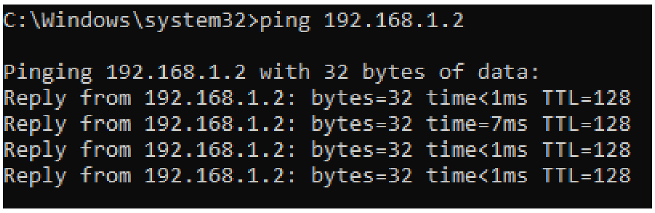
```
C:\Windows\system32>ipconfig

Windows IP Configuration


Ethernet adapter Ethernet0:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . : localdomain

Ethernet adapter Ethernet1:

   Connection-specific DNS Suffix  . : localdomain
   Link-local IPv6 Address . . . . . : fe80::d8d0:4080:1b5f:5433%11
   IPv4 Address. . . . . . . . . . . : 192.168.1.3
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . :
```



```
C:\Users\user1>ipconfig

Windows IP Configuration


Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . : localdomain
   Link-local IPv6 Address . . . . . : fe80::6d34:1629:fde1:f3b1%13
   IPv4 Address. . . . . . . . . . . : 192.168.1.2
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . :
```

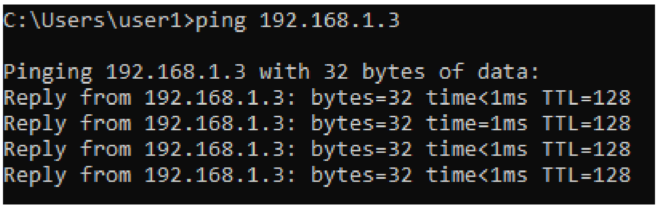
Checking connection between them:

The attacker's machine pinged the victim's machine successfully



```
C:\Windows\system32>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time=7ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
```

The victim's machine pinged the attacker's machine successfully

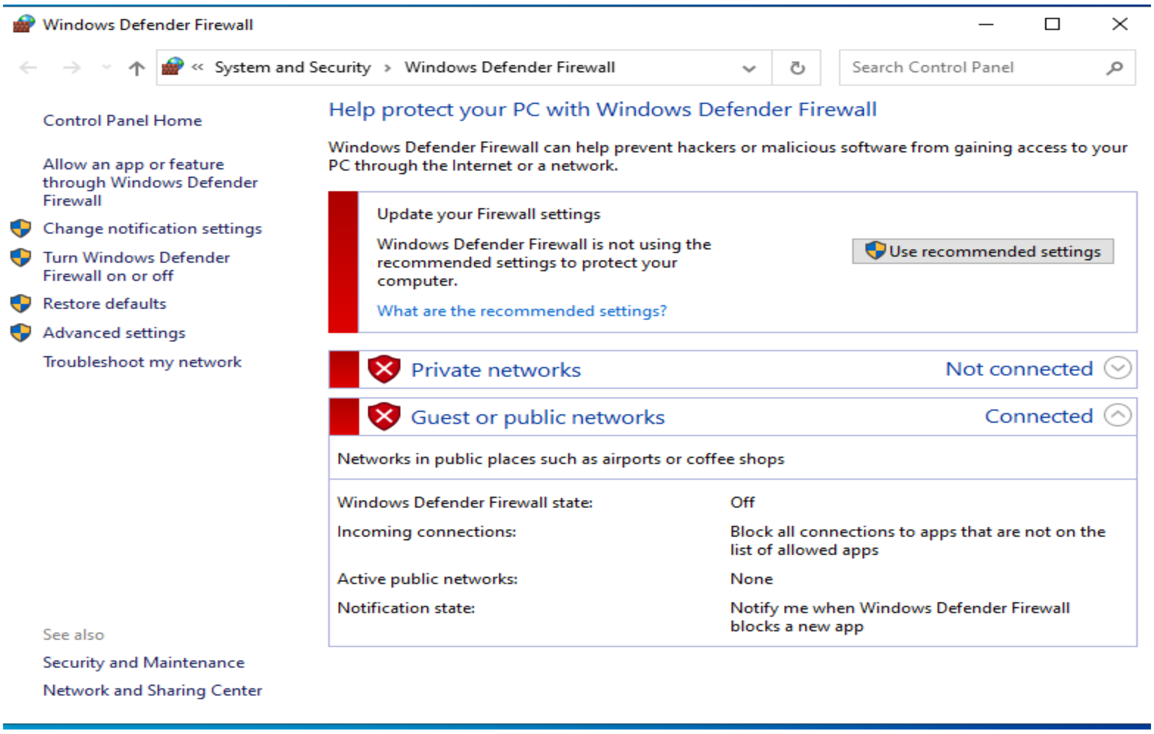


```
C:\Users\user1>ping 192.168.1.3

Pinging 192.168.1.3 with 32 bytes of data:
Reply from 192.168.1.3: bytes=32 time<1ms TTL=128
Reply from 192.168.1.3: bytes=32 time=1ms TTL=128
Reply from 192.168.1.3: bytes=32 time<1ms TTL=128
Reply from 192.168.1.3: bytes=32 time<1ms TTL=128
```

3) Preparing the machines for the analysis

In order to prevent the malware file from being on the computer or from being executed on the computer, both the firewall and the real-time protection need to be turned off so they won't block or prevent the malware from taking actions.

Windows defender firewall is disabled on both machines



Windows Defender Firewall
« System and Security › Windows Defender Firewall
Search Control Panel
Control Panel Home
Allow an app or feature through Windows Defender Firewall
Change notification settings
Turn Windows Defender Firewall on or off
Restore defaults
Advanced settings
Troubleshoot my network
See also
Security and Maintenance
Network and Sharing Center
Use recommended settings

Help protect your PC with Windows Defender Firewall

Windows Defender Firewall can help prevent hackers or malicious software from gaining access to your PC through the Internet or a network.

Update your Firewall settings

Windows Defender Firewall is not using the recommended settings to protect your computer.

What are the recommended settings?

Private networks — Not connected

Guest or public networks — Connected

Networks in public places such as airports or coffee shops

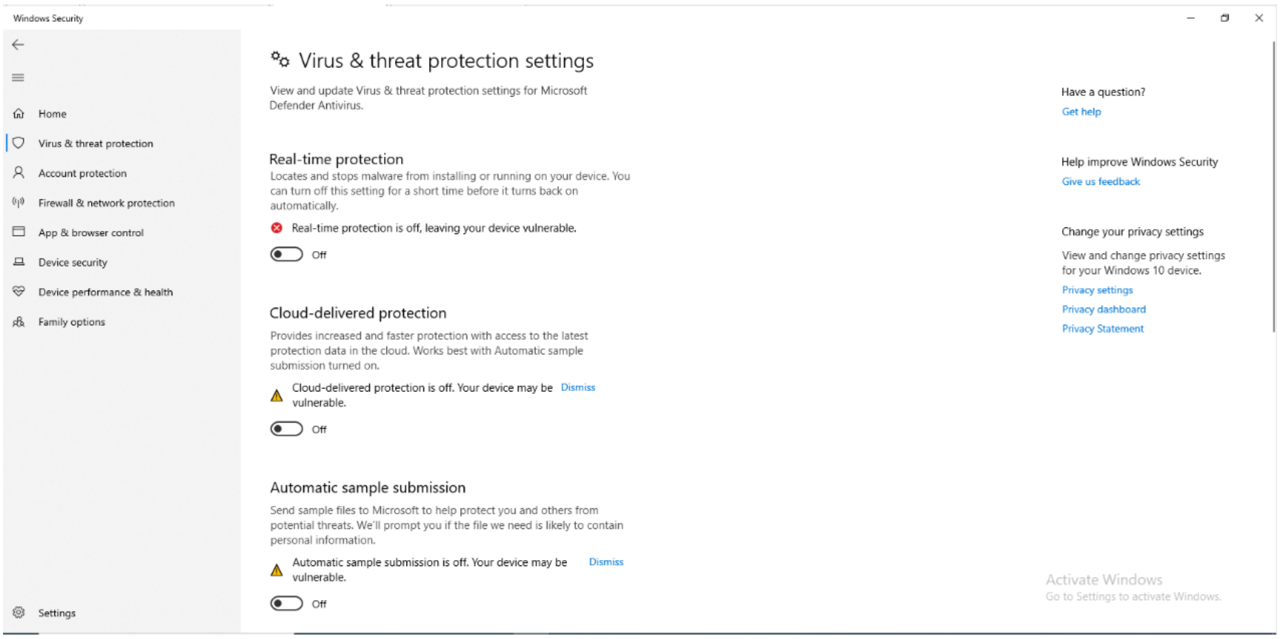
Windows Defender Firewall state: Off

Incoming connections: Block all connections to apps that are not on the list of allowed apps

Active public networks: None

Notification state: Notify me when Windows Defender Firewall blocks a new app

Windows Real-Time Protection is disabled on both machines


Windows Security
Home
Virus & threat protection
Account protection
Firewall & network protection
App & browser control
Device security
Device performance & health
Family options
Settings
Have a question?
Get help
Help improve Windows Security
Give us feedback
Change your privacy settings
View and change privacy settings for your Windows 10 device.
Privacy settings
Privacy dashboard
Privacy Statement
Activate Windows
Go to Settings to activate Windows.

Virus & threat protection settings

View and update Virus & threat protection settings for Microsoft Defender Antivirus.

Real-time protection

Locates and stops malware from installing or running on your device. You can turn off this setting for a short time before it turns back on automatically.

Real-time protection is off, leaving your device vulnerable.

Off

Cloud-delivered protection

Provides increased and faster protection with access to the latest protection data in the cloud. Works best with Automatic sample submission turned on.

Cloud-delivered protection is off. Your device may be vulnerable. Dismiss

Off

Automatic sample submission

Send sample files to Microsoft to help protect you and others from potential threats. We'll prompt you if the file we need is likely to contain personal information.

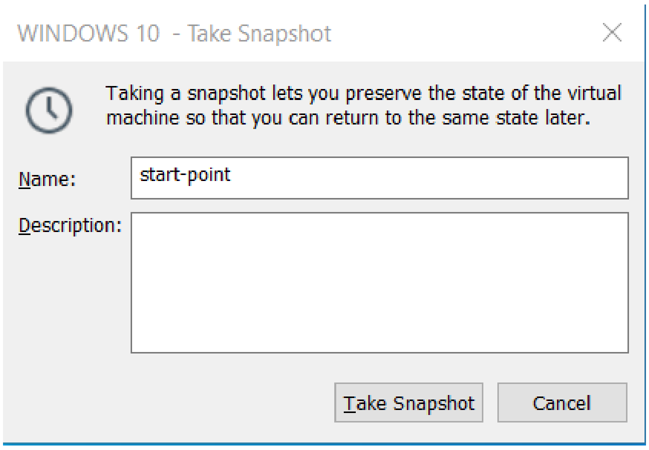
Automatic sample submission is off. Your device may be vulnerable. Dismiss

Off

4) Taking snapshots

VMware's virtual machine "snapshot" option allows us to save a computer's current state and return to the same state later, in case of need. Taking snapshots when performing "malware analysis" lab is extremely important because of the high probability that the malware would perform changes on the machine (the malware might add keys, delete files, etc.) and snapshots can help us get back to the starting point at any time.

After I've installed the OS and the required analysis tools on the virtual machine and configured the network settings, I took a snapshot. That snapshot is now the foundation of my lab.



Taking a snapshot lets you preserve the state of the virtual machine so that you can return to the same state later.

Name: start-point

Description:

Afterwards, I installed all the tools for the analysis and took another snapshot.
I executed the malware, performed the analysis, and reverted to the initial snapshot in order to run another lab under the same conditions as the previous lab.
(This will be presented below)

The Malware

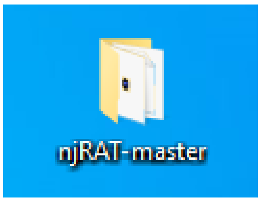
In this project I will analyze the malware "NJrat". I chose this malware because of its extensive capabilities and also because "NJrat" is one of the most used RAT's in the world.

NJrat is a variant of Remote Access Trojan that is also known as "Bladabindi" and it is used to gain remote control over infected machines. NJrat has the ability to activate the machine's webcam, log keystrokes, steal passwords from the web browser, access the machine's command line, kill running processes, and manipulate the machine's registry and more.

The NJrat malware uses quite a few attack vectors to infect its "victims". For example, the malware targets discord users as span campaigns. Another way the malware finds its way to infect machines is through a compromised website. The website tricks users to download a fake software update that installs the malware on the machine.
Once the NJrat arrived at the targeted machine and infected it, its malicious activity has begun.

The creation of the malware - attacker's computer:
This machine has 2 network adapters:
- NAT- (temporary) - to enable internet connection in order to download the NJrat malware.
- VMnet1 - the internal network that consists of the 2 machines discussed in my lab.

I got the NJrat from https://github.com/AliBawazeEer/RAT-NjRat-0.7d-modded-source-code.git , I extracted the files and saved it on the desktop.

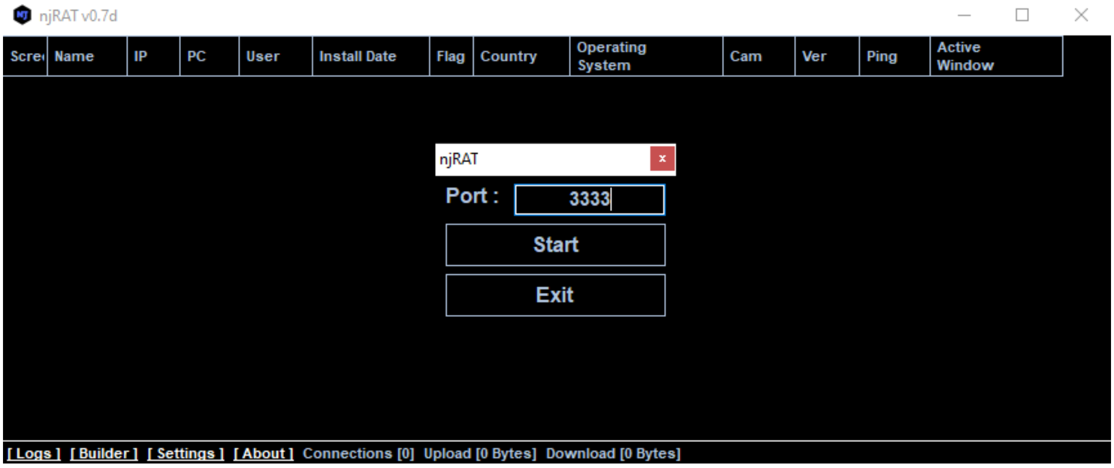


After downloading, the NAT network adapter is not necessary and therefore disconnected.

This application is the malware building. NjRAT provides a simple C2 (command & control) interface allowing the threat actor to easily interact with victim's machines.

njRAT v0.7d 4/12/2021 12:15 PM Application 1,684 KB

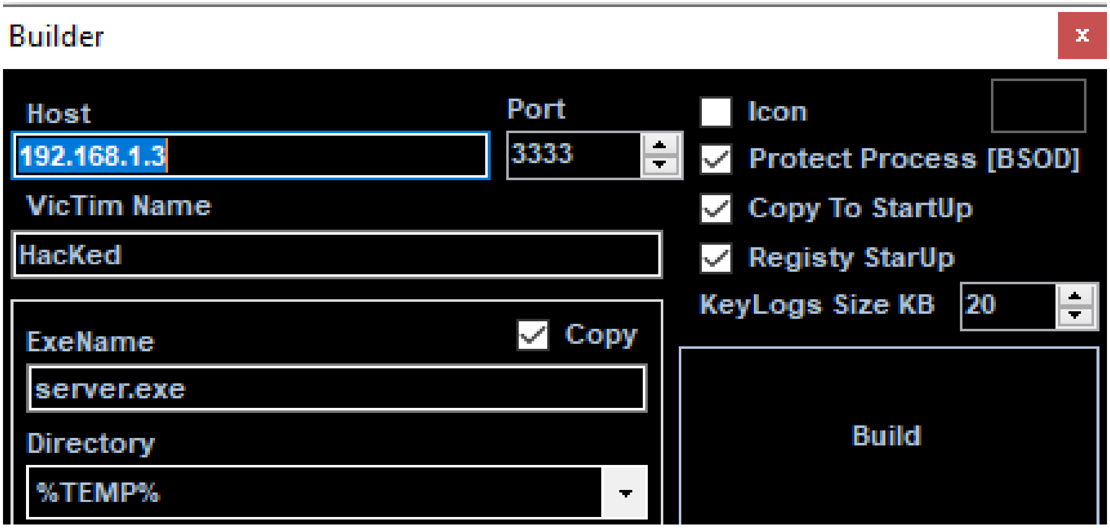
When I open the NJrat malware I first choose the port number I want the software to listen to. I chose port 3333 since it is easy to find when using "Wireshark". After choosing the port number, I hit "start



Port : 3333

Connections [0] Upload [0 Bytes] Download [0 Bytes]

After this, I built the “server” file.



Host: 192.168.1.3

Port: 3333

VicTim Name: HacKed

ExeName: server.exe ☑ Copy

Directory: %TEMP%

□ Icon

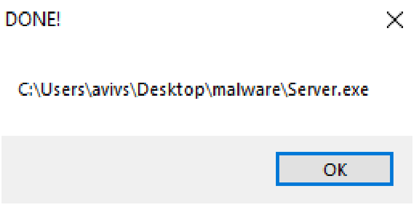
☑ Protect Process [BSOD]

☑ Copy To StartUp

☑ Registy StarUp

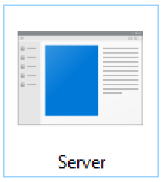
KeyLogs Size KB: 20

The Host – 192.168.1.3 (the attacker's machine).
The malware will connect to port 3333 on the attacker's machine. I pressed "Build" button and the "Server" file was created. I saved it inside a folder named "Malware". The .exe file will open a remote connection to the "victim's" machine.



DONE!

C:\Users\avivs\Desktop\malware\Server.exe



Before I infect the "victim's" machine with the malware, I must prepare the machine and install the analysis tools required for my investigation of the malware and take a snapshot of this state in order to come back in case of need.

The Tools

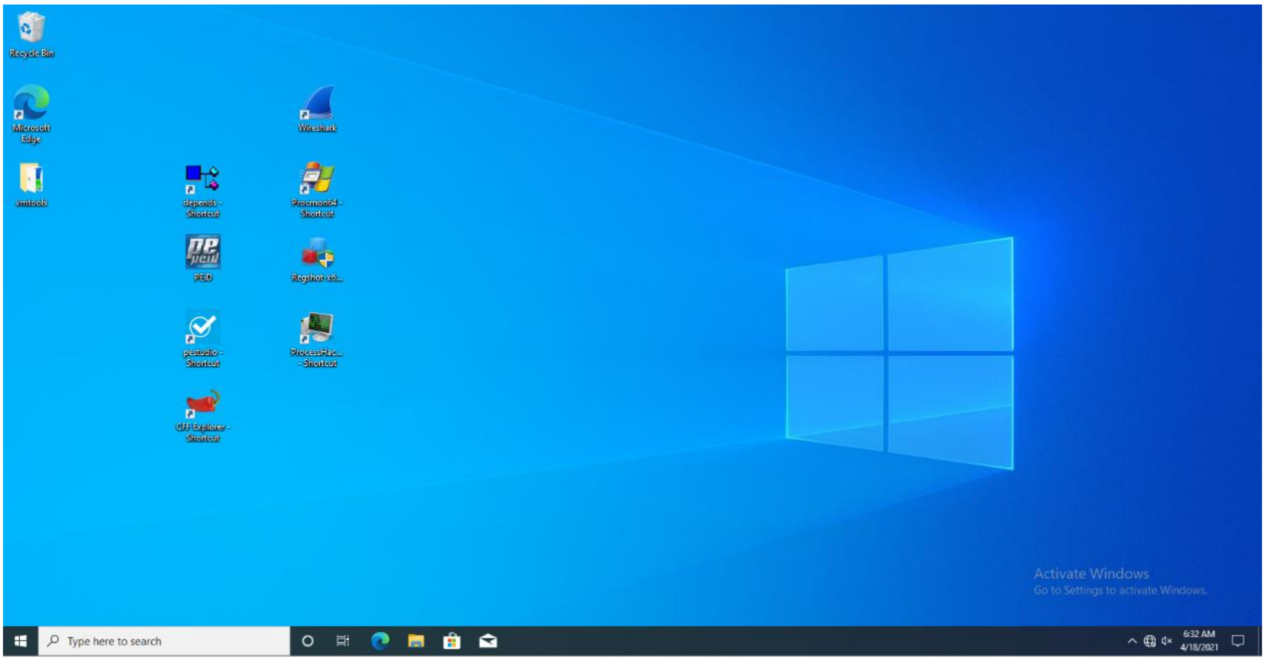
Static analysis tools-

DIE- Detect It Easy
Dependency Walker
PEstudio
CFF Explorer

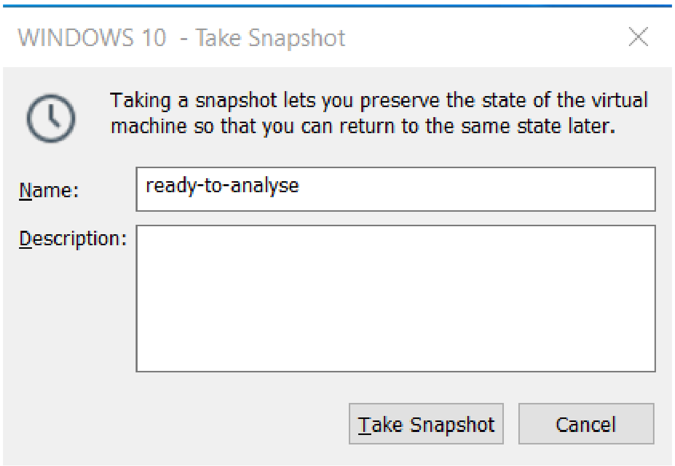
Dynamic analysis tools-

Wireshark
Process Monitor
Process Hacker
Regshot

The victim's computer- with the tools



Snapshot of this state- Ready-to-analyze


WINDOWS 10 - Take Snapshot

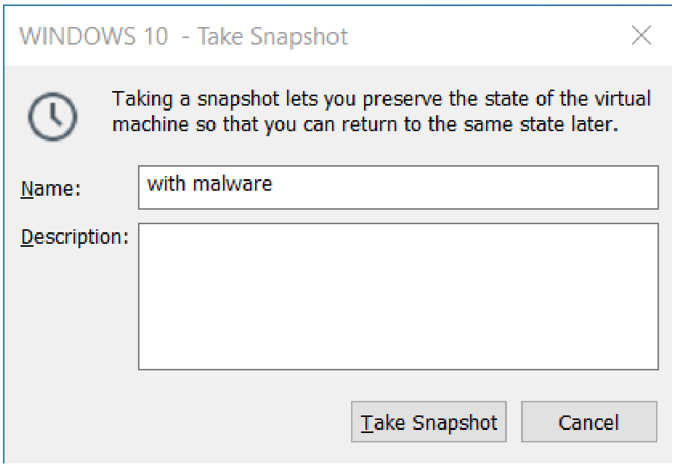
Taking a snapshot lets you preserve the state of the virtual machine so that you can return to the same state later.

Name: ready-to-analyse

Description:

In order to get the malware to be on my machine that is used for analysis purposes, I shared the "malware" folder from the attacker's machine onto the network, on the "victim's" machine I copied the file "server.exe" and put it on the desktop.

As soon as the malware was present on my analysis machine plus I had all the required tools to analyze it, I took a snapshot of the machine's current state and called it "With Malware".



WINDOWS 10 - Take Snapshot

Taking a snapshot lets you preserve the state of the virtual machine so that you can return to the same state later.

Name: with malware

Description:

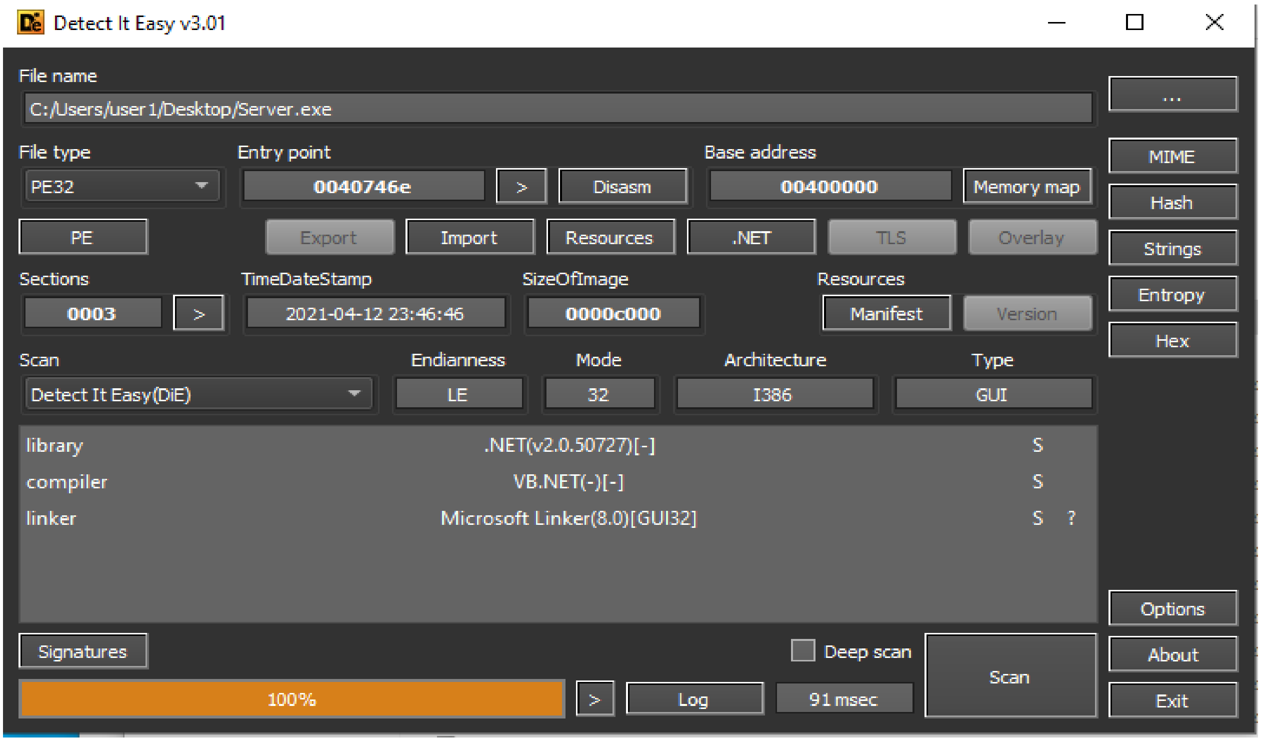
Static Analysis

The "Static Analysis" phase consists of examining the executable file without viewing the actual instructions. Basic static analysis can confirm whether or not a file is malicious, provide information about its functionality and sometimes even provide information that will allow you to produce simple network signatures. Basic static analysis is a straight-forward procedure that can be quick but it is largely ineffective against more sophisticated malware and it can miss important behaviors.

Packers:
Malware writers or creators often use packing or obfuscation to make their files more difficult to analyze and detect. Obfuscated programs are processes in which the author of the malware has attempted to hide. Packed programs are a subset of obfuscated programs in which the malicious file is compressed and cannot be analyzed. Both techniques would severely limit your chances to successfully statically analyze the malware.

DIE- Detect It Easy is an application that has been built as a packer identifier in order to help define a file type.

I'm using DIE to find whether the malware is packed or not:



Detect It Easy v3.01

File name: C:/Users/user1/Desktop/Server.exe

File type: PE32

Entry point: 0040746e

Base address: 00400000

Sections: 0003

TimeDateStamp: 2021-04-12 23:46:46

SizeOfImage: 0000c000

Scan: Detect It Easy(DiE)

Endianness: LE

Mode: 32

Architecture: I386

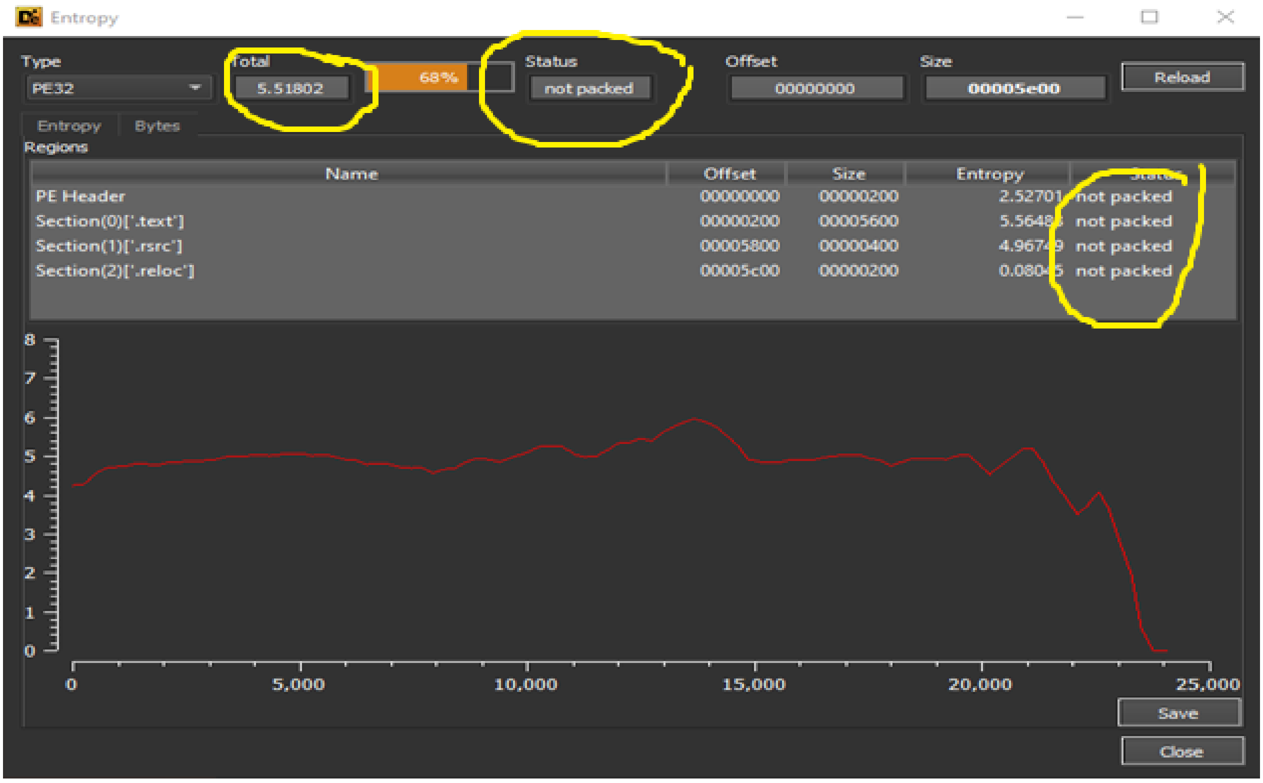
Type: GUI

```
library     .NET(v2.0.50727)[-]            S
compiler    VB.NET(-)[-]                   S
linker      Microsoft Linker(8.0)[GUI32]   S  ?
```

100%

91 msec

measuring the code's entropy may help malware researchers determine whether or not a sample of the malware has been obfuscated in a way, compressed or encrypted. The most popular way to measure the entropy of the code is based on "Shannon's Formula". Using this formula, each binary is measured on a scale from 0 to 8.15. Entropy has a very clear relationship with malware, the more the entropy the more it is likely to be obfuacated or contain encrypted data; therefore the likeliness of a malicious file increases.



Entropy

Type: PE32

Total: 5.51802 68%

Status: not packed

Offset: 00000000

Size: 00005e00

Regions

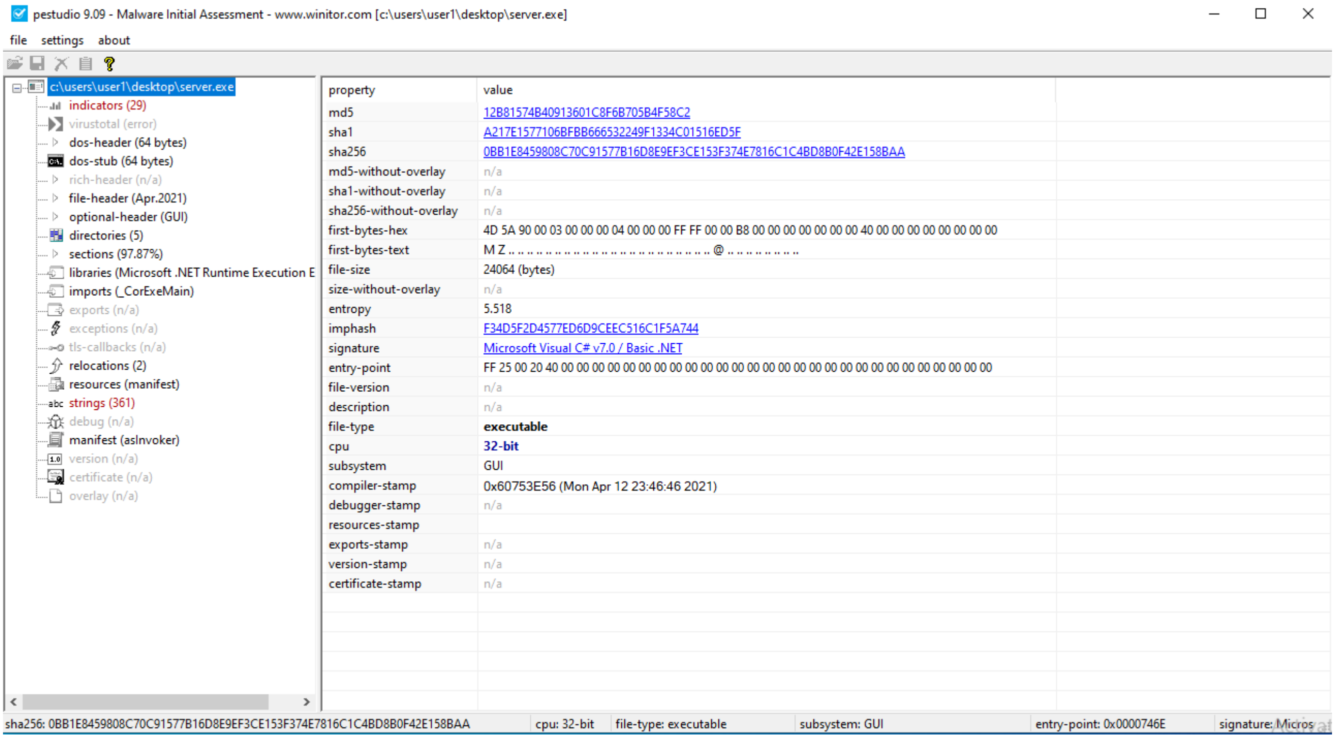
Name	Offset	Size	Entropy	Status
PE Header	00000000	00000200	2.52701	not packed
Section(0)['.text']	00000200	00005600	5.5648[illegible]	not packed
Section(1)['.rsrc']	00005800	00000400	4.967[illegible]9	not packed
Section(2)['.reloc']	00005c00	00000200	0.080[illegible]5	not packed

These finding show:
- The file is compiled with .NET4.0 which means it's written in C or C++. Malware creators often use multiple .NET obfuscations in order to avoid detection by antivirus software.
- The malware is not packed (text entropy is under 5).

Another way to confirm it is by looking at the strings of the malware's code and if the strings are readable, it means that the code isn’t packed and obfuscated.
For looking at the strings, I'm using a tool called- PEstudio.

PEstudio is a free and portable tool which uses static analysis (and other techniques) to help you discover more about suspicious applications.

Before I present the strings, I want to show what other important data I can retrieve about the malware using PEstudio:



pestudio 9.09 - Malware Initial Assessment - www.winitor.com [c:\users\user1\desktop\server.exe]

- c:\users\user1\desktop\server.exe
  - indicators (29)
  - virustotal (error)
  - dos-header (64 bytes)
  - dos-stub (64 bytes)
  - rich-header (n/a)
  - file-header (Apr.2021)
  - optional-header (GUI)
  - directories (5)
  - sections (97.87%)
  - libraries (Microsoft .NET Runtime Execution E
  - imports (_CorExeMain)
  - exports (n/a)
  - exceptions (n/a)
  - tls-callbacks (n/a)
  - relocations (2)
  - resources (manifest)
  - strings (361)
  - debug (n/a)
  - manifest (asInvoker)
  - version (n/a)
  - certificate (n/a)
  - overlay (n/a)

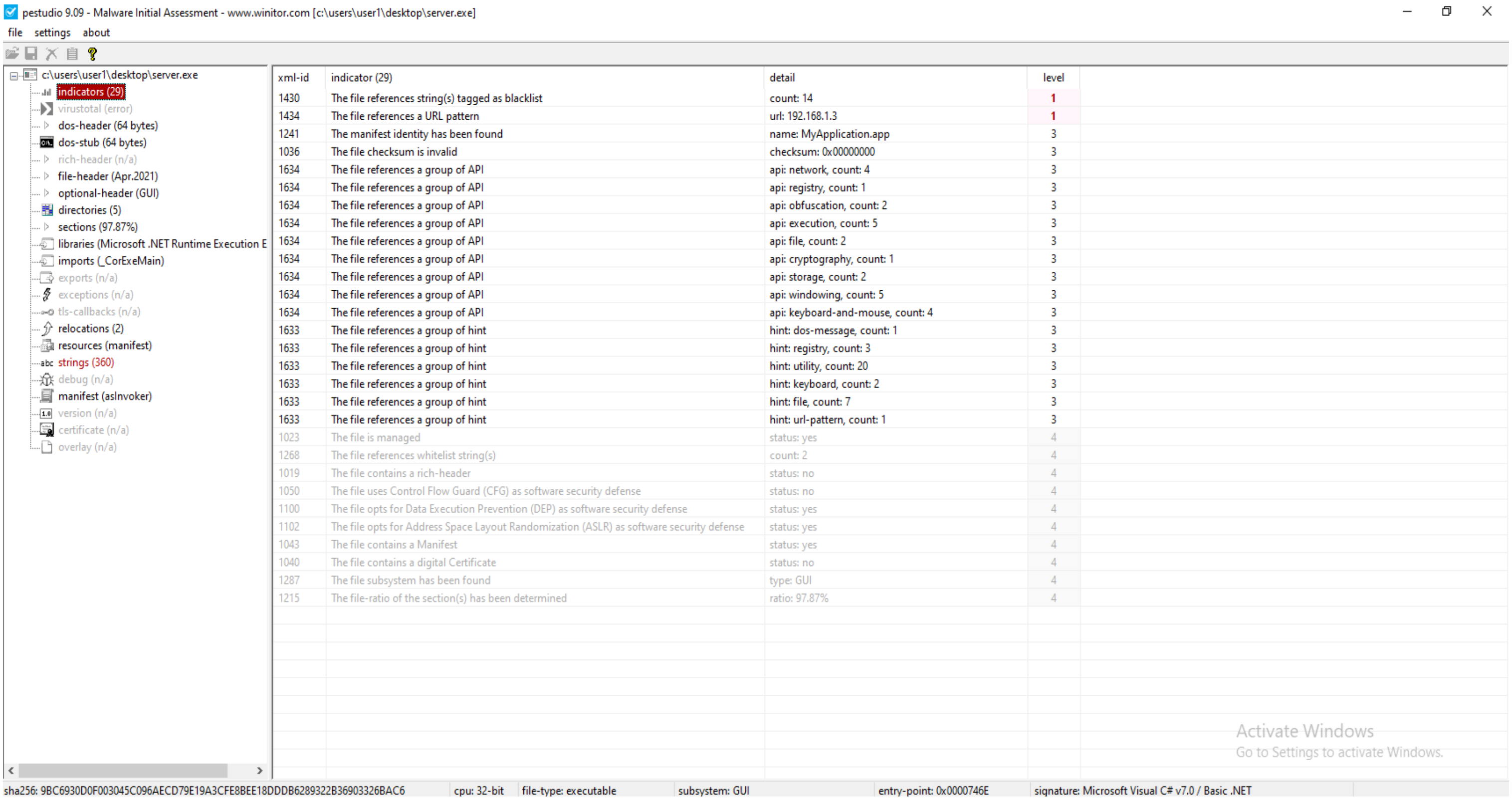
property	value
md5	12B81574B40913601C8F6B705B4F58C2
sha1	A217E1577106BFBB666532249F1334C01516ED5F
sha256	0BB1E8459808C70C91577B16D8E9EF3CE153F374E7816C1C4BD8B0F42E158BAA
md5-without-overlay	n/a
sha1-without-overlay	n/a
sha256-without-overlay	n/a
first-bytes-hex	4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00 B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00 00
first-bytes-text	M Z .. .. @ .. ..
file-size	24064 (bytes)
size-without-overlay	n/a
entropy	5.518
imphash	F34D5F2D4577ED6D9CEEC516C1F5A744
signature	Microsoft Visual C# v7.0 / Basic .NET
entry-point	FF 25 00 20 40 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
file-version	n/a
description	n/a
file-type	**executable**
cpu	**32-bit**
subsystem	GUI
compiler-stamp	0x60753E56 (Mon Apr 12 23:46:46 2021)
debugger-stamp	n/a
resources-stamp	
exports-stamp	n/a
version-stamp	n/a
certificate-stamp	n/a

sha256: 0BB1E8459808C70C91577B16D8E9EF3CE153F374E7816C1C4BD8B0F42E158BAA | cpu: 32-bit | file-type: executable | subsystem: GUI | entry-point: 0x0000746E | signature: Micros

Hashes:
Md5- 12B81574B40913601C8F6B705B4F58C2
Sha1- A217E1577106BFBB666532249F1334C01516ED5F
Sha256- 0BB1E8459808C70C91577B16D8E9EF3CE153F374E7816C1C4BD8B0F42E158BAA

File information:
File type- executable, first bytes hex- 4D 5A…, First bytes text- M Z …
File size- 24064 (bytes)
System type- GUI x32 bit
Signature- Microsoft Visual C# v7.0 / Basic .NET (written in C# language)

IOC- Indictors Of Compromise:

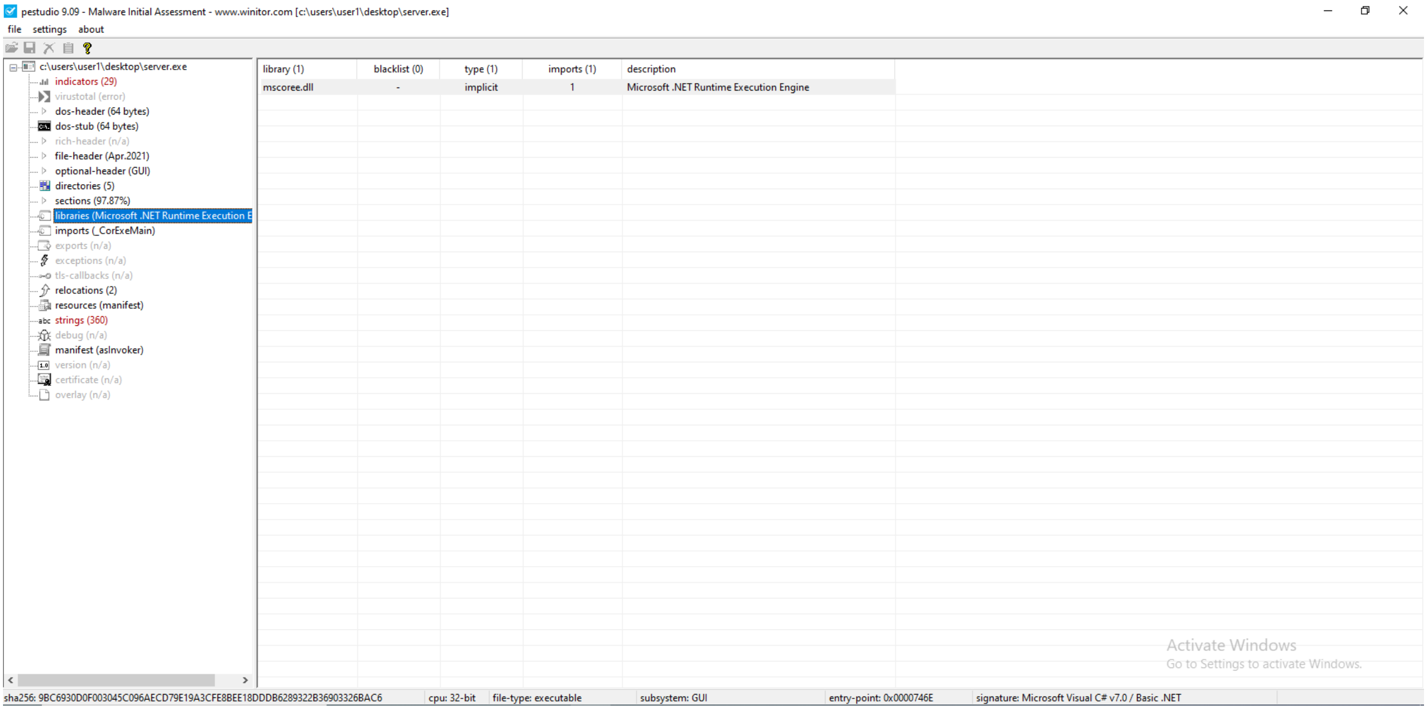


pestudio 9.09 - Malware Initial Assessment - www.winitor.com [c:\users\user1\desktop\server.exe]

xml-id	indicator (29)	detail	level
1430	The file references string(s) tagged as blacklist	count: 14	1
1434	The file references a URL pattern	url: 192.168.1.3	1
1241	The manifest identity has been found	name: MyApplication.app	3
1036	The file checksum is invalid	checksum: 0x00000000	3
1634	The file references a group of API	api: network, count: 4	3
1634	The file references a group of API	api: registry, count: 1	3
1634	The file references a group of API	api: obfuscation, count: 2	3
1634	The file references a group of API	api: execution, count: 5	3
1634	The file references a group of API	api: file, count: 2	3
1634	The file references a group of API	api: cryptography, count: 1	3
1634	The file references a group of API	api: storage, count: 2	3
1634	The file references a group of API	api: windowing, count: 5	3
1634	The file references a group of API	api: keyboard-and-mouse, count: 4	3
1633	The file references a group of hint	hint: dos-message, count: 1	3
1633	The file references a group of hint	hint: registry, count: 3	3
1633	The file references a group of hint	hint: utility, count: 20	3
1633	The file references a group of hint	hint: keyboard, count: 2	3
1633	The file references a group of hint	hint: file, count: 7	3
1633	The file references a group of hint	hint: url-pattern, count: 1	3
1023	The file is managed	status: yes	4
1268	The file references whitelist string(s)	count: 2	4
1019	The file contains a rich-header	status: no	4
1050	The file uses Control Flow Guard (CFG) as software security defense	status: no	4
1100	The file opts for Data Execution Prevention (DEP) as software security defense	status: yes	4
1102	The file opts for Address Space Layout Randomization (ASLR) as software security defense	status: yes	4
1043	The file contains a Manifest	status: yes	4
1040	The file contains a digital Certificate	status: no	4
1287	The file subsystem has been found	type: GUI	4
1215	The file-ratio of the section(s) has been determined	ratio: 97.87%	4

sha256: 9BC6930D0F003045C096AECD79E19A3CFE8BEE18DDDB6289322B36903326BAC6 | cpu: 32-bit | file-type: executable | subsystem: GUI | entry-point: 0x0000746E | signature: Microsoft Visual C# v7.0 / Basic .NET

Present pieces of code that indicate that this is a suspicious file
-14 strings tagged as blacklist.
- An IP address (to which the malware is connected to).
-The manifest identity – MyApplication.app.
-The API's groups (network, registry, keyboard-and-mouse, etc.)



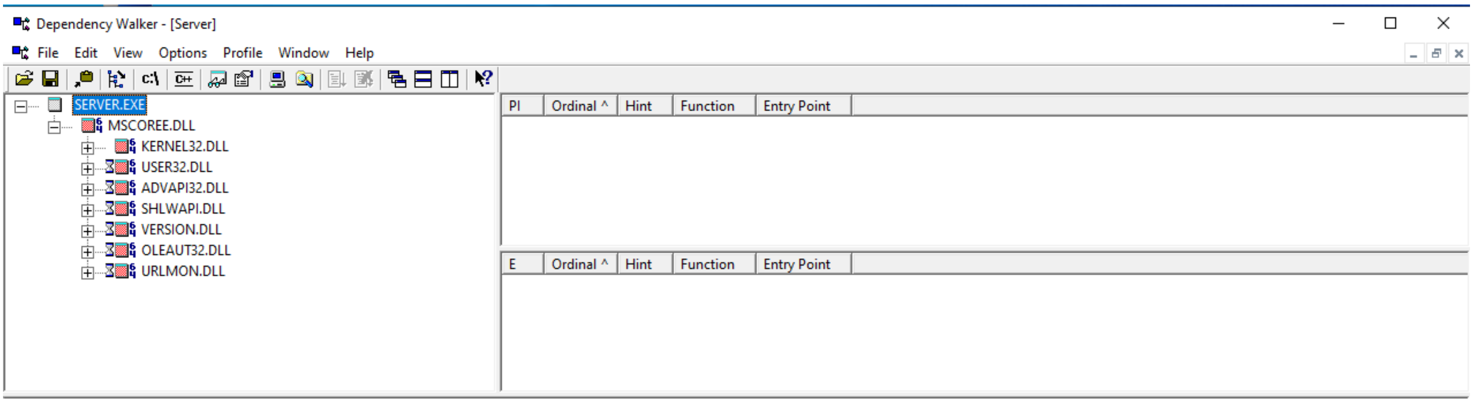
pestudio 9.09 - Malware Initial Assessment - www.winitor.com [c:\users\user1\desktop\server.exe]

```
c:\users\user1\desktop\server.exe
  indicators (29)
  virustotal (error)
  dos-header (64 bytes)
  dos-stub (64 bytes)
  rich-header (n/a)
  file-header (Apr.2021)
  optional-header (GUI)
  directories (5)
  sections (97.87%)
  libraries (Microsoft .NET Runtime Execution E
  imports (_CorExeMain)
  exports (n/a)
  exceptions (n/a)
  tls-callbacks (n/a)
  relocations (2)
  resources (manifest)
  strings (360)
  debug (n/a)
  manifest (asInvoker)
  version (n/a)
  certificate (n/a)
  overlay (n/a)
```

library (1)	blacklist (0)	type (1)	imports (1)	description
mscoree.dll	-	implicit	1	Microsoft .NET Runtime Execution Engine

sha256: 9BC6930D0F003045C096AECD79E19A3CFE8BEE18DDDB6289322B36903326BAC6 | cpu: 32-bit | file-type: executable | subsystem: GUI | entry-point: 0x0000746E | signature: Microsoft Visual C# v7.0 / Basic .NET

The only library that was found is- mscoree.dll.
 MSCorEE.dll is a Microsoft library file which is essential for the execution of "managed code" applications written for use with the .NET Framework.
This library contains all the sub libraries the executable (.exe) file is using. In order to see all the libraries the file is using, I use a tool named "Dependency Walker"- This tool can recursively scan all the dependent DLLs that are used by a program:



Dependency Walker - [Server]

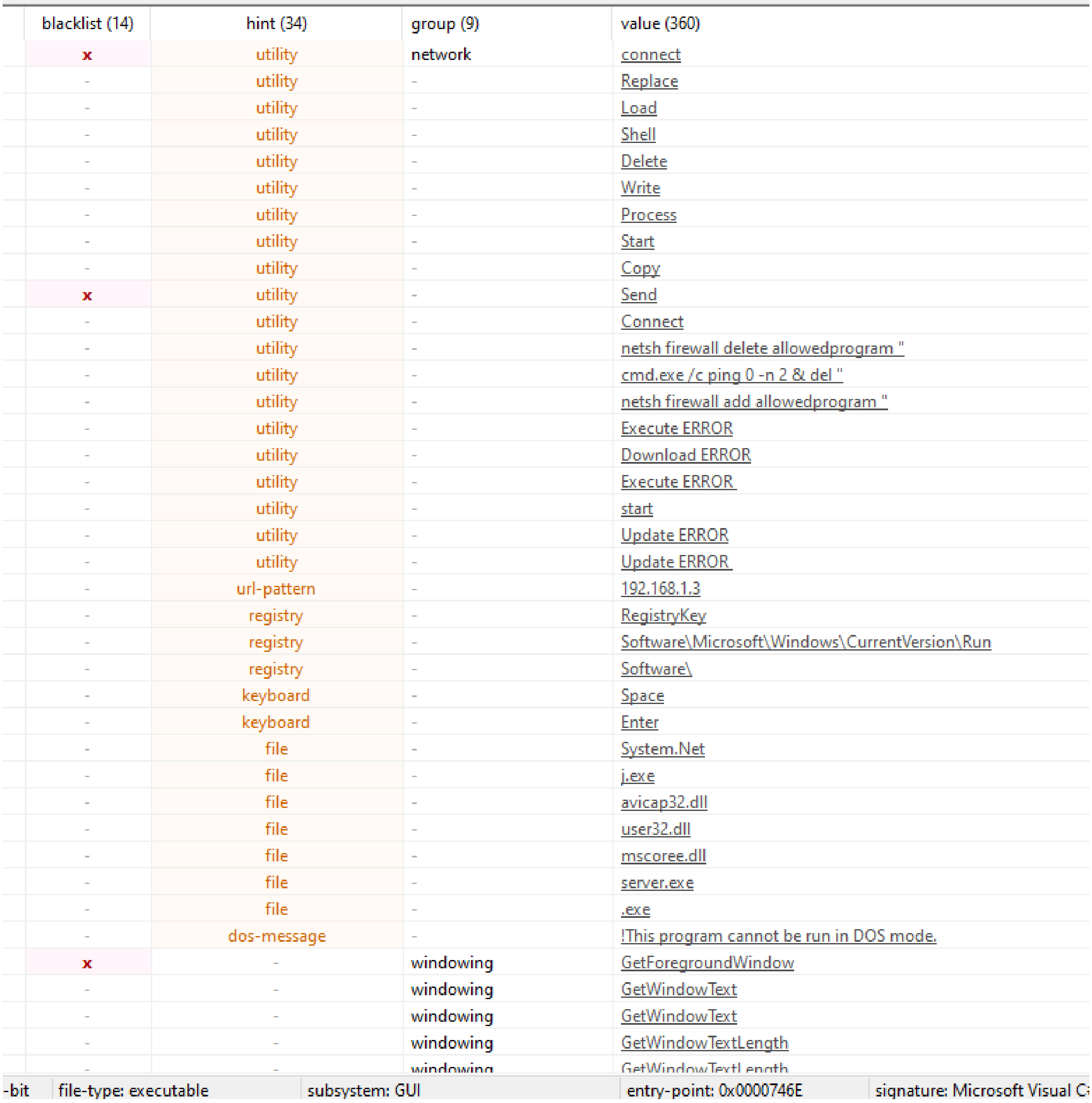
```
SERVER.EXE
  MSCOREE.DLL
    KERNEL32.DLL
    USER32.DLL
    ADVAPI32.DLL
    SHLWAPI.DLL
    VERSION.DLL
    OLEAUT32.DLL
    URLMON.DLL
```

PI	Ordinal ^	Hint	Function	Entry Point

E	Ordinal ^	Hint	Function	Entry Point

In addition, those libraries will be presented on the strings list.
A program contains strings if it prints a message, connects to a URL, or copies a file to a specific location, etc. Searching through the strings can be a simple way to get hints about the functionality of a program.

PEstudio- Strings:



blacklist (14)	hint (34)	group (9)	value (360)
x	utility	network	connect
-	utility	-	Replace
-	utility	-	Load
-	utility	-	Shell
-	utility	-	Delete
-	utility	-	Write
-	utility	-	Process
-	utility	-	Start
-	utility	-	Copy
x	utility	-	Send
-	utility	-	Connect
-	utility	-	netsh firewall delete allowedprogram "
-	utility	-	cmd.exe /c ping 0 -n 2 & del "
-	utility	-	netsh firewall add allowedprogram "
-	utility	-	Execute ERROR
-	utility	-	Download ERROR
-	utility	-	Execute ERROR
-	utility	-	start
-	utility	-	Update ERROR
-	utility	-	Update ERROR
-	url-pattern	-	192.168.1.3
-	registry	-	RegistryKey
-	registry	-	Software\Microsoft\Windows\CurrentVersion\Run
-	registry	-	Software\
-	keyboard	-	Space
-	keyboard	-	Enter
-	file	-	System.Net
-	file	-	j.exe
-	file	-	avicap32.dll
-	file	-	user32.dll
-	file	-	mscoree.dll
-	file	-	server.exe
-	file	-	.exe
-	dos-message	-	!This program cannot be run in DOS mode.
x	-	windowing	GetForegroundWindow
-	-	windowing	GetWindowText
-	-	windowing	GetWindowText
-	-	windowing	GetWindowTextLength
-	-	windowing	GetWindowTextLength

-bit | file-type: executable | subsystem: GUI | entry-point: 0x0000746E | signature: Microsoft Visual C

Connect, 192.168.1.3- tries to connect the URL address.

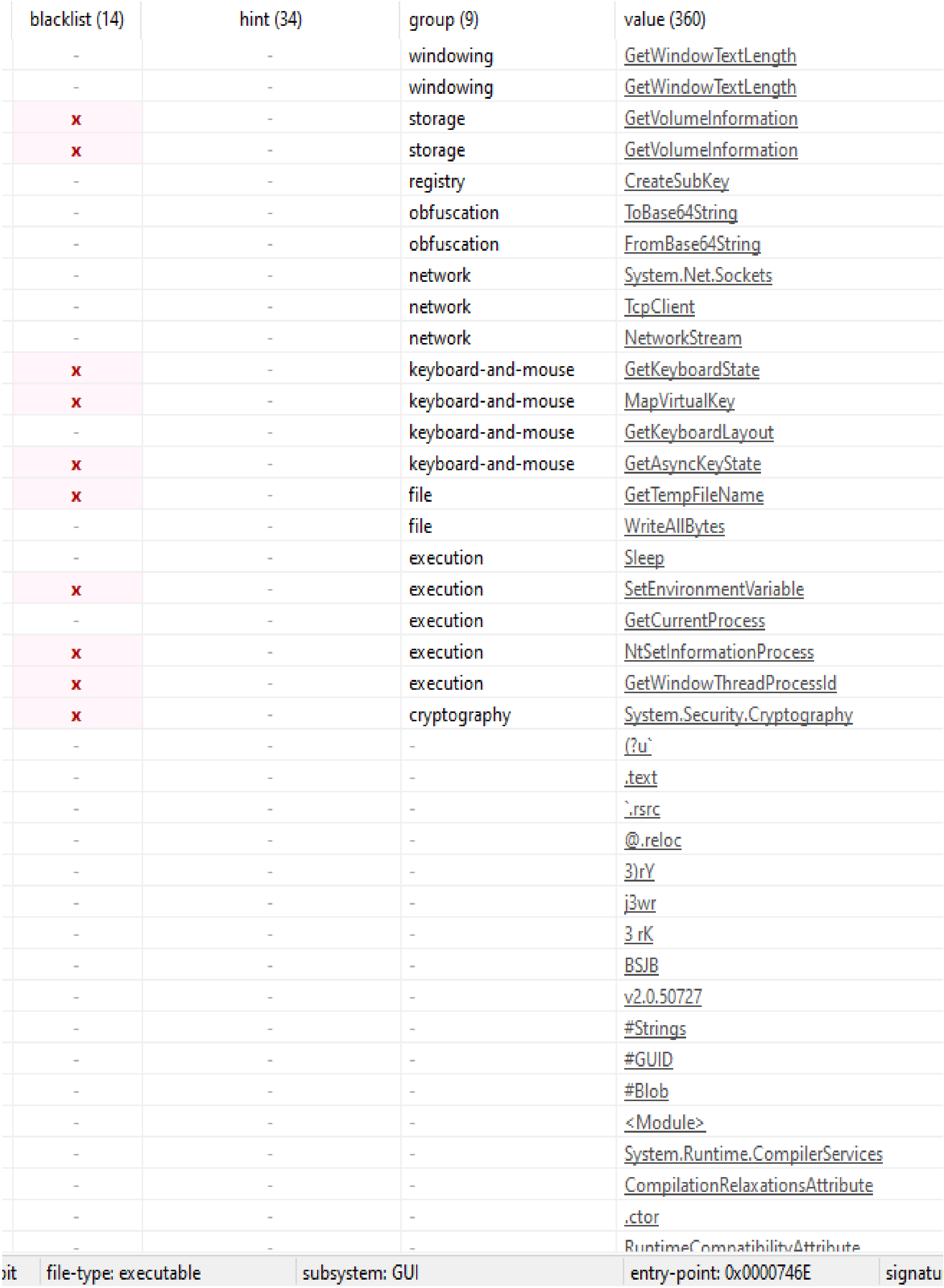
Replace, Copy, Write, Load, Delete, Start, Process- All of which are actions taken on processes.

Shell, cmd.exe, ping 0 –n2 - using command line trying to make a connection.

Netsh firewall delete\add aloowedprogram- Netsh command is used for configuring firewall and its exceptions. It will create a firewall policy to add itself as an ‘allowedprogram’.

Registrykey, Software\Microsoft\Windows\CurrentVersion\Run- making changes on the registry- keys.

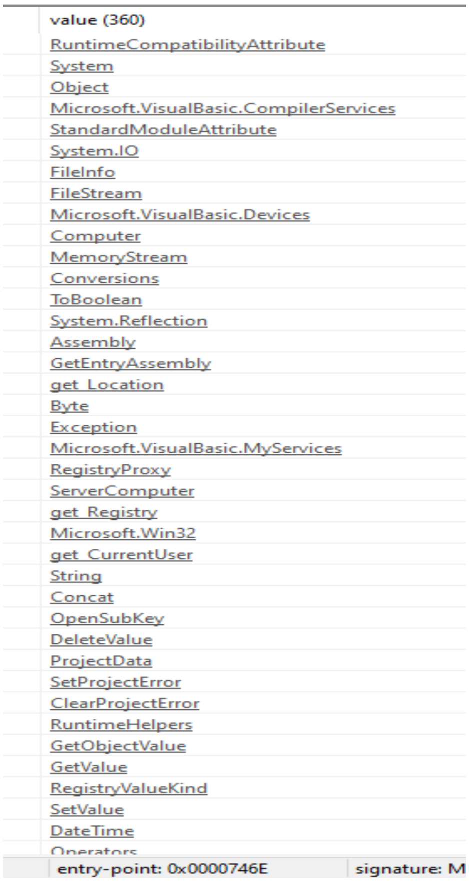
Avicap32.dll, user32.dll mscoree.dll- the dll files (shown before)



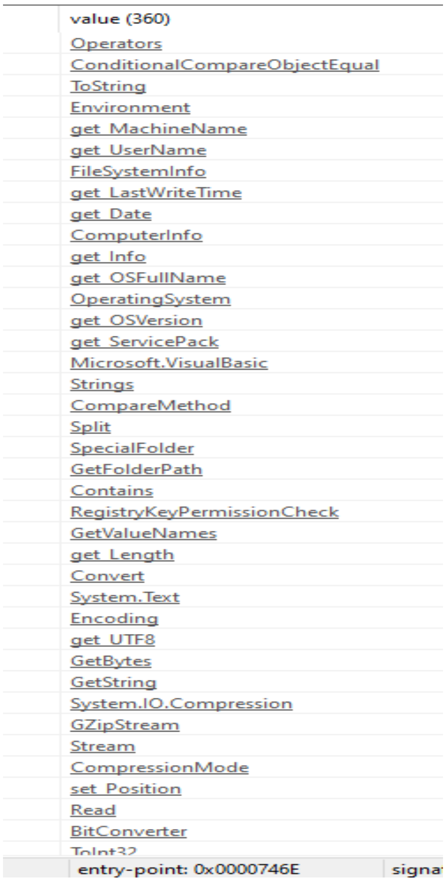
blacklist (14)	hint (34)	group (9)	value (360)
-	-	windowing	GetWindowTextLength
-	-	windowing	GetWindowTextLength
x	-	storage	GetVolumeInformation
x	-	storage	GetVolumeInformation
-	-	registry	CreateSubKey
-	-	obfuscation	ToBase64String
-	-	obfuscation	FromBase64String
-	-	network	System.Net.Sockets
-	-	network	TcpClient
-	-	network	NetworkStream
x	-	keyboard-and-mouse	GetKeyboardState
x	-	keyboard-and-mouse	MapVirtualKey
-	-	keyboard-and-mouse	GetKeyboardLayout
x	-	keyboard-and-mouse	GetAsyncKeyState
x	-	file	GetTempFileName
-	-	file	WriteAllBytes
-	-	execution	Sleep
x	-	execution	SetEnvironmentVariable
-	-	execution	GetCurrentProcess
x	-	execution	NtSetInformationProcess
x	-	execution	GetWindowThreadProcessId
x	-	cryptography	System.Security.Cryptography
-	-	-	(?u`
-	-	-	.text
-	-	-	`.rsrc
-	-	-	@.reloc
-	-	-	3)rY
-	-	-	j3wr
-	-	-	3 rK
-	-	-	BSJB
-	-	-	v2.0.50727
-	-	-	#Strings
-	-	-	#GUID
-	-	-	#Blob
-	-	-	<Module>
-	-	-	System.Runtime.CompilerServices
-	-	-	CompilationRelaxationsAttribute
-	-	-	.ctor
-	-	-	RuntimeCompatibilityAttribute

it | file-type: executable | subsystem: GUI | entry-point: 0x0000746E | signatu

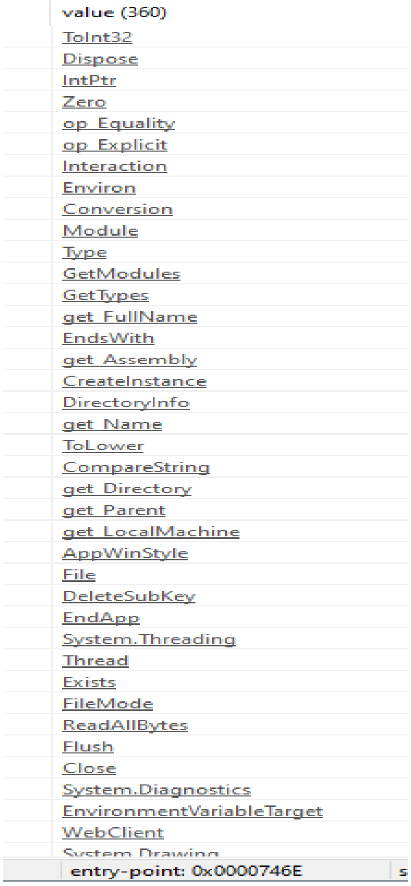
GetWindowText, GetWindowTextLength- Changes the text of the specified window's title bar.
GetVolumeInformation- Retrieves information about the file system and volume associated with the specified root directory.
CreateSubKey- adding sub key.
System.Net.Sockets, TcpClient, NetworkStream-Creating netork connections.
GetKeyboardState, MapVirtualKey, GetKeyboardLayout, GetAsyncKeyState- using keyboard and mouse state.
Sleep- has the ability to turn the computer to "sleep" mode.
SetEnvironmentVariable, GetCurrentProcess, NtSetInformationProcess, GetWindowThreadProcessId- intervention in processes.
System.Security.Cryptography- using cryptographic services, including secure encoding and decoding of data, as well as many other operations, such as hashing, random number generation, and message authentication.



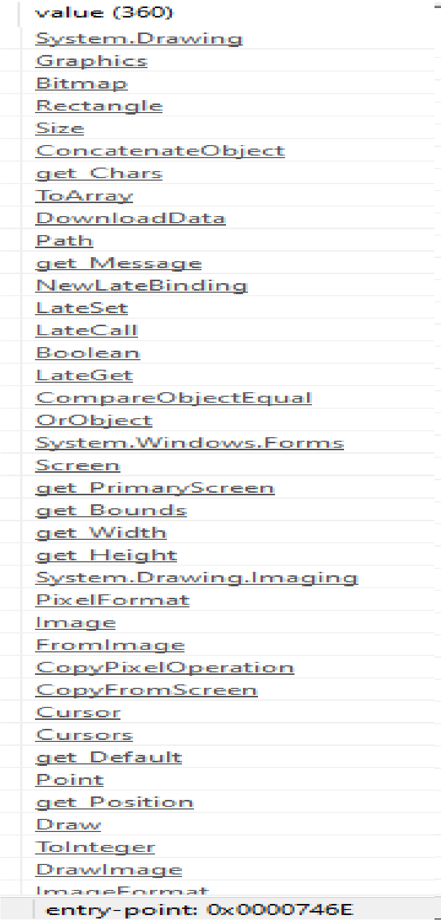
```
value (360)
RuntimeCompatibilityAttribute
System
Object
Microsoft.VisualBasic.CompilerServices
StandardModuleAttribute
System.IO
FileInfo
FileStream
Microsoft.VisualBasic.Devices
Computer
MemoryStream
Conversions
ToBoolean
System.Reflection
Assembly
GetEntryAssembly
get_Location
Byte
Exception
Microsoft.VisualBasic.MyServices
RegistryProxy
ServerComputer
get_Registry
Microsoft.Win32
get_CurrentUser
String
Concat
OpenSubKey
DeleteValue
ProjectData
SetProjectError
ClearProjectError
RuntimeHelpers
GetObjectValue
GetValue
RegistryValueKind
SetValue
DateTime
Operators
entry-point: 0x0000746E          signature: M
```



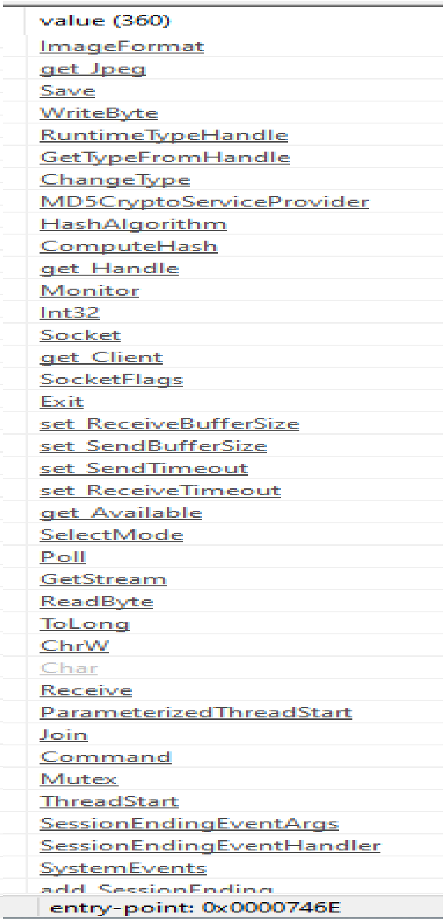
```
value (360)
Operators
ConditionalCompareObjectEqual
ToString
Environment
get_MachineName
get_UserName
FileSystemInfo
get_LastWriteTime
get_Date
ComputerInfo
get_Info
get_OSFullName
OperatingSystem
get_OSVersion
get_ServicePack
Microsoft.VisualBasic
Strings
CompareMethod
Split
SpecialFolder
GetFolderPath
Contains
RegistryKeyPermissionCheck
GetValueNames
get_Length
Convert
System.Text
Encoding
get_UTF8
GetBytes
GetString
System.IO.Compression
GZipStream
Stream
CompressionMode
set_Position
Read
BitConverter
ToInt32
entry-point: 0x0000746E          signa
```



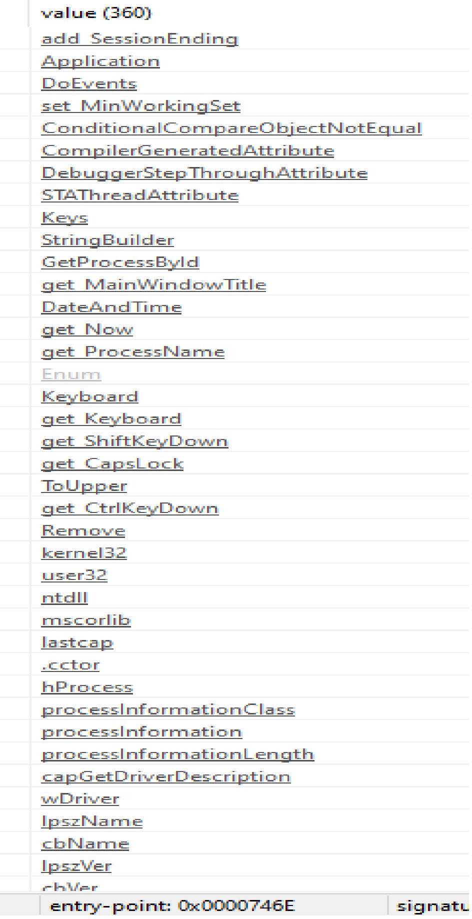
```
value (360)
ToInt32
Dispose
IntPtr
Zero
op_Equality
op_Explicit
Interaction
Environ
Conversion
Module
Type
GetModules
GetTypes
get_FullName
EndsWith
get_Assembly
CreateInstance
DirectoryInfo
get_Name
ToLower
CompareString
get_Directory
get_Parent
get_LocalMachine
AppWinStyle
File
DeleteSubKey
EndApp
System.Threading
Thread
Exists
FileMode
ReadAllBytes
Flush
Close
System.Diagnostics
EnvironmentVariableTarget
WebClient
System.Drawing
entry-point: 0x0000746E          s
```



```
value (360)
System.Drawing
Graphics
Bitmap
Rectangle
Size
ConcatenateObject
get_Chars
ToArray
DownloadData
Path
get_Message
NewLateBinding
LateSet
LateCall
Boolean
LateGet
CompareObjectEqual
OrObject
System.Windows.Forms
Screen
get_PrimaryScreen
get_Bounds
get_Width
get_Height
System.Drawing.Imaging
PixelFormat
Image
FromImage
CopyPixelOperation
CopyFromScreen
Cursor
Cursors
get_Default
Point
get_Position
Draw
ToInteger
DrawImage
ImageFormat
entry-point: 0x0000746E
```

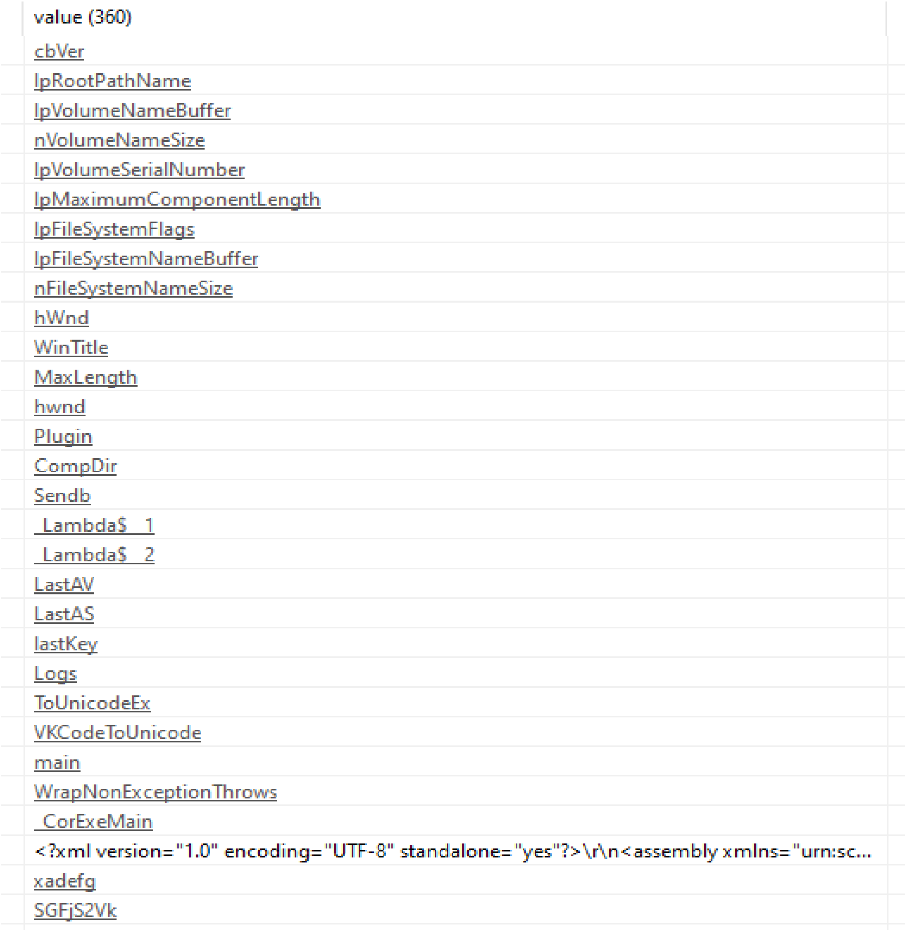


```
value (360)
ImageFormat
get_Jpeg
Save
WriteByte
RuntimeTypeHandle
GetTypeFromHandle
ChangeType
MD5CryptoServiceProvider
HashAlgorithm
ComputeHash
get_Handle
Monitor
Int32
Socket
get_Client
SocketFlags
Exit
set_ReceiveBufferSize
set_SendBufferSize
set_SendTimeout
set_ReceiveTimeout
get_Available
SelectMode
Poll
GetStream
ReadByte
ToLong
ChrW
Char
Receive
ParameterizedThreadStart
Join
Command
Mutex
ThreadStart
SessionEndingEventArgs
SessionEndingEventHandler
SystemEvents
add_SessionEnding
entry-point: 0x0000746E
```



```
value (360)
add_SessionEnding
Application
DoEvents
set_MinWorkingSet
ConditionalCompareObjectNotEqual
CompilerGeneratedAttribute
DebuggerStepThroughAttribute
STAThreadAttribute
Keys
StringBuilder
GetProcessById
get_MainWindowTitle
DateAndTime
get_Now
get_ProcessName
Enum
Keyboard
get_Keyboard
get_ShiftKeyDown
get_CapsLock
ToUpper
get_CtrlKeyDown
Remove
kernel32
user32
ntdll
mscorlib
lastcap
.cctor
hProcess
processInformationClass
processInformation
processInformationLength
capGetDriverDescription
wDriver
lpszName
cbName
lpszVer
cbVer
entry-point: 0x0000746E          signatu
```

AVIV SKITAL



```
value (360)
cbVer
lpRootPathName
lpVolumeNameBuffer
nVolumeNameSize
lpVolumeSerialNumber
lpMaximumComponentLength
lpFileSystemFlags
lpFileSystemNameBuffer
nFileSystemNameSize
hWnd
WinTitle
MaxLength
hwnd
Plugin
CompDir
Sendb
_Lambda$__1
_Lambda$__2
LastAV
LastAS
lastKey
Logs
ToUnicodeEx
VKCodeToUnicode
main
WrapNonExceptionThrows
_CorExeMain
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>\r\n<assembly xmlns="urn:sc...
xadefg
SGFjS2Vk
0.7d
TEMP
8f3067438cc6bd847dbef75384743d67
3333
|'|'|
True
yy-MM-dd
??-??-??
Microsoft
entry-point: 0x0000746E    signature: Microsoft V
```

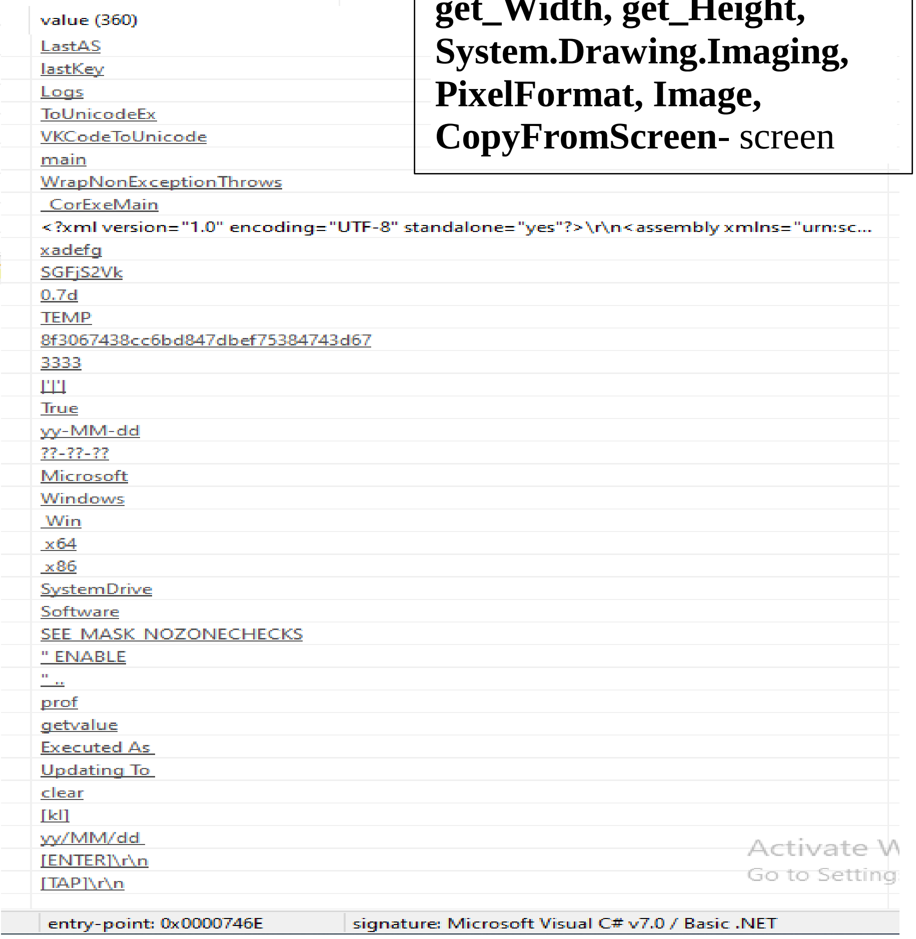
get_MachineName, get_UserName, FileSystemInfo, get_LastWriteTime, ComputerInfo- Collect information about the infected pc.

get_OSFullName, OperatingSystem, get_OSVersion- Collect information about the OS type+version, computer version+ name.

DirectoryInfo, get_FullName, get_Name, get_Parent- collect information from the directories.

CompareString- maybe try hashing. to decipher

get_PrimaryScreen, get_Width, get_Height, System.Drawing.Imaging, PixelFormat, Image, CopyFromScreen- screen



```
value (360)
LastAS
lastKey
Logs
ToUnicodeEx
VKCodeToUnicode
main
WrapNonExceptionThrows
_CorExeMain
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>\r\n<assembly xmlns="urn:sc...
xadefg
SGFjS2Vk
0.7d
TEMP
8f3067438cc6bd847dbef75384743d67
3333
|'|'|
True
yy-MM-dd
??-??-??
Microsoft
Windows
Win
x64
x86
SystemDrive
Software
SEE_MASK_NOZONECHECKS
" ENABLE
" ..
prof
getvalue
Executed As 
Updating To 
clear
[kl]
yy/MM/dd 
[ENTER]\r\n
[TAP]\r\n
entry-point: 0x0000746E    signature: Microsoft Visual C# v7.0 / Basic .NET
```

Keyboard, get_Keyboard, get_ShiftKeyDown, get_CapsLock, ToUpper, get_CtrlKeyDown- checking keyboard station (clicks)

3333- Listening on port 3333.

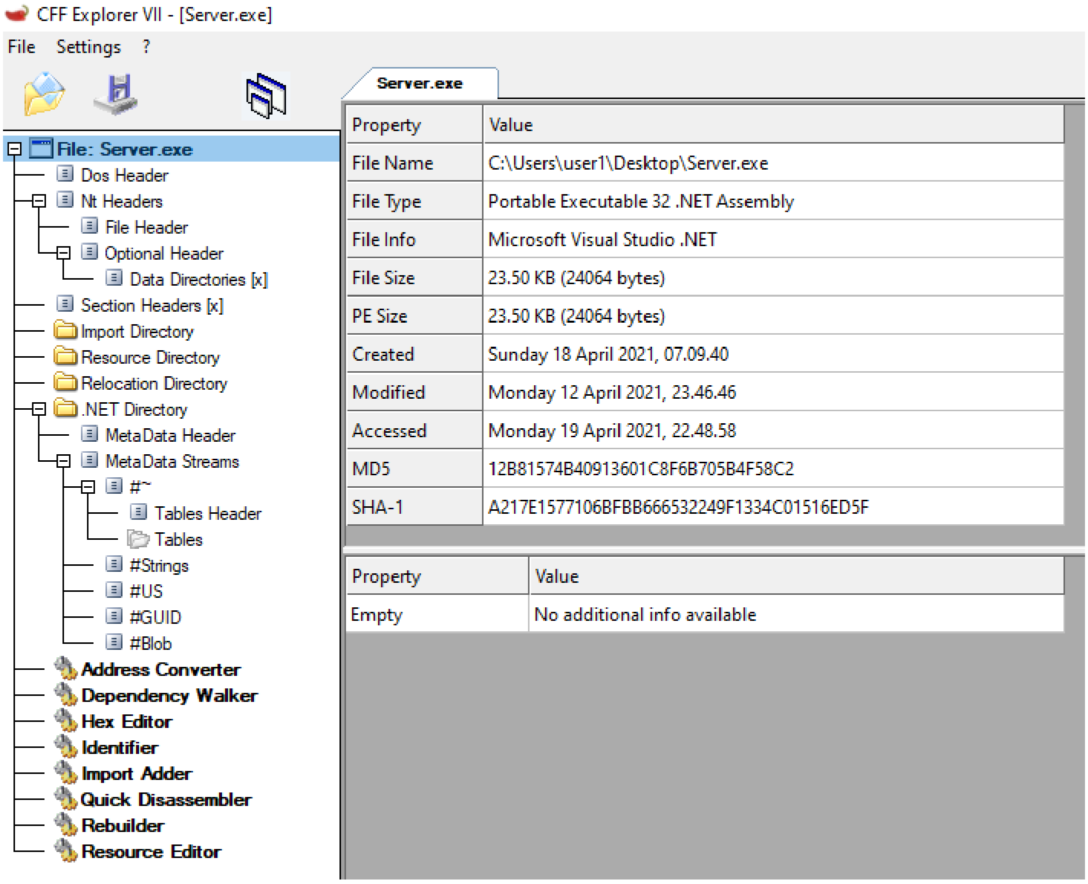
TEMP folder- explanation in the dynamic analysis.

get_Date, yy-mm-dd, yy/mm/dd- date templates, collect the date information.

The strings are clearly readable, the words are written in clear text - which confirms that the code is not obfuscated.

Another tool that can provide some information about the malware file is –

CFF explorer - PE editor tool.



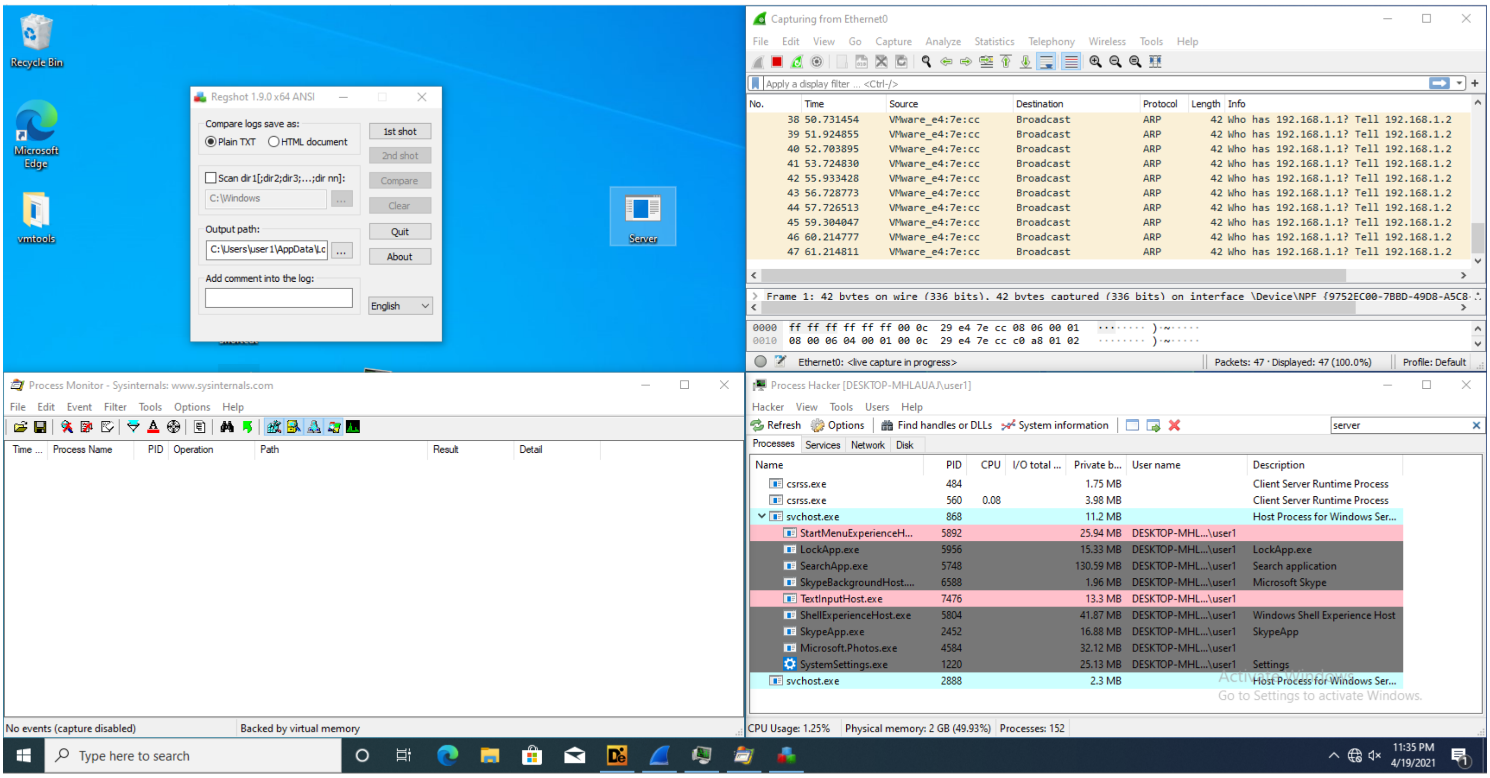
Property	Value
File Name	C:\Users\user1\Desktop\Server.exe
File Type	Portable Executable 32 .NET Assembly
File Info	Microsoft Visual Studio .NET
File Size	23.50 KB (24064 bytes)
PE Size	23.50 KB (24064 bytes)
Created	Sunday 18 April 2021, 07.09.40
Modified	Monday 12 April 2021, 23.46.46
Accessed	Monday 19 April 2021, 22.48.58
MD5	12B81574B40913601C8F6B705B4F58C2
SHA-1	A217E1577106BFBB666532249F1334C01516ED5F

Property	Value
Empty	No additional info available

Dynamic Analysis

Dynamic analysis is any examination performed after executing malware. Dynamic analysis techniques are the second step in the malware analysis process. Dynamic analysis is typically performed after basic static analysis has reached to the end. It involves monitoring malware as it runs and examining the system after the malware has executed.

Prior to running the malware, the following tools need to be opened and ready for use so that during the malware's activity I will be able to capture its activity in the background.



Regshot 1.9.0 x64 ANSI

Compare logs save as: ● Plain TXT ○ HTML document

□ Scan dir1[;dir2;dir3;...;dir nn]: C:\Windows

Output path: C:\Users\user1\AppData\Lc

Add comment into the log:

English

Capturing from Ethernet0

```
No. Time        Source             Destination  Protocol Length Info
 38 50.731454   VMware_e4:7e:cc    Broadcast    ARP      42 Who has 192.168.1.1? Tell 192.168.1.2
 39 51.924855   VMware_e4:7e:cc    Broadcast    ARP      42 Who has 192.168.1.1? Tell 192.168.1.2
 40 52.703895   VMware_e4:7e:cc    Broadcast    ARP      42 Who has 192.168.1.1? Tell 192.168.1.2
 41 53.724830   VMware_e4:7e:cc    Broadcast    ARP      42 Who has 192.168.1.1? Tell 192.168.1.2
 42 55.933428   VMware_e4:7e:cc    Broadcast    ARP      42 Who has 192.168.1.1? Tell 192.168.1.2
 43 56.728773   VMware_e4:7e:cc    Broadcast    ARP      42 Who has 192.168.1.1? Tell 192.168.1.2
 44 57.726513   VMware_e4:7e:cc    Broadcast    ARP      42 Who has 192.168.1.1? Tell 192.168.1.2
 45 59.304047   VMware_e4:7e:cc    Broadcast    ARP      42 Who has 192.168.1.1? Tell 192.168.1.2
 46 60.214777   VMware_e4:7e:cc    Broadcast    ARP      42 Who has 192.168.1.1? Tell 192.168.1.2
 47 61.214811   VMware_e4:7e:cc    Broadcast    ARP      42 Who has 192.168.1.1? Tell 192.168.1.2

> Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface \Device\NPF_{9752EC00-7BBD-49D8-A5C8-

0000  ff ff ff ff ff ff 00 0c  29 e4 7e cc 08 06 00 01   ········ )·~·····
0010  08 00 06 04 00 01 00 0c  29 e4 7e cc c0 a8 01 02   ········ )·~·····

Ethernet0: <live capture in progress>    Packets: 47 · Displayed: 47 (100.0%)    Profile: Default
```

Process Monitor - Sysinternals: www.sysinternals.com

Time ...	Process Name	PID	Operation	Path	Result	Detail

No events (capture disabled) Backed by virtual memory

Process Hacker [DESKTOP-MHLAUAJ\user1]

Filter: server

Name	PID	CPU	I/O total ...	Private b...	User name	Description
csrss.exe	484			1.75 MB		Client Server Runtime Process
csrss.exe	560	0.08		3.98 MB		Client Server Runtime Process
svchost.exe	868			11.2 MB		Host Process for Windows Ser...
StartMenuExperienceH...	5892			25.94 MB	DESKTOP-MHL...\user1	
LockApp.exe	5956			15.33 MB	DESKTOP-MHL...\user1	LockApp.exe
SearchApp.exe	5748			130.59 MB	DESKTOP-MHL...\user1	Search application
SkypeBackgroundHost....	6588			1.96 MB	DESKTOP-MHL...\user1	Microsoft Skype
TextInputHost.exe	7476			13.3 MB	DESKTOP-MHL...\user1	
ShellExperienceHost.exe	5804			41.87 MB	DESKTOP-MHL...\user1	Windows Shell Experience Host
SkypeApp.exe	2452			16.88 MB	DESKTOP-MHL...\user1	SkypeApp
Microsoft.Photos.exe	4584			32.12 MB	DESKTOP-MHL...\user1	
SystemSettings.exe	1220			25.13 MB	DESKTOP-MHL...\user1	Settings
svchost.exe	2888			2.3 MB		Host Process for Windows Ser...

CPU Usage: 1.25% Physical memory: 2 GB (49.93%) Processes: 152

Recycle Bin
Microsoft Edge
vmtools
Server
1st shot
2nd shot
Compare
Clear
Quit
About
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
File Edit Event Filter Tools Options Help
Hacker View Tools Users Help
Refresh Options Find handles or DLLs System information
Processes Services Network Disk
Activate Windows
Go to Settings to activate Windows.
Type here to search
11:35 PM
4/19/2021

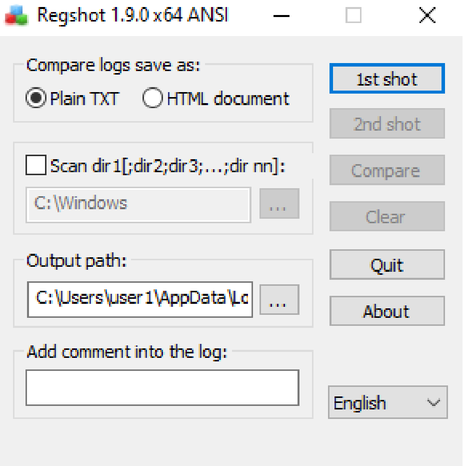
"Wireshark"- with Wireshark I capture the network traffic between the malware to the C&C server- on the attacker's machine.

"Process Hacker" - with this tool it is possible to see all the active processes that are running on the machine. For example, I can check whether the Server.exe file is running.
In the filter bar I wrote - "Server".

"Process Monitor" - with this tool it is possible to see the real-time file-system, registry and thread or process activity. I stopped and deleted all the running activities so that there are no any unnecessary processes running on the machine.

"Regshot" – this tool allows me to quickly take a snapshot of the machine's registry and then compare it with a second one, which is done after doing system changes.

At first, I will take a snapshot of the machine's registry before running the malware:



Compare logs save as: ● Plain TXT ○ HTML document

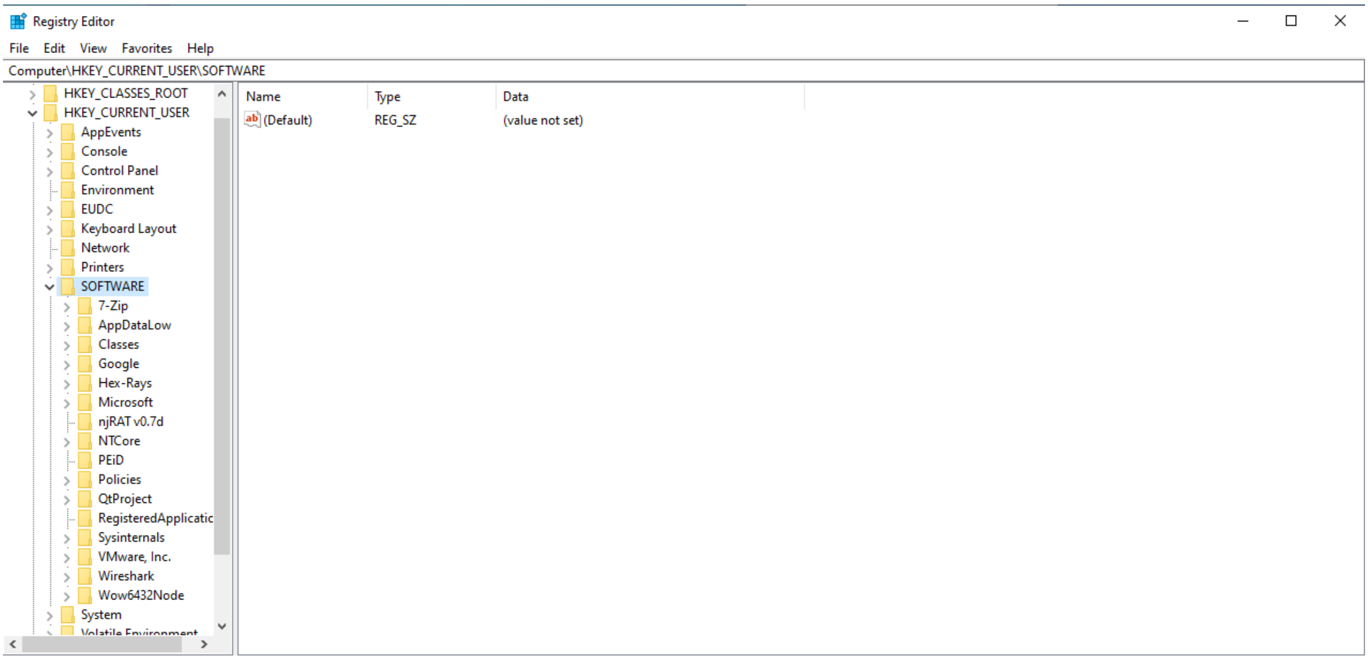
□ Scan dir1[;dir2;dir3;...;dir nn]: C:\Windows

Output path: C:\Users\user1\AppData\Lc

Add comment into the log:

English

secondly, I want to look at the Registry Editor before the execution:

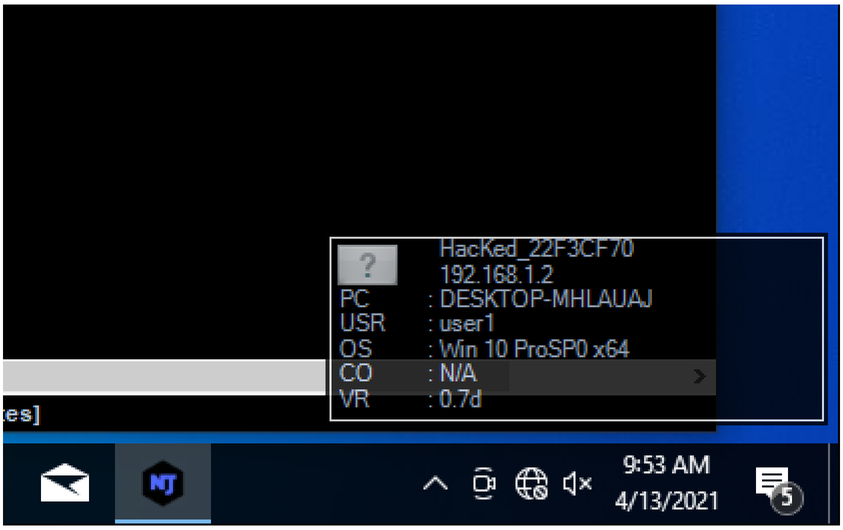


Computer\HKEY_CURRENT_USER\SOFTWARE

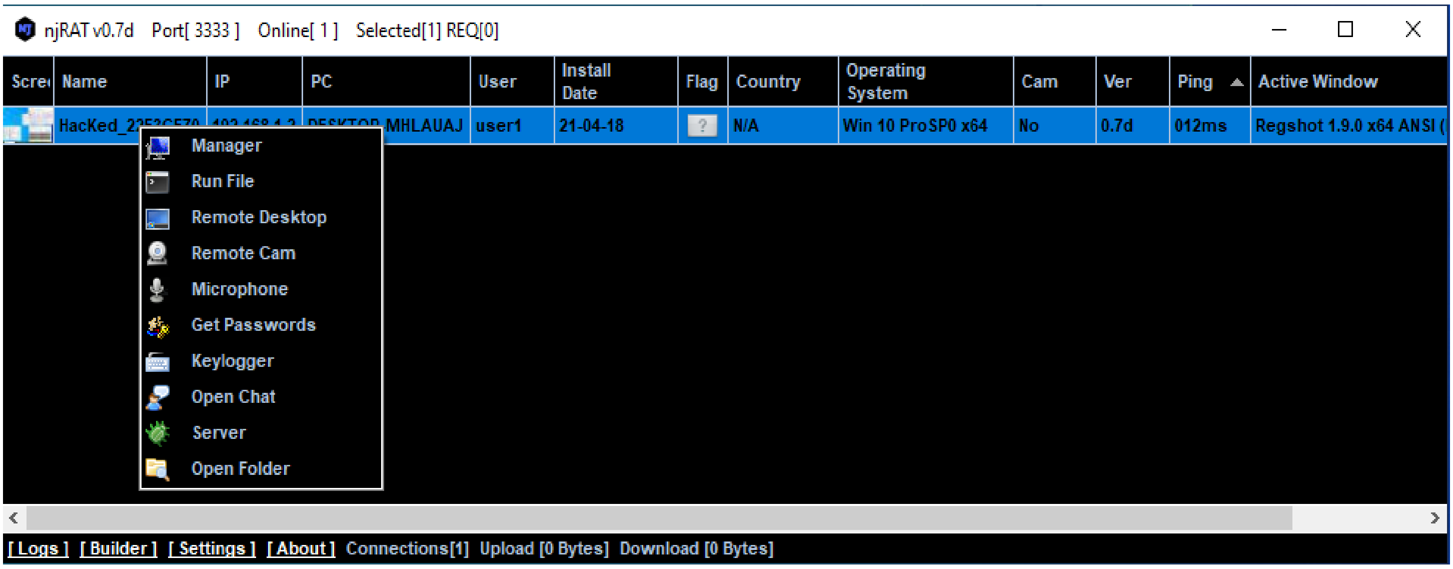
Name	Type	Data
(Default)	REG_SZ	(value not set)

Thirdly, I started monitoring the activities using "Process Monitor".

At this point, the machine is ready for the execution of the malware and all the necessary tools are up and running.
In order to execute the malware I double-clicked the "Server.exe" file. A short while afterwards an alert popped up on the attacker's machine and all the options were available:



HacKed_22F3CF70
192.168.1.2
PC : DESKTOP-MHLAUAJ
USR : user1
OS : Win 10 ProSP0 x64
CO : N/A
VR : 0.7d



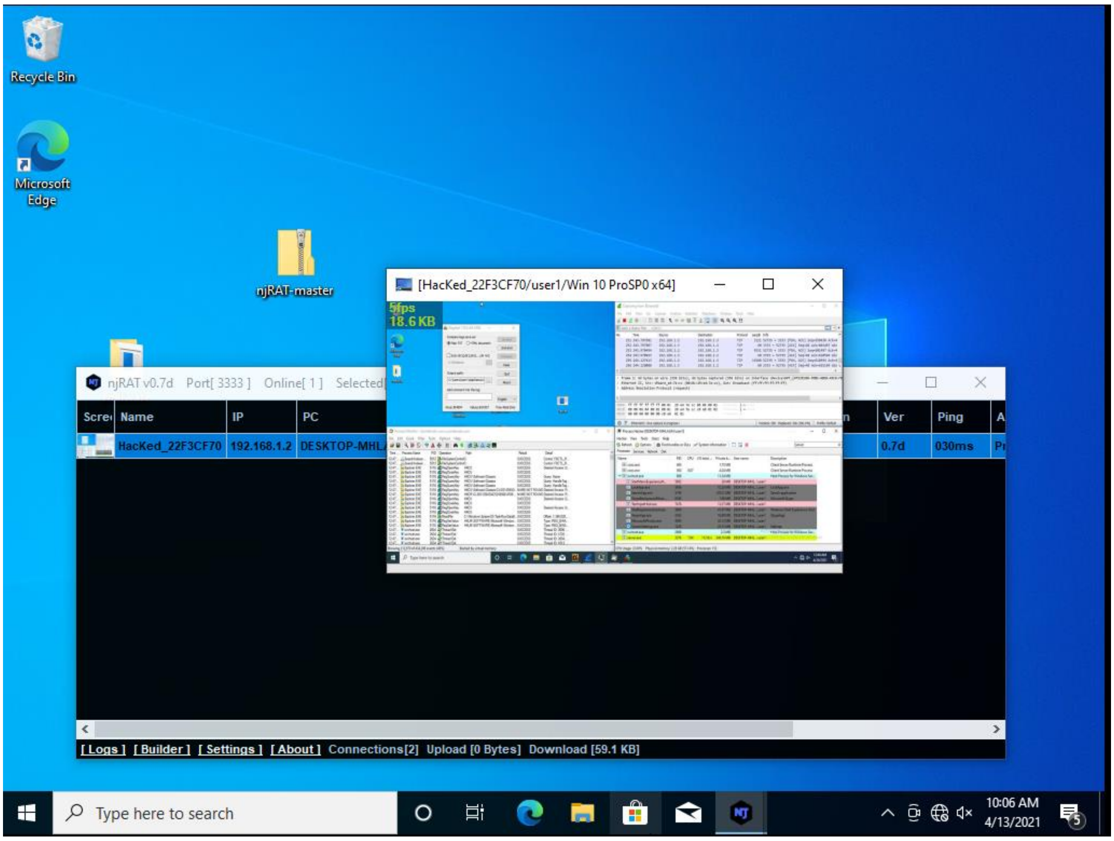
njRAT v0.7d Port[3333] Online[1] Selected[1] REQ[0]

Scre	Name	IP	PC	User	Install Date	Flag	Country	Operating System	Cam	Ver	Ping	Active Window
	HacKed_22F3CF70	192.168.1.2	DESKTOP-MHLAUAJ	user1	21-04-18	?	N/A	Win 10 ProSP0 x64	No	0.7d	012ms	Regshot 1.9.0 x64 ANSI (

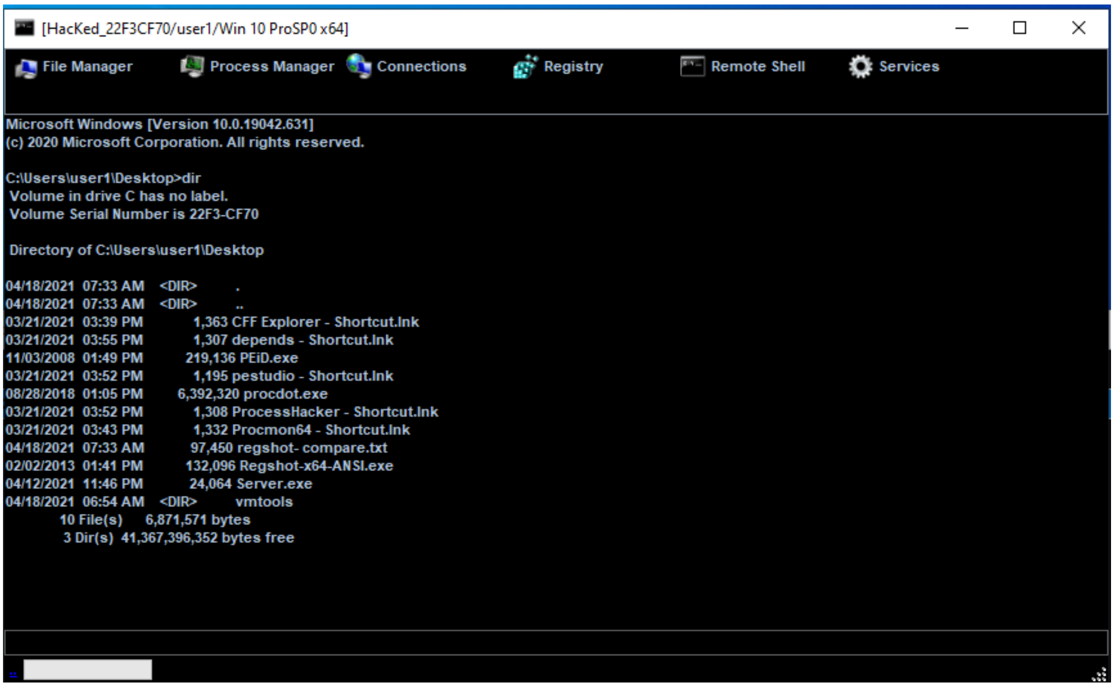
[Logs] [Builder] [Settings] [About] Connections[1] Upload [0 Bytes] Download [0 Bytes]

Examples:

Remote Desktop- gives me the picture of the victim's pc state.

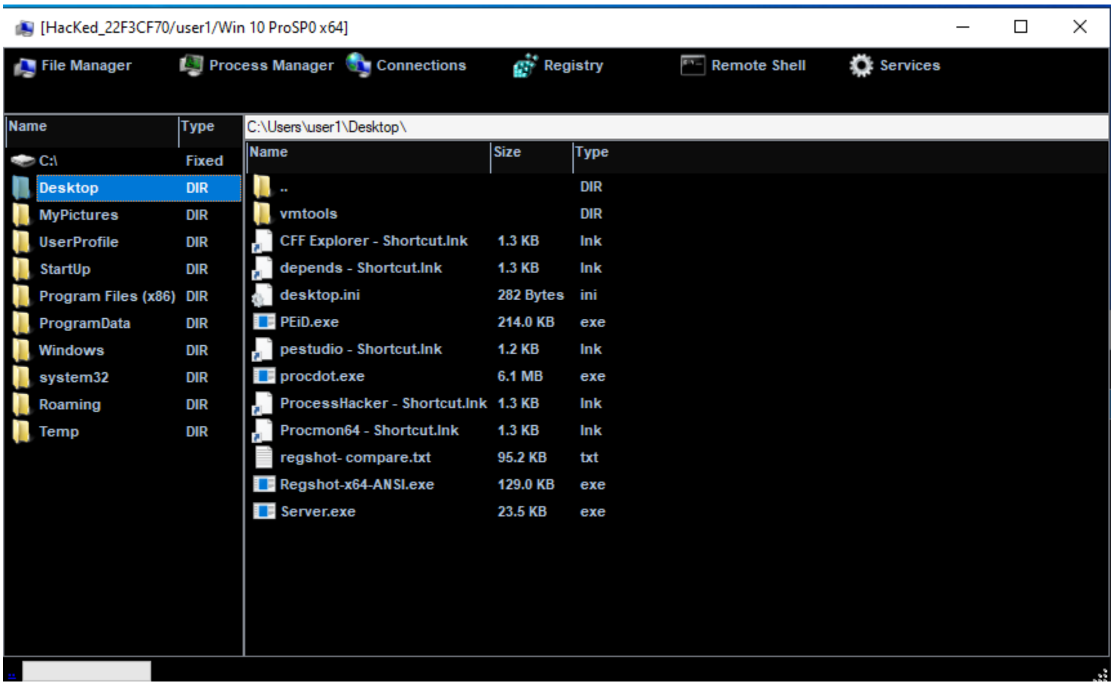

Recycle Bin
Microsoft Edge
njRAT-master
[HacKed_22F3CF70/user1/Win 10 ProSP0 x64]
5fps
18.6 KB
njRAT v0.7d Port[3333] Online[1] Selected
Scre Name IP PC Ver Ping A
HacKed_22F3CF70 192.168.1.2 DESKTOP-MHL 0.7d 030ms Pr
[Logs] [Builder] [Settings] [About] Connections[2] Upload [0 Bytes] Download [59.1 KB]
Type here to search
10:06 AM
4/13/2021

Remote Shell- launch a command shell interface for executing



```
Microsoft Windows [Version 10.0.19042.631]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\user1\Desktop>dir
 Volume in drive C has no label.
 Volume Serial Number is 22F3-CF70

 Directory of C:\Users\user1\Desktop

04/18/2021  07:33 AM    <DIR>          .
04/18/2021  07:33 AM    <DIR>          ..
03/21/2021  03:39 PM             1,363 CFF Explorer - Shortcut.lnk
03/21/2021  03:55 PM             1,307 depends - Shortcut.lnk
11/03/2008  01:49 PM           219,136 PEiD.exe
03/21/2021  03:52 PM             1,195 pestudio - Shortcut.lnk
08/28/2018  01:05 PM         6,392,320 procdot.exe
03/21/2021  03:52 PM             1,308 ProcessHacker - Shortcut.lnk
03/21/2021  03:43 PM             1,332 Procmon64 - Shortcut.lnk
04/18/2021  07:33 AM            97,450 regshot- compare.txt
02/02/2013  01:41 PM           132,096 Regshot-x64-ANSI.exe
04/12/2021  11:46 PM            24,064 Server.exe
04/18/2021  06:54 AM    <DIR>          vmtools
              10 File(s)      6,871,571 bytes
               3 Dir(s)  41,367,396,352 bytes free
```

File Manager- remotely execute and manipulate files

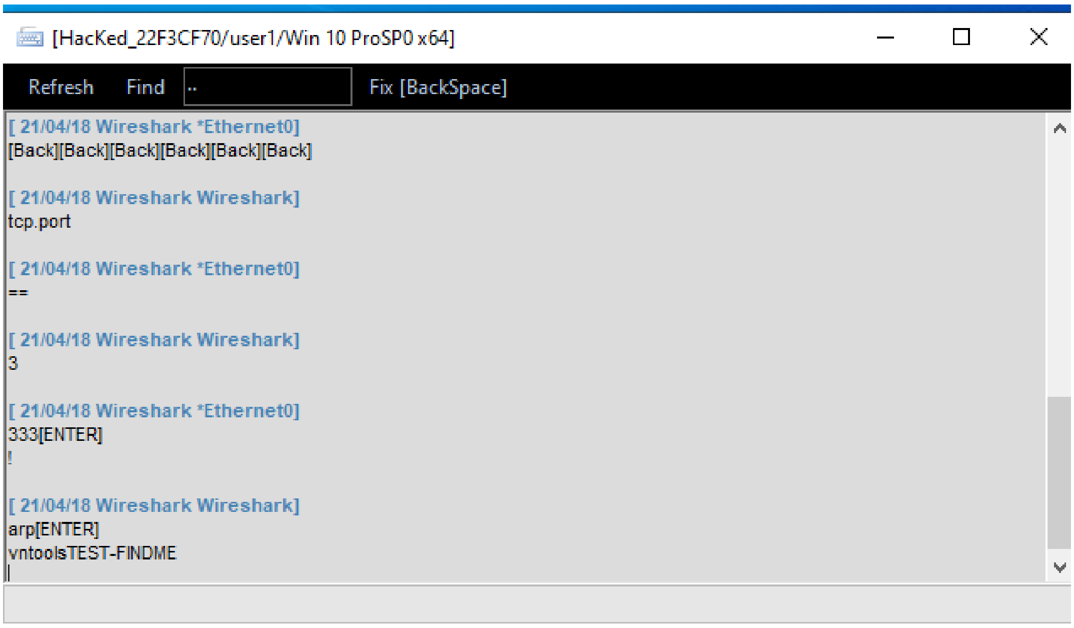


Name	Type
C:\	Fixed
Desktop	DIR
MyPictures	DIR
UserProfile	DIR
StartUp	DIR
Program Files (x86)	DIR
ProgramData	DIR
Windows	DIR
system32	DIR
Roaming	DIR
Temp	DIR

C:\Users\user1\Desktop\

Name	Size	Type
..		DIR
vmtools		DIR
CFF Explorer - Shortcut.lnk	1.3 KB	lnk
depends - Shortcut.lnk	1.3 KB	lnk
desktop.ini	282 Bytes	ini
PEiD.exe	214.0 KB	exe
pestudio - Shortcut.lnk	1.2 KB	lnk
procdot.exe	6.1 MB	exe
ProcessHacker - Shortcut.lnk	1.3 KB	lnk
Procmon64 - Shortcut.lnk	1.3 KB	lnk
regshot- compare.txt	95.2 KB	txt
Regshot-x64-ANSI.exe	129.0 KB	exe
Server.exe	23.5 KB	exe

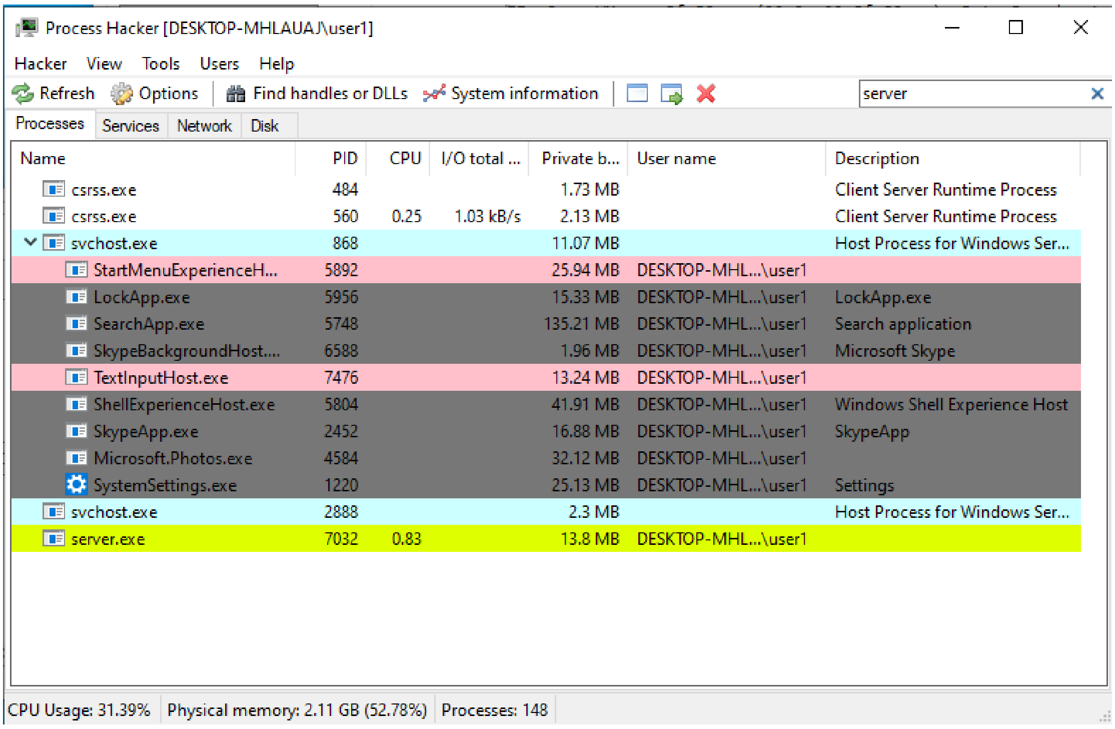
Keylogger- record keystrokes made by the victim



```
[ 21/04/18 Wireshark *Ethernet0]
[Back][Back][Back][Back][Back][Back]

[ 21/04/18 Wireshark Wireshark]
tcp.port

[ 21/04/18 Wireshark *Ethernet0]
==

[ 21/04/18 Wireshark Wireshark]
3

[ 21/04/18 Wireshark *Ethernet0]
333[ENTER]
!

[ 21/04/18 Wireshark Wireshark]
arp[ENTER]
vntoolsTEST-FINDME
|
```

After a few operations on the attacker's side, I returned to analyzing the software on the "victim's" machine:

On "Process Hacker" I can see that the "Server.exe" file is running:

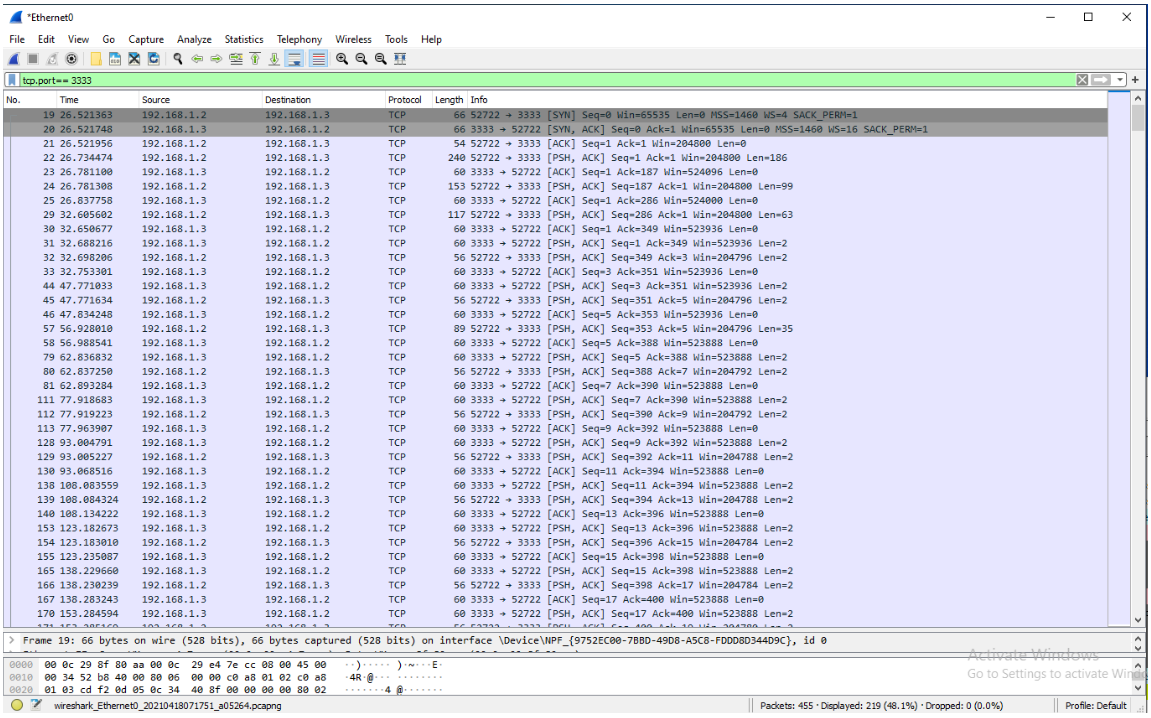


Process Hacker [DESKTOP-MHLAUAJ\user1]

Name	PID	CPU	I/O total ...	Private b...	User name	Description
csrss.exe	484			1.73 MB		Client Server Runtime Process
csrss.exe	560	0.25	1.03 kB/s	2.13 MB		Client Server Runtime Process
svchost.exe	868			11.07 MB		Host Process for Windows Ser...
StartMenuExperienceH...	5892			25.94 MB	DESKTOP-MHL...\user1	
LockApp.exe	5956			15.33 MB	DESKTOP-MHL...\user1	LockApp.exe
SearchApp.exe	5748			135.21 MB	DESKTOP-MHL...\user1	Search application
SkypeBackgroundHost....	6588			1.96 MB	DESKTOP-MHL...\user1	Microsoft Skype
TextInputHost.exe	7476			13.24 MB	DESKTOP-MHL...\user1	
ShellExperienceHost.exe	5804			41.91 MB	DESKTOP-MHL...\user1	Windows Shell Experience Host
SkypeApp.exe	2452			16.88 MB	DESKTOP-MHL...\user1	SkypeApp
Microsoft.Photos.exe	4584			32.12 MB	DESKTOP-MHL...\user1	
SystemSettings.exe	1220			25.13 MB	DESKTOP-MHL...\user1	Settings
svchost.exe	2888			2.3 MB		Host Process for Windows Ser...
server.exe	7032	0.83		13.8 MB	DESKTOP-MHL...\user1	

CPU Usage: 31.39% | Physical memory: 2.11 GB (52.78%) | Processes: 148

On "Wireshark" I filtered the traffic by using the "tcp.port==3333" display filter in order to present only the network traffic related to the malware:



*Ethernet0

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

tcp.port== 3333

```
No.  Time        Source       Destination  Protocol Length Info
  19 26.521363   192.168.1.2  192.168.1.3  TCP       66 52722 → 3333 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=4 SACK_PERM=1
  20 26.521748   192.168.1.3  192.168.1.2  TCP       66 3333 → 52722 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=16 SACK_PERM=1
  21 26.521956   192.168.1.2  192.168.1.3  TCP       54 52722 → 3333 [ACK] Seq=1 Ack=1 Win=204800 Len=0
  22 26.734474   192.168.1.2  192.168.1.3  TCP      240 52722 → 3333 [PSH, ACK] Seq=1 Ack=1 Win=204800 Len=186
  23 26.781100   192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [ACK] Seq=1 Ack=187 Win=524096 Len=0
  24 26.781308   192.168.1.2  192.168.1.3  TCP      153 52722 → 3333 [PSH, ACK] Seq=187 Ack=1 Win=204800 Len=99
  25 26.837758   192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [ACK] Seq=1 Ack=286 Win=524000 Len=0
  29 32.605602   192.168.1.2  192.168.1.3  TCP      117 52722 → 3333 [PSH, ACK] Seq=286 Ack=1 Win=204800 Len=63
  30 32.650677   192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [ACK] Seq=1 Ack=349 Win=523936 Len=0
  31 32.688216   192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [PSH, ACK] Seq=1 Ack=349 Win=523936 Len=2
  32 32.698206   192.168.1.2  192.168.1.3  TCP       56 52722 → 3333 [PSH, ACK] Seq=349 Ack=3 Win=204796 Len=2
  33 32.753301   192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [ACK] Seq=3 Ack=351 Win=523936 Len=0
  44 47.771033   192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [PSH, ACK] Seq=3 Ack=351 Win=523936 Len=2
  45 47.771634   192.168.1.2  192.168.1.3  TCP       56 52722 → 3333 [PSH, ACK] Seq=351 Ack=5 Win=204796 Len=2
  46 47.834248   192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [ACK] Seq=5 Ack=353 Win=523936 Len=0
  57 56.928010   192.168.1.2  192.168.1.3  TCP       89 52722 → 3333 [PSH, ACK] Seq=353 Ack=5 Win=204796 Len=35
  58 56.988541   192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [ACK] Seq=5 Ack=388 Win=523888 Len=0
  79 62.836832   192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [PSH, ACK] Seq=5 Ack=388 Win=523888 Len=2
  80 62.837250   192.168.1.2  192.168.1.3  TCP       56 52722 → 3333 [PSH, ACK] Seq=388 Ack=7 Win=204792 Len=2
  81 62.893284   192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [ACK] Seq=7 Ack=390 Win=523888 Len=0
 111 77.918683   192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [PSH, ACK] Seq=7 Ack=390 Win=523888 Len=2
 112 77.919223   192.168.1.2  192.168.1.3  TCP       56 52722 → 3333 [PSH, ACK] Seq=390 Ack=9 Win=204792 Len=2
 113 77.963907   192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [ACK] Seq=9 Ack=392 Win=523888 Len=0
 128 93.004791   192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [PSH, ACK] Seq=9 Ack=392 Win=523888 Len=2
 129 93.005227   192.168.1.2  192.168.1.3  TCP       56 52722 → 3333 [PSH, ACK] Seq=392 Ack=11 Win=204788 Len=2
 130 93.068516   192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [ACK] Seq=11 Ack=394 Win=523888 Len=0
 138 108.083559  192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [PSH, ACK] Seq=11 Ack=394 Win=523888 Len=2
 139 108.084324  192.168.1.2  192.168.1.3  TCP       56 52722 → 3333 [PSH, ACK] Seq=394 Ack=13 Win=204788 Len=2
 140 108.134222  192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [ACK] Seq=13 Ack=396 Win=523888 Len=0
 153 123.182673  192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [PSH, ACK] Seq=13 Ack=396 Win=523888 Len=2
 154 123.183010  192.168.1.2  192.168.1.3  TCP       56 52722 → 3333 [PSH, ACK] Seq=396 Ack=15 Win=204784 Len=2
 155 123.235087  192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [ACK] Seq=15 Ack=398 Win=523888 Len=0
 165 138.229660  192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [PSH, ACK] Seq=15 Ack=398 Win=523888 Len=2
 166 138.230239  192.168.1.2  192.168.1.3  TCP       56 52722 → 3333 [PSH, ACK] Seq=398 Ack=17 Win=204784 Len=2
 167 138.283243  192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [ACK] Seq=17 Ack=400 Win=523888 Len=0
 170 153.284594  192.168.1.3  192.168.1.2  TCP       60 3333 → 52722 [PSH, ACK] Seq=17 Ack=400 Win=523888 Len=2
```

```
> Frame 19: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface \Device\NPF_{9752EC00-7BBD-49D8-A5C8-FDDD8D344D9C}, id 0
```

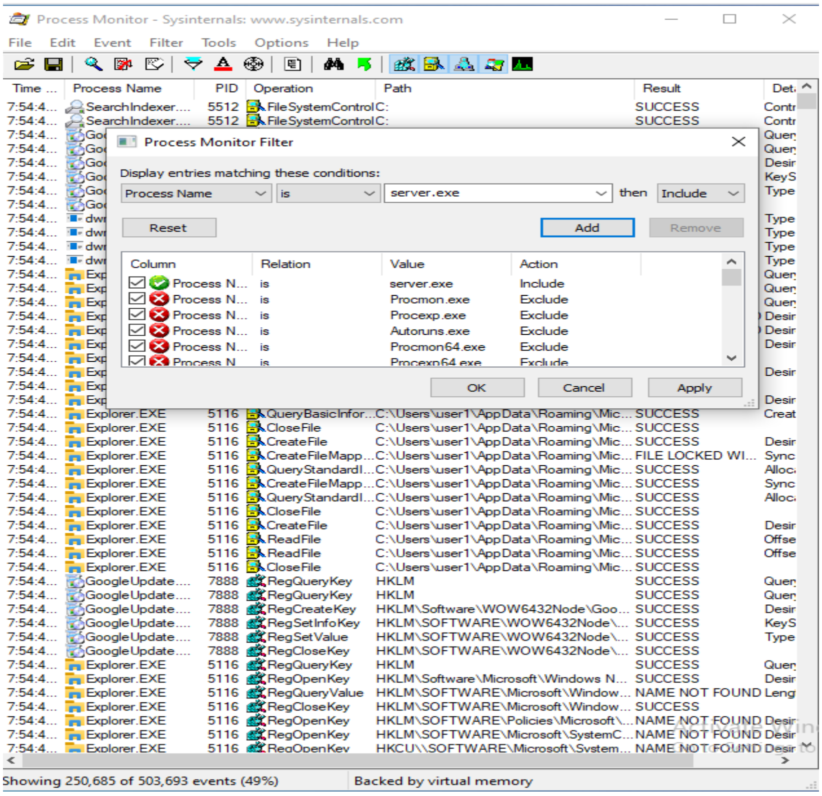
```
0000  00 0c 29 8f 80 aa 00 0c  29 e4 7e cc 08 00 45 00   ··)····· )·~···E·
0010  00 34 52 b8 40 00 80 06  00 00 c0 a8 01 02 c0 a8   ·4R·@··· ········
0020  01 03 cd f2 0d 05 0c 34  40 8f 00 00 00 00 80 02   ·······4 @·······
```

wireshark_Ethernet0_20210418071751_a05264.pcapng | Packets: 455 · Displayed: 219 (48.1%) · Dropped: 0 (0.0%) | Profile: Default

This picture (or screenshot if you took a screenshot \ snapshot) presents the TCP connection between the malware ("victim's" machine - 192.168.1.2) and the C&C server (attacker's machine - 192.168.1.3). The first 3 packets are the TCP 3-way-handshake and all the other packets are transformation of data \ TCP stream between the two machines..

Since the victim and the attacker's machines are on the same network, the malware doesn't try to go out on the internet and connect to its C&C server.
However, when the incident response team are dealing with an unknown malware, they will investigate the malware to its source.
In this situation, it is extremely important to capture the malware's network traffic. Based on the malware's network traffic behavior it will be possible to retrieve important information such as: the attacker's IP address, DNS queries made by the malware, what information it sends to its server and more.

In "Process Monitor" I filtered all the events in order to see only the events that are linked to the "Server.exe" process.
I added the process name – "Server.exe", I presses the "add" button, the filter was added to the list, I pressed "apply" and "ok"


Process Monitor - Sysinternals: www.sysinternals.com
File Edit Event Filter Tools Options Help

Process Monitor Filter

Display entries matching these conditions:

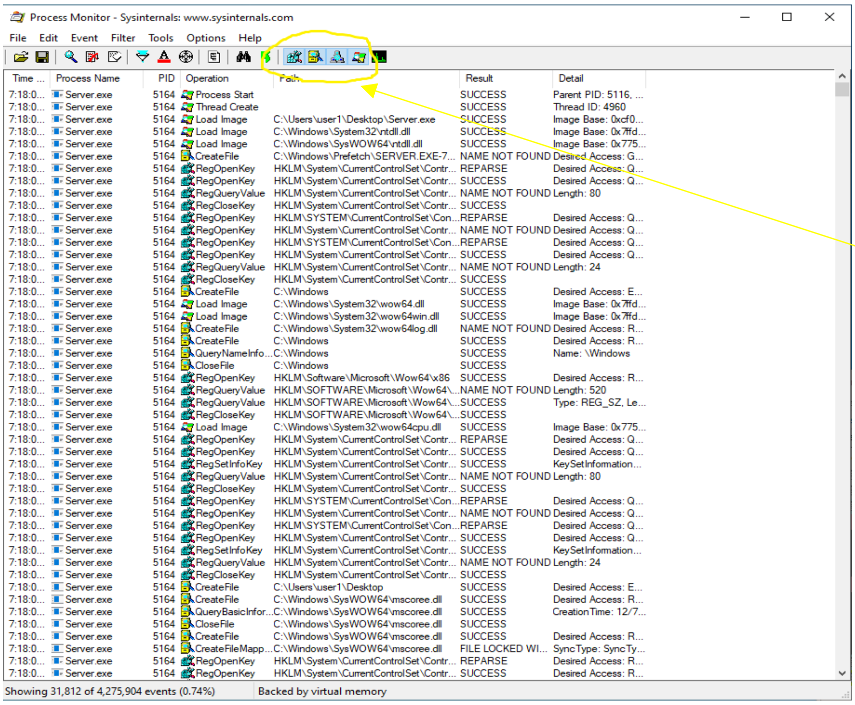
Process Name | is | server.exe | then | Include

Column	Relation	Value	Action
☑ Process N...	is	server.exe	Include
☑ Process N...	is	Procmon.exe	Exclude
☑ Process N...	is	Procexp.exe	Exclude
☑ Process N...	is	Autoruns.exe	Exclude
☑ Process N...	is	Procmon64.exe	Exclude
☑ Process N...	is	Procexp64.exe	Exclude

```
Time ...   Process Name     PID   Operation          Path                                        Result             Det:
7:54:4...  SearchIndexer....  5512  FileSystemControl C:                                          SUCCESS            Contr
7:54:4...  SearchIndexer....  5512  FileSystemControl C:                                          SUCCESS            Contr
7:54:4...  Goo                                                                                                       Quer
7:54:4...  Goo                                                                                                       Quer
7:54:4...  Goo                                                                                                       Desir
7:54:4...  Goo                                                                                                       KeyS
7:54:4...  Goo                                                                                                       Type
7:54:4...  Goo
7:54:4...  dwi                                                                                                       Type
7:54:4...  dwi                                                                                                       Type
7:54:4...  dwi                                                                                                       Type
7:54:4...  dwi                                                                                                       Type
7:54:4...  Exp                                                                                                       Quer
7:54:4...  Exp                                                                                                       Quer
7:54:4...  Exp                                                                                                       Quer
7:54:4...  Exp                                                                                                       Desir
7:54:4...  Exp                                                                                                       Desir
7:54:4...  Exp                                                                                                       Desir
7:54:4...  Exp
7:54:4...  Exp                                                                                                       Desir
7:54:4...  Exp
7:54:4...  Exp                                                                                                       Desir
7:54:4...  Explorer.EXE     5116  QueryBasicInfor...C:\Users\user1\AppData\Roaming\Mic... SUCCESS            Creat
7:54:4...  Explorer.EXE     5116  CloseFile          C:\Users\user1\AppData\Roaming\Mic... SUCCESS
7:54:4...  Explorer.EXE     5116  CreateFile         C:\Users\user1\AppData\Roaming\Mic... SUCCESS            Desir
7:54:4...  Explorer.EXE     5116  CreateFileMapp...C:\Users\user1\AppData\Roaming\Mic... FILE LOCKED WI... Sync
7:54:4...  Explorer.EXE     5116  QueryStandardI...C:\Users\user1\AppData\Roaming\Mic... SUCCESS            Alloc
7:54:4...  Explorer.EXE     5116  CreateFileMapp...C:\Users\user1\AppData\Roaming\Mic... SUCCESS            Sync
7:54:4...  Explorer.EXE     5116  QueryStandardI...C:\Users\user1\AppData\Roaming\Mic... SUCCESS            Alloc
7:54:4...  Explorer.EXE     5116  CloseFile          C:\Users\user1\AppData\Roaming\Mic... SUCCESS
7:54:4...  Explorer.EXE     5116  CreateFile         C:\Users\user1\AppData\Roaming\Mic... SUCCESS            Desir
7:54:4...  Explorer.EXE     5116  ReadFile           C:\Users\user1\AppData\Roaming\Mic... SUCCESS            Offse
7:54:4...  Explorer.EXE     5116  ReadFile           C:\Users\user1\AppData\Roaming\Mic... SUCCESS            Offse
7:54:4...  Explorer.EXE     5116  CloseFile          C:\Users\user1\AppData\Roaming\Mic... SUCCESS
7:54:4...  GoogleUpdate....  7888  RegQueryKey        HKLM                                        SUCCESS            Quer
7:54:4...  GoogleUpdate....  7888  RegQueryKey        HKLM                                        SUCCESS            Quer
7:54:4...  GoogleUpdate....  7888  RegCreateKey       HKLM\Software\WOW6432Node\Goo...            SUCCESS            Desir
7:54:4...  GoogleUpdate....  7888  RegSetInfoKey      HKLM\SOFTWARE\WOW6432Node\...               SUCCESS            KeyS
7:54:4...  GoogleUpdate....  7888  RegSetValue        HKLM\SOFTWARE\WOW6432Node\...               SUCCESS            Type
7:54:4...  GoogleUpdate....  7888  RegCloseKey        HKLM\SOFTWARE\WOW6432Node\...               SUCCESS
7:54:4...  Explorer.EXE     5116  RegQueryKey        HKLM                                        SUCCESS            Quer
7:54:4...  Explorer.EXE     5116  RegOpenKey         HKLM\Software\Microsoft\Windows N...        SUCCESS            Desir
7:54:4...  Explorer.EXE     5116  RegQueryValue      HKLM\SOFTWARE\Microsoft\Window...           NAME NOT FOUND     Leng
7:54:4...  Explorer.EXE     5116  RegCloseKey        HKLM\SOFTWARE\Microsoft\Window...           SUCCESS
7:54:4...  Explorer.EXE     5116  RegOpenKey         HKLM\SOFTWARE\Policies\Microsoft\...        NAME NOT FOUND     Desir
7:54:4...  Explorer.EXE     5116  RegOpenKey         HKLM\SOFTWARE\Microsoft\SystemC...          NAME NOT FOUND     Desir
7:54:4...  Explorer.EXE     5116  RegOpenKey         HKCU\\SOFTWARE\Microsoft\System...          NAME NOT FOUND     Desir
```

Showing 250,685 of 503,693 events (49%) | Backed by virtual memory

Process Monitor- with server.exe filter



Time ...	Process Name	PID	Operation	Path	Result	Detail
7:18:0...	Server.exe	5164	Process Start		SUCCESS	Parent PID: 5116, ...
7:18:0...	Server.exe	5164	Thread Create		SUCCESS	Thread ID: 4960
7:18:0...	Server.exe	5164	Load Image	C:\Users\user1\Desktop\Server.exe	SUCCESS	Image Base: 0xcf0...
7:18:0...	Server.exe	5164	Load Image	C:\Windows\System32\ntdll.dll	SUCCESS	Image Base: 0x7ffd...
7:18:0...	Server.exe	5164	Load Image	C:\Windows\SysWOW64\ntdll.dll	SUCCESS	Image Base: 0x775...
7:18:0...	Server.exe	5164	CreateFile	C:\Windows\Prefetch\SERVER.EXE-7...	NAME NOT FOUND	Desired Access: G...
7:18:0...	Server.exe	5164	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	REPARSE	Desired Access: Q...
7:18:0...	Server.exe	5164	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	Desired Access: Q...
7:18:0...	Server.exe	5164	RegQueryValue	HKLM\System\CurrentControlSet\Contr...	NAME NOT FOUND	Length: 80
7:18:0...	Server.exe	5164	RegCloseKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	
7:18:0...	Server.exe	5164	RegOpenKey	HKLM\SYSTEM\CurrentControlSet\Con...	REPARSE	Desired Access: Q...
7:18:0...	Server.exe	5164	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	NAME NOT FOUND	Desired Access: Q...
7:18:0...	Server.exe	5164	RegOpenKey	HKLM\SYSTEM\CurrentControlSet\Con...	REPARSE	Desired Access: Q...
7:18:0...	Server.exe	5164	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	Desired Access: Q...
7:18:0...	Server.exe	5164	RegQueryValue	HKLM\System\CurrentControlSet\Contr...	NAME NOT FOUND	Length: 24
7:18:0...	Server.exe	5164	RegCloseKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	
7:18:0...	Server.exe	5164	CreateFile	C:\Windows	SUCCESS	Desired Access: E...
7:18:0...	Server.exe	5164	Load Image	C:\Windows\System32\wow64.dll	SUCCESS	Image Base: 0x7ffd...
7:18:0...	Server.exe	5164	Load Image	C:\Windows\System32\wow64win.dll	SUCCESS	Image Base: 0x7ffd...
7:18:0...	Server.exe	5164	CreateFile	C:\Windows\System32\wow64log.dll	NAME NOT FOUND	Desired Access: R...
7:18:0...	Server.exe	5164	CreateFile	C:\Windows	SUCCESS	Desired Access: R...
7:18:0...	Server.exe	5164	QueryNameInfo...	C:\Windows	SUCCESS	Name: \Windows
7:18:0...	Server.exe	5164	CloseFile	C:\Windows	SUCCESS	
7:18:0...	Server.exe	5164	RegOpenKey	HKLM\Software\Microsoft\Wow64\x86	SUCCESS	Desired Access: R...
7:18:0...	Server.exe	5164	RegQueryValue	HKLM\SOFTWARE\Microsoft\Wow64\...	NAME NOT FOUND	Length: 520
7:18:0...	Server.exe	5164	RegQueryValue	HKLM\SOFTWARE\Microsoft\Wow64\...	SUCCESS	Type: REG_SZ, Le...
7:18:0...	Server.exe	5164	RegCloseKey	HKLM\SOFTWARE\Microsoft\Wow64\...	SUCCESS	
7:18:0...	Server.exe	5164	Load Image	C:\Windows\System32\wow64cpu.dll	SUCCESS	Image Base: 0x775...
7:18:0...	Server.exe	5164	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	REPARSE	Desired Access: Q...
7:18:0...	Server.exe	5164	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	Desired Access: Q...
7:18:0...	Server.exe	5164	RegSetInfoKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	KeySetInformation...
7:18:0...	Server.exe	5164	RegQueryValue	HKLM\System\CurrentControlSet\Contr...	NAME NOT FOUND	Length: 80
7:18:0...	Server.exe	5164	RegCloseKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	
7:18:0...	Server.exe	5164	RegOpenKey	HKLM\SYSTEM\CurrentControlSet\Con...	REPARSE	Desired Access: Q...
7:18:0...	Server.exe	5164	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	NAME NOT FOUND	Desired Access: Q...
7:18:0...	Server.exe	5164	RegOpenKey	HKLM\SYSTEM\CurrentControlSet\Con...	REPARSE	Desired Access: Q...
7:18:0...	Server.exe	5164	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	Desired Access: Q...
7:18:0...	Server.exe	5164	RegSetInfoKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	KeySetInformation...
7:18:0...	Server.exe	5164	RegQueryValue	HKLM\System\CurrentControlSet\Contr...	NAME NOT FOUND	Length: 24
7:18:0...	Server.exe	5164	RegCloseKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	
7:18:0...	Server.exe	5164	CreateFile	C:\Users\user1\Desktop	SUCCESS	Desired Access: E...
7:18:0...	Server.exe	5164	CreateFile	C:\Windows\SysWOW64\mscoree.dll	SUCCESS	Desired Access: R...
7:18:0...	Server.exe	5164	QueryBasicInfor...	C:\Windows\SysWOW64\mscoree.dll	SUCCESS	CreationTime: 12/7...
7:18:0...	Server.exe	5164	CloseFile	C:\Windows\SysWOW64\mscoree.dll	SUCCESS	
7:18:0...	Server.exe	5164	CreateFile	C:\Windows\SysWOW64\mscoree.dll	SUCCESS	Desired Access: R...
7:18:0...	Server.exe	5164	CreateFileMapp...	C:\Windows\SysWOW64\mscoree.dll	FILE LOCKED WI...	SyncType: SyncTy...
7:18:0...	Server.exe	5164	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	REPARSE	Desired Access: R...
7:18:0...	Server.exe	5164	RegOpenKey	HKLM\System\CurrentControlSet\Contr...	SUCCESS	Desired Access: R...

Showing 31,812 of 4,275,904 events (0.74%) Backed by virtual memory

As these indicators are marked we can see: (from right to left)

-process and thread activity.

-network activity

- file system activity

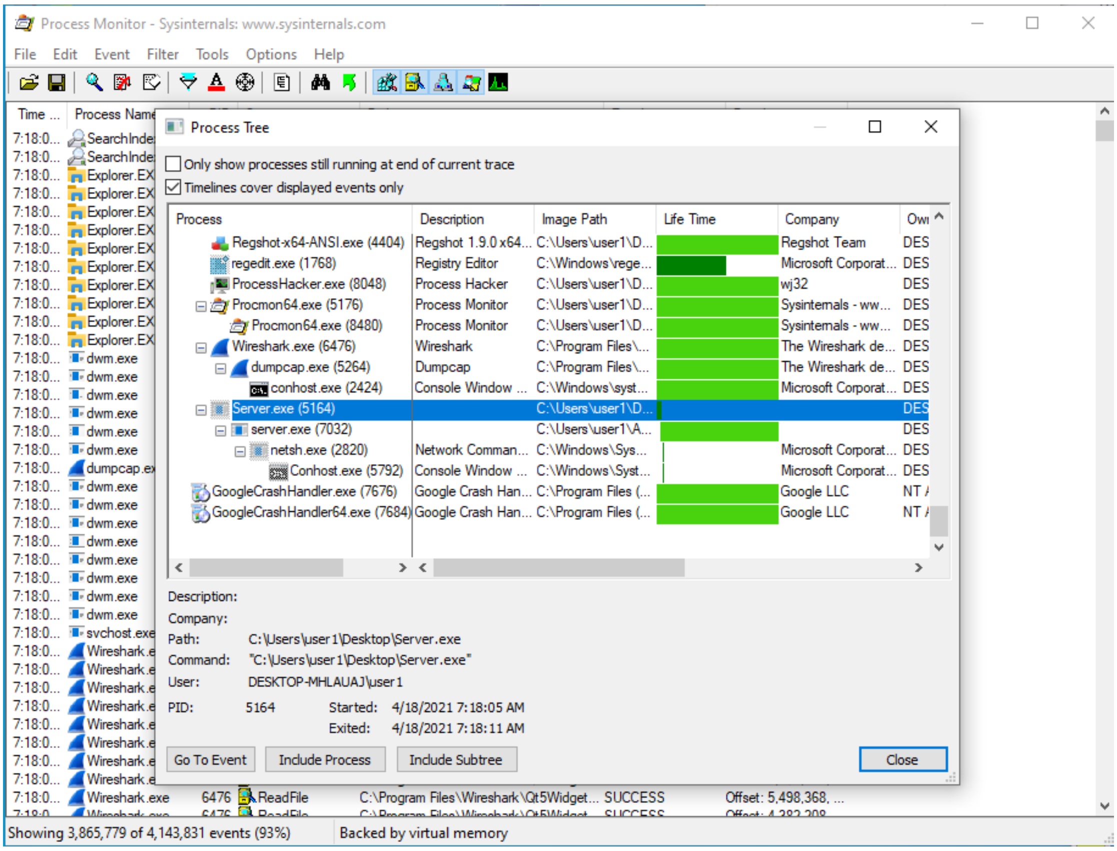
- Registry activity

We can see over 4 million events that have occurred while the malware was running on the machine: Registry modifications, files' changes, processes intervention and more.

"Process Monitor" has a toll called "Tree".
The "Tree" tool shows the relationships of all the running processes (parent-processes and child-processes).

Using this tool we can see that the main "server.exe" file ran and it has a very short "Life Time".

Below it, we can see another process called "server.exe" but this one has a long "Life Time".


Process Monitor - Sysinternals: www.sysinternals.com
File Edit Event Filter Tools Options Help
Process Tree

☐ Only show processes still running at end of current trace
☑ Timelines cover displayed events only

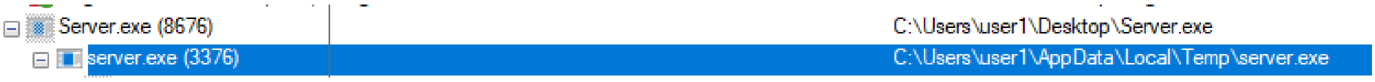
Process	Description	Image Path	Life Time	Company	Own
Regshot-x64-ANSI.exe (4404)	Regshot 1.9.0 x64...	C:\Users\user1\D...	🟩	Regshot Team	DES
regedit.exe (1768)	Registry Editor	C:\Windows\rege...	🟩	Microsoft Corporat...	DES
ProcessHacker.exe (8048)	Process Hacker	C:\Users\user1\D...	🟩	wj32	DES
⊟ Procmon64.exe (5176)	Process Monitor	C:\Users\user1\D...	🟩	Sysinternals - ww...	DES
Procmon64.exe (8480)	Process Monitor	C:\Users\user1\D...	🟩	Sysinternals - ww...	DES
⊟ Wireshark.exe (6476)	Wireshark	C:\Program Files\...	🟩	The Wireshark de...	DES
⊟ dumpcap.exe (5264)	Dumpcap	C:\Program Files\...	🟩	The Wireshark de...	DES
conhost.exe (2424)	Console Window ...	C:\Windows\syst...	🟩	Microsoft Corporat...	DES
⊟ Server.exe (5164)		C:\Users\user1\D...			DES
⊟ server.exe (7032)		C:\Users\user1\A...	🟩		DES
⊟ netsh.exe (2820)	Network Comman...	C:\Windows\Sys...		Microsoft Corporat...	DES
Conhost.exe (5792)	Console Window ...	C:\Windows\Syst...		Microsoft Corporat...	DES
GoogleCrashHandler.exe (7676)	Google Crash Han...	C:\Program Files (...	🟩	Google LLC	NT /
GoogleCrashHandler64.exe (7684)	Google Crash Han...	C:\Program Files (...	🟩	Google LLC	NT /

Description:
Company:
Path: C:\Users\user1\Desktop\Server.exe
Command: "C:\Users\user1\Desktop\Server.exe"
User: DESKTOP-MHLAUAJ\user1
PID: 5164 Started: 4/18/2021 7:18:05 AM
Exited: 4/18/2021 7:18:11 AM

```
7:18:0... Wireshark.exe   6476 ReadFile   C:\Program Files\Wireshark\Qt5Widget... SUCCESS   Offset: 5,498,368, ...
```

Showing 3,865,779 of 4,143,831 events (93%)   Backed by virtual memory

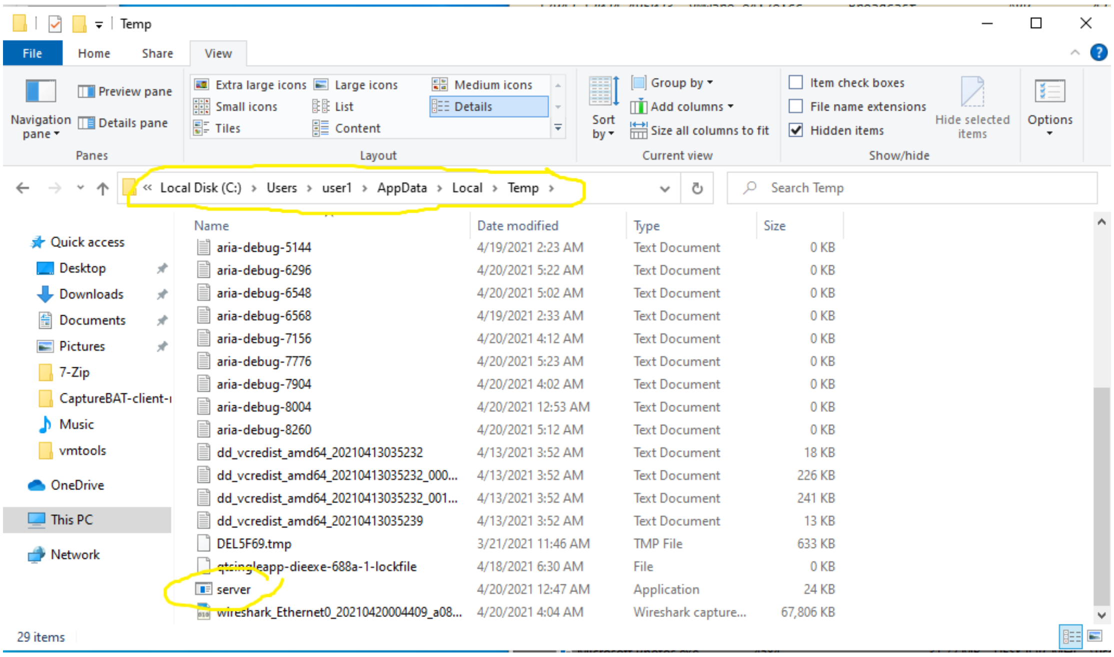
If we focus on the path of the file we will be able to get some important information:



```
⊟ Server.exe (8676)        C:\Users\user1\Desktop\Server.exe
  ⊟ server.exe (3376)      C:\Users\user1\AppData\Local\Temp\server.exe
```

The first "Server.exe" file that is located on the desktop is the file I executed because this is the malware I intended to run on my "victim's" machine.
After quite a short while the second "Server.exe" file that is located in the Temp directory was running and the first file's process was killed.
 That happened because the malware copied itself to the "TEMP" directory.

Malware software tend to copy themselves into different and hidden directories in order to hide themselves so that they would still be present on the machine even after the original file has been deleted or removed. If I open this path (C:\Users\User1\AppData\Local\Temp), I will see the "Server.exe" file:



« Local Disk (C:) › Users › user1 › AppData › Local › Temp ›

Name	Date modified	Type	Size
aria-debug-5144	4/19/2021 2:23 AM	Text Document	0 KB
aria-debug-6296	4/20/2021 5:22 AM	Text Document	0 KB
aria-debug-6548	4/20/2021 5:02 AM	Text Document	0 KB
aria-debug-6568	4/19/2021 2:33 AM	Text Document	0 KB
aria-debug-7156	4/20/2021 4:12 AM	Text Document	0 KB
aria-debug-7776	4/20/2021 5:23 AM	Text Document	0 KB
aria-debug-7904	4/20/2021 4:02 AM	Text Document	0 KB
aria-debug-8004	4/20/2021 12:53 AM	Text Document	0 KB
aria-debug-8260	4/20/2021 5:12 AM	Text Document	0 KB
dd_vcredist_amd64_20210413035232	4/13/2021 3:52 AM	Text Document	18 KB
dd_vcredist_amd64_20210413035232_000...	4/13/2021 3:52 AM	Text Document	226 KB
dd_vcredist_amd64_20210413035232_001...	4/13/2021 3:52 AM	Text Document	241 KB
dd_vcredist_amd64_20210413035239	4/13/2021 3:52 AM	Text Document	13 KB
DEL5F69.tmp	3/21/2021 11:46 AM	TMP File	633 KB
qtsingleapp-dieexe-688a-1-lockfile	4/18/2021 6:30 AM	File	0 KB
server	4/20/2021 12:47 AM	Application	24 KB
wireshark_Ethernet0_20210420004409_a08...	4/20/2021 4:04 AM	Wireshark capture...	67,806 KB

29 items

Netsh.exe- network shell, is a command-line utility included in Microsoft's Windows NT line of operating system. It allows local or remote configuration of network devices.

Conhost.exe- Console Window Host is the process of a program (cryptominer) that is designed to mine Monero cryptocurrency. Generally, cyber criminals trick people into downloading and installing this program to generate revenue. In summary, the program uses computer resources to mine cryptocurrency. The presence of this malware significantly diminishes computer performance.

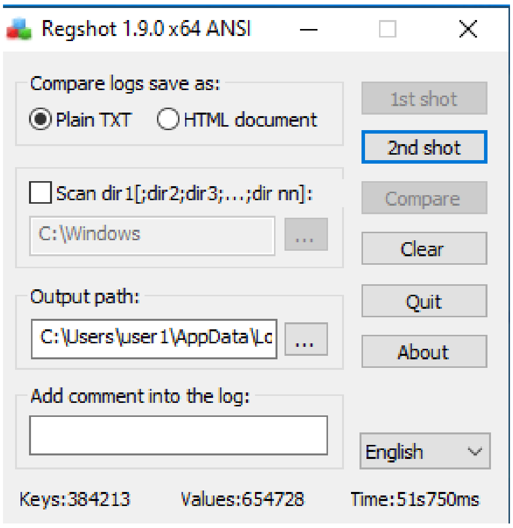
NJrat can target cryptocurrency wallet applications and steal cryptocurrency from PCs. For example, it can grab a bitcoin wallet and access credit card information, which is usually stored in cryptocurrency apps as a way to purchase cryptocurrency.

Both the "Conshot.exe" file and the "netsh.exe" file are having a short "Life Time" because the machines have no internet connection therefore these processes were terminated.

Regshot - snapshot #2:

After executing the malware and giving it time to make changes on the machine, I took a second snapshot of the registry in order to compare it to the first snapshot (taken before the execution of the malware):

1- I pressed - "2nd shot"



Regshot 1.9.0 x64 ANSI

Compare logs save as: ● Plain TXT ○ HTML document

□ Scan dir1[;dir2;dir3;...;dir nn]: C:\Windows

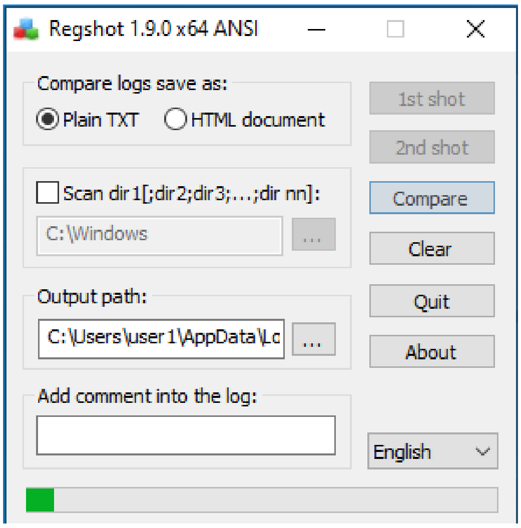
Output path: C:\Users\user1\AppData\Lc

Add comment into the log:

English

Keys:384213 Values:654728 Time:51s750ms

2- I pressed - "Compare"



Regshot 1.9.0 x64 ANSI

Compare logs save as: ● Plain TXT ○ HTML document

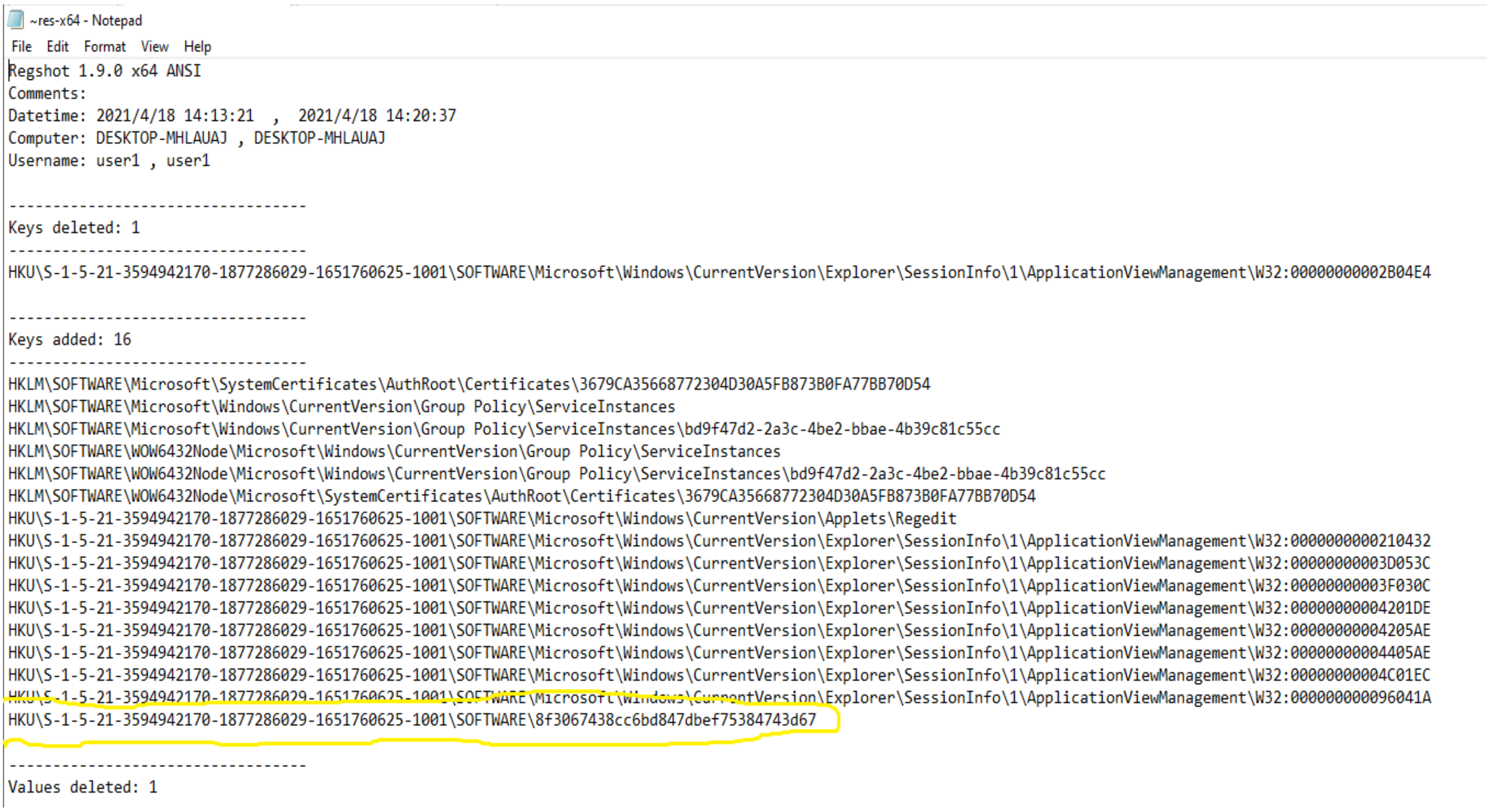
□ Scan dir1[;dir2;dir3;...;dir nn]: C:\Windows

Output path: C:\Users\user1\AppData\Lc

Add comment into the log:

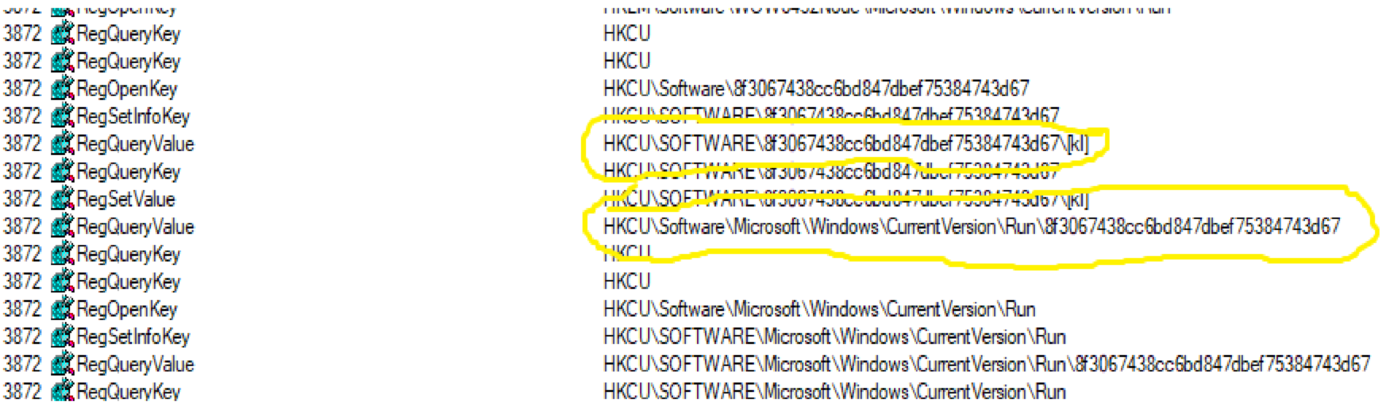
English

The Comparing File



```
Regshot 1.9.0 x64 ANSI
Comments:
Datetime: 2021/4/18 14:13:21  ,  2021/4/18 14:20:37
Computer: DESKTOP-MHLAUAJ , DESKTOP-MHLAUAJ
Username: user1 , user1

----------------------------------
Keys deleted: 1
----------------------------------
HKU\S-1-5-21-3594942170-1877286029-1651760625-1001\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\SessionInfo\1\ApplicationViewManagement\W32:00000000002B04E4

----------------------------------
Keys added: 16
----------------------------------
HKLM\SOFTWARE\Microsoft\SystemCertificates\AuthRoot\Certificates\3679CA35668772304D30A5FB873B0FA77BB70D54
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Group Policy\ServiceInstances
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Group Policy\ServiceInstances\bd9f47d2-2a3c-4be2-bbae-4b39c81c55cc
HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Group Policy\ServiceInstances
HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Group Policy\ServiceInstances\bd9f47d2-2a3c-4be2-bbae-4b39c81c55cc
HKLM\SOFTWARE\WOW6432Node\Microsoft\SystemCertificates\AuthRoot\Certificates\3679CA35668772304D30A5FB873B0FA77BB70D54
HKU\S-1-5-21-3594942170-1877286029-1651760625-1001\SOFTWARE\Microsoft\Windows\CurrentVersion\Applets\Regedit
HKU\S-1-5-21-3594942170-1877286029-1651760625-1001\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\SessionInfo\1\ApplicationViewManagement\W32:0000000000210432
HKU\S-1-5-21-3594942170-1877286029-1651760625-1001\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\SessionInfo\1\ApplicationViewManagement\W32:00000000003D053C
HKU\S-1-5-21-3594942170-1877286029-1651760625-1001\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\SessionInfo\1\ApplicationViewManagement\W32:00000000003F030C
HKU\S-1-5-21-3594942170-1877286029-1651760625-1001\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\SessionInfo\1\ApplicationViewManagement\W32:00000000004201DE
HKU\S-1-5-21-3594942170-1877286029-1651760625-1001\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\SessionInfo\1\ApplicationViewManagement\W32:00000000004205AE
HKU\S-1-5-21-3594942170-1877286029-1651760625-1001\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\SessionInfo\1\ApplicationViewManagement\W32:00000000004405AE
HKU\S-1-5-21-3594942170-1877286029-1651760625-1001\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\SessionInfo\1\ApplicationViewManagement\W32:00000000004C01EC
HKU\S-1-5-21-3594942170-1877286029-1651760625-1001\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\SessionInfo\1\ApplicationViewManagement\W32:000000000096041A
HKU\S-1-5-21-3594942170-1877286029-1651760625-1001\SOFTWARE\8f3067438cc6bd847dbef75384743d67

----------------------------------
Values deleted: 1
```

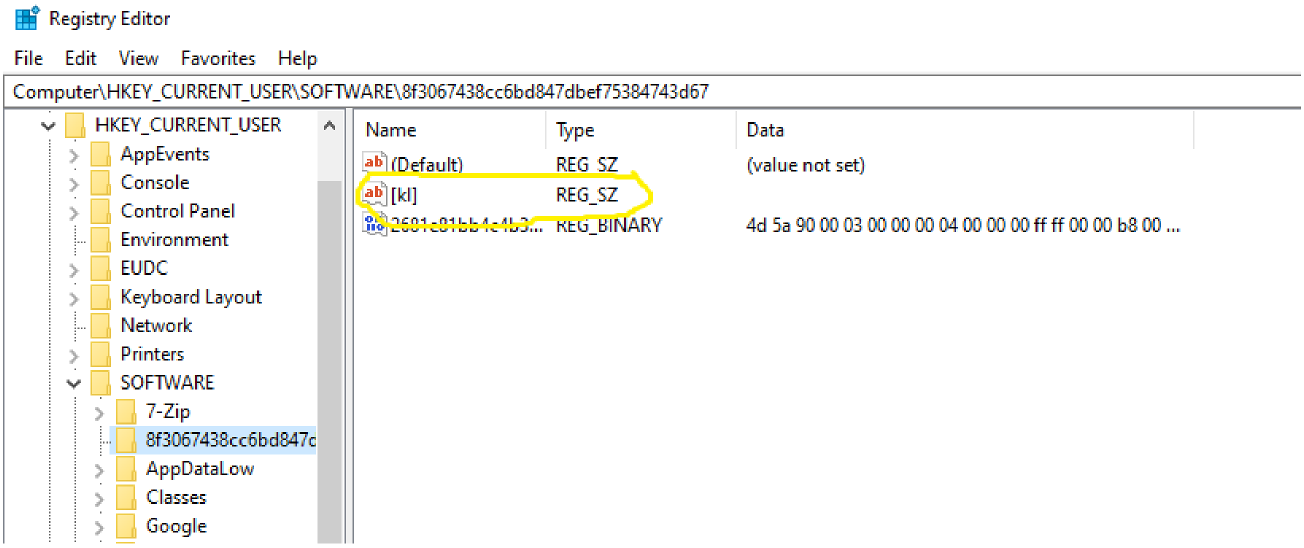
In addition, I examine the registry activities using "Process Monitor".
(By canceling all the others signs)



```
3872  RegOpenKey      HKLM\Software\WOW6432Node\Microsoft\Windows\CurrentVersion\Run
3872  RegQueryKey     HKCU
3872  RegQueryKey     HKCU
3872  RegOpenKey      HKCU\Software\8f3067438cc6bd847dbef75384743d67
3872  RegSetInfoKey   HKCU\SOFTWARE\8f3067438cc6bd847dbef75384743d67
3872  RegQueryValue   HKCU\SOFTWARE\8f3067438cc6bd847dbef75384743d67\[kl]
3872  RegQueryKey     HKCU\SOFTWARE\8f3067438cc6bd847dbef75384743d67
3872  RegSetValue     HKCU\SOFTWARE\8f3067438cc6bd847dbef75384743d67\[kl]
3872  RegQueryValue   HKCU\Software\Microsoft\Windows\CurrentVersion\Run\8f3067438cc6bd847dbef75384743d67
3872  RegQueryKey     HKCU
3872  RegQueryKey     HKCU
3872  RegOpenKey      HKCU\Software\Microsoft\Windows\CurrentVersion\Run
3872  RegSetInfoKey   HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
3872  RegQueryValue   HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\8f3067438cc6bd847dbef75384743d67
3872  RegQueryKey     HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
```

The information written above indicates that the malware has added a new key to the registry - **8f3067438cc6bd847dbef75384743d67** with the **value** of - **[kl].**

In order to confirm it I checked it on the Registry Editor:



Computer\HKEY_CURRENT_USER\SOFTWARE\8f3067438cc6bd847dbef75384743d67

Name	Type	Data
(Default)	REG_SZ	(value not set)
[kl]	REG_SZ	
2681c81bb4c4b3...	REG_BINARY	4d 5a 90 00 03 00 00 00 04 00 00 00 ff ff 00 00 b8 00 ...

1- The malware created a key with the name **[kl]** into the path **HKEY_CURRENT_USER\Software\8f3067438cc6bd847dbef75384743d67.**

2- The malware has added another key into the following path **HKEY_CURRENT_USER\Software\Microsoft\CurrentVersion\Run\8f3067438cc6bd847dbef75384743d67.**

The above key is at the USER level and it is used by the malware to achieve persistence
Malware often use persistence so that the malware can communicate with the infected system even after the system reboots or logs-off. The persistence on the system, the malware's author can also use the affected system to infiltrate to other systems in the local network.

Summary:

NjRAT has the ability to:

* Remote control over the infected machine.14

* See the victim's IP address and view the machine's information such as computer name, username, operating system, install-date, version, etc.

* Remotely execute a file.

* Manipulate files.

* Open a remote shell, providing the attacker with access to a command line.

* Open "Process Manager" tool and kill running processes.

* Manipulate the remote-system's registry.

* Record the computer's camera footage and microphone.

* Log keystrokes.

* Steal passwords that are stored on the installed web browsers or on other applications.

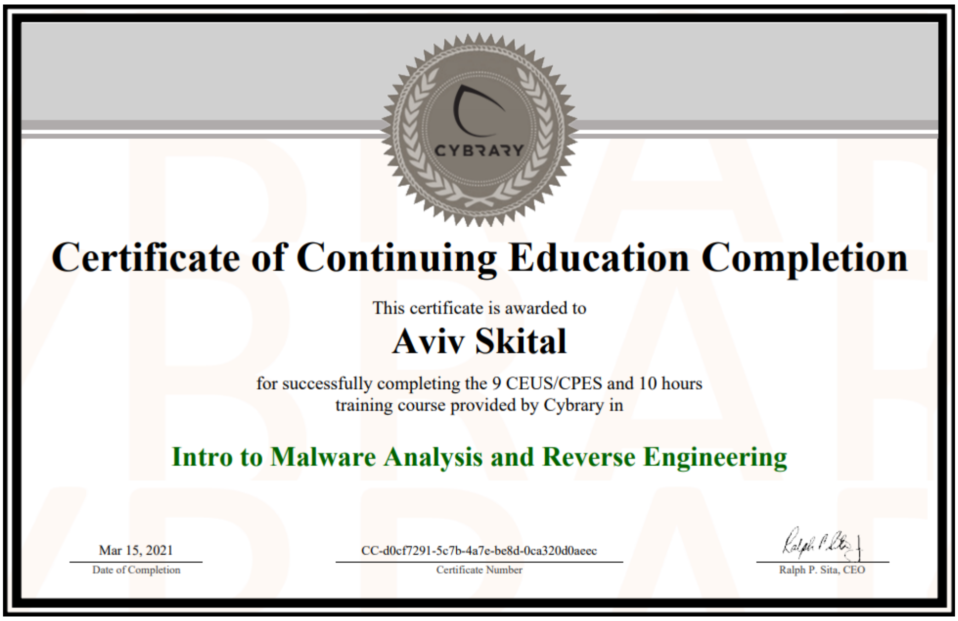
* Target cryptocurrency wallet applications and steal cryptocurrency from PC's.

Self-reflection and takeaways:

This project has given me the ability to study and have an in-depth understanding of how malware work. I learned to investigate and search for information over the internet in an accurate and consistent manner. Furthermore, I learned about the "Windows" operating system and the tools included in it. This project has given me many tools in the field of Cyber-Security in general and in the field of incident response in particular.

Sources of information

1- Online course on Cybrary.it:



CYBRARY

Certificate of Continuing Education Completion

This certificate is awarded to

Aviv Skital

for successfully completing the 9 CEUS/CPES and 10 hours training course provided by Cybrary in

Intro to Malware Analysis and Reverse Engineering

Mar 15, 2021
Date of Completion

CC-d0cf7291-5c7b-4a7e-be8d-0ca320d0aeec
Certificate Number

Ralph P. Sita, CEO

2- Websites:

https://www.logsign.com/blog/malware-analysis-things-you-should-know

https://www.cynet.com/attack-techniques-hands-on/njrat-report-bladabindi

https://www.file.net/process/mscoree.dll.html

https://blog.cyberint.com/njrat-bulletin

https://www.carbonblack.com/blog/threat-analysis-unit-tau-threat-/intelligence-notification-njrat

https://www.pcrisk.com/removal-guides/14828-conhost-exe-virus

https://resources.infosecinstitute.com/topic/common-malware-/persistence-mechanisms

https://whiteheart0.medium.com/entropy-analysis-a-critical-test-for-malwares-69939f5b8b1

3- Books:

* Practical Malware Analysis- The Hands-On Guide to Dissecting.
* Malicious Software/ Richard Bejtlich.
* Operating Systems / Barak Gonen.