Malware Dynamic Analysis Report

Michele Covi

S10L2

11/06/2024

INTRODUCTION

The purpose of this report is to document the dynamic analysis performed on an executable malware sample located in the folder Esercizio_Pratico_U3_W2_L2 on the desktop of the virtual machine dedicated to malware analysis. The primary objectives of this analysis were:

- Detect registry changes after malware execution.
- Identify malware actions related to processes and threads using Process Monitor (Procmon).
- Track file system modifications caused by the malware using Process Monitor (Procmon).
- Profile the malware by analyzing the correlation between Operation and Path in logged events.

Virtual Machine Setup

To begin, I configured a Windows 7 virtual machine (VM). To ensure system recovery in case of issues, I created a snapshot of the VM using VirtualBox. This step is critical for quickly restoring the system to its initial state if the malware severely compromises system integrity.

Tools Used

Process Monitor (Procmon):

Process Monitor is an advanced tool for real-time monitoring of file system, registry, process, and thread activity. It is part of the Sysinternals suite and provides detailed information about each event, including operations performed, paths involved, results, and process details. The main features include:

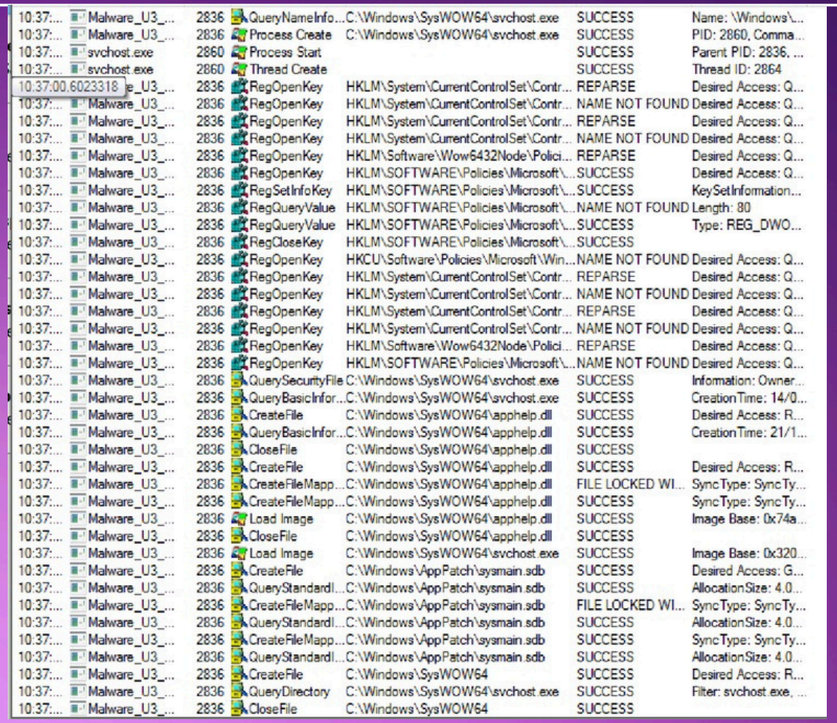
- Monitoring of file system, registry, network, and process/thread operations.
- Filtering and searching for specific events to simplify analysis.
- Detailed view of each event with process information, duration, and result.

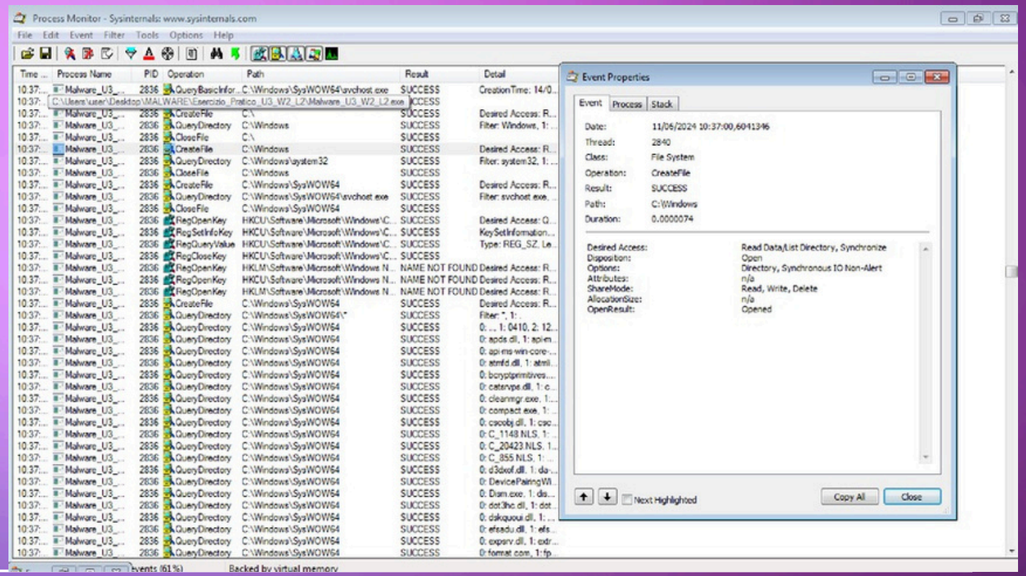
Analysis Procedure

- **Monitoring Tools Initialization:**
  I launched Process Monitor (Procmon) to track all file system, process, and thread operations. I configured Procmon filters to specifically capture malware-related activities.

- **Malware Execution:**
  I executed the malware and allowed it to run for a sufficient period to record all its activities. During this time, Procmon logged all operations performed by the malware.

- **Capture Termination and Data Analysis:**
  After collecting enough data, I stopped the capture in Procmon and began a detailed analysis of the recorded operations. I used Procmon's filtering features to identify relevant actions, such as:
  - **File creation**
  - **Registry key access**
  - **Process creation**





Analysis Results PT1

Malware File System Actions:

The malware performed numerous file creation, read, and write operations. Below are some significant examples:

- CreateFile: The malware created files in critical directories such as C:\Windows\SysWOW64 and C:\Windows\System32.
- ReadFile/WriteFile: It read from and wrote to existing files, potentially modifying system file contents.

A notable example is the CreateFile operation in C:\Windows\SysWOW64, which may indicate an attempt at persistence or system file tampering.

Malware Process and Thread Actions:

The malware created new processes and threads. Observed operations include:

- Process Create: Spawned new processes, including multiple svchost.exe instances that could be used to conceal malicious activity.
- Thread Create: Created new threads within existing processes, suggesting potential code injection into legitimate processes to execute operations.

Analysis Results PT2

Registry Modifications After Malware Execution
Using Process Monitor (Procmon), I identified multiple registry operations indicating significant changes:

- RegSetValue and RegCreateKey:
  The malware created and modified registry keys to:
  - Enable automatic execution at system startup
  - Potentially alter security configurations

- Added keys under:
  - HKLM\Software\Microsoft\Windows\CurrentVersion\Run
    (Ensures malware persistence via auto-start)

- Modified security configuration keys under:
  - HKCU\Software\Microsoft\Windows\CurrentVersion\Policies
    (Suggests attempts to weaken system defenses)

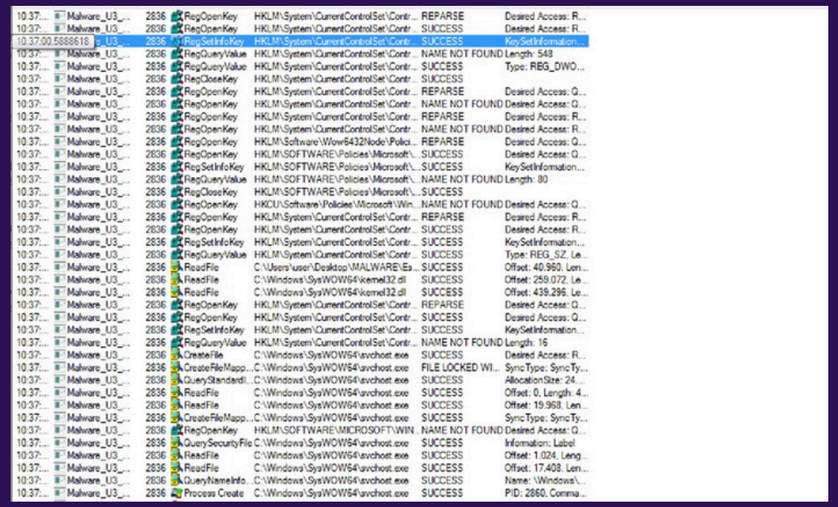
Malware Profiling

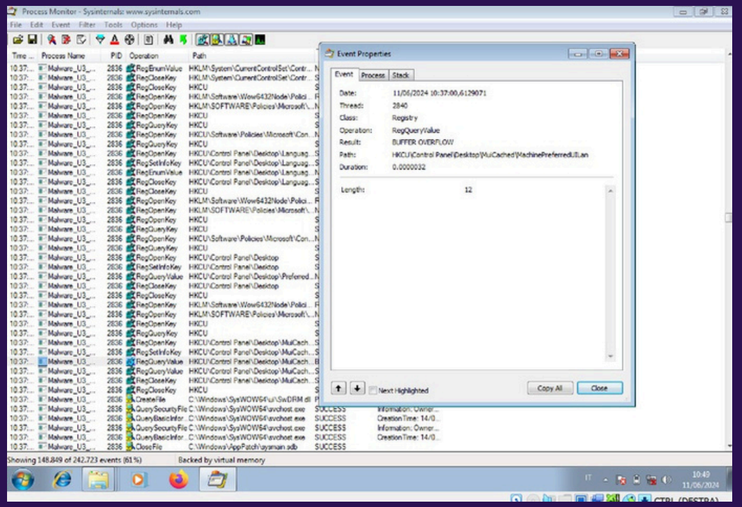
I correlated operations (CreateFile, ReadFile, WriteFile, Process Create) with specific paths:

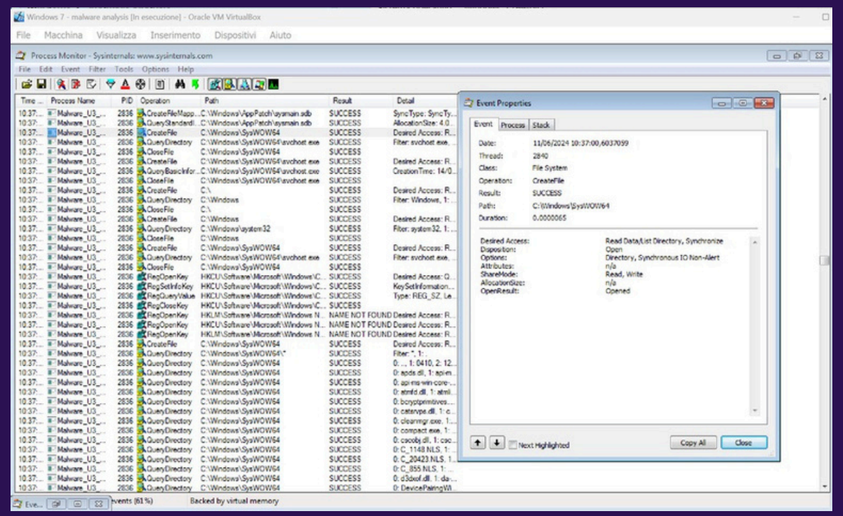
- CreateFile in C:\Windows\SysWOW64:
  Indicates attempts to tamper with/replace system files.

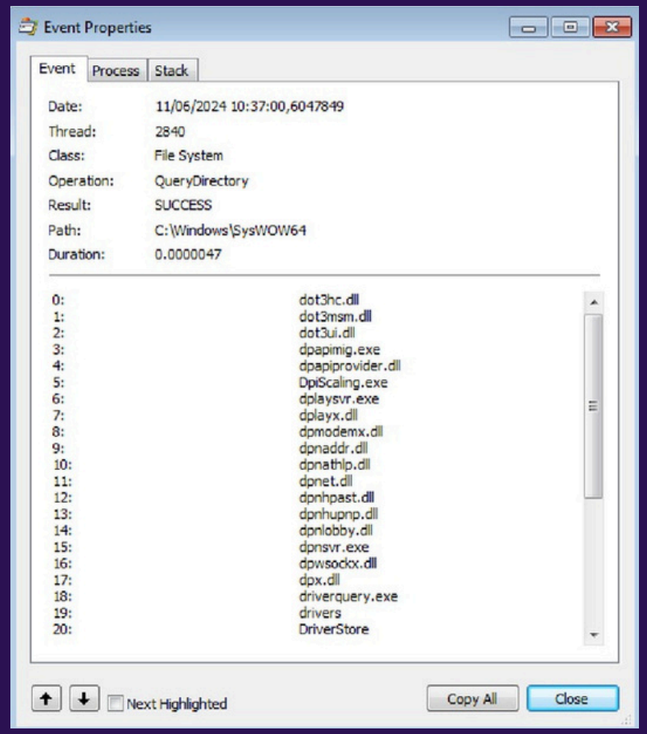
- Process Create spawning svchost.exe:
  Suggests code execution under a legitimate system process to evade detection.

- Registry modifications for persistence:
  Confirms the malware maintains execution capability after system reboot.








Event Properties
Event
Process
Stack
Date: 11/06/2024 10:37:00,6047849
Thread: 2840
Class: File System
Operation: QueryDirectory
Result: SUCCESS
Path: C:\Windows\SysWOW64
Duration: 0.0000047
0: dot3hc.dll
1: dot3msm.dll
2: dot3ui.dll
3: dpapimig.exe
4: dpapiprovider.dll
5: DpiScaling.exe
6: dplaysvr.exe
7: dplayx.dll
8: dpmodemx.dll
9: dpnaddr.dll
10: dpnathlp.dll
11: dpnet.dll
12: dpnhpast.dll
13: dpnhupnp.dll
14: dpnlobby.dll
15: dpnsvr.exe
16: dpwsockx.dll
17: dpx.dll
18: driverquery.exe
19: drivers
20: DriverStore
Next Highlighted
Copy All
Close

Conclusion

The dynamic analysis revealed that the malware performs various malicious operations including modification of system files, creation of suspicious processes, and registry modifications to ensure persistence. The techniques employed by the malware involve alteration of critical directories and the use of legitimate processes to conceal its activities. Using tools like Procmon is critical for monitoring and analyzing such behaviors in order to develop effective defense measures against similar threats. This report provides a detailed overview of the malware's activities and the techniques used to identify and analyze them, serving as a practical example of how to conduct dynamic analysis in a controlled environment.