# Ransomware Analysis: Rhysida, BlackCat (ALPHV & Sphynx), and LockBit (2.0 / 3.0)

Author: Victor Chinedu Ndukwe

Repository: GitHub — Research Write-up

Date: 23 October 2025

## Abstract

This document summarises static and dynamic analyses of several ransomware strains—Rhysida, BlackCat (ALPHV and Sphynx), and LockBit (2.0 and 3.0)—performed on a Windows 10 client within a controlled virtual lab. The goal is to present findings useful for defenders and researchers, including indicators of compromise (IOCs), YARA markers, behaviour summaries, and detection capabilities of IDS tools (Snort, Suricata, OSSEC). Images from the original analysis (PeStudio screenshots, Process Hacker views, ransom notes, and IDS logs) are embedded throughout.

## Table of Contents

1. Rhysida  
2. BlackCat ALPHV  
3. BlackCat Sphynx  
4. LockBit 2.0  
5. LockBit 3.0  
6. Data collection and IDS evaluation  
7. Conclusions and recommendations  
8. Appendix — Figures and IOCs

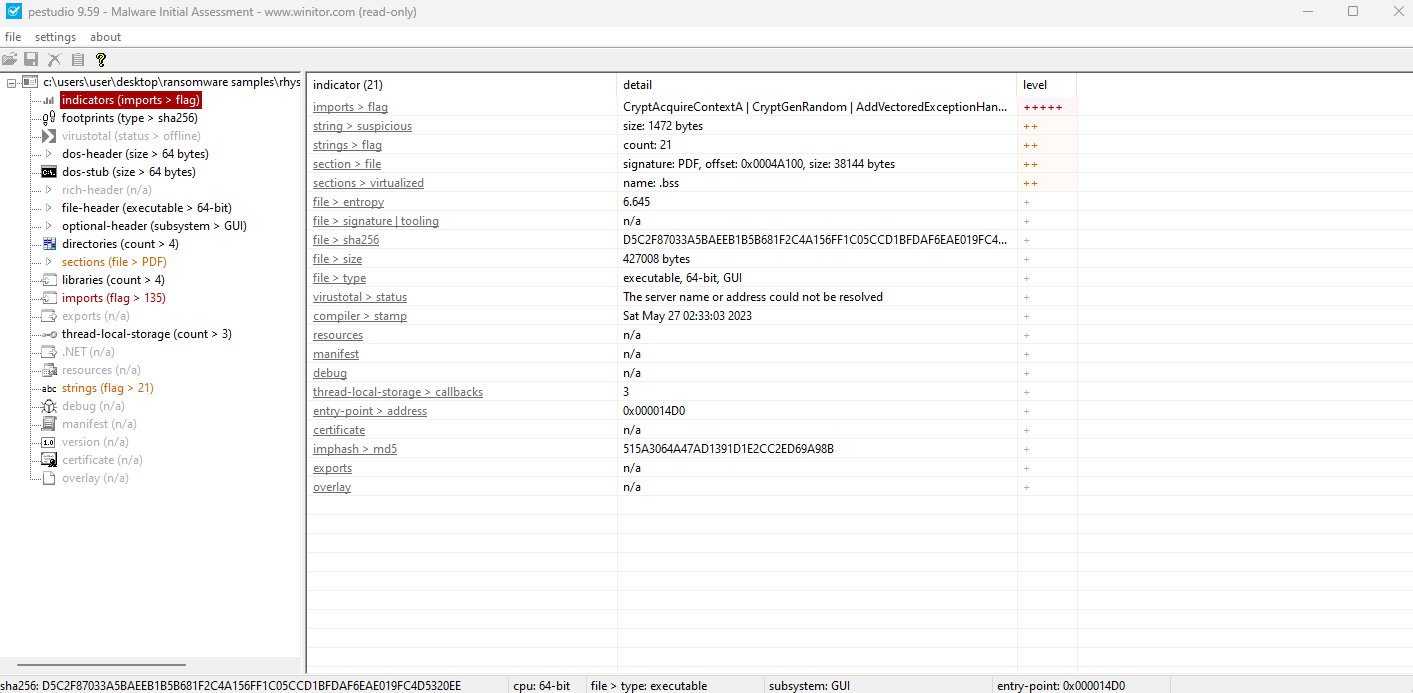
## 1. Rhysida

Rhysida is a ransomware strain notable for high encryption speed, advanced encryption techniques, and evasion methods. It encrypts user files and demands ransom for decryption. The following summarises static and dynamic analysis findings.

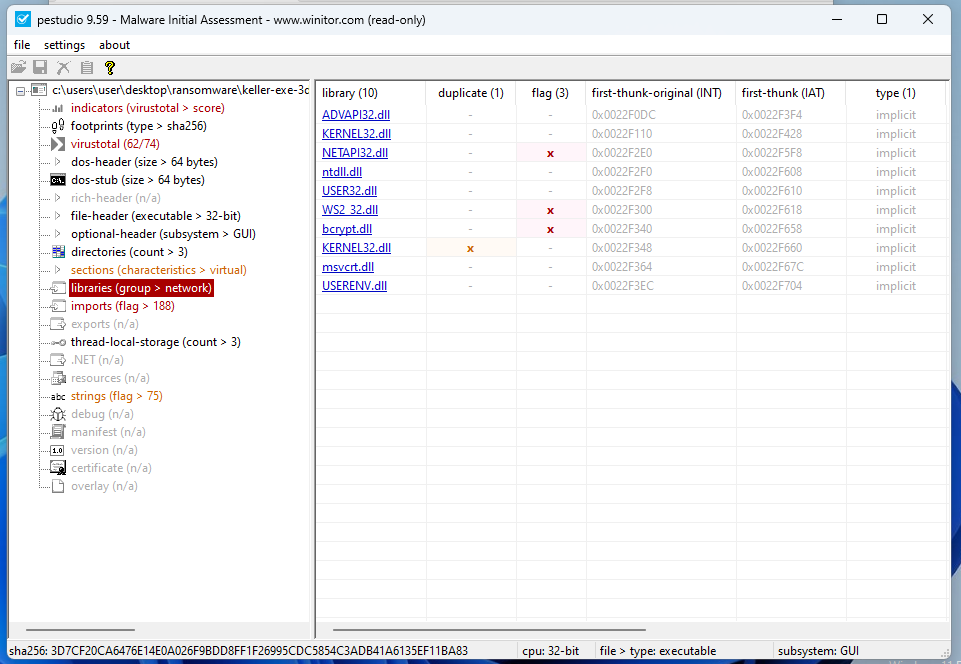
### 1.1 Static Analysis

Entropy and file structure: PeStudio reported an entropy value of 6.645 for the Rhysida sample, indicating packing or encryption. A section flagged with a PDF signature (size ≈ 38,144 bytes) suggests the sample embeds or masquerades as a PDF, possibly as a decoy or payload carrier.

Imports and libraries: Rhysida imports advapi32.dll, kernel32.dll, msvcrt.dll, user32.dll and uses functions such as CryptAcquireContextA, CryptGenRandom, GetCurrentProcessId, VirtualProtect, VirtualQuery, and AddVectoredExceptionHandler. These indicate cryptographic operations, memory manipulation, anti-debugging and process control.



*Figure: image1.png*



*Figure: image10.png*

### 1.2 Dynamic Analysis

Observed behaviours: using Process Hacker, VirusTotal and APIminer, Rhysida created files (notably 'criticalbreachdetected.pdf'), dropped files with a .rhysida extension, modified registry keys (including .pdf associations), and interacted with processes such as cmd.exe, sc.exe, reg.exe, dllhost.exe and SearchIndexer. No network traffic was observed during the analysis, suggesting the sample can operate offline.

YARA markers and telemetry indicators: powershell, ThreadControl\_\_Context, SEH\_\_\_vectored, Check\_OutputDebugStringA\_iat, anti\_dbg. Encryption methodology: ChaCha20-based CSPRNG for file encryption, intermittent 1 MB block encryption, AES-256-CTR for file content and RSA-4096 for key wrapping.

## 2. BlackCat (ALPHV)

BlackCat (ALPHV) is a modular, cross-platform ransomware family. Static analysis shows high entropy (≈6.863) and YARA matches indicating network and spreading capabilities.

### 2.1 Static Analysis

Key libraries: WS2\_32.dll, NETAPI32.dll, bcrypt.dll, USERENV.dll, ntdll.dll, msvcrt.dll, USER32.dll, ADVAPI32.dll, KERNEL32.dll. YARA matches included network and spreading indicators, registry manipulation and mutex usage.

### 2.2 Dynamic Analysis

Behaviour: drops files with .sykffle extension, deletes temporary files and logs, interacts with registry keys to persist and possibly disable security features, injects into processes like wuapihost.exe and wmiprvse.exe, and drops ransom note RECOVER-${EXTENSION}-FILES.txt. Some images of infected files and ransom notes are included.

## 3. BlackCat Sphynx

Sphynx is a stealthy variant of BlackCat. Static entropy observed ≈6.805. The sample demonstrates anti-VM and anti-sandbox techniques and uses a broad set of Windows libraries.

### 3.1 Static Analysis

Notable libraries and functions: advapi32.dll (AdjustTokenPrivileges), bcrypt.dll (BCryptGenRandom), kernel32.dll (CreateProcessW, VirtualProtect), netapi32.dll (NetShareEnum), mpr.dll (WNetAddConnection2W), ole32/oleaut32, rstrtmgr.dll, secur32.dll and IPHLPAPI.DLL.

### 3.2 Dynamic Analysis

Behaviour: deletes WER logs, drops files in WER\Temp, modifies registry keys, creates and injects into processes such as wuapihost.exe and wmiadap.exe. API calls observed (APIminer) include NtProtectVirtualMemory, NtClose, LdrGetDllHandle. No network traffic was observed during the execution in the controlled lab.

## 4. LockBit 2.0

LockBit 2.0 is a fast and efficient ransomware family. Static entropy ≈6.781 and contains YARA rules suggesting anti-VM, privilege escalation and spreading capabilities.

### 4.1 Static Analysis

Key libraries: WS2\_32.dll, CRYPT32.dll, gdiplus.dll, SHLWAPI.dll, MPR.dll, ntdll.dll, msvcrt.dll, KERNEL32.dll, USER32.dll, ADVAPI32.dll, SHELL32.dll, ole32.dll.

### 4.2 Dynamic Analysis

Behaviour: executes system commands to delete shadow copies and backups (vssadmin, wmic, bcdedit, wbadmin), encrypts files with .lockbit extension, modifies registry for persistence, creates mutexes and spawns process trees to disable recovery and cover tracks.

## 5. LockBit 3.0

LockBit 3.0 improves upon LockBit 2.0 with enhanced evasion techniques (HeavensGate, DebuggerHiding\_\_Thread) and refined file-system manipulation. Entropy ≈6.877.

### 5.1 Static Analysis

Notable functions: MessageBoxW, LoadResource, WriteFile, CreateFileW, NtClose, RtlAllocateHeap. Uses HeavensGate technique to execute 64-bit code from 32-bit processes.

### 5.2 Dynamic Analysis

Behaviour: extensive file operations, registry manipulation, process injection (WMIADAP.EXE), and capability to capture screenshots and manipulate tokens for privilege escalation. Ransom note delivery and potential data exfiltration are consistent with modern ransomware tactics.

## 6. Data collection and IDS evaluation

Methodology: A baseline was captured, each ransomware sample was deployed to a Windows 10 client machine, IDS tools (Snort, Suricata, OSSEC) logged activity, and snapshots restored after each test.

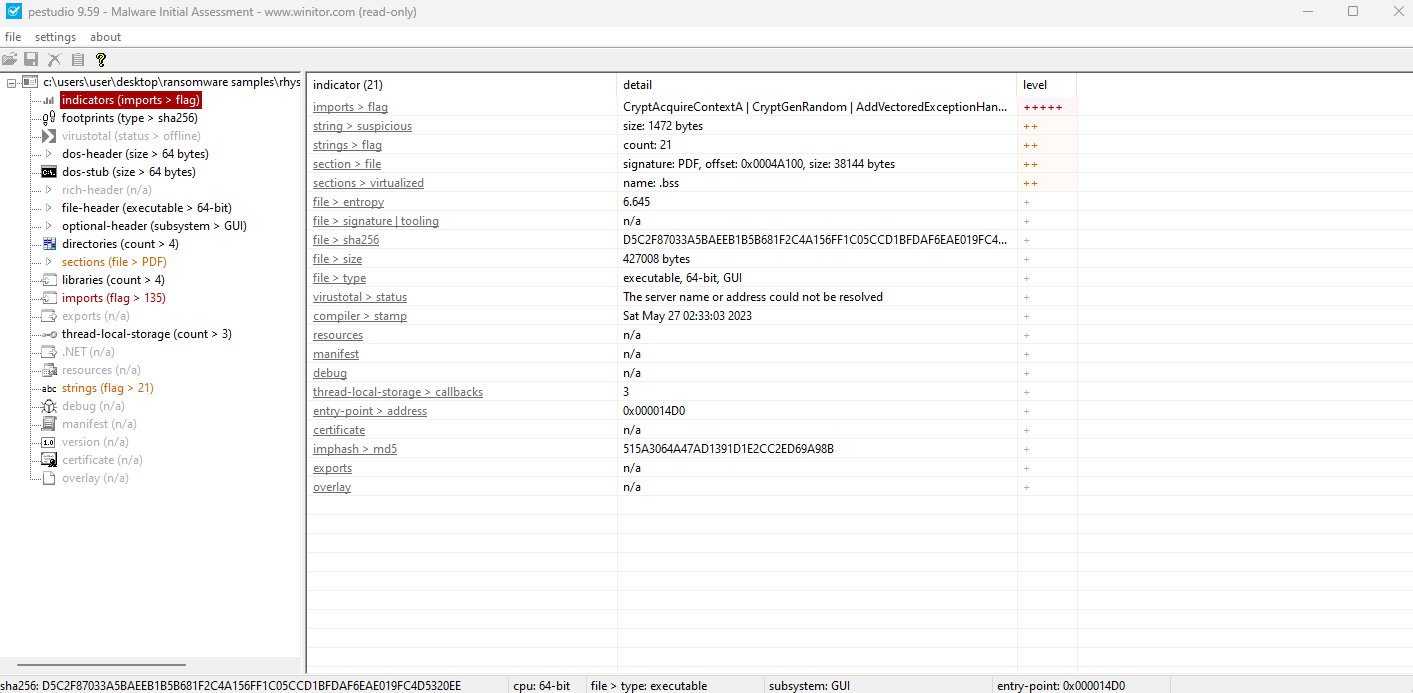
Findings: Snort and Suricata (network-based IDS) generated no significant alerts for the samples in the isolated virtual lab. OSSEC (host-based IDS) detected registry changes, file modifications, ransom note creation and process anomalies for multiple samples, demonstrating host-level detection strengths.

## 7. Conclusions and recommendations

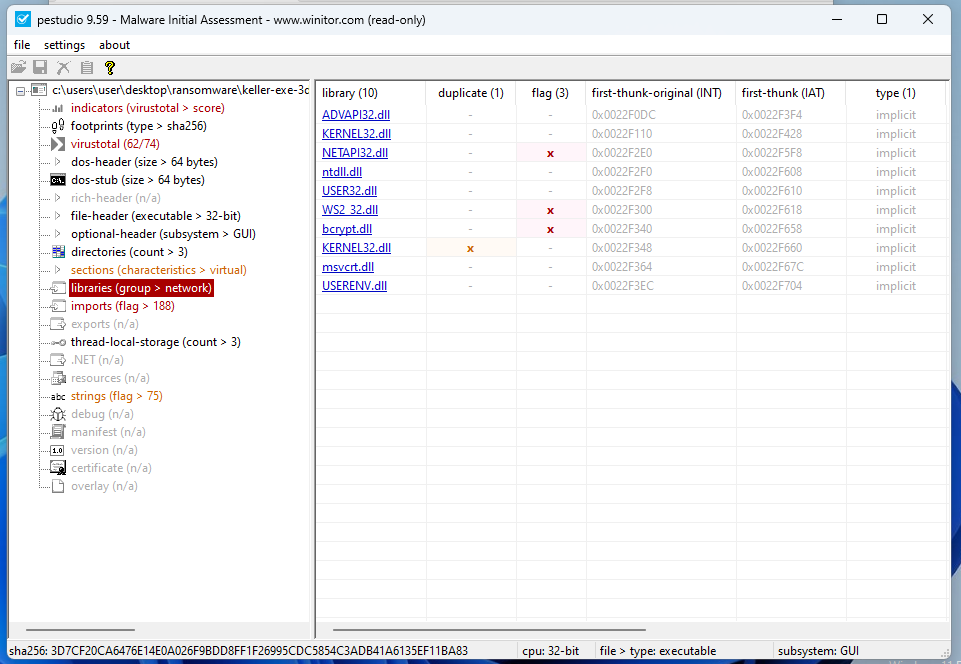
Summary: Host-based detection (OSSEC) outperformed network-based IDS (Snort/Suricata) in this lab for ransomware that operated offline and primarily modified the host. Recommendations: Employ layered detection combining host and network telemetry, deploy endpoint detection and response (EDR), ensure regular backups and immutable snapshots, monitor for registry changes and abnormal process creation, and craft YARA rules and Sigma signatures from the observed markers.

## 8. Appendix — Figures and IOCs

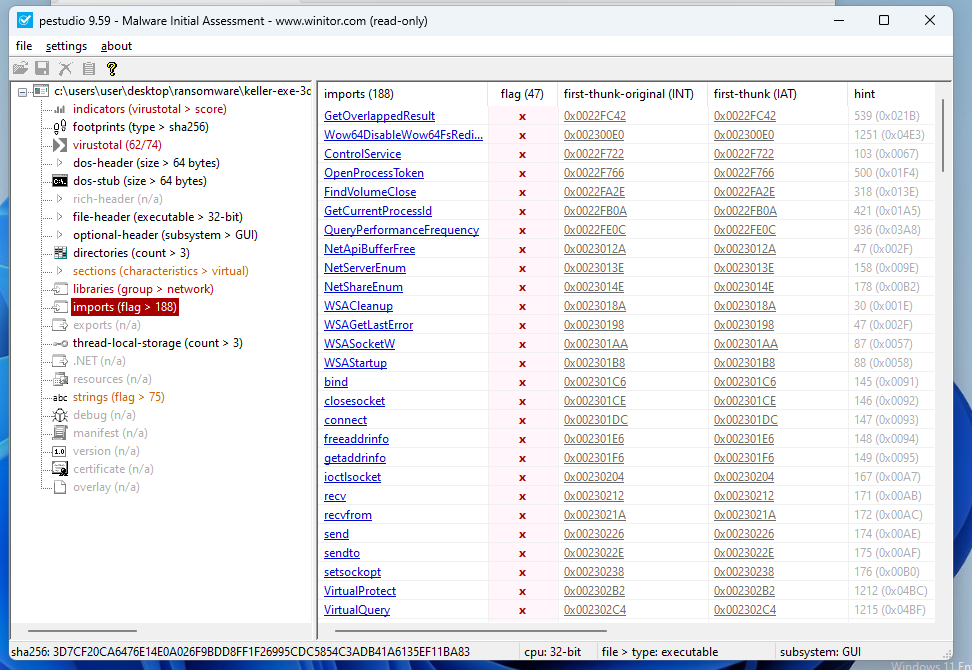
This appendix includes the figures extracted from the original analysis and suggested IOCs derived from the observations.



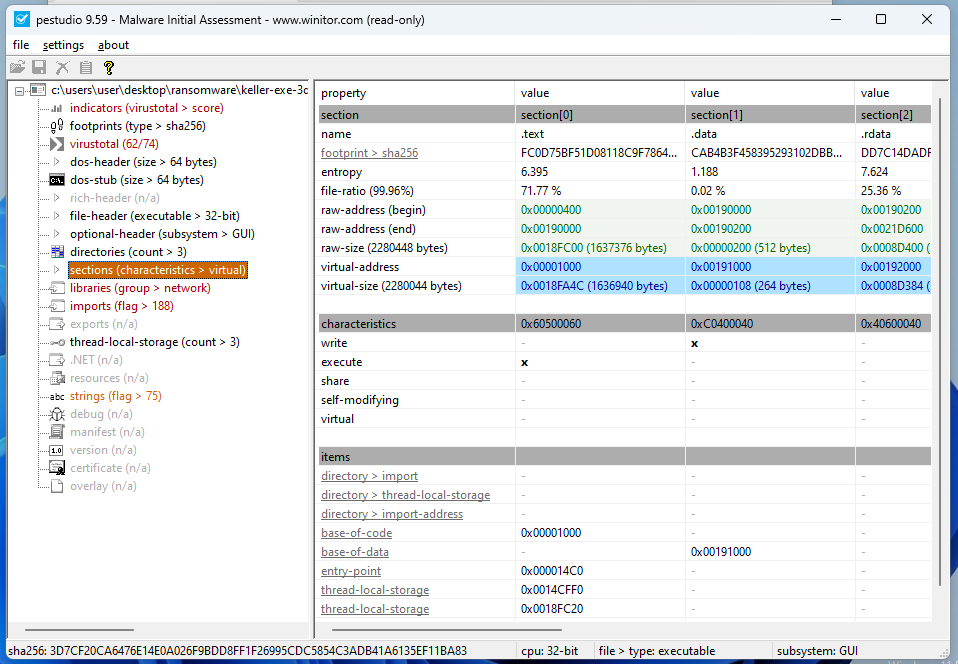
*Appendix Figure: image1.png*



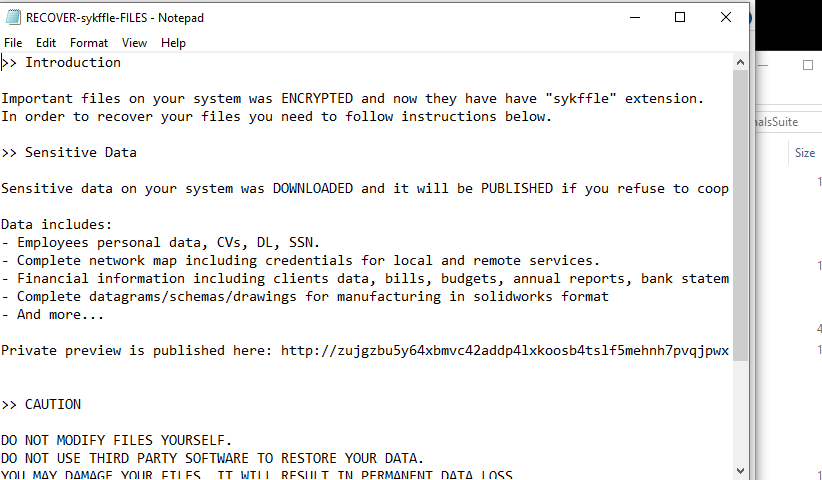
*Appendix Figure: image10.png*



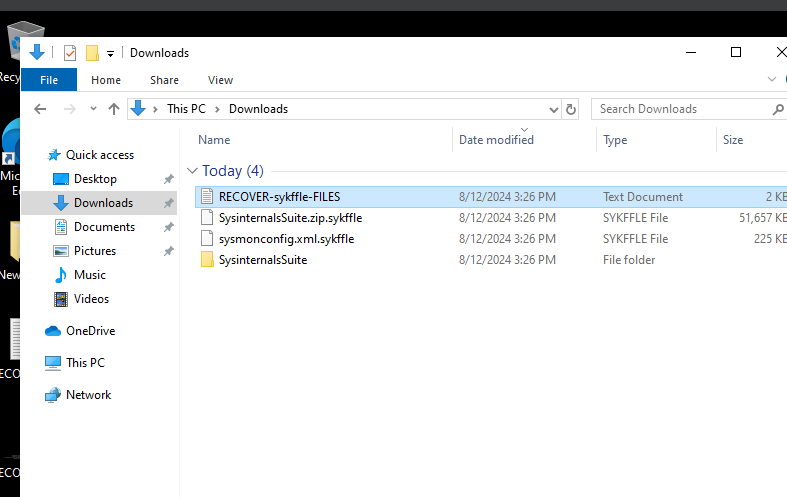
*Appendix Figure: image11.png*



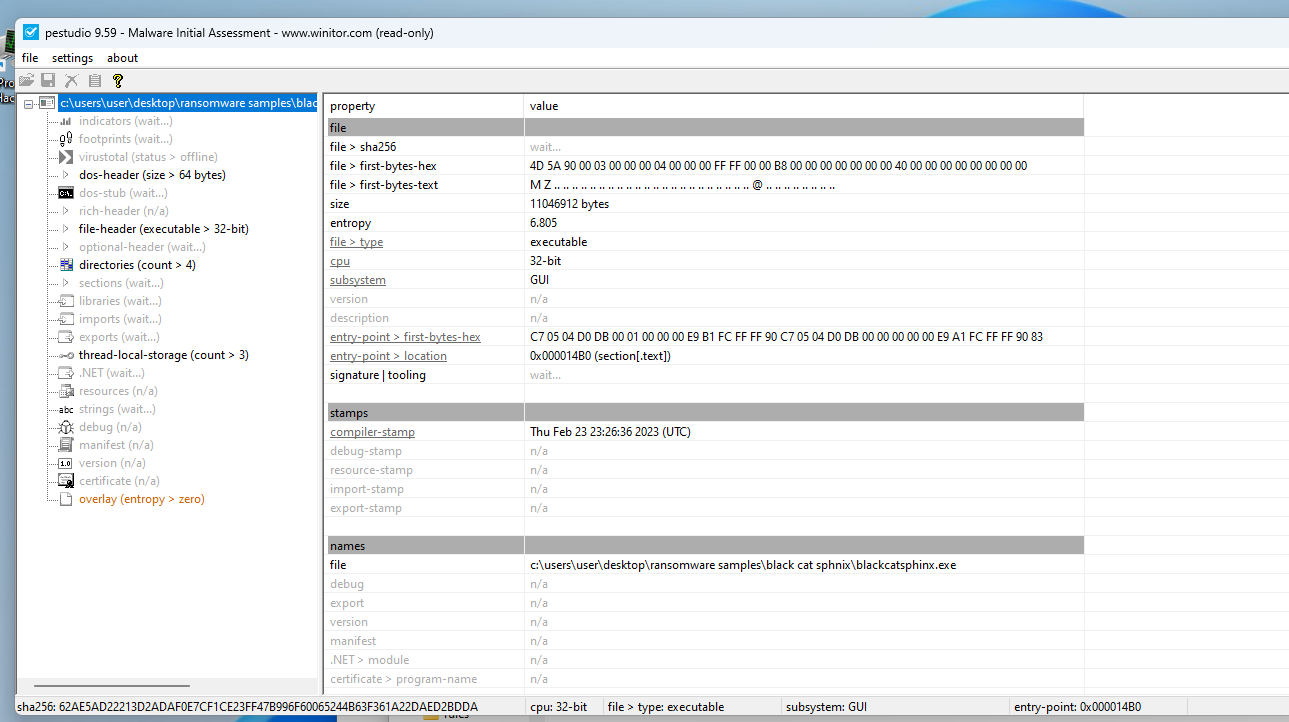
*Appendix Figure: image12.png*



*Appendix Figure: image13.png*



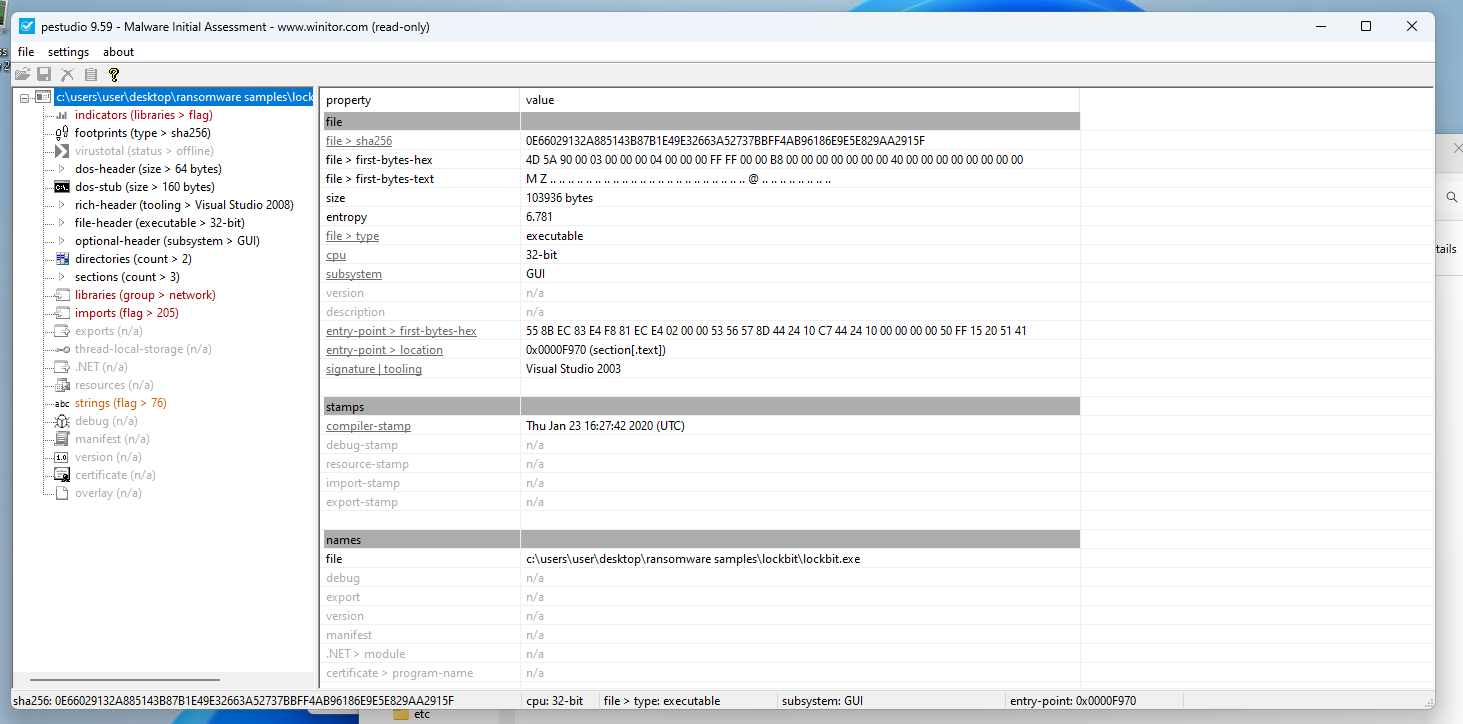
*Appendix Figure: image14.png*



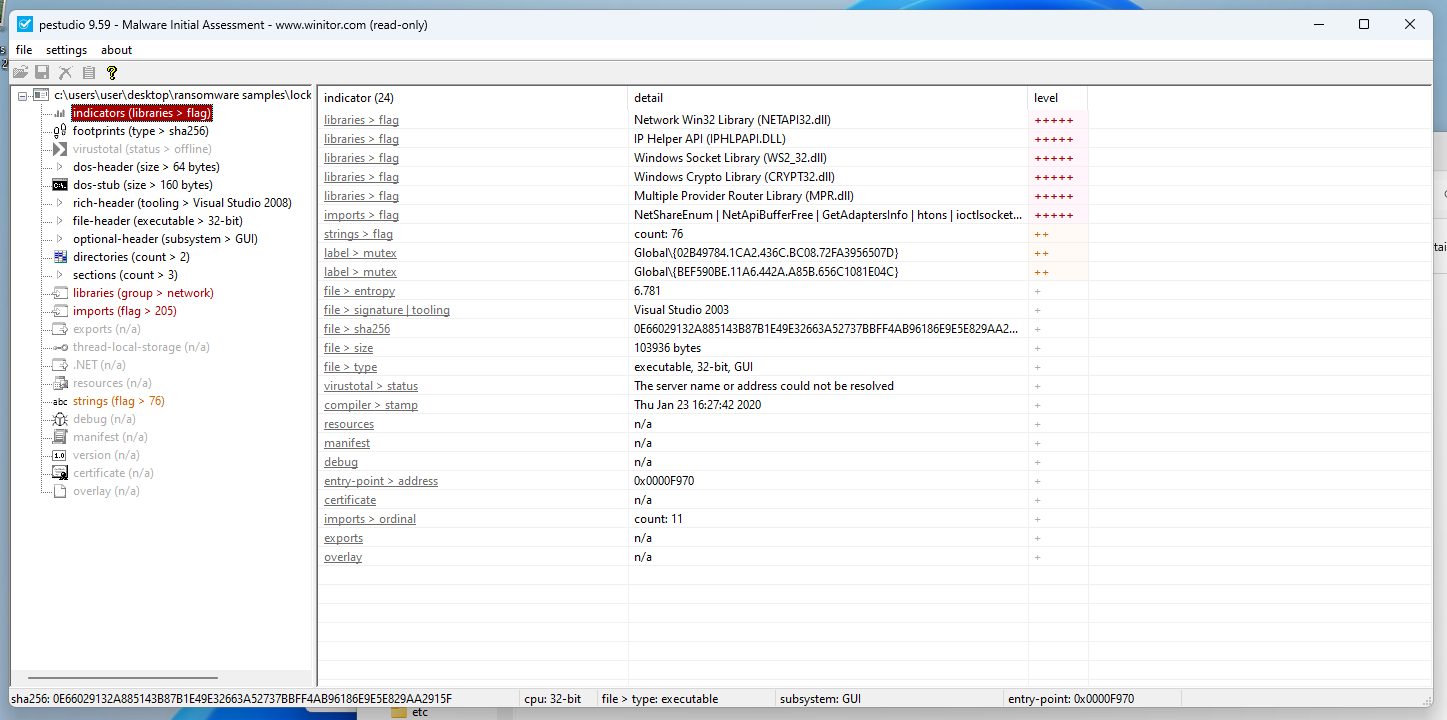
*Appendix Figure: image15.png*



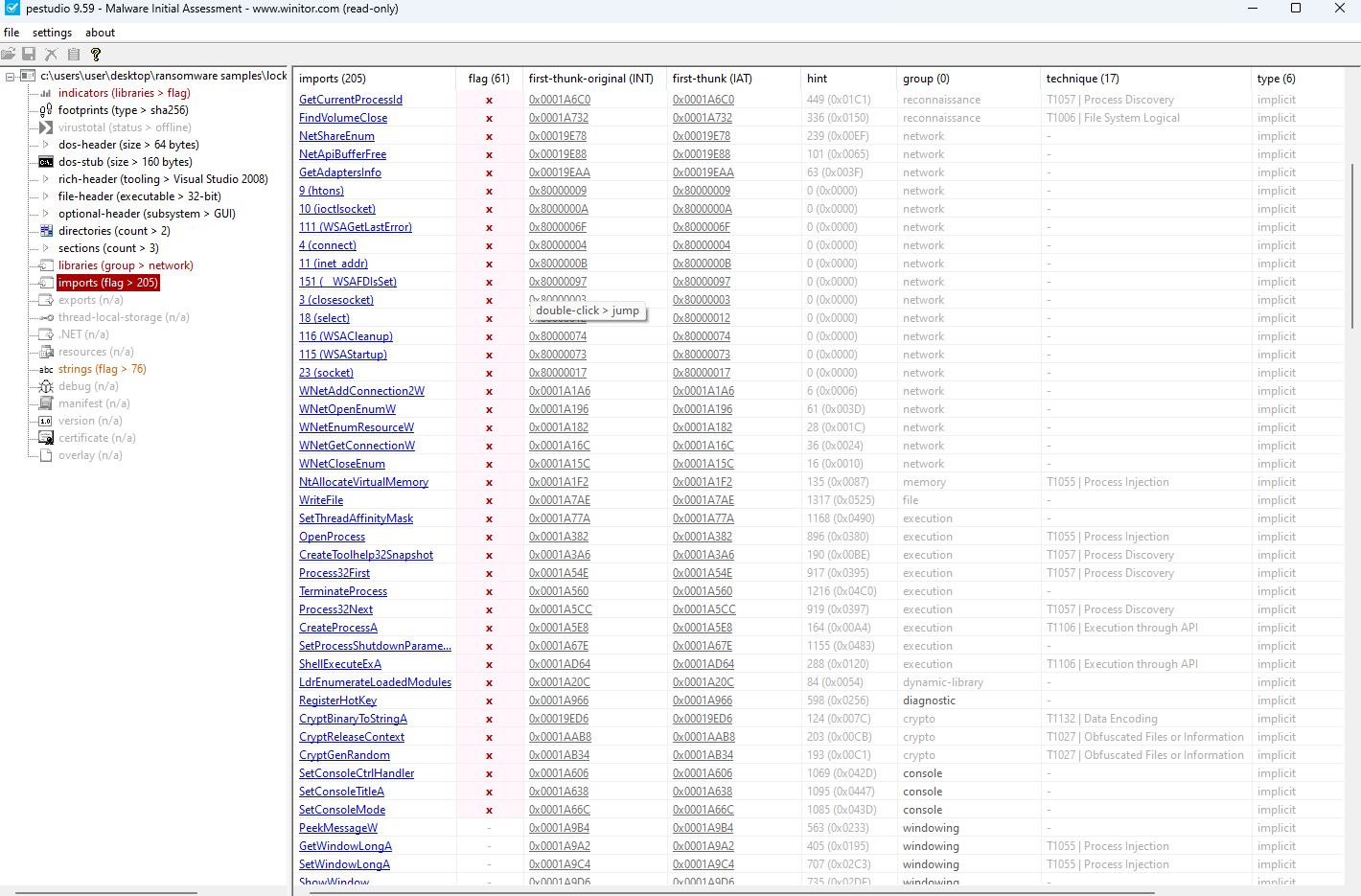
*Appendix Figure: image16.png*



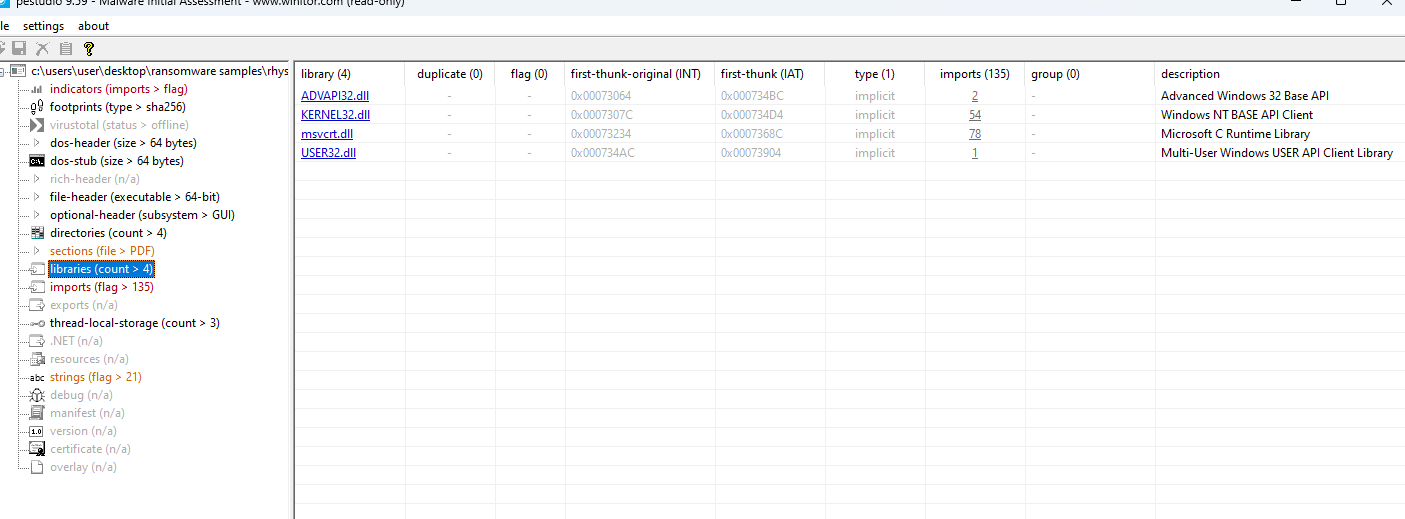
*Appendix Figure: image17.png*



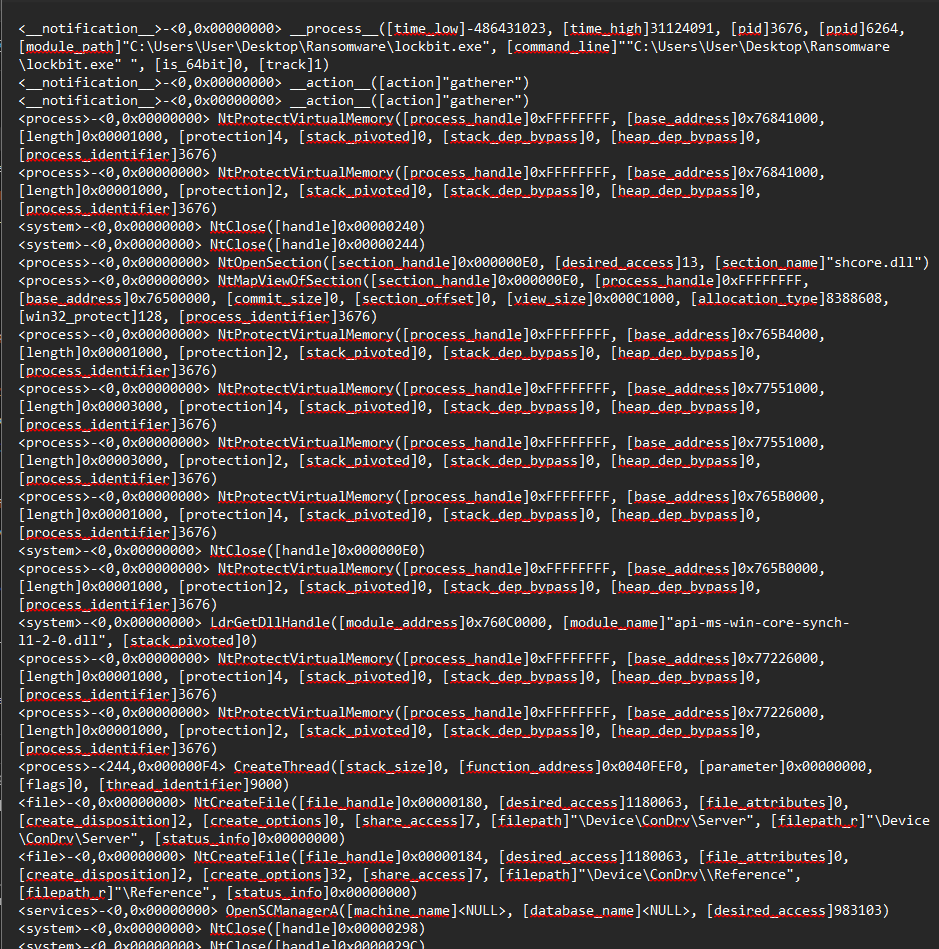
*Appendix Figure: image18.png*



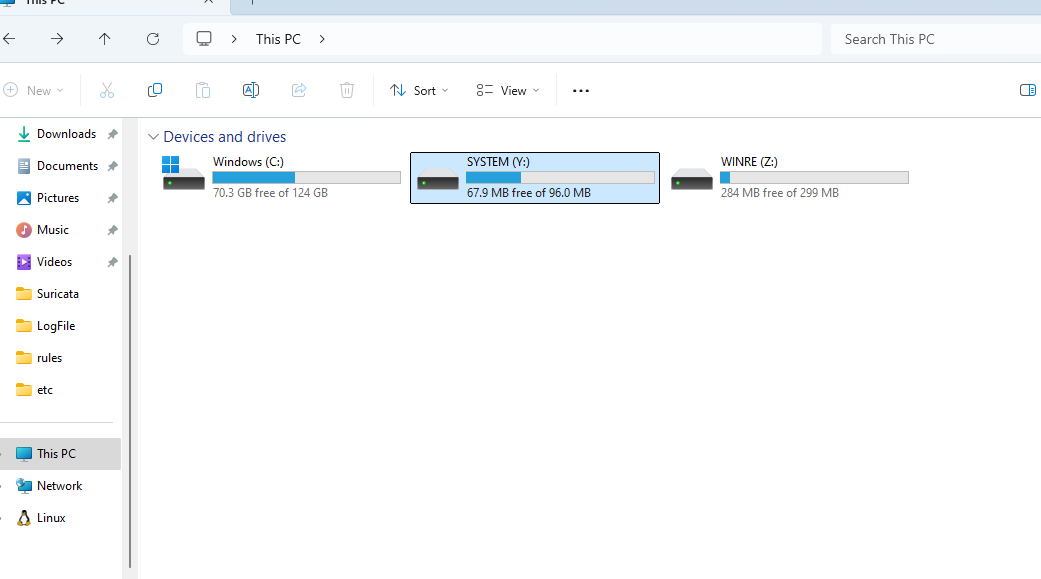
*Appendix Figure: image19.png*



*Appendix Figure: image2.png*



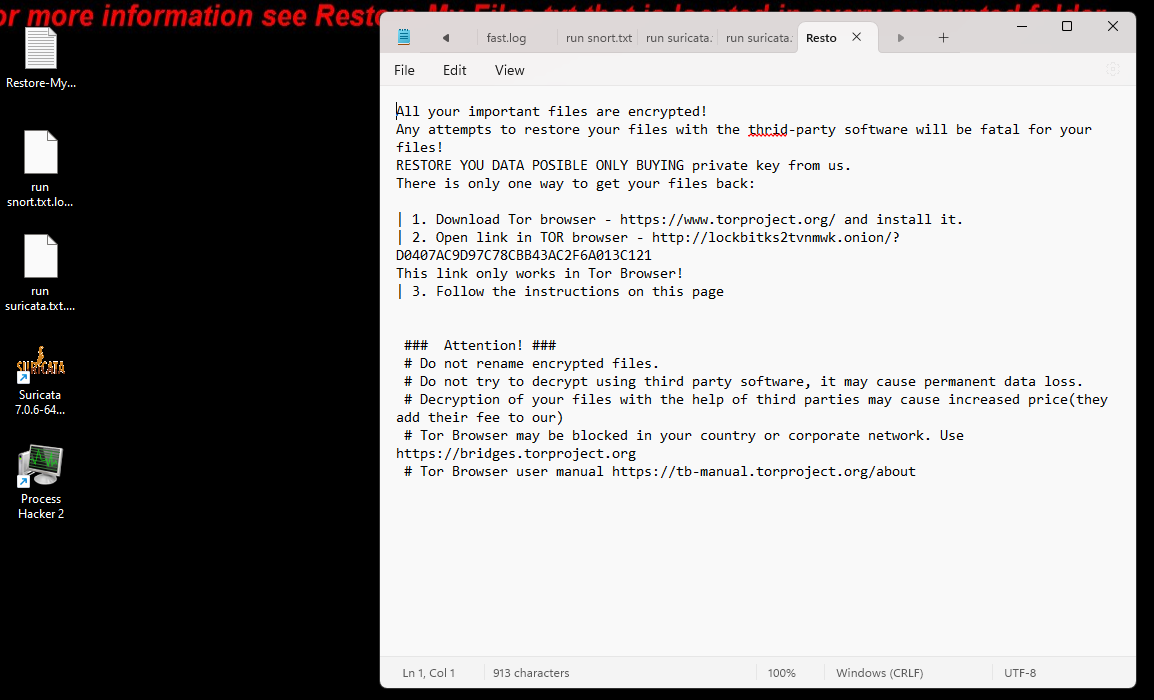
*Appendix Figure: image20.png*



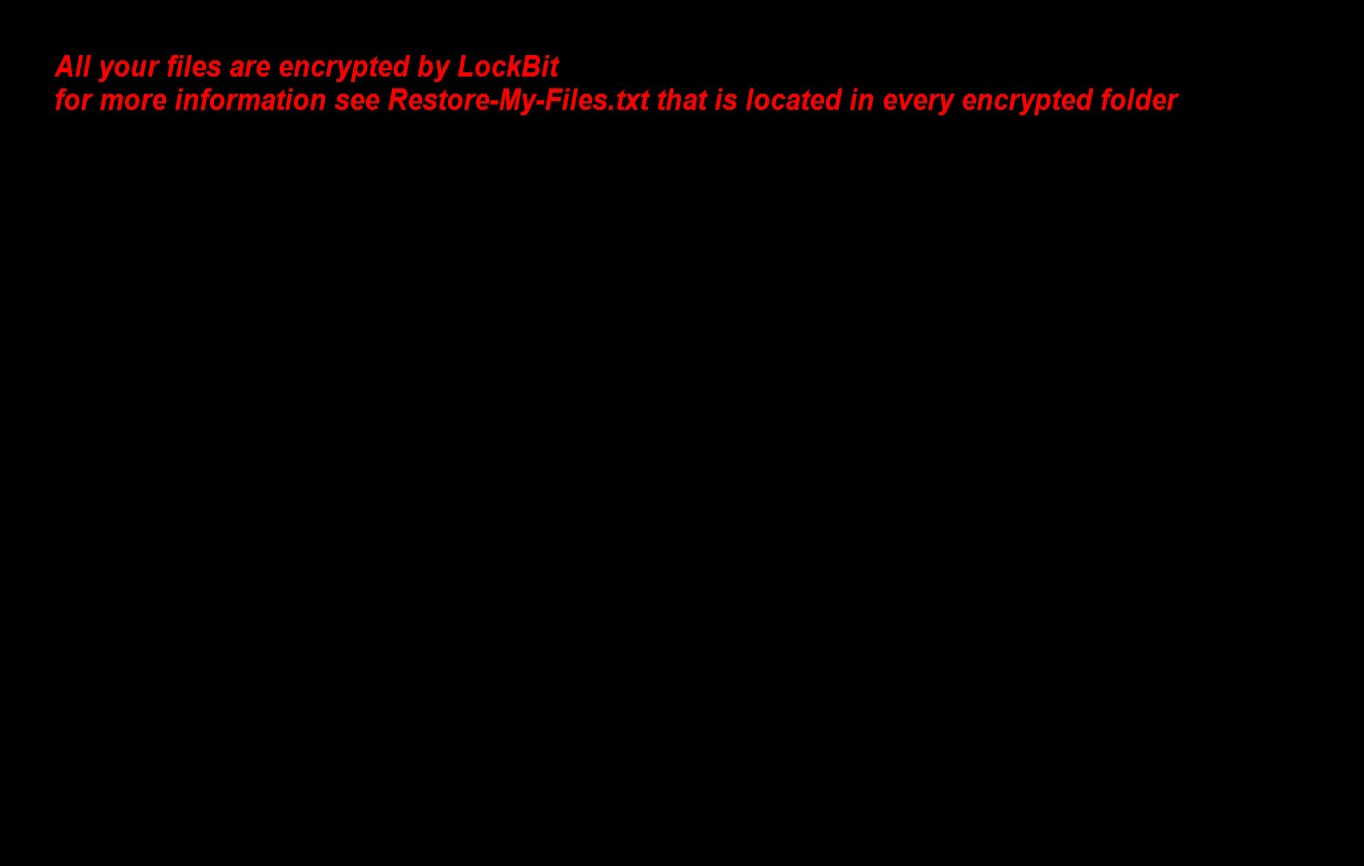
*Appendix Figure: image21.png*



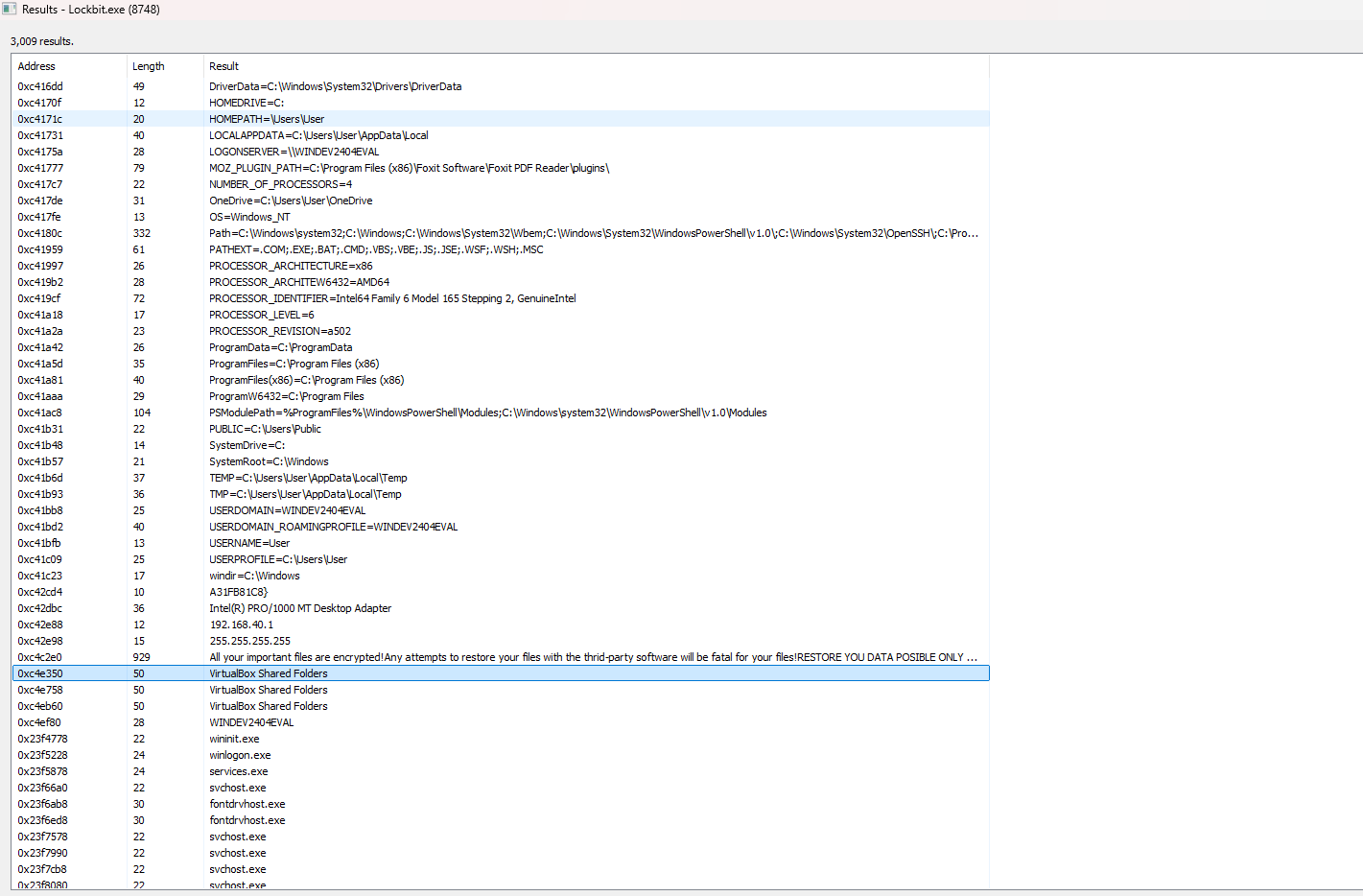
*Appendix Figure: image22.png*



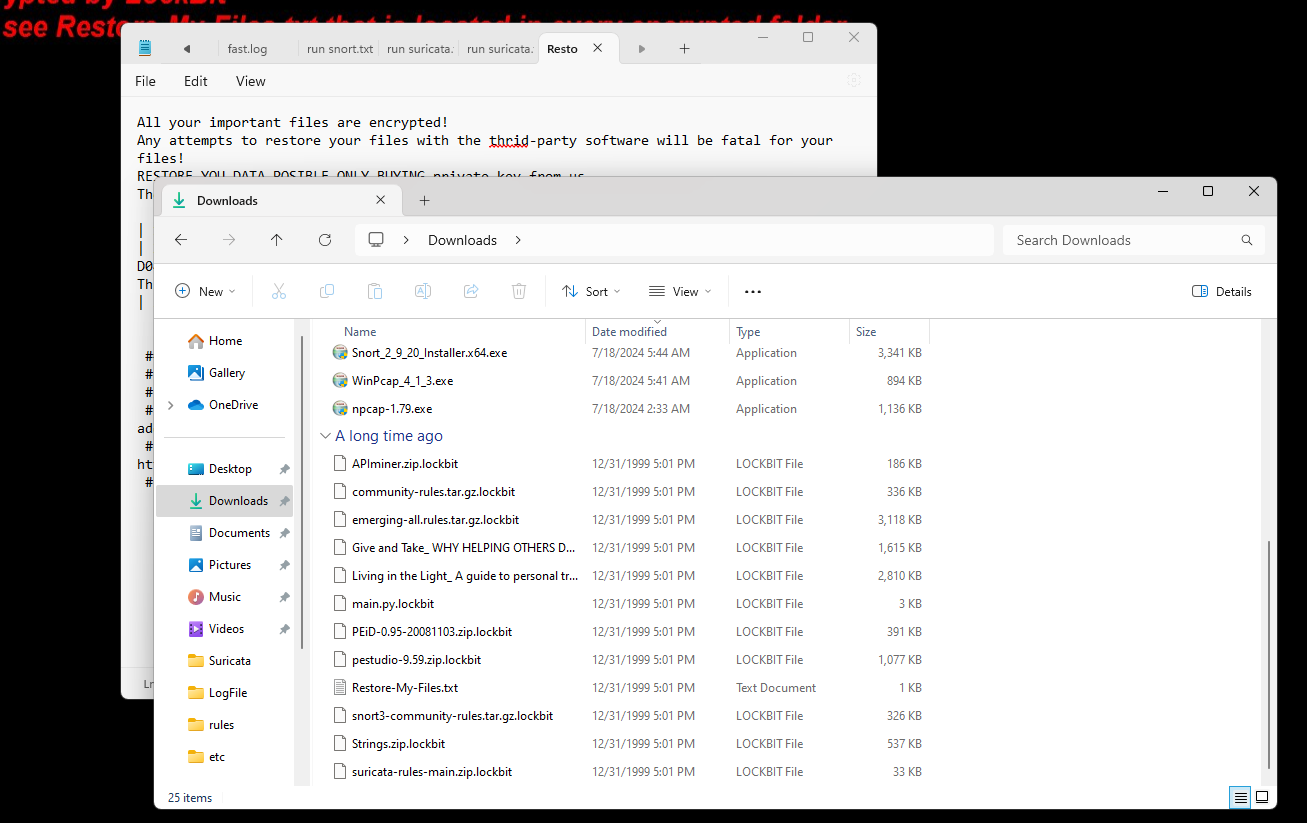
*Appendix Figure: image23.png*



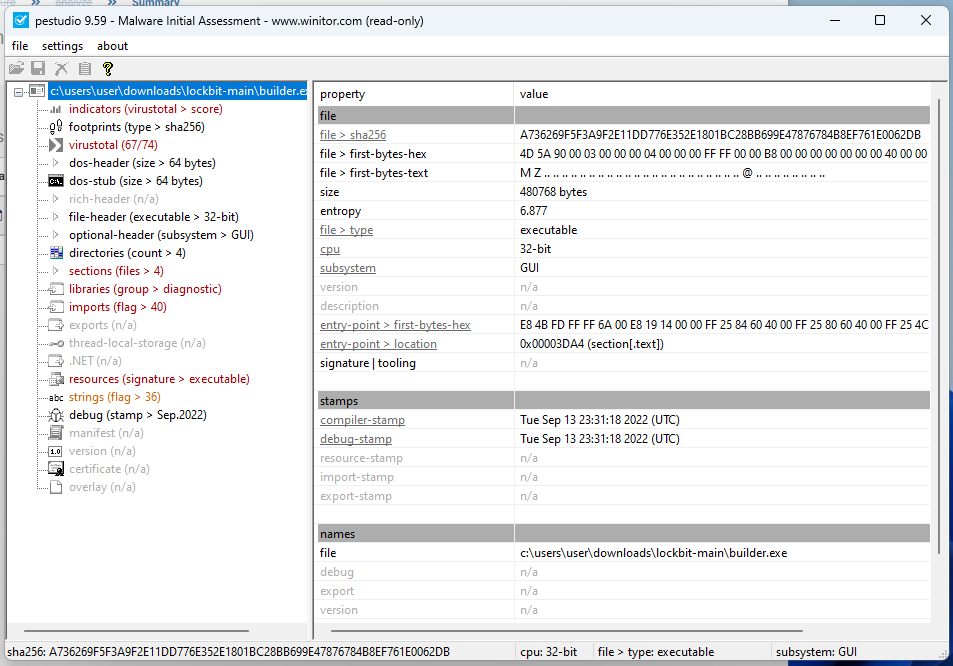
*Appendix Figure: image24.png*



*Appendix Figure: image25.png*



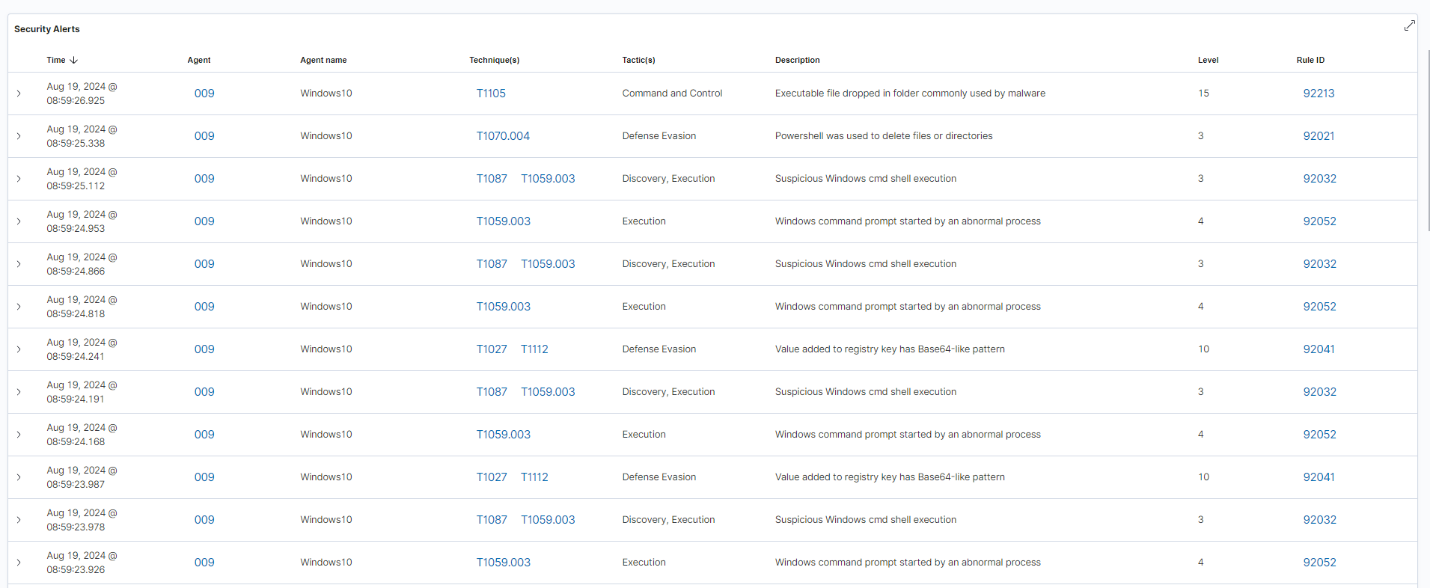
*Appendix Figure: image26.png*



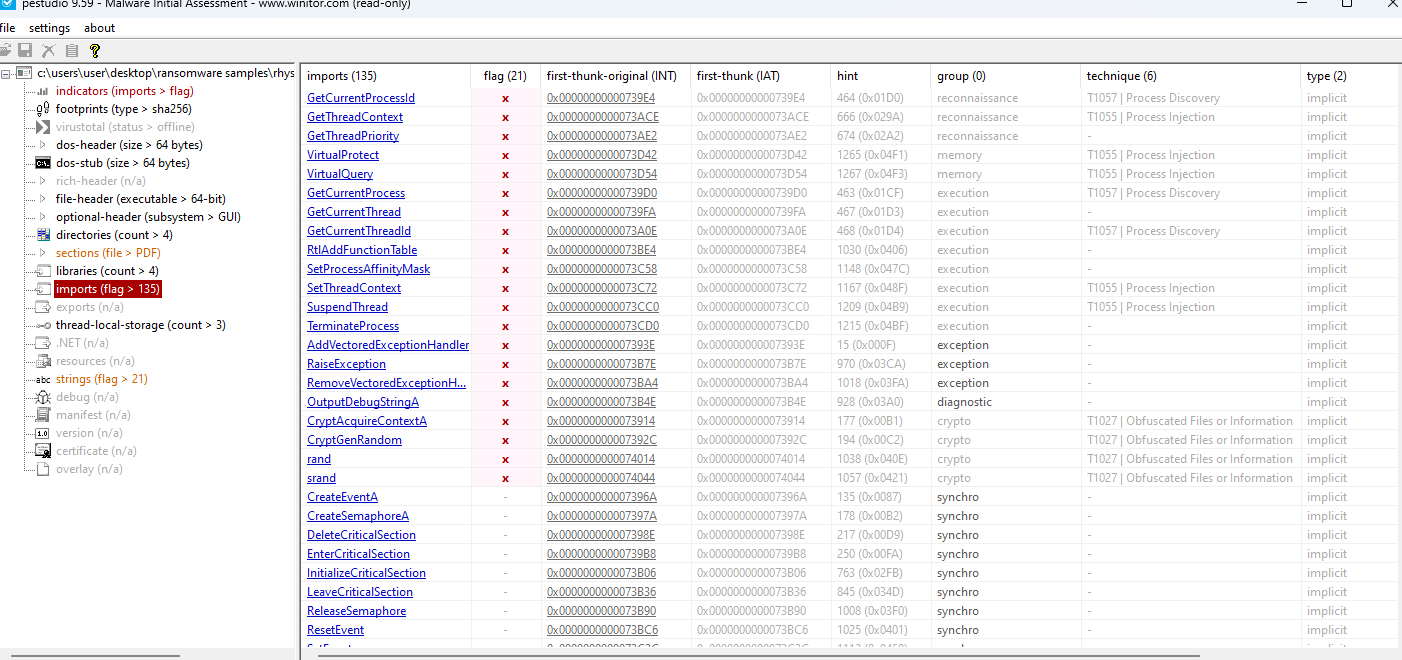
*Appendix Figure: image27.png*



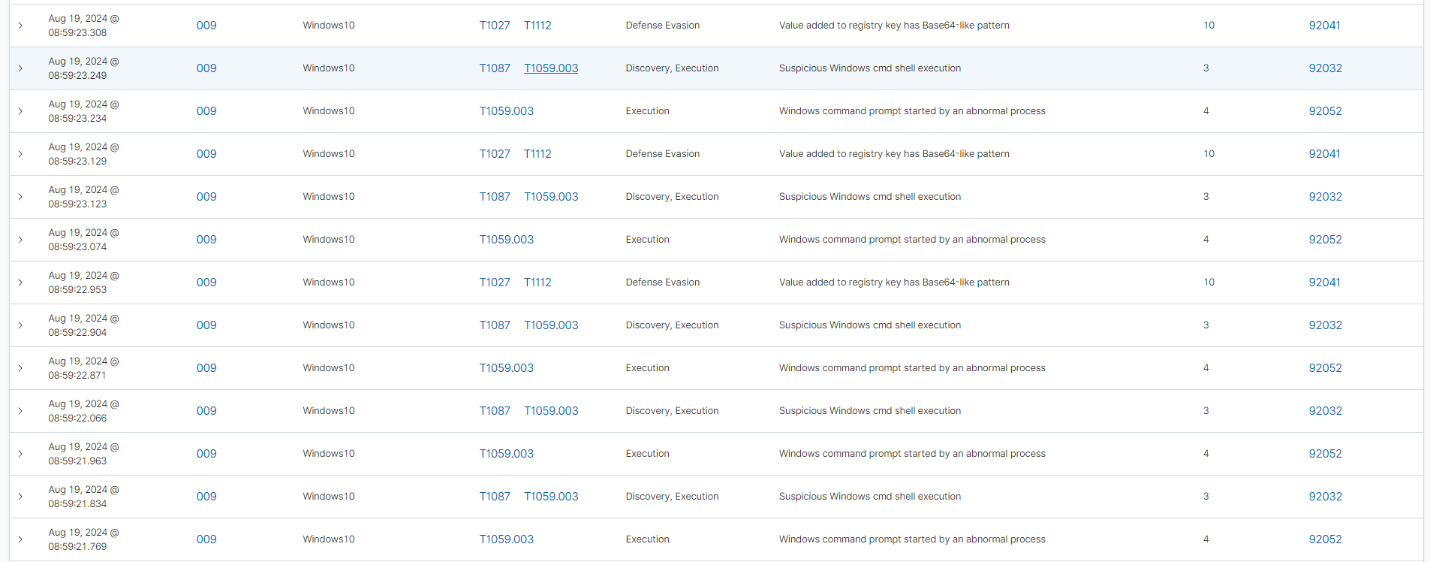
*Appendix Figure: image28.png*



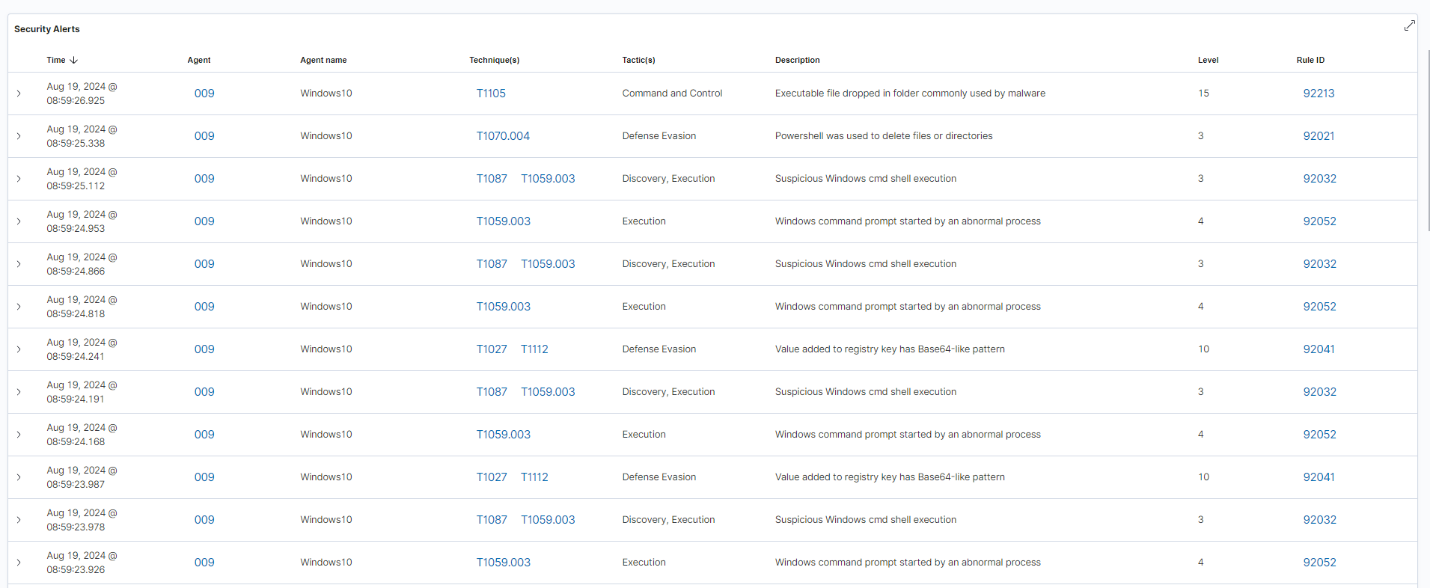
*Appendix Figure: image29.png*



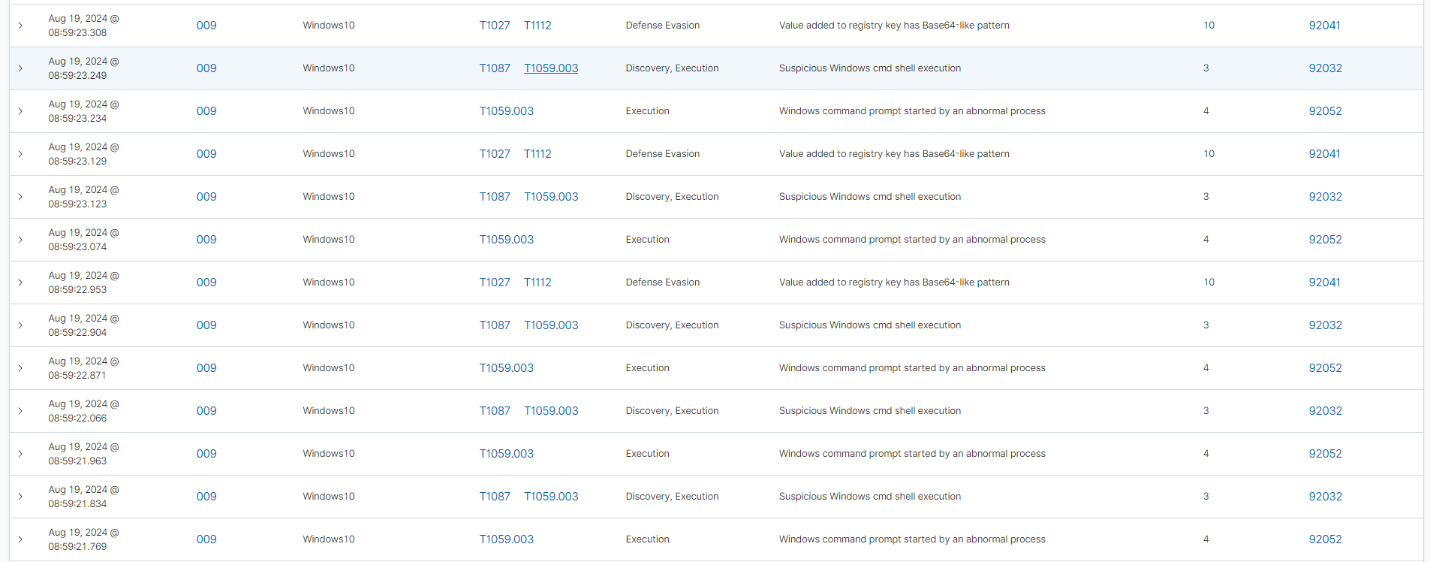
*Appendix Figure: image3.png*



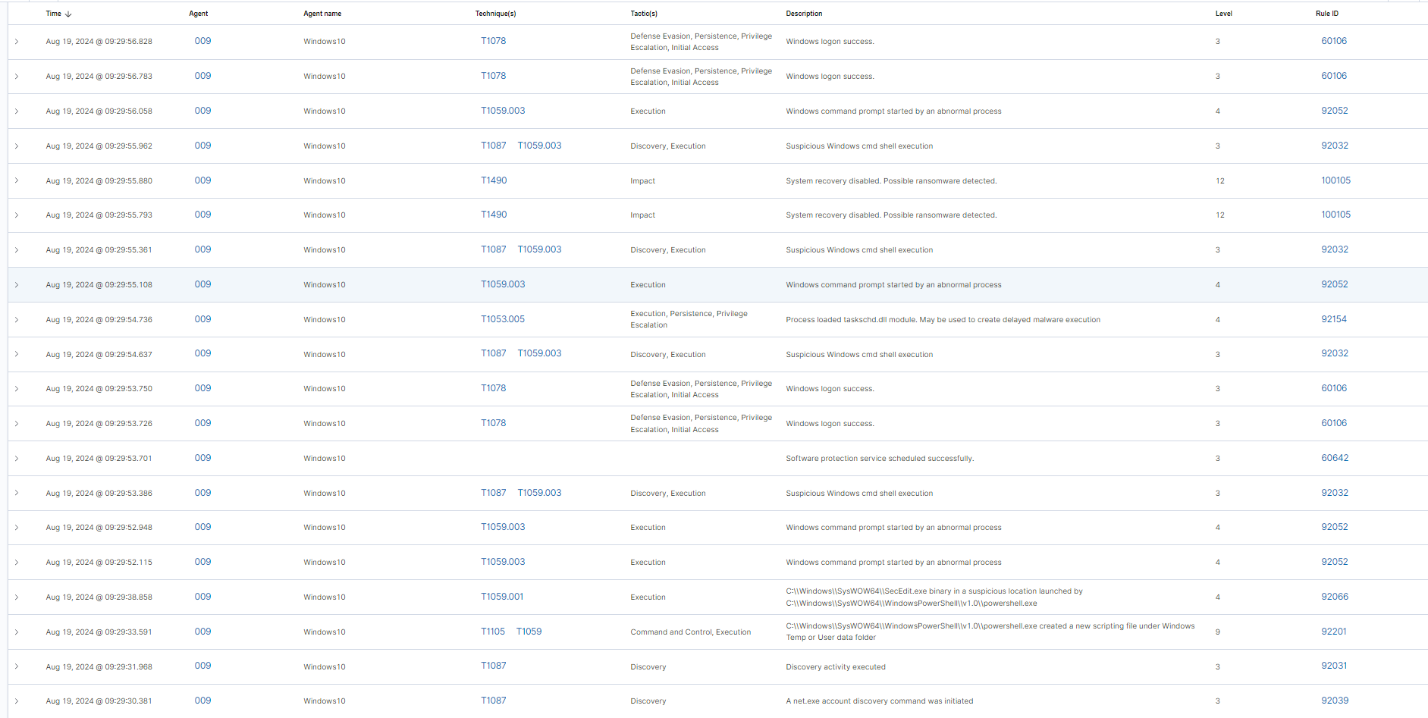
*Appendix Figure: image30.png*



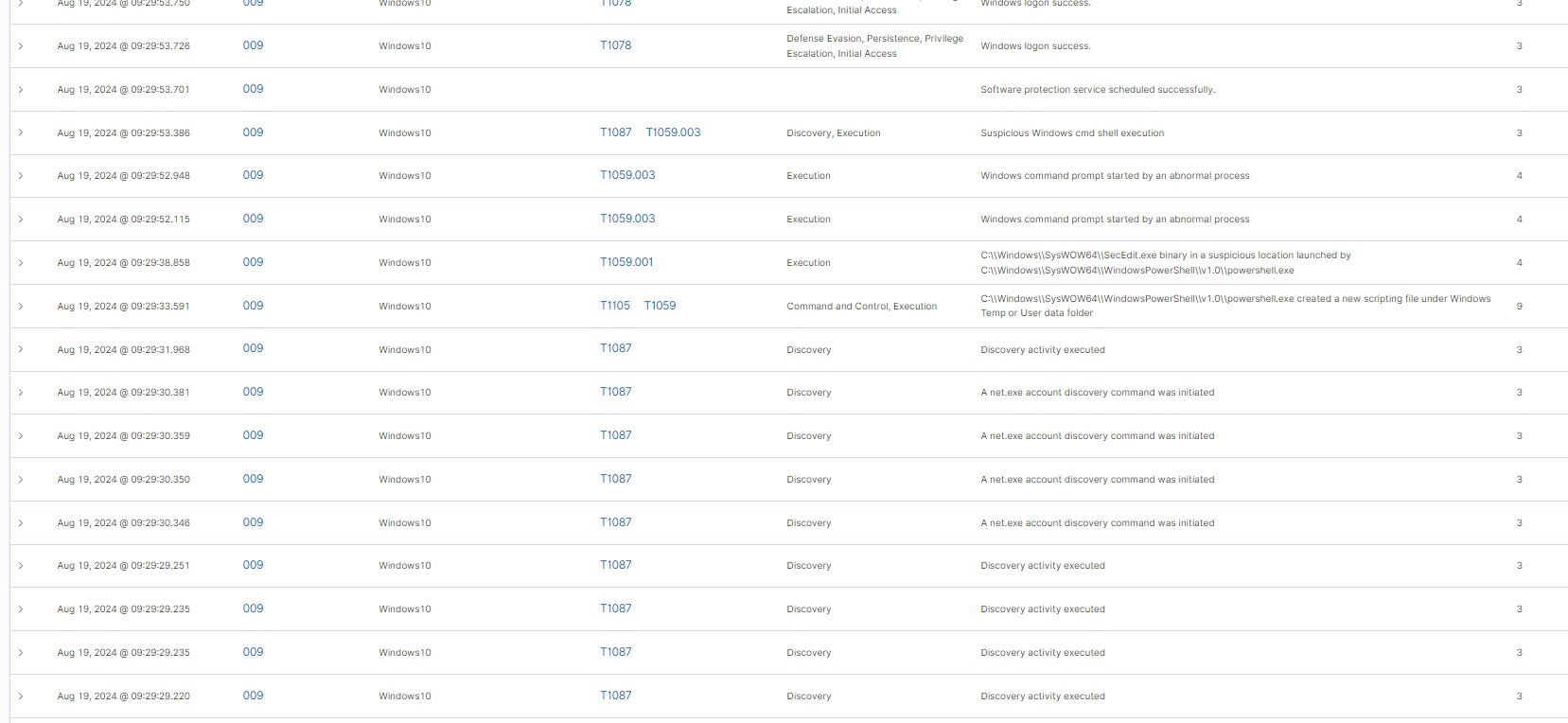
*Appendix Figure: image31.png*



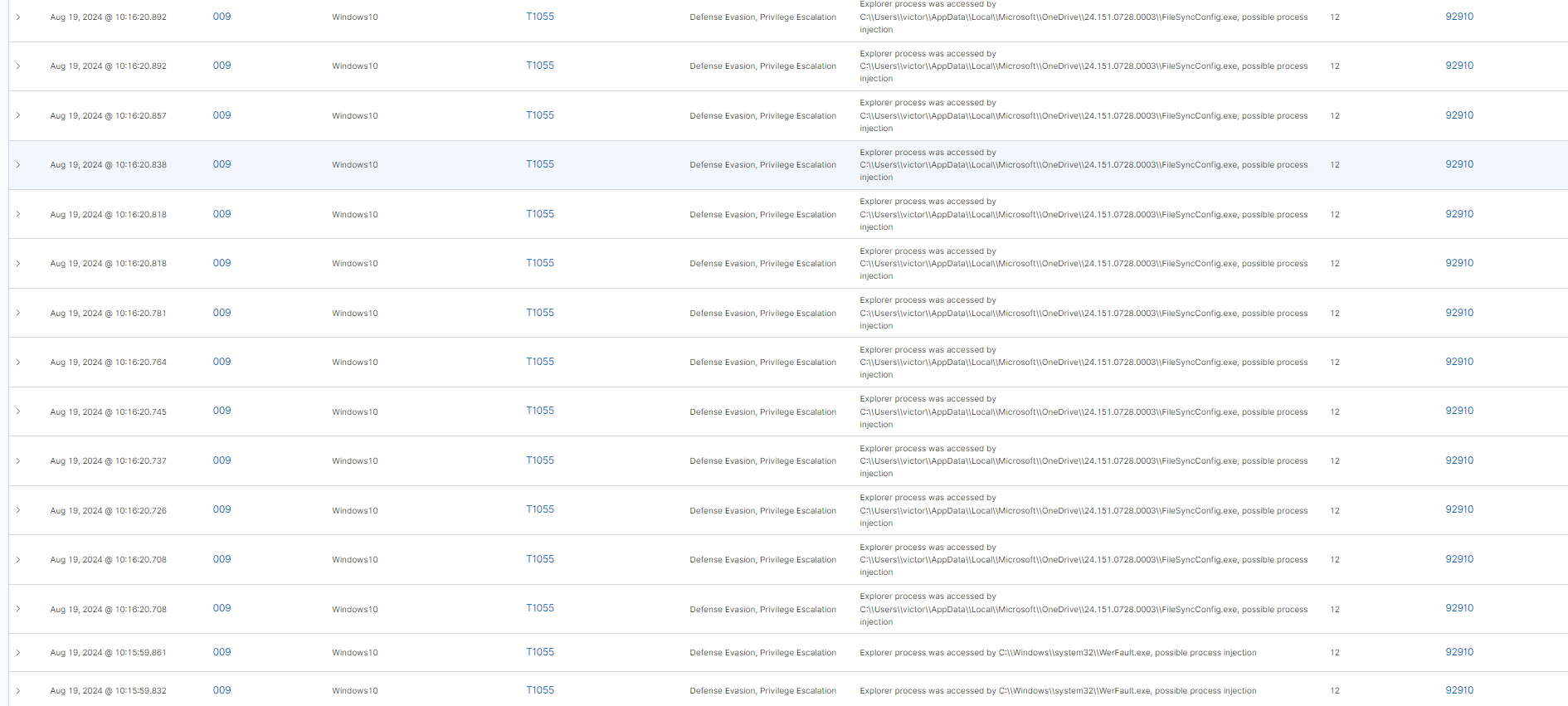
*Appendix Figure: image32.png*



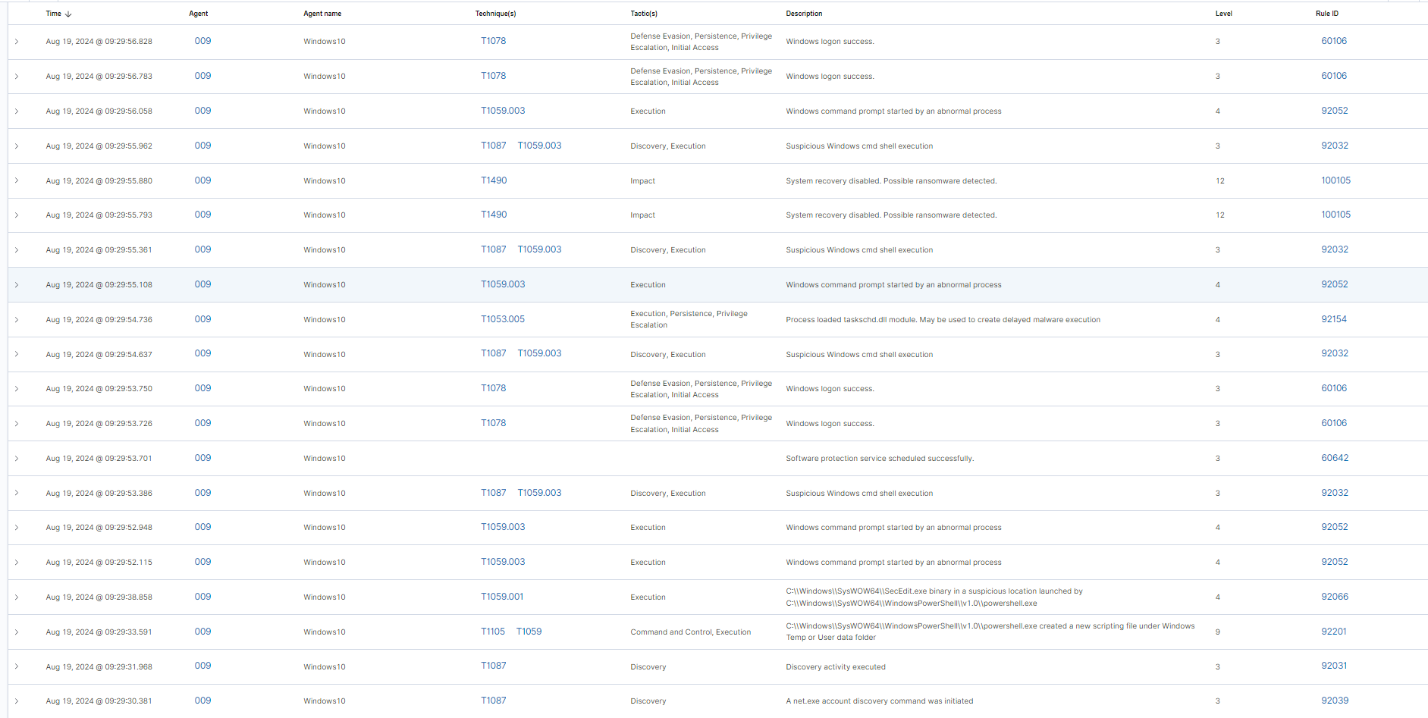
*Appendix Figure: image33.png*



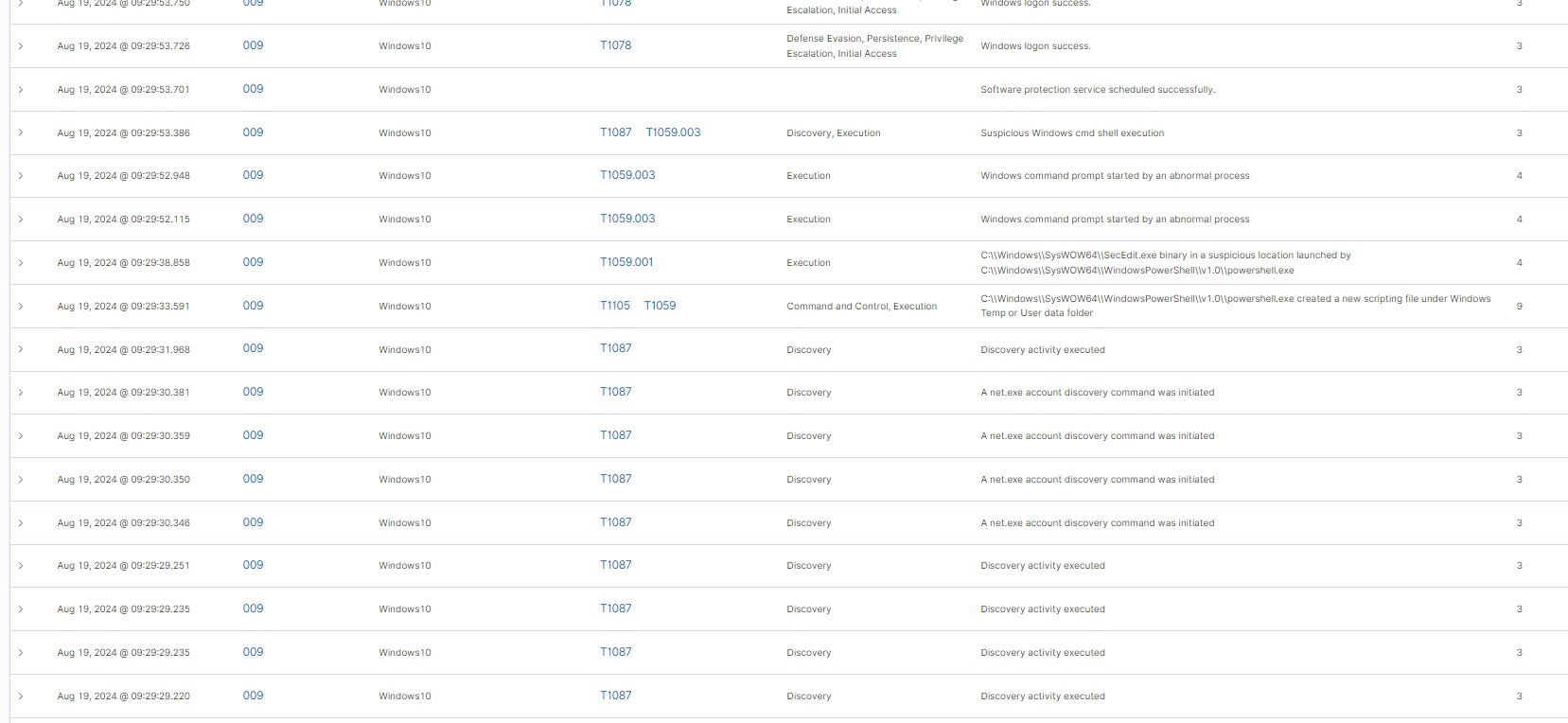
*Appendix Figure: image34.png*



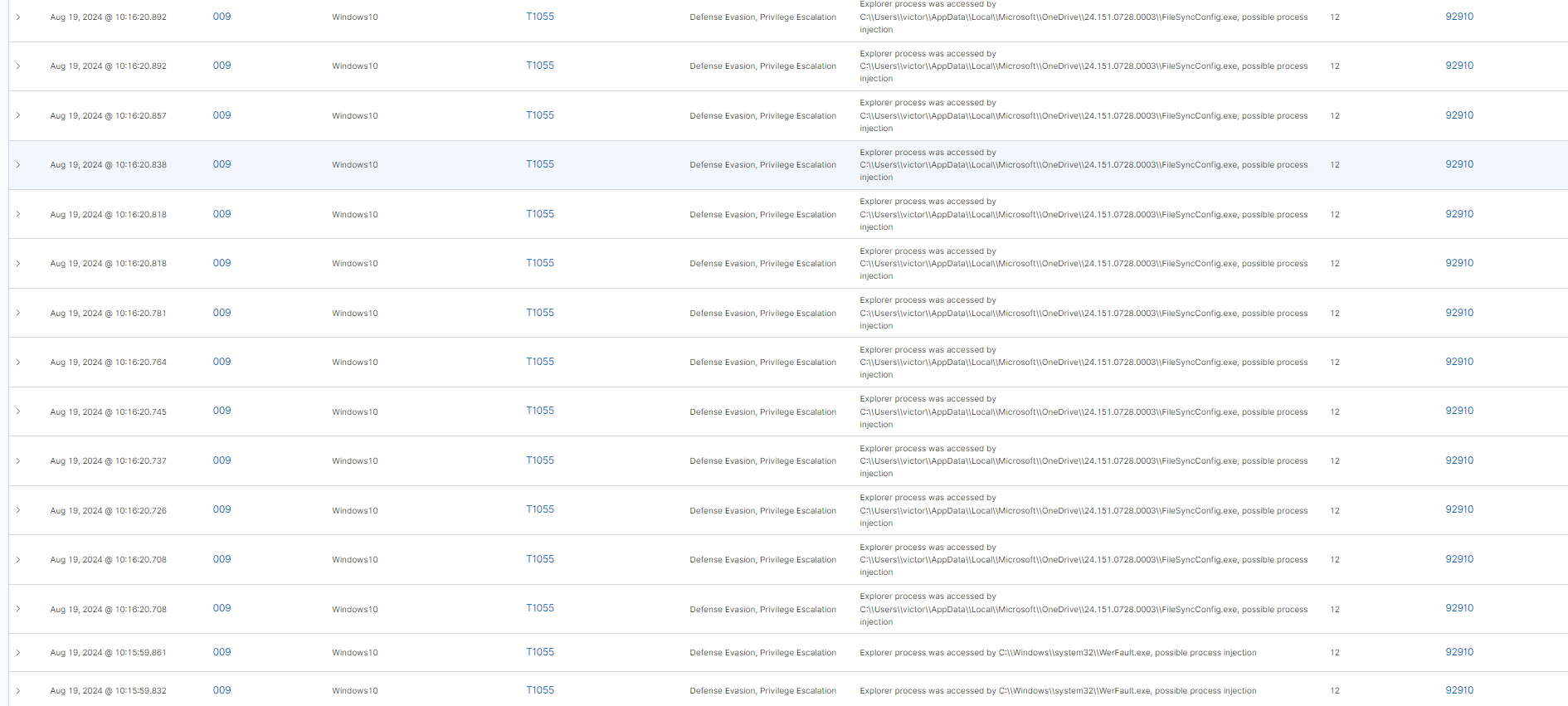
*Appendix Figure: image35.png*



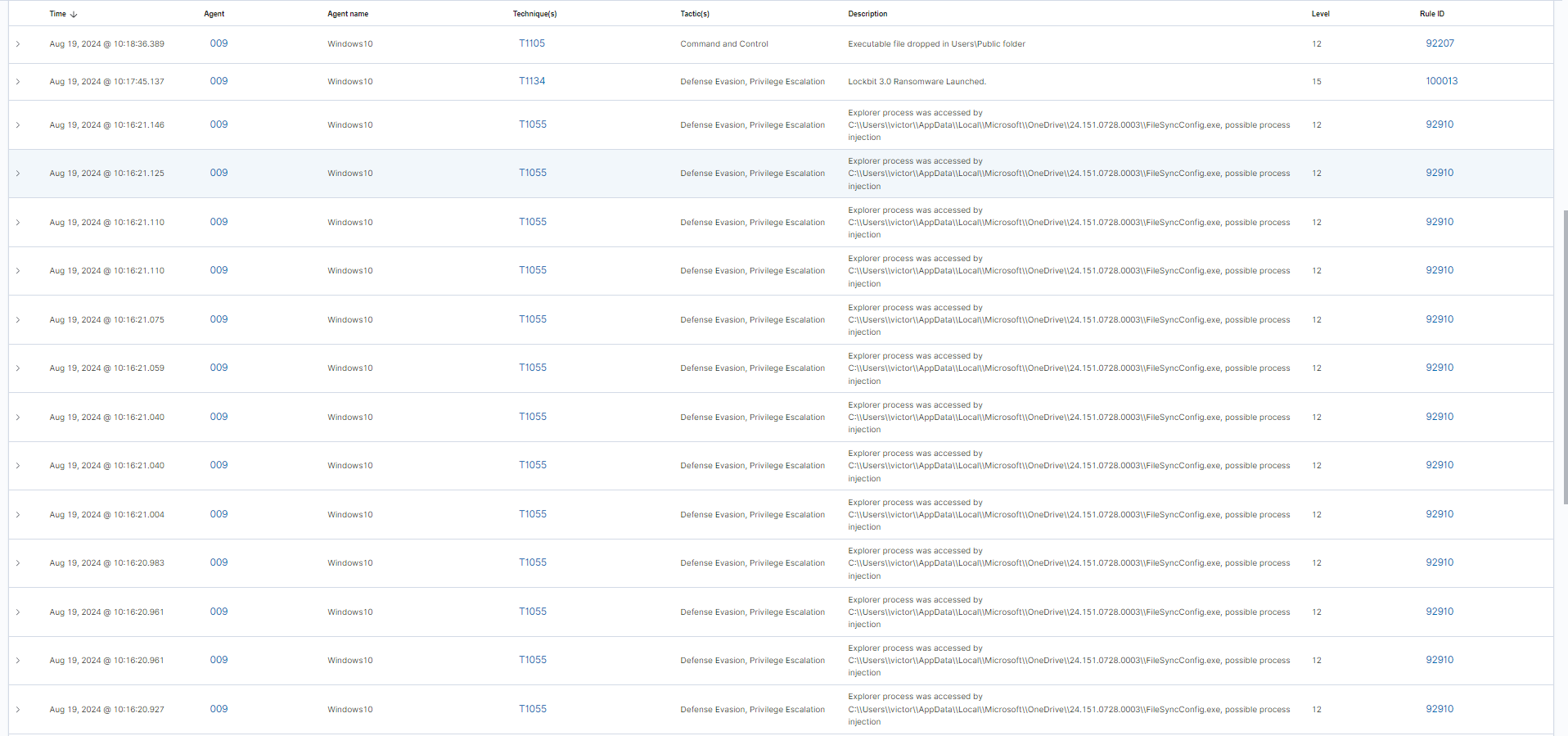
*Appendix Figure: image36.png*



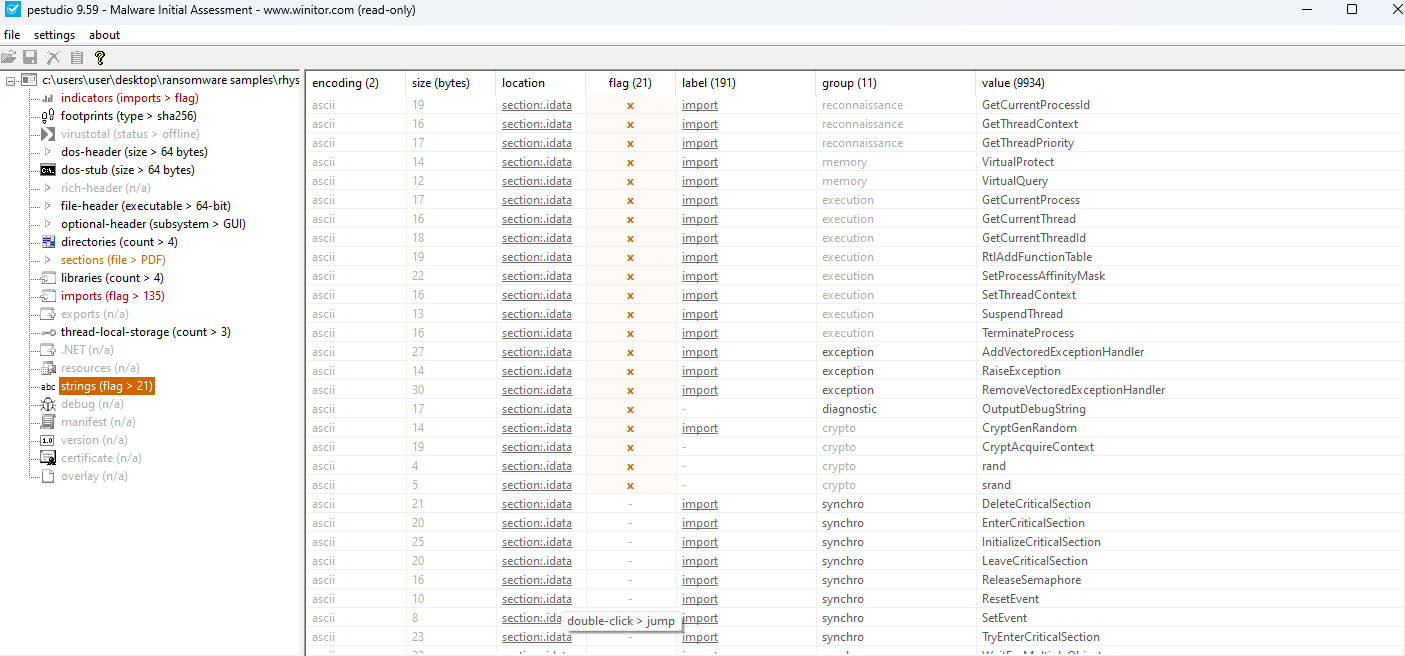
*Appendix Figure: image37.png*



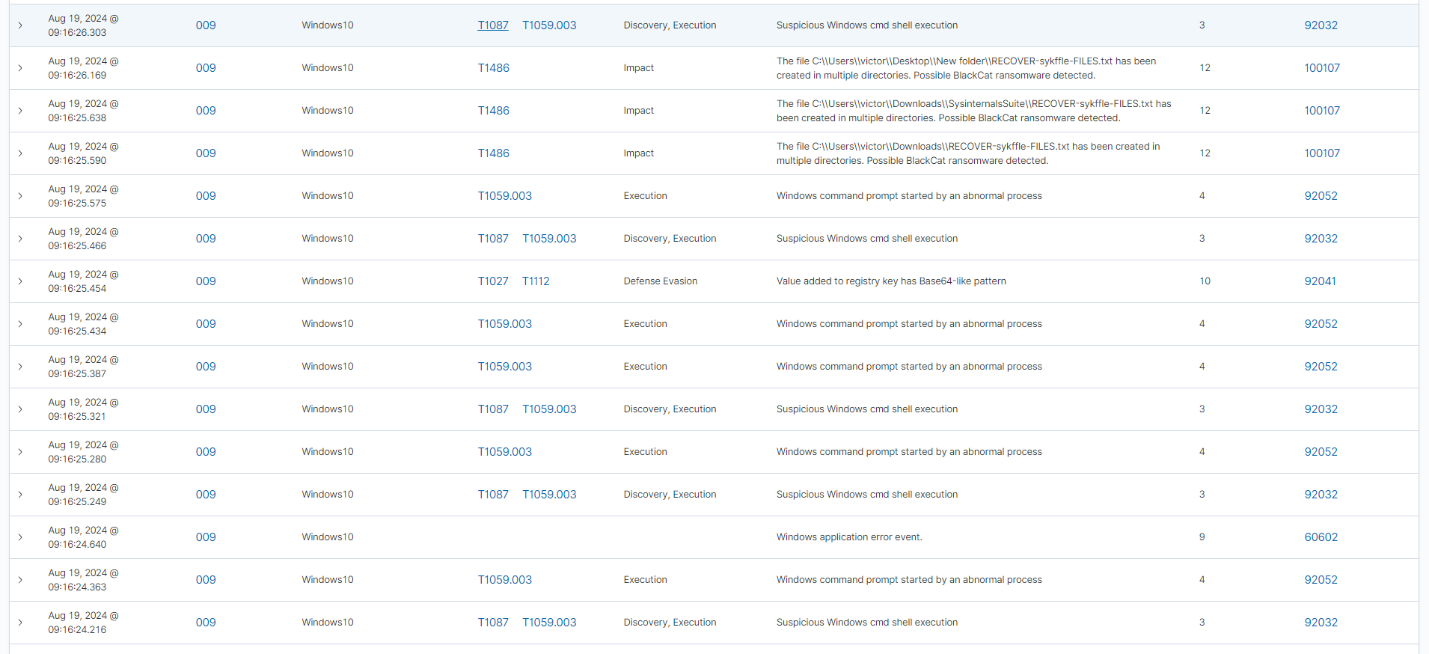
*Appendix Figure: image38.png*



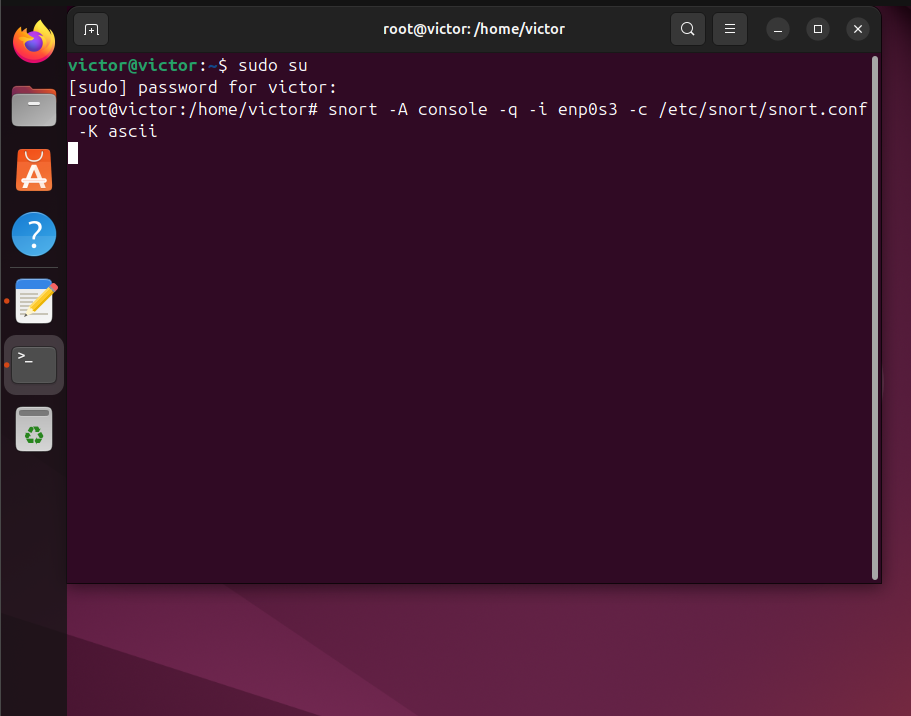
*Appendix Figure: image39.png*



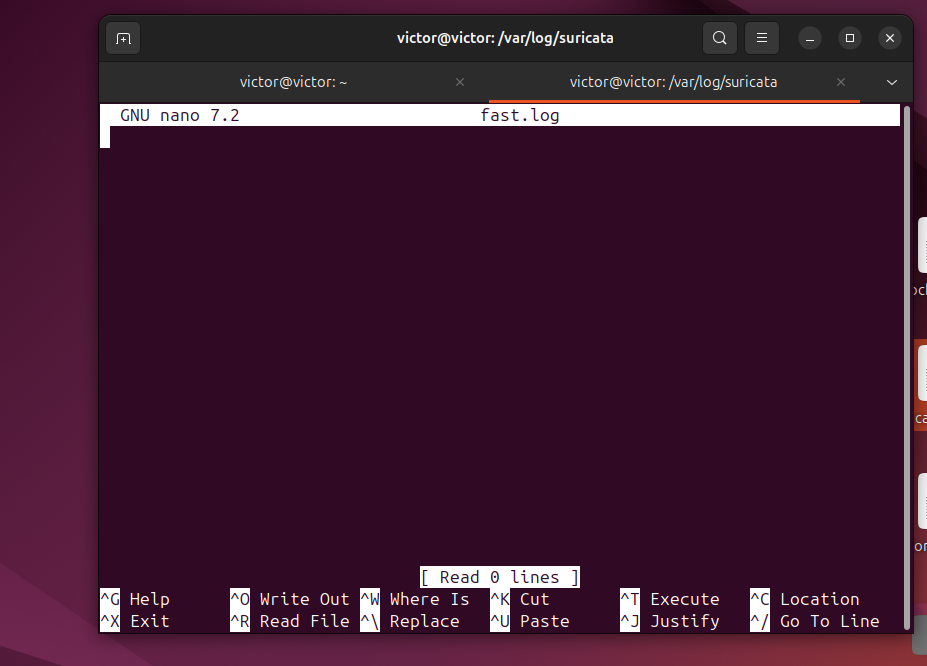
*Appendix Figure: image4.png*



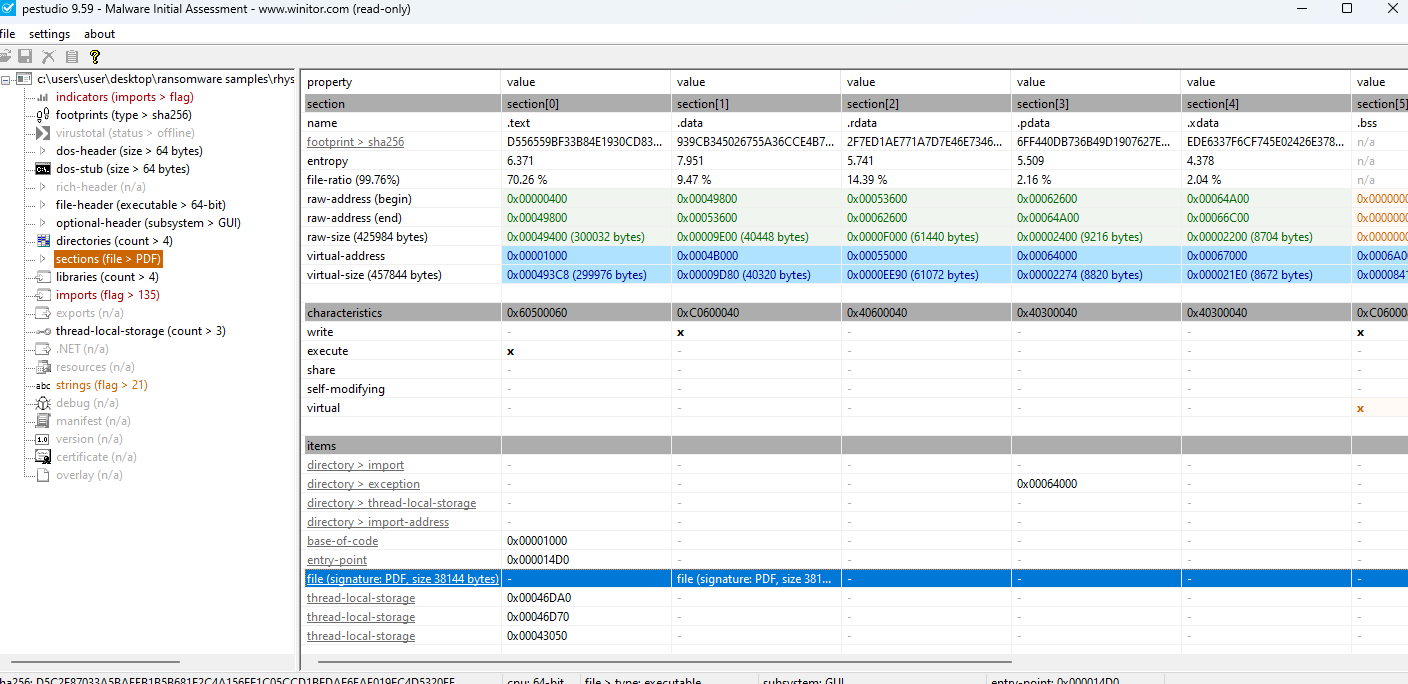
*Appendix Figure: image40.png*



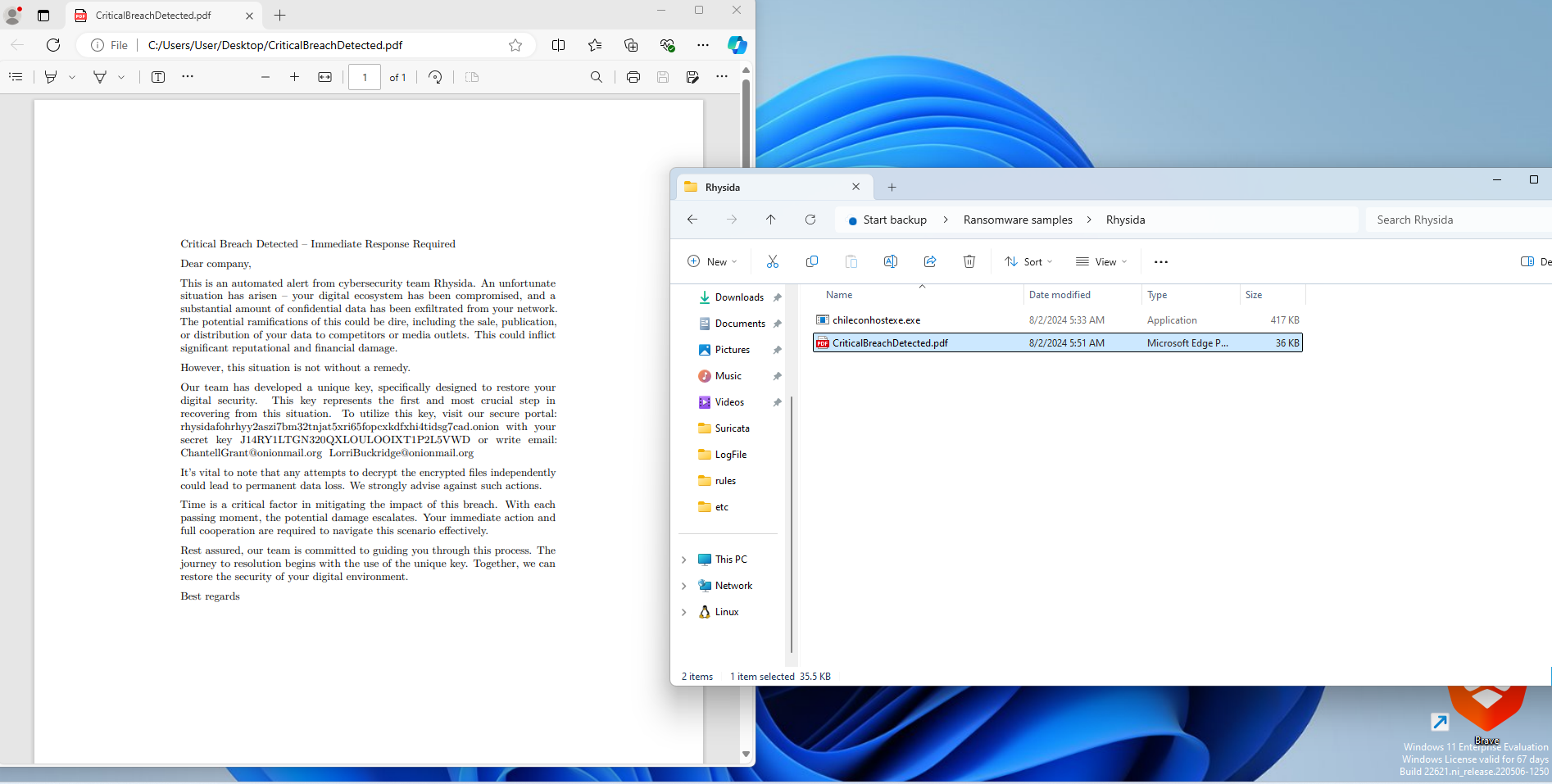
*Appendix Figure: image41.png*



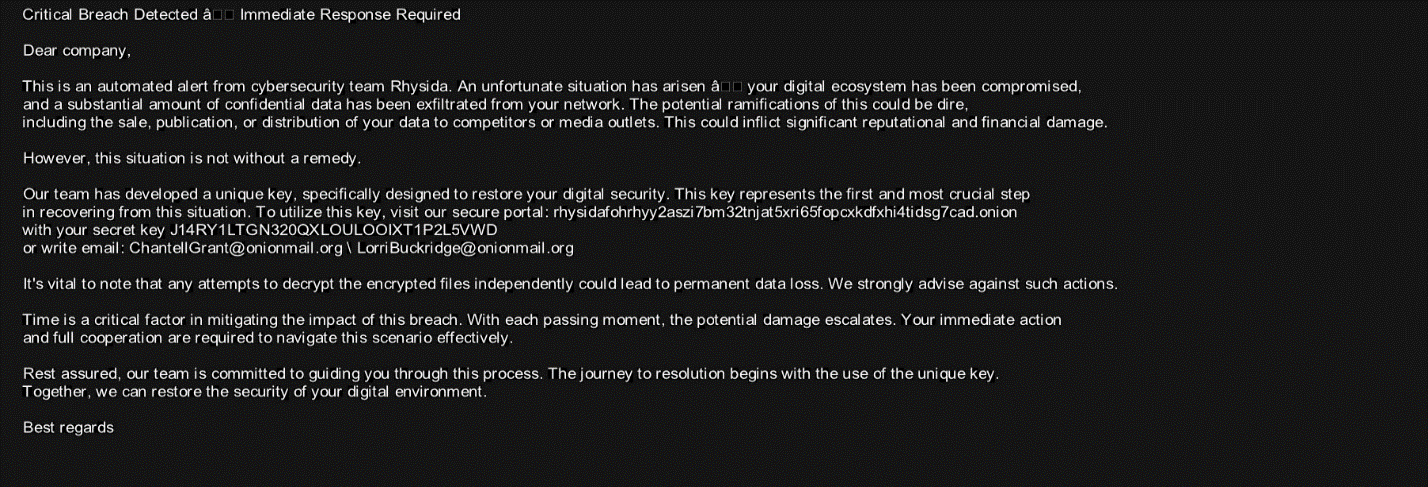
*Appendix Figure: image42.png*



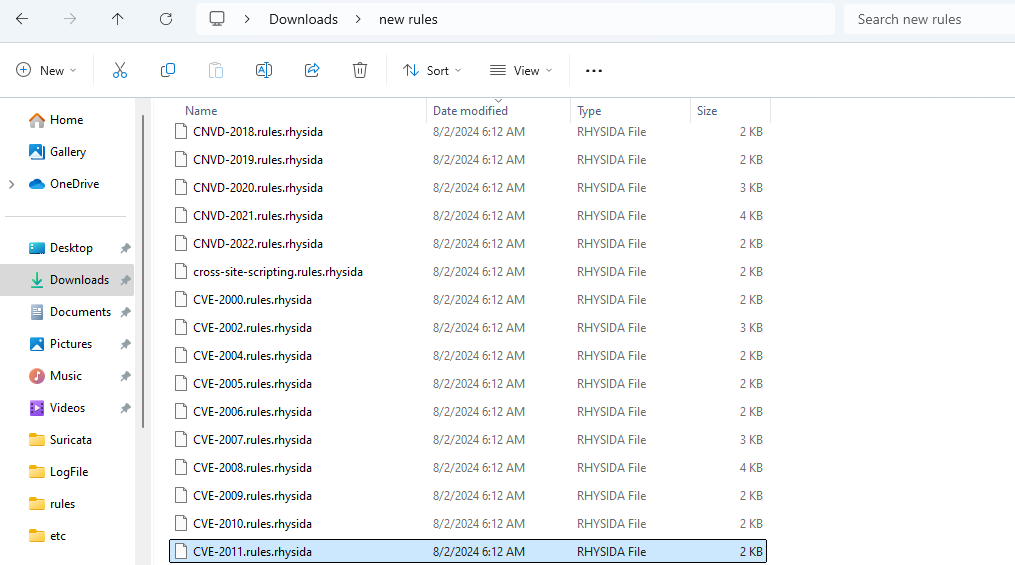
*Appendix Figure: image5.png*



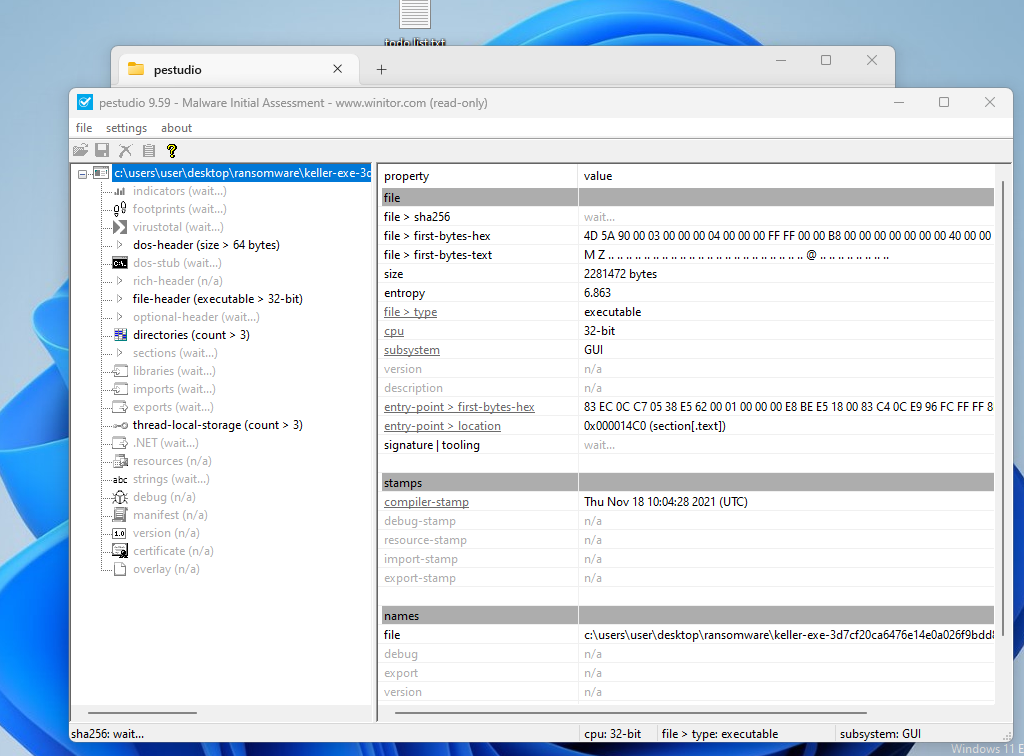
*Appendix Figure: image6.png*



*Appendix Figure: image7.png*



*Appendix Figure: image8.png*



*Appendix Figure: image9.png*