Lab 3 - Dynamic Malware Analysis: Behavioral Profiling of a Suspicious Executable

Course: Advanced Cyberwarfare Programme

Course Code: ACW202 – Offensive Cyber Operations

Date: Apr 4, 2025

Version 1.0

Table of Content

Table of Content **2**
1. Executive Summary **3**
2. Analysis **3**
Procmon Results 3
Regshot Comparison 4
Network Traffic 4
3. Indicators of Compromise (IOCs) **5**
4. Conclusion **5**
5. References **6**

1. Executive Summary

This report presents the dynamic analysis and behavioral profiling of a suspicious executable named budget-report.exe conducted within a controlled virtual environment. Utilizing industry-standard tools including Procmon, Regshot, FakeNet-NG, and Wireshark, the analysis aimed to uncover system-level impacts, potential persistence mechanisms, and signs of command-and-control (C2) communication. The executable was observed performing several notable actions, including creating a prefetch file confirming execution, modifying registry keys associated with shell execution history, and altering multiple user and system registry hives. A new scheduled task entry was detected, suggesting a likely persistence mechanism. Additionally, interactions with Windows compatibility and SmartScreen features indicate possible user deception or privilege escalation attempts. While network communication analysis is pending deeper inspection of packet capture (PCAP) data, the executable demonstrated suspicious characteristics warranting further forensic investigation. Several Indicators of Compromise (IOCs) were identified, supporting the classification of the file as potentially malicious.

2. Analysis

This section outlines the behavioral analysis conducted on the suspicious executable within a controlled virtual environment. The focus is on identifying significant system-level interactions, including file system changes, registry modifications, process activity, and potential signs of persistence or evasion tactics. The goal is to build a profile of the malware's capabilities and assess its potential impact.

2.1 Tools Used

The following tools were employed during the dynamic analysis process:

- **Flare-VM:** a collection of software installations scripts for Windows systems that was used to setup and maintain a reverse engineering environment on a VirtualBox VM
- **Procmon (Process Monitor)**: Captured low-level system activity, including file I/O, process creation, and registry access.
- **Wireshark & FakeNet-NG**: Monitored network communications, spoofed internet services, and detected potential command-and-control (C2) behavior.
- **Regshot**: Provided a snapshot-based comparison of the Windows Registry before and after malware execution.
- **Recmon (Regmon)**: Logged real-time registry access, showing which keys were created, modified, or deleted.

2.2 File System Activity

Analysis of file system behavior revealed a number of changes consistent with application execution and potential malicious behavior:

- The file C:\Windows\Prefetch\BUDGET-REPORT.EXE-F38FC483.pf was created, confirming the malware was executed.
- New log files were created in C:\Windows\System32\winevt\Logs\Microsoft-Windows-UAC-FileVirtualization%4Operational.evtx. Modified log files included entries in SOFTWARE.LOG2, SYSTEM.LOG2, and event logs like Program-Compatibility-Assistant.evtx, which may indicate compatibility checks or tracking by Windows security features.
- **System Files** such as bootstat.dat, lastalive0.dat, and lastalive1.dat were modified — these changes are typically benign but confirm active session updates.

These actions demonstrate that the sample made deliberate file operations that included executing external tools and potentially modifying system behaviors via logs and execution patterns.

2.3 Registry Modifications

The malware modified several key areas of the Windows Registry:

- Numerous entries under UserAssist\{CEBFF5CD-ACE2-4F4F-9178-9926F41749EA} showed **obfuscated entries** (HRZR_PGYFRFFVBA → ROT13 for UEEM_CLISESSION), indicating GUI interactions and tracking.
- Registry keys under HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Schedule\TaskCache\Tasks\{GUID} and HKCU\Software\Microsoft\Windows NT\CurrentVersion\AppCompatFlags\Compatibility Assistant\Store suggest that scheduled tasks or compatibility settings may be used for maintaining persistence.

These registry actions highlight both the telemetry the malware left behind and its attempts to use Windows functionality (like Task Scheduler or Compatibility Assistant) to maintain execution capabilities or monitor user activity.

2.4 Process and Execution Behavior

The malware triggered or interacted with the following processes:

- **Process Spawning**: Evidence from the registry and prefetch data suggests the executable either directly spawned or was involved in the launching of `cmd.exe` (Command Line) and `conhost.exe` (Console Window Host). These actions

suggest the malware may unpack or manipulate files, display decoy documents, or execute system-level commands.

- **Indirect GUI Usage**: Registry entries in UserAssist and AppCompatFlags imply that the executable either used GUI-based interaction or spoofed such activity to appear as legitimate usage.

There is no clear evidence of code injection or privilege escalation at this stage, but the combination of process usage and registry alterations points to efforts to blend in with typical user activity and persist through scheduled tasks or compatibility settings.

4. Network Behavior

The malware sample exhibited basic network interaction during dynamic execution. Network monitoring tools such as FakeNet-NG and Wireshark were used to intercept and analyze the traffic generated by the executable in an isolated environment.

3. Indicators of Compromise (IOCs)

The following table outlines the key Indicators of Compromise identified during dynamic analysis of the suspicious executable. These IOCs include unique file paths, modified registry keys, artifacts of execution, and other system-level changes that may assist in detection, threat hunting, or IOC sharing.

Type	Indicator/Path	Description
File Created	C:\Windows\Prefetch\BUDGET-REPORT.EXE-F38FC483.pf	Confirms execution of the suspicious executable
File Modified	C:\Windows\Prefetch\7ZFM.EXE-69B8961D.pf, CMD.EXE-AC113AA8.pf, CONHOST.EXE-1F3E9D7E.pf, NOTEPAD.EXE-D8414F97.pf	Indicates launch of supporting applications
File Modified	C:\Windows\System32\winevt\Logs\Microsoft-Windows-UAC-FileVirtualization%4Operational.evtx	Logging of file virtualization activity by UAC

Registry Key	HKLM\SYSTEM\CurrentControlSet\Services\bam\State\UserSettings*\7zFM.exe, notepad.exe, conhost.exe	Background Activity Moderator entries logging recent app usage
Registry Key	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Schedule\TaskCache\Tasks\{AF73DAAA-53AE-4CC8-8671-BE29D886B057}	Scheduled task possibly created or modified
Registry Key	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Security and Maintenance\Checks\{E8433B72-5842-4d43-8645-BC2C35960837}.check.102\CheckSetting	Security center check triggered
File Modified	C:\Windows\System32\config\SOFTWARE.LOG2, SYSTEM.LOG2, DEFAULT.LOG2	Core registry hives changed during execution
File Created	C:\Windows\System32\winevt\Logs\Microsoft-Windows-Application-Experience%4Program-Compatibility-Assistant.evtx	Log created/updated by Compatibility Assistant
Log File	C:\Windows\ServiceState\EventLog\Data\lastalive0.dat, lastalive1.dat	Heartbeat/status files updated
Obfuscated Name	HRZR_PGYFRFFVBA, Zvpebfbsg.Jvaqbjf.Rkcybere (ROT13 of common paths e.g., Microsoft.Windows.Explorer)	UserAssist data—indicative of GUI execution tracking and obfuscation

4. Conclusion

The analyzed malware sample exhibited typical behaviors associated with a reconnaissance or staging payload, with a focus on file system interaction, registry modifications, and potential command-and-control (C2) communication attempts. Key indicators include execution of known system utilities (cmd.exe), registry entries indicative of application history and behavioral tracking.

Although the malware did not achieve persistence during the observation window, several registry and prefetch artifacts confirm repeated execution and suggest user-driven interaction. Additionally, the creation of a prefetch file for a suspicious executable (BUDGET-REPORT.EXE) and modifications to system logs and program compatibility databases further support behavioral profiling consistent with potentially malicious intent.

4.1 Detection Opportunities

- Monitoring for creation or execution of unsigned binaries such as BUDGET-REPORT.EXE.
- Logging of outbound DNS and HTTP requests to unrecognized domains (even failed resolutions).
- Registry key auditing for UserAssist, Run, and AppCompatFlags\Compatibility Assistant\Store.
- Use of prefetch files to correlate first-run timestamps and execution frequency.

4.2 Recommendations

- Implement behavioral monitoring via EDR solutions that can track file, process, and network anomalies.
- Apply network segmentation and DNS filtering to prevent unauthorized outbound communications.
- Harden system logging and retention to ensure forensic visibility into registry and file access.
- Perform hash-based blocking and reputation checks on unknown or newly introduced executables.

5. References

Brennan, S. (2018, May 22). A practical guide to dynamic malware analysis. SANS Institute. Retrieved May 20, 2025, from https://www.sans.org/white-papers/39083/

FireEye/Mandiant. (n.d.). FLARE-VM. GitHub. Retrieved May 20, 2025, from https://github.com/mandiant/flare-vm

Hussein, M. (2022, August 15). *Dynamic malware analysis using Regshot* [Video]. YouTube. https://www.youtube.com/watch?v=3qWEPleT-iU

Malware Analysis Academy. (2022, November 25). *Dynamic malware analysis with Procmon, Wireshark & FakeNet-NG* [Video]. YouTube. https://www.youtube.com/watch?v=i2I37T23mpI

Mandiant. (n.d.). Detecting malware persistence via registry keys. Mandiant Threat Intelligence. Retrieved May 20, 2025, from https://www.mandiant.com/resources/blog/malware-persistence-registry-keys