

REAL WORLD HANDS ON EXAMPLES

Malware Analysis

SIEM Log analysis

WannaCry

Image Forensics

Sandboxing

CYBER DEFENSE FORENSICS ANALYST

SANDEEP KUMAR SEERAM

Cyber Defense Forensics Analyst

“Real World Hands on Examples”

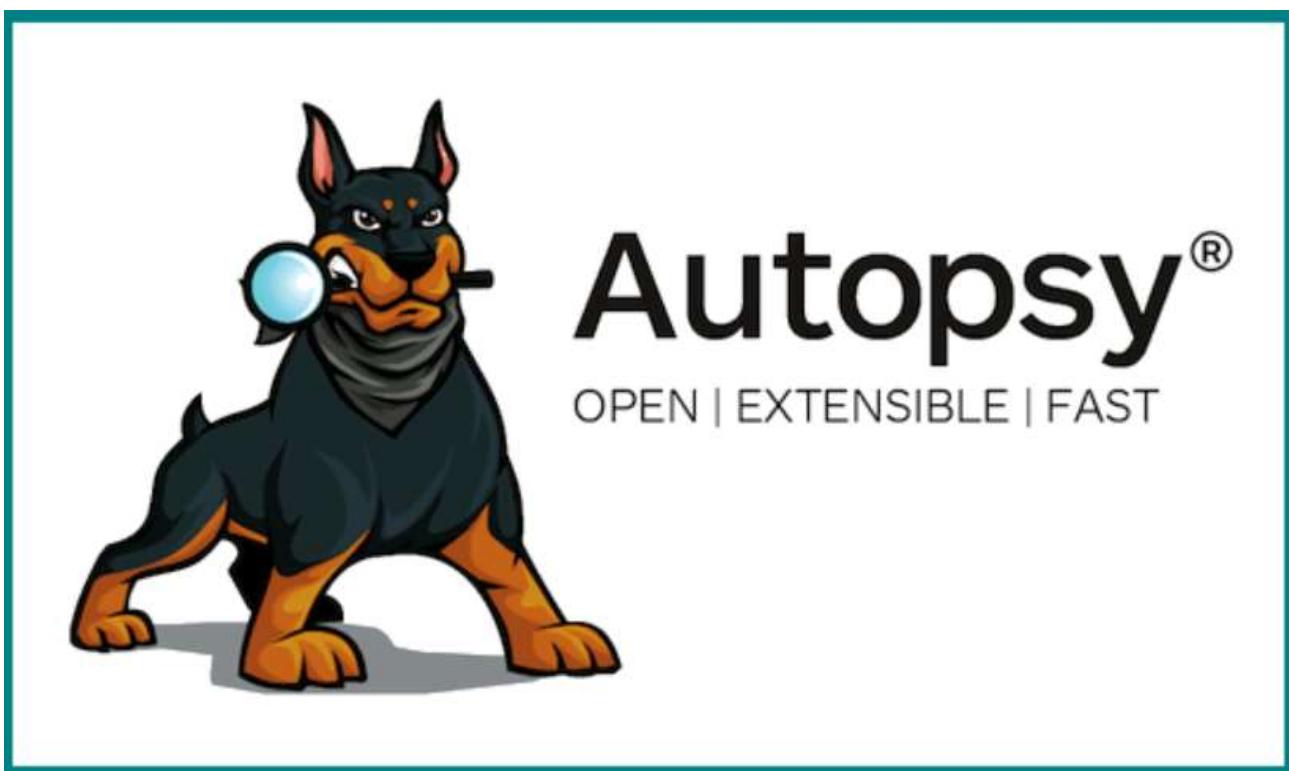
Chapter 6: Windows Forensics

Computer forensics (also known as computer forensic science) is a branch of digital forensic science pertaining to evidence found in computers and digital storage media.

The goal of computer forensics is to examine digital media in a forensically sound manner with the aim of identifying, preserving, recovering, analysing and presenting facts and opinions about digital information.

For this chapter, we will use [Autopsy](#) - an open source forensic analysis tool. Using the tool, you'll need to identify user behaviour and files inside a system image.

We will use Autopsy and create a case to analyse the Windows Forensics image. Using Autopsy's features, and through browsing the file system to understand the image.



We will start by creating the case:

The image shows the 'New Case Information' dialog box. The left sidebar lists 'Steps' with '1. Case Info' and '2. Additional Information'. The main panel is titled 'Case Info' and contains the following fields:

- Case Name:
- Base Directory:
- Case Type: Single-user Multi-user
- Case data will be stored in the following directory:

At the bottom, there are navigation buttons: '< Back' (disabled), 'Next >', 'Finish' (disabled), 'Cancel', and 'Help'.

New Case Information

Steps

1. Case Info
2. Additional Information

Additional Information

Optional: Set Case Number and Examiner

Case Number: 12345

Examiner: Sandeep Kumar Seeram

< Back Next > **Finish** Cancel Help

The case database will be built upon providing the necessary details. Proceed with adding the source disk/image for forensic analysis.

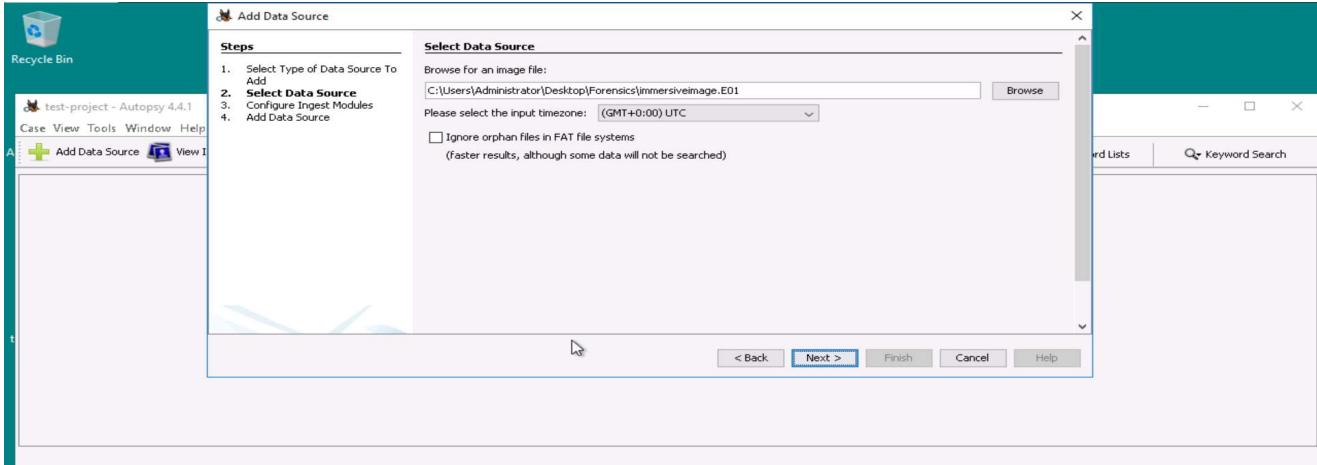
Add Data Source

Steps

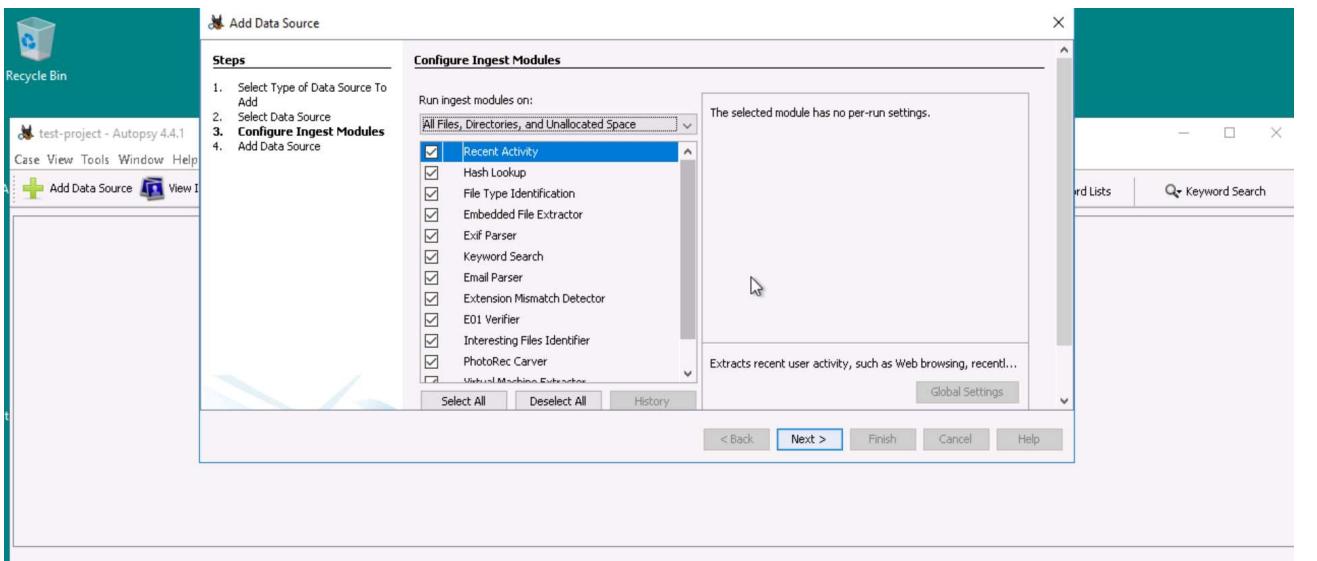
1. **Select Type of Data Source To Add**
2. Select Data Source
3. Configure Ingest Modules
4. Add Data Source

Select Type of Data Source To Add

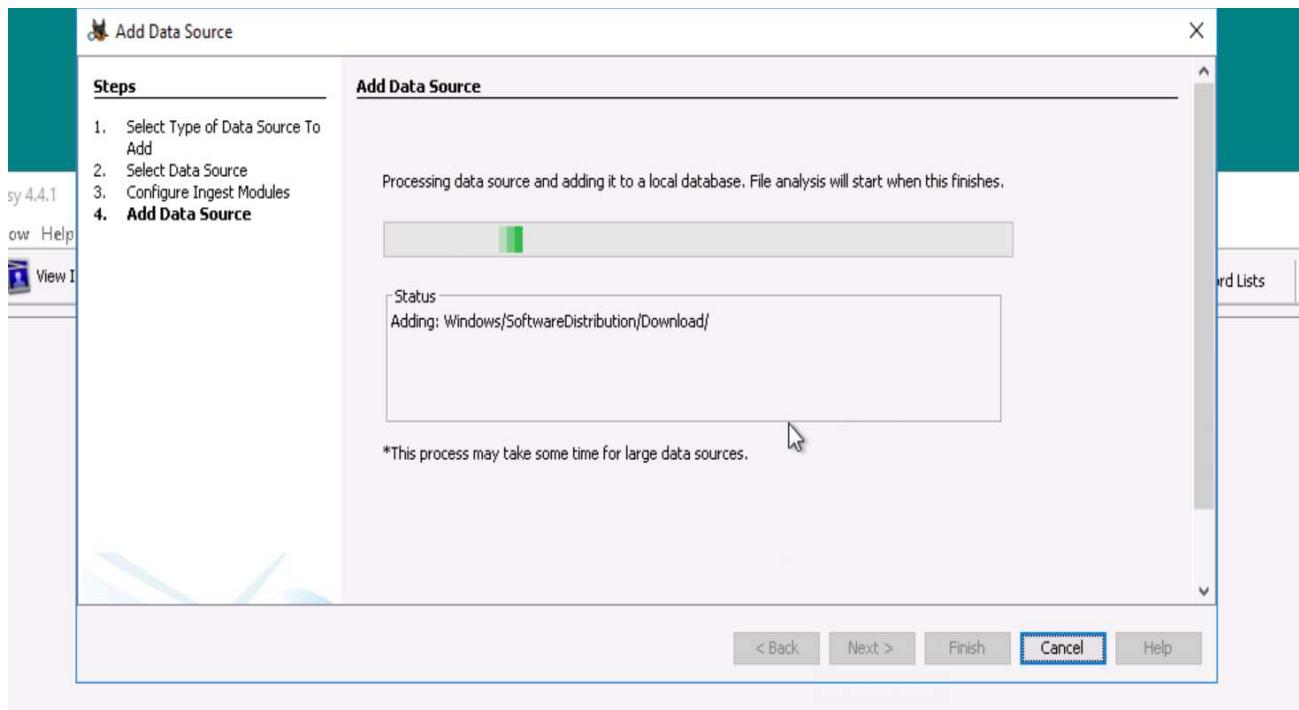
Disk Image or VM File
Local Disk
Logical Files
Unallocated Space Image File



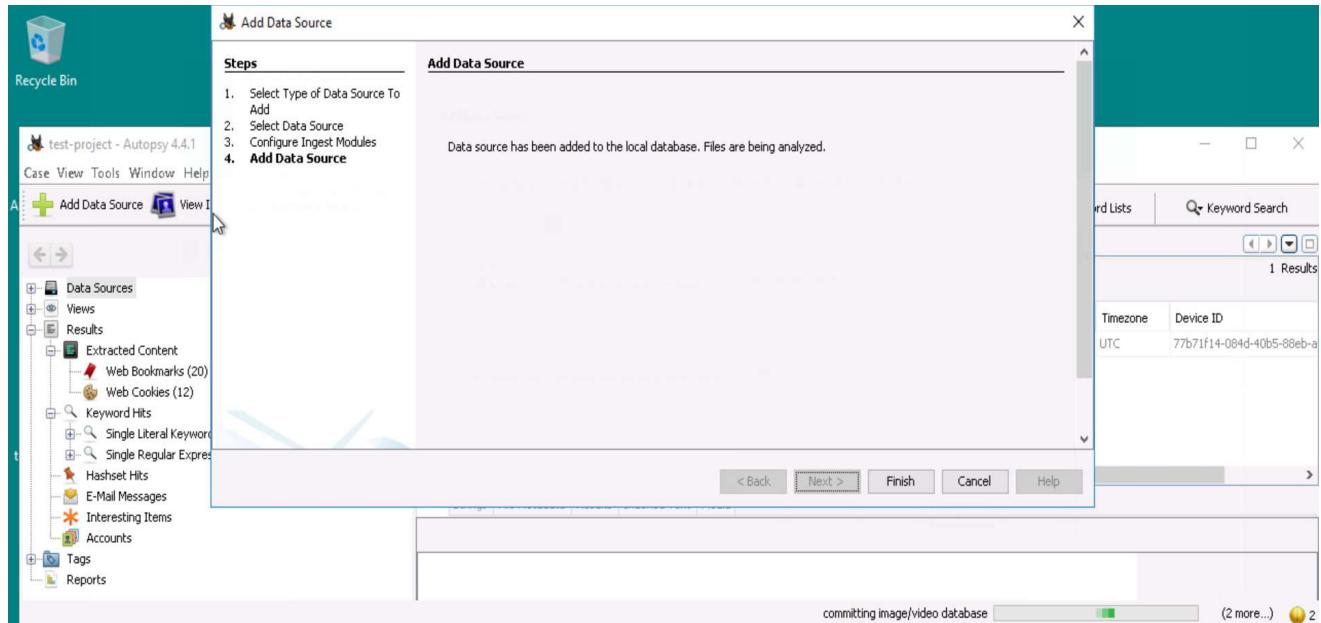
Next step will be selecting the modules – each module is configured to look into the forensic aspect of an image.



Adding disk to the database:



Once data is added to the database, files will be analysed.



The screenshot shows the Autopsy 4.4.1 interface. The left sidebar displays a tree view of data sources, views, and results. Under 'Results', 'Extracted Content' is expanded, showing 'Recent Documents (14)' which is selected. Other categories like 'Extension Mismatch Detected (1)', 'Web Bookmarks (20)', 'Web Cookies (12)', 'Web History (1338)', and 'Web Search (1)' are also listed. The main pane shows a 'Listing' of files with columns for 'Source File', 'Path', 'Date/Time', 'Data Source', and 'Tags'. The table has 14 results. The first row is 'Appearance and Personalization.lnk' with 'No preferred path found'. The second row is 'bgconfig.lnk' located at 'C:\Wallpaper\bgconfig.bgl'. The third row is 'Bginfo_wallpaper.lnk' located at 'C:\Wallpaper\Bginfo_wallpaper.zip'. The fourth row is 'Hidden.lnk' located at 'C:\Users\IEUser\Desktop\Hidden.zip'. The fifth row is 'Hidden.lnk' located at 'C:\Users\IEUser\Pictures'. The bottom of the interface shows tabs for 'Hex', 'Strings', 'File Metadata', 'Results', 'Indexed Text', and 'Media'. A status bar at the bottom right shows 'Recent Activity for immersiveimage.E01', '83%', '(1 more...)', and a yellow warning icon with the number '2'.

| Source File | Path | Date/Time | Data Source | Tags |
|------------------------------------|------------------------------------|-------------------------|--------------------|------|
| Appearance and Personalization.lnk | No preferred path found | 2013-10-23 22:04:02 UTC | immersiveimage.E01 | |
| bgconfig.lnk | C:\Wallpaper\bgconfig.bgl | 2013-10-23 22:03:32 UTC | immersiveimage.E01 | |
| Bginfo_wallpaper.lnk | C:\Wallpaper\Bginfo_wallpaper.zip | 2013-10-23 21:51:04 UTC | immersiveimage.E01 | |
| Hidden.lnk | C:\Users\IEUser\Desktop\Hidden.zip | 2017-08-10 07:29:24 UTC | immersiveimage.E01 | |
| Hidden.lnk | C:\Users\IEUser\Pictures | 2017-08-10 07:42:11 UTC | immersiveimage.E01 | |

You will be presented with lot of information on the image, files, recent activity, bookmarks, web history, search history, email client information etc..

Generate the report and submit it as your evidence.

Chapter 11: Malware Analysis

If you are working for a large enterprise in the Security team, you will see lot of malware targeted at your company every single day. As a Cyber Defence Forensics Analyst, you will be tasked to conduct analysis on the malware to understand the true nature and behaviour.

Malware analysis is growing complex and now it's an ever-evolving skill, with tools continually being created and updated to analyse modern malware. On the other side, malware authors are creating complex samples that cannot be fully analysed without a combination of tools and techniques.

Normally when analysing malware, the analyst will have limited time to learn what the malware is doing (and how to isolate it).

For example, is the malware using a static web domain? Block that domain. Read how WannaCry got mitigated, the backdoor domain was identified and blocked.

There are a number of questions that analysts need to answer as quickly as possible; these are:

- What classification is the malware?
- Is the malware making any connections?
- Is the malware changing the system in anyway?
- What functions is the malware using?

Every piece of malware can be analysed with two different techniques:

1. Static Malware Analysis
2. Dynamic Malware Analysis

In this chapter, we will start with Static Malware Analysis and learn the popular tools and techniques used and then we dive deep into Dynamic Malware Analysis with some real-world examples.

Static Malware Analysis:

Static analysis is analysing a piece of malware without executing it. This means that the malware never gets loaded into memory and the instructions are never run. As a analyst you need to look through the instructions stored in the .text section to see what the program would do if it was loaded into memory.

Static analysis is difficult as there is no memory allocated to the program, such as the stack; therefore, you cannot check values in memory at certain points, rendering this type of analysis is slow and quite difficult.

However, there are many tools that can be used to make this process easier. The analyst does not need to read machine code to understand what is going on; there are tools, such as disassemblers and executable viewers which help to understand and analyse a piece of malware.

[Disassemblers](#) are a static analyst's dream: they take machine code and convert it into corresponding assembly code. The analyst then has to read the assembly language to understand

what the program is doing. There are a variety of disassemblers on the market, but to get the best tools you will have to pay. Luckily, there are free demo versions of the paid-for tools.

- IDA pro (demo version IDA free)
- Radare2

Go through the chapter Reverse Engineering to see how Radare2 performed a binary analysis.

[Executable Viewer](#): When analysing a piece of malware, it is worth looking at the type of data, not just instructions to be run. Sometimes malware holds valuable information in other sections – such as the .data section, where initialised global variables are stored. Other tools can be used to get data about a file, which will change the way you analyse it. Is the file a PE file or an ELF file? Understanding this information changes the way you will analyse the file. The following are OpenSource tools commonly used for this

- File
- Strings
- Readelf
- PEStudio

Dynamic Analysis

Dynamic analysis is interacting with malware in a way that executes it. Once the malware is executed and running, there is an active effort to understand what it is doing to the system it's running on. This can be done in a number of ways.

First, The Malware Analyst can execute the malware before taking a snapshot of the system for further investigation to see what has changed? Any files changed? Any connections been made?

Second, The analyst can also use a debugger to execute a program step by step.

But running malware can be scary, especially when the result is unknown. Because of this, the analyst needs to think about the system that they are running the malware on. There are a few guidelines that are important to think about when executing malware:

- Do not connect the system to the internet
- Do not run on a host machine that has important information stored
- Give the malware least possible privileges and work your way up

If the malware needs internet in order to fully unpack itself, then there are a few tools that can be used to simulate network connections. My favourite one is "fakenet".

These tools will reply to any request with the relevant acknowledgement that the protocol specifies. For example, it will respond with a SYN ACK packet to a SYN TCP packet.

The best in class debuggers are not limited to the list, but the below listed products are widely used:

- Ollydbg
- X64dbg
- Windbg

- ImmunityDebugger

Analysis Environments are generally isolated sandbox environments used to analyse the malware. These are tools that will execute the malware, and then record any changes to the system that the malware makes, including any connections created and any odd behaviour that is relevant. These are a couple of malware analysis environments to note, both of which are open source:

- Cuckoo
- VxStream

We will use a [cloud malware analysis environment](#) in this book and test some real-world malwares and witness the power of malware analysis.

Malware detected in organizations comes in different shapes and forms. Sometimes an executable (.exe) sometimes a (.pdf) file and sometimes it's just a domain.

Whenever anything is identified as malicious/suspicious the file goes through analysis. Many organizations have automated the malware analysis process, there are tools/services available that can help enterprises automate the entire analysis.

For this book, we are going to use [Netflix Account Generator.exe](#) which is classified as an evasion and remote access trojan malware.

We will execute this malware in a controlled/isolated cloud sandbox which is running below configuration. In malware analysis, we need to have the capability to test the malware in various Operating Systems with different environments variables, applications and hotfixes.

Windows 7 Professional 32bit is selected for this malware analysis.



Next we define the Applications and Hotfixes, if you are malware researcher working on a particular sample, we can use this functionality to identify the vulnerable application and also find a proper hotfix for it.

| APPLICATIONS | HOT FIXES |
|--|----------------|
| Internet Explorer | 8.0.7601.17514 |
| Microsoft Visual C++ 2013 x86 Additional Run... | 12.0.21005 |
| Microsoft Visual C++ 2013 Redistributable (x8... | 12.0.30501.0 |
| Microsoft Visual C++ 2010 x86 Redistributab... | 10.0.40219 |
| Adobe Acrobat Reader DC MUI | 15.023.20070 |
| Adobe Refresh Manager | 1.8.0 |
| Microsoft Visual C++ 2008 Redistributable - x... | 9.0.30729.6161 |
| Update for Microsoft .NET Framework 4.7.2 (K... | 1 |
| Microsoft .NET Framework 4.7.2 | 4.7.03062 |
| Microsoft Office Access Setup Metadata MUI ... | 14.0.6029.1000 |
| + Install Soft | |
| + Install Soft | |

The next granularity setting is to select the malware file that you want to analysis and the network settings. Again my personal favourite is Fake Net, helps to mimic a real network world with an ability to respond to TCP requests. But if you want to run the analysis without disclosing your whereabouts, you have an option to route the traffic via TOR network. It will be impossible to trace the activity back to you.

OBJECT

Type URL or choose a file to run

or

Open in browser

Download with User Agent

Change extension to valid ON OFF

Command Line:

* Type %FILENAME% for replacing on path to the uploaded file in testing system

Start object from

OPTIONS

Duration: or SMART

Privacy:
 Public submission Who has a link Only me

Network: Connected Disconnected

HTTPS MITM proxy Fake Net

Route internet traffic through (optional): Route via TOR User's VPN (OpenVPN)

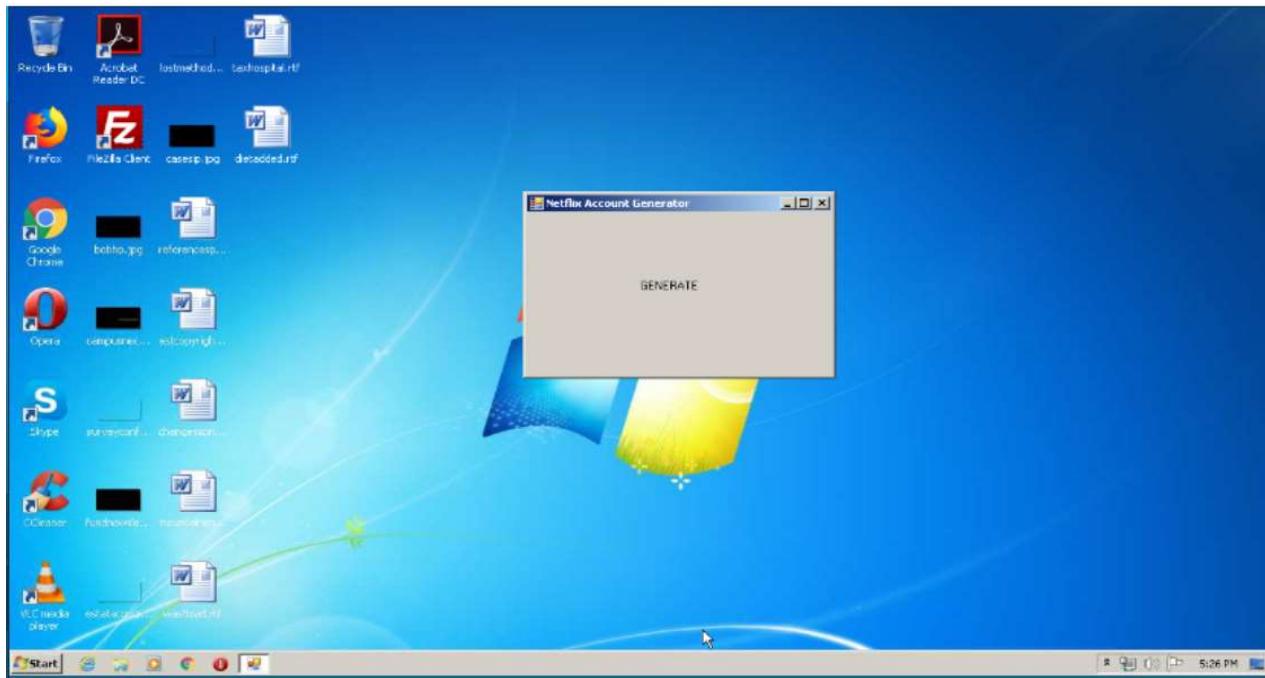
Sandbox building happens in stages and the file we uploaded for the malware analysis is on auto-run.

WINDOWS 7 PROFESSIONAL 32 bit



- 🕒 Loading analyzed objects
- 🕒 Allocating a new environment
- 🕒 Creating a network connection
- 🕒 Preparing to start
- 🕒 Connecting to the incident!

The sandbox environment is built and the file gets auto-executed.



The first thing we try to understand is how the file was executed, any new process started and its entire nature. In our case here, Netflix Account Generator. exe initiated 2 more child processes, each having its malicious nature.

PROCESS Filter by name or PID Show only important

- 3772 Netflix Account Generator.exe PE
 - ↳ EXE
- 2816 lawn mower.exe PE
 - ↳ EXE
 - ↳ POC
- 3036 lawn mower.exe PE
 - ↳ POC
 - ↳ EXE
 - ↳ POC

We will be collecting information about the processes initiated and identified malicious.

PROCESS DETAILS:

Netflix Account Generator.exe (id: 3772)
1.0.0.0
Netflix Account Generator

Username: admin
Start: +0ms

More Info

100 out of 100

Malicious

WARNING

Executable content was dropped or overwritten

Netflix Account Generator.exe (id: 3772)
C:\Users\admin\AppData\Local\Temp\Netflix Account Generator.exe

User: admin
SID: S-1-5-21-1302019708-1500728564-335382590-1000
IL: MEDIUM

Timeline
Created 0 +273 Was run
Terminated 134

Children 2816 | lawn mower.exe

Download

Look up on VT

Command Line: "C:\Users\admin\AppData\Local\Temp\Netflix Account Generator.exe"

Version Information:
Description: Netflix Account Generator
Version: 1.0.0.0

INDICATORS OF SUSPICIOUS BEHAVIOUR

WARNING

Executable content was dropped or overwritten
Creates files in the user directory

EVENTS

MODIFIED FILES 1 REGISTRY CHANGES 26 HTTP REQUESTS 0 CONNECTIONS 2

+1187ms C:\Users\admin\AppData\Roaming\lawnmower.exe
Size: 202 Kb
MD5: 614E1AFB1B3686B14A93B25121D3941

Coming to network connections, we need to verify all the outbound connections and identify any data was exfiltrated from our system.

| Time | Protocol | CN | Rep | ID | Process | IP | Domain | ASN |
|---------|----------|---------|-----|------|----------------------|----------------|----------------|--------------------|
| 612ms | TCP | Germany | 🔥 | 3772 | Netflix Account G... | 88.99.66.31 | 2no.co | Hetzner Online ... |
| 1640ms | TCP | France | ⚠️ | 3772 | Netflix Account G... | 188.165.215.31 | femto.pw | OVH SAS |
| 5732ms | UDP | USA | ✅ | 3036 | lawnmower.exe | 8.8.8.8 | | Google Inc. |
| 5735ms | TCP | Russia | 🔥 | 3036 | lawnmower.exe | 193.161.193.99 | ionusos-255... | OOO Bitree Netw... |
| 10855ms | UDP | USA | ✅ | --- | --- | 8.8.8.8 | | Google Inc. |
| 10858ms | TCP | Russia | 🔥 | 3036 | lawnmower.exe | 193.161.193.99 | ionusos-255... | OOO Bitree Netw... |

| Time | Status | Rep | Domain | IP |
|---------|-----------|-----|----------------------------|----------------|
| 610ms | RESPONDED | 🔥 | 2no.co | 88.99.66.31 |
| 1639ms | RESPONDED | ⚠️ | femto.pw | 188.165.215.31 |
| 5729ms | RESPONDED | 🔥 | ionusos-25533.portmap.host | 193.161.193.99 |
| 10853ms | RESPONDED | 🔥 | ionusos-25533.portmap.host | 193.161.193.99 |

| Time | Class | ID | Process | Message |
|---------|---------------------------------------|------|---------------|---|
| 835ms | Potentially Bad Traffic | 1052 | svchost.exe | ET DNS Query to a *.pw domain - Likely Hostile |
| 4813ms | Potential Corporate Privacy Violation | 3036 | lawnmower.exe | ET POLICY DNS Query to a Reverse Proxy Service Observed |
| 9999ms | Potential Corporate Privacy Violation | 3036 | lawnmower.exe | ET POLICY DNS Query to a Reverse Proxy Service Observed |
| 15124ms | Potential Corporate Privacy Violation | 3036 | lawnmower.exe | ET POLICY DNS Query to a Reverse Proxy Service Observed |

In our case, we have noticed potentially bad traffic out of our system. We can notice the list of file modifications, in case of a ransomware the number of files modified will increase, because ransomware encrypts all the files.

| FILES MODIFICATION | | | | Filter by name | <input checked="" type="checkbox"/> Show only important |
|--------------------|------|------------------------------|---|----------------|---|
| Time offset | ID | Process | Filename | Size | Type |
| 1187ms | 3772 | Netflix Account Generator... | C:\Users\admin\AppData\Roaming\lawnmower.exe | 202 Kb | executable |
| 3328ms | 2816 | lawnmower.exe | C:\Users\admin\AppData\Roaming\90059C37-1320-41A4-B58D-2B75A9850D2F\run.dat | 8 b | binary |
| 3328ms | 2816 | lawnmower.exe | C:\Users\admin\AppData\Roaming\90059C37-1320-41A4-B58D-2B75A9850D2F\TCP Monit or\tcpmon.exe | 202 Kb | executable |

Understanding the complete execution – step by step:

| TITLE | TYPE | IOC | REP | ACTION |
|--|--|-----|--------------------------|-------------------------------------|
| Main object - "Netflix Account Generator.exe" | | | | |
| SHA256 | E8CD9B6446D959B6DABD75A6BAEAF7125411107A559F3CB14648ABE65F407D5E | ? | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| SHA1 | D3D081AE61A35329B7E837FF2FB1B98DD41823C4 | ? | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| MDS | 32C4FDD1110DBFF69C2E003C0829856C | ? | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| Dropped executable file | | | | |
| SHA256 | C:\Users\admin\AppData\Roaming\lawnmower.exe | ? | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| | 7F1AC39D5553C36D78AAC60816D9D018D129E0C7F5274A6798C51F337D5865 | ? | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| DNS requests | | | | |
| DOMAIN | femto.pw | ! | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| DOMAIN | 2no.co | ! | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| DOMAIN | ionusos-25533.portmap.host | ! | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| Connections | | | | |
| IP | 88.99.66.31 | ! | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| IP | 193.161.193.99 | ! | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| IP | 188.165.215.31 | ? | <input type="checkbox"/> | <input checked="" type="checkbox"/> |

Execution Map:



MITRE ATT&CK Matrix maps the stages of an attack. The end goal of any attack is either Command and Control or Data Exfiltration. Generally in organizations we see lot of stages of attacks. I have seen real scenarios, where an attacker got access into an environment – initial access – performed a privilege escalation – performed data collection, installed a compression software (gzip) the a flag was raised in the [security operations center](#), the event was investigated and analysed. Quickly identified the threat and before the attacker was about to perform Data Exfiltration. The attack chain is killed. So its very important to understand the MITRE ATT&CK matrix map. Every investigation report should have the ATT&CK matrix done.



So what is [Netflix Account Generator.exe](#)?

Its classified as Nanocore – a RAT (Remote Access Trojan)

• Nanocore

[nanocore](#) [trojan](#) [rat](#) [loader](#)

NanoCore is a Remote Access Trojan or RAT. This malware is highly customizable with plugins which allow attackers to tailor its functionality to their needs. Nanocore is created with the .NET framework and it's available for purchase for just \$25 from its "official" website.

Type
Trojan

Origin
USA

First seen
1 January, 2013

Last seen
7 March, 2020

We can generate a detailed report on the entire analysis.
