GENERAL INFORMATION

Mission Name

ForensicsTest

History Background

In the previous mission, ETHAN is arrested by Claire and Tarain. Claire and Ethan are going to work together to catch the SHAX hackers and both must be tested on their skills.

Technical High-Level Overview

Player must put in practice his knowledge about recovering event logs. The goal of this mission is to catch the application used by Phaldra to hide his footprints.

Short Mission Description

You´re going to be tested to ensure your forensic knowledge is up to date. Please catch the threat name of the process ID. Please insert it in Hexadecimal, like this 0xAABB.

Mission Description

Player must put in practice his knowledge about locating PID (Process ID). You´re going to be tested to ensure that your knowledge is up to date. Please catch Process ID of the binary whose parent process ID is 4804 on Phaldras´s computer in 2049 when Phaldra tried to hide his footprints. Please insert it in Hexadecimal, like this 0xAABB.

Location

SYLVARCON | EBAND DEPARTMENT - RECON HQ

Tools

- FTK Imager
- Event Log Explorer
- EvtxCMD
- Bulkextractor
- Timeline Explorer

Questions

Which executable was used to clear event logs? Insert the full name, including dot.exe

- wevtutil.exe

What was the full command line used to clear event log?

- wevtutil cl security

Which is the security event id which shows that Security event logs were cleared?

- 1102

Hints

1. Use Bulk extractor to recover deleted evtx.
2. Use Evtxcmd to analyse evtx files recovered.
3. Filter information using Timeline Explorer and PID 18B0

Write Up

Linux Method - no root necessary

- apt install libevtx-utils
- apt install python3-evtx
- mount
- ewfmount
- https://github.com/williballenthin/python-evtx
- Git clone https://github.com/williballenthin/python-evtx.git

Windows Method

First of all, player should mount evidence provided to extract event logs, using FTK imager:

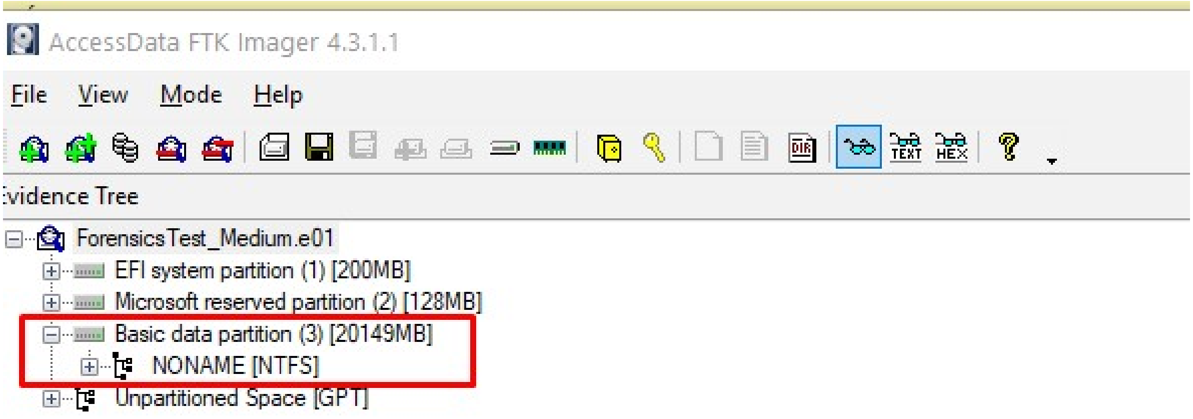
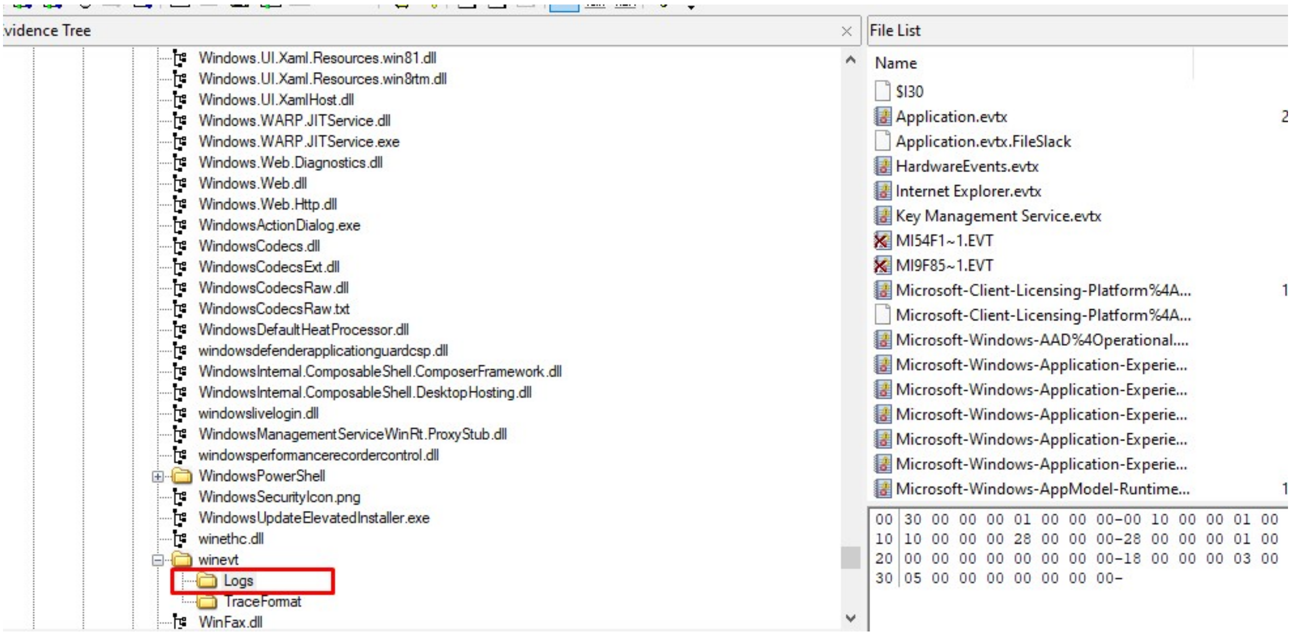

AccessData FTK Imager 4.3.1.1
File
View
Mode
Help
TEXT
HEX
:vidence Tree
ForensicsTest_Medium.e01
EFI system partition (1) [200MB]
Microsoft reserved partition (2) [128MB]
Basic data partition (3) [20149MB]
NONAME [NTFS]
Unpartitioned Space [GPT]

Figure 1

Later, player must extract security event logs located at: C:\Windows\System32\winevt\Logs:



vidence Tree

File List

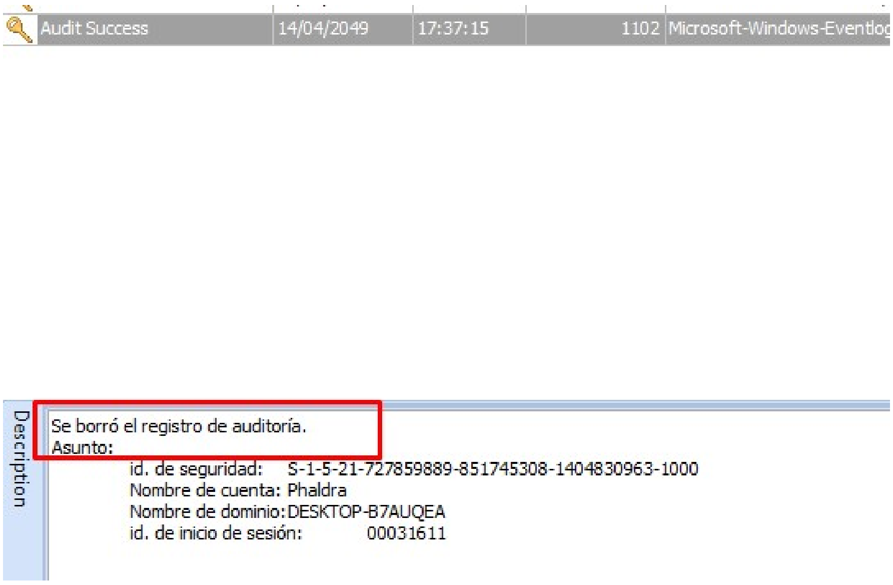
```
Windows.UI.Xaml.Resources.win81.dll
Windows.UI.Xaml.Resources.win8rtm.dll
Windows.UI.XamlHost.dll
Windows.WARP.JITService.dll
Windows.WARP.JITService.exe
Windows.Web.Diagnostics.dll
Windows.Web.dll
Windows.Web.Http.dll
WindowsActionDialog.exe
WindowsCodecs.dll
WindowsCodecsExt.dll
WindowsCodecsRaw.dll
WindowsCodecsRaw.txt
WindowsDefaultHeatProcessor.dll
windowsdefenderapplicationguardcsp.dll
WindowsInternal.ComposableShell.ComposerFramework.dll
WindowsInternal.ComposableShell.DesktopHosting.dll
windowslivelogin.dll
WindowsManagementServiceWinRt.ProxyStub.dll
windowsperformancerecordercontrol.dll
WindowsPowerShell
WindowsSecurityIcon.png
WindowsUpdateElevatedInstaller.exe
winethc.dll
winevt
    Logs
    TraceFormat
WinFax.dll
```

```
Name
$I30
Application.evtx                                   2
Application.evtx.FileSlack
HardwareEvents.evtx
Internet Explorer.evtx
Key Management Service.evtx
MI54F1~1.EVT
MI9F85~1.EVT
Microsoft-Client-Licensing-Platform%4A...          1
Microsoft-Client-Licensing-Platform%4A...
Microsoft-Windows-AAD%4Operational....
Microsoft-Windows-Application-Experie...
Microsoft-Windows-Application-Experie...
Microsoft-Windows-Application-Experie...
Microsoft-Windows-Application-Experie...
Microsoft-Windows-Application-Experie...
Microsoft-Windows-AppModel-Runtime...              1
```

```
00 30 00 00 00 01 00 00 00-00 10 00 00 01 00
10 10 00 00 00 28 00 00 00-28 00 00 00 01 00
20 00 00 00 00 00 00 00 00-18 00 00 03 00
30 05 00 00 00 00 00 00 00-
```

Figure 2

Double click on Logs and the Export the whole folder. After, player should analyse Security event log to identify if there is any event:



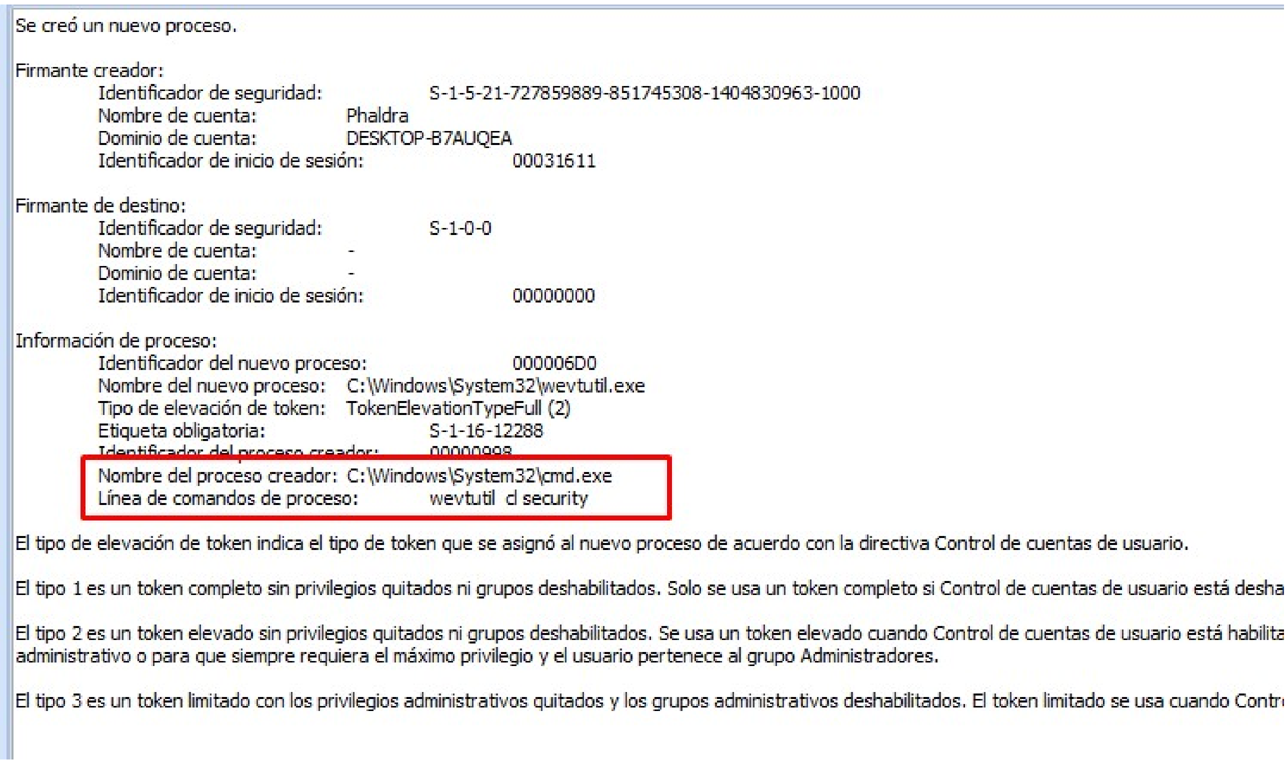
```
Audit Success    14/04/2049    17:37:15    1102 Microsoft-Windows-Eventlog
```

Description

```
Se borró el registro de auditoría.
Asunto:
        id. de seguridad:     S-1-5-21-727859889-851745308-1404830963-1000
        Nombre de cuenta: Phaldra
        Nombre de dominio: DESKTOP-B7AUQEA
        id. de inicio de sesión:        00031611
```

Figure 3

As we can see above, there is an 1102 evernt whidh indicates that Security events logs were cleared and other event shows how security events were cleared:



```
Se creó un nuevo proceso.

Firmante creador:
        Identificador de seguridad:             S-1-5-21-727859889-851745308-1404830963-1000
        Nombre de cuenta:               Phaldra
        Dominio de cuenta:              DESKTOP-B7AUQEA
        Identificador de inicio de sesión:              00031611

Firmante de destino:
        Identificador de seguridad:             S-1-0-0
        Nombre de cuenta:               -
        Dominio de cuenta:              -
        Identificador de inicio de sesión:              00000000

Información de proceso:
        Identificador del nuevo proceso:                000006D0
        Nombre del nuevo proceso:       C:\Windows\System32\wevtutil.exe
        Tipo de elevación de token:     TokenElevationTypeFull (2)
        Etiqueta obligatoria:                   S-1-16-12288
        Identificador del proceso creador:      00000998
        Nombre del proceso creador:     C:\Windows\System32\cmd.exe
        Línea de comandos de proceso:           wevtutil  cl security

El tipo de elevación de token indica el tipo de token que se asignó al nuevo proceso de acuerdo con la directiva Control de cuentas de usuario.

El tipo 1 es un token completo sin privilegios quitados ni grupos deshabilitados. Solo se usa un token completo si Control de cuentas de usuario está desha

El tipo 2 es un token elevado sin privilegios quitados ni grupos deshabilitados. Se usa un token elevado cuando Control de cuentas de usuario está habilita
administrativo o para que siempre requiera el máximo privilegio y el usuario pertenece al grupo Administradores.

El tipo 3 es un token limitado con los privilegios administrativos quitados y los grupos administrativos deshabilitados. El token limitado se usa cuando Contr
```

Figure 4

Considering this behaviour, it´s essential to perform a recover task to get deleted Security event logs. T. To achieve this, player should use Bulk Extractor to load the Encase image and select proper checks to recovery evtx files:

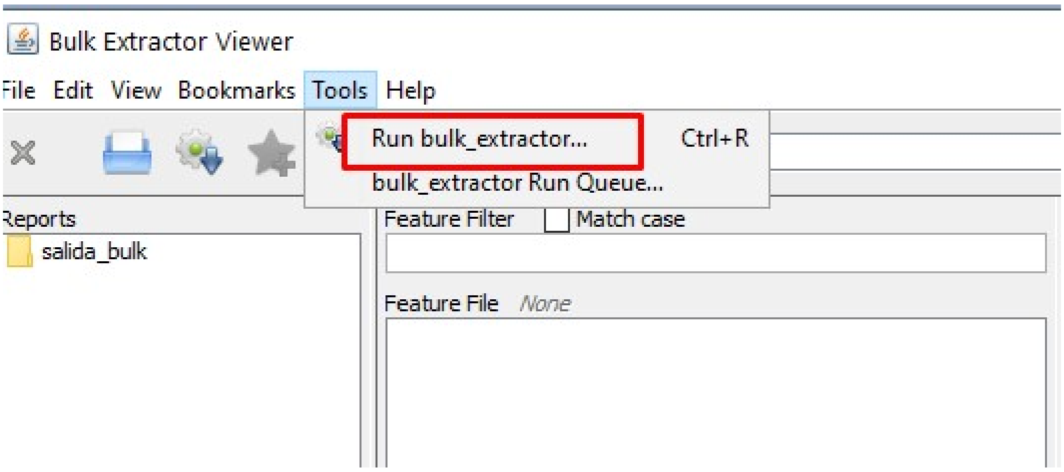
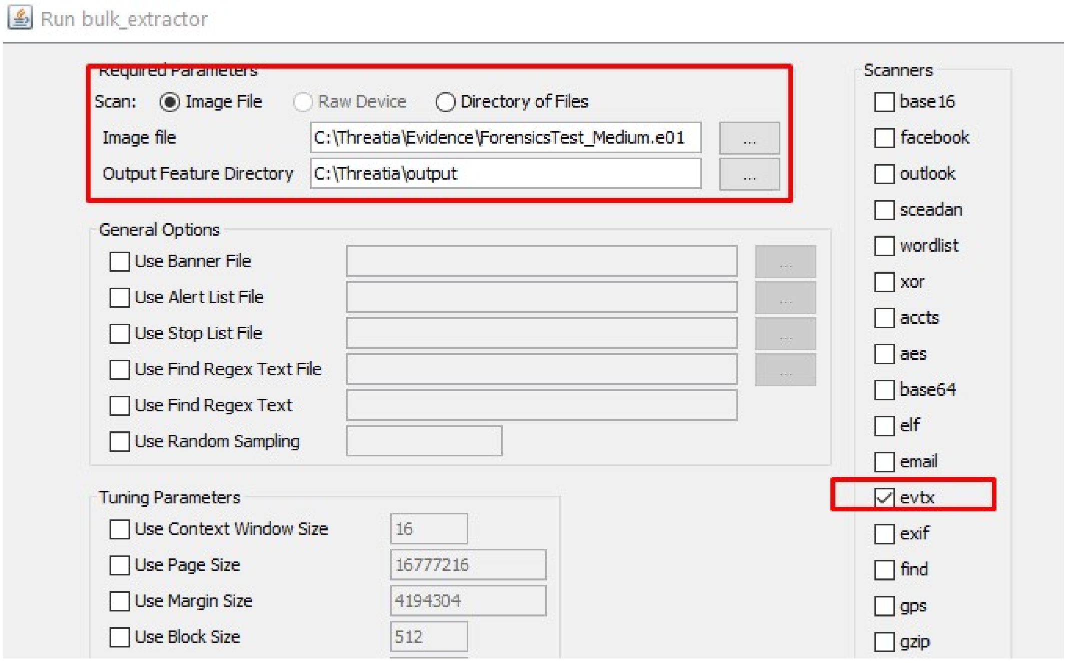


Figure 5



Run bulk_extractor

Required Parameters

Scan: ● Image File ○ Raw Device ○ Directory of Files

Image file: C:\Threatia\Evidence\ForensicsTest_Medium.e01

Output Feature Directory: C:\Threatia\output

General Options

- □ Use Banner File
- □ Use Alert List File
- □ Use Stop List File
- □ Use Find Regex Text File
- □ Use Find Regex Text
- □ Use Random Sampling

Tuning Parameters

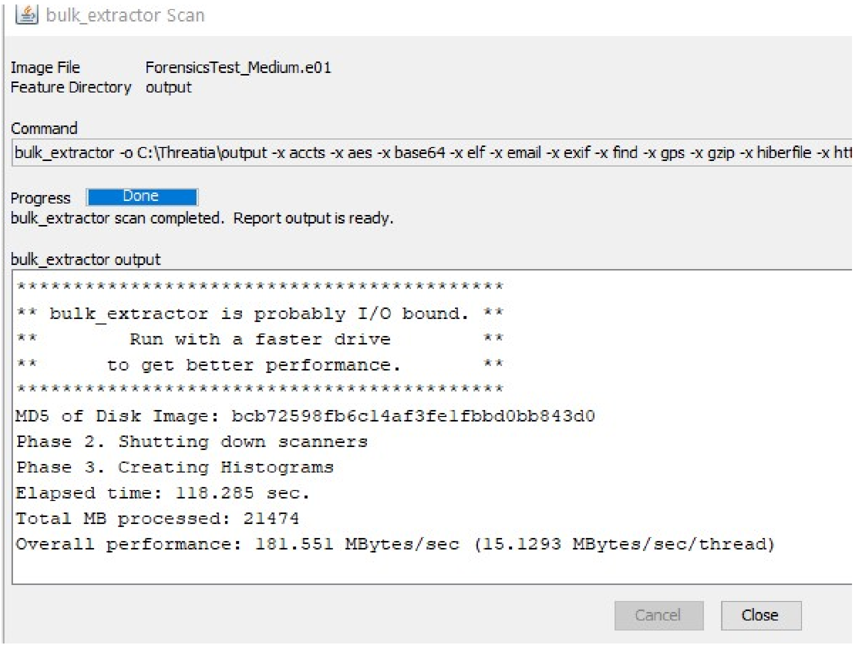
- □ Use Context Window Size: 16
- □ Use Page Size: 16777216
- □ Use Margin Size: 4194304
- □ Use Block Size: 512

Scanners

- □ base16
- □ facebook
- □ outlook
- □ sceadan
- □ wordlist
- □ xor
- □ accts
- □ aes
- □ base64
- □ elf
- □ email
- ☑ evtx
- □ exif
- □ find
- □ gps
- □ gzip

Figure 6

Finaly player will execute Bulk Extractor, clicking on “Submit Run”. Once Bulk extractor has finished, player could find results on the previous selected output:



bulk_extractor Scan

Image File: ForensicsTest_Medium.e01

Feature Directory: output

Command

```
bulk_extractor -o C:\Threatia\output -x accts -x aes -x base64 -x elf -x email -x exif -x find -x gps -x gzip -x hiberfile -x htt
```

Progress: Done

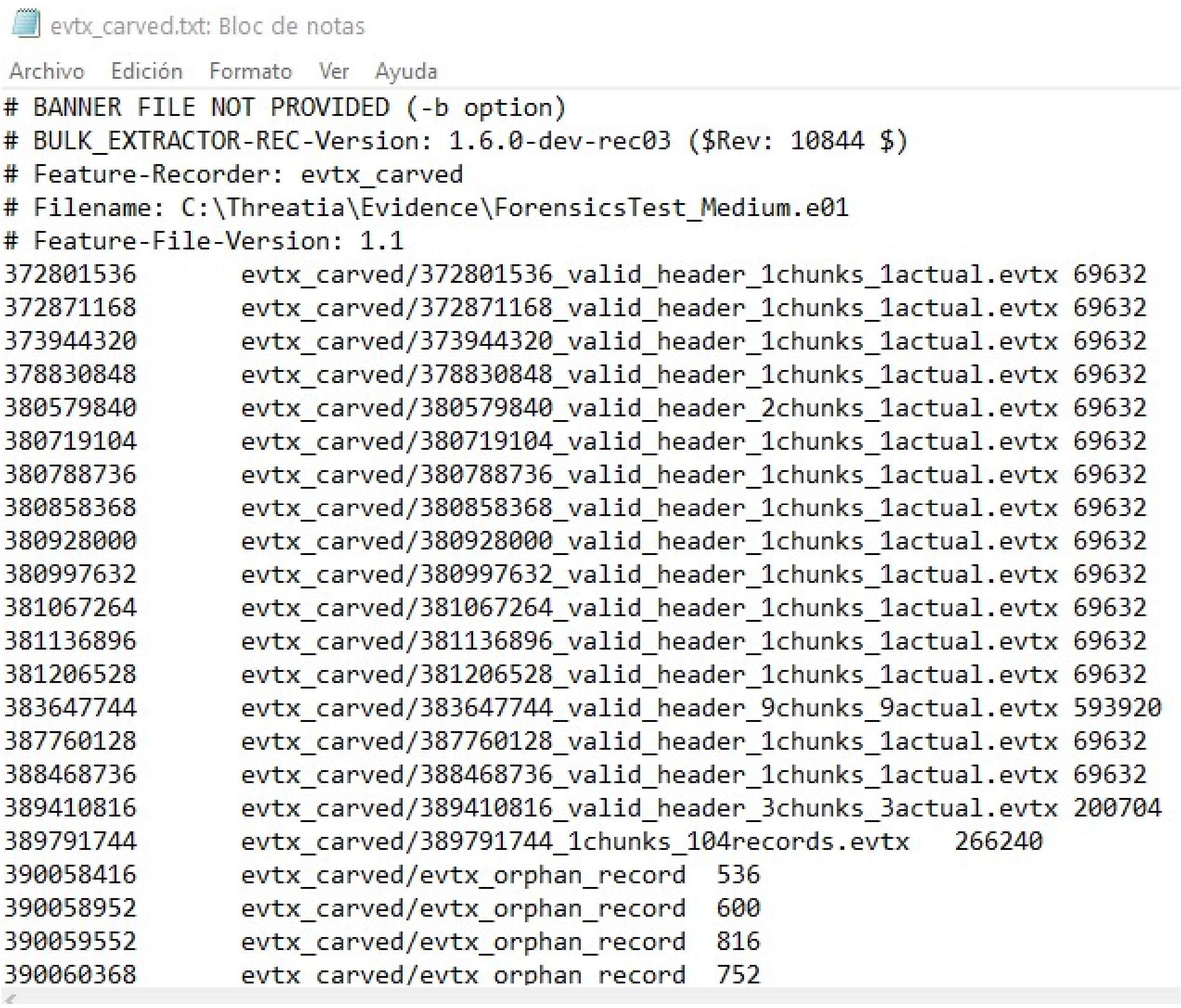
bulk_extractor scan completed. Report output is ready.

bulk_extractor output

```
*******************************************
** bulk_extractor is probably I/O bound. **
**        Run with a faster drive        **
**      to get better performance.       **
*******************************************
MD5 of Disk Image: bcb72598fb6c14af3fe1fbbd0bb843d0
Phase 2. Shutting down scanners
Phase 3. Creating Histograms
Elapsed time: 118.285 sec.
Total MB processed: 21474
Overall performance: 181.551 MBytes/sec (15.1293 MBytes/sec/thread)
```

Figure 7

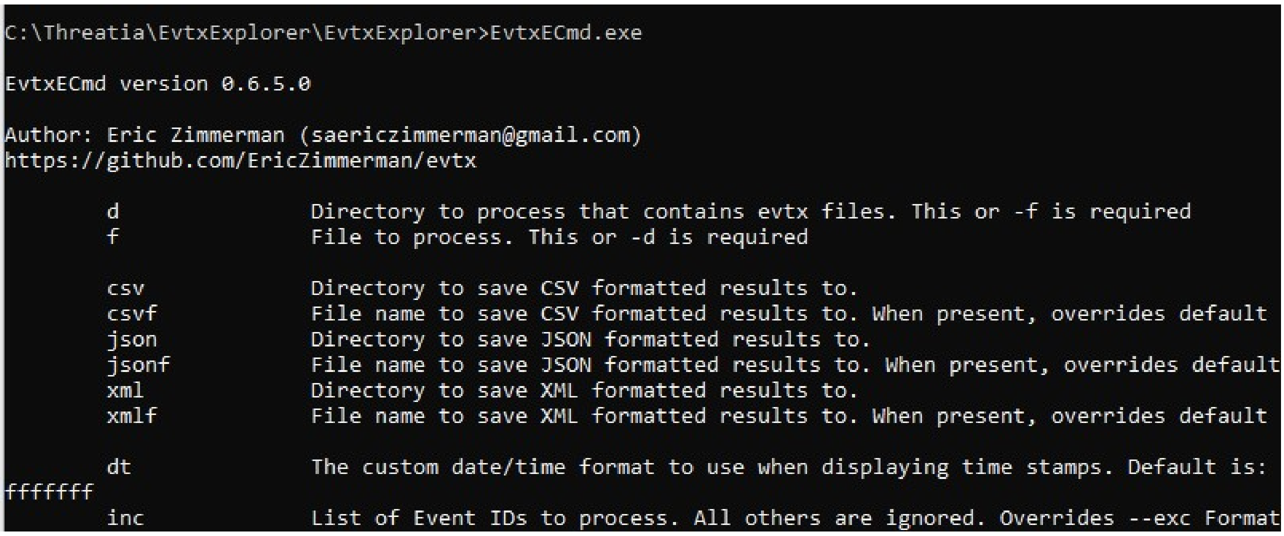
If player tries to open each of recovered files, they won´t be able select the proper event log file (no name to identify security event logs):



```
# BANNER FILE NOT PROVIDED (-b option)
# BULK_EXTRACTOR-REC-Version: 1.6.0-dev-rec03 ($Rev: 10844 $)
# Feature-Recorder: evtx_carved
# Filename: C:\Threatia\Evidence\ForensicsTest_Medium.e01
# Feature-File-Version: 1.1
372801536       evtx_carved/372801536_valid_header_1chunks_1actual.evtx 69632
372871168       evtx_carved/372871168_valid_header_1chunks_1actual.evtx 69632
373944320       evtx_carved/373944320_valid_header_1chunks_1actual.evtx 69632
378830848       evtx_carved/378830848_valid_header_1chunks_1actual.evtx 69632
380579840       evtx_carved/380579840_valid_header_2chunks_1actual.evtx 69632
380719104       evtx_carved/380719104_valid_header_1chunks_1actual.evtx 69632
380788736       evtx_carved/380788736_valid_header_1chunks_1actual.evtx 69632
380858368       evtx_carved/380858368_valid_header_1chunks_1actual.evtx 69632
380928000       evtx_carved/380928000_valid_header_1chunks_1actual.evtx 69632
380997632       evtx_carved/380997632_valid_header_1chunks_1actual.evtx 69632
381067264       evtx_carved/381067264_valid_header_1chunks_1actual.evtx 69632
381136896       evtx_carved/381136896_valid_header_1chunks_1actual.evtx 69632
381206528       evtx_carved/381206528_valid_header_1chunks_1actual.evtx 69632
383647744       evtx_carved/383647744_valid_header_9chunks_9actual.evtx 593920
387760128       evtx_carved/387760128_valid_header_1chunks_1actual.evtx 69632
388468736       evtx_carved/388468736_valid_header_1chunks_1actual.evtx 69632
389410816       evtx_carved/389410816_valid_header_3chunks_3actual.evtx 200704
389791744       evtx_carved/389791744_1chunks_104records.evtx   266240
390058416       evtx_carved/evtx_orphan_record  536
390058952       evtx_carved/evtx_orphan_record  600
390059552       evtx_carved/evtx_orphan_record  816
390060368       evtx_carved/evtx_orphan_record  752
```

Figure 8

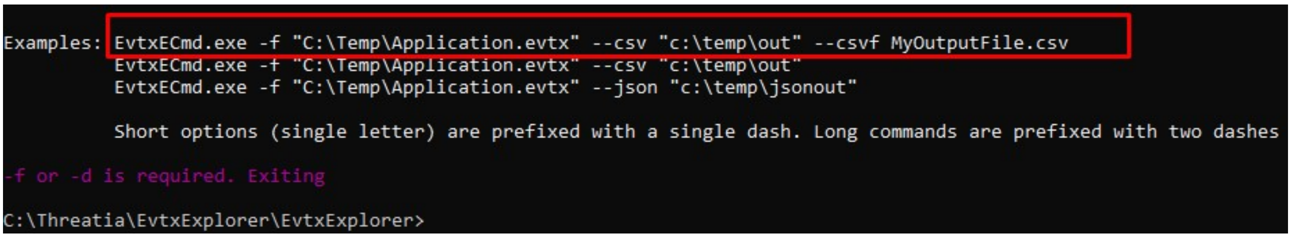
It would be better to use any other tool which allow to analyze the whole package of recevered evtx: Evtxcmd:



```
C:\Threatia\EvtxExplorer\EvtxExplorer>EvtxECmd.exe

EvtxECmd version 0.6.5.0

Author: Eric Zimmerman (saericzimmerman@gmail.com)
https://github.com/EricZimmerman/evtx

        d               Directory to process that contains evtx files. This or -f is required
        f               File to process. This or -d is required

        csv             Directory to save CSV formatted results to.
        csvf            File name to save CSV formatted results to. When present, overrides default
        json            Directory to save JSON formatted results to.
        jsonf           File name to save JSON formatted results to. When present, overrides default
        xml             Directory to save XML formatted results to.
        xmlf            File name to save XML formatted results to. When present, overrides default

        dt              The custom date/time format to use when displaying time stamps. Default is:
fffffff
        inc             List of Event IDs to process. All others are ignored. Overrides --exc Format
```

Figure 9

To launch this analyzer it´s essential how it works:

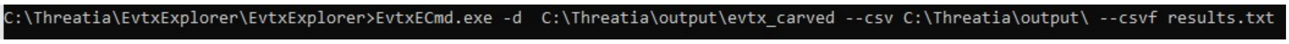


```
Examples: EvtxECmd.exe -f "C:\Temp\Application.evtx" --csv "c:\temp\out" --csvf MyOutputFile.csv
          EvtxECmd.exe -f "C:\Temp\Application.evtx" --csv "c:\temp\out"
          EvtxECmd.exe -f "C:\Temp\Application.evtx" --json "c:\temp\jsonout"

          Short options (single letter) are prefixed with a single dash. Long commands are prefixed with two dashes

-f or -d is required. Exiting

C:\Threatia\EvtxExplorer\EvtxExplorer>
```

Figure 10

Player must indicate where the output folder is that contains evtx recovered and the output results.

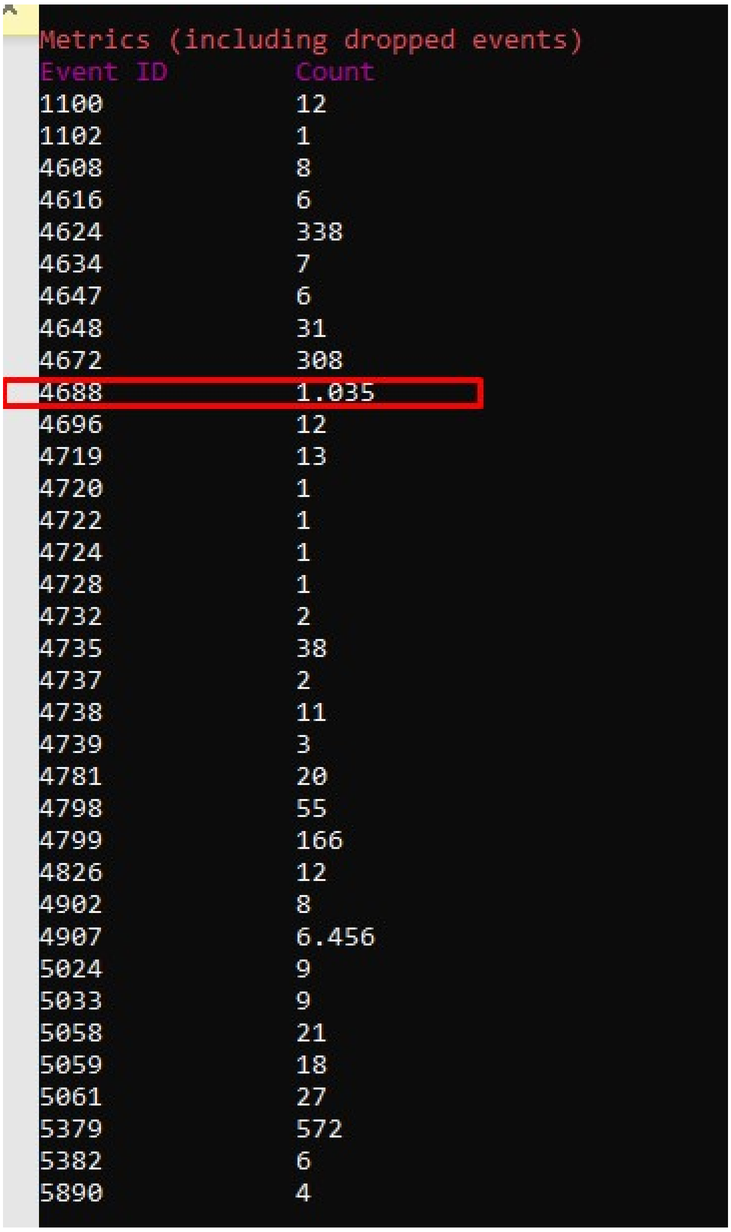
Evtxecmd.exe -f (path to evtx recovered) –csv (ouput path) –csvf (output file)



```
C:\Threatia\EvtxExplorer\EvtxExplorer>EvtxECmd.exe -d  C:\Threatia\output\evtx_carved --csv C:\Threatia\output\ --csvf results.txt
```

Figure 11

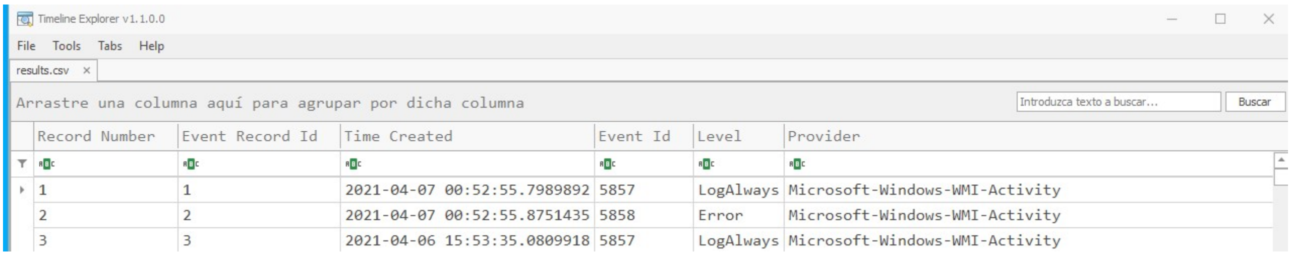
Once Evtxcmd finishes, player will be able to identify a summary of event IDs recovered:



```
Metrics (including dropped events)
Event ID            Count
1100                12
1102                1
4608                8
4616                6
4624                338
4634                7
4647                6
4648                31
4672                308
4688                1.035
4696                12
4719                13
4720                1
4722                1
4724                1
4728                1
4732                2
4735                38
4737                2
4738                11
4739                3
4781                20
4798                55
4799                166
4826                12
4902                8
4907                6.456
5024                9
5033                9
5058                21
5059                18
5061                27
5379                572
5382                6
5890                4
```

Figure 12

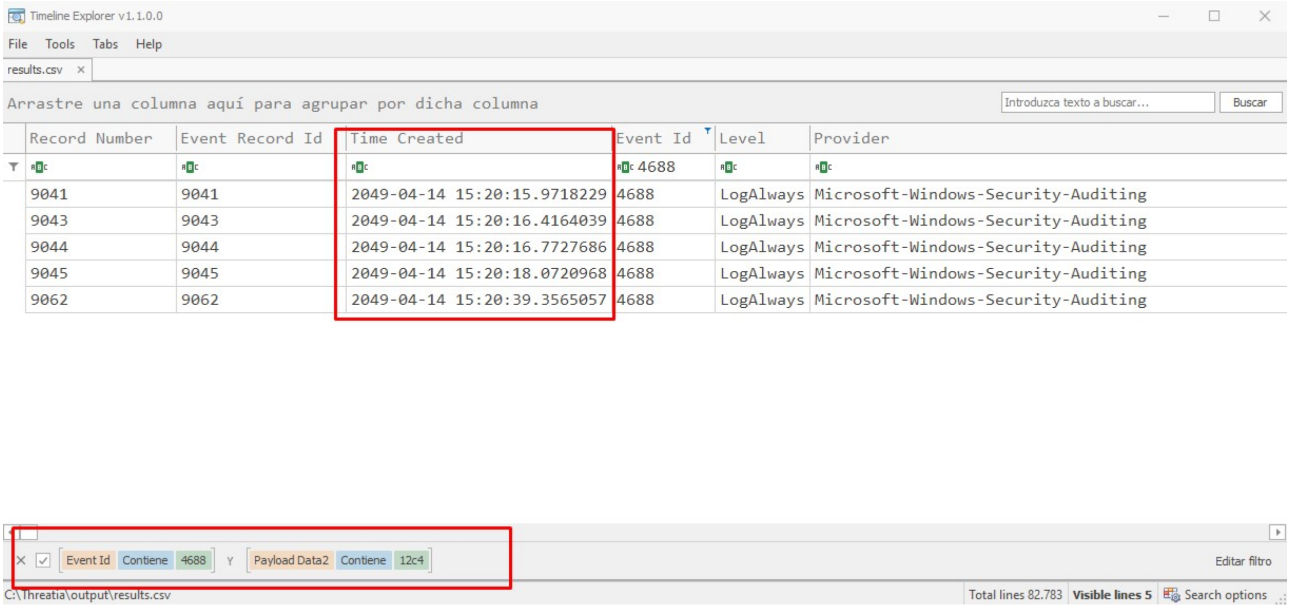
The key is to analyse 4688 events, so it´s necessary to open CSV file using Timeline Explorer:



Arrastre una columna aquí para agrupar por dicha columna

Record Number	Event Record Id	Time Created	Event Id	Level	Provider
1	1	2021-04-07 00:52:55.7989892	5857	LogAlways	Microsoft-Windows-WMI-Activity
2	2	2021-04-07 00:52:55.8751435	5858	Error	Microsoft-Windows-WMI-Activity
3	3	2021-04-06 15:53:35.0809918	5857	LogAlways	Microsoft-Windows-WMI-Activity

Player has noticed that inside the description their challenge, player has to focus on: 2049, a tool to hide Phaldra´s footprints and a program ID 4804. So, keeping in to account the previous information, it must transform 4804 into HEX: 12C4



Arrastre una columna aquí para agrupar por dicha columna

Record Number	Event Record Id	Time Created	Event Id	Level	Provider
			4688		
9041	9041	2049-04-14 15:20:15.9718229	4688	LogAlways	Microsoft-Windows-Security-Auditing
9043	9043	2049-04-14 15:20:16.4164039	4688	LogAlways	Microsoft-Windows-Security-Auditing
9044	9044	2049-04-14 15:20:16.7727686	4688	LogAlways	Microsoft-Windows-Security-Auditing
9045	9045	2049-04-14 15:20:18.0720968	4688	LogAlways	Microsoft-Windows-Security-Auditing
9062	9062	2049-04-14 15:20:39.3565057	4688	LogAlways	Microsoft-Windows-Security-Auditing

× ☑ Event Id Contiene 4688 Y Payload Data2 Contiene 12c4

Editar filtro

C:\Threatia\output\results.csv Total lines 82.783 **Visible lines 5** Search options

Figure 13

Player wil be able to export filtered results in order to get the catch the threat:

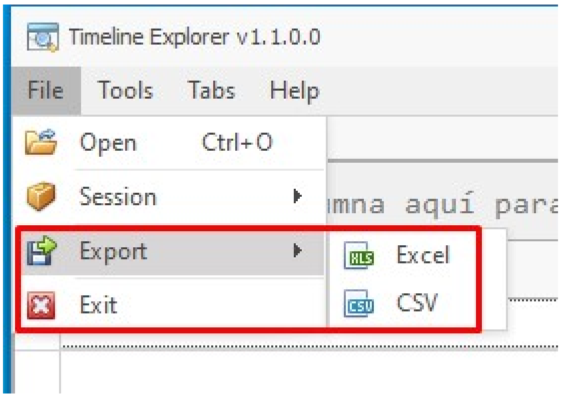
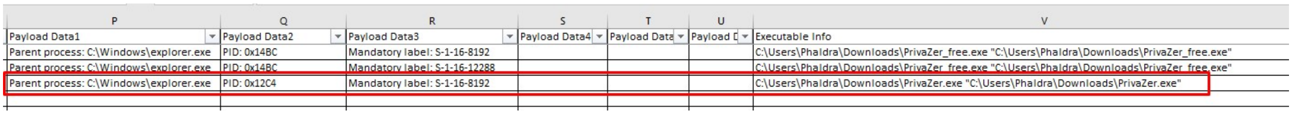


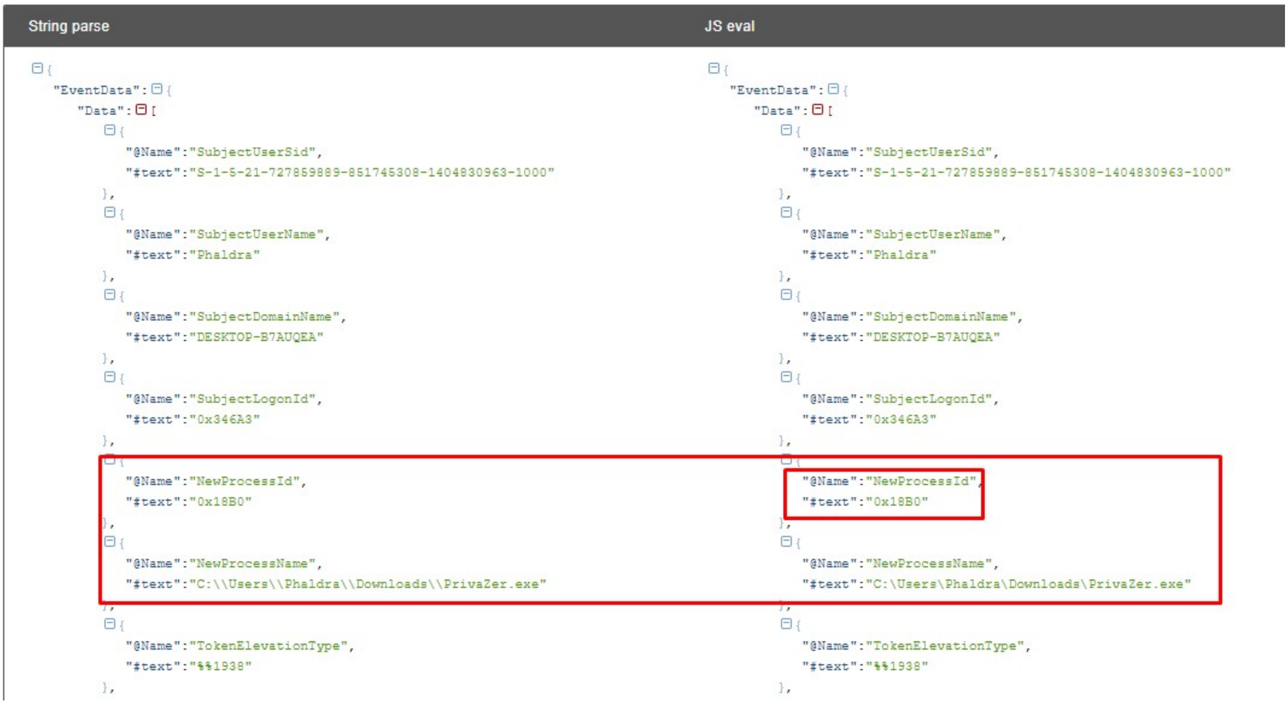
Figure 14

Once exported, check exectuble info, opening the XLSX file:



P	Q	R	S	T	U	V
Payload Data1	Payload Data2	Payload Data3	Payload Data4	Payload Data	Payload D	Executable Info
Parent process: C:\Windows\explorer.exe	PID: 0x14BC	Mandatory label: S-1-16-8192				C:\Users\Phaldra\Downloads\PrivaZer_free.exe "C:\Users\Phaldra\Downloads\PrivaZer_free.exe"
Parent process: C:\Windows\explorer.exe	PID: 0x14BC	Mandatory label: S-1-16-12288				C:\Users\Phaldra\Downloads\PrivaZer_free.exe "C:\Users\Phaldra\Downloads\PrivaZer_free.exe"
Parent process: C:\Windows\explorer.exe	PID: 0x12C4	Mandatory label: S-1-16-8192				C:\Users\Phaldra\Downloads\PrivaZer.exe "C:\Users\Phaldra\Downloads\PrivaZer.exe"

And finally, check Payload fata of Privazer (json format):



String parse

```
{
   "EventData":{
      "Data":[
         {
            "@Name":"SubjectUserSid",
            "#text":"S-1-5-21-727859889-851745308-1404830963-1000"
         },
         {
            "@Name":"SubjectUserName",
            "#text":"Phaldra"
         },
         {
            "@Name":"SubjectDomainName",
            "#text":"DESKTOP-B7AUQEA"
         },
         {
            "@Name":"SubjectLogonId",
            "#text":"0x346A3"
         },
         {
            "@Name":"NewProcessId",
            "#text":"0x18B0"
         },
         {
            "@Name":"NewProcessName",
            "#text":"C:\\Users\\Phaldra\\Downloads\\PrivaZer.exe"
         },
         {
            "@Name":"TokenElevationType",
            "#text":"%%1938"
         },
```

JS eval

```
{
   "EventData":{
      "Data":[
         {
            "@Name":"SubjectUserSid",
            "#text":"S-1-5-21-727859889-851745308-1404830963-1000"
         },
         {
            "@Name":"SubjectUserName",
            "#text":"Phaldra"
         },
         {
            "@Name":"SubjectDomainName",
            "#text":"DESKTOP-B7AUQEA"
         },
         {
            "@Name":"SubjectLogonId",
            "#text":"0x346A3"
         },
         {
            "@Name":"NewProcessId",
            "#text":"0x18B0"
         },
         {
            "@Name":"NewProcessName",
            "#text":"C:\Users\Phaldra\Downloads\PrivaZer.exe"
         },
         {
            "@Name":"TokenElevationType",
            "#text":"%%1938"
         },
```

Figure 15

This is the PID: 0x18B0

Flag Information

flag{Ransom: 0x18B0}