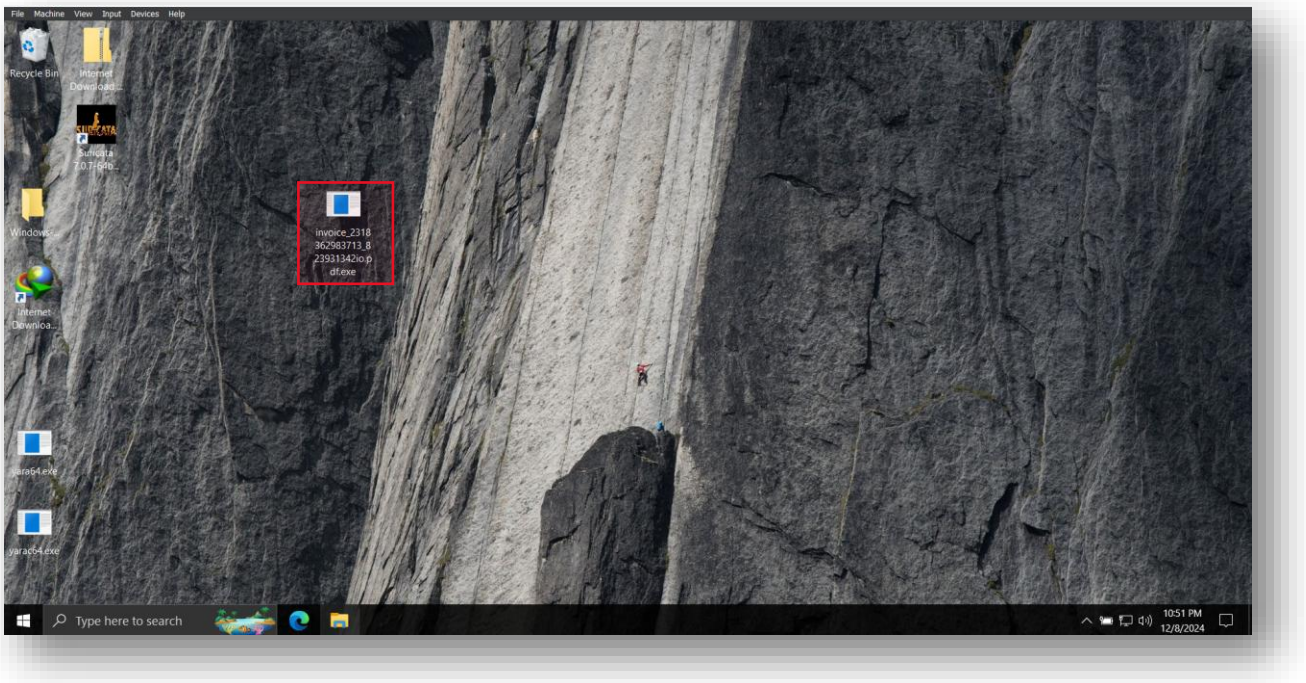
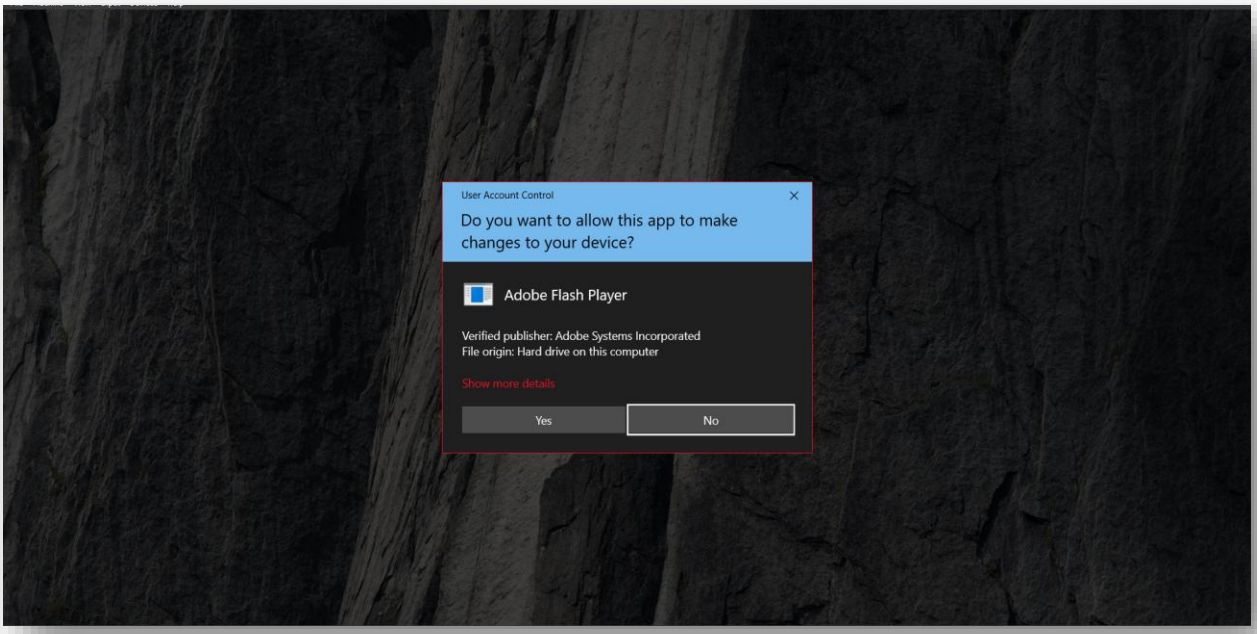
Proactive Project

Simulate Malware Execution:

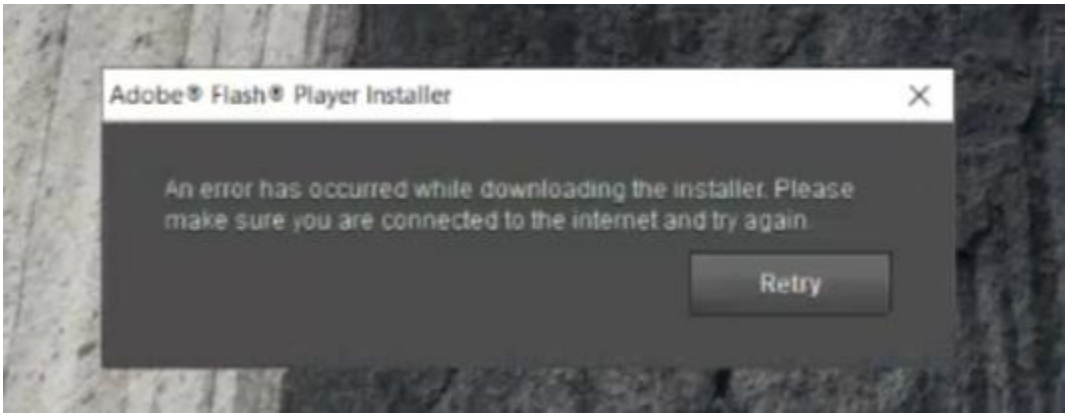
We are using a Windows 10 Enterprise Virtual Machine on Virtual Box



Here we have the **ZEUS malware** before we run the malware, we will open **Wireshark** and start capturing the packets to further analysis.



After Running it a UAC pops and requires changes approval from Adobe Flash Player. It doesn't matter if we click yes or no it will do the same because if we clicked no, the UAC window will reappear.



If we click on retry it will just reload and fail again.

Finally, the malware disappears from the desktop and the flash player tries to connect to something but fails.

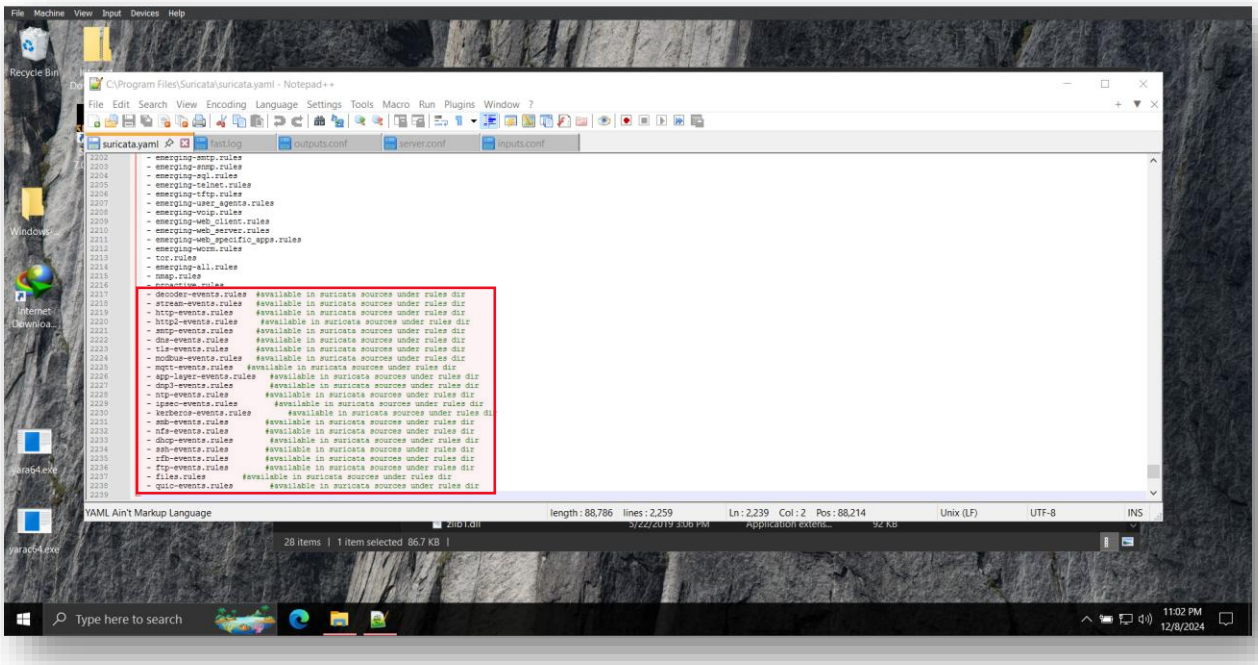
Basic Conclusion:

- Has Anti-VM capabilities

- Dropped an executable

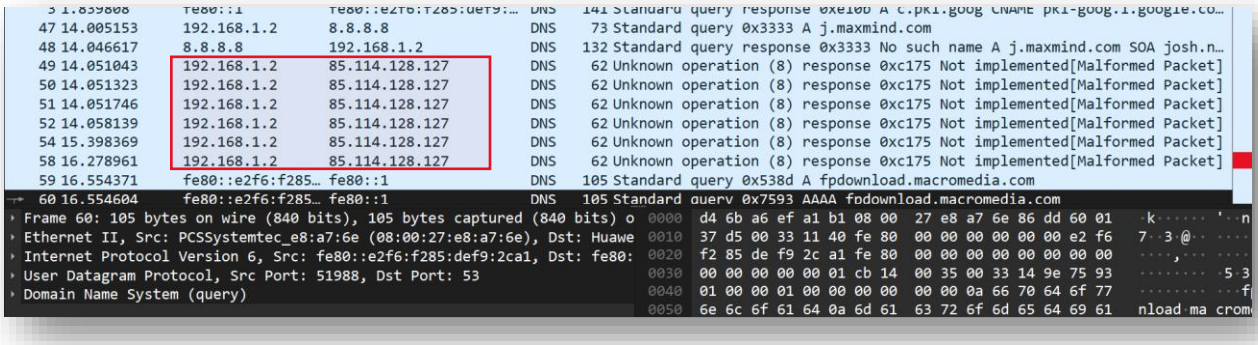
- Deleted / Moved the main executable

Configure Suricata for Network Monitoring:

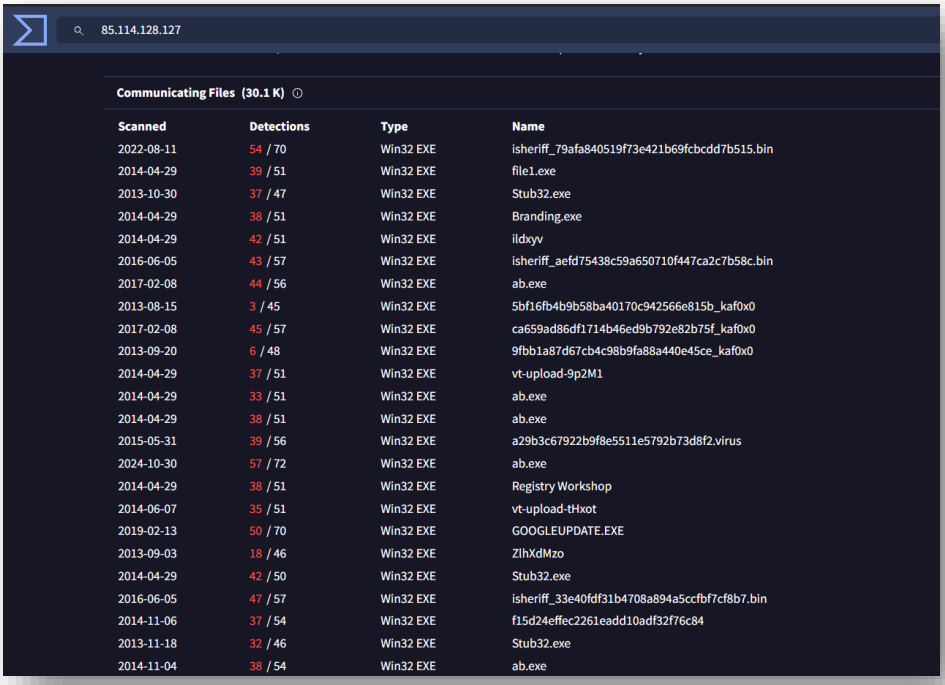


After Installing Suricata we went to its directory specifically suricata.yaml file to uncomment the default rules and we downloaded a collection of rules in one file called emerging-all.rules to work with in our project.

We opened the captured packets to check its behavior using **Wireshark**



When analyzing the packets using Wireshark, I noticed that the malware sends a malformed DNS request to 85.114.128.127.



When searching for it on Virus Total It was flagged as safe, but by noticing the files that communicates with this Ip I knew It was malicious.

```
1 alert ip any any -> 85.114.128.127 any (msg:"ZEUS Trojan C2 Connect: 85.114.128.127"; sid:1000123; rev:1;)
```

By knowing this I made a Suricata rule to alert when connecting to that IP and adding it to the suricata.yaml file.

```
.\suricata.exe -c suricata.yaml -i 192.168.1.2 -l log --service-install
```

So, by using these commands I used the suricata.yaml file as the config file and I listened on my ip address and made the logs dumps into the log folder and finally

--service-install: to add it as a service to run on startup.

Integrate with Splunk:

```
[tcpout]
defaultGroup = default-autolb-group

[tcpout:default-autolb-group]
server = 192.168.1.9:9997

[tcpout-server://192.168.1.9:9997]
```

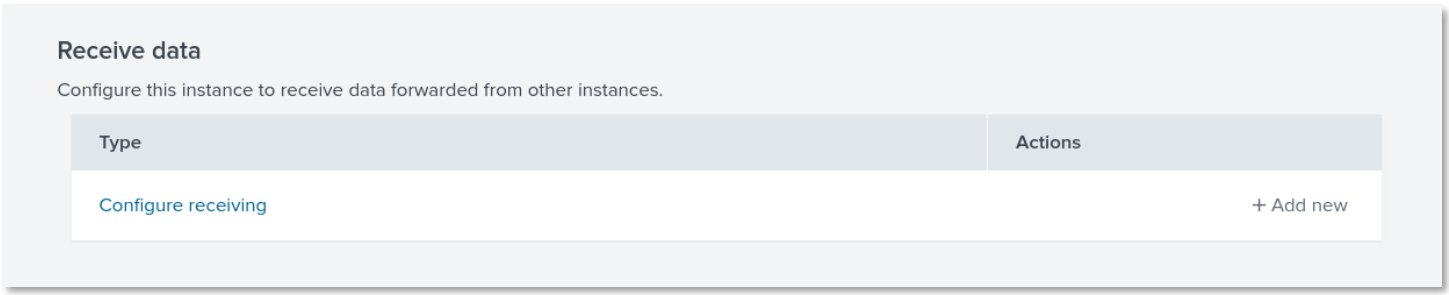
Now we need to forward these alerts to Splunk for centralized analysis.

We started by configuring the main server to send the alerts on using this path:

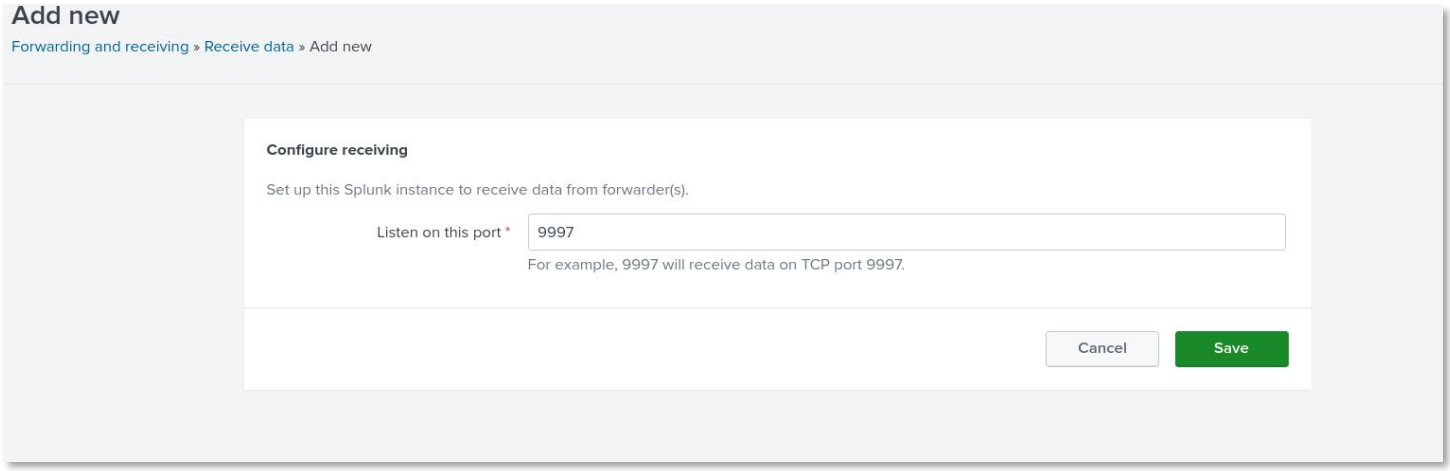
```
C:\Program Files\SplunkUniversalForwarder\etc\system\local\outputs.conf
```

```
[monitor://C:\Program Files\Suricata\log\fast.log]
disabled = false
index = suricata
```

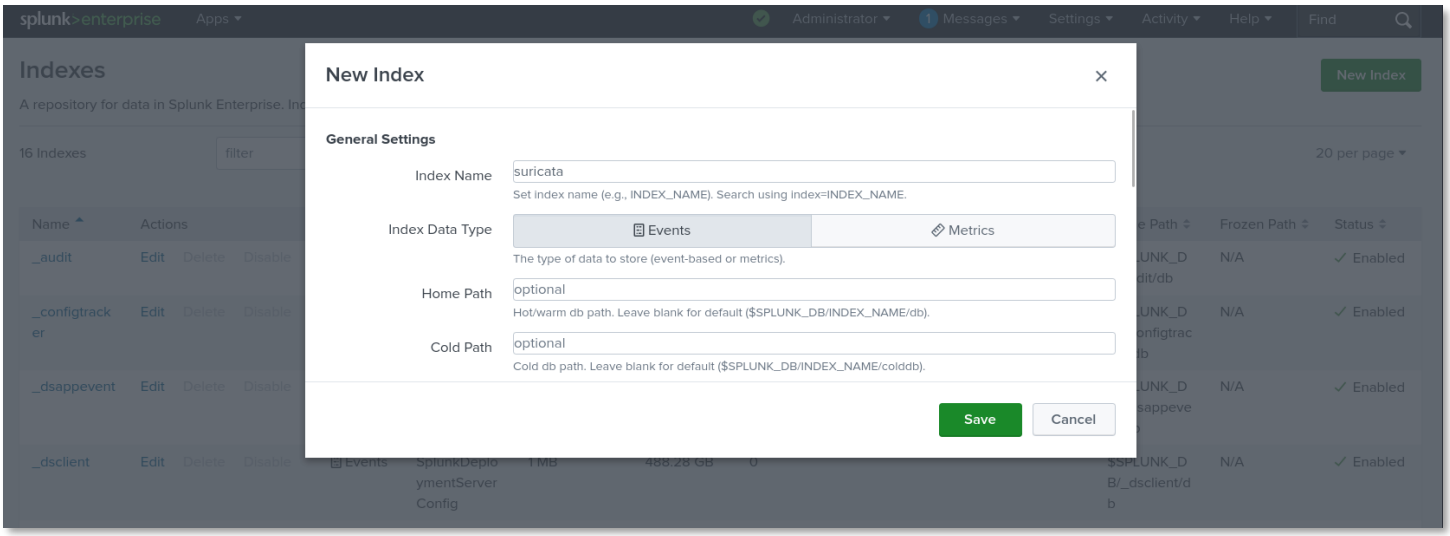
Suricata appends detailed event logs to fast.log in real-time, enabling Splunk to ingest and analyze security events promptly.



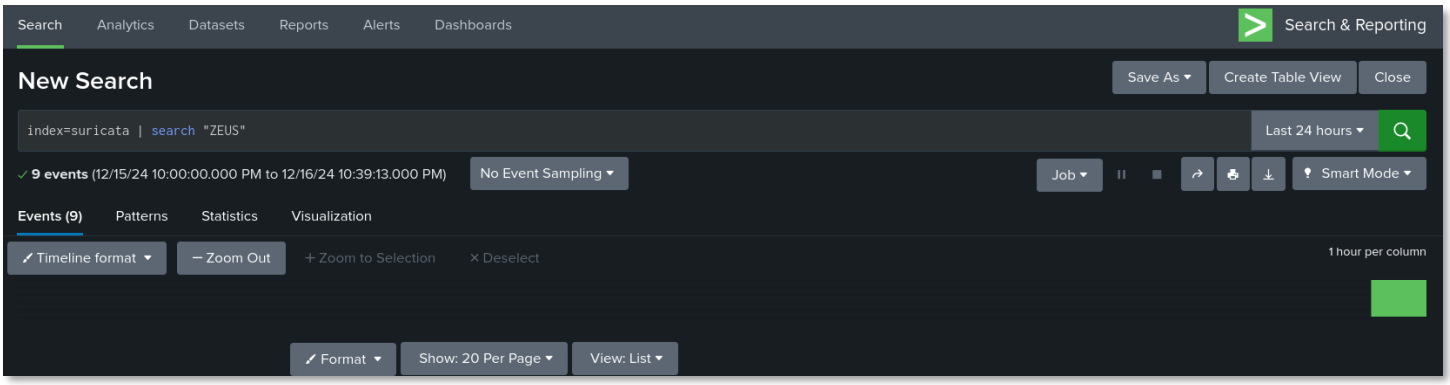
We go to settings then forwarding and receiving



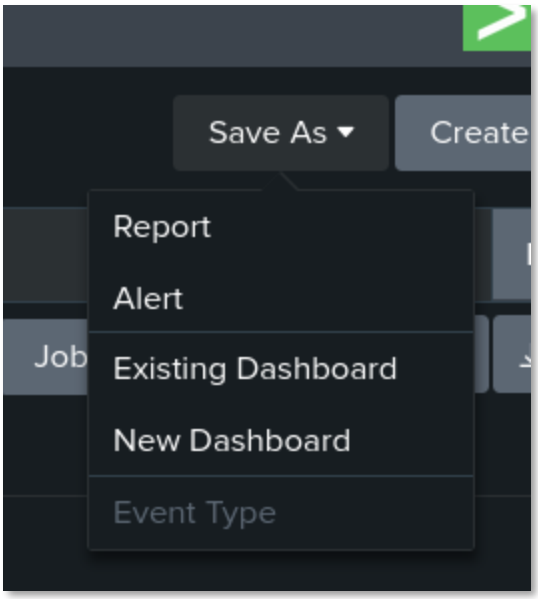
Then we add a new receiving to listen on port 9997



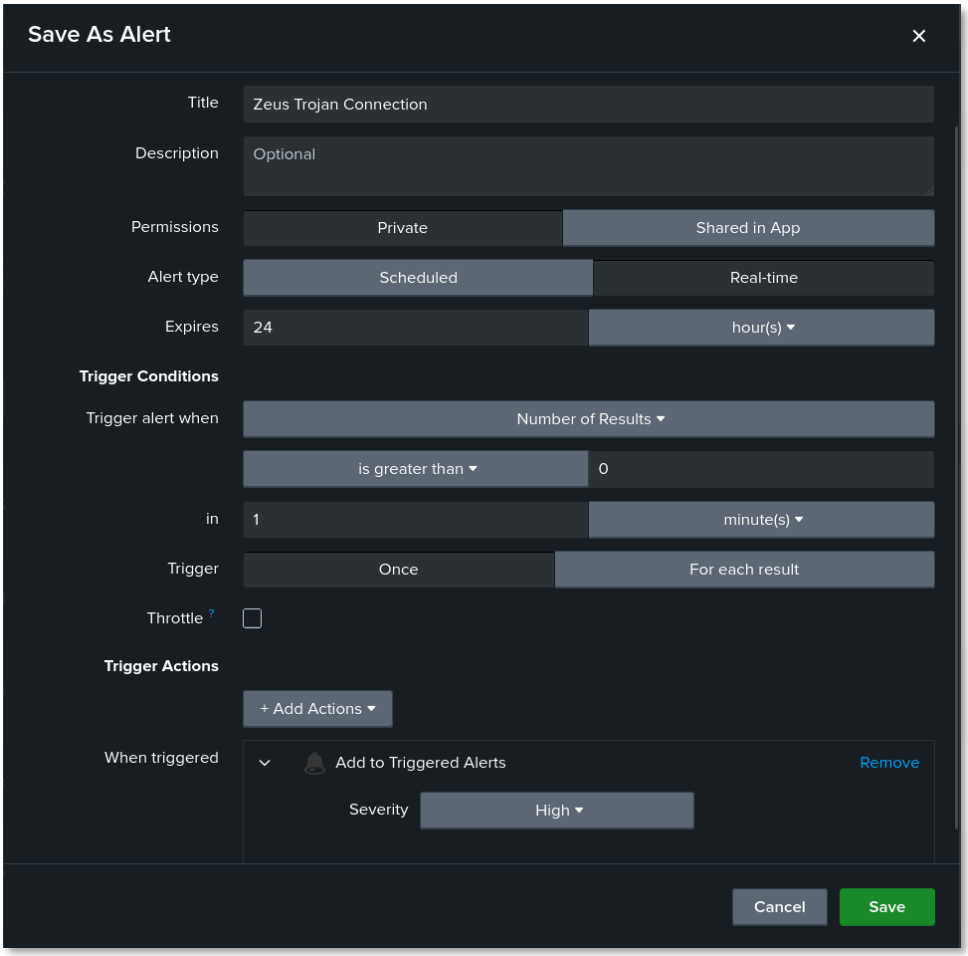
Then we go to settings/index and create a new index to isolate Suricata events.

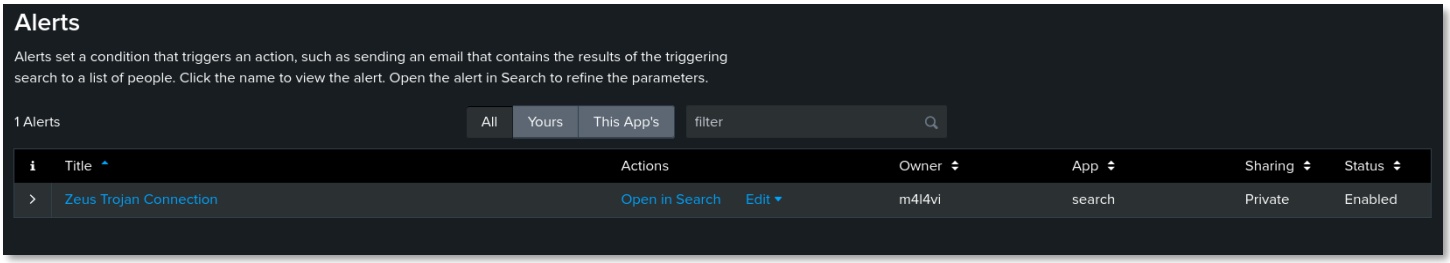


Then we go to search for ZEUS from Suricata index

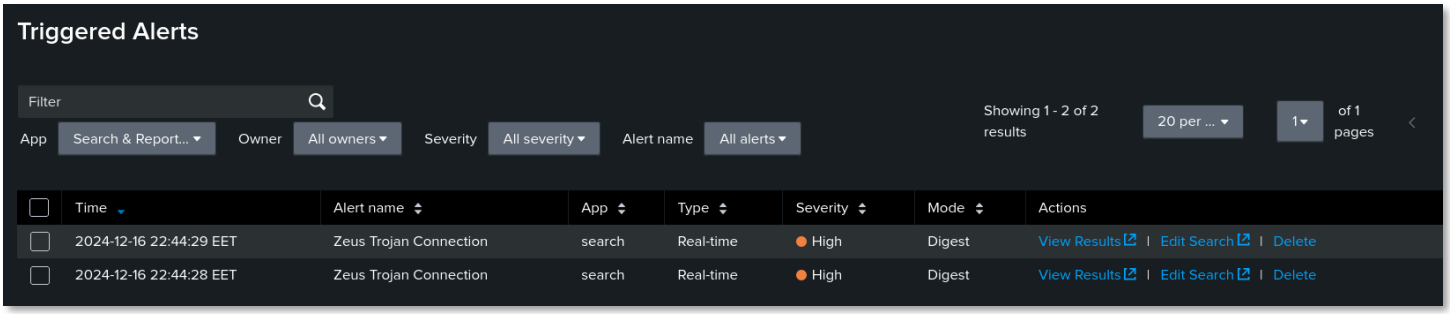


Then we will save it as an alert.

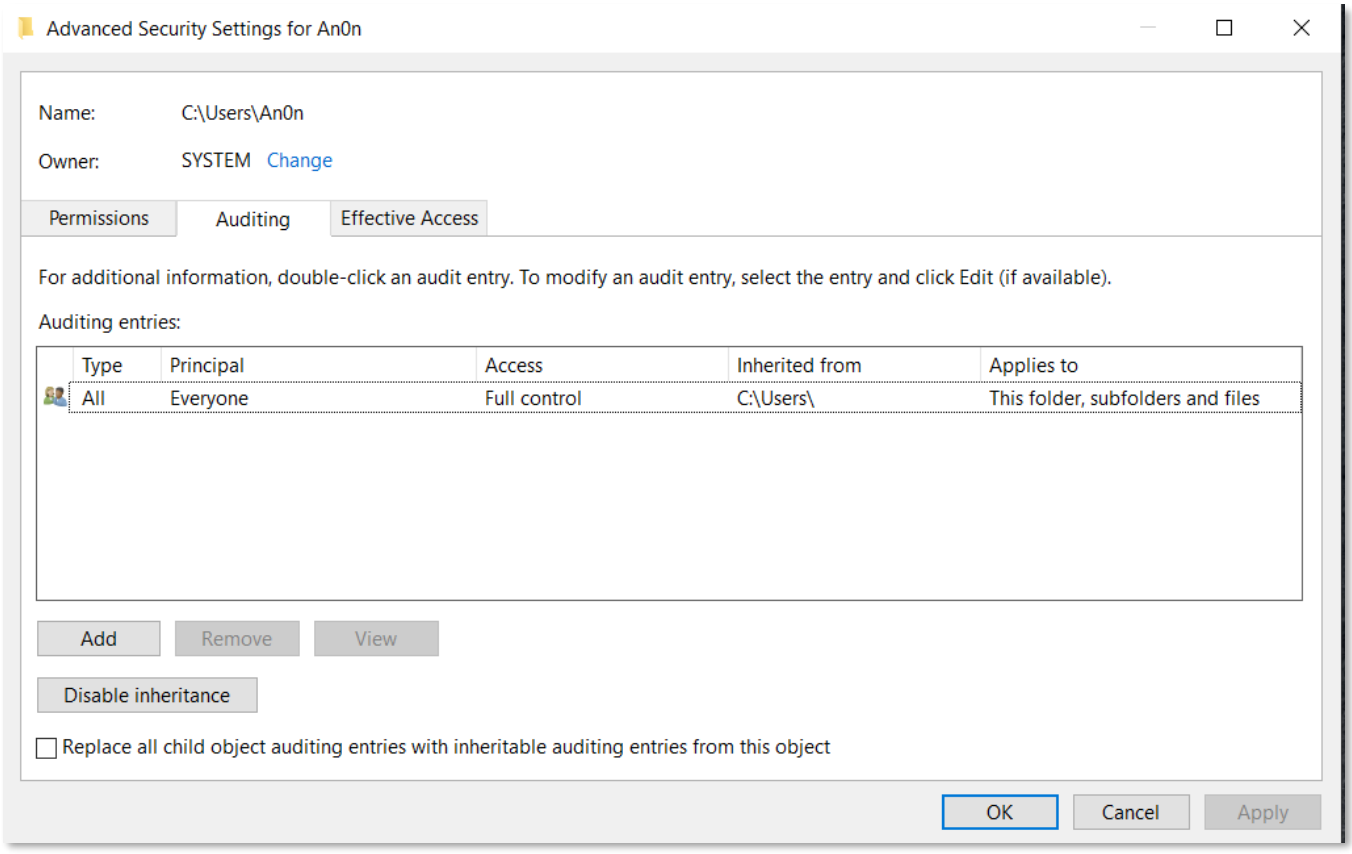




Here we can show the alerts we wrote for now.



Next, we executed the malware again to identify any triggered alerts.



We audit the **Users** folder to the windows event logs of any file creations or deletions

```
host="DESKTOP-R74UPAT" EventCode=4663
```

Event ID 4663: Indicates an attempt to access an object, such as file creation, modification, or deletion.

After some investigation we found that the malware adding files in temp we can confirm it using the event code above.

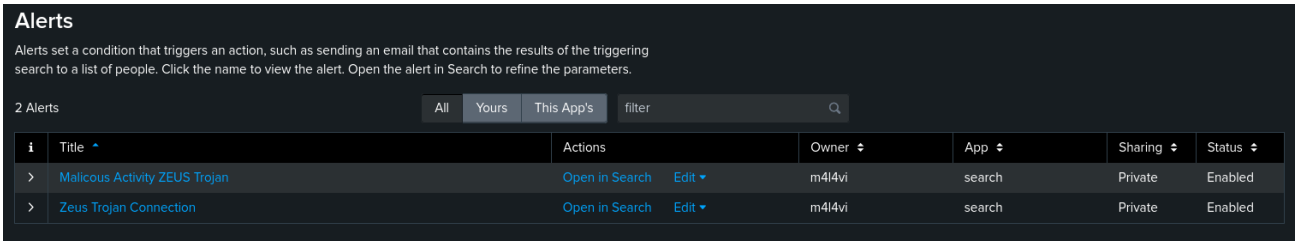
```
12/16/24
9:32:36.000 PM    12/16/2024 09:32:36 PM
                  LogName=Security
                  EventCode=4663
                  EventType=0
                  ComputerName=DESKTOP-R74UPAT
                  SourceName=Microsoft Windows security auditing.
                  Type=Information
                  RecordNumber=466651
                  Keywords=Audit Success
                  TaskCategory=File System
                  OpCode=Info
                  Message=An attempt was made to access an object.

                  Subject:
                          Security ID:            S-1-5-21-2826541408-2641107319-1920346365-1001
                          Account Name:           An0n
                          Account Domain:         DESKTOP-R74UPAT
                          Logon ID:               0x13F6B

                  Object:
                          Object Server:          Security
                          Object Type:            File
                          Object Name:            C:\Users\An0n\AppData\Local\Temp\InstallFlashPlayer.exe
                          Handle ID:              0x5cc
                          Resource Attributes:    S:AI

                  Process Information:
                          Process ID:             0x2944
                          Process Name:           C:\Users\An0n\Desktop\invoice_2318362983713_823931342io.exe
```

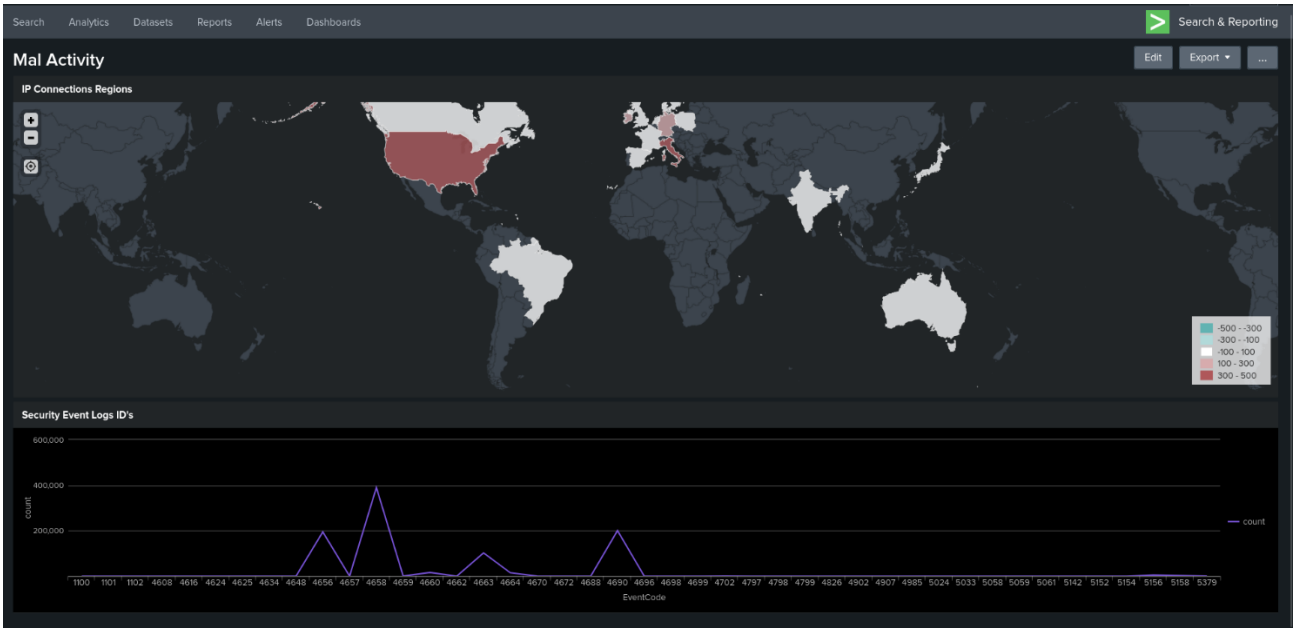
A thorough examination of the temporary directory revealed that the malicious invoice.exe executable had dropped falshplayer.exe, with the intent of launching it.

Malicous Activity ZEUS Trojan

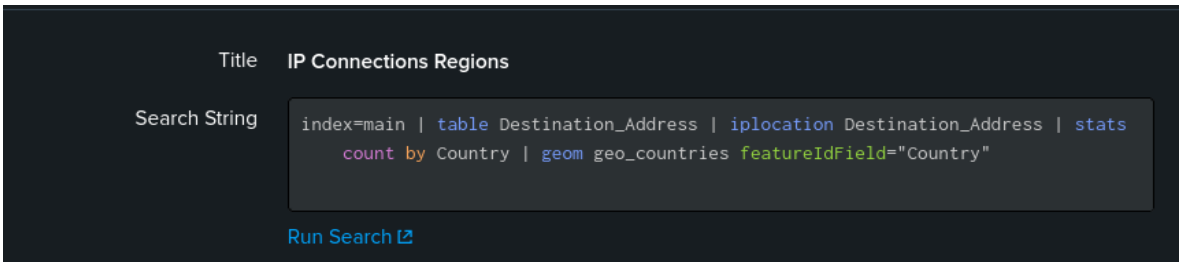
```
Process_Name="*\\invoice_2318362983713_823931342io.pdf.exe"
```

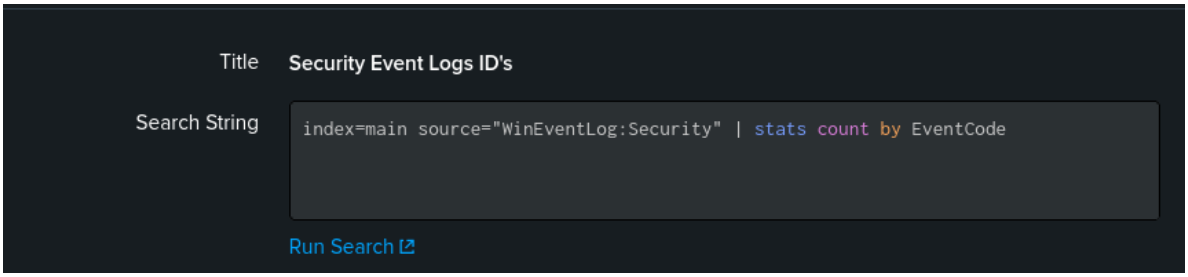
Now we searched for this file in any path to find out if this file manipulates any process





Finally, we make this query to create a dashboard





Queries to create this dashboard.

Volatility & Yara Rules

Working on the memory dump that you provided to us on classroom 🔗

I'll be working on Volatility Workbench (GUI Volatility) and Volatility 2

First, we need to list all processes we've got using (pslist and pstree) volatility plugins

pslist

PID	PPID	ImageFileName	Offset(V)	Threads	Handles	SessionId	Wow64	CreateTime	ExitTime	File output
4	0	System	0x823c8a00	57	671	N/A	False	N/A	N/A	Disabled
596	4	smss.exe	0x82292da0	3	19	N/A	False	2010-09-02 12:25:18.000000 UTC	N/A	Disabled
668	596	csrss.exe	0x821f2978	14	471	0	False	2010-09-02 12:25:21.000000 UTC	N/A	Disabled
692	596	winlogon.exe	0x822c09f8	21	588	0	False	2010-09-02 12:25:22.000000 UTC	N/A	Disabled
744	692	services.exe	0x821a5da0	15	279	0	False	2010-09-02 12:25:22.000000 UTC	N/A	Disabled
756	692	lsass.exe	0x822c8798	24	437	0	False	2010-09-02 12:25:22.000000 UTC	N/A	Disabled
912	744	svchost.exe	0x82150b90	20	202	0	False	2010-09-02 12:25:22.000000 UTC	N/A	Disabled
992	744	svchost.exe	0x822c8bf8	10	277	0	False	2010-09-02 12:25:22.000000 UTC	N/A	Disabled
1084	744	svchost.exe	0x82151da0	58	1327	0	False	2010-09-02 12:25:22.000000 UTC	N/A	Disabled
1140	744	svchost.exe	0x821521b0	6	81	0	False	2010-09-02 12:25:22.000000 UTC	N/A	Disabled
1192	744	svchost.exe	0x8214f488	13	175	0	False	2010-09-02 12:25:23.000000 UTC	N/A	Disabled
1436	744	iscsiexe.exe	0x8221e278	6	78	0	False	2010-09-02 12:25:24.000000 UTC	N/A	Disabled
1616	744	spoolsv.exe	0x82095500	13	140	0	False	2010-09-02 12:25:24.000000 UTC	N/A	Disabled
1752	1720	explorer.exe	0x821b2020	22	520	0	False	2010-09-02 12:25:25.000000 UTC	N/A	Disabled
1900	1752	SharedIntApp.ex	0x822b96c0	3	75	0	False	2010-09-02 12:25:25.000000 UTC	N/A	Disabled
1908	1752	prl_cc.exe	0x820ee580	14	133	0	False	2010-09-02 12:25:25.000000 UTC	N/A	Disabled
1936	1752	jusched.exe	0x8212ada0	1	43	0	False	2010-09-02 12:25:26.000000 UTC	N/A	Disabled
364	744	svchost.exe	0x82129370	4	88	0	False	2010-09-02 12:25:33.000000 UTC	N/A	Disabled
472	744	jqs.exe	0x82089558	5	146	0	False	2010-09-02 12:25:33.000000 UTC	N/A	Disabled
488	744	sqlservr.exe	0x8208abf0	25	306	0	False	2010-09-02 12:25:33.000000 UTC	N/A	Disabled
572	744	coherence.exe	0x82077da0	4	51	0	False	2010-09-02 12:25:36.000000 UTC	N/A	Disabled
436	744	prl_tools_servi	0x82189530	3	78	0	False	2010-09-02 12:25:36.000000 UTC	N/A	Disabled
632	436	prl_tools.exe	0x82086798	9	107	0	False	2010-09-02 12:25:36.000000 UTC	N/A	Disabled
660	744	sqlwriter.exe	0x821aa7e8	4	84	0	False	2010-09-02 12:25:36.000000 UTC	N/A	Disabled
2180	1084	wscntfy.exe	0x8213dda0	3	48	0	False	2010-09-02 12:25:41.000000 UTC	N/A	Disabled
2588	744	alg.exe	0x81e8a368	6	107	0	False	2010-09-02 12:25:44.000000 UTC	N/A	Disabled
940	1084	wuauclt.exe	0x8205dda0	4	126	0	False	2010-09-02 12:26:40.000000 UTC	N/A	Disabled
2972	1752	ImmunityDebugge	0x82001ad0	2	87	0	False	2010-09-08 19:14:36.000000 UTC	N/A	Disabled
2204	2972	nifek_locked.ex	0x8207bda0	2	38	0	False	2010-09-08 19:14:36.000000 UTC	N/A	Disabled
1932	1752	ImmunityDebugge	0x82282380	2	86	0	False	2010-09-08 19:23:02.000000 UTC	N/A	Disabled
952	1932	vaelh.exe	0x8223c020	2	40	0	False	2010-09-08 19:23:02.000000 UTC	N/A	Disabled
3788	1752	ImmunityDebugge	0x81ffb6d8	2	103	0	False	2010-09-08 22:39:40.000000 UTC	N/A	Disabled
3508	3788	anaxu.exe	0x8219e5c8	2	54	0	False	2010-09-08 22:39:40.000000 UTC	N/A	Disabled
3984	1084	wuauclt.exe	0x81eab2f8	8	325	0	False	2010-09-09 19:52:45.000000 UTC	N/A	Disabled
2404	1752	ImmunityDebugge	0x82066478	2	85	0	False	2010-09-09 19:56:19.000000 UTC	N/A	Disabled
3772	2404	b98679df6defbb3	0x81f4bb28	1	46	0	False	2010-09-09 19:56:19.000000 UTC	N/A	Disabled
3276	3772	ihah.exe	0x81e87da0	1	45	0	False	2010-09-09 19:56:32.000000 UTC	N/A	Disabled
3768	1084	rundll32.exe	0x82311648	1	53	0	False	2010-09-09 19:56:33.000000 UTC	N/A	Disabled

Time Stamp: Mon Dec 16 18:12:00 2024

pstree

PID	PPID	ImageFileName	Offset(V)	Threads	Handles	SessionId	Wow64	CreateTime	ExitTime
4	0	System	0x823c8a00	57	671	N/A	False	N/A	N/A
* 596	4	smss.exe	0x82292da0	3	19	N/A	False	2010-09-02 12:25:18.000000 UTC	N/A
** 692	596	winlogon.exe	0x822c09f8	21	588	0	False	2010-09-02 12:25:22.000000 UTC	N/A
*** 744	692	services.exe	0x821a5da0	15	279	0	False	2010-09-02 12:25:22.000000 UTC	N/A
**** 992	744	svchost.exe	0x822c8bf8	10	277	0	False	2010-09-02 12:25:22.000000 UTC	
**** 1192	744	svchost.exe	0x8214f488	13	175	0	False	2010-09-02 12:25:23.000000 UTC	
**** 488	744	sqlservr.exe	0x8208abf0	25	306	0	False	2010-09-02 12:25:33.000000 UTC	
**** 364	744	svchost.exe	0x82129370	4	88	0	False	2010-09-02 12:25:33.000000 UTC	
**** 912	744	svchost.exe	0x82150b90	20	202	0	False	2010-09-02 12:25:22.000000 UTC	
**** 1616	744	spoolsv.exe	0x82095500	13	140	0	False	2010-09-02 12:25:24.000000 UTC	
**** 1140	744	svchost.exe	0x821521b0	6	81	0	False	2010-09-02 12:25:22.000000 UTC	
**** 436	744	prl_tools_servi	0x82189530	3	78	0	False	2010-09-02 12:25:36.000000 UTC	
***** 632	436	prl_tools.exe	0x82086798	9	107	0	False	2010-09-02 12:25:36.000000 UTC	
**** 660	744	sqlwriter.exe	0x821aa7e8	4	84	0	False	2010-09-02 12:25:36.000000 UTC	
**** 472	744	jqs.exe	0x82089558	5	146	0	False	2010-09-02 12:25:33.000000 UTC	N/A
**** 1436	744	iscsiexe.exe	0x8221e278	6	78	0	False	2010-09-02 12:25:24.000000 UTC	
**** 1084	744	svchost.exe	0x82151da0	58	1327	0	False	2010-09-02 12:25:22.000000 UTC	
***** 3984	1084	wuauclt.exe	0x81eab2f8	8	325	0	False	2010-09-09 19:52:45.000000 UTC	
***** 940	1084	wuauclt.exe	0x8205dda0	4	126	0	False	2010-09-02 12:26:40.000000 UTC	
***** 2180	1084	wscntfy.exe	0x8213dda0	3	48	0	False	2010-09-02 12:25:41.000000 UTC	
***** 3768	1084	rundll32.exe	0x82311648	1	53	0	False	2010-09-09 19:56:33.000000 UTC	
**** 2588	744	alg.exe	0x81e8a368	6	107	0	False	2010-09-02 12:25:44.000000 UTC	N/A
**** 572	744	coherence.exe	0x82077da0	4	51	0	False	2010-09-02 12:25:36.000000 UTC	
*** 756	692	lsass.exe	0x822c8798	24	437	0	False	2010-09-02 12:25:22.000000 UTC	N/A
** 668	596	csrss.exe	0x821f2978	14	471	0	False	2010-09-02 12:25:21.000000 UTC	N/A
1752	1720	explorer.exe	0x821b2020	22	520	0	False	2010-09-02 12:25:25.000000 UTC	N/A
* 2404	1752	ImmunityDebugge	0x82066478	2	85	0	False	2010-09-09 19:56:19.000000 UTC	N/A
** 3772	2404	b98679df6defbb3	0x81f4bb28	1	46	0	False	2010-09-09 19:56:19.000000 UTC	N/A
*** 3276	3772	ihah.exe	0x81e87da0	1	45	0	False	2010-09-09 19:56:32.000000 UTC	
* 1900	1752	SharedIntApp.ex	0x822b96c0	3	75	0	False	2010-09-02 12:25:25.000000 UTC	N/A
* 1932	1752	ImmunityDebugge	0x82282380	2	86	0	False	2010-09-08 19:23:02.000000 UTC	N/A
** 952	1932	vaelh.exe	0x8223c020	2	40	0	False	2010-09-08 19:23:02.000000 UTC	N/A
* 3788	1752	ImmunityDebugge	0x81ffb6d8	2	103	0	False	2010-09-08 22:39:40.000000 UTC	N/A
** 3508	3788	anaxu.exe	0x8219e5c8	2	54	0	False	2010-09-08 22:39:40.000000 UTC	N/A
* 1936	1752	jusched.exe	0x8212ada0	1	43	0	False	2010-09-02 12:25:26.000000 UTC	N/A
* 1908	1752	prl_cc.exe	0x820ee580	14	133	0	False	2010-09-02 12:25:25.000000 UTC	N/A
* 2972	1752	ImmunityDebugge	0x82001ad0	2	87	0	False	2010-09-08 19:14:36.000000 UTC	N/A
** 2204	2972	nifek_locked.ex	0x8207bda0	2	38	0	False	2010-09-08 19:14:36.000000 UTC	N/A

Time Stamp: Mon Dec 16 18:14:12 2024

We can see from the process tree there's a sql server and there's a VM running on the system on startup (sqlserver.exe),(prl_tools.exe) which Parallels Tools for the running VM.

And some services for running it, in addition to some dll's

These are the cmd command and the full path for all these processes that we talked about

```
\system32\smss.exe      \SystemRoot\System32\smss.exe   \SystemRoot\System32\smss.exe
\system32\winlogon.exe   winlogon.exe    \??\C:\WINDOWS\system32\winlogon.exe
\system32\services.exe   C:\WINDOWS\system32\services.exe       C:\WINDOWS\system32\services.exe
\WINDOWS\system32\svchost.exe    C:\WINDOWS\system32\svchost -k rpcss    C:\WINDOWS\system32\svchost.exe
\WINDOWS\system32\svchost.exe    C:\WINDOWS\system32\svchost.exe -k LocalService C:\WINDOWS\system32\svchost.exe
\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL\Binn\sqlservr.exe      "c:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL\Binn\sqlservr.exe" -sSQLEXPRESS    c:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL\Binn\sqlservr.exe
\WINDOWS\system32\svchost.exe    -      -
\WINDOWS\system32\svchost.exe    C:\WINDOWS\system32\svchost -k DcomLaunch       C:\WINDOWS\system32\svchost.exe
\WINDOWS\system32\spoolsv.exe    C:\WINDOWS\system32\spoolsv.exe C:\WINDOWS\system32\spoolsv.exe
\WINDOWS\system32\svchost.exe    C:\WINDOWS\system32\svchost.exe -k NetworkService       C:\WINDOWS\system32\svchost.exe
\Program Files\Parallels\Parallels Tools\Services\prl_tools_service.exe  -       -
\Program Files\Parallels\Parallels Tools\Services\prl_tools.exe  "C:\Program Files\Parallels\Parallels Tools\Services\prl_tools.exe"     C:\Program Files\Parallels\Parallels Tools\Services\prl_tools.exe
\Program Files\Microsoft SQL Server\90\Shared\sqlwriter.exe      "c:\Program Files\Microsoft SQL Server\90\Shared\sqlwriter.exe" c:\Program Files\Microsoft SQL Server\90\Shared\sqlwriter.exe
 Files\Java\jre6\bin\jqs.exe     "C:\Program Files\Java\jre6\bin\jqs.exe" -service -config "C:\Program Files\Java\jre6\lib\deploy\jqs\jqs.conf"  C:\Program Files\Java\jre6\bin\jqs.exe
\WINDOWS\system32\iscsiexe.exe   C:\WINDOWS\System32\iscsiexe.exe       C:\WINDOWS\System32\iscsiexe.exe
\WINDOWS\system32\svchost.exe    C:\WINDOWS\System32\svchost.exe -k netsvcs      C:\WINDOWS\System32\svchost.exe
\WINDOWS\system32\wuauclt.exe    "C:\WINDOWS\system32\wuauclt.exe" /RunStoreAsComServer Local\[43c]SUSDS82f0c54ad7b58b46a717b19ec999e73f C:\WINDOWS\system32\wuauclt.exe
\WINDOWS\system32\wuauclt.exe    "C:\WINDOWS\system32\wuauclt.exe"       C:\WINDOWS\system32\wuauclt.exe
\WINDOWS\system32\wscntfy.exe    C:\WINDOWS\system32\wscntfy.exe C:\WINDOWS\system32\wscntfy.exe
\WINDOWS\system32\rundll32.exe   C:\WINDOWS\System32\rundll32.exe C:\WINDOWS\System32\firewall.cpl,ShowNotificationDialog /configure "C:\WINDOWS\explorer.exe"   C:\WINDOWS\System32\rundll32.exe
\system32\alg.exe        -      -
\Program Files\Parallels\Parallels Tools\Services\coherence.exe  -       -
\system32\lsass.exe      C:\WINDOWS\system32\lsass.exe   C:\WINDOWS\system32\lsass.exe
```

Nothing really seems suspicious to me till now

So, let's see and investigate the next processes block (that have a fresh start process tree)

```
1752    1720    explorer.exe     0x821b2020      22      520     0
* 2404  1752    ImmunityDebugge 0x82066478      2       85      0
** 3772 2404    b98679df6defbb3 0x81f4bb28      1       46      0
*** 3276        3772    ihah.exe        0x81e87da0      1       45
* 1900  1752    SharedIntApp.ex 0x822b96c0      3       75      0
* 1932  1752    ImmunityDebugge 0x82282380      2       86      0
** 952  1932    vaelh.exe        0x8223c020      2       40      0
* 3788  1752    ImmunityDebugge 0x81ffb6d8      2       103     0
** 3508 3788    anaxu.exe        0x8219e5c8      2       54      0
* 1936  1752    jusched.exe      0x8212ada0      1       43      0
* 1908  1752    prl_cc.exe       0x820ee580      14      133     0
* 2972  1752    ImmunityDebugge 0x82001ad0      2       87      0
** 2204 2972    nifek locked.ex 0x8207bda0      2       38      0
```

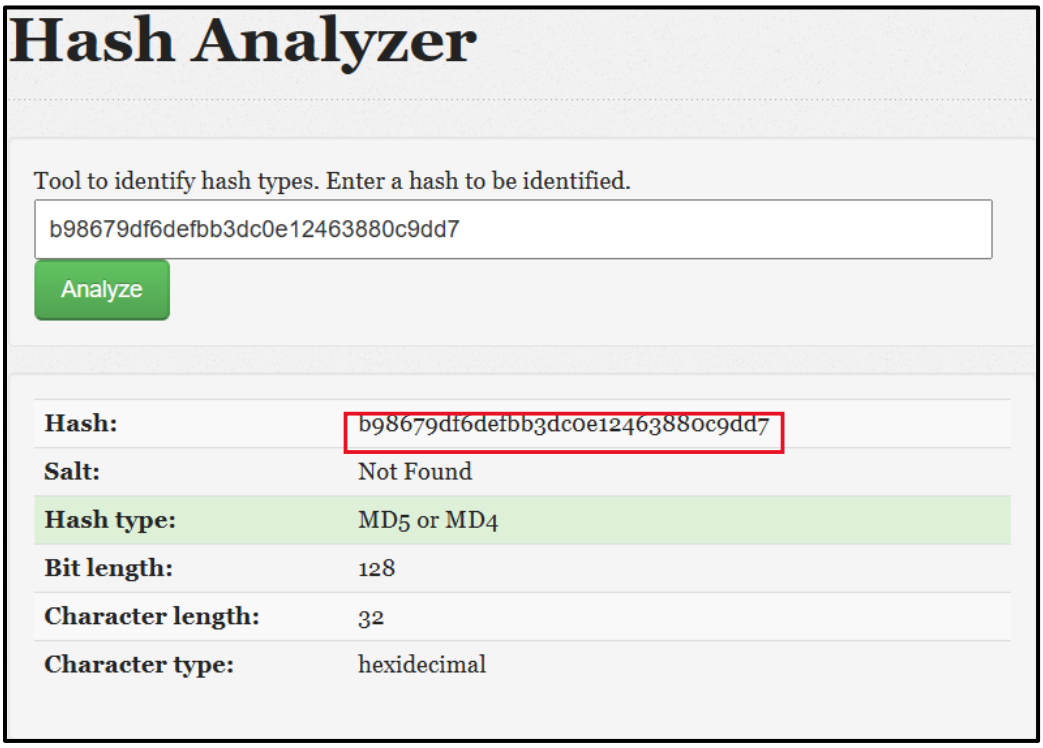
We can see there's a process for Immunity Debugger that has been executed from the explorer, and then it opened another application that has a very suspicious name (b98679df6defbb3), and this suspicious process executed another process (vaelh.exe).

Each time Immunity Debugger executes, it executes another suspicious process.

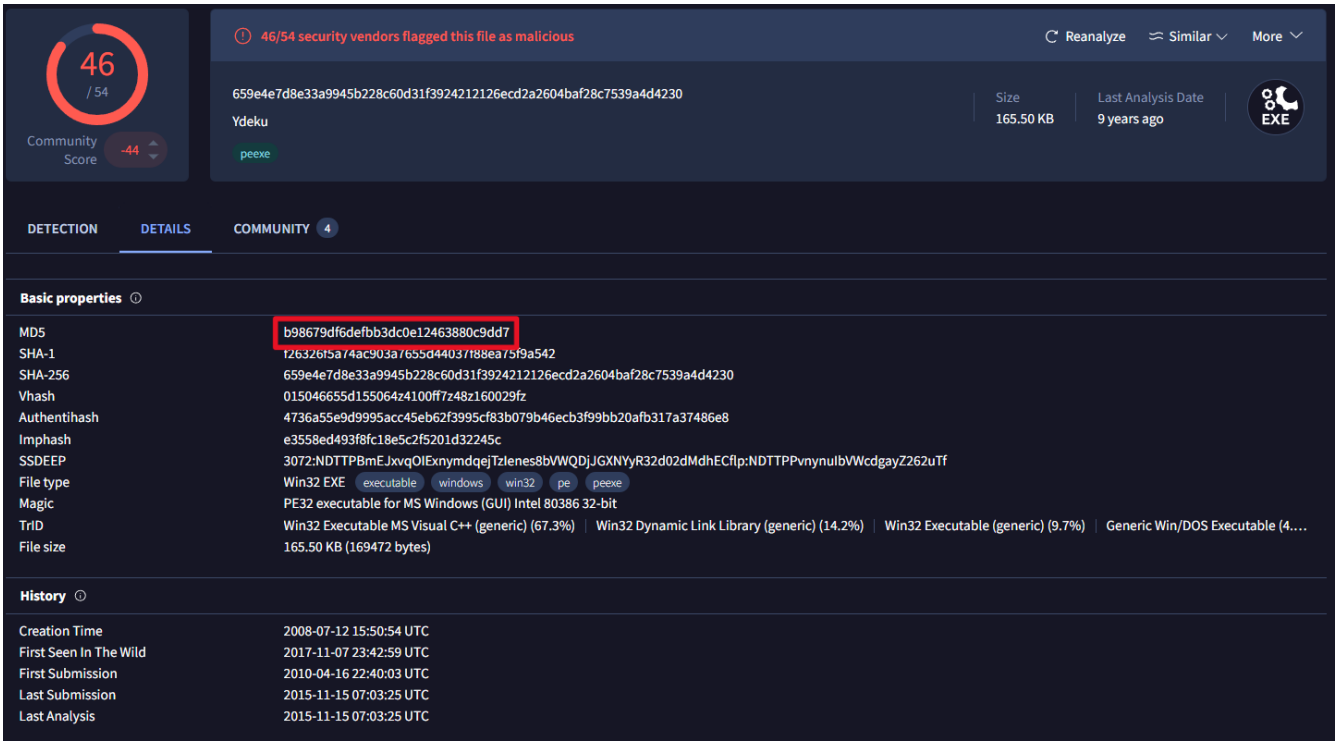
We need to see the full name and the full path of these processes

```
\Device\HarddiskVolume1\WINDOWS\explorer.exe    C:\WINDOWS\Explorer.EXE C:\WINDOWS\Explorer.EXE
\Device\HarddiskVolume1\Program Files\Immunity Inc\Immunity Debugger\ImmunityDebugger.exe       "C:\Program
\Device\HarddiskVolume1\Documents and Settings\Administrator\Desktop\b98679df6defbb3dc0e12463880c9dd7.exe
N/A     \Device\HarddiskVolume1\Documents and Settings\Administrator\Application Data\Obyt\ihah.exe     "C:
\Device\HarddiskVolume1\Program Files\Parallels\Parallels Tools\SIA\SharedIntApp.exe    "C:\Program Files\P
\Device\HarddiskVolume1\Program Files\Immunity Inc\Immunity Debugger\ImmunityDebugger.exe       "C:\Program
\Device\HarddiskVolume1\Documents and Settings\Administrator\Desktop\vaelh.exe "C:\Documents and Settings\
\Device\HarddiskVolume1\Program Files\Immunity Inc\Immunity Debugger\ImmunityDebugger.exe       "C:\Program
\Device\HarddiskVolume1\Documents and Settings\Administrator\Desktop\anaxu.exe  "C:\Documents and Settings\
\Device\HarddiskVolume1\Program Files\Java\jre6\bin\jusched.exe "C:\Program Files\Java\jre6\bin\jusched.exe
\Device\HarddiskVolume1\Program Files\Parallels\Parallels Tools\prl_cc.exe      "C:\Program Files\Parallels
\Device\HarddiskVolume1\Program Files\Immunity Inc\Immunity Debugger\ImmunityDebugger.exe       "C:\Program
\Device\HarddiskVolume1\Documents and Settings\Administrator\Desktop\nifek locked.exe   "C:\Documents and S
```

We can see the full name and path for the executed programs, there's one of them that has a long alphanumeric name that seems like a hash (**b98679df6defbb3dc0e12463880c9dd7**.exe) Let's check if it's a valid hash or not using hash analyzer online tool 🔗:

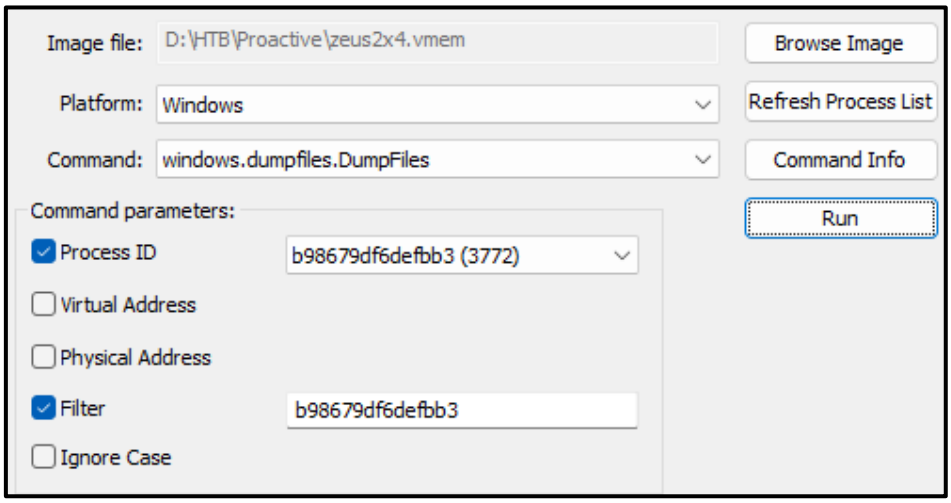


Now we are sure that the process name is a valid MD5 hash, Let's check Virustotal



There won't be a clear way to confirm if it's harmful or not such as Virustotal score.

Now let's dump the process and make a quick investigation into it



Using Exiftool on it to see the metadata of the file

```
┌──(kali㉿kali)-[/media/sf_HTB/Proactive]
└─$ exiftool file.0×82069028.0×82005ca0.ImageSectionObject.b98679df6defbb3dc0e12463880c9dd7.exe.img
ExifTool Version Number         : 13.00
File Name                       : file.0×82069028.0×82005ca0.ImageSectionObject.b98679df6defbb3dc0e12463880c9dd7.exe.img
Directory                       : .
File Size                       : 172 kB
File Modification Date/Time     : 2024:12:16 12:34:01-05:00
File Access Date/Time           : 2024:12:16 12:54:28-05:00
File Inode Change Date/Time     : 2024:12:16 12:34:20-05:00
File Permissions                : -rwxrwx——
File Type                       : Win32 EXE
File Type Extension             : exe
MIME Type                       : application/octet-stream
Machine Type                    : Intel 386 or later, and compatibles
Time Stamp                      : 2008:07:12 11:50:54-04:00
Image File Characteristics      : No relocs, Executable, 32-bit
PE Type                         : PE32
Linker Version                  : 9.0
Code Size                       : 141824
Initialized Data Size           : 59392
Uninitialized Data Size         : 0
Entry Point                     : 0×1880b
OS Version                      : 5.0
Image Version                   : 0.0
Subsystem Version               : 5.0
Subsystem                       : Windows GUI
File Version Number             : 10.6.1094.74
Product Version Number          : 10.6.1094.74
File Flags Mask                 : 0×003f
File Flags                      : (none)
File OS                         : Windows NT 32-bit
Object File Type                : Executable application
File Subtype                    : 0
Language Code                   : English (U.S.)
Character Set                   : Unicode
Company Name                    : Yldaepupreapagytek
File Description                : Iqkyliybotetafwa
Original File Name              : Ydeku
Internal Name                   : Wefietrenuyz
Legal Copyright                 : Egewylvekeez
Legal Trademarks                : Ufhokibiedqereo
Product Name                    : Iwhahu
```

We can see these attributes are not specific or even written using an appropriate English language so it's more suspicious to be malicious, we can see the original file name is the same file name we see on virustotal (**Ydeku**).

Now let's return to this image and analyze the rest suspicious files that are child processes for this malicious file:

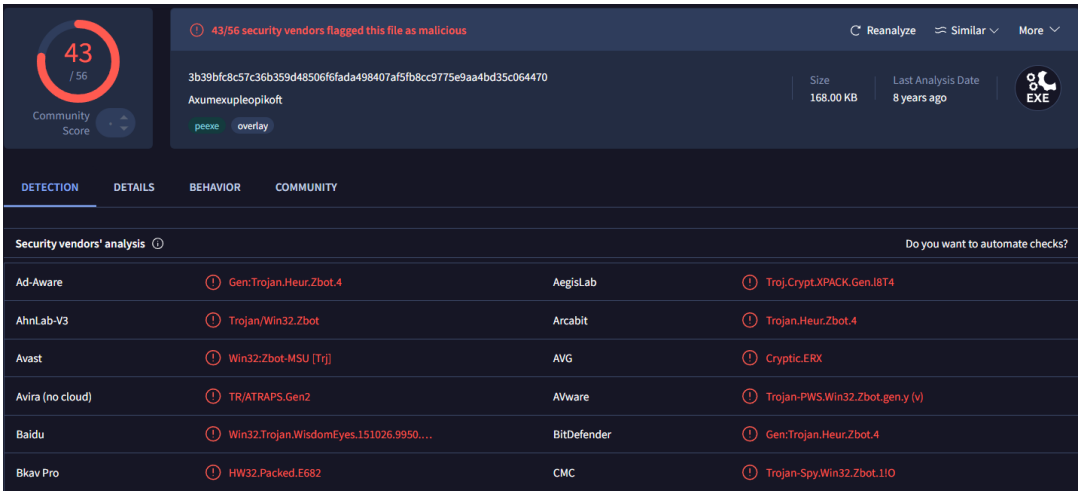
```
1752    1720    explorer.exe     0x821b2020      22      520     0
* 2404  1752    ImmunityDebugge  0x82066478      2       85      0
** 3772 2404    b98679df6defbb3  0x81f4bb28      1       46      0
*** 3276        3772    ihah.exe         0x81e87da0      1       45
* 1900  1752    SharedIntApp.ex  0x822b96c0      3       75      0
* 1932  1752    ImmunityDebugge  0x82282380      2       86      0
** 952  1932    vaelh.exe        0x8223c020      2       40      0
* 3788  1752    ImmunityDebugge  0x81ffb6d8      2       103     0
** 3508 3788    anaxu.exe        0x8219e5c8      2       54      0
* 1936  1752    jusched.exe      0x8212ada0      1       43      0
* 1908  1752    prl_cc.exe       0x820ee580      14      133     0
* 2972  1752    ImmunityDebugge  0x82001ad0      2       87      0
** 2204 2972    nifek_locked.ex  0x8207bda0      2       38      0
```

Now Let's Dump (**ihah.exe**) and investigate it:

```
┌──(kali㉿kali)-[/media/sf_HTB/Proactive]
└─$ exiftool file.0×81e815c8.0×82247520.ImageSectionObject.ihah.exe.img
ExifTool Version Number         : 13.00
File Name                       : file.0×81e815c8.0×82247520.ImageSectionObject.ihah.exe.img
Directory                       : .
File Size                       : 172 kB
File Modification Date/Time     : 2024:12:16 13:27:21-05:00
File Access Date/Time           : 2024:12:16 13:27:46-05:00
File Inode Change Date/Time     : 2024:12:16 13:27:37-05:00
File Permissions                : -rwxrwx——
File Type                       : Win32 EXE
File Type Extension             : exe
MIME Type                       : application/octet-stream
Machine Type                    : Intel 386 or later, and compatibles
Time Stamp                      : 2008:01:25 09:18:42-05:00
Image File Characteristics      : No relocs, Executable, 32-bit
PE Type                         : PE32
Linker Version                  : 9.0
Code Size                       : 142336
Initialized Data Size           : 59392
Uninitialized Data Size         : 0
Entry Point                     : 0×c294
OS Version                      : 5.0
Image Version                   : 0.0
Subsystem Version               : 5.0
Subsystem                       : Windows GUI
File Version Number             : 7.3.2484.1905
Product Version Number          : 7.3.2484.1905
File Flags Mask                 : 0×003f
File Flags                      : (none)
File OS                         : Windows NT 32-bit
Object File Type                : Executable application
File Subtype                    : 0
Language Code                   : English (U.S.)
Character Set                   : Unicode
Company Name                    : Ynagzakoykfeziucbye
File Description                : Vuupfaopescoydba
Original File Name              : Axumexupleopikoft
Internal Name                   : Wyzesofooqbukyabykb
Legal Copyright                 : Haezmoaxcamedua
Legal Trademarks                : Ziypewurkaucitamifg
Product Name                    : Bydoycgui
```

Also, another random characters that doesn't represent a name at all.

The original name of the file (**Axumexupleopikoft**) that we got from Exiftool, is the same as shown in virustotal.



And so on for all other files, all of them has a High Community Score and seems to be malicious

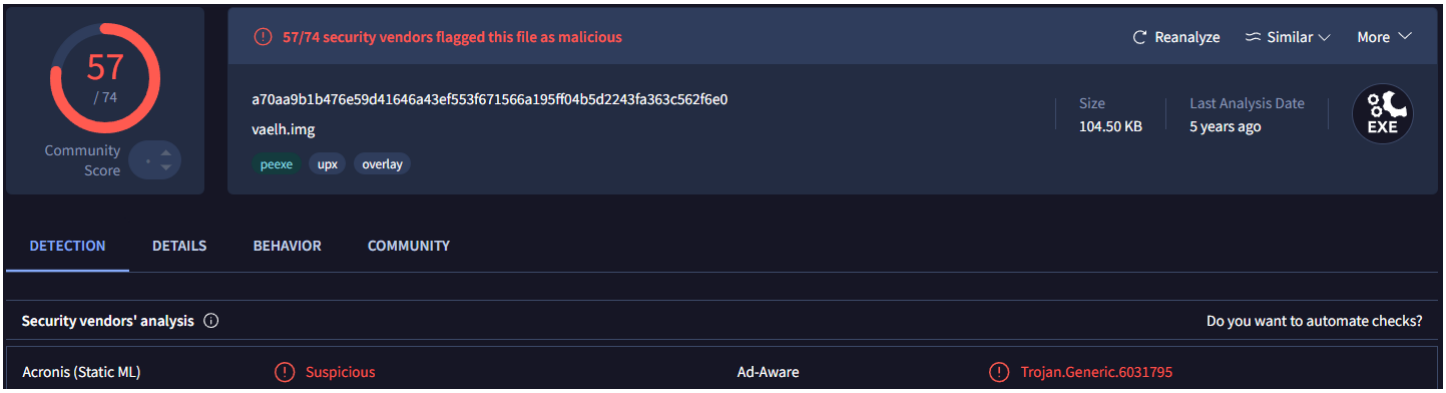
```
1752    1720    explorer.exe    0x821b2020      22      520     0
* 2404  1752    ImmunityDebugge 0x82066478      2       85      0
** 3772 2404    b98679df6defbb3 0x81f4bb28      1       46      0
*** 3276        3772    ihah.exe        0x81e87da0      1       45
* 1900  1752    SharedIntApp.ex 0x822b96c0      3       75      0
* 1932  1752    ImmunityDebugge 0x82282380      2       86      0
** 952  1932    vaelh.exe       0x8223c020      2       40      0
* 3788  1752    ImmunityDebugge 0x81ffb6d8      2       103     0
** 3508 3788    anaxu.exe       0x8219e5c8      2       54      0
* 1936  1752    jusched.exe     0x8212ada0      1       43      0
* 1908  1752    prl_cc.exe      0x820ee580      14      133     0
* 2972  1752    ImmunityDebugge 0x82001ad0      2       87      0
** 2204 2972    nifek_locked.ex 0x8207bda0      2       38      0
```

vaelh.exe Analysis :

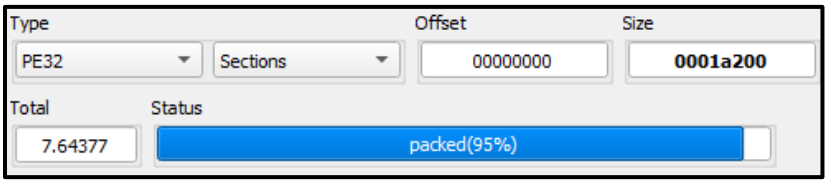
```
┌──(kali㉿kali)-[/media/sf_HTB/Proactive]
└─$ exiftool file.0×82263470.0×820f3eb8.ImageSectionObject.vaelh.exe.img
ExifTool Version Number         : 13.00
File Name                       : file.0×82263470.0×820f3eb8.ImageSectionObject.vaelh.exe.img
Directory                       : .
File Size                       : 107 kB
File Modification Date/Time     : 2024:12:16 13:36:50-05:00
File Access Date/Time           : 2024:12:16 13:39:38-05:00
File Inode Change Date/Time     : 2024:12:16 13:37:02-05:00
File Permissions                : -rwxrwx——
File Type                       : Win32 EXE
File Type Extension             : exe
MIME Type                       : application/octet-stream
Machine Type                    : Intel 386 or later, and compatibles
Time Stamp                      : 1998:06:28 02:34:08-04:00
Image File Characteristics      : No relocs, Executable, 32-bit
PE Type                         : PE32
Linker Version                  : 8.8
Code Size                       : 98304
Initialized Data Size           : 8192
Uninitialized Data Size         : 36864
Entry Point                     : 0×21c90
OS Version                      : 4.0
Image Version                   : 0.0
Subsystem Version               : 4.0
Subsystem                       : Windows GUI
File Version Number             : 8.5.0.3
Product Version Number          : 8.5.0.3
File Flags Mask                 : 0×003f
File Flags                      : (none)
File OS                         : Windows NT 32-bit
Object File Type                : Dynamic link library
File Subtype                    : 0
Language Code                   : Neutral
Character Set                   : Unicode
File Version                    : 8.5.0.3
Product Version                 : 8.5.0.3
File Description                : Kaspersky Anti-Virus
Company Name                    : Kaspersky Lab
Legal Copyright                 : Copyright © Kaspersky Lab 1997-2009.
Product Name                    : Kaspersky Anti-Virus
Legal Trademarks                : Kaspersky™ Anti-Virus ®  is registered trademark of Kaspersky Lab.
Internal Name                   : AVP
```

Based on the information from the metadata of the (**vaelh.exe**) file we can see that it doesn't seem suspicious , whether virustotal has a high score to me malicious,

But virustotal has a high score based on 57 security vendors that flagged this as a malicious file, whether it seems that it has a valid File Description, Company name, legal copyright, product name. As shown in the following image it seems that it's packed with UPX



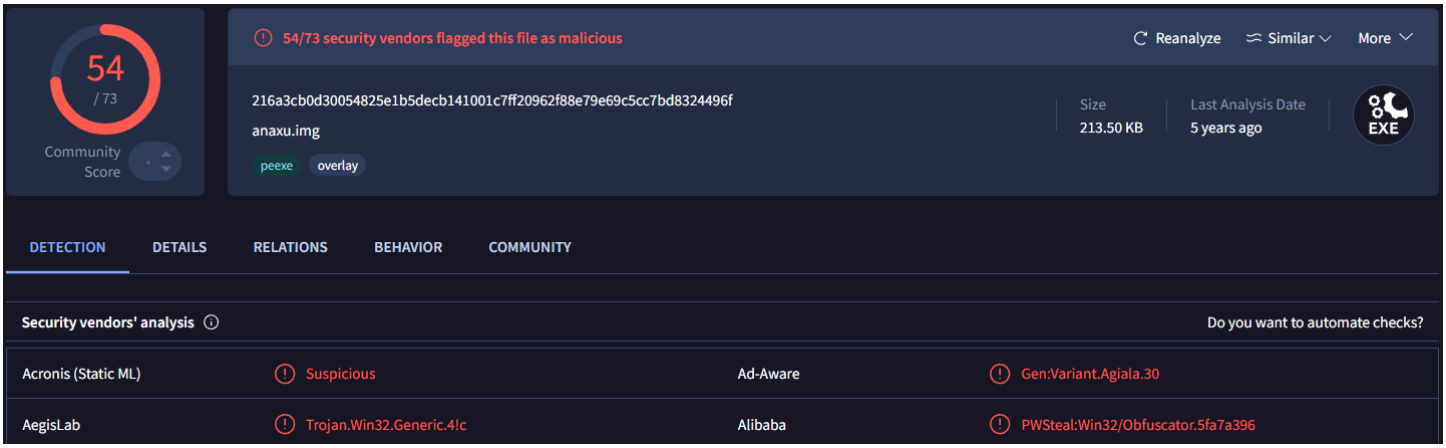
We can see it also from Detect it easy that it's packed with UPX and has a high entropy in the code which means that the file contains data that appears highly random or lacks recognizable patterns. (Entropy is a measure of uncertainty or randomness in data)



anaxu.exe Analysis:

```
┌──(kali㉿kali)-[/media/sf_HTB/Proactive]
└─$ exiftool file.0×82386bb0.0×8204f288.ImageSectionObject.anaxu.exe.img
ExifTool Version Number         : 13.00
File Name                       : file.0×82386bb0.0×8204f288.ImageSectionObject.anaxu.exe.img
Directory                       : .
File Size                       : 219 kB
File Modification Date/Time     : 2024:12:16 13:51:46-05:00
File Access Date/Time           : 2024:12:16 13:52:14-05:00
File Inode Change Date/Time     : 2024:12:16 13:51:58-05:00
File Permissions                : -rwxrwx---
File Type                       : Win32 EXE
File Type Extension             : exe
MIME Type                       : application/octet-stream
Machine Type                    : Intel 386 or later, and compatibles
Time Stamp                      : 2005:07:23 17:40:54-04:00
Image File Characteristics      : Executable, No line numbers, No symbols, 32-bit
PE Type                         : PE32
Linker Version                  : 7.0
Code Size                       : 53248
Initialized Data Size           : 185344
Uninitialized Data Size         : 0
Entry Point                     : 0×da30
OS Version                      : 8.4
Image Version                   : 6.0
Subsystem Version               : 4.0
Subsystem                       : Windows GUI
File Version Number             : 60.40.109.60
Product Version Number          : 60.40.109.60
File Flags Mask                 : 0×003f
File Flags                      : (none)
File OS                         : Windows NT 32-bit
Object File Type                : Executable application
File Subtype                    : 0
Language Code                   : Unknown (0681)
Character Set                   : Unknown (7EA3)
Company Name                    : ЛюЛрОюзХрИчДъфцННйЗХоБчСШжИ
File Description                : цЩхЫжГдЬЬх6рчЩздпНИИдЩьдЛ
File Version                    : 60.40.109.60
Internal Name                   : ИфиврeоЫкУглКцЮюБКилншэШф
Legal Copyright                 : 7119-6856
Original File Name              : 6sHkp.exe
Product Name                    : стГЦЖтЕЩЬХЖУдгюГТвРраЖЖ
Product Version                 : 60.40.109.60
```

Seems so suspicious to me, let's check virustotal also:



NOTE: I didn't use (**windows.malfind**) plugin as it's not precise also, all files that I scanned above **malfind** found in addition to a lot of other false positive files, so I don't recommend using it.

All these files that we scanned were malicious, so let's see the network stats

We have two plugins in **volatility 2** (**connscan**) which makes pool scanner for tcp connections, and (**connections**) which Prints list of open connections

connscan plugin

Offset(P)	Local Address	Remote Address	Pid
0×020f5410	10.211.55.5:1427	65.54.81.89:80	1084
0×02125008	10.211.55.5:1423	207.46.21.123:80	1084
0×022ace08	10.211.55.5:1432	193.43.134.14:80	1752

pid 1084 → svchost.exe

pid 1752 → explorer.exe

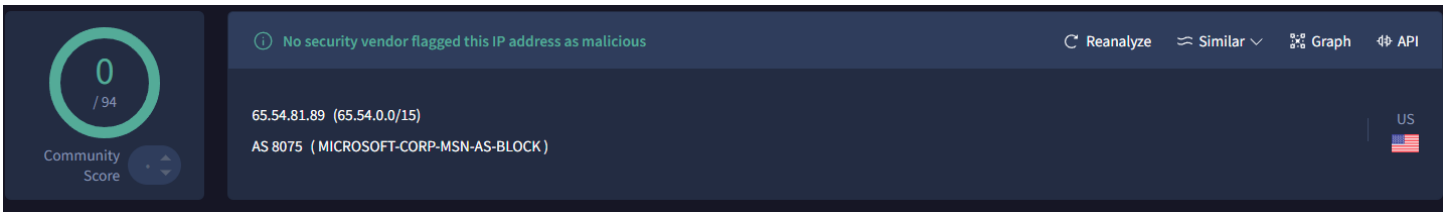
connection plugin

Offset(V)	Local Address	Remote Address	Pid
0×822ace08	10.211.55.5:1432	193.43.134.14:80	1752

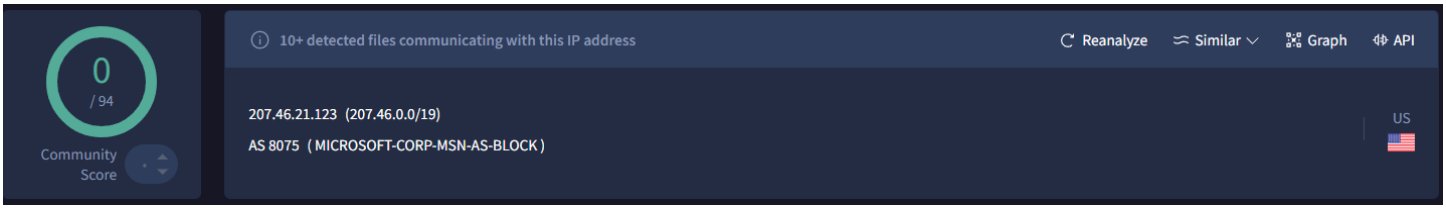
Explanation:

Since the malicious file has ben executed from the explorer, (**ppid of his ppid is explorer.exe**) , so we will focus on the explorer.exe connection, also because the other connections in connscan had belonged to svchost.exe, which is the main process of any running windows machine. Let's also check what I've said using virustotal

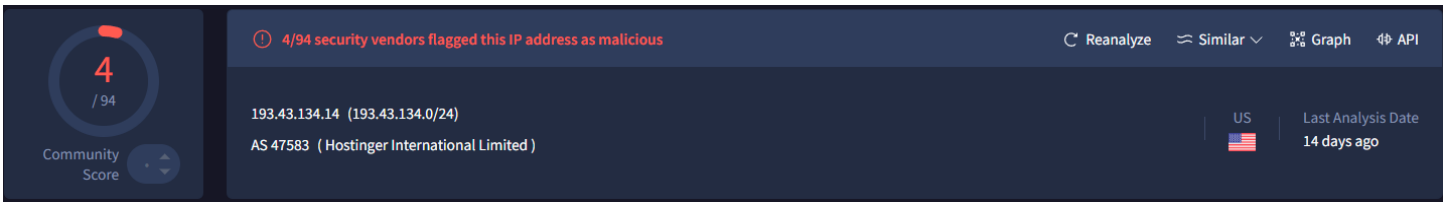
Scanning the ip address (**65.54.81.89**) which belongs to (**svchost.exe**) process



Scanning the ip address (**207.46.21.123**) which belongs to (**svchost.exe**) process

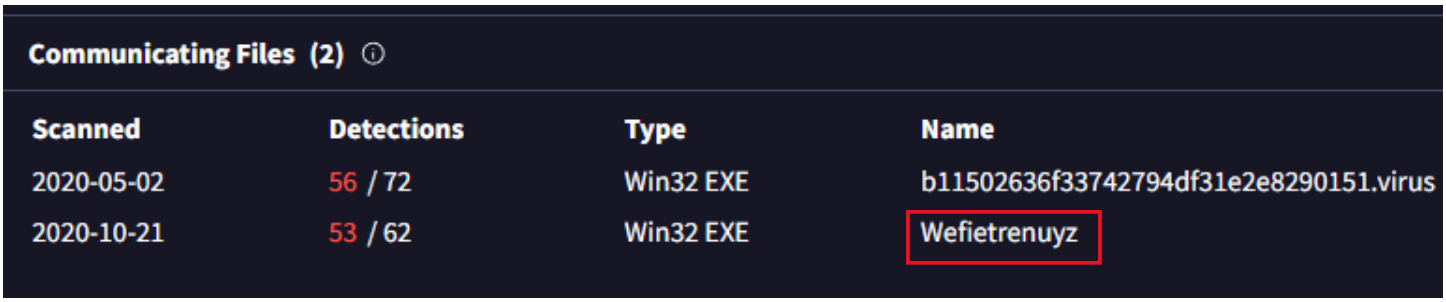


Now, Scanning the ip address (**193.43.134.14**) which belongs to (**explorer.exe**) process

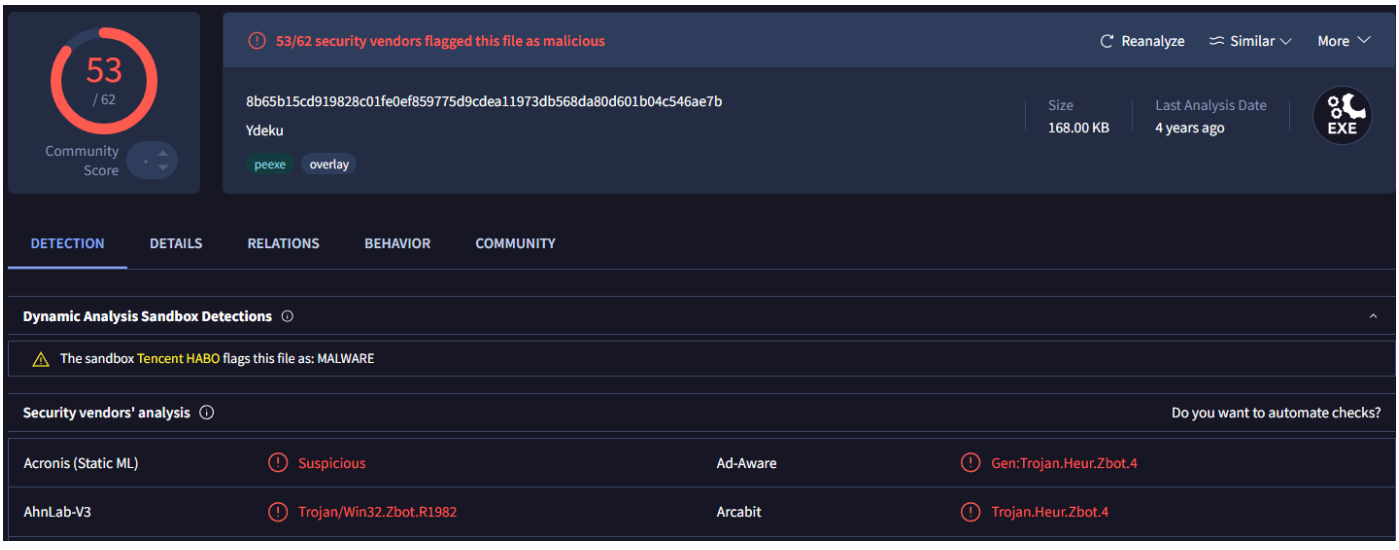


So, I'm right, the malicious ip address must be related to (**explorer.exe**) process

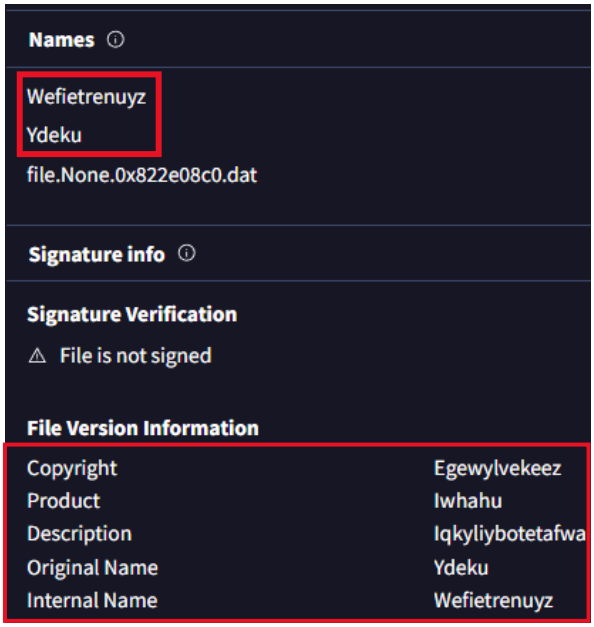
Also, it has suspicious files dropped related to that ip address (**193.43.134.14**)



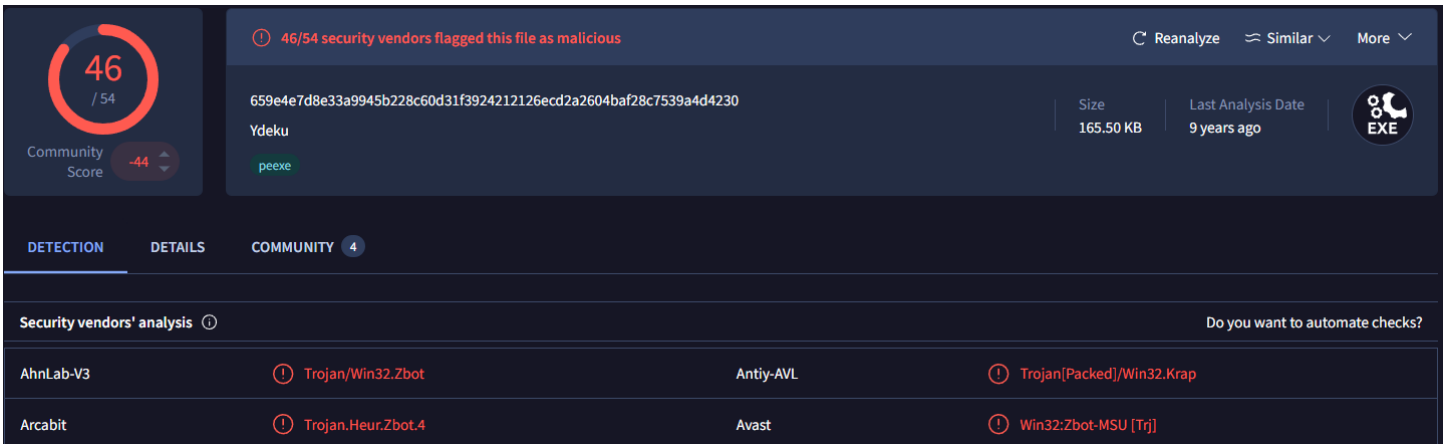
Let's check the second one :



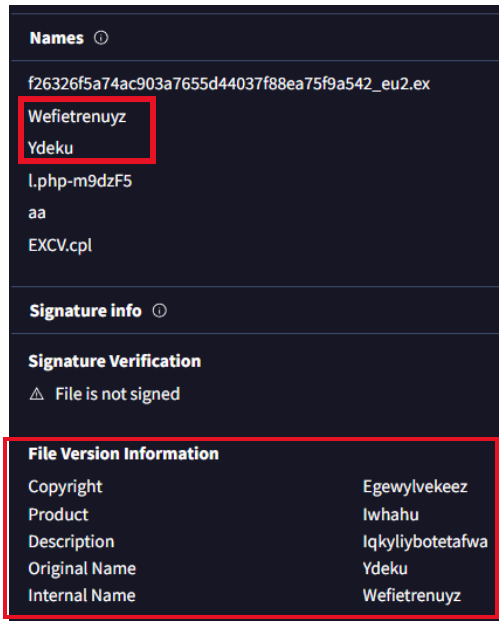
(Ydeku), I think we've seen this before. Let's go to Details section and get the names and Signature info :



From the first malicious process which its name was a valid MD5 hash (**b98679df6defbb3dc0e12463880c9dd7.exe**) has this scan :



And has these details:



YARA Rules

Create rules based on all malicious files that we've got from the memory dump and from Zeus Banking Malware.

First, Creating a custom rule based on all malicious files, all malicious file we need :

- **invoice_2318362983713_823931342io.pdf.exe**
- **b98679df6defbb3dc0e12463880c9dd7.exe**
- **ihah.exe**
- **vaelh.exe**
- **anaxu.exe**

```
┌──(kali㉿kali)-[~/Desktop/Proactive]
└─$ ls
file.0×81e815c8.0×82247520.ImageSectionObject.ihah.exe.img
file.0×82069028.0×82005ca0.ImageSectionObject.b98679df6defbb3dc0e12463880c9dd7.exe.img
file.0×82263470.0×820f3eb8.ImageSectionObject.vaelh.exe.img
file.0×82386bb0.0×8204f288.ImageSectionObject.anaxu.exe.img
invoice_2318362983713_8239131342io.pdf.exe
```

Generating the YARA rules using this command:

```
└─$ sudo python3 yarGen.py -m /home/kali/Desktop/Proactive/ -o /home/kali/Desktop/YaraGen.yar
```

```
┌──(kali㉿kali)-[~/yarGen]
└─$ sudo python3 yarGen.py -m /home/kali/Desktop/Proactive/ -o /home/kali/Desktop/YaraGen.yar
/home/kali/yarGen/yarGen.py:1152: SyntaxWarning: invalid escape sequence '\w'
  cleanedName = re.sub('[^\w]', r'_', cleanedName)
/home/kali/yarGen/yarGen.py:1354: SyntaxWarning: invalid escape sequence '\w'
  rule_name = re.sub('[^\w]', r'_', rule_name)
/home/kali/yarGen/yarGen.py:1494: SyntaxWarning: invalid escape sequence '\w'
  cleanedName = re.sub('[^\w]', r'_', cleanedName)
________________________________________________________________________

                    _____
   __ _____ _____/ ___/__ ___
  / // / _ `/ __/ (_ / -_) _ \
  \_, /\_,_/_/  \___/\__/_//_/
 /___/  Yara Rule Generator
        Florian Roth, August 2023, Version 0.24.0

  Note: Rules have to be post-processed
  See this post for details: https://medium.com/@cyb3rops/121d29322282
_____________________________________________________________________-
[+] Using identifier 'Proactive'
[+] Using reference 'https://github.com/Neo23×0/yarGen'
[+] Using prefix 'Proactive'
[+] Processing PEStudio strings ...
[+] Reading goodware strings from database 'good-strings.db' ...
    (This could take some time and uses several Gigabytes of RAM depending on your db size)
```

Let’s read a sample of the YARA Rule that we generated:

```
/*
   YARA Rule Set
   Author: yarGen Rule Generator
   Date: 2024-12-16
   Identifier: Malicious_Findings
   Reference: https://github.com/Neo23x0/yarGen
*/

/* Rule Set ----------------------------------------------------------------- */

rule _home_kali_Desktop_Malicious_Findings_ihah {
   meta:
      description = "Malicious_Findings - file ihah.exe"
      author = "yarGen Rule Generator"
      reference = "https://github.com/Neo23x0/yarGen"
      date = "2024-12-16"
      hash1 = "3b39bfc8c57c36b359d48506f6fada498407af5fb8cc9775e9aa4bd35c064470"
   strings:
      $s1 = "        <requestedExecutionLevel level=\"asInvoker\" uiAccess=\"false\"></requestedExecutionLevel>" fullword ascii
      $s2 = "AAA8AAAAAAAA" ascii /* base64 encoded string '<<<<<<' */
      $s3 = "  <description>Kea i l hiywildy ow</description>" fullword ascii
      $s4 = "      processorArchitecture=\"*\"" fullword ascii
      $s5 = "          processorArchitecture=\"X86\"" fullword ascii
      $s6 = "          publicKeyToken=\"6595b64144ccf1df\"" fullword ascii
      $s7 = "          version=\"6.0.0.0\"" fullword ascii
      $s8 = "  <trustInfo xmlns=\"urn:schemas-microsoft-com:asm.v3\">" fullword ascii
      $s9 = "IIIJIIVIII" fullword ascii
      $s10 = "BBBBBBBBHBBBBBBB" fullword ascii
      $s11 = "JJJJJJJJJHJJJJJ" fullword ascii
      $s12 = "IIIVIIBI" fullword ascii
      $s13 = "JJFJJJJJ" fullword ascii
      $s14 = "DINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDING" fullword ascii
      $s15 = "          name=\"Microsoft.Windows.Common-Controls\"" fullword ascii
      $s16 = "Ynagzakoykfeziucbye" fullword wide
      $s17 = "Vuupfaopescoydba" fullword wide
      $s18 = "Axumexupleopikoft" fullword wide
      $s19 = "Wyzesofooqbukyabykb" fullword wide
      $s20 = "Ziypewurkaucitamifg" fullword wide
   condition:file_0x82069028_0x82005ca0_ImageSectionObject_b98679df6defbb3dc0e12463880c9dd7_exe
      uint16(0) == 0x5a4d and filesize < 500KB and
      8 of them
}
```

Now let’s run the YARA rule on all malicious files inside (Malicious_Findings) Directory using this command:

1. `└─$ yara YaraGen.yar Malicious_Findings`

```
┌──(kali㉿kali)-[~/Desktop]
└─$ yara YaraGen.yar Malicious_Findings
_home_kali_Desktop_Malicious_Findings_ihah Malicious_Findings/ihah.exe
b98679df6defbb3dc0e12463880c9dd7 Malicious_Findings/ihah.exe
_home_kali_Desktop_Malicious_Findings_vaelh Malicious_Findings/vaelh.exe
_home_kali_Desktop_Malicious_Findings_anaxu Malicious_Findings/anaxu.exe
invoice_2318362983713_823931342io_pdf Malicious_Findings/invoice_2318362983713_823931342io.pdf.exe
b98679df6defbb3dc0e12463880c9dd7 Malicious_Findings/b98679df6defbb3dc0e12463880c9dd7.exe
```

Run YARA command to print matching strings (Here's a sample of the whole output):

```
1.  └─$ yara -s YaraGen.yar Malicious_Findings
```

```
┌──(kali㉿kali)-[~/Desktop]
└─$ yara -s YaraGen.yar Malicious_Findings
_home_kali_Desktop_Malicious_Findings_ihah Malicious_Findings/ihah.exe
0x29664:$s1:          <requestedExecutionLevel level="asInvoker" uiAccess="false"></requestedExecutionLevel>
0x7e68:$s2: AAA8AAAAAAAA
0x83ba:$s2: AAA8AAAAAAAA
0x29481:$s3:    <description>Kea i l hiywildy ow</description>
0x29424:$s4:        processorArchitecture="*"
0x2955e:$s5:            processorArchitecture="X86"
0x29585:$s6:            publicKeyToken="6595b64144ccf1df"
0x29541:$s7:            version="6.0.0.0"
0x295ff:$s8:    <trustInfo xmlns="urn:schemas-microsoft-com:asm.v3">
0x84e1:$s9: IIIJIIVIII
0x8373:$s10: BBBBBBBBHBBBBBBBB
0x7f67:$s11: JJJJJJJJJHJJJJJ
0x8220:$s12: IIIVIIBI
0x81c1:$s13: JJFJJJJJ
0x2950d:$s15:           name="Microsoft.Windows.Common-Controls"
0x29158:$s16: Y\x00n\x00a\x00g\x00z\x00a\x00k\x00o\x00y\x00k\x00f\x00e\x00z\x00i\x00u\x00c\x00b\x00y\x00e\x00
0x291a8:$s17: V\x00u\x00u\x00p\x00f\x00a\x00o\x00p\x00e\x00s\x00c\x00o\x00y\x00d\x00b\x00a\x00
0x291f4:$s18: A\x00x\x00u\x00m\x00e\x00x\x00u\x00p\x00l\x00e\x00o\x00p\x00i\x00k\x00o\x00f\x00t\x00
0x29238:$s19: W\x00y\x00z\x00e\x00s\x00o\x00f\x00o\x00o\x00q\x00b\x00u\x00k\x00y\x00a\x00b\x00y\x00k\x00b\x00
0x292cc:$s20: Z\x00i\x00y\x00p\x00e\x00w\x00u\x00r\x00k\x00a\x00u\x00c\x00i\x00t\x00a\x00m\x00i\x00f\x00g\x00
b98679df6defbb3dc0e12463880c9dd7 Malicious_Findings/ihah.exe
0x29664:$s1:          <requestedExecutionLevel level="asInvoker" uiAccess="false"></requestedExecutionLevel>
0x29562:$s3:        processorArchitecture="X86"
0x29585:$s5:            publicKeyToken="6595b64144ccf1df"
0x29541:$s7:            version="6.0.0.0"
0x295ff:$s8:    <trustInfo xmlns="urn:schemas-microsoft-com:asm.v3">
0x29712:$s9: DDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPA
0x29722:$s9: DDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPA
0x29732:$s9: DDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPA
0x29742:$s9: DDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPA
0x29752:$s9: DDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPA
0x29762:$s9: DDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPA
0x29772:$s9: DDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPADDINGXXPADDINGPA
0x2950d:$s11:           name="Microsoft.Windows.Common-Controls"
0x294b3:$s17:   <dependency>
0x296f3:$s18:   </trustInfo>
0x295ee:$s19:   </dependency>
_home_kali_Desktop_Malicious_Findings_vaelh Malicious_Findings/vaelh.exe
0x19891:$s1:        <requestedExecutionLevel level="asInvoker" uiAccess="false"/>
0x17b25:$s2: GetCompu
```