Practical Threat Hunting

TOM UELTSCHI

CERT-EU 2019 ANNUAL CONFERENCE

CERT-EU

FOR THE EU INSTITUTIONS, BODIES AND AGENCIES

CERT-EU 2019 ANNUAL CONFERENCE

5-6 NOVEMBER 2019, BRUSSELS, BELGIUM

C:> whoami /all

- Tom Ueltschi
- Swiss Post CERT / SOC / CSIRT since 2007 *(over 12 years!)*
- Focus & Interests: Malware Analysis, Threat Intel, Threat Hunting, Red / Purple Teaming
- Member of many trust groups & infosec communities
- FIRST SIG member (malware analysis, red teaming, CTI)
- Twitter: @c_APT_ure

Previous presentations including “Threat Hunting”

- “Advanced Incident Detection and Threat Hunting using Sysmon (and Splunk)“
  - BotConf 2016 (almost 3 years ago)
  - FIRST Annual Conference 2017
  - FIRST TC Amsterdam 2018
- “Hunting and Detecting APTs using Sysmon and PowerShell Logging“
  - BotConf 2018

Previous presentations including “Threat Hunting”

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 in December 2016 at BotConf:

"Advanced Incident Detection and Threat Hunting using Sysmon (and Splunk)"

Slides: https://www.botconf.eu/wp-content/uploads/2016/11/PR12-Sysmon-UELTSCHI.pdf
Video: https://www.youtube.com/watch?v=vv_VXntQTpE

In 2017 I gave an updated version on the same topic at the FIRST annual conference.

Slides: https://www.first.org/resources/papers/conf2017/Advanced-Incident-Detection-and-Threat-Hunting-using-Sysmon-and-Splunk.pdf

In April 2018 at FIRST TC Amsterdam, I gave an updated version from the FIRST 2017 talk.

Slides: FIRST-TC-2018_Tom-Ueltschi_Sysmon_PUBLIC.pdf (Github download)

!! NEW !!
At BotConf 2018, I presented again on using Sysmon and Splunk, but also including Powershell Logging and MITRE ATT&CK as well.
"Hunting and Detecting APTs using Sysmon and PowerShell Logging"

Slides: 2018-Tom-Ueltschi-Sysmon.pdf
Video: *(was recorded and will be published soon)*

https://c-apt-ure.blogspot.com/2017/12/is-this-blog-still-alive.html

Outline

- Introduction
- New stuff:
  - T1064: Scripting - VBS Scripts
  - T1060: Registry Run Keys / Startup Folder - dropping VBS file in Startup
  - T1071: Standard Application Layer Protocol - Command and Control via DNS

  (T1234 = MITRE ATT&CK Technique #)
- Quick review: 3 techniques from MITRE ATT&CK
  - BotConf 2018 presentation
    "Hunting and Detecting APTs using Sysmon and PowerShell Logging"

Threat Hunting with[out] (the right) data?

https://cyberwardog.blogspot.com/2017/12/ready-to-hunt-first-show-me-your-data.html

Cyber Wardog Lab

by Roberto Rodriguez

Home

Friday, December 15, 2017

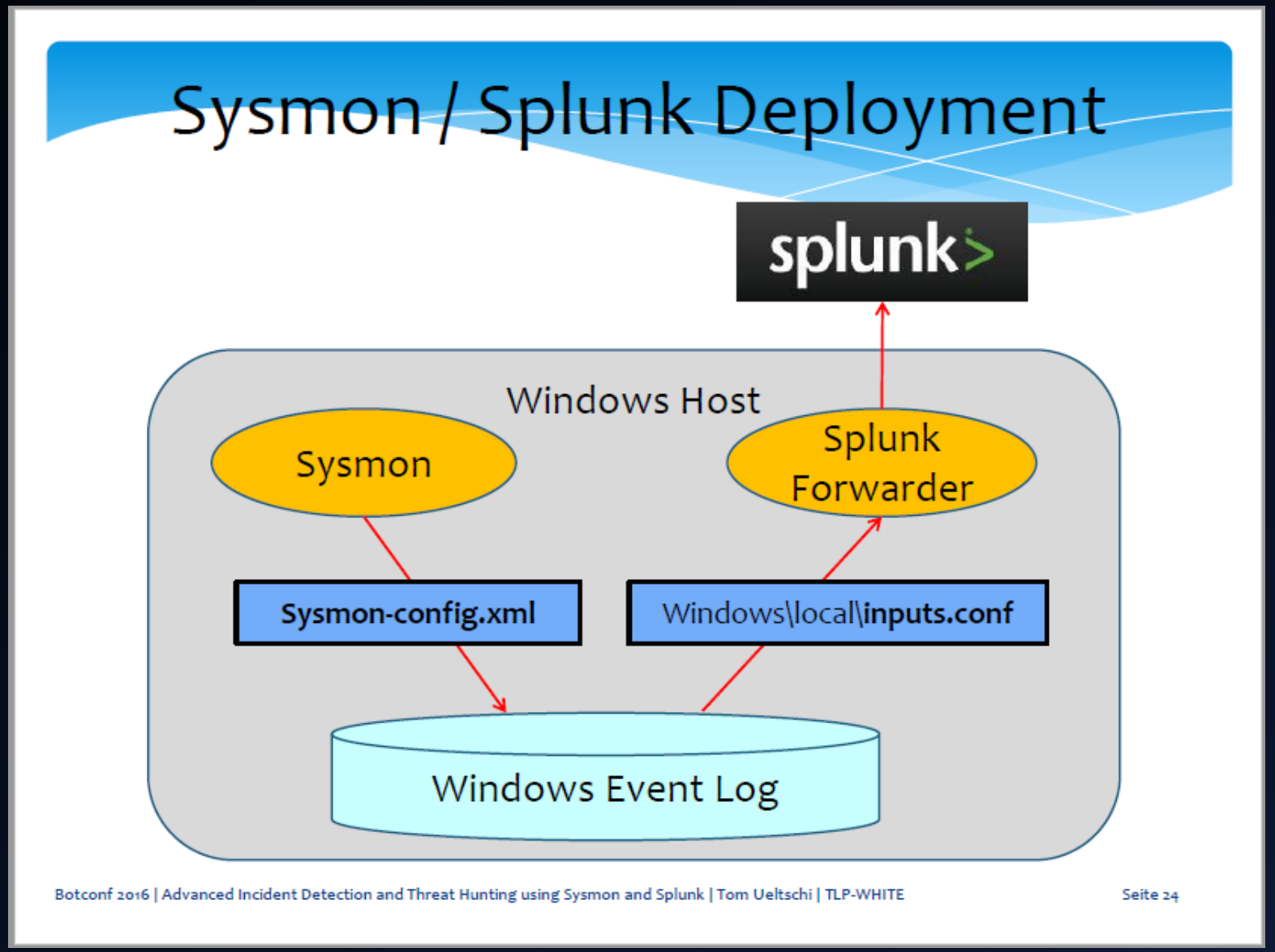
Ready to hunt? First, Show me your data!

Ready to hunt?

First, show me your data!

Our setup

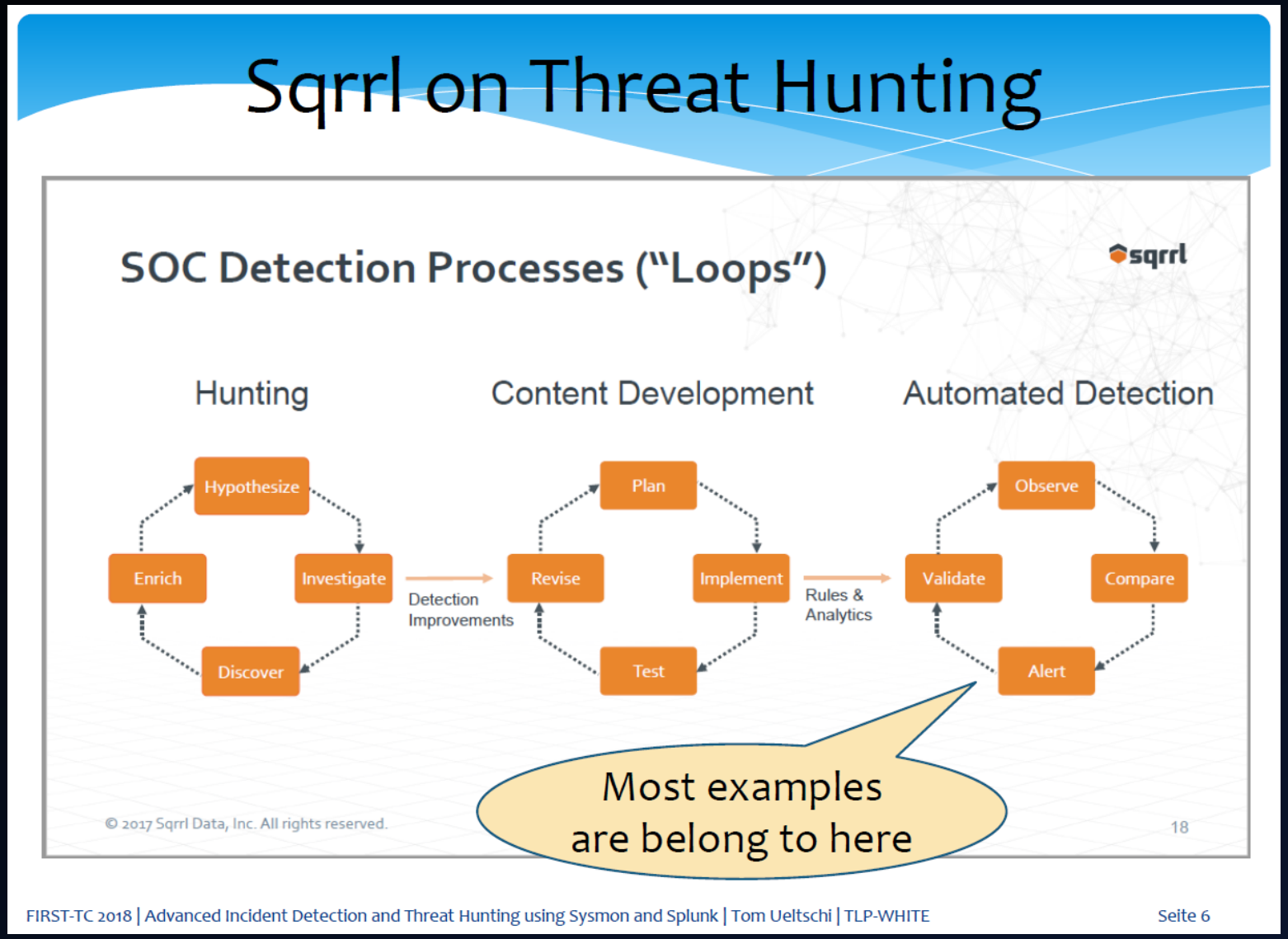
- ~25’000 hosts
- ~150 GB/day
- Event logs
  - Windows
  - Sysmon
  - Powershell



Data Sources & Event Logs

- Sysmon
- PowerShell ScriptBlock Logging
- Windows Event Logs
- Network
  - DNS, Web Proxy, Netflow, Firewalls

Threat Hunting → Automated Detection



The ThreatHunter Playbook Project

github.com/hunters-forge/ThreatHunter-Playbook

The ThreatHunter-Playbook

launch binder | License GPLv3 | Follow 1.2k | Open Source

Goals

- Expedite the development of techniques an hypothesis for hunting campaigns.
- Help Threat Hunters understand patterns of behavior observed during post-exploitation.
- Reduce the number of false positives while hunting by providing more context around suspicious events.
- Share real-time analytics validation examples through cloud computing environments for free.
- Distribute Threat Hunting concepts and processes around the world for free.
- Map pre-recorded datasets to adversarial techniques.

Author

- Roberto Rodriguez @Cyb3rWard0g
- Jose Luis Rodriguez @Cyb3rPandaH

The ThreatHunter Playbook Project (Playbook)

github.com/hunters-forge/ThreatHunter-Playbook/blob/master/playbooks/windows/08_lateral_movement/T1047_windows_management_instrumentation/w...

hunters-forge / ThreatHunter-Playbook

Watch 278 | Star 1.7k | Fork 394

Code | Issues 2 | Pull requests 3 | Projects 0 | Wiki | Security | Insights

Branch: master | ThreatHunter-Playbook / playbooks / windows / 08_lateral_movement / T1047_windows_management_instrumentation / wmi_win32_process_create_remote.ipynb | Find file | Copy path

cyb3rward0g ThreatHunter Playbook 2.0 e18bf20 on Aug 16

0 contributors

336 lines (335 sloc) | 15.2 KB | Raw | Blame | History

WMI Win32_Process Class and Create Method for Remote Execution

Playbook Tags

ID: WINEXEC190810201010

Author: Roberto Rodriguez @Cyb3rWard0g

References:

ATT&CK Tags

Tactic: Execution, Lateral Movement

Technique: Windows Management Instrumentation (T1047)

The ThreatHunter Playbook Project (Playbook)

github.com/hunters-forge/ThreatHunter-Playbook/blob/master/playbooks/windows/08_lateral_movement/T1047_windows_management_instrumentation/w...

hunters-forge / ThreatHunter-Playbook

Watch 278 | Star 1.7k | Fork 394

Code | Issues 2 | Pull requ

Branch: master | ThreatHunter-Play
T1047_windows_m

cyb3rward0g ThreatHunter Playbook 2.0

0 contributors

336 lines (335 sloc) | 15.2 KB

WMI Win32_P
Execution

Playbook Tags

ID: WINEXEC190810201

Author: Roberto Rodrigue

References:

ATT&CK Tags

Tactic: Execution, Lateral

Technique: Windows Ma

Technical Description

WMI is the Microsoft implementation of the Web-Based Enterprise Management (WBEM) and Common Information Model (CIM). Both standards aim to provide an industry-agnostic means of collecting and transmitting information related to any managed component in an enterprise. An example of a managed component in WMI would be a running process, registry key, installed service, file information, etc. At a high level, Microsoft's implementation of these standards can be summarized as follows: Managed Components Managed components are represented as WMI objects — class instances representing highly structured operating system data. Microsoft provides a wealth of WMI objects that communicate information related to the operating system. E.g. Win32_Process, Win32_Service, AntiVirusProduct, Win32_StartupCommand, etc.

One well known lateral movement technique is performed via the WMI object — class Win32_Process and its method Create. This is because the Create method allows a user to create a process either locally or remotely. One thing to notice is that when the Create method is used on a remote system, the method is run under a host process named "Wmiprvse.exe".

The process WmiprvSE.exe is what spawns the process defined in the CommandLine parameter of the Create method. Therefore, the new process created remotely will have Wmiprvse.exe as a parent. WmiprvSE.exe is a DCOM server and it is spawned underneath the DCOM service host svchost.exe with the following parameters C:\WINDOWS\system32\svchost.exe -k DcomLaunch -p.

From a logon session perspective, on the target, WmiprvSE.exe is spawned in a different logon session by the DCOM service host. However, whatever is executed by WmiprvSE.exe occurs on the new network type (3) logon session created by the user that authenticated from the network.

Additional Reading

- Logon Session

The ThreatHunter Playbook Project (Playbook)

github.com/hunters-forge/ThreatHunter-Playbook/blob/master/playbooks/windows/08_lateral_movement/T1047_windows_management_instrumentation/w...

hunters-forge / ThreatHunter-Playbook

Watch 278 | Star 1.7k | Fork 394

Code | Issues 2 | Pull requests 3

Branch: master | ThreatHunter-Playbook / play... T1047_windows_managemen...

cyb3rward0g ThreatHunter Playbook 2.0

0 contributors

336 lines (335 sloc) | 15.2 KB

WMI Win32_Proces... Execution

Playbook Tags

ID: WINEXEC190810201010

Author: Roberto Rodriguez @Cyb3rW...

References:

ATT&CK Tags

Tactic: Execution, Lateral Movement

Technique: Windows Management In...

Hypothesis

Adversaries might be leveraging WMI Win32_Process class and method create to execute code remotely across my environment.

Attack Simulation Dataset

Environment	Name	Description
Shire	empire_invoke_wmi	A mordor dataset to simulate the use of of WMI Win32_Process class and method Create to execute code remotely
Shire	empire_wmic_add_user_backdoor	A mordor dataset to simulate the use of of WMI Win32_Process class and method Create to execute code remotely
Shire	empire_invoke_wmi_debugger	A mordor dataset to simulate the use of of WMI Win32_Process class and method Create to execute code remotely

Recommended Data Sources

Event ID	Event Name	Log Provider	Audit Category	Audit Sub-Category	ATT&CK Data Source
4688	A new process has been created	Microsoft-Windows-Security-Auditing	Detailed Tracking	Process Creation	Windows Event Logs
4624	An account was successfully logged on	Microsoft-Windows-Security-Auditing	Audit Logon/Logoff	Audit Logon	Windows Event Logs
1	Process Creation	Microsoft-Windows-Sysmon			Process Monitoring

The ThreatHunter Playbook Project (Playbook)

github.com/hunters-forge/ThreatHunter-Playbook/blob/master/playbooks/windows/08_lateral_movement/T1047_windows_management_instrumentation/w...

hunters-forge / ThreatHunter-Playbook

Watch 278 | Star 1.7k | Fork 394

Code | Issues 2 | Pull req

Branch: master | ThreatHunter-Play... T1047_windows_m...

cyb3rward0g ThreatHunter Playbook 2...

0 contributors

336 lines (335 sloc) | 15.2 KB

WMI Win32_... Execution

Playbook Tags

ID: WINEXEC190810201

Author: Roberto Rodrigu...

References:

ATT&CK Tags

Tactic: Execution, Latera...

Technique: Windows Ma...

Data Analytics

Initialize Analytics Engine

In [1]:
```
from openhunt.logparser import winlogbeat
from pyspark.sql import SparkSession
```

In [2]:
```
win = winlogbeat()
spark = SparkSession.builder.appName("Mordor").config("spark.sql.caseSensitive", "True").getOrCreate()
print(spark)
```

```
<pyspark.sql.session.SparkSession object at 0x7f0f4640ffd0>
```

Prepare & Process Mordor File

In [3]:
```
mordor_file = win.extract_nested_fields("mordor/small_datasets/empire_wmic_add_user_2019-05-18231333.jso
n",spark)
```

```
[+] Processing a Spark DataFrame..
[+] Reading Mordor file..
[+] Processing Data from Winlogbeat version 6..
[+] DataFrame Returned !
```

Register Mordor DataFrame as a SQL temporary view

In [4]:
```
mordor_file.createOrReplaceTempView("mordor_file")
```

The ThreatHunter Playbook Project (Playbook)

github.com/hunters-forge/ThreatHunte

hunters-forge / ThreatHunter-

Code | Issues 2 | Pull requ

Branch: master | ThreatHunter-Playl
T1047_windows_m

cyb3rward0g ThreatHunter Playbook 2.0

0 contributors

336 lines (335 sloc) | 15.2 KB

WMI Win32_P
Execution

Playbook Tags

ID: WINEXEC1908102010

Author: Roberto Rodrigue

References:

ATT&CK Tags

Tactic: Execution, Lateral

Technique: Windows Man

Validate Analytic II

FP Rate	Source	Analytic Logic	Description
Medium	Sysmon	SELECT `@timestamp`, computer_name, User, Image, CommandLine FROM mordor_file WHERE channel = "Microsoft-Windows-Sysmon/Operational" AND event_id = 1 AND lower(ParentImage) LIKE "%wmiprvse.exe" AND NOT LogonId = "0x3e7"	Look for wmiprvse.exe spawning processes that are part of non-system account sessions.

In [6]:
```
sysmon_process_df = spark.sql(
    '''
    SELECT `@timestamp`, computer_name, User, Image, CommandLine
    FROM mordor_file
    WHERE channel = "Microsoft-Windows-Sysmon/Operational"
        AND event_id = 1
        AND lower(ParentImage) LIKE "%wmiprvse.exe"
        AND NOT LogonId = "0x3e7"
    '''
)
sysmon_process_df.show(10,False)
```

```
+-----------------------+--------------+-------------+-------------------------+--------------------
---------+
|@timestamp             |computer_name |User         |Image                    |CommandLine
|
+-----------------------+--------------+-------------+-------------------------+--------------------
---------+
|2019-05-18T23:14:57.079Z|IT001.shire.com|SHIRE\pgustavo|C:\Windows\System32\net.exe|net user /add backdoo
r paw0rd1|
+-----------------------+--------------+-------------+-------------------------+--------------------
---------+
```

The ThreatHunter Playbook Project (Playbook)

github.com/hunters-forge/ThreatHunter-Playbook/blob/master/playbooks/windows/08_lateral_movement/T1047_windows_management_instrumentation/w...

hunters-forge / ThreatHunter-Playbook

Watch 278 | Star 1.7k | Fork 394

Code | Issues 2 | Pull requests 3 | Pro

Branch: master | ThreatHunter-Playbook / playbook / T1047_windows_management_inst

cyb3rward0g ThreatHunter Playbook 2.0

0 contributors

336 lines (335 sloc) | 15.2 KB

WMI Win32_Process C Execution

Playbook Tags

ID: WINEXEC190810201010

Author: Roberto Rodriguez @Cyb3rWard0g

References:

ATT&CK Tags

Tactic: Execution, Lateral Movement

Technique: Windows Management Instrumentation (T1047)

Detection Blind Spots

Hunter Notes

- Stack the child processes of wmiprvse.exe in your environment. This is very helpful to reduce the number of false positive and understand your environment. You can categorize the data returned by business unit.
- Look for wmiprvse.exe spawning new processes that are part of a network type logon session.
- Enrich events with Network Logon events (4624 - Logon Type 3)

Hunt Output

Category	Type	Name
Signature	Sigma Rule	sysmon_wmiprvse_spawning_process.yml
Signature	Sigma Rule	win_wmiprvse_spawning_process.yml

References

- https://posts.specterops.io/threat-hunting-with-jupyter-notebooks-part-4-sql-join-via-apache-sparksql-6630928c931e
- https://posts.specterops.io/real-time-sysmon-processing-via-ksql-and-helk-part-3-basic-use-case-8fbf383cb54f
- https://www.youtube.com/watch?v=iiaPeXEn5_E

The ThreatHunter Playbook Project (SIGMA Rules)

github.com/hunters-forge/ThreatHunter-Playbook/tree/master/signatures/sigma

hunters-forge / ThreatHunter-Playbook

Watch 278 | Star 1.7k | Fork 394

Code | Issues 2 | Pull requests 3 | Projects 0 | Wiki | Security | Insights

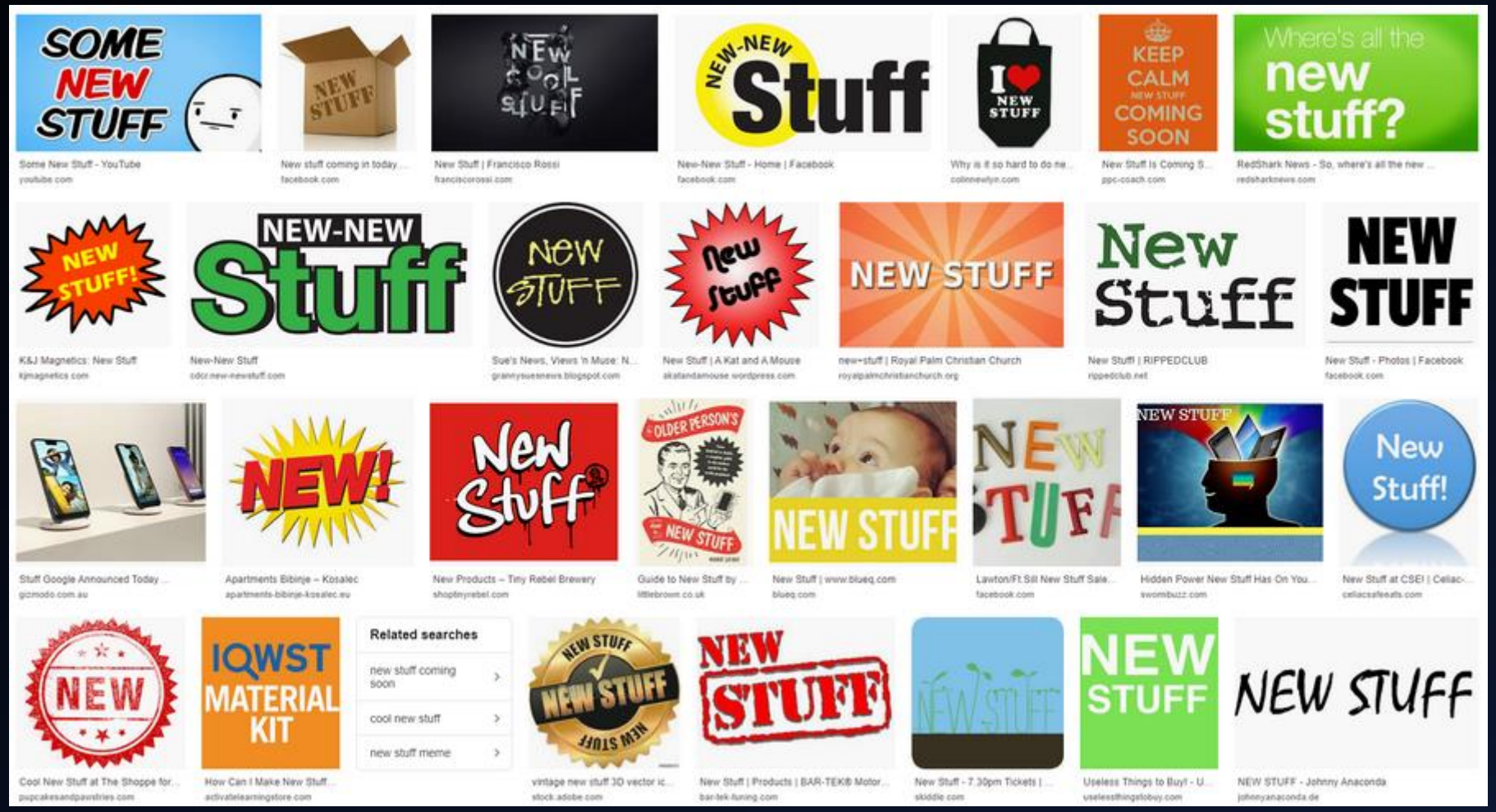
Branch: master | ThreatHunter-Playbook / signatures / sigma / | Create new file | Upload files | Find file | History

cyb3rward0g ThreatHunter Playbook 2.0 ... Latest commit e18bf20 on Aug 16

powershell_alternate_powershell_hosts.yml	ThreatHunter Playbook 2.0
powershell_remote_powershell_session.yml	ThreatHunter Playbook 2.0
sysmon_alternate_powershell_hosts_moduleload.yml	ThreatHunter Playbook 2.0
sysmon_alternate_powershell_hosts_pipe.yml	ThreatHunter Playbook 2.0
sysmon_createremotethread_loadlibrary.yml	ThreatHunter Playbook 2.0
sysmon_non_interactive_powershell_execution.yml	ThreatHunter Playbook 2.0
sysmon_powershell_execution_moduleload.yml	ThreatHunter Playbook 2.0
sysmon_powershell_execution_pipe.yml	ThreatHunter Playbook 2.0
sysmon_rdp_registry_modification.yml	ThreatHunter Playbook 2.0
sysmon_remote_powershell_session_network.yml	ThreatHunter Playbook 2.0
sysmon_remote_powershell_session_process.yml	ThreatHunter Playbook 2.0
sysmon_wdigest_registry_modification.yml	ThreatHunter Playbook 2.0
sysmon_wmi_module_load.yml	ThreatHunter Playbook 2.0
sysmon_wmiprvse_spawning_process.yml	ThreatHunter Playbook 2.0

win_ad_object_writedac_access.yml	ThreatHunter Playbook 2.0
win_ad_replication_non_machine_account.yml	ThreatHunter Playbook 2.0
win_ad_replication_user_backdoor.yml	ThreatHunter Playbook 2.0
win_dpapi_domain_backupkey_extraction.yml	ThreatHunter Playbook 2.0
win_dpapi_domain_masterkey_backup_attempt.yml	ThreatHunter Playbook 2.0
win_lsass_access_non_system_account.yml	ThreatHunter Playbook 2.0
win_non_interactive_powershell.yml	ThreatHunter Playbook 2.0
win_protected_storage_service_access.yml	ThreatHunter Playbook 2.0
win_remote_powershell_session.yml	ThreatHunter Playbook 2.0
win_sam_registry_hive_dump_via_reg_utility.yml	ThreatHunter Playbook 2.0
win_sam_registry_hive_handle_request.yml	ThreatHunter Playbook 2.0
win_scm_database_handle_failure.yml	ThreatHunter Playbook 2.0
win_scm_database_privileged_operation.yml	ThreatHunter Playbook 2.0
win_syskey_registry_access.yml	ThreatHunter Playbook 2.0
win_wmiprvse_spawning_process.yml	ThreatHunter Playbook 2.0

Outline – New Stuff



Outline – New Stuff

- T1064 - Scripting
  VBS Scripts

Scripting

Adversaries may use scripts to aid in operations and perform multiple actions that would otherwise be manual. Scripting is useful for speeding up operational tasks and reducing the time required to gain access to critical resources. Some scripting languages may be used to bypass process monitoring mechanisms by directly interacting with the operating system at an API level instead of calling other programs. Common scripting languages for Windows include VBScript and PowerShell but could also be in the form of command-line batch scripts.

Scripts can be embedded inside Office documents as macros that can be set to execute when files used in Spearphishing Attachment and other types of spearphishing are opened. Malicious embedded macros are an alternative means of execution than software exploitation through Exploitation for Client Execution, where adversaries will rely on macros being allowed or that the user will accept to activate them.

ID: T1064

Tactic: Defense Evasion, Execution

Platform: Linux, macOS, Windows

Permissions Required: User

Data Sources: Process monitoring, File monitoring, Process command-line parameters

Defense Bypassed: Process whitelisting, Data Execution Prevention, Exploit Prevention

Version: 1.0

Outline – New Stuff

- T1064 - Scripting
  VBS Scripts

Scripting

Adversaries may use scripts to aid in operations and perf
otherwise be manual. Scripting is useful for speeding up
time required to gain access to critical resources. Some s
bypass process monitoring mechanisms by directly intera
API level instead of calling other programs. Common scri
VBScript and PowerShell but could also be in the form of

Scripts can be embedded inside Office documents as ma
files used in Spearphishing Attachment and other types o
Malicious embedded macros are an alternative means of
through Exploitation for Client Execution, where adversari
or that the user will accept to activate them.

Procedure Examples

Name	Description
JCry	JCry has used VBS scripts. [61]
JHUHUGIT	JHUHUGIT uses a .bat file to execute a .dll. [27]
jRAT	jRAT has been distributed as HTA files with VBScript+JScript. [54]
Ke3chang	Ke3chang has used batch scripts in its malware to install persistence mechanisms. [95]
KeyBoy	KeyBoy uses Python and VBS scripts for installing files and performing execution. [60]
Keydnap	Keydnap uses Python for scripting to execute additional commands. [22]
Koadic	Koadic performs most of its operations using Windows Script Host (Jscript and VBScript) and runs arbitrary shellcode . [8]
Lazarus Group	A Destover-like variant used by Lazarus Group uses a batch file mechanism to delete its binaries from the system. [13]
Leafminer	Leafminer infected victims using JavaScript code. [90]
Leviathan	Leviathan has used multiple types of scripting for execution, including JavaScript, JavaScript Scriptlets in XML, and VBScript. [31]
Magic Hound	Magic Hound malware has used .vbs scripts for execution. [68]
menuPass	menuPass has used malicious macros embedded inside Office documents to execute files. [78] [79]
MoonWind	MoonWind uses batch scripts for various purposes, including to restart and uninstall itself. [11]
MuddyWater	MuddyWater has used VBScript and JavaScript files to execute its POWERSTATS payload. MuddyWater has also used Microsoft scriptlets, macros, and PowerShell scripts.[[69] [70] [71] [72] [21]
NanHaiShu	NanHaiShu executes additional Jscript and VBScript code on the victim's machine. [33]
NanoCore	NanoCore uses VBS and JavaScript files. [29]
NavRAT	NavRAT loads malicious shellcode and executes it in memory. [36]
OceanSalt	OceanSalt has been executed via malicious macros. [41]
OilRig	OilRig has used various types of scripting for execution, including .bat and .vbs scripts. The group has also used macros to deliver malware such as QUADAGENT and OopsIE. [75] [76] [25] [18] [77]
OopsIE	OopsIE creates and uses a VBScript as part of its persistent execution. [25] [26]

Why should I care about VBS scripts?

- VBS based Malware & RAT families
  - vjWorm
  - H-Worm / Houdini RAT
  - WSH-RAT (let's call it "Wish-RAT")
- Persistence methods using VBS scripts
  - Dropping VBS to Startup Folder ---------->
  - Dropping URL file to Startup Folder calling VBS
  - Used by many Malware families
    - NanoCore RAT
    - NetWire RAT
    - AdWind / JBifrost
    - …

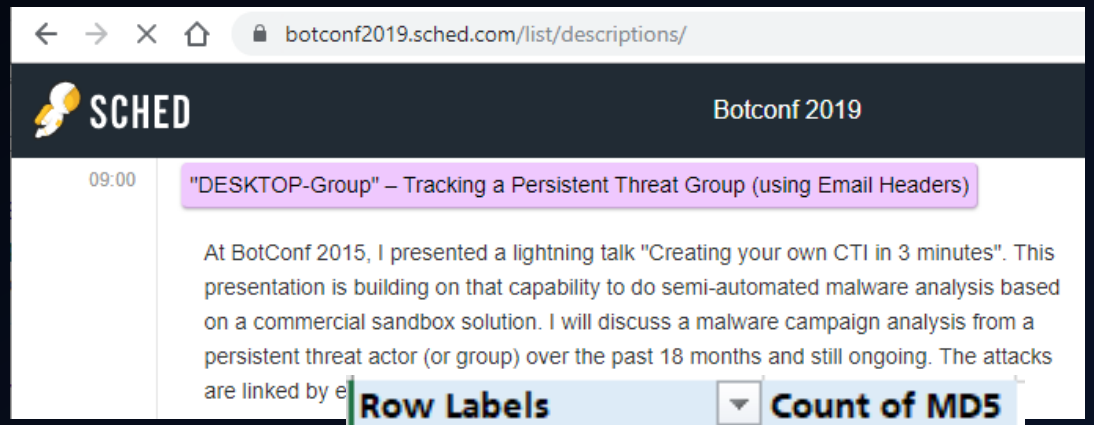
```
  2 AveMaria_infostealer
 30 crime_TrojanPSWFareit_mem

 28 memstr_HawkEye_Keylogger
  1 memstr_Predator_Pain
 36 memstr_rat_houdini
 85 memstr_rat_nanocore
 38 memstr_rat_remcos
  6 memstr_rat_wshrat

  1 pcap_java_rat_Luminosity_Link_p4ck3t
120 pcap_java_rat_adwind_JBifrost
 11 pcap_rat_Revenge_RAT
 16 pcap_rat_netwire
```

Why should I care about VBS scripts?

- VBS based Malware & RAT families
  - vjWorm
  - H-Worm / Houdini RAT
  - WSH-RAT (let's call it "Wish-RAT")
- Persistence methods using VBS scripts
  - Dropping VBS to Startup Folder
  - Dropping URL file to Startup Folder calling VBS
  - Used by many Malware families
    - NanoCore RAT
    - NetWire RAT
    - AdWind / JBifrost
    - …



Row Labels	Count of MD5
Adwind RAT	9
Imminent Monitor RAT	8
n/a (yet unknown)	6
NanoCore RAT	20
NanoCore RAT CyberGate RAT	1
NanoCore RAT VBS/Dunihi / Houdini RAT	1
Netwire RAT	5
VBS/Dunihi / Houdini RAT	12
WSH-RAT	2
WSH-RAT NanoCore RAT	1
(blank)	1
Grand Total	**66**

Why should I care about VBS scripts?

- vjWorm
  [JBX report link]

joesandbox.com/analysis/99489/0/html

JoeSandbox Cloud BASIC

Coded by v_B01 3/3

Overview Startup Dropped Domains / IPs Static Network Hooks Stats Behavior Disassembly

Analysis Process: wscript.exe PID: 3924 Parent PID: 4764

General

.vbs'

Show windows behavior

File Written

File Path	Offset	Length	Value	Ascii	Completion	Count	Source Address	Symbol
C:\Users\user\AppData\Roaming\Colis-1.vbs	0	11723	27 20 43 6f 64 65 64 20 62 79 20 76	' Coded by v_B01..On error resume next....j =	success or wait	1	7395A6D2	CopyFileW

Google

"Coded by v_B01"

All Images Videos Maps Shopping More Settings Tools

About 248 results (0.23 seconds)

[PDF] Automated Malware Analysis Report for Colis-1.vbs - Joe ...

https://www.joesandbox.com › analysis › pdf

Dec 23, 2018 - 11d470c0e7a5603da35414499ff1854f7dd9c8b142a772c f3719264c8c53689286eae1696a574b5af. File Content Preview: ' **Coded by v_B01**.

Why should I care about VBS scripts?

- vjWorm
  [JBX report link]

joesandbox.com/analysis/99489/0/html

JoeSandbox Cloud BASIC

Coded by v_B01 3/3

Overview Startup Dropped Domains / IPs Static Network Hooks Stats Behavior Disassembly

Analysis Process: wscript.exe PID: 3924 Parent PID: 4764

General

Path:	C:\Windows\SysWOW64\wscript.exe
Wow64 process (32bit):	true
Commandline:	'C:\Windows\System32\WScript.exe' 'C:\Users\user\Desktop\Colis-1.vbs'

Show windows behavior

File Written

File Path	Offset	Length	Value	Ascii
C:\Users\user\AppData\Roaming\Colis-1.vbs	0	11723	27 20 43 6f 64 65 64 20 62 79 20 76 5f 42 30 31 0d 0a	' Coded by v_B01..On error resume next....j = array(ChrW(87) &
C:\Users\user\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Colis-1.vbs	0	11723	27 20 43 6f 64 65 64 20 62 79 20 76 5f 42 30 31 0d 0a 4f 6e 20	' Coded by v_B01..On error resume next....j = array(ChrW(87) &

Completion	Count	Source Address	Symbol
success or wait	1	7395A6D2	CopyFileW

Why should I care about VBS scripts?

- joesandbox.com/analysis/99489/0/html

Persistence and Installation Behavior:

Windows Shell Script Host drops VBS files	
Source: C:\Windows\SysWOW64\wscript.exe	File created: C:\Users\user\AppData\Roaming\Colis-1.vbs
Source: C:\Windows\SysWOW64\wscript.exe	File created: C:\Users\user\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Colis-1.vbs

Boot Survival:

Drops VBS files to the startup folder	
Source: C:\Windows\SysWOW64\wscript.exe	File created: C:\Users\user\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Colis-1.vbs
Source: C:\Windows\System32\wscript.exe	File created: C:\Users\user\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Colis-1.vbs
Source: C:\Windows\System32\wscript.exe	File created: C:\Users\user\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Colis-1.vbs

Uses schtasks.exe or at.exe to add and modify task schedules	
Creates a start menu entry (Start Menu\Programs\Startup)	
Source: C:\Windows\SysWOW64\wscript.exe	File created: C:\Users\user\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Colis-1.vbs

Why should I care about VBS scripts?

- Persistence methods using VBS scripts
  - Dropping URL file to Startup Folder calling VBS

```
<path>C:\Users\user\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\fg1LK3rQ.url</path>
2019-08-25_7/dropped/fg1LK3rQ.url.0.dr
[InternetShortcut]
URL=file:///C:/Users/luketaylor/AppData/Roaming/WIN45/kI4Rg6US.vbs
--
<path>C:\Users\user\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\1sr5BcvD.url</path>
2019-08-29_3/dropped/1sr5BcvD.url.0.dr
[InternetShortcut]
URL=file:///C:/Users/luketaylor/AppData/Roaming/L2Schemas/O5tGbHD7.vbs
--
<path>C:\Users\user\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\F1Grb228.url</path>
2019-09-08_9/dropped/F1Grb228.url.0.dr
[InternetShortcut]
URL=file:///C:/Users/luketaylor/AppData/Roaming/Downloaded Program Files/3DwqH24f.vbs
--
<path>C:\Users\user\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\4W08Xh2V.url</path>
2019-09-10_30/dropped/4W08Xh2V.url.0.dr
[InternetShortcut]
URL=file:///C:/Users/LUKETA~1/AppData/Local/Temp/servicing/c3vLf44c.vbs
```

Why should I care about VBS scripts?

- Persistence methods using VBS scripts
  - Dropping URL file to Startup Folder calling VBS

```
<path>C:\Users\user\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\fglLK3rQ.url</path>
2019-08-25_7/dropped/fglLK3rQ.url.0.dr
[InternetShortcut]
URL=file:///C:/Users/luketaylor/AppData/Roaming/WIN45/kI4Rg6US.vbs
```

```
<path>C:\Users\user\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\lsr5BcvD.url</path>
2019-08-29_3/dropped/lsr5BcvD.url.0.dr
[InternetShortcut]
URL=file:///C:/Users/luketaylor/AppData/Roaming/L2Schemas/O5tGbHD7.vbs
--
<path>C:\Users\user\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\F1Grb228.url</path>
2019-09-08_9/dropped/F1Grb228.url.0.dr
[InternetShortcut]
URL=file:///C:/Users/luketaylor/AppData/Roaming/Downloaded Program Files/3DwqH24f.vbs
--
<path>C:\Users\user\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\4W08Xh2V.url</path>
2019-09-10_30/dropped/4W08Xh2V.url.0.dr
[InternetShortcut]
URL=file:///C:/Users/LUKETA~1/AppData/Local/Temp/servicing/c3vLf44c.vbs
```

Why should I care about VBS scripts?

- Persistence methods using VBS scripts
  - Dropping VBS to Startup Folder

```
  8 vbs-startup-folder_nanocore
  5 vbs-startup-folder_netwire
```

  - Dropping URL file to Startup Folder calling VBS

```
 32 url-startup-folder_nanocore
  7 url-startup-folder_netwire
```

  - NanoCore and NetWire samples analyzed in 2019

```
176 2019-samples-nanocore
 69 2019-samples-netwire
```

Hunting for suspicious VBS scripts

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ProcessCreate (cscript.exe OR wscript.exe)
| search (Image="*\\cscript.exe" OR Image="*\\wscript.exe")
| rex field=CommandLine ".*\\\\(?<VbsFilename_NoPath>[^\\\\:]*\.([cCvVwW][bBmMsS][aAdDeEfFhHsS]|[jJ][sS]))[^a-zA-Z].*"
| rex field=VbsFilename_NoPath ".*\.(?<VbsFilename_Ext>[a-zA-Z]{2,3})"
| eval VbsFilename_Ext=lower(VbsFilename_Ext)
| rex field=Image ".*\\\\(?<Image_fn>[^\\\\]*)"
| rex field=ParentImage ".*\\\\(?<ParentImage_fn>[^\\\\]*)"
| eval len_filename = len(VbsFilename_NoPath)
| search VbsFilename_NoPath!="Lohn Tabelle1.xlsx  *.vbs"
| stats
    dc(VbsFilename_NoPath)
    dc(Image_fn)
    dc(ParentImage_fn)
    dc(CommandLine) AS CmdLines
    dc(ComputerName) AS Clients
    count by VbsFilename_Ext
| sort -count
```

✓ 2,857,354 events (7/1/19 12:00:00.000 AM to 10/22/19 12:00:00.000 AM) No Event Sampling ▾ Job ▾

> 90 days → 2.8M events from > 25K endpoints (= all)

Hunting for suspicious VBS scripts

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ProcessCreate (cscript.exe OR wscript.exe)
| search (Image="*\\cscript.exe" OR Image="*\\wscript.exe")
| rex field=CommandLine ".*\\\\(?<VbsFilename_NoPath>[^\\\\:]*\.([cCvVwW][bBmMsS][aAdDeEfFhHsS]|[jJ][sS]))[^a-zA-Z].*"
| rex field=VbsFilename_NoPath ".*\.(?<VbsFilename_Ext>[a-zA-Z]{2,3})"
| eval VbsFilename_Ext=lower(VbsFilename_Ext)
| rex field=Image ".*\\\\(?<Image_fn>[^\\\\]*)"
| rex field=ParentImage ".*\\\\(?<ParentImage_fn>[^\\\\]*)"
```

VbsFilename_Ext	dc(VbsFilename_NoPath)	dc(Image_fn)	dc(ParentImage_fn)	CmdLines	Clients	count
vbs	4352	2	62	33687	25781	2849468
wsf	9	2	7	36	99	5073
js	29	2	15	353	217	1652
cmd	4	2	2	12	88	1148
vbe	1	2	2	6	2	13

✓ 2,857,354 events (7/1/19 12:00:00.000 AM to 10/22/19 12:00:00.000 AM) No Event Sampling ▾ Job ▾

> 90 days → 2.8M events from > 25K endpoints (= all)

Parent - Child Relationship

Hunting for suspicious VBS scripts (ChildProcess)

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ProcessCreate (cscript.exe OR wscript.exe)
| search (ParentImage="*\\cscript.exe" OR ParentImage="*\\wscript.exe")
| rex field=Image ".*\\\\(?<Image_fn>[^\\\\]*)"
| rex field=ParentImage ".*\\\\(?<ParentImage_fn>[^\\\\]*)"
| stats
    dc(CommandLine) AS CmdLines
    dc(ComputerName) AS Clients
    count by ParentImage_fn Image_fn
| sort -count
```

✓ 2,530,634 events (10/14/19 1:00:00.000 AM to 10/21/19 1:09:53.000 AM) No Event Sampling ▾

- Extract Image filename
- Extract ParentImage filename

Hunting for suspicious VBS scripts (ChildProcess)

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ProcessCreate (cscript.exe OR wscript.exe)
| search (ParentImage="*\\cscript.exe" OR ParentImage="*\\wscript.exe")
| rex field=Image ".*\\\\(?<Image_fn>[^\\\\]*)"
| rex field=ParentImage ".*\\\\(?<ParentImage_fn>[^\\\\]*)"
| stats
```

ParentImage_fn	Image_fn	CmdLines	Clients	count
cscript.exe	powershell.exe	5070	21641	497514
cscript.exe	WinHTTPproxy2MIF.exe	2493	21626	125796
cscript.exe	LocalPowerUsers2MIF.exe	2493	21623	125792
cscript.exe	Zebra2MIF.exe	2493	21624	125779
cscript.exe	UserProfileInfo2MIF.exe	2491	21601	125060
cscript.exe	HBAWWN2MIF.exe	2491	21603	125041

- Powershell most frequently executed from VBS scripts
- **2MIF.exe all have almost equal numbers, look related (legit)

Hunting for suspicious VBS scripts (ChildProcess)

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ProcessCreate (cscript.exe OR wscript.exe)
| search (ParentImage="*\\cscript.exe" OR ParentImage="*\\wscript.exe")
| rex field=Image ".*\\\\(?<Image_fn>[^\\\\]*)"
| rex field=ParentImage ".*\\\\(?<ParentImage_fn>[^\\\\]*)"
| stats
```

ParentImage_fn	Image_fn	CmdLines	Clients	count
cscript.exe	powershell.exe	5070	21641	497514

ParentImage_fn	Image_fn	CmdLines	Clients	count
cscript.exe	Uedit32.exe	1	1	1
cscript.exe	msiexec.exe	1	1	1
cscript.exe	regsvr32.exe	1	1	1
cscript.exe	rundll32.exe	1	1	1
wscript.exe	regsvr32.exe	1	1	1
cscript.exe	ovconfget.exe	25	2	301
wscript.exe	mstsc.exe	32	2	126

Hunting for suspicious VBS scripts (ChildProcess)

Time	Event
10/17/19 2:33:01.000 PM	10/17/2019 02:33:01 PM LogName=Microsoft-Windows-Sysmon/Operational SourceName=Microsoft-Windows-Sysmon EventCode=1 EventType=4 Type=Information

```
Image: C:\Windows\System32\regsvr32.exe
FileVersion: 10.0.17763.1 (WinBuild.160101.0800)
Description: Microsoft(C) Register Server
Product: Microsoft® Windows® Operating System
Company: Microsoft Corporation
CommandLine: "C:\WINDOWS\SYSTEM32\regsvr32.exe" \\[redacted]\package-development\Tools\Packaging\Scripts\Dll\Guid.dll /s
CurrentDirectory: \\[redacted]\Package-Development\Appl\WXX\Common\SAP-CRYSTALREPORTS-RUNTIME\2008.3\
```

```
ParentImage: C:\Windows\System32\wscript.exe
ParentCommandLine: "C:\WINDOWS\System32\WScript.exe" "\\[redacted]\Package-Development\Tools\Packaging\Scripts\AddSettingsToMsiScript.vbs"  "\\[redacted]\package-development\Appl\WXX\Common\SAP-CRYSTALREPORTS-RUNTIME\2008.3\001\SAP-CRYSTALREPORTS-RUNTIME_2008.3_C_001.msi"
```

```
TerminalSessionId: 1
IntegrityLevel: Medium
Hashes: MD5=DA0E9A7777D16AE18BD9C642A9F42223,IMPHASH=0235FF9A007804882636BCCCFB4D1A2F
ParentProcessGuid: {790d6656-5f7c-5da8-0000-0010f0258b06}
ParentProcessId: 11596
ParentImage: C:\Windows\System32\wscript.exe
ParentCommandLine: "C:\WINDOWS\System32\WScript.exe" "\\H058L5\Package-Development\Tools\Packaging\Scripts\AddSettingsToMsiScript.vbs"  "\\h058l5\package-development\Appl\WXX\Common\SAP-CRYSTALREPORTS-RUNTIME\2008.3\001\SAP-CRYSTALREPORTS-RUNTIME_2008.3_C_001.msi"
Collapse
```

Measure Length

Hunting for suspicious VBS scripts (filename length)

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ProcessCreate (cscript.exe OR wscript.exe)
| search (Image="*\\cscript.exe" OR Image="*\\wscript.exe") CommandLine="*\\Users\\*"
| rex field=CommandLine ".*\\\\(?<VbsFilename_NoPath>[^\\\\:]*\.([cCvVwW][bBmMsS][aAdDeEfFhHsS]|[jJ][sS]))[^a-zA-Z].*"
| rex field=VbsFilename_NoPath ".*\.(?<VbsFilename_Ext>[a-zA-Z]{2,3})"
| eval len_filename = len(VbsFilename_NoPath)
| where len_filename >= 30
| eval VbsFilename_NoPath = replace(VbsFilename_NoPath, "[ ]{10,}", " [many_SPACES_removed] ")
| eval len_filename_trimmed = len(VbsFilename_NoPath)
| stats
    values(Image)
    values(ParentImage)
    dc(CommandLine) AS CmdLines
    dc(ComputerName) AS Clients
    count by len_filename len_filename_trimmed VbsFilename_NoPath
| sort -len_filename
```

✓ 346 events (7/4/19 12:00:00.000 AM to 10/2/19 4:36:09.000 PM) No Event Sampling ▾

→ Look for (very) long filenames, e.g. to “hide” the real extension (double-ext.)

Hunting for suspicious VBS scripts (filename length)

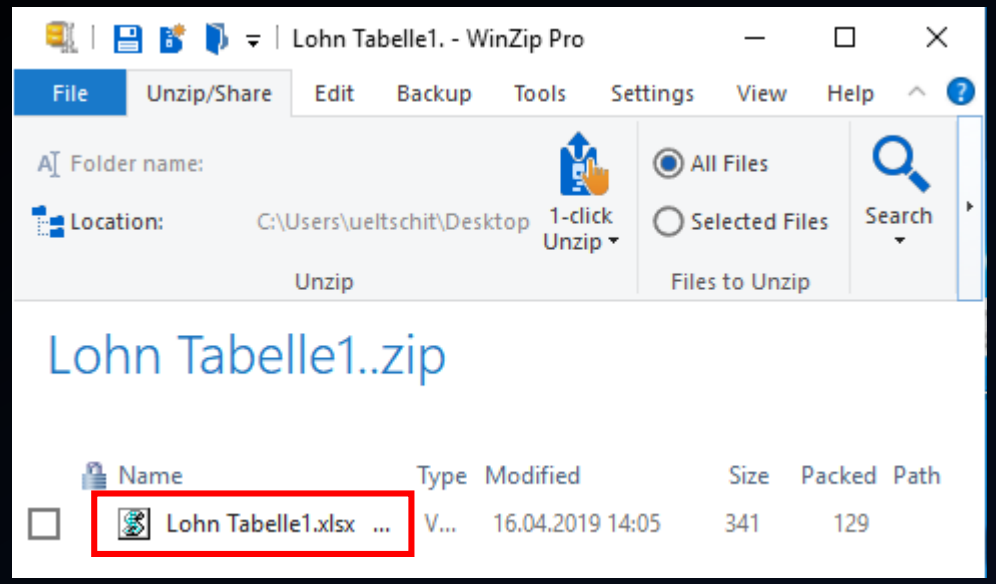
```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ProcessCreate (cscript.exe OR wscript.exe)
| search (Image="*\\cscript.exe" OR Image="*\\wscript.exe") CommandLine="*\\Users\\*"
| rex field=CommandLine ".*\\\\(?<VbsFilename_NoPath>[^\\\\:]*\.([cCvVwW][bBmMsS][aAdDeEfFhHsS]|[jJ][sS]))[^a-zA-Z].*"
```

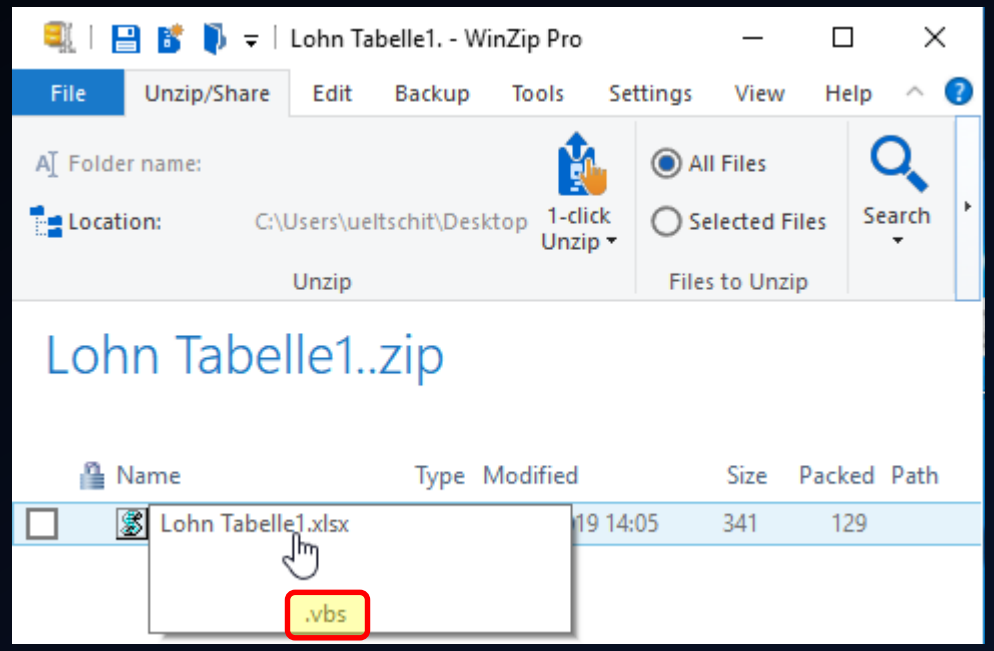
len_filename	len_filename_trimmed	VbsFilename_NoPath	values(Image)	values(ParentImage)	CmdLines	Clients	count
218	45	Lohn Tabelle1.xlsx [many_SPACES_removed] .vbs	C:\Windows\System32\cscript.exe	C:\Windows\explorer.exe	1	1	1
47	47	WPE-History.log -l WPE, Install, Config_WPE.vbs	C:\Windows\SysWOW64\cscript.exe	C:\Windows\SysWOW64\cscript.exe	3	1	16
46	46	Test2.vbs -o -l WPE, Install, Config_WPE.vbs	C:\Windows\SysWOW64\cscript.exe	C:\Windows\SysWOW64\cscript.exe	1	1	1
35	35	10/installer/server/initcluster.vbs	C:\Windows\System32\cscript.exe	C:\Users[redacted]Downloads\postgresql-10.10-1-windows-x64.exe	1	1	1
35	35	11/installer/server/initcluster.vbs	C:\Windows\System32\cscript.exe	C:\Users[redacted]\Downloads\postgresql-11.5-1-windows-x64.exe C:\Users\[redacted]\Downloads\postgresql-11.5-1-windows-x64.exe C:\Users\[redacted]\Downloads\postgresql-11.5-1-windows-x64.exe	3	3	3

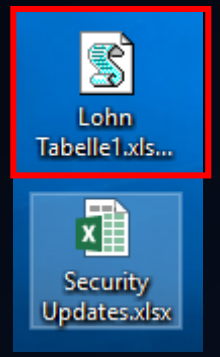
len_filename	len_filename_trimmed	VbsFilename_NoPath
218	45	Lohn Tabelle1.xlsx [many_SPACES_removed] .vbs

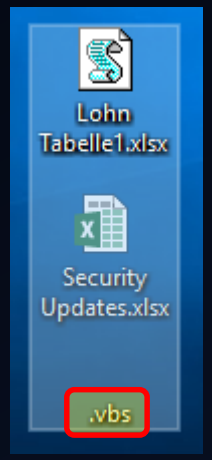
→ Look for (very) long filenames, e.g. to “hide” the real extension (double-ext.)

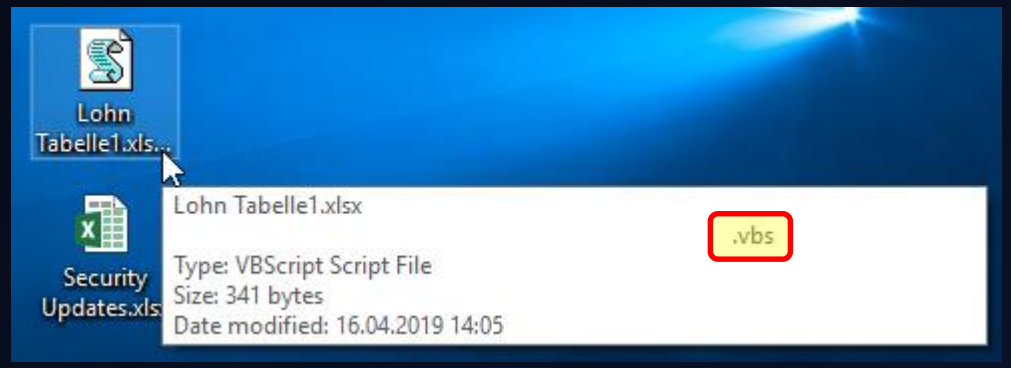
Hunting for suspicious VBS scripts (filename length)











→ Look for (very) long filenames, e.g. to “hide” the real extension (double-ext.)

Hunting for suspicious VBS scripts (susp FN / parents)

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ProcessCreate (cscript.exe OR wscript.exe)
| search (Image="*\\cscript.exe" OR Image="*\\wscript.exe") CommandLine="*\\Users\\*"
| rex field=CommandLine ".*\\\\(?<VbsFilename_NoPath>[^\\\\:]*\.([cCvVwW][bBmMsS][aAdDeEfFhHsS]|[jJ][sS]))[^a-zA-Z].*"
| rex field=VbsFilename_NoPath ".*\.(?<VbsFilename_Ext>[a-zA-Z]{2,3})"
| eval len_filename = len(VbsFilename_NoPath)
| search VbsFilename_NoPath!="Lohn Tabelle1.xlsx  *.vbs"
| stats
    values(VbsFilename_NoPath)
    values(Image)
    values(ParentImage)
    dc(CommandLine) AS CmdLines
    dc(ComputerName) AS Clients
    count by VbsFilename_Ext
| sort -count
```

✓ 22,688 events (7/4/19 12:00:00.000 AM to 10/2/19 4:13:54.000 PM) No Event Sampling ▾

Hunting for suspicious VBS scripts (susp FN / parents)

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ProcessCreate (cscript.exe OR wscript.exe)
| search (Image="*\\cscript.exe" OR Image="*\\wscript.exe") CommandLine="*\\Users\\*"
| rex field=CommandLine ".*\\\\(?<VbsFilename_NoPath>[^\\\\:]*\.([cCvVwW][bBmMsS][aAdDeEfFhHsS]|[jJ][sS]))[^a-zA-Z].*"
```

VbsFilename_Ext	values(VbsFilename_NoPath)	values(Image)	values(ParentImage)	CmdLines	Clients	count
vbs	10/installer/server/initcluster.vbs 10059.vbs 11/installer/server/initcluster.vbs ADtoIntranetBildschirmAusgabe.vbs AlleGLSEXSAT_kopieren - Kopie.vbs CommandPromptHere.vbs CopyForProduktion.vbs CopyForProduktionServer.vbs CopyForProduktionVDI.vbs CreateShortcut.vbs DownLoadTest.vbs Fichier.vbs Focus.vbs	C:\WINDOWS\SysWOW64\wscript.exe C:\WINDOWS\System32\cscript.exe C:\Windows\SysWOW64\cscript.exe C:\Windows\SysWOW64\wscript.exe C:\Windows\System32\cscript.exe C:\Windows\System32\wscript.exe C:\windows\System32\cscript.exe	C:\Program Files (x86)\Beyond Compare 3\BCompare.exe C:\Program Files (x86)\Hard Disk Sentinel\HDSentinel.exe C:\Program Files (x86)\Java\jre8\bin\java.exe C:\Program Files (x86)\Microsoft Office\Office16\OUTLOOK.EXE C:\Program Files (x86)\Microsoft SQL Server\130\Tools\Binn\ManagementStudio\Ssms.exe C:\Program Files (x86)\SAP\FrontEnd\SapGui\saplogon.exe C:\Program Files (x86)\SWIFT\Alliance Lite2\ConfigTool.exe C:\Program Files\7-Zip\7zFM.exe C:\Program Files\Avidemux 2.7 VC++ 64bits\Uninstall Avidemux VC++ 64bits.exe C:\Program Files\Java\jre8\bin\java.exe C:\Program Files\Mythicsoft\FileLocator Lite\FileLocatorLite.exe C:\Program Files\PowerISO\PowerISO.exe C:\Temp\7-ZipPortable\7zFM.exe	4586	2421	22065

```
    count by VbsFilename_Ext
| sort -count
```

✓ 22,688 events (7/4/19 12:00:00.000 AM to 10/2/19 4:13:54.000 PM) No Event Sampling

CmdLines	Clients	count
4586	2421	22065

Hunting for suspicious VBS scripts (susp FN / parents)

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ProcessCreate (cscript.exe OR wscript.exe)
| search (Image="*\\cscript.exe" OR Image="*\\wscript.exe") CommandLine="*\\Users\\*"
| rex
```

VbsFilename_Ext	values(Vb	count
vbs	10/instal 10059.vbs 11/instal ADtoIntra AlleGLSEX CommandPr CopyForPr CopyForPr CopyForPr CreateSho DownLoadT Fichier.v Focus.vbs	22065

```
14      c
15  | sort
```

✓ 22,688 ev

values(VbsFilename_NoPath)
10/installer/server/initcluster.vbs
10059.vbs
11/installer/server/initcluster.vbs
ADtoIntranetBildschirmAusgabe.vbs
AlleGLSEXSAT_kopieren - Kopie.vbs
CommandPromptHere.vbs
CopyForProduktion.vbs
CopyForProduktionServer.vbs
CopyForProduktionVDI.vbs
CreateShortcut.vbs
DownLoadTest.vbs
Fichier.vbs
Focus.vbs
IE_CheckRegUninstallKey.vbs
IT-Shop.vbs
KillTaskAP_ProgramList.vbs
Loop.vbs
Main-Srv.vbs
MakeSaveCopy.vbs
MinToMajus.vbs

values(ParentImage)
C:\Program Files (x86)\Beyond Compare 3\BCompare.exe
C:\Program Files (x86)\Hard Disk Sentinel\HDSentinel.exe
C:\Program Files (x86)\Java\jre8\bin\java.exe
C:\Program Files (x86)\Microsoft Office\Office16\OUTLOOK.EXE
C:\Program Files (x86)\Microsoft SQL Server\130\Tools\Binn\ManagementStudio\Ssms.exe
C:\Program Files (x86)\SAP\FrontEnd\SapGui\saplogon.exe
C:\Program Files (x86)\SWIFT\Alliance Lite2\ConfigTool.exe
C:\Program Files\7-Zip\7zFM.exe
C:\Program Files\Avidemux 2.7 VC++ 64bits\Uninstall Avidemux VC++ 64bits.exe
C:\Program Files\Java\jre8\bin\java.exe
C:\Program Files\Mythicsoft\FileLocator Lite\FileLocatorLite.exe
C:\Program Files\PowerISO\PowerISO.exe
C:\Temp\7-ZipPortable\7zFM.exe
C:\Users\[redacted]a\AppData\Local\Temp\SapSmartDel.exe
C:\Users\[redacted]1\AppData\Local\Temp\wpb_cloud\10.3.7.1269\producer.exe
C:\Users\[redacted]\Downloads\postgresql-11.5-1-windows-x64.exe
C:\Users\[redacted]\AppData\Local\Temp\SapSmartDel.exe
C:\Users\[redacted]\AppData\Local\Learnpulse\Screenpresso\Screenpresso.exe
C:\Users\[redacted]Downloads\postgresql-11.5-1-windows-x64.exe
C:\Users\[redacted]\Downloads\postgresql-11.5-1-windows-x64.exe

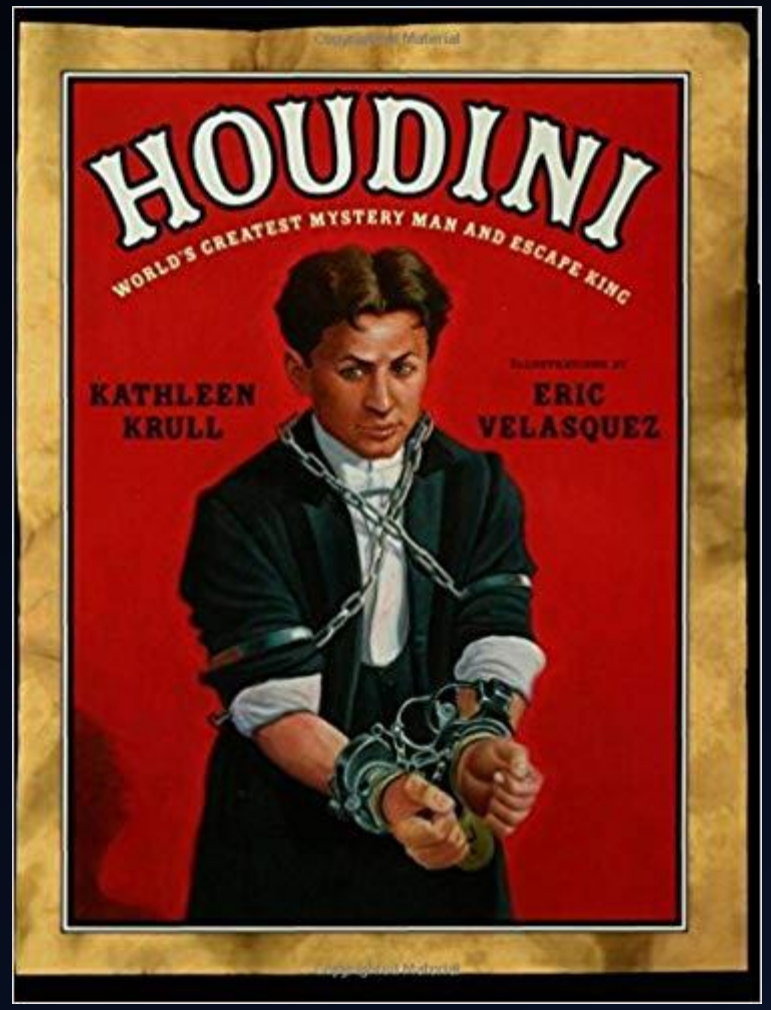
Hunting for suspicious VBS scripts (susp FN / parents)

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ProcessCreate (cscript.exe OR wscript.exe)
| search (Image="*\\cscript.exe" OR Image="*\\wscript.exe") CommandLine="*\\Users\\*"
```

VbsFilename_Ext	values(VbsFilename_NoPath)
wsf	GetReportsForAllGPOs.wsf GetReportsForGPO.wsf QueryFileInfo.wsf
VBS	CleanUpADGroups.VBS MoveitDoneSetStatus_JBL.VBS jbl-MoveitDoneSetStatus.VBS
js	9-es5.7983c45b9e29644b00d5.js Installer.js Javascript .js babel.config.js diff-doc.js diff-xls.js hostscript.js jquery.nanogallery2.core.min.js jquery.nanogallery2.js puttesession.js vue.config.js

values(ParentImage)
C:\Users\[redacted]1\AppData\Local\Temp\nsb74A0.tmp\eft-com-interface-installer.exe C:\Users\[redacted]AppData\Local\Temp\nsoAAB5.tmp\eft-com-interface-installer.exe C:\Windows\explorer.exe
C:\Users\[redacted]\AppData\Local\Adersoft\Vbsedit\x64\vbsedit.exe C:\Windows\explorer.exe
C:\PROGRA~1\AdoptOpenJDK\jdk-12.0.1.12-hotspot\bin\javaw.exe C:\Program Files (x86)\Common Files\Oracle\Java\javapath_target_4707000\javaw.exe C:\Program Files\Java\jre1.8.0_191\bin\javaw.exe C:\Program Files\Java\jre1.8.0_221\bin\javaw.exe C:\Program Files\TortoiseGit\bin\TortoiseGitProc.exe C:\Program Files\TortoiseSVN\bin\TortoiseProc.exe C:\Program Files\nodejs\node.exe C:\Windows\explorer.exe

Who is Houdini?



- vjWorm
- H-Worm
- WSH-RAT

Hunting for suspicious VBS scripts (Houdini detection)

```
08/09/2018 02:35:36 PM
LogName=Microsoft-Windows-Sysmon/Operational
SourceName=Microsoft-Windows-Sysmon
EventCode=1
EventType=4
Type=Information
ComputerName=[redacted]
User=NOT_TRANSLATED
Sid=S-1-5-18
SidType=0
TaskCategory=Process Create (rule: ProcessCreate)
OpCode=Info
RecordNumber=229494
Keywords=None
Message=Process Create:
UtcTime: 2018-08-09 12:35:36.423
ProcessGuid: {C2BF324B-3518-5B6C-0000-0010FB5AE503}
ProcessId: 6736
Image: C:\Windows\System32\wscript.exe
FileVersion: 5.812.10240.16384
Description: Microsoft ® Windows Based Script Host
Product: Microsoft ® Windows Script Host
Company: Microsoft Corporation
CommandLine: "C:\Windows\System32\wscript.exe" //B "C:\Users\[redacted]\AppData\Roaming\WZuMSKZzkg.vbs"
CurrentDirectory: C:\windows\system32\
User: POST\[redacted]
LogonGuid: {C2BF324B-E07B-5B6B-0000-00204E840C00}
LogonId: 0xC844E
TerminalSessionId: 1
IntegrityLevel: Medium
Hashes: MD5=03E5DFD4C18D75763EB6136CF22C7A84,IMPHASH=992748372A975981625241A4E77CA0B5
ParentProcessGuid: {C2BF324B-3518-5B6C-0000-00101B46E503}
ParentProcessId: 13876
ParentImage: C:\Windows\System32\cscript.exe
ParentCommandLine: "C:\windows\System32\CScript.exe"
    "C:\Users\[redacted]\AppData\Local\Temp\Temp1_Papu_Questionnaire.zip\Questionnaire_Secretariat_Papu.vbs"
```

Hunting for suspicious VBS scripts (Houdini detection)

```
08/09/2018 02:35:36 PM
LogName=Microsoft-Windows-Sysmon/Operational
SourceName=Microsoft-Windows-Sysmon
EventCode=1
EventType=4
Type=Information
ComputerName=[redacted]
User=NOT_TRANSLATED
Sid=S-1-5-18
SidType=0
TaskCategory=Process Create (rule: ProcessCreate)
```

```
ParentImage: C:\Windows\System32\cscript.exe
ParentCommandLine: "C:\windows\System32\CScript.exe"
    "C:\Users\[redacted]\AppData\Local\Temp\Temp1_Papu_Questionnaire.zip\Questionnaire_Secretariat_Papu.vbs"
```

```
ProcessId: 6736
```

```
Image: C:\Windows\System32\wscript.exe
FileVersion: 5.812.10240.16384
Description: Microsoft ® Windows Based Script Host
Product: Microsoft ® Windows Script Host
Company: Microsoft Corporation
CommandLine: "C:\Windows\System32\wscript.exe" //B "C:\Users\[redacted]\AppData\Roaming\WZuMSKZzkg.vbs"
```

```
LogonId: 0xC844E
TerminalSessionId: 1
IntegrityLevel: Medium
Hashes: MD5=03E5DFD4C18D75763EB6136CF22C7A84,IMPHASH=992748372A975981625241A4E77CA0B5
ParentProcessGuid: {C2BF324B-3518-5B6C-0000-00101B46E503}
ParentProcessId: 13876
ParentImage: C:\Windows\System32\cscript.exe
ParentCommandLine: "C:\windows\System32\CScript.exe"
    "C:\Users\[redacted]\AppData\Local\Temp\Temp1_Papu_Questionnaire.zip\Questionnaire_Secretariat_Papu.vbs"
```

Hunting for suspicious VBS scripts (Houdini detection)

alert_sysmon_houdini_infection_filecreate_5m

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ( FileCreate OR FileCreateStreamHash ) AND ( cscript.exe OR wscript.exe )
| search ( Image="*\\cscript.exe" OR Image="*\\wscript.exe" )
    TargetFilename="*\\Start Menu\\*\\Startup\*.vbs*" TargetFilename!="*:Zone.Identifier"
| strcat "TargetFilename: " TargetFilename ", CreationUtcTime: " CreationUtcTime ", Hash: " Hash  Details
| stats count by ComputerName TaskCategory ProcessId Image Details
| sort ComputerName TaskCategory ProcessId
```

alert_sysmon_houdini_infection_processcreate_5m

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ProcessCreate (cscript.exe OR wscript.exe)
| search (Image="*\\cscript.exe" OR Image="*\\wscript.exe")
    (CommandLine="*\\Users\\*\\AppData\\*.vbs*" OR CommandLine="*\\ProgramData\\*.vbs*")
    (ParentImage="*\\cscript.exe" OR ParentImage="*\\wscript.exe")
| stats count by ComputerName User ProcessId Image CommandLine ParentImage ParentCommandLine
```

Hunting for suspicious VBS scripts (Houdini detection)

alert_sysmon_houdini_infection_processcreate_5m

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ProcessCreate (cscript.exe OR wscript.exe)
| search (Image="*\\cscript.exe" OR Image="*\\wscript.exe")
```

ProcessId	Image	CommandLine	ParentImage	ParentCommandLine
1672	C:\Windows\System32\wscript.exe	"C:\Windows\System32\wscript.exe" //B "C:\Users\[redacted]\AppData\Local\Temp\Questionnaire_Secretariat_Papu.vbs"	C:\Windows\System32\cscript.exe	"C:\windows\System32\CScript.exe" "C:\Users\[redacted]\AppData\Local\Temp\Temp1_Papu_Questionnaire.zip\Questionnaire_Secretariat_Papu.vbs"
6736	C:\Windows\System32\wscript.exe	"C:\Windows\System32\wscript.exe" //B "C:\Users\[redacted]\AppData\Roaming\WZuMSKZzkg.vbs"	C:\Windows\System32\cscript.exe	"C:\windows\System32\CScript.exe" "C:\Users\[redacted]\AppData\Local\Temp\Temp1_Papu_Questionnaire.zip\Questionnaire_Secretariat_Papu.vbs"
12604	C:\Windows\System32\wscript.exe	"C:\Windows\System32\wscript.exe" //B "C:\Users\[redacted]\AppData\Roaming\WZuMSKZzkg.vbs"	C:\Windows\System32\wscript.exe	"C:\Windows\System32\wscript.exe" //B "C:\Users\[redacted]\AppData\Local\Temp\Questionnaire_Secretariat_Papu.vbs"

Hunting for suspicious VBS scripts (Houdini detection)

alert_sysmon_houdini_infection_filecreate_5m

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ( FileCreate OR FileCreateStreamHash ) AND ( cscript.exe OR wscript.exe )
| search ( Image="*\\cscript.exe" OR Image="*\\wscript.exe" )
    TargetFilename="*\\Start Menu\\*\\Startup\*.vbs*" TargetFilename!="*:Zone.Identifier"
```

TaskCategory	ProcessId	Image	Details	count
File created (rule: FileCreate)	6736	C:\Windows\System32\wscript.exe	TargetFilename: C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\WZuMSKZzkg.vbs, CreationUtcTime: 2018-08-09 12:35:36.583, Hash:	911
File created (rule: FileCreate)	13876	C:\windows\System32\CScript.exe	TargetFilename: C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Questionnaire_Secretariat_Papu.vbs, CreationUtcTime: 2018-08-09 12:35:36.533, Hash:	1
File stream created (rule: FileCreateStreamHash)	13876	C:\windows\System32\CScript.exe	TargetFilename: C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Questionnaire_Secretariat_Papu.vbs, CreationUtcTime: 2018-08-09 12:35:36.533, Hash: MD5=8248499D226833FECCB26DB4838EFC35,IMPHASH=00000000000000000000000000000000	1

Hunting for suspicious VBS scripts (Houdini detection)

alert_sysmon_houdini_infection_filecreate_5m

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ( FileCreate OR FileCreateStreamHash ) AND ( cscript.exe OR wscript.exe )
| search ( Image="*\\cscript.exe" OR Image="*\\wscript.exe" )
    TargetFilename="*\\Start Menu\\*\\Startup\*.vbs*" TargetFilename!="*:Zone.Identifier"
| strcat "TargetFilename: " TargetFilename ", CreationUtcTime: " CreationUtcTime ", Hash: " Hash  Details
```

TaskCategory	ProcessId	Image
File created (rule: FileCreate)	8224	C:\WINDOWS\System32\CScript.exe
File stream created (rule: FileCreateStreamHash)	8224	C:\WINDOWS\System32\CScript.exe

Details	count
TargetFilename: C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Colis-1.vbs, CreationUtcTime: 2018-11-05 11:55:14.908, Hash:	1
TargetFilename: C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Colis-1.vbs, CreationUtcTime: 2018-11-05 11:55:14.908, Hash: MD5=67FDAC001C11D11E0C35D35E5D30D6E0,IMPHASH=00000000000000000000000000000000	1

Hunting for suspicious VBS scripts (Houdini detection)

alert_sysmon_houdini_infection_filecreate_5m

```
(index=it_bapo OR index=it_sys
    ( FileCreate OR FileCreate
| search ( Image="*\\cscript.e
    TargetFilename="*\\Start M
| strcat "TargetFilename: " Ta
```

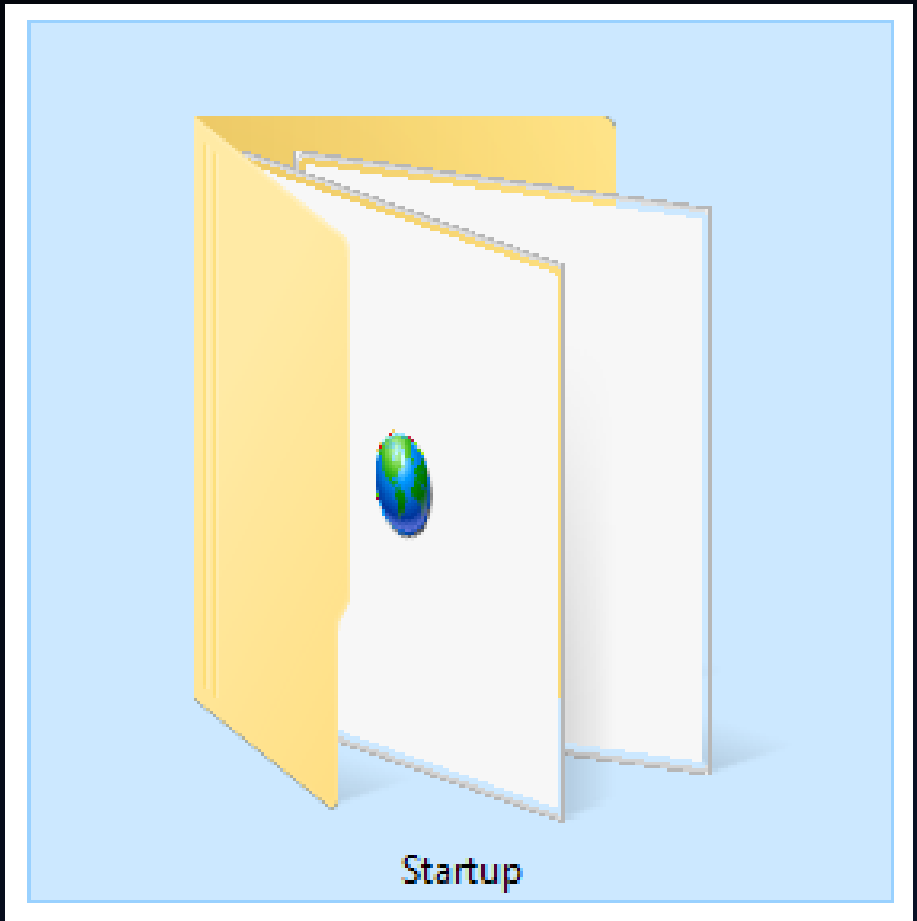
TaskCategory	ProcessId	Ima
File created (rule: FileCreate)	8224	C:\W
File stream created (rule: FileCreateStreamHash)	8224	C:\W

Created / dropped Files

C:\Users\user\AppData\Roaming\Colis-1.vbs	
Process:	C:\Windows\System32\wscript.exe
File Type:	ASCII text, with very long lines, with CRLF line terminators
Size (bytes):	11723
Entropy (8bit):	4.623575735612297
Encrypted:	false
MD5:	67FDAC001C11D11E0C35D35E5D30D6E0
SHA1:	7D0EA8F89384E73FE116E09AC985686A18F3F48B
SHA-256:	56680FD4AA08544BFAC5D3043ECE1E9162DF322959273510C3FE14457F04551F

Details	unt
TargetFilename Menu\Programs	
TargetFilename: C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Colis-1.vbs, CreationUtcTime: 2018-11-05 11:55:14.908, Hash: MD5=67FDAC001C11D11E0C35D35E5D30D6E0,IMPHASH=00000000000000000000000000000000	1

And now for something completely different…



Outline – New Stuff

- T1060 - Registry Run Keys / Startup Folder
  dropping VBS file in Startup folder

Registry Run Keys / Startup Folder

Adding an entry to the "run keys" in the Registry or startup folder will cause the program referenced to be executed when a user logs in. [1] These programs will be executed under the context of the user and will have the account's associated permissio

ID: T1060

Tactic: Persistence

Platform: Windows

System Requirements: HKEY_LOCAL_MACHINE keys require administrator access to create and modify

Permissions Required: User, Administrator

Data Sources: Windows Registry, File monitoring

CAPEC ID: CAPEC-270

Contributors: Oddvar Moe, @oddvarmoe

Version: 1.0

Outline – New Stuff

- T1060 - Registry Run Keys / Startup Folder
  dropping VBS file in Startup folder

Registry

Adding an entry to the

[1] These programs wi

fuzzysecurity.com/tutorials/19.html

Windows Startup Folder

The final technique is a classic, all windows versions, going back to "Windows 3", have starup directories. Any binary, script or application shortcut which is put in that directory will be executed when the user logs on to the system.

Links:

List Of Major Windows Versions - here

Startup Directories:

```
# Windows NT 6.0 - 10.0 / All Users
%SystemDrive%\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup

# Windows NT 6.0 - 10.0 / Current User
%SystemDrive%\Users\%UserName%\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup

# Windows NT 5.0 - 5.2
%SystemDrive%\Documents and Settings\All Users\Start Menu\Programs\Startup

# Windows NT 3.5 - 4.0
%SystemDrive%\WINNT\Profiles\All Users\Start Menu\Programs\Startup
```

_MACHINE keys
and modify
strator
monitoring
noe

Hunting for suspicious VBS scripts (Startup Folder)

```
(index=it_bapo OR index=it_sysmon) SourceName="Microsoft-Windows-Sysmon" ProcessCreate
    ( cmd.exe OR cscript.exe OR wscript.exe ) AND startup
| search ( Image="*\\cmd.exe" OR Image="*\\cscript.exe" OR Image="*\\wscript.exe" ) AND
   CommandLine="*\\startup\\*"
| rex field=Image ".*\\\\(?<Image_fn>[^\\\\]*)"
| rex field=ParentImage ".*\\\\(?<ParentImage_fn>[^\\\\]*)"
| rex mode=sed field=CommandLine "s/[\\\\]Users[\\\\][a-zA-Z0-9~]+[\\\\]/\\\\Users\\\\[redacted]\\\\/g"
| stats dc(ComputerName) AS dc_CN count by ParentImage_fn Image_fn CommandLine
| sort -Image_fn -dc_CN
```

✓ 67 events (8/28/19 12:00:00.000 AM to 10/27/19 9:11:41.000 PM) No Event Sampling ▾

Hunting for suspicious VBS scripts (Startup Folder)

```
(index=it_bapo OR index=it_sysmon) SourceName="Microsoft-Windows-Sysmon" ProcessCreate
    ( cmd.exe OR cscript.exe OR wscript.exe ) AND startup
| search ( Image="*\\cmd.exe" OR Image="*\\cscript.exe" OR Image="*\\wscript.exe" ) AND
    CommandLine="*\\startup\\*"
| rex field=Image ".*\\\\(?<Image_fn>[^\\\\]*)"
| rex field=ParentImage ".*\\\\(?<ParentImage_fn>[^\\\\]*)"
```

ParentImage_fn	Image_fn	CommandLine	dc_CN	count
explorer.exe	cscript.exe	"C:\WINDOWS\System32\CScript.exe" "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\test1.vbs"	1	7
explorer.exe	cscript.exe	"C:\WINDOWS\System32\CScript.exe" "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\update.vbs"	1	2
explorer.exe	cmd.exe	C:\Windows\system32\cmd.exe /c ""C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\blackdress_stageless_x64.cmd" "	5	5
explorer.exe	cmd.exe	C:\WINDOWS\system32\cmd.exe /c ""C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp\LogonScript.cmd" "	1	8
explorer.exe	cmd.exe	C:\WINDOWS\system32\cmd.exe /c ""C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp\test1.bat" "	1	8
explorer.exe	cmd.exe	C:\WINDOWS\system32\cmd.exe /c ""C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp\test1.cmd" "	1	8
explorer.exe	cmd.exe	C:\WINDOWS\system32\cmd.exe /c ""C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\test1.bat" "	1	7
explorer.exe	cmd.exe	C:\WINDOWS\system32\cmd.exe /c ""C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\test1.cmd" "	1	7
explorer.exe	cmd.exe	C:\Windows\system32\cmd.exe /c ""C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup\test1.bat" "	1	3
explorer.exe	cmd.exe	C:\Windows\system32\cmd.exe /c ""C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup\test1.cmd" "	1	3
explorer.exe	cmd.exe	C:\Windows\system32\cmd.exe /c ""C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\LogonScript.cmd" "	1	3
explorer.exe	cmd.exe	C:\Windows\system32\cmd.exe /c ""C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\test1.bat" "	1	3
explorer.exe	cmd.exe	C:\Windows\system32\cmd.exe /c ""C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\test1.cmd" "	1	3

Hunting for suspicious VBS scripts (Startup Folder)

```
(index=it_bapo OR index=it_sysmon) SourceName="Microsoft-Windows-Sysmon" ProcessCreate
    ( c
| searc
    Con
| rex f
```

ParentImage_fn	Image_fn		count
explorer.exe	cscript.e	1	7
explorer.exe	cscript.e	1	2
explorer.exe	cmd.exe	5	5
explorer.exe	cmd.exe	1	8
explorer.exe	cmd.exe	1	8
explorer.exe	cmd.exe	1	8
explorer.exe	cmd.exe	1	7
explorer.exe	cmd.exe	1	7
explorer.exe	cmd.exe	1	3
explorer.exe	cmd.exe	1	3
explorer.exe	cmd.exe	1	3
explorer.exe	cmd.exe	1	3
explorer.exe	cmd.exe	1	3

```
" "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\test1.vbs"
" "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\update.vbs"
"C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\blackdress_stageless_x64.cmd"
"C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp\LogonScript.cmd" "
"C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp\test1.bat" "
"C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp\test1.cmd" "
"C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\test1.bat" "
"C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\test1.cmd" "
"C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup\test1.bat" "
"C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup\test1.cmd" "
"C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\LogonScript.cmd" "
"C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\test1.bat" "
"C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\test1.cmd" "
```

Hunting for suspicious VBS scripts (Startup Folder)

```
(index=it_bapo OR index=it_sysmon) SourceName="Microsoft-Windows-Sysmon" ProcessCreate
    ( cmd.exe OR cscript.exe OR wscript.exe ) AND startup
| search ( Image="*\\cmd.exe" OR Image="*\\cscript.exe" OR Image="*\\wscript.exe" ) AND
    CommandLine="*\\startup\\*"
| rex field=Image ".*\\\\(?<Image_fn>[^\\\\]*)"
```

CommandLine	dc_CN	count
"C:\WINDOWS\System32\CScript.exe" "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\absolute-path-to-clipboard.vbs"	1	6
"C:\WINDOWS\System32\CScript.exe" "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\sshagent.vbs"	1	6
"C:\WINDOWS\System32\CScript.exe" "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\sshd.vbs"	1	6
"C:\WINDOWS\System32\CScript.exe" "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\test1.vbs"	1	7
"C:\WINDOWS\System32\CScript.exe" "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\update.vbs"	1	2
"C:\WINDOWS\System32\CScript.exe" "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\xserver.vbs"	1	6

Hunting for suspicious VBS scripts (Startup Folder)

```
(index=it_bapo OR index=it_sysmon) SourceName="Microsoft-Windows-Sysmon" ProcessCreate
    ( cmd.exe OR cscript.exe OR wscript.exe ) AND startup
| search ( Image="*\\cmd.exe" OR Image="*\\cscript.exe" OR Image="*\\wscript.exe" ) AND
    CommandLine="*\\startup\\*"
| rex field=Image ".*\\\\(?<Image_fn>[^\\\\]*)"
```

CommandLine	dc_CN	count
"C:\WIN		6
"C:\WIN		6
"C:\WIN		6
"C:\WIN "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\absolute-path-to-clipboard.vbs"		7
"C:\WIN "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\sshagent.vbs"		2
"C:\WIN "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\sshd.vbs"		6
"C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\test1.vbs"		
"C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\update.vbs"		
"C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\xserver.vbs"		

Hunting for suspicious VBS scripts (Startup Folder)

```
(index=it_bapo OR index=it_sysmon) SourceName="Microsoft-Windows-Sysmon" ProcessCreate
    ( cmd.exe OR cscript.exe OR wscript.exe ) AND startup
| search ( Image="*\\cmd.exe" OR Image="*\\cscript.exe" OR Image="*\\wscript.exe" ) AND
    CommandLine="*\\startup\\*"
| rex field=Image ".*\\\\(?<Image_fn>[^\\\\]*)"
```

CommandLine	dc_CN	count
C:\WINDOWS\system32\cmd.exe /c ""C:\Users\ ams\Startup\FW.bat" "	1	14
C:\WINDOWS\system32\cmd.exe /c ""C:\Users\ ams\Startup\LW_Verbinden.bat" "	1	27
C:\WINDOWS\system32\cmd.exe /c ""C:\Users\ ams\Startup\Laufwerke.bat" "	1	2
C:\WINDOWS\system32\cmd.exe /c ""C:\Users\ ams\Startup\LogonScript.cmd" "	1	5
C:\WINDOWS\system32\cmd.exe /c ""C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Netzlaufwerk.bat" "	1	9
C:\WINDOWS\system32\cmd.exe /c ""C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\PresentationMode.bat" "	1	8
C:\WINDOWS\system32\cmd.exe /c ""C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\PublicKey.bat" "	1	4
C:\WINDOWS\system32\cmd.exe /c ""C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Startup Chrome.bat" "	1	8
C:\WINDOWS\system32\cmd.exe /c ""C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Substitude_B.bat" "	1	13
C:\WINDOWS\system32\cmd.exe /c ""C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\coherence.cmd" "	1	1
C:\WINDOWS\system32\cmd.exe /c ""C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\del_public_doc.bat" "	1	4
C:\WINDOWS\system32\cmd.exe /c ""C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\fw.bat" "	1	5

Don’t stop at VBS
Remember CMD, BAT, etc.
All script types

Hunting for suspicious VBS scripts (Startup Folder)

alert_sysmon_persistence_startup_folder_5m

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational" ProcessCreate
    ( update.exe OR update.vbs OR "\\Windows\\Start Menu\\Programs\\Startup\\" )
    ( cscript.exe OR wscript.exe OR update.exe )
| search ( CommandLine="*\\Windows\\Start Menu\\Programs\\Startup\\*.vbs*" OR
           CommandLine="*\\Windows\\Start Menu\\Programs\\Startup\\*.exe*" OR
           ParentCommandLine="*\\Windows\\Start Menu\\Programs\\Startup\\*.vbs*" OR
           Image="*\\appdata\\update.exe" ) AND
         ( Image="*\\cscript.exe" OR Image="*\\wscript.exe" OR Image="*\\appdata\\update.exe" )

         NOT (ParentImage=[redacted]
         Image!="*\\Startu
         CommandLine!="*\\
         CommandLine!="*\\
         CommandLine!="*\\
         CommandLine!="*\\
         CommandLine!="*\\
| rex field=Image ".*\\\\(?<Image_filename>[^\\\\]+)"
| rex field=ParentImage ".*\\\\(?<ParentImage_filename>[^\\\\]+)"
| stats values(ParentCommandLine) count by CommandLine
```

✓ 12 events (8/28/19 12:00:00.000 AM to 10/27/19 11:17:00.000 PM) No Event Sampling ▾

Hunting for suspicious VBS scripts (Startup Folder)

alert_sysmon_persistence_startup_folder_5m

```
(index=it_bapo OR index=it_sysmon) sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational" ProcessCreate
    ( update.exe OR update.vbs OR "\\Windows\\Start Menu\\Programs\\Startup\\" )
    ( cscript.exe OR wscript.exe OR update.exe )
| search ( CommandLine="*\\Windows\\Start Menu\\Programs\\Startup\\*.vbs*" OR
         CommandLine="*\\Windows\\Start Menu\\Programs\\Startup\\*.exe*" OR
         ParentCommandLine="*\\Windows\\Start Menu\\Programs\\Startup\\*.vbs*" OR
```

CommandLine	values(ParentCommandLine)	count
"C:\Users\[redacted]\AppData\Roaming\appdata\update.exe"	C:\WINDOWS\explorer.exe /factory,{ceff45ee-c862-41de-aee2-a022c81eda92} -Embedding	1
"C:\WINDOWS\System32\CScript.exe" "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\test1.vbs"	C:\WINDOWS\Explorer.EXE	7
"C:\WINDOWS\System32\CScript.exe" "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\update.vbs"	C:\WINDOWS\Explorer.EXE C:\WINDOWS\explorer.exe /factory,{ceff45ee-c862-41de-aee2-a022c81eda92} -Embedding	2
C:\Users\[redacted]\AppData\Roaming\appdata\update.exe	"C:\WINDOWS\System32\CScript.exe" "C:\Users\[redacted]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\update.vbs"	2


DNS
Tunnel
An underground or underwater passage.

Outline – New Stuff

- T1071 / Standard Application Layer Protocol
  Command and Control via DNS

Standard Application Layer Protocol

Adversaries may communicate using a common, standardized application layer protocol such as HTTP, HTTPS, SMTP, or DNS to avoid detection by blending in with existing traffic. Commands to the remote system, and often the results of those commands, will be embedded within the protocol traffic between the client and server.

For connections that occur internally within an enclave (such as those between a proxy or pivot node and other nodes protocols are RPC, SSH, or RDP.

ID: T1071

Tactic: Command And Control

Platform: Linux, macOS, Windows

Data Sources: Packet capture, Netflow/Enclave netflow, Process use of network, Malware reverse engineering, Process monitoring

Requires Network: Yes

Version: 1.0

Hunting for C&C via DNS

```
21-Oct-2019 17:22:53.875 queries: info: client @0x7fe497ff9cc0 10.226.160.152#58359 (post.clien[REDACTED].ch.p
ost.ch): query: post.clien[REDACTED].ch.post.ch IN A + (10.1.102.12)
```

```
21-Oct-2019 17:12:00.362 queries: info: client @0x7f79c000b730 172.27.136.16#61732 (post.12ebb1341fd83c237c560
c44d5843939ac8e5616a6f62d51aaf939530.16cd5041c.62796.dns.clien[REDACTED].ch): query: post.12ebb1341fd83c237c56
0c44d5843939ac8e5616a6f62d51aaf939530.16cd5041c.62796.dns.clien[REDACTED].ch IN A + (172.27.59.12)
```

```
21-Oct-2019 17:12:00.354 queries: info: client @0x7f1ffc5d5b50 172.27.136.16#55605 (post.120.06cd5041c.62796.d
ns.clien[REDACTED].ch): query: post.120.06cd5041c.62796.dns.clien[REDACTED].ch IN A + (172.27.59.11)
```

```
21-Oct-2019 17:12:00.345 queries: info: client @0x7f79c839b060 172.27.136.16#65282 (api.175f2895.62796.dns.cli
en[REDACTED].ch): query: api.175f2895.62796.dns.clien[REDACTED].ch IN TXT + (172.27.59.12)
```

- Do you have DNS logs with client (source) information? (could be useful!)

Hunting for C&C via DNS

```
index=it_dns sourcetype=clientlog host!=[redacted]
| bin _time span=1h
| rex field=_raw ".*info: client (?<c_ip>[0-9.]*)#.* query: (?<query>[^ ]*) IN (?<type>[A-Z]*) .*"
| rex field=query ".*\.(?<domain_2nd>[^\.]*)\.(?<domain_1st>[^\.]*\.[^\.]*)"
| search query="*.*.*" c_ip!=[redacted]
| eval len_query = len(query) | eval len_dom1 = len(domain_1st) | eval len_dom2 = len(domain_2nd)
| eval len_c2 = len_query - len_dom1 - len_dom2
| where len_dom1 > [redacted]
| stats avg(len_query) AS AVG_LEN_QUERY avg(len_c2) AS AVG_LEN_C2 dc(query) AS DC_QUERIES
    values(domain_2nd) AS DOMAIN_2ND values(type) AS TYPES
    count by _time c_ip domain_1st
| where DC_QUERIES > 50 and AVG_LEN_C2 > [redacted]
| sort -DC_QUERIES
```

Hunting for C&C via DNS

```
index=it_dns sourcetype=clientlog host!=[redacted]
```

```
| bin _time span=1h
| rex field=_raw ".*info: client (?<c_ip>[0-9.]*)#.* query: (?<query>[^ ]*) IN (?<type>[A-Z]*) .*"
| rex field=query ".*\.(?<domain_2nd>[^\.]*)\.(?<domain_1st>[^\.]*\.[^\.]*)"
| search query="*.*.*" c_ip!="[redacted]"
| eval len_query = len(query) | eval len_dom1 = len(domain_1st) | eval len_dom2 = len(domain_2nd)
| eval len_c2 = len_query - len_dom1 - len_dom2
| where len_dom1 > [redacted]
| stats avg(len_query) AS AVG_LEN_QUERY avg(len_c2) AS AVG_LEN_C2 dc(query) AS DC_QUERIES
    values(domain_2nd) AS DOMAIN_2ND values(type) AS TYPES
    count by _time c_ip domain_1st
| where DC_QUERIES > 50 and AVG_LEN_C2 > [redacted]
| sort -DC_QUERIES
```

```
    values(domain_2nd) AS DOMAIN_2ND values(type) AS TYPES
    count by _time c_ip domain_1st
| where DC_QUERIES > 50 and AVG_LEN_C2 > 10
| sort -DC_QUERIES
```

Hunting for C&C via DNS

- query = "subdom3.subdom2.subdom1.domain.tld"
  - domain_1st = "domain.tld"
  - domain_2nd = "subdom1"
  - len_query = 34
  - len_dom1 = 10
  - len_dom2 = 7
  - len_c2 = len_query - len_dom1 - len_dom2 = 34 - 10 - 7 = 17
  - AVG_LEN_QUERY = avg(len_query)
  - AVG_LEN_C2 = avg(len_c2)
  - DC_QUERIES = distinct_count(query)
  - Within 1h span: DC_QUERIES > 50 and AVG_LEN_C2 >

Hunting for C&C via DNS

_time	c_ip	domain_1st	AVG_LEN_QUERY	AVG_LEN_C2	DC_QUERIES	DOMAIN_2ND	TYPES	count
Mon Jun 10 10:00:00 2019	10.1.96.150	clien[redacted].ch	85.63697592187236	67.63697592187236	3359	dns	A TXT	5939

_time	c_ip	domain_1st	AVG_LEN_QUERY	AVG_LEN_C2	DC_QUERIES	DOMAIN_2ND	TYPES	count
Mon Jun 24 11:00:00 2019	10.1.96.46	clien[redacted].ch	94.84922728986054	76.84922728986054	1709	dns	A TXT	2653
Mon Jun 24 11:00:00 2019	10.1.104.12	clien[redacted].ch	86.94143780290791	68.94143780290791	1551	dns	A TXT	2476
Mon Jun 24 11:00:00 2019	10.226.160.2	clien[redacted].ch	203.35014272121788	185.35014272121788	1013	dns	A TXT	1051

_time	c_ip	domain_1st	AVG_LEN_QUERY	AVG_LEN_C2	DC_QUERIES	DOMAIN_2ND	TYPES	count
Mon Jun 24 13:00:00 2019	10.1.96.46	clien[redacted].ch	35.307479224376735	17.30747922437673	965	dns	A TXT	3971
Mon Jun 24 13:00:00 2019	10.1.104.12	clien[redacted].ch	38.31662548535122	20.316625485351217	931	dns	A TXT	2833

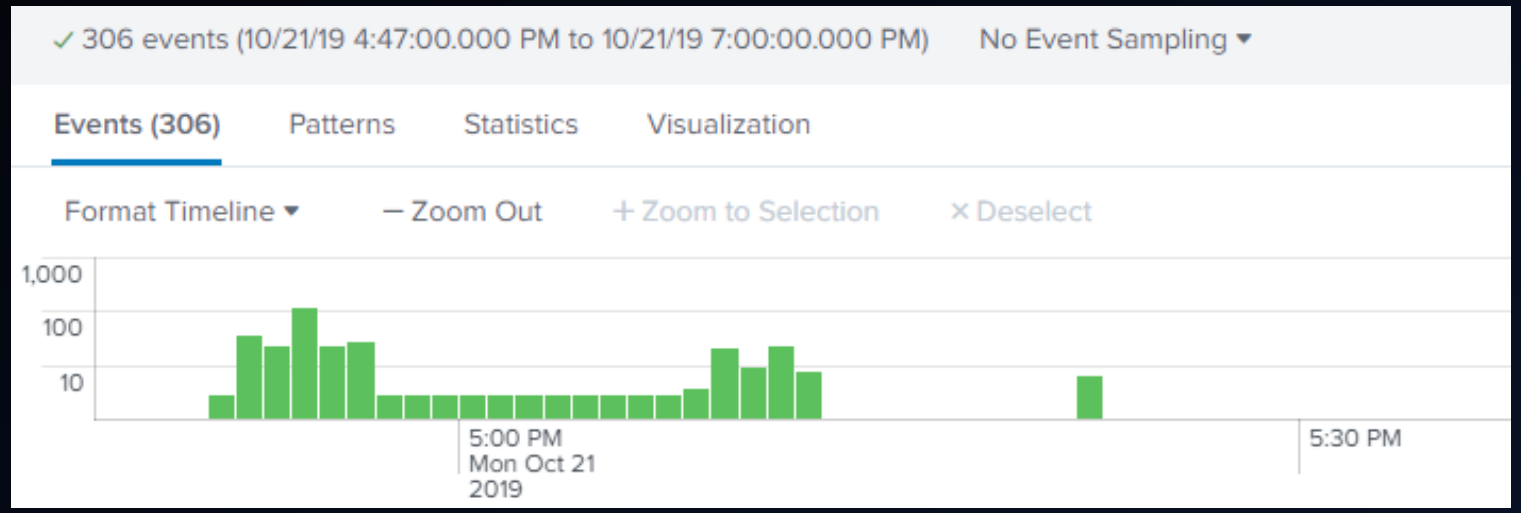
_time	c_ip	domain_1st	AVG_LEN_QUERY	AVG_LEN_C2	DC_QUERIES	DOMAIN_2ND	TYPES	count
Thu Jul 4 16:00:00 2019	10.1.96.46	clien[redacted].ch	34.82745419742959	16.82745419742959	1127	dns	A TXT	3657

- Within 1h span: DC_QUERIES > 50 and AVG_LEN_C2 > [redacted]

Hunting for C&C via DNS

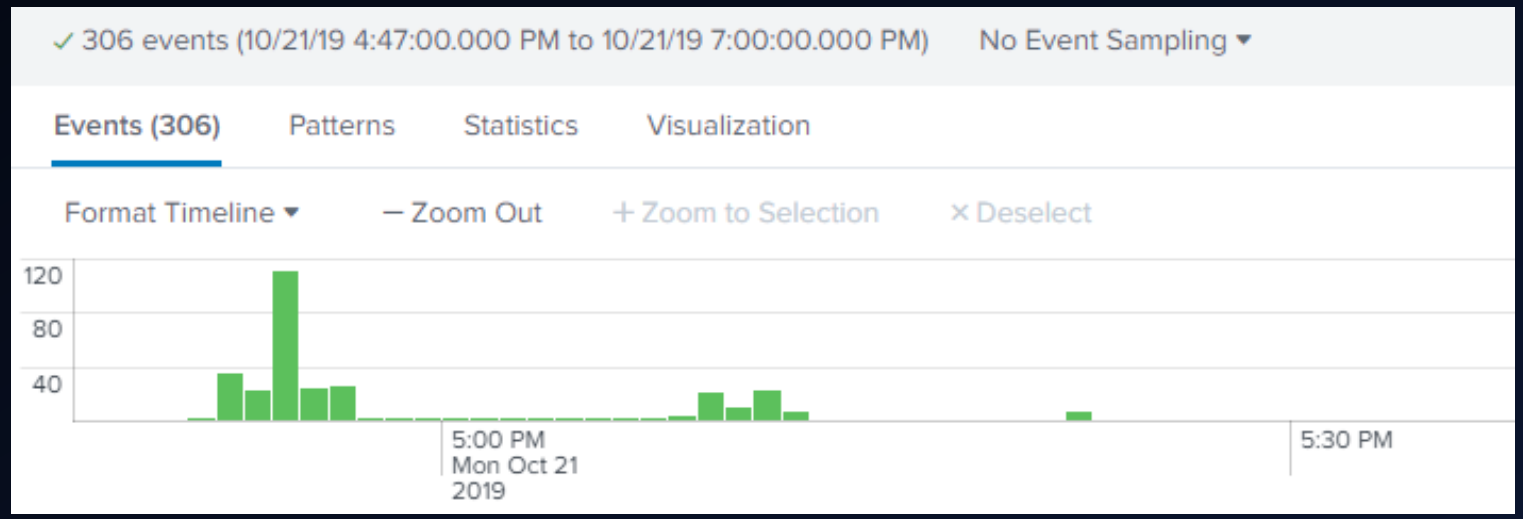
Top:

Log scale



Bottom:

Linear scale



Outline

- Introduction
- 1st of 3 techniques from MITRE ATT&CK

Windows Management Instrumentation Event Subscription	
Technique	
ID	T1084
Tactic	Persistence
Platform	Windows
Permissions Required	Administrator, SYSTEM
Data Sources	WMI Objects

WMI Event Subscription (Persistence)

https://attack.mitre.org/wiki/Technique/T1084

ATT&CK™
Adversarial Tactics, Techniques & Common Knowledge

Page | Discussion | Read | View form | View history | Search enterprise

Last 5 Pages Viewed:

Main page
Help
Contribute
References
Using the API
Contact us
Terms of Use

Tactics
- Initial Access
- Execution
- Persistence
- Privilege Escalation
- Defense Evasion
- Credential Access
- Discovery
- Lateral Movement
- Collection
- Exfiltration
- Command and Control

Techniques
- Technique Matrix

Windows Management Instrumentation Event Subscription

Windows Management Instrumentation (WMI) can be used to install event filters, providers, consumers, and bindings that execute code when a defined event occurs. Adversaries may use the capabilities of WMI to subscribe to an event and execute arbitrary code when that event occurs, providing persistence on a system. Adversaries may attempt to evade detection of this technique by compiling WMI scripts.[1] Examples of events that may be subscribed to are the wall clock time or the computer's uptime.[2] Several threat groups have reportedly used this technique to maintain persistence.[3]

Windows Management Instrumentation Event Subscription	
Technique	
ID	T1084
Tactic	Persistence
Platform	Windows
Permissions Required	Administrator, SYSTEM
Data Sources	WMI Objects

Contents [hide]

Examples

- APT29 has used WMI event filters to establish persistence.[4]
- Leviathan has used WMI for persistence.[5]
- POSHSPY uses a WMI event subscription to establish persistence.[6]

Examples

- APT29 has used WMI event filters to establish persistence.[4]

WMI Event Subscription

Figure 5:
SEADADDY WMI persistence with PowerShell

```
$filterName='BotFilter82'
$consumerName='BotConsumer23'
$exePath='C:\Windows\System32\evil.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;ExecutablePa
th=$exePath;CommandLineTemplate=$exePath}
Set-WmiInstance-Class__FilterToConsumerBinding-
Namespace"root\subscription"-Arguments
@{Filter=$WMIEventFilter;Consumer=$WMIEventConsumer}
```

Source:
https://www.fireeye.com/content/dam/fireeye-www/global/en/current-threats/pdfs/wp-windows-management-instrumentation.pdf

WMI Event Subscription

- Generating test events using “PowerLurk” Github project
- Likely won’t catch many APTs searching for `Register-MaliciousWmiEvent` ;-)

```
PS C:\PowerShell\PowerLurk-master\PowerLurk-master> Set-ExecutionPolicy Bypass

PS C:\PowerShell\PowerLurk-master\PowerLurk-master> . .\PowerLurk.ps1

PS C:\PowerShell\PowerLurk-master\PowerLurk-master> Register-MaliciousWmiEvent
-EventName LogNotepad -PermanentCommand "cmd.exe /c echo %ProcessId% >>
C:\\Users\\Public\\notepad-log.txt" -Trigger ProcessStart -ProcessName notepad.exe
```

```
PS C:\PowerShell\PowerLurk-master\PowerLurk-master> Register-MaliciousWmiEvent
-EventName Logonlog -PermanentCommand "cmd.exe /c echo %TargetInstance.Antecedent%
>> C:\Users\Public\logon.txt" -Trigger UserLogon -Username any
```

How noisy is the Sysmon WmiEvent?

> 90 days
> 270 EP's
< 600 events
4 diff types

```
sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational" WmiEvent
    (WmiFilterEvent OR WmiConsumerEvent OR WmiBindingEvent)
| search (EventCode=19 OR EventCode=20 OR EventCode=21)
| rex field=Message ".*EventType: (?<WmiEventType>.*)"
| stats dc(Name) dc(Query) dc(EventNamespace) dc(Consumer) dc(Filter) dc(ComputerName)
    count by TaskCategory EventCode WmiEventType
| sort EventCode
```

✓ 1,764 events (6/1/18 12:00:00.000 AM to 10/18/18 12:00:00.000 AM) No Event Sampling Job Fast Mode

Events Patterns **Statistics (3)** Visualization

100 Per Page Format Preview

TaskCategory	EventCode	WmiEventType	dc(Name)	dc(Query)	dc(EventNamespace)	dc(Consumer)	dc(Filter)	dc(ComputerName)	count
WmiEventFilter activity detected (rule: WmiEvent)	19	WmiFilterEvent	5	5	2	0	0	271	586
WmiEventConsumer activity detected (rule: WmiEvent)	20	WmiConsumerEvent	4	0	0	0	0	273	594
WmiEventConsumerToFilter activity detected (rule: WmiEvent)	21	WmiBindingEvent	0	0	0	4	4	271	584

```
sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational" WmiEvent
    [redacted]
| search EventCode=19 OR EventCode=20 OR EventCode=21
| rex field=Message ".*User: ([redacted]|NT AUTHORITY)\\\\(?<USER1>.*)"
| table _time EventCode TaskCategory Message ComputerName USER1
```

_time	EventCode	TaskCategory	Message
2018-07-03 11:25:25	21	WmiEventConsumerToFilter activity detected (rule: WmiEvent)	WmiEventConsumerToFilter activity detected: EventType: WmiBindingEvent UtcTime: 2018-07-03 09:25:25.382 Operation: Created User: [redacted] Consumer: "CommandLineEventConsumer.Name=\"Logonlog\"" Filter: "__EventFilter.Name=\"Logonlog\""
2018-07-03 11:25:25	19	WmiEventFilter activity detected (rule: WmiEvent)	WmiEventFilter activity detected: EventType: WmiFilterEvent UtcTime: 2018-07-03 09:25:25.339 Operation: Created User: [redacted] EventNamespace: "root/cimv2" Name: "Logonlog" Query: "SELECT * FROM __InstanceCreationEvent WITHIN 10 WHERE TargetInstance ISA 'Win32_LoggedOnUser'"
2018-07-03 11:25:25	20	WmiEventConsumer activity detected (rule: WmiEvent)	WmiEventConsumer activity detected: EventType: WmiConsumerEvent UtcTime: 2018-07-03 09:25:25.316 Operation: Created User: [redacted] Name: "Logonlog" Type: Command Line Destination: "cmd.exe /c echo %TargetInstance.Antecedent% >> C:\\Users\\Public\\logon.txt"

```
sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational" WmiEvent
    [redacted]
| search EventCode=19 OR EventCode=20 OR EventCode=21
| rex field=Message ".*User: ([redacted]|NT AUTHORITY)\\\\(?<USER1>.*)"
| table _time EventCode TaskCategory Message ComputerName USER1
```

_time	EventCode	TaskCategory	Message
2018-07-03 11:25:40	21	WmiEventConsumerToFilter activity detected (rule: WmiEvent)	WmiEventConsumerToFilter activity detected: EventType: WmiBindingEvent UtcTime: 2018-07-03 09:25:40.004 Operation: Created User: [redacted] Consumer: "CommandLineEventConsumer.Name=\"LogNotepad\"" Filter: "__EventFilter.Name=\"LogNotepad\""
2018-07-03 11:25:39	19	WmiEventFilter activity detected (rule: WmiEvent)	WmiEventFilter activity detected: EventType: WmiFilterEvent UtcTime: 2018-07-03 09:25:39.910 Operation: Created User: [redacted] EventNamespace: "root/cimv2" Name: "LogNotepad" Query: "SELECT * FROM Win32_ProcessStartTrace WHERE ProcessName='notepad.exe'"
2018-07-03 11:25:39	20	WmiEventConsumer activity detected (rule: WmiEvent)	WmiEventConsumer activity detected: EventType: WmiConsumerEvent UtcTime: 2018-07-03 09:25:39.883 Operation: Created User: [redacted] Name: "LogNotepad" Type: Command Line Destination: "cmd.exe /c echo %ProcessId% >> C:\\\\Users\\\\Public\\\\notepad-log.txt"

Outline

- Introduction
- 2nd of 3 techniques from MITRE ATT&CK

Logon Scripts	
	Technique
ID	T1037
Tactic	Lateral Movement, Persistence
Platform	macOS, Windows
System Requirements	Write access to system or domain logon scripts
Data Sources	File monitoring, Process monitoring
CAPEC ID	CAPEC-564

Idea for detection

- Search for child processes of “userinit.exe”
- Exclude “explorer.exe” (normal)
- Exclude logon scripts (after baselining & vetting)
- Possibly a small number of other legitimate executables, but feasible to enumerate and filter out

- Search for ProcessCreate or RegistryEvents with the registry key name “UserInitMprLogonScript”

```
sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
    ( ProcessCreate userinit.exe ) OR ( ProcessCreate OR RegistryEvent UserInitMprLogonScript )
| search (ParentImage="*\\userinit.exe" Image!="*\\explorer.exe"
        [redacted]
        [redacted]
        [redacted]
         CommandLine!="*\\netlogon\\netlogon.bat*") OR
        UserInitMprLogonScript
| stats values(CommandLine) dc(ComputerName) AS DC_host count by ParentImage Image
```

ParentImage	Image	values(CommandLine)	DC_host	count
C:\Windows\System32\cmd.exe	C:\Windows\System32\reg.exe	REG ADD HKCU\Environment /v UserInitMprLogonScript /t REG_SZ /d "notepad.exe C:\Users\[redacted]\Desktop\UserInitMprLogonScript.txt" reg query HKCU\Environment /v UserInitMprLogonScript reg query HKCU\Environment\UserInitMprLogonScript	2	4
C:\Windows\System32\userinit.exe	C:\Windows\System32\notepad.exe	notepad.exe notepad.exe C:\Users\[redacted]\Desktop\UserInitMprLogonScript.txt	3	4
C:\Windows\explorer.exe	C:\Windows\System32\notepad.exe	"C:\WINDOWS\system32\NOTEPAD.EXE" C:\Users\[redacted]\Desktop\userinitMprLogonScript_notepad_reg.txt	1	2

```
1  sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
2      ( Process                                                    onScript )
3  | search (Par
4
5
6
7          Com
8        User
9  | stats value
```

Message

```
Process Create:
UtcTime: 2019-10-23 12:32:15.127
ProcessGuid: {5c2fa88c-484f-5db0-0000-001050e6a701}
ProcessId: 7948
Image: C:\Windows\System32\notepad.exe
FileVersion: 10.0.17763.1 (WinBuild.160101.0800)
Description: Notepad
Product: Microsoft® Windows® Operating System
Company: Microsoft Corporation
CommandLine: notepad.exe C:\Users\[redacted]\Desktop\UserInitMprLogonScript.txt
CurrentDirectory: C:\WINDOWS\system32\
User: POST\ueltschit
LogonGuid: {5c2fa88c-4844-5db0-0000-0020102ca201}
LogonId: 0x1A22C10
TerminalSessionId: 2
IntegrityLevel: Medium
Hashes: MD5=0E61079D3283687D2E279272966AE99D,IMPHASH=C8922BE3DCDFEB5994C9EEE7745DC22E
ParentProcessGuid: {5c2fa88c-484e-5db0-0000-00102e6da701}
ParentProcessId: 7504
ParentImage: C:\Windows\System32\userinit.exe
ParentCommandLine: C:\windows\system32\userinit.exe
```

```
1  sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
2      ( Process                                                    onScript )
3  | search (Par
4
5
6
7          Com
8
9  |
```

Message

```
Process Create:
UtcTime: 2019-10-23 12:32:15.127
ProcessGuid: {5c2fa88c-484f-5db0-0000-001050e6a701}
ProcessId: 7948
Image: C:\Windows\System32\notepad.exe
```

```
Image: C:\Windows\System32\notepad.exe
FileVersion: 10.0.17763.1 (WinBuild.160101.0800)
Description: Notepad
Product: Microsoft® Windows® Operating System
Company: Microsoft Corporation
CommandLine: notepad.exe C:\Users\[redacted]\Desktop\UserInitMprLogonScript.txt
```

```
LogonId: 0x1A22C10
```

```
ParentImage: C:\Windows\System32\userinit.exe
ParentCommandLine: C:\windows\system32\userinit.exe
```

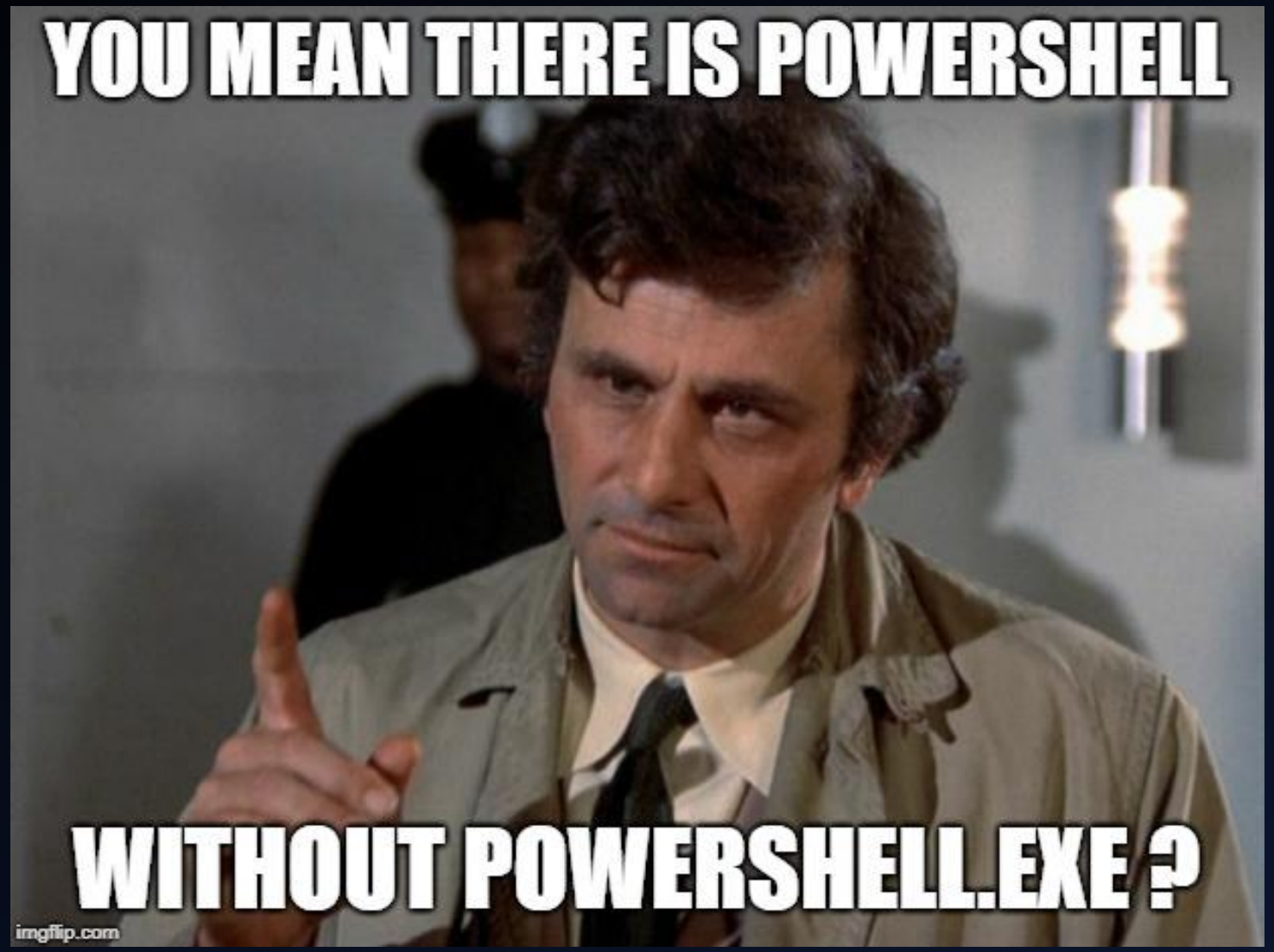
```
ParentProcessGuid: {5c2fa88c-484e-5db0-0000-00102e6da701}
ParentProcessId: 7504
ParentImage: C:\Windows\System32\userinit.exe
ParentCommandLine: C:\windows\system32\userinit.exe
```

Outline

- Introduction
- 3rd of 3 techniques from MITRE ATT&CK

PowerShell	
Technique	
ID	T1086
Tactic	Execution
Platform	Windows
Permissions Required	User, Administrator
Data Sources	Windows Registry, File monitoring, Process command-line parameters, Process monitoring
Supports Remote	Yes

Unmanaged PowerShell



Idea for detection

- Search PowerShell Transcript Files for “Host Application:” which is NOT any of
  - `powershell.exe`
  - `powershell_ise.exe`
  - `wsmprovhost.exe`
  - and possibly very few others

```
sourcetype="PowerShell_transcript.*" "Host Application:" NOT powershell.exe
| search NOT "Host Application: C:\\*\\powershell.exe"
| rex  field=_raw ".*Host Application: (?<Host_Application>[^ \n]*).*"
| rex  field=_raw ".*Username: (NT AUTHORITY)\\\\(?<Username>.*)"
| search Host_Application!="powershell"
    Host_Application!="*\\PowerShell_ISE.exe"
    Host_Application!="*\\wsmprovhost.exe"
    Host_Application!="*\\"
| stats count by host Username Host_Application
```

Host_Application	count
C:\WINDOWS\sysnative\rundll32.exe	5
C:\Windows\SysWOW64\WindowsPowerShell\v1.0\evading-PS-CLI-detections.exe	12
PSAttack.exe	203

Thanks for your attention!!

Time left for questions?

- Twitter: @c_APT_ure
- Blog: http://c-apt-ure.blogspot.com/2017/12/is-this-blog-still-alive.html

→ all my presentations linked in one place