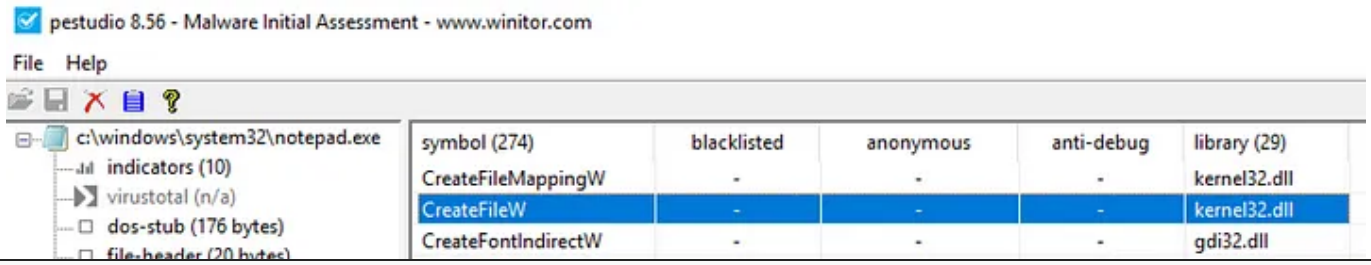
Detecting Adversary Tradecraft with Image Load Event Logging and EQL

A Windows process can load a dynamic link library (DLL) in order to use one or more of the DLL's functions to carry out certain actions. For example, `notepad.exe` loads the DLL, `kernel32.dll` before it can use the `CreateFileW` function or API call to create or open files. This is an image load event.



Microsoft Sysmon can be configured to log `Image Loaded` events to provide visibility into what DLLs are loaded by running processes.

Event ID 7: Image loaded

The image loaded event logs when a module is loaded in a specific process. This event is disabled by default and needs to be configured with the –l option. It indicates the process in which the module is loaded, hashes and signature information. The signature is created asynchronously for performance reasons and indicates if the file was removed after loading. This event should be configured carefully, as monitoring all image load events will generate a large number of events.

Description of Sysmon Event ID 7

```
ProcessId: 4132
Image: C:\Windows\System32\notepad.exe
ImageLoaded: C:\Windows\System32\kernel32.dll
FileVersion: 10.0.14393.206 (rs1_release.160915-0644)
Description: Windows NT BASE API Client DLL
Product: Microsoft® Windows® Operating System
Company: Microsoft Corporation
Hashes:
SHA1=6EE3E2D33012161659609DADEA59A2164C5A5CEB,MD5=6955067712F2F475
2CA12192B08EF860,SHA256=E02A3B57EA8B393408FF782866A1D342DD8C6B5F59
25BA527981DBB21B6A4080,IMPHASH=3CE0779E0F4E275CD51A359A98CCC682
Signed: true
Signature: Microsoft Windows
SignatureStatus: Valid
```

Process Monitor output shows vaultcli.dll loaded by vaultcmd.exe

Examining the exports table of `vaultcli.dll` suggests that this DLL provides the functionality to enumerate or get information from the credential vault.

```
Signature: Microsoft Windows
SignatureStatus: Valid
```

We can write an Event Query Language (EQL) query to detect unexpected processes loading `vaultcli.dll` as follows. If you're not familiar with EQL, you can find the getting started guide here.

Credential Enumeration via Credential Vault CLI

Example 2: Stealthy Scheduled Task Creation via VBA Macro

A Microsoft Office document can contain VBA code to create a scheduled task for persistence without using the native scheduled tasks (`schtasks.exe`)

```
Signature: Microsoft Windows
SignatureStatus: Valid
```

We can detect the behavior of Microsoft Office applications loading `taskschd.dll` with the following EQL query.

Scheduled Task Creation via Microsoft Office Application

By invoking WMI to execute a malicious `powershell.exe` command, `powershell.exe` is spawned with the parent process `wmiprvse.exe`, not `winword.exe` or whatever application is used to execute the macro.

```
C:\Windows\SysWOW64\wbem\wbemsvc.dll
C:\Windows\SysWOW64\wbem\fastprox.dll
```

Process Monitor output showing WMI-related DLLs loaded by winword.exe

What adversary tradecraft can you detect by leveraging image load event logging or by combining these events with other event types such as process, network, or file events? I'd be interested in hearing any feedback, experiences, or findings that you would like to share. For anyone who would like to share any analytics for detection, please see the EQL Analytics Library contribution guide.

Threat Detection | Threat Hunting | Information Security | Cybersecurity