Malware Analysis Report

SHA265: afd631575dfde7ec9da22db771863e810412db7e243be73137f915fc538be3ba

12/31/2024

Executive Summary

This report examines the technical attributes and operational behavior of a sophisticated malware sample**,** **file.exe**, detected in an enterprise environment. The analysis identifies advanced capabilities in system compromise, persistence, and data exfiltration, with notable techniques for sandbox evasion and exploitation of critical Windows subsystems.

Key Findings

Sophistication and Targeted Behavior:

The malware employs sandbox and virtualization evasion techniques (MITRE T1497), making detection and analysis difficult. It demonstrates customization for enterprise systems, evident in its interaction with group policy configurations, registry keys, and task scheduling. Communication is facilitated through covert application-layer protocols, indicating advanced command-and-control (C2) mechanisms.

Exploitation Techniques:

Exploits vulnerabilities in Windows Registry, Task Scheduler, Group Policy Management, and OneDrive integration. Alters registry keys to disrupt backup operations, compromise startup behaviors, and interfere with BitLocker security settings. Uses Themida packer to evade signature-based detection, requiring de-obfuscation for further analysis.

Functional Impacts:

- System Reconnaissance: Collects detailed hardware, software, and network configurations.
- Data Exfiltration: Leverages OneDrive configurations and HTTP requests to specific IPs and domains for covert data transfer.
- Remote Command Execution: Executes attacker-specified commands to extend compromise and deploy additional payloads.
- Persistence: Ensures reactivation through scheduled tasks, registry modifications, and cloud-based persistence mechanisms.

Indicators of Compromise (IOCs):

- Network artifacts include outbound connections to specific IP addresses (e.g., 185.215.113.37) and domains (e.g., go.microsoft.com).
- Registry keys and file system modifications highlight attempts to blend malicious operations with legitimate system processes.
- Packaged with encoded and randomized strings to obfuscate its operations further.

Case Details

Date	**30 December 2024**
Analyst	**Mantone Malikhetla**

Sample information

File name	**file.exe**
File size	**1861632 bytes**
File type	**executable, 32 bit, GUI**
MD5	**88268992617be8a686f04a9c90726839**
SHA1	**fcd4442a98d04fbce0ebcb94523f2d8ca7afaa55**
SHA256	**afd631575dfde7ec9da22db771863e810412db7e243be73137f915fc538be3ba**
Packer / compiler info	**Microsoft Visual C/C++(16.00.30319)[LTCG/C++]**
Compile time	**29/09/2024 18:19:54 UTC**

Case Specific Requirements

- **Why is this sample interesting?**
    a. Evasion Techniques: It employs sandbox evasion and artifact detection methods, indicating sophistication.
    b. Target-Specific Behavior: Initial analysis suggests customization for enterprise systems.
    c. Communication Patterns: Uses covert application-layer protocols for C2 communication, indicating advanced capabilities.
    d. Potential Exploitation of Known or Unknown CVEs: Its behavior suggests targeting specific software vulnerabilities.

Standing Information Requirements

- **Functionality the malware provides the attacker once it installed successfully**
    a. System Reconnaissance: Gathers hardware, software, and network details.
    b. Data Exfiltration: Transfers sensitive files and information to a C2 server.
    c. Remote Command Execution: Executes commands from the attacker to perform actions like file deletion, payload deployment, and further reconnaissance.
    d. Persistence Mechanisms: Ensures it reactivates after reboots through registry modifications or scheduled tasks.

- **Malware affecting multiple organizations and/or indicators of a tailored attack**
    a. The malware queries specific registry keys and configurations commonly associated with enterprise systems.

- **Indicators of compromise (IOCs) associated with this malware**
    a. **Network related (IP Addresses, URLs, email addresses, unique traffic patterns)**
    *IP Addresses***:**
    Outbound Connections to: 185.215.113.37, 192.168.100.255, 13.71.55.58, 2.16.164.49

    Domains:
    Connections to: go.microsoft.com, nexusrules.officeapps.live.com

    *Traffic Patterns***:**
    Requests to:
    http://185.215.113.37/e2b1563c6670f193.php
    http://www.microsoft.com/pkiops/crl/Microsoft%20ECC%20Product%20Root%20Certificate%20Authority%202018.crl

    b. **Running processes / RAM artifacts**
    HKLM\SYSTEM\CurrentControlSet\Control

    c. **Registry keys of interest**
    Keys added:
    HKLM\SOFTWARE\Microsoft\OneDrive\24.186.0915.0004
    HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Group Policy\ServiceInstances\2cd129ab-b794-4cb9-8871-2043d82c17dd

    Keys deleted:
    HKLM\DRIVERS\DriverDatabase\DeviceIds

    d. **File created and/or deleted:**
    File deleted:
    C:\Windows\System32\Tasks\GoogleSystem\GoogleUpdater\GoogleUpdaterTaskSystem130.0.6679.0{26B7157F-9B78-4FFD-90CF-52A5AF0D059D}

    File added:
    C:\Windows\System32\SleepStudy\user-not-present-trace-2024-10-17-01-37-03.etl

- **Mitre Attack Details**
    a. Defense Evasion: Virtualbox/Sandbox Evasion
    b. Discovery: Query Registration, System Information Discovery
    c. Command & Control: Application Layer Protocol

- **Application, service or other vulnerability this malware exploits**
    a. Windows Registry (Core System Configuration)
    Targeted Keys:
    HKLM\SYSTEM\CurrentControlSet\Control\Session Manager
    HKLM\SYSTEM\ControlSet001\Control\BackupRestore
    HKLM\SYSTEM\ControlSet001\Control\BitLocker

    The malware manipulates registry keys to modify system startup behavior, disrupt backup operations, and interfere with security features like BitLocker.
    Alters Session Manager settings to potentially execute payloads or modify system-level operations during boot.

    b. Windows Task Scheduler
    Key Modified:
    HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Schedule\TaskCache\Plain\{1257299A-9F8C-4823-A354-D27D0C680DD4}

    Creates scheduled tasks for persistence, ensuring malware execution during system startup or at specific intervals.
    Exploits the Task Scheduler service to blend malicious activities with legitimate system functions.

    c. Group Policy Management
    Key Modified:
    HKLM\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentVersion\Group Policy\ServiceInstances

    Manipulates group policy settings to override administrative controls, enforce malicious policies, or introduce exceptions that facilitate malware activities.
    Targets environments reliant on centralized policy management, such as enterprise or healthcare systems.

    d. OneDrive Integration
    Key Added:
    HKLM\SOFTWARE\Microsoft\OneDrive\24.186.0915.0004

    Leverages OneDrive configurations, possibly for data exfiltration or persistence under the guise of legitimate cloud operations.

    e. Driver Database and Device Management
    Key Deleted:
    HKLM\SYSTEM\DriverDatabase\DriverPackages\netevbda.inf_amd64_1503f4d5a0d6ba56\Configurations\bcm57810NP_152d_89ab_amd64wlh\Device\Interrupt Management
    HKLM\DRIVERS\DriverDatabase\DeviceIds*AEI0276

Modifies or removes driver configurations to disrupt diagnostics or analysis. This could limit visibility into network activity or disable device functionalities that could interfere with malware operations.

Static Analysis

Detect It Easy data below shows malware has properties tailored for the Windows platform

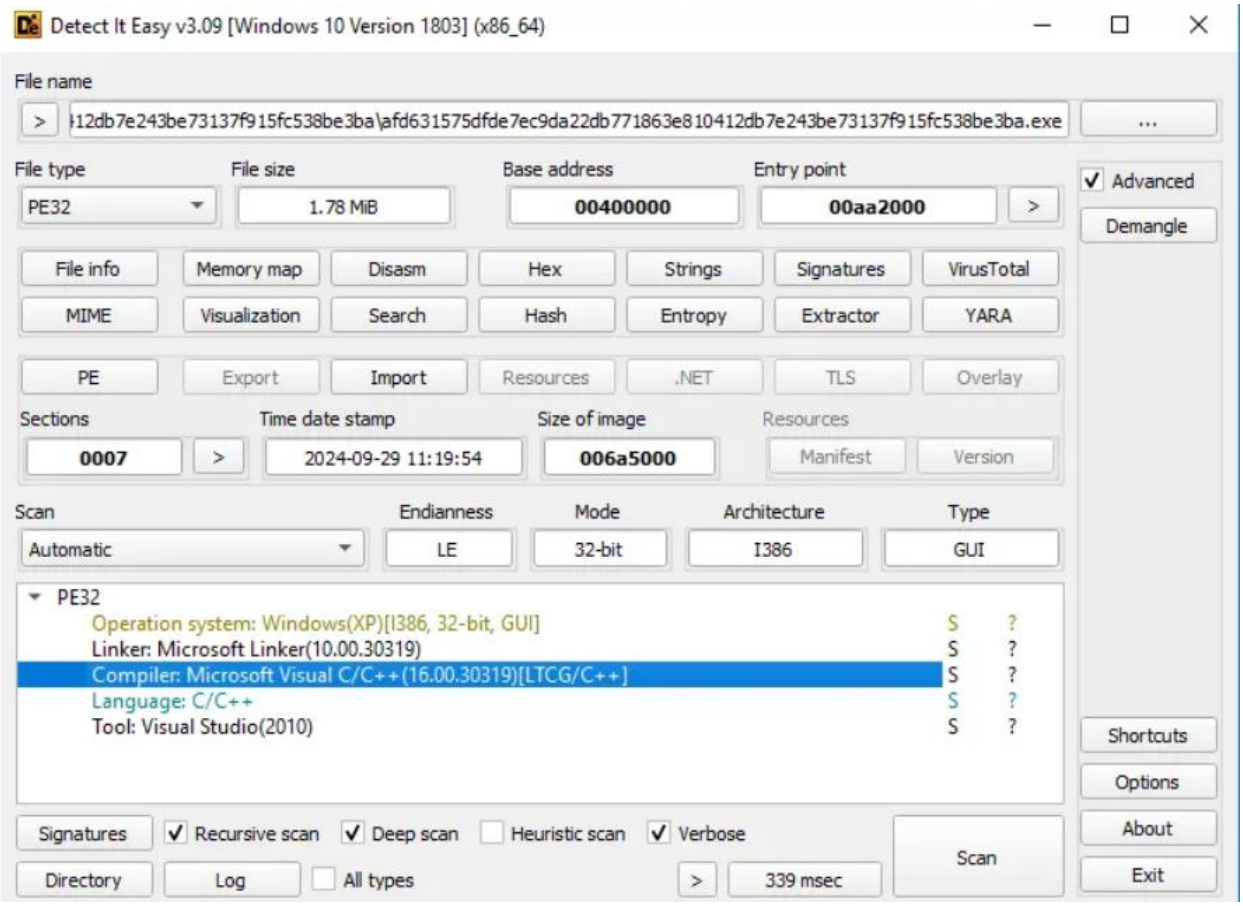


Figure 1 - Detect It Easy: Malware Metadata

Exeinfo PE shows sample is packed with Themida Packer. The use of packing signifies the malware's intent to evade signature-based detection, requiring advanced unpacking techniques for further analysis. **This aligns with T1497 (Virtualization/Sandbox Evasion)**

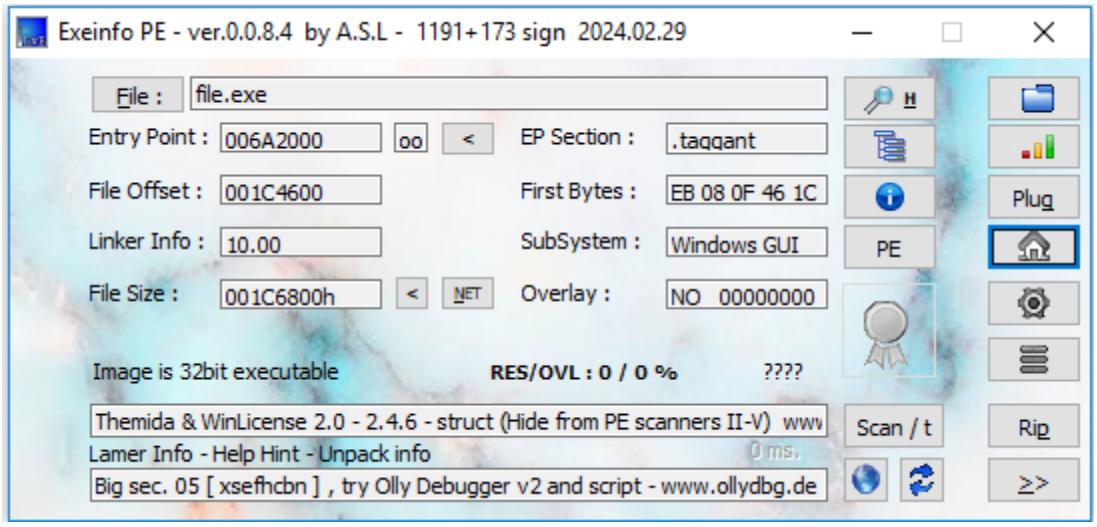


Figure 2 - PE Info: Unpacking and De-obfuscation

FLOSS Strings Analysis shows presence of encoded or encrypted Content, randomized strings like G\v, wIV, and mN. suggest encryption or packing.

Figure 3 - FLOSS Strings Analysis

Die PE Entropy shows that the sample is packed because of the high entropy above 7, Section 0 and Section 4 are packed.

Entropy analysis provides insights into the structure of the executable.

- High entropy in specific sections confirms the presence of packed or encrypted data.
- The packer compresses or encrypts the executable's code and data sections. The most common sections targeted for packing include:
  .text Contains the executable code (instructions).
  .rdata: Read-only data such as strings and constants.

Suspicious section names such as xdmmvtdk, xsefhcbn, and .taggant are used by the malware for obfuscation.

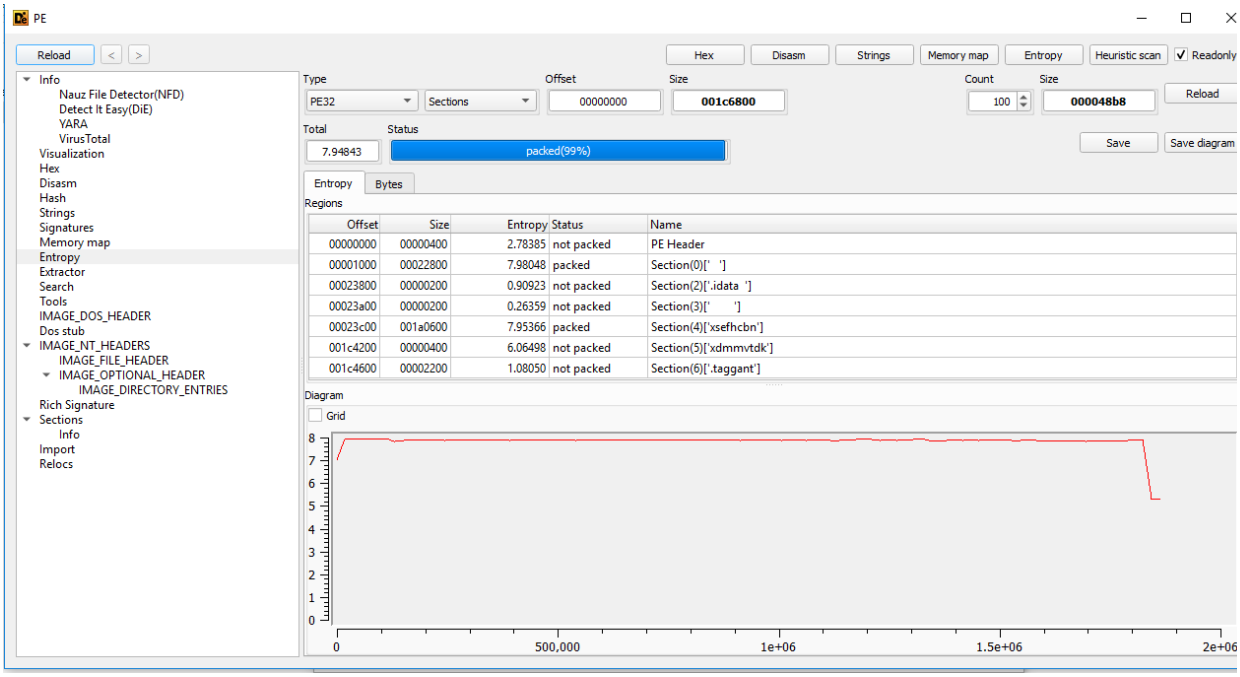


Figure 4 - Die PE: Entropy

CFF Explorer shows that sample has one import, the kernel32.dll

- Provides core Windows API functions related to memory management, input/output operations, process and thread creation, and synchronization.
- Malware often hooks or replaces functions in this DLL to perform actions like hiding files, processes, or creating new processes stealthily.

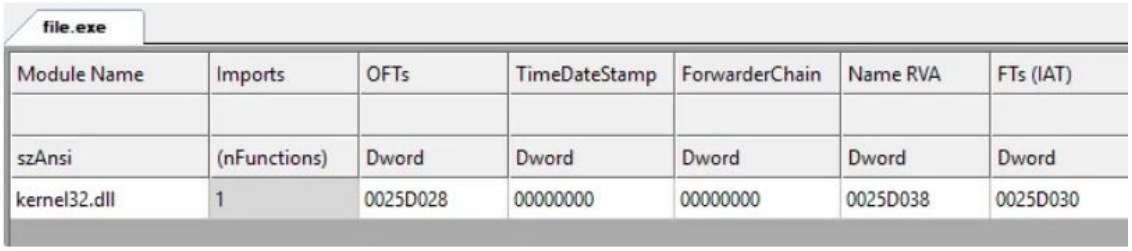


Figure 5 - CFF Explorer: Imports

Unpacking with Magicmida, shows unpacked fileU.exe size of 4GB in comparison to packed file.exe of 1MB

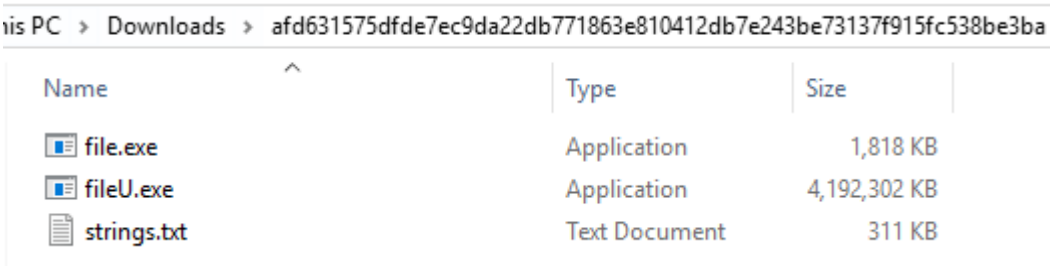


Figure 6 - Packed (file.exe) and Unpacked (fileU.exe) samples

Unpacked DIE PE entropy show low entropy values and additional sections which were not visible in the packed sample.

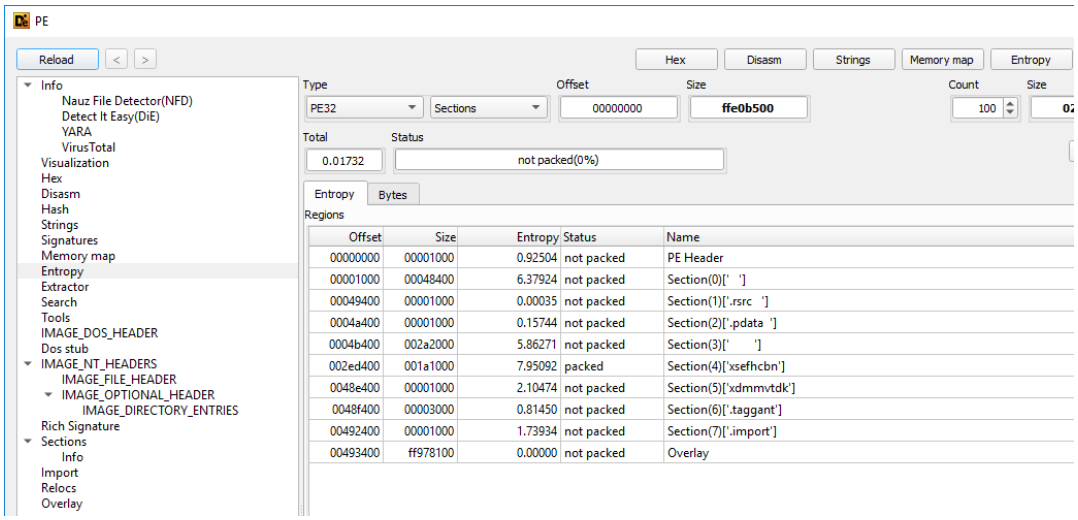
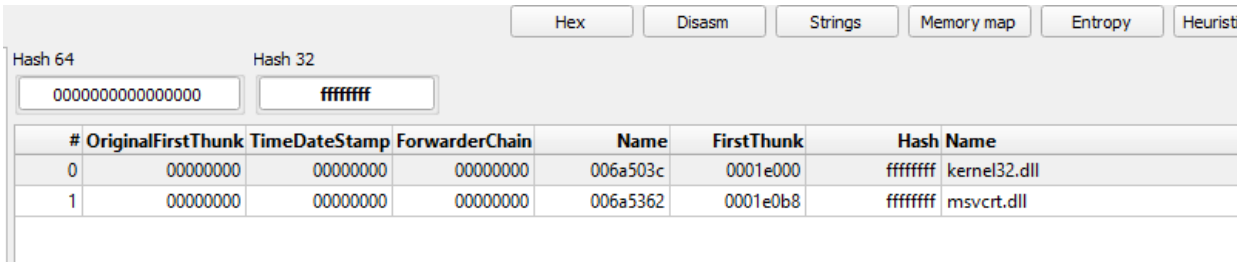


Figure 7 - Die PE: Entropy of Unpacked sample

An additional imports msvcrt.dll is available after unpacking

- Contains functions from the Microsoft C Runtime library, including memory allocation, string handling, and input/output processing.
- Malware can exploit vulnerabilities in these functions to gain control over execution flow.



#	OriginalFirstThunk	TimeDateStamp	ForwarderChain	Name	FirstThunk	Hash	Name
0	00000000	00000000	00000000	006a503c	0001e000	ffffffff	kernel32.dll
1	00000000	00000000	00000000	006a5362	0001e0b8	ffffffff	msvcrt.dll

Figure 8 - CFF Explorer: Imports unpacked sample

Dynamic Analysis

This figure compares pre- and post-infection registry states. Registry activities align with behaviours observed in advanced, targeted malware campaigns, indicating a focus on persistence, stealth, and disrupting defensive mechanisms.

- Persistence through HKLM\Software\Microsoft\Windows\CurrentVersion\Run.
- Adds keys for scheduling tasks (e.g., GoogleUpdaterTaskSystem130.0.6679.0).
- Deletes keys related to legitimate services, potentially to disrupt detection mechanisms. Removes traces of specific hardware configurations, particularly those tied to analysis environments or sandbox systems.

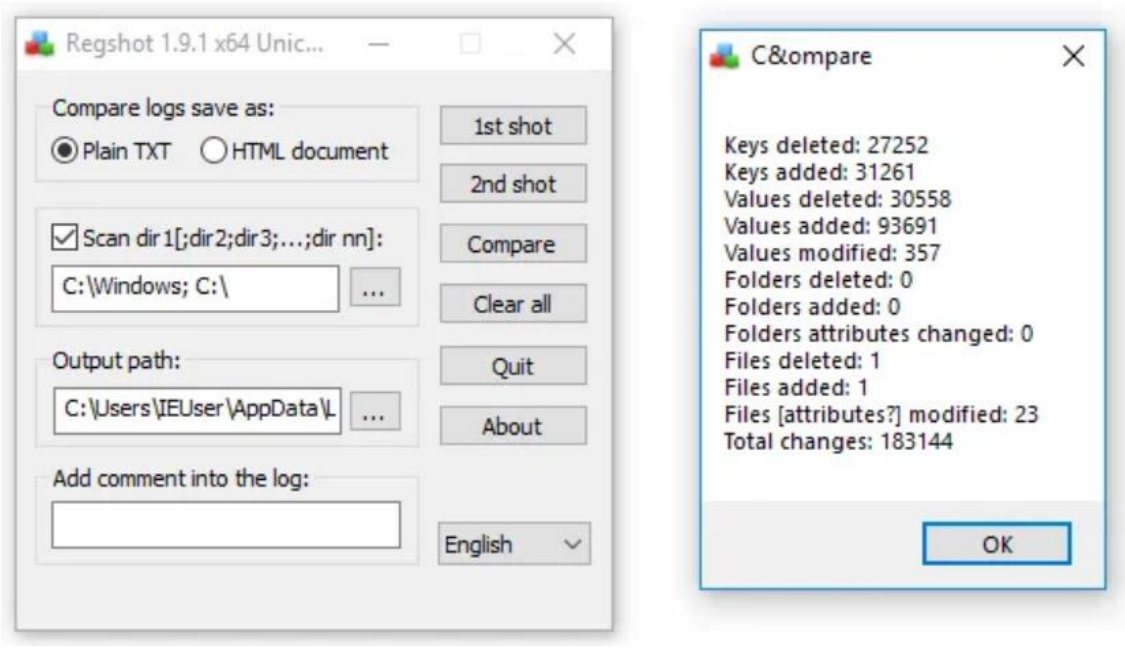


Figure 9 - Regshot: Registry Changes

This figure shows processes initiated by the malware. Runtime behavior confirms the use of legitimate processes for camouflage and execution of additional payloads.

Details:

- Accesses the HKLM\System\CurrentControlSet\Control\SafeBoot\Option registry key to Windows Safe Boot settings, which control how the system behaves when started in Safe Mode potentially modify Safe Boot settings to disable the feature, hindering recovery efforts by preventing booting into Safe Mode.
- Identification of certain user profiles - HKLM\System\CurrentControlSet\hivelist\\\Registry\User\ - it may tailor its actions based on the detected environment**.**
- Ensures its actions occur at a low level, making them harder to detect or reverse.

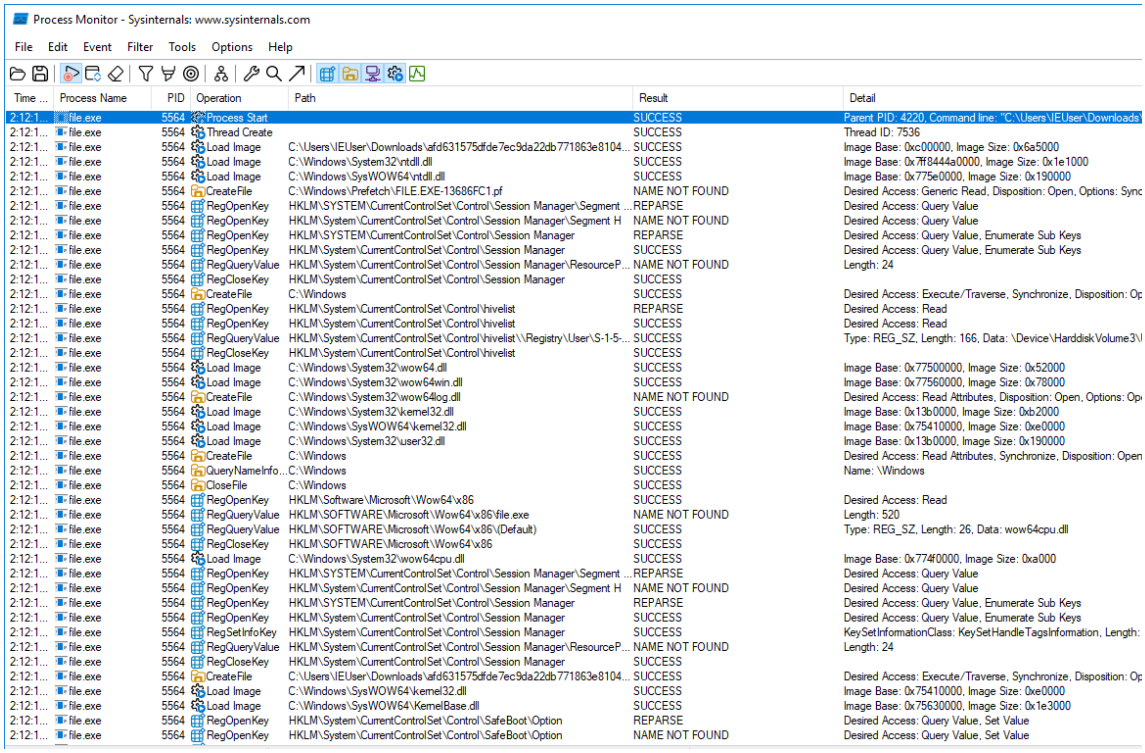


Figure 10 - Process Monitor: Runtime Behavior

This figure captures network activity generated by the malware. The malware leverages application-layer protocols (T1071) to blend into legitimate network traffic, ensuring stealthy C2 communication.

- Outbound connections to 185.215.113.37, nexusrules.officeapps.live.com.
- Traffic contains obfuscated commands and data.
- DNS tunneling detected for covert data exchange.

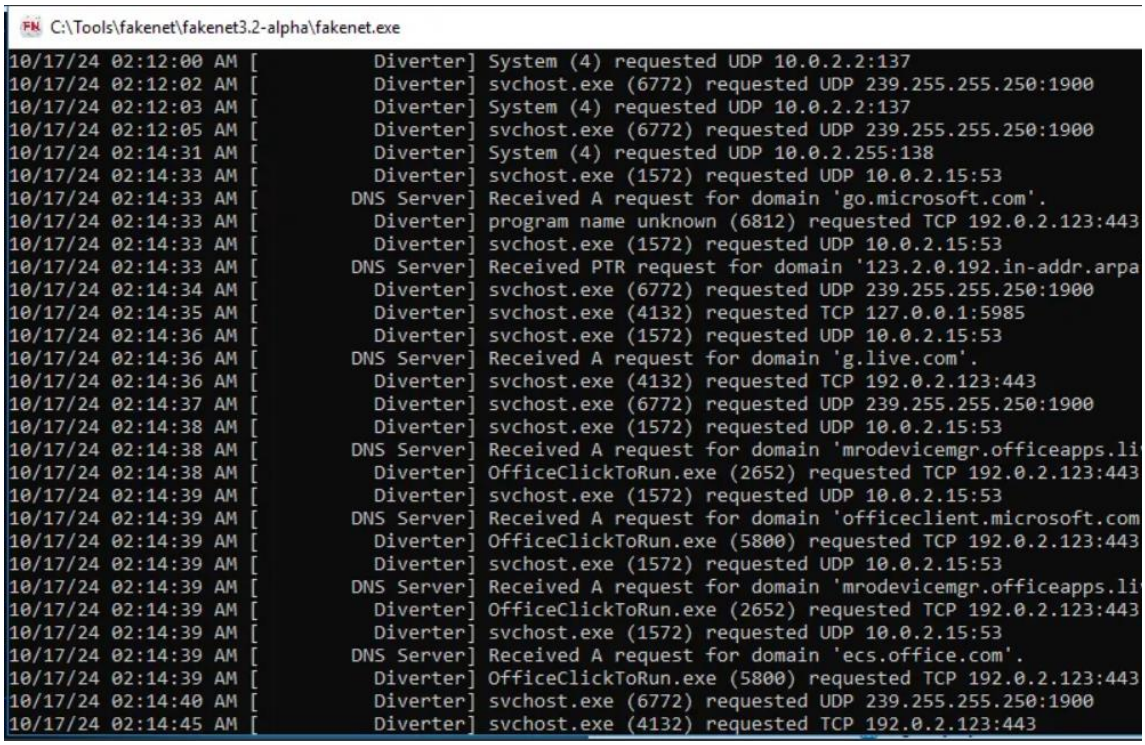


Figure 11 - FakeNet Logs: Network Communication

AnyRun Sandbox Process Graph provides sandbox-detected behaviours. The highlighted file.exe is the central malicious payload.

- Process begins with file.exe as the main executable responsible for initiating the malicious activity.
- file.exe leads to the execution of sppextcomobj.exe, followed by slui.exe.
- Both of these secondary executables are legitimate Windows components:
  sppextcomobj.exe: Typically associated with Software Protection Platform, often targeted by malware for persistence or license tampering.
  slui.exe: Windows Activation Client, often used in process injection or masquerading attacks.

The malware is executing additional malicious operations as disguising as trusted Windows processes.

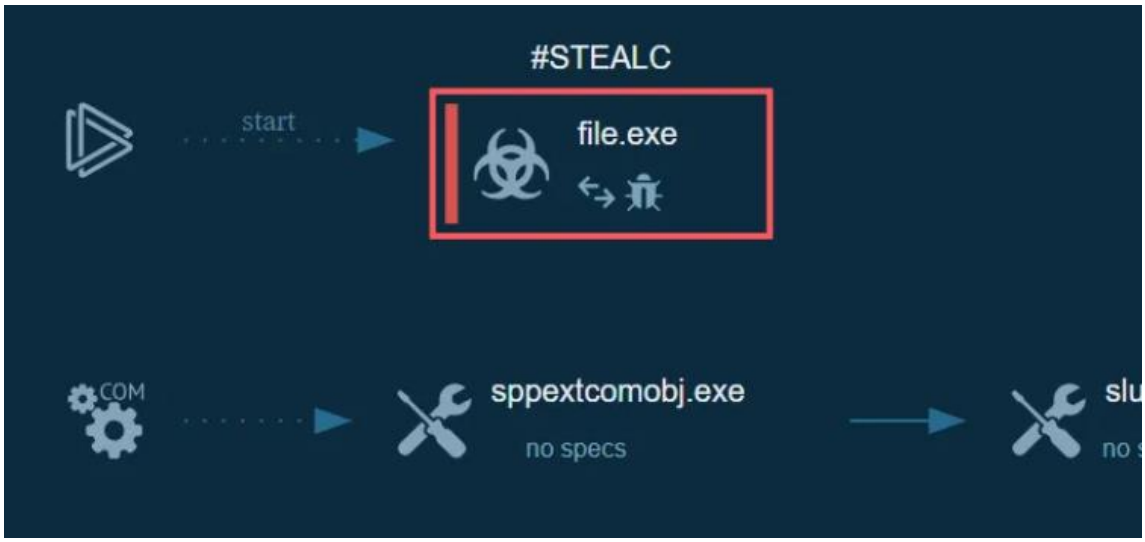


Figure 12 - AnyRun Sandbox Process Graph

Figure below shows logs of HTTP requests generated in a sandboxed environment. The use of HTTP ensures compatibility with corporate firewalls, and the encoded payloads confirm ongoing C2 communication post-infection

- HTTP GET requests to C2 servers embedding encoded instructions.
- Responses suggest staged payload delivery or command execution.

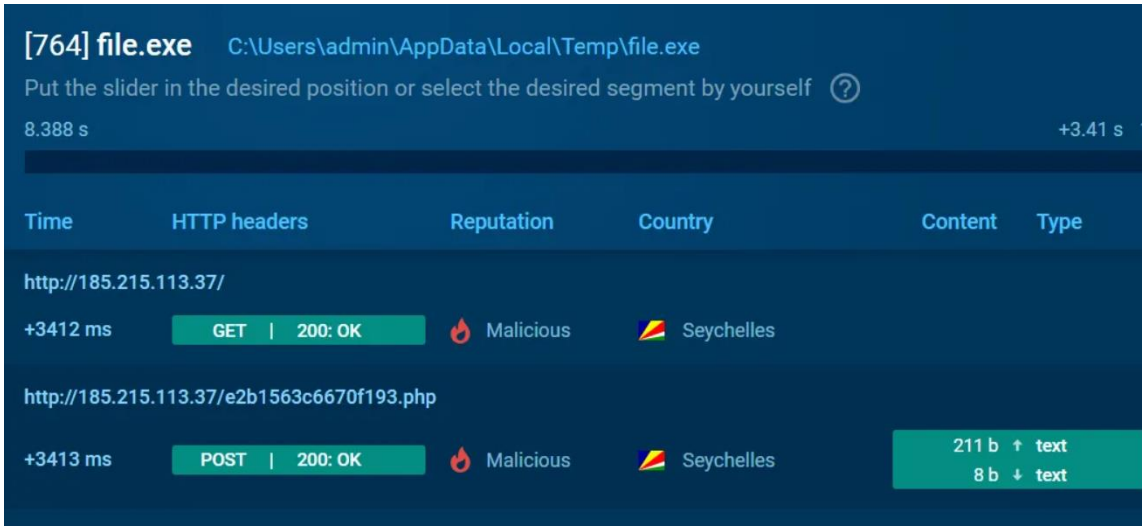


Figure 13 - AnyRun Sandbox: HTTP Request Details

Recommendations / Additional Information

Remediation options available to effectively remove the malware and return the system to a secure state

1. Isolation

Immediate Action:

- Disconnect infected systems from the network to prevent further propagation or data exfiltration.
- Block suspicious IPs and domains identified in the IOCs (e.g., 185.215.113.37, nexusrules.officeapps.live.com).
- Disable shared drives and accounts that might allow lateral movement.

2. Detection and Forensic Analysis

Host-Based Scanning:

- Use EDR (Endpoint Detection and Response) solutions to identify malicious files, processes (e.g., svchost.exe with unusual arguments), and registry changes.
- Scan for dropped files in %TEMP%, %APPDATA%, and %SystemRoot%\System32\.

Registry Inspection:

- Check for persistence keys in:
  - HKLM\Software\Microsoft\Windows\CurrentVersion\Run.
  - Task-related keys in HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Schedule.
- Cross-verify added and deleted keys from **Regshot** results.

Network Traffic Analysis:

- Review FakeNet logs for outbound HTTPS requests or DNS tunneling to suspicious domains.
- Use SIEM tools to detect traffic patterns associated with the malware.

3. Neutralization

Malware Removal:

- Use dedicated malware removal tools (e.g., Malwarebytes, CrowdStrike, or Kaspersky).
- Manually delete:
  - Malicious files identified during analysis.
  - Scheduled tasks like GoogleUpdaterTaskSystem130.0.6679.0.
  - Encrypted payloads in %TEMP% or %APPDATA%.

Kill Malicious Processes:

- Terminate processes identified as malicious (e.g., anomalous svchost.exe instances).
- Use tools like Process Explorer to trace and kill malware-associated processes.

4. Recovery

Restore Registry and Configuration:

- Revert unauthorized registry changes by importing clean backups.
- Delete keys added by the malware to disrupt persistence mechanisms.

Rebuild Critical Systems:

- For heavily compromised systems, consider reimaging or reinstalling the OS to ensure complete eradication.
- Verify integrity of critical files using checksums or hashes.

5. Patch Management

Apply Security Patches:

- Patch all systems for vulnerabilities linked to known CVEs and potential exploitation paths.
- Update software, browsers, and dependencies to close existing security gaps.

Endpoint Security Updates:

- Ensure endpoint protection tools are updated with the latest threat signatures and rules to detect similar malware.