

SPLUNK

Conclusion:

In this fun exercise, as a SOC Analyst, we have investigated a cyber-attack where the attacker had defaced a website 'imreallynotbatman.com' of the Wayne Enterprise. We mapped the attacker's activities into the 7 phases of the Cyber Kill Chain. Let us recap everything we have found so far:

Reconnaissance Phase:

We first looked at any reconnaissance activity from the attacker to identify the IP address and other details about the adversary.

Findings:

IP Address 40.80.148.42 was found to be scanning our webserver.

The attacker was using Acunetix as a web scanner.

Exploitation Phase:

We then looked into the traces of exploitation attempts and found brute-force attacks against our server, which were successful.

Findings:

Brute force attack originated from IP 23.22.63.114.

The IP address used to gain access: 40.80.148.42

142 unique brute force attempts were made against the server, out of which one attempt was successful

Installation Phase:

Next, we looked at the installation phase to see any executable from the attacker's IP Address uploaded to our server.

Findings:

A malicious executable file 3791.exe was observed to be uploaded by the attacker.

We looked at the sysmon logs and found the MD5 hash of the file.

Action on Objective:

After compromising the web server, the attacker defaced the website.

Findings:

We examined the logs and found the file name used to deface the webserver.

Weaponization Phase:

We used various threat Intel platforms to find the attacker's infrastructure based on the following information we saw in the above activities.

Information we had:

Domain: prankglassinebracket.jumpingcrab.com

IP Address: 23.22.63.114

Findings:

Multiple masquerading domains were found associated with the attacker's IPs.

An email of the user Lillian.rose@polson1vy.com was also found associated with the attacker's IP address.

Deliver Phase:

In this phase, we again leveraged online Threat Intel sites to find malware associated with the adversary's IP address, which appeared to be a secondary attack vector if the initial compromise failed.

Findings:

A malware name MirandaTateScreensaver.scr.exe was found associated with the adversary.

MD5 of the malware was c99131e0169171935c5ac32615ed6261

Volatility

Volatility is a free memory forensics tool developed and maintained by Volatility Foundation, commonly used by malware and SOC analysts within a blue team or as part of their detection and monitoring solutions. Volatility is written in Python and is made up of python plugins and modules designed as a plug-and-play way of analyzing memory dumps.

Simple example of using a plugin.

PID	PPID	ImageFileName	Offset	Threads	Handles	SessionId	Wow64	CreateTime	ExitTime	File output
998	652	svchost.exe	0x2029ab8	9	226	0	False	2012-07-22 02:42:33.000000	N/A	Disabled
664	608	lsass.exe	0x202a3b8	24	330	0	False	2012-07-22 02:42:32.000000	N/A	Disabled
652	608	services.exe	0x202ab28	16	243	0	False	2012-07-22 02:42:32.000000	N/A	Disabled
1640	1484	reader_sl.exe	0x207bd00	5	39	0	False	2012-07-22 02:42:36.000000	N/A	Disabled
1512	652	spoolsv.exe	0x20b17b8	14	113	0	False	2012-07-22 02:42:36.000000	N/A	Disabled
1588	1004	wuauctl.exe	0x225bda0	5	132	0	False	2012-07-22 02:44:01.000000	N/A	Disabled
788	652	alg.exe	0x22e8da0	7	104	0	False	2012-07-22 02:43:01.000000	N/A	Disabled
1484	1464	explorer.exe	0x23dea70	17	415	0	False	2012-07-22 02:42:36.000000	N/A	Disabled
1056	652	svchost.exe	0x23dfa00	5	60	0	False	2012-07-22 02:42:33.000000	N/A	Disabled
1136	1004	wuauctl.exe	0x23fcda0	8	173	0	False	2012-07-22 02:43:46.000000	N/A	Disabled
1220	652	svchost.exe	0x2495650	15	197	0	False	2012-07-22 02:42:35.000000	N/A	Disabled
608	368	winlogon.exe	0x2498700	23	519	0	False	2012-07-22 02:42:32.000000	N/A	Disabled
584	368	csrss.exe	0x24a0598	9	326	0	False	2012-07-22 02:42:32.000000	N/A	Disabled
368	4	smss.exe	0x24f1020	3	19	N/A	False	2012-07-22 02:42:31.000000	N/A	Disabled
1004	652	svchost.exe	0x25001d0	64	1118	0	False	2012-07-22 02:42:33.000000	N/A	Disabled
824	652	svchost.exe	0x2511360	20	194	0	False	2012-07-22 02:42:33.000000	N/A	Disabled
4	0	System	0x25c89c8	53	240	N/A	False	N/A	N/A	Disabled

Exercise.

For case 1

Most things were straightforward in answering , we just had to use the required plugins however to find the user agent requires a special command which could be confusing.

```
thmanalyst@ubuntu:/Scenarios/Investigations$ strings /Scenarios/Investigations/"Investigation-1.vmem" | grep -i "user-agent"
User-Agent: Mozilla/5.0 (Windows; U; MSIE 7.0; Windows NT 6.0; en-US)
User-Agent: RPC
User-Agent: RPC
User-Agent: Mozilla/5.0 (Windows; U; MSIE 7.0; Windows NT 6.0; en-US)
User-Agent
user-agent
USER-AGENT:
User-Agent:
:cs(User-Agent)
User-Agent
User-Agent
User-Agent
thmanalyst@ubuntu:/Scenarios/Investigations$
```

Case2

In addition, there were outside OSINT that were required to find some of the solutions. Because we were dealing with a suspicious process called **@WanaDecryptor@**, OSINT was required. Some notable sources were virus total and blogs to find the correct mutex for the known indicator of the malware.

(MsWinZonesCacheCounterMutexA0x0)

One important thing to note is that this suspicious process is part of the famous malware **wannacry**. Finally to summarize, it's important to recognize the suspicious pid and parent process PID, so that we can trace and link things up.

Velociraptor

In this room, we will explore Rapid7's newly acquired tool known as [Velociraptor](#).

Per the official Velociraptor [documentation](#), "*Velociraptor is a unique, advanced open-source endpoint monitoring, digital forensic and cyber response platform. It was developed by Digital Forensic and Incident Response (DFIR) professionals who needed a powerful and efficient way to hunt for specific artifacts and monitor activities across fleets of endpoints. Velociraptor provides you with the ability to more effectively respond to a wide range of digital forensic and cyber incident response investigations and data breaches*".

This tool was created by Mike Cohen, a former Google employee who worked on tools such as [GRR](#) (GRR Rapid Response) and [Rekall](#) (Rekall Memory Forensic Framework). Mike joined Rapid7's Detection and Response Team and continues to work on improving Velociraptor. At the date of this entry, the latest release for Velociraptor is [0.6.3](#).

Learning Objectives

- Learn what is Velociraptor
- Learn how to interact with agents and create collections
- Learn how to interact with the virtual file system
- Learn what is VQL and how to create basic queries
- Use Velociraptor to perform a basic hunt

ct tryhackme@thm-velociraptor: ~

```
config.yaml velociraptor velociraptor.config.yaml velociraptor-v0.5.8-linux-amd64
ct tryhackme@thm-velociraptor: ~$ ./velociraptor-v0.5.8-linux-amd64 --config velociraptor.config.yaml frontend -v
2023-08-28T17:19:21Z [INFO] Starting Velociraptor
2023-08-28T17:19:21Z [INFO] This is Velociraptor 0.5.8 built on 2021-04-11T22:11:10Z (e468f54c)
2023-08-28T17:19:21Z [INFO] Loading config from file velociraptor.yaml
2023-08-28T17:19:21Z [INFO] Starting Frontend. ("build_time": "operable program or batch file.

2023-08-28T17:19:21Z [INFO] Error increasing limit invalid argc:Users\Administrator> cd C:
2023-08-28T17:19:21Z [INFO] Starting Journal service. The system cannot find the path specified.
2023-08-28T17:19:21Z [INFO] Starting the notification service.
2023-08-28T17:19:21Z [INFO] Starting Inventory Service C:Users\Administrator>cd "C:\Program Files\Velociraptor"
2023-08-28T17:19:22Z [INFO] Loaded 250 built in artifacts in 1s
2023-08-28T17:19:21Z [INFO] Starting Label service. C:\Program Files\Velociraptor\velociraptor-v0.5.8-windows-amd64.exe --config velociraptor.config.yaml client -v
2023-08-28T17:19:22Z [INFO] Selected frontend configuration location: C:\Program Files\Velociraptor\velociraptor-v0.5.8-windows-amd64.exe --config velociraptor.config.yaml client -v
2023-08-28T17:19:22Z [INFO] Starting Client Monitoring Service [INFO] 2023-08-28T17:31:13Z
2023-08-28T17:19:22Z [INFO] Reloading client monitoring tables [INFO] 2023-08-28T17:31:13Z
2023-08-28T17:19:22Z [INFO] Starting Hunt Dispatcher Service. [INFO] 2023-08-28T17:31:13Z
2023-08-28T17:19:22Z [INFO] Starting the hunt manager service. [INFO] 2023-08-28T17:31:13Z
2023-08-28T17:19:22Z [INFO] server monitoring: Starting Server [INFO] 2023-08-28T17:31:13Z
2023-08-28T17:19:22Z [INFO] Closing Server Monitoring Event loop [INFO] 2023-08-28T17:31:13Z Digging deeper! https://www.velocidex.com
2023-08-28T17:19:22Z [INFO] server monitoring: Updating monitor [INFO] 2023-08-28T17:31:13Z This is Velociraptor 0.5.8 built on 2021-04-11T22:11:10Z (e468f54c)
2023-08-28T17:19:22Z [INFO] Starting Enrollment service. [INFO] 2023-08-28T17:31:13Z Loading config from file velociraptor.config.yaml
2023-08-28T17:19:22Z [INFO] server monitoring: Collecting Server [INFO] 2023-08-28T17:31:13Z Loading writeback from C:\Program Files\Velociraptor\velociraptor.writeback.yaml
2023-08-28T17:19:22Z [INFO] Starting VFS writing service. [INFO] 2023-08-28T17:31:13Z Setting temp directory to C:\Program Files\Velociraptor\Tools
```

Velociraptor

Search clients

State	FlowId	Artifacts	Created	Last Active	Creator	Mb	Rows
✓	F CJMGI02JVI10M	Generic.Client.Info	2023-08-28 20:53:52 UTC	2023-08-28 20:53:54 UTC		8	

Artifact Collection Uploaded Files Requests Results Log Notebook

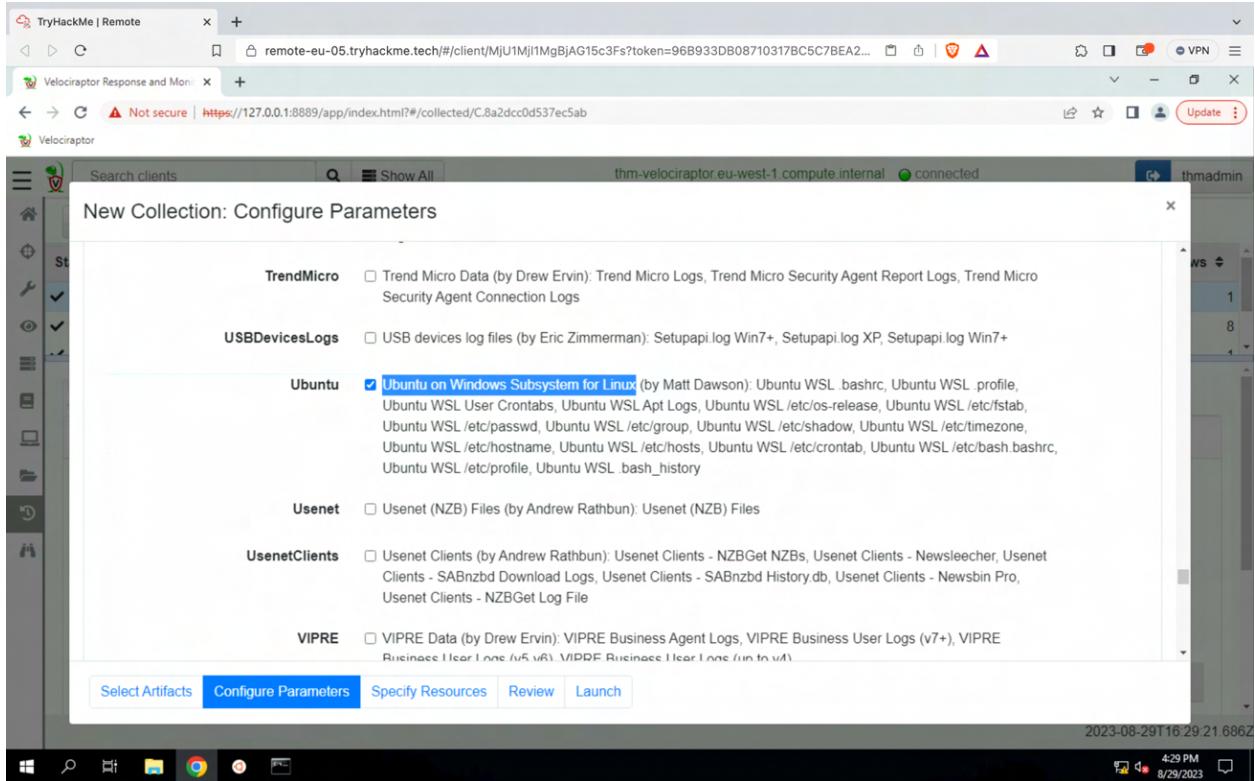
Request sent to client

```
14     "max_row": "1000"
15   },
16   {
17     "query_id": "1",
18     "task_id": "1693256032508714",
19     "VQLClientAction": {
20       "precondition": "SELECT OS From info() where OS = 'windows'",
21       "Query": [
22         {
23           "VQL": "LET precondition_Generic_Client_Info_Users_0=SELECT OS FROM info() WHERE OS = 'windows'"
24         },
25         {
26           "VQL": "LET Generic_Client_Info_Users_0=SELECT Name, Description, Htme AS Lastlogin FROM Artifact.Windows.Sys.Users()"
27         },
28       ],
29       "Name": "$4b3003c65600da34ca73094cf23e1a4c63c719330d3604ea3cad424f6e2cf7f",
30       "VQL": "SELECT * FROM if_then=Generic_Client_Info_Users_0, condition=precondition_Generic_Client_Info_Users_0, else= ( SELECT * FROM scope() WHERE log(message='Query skip'))"
31     }
32   }
```

Windows.sys.Users()

1 of 1

9:15 PM 8/28/2023



The VFS :: Velociraptor - Deployment :: Velociraptor - Deployment :: Velociraptor - TryHackMe | Learning Paths + Tue Aug 29 10:07 AM

BETA

Velociraptor

VFS accessors

The top level directory in the VFS tree view represents the **accessor**. An accessor is simply a dedicated code used to fetch filesystem information from the endpoint.

The **file** accessor simply uses the OS's APIs to list files or directories and fetch data. The **ntfs** accessor uses Velociraptor's built in NTFS parser to be able to access hidden NTFS files and Alternate Data Streams (ADS).

Similarly the **registry** accessor provides file like access to the registry.

Registry Accessor

Interactively investigating an endpoint

Although the VFS presents a familiar interface, it is not ideal for quickly finding the files and registry keys we are usually interested in. One would need to know exactly which files are of interest and then click over multiple

The screenshot shows a Brave browser window with multiple tabs open. The active tab displays a search interface for filenames. A modal window titled "Raw Response JSON" shows the following JSON data:

```

1+ [
2+   {
3+     "Name": "ChromeSetup.exe",
4+     "ModTime": "2020-05-31T17:38:42Z",
5+     "FullPath": "C:\\Users\\mike\\Downloads\\ChromeSetup.exe",
6+     "Mtime": "2020-05-31T17:38:42Z",
7+     "Btime": "2021-01-19T08:52:33.1732915Z",
8+     "Ctime": "2020-05-31T17:38:42Z",
9+     "Atime": "2021-01-22T14:23:00.5879832Z",
10+    "Data": {},
11+    "Size": 1995576,
12+    "IsDir": "false",
13+    "IsLink": "false",
14+    "Mode": 438,
15+    "Sys": {
16+      "FileAttributes": 32,
17+      "CreationTime": {
18+        "LowDateTime": 1832406963,
19+        "HighDateTime": 30862912
20+      },
21+      "LastAccessTime": {
22+        "LowDateTime": 1737614000000000000
23+      }
24+    }
25+
26+  }
27+
28+ ]
  
```

The modal has a "Close" button at the bottom right. Below the modal, the text "Glob output" is visible.

Some of the more important columns available are

<https://docs.velociraptor.app/docs/forensic/filesystem/image12.png>

The screenshot shows a Firefox browser window with the URL <https://127.0.0.1:8889/app/index.html#/notebooks/N.CJOGPUACOD69C>. The page displays a table of notebook details and a detailed view of a file's PE header.

NotebookId	Name	Description	Creation Time	Modified Time	Creator	Collaborators
N.CJOGPUACOD69C	notebook	notebook	2023-08-31 21:58:17 UTC	2023-08-31 21:58:17 UTC	admin	admin

Below the table, a file detail view for "FXDRV.DLL" from "C:/Windows/System32/spool/drivers/x64/3/FXSDRV.DLL" is shown. The "PE" section is expanded, displaying:

```

{
  "FileHeader": {...},
  "GUIDAge": "CE127918A8E41A19888BBB312F6806E51",
  "PDB": "FXDRV.pdb",
  "Sections": [...],
  "VersionInformation": {...}
}
  
```

The timestamp for this detail view is 2023-08-31 22:08:27 UTC (71s).

```
SELECT "C:/" + FullPath AS Full_Path,FileName AS  
File_Name,parse_pe(file="C:/" + FullPath) AS PE  
  
FROM parse_mft(filename="C:/$MFT", accessor="ntfs")  
  
WHERE NOT IsDir  
  
AND FullPath =~ "Windows/System32/spool/drivers"  
  
AND PE
```

Utilized VQL Query to find a suspicious DLL.

Malware Analysis

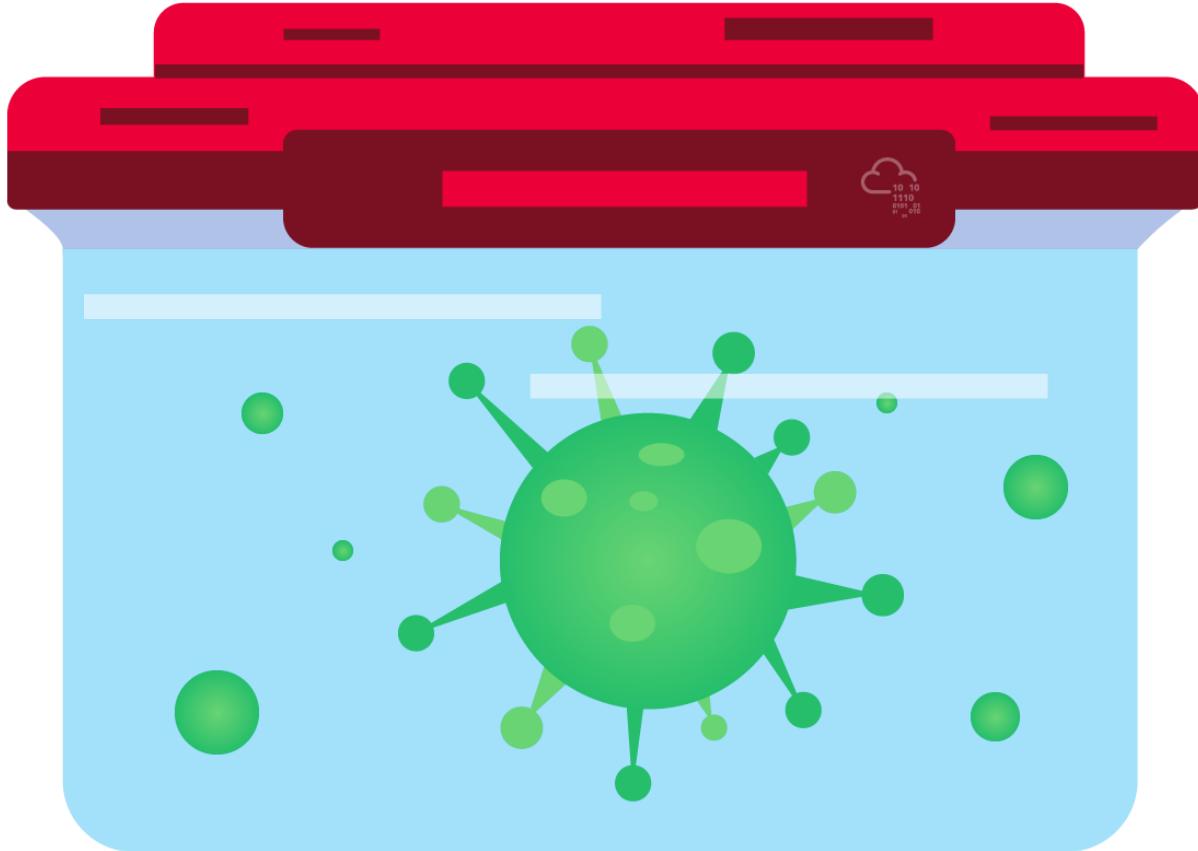
- Never analyze malware or suspected malware on a machine that does not have the sole purpose of analyzing malware.
- When not analyzing or moving malware samples around to different locations, always keep them in password-protected zip/rar or other archives so that we can avoid accidental detonation.
- Only extract the malware from this password-protected archive inside the isolated environment, and only when analyzing it.
- Create an isolated VM specifically for malware analysis, which has the capability of being reverted to a clean slate once you are done.
- Ensure that all internet connections are closed or at least monitored.
- Once you are done with malware analysis, revert the VM to its clean slate for the next malware analysis session to avoid residue from a previous malware execution corrupting the next one.

Malware Analysis is like solving a puzzle. Different tools and techniques are used to find the pieces of this puzzle, and joining those pieces gives us the complete picture of what the malware is trying to do. Most of the time, you will have an executable file (also called a binary or a PE file. PE stands for Portable Executable), a malicious document file, or a Network Packet Capture (Pcap). The Portable Executable is the most prevalent type of file analyzed while performing Malware Analysis.

To find the different puzzle pieces, you will often use various tools, tricks, and shortcuts. These techniques can be grouped into the following two categories:

- Static Analysis
- Dynamic Analysis

Static Analysis

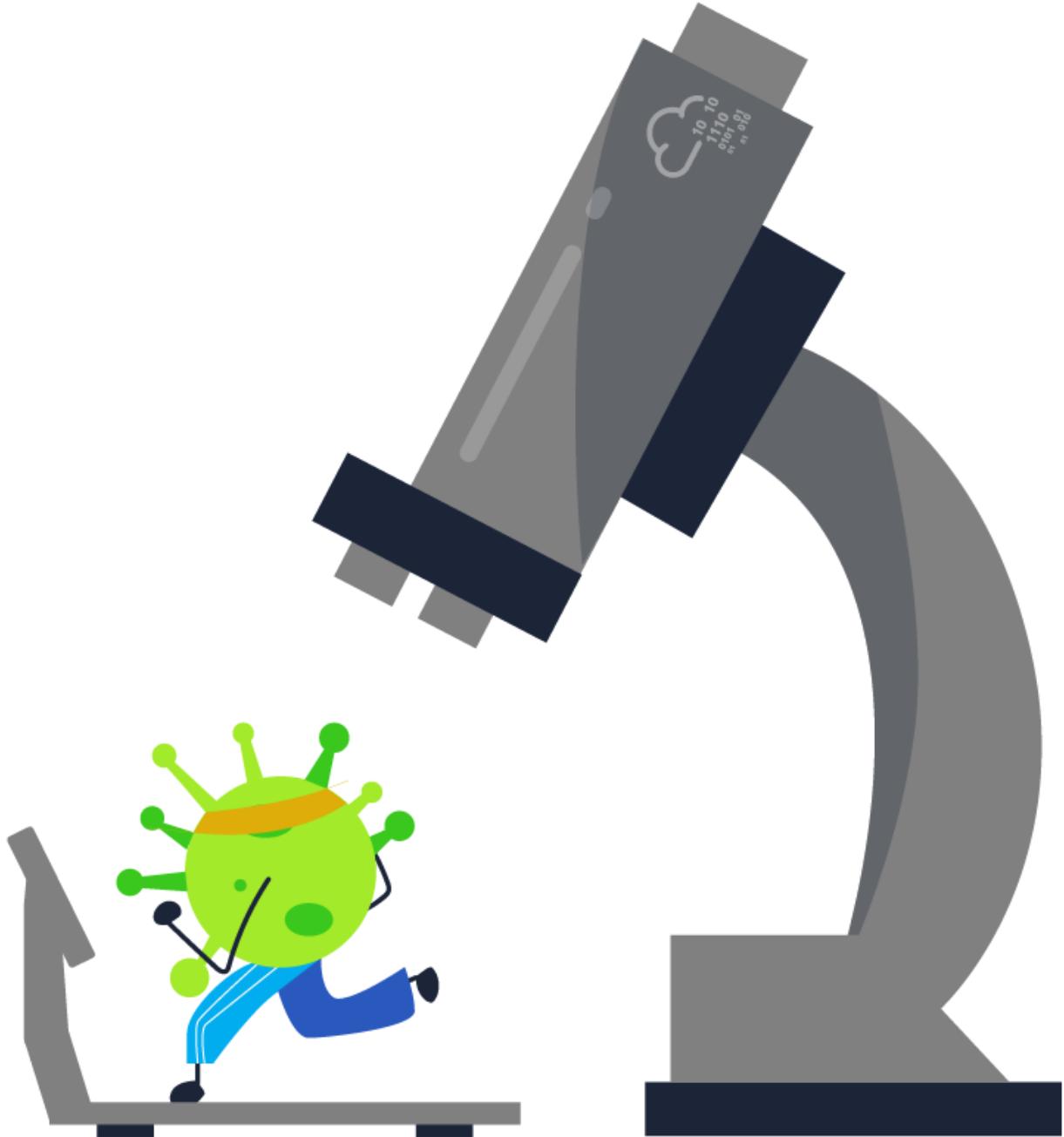


When malware is analyzed without being executed, it is called Static Analysis. In this case, the different properties of the PE file are analyzed without running it. Similarly, in the case of a malicious document, exploring the document's properties without analyzing it will be considered Static Analysis. Examples of static analysis include checking for strings in malware, checking the PE header for information related to different sections, or looking at the code using a disassemble. We will look at some of these techniques later in the room.

Malware often uses techniques to avoid static analysis. Some of these techniques use obfuscation, packing, or other means of hiding its properties. To circumvent these techniques, we often use dynamic analysis.

Dynamic Analysis

Malware faces a dilemma. It has to execute to fulfill its purpose, and no matter how much obfuscation is added to the code, it becomes an easy target for detection once it runs.



Static analysis might provide us with crucial information regarding malware, but sometimes that is not enough. We might need to run the malware in a controlled environment to observe what it does in these cases. Malware can often hide its properties to thwart Static Analysis. However, in most of those cases, Dynamic Analysis can prove fruitful. Dynamic analysis techniques include running the malware in a VM, either in a manual fashion with tools installed to monitor the

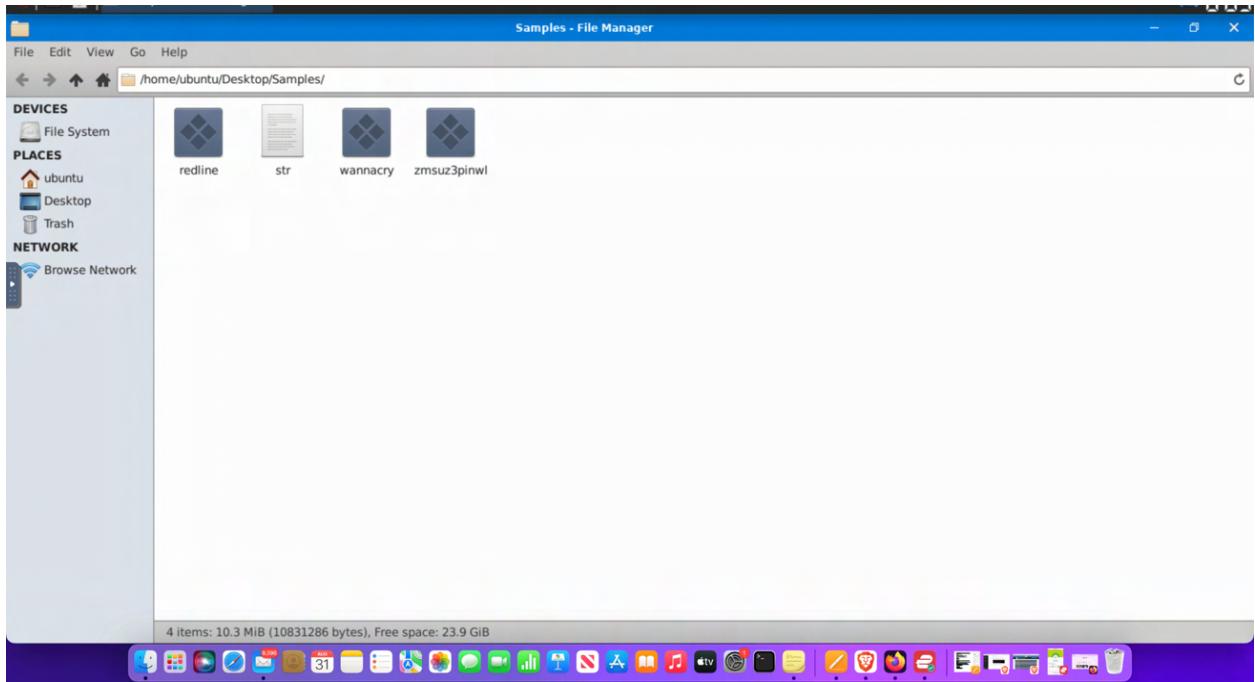
malware's activity or in the form of sandboxes that perform this task automatically. We will learn about some of these techniques later in this room. Once we run the malware in a controlled environment, we can use our knowledge from the Windows Forensics rooms to identify what it did in our environment. The advantage here is that since we control the environment, we can configure it to avoid noise, like activity from a legitimate user or Windows Services. Thus, everything we observe in such an environment points to malware activity, making it easier to identify what the malware did in this scenario.

Malware, however, often uses techniques to prevent an analyst from performing dynamic analysis. Since most dynamic analysis is performed in a controlled environment, most methods to bypass dynamic analysis include detecting the environment in which it is being run. Therefore, in these cases, the malware uses a different, benign code path if it identifies that it is being run in a controlled environment.

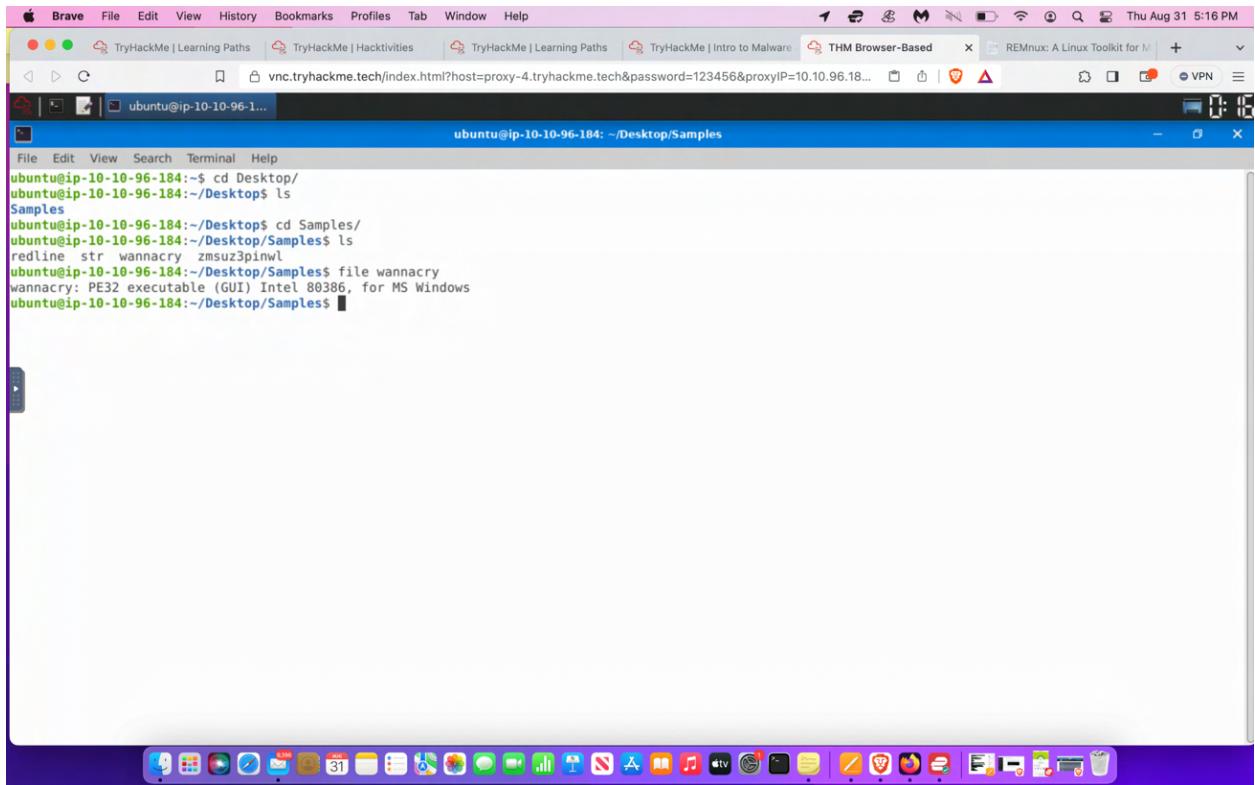
Advanced Malware Analysis

Advanced Malware Analysis Techniques

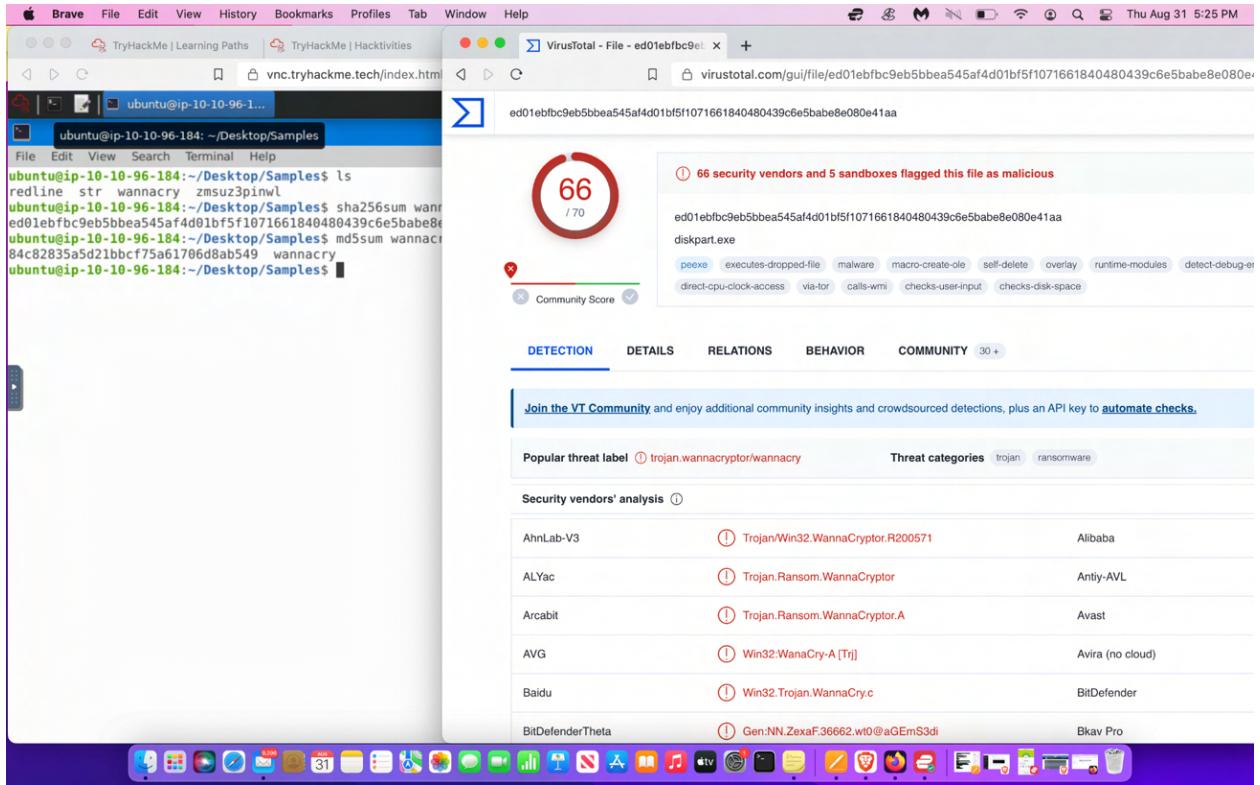
- Objective: Analyze malware that evades basic static and dynamic analysis.
- Tools: Disassemblers and debuggers.
- Disassemblers: Convert malware code from binary to assembly language.
 - Allows static examination of malware instructions.
- Debuggers: Attach to a program and monitor malware instructions during runtime.
 - Enables starting and stopping malware at different points.
 - Identifies crucial information.
 - Provides insights into system memory and CPU utilization.



Remnux the linux distribution used for malware analysis.

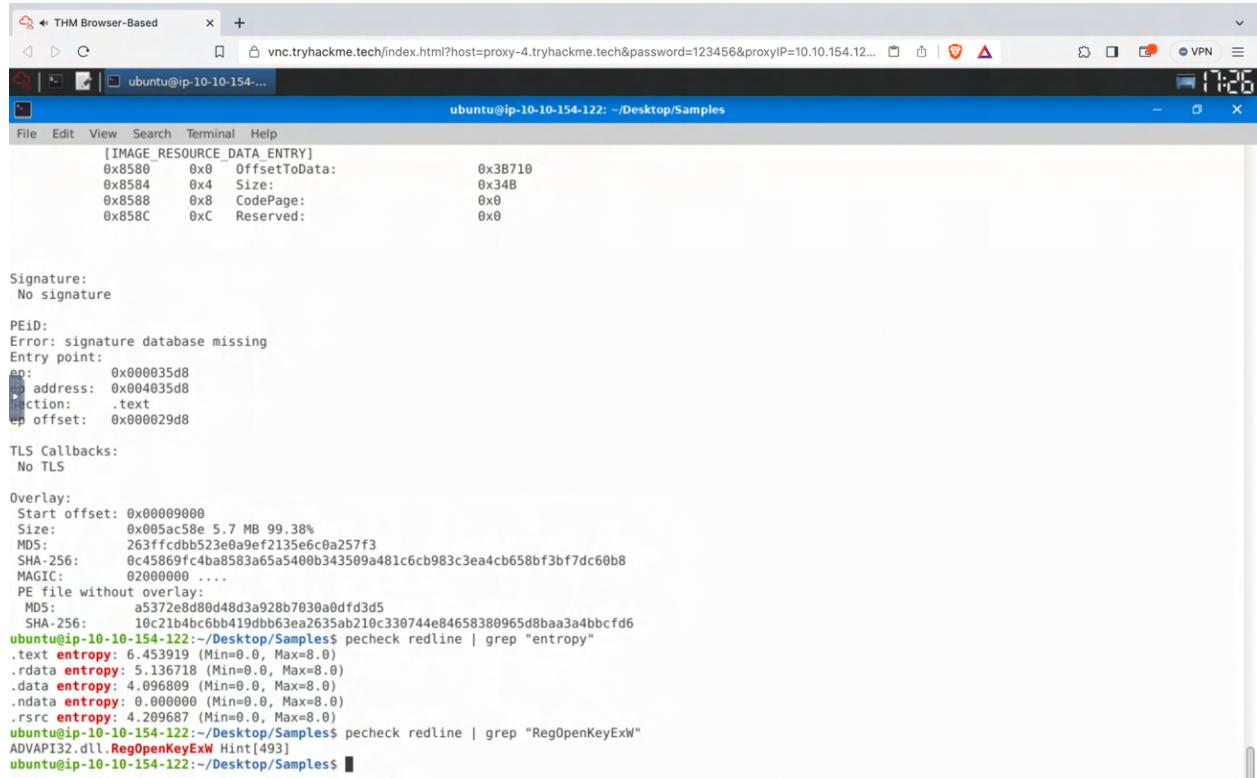


Details for the famous wannacry malware present in samples directory. uses the X-86. It is a PE32 Executable GUI for windows.



Calculate md5 or sha256 hash and look up the hash on virus total.

Analysed the PE header using the pecheck command to find various sections such as .text , .data, .rdata, .ndata, .rsrs



```

ubuntu@ip-10-10-154-122: ~/Desktop/Samples$ pecheck redline
[IMAGE RESOURCE DATA_ENTRY]
0x8580 0x0 OffsetToData: 0x3B710
0x8584 0x4 Size: 0x34B
0x8588 0x8 CodePage: 0x0
0x858C 0xC Reserved: 0x0

Signature:
No signature

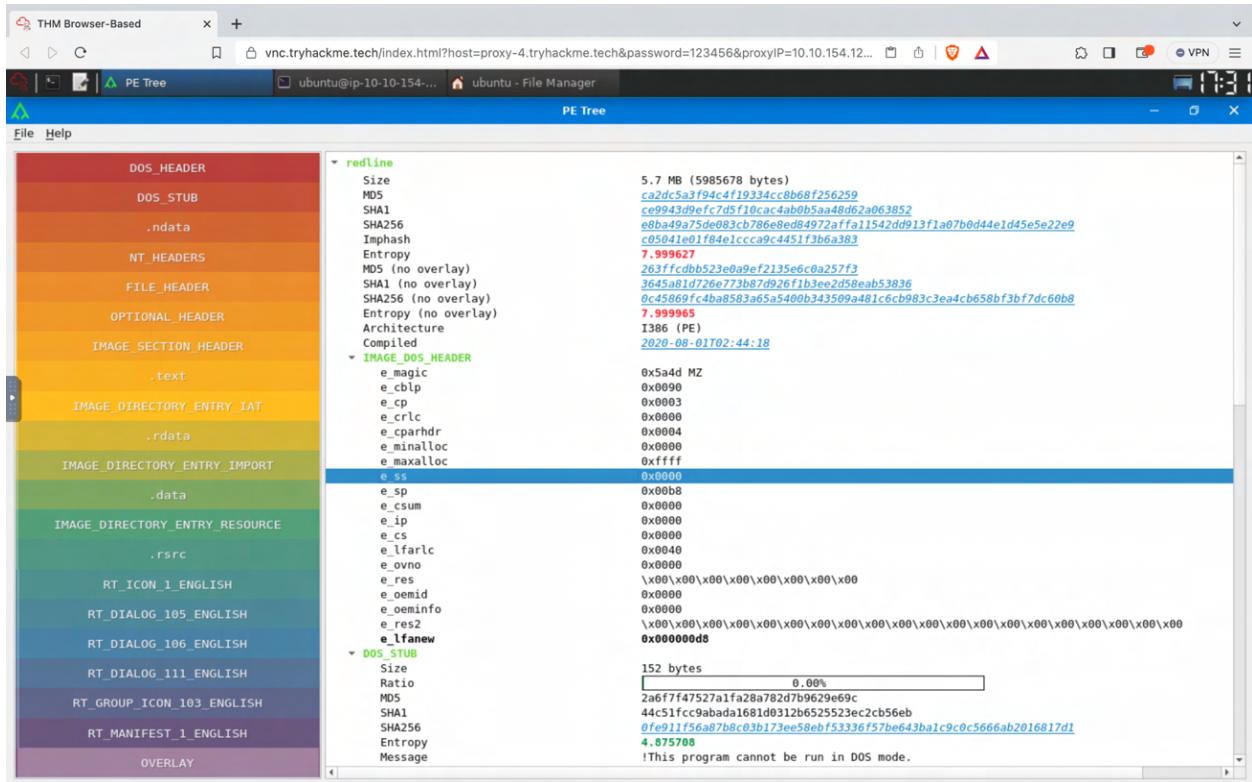
PEid:
Error: signature database missing
Entry point:
  ep: 0x000035d8
  ip address: 0x004035d8
  section: .text
  ep offset: 0x000029d8

TLS Callbacks:
No TLS

Overlay:
Start offset: 0x00009000
Size: 0x005ac58e 5.7 MB 99.38%
MD5: 263ffcd8b523e0a9ef2135e6c0a257f3
SHA-256: 0c45869fc4ba8583af5a540b0343509a481c6cb983c3ea4cb658bf3bf7dc60b8
MAGIC: 02000000 ...
PE file without overlay:
  MD5: a5372e8d80dd48d3a928b7030a0dfd3d5
  SHA-256: 10c21b4bc6bb419dbb63ea2635ab210c330744e84658380965d8baa3a4bbcf0
ubuntu@ip-10-10-154-122:~/Desktop/Samples$ pecheck redline | grep "entropy"
.text entropy: 6.453919 (Min=0.0, Max=8.0)
.rdata entropy: 5.136718 (Min=0.0, Max=8.0)
.data entropy: 4.096889 (Min=0.0, Max=8.0)
.ndata entropy: 0.000000 (Min=0.0, Max=8.0)
.rsrc entropy: 4.209687 (Min=0.0, Max=8.0)
ubuntu@ip-10-10-154-122:~/Desktop/Samples$ pecheck redline | grep "RegOpenKeyExW"
ADVAPI32.dll.RegOpenKeyExW Hint[493]
ubuntu@ip-10-10-154-122:~/Desktop/Samples$ 
```

Executed the pe-tree command on the terminal `pe-tree redline`

for the redline sample given to view a GUI based interface in a more readable format. Ultimately this contains metadata which can be helpful for our malware analysis.



Dynamic Analysis

Uploaded samples on hybrid analysis to analyze the report given for the redline sample. This was done my using the md5 hash as opposed to uploading an actual sample.

The screenshot shows a web browser displaying the Hybrid Analysis report for a sample identified as 'redline'. The report is labeled as 'malicious' with a threat score of 100/100 and 62% AV detection. It includes sections for Risk Assessment, Incident Response, and Related Sandbox Artifacts. The Risk Assessment section details various malicious behaviors observed by the malware, such as spyware, persistence, fingerprinting, and evasion techniques. The Incident Response section provides links for sharing the report. The Related Sandbox Artifacts section lists various analysis tools and reports generated during the analysis process. The bottom of the page features a purple navigation bar with various application icons.

This report is generated from a file or URL submitted to this webservice on December 9th 2022 22:05:30 (UTC)

Guest System: Windows 10 64 bit, Professional, 10.0 (build 16299).

Report generated by [Falcon Sandbox](#) v9.5.1 © Hybrid Analysis

[🔗 Overview](#) [🔗 Sample unavailable](#) [🔗 Downloads](#) [🔗 External Reports](#) [🔗 Re-analyze](#) [🔗 Hash Seen Before](#) [🔗 Show Similar Samples](#)

[🔗 Report False-Positive](#) [🔗 Request Report Deletion](#)

malicious

Threat Score: 100/100
AV Detection: 62%

Labeled as: Win/malicious_confidence_100%
#evasive #pua #soclars #stealer

[🔗 Link](#) [🔗 Twitter](#) [🔗 E-Mail](#)

Incident Response
[Related Sandbox Artifacts](#)

Indicators
CrowdStrike AI
File Details
Screenshots (3)
Hybrid Analysis (50)
Network Analysis
Extracted Strings
Extracted Files (35)
Notifications
Community (0)

[Back to top](#)

Risk Assessment

Spyware Found browser information locations related strings
Hooks API calls
POSTs files to a webserver
Sets a computer-based training (CBT) hook
Tries to steal browser sensitive information (file access)

Persistence Installs hooks/patches the running process
Spawns a lot of processes
Writes data to a remote process

Fingerprint Queries kernel debugger information
Queries process information
Queries sensitive IE security settings
Queries the display settings of system associated file extensions
Queries the internet cache settings (often used to hide footprints in index.dat or internet cache)
Reads the windows installation language
Tries to identify its external IP address

Evasive Contains ability to adjust token privileges
Contains ability to check if a debugger is running

The screenshot shows the Hybrid Analysis interface with the following details:

- Contacted Hosts:**

IP Address	Port/Protocol	Associated Process	Details
149.28.253.196	443	62237fa56f568_sat15e879a10c5.exe	United States PID: 2488
208.95.112.1	80	62237fa56f568_sat15e879a10c5.exe	Reserved PID: 3144
8.25.82.208	80	62237fa56f568_sat15e879a10c5.exe	United States PID: 2488
151.15.10.1	80	62237fa56f568_sat159fc8bb76b4.tmp	United Kingdom PID: 68444
148.251.234.83	443	62237fa56f568_sat15e879a10c5.exe	Germany PID: 2488
23.59.204.217	80	62237fa56f568_sat15e879a10c5.exe	United States PID: 2488
45.66.159.18	80	62237fa56f568_sat15e879a10c5.exe	Russian Federation PID: 2488
- Contacted Countries:** A world map showing connections from various countries to the analyzed host.
- Network Analysis:** A sidebar with links to DNS Requests (17), Contacted Hosts (10), Contacted Countries, HTTP Traffic (22), Memory Forensics (1), Network Analysis (50), Extracted Strings, Extracted Files (35), Notifications, and Community (0).
- Bottom Bar:** A Mac-style dock with various application icons.

Contacted hosts findings in the network analysis of the incidence response.

Summary

Malware analysis. However, this was just scratching the surface.

- Static and Dynamic analysis of malware
- Finding strings, calculating hashes, and running AV scans on malware
- Introduction to the PE header and how to use information from it in malware analysis
- Sandboxing and different online sandboxes that we can use
- How malware evades the techniques we just discussed.

PHISHING

Email Protocols and Message Flow

Email Protocols:

- Email transactions involve specific protocols designed for network-related tasks.
- Three main protocols facilitate outgoing and incoming email messages.

SMTP (Simple Mail Transfer Protocol):

- Handles the sending of emails.

POP3 (Post Office Protocol):

- Responsible for transferring email between a client and a mail server.
- Emails are downloaded and stored on a single device.
- Sent messages are stored on the same device.
- Accessible only from the device where emails were downloaded.
- Enable "Keep email on server" to retain messages on the server.

IMAP (Internet Message Access Protocol):

- Also responsible for transferring email between a client and a mail server.
- Emails are stored on the server and can be downloaded to multiple devices.
- Sent messages are stored on the server.
- Messages can be synced and accessed across multiple devices.

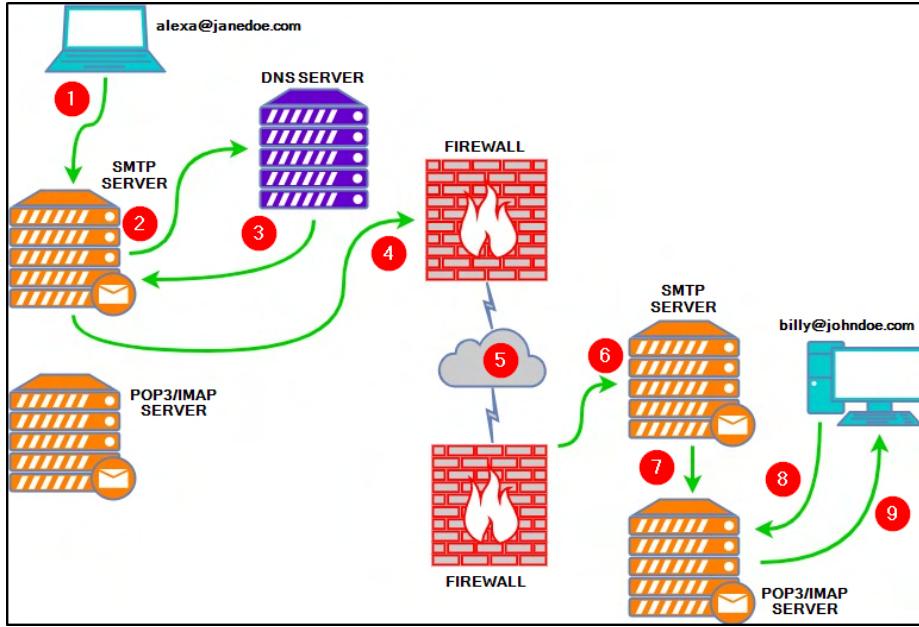
Email Transmission Steps:

Below is an explanation of each numbered point from the above diagram:

1. Alexa composes an email to Billy (billy@johndoe.com) in her favorite email client. After she's done, she hits the send button.

- 2 . The **SMTP** server needs to determine where to send Alexa's email. It queries **DNS** for information associated with johndoe.com.
- 3 . The **DNS** server obtains the information johndoe.com and sends that information to the **SMTP** server.
- 4 . The **SMTP** server sends Alexa's email across the Internet to Billy's mailbox at johndoe.com.
- 5 . In this stage, Alexa's email passes through various **SMTP** servers and is finally relayed to the destination **SMTP** server.
- 6 . Alexa's email finally reached the destination **SMTP** server.
- 7 . Alexa's email is forwarded and is now sitting in the local **POP3/IMAP** server waiting for Billy.
- 8 . Billy logs into his email client, which queries the local **POP3/IMAP** server for new emails in his mailbox.
- 9 . Alexa's email is copied (**IMAP**) or downloaded (**POP3**) to Billy's email client.

Lastly, each protocol has its associated default ports and recommended ports. For example, **SMTP** is port 25.



Email Body

So far everything has been straightforward . One important thing to note is that contents within a pdf file that are encoded from base64 data have to be decoded in human readable format. One notable for example is a email , which consists of a pdf file. Since the base 64 data is very lengthy , we simply just copied the contents and pasted in in cyberchef.

The screenshot shows the CyberChef interface with the 'From Base64' recipe selected. The input is a long base64 encoded string, and the output is a PDF document titled 'download.pdf' containing the decoded content.

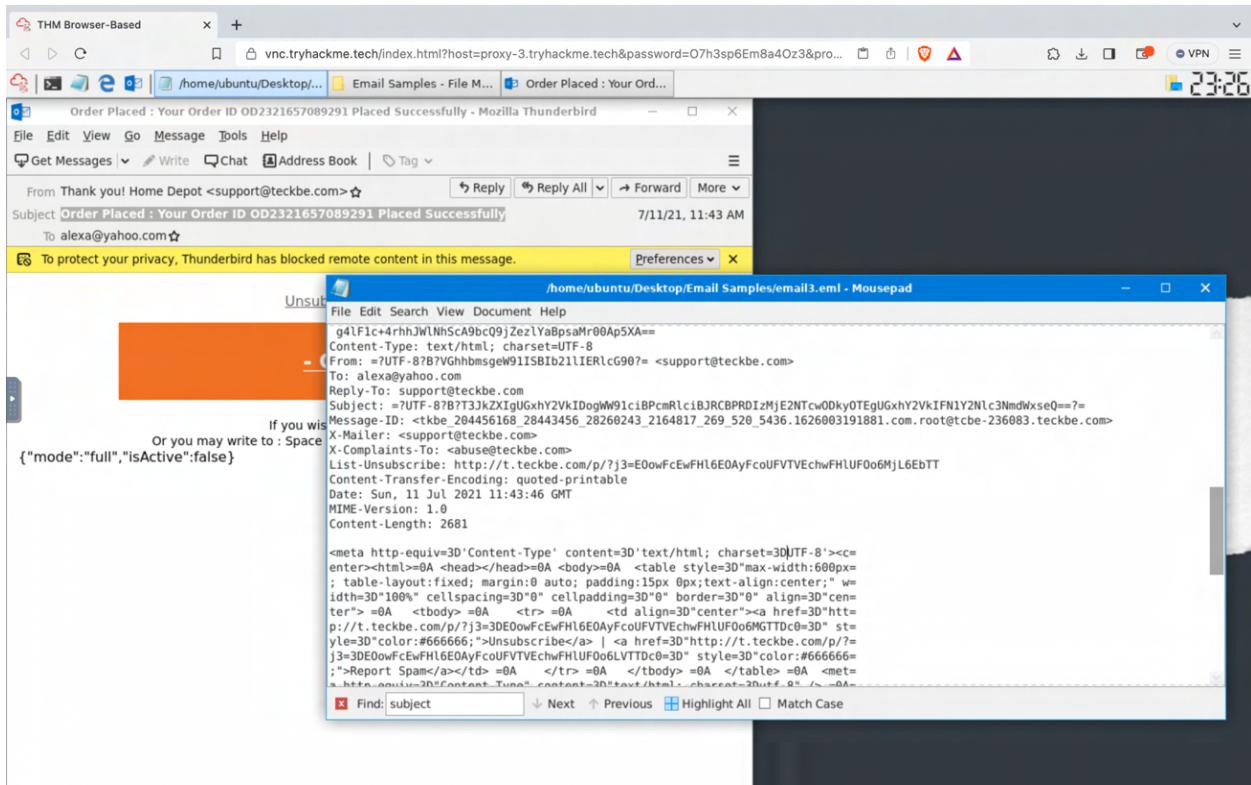
Then we selected the from base64 recipe and dragged it, and bake. We then saved the output to a pdf file .

The screenshot shows the Mac OS X Finder window displaying the 'download.pdf' file in the 'Downloads' folder. The file is a PDF document.

Once we saved it to a pdf file, then we should be able to view the BASE64 contents decoded of the pdf file

The screenshot shows a PDF viewer displaying the 'download.pdf' file. The content of the PDF is a single page with the text: 'THIRTEEN_PDF_ATTACHMENT'.

EMAIL ANALYSIS PHISING



From the given samples we can find some useful information given the email headers. We can find the sender's address which is stated in the above screen shot. A way to find other valuable information such as which entity the sender has masqueraded as or in other words spoofed, we can use the given platform to open the email sample. From the analysis we can see in the email that the sender has spoofed using home depot. This information apparently was not visible within the sample file. Another factor was subject, that was encoded, we learned that the subject of the email could be retrieved by simply opening the email through the thunderbird application. A link was placed by the sender, which was also retrieved through the thunderbird application.

PHISHING CASE

Used email headers to find relevant information. There are many useful tools one can use to find useful information from a phishing email. The one listed below is from google, called message header. It appears to be user friendly and one can navigate through the findings to for analysis

The screenshot shows the Google Admin Toolbox's "Messageheader" tool. The main pane displays the raw email header information, which includes fields like Received, Return-Path, X-Originating-IP, Received-SPF, Authentication-Results, and DKIM-Signature. Below the header, there is a summary of the message's metadata (Created at, To, Subject) and a detailed breakdown of various security-related fields (SPF, DKIM, DMARC). A scroll bar indicates there is more content below the visible area.

The screenshot shows the ConvertCSV URL Extractor tool. It has a "From list of web pages" input field containing a URL to a phishing page. Below it, there are sections for "Step 2: Choose output options" (with "Extract" selected) and "Step 3: Extract URLs". A large text area displays the extracted URLs, which include various links such as tryhackme.com, [convercsv.com](https://www.convercsv.com), and several internal links within the tryhackme site. At the bottom, there are download and search buttons.

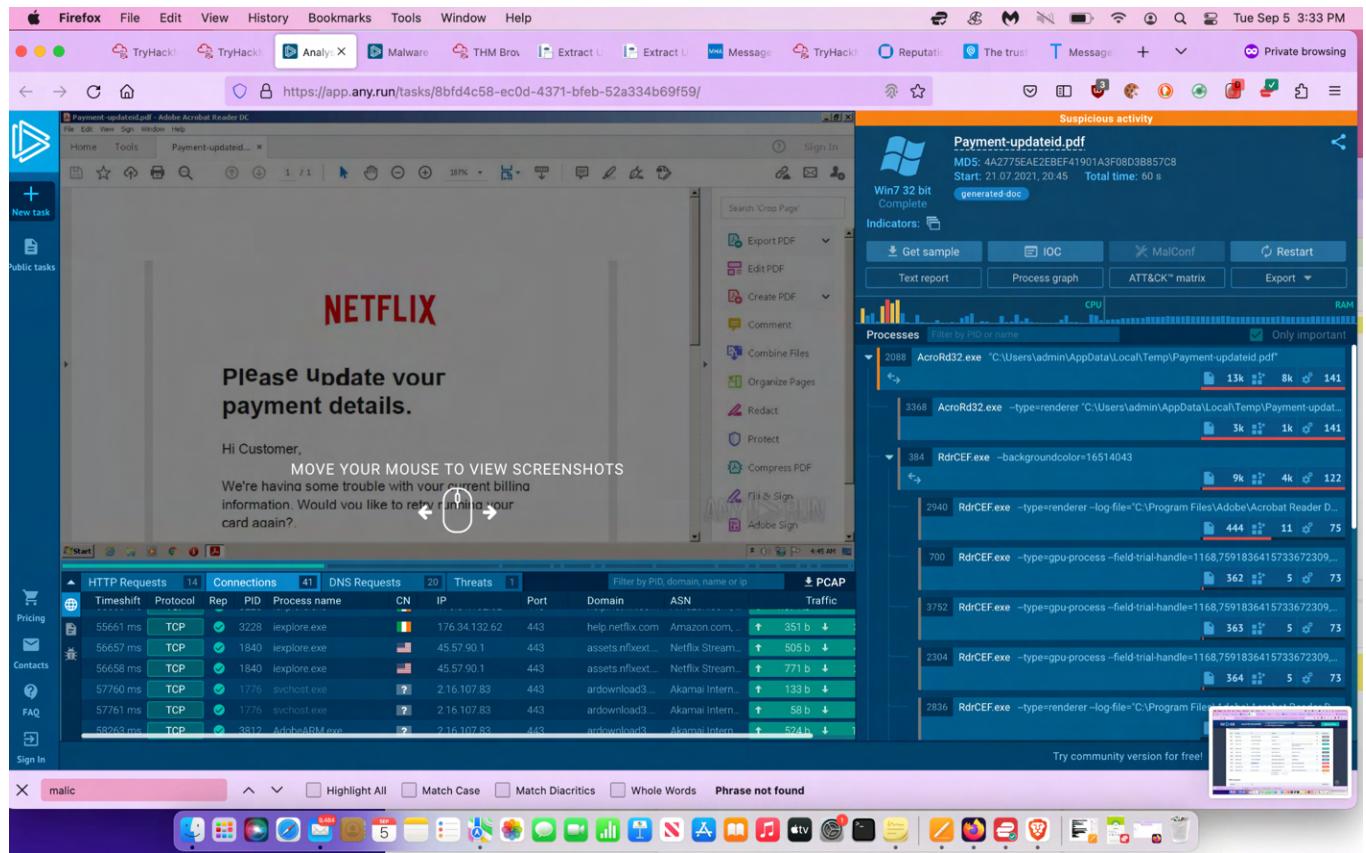
The url extractor is a pretty useful tool for finding all of the urls that are listed. Typically this could also be located in the e-mail body.

7	X-Google-DKIM-Signature	vt=1; ansa-sha256; c=relaxed/relaxed; d=1e100.net; sv=20161025; hr=x-gm-message-state:message-id-date:to:from:subject:mime-version; bh=HLZRRPnbarLrnjKEgNsm2TnkOdazsljRPO2QPMc; b=hPz8lyziahNDXDeDdUrm56t1;p72zlfF2P2k4XmJwbesl2BMZYohh3ek>30eh3 3dECPmIEOnvAegzHrJL1a74mhJhqv2f2Bsg1t9mYIKSEN+xyGGs1gsf1ZGTwkvUlTCkeym6FbzJuQ9k78r++VX1YR40DT/kOFN1CGXitZOKIDHNULxAbpKZMaPsfjZ6PQUsJc6tYyVP7b859WOS9MhAxBAjUyuPTMXZMN1T1k1BESCWQ Qoof1MpcOwhYb6YAJZJ9TrgSsN338Jrn/vutbxgScu+bVIRjOB2j0.U6mWQJls4cvg==
8	X-Gm-Message-State	AOAM533EzimkLM39ewHUBGUjxphifeErIgPuabhfJWWKu2wn4r+yJnjqymCICQhUyp/Zfwp/WbqgvdBtC9jQ8XBsq2S/dspckOQ==
9	X-Google-SMTP-Source	ABdhPJz6YRX0AT7zgRFB2ugWTiJ2l0BYxm4hfbUuawJapAdMqALhHWGCRAl13lZk9h5o99nVEc1
10	X-Received	by 2002.aca.eff06; with SMTP id n6m16727904oh.66.1625624086463; Tue, 06 Jul 2021 19:14:46 -0700 (PDT)
11	X-Relaying-Domain	elektro.ng
12	X-Mailer	Microsoft Office Outlook, Build 11.0.5510
13	X-ContentID	19380
14	X-ContentBase	/Portal/content/DollarGeneral

The message header analysis is also a useful tool to find important information such as domain of interest, submitting host, received host, protocols and IP addresses.

Phishing case 2 lab

In this lab we analyzed a malicious attachment from a phishing email through a online malware sandbox platform for analysis, we can analyze the network and discover other important details. In summary we discovered how the email was classified which was suspicious. We discovered a pdf file, SHA 256 of the pdf file was available. A windows process file was flagged as potentially bad traffic. In addition we can click and view screen shots of the live e-mail, in terms of how a potential victim could be a target.



ANY RUN ANALYZE MALWARE

Connections

PID	Process	IP	Domain	ASN	CN	Reputation
384	RdrCEF.exe	3.233.129.217:443	p13n.adobe.io	—	US	unknown
980	iexplore.exe	35.244.149.249:443	lihi1.com	—	US	unknown
3228	iexplore.exe	93.184.220.29:80	ocsp.digicert.com	MCI Communications Services, Inc. d/b/a Verizon Business	US	whitelisted
3228	iexplore.exe	2.18.232.136:443	help.netflix.com	Akamai International B.V.	—	whitelisted
3228	iexplore.exe	176.34.132.62:443	help.netflix.com	Amazon.com, Inc.	IE	unknown
3228	iexplore.exe	104.16.149.64:443	cdn.cookielaw.org	Cloudflare Inc	US	unknown
3228	iexplore.exe	104.20.184.68:443	geolocation.onetrust.com	Cloudflare Inc	US	shared
1776	svchost.exe	2.16.107.83:443	ardownload3.adobe.com	Akamai International B.V.	—	malicious
3812	AdobeARM.exe	2.16.107.83:443	ardownload3.adobe.com	Akamai International B.V.	—	malicious
1840	iexplore.exe	45.57.90.1:443	assets.netflixext.com	Netflix Streaming Services Inc.	US	suspicious

DNS requests

Domain	IP	Reputation
ardownload3.adobe.com	2.16.107.83	malicious

A helpful feature from the platform was a self generated report one could create . This lists important details such as PID, and ip address. Luckily it was readable and important labels such as malicious IP were available to analyze from the given report.

Phishing Case 3 lab.

For this case , a sandbox with the suspicious email was provided. We retrieved the details , and we discovered a Malicious activity had been classified. From the given report generated, We learned that this time there was an excel file present. We traced 3 malicious domains, along with 3 malicious IP addresses. In addition we discovered the vulnerability that the malicious attachment attempted to exploit. That vulnerability happens to be **CVE-2017-11882**.

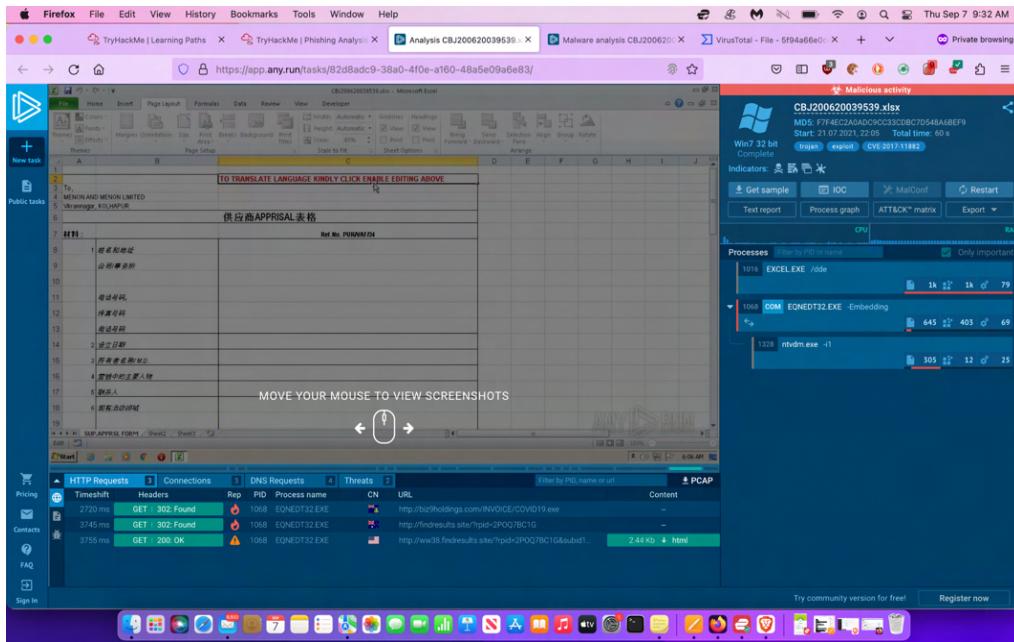
The screenshot shows the ANY.RUN malware analysis interface. At the top, it displays a summary of network activity: 3 HTTP(S) requests, 3 TCP/UDP connections, 4 DNS requests, and 0 Threats. Below this, the "HTTP requests" section lists three entries:

PID	Process	Method	HTTP Code	IP	URL	CN	Type	Size	Reputation
1068	EQNEDT32.EXE	GET	302	204.11.56.48.80	http://biz9holdings.com/INVOICE/COVID19.exe	VG	—	—	malicious
1068	EQNEDT32.EXE	GET	302	103.224.182.251.80	http://findresults.site/?pid=2P0Q7BC1G	AU	—	—	malicious
1068	EQNEDT32.EXE	GET	200	75.2.11.242.80	http://www.findresults.site/?pid=2P0Q7BC1G&subid=20210722-1505-2609-be9-8bc8329e748d	US	html	2.44 Kb	malicious

Below the table, a note says "Download PCAP: analyze network streams, HTTP content and a lot more at the [full report](#)".

The "Connections" section shows the same three network interactions with their respective IP addresses and domain names.

The "DNS requests" section is partially visible at the bottom.



CVE-2017-11882 Detail

Description

Microsoft Office 2007 Service Pack 3, Microsoft Office 2010 Service Pack 2, Microsoft Office 2013 Service Pack 1, and Microsoft Office 2016 allow an attacker to run arbitrary code in the context of the current user by failing to properly handle objects in memory, aka "Microsoft Office Memory Corruption Vulnerability". This CVE ID is unique from CVE-2017-11884.

Severity	CVSS Version 3.x	CVSS Version 2.0
NIST: NVD	Base Score: 7.8 HIGH	Vector: CVSS:3.1/AV:L/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

QUICK INFO

CVE Dictionary Entry: CVE-2017-11882
NVD Published Date: 11/14/2017
NVD Last Modified: 03/16/2021
Source: Microsoft Corporation

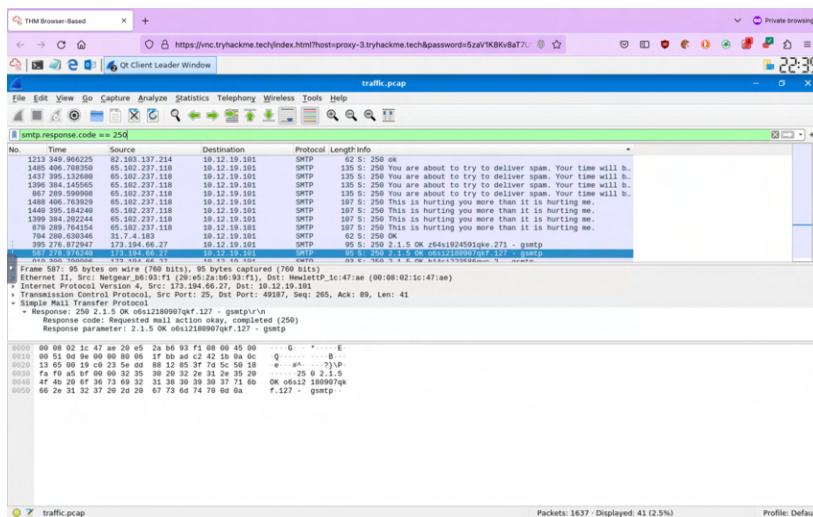
Phishing Prevention

In this lab we explored the various the various methods defenders can take to protect users from falling as a victim. The main focus for the first portion was email security using SPF, DKIM, DMARC. We explored mitigations as per the MITRE ATTA&CK Framework in terms of tricking targets.

A notable concept we explored was S/MIME which is an important concept learned from our SECURITY+ studies. We briefly touched on digital signatures and encryption, not ignoring the importance of public key cryptography.

SMTP STATUS CODES

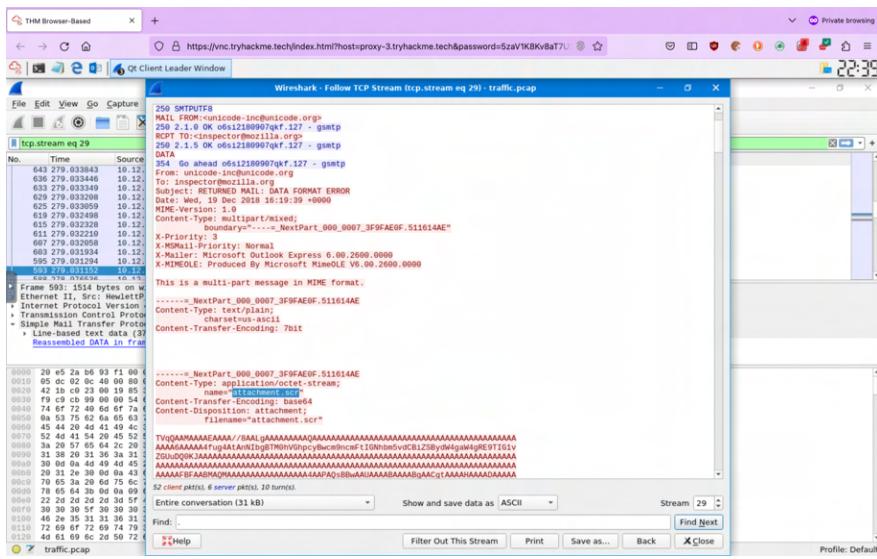
We proceeded on to the lab portion which involved SMTP status codes. We deployed wireshark, and completed the given scenario. Most of the portions were straightforward because we had our trusty wireshark documentation to guide us when filtering to analyze the given packet capture.



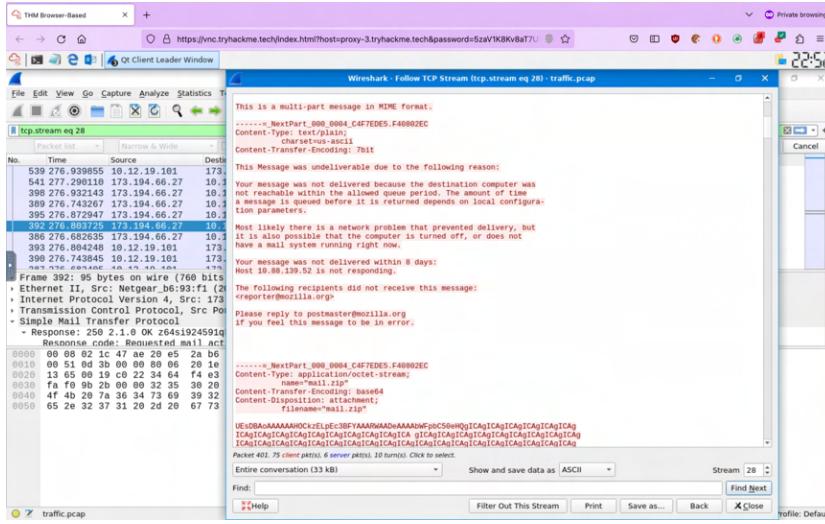
SMTP TRAFFIC ANALYSIS

In this portion of the lab we again explored packet capture analysis using wireshark on a virtual machine. Again most of it was straight forward using the given wire shark documentation. However, during our analysis we discovered some interesting findings which i will share.

During our analysis we discovered an attachment file



During our analysis we discovered an attachment file, we simply followed the tcp stream to retrieve the finding. We learned that the port the traffic was using was 25 catered to SMTP. The source ip address was 10.12.19.101 involved for all SMTP traffic. Finally relating to this we discovered the final attachment , which was mail.zip



Conclusion

Finally in conclusion an important aspect for SOC analysts to study was a given phishing IR playbook. The phishing playbook <https://www.incidentresponse.org/playbooks/phishing> summarizes an incident response by the NIST incident process. It covers **Prepare, Detect , Analyze, Contain , Eradicate , Recovery, Post-Incident Handling.**

Lab Scenario:

Here we investigated a E-mail sample via a virtual machine to determine if it was legitimate. The e-mail also contained an attachment . One important thing to note here is that it was a generic email with a generic good day greeting. We analyzed the email headers using thunderbird as the application.

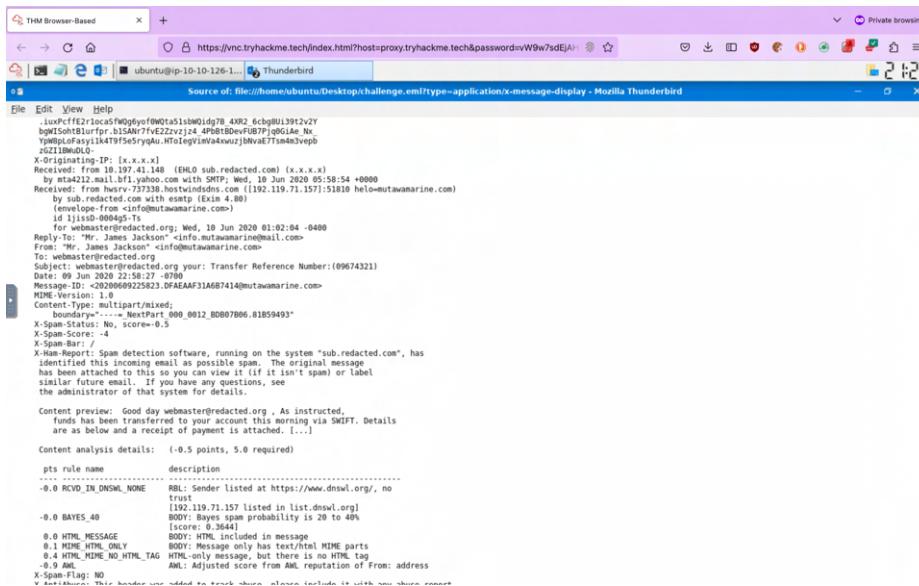
From the analysis we notice the date the e-mail was received, which was 6/10/20.

Email was from Mr James. Jackson

Email address: info@mutawamarine.com

Email address to receive a reply of email: info@mutawamarine.com

Originating ip address: 192.119.71.157



Next during our analysis we needed to run the whois command on our local terminal , so as we went ahead and did that .

Command : whois 192.119.71.157

From the terminal output , we notice that the owner of the originating IP address is Hostwinds LLC.

```
remarks: 192.168.0.0/16 are found Indiana-IPv4-Special-Registry.
remarks: 192.0.0/24 reserved for IANA IPv4 Special Purpose
remarks: Address Registry (RFC5736). Complete registration
remarks: details for 192.0.0/24 are found
remarks: Indiana-IPv4-Special-Registry.

whois: whois.arin.net

changed: 1993-05
source: IANA

# whois.arin.net

NetRange: 192.119.64.0 - 192.119.127.255
CIDR: 192.119.64.0/18
NetName: HOSTWINDS-18-2
NetHandle: NET-192-119-64-0-1
Parent: NET192 (NET-192-0-0-0-0)
NetType: Direct Allocation
OrgASes: ASS4296
Organization: Hostwinds LLC. (HL-29)
RegDate: 2012-11-12
Updated: 2021-09-23
Comment: https://www.hostwinds.com
Comment: Abuse Contact: abuse@hostwinds.com
Ref: https://rdap.arin.net/registry/p/192.119.64.0

OrgName: Hostwinds LLC
OrgId: HL-29
Address: 12181 Tukwila International Blvd, 3rd Floor, Suite 320
City: Seattle
StateProv: WA
PostalCode: 98168
Country: US
RegDate: 2011-11-30
Updated: 2021-09-23
Comment: https://www.hostwinds.com
Comment: Abuse Contact: abuse@hostwinds.com
Ref: https://rdap.arin.net/registry/entity/HL-29

ReferralServer: rwhois://rwhois.hostwinds.net:4321

OrgTechHandle: HNOC9-ARIN
OrgTechName: Hostwinds Network Operations Center
OrgTechPhone: +1-206-886-8665
OrgTechEmail: support@hostwinds.com
OrgTechRef: https://rdap.arin.net/registry/entity/HNOC9-ARIN

OrgAbuseHandle: HAC3-ARIN
OrgAbuseName: Hostwinds Abuse Center
OrgAbusePhone: +1-206-886-8665
OrgAbuseEmail: abuse@hostwinds.com
OrgAbuseRef: https://rdap.arin.net/registry/entity/HAC3-ARIN

OrgNOCHandle: HNOC9-ARIN
OrgNOCName: Hostwinds Network Operations Center
OrgNOCPhone: +1-206-886-8665
OrgNOCEmail: support@hostwinds.com
OrgNOCRef: https://rdap.arin.net/registry/entity/HNOC9-ARIN

(base) Faizs-MacBook-Pro:~ faizs [ ]
```

To find SPF record for return path

Command: nslookup -type=txt mutamarine.com

```
(base) Faizs-MacBook-Pro:~ faiz$ nslookup -type=txt mutamarine.com
Server:      100.64.100.1
Address:     100.64.100.1#53

** server can't find mutamarine.com: NXDOMAIN

(base) Faizs-MacBook-Pro:~ faiz$ nslookup -type=txt mutawamarine.com
Server:      100.64.100.1
Address:     100.64.100.1#53

Non-authoritative answer:
mutawamarine.com      text = "MS=ms97822417"
mutawamarine.com      text = "MS=842BCB91F2AB2807BE05D25DC690D1226B349676"
mutawamarine.com      text = "v=spf1 include:spf.protection.outlook.com -all"

Authoritative answers can be found from:

(base) Faizs-MacBook-Pro:~ faiz$
```

To find the DMARC record for the return path domain we used the dig command

Command: dig TXT _dmarc.mutawamarine.com

We notice that it is **v=DMARC1; p=quarantine; fo=1**

```
(base) Faizs-MacBook-Pro:~ faiz$ dig TXT _dmarc.mutawamarine.com
; <<>> DiG 9.10.6 <<>> TXT _dmarc.mutawamarine.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 62614
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;;
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 1232
;; QUESTION SECTION:
;_dmarc.mutawamarine.com.      IN      TXT
;;
;; ANSWER SECTION:
_dmarc.mutawamarine.com. 3600    IN      TXT      "v=DMARC1; p=quarantine; fo=1"
;;
;; Query time: 111 msec
;; SERVER: 100.64.100.1#53(100.64.100.1)
;; WHEN: Mon Sep 11 17:34:45 PDT 2023
;; MSG SIZE  rcvd: 93
(base) Faizs-MacBook-Pro:~ faiz$ █
```

Finally from our analysis we were found out that the attachment was named **SWT_#09674321__PDF__.CAB**. From there on we used the hybrid analysis to find out the **SHA256: 2e91c533615a9bb8929ac4bb76707b2444597ce063d84a4b33525e25074fff3f**
File Extension: RAR

Analysis Overview

Submission name: **SWT_#09674321__PDF__.CAB**

Size: 400KB

Type: **unknown**

Mime: application/x-rar

SHA256: **2e91c533615a9bb8929ac4bb76707b2444597ce063d84a4b33525e25074fff3f**

Last Anti-Virus Scan: 07/21/2023 18:13:12 (UTC)

Last Sandbox Report: 01/02/2022 16:57:41 (UTC)

Threat Score: 100/100

AV Detection: 32%

Labeled as: **Trojan.Generic**

Analysis Overview

Anti-Virus Scanner Results

Related Hashes

Community (0)

Back to top

Anti-Virus Results

MetaDefender: 3% Multi Scan Analysis (Last Update: 07/21/2023 18:13:12 UTC)

VirusTotal: 60% Multi Scan Analysis (Last Update: 07/21/2023 18:13:12 UTC)

From our analysis we can infer that the file attachment was indeed malicious , marked as a **Trojan.Generic**

Timestamp	Input	Threat level	Analysis Summary	Countries	Environment	Action
March 11th 2023 10:01:01 (UTC)	SWT_#09674321__PDF__.CAB_PW_infected.zip Zip archive data, at least v2.0 to extract c1cd30b4-4277-4840-8750-9648bd7fed9b0ddaf62c2aa4d0fa2193d49fd9b0d6ca5996	no specific threat	AV Detection: Marked as clean	-	Windows 7 32 bit	<input type="checkbox"/>
January 2nd 2022 16:57:41 (UTC)	attachment.pdf.cab RAR archive data, v87. 2e91c533615a9bb8929ac4bb76707b2444597ce063d84a4b33525e25074fff3f	malicious	AV Detection: 32% Trojan.Generic	-	Windows 7 32 bit	<input type="checkbox"/>
December 31st 2021 20:30:12 (UTC)	SWT_#09674321__PDF__.CAB RAR archive data, v87. 2e91c533615a9bb8929ac4bb76707b2444597ce063d84a4b33525e25074fff3f	malicious	AV Detection: 32% Trojan.Generic	-	Windows 7 64 bit	<input type="checkbox"/>

Finally we utilized virus total to find the exact file size of the pdf attachment (SWT_#09674321____PDF__.CAB) plugging in the SHA-256 we discovered on hybrid analysis.

We discovered that the file size of the attachment was actually 400.26 KB

VirusTotal - File - 2e91c533615a9bb8929ac4bb76707b2444597ce063d84a4b33525e25074fff3f

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Basic properties

MD5	f4dd3456ccb976a145c1179ad4461ec
SHA-1	5a2bb0188377c15c036843b4a6ab9bc0f2c1607
SHA-256	2e91c533615a9bb8929ac4bb76707b2444597ce063d84a4b33525e25074fff3f
SSDEEP	12288:Mj6ygt8RoYqMANuL8IOA81aYolm9-X3B4k56:EgoRJCul87tolC+X3O
TLSH	T12C94238893562439A8F7385DAFD0CFB5EFE898E74E8F97709CFD609E5D140446205AC2
File type	RAR compressed rar
Magic	RAR archive data, v5
TrID	RAR compressed archive (v5.0) (61.5%) RAR compressed archive (gen) (38.4%)
File size	400.26 KB (409868 bytes)

History

- First Submission 2020-06-10 07:06:14 UTC
- Last Submission 2023-08-17 17:23:53 UTC
- Last Analysis 2023-09-07 17:45:53 UTC

X EXTENSION ^ v Highlight All Match Case Match Diacritics Whole Words 1 of 1 match

