Dynamic Analysis

Sean Kilgallon
Dept of Computer & Information Sciences
University of Delaware
Dynamic or behavioral analysis is observing the behavior of the malware while it is actually running on a host system.
Uses Cases

- Computer security incident management
- Malware research
- Indicator of compromise extraction
• Helping malware researchers to identify and classify malware samples

```plaintext
rule silent_banker : banker
{
    meta:
        description = "This is just an example"
        thread_level = 3
        in_the_wild = true

    strings:
        $a = {6A 40 68 00 30 00 00 6A 14 8D 91}
        $b = {8D 4D B0 2B C1 83 C0 27 99 6A 4E 59 F7 F9}
        $c = "UVODFRYSIHLNWPEJXQZAKCBGMT"

    condition:
        $a or $b or $c
}
```
“A standardized language for encoding and communicating high-fidelity information about malware based upon attributes such as behaviors, artifacts, and attack patterns”

A standard is necessary to provide a common way to share malware analysis results among organizations to avoid duplicate, inaccurate work.

https://maec.mitre.org/index.html
Sandbox Goals

Visibility

Resistance to detection

Scalability
Necessary Questions

- What kind of files do I want to analyze?
- What volume of analyses do I want to be able to handle?
- Which platform do I want to use to run my analysis on?
- What kind of information I want about the file?
Environment Setup

- What operating system should I use? Hardware?
- Intentional traces of normal usage
  - browsing history
  - Cookies
  - Documents
  - Images
- Necessary applications for malware to execute
Features

• Traces of calls performed by all processes spawned by the malware.
• Files being created, deleted and downloaded by the malware during its execution.
• Memory dumps of the malware processes.
• Network traffic trace in PCAP format.
• Screenshots taken during the execution of the malware.
• Full memory dumps of the machines.
Examples of Features

"NtQuerySystemInformation": 1,
"NtEnumerateValueKey": 2,
"GetSystemTimeAsFileTime": 4,
"CreateThread": 1,
"GetSystemInfo": 1,
"NtOpenKeyEx": 2,
"GetSystemDirectoryW": 2,
"LdrGetProcedureAddress": 45,
"NtProtectVirtualMemory": 3,
"NtCreateMutant": 1,
"SetUnhandledExceptionFilter": 1,
"NtEnumerateKey": 2,
"NtClose": 20,
"NtQueryValueKey": 12,
"NtAllocateVirtualMemory": 7,
"ReadProcessVirtualMemory": 5,
"LdrLoadDll": 3,
"LdrGetDllHandle": 38,
"NtOpenKey": 23,
"NtFreeVirtualMemory": 1
"regkey_read": [ 
  "HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\Nls\Language\InstallLanguageFallback",
  "HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Windows NT\CurrentVersion\Windows\LoadAppInit_DLLs",
  "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\GRE_Initialize\DisableMetaFiles",
  "HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\Terminal Server\TSUserEnabled",
  "HKEY_CURRENT_USER\Control Panel\Desktop\PreferredUILanguages",
  "HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\MUI\UILanguages\en-US\Type",
  "HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\Terminal Server\TSAppCompat",
  "HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\MUI\UILanguages\en-US\AlternateCodePage",
  "HKEY_CURRENT_USER\Control Panel\Desktop\MuiCached\MachinePreferredUILanguages",
  "HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Control\Nls\CustomLocale\EMPTY"
],
"dll_loaded": [ 
  "psapi.dll",
  "C:\Windows\system32\IMM32.DLL",
  "User32.dll"
]
Dynamic Downfalls

- Dynamic malware analysis is not deterministic
  - Success depends on a billion factors

- Anti-sandboxing malware
  - Environmental awareness
  - Obfuscating internal data
  - Timing based evasion
  - Simulated Randomness
What tools are available?

https://github.com/rshipp/awesome-malware-analysis