Complete Windows Forensics

Project Overview

This project is a **complete toolkit** and guide for **investigating Windows computers**. It helps investigators **collect and analyze memory, disk, and system artifacts** to understand **user activity and system behavior**. From **capturing RAM and full disk images** to analyzing **registry entries, system logs, and artifacts like MFT, USN Journal, and Prefetch**, the toolkit provides a full picture of what happened on a computer. It also includes **automated collection, evidence verification**, and **timeline creation** to make investigations faster, safer, and more organized.

Current Session: VM Acquisition & Triage Analysis :

Session Scope

In this practical session, we will cover:

- **Memory and disk acquisition** procedures within virtual environments
- **Alternative tools available** for physical machine acquisition
- **Mounting disk images** using Arsenal Image Mounter for analysis
- **Automated triage collection** using Kape for rapid evidence gathering
- **Registry examination** with Registry Explorer and bulk parsing techniques

Session Objectives - Key Questions to Answer:

- **Active accounts during the attack timeframe ?**
- **Which account(s) were created ?**
- **Which accounts are Administrator group members ?**
- **Which users have profiles on the system ?**

Note: This session covers the basic acquisition and triage phase. Advanced artifact analysis will be addressed in later sessions.

Tools Required :

Tool	Purpose
VirtualBox & VBoxManage	Virtual machine management and evidence acquisition
FTK Imager	Disk imaging and forensic analysis
Arsenal Image Mounter	Mounting disk images for analysis
Kape	Automated forensic triage and artifact collection
Registry Explorer	Windows registry analysis and exploration
RegRipper	Automated registry parsing and data extraction
Notepad++	Log analysis and text file examination

Investigation Documentation :

Field	Details
Case Number	CF-2025-001
Investigator	NIZAR ADERBAZ
Date	2025-11-11

Target System	**Windows 10 Pro**
Investigation Type	**Complete Windows System Forensics**

1) Memory Forensics: Acquisition

Memory forensics involves **capturing and analyzing a computer's RAM (Random Access Memory)** to extract **volatile evidence** that disappears when the system is powered off. This includes **running processes**, **network connections**, **open files**, **encryption keys**, and **malware activities.**

Memory Acquisition Methods :

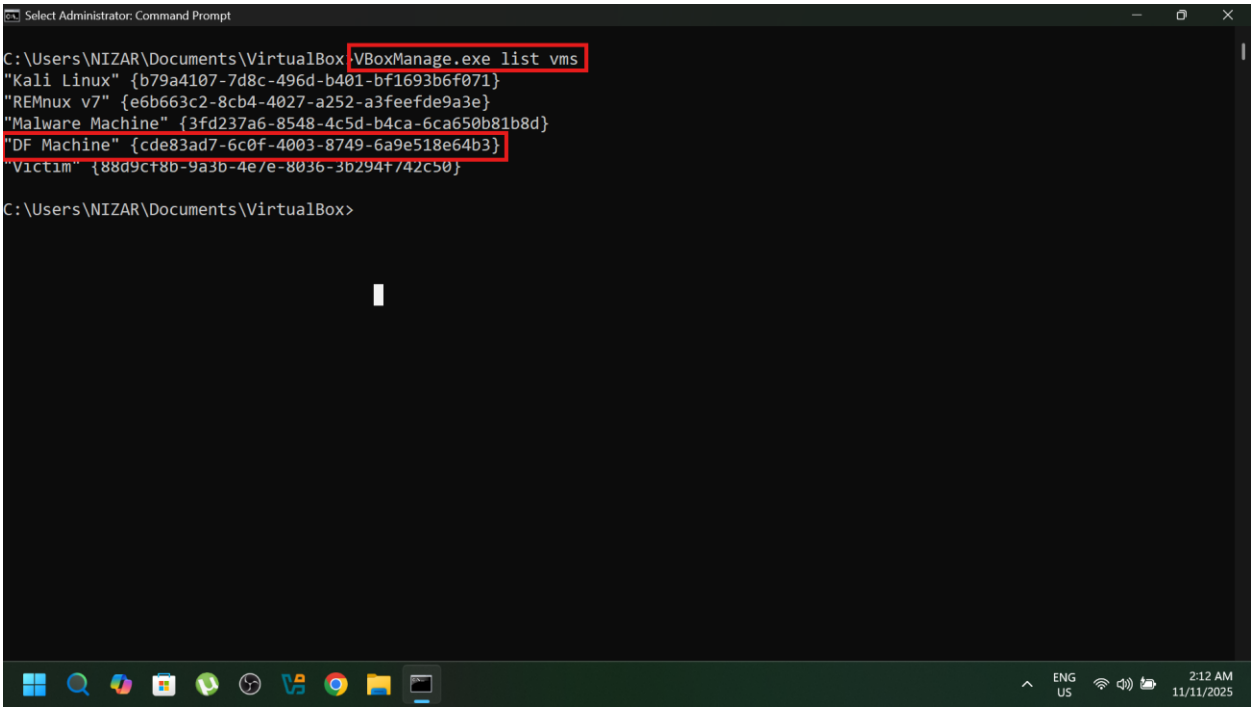
To capture a memory image, we can use various public tools like **FTK Imager** for physical machines, or if we're in a **VirtualBox environment,** we can use the built-in **VBoxManage tool.**

Step 1: Identify Virtual Machine

First, we need to identify our **VMs** and obtain **the UUID** of **the target machine**:

Command :

- ⇨ **.\VBoxManage.exe list vms**



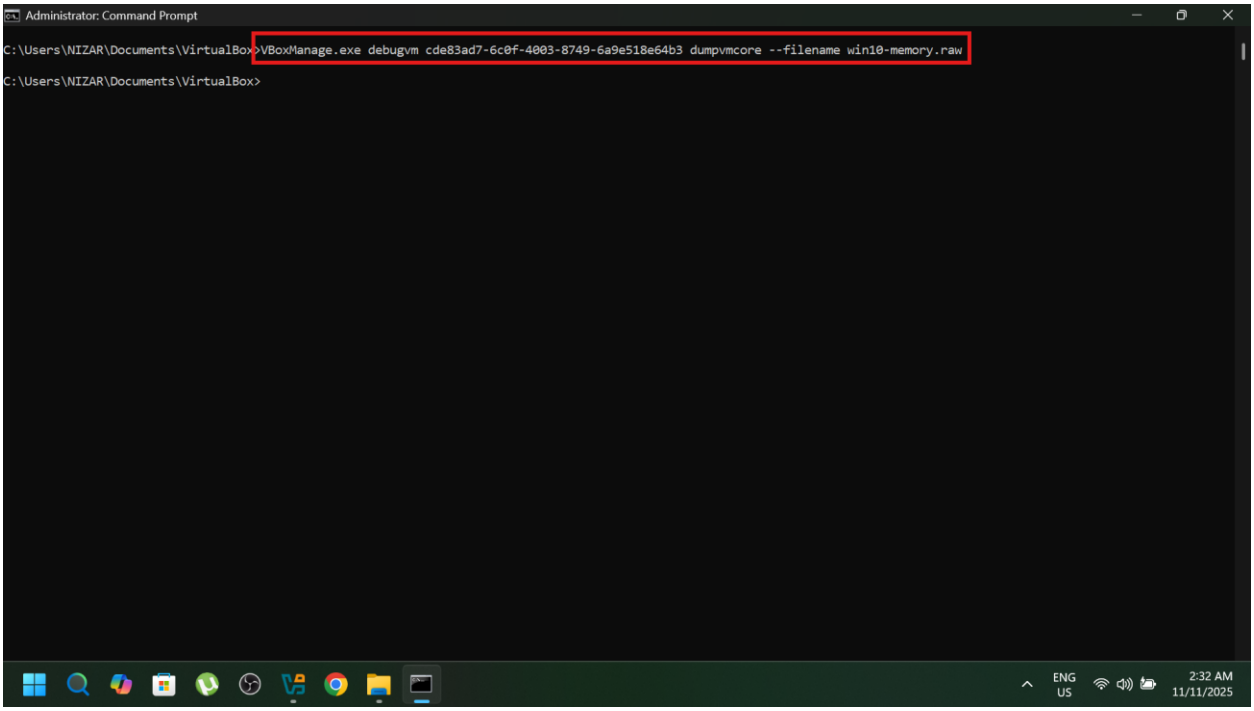
Output showing all available virtual machines with their UUIDs

Step 2: Capture Memory Image

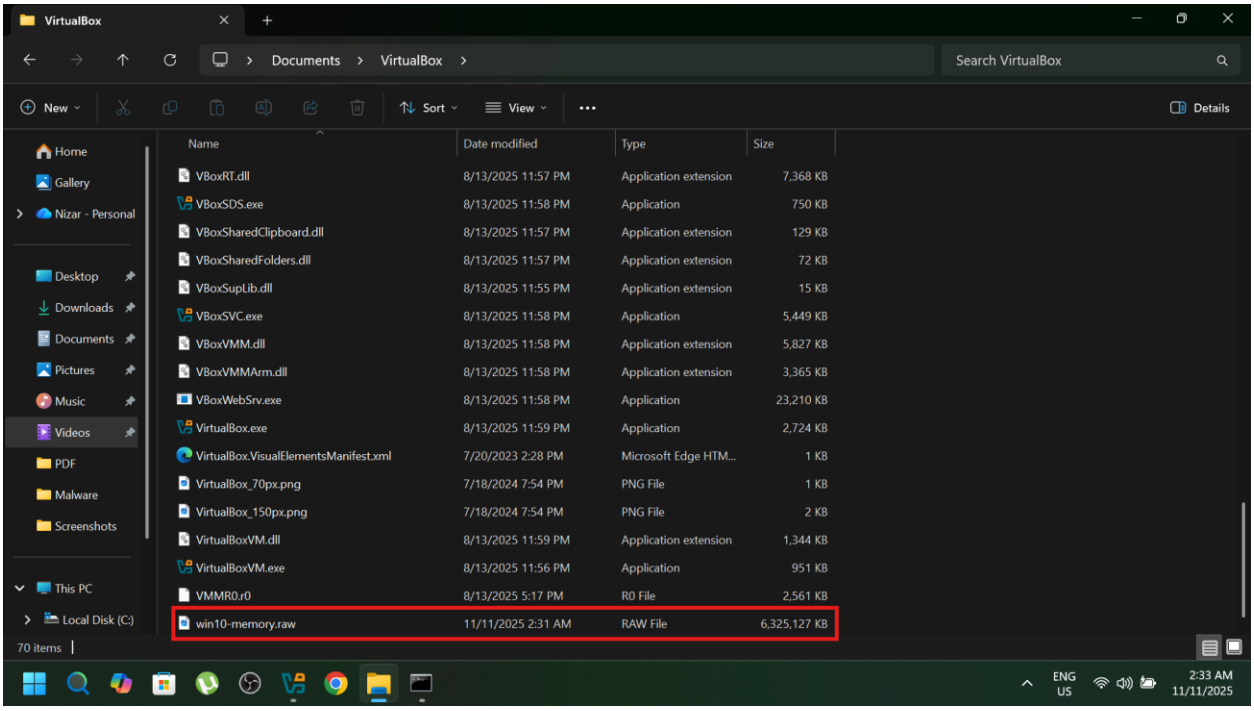
Using the **UUID** from the previous step, we capture **the memory image**:

Command :

- ⇨ **.\VBoxManage.exe debugvm cde83ad7-6c0f-4003-8749-6a9e518e64b3 dumpvmcore --filename win10-memory.raw**



Executing VBoxManage debugvm command to capture memory dump from target virtual machine



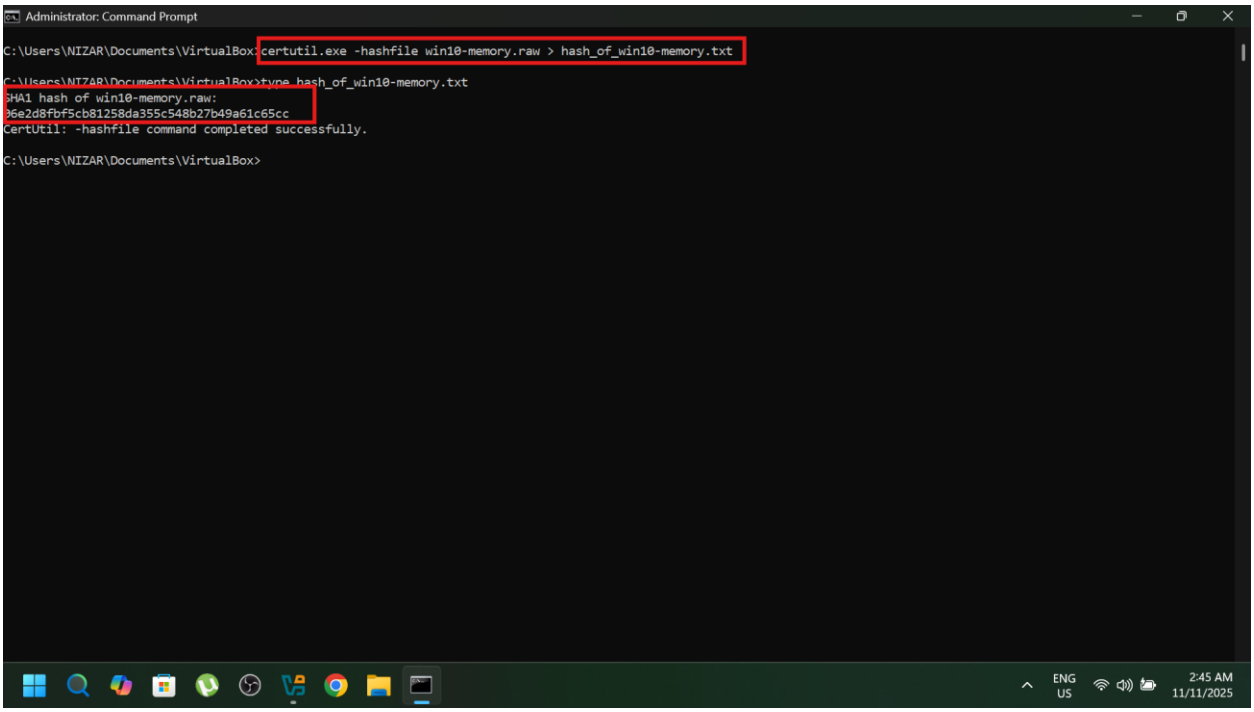
Memory dump file (win10-memory.raw) successfully created in the target folder

Step 3: Verify Integrity

After acquiring **the memory dump**, we calculate the **hash** for integrity verification:

Command :

- ⇨ **certutil.exe -hashfile win10-memory.raw > hash_of_win10-memory.txt**



SHA-1 hash verification of memory dump file using certutil, ensuring evidence integrity

2) Disk Forensics: Acquisition

Disk forensics involves **creating a forensic image** of **a computer's storage media (HDD, SSD)** to **preserve file systems**, **documents**, **applications**, and **system artifacts for analysis**. Unlike memory, **disk evidence persists after shutdown.**

Disk Acquisition Methods

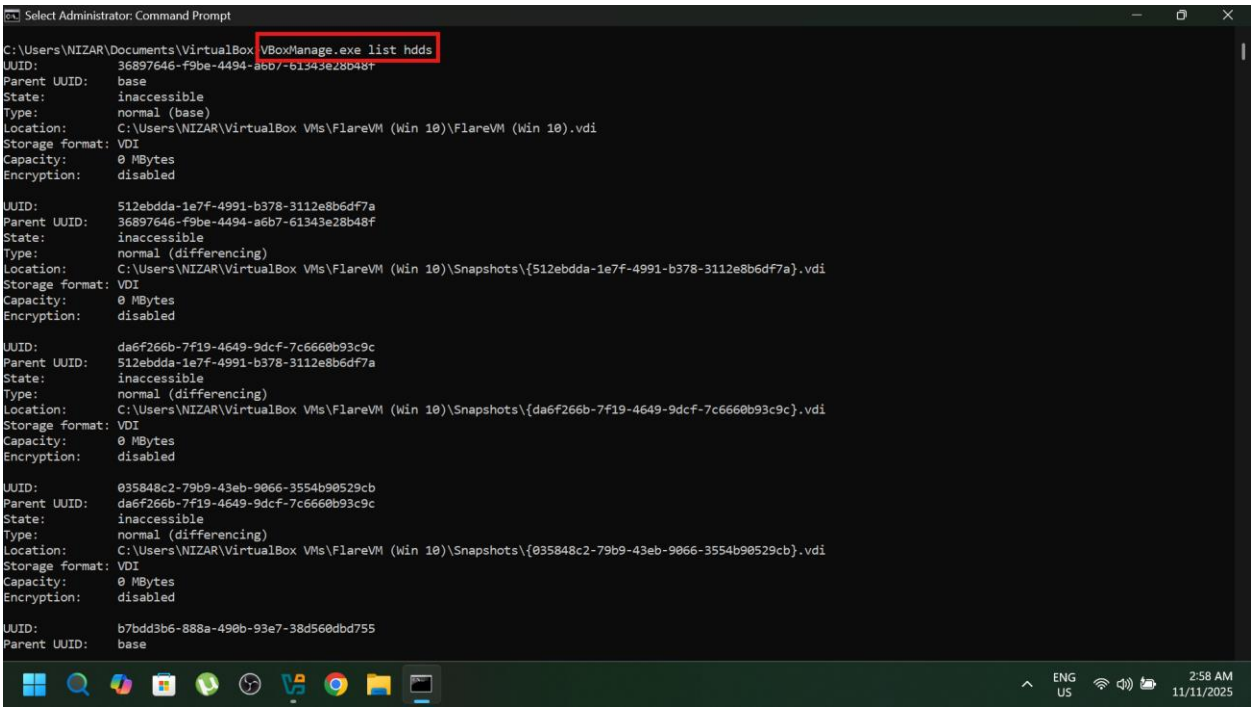
Now that memory acquisition is complete, we can safely shut down the target system and proceed with disk imaging using **VBoxManage**.

Step 1: Identify Disk Images

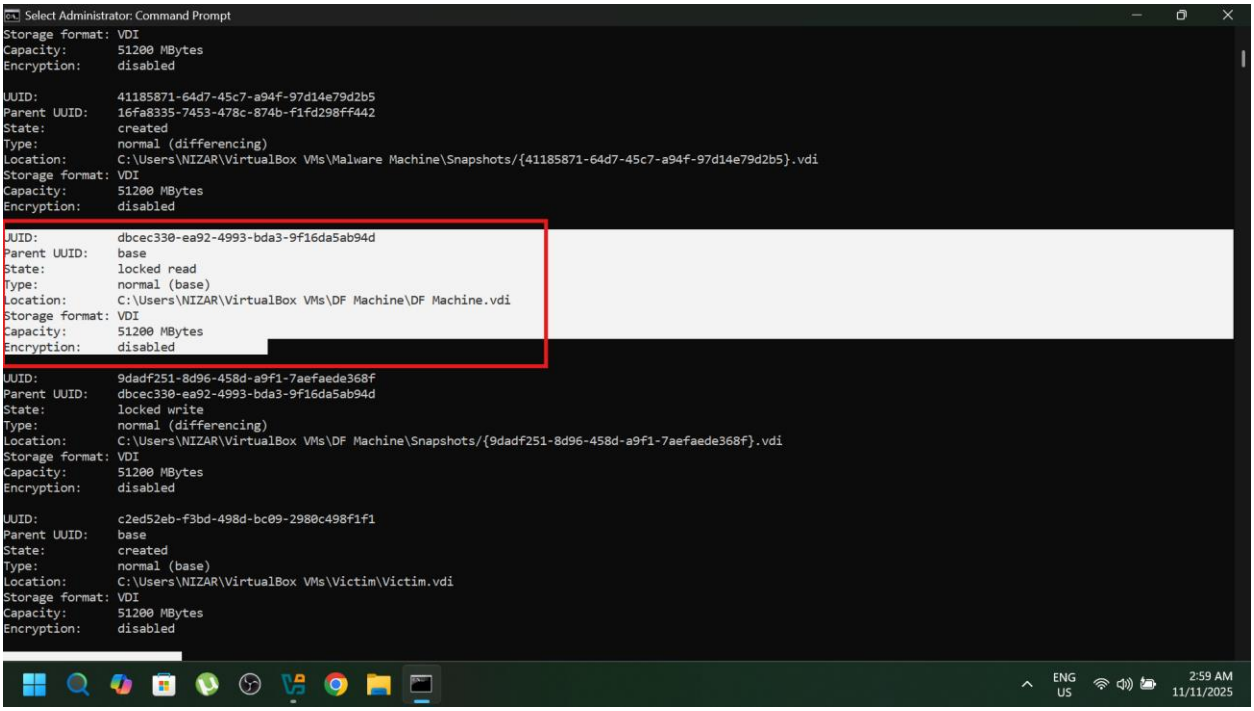
First, list all available disk images to identify **the target**:

Command :

- ⇨ **.\VBoxManage.exe list hdds**



Listing all virtual disk images with VBoxManage to identify the target disk UUID for acquisition



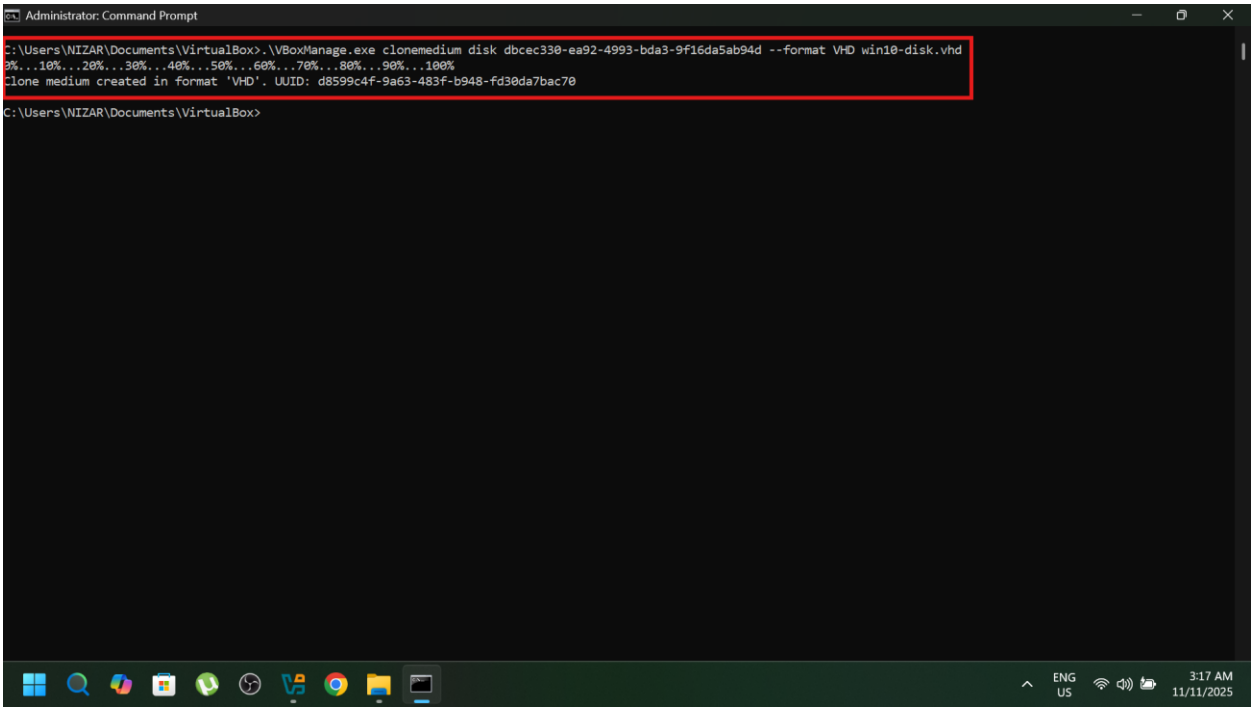
Identifying the victim's virtual disk from the list of available disk images

Step 2: Capture Disk Image

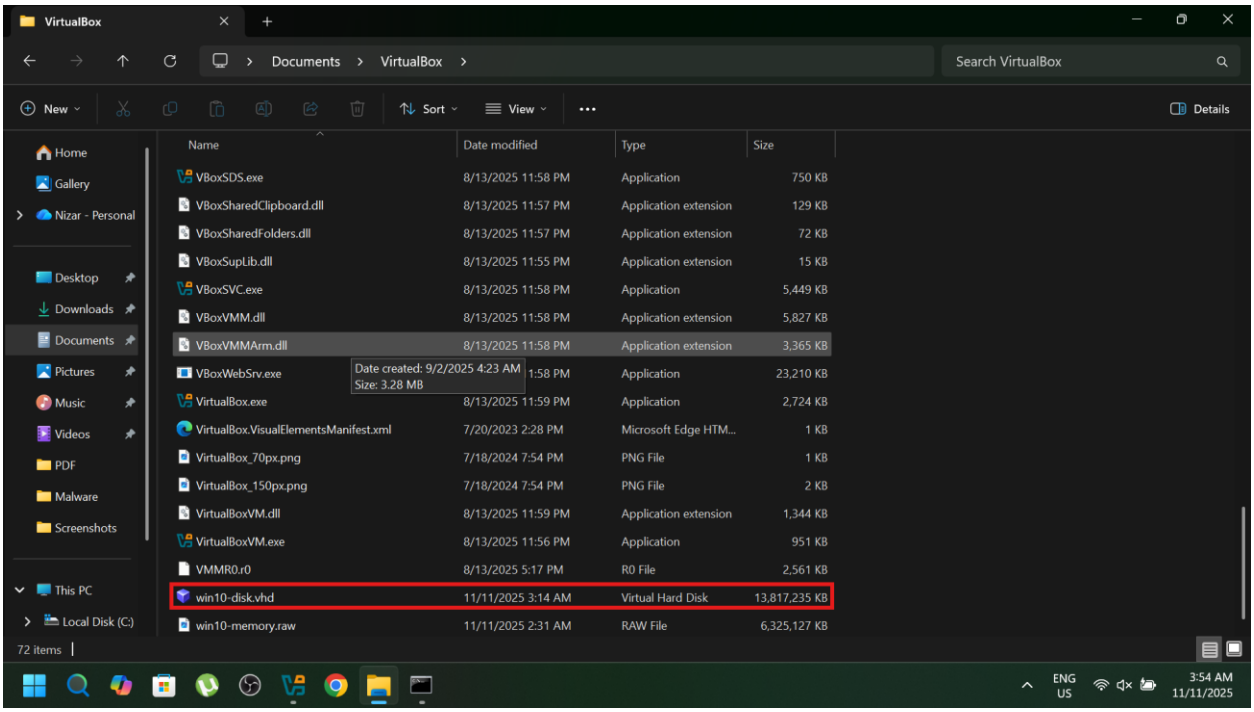
Using **the disk UUID**, create a forensic **copy of the disk**:

Command :

- ⇨ **.\VBoxManage.exe clonemedium disk dbcec330-ea92-4993-bda3-9f16da5ab94d --format VHD win10-disk.vhd**



Creating forensic image of victim's disk using VBoxManage clonemedium command



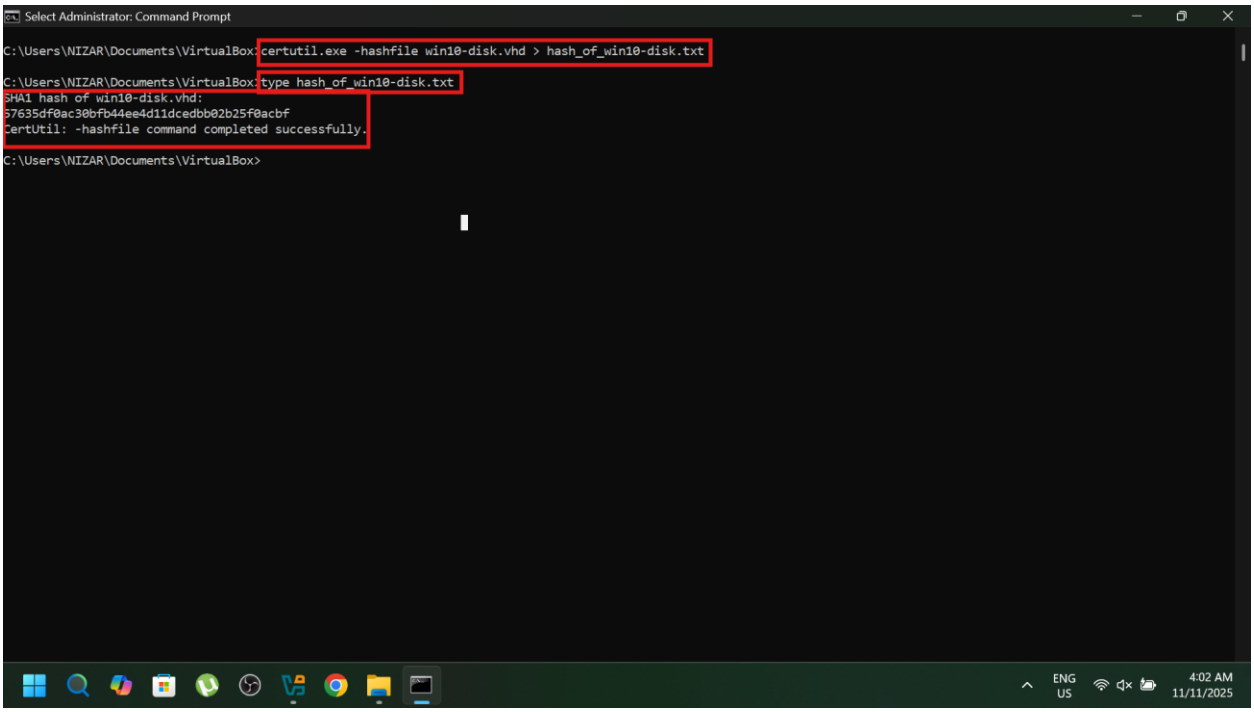
Victim's disk image (win10-disk.vhd) successfully created and visible in the evidence folder

Step 3: Verify Integrity

Generate hash verification for the disk image:

Command :

- ⇨ **certutil.exe -hashfile win10-disk.vhd > hash_of_win10-disk.txt**



Verifying integrity of victim's disk image using SHA-1 hash calculation

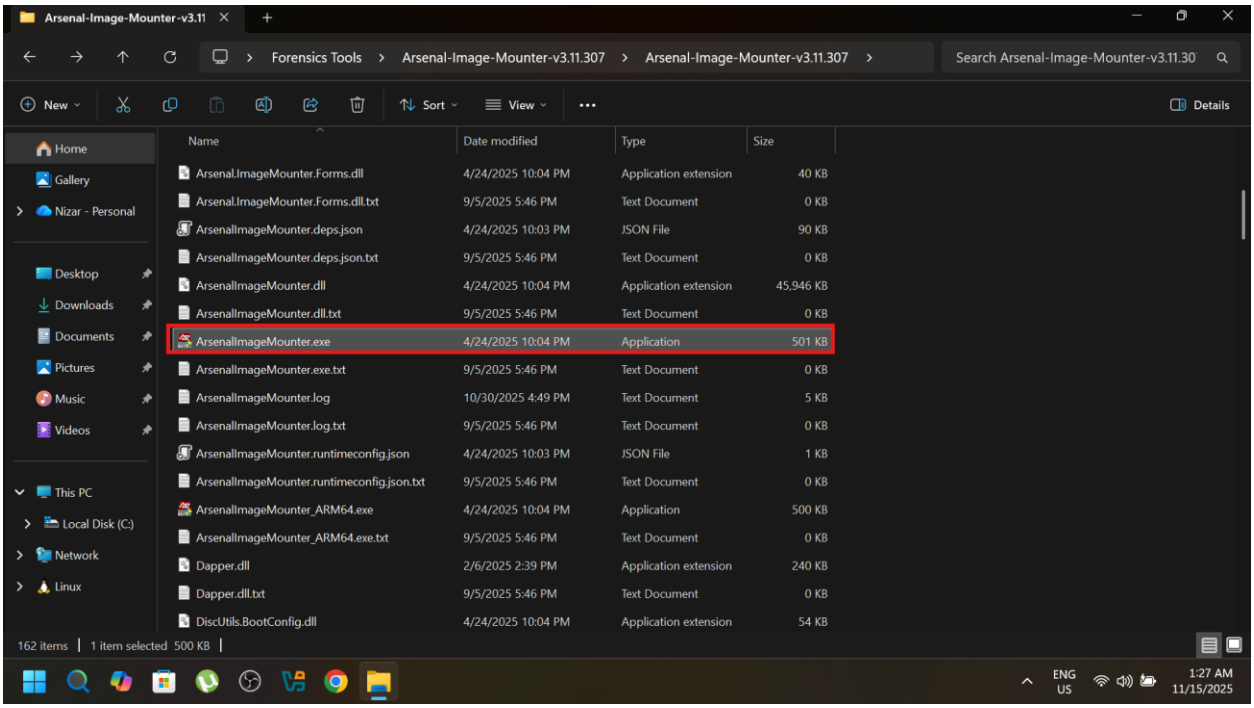
3) Disk Forensics: Mounting & Analysis

Disk mounting allows us to access and interact with forensic disk images as if they were physical drives, enabling file system exploration and analysis while maintaining evidence integrity.

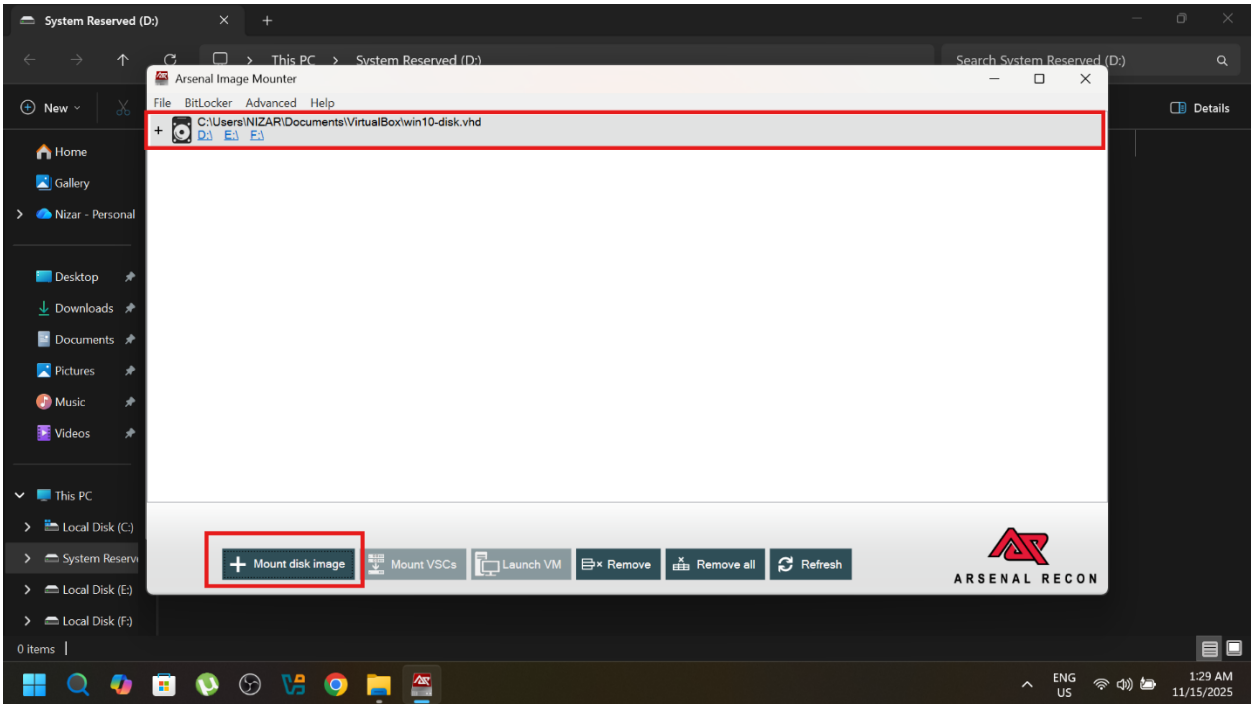
Now that we have our disk image, we'll use **Arsenal Image Mounter** to mount it for analysis.

Step 1: Open Arsenal Image Mounter

- Launch **Arsenal Image Mounter** application
- Click **"Mount Image"** and select our **win10-disk.vhd** file



Launching Arsenal Image Mounter application to begin the disk mounting process



Selecting "Mount Image" in Arsenal and choosing the victim's win10-disk.vhd file for analysis

Important Mounting Configuration !!!

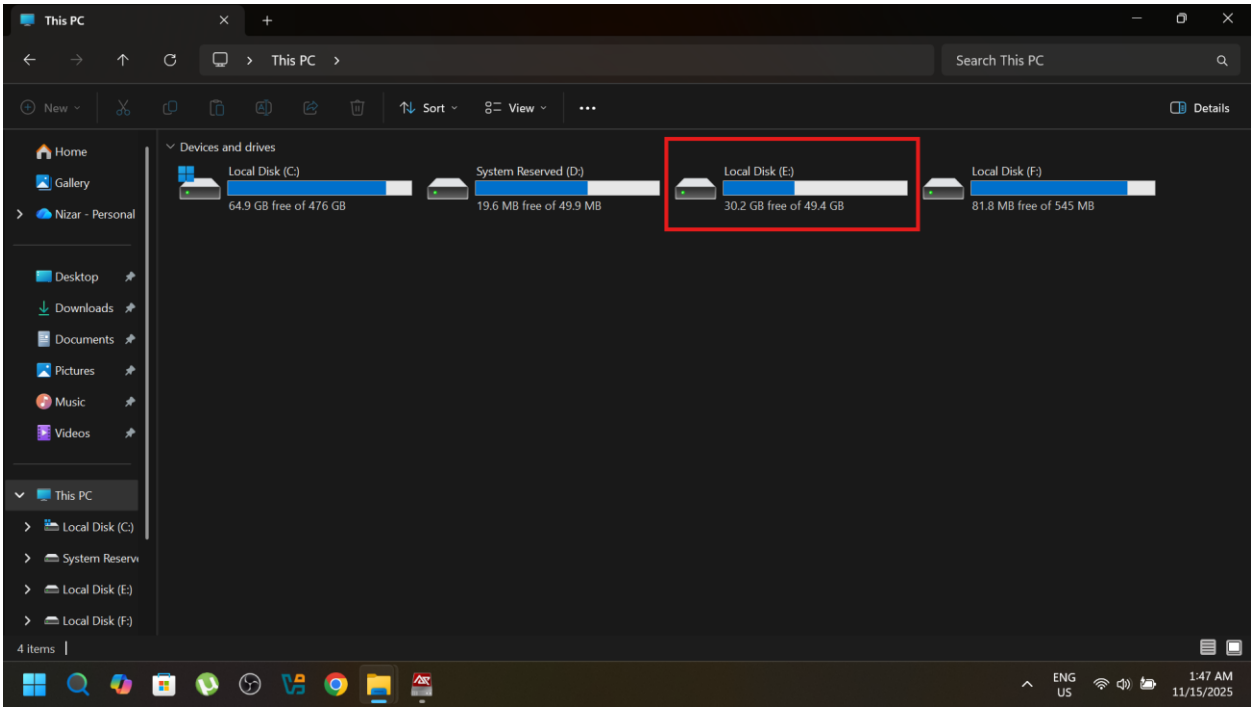
When Arsenal prompts for mounting options, ensure you select:

- **Mount on: Physical drive letter (e.g., E:)**
- **Access mode: Write temporary (creates temporary writeable copy)**
- **File system: Automatic detection**

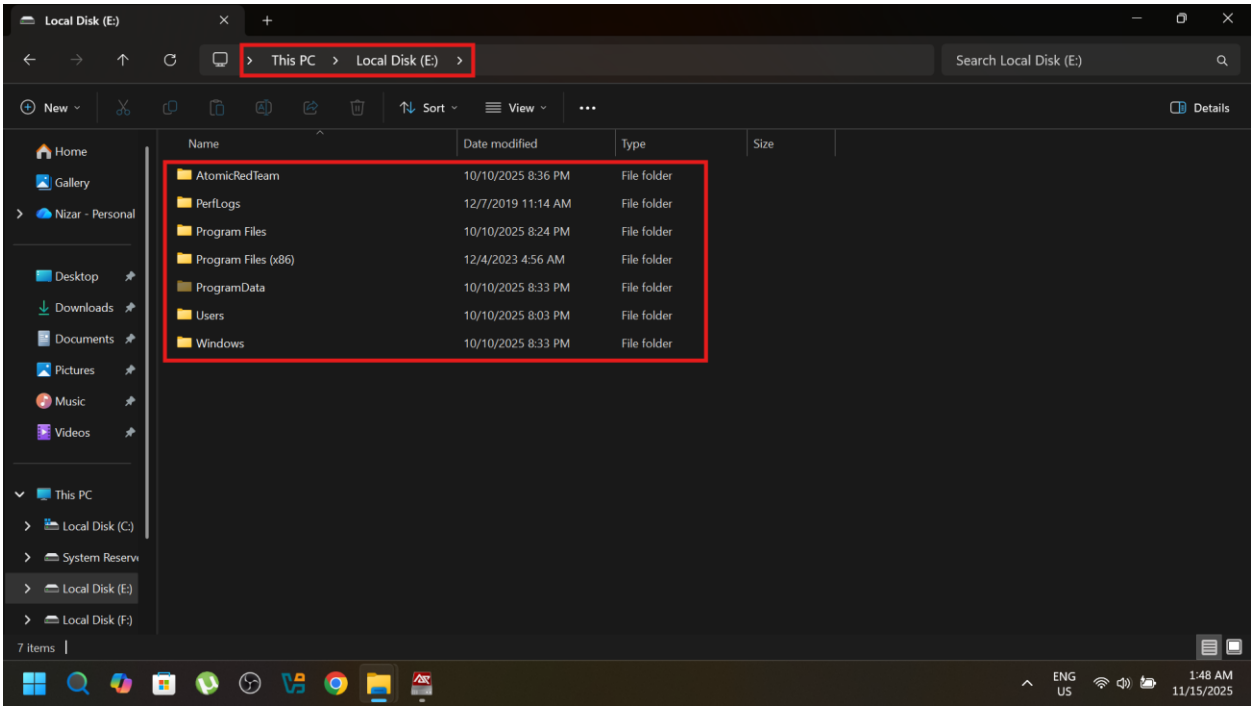
Note: The **"Write temporary"** option is crucial as it creates a temporary copy for analysis while preserving the original evidence integrity.

Step 2: Access Mounted Disk

Once mounted, **the disk will appear** as a new drive **in Windows Explorer**, but we'll use **command line** for complete access.



Victim's disk successfully mounted as drive E: in Windows Explorer - ready for command line analysis



Initial view inside the mounted victim's disk showing Windows system files and folders

Forensic Triage with Kape

Forensic triage involves **rapidly collecting and analyzing key digital artifacts** to quickly understand **system activity**, **user behavior**, and **potential security incidents**.

Now that our disk is mounted, we'll use **Kape** to automatically collect crucial forensic artifacts from the victim's system.

Step 1: Prepare Kape Command

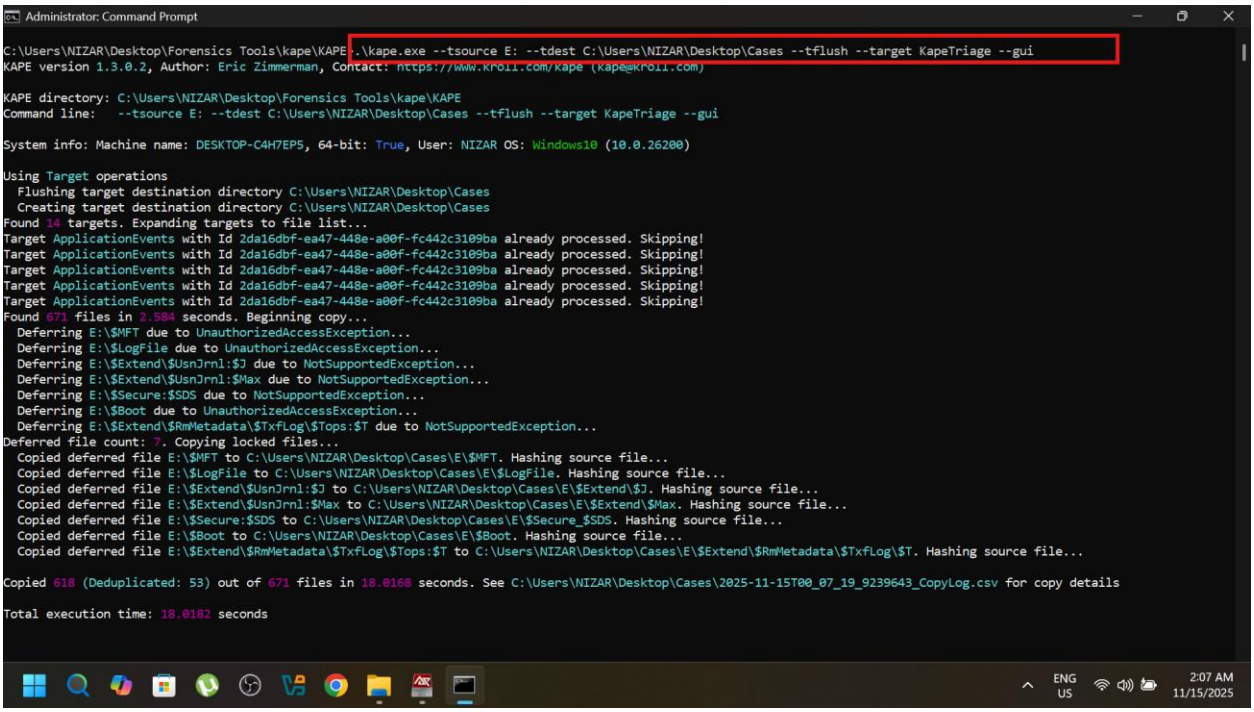
We'll use **Kape** to target our mounted **drive (E:)** and collect essential artifacts:

Command :

- **.\kape.exe --tsource E: --tdest C:\Users\NIZAR\Desktop\Cases --tflush --target KapeTriage –gui**

Command Breakdown

- **--tsource E:** : Target source (our mounted victim disk)
- **--tdest C** :\Users\NIZAR\Desktop\Cases: Destination for collected artifacts
- **--tflush** : Clear destination before collection
- **--target KapeTriage**: Predefined collection targets
- **--gui** : Show graphical interface



Kape command executing successfully, showing the triage collection process in progress

Registry Forensics Analysis :

The Windows Registry is a **hierarchical database that stores system configuration, user settings, and application data**. Registry forensics examines these artifacts **to reconstruct user activities, system changes, and potential malicious behavior**.

Now that we have our triage data, we'll analyze **the registry** using **specialized forensic tools**.

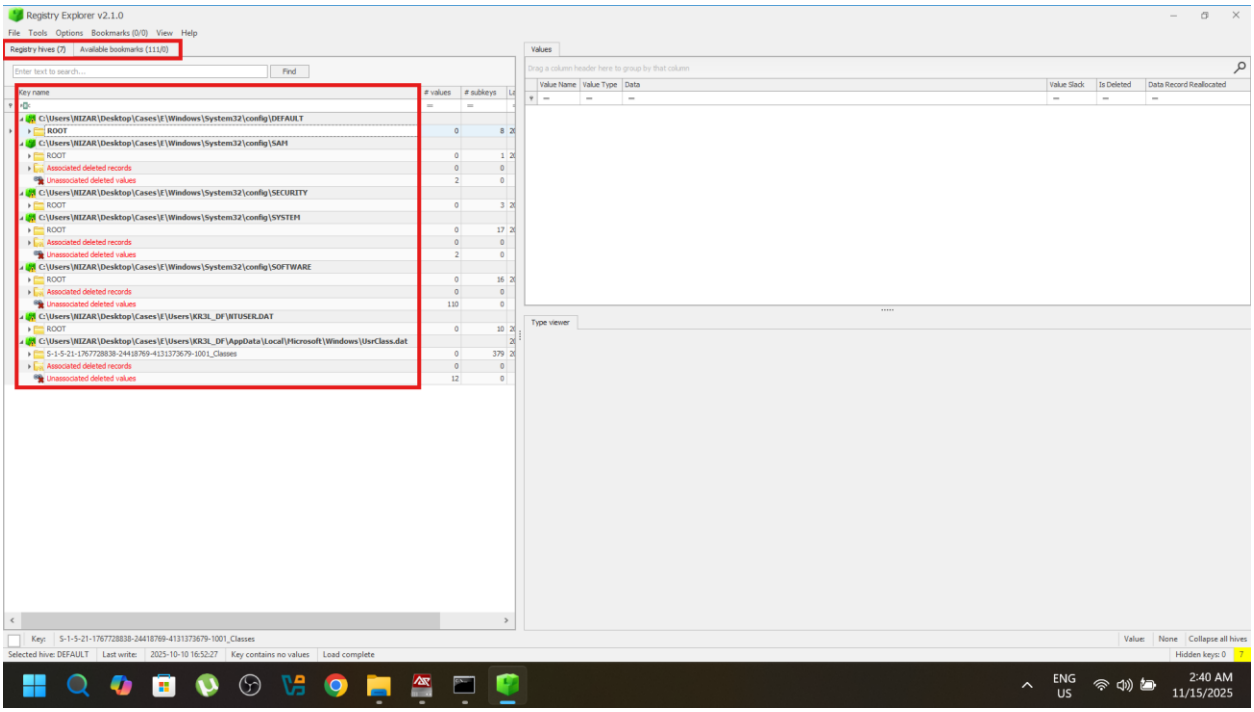
Step 1: Locate Registry Hives

Key registry hives are located at:

- **User Profiles:** E:\Users\Target\NTUSER.DAT
- **User Classes:** E:\Users\Target\AppData\Local\Microsoft\Windows\UsrClass.dat
- **System Hives:** E:\Windows\System32\config\ (SYSTEM, SOFTWARE, SAM, SECURITY)

Step 2: Analyze with Registry Explorer

We'll use **Registry Explorer** to load and examine the registry hives with **built-in bookmarks** for common forensic artifacts.



Using Registry Explorer to load victim's registry hives with built-in forensic bookmarks for artifact analysis

Step 3: Bulk Parsing with RegRipper

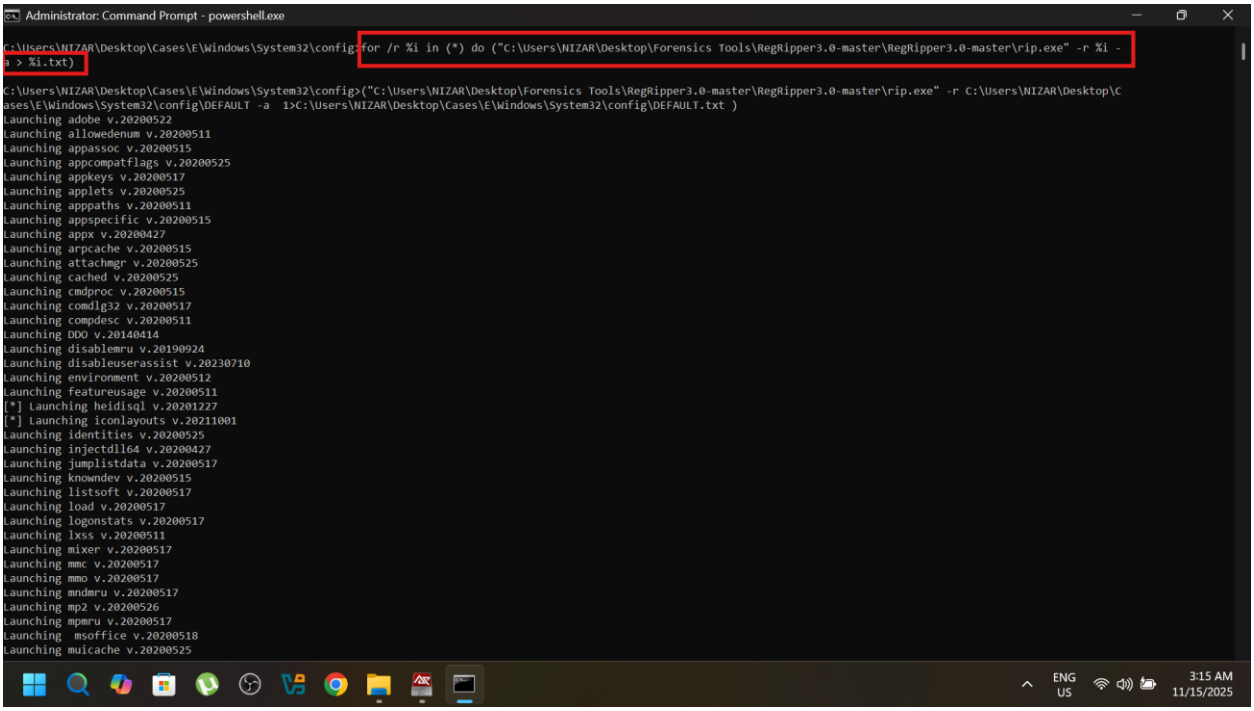
For comprehensive analysis, we'll use **RegRipper** to parse all registry hives automatically using **bulk processing**.

Command :

+> for /r %i in (*.dat) do ("C:\Users\NIZAR\Desktop\Tools\RegRipper3.0-master\RegRipper3.0-master\rip.exe" -r %i -a > %i.txt)

Command Breakdown :

- **for /r %i in (*.dat) do**: Loops through all **(.dat) files** in current directory
- **rip.exe -r %i -a**: Runs **RegRipper** on each file with automatic plugin selection
- **%i.txt**: Outputs results to text files

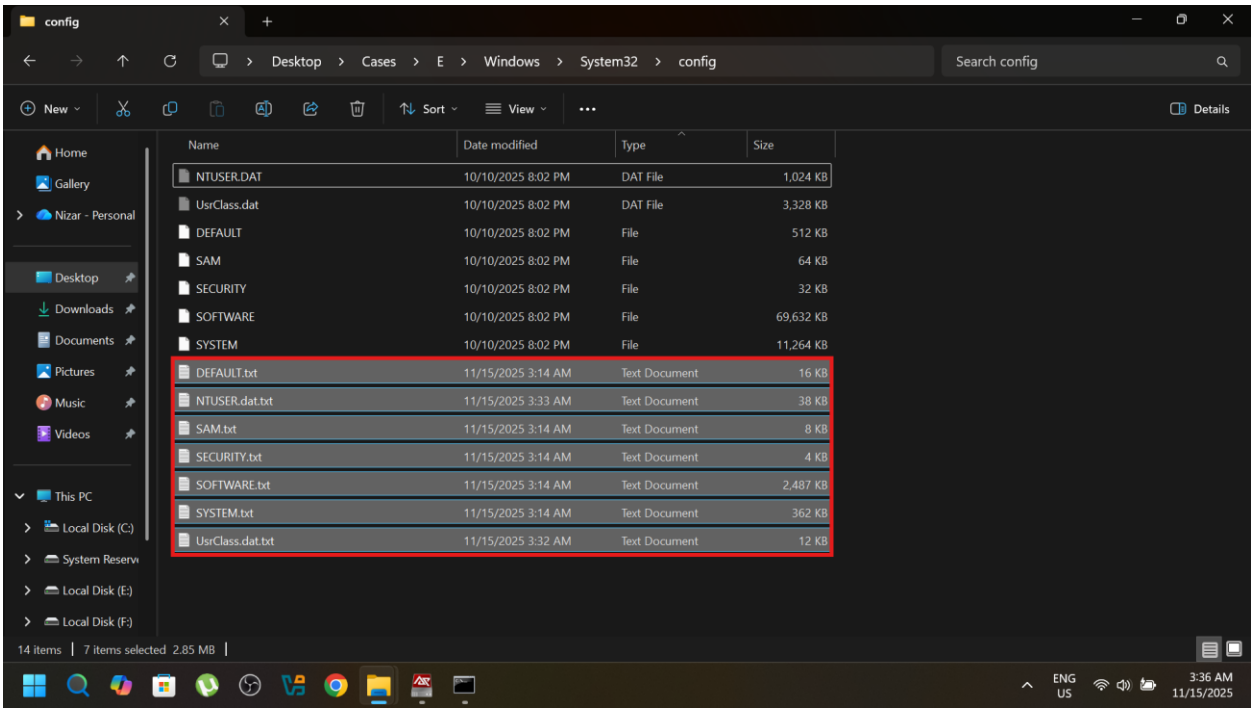


Bulk registry parsing command successfully completed, generating text files for all analyzed registry hives

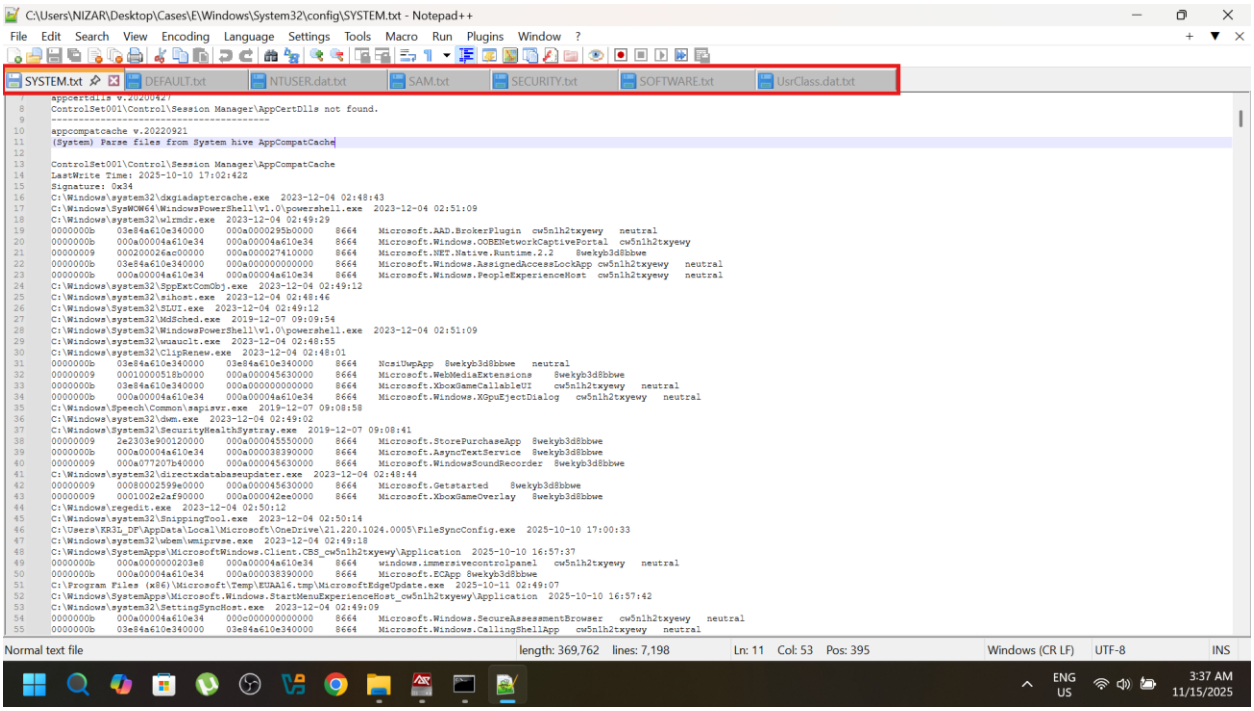
After processing:

- Select all **.txt files** in the output directory
- Open with **Notepad++**
- Use **Ctrl+F** to search across all files for **specific artifacts** and **plugins**

This automated approach ensures **we extract all available registry evidence efficiently**.



Generated text files containing parsed registry data from all hives, ready for analysis in Notepad++



All registry text files opened in Notepad++ for comprehensive search and analysis across parsed data

Answering Investigation Questions :

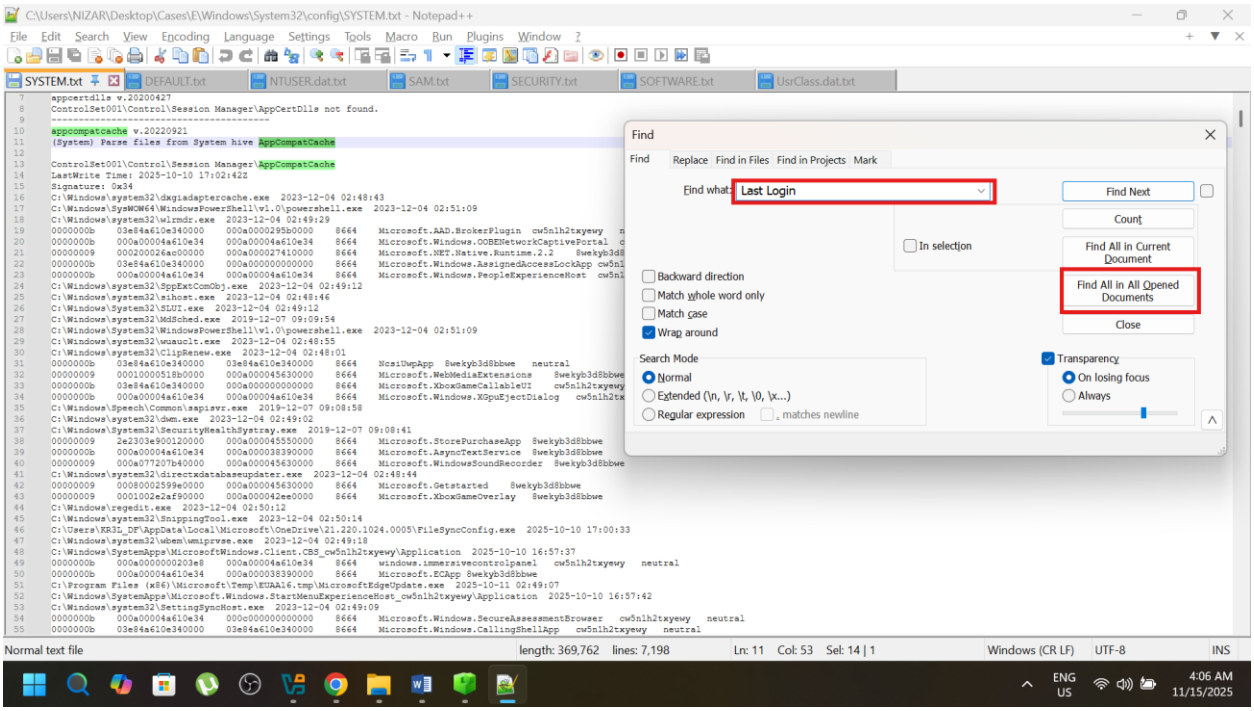
Now that we have completed our **forensic acquisition**, **triage**, and **parsing**, we can answer the key investigation questions using the collected evidence.

Questions to Answer:

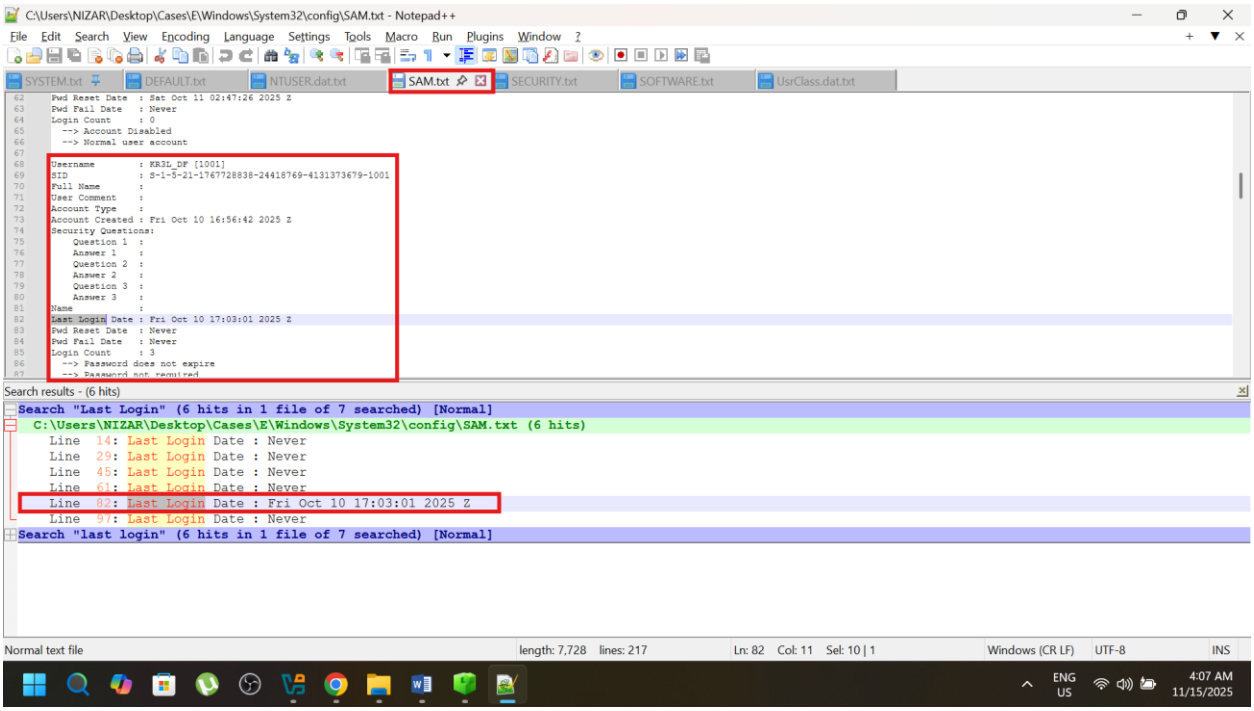
1. **Active accounts during the attack timeframe?**

To identify active accounts:

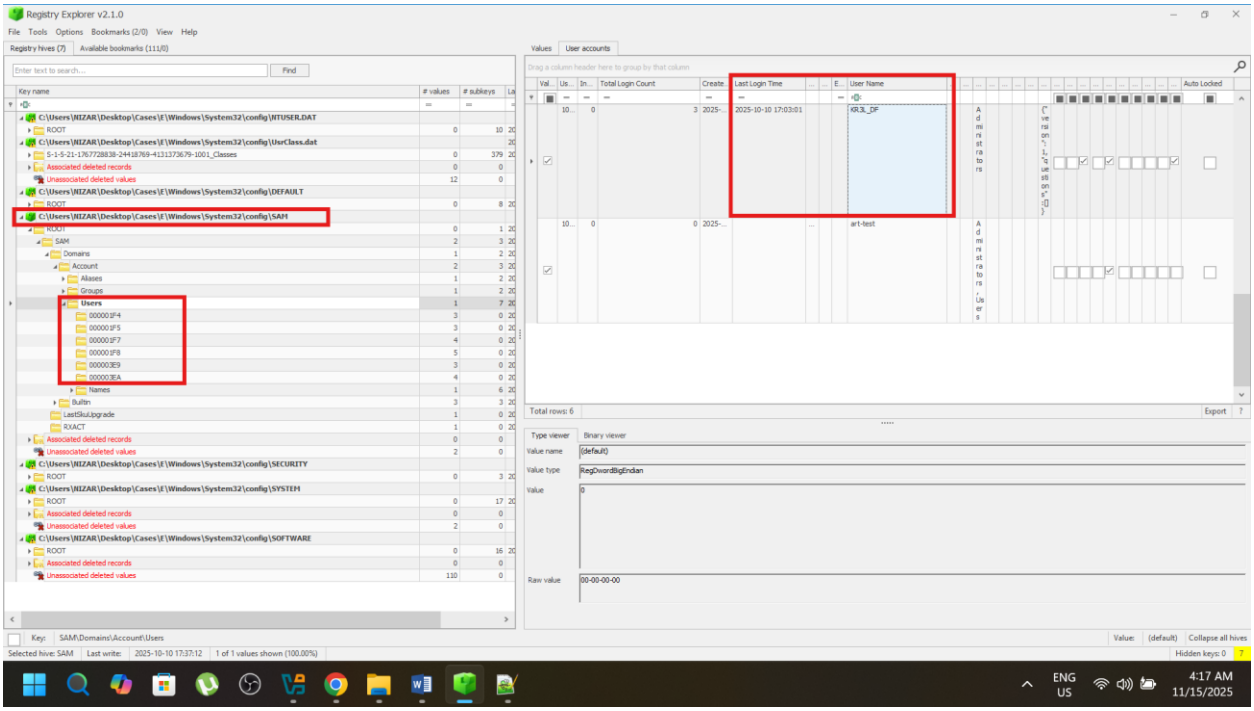
- Open all registry text files in **Notepad++**
- Press **Ctrl + F** to open search
- Search for: **Last Login**



Using Ctrl+F in Notepad++ to search "Last Login" across all registry files to identify active accounts during attack timeframe



Discovery of active user account "KR3l_DF" during the attack timeframe through registry analysis



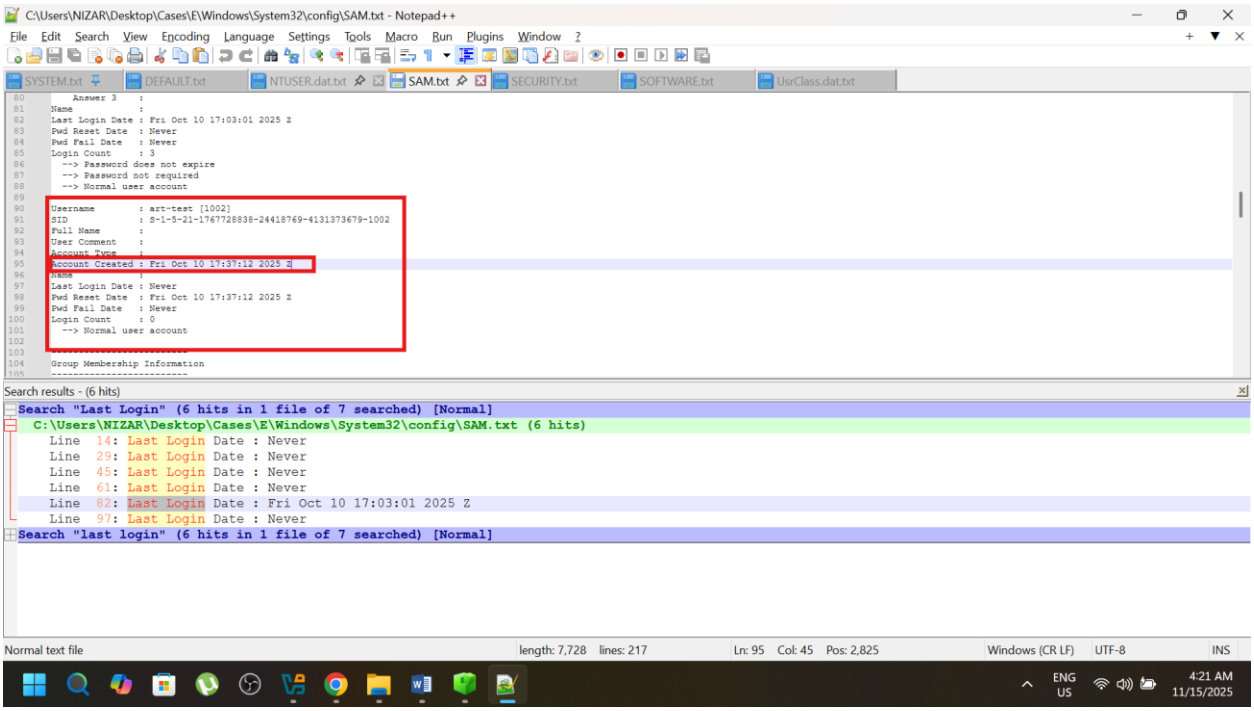
Identifying active user account "KR3l_DF" in Registry Explorer during the attack timeframe

So the answer is : KR3l_DF

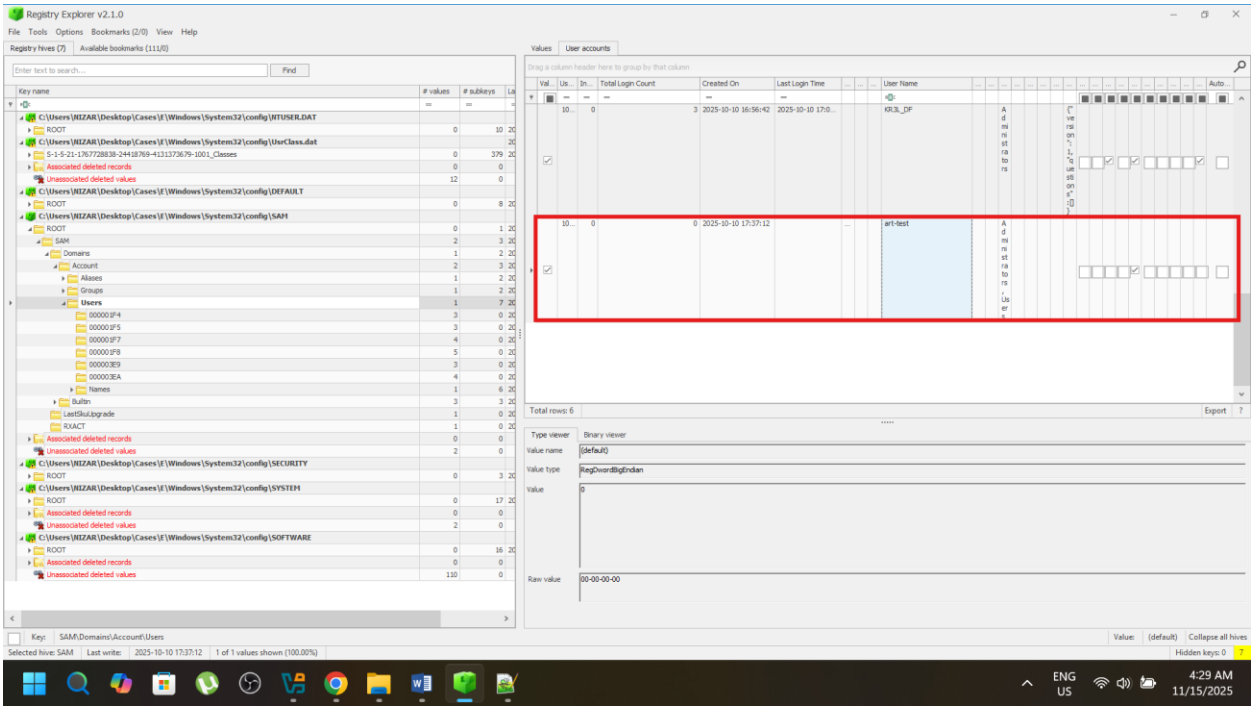
2. Which account(s) were created?

As we investigated, we discovered **a new suspicious account** was created just **30 minutes** after **KR3l_DF** logged in, indicating potential attacker **persistence activity**.

Maybe this was the attacker creating a **backdoor account** to maintain access to the system



Newly created suspicious account "art-test" detected in registry just 30 minutes after KR3I_DF login



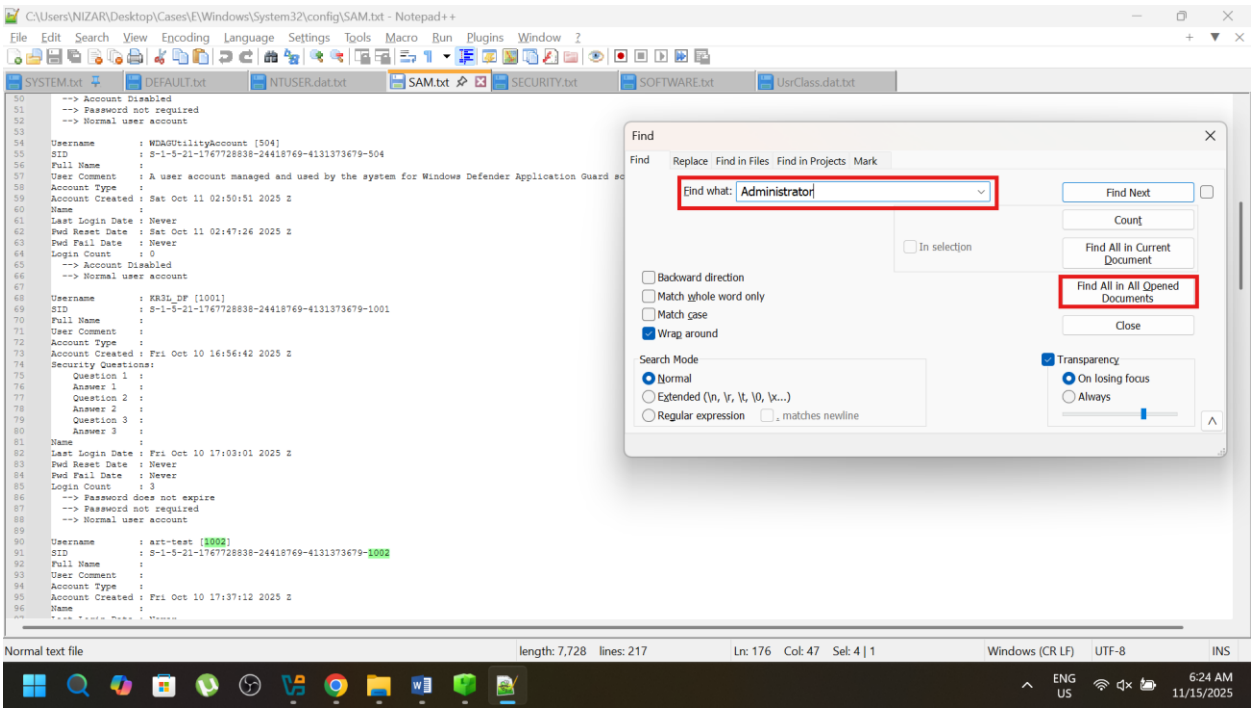
Recently created "art-test" account shown in Registry Explorer, timestamped 30 minutes after initial KR3I_DF compromise

So the answer is : art-test

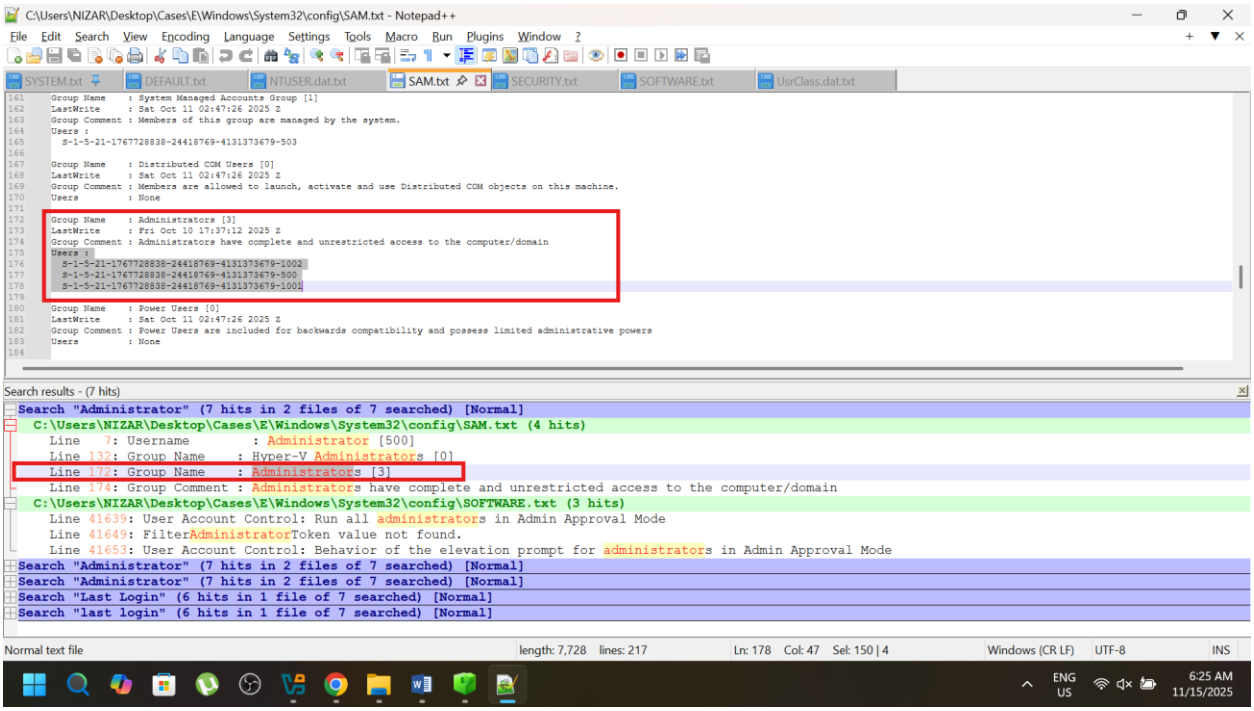
3. Which accounts are Administrator group members?

By examining the **Administrators group** membership in the registry, we can identify all user **SIDs** with administrative privileges on the system

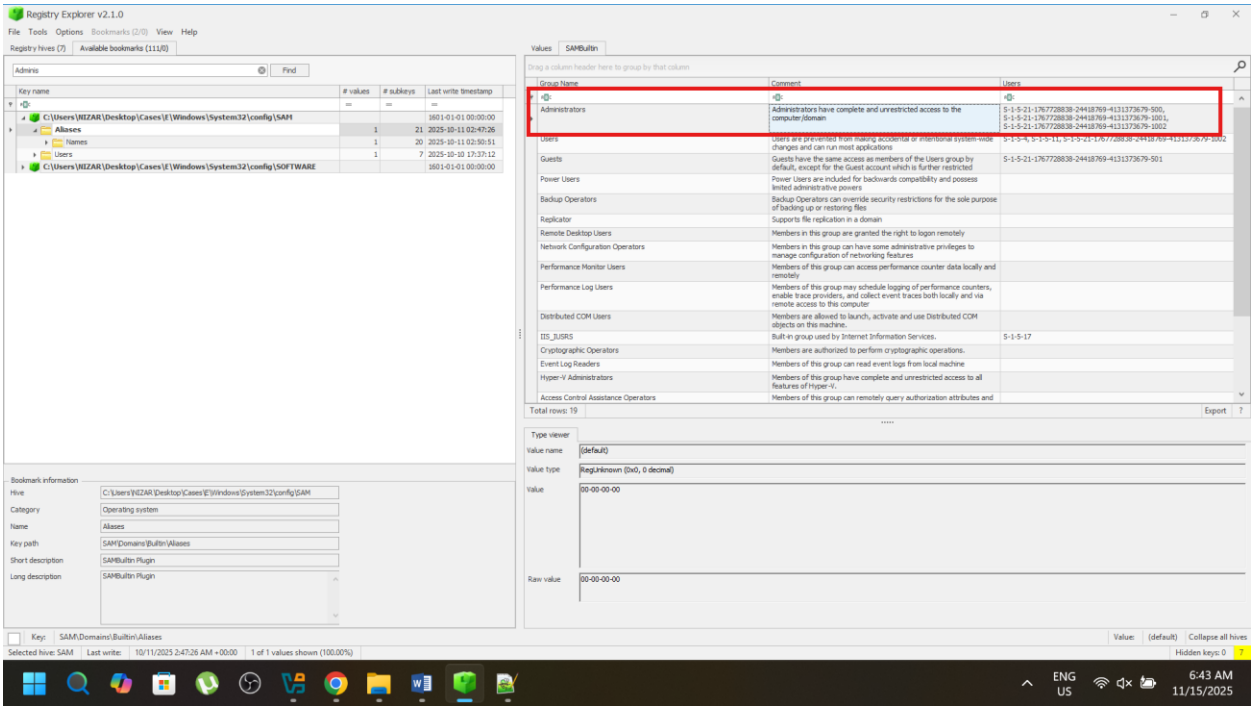
SIDs (Security Identifiers) are unique numbers that identify users, groups, and computers in Windows - like digital fingerprints for each account.



Searching for "Administrators" group in Notepad++ to identify all member SIDs with administrative privileges



Viewing all SID members within the Administrators group, showing accounts with administrative privileges



Administrators group membership viewed in Registry Explorer, displaying all member SIDs with administrative rights

So the answer is :

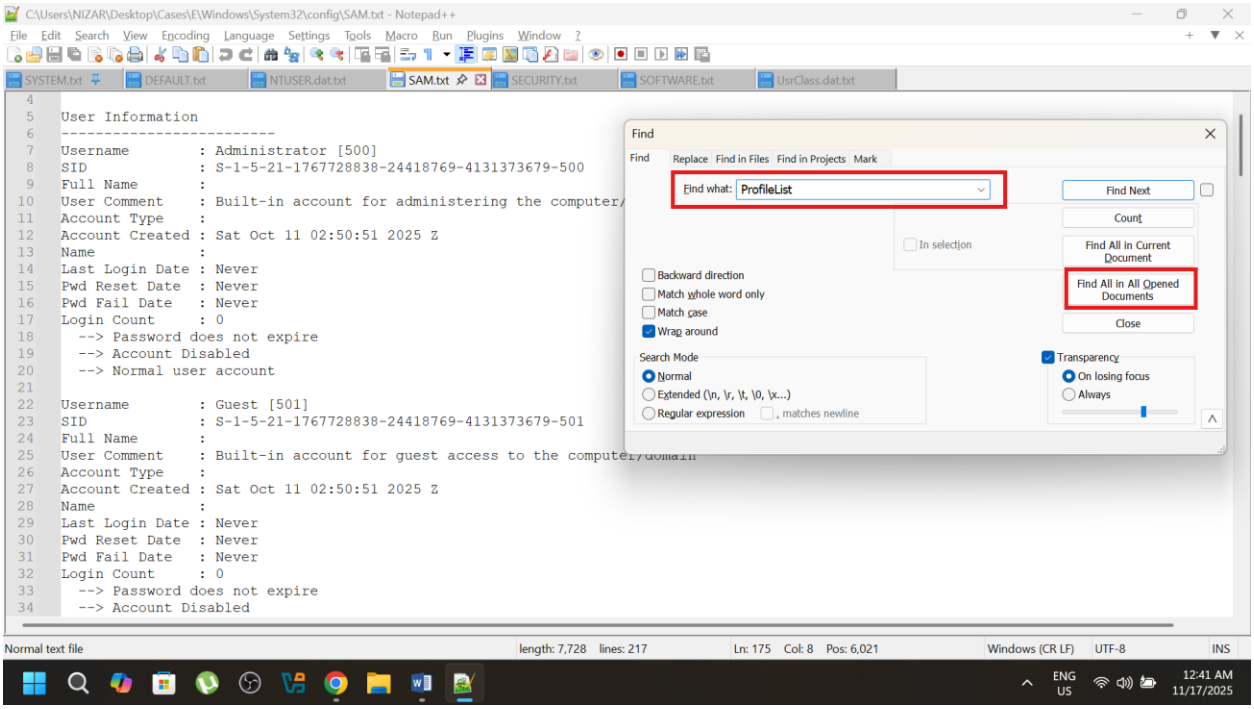
S-1-5-21-1767728838-24418769-4131373679-1002

S-1-5-21-1767728838-24418769-4131373679-500

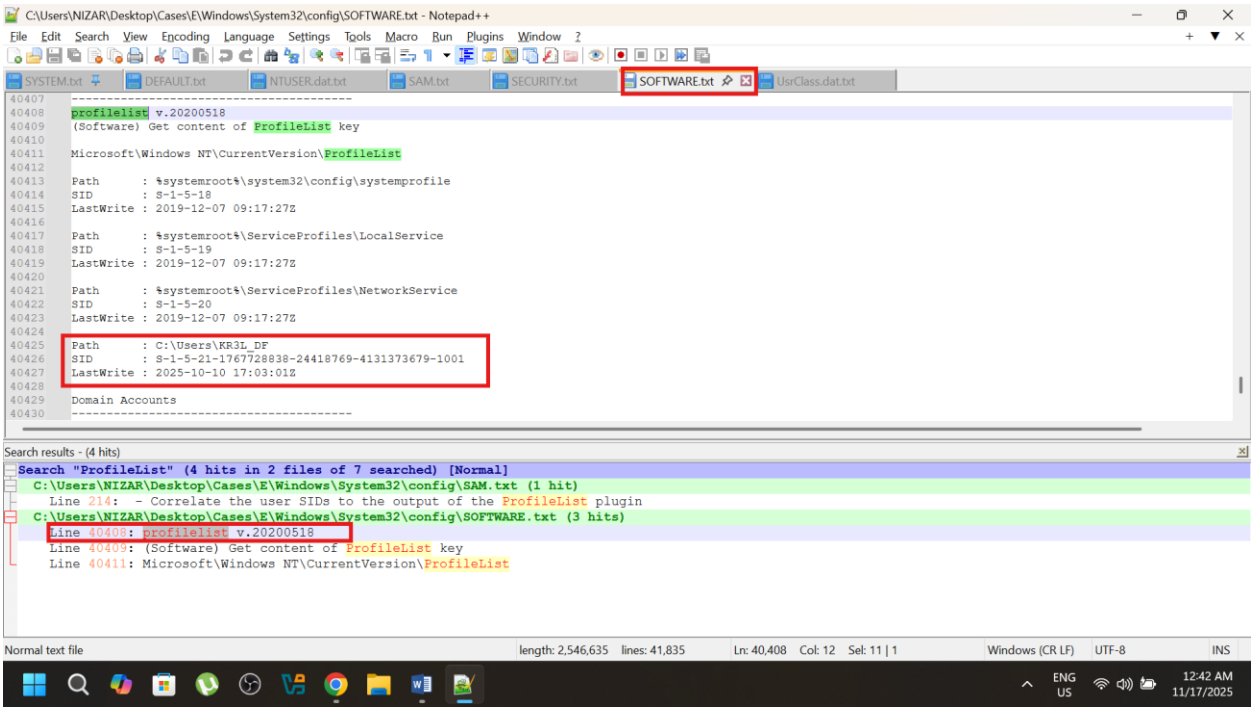
S-1-5-21-1767728838-24418769-4131373679-1001

4. Which users have profiles?

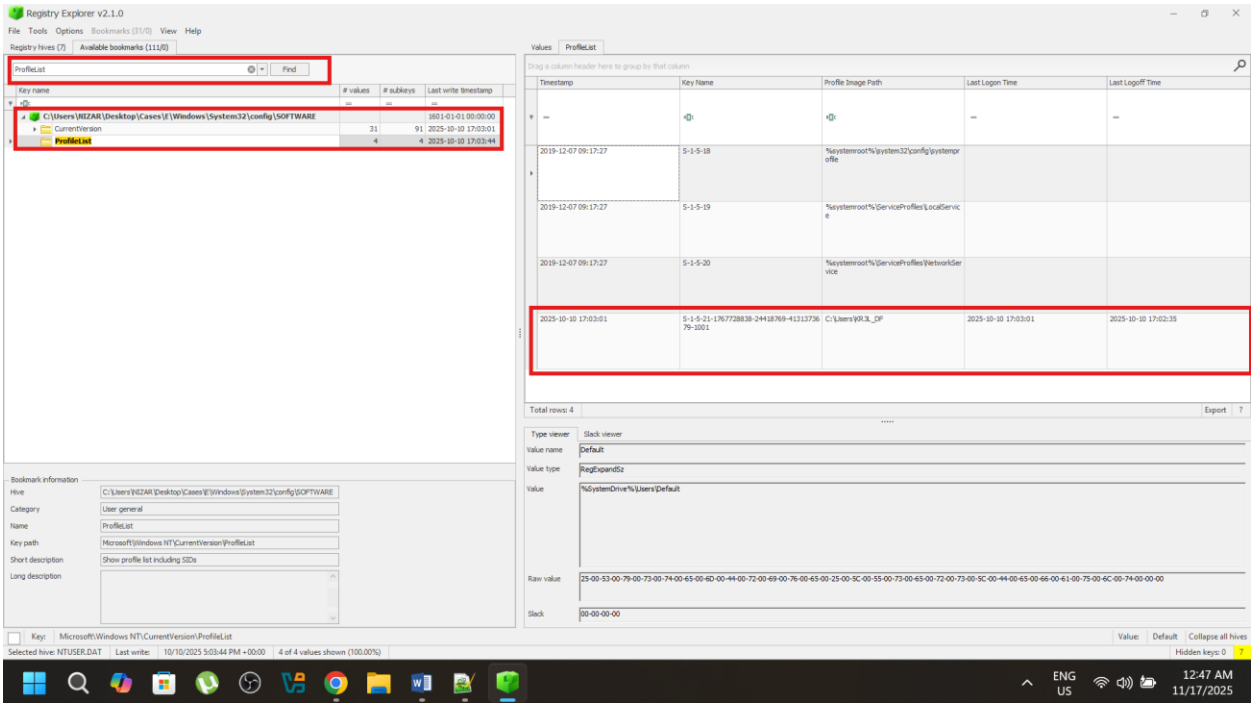
A user profile is a collection of settings and data that Windows maintains for each user account - it contains their **desktop**, **documents**, **browser history**, and **personal configurations**.



Using Notepad++ to search for "ProfileList" across registry files to identify all user profiles on the system



KR3L_DF user profile identified in ProfileList with corresponding SID and profile path



KR3L_DF user profile visible in Registry Explorer under ProfileList with complete profile path details

So the answer is : KR3L_DF