Sysmon FTW!

BOTCONF 2017 – LIGHTNING TALK
C:\> whoami /all

- Tom Ueltschi
- Swiss Post CERT / SOC / CSIRT, since 2007 (10 years!)
  - Focus: Malware Analysis, Threat Intel, Threat Hunting, Red Teaming
- Talks about «Ponmocup Hunter» (Botconf, DeepSec, SANS DFIR Summit)
- BotConf 2016 talk with same title
- Member of many trust groups / infosec communities
- FIRST SIG member (Malware Analysis, Red Teaming)
- Twitter: @c_APT_uRe
BotConf Speaker history

• 2013 - My Name is Hunter, Ponmocup Hunter
• 2014 - Ponmocup Hunter 2.0 – The Sequel
• 2015 - LT: Creating your own CTI (in 3 minutes.. or 5 😊)
• 2016 - Advanced Incident Detection and Threat Hunting using Sysmon (and Splunk)
• 2017 - CFP got REJECTED (for the first time 😞) ... Now what???

Write a book on how to deal with rejection? Maybe not, but...
BotConf Speaker history

• 2013 - My Name is Hunter, Ponmocup Hunter
• 2014 - Ponmocup Hunter 2.0 – The Sequel
• 2015 - LT: Creating your own CTI (in 3 minutes.. or 5 😊)
• 2016 - Advanced Incident Detection and Threat Hunting using Sysmon (and Splunk)
• 2017 - LT: Sysmon FTW! 😊

or in Botnet terms: turn a “reject” into an “inject”
Sysmon v6.20

Published: November 19, 2017

By Mark Russinovich and Thomas Garnier

Download Sysmon (1.4 MB)

Introduction

System Monitor (Sysmon) is a Windows system service and device driver that, once installed on a system, remains resident across system reboots to monitor and log system activity to the Windows event log. It provides detailed information about process creations, network connections, and changes to file creation time. By collecting the events it generates using Windows Event Collection or SIEM agents and subsequently analyzing them, you can identify malicious or anomalous activity and understand how intruders and malware operate on your network.

Note that Sysmon does not provide analysis of the events it generates, nor does it attempt to protect or hide itself from attackers.
Sysmon v6.20

Published: November 19, 2017

By Mark Russinovich and Thomas Garnier

Event ID 19: WmiEvent (WmiEventFilter activity detected)

When a WMI event filter is registered, which is a method used by malware to execute, this event logs the WMI namespace, filter name and filter expression.

Event ID 20: WmiEvent (WmiEventConsumer activity detected)

This event logs the registration of WMI consumers, recording the consumer name, log, and destination.

Event ID 21: WmiEvent (WmiEventConsumerToFilter activity detected)

When a consumer binds to a filter, this event logs the consumer name and filter path.
Sysmon v6.20

This Sysmon release adds the ability to change the Sysmon service and driver names to foil malware that use them to detect its presence.
MAINTAIN PERSISTENCE

- WMI Persistence requires three components
  - An event filter – the condition we’re waiting for
    - _EventFilter objects have a name and a “trigger”
  - An event consumer – the persistence payload
    - _EventConsumer objects have a name and one of the following:
      - A script (located in objects.data)
      - A path to an external script (somewhere on disk)
      - A path to an executable (not a script, also on disk)
    - Pre-Vista ran as SYSTEM
    - Post-Vista run as LOCAL SERVICE
  - A binding that associates a filter to a consumer
    - _FilterToConsumerBinding objects reference an event filter and an event consumer
Malicious WMI Persistence Example

The PowerShell code in Figure 5 is a modified instance of the WMI persistence code present in the SEADADDY* malware family. The event filter was taken from the Powersploit persistence module and is designed to trigger shortly after system startup. The event consumer simply executes an executable with SYSTEM privileges.

```powershell
$filterName='BotFilter82'
$consumerName='BotConsumer23'
$exePath='C:\Windows\System32\ev11.exe'
$query="SELECT * FROM __InstanceModificationEvent
      WITHIN 60 WHERE TargetInstance ISA 'Win32_PerfFormattedData_PerfOS_System' AND
      TargetInstance.SystemUpTime >= 200 AND
      TargetInstance.SystemUpTime < 320"

$WMIEventFilter=Set-WmiInstance-Class__EventFilter-NameSpace"root\\subscription"-Arguments @
{Name=$filterName;EventNameSpace="root\\cimv2";QueryLanguage="WQL";Query=$query} -ErrorActionStop

$WMIEventConsumer=Set-WmiInstance-ClassCommandLineEventConsumer-NameSpace"root\\
subscription"-Arguments={$consumerName;ExecutablePath=$exePath;CommandLineTemplate=$exePath} -ErrorActionStop

Set-WmiInstance-Class__FilterToConsumerBinding-
```
Detecting Internal Recon

* 3 or more (of 7) different commands executed within 15 min

```
index= source=type= "WinEventLog:Microsoft-Windows-Sysmon/Operational" ProcessCreate
    (ipconfig OR net.exe OR whoami OR netstat OR nbtstat OR hostname OR tasklist)

| search EventCode=1
  - Image="\\\ipconfig.exe" OR Image="\\\net.exe" OR Image="\\\whoami.exe" OR Image="\\\netstat.exe" OR Image="\\\nbtstat.exe" OR Image="\\\hostname.exe" OR Image="\\\tasklist.exe"
  - bin.time span=15m
| rex field=Message "\.(User: (?<USER1>)\{NT AUTHORITY\}\\(\?<USER1>\),\")"
| stats dc(Image) AS CNT_CMDS values(CommandLine) values(ParentImage) values(ParentCommandLine)
  count by _time ComputerName USER1
| where CNT_CMDS > 2
```
Detecting C2 using Named Pipes

* Search for Processes
  - Connecting through Web Proxy and
  - Creating Named Pipes

whitelisting vetted good processes
Detecting Mimikatz

* Search for ProcessAccess of LSASS.exe
  - GrantedAccess of: `0x1010, 0x1410, 0x143A`
  - CallTrace: `KERNELBASE.dll and (ntdll.dll or UNKNOWN)`
Is this blog still alive?

Is this blog still alive? That’s a valid question since I haven’t blogged for quite some time. (now, has it really been more than 3 years?!) So I finally decided to write another post about some stuff that happened in the meantime.

For the past few years I have been more active on Twitter (@c_APT_ure) and also presenting at conferences and collaborating in closed / trusted groups.

My most recent area of interest has been increasing endpoint visibility using Sysinternals Sysmon and sending logs into Splunk for incident detection and threat hunting.

My first presentation was a year ago at BotConf 2016:

“Advanced Incident Detection and Threat Hunting using Sysmon (and Splunk)”

Video: https://www.youtube.com/watch?v=vv_YXnlQ7pE

This year I gave an updated version on the same topic at the FIRST annual conference.


There are many good resources for further reading that I can suggest.

- Sysmon - DFIR (Mike Haag / @MIHaggs)
  - Sysmon Config files
- ThreatHunter-Playbook (Roberto Rodriguez / @Cyb3rWord0g)
- Operational Look at Sysinternals Sysmon 6.20 Update

The list of resources may get updated every so often...
Thanks for accepting my LT!!
I could not have accepted another rejection

• Twitter: @c_APT_ure

⇒ many resources about Sysmon linked in one place