

---

# **EQL Analytics Library**

**endgame**

**Feb 12, 2020**



---

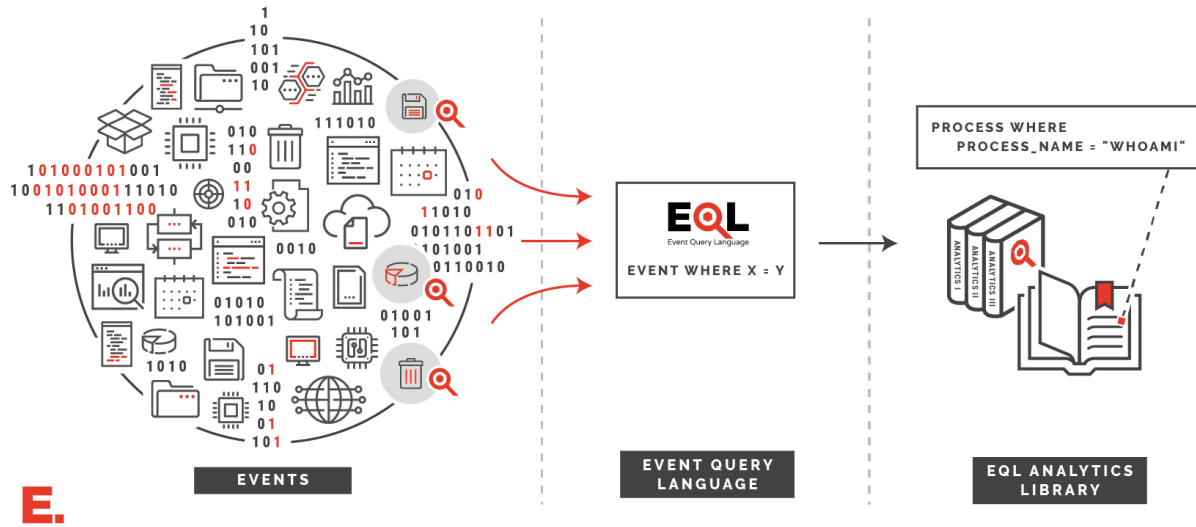
## Contents

---

<b>1</b>	<b>Next Steps</b>	<b>3</b>
1.1	Getting Started . . . . .	3
1.2	Analytics . . . . .	7
1.3	Atomic Blue Detections . . . . .	100
1.4	Enterprise ATT&CK Matrix . . . . .	103
1.5	Schemas . . . . .	154
1.6	Resources . . . . .	160
1.7	License . . . . .	161
	<b>Index</b>	<b>163</b>



WHAT DOES THE EVENT QUERY LANGUAGE DO?



**eqlib** is a library of event based analytics, written in **EQL** to detect adversary behaviors identified in MITRE ATT&CK™.



- *Get started* with EQL on your own computer
- Explore the *analytics* that map to ATT&CK.
- Learn how to *write queries* in EQL syntax
- Browse our *schemas* and existing normalizations
- View additional *resources*
- Check the *license* status

## 1.1 Getting Started

The EQL library current supports Python 2.7 and 3.5 - 3.7. Assuming a supported Python version is installed, run the command:

```
$ git clone https://github.com/endgameinc/eqllib
$ cd eqllib
$ python setup.py install
```

If Python is configured and already in the PATH, then eqllib will be readily available, and can be checked by running the command:

```
$ eqlib -h
usage: eqlib [-h] {convert-query,convert-data,query,survey} ...

EQL Analytics

positional arguments:
  {convert-query,convert-data,query,survey}
                        Sub Command Help
  convert-query         Convert a query to specific data source
  convert-data          Convert data from a specific data source
```

(continues on next page)

query	Query over a data source
survey	Run multiple analytics over JSON data

### 1.1.1 eqllib Command-Line Interface

The EQL Analytics Library comes with a utility that can search, normalize, and survey JSON data. See *Getting Started* for instructions on installing `eqllib` locally.

#### convert-data

`eqllib convert-data [OPTIONS] <input-json-file> <output-json-file>`

The `convert-data` command normalizes data, generating a new JSON file that matches the schema.

#### Arguments

##### output-json-file

Path to an output JSON file to store normalized events.

#### Options

`-h`

Show the help message and exit

`--file, -f`

Path to a JSON file of unnormalized events. Defaults to stdin if not specified

`--format`

Format for the input file. One of `json`, `json.gz`, `jsonl`, `jsonl.gz`

`-s <data-source>, --source <data-source>`

Required: the source schema for the events. (e.g. "Microsoft Sysmon")

`-e <encoding>`

Source file encoding. (e.g. `ascii`, `utf8`, `utf16`, etc.)

#### convert-query

`eqllib convert-query [OPTIONS] <eql-query>`

The `convert-query` command takes an EQL query that matches a normalized schema, and will print out the query converted to match a different schema.

#### Arguments

##### eql-query

Input EQL query written for the normalization schema



## Options

- h** Show the help message and exit
- s** <data-source>, **--source** <data-source>  
Required: the source schema for the events. (e.g. "Microsoft Sysmon")

## query

The **query** command reads JSON events and print matching output events back as JSON. Unless specified with **-s**, data is assumed to already be normalized against the schema.

**eqllib query** [*OPTIONS*] <input-query> <json-file>

## Arguments

**input-query**  
Query in EQL syntax that matches the common schema.

## Options

- h** Show the help message and exit
- file, -f**  
Path to a JSON file of unnormalized events. Defaults to stdin if not specified
- format**  
Format for the input file. One of json, json.gz, jsonl, jsonl.gz
- s** <data-source>, **--source** <data-source>  
Required: the source schema for the events. (e.g. "Microsoft Sysmon")
- e** <encoding>  
Source file encoding. (e.g. ascii, utf8, utf16, etc.)

## survey

**eqllib survey** [*OPTIONS*] <json-file> <analytic-path> [*analytic-path, ...*]

The **survey** command can be used to run multiple analytics against a single JSON file. Unless specified with **-s**, data is assumed to already be normalized against the schema.

## Arguments

**analytic-path** [*analytic-path, ...*]  
Path(s) to analytic TOML files or a directory of analytics.

### Options

- h** Show the help message and exit
- file, -f** Path to a JSON file of unnormalized events. Defaults to stdin if not specified
- format** Format for the input file. One of json, json.gz, jsonl, jsonl.gz
- s <data-source>, --source <data-source>** Required: the source schema for the events. (e.g. "Microsoft Sysmon")
- e <encoding>** Source file encoding. (e.g. ascii, utf8, utf16, etc.)
- c** Output counts per analytic instead of the individual hits.

View usage for the related [EQL utility](#).

### 1.1.2 Guide to Microsoft Sysmon

Microsoft Sysmon is a freely available tool provided by SysInternals for endpoint logging.

#### Installing Sysmon

Download Sysmon from SysInternals.

To install Sysmon, from a terminal, simply change to the directory where the unzipped binary is located, then run the following command as an Administrator

To capture all default event types, with all hashing algorithms, run

```
Sysmon.exe -AcceptEula -i -h * -n -l
```

To configure Sysmon with a specific XML configuration file, run

```
Sysmon.exe -AcceptEula -i myconfig.xml
```

Full details of what each flag does can be found on the [Microsoft Sysmon](#) page

**Warning:** Depending on the configuration, Sysmon can generate a significant amount of data. When deploying Sysmon to production or enterprise environments, it is usually best to tune it to your specific environment. There are several Sysmon configuration files in common use which can be used or referenced for this purpose.

- @SwiftOnSecurity's [scalable config file](#).
- @olafhartong's more [verbose config file](#).

#### Getting Sysmon logs with PowerShell

Helpful PowerShell functions for parsing Sysmon events from Windows Event Logs are found in the Github at [utils/scrape-events.ps1](#)

Getting logs into JSON format can be done by piping to PowerShell cmdlets within an elevated powershell.exe console.

```
# Import the functions provided within scrape-events
Import-Module .\utils\scrape-events.ps1

# Save the most recent 5000 Sysmon logs
Get-LatestLogs | ConvertTo-Json | Out-File -Encoding ASCII -FilePath my-sysmon-data.
↪json

# Save the most recent 1000 Sysmon process creation events
Get-LatestProcesses | ConvertTo-Json | Out-File -Encoding ASCII -FilePath my-sysmon-
↪data.json
```

To get *all* Sysmon logs from Windows Event Logs, run the powershell command

```
Get-WinEvent -filterhashtable @{logname="Microsoft-Windows-Sysmon/Operational"} -
↪Oldest | Get-EventProps | ConvertTo-Json | Out-File -Encoding ASCII -FilePath my-
↪sysmon-data.json
```

**Warning:** Use this with caution as it will process all events, which may take time and likely generate a large file

## Example searches with EQL

Once you have logs in JSON format, they can now be queried using EQL. To do so, either the *query* or the *data* will need to be converted (normalized). Because EQL is built to be able to be flexible across all data sources, it is necessary to translate the query to match the underlying data, or to change the data to match the query. The conversion functionality is described in more detail in the *eqlib Command-Line Interface* guide.

For example, to find suspicious reconnaissance commands over the generated data

```
eqlib query -f my-sysmon-data.json --source "Microsoft Sysmon" "process where_
↪process_name in ('ipconfig.exe', 'netstat.exe', 'systeminfo.exe', 'route.exe')"
```

## 1.2 Analytics

### 1.2.1 Access of Outlook Email Archives

Collection of sensitive information via .ost and .pst outlook archive files.

**id** 15d87029-42c1-4992-a49b-aac74d451c06

**categories** enrich

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Collection

**techniques** T1114 Email Collection

#### Query

```
process where subtype.create and wildcard(command_line, "*.ost *", "*.pst *")
```

#### Contributors

- Endgame

### 1.2.2 Account Discovery via Built-In Tools

Adversaries may use built-in applications to get a listing of local system or domain accounts

**id** 56fdf859-b2a7-4009-88e0-69fec4c3deef

**categories** enrich

**confidence** low

**os** windows, macos, linux

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Discovery

**techniques** T1087 Account Discovery

#### Query

```
process where subtype.create and (  
  process_name == "net.exe" and wildcard(command_line, "* user*", "*localgroup *",  
  ↳ "*group *") or  
  process_name in ("groups", "id") or  
  process_name == "dscl" and command_line == "*list /groups*" or  
  process_name == "dscacheutil" and command_line == "*group*" or  
  wildcard(command_line, "*/etc/passwd*", "*/etc/master.passwd*")  
)
```

#### Contributors

- Endgame

### 1.2.3 AD Dumping via Ntdsutil.exe

Identifies usage of `ntdsutil.exe` to export an Active Directory database to disk.

**id** 19d59f40-12fc-11e9-8d76-4d6bb837cda4  
**categories** detect  
**confidence** medium  
**os** windows  
**created** 01/07/2019  
**updated** 01/07/2019

#### MITRE ATT&CK™ Mapping

**tactics** Credential Access  
**techniques** T1003 Credential Dumping

#### Query

```
file where file_name == "ntds.dit" and process_name == "ntdsutil.exe"
```

#### Detonation

Atomic Red Team: T1003

#### Contributors

- Tony Lambert

### 1.2.4 Adding the Hidden File Attribute with via attrib.exe

Adversaries can add the *hidden* attribute to files to hide them from the user in an attempt to evade detection

**id** 9051814c-a142-4b1c-965b-76a09dace760  
**categories** enrich  
**confidence** low  
**os** windows  
**created** 7/26/2019  
**updated** 7/26/2019

#### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion, Persistence  
**techniques** T1158 Hidden Files and Directories

### Query

```
process where subtype.create and  
process_name == "attrib.exe" and  
command_line == "* +h*"
```

### Contributors

- Endgame

## 1.2.5 AppCert DLLs Registry Modification

Dynamic-link libraries (DLLs) that are specified in the AppCertDLLs value in the Registry key can be abused to obtain persistence and privilege escalation by causing a malicious DLL to be loaded and run in the context of separate processes on the computer.

**id** 14f90406-10a0-4d36-a672-31cabe149f2f  
**categories** enrich  
**confidence** low  
**os** windows  
**created** 7/26/2019  
**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Privilege Escalation, Persistence  
**techniques** T1182 AppCert DLLs

### Query

```
registry where registry_path == "*\\System\\ControlSet*\\Control\\Session_  
↳Manager\\AppCertDLLs\\*"
```

### Contributors

- Endgame

## 1.2.6 Audio Capture via PowerShell

Detect attacker collecting audio via PowerShell Cmdlet.

**id** ab7a6ef4-0983-4275-a4f1-5c6bd3c31c23  
**categories** detect  
**confidence** medium  
**os** windows

**created** 11/30/2018

**updated** 11/30/2018

## MITRE ATT&CK™ Mapping

**tactics** Collection

**techniques** T1123 Audio Capture

## Query

```
process where subtype.create and
  process_name == "powershell.exe" and command_line == "* WindowsAudioDevice-
  ↳Powershell-Cmdlet *"
```

## Detonation

Atomic Red Team: T1123

## Contributors

- Endgame

## 1.2.7 Audio Capture via SoundRecorder

Detect audio collection via SoundRecorder application.

**id** f72a98cb-7b3d-4100-99c3-a138b6e9ff6e

**categories** detect

**confidence** medium

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

## MITRE ATT&CK™ Mapping

**tactics** Collection

**techniques** T1123 Audio Capture

## Query

```
process where subtype.create and
  process_name == "SoundRecorder.exe" and command_line == "* /FILE*"
```

### Detonation

Atomic Red Team: T1123

### Contributors

- Endgame

## 1.2.8 Bypass UAC via CMSTP

Detect child processes of automatically elevated instances of Microsoft Connection Manager Profile Installer (`cmstp.exe`).

**id** e584f1a1-c303-4885-8a66-21360c90995b  
**categories** detect  
**confidence** medium  
**os** windows  
**created** 11/30/2018  
**updated** 11/30/2018

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion, Execution

**techniques** T1191 CMSTP, T1088 Bypass User Account Control

### Query

```
sequence
[ process where subtype.create and
  process_name == "cmstp.exe" and command_line == "*/s*" and command_line == "*/au*"
↔ ] by unique_pid
[ process where subtype.create ] by unique_ppid
```

### Detonation

Atomic Red Team: T1191

### Contributors

- Endgame



## 1.2.9 Bypass UAC via CompMgmtLauncher

Identifies use of CompMgmtLauncher.exe to bypass User Account Control. Adversaries use this technique to execute privileged processes.

**id** 7efc7afe-8396-4bf0-ac7d-1a860a401d22

**categories** detect

**confidence** medium

**os** windows

**created** 12/04/2019

**updated** 12/04/2019

### MITRE ATT&CK™ Mapping

**tactics** Privilege Escalation

**techniques** T1088 Bypass User Account Control

### Query

```
sequence with maxspan=10s
[registry where registry_path == "*\\mscfile\\shell\\open\\command*" and user_name !
↪= "SYSTEM"]
[process where subtype.create and parent_process_path ==
↪"C:\\Windows\\System32\\CompMgmtLauncher.exe"]
```

### Contributors

- Daniel Stepanic

### References

- <https://enigma0x3.net/2016/08/15/fileless-uac-bypass-using-eventvwr-exe-and-registry-hijacking>
- <https://www.elastic.co/blog/embracing-offensive-tooling-building-detections-against-koadic-using-eql>

## 1.2.10 Bypass UAC via Fodhelper.exe

Identifies use of Fodhelper.exe to bypass User Account Control. Adversaries use this technique to execute privileged processes.

**id** e491ce22-792f-11e9-8f5c-d46d6d62a49e

**categories** detect

**confidence** high

**os** windows

**created** 05/17/2019

**updated** 05/17/2019

### MITRE ATT&CK™ Mapping

**tactics** Privilege Escalation

**techniques** T1088 Bypass User Account Control

#### Query

```
process where subtype.create and  
parent_process_name == "fodhelper.exe"
```

#### Detonation

Atomic Red Team: T1088

#### Contributors

- Tony Lambert

### 1.2.11 Bypass UAC via WSReset.exe

Identifies use of WSReset.exe to bypass User Account Control. Adversaries use this technique to execute privileged processes.

**id** 532b5ed4-7930-11e9-8f5c-d46d6d62a49e

**categories** detect

**confidence** high

**os** windows

**created** 05/17/2019

**updated** 05/17/2019

### MITRE ATT&CK™ Mapping

**tactics** Privilege Escalation

**techniques** T1088 Bypass User Account Control

#### Query

```
process where subtype.create and  
parent_process_name == "wsreset.exe" and process_name != "conhost.exe"
```

#### Detonation

Atomic Red Team: T1088

## Contributors

- Tony Lambert

### 1.2.12 Change Default File Association

Detect changes to default File Association handlers.

**id** 26f0ebab-b315-492d-a5be-aa665fba2f35  
**categories** hunt  
**confidence** medium  
**os** windows  
**created** 11/30/2018  
**updated** 11/30/2018

## MITRE ATT&CK™ Mapping

**tactics** Persistence  
**techniques** T1042 Change Default File Association

## Query

```
sequence by unique_pid with maxspan=1s
  [ registry where registry_path == "*\\SOFTWARE\\Classes\\*\\*" ]
  [ registry where registry_path ==
↵ "*\\SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Explorer\\GlobalAssocChangedCounter
↵ "]
| unique_count process_name, registry_path
```

## Detonation

Atomic Red Team: T1042

## Contributors

- Endgame

### 1.2.13 Clearing Windows Event Logs with wevtutil

Identifies attempts to clear Windows event logs with the command `wevtutil`.

**id** 5b223758-07d6-4100-9e11-238cfdd0fe97  
**categories** detect  
**confidence** low  
**os** windows

**created** 11/30/2018

**updated** 11/30/2018

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion

**techniques** T1070 Indicator Removal on Host

### Query

```
process where subtype.create and
  process_name == "wevtutil.exe" and command_line == "* cl *"
```

### Detonation

Atomic Red Team: T1070

### Contributors

- Endgame

## 1.2.14 COM Hijack via Script Object

Identifies COM hijacking using the script object host `scrobj.dll`, which allows for stealthy execution of scripts in legitimate processes.

**id** 9d556fd6-76a3-45d5-9d8d-cb8edf0282f2

**categories** detect

**confidence** medium

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

### MITRE ATT&CK™ Mapping

**tactics** Persistence, Defense Evasion

**techniques** T1122 Component Object Model Hijacking

### Query

```
registry where
  registry_path == "*_Classes\\CLSID\\{*}\\InprocServer32*" and
  (registry_data == "scrobj*" or registry_data == "*\\scrobj*")
```

## Detonation

Atomic Red Team: T1122

## Contributors

- Endgame

### 1.2.15 Command-Line Creation of a RAR file

Detect compression of data into a RAR file using the `rar.exe` utility.

```
id 1ec33c93-3d0b-4a28-8014-dbdaae5c60ae
categories detect
confidence medium
os windows
created 11/30/2018
updated 11/30/2018
```

## MITRE ATT&CK™ Mapping

```
tactics Exfiltration
techniques T1002 Data Compressed
```

## Query

```
process where subtype.create and process_name == "rar.exe" and
  command_line == "* a *"
```

## Detonation

Atomic Red Team: T1002

## Contributors

- Endgame

### 1.2.16 Control Panel Items

Windows Control Panel items are utilities that allow users to view and adjust computer settings. Adversaries can use Control Panel items as execution payloads to execute arbitrary commands.

```
id 3b9bbf6b-dde2-4f82-b1ad-b3b625f44a26
categories enrich
confidence low
```

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion, Execution

**techniques** T1196 Control Panel Items

### Query

```
process where subtype.create and
  process_name in ("control.exe", "rundll32.exe") and
  command_line == "*.cpl *"
```

### Contributors

- Endgame

## 1.2.17 Creation of an Archive with Common Archivers

Adversaries may collect and stage data in a central location or directory in preparation of exfiltration

**id** f43f66f3-7e86-4cd1-9850-df7b4ac7822e

**categories** enrich

**confidence** low

**os** macos, linux

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Collection

**techniques** T1074 Data Staged

### Query

```
sequence by unique_pid with maxspan=1m
  [ process where subtype.create and process_name in ("zip", "tar", "gzip", "hdiutil
  ↳") ]
  [ file where wildcard(file_name, "*.zip", "*.tar", "*.gzip", "*.gz") ]
```

## Contributors

- Endgame

### 1.2.18 Creation of Kernel Module

Identify activity related to loading kernel modules on Linux via creation of new ko files in the LKM directory

**id** 9e711823-72f1-4c5c-843d-9afc90c4e6a1

**categories** enrich

**confidence** low

**os** linux

**created** 7/26/2019

**updated** 7/26/2019

## MITRE ATT&CK™ Mapping

**tactics** Persistence

**techniques** T1215 Kernel Modules and Extensions

## Query

```
file where subtype.create and
  file_path == "/lib/modules/*" and file_name == "*.ko"
```

## Contributors

- Endgame

### 1.2.19 Creation of Scheduled Task with schtasks.exe

A scheduled task can be used by an adversary to establish persistence, move laterally, and/or escalate privileges.

**id** 9583c2ff-508d-4ebb-8b89-712b0a4d3186

**categories** hunt

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

## MITRE ATT&CK™ Mapping

**tactics** Privilege Escalation, Execution, Persistence

**techniques** T1053 Scheduled Task

### Query

```
process where subtype.create and  
  process_name = "schtasks.exe" and  
  command_line = "*create*"
```

### Contributors

- Endgame

## 1.2.20 Creation or Modification of Systemd Service

Systemd services can be used to establish persistence on a Linux system. The systemd service manager is commonly used for managing background daemon processes (also known as services) and other system resources.

```
id 1a568233-9ca1-4c2c-b2e7-b15b90e2c954  
categories enrich  
confidence low  
os linux  
created 7/26/2019  
updated 7/26/2019
```

### MITRE ATT&CK™ Mapping

```
tactics Persistence  
techniques T1501 Systemd Service
```

### Query

```
file where not subtype.delete and  
  file_name == "*.service*" and  
  wildcard(file_path, "/etc/systemd/system/*", "/usr/lib/systemd/system/*")
```

### Contributors

- Endgame

## 1.2.21 Credential Enumeration via Credential Vault CLI

Identifies use of the Credential Vault command line interface to enumerate a user's saved credentials.

```
id 11968244-6db0-4e03-886c-e3983f9d9024  
categories detect  
confidence high  
os windows
```



**created** 8/16/2019

**updated** 8/16/2019

## MITRE ATT&CK™ Mapping

**tactics** Credential Access

**techniques** T1003 Credential Dumping

## Query

```
process where subtype.create and
  process_name == "vaultcmd.exe" and
  command_line == "* /list*"
```

## Contributors

- David French

## References

- <https://rastamouse.me/2017/08/jumping-network-segregation-with-rdp/>
- <https://medium.com/threatpunter/detecting-adversary-tradecraft-with-image-load-event-logging-8de93338c16>

## 1.2.22 Delete Volume USN Journal with fsutil

Identifies use of the fsutil command to delete the volume USNJRNL. This technique is used by attackers to eliminate evidence of files created during post-exploitation activities.

**id** c91f422a-5214-4b17-8664-c5fcf115c0a2

**categories** detect

**confidence** low

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

## MITRE ATT&CK™ Mapping

**tactics** Defense Evasion

**techniques** T1070 Indicator Removal on Host

## Query

```
process where subtype.create and
  process_name == "fsutil.exe" and command_line == "* usn *" and command_line == "*_
↳deletejournal*"
```

(continues on next page)

### Detonation

Atomic Red Team: T1070

### Contributors

- Endgame

## 1.2.23 Disconnecting from Network Shares with net.exe

Identifies attempts to remove network shares with the Windows built-in command net.exe

**id** 7d328c61-8f63-4411-9ae7-e5b502a80e7e

**categories** enrich

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion

**techniques** T1126 Network Share Connection Removal

### Query

```
process where subtype.create and  
process_name == "net.exe" and command_line == "* /d*"
```

### Contributors

- Endgame

## 1.2.24 Discovery and Enumeration of System Information via Rundll32

Identifies initial system enumeration and discovery commands tied to remote access tools that leverage “rundll32.exe”.

**id** 35d27938-d13d-4bcd-9be7-3a69d208c63f

**categories** detect

**confidence** medium

**os** windows

**created** 12/04/2019

**updated** 12/04/2019

## MITRE ATT&CK™ Mapping

**tactics** Discovery

**techniques** T1087 Account Discovery, T1096 NTFS File Attributes, T1033 System Owner/User Discovery

## Query

```
sequence with maxspan=1h
  [process where subtype.create and process_name == "rundll32.exe"] by unique_pid
  [network where subtype.outgoing and process_name == "rundll32.exe"] by unique_pid
  [process where subtype.create and parent_process_name == "rundll32.exe"] by unique_
  ↳ ppid
```

## Contributors

- Daniel Stepanic

## References

- <https://www.elastic.co/blog/embracing-offensive-tooling-building-detections-against-koadic-using-eql>

## 1.2.25 Discovery of a Remote System's Time

Identifies use of various commands to query a remote system's time. This technique may be used before executing a scheduled task or to discover the time zone of a target system

**id** fcdb99c2-ac3c-4bde-b664-4b336329bed2

**categories** detect

**confidence** low

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

## MITRE ATT&CK™ Mapping

**tactics** Discovery

**techniques** T1124 System Time Discovery

### Query

```
process where subtype.create and process_name == "net.exe" and
  command_line == "* time *" and command_line == "*\\\\\\*"
| unique parent_process_path, command_line
```

### Detonation

Atomic Red Team: T1124

### Contributors

- Endgame

## 1.2.26 Discovery of Domain Groups

Identify usage of known commands for discovery of local groups

**id** cd2124cb-718d-4ecf-bc96-5571f8e3dbce

**categories** enrich

**confidence** low

**os** linux, macos

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Discovery

**techniques** T1069 Permission Groups Discovery

### Query

```
process where subtype.create and (
  process_name in ("ldapsearch", "dscacheutil") or
  process_name == "dscl" and command_line == "*-list*"
)
```

### Contributors

- Endgame

## 1.2.27 Discovery of Network Environment via Built-in Tools

Built-in tools can be used to enumerate and discover network environment on unix systems.

**id** fd7a0c56-60fa-4f14-8c8e-0e41ad955725  
**categories** enrich  
**confidence** low  
**os** macos, linux  
**created** 7/26/2019  
**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Discovery  
**techniques** T1016 System Network Configuration Discovery

### Query

```
process where subtype.create and (  
  process_name in ("ifconfig", "arp", "networkctl", "netstat", "route", "ntop") or (  
    process_name in ('cat', 'more', 'less', 'vim', 'vi', 'nano', 'gedit') and  
    command_line == "* /etc/hosts*"  
  )  
)
```

### Contributors

- Endgame

## 1.2.28 Discovery of Network Environment via Built-in Tools

Built-in tools can be used to enumerate and discover network environment on windows systems.

**id** 3a78a9fb-3714-43fa-90ca-7cf85da5a710  
**categories** enrich  
**confidence** low  
**os** windows  
**created** 7/26/2019  
**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Discovery  
**techniques** T1016 System Network Configuration Discovery

### Query

```
process where subtype.create and
  process_name in ("ipconfig.exe", "route.exe", "nbtstat.exe", "arp.exe")
| unique command_line
```

### Contributors

- Endgame

## 1.2.29 DLL Search Order Hijacking with known programs

Detects writing DLL files to known locations associated with Windows files vulnerable to DLL search order hijacking.

**id** afd1fba7-5301-4d5c-ae66-f8608bc98ae9

**categories** detect

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Privilege Escalation, Defense Evasion, Persistence

**techniques** T1038 DLL Search Order Hijacking

### Query

```
file where not subtype.delete and
  not user_sid in ("S-1-5-18", "S-1-5-19", "S-1-5-20") and (
    file_path == "*\\windows\\ehome\\cryptbase.dll" or
    file_path == "*\\windows\\system32\\sysprep\\cryptbase.dll" or
    file_path == "*\\windows\\system32\\sysprep\\cryptsp.dll" or
    file_path == "*\\windows\\system32\\sysprep\\rpcrtremote.dll" or
    file_path == "*\\windows\\system32\\sysprep\\uxtheme.dll" or
    file_path == "*\\windows\\system32\\sysprep\\dwmapi.dll" or
    file_path == "*\\windows\\system32\\sysprep\\shcore.dll" or
    file_path == "*\\windows\\system32\\sysprep\\oleacc.dll" or
    file_path == "*\\windows\\system32\\ntwdblib.dll"
  )
| unique process_path, file_path
```

### Contributors

- Endgame

## 1.2.30 Domain Trust Discovery

Detect commands used to enumerate a list of trusted domains.

**id** bccb1c48-305c-4b1f-affb-a7a50bf4654b  
**categories** enrich  
**confidence** low  
**os** windows  
**created** 7/26/2019  
**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Discovery  
**techniques** T1482 Domain Trust Discovery

### Query

```
process where subtype.create and (  
  (process_name == "dsquery.exe") and command_line == "(objectClass=trustedDomain)*"  
  or  
  (process_name == "nltest.exe") and command_line == "*domain_trusts*"  
)
```

### Contributors

- Endgame

## 1.2.31 Domain Trust Discovery via Nltest.exe

Identifies execution of nltest.exe for domain trust discovery. This technique is used by attackers to enumerate Active Directory trusts.

**id** 03e231a6-74bc-467a-acb1-e5676b0fb55e  
**categories** hunt  
**confidence** low  
**os** windows  
**created** 05/17/2019  
**updated** 05/17/2019

### MITRE ATT&CK™ Mapping

**tactics** Discovery  
**techniques** T1482 Domain Trust Discovery

### Query

```
process where subtype.create and
  process_name == "nltest.exe" and command_line == "*domain_trusts*"
```

### Detonation

Atomic Red Team: T1482

### Contributors

- Tony Lambert

## 1.2.32 Encoding or Decoding Files via CertUtil

Find execution of the Windows tool certutil.exe to decode or encode files.

```
id c6facc54-4894-4722-b873-062baaae851f
categories detect
confidence medium
os windows
created 11/30/2018
updated 11/30/2018
```

### MITRE ATT&CK™ Mapping

```
tactics Defense Evasion
techniques T1140 Deobfuscate/Decode Files or Information
```

### Query

```
process where subtype.create and
  process_name == "certutil.exe" and
  (command_line == "*encode *" or command_line == "*decode *")
```

### Detonation

Atomic Red Team: T1140

### Contributors

- Endgame



### 1.2.33 Enumeration of Local Shares

Identifies enumeration of local shares with the built-in Windows tool `net . exe`.

**id** bc1944cd-97fc-4b9a-b068-46203b6bbcde  
**categories** detect  
**confidence** low  
**os** windows  
**created** 11/30/2018  
**updated** 11/30/2018

#### MITRE ATT&CK™ Mapping

**tactics** Discovery  
**techniques** T1135 Network Share Discovery

#### Query

```
process where subtype.create and  
  (process_name == "net.exe" or (process_name == "net1.exe" and parent_process_name !=  
  ↳ "net.exe")) and  
  command_line == "* share*" and command_line != "* * *"
```

#### Contributors

- Endgame

### 1.2.34 Enumeration of Mounted Shares

Identifies enumeration of mounted shares with the built-in Windows tool `net . exe`.

**id** 4d2e7fc1-af0b-4915-89aa-03d25ba7805e  
**categories** detect  
**confidence** low  
**os** windows  
**created** 11/30/2018  
**updated** 11/30/2018

#### MITRE ATT&CK™ Mapping

**tactics** Discovery  
**techniques** T1049 System Network Connections Discovery

### Query

```
process where subtype.create and
  (process_name == "net.exe" or (process_name == "net1.exe" and parent_process_name !=
  ↳ "net.exe")) and
  (command_line == "* use" or command_line == "* use *") and

  // since this command is looking for discovery only, we want to ignore mounting_
  ↳ shares
  command_line != "* \\\\*"
| unique parent_process_path, command_line, user_name
```

### Detonation

Atomic Red Team: T1049

### Contributors

- Endgame

## 1.2.35 Enumeration of Remote Shares

Identifies enumeration of remote shares with the built-in Windows tool `net . exe`.

**id** e61f557c-a9d0-4c25-ab5b-bbc46bb24deb

**categories** detect

**confidence** low

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

### MITRE ATT&CK™ Mapping

**tactics** Discovery

**techniques** T1135 Network Share Discovery

### Query

```
process where subtype.create and
  (process_name == "net.exe" or (process_name == "net1.exe" and parent_process_name !=
  ↳ "net.exe")) and
  command_line == "* view*" and command_line == "* \\\\*"
```

### Detonation

Atomic Red Team: T1135

## Contributors

- Endgame

### 1.2.36 Enumeration of System Information

System information enumeration and discovery via built-in tools.

**id** 6a1247d5-8b8a-4a5c-8d35-dd9ef220e7d1

**categories** enrich

**confidence** low

**os** linux

**created** 7/26/2019

**updated** 7/26/2019

## MITRE ATT&CK™ Mapping

**tactics** Discovery

**techniques** T1082 System Information Discovery

## Query

```
process where subtype.create and (
  process_name == "uname" or (
    process_name in ("cat", "more", "less") and
    wildcard(command_line,
      "* /etc/issue*", "* /proc/version*", "* /etc/profile*",
      "* /etc/services*", "* /proc/cpuinfo*",)
  ))
```

## Contributors

- Endgame

### 1.2.37 Enumeration of System Information

Windows contains several built-in commands to report system information. These may be used by an actor to gain detailed information about the target machine.

**id** 507f19c1-dfa9-475b-925e-61e417a10967

**categories** enrich

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Discovery

**techniques** T1082 System Information Discovery

#### Query

```
process where subtype.create and (
  process_name in ("systeminfo.exe", "hostname.exe") or
  process_name == "cmd.exe" and wildcard(command_line, "* ver*", "%COMPUTERNAME%",
  ↳ "%PROCESSOR_*%")
)
```

#### Contributors

- Endgame

### 1.2.38 Executable Written and Executed by Microsoft Office Applications

Identifies an executable file written by a Microsoft Office application where that same executable is later ran as it's own process. This behavior can be indicative of suspicious activity possibly tied to macro objects or technologies used for command execution such as Dynamic Data Exchange (DDE).

**id** 2b512bec-b28d-4a84-9253-2c691bedb7bc

**categories** detect

**confidence** high

**os** windows

**created** 12/04/2019

**updated** 12/04/2019

### MITRE ATT&CK™ Mapping

**tactics** Execution

**techniques** T1204 User Execution, T1173 Dynamic Data Exchange

#### Query

```
sequence with maxspan=3d
  [file where file_name == "*.exe" and process_name in ("winword.exe", "excel.exe",
  ↳ "powerpnt.exe")] by file_path
  [process where true] by process_path
```

#### Contributors

- Daniel Stepanic

## References

- <https://www.elastic.co/blog/embracing-offensive-tooling-building-detections-against-koadic-using-eql>

### 1.2.39 Execution of a Command via a SYSTEM Service

Detect the usage of an intermediate service used to launch a SYSTEM-level command via `cmd.exe` or `powershell.exe`.

**id** dcb72010-c3f5-42bc-bc5e-f4f015aed1e8

**categories** detect

**confidence** medium

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

#### MITRE ATT&CK™ Mapping

**tactics** Privilege Escalation

**techniques** T1035 Service Execution, T1050 New Service

#### Query

```
registry where
  registry_path == "*\\System\\*ControlSet*\\Services\\*\\*\\ImagePath"
  and wildcard(registry_data, "%COMSPEC%", "cmd.exe", "powershell*", "cmd *")
```

#### Detonation

Atomic Red Team: T1035

#### Contributors

- Endgame

### 1.2.40 Execution of Existing Service via Command

Identifies attempts to execute an existing service by running a built-in Windows command.

**id** 45861478-8ba3-4302-9600-1970d5d8b074

**categories** enrich

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Execution

**techniques** T1035 Service Execution

#### Query

```
process where subtype.create and (  
  process_name == "sc.exe" and command_line == "* start *" or  
  process_name == "net.exe" and match(command_line, ".*? start *[\s].*") or  
  process_name == "powershell.exe" and wildcard(command_line, "*Start-Service*") or  
  process_name == "wmic.exe" and wildcard(command_line, "*service*call*startservice*")  
)
```

#### Contributors

- Endgame

### 1.2.41 Execution via cmstp.exe

Identifies potentially stealthy execution via the Microsoft Connection Manager Profile Installer.

**id** 56c64a8c-a787-488a-a7f2-b992d332679d

**categories** enrich

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion, Execution

**techniques** T1191 CMSTP

#### Query

```
process where subtype.create and  
  process_name == "cmstp.exe" and  
  command_line == "* /s *"
```

#### Contributors

- Endgame

## 1.2.42 HH.exe execution

Identifies usage of hh.exe executing recently modified .chm files.

**id** b25aa548-7937-11e9-8f5c-d46d6d62a49e

**categories** detect

**confidence** medium

**os** windows

**created** 08/08/2019

**updated** 09/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion, Execution

**techniques** T1223 Compiled HTML File

### Query

```
sequence with maxspan=1d
  [file where file_name == "*.chm"]
  [process where subtype.create and process_name == "hh.exe" and command_line == "*
↳ *.chm*"]
```

### Detonation

Atomic Red Team: T1223

### Contributors

- Dan Beavin

## 1.2.43 Host Artifact Deletion

Adversaries may delete artifacts on a host system, including logs, browser history, or directories.

**id** 339d4a19-dfb8-4d86-89c8-6a3ac807a57f

**categories** enrich

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion

**techniques** T1070 Indicator Removal on Host

#### Query

```
process where subtype.create and (  
  (process_name == "rundll32.exe" and command_line == "*InetCpl.cpl,Clear*") or  
  (process_name == "reg.exe" and command_line == "* delete *") or  
  (process_name == "cmd.exe" and command_line == "* *rmdir *")  
)
```

#### Contributors

- Endgame

### 1.2.44 Image Debuggers for Accessibility Features

The Debugger registry key allows an attacker to launch intercept the execution of files, causing a different process to be executed. This functionality is used by attackers and often targets common programs to establish persistence.

**id** 279773ee-7c69-4043-870c-9ed731c7989a

**categories** detect

**confidence** medium

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

### MITRE ATT&CK™ Mapping

**tactics** Persistence, Privilege Escalation, Defense Evasion

**techniques** T1015 Accessibility Features, T1183 Image File Execution Options Injection

#### Query

```
registry where wildcard(registry_path,  
  "*\\Software\\Microsoft\\Windows NT\\CurrentVersion\\Image File Execution_  
↪Options\\*\\Debugger",  
  "*\\Software\\Wow6432Node\\Microsoft\\Windows NT\\CurrentVersion\\Image File_  
↪Execution Options\\*\\Debugger"  
)  
  
and wildcard(registry_path,  
  // Accessibility Features  
  "*\\sethc.exe\\*"  
  "*\\utilman.exe\\*"
```

(continues on next page)



(continued from previous page)

```
"*\\narrator.exe\\"*,
"*\\osk.exe\\"*,
"*\\magnify.exe\\"*,
"*\\displayswitch.exe\\"*,
"*\\atbroker.exe\\"*,
)
```

## Detonation

Atomic Red Team: T1015

## Contributors

- Endgame

## 1.2.45 Incoming Remote PowerShell Sessions

Incoming lateral movement via Windows Remote Management (WinRM)

**id** 3abf86e1-3ba3-4473-90ea-5fc37ff57d18

**categories** enrich

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

## MITRE ATT&CK™ Mapping

**tactics** Lateral Movement, Execution

**techniques** T1028 Windows Remote Management

## Query

```
sequence with maxspan=2s
[network where subtype.incoming and destination_port in (5985, 5986)]
[process where subtype.create and
 process_name == "wsmprovhost.exe" and parent_process_name == "svchost.exe"]
```

## Contributors

- Endgame

### 1.2.46 Indirect Command Execution

Detect indirect command execution via Program Compatibility Assistant `pcalua.exe` or `forfiles.exe`.

**id** 884a7ccd-7305-4130-82d0-d4f90bc118b6

**categories** hunt

**confidence** medium

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

#### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion

**techniques** T1202 Indirect Command Execution

---

**Note:** These processes can be used in legitimate scripts, so `| unique_count` and `| filter` are used to focus on outliers as opposed to commonly seen artifacts.

---

#### Query

```
process where subtype.create and
  parent_process_name in ("pcalua.exe", "forfiles.exe")
| unique_count command_line, process_name
| filter count < 10
```

#### Detonation

Atomic Red Team: T1202

#### Contributors

- Endgame

### 1.2.47 Installation of Port Monitor

A port monitors can be registered by calling the `AddMonitor` API with a path to a DLL. This functionality can be abused by attackers to establish persistence.

**id** dce405ba-0f30-4278-b6c6-80d57847ba6b

**categories** hunt

**confidence** low

**os** windows

**created** 7/26/2019

updated 7/26/2019

## MITRE ATT&CK™ Mapping

**tactics** Privilege Escalation, Persistence

**techniques** T1013 Port Monitors

## Query

```
registry where registry_path == "*ControlSet*\Control\Print\Monitors"
```

## Contributors

- Endgame

## 1.2.48 Installation of Security Support Provider

Adversaries can establish persistence by modifying registry keys related to the Windows Security Support Provider (SSP) configuration

**id** 43cfcfb8-e52d-4c1a-a110-3aecc09e6206

**categories** enrich

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

## MITRE ATT&CK™ Mapping

**tactics** Persistence

**techniques** T1101 Security Support Provider

## Query

```
registry where
  wildcard(registry_path,
    "*\\SYSTEM\\CurrentControlSet\\Control\\Lsa\\Security Packages*",
    "*\\SYSTEM\\CurrentControlSet\\Control\\Lsa\\OSConfig\\Security Packages*")
```

## Contributors

- Endgame

### 1.2.49 Installation of Time Providers

Attackers may establish persistence by registering a DLL with Windows as a valid time provider.

**id** 3056a14a-59d9-43d3-84b5-738b4b8c3dd7  
**categories** enrich  
**confidence** low  
**os** windows  
**created** 7/26/2019  
**updated** 7/26/2019

#### MITRE ATT&CK™ Mapping

**tactics** Persistence  
**techniques** T1209 Time Providers

#### Query

```
registry where  
  registry_path == "*\\System\\CurrentControlSet\\Services\\W32Time\\TimeProviders\\*"
```

#### Contributors

- Endgame

### 1.2.50 Installing Custom Shim Databases

Identifies the installation of custom Application Compatibility Shim databases.

**id** 0e9a0a32-acf4-4969-9828-215a692c436e  
**categories** detect  
**confidence** medium  
**os** windows  
**created** 11/30/2018  
**updated** 11/30/2018

#### MITRE ATT&CK™ Mapping

**tactics** Persistence, Privilege Escalation  
**techniques** T1138 Application Shimming

## Query

```
registry where registry_path == "*\\SOFTWARE\\Microsoft\\Windows_
↳NT\\CurrentVersion\\AppCompatFlags\\Custom\\*.sdb"
  and not event of [process where subtype.create and

                                // Ignore legitimate usage of sdbinst.exe
                                not (process_name == "sdbinst.exe" and parent_process_name ==
↳"msiexec.exe")
                                ]
```

## Detonation

Atomic Red Team: T1138

## Contributors

- Endgame

### 1.2.51 InstallUtil Execution

InstallUtil may be abused to bypass process whitelisting or proxy the execution of code through a trusted Windows utility.

**id** b937f762-466f-4242-a461-d68e6e4bfc5a

**categories** hunt

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

## MITRE ATT&CK™ Mapping

**tactics** Execution, Defense Evasion

**techniques** T1118 InstallUtil

## Query

```
process where subtype.create and
  process_name == "installutil.exe" and
  command_line == "* *"
| unique parent_process_name, command_line
```

## Contributors

- Endgame

### 1.2.52 Interactive AT Job

Detect an interactive AT job, which may be used as a form of privilege escalation.

**id** d8db43cf-ed52-4f5c-9fb3-c9a4b95a0b56  
**categories** detect  
**confidence** medium  
**os** windows  
**created** 11/30/2018  
**updated** 11/30/2018

#### MITRE ATT&CK™ Mapping

**tactics** Privilege Escalation  
**techniques** T1053 Scheduled Task

---

#### Note:

As of Windows 8, the `at .exe` command was deprecated and prints the error message `The AT command has been deprecated. Please use schtasks.exe instead.`

---

#### Query

```
process where subtype.create and  
process_name == "at.exe" and command_line == "* interactive *"
```

#### Detonation

Atomic Red Team: T1053

#### Contributors

- Endgame

#### References

- <https://blogs.technet.microsoft.com/supportingwindows/2013/07/05/whats-new-in-task-scheduler-for-windows-8-server-2012/>

### 1.2.53 Launch Daemon Persistence

An adversary can maintain persistence by installing a new launch daemon that can be configured to execute upon startup

**id** 24cb8b7c-92fe-4d62-af0e-d3de993cd48b  
**categories** enrich

**confidence** low

**os** macos

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Privilege Escalation, Persistence

**techniques** T1160 Launch Daemon

### Query

```
process where subtype.create and  
parent_process_name == "launchd"
```

### Contributors

- Endgame

## 1.2.54 Loading Kernel Modules with kextload

Identify activity related to loading kernel modules on MacOS via the kextload command

**id** deca3ab9-93f2-4e1e-b782-97863bc26089

**categories** hunt

**confidence** low

**os** macos

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Persistence

**techniques** T1215 Kernel Modules and Extensions

### Query

```
process where subtype.create and  
process_name == "kextload"
```

### Contributors

- Endgame

### 1.2.55 Local Job Scheduling Paths

On Linux and macOS systems, multiple methods are supported for creating pre-scheduled and periodic background jobs.

**id** 01fa72dc-5ce4-443b-96f9-703edfeefa5d  
**categories** enrich  
**confidence** low  
**os** macos, linux  
**created** 7/26/2019  
**updated** 7/26/2019

#### MITRE ATT&CK™ Mapping

**tactics** Execution, Persistence  
**techniques** T1168 Local Job Scheduling

#### Query

```
file where wildcard(file_path, "/etc/crontab", "/etc/cron.d", "*LaunchDaemons*")
```

#### Contributors

- Endgame

### 1.2.56 Local Job Scheduling Process

On Linux and macOS systems, multiple methods are supported for creating pre-scheduled and periodic background jobs.

**id** 7f490015-20b2-43e3-acf7-e2f2d098505d  
**categories** enrich  
**confidence** low  
**os** macos, linux  
**created** 7/26/2019  
**updated** 7/26/2019

#### MITRE ATT&CK™ Mapping

**tactics** Execution, Persistence  
**techniques** T1168 Local Job Scheduling



## Query

```
process where subtype.create and  
  process_name in ("cron", "at", "launchd")
```

## Contributors

- Endgame

## 1.2.57 Logon Scripts with UserInitMprLogonScript

Detect modification of Windows logon scripts stored in HKCU\Environment\UserInitMprLogonScript and trigger when a user logs in.

**id** 54fff7e8-f81d-4169-b820-4cbff0133e2d  
**categories** detect  
**confidence** medium  
**os** windows  
**created** 11/30/2018  
**updated** 11/30/2018

## MITRE ATT&CK™ Mapping

**tactics** Persistence  
**techniques** T1037 Logon Scripts

## Query

```
registry where registry_path == "*\\Environment\\UserInitMprLogonScript"
```

## Detonation

Atomic Red Team: T1037

## Contributors

- Endgame

## 1.2.58 LSA Authentication Package

Adversaries can use the auto-start mechanism provided by LSA Authentication Packages for persistence.

**id** 077b1d1b-34ff-42d2-bd48-b0e6cdd1a359  
**categories** enrich

**confidence** low  
**os** windows  
**created** 7/26/2019  
**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Persistence  
**techniques** T1131 Authentication Package

### Query

```
registry where hive.hklm and  
registry_path == "*ControlSet*\\Control\\Lsa\\Authentication Packages*"
```

### Contributors

- Endgame

## 1.2.59 LSASS Memory Dumping

Detect creation of dump files containing the memory space of lsass.exe, which contains sensitive credentials.

**id** 210b4ea4-12fc-11e9-8d76-4d6bb837cda4  
**categories** detect  
**confidence** high  
**os** windows  
**created** 01/07/2019  
**updated** 01/07/2019

### MITRE ATT&CK™ Mapping

**tactics** Credential Access  
**techniques** T1003 Credential Dumping

### Query

```
file where file_name == "lsass*.dmp" and process_name != "werfault.exe"
```

### Detonation

Atomic Red Team: T1003

## Contributors

- Tony Lambert

### 1.2.60 LSASS Memory Dumping via ProcDump.exe

Identifies usage of Sysinternals `procdump.exe` to export the memory space of `lsass.exe` which contains sensitive credentials.

**id** 1e1ef6be-12fc-11e9-8d76-4d6bb837cda4

**categories** detect

**confidence** high

**os** windows

**created** 01/07/2019

**updated** 01/07/2019

## MITRE ATT&CK™ Mapping

**tactics** Credential Access

**techniques** T1003 Credential Dumping

## Query

```
process where subtype.create and  
  process_name == "procdump*.exe" and command_line == "*lsass*"
```

## Detonation

Atomic Red Team: T1003

## Contributors

- Tony Lambert

### 1.2.61 Modification of Boot Configuration

Identifies use of the `bcdedit` command to delete boot configuration data. This tactic is sometimes used as by malware or an attacker as a destructive technique.

**id** c4732632-9c1d-4980-9fa8-1d98c93f918e

**categories** detect

**confidence** low

**os** windows

**created** 11/30/2018

**updated** 05/17/2019

### MITRE ATT&CK™ Mapping

**tactics** Impact

**techniques** T1490 Inhibit System Recovery

### Query

```
process where subtype.create and
  process_name == "bcdedit.exe" and command_line == "*set *" and
  (command_line == "* bootstatuspolicy *ignoreallfailures*" or command_line == "*_
↵recoveryenabled* no*")
```

### Detonation

Atomic Red Team: T1490

### Contributors

- Endgame

## 1.2.62 Modification of ld.so.preload

Identifies modification of ld.so.preload for shared object injection. This technique is used by attackers to load arbitrary code into processes.

**id** fd9b987a-1101-4ed3-bda6-a70300eaf57e

**categories** detect

**confidence** medium

**os** linux

**created** 05/17/2019

**updated** 05/17/2019

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion

**techniques** T1055 Process Injection

### Query

```
file where file_path="/etc/ld.so.preload"
```

## Detonation

Atomic Red Team: T1055

### Contributors

- Tony Lambert

## 1.2.63 Modification of Logon Scripts from Registry

Windows allows logon scripts to be run whenever a specific user or group of users log into a system.

**id** af99d7ec-b1c7-4648-9188-063ca27544ac  
**categories** enrich  
**confidence** low  
**os** windows  
**created** 7/26/2019  
**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Lateral Movement, Persistence  
**techniques** T1037 Logon Scripts

### Query

```
registry where registry_path == "*\\Environment\\UserInitMprLogonScript"
```

### Contributors

- Endgame

## 1.2.64 Modification of rc.common Script

During the boot process, macOS executes `source /etc/rc.common`, which is a shell script containing various utility functions. Adversaries can use the `rc.common` file as a way to hide code for persistence.

**id** 11db63f4-15eb-47f7-8e69-e4879bace2b0  
**categories** enrich  
**confidence** low  
**os** macos  
**created** 7/26/2019  
**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Persistence

**techniques** T1163 Rc.common

#### Query

```
file where file_name == "rc.common"
```

#### Contributors

- Endgame

### 1.2.65 Modifications of .bash\_profile and .bashrc

Detect modification of .bash\_profile and .bashrc files for persistent commands

**id** 3567621a-1564-11e9-8e67-d46d6d62a49e

**categories** hunt

**confidence** low

**os** linux, macos

**created** 01/10/2019

**updated** 01/10/2019

### MITRE ATT&CK™ Mapping

**tactics** Persistence

**techniques** T1156 .bash\_profile and .bashrc

#### Query

```
file where subtype.modify and  
(file_name == ".bash_profile" or file_name == ".bashrc")
```

#### Detonation

Atomic Red Team: T1156

#### Contributors

- Tony Lambert

## 1.2.66 Mounting Hidden Shares

Identifies enumeration of mounted shares with the built-in Windows tool `net .exe`.

**id** 9b3dd402-891c-4c4d-a662-28947168ce61

**categories** detect

**confidence** low

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

### MITRE ATT&CK™ Mapping

**tactics** Lateral Movement

**techniques** T1077 Windows Admin Shares

### Query

```
process where subtype.create and
  (process_name == "net.exe" or (process_name == "net1.exe" and parent_process_name !
  ↳= "net.exe")) and
  (command_line == "* use" or command_line == "* use *") and

  // since this command is looking for discovery only, we want to ignore mounting_
  ↳shares
  command_line == "* \\*\\*\\*"
| unique parent_process_path, command_line, user_name
```

### Detonation

Atomic Red Team: T1077

### Contributors

- Endgame

## 1.2.67 Mounting Windows Hidden Shares with net.exe

Identifies hidden Windows Admin Network shares

**id** 8e7c9bce-565b-4ee1-bb70-37dc61afc8d0

**categories** hunt

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Lateral Movement

**techniques** T1077 Windows Admin Shares

#### Query

```
process where subtype.create and
  (process_name == "net.exe" or (process_name == "net1.exe" and parent_process_name !
  ↳= "net.exe")) and
  (command_line == "* use \\\\*\*\$*" or command_line == "* use \\\\*/*$*")
```

#### Contributors

- Endgame

### 1.2.68 MS Office Template Injection

Microsoft's Open Office XML (OOXML) specification defines an XML-based format for Office documents. Adversaries may abuse this technology to initially conceal malicious code to be executed via documents.

**id** bba65411-cf61-4d7c-a9a8-a2021684e9ca

**categories** detect

**confidence** low

**os** windows

**created** 02/12/2020

**updated** 02/12/2020

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion

**techniques** T1221 Template Injection

#### Query

```
sequence by unique_pid
  [process where process_name in ("winword.exe", "excel.exe", "powerpnt.exe")]
  [dns where not wildcard(query_name, "*.microsoft.com", "*.skype.com")]
  [network where true]
```

#### Contributors

- Daniel Stepanic



## References

- <https://www.elastic.co/blog/playing-defense-against-gamaredon-group>

### 1.2.69 Mshta Descendant of Microsoft Office

Identifies the execution of `mshta.exe` as a descendant of a Microsoft Office process.

**id** d49fc9fe-df80-416d-a861-0be02bef0df5

**categories** detect

**confidence** medium

**os** windows

**created** 12/04/2019

**updated** 12/04/2019

### MITRE ATT&CK™ Mapping

**tactics** Execution, Defense Evasion, Command and Control

**techniques** T1170 Mshta

### Query

```
process where subtype.create and process_name == "mshta.exe"
  and descendant of
    [process where process_name in ("outlook.exe", "winword.exe", "excel.exe",
  ↪ "powerpnt.exe")]
```

### Contributors

- Daniel Stepanic

### References

- <https://www.elastic.co/blog/embracing-offensive-tooling-building-detections-against-koadic-using-eql>

### 1.2.70 Mshta Network Connections

Identifies suspicious `mshta.exe` commands that make outbound network connections.

**id** 6bc283c4-21f2-4aed-a05c-a9a3ffa95dd4

**categories** detect

**confidence** medium

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

### MITRE ATT&CK™ Mapping

**tactics** Execution, Defense Evasion, Command and Control

**techniques** T1170 Mshta

#### Query

```
sequence by unique_pid
  [process where subtype.create and process_name == "mshta.exe" and command_line ==
  ↳ "*javascript*"]
  [network where process_name == "mshta.exe"]
```

#### Detonation

Atomic Red Team: T1170

#### Contributors

- Endgame

### 1.2.71 Network Service Scanning via Port

Network Service Scanning via incoming network port scanning

**id** 4f64ef9e-ee9b-4245-a3f4-777e550ebb37

**categories** hunt

**confidence** low

**os** windows, macos, linux

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Discovery

**techniques** T1046 Network Service Scanning

#### Query

```
network where subtype.incoming
| unique unique_pid destination_port
| unique_count unique_pid
| filter count > 25
```

## Contributors

- Endgame

### 1.2.72 Non-browser processes making DNS requests to Dynamic DNS Providers

Identifies non-browser processes making DNS requests to Dynamic DNS Providers used by GAMAREDON GROUP.

**id** de828f75-33bb-41ab-bc52-92dc2e0ef58b

**categories** detect

**confidence** low

**os** windows

**created** 02/12/2020

**updated** 02/12/2020

## MITRE ATT&CK™ Mapping

**tactics** Command and Control

**techniques** T1071 Standard Application Layer Protocol

## Query

```
dns where wildcard(query_name, "*.ddns.net", "*.hopto.org", "*.bounceme.net") and
  process_name not in ("chrome.exe", "iexplore.exe", "firefox.exe")
| unique unique_pid
```

## Contributors

- Daniel Stepanic

## References

- <https://www.elastic.co/blog/playing-defense-against-gamaredon-group>

### 1.2.73 Office Application Startup via Template File Modification

Adversaries can modify default Microsoft Office templates in order to establish persistence

**id** d763c9bb-c0f7-4a4f-82b0-06105e178afa

**categories** enrich

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Persistence

**techniques** T1137 Office Application Startup

#### Query

```
file where not subtype.delete and
  wildcard(file_path,
    ".*\\Users\\*\\AppData\\Roaming\\Microsoft\\Templates\\Normal.dotm",
    ".*\\Users\\AppData\\Roaming\\Microsoft\\Excel\\XLSTART\\PERSONAL.XLSB",
  )
```

#### Contributors

- Endgame

### 1.2.74 Office Application Startup via Template Registry Modification

Adversaries can modify Microsoft Office-related registry keys to establish persistence.

**id** 100e0ff0-fae0-4dc0-998d-c168d7e4dcb7

**categories** enrich

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Persistence

**techniques** T1137 Office Application Startup

#### Query

```
registry where wildcard(registry_path,
  ".*\\Software\\Microsoft\\Office\\*\\Outlook\\Today\\UserDefinedUrl",
  ".*\\Software\\Microsoft\\Office\\*\\Excel\\Options\\Open",
  ".*\\Software\\Microsoft\\Office\\*\\PowerPoint\\AddIns",
  ".*\\Software\\Microsoft\\Office\\*\\Addins\\*",
  ".*\\SOFTWARE\\Microsoft\\Office\\*\\Excel\\Options",
  ".*\\Software\\Microsoft\\VBA\\VBE\\*\\Addins\\*")
```

#### Contributors

- Endgame

## 1.2.75 Password Policy Enumeration

Identifies enumeration of local or global password policies using built-in commands.

**id** 94a5cbe1-851a-4b8f-bd9c-04c62097ae5e  
**categories** enrich  
**confidence** low  
**os** linux  
**created** 7/26/2019  
**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Discovery  
**techniques** T1201 Password Policy Discovery

### Query

```
process where subtype.create and (  
  process_name == "chage" and command_line == "* -l *" or  
  process_name == "cat" and command_line == "*/etc/pam.d/common-password*"  
)
```

### Contributors

- Endgame

## 1.2.76 Persistence via AppInit DLL

Detect registry modifications of the AppInit\_Dlls key, which is used by attackers to maintain persistence. AppInit DLLs are loaded into every process that uses the common library `user32.dll`.

**id** 822dc4c5-b355-4df8-bd37-29c458997b8f  
**categories** detect  
**confidence** low  
**os** windows  
**created** 11/30/2018  
**updated** 11/30/2018

### MITRE ATT&CK™ Mapping

**tactics** Persistence, Privilege Escalation  
**techniques** T1103 AppInit DLLs

### Query

```
registry where wildcard(registry_path,
    "*\\SOFTWARE\\Microsoft\\Windows NT\\CurrentVersion\\Windows\\AppInit_Dlls",
    "*\\SOFTWARE\\Wow6432Node\\Microsoft\\Windows_
↵NT\\CurrentVersion\\Windows\\AppInit_Dlls"
)
and not wildcard(process_path, "*\\system32\\msiexec.exe", "*\\syswow64\\msiexec.exe
↵")
| unique registry_data
```

### Detonation

Atomic Red Team: T1103

### Contributors

- Endgame

## 1.2.77 Persistence via NetSh Key

The tool NetShell allows for the creation of helper DLLs, which are loaded into `netsh.exe` every time it executes. This is used by attackers to establish persistence.

```
id 5f9a71f4-f5ef-4d35-aff8-f67d63d3c896
categories detect
confidence medium
os windows
created 11/30/2018
updated 11/30/2018
```

### MITRE ATT&CK™ Mapping

```
tactics Persistence
techniques T1128 Netsh Helper DLL
```

### Query

```
registry where registry_path == "*\\Software\\Microsoft\\NetSh\\*"
```

### Detonation

Atomic Red Team: T1128

## Contributors

- Endgame

### 1.2.78 Persistence via Screensaver

Detect persistence via screensaver when attacker writes payload to registry within screensaver key path.

**id** dd2eee76-9b44-479e-9860-435357e82db8

**categories** detect

**confidence** medium

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

## MITRE ATT&CK™ Mapping

**tactics** Persistence

**techniques** T1180 Screensaver

## Query

```
registry where registry_path == "*\\Control Panel\\Desktop\\SCRNSAVE.EXE"

// Ignore when the screensaver is legitimately set via the dialog
and not event of [ process where subtype.create
                    and process_path == "\\system32\\rundll32.exe"
                    and parent_process_path == "\\explorer.exe"
                    and command_line == "shell32.dll,Control_RunDLL desk.cpl,
↪ScreenSaver, *"
                    ]
```

## Detonation

Atomic Red Team: T1180

## Contributors

- Endgame

## References

- <https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1180/T1180.yaml>

### 1.2.79 Persistent process via Launch Agent

An adversary can establish persistence by installing a new launch agent that executes at login by using `launchd` or `launchctl` to load a plist into the appropriate directories

```
id 8b3a3f3b-f4f0-4cd4-82f4-28f79a3cf95b
categories enrich
confidence low
os macos
created 7/26/2019
updated 7/26/2019
```

#### MITRE ATT&CK™ Mapping

```
tactics Persistence
techniques T1159 Launch Agent
```

#### Query

```
file where not subtype.delete and
  file_path == "*/library/launchagents/*"
```

#### Contributors

- Endgame

### 1.2.80 Plist Modification

Property list (plist) files contain all of the information that macOS and OS X uses to configure applications and services. Adversaries can modify these plist files to point to their own code, can use them to execute their code in the context of another user, bypass whitelisting procedures, or even use them as a persistence mechanism.

```
id 9424fa5e-466a-40df-bb69-7cf31b7bd398
categories enrich
confidence low
os macos
created 7/26/2019
updated 7/26/2019
```

#### MITRE ATT&CK™ Mapping

```
tactics Privilege Escalation, Defense Evasion, Persistence
techniques T1150 Plist Modification
```



## Query

```
file where file_name == "*Library/Preferences/*.plist"
```

## Contributors

- Endgame

### 1.2.81 Potential Gatekeeper Bypass

In macOS, when applications or programs are downloaded from the internet, there is a special attribute set on the file. This attribute is read by Apple's Gatekeeper defense program at execution time.

**id** a4fe6af5-bc33-4e72-8241-eea885b95c46

**categories** detect

**confidence** low

**os** macos

**created** 7/26/2019

**updated** 7/26/2019

## MITRE ATT&CK™ Mapping

**tactics** Defense Evasion

**techniques** T1144 Gatekeeper Bypass

## Query

```
process where subtype.create and
  process_name == "xattr" and
  command_line == "*com.apple.quarantine*"
| unique command_line
```

## Contributors

- Endgame

### 1.2.82 Process Discovery via Built-In Applications

Built-in tools can be used to discover running processes on an endpoint

**id** 737c7bed-364f-4b47-a0aa-763c80c8aa6c

**categories** enrich

**confidence** low

**os** macos, linux

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Discovery

**techniques** T1057 Process Discovery, T1063 Security Software Discovery

### Query

```
process where subtype.create and
  (process_name in ("ps", "pstree", "htop", "pgrep") or
   match(command_line, "?.*? /proc/\d+"))
```

### Contributors

- Endgame

## 1.2.83 Process Discovery via Windows Tools

Attackers will enumerate running processes to gain further comprehension of the environment.

**id** 555a76e1-d5fe-44b9-a6bc-d275c4c446cc

**categories** enrich

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Discovery

**techniques** T1057 Process Discovery, T1063 Security Software Discovery

### Query

```
process where subtype.create and (
  process_name == "tasklist.exe" and not matchLite("?.* [-/]svc", command_line) or
  process_name == "quser.exe" or
  (process_name == "powershell.exe" and command_line == "*Get-Process*")
)
```

### Contributors

- Endgame

## 1.2.84 Processes Running with Unusual Extensions

Processes should always be executing with PE extensions, such as `.exe`, so any execution from non-PE extensions, such as `.gif` are immediately suspicious.

**id** 251c26ff-658b-42d1-a808-bafcd4b52284  
**categories** detect  
**confidence** low  
**os** windows  
**created** 7/26/2019  
**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion  
**techniques** T1036 Masquerading

### Query

```
process where subtype.create
and wildcard(process_name,
    "*.pif", "*.pdf", "*.docx", "*.doc",
    "*.xlsx", "*.xls", "*.pptx", "*.ppt",
    "*.txt", "*.rtf", "*.gif", "*.jpg",
    "*.png", "*.bmp", "*.vbs", "*.vbe",
    "*.bat", "*.js", "*.cmd",
    "*.wsh", "*.ps1", "*"
)
```

### Contributors

- Endgame

## 1.2.85 Processes with Trailing Spaces

Identifies processes running with a trailing space, which can be used to look like an ordinary file while evading default file handlers.

**id** 391c27cf-68d5-4416-9315-cdfde096a33b  
**categories** detect  
**confidence** low  
**os** macos, linux  
**created** 7/26/2019  
**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion, Execution

**techniques** T1151 Space after Filename

#### Query

```
process where subtype.create  
and process_name == "*" "
```

#### Contributors

- Endgame

### 1.2.86 Proxied Execution via Signed Scripts

Signed script scripts such as PubPrn.vbs can be used to proxy execution from a remote site while bypassing signature validation restrictions and potentially application whitelisting.

**id** 0d62a884-1052-44d0-a76c-1f4845e348d2

**categories** enrich

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion, Execution

**techniques** T1216 Signed Script Proxy Execution

#### Query

```
process where subtype.create and  
  process_name in ("cscript.exe", "wscript.exe") and  
  command_line == "*" *.vbs* *script:http*
```

#### Contributors

- Endgame

## 1.2.87 Reading the Clipboard with pbpaste

Adversaries may collect data stored in the clipboard from users copying information within or between applications.

**id** 4e026838-f673-4a5b-b380-615d624fbd00  
**categories** enrich  
**confidence** low  
**os** macos  
**created** 7/26/2019  
**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Collection  
**techniques** T1115 Clipboard Data

### Query

```
process where subtype.create and process_name == "pbpaste"
```

### Contributors

- Endgame

## 1.2.88 Registration of a Password Filter DLL

Identifies the installation of password filter DLLs which may be used to steal credentials from LSA.

**id** ae6ae50f-69f3-4e85-bfe2-2db9d1422517  
**categories** detect  
**confidence** low  
**os** windows  
**created** 7/26/2019  
**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Credential Access  
**techniques** T1174 Password Filter DLL

### Query

```
registry where hive.hklm and
  registry_path == "*SYSTEM\\ControlSet\\Control\\Lsa\\Notification Packages*"
| unique registry_path, process_path
```

### Contributors

- Endgame

## 1.2.89 Registration of Winlogon Helper DLL

A winlogon registry key was modified to establish persistence.

**id** 46de6f8f-e30e-45f7-a136-7ab140c9af08  
**categories** hunt  
**confidence** low  
**os** windows  
**created** 7/26/2019  
**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Persistence  
**techniques** T1004 Winlogon Helper DLL

### Query

```
registry where
  wildcard(registry_path,
    "*\\Software[Wow6432Node]Microsoft\\Windows NT\\CurrentVersion\\Winlogon\\"*
↪ ",
    "*\\Software\\Microsoft\\Windows NT\\CurrentVersion\\Winlogon\\"*")
```

### Contributors

- Endgame

## 1.2.90 Registry Persistence via Run Keys

Adversaries can establish persistence by adding an entry to the “run keys” in the registry or startup folder. The referenced program will be executed when a user logs in.

**id** c457d0c5-3ec8-4e9e-93f5-6ddcbfeec498  
**categories** enrich  
**confidence** low  
**os** windows

**created** 7/26/2019

**updated** 7/26/2019

## MITRE ATT&CK™ Mapping

**tactics** Persistence

**techniques** T1060 Registry Run Keys / Startup Folder

### Query

```
registry where
  registry_path == "*\\Software\\Microsoft\\Windows\\CurrentVersion\\Run*"
```

### Contributors

- Endgame

## 1.2.91 Registry Persistence via Shell Folders

Adversaries can establish persistence by adding an entry to the “run keys” in the registry or startup folder. The referenced program will be executed when a user logs in.

**id** f8b1720c-7116-4ec3-b38a-402f984e4972

**categories** detect

**confidence** low

**os** windows

**created** 7/22/2019

**updated** 7/22/2019

## MITRE ATT&CK™ Mapping

**tactics** Persistence

**techniques** T1060 Registry Run Keys / Startup Folder

### Query

```
registry where
  registry_path == "\\Software\\Microsoft\\Windows\\CurrentVersion\\Explorer\\*Shell_
↳Folders*"
```

### Contributors

- Endgame

### 1.2.92 Registry Preparation of Event Viewer UAC Bypass

Identifies preparation for User Account Control (UAC) bypass via Event Viewer registry hijacking. Attackers bypass UAC to stealthily execute code with elevated permissions.

**id** f90dd84d-6aa1-4ffd-8f0e-933f51c20fbe

**categories** detect

**confidence** low

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

#### MITRE ATT&CK™ Mapping

**tactics** Privilege Escalation

**techniques** T1088 Bypass User Account Control

#### Query

```
registry where
  registry_path == "*\\MSCFile\\shell\\open\\command\\" and

  // Ignore cases where the original avalue is restored
  registry_data != '*\\system32\\mmc.exe \"%1\"*'

  // SYSTEM will never need to bypass uac
and not user_sid in ("S-1-5-18", "S-1-5-19", "S-1-5-20")
```

#### Detonation

Atomic Red Team: T1088

#### Contributors

- Endgame

### 1.2.93 RegSvr32 Scriptlet Execution

Detect regsvr32 loading a script object (scrobj).

**id** 82200c71-f3c3-4b6c-aead-9cafeab602f5

**categories** detect

**confidence** medium

**os** windows

**created** 11/30/2018

**updated** 11/30/2018



## MITRE ATT&CK™ Mapping

**tactics** Execution

**techniques** T1117 Regsvr32

### Query

```
process where subtype.create and
  process_name == "regsvr32.exe" and
  wildcard(command_line, "*scrobj*", "*/i:*", "*/i:*", "*/i:*")
```

### Detonation

Atomic Red Team: T1117

### Contributors

- Endgame

## 1.2.94 Remote Desktop Protocol Hijack

Identifies possible Remote Desktop Protocol session hijacking

**id** 46ff4da0-2f55-4023-8de3-1709fbd33f1d

**categories** hunt

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

## MITRE ATT&CK™ Mapping

**tactics** Lateral Movement

**techniques** T1076 Remote Desktop Protocol

### Query

```
process where subtype.create and
  process_name == "tscon.exe" and command_line == "* *"
```

### Contributors

- Endgame

### 1.2.95 Remote Execution via WMIC

Identifies use of `wmic.exe` to run commands on remote hosts.

**id** 07b1481c-2a20-4274-a64e-effcd40941a5  
**categories** detect  
**confidence** low  
**os** windows  
**created** 11/30/2018  
**updated** 11/30/2018

#### MITRE ATT&CK™ Mapping

**tactics** Lateral Movement, Execution  
**techniques** T1047 Windows Management Instrumentation

#### Query

```
process where subtype.create and process_name == "wmic.exe" and  
(command_line == "* /node:*" or command_line == "* -node:*") and  
(command_line == "* *process* call *")
```

#### Contributors

- Endgame

### 1.2.96 Remote System Discovery Commands

Commands used to obtain information about the remote system.

**id** 9be90e44-c0f7-4fd2-9378-be00c25a02d7  
**categories** enrich  
**confidence** low  
**os** windows  
**created** 7/26/2019  
**updated** 7/26/2019

#### MITRE ATT&CK™ Mapping

**tactics** Discovery  
**techniques** T1018 Remote System Discovery

## Query

```
process where subtype.create and (
  process_name == "nbtstat.exe" and wildcard(command_line, "* -n*", "* -s*") or
  process_name == "arp.exe" and command_line == "* -a*"
)
```

## Contributors

- Endgame

## 1.2.97 Remote Terminal Sessions

An adversary may use Valid Accounts to log into a service specifically designed to accept remote connections.

**id** 5c310aff-d4a8-43fb-beed-b17dab1f1df0

**categories** enrich

**confidence** low

**os** windows, macos, linux

**created** 7/26/2019

**updated** 7/26/2019

## MITRE ATT&CK™ Mapping

**tactics** Lateral Movement

**techniques** T1021 Remote Services

## Query

```
process where subtype.create and
  process_name in ("telnet.exe", "putty.exe", "ssh")
| unique_count parent_process_name, command_line
```

## Contributors

- Endgame

## 1.2.98 Resumed Application on Reboot

Starting in Mac OS X 10.7 (Lion), users can specify certain applications to be re-opened when a user reboots their machine.

**id** 491db9c2-8b06-4076-8f9b-de44b9bae8d0

**categories** enrich

**confidence** low

**os** macos

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Persistence

**techniques** T1164 Re-opened Applications

### Query

```
file where file_name == "*Library/Preferences/com.apple.loginwindow.plist"
```

### Contributors

- Endgame

## 1.2.99 Root Certificate Install

Identifies modifications to the local trusted root certificates via known Windows tools. The install of a malicious root certificate would allow an attacker the ability to masquerade malicious files as valid signed components from any entity (e.g. Microsoft). It could also allow an attacker to decrypt SSL traffic on this machine. However, software may also install root certificates for the purpose of inspecting SSL traffic.

**id** 7a2efea5-42d9-4bb1-8e53-6e6d47167a96

**categories** hunt

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion

**techniques** T1130 Install Root Certificate

### Query

```
registry where wildcard(registry_path,  
↪ "*Software\\Microsoft\\SystemCertificates\\Root\\Certificates\\*\\Blob",  
↪ "*Software\\Microsoft\\SystemCertificates\\AuthRoot\\Certificates\\*\\Blob",  
↪ "*Software\\Policies\\Microsoft\\SystemCertificates\\Root\\Certificates\\*\\Blob",
```

(continues on next page)

(continued from previous page)

```
↔"*Software\Policies\Microsoft\SystemCertificates\AuthRoot\Certificates\*\Blob
↔")
| unique process_path, registry_path
```

## Contributors

- Endgame

### 1.2.100 SAM Dumping via Reg.exe

Identifies usage of `reg.exe` to export registry hives which contain the SAM and LSA secrets.

**id** aed95fc6-5e3f-49dc-8b35-06508613f979

**categories** detect

**confidence** low

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

## MITRE ATT&CK™ Mapping

**tactics** Credential Access

**techniques** T1003 Credential Dumping

## Query

```
process where subtype.create and
  process_name == "reg.exe" and
  (command_line == "* save *" or command_line == "* export *") and
  (command_line == "*hklm*" or command_line == "*hkey_local_machine*" ) and
  (command_line == "*\sam *" or command_line == "*\security *" or command_line ==
↔"*\\system *")
```

## Detonation

Atomic Red Team: T1003

## Contributors

- Endgame

### 1.2.101 Scheduled Task Creation via Microsoft Office Application

Identifies the creation of a scheduled task via a Microsoft Office application to establish persistence.

**id** 8e98bf09-e662-4908-b68e-5c96ad5c6860

**categories** detect

**confidence** medium

**os** windows

**created** 8/16/2019

**updated** 8/16/2019

#### MITRE ATT&CK™ Mapping

**tactics** Persistence

**techniques** T1053 Scheduled Task

#### Query

```
image_load where
  process_name in ("excel.exe", "winword.exe", "powerpnt.exe", "outlook.exe") and
  image_name == "taskschd.dll"
```

#### Contributors

- David French

#### References

- <https://medium.com/threatpunter/detecting-adversary-tradecraft-with-image-load-event-logging-8de93338c16>
- <https://twitter.com/DanielStepanic/status/1161983008582393856?s=20>
- <https://twitter.com/SBousseaden/status/1161919993652662273?s=20>

### 1.2.102 Searching for Passwords in Files

Adversaries may search local file systems and remote file shares for files containing passwords.

**id** 53de420f-7618-4330-87b1-1e57bafa7da5

**categories** enrich

**confidence** low

**os** macos, linux

**created** 7/26/2019

**updated** 7/26/2019

## MITRE ATT&CK™ Mapping

**tactics** Credential Access

**techniques** T1081 Credentials in Files

### Query

```
process where subtype.create
and process_name in ("cat", "grep")
and wildcard(command_line, "*.bash_history*", "*password*", "*passwd*")
```

### Contributors

- Endgame

## 1.2.103 Searching for Passwords in Files

Adversaries may search local file systems and remote file shares for files containing passwords.

**id** 62b7273b-67b2-4698-95b5-f6fafabc3390

**categories** detect

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

## MITRE ATT&CK™ Mapping

**tactics** Credential Access

**techniques** T1081 Credentials in Files

### Query

```
process where subtype.create and
  process_name == "findstr.exe" and command_line == "*password*"
| unique parent_process_name, command_line
```

### Contributors

- Endgame

### 1.2.104 Service Path Modification with sc.exe

Identifies usage of the sc.exe command to modify existing services.

**id** 15c17f6b-29c5-43a4-8adc-d298f2c4c141  
**categories** hunt  
**confidence** low  
**os** windows  
**created** 7/26/2019  
**updated** 7/26/2019

#### MITRE ATT&CK™ Mapping

**tactics** Persistence  
**techniques** T1031 Modify Existing Service

#### Query

```
process where subtype.create and  
  process_name == "sc.exe" and  
  wildcard(command_line, "* config *", "*binPath*")
```

#### Contributors

- Endgame

### 1.2.105 Service Stop or Disable with sc.exe

Detects when running services are stopped with the sc.exe command

**id** 591da84a-0382-40e7-afc8-12bd58c40425  
**categories** enrich  
**confidence** low  
**os** windows  
**created** 7/26/2019  
**updated** 7/26/2019

#### MITRE ATT&CK™ Mapping

**tactics** Impact  
**techniques** T1489 Service Stop



## Query

```
process where subtype.create and
  process_name == "sc.exe" and
  wildcard(command_line, "* stop*", "* config *disabled*")
```

## Contributors

- Endgame

### 1.2.106 Startup Folder Execution via VBScript

Adversaries abuse common persistence mechanisms such as placing their malware/implants into a compromised user's startup folder. This detection identifies the execution portion of GAMAREDON GROUP's technique of placing short-cut and VBScript files into this folder.

**id** 7b4bd51e-4165-43f8-b0c8-fb2d7cd9cf94

**categories** detect

**confidence** low

**os** windows

**created** 02/12/2020

**updated** 02/12/2020

## MITRE ATT&CK™ Mapping

**tactics** Persistence

**techniques** T1060 Registry Run Keys / Startup Folder

## Query

```
sequence by user_name with maxspan=90d
[file where subtype.create and file_path == "*\\Programs\\Startup\\*.vbs"]
[process where subtype.create and parent_process_name=="explorer.exe"
  and process_name == "wscript.exe" and command_line == "*\\Programs\\Startup\\*"]
```

## Contributors

- Daniel Stepanic

## References

- <https://www.elastic.co/blog/playing-defense-against-gamaredon-group>

### 1.2.107 Startup Folder Persistence with Shortcut/VBScript Files

Adversaries abuse common persistence mechanisms such as placing their malware/implants into a compromised user's startup folder. This detection identifies GAMAREDON GROUP's technique of placing shortcut and VBScript files into this folder.

**id** 5430be26-4019-4bc3-bb04-056019304dc9

**categories** detect

**confidence** low

**os** windows

**created** 02/12/2020

**updated** 02/12/2020

#### MITRE ATT&CK™ Mapping

**tactics** Persistence

**techniques** T1060 Registry Run Keys / Startup Folder

#### Query

```
file where subtype.create
  and process_name in ("powershell.exe", "wscript.exe", "cscript.exe", "cmd.exe",
↳ "winword.exe", "excel.exe", "powerpnt.exe")
  and (file_path == "*\\Programs\\Startup\\*.lnk" or
       file_path == "*\\Programs\\Startup\\*.vbs")
| unique process_name, file_path, user_name
```

#### Contributors

- Daniel Stepanic

#### References

- <https://www.elastic.co/blog/playing-defense-against-gamaredon-group>

### 1.2.108 Stopping Services with net.exe

Detects when running services are stopped with the net.exe command.

**id** 0b2ea078-b2ef-4cf7-aef1-564a63662e3b

**categories** enrich

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

## MITRE ATT&CK™ Mapping

**tactics** Impact

**techniques** T1489 Service Stop

### Query

```
process where subtype.create and
  process_name == "net.exe" and
  command_line == "* stop *"
```

### Contributors

- Endgame

## 1.2.109 Suspicious ADS File Creation

Detect suspicious creation or modification of NTFS Alternate Data Streams.

**id** 6624038b-05e6-4f9b-9830-346af38de870

**categories** detect

**confidence** medium

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

## MITRE ATT&CK™ Mapping

**tactics** Defense Evasion

**techniques** T1096 NTFS File Attributes

### Query

```
file where
  file_name == ":*" and file_name != "*:Zone.Identifier" and
  (file_name == "*.dll*" or file_name == "*.exe*")
```

### Detonation

Atomic Red Team: T1096

### Contributors

- Endgame

### 1.2.110 Suspicious Bitsadmin Job via bitsadmin.exe

Detect download of BITS jobs via bitsadmin.exe.

**id** ef9fe5c0-b16f-4384-bb61-95977799a84c  
**categories** detect  
**confidence** medium  
**os** windows  
**created** 11/30/2018  
**updated** 11/30/2018

#### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion, Persistence

**techniques** T1197 BITS Jobs

#### Query

```
process where subtype.create  
and process_name == "bitsadmin.exe"  
and wildcard(command_line, "* /download *", "*transfer*")
```

#### Detonation

Atomic Red Team: T1197

#### Contributors

- Endgame

### 1.2.111 Suspicious Bitsadmin Job via PowerShell

Detect download of BITS jobs via PowerShell.

**id** ec5180c9-721a-460f-bddc-27539a284273  
**categories** detect  
**confidence** medium  
**os** windows  
**created** 11/30/2018  
**updated** 11/30/2018

## MITRE ATT&CK™ Mapping

**tactics** Defense Evasion, Persistence

**techniques** T1197 BITS Jobs

### Query

```
process where subtype.create and
  process_name == "powershell.exe" and command_line == "*Start-BitsTransfer*
```

### Detonation

Atomic Red Team: T1197

### Contributors

- Endgame

## 1.2.112 Suspicious File Creation via Browser Extensions

Malicious browser extensions can be installed via app store downloads masquerading as legitimate extensions, social engineering, or by an adversary that has already compromised a system

**id** 7797d204-3205-4033-bac7-658fc203198d

**categories** enrich

**confidence** low

**os** macos, windows

**created** 7/26/2019

**updated** 7/26/2019

## MITRE ATT&CK™ Mapping

**tactics** Persistence

**techniques** T1176 Browser Extensions

### Query

```
file where not subtype.delete and
  wildcard(file_name, "*.exe", "*.dll", "*.ps1", "*.vbs", "*.bat") and
  wildcard(file_path,
    // windows
    "*\AppData\Local\Google\Chrome\User Data\Default\Extensions",
    ":\Program Files\Mozilla Firefox\plugins\*",
    ":\Program Files\Internet Explorer\Plugins\*",
```

(continues on next page)

(continued from previous page)

```
// macos
"/Applications/Firefox.app/Contents/MacOS/firefox/plugins/*",
"/Users/*/Library/Safari/Extensions/*",
"/Users/*/Library/Application Support/Google/Chrome/Default/Extensions/*"
)
```

### Contributors

- Endgame

### 1.2.113 Suspicious MS Office Registry Modifications

Adversaries may attempt to lower security controls around macro-enabled objects via malicious documents. By modifying these settings such as trusting future macros or disabling security warnings, adversaries increase their chances of success to re-gain access to machine.

**id** 53745477-dafc-43ba-8eaf-6578a6758794

**categories** detect

**confidence** low

**os** windows

**created** 02/12/2020

**updated** 02/12/2020

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion

**techniques** T1112 Modify Registry

### Query

```
sequence by unique_pid
  [process where process_name in ("winword.exe", "excel.exe", "powerpnt.exe")]
  [registry where wildcard(registry_path,
↪ "*\\Software\\Microsoft\\Office\\*\\Word\\Security\\AccessVBOM",
↪ "*\\Software\\Microsoft\\Office\\*\\Word\\Security\\VBAWarnings")]
| unique unique_pid
```

### Contributors

- Daniel Stepanic

### References

- <https://www.elastic.co/blog/playing-defense-against-gamaredon-group>

## 1.2.114 Suspicious Process Loading Credential Vault DLL

Identifies an unexpected process loading the Windows Credential Vault DLL in preparation of enumerating/stealing a user's saved credentials.

**id** 679560ee-0ea0-4358-bf83-e4c478d9d1c8

**categories** detect

**confidence** high

**os** windows

**created** 8/16/2019

**updated** 8/16/2019

### MITRE ATT&CK™ Mapping

**tactics** Credential Access

**techniques** T1003 Credential Dumping

### Query

```
image_load where process_name != "vaultcmd.exe" and  
image_name == "vaultcli.dll"
```

### Contributors

- David French

### References

- <https://medium.com/threatpunter/detecting-adversary-tradecraft-with-image-load-event-logging-8de93338c16>

## 1.2.115 Suspicious Script Object Execution

Identifies scrobj.dll loaded into unusual Microsoft processes, often indicating a *Squiblydoo* attack.

**id** a792cb37-fa56-43c2-9357-4b6a54b559c7

**categories** detect

**confidence** medium

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion, Execution

**techniques** T1117 Regsvr32

### Query

```
image_load where image_name == "scrobj.dll" and  
process_name in ("regsvr32.exe", "rundll32.exe", "certutil.exe")
```

### Detonation

Atomic Red Team: T1117

### Contributors

- Endgame

### References

- <https://web.archive.org/web/20170427203617/http://subt0x10.blogspot.com/2017/04/bypass-application-whitelisting-script.html>
- <https://gist.github.com/subTee/24c7d8e1ff0f5602092f58cbb3f7d302>

## 1.2.116 System Information Discovery

Detect enumeration of Windows system information via `systeminfo.exe`

**id** 4b9c2df7-87e2-4bbc-9123-9779ecb2dbf2

**categories** hunt

**confidence** medium

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

### MITRE ATT&CK™ Mapping

**tactics** Discovery

**techniques** T1082 System Information Discovery

### Query

```
process where subtype.create and process_name == "systeminfo.exe"  
| unique user_name, command_line
```



## Detonation

Atomic Red Team: T1082

### Contributors

- Endgame

## 1.2.117 System Network Connections Discovery

Adversaries may attempt to get a listing of network connections to or from a compromised system.

**id** df696af0-8d3f-4557-8278-d10f40ba7c07

**categories** enrich

**confidence** low

**os** macos, linux

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Discovery

**techniques** T1049 System Network Connections Discovery

### Query

```
process where subtype.create and
  process_name in ("netstat", "lsof", "who", "w")
| unique command_line
```

### Contributors

- Endgame

## 1.2.118 System Owner and User Discovery

Windows contains several built-in commands to report the active user. These may be used by an actor to learn privileges levels or determine if a session is active.

**id** 4d8563cb-f6cb-4758-9255-92479260031f

**categories** enrich

**confidence** low

**os** windows

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Discovery

**techniques** T1033 System Owner/User Discovery

#### Query

```
process where subtype.create and (  
  process_name in ("hostname.exe", "whoami.exe", "systeminfo.exe", "quser.exe") or  
  process_name == "cmd.exe" and wildcard(command_line, "*echo *%USERNAME%*", "*echo *  
↳%USERDOMAIN%*")  
)
```

#### Contributors

- Endgame

### 1.2.119 Trap Signals Usage

The trap command allows programs and shells to specify commands that will be executed upon receiving interrupt signals.

**id** 3ecbba23-0d1e-4870-8b9e-016b423aebee

**categories** enrich

**confidence** low

**os** macos, linux

**created** 7/26/2019

**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Execution, Persistence

**techniques** T1154 Trap

#### Query

```
process where subtype.create and  
  process_name == "trap" and command_line == "* signals*"
```

#### Contributors

- Endgame

## 1.2.120 Unload Sysmon Filter Driver with fltmc.exe

Detect the unloading of the Sysinternals Sysmon filter driver via the `unload` command line parameter.

**id** 1261d02a-ee99-4954-8404-8376a8d441b2  
**categories** detect  
**confidence** medium  
**os** windows  
**created** 11/30/2018  
**updated** 11/30/2018

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion  
**techniques** T1089 Disabling Security Tools

---

**Note:** The Sysmon driver can be installed with various service names. The analytic should be changed to reflect the installed service name if Sysmon is installed with a different name.

---

### Query

```
process where subtype.create and  
  process_name == "fltmc.exe" and command_line == "* unload *sysmon*"
```

### Detonation

Atomic Red Team: T1089

### Contributors

- Endgame

## 1.2.121 Unusual Child Process

Identifies processes launched with suspicious parents.

**id** 3b1b9720-179b-47e2-930e-d3757bbe345e  
**categories** detect  
**confidence** low  
**os** windows  
**created** 11/30/2018  
**updated** 11/30/2018

## MITRE ATT&amp;CK™ Mapping

**tactics** Defense Evasion, Execution

**techniques** T1093 Process Hollowing, T1055 Process Injection

## Query

```
process where subtype.create and
(
  (process_name == "smss.exe" and not parent_process_name in ("System", "smss.exe"))
  ↪ or
  (process_name == "csrss.exe" and not parent_process_name in ("smss.exe", "svchost.
  ↪ exe")) or
  (process_name == "wininit.exe" and parent_process_name != "smss.exe") or
  (process_name == "winlogon.exe" and parent_process_name != "smss.exe") or
  (process_name == "lsass.exe" and parent_process_name != "wininit.exe") or
  (process_name == "LogonUI.exe" and not parent_process_name in ("winlogon.exe",
  ↪ "wininit.exe")) or
  (process_name == "services.exe" and parent_process_name != "wininit.exe") or
  (process_name == "svchost.exe" and parent_process_name != "services.exe" and
    // When a 32-bit DLL is loaded, the syswow64\svchost.exe service will be called
    not (parent_process_path == "*\\system32\\svchost.exe" and process_path ==
  ↪ "*\\syswow64\\svchost.exe"))
  ) or
  (process_name == "spoolsv.exe" and parent_process_name != "services.exe") or
  (process_name == "taskhost.exe" and not parent_process_name in ("services.exe",
  ↪ "svchost.exe")) or
  (process_name == "taskhostw.exe" and not parent_process_name in ("services.exe",
  ↪ "svchost.exe")) or
  (process_name == "userinit.exe" and not parent_process_name in ("dwm.exe",
  ↪ "winlogon.exe"))
)
```

## Contributors

- Endgame

## References

- <https://web.archive.org/web/20140119132337/https://sysforensics.org/2014/01/know-your-windows-processes.html>

## 1.2.122 User Account Creation

Identifies creation of local users via the net .exe command.

**id** 014c3f51-89c6-40f1-ac9c-5688f26090ab

**categories** detect, hunt

**confidence** low

**os** windows

**created** 11/30/2018

**updated** 11/30/2018

### MITRE ATT&CK™ Mapping

**tactics** Persistence, Credential Access

**techniques** T1136 Create Account

### Query

```
process where subtype.create and
  (process_name == "net.exe" or (process_name == "net1.exe" and parent_process_name !
  ↳= "net.exe")) and
  command_line == "* user */ad"
```

### Detonation

Atomic Red Team: T1136

### Contributors

- Endgame

## 1.2.123 Volume Shadow Copy Deletion via VssAdmin

Identifies suspicious use of vssadmin.exe to delete volume shadow copies.

**id** d3a327b6-c517-43f2-8e97-1f06b7370705

**categories** detect

**confidence** medium

**os** windows

**created** 11/30/2018

**updated** 05/17/2019

### MITRE ATT&CK™ Mapping

**tactics** Impact

**techniques** T1490 Inhibit System Recovery

### Query

```
process where subtype.create and
  process_name == "vssadmin.exe" and command_line == "*delete* *shadows*"
```

### Detonation

Atomic Red Team: T1490

### Contributors

- Endgame

## 1.2.124 Volume Shadow Copy Deletion via WMIC

Identifies use of wmic for shadow copy deletion on endpoints. This commonly occurs in tandem with ransomware or other destructive attacks.

**id** 7163f069-a756-4edc-a9f2-28546dcb04b0

**categories** detect

**confidence** medium

**os** windows

**created** 11/30/2018

**updated** 05/17/2019

### MITRE ATT&CK™ Mapping

**tactics** Impact

**techniques** T1490 Inhibit System Recovery

### Query

```
process where subtype.create and  
process_name == "wmic.exe" and command_line == "* *shadowcopy* *delete*"
```

### Detonation

Atomic Red Team: T1490

### Contributors

- Endgame

## 1.2.125 Windows File Permissions Modification

File permissions are commonly managed by discretionary access control lists (DACLS) specified by the file owner. Adversaries may modify file permissions/attributes to evade intended DACLS.

**id** a099cb16-1a92-4503-9102-56cc84a51ad1

**categories** enrich

**confidence** low  
**os** windows  
**created** 7/26/2019  
**updated** 7/26/2019

### MITRE ATT&CK™ Mapping

**tactics** Defense Evasion  
**techniques** T1222 File Permissions Modification

### Query

```
process where subtype.create and (
  process_name == "attrib.exe" and command_line == "* +h*" or
  process_name == "takeown.exe" or
  process_name == "icacls.exe" and command_line == "*grant*"
)
```

### Contributors

- Endgame

## 1.2.126 Windows Network Enumeration

Identifies attempts to enumerate hosts in a network using the built-in Windows `net .exe` tool.

**id** b8a94d2f-dc75-4630-9d73-1edc6bd26fff  
**categories** detect  
**confidence** low  
**os** windows  
**created** 11/30/2018  
**updated** 11/30/2018

### MITRE ATT&CK™ Mapping

**tactics** Discovery  
**techniques** T1018 Remote System Discovery

### Query

```
process where subtype.create and
  process_name == "net.exe" and command_line == "* view*" and command_line !=
  ↳ "* \\ \\ *"
```

### Detonation

Atomic Red Team: T1018

### Contributors

- Endgame

## 1.2.127 WMI Execution via Microsoft Office Application

Identifies the execution of Windows Management Instrumentation (WMI) via a Microsoft Office application.

**id** e6be5ffe-c765-4e13-962d-7eaae07aeac

**categories** detect

**confidence** medium

**os** windows

**created** 8/16/2019

**updated** 8/16/2019

### MITRE ATT&CK™ Mapping

**tactics** Execution

**techniques** T1047 Windows Management Instrumentation

### Query

```
image_load where
  process_name in ("excel.exe", "winword.exe",
                  "powerpnt.exe", "outlook.exe") and
  image_name in ("wbemdisp.dll", "wbemcomn.dll", "wbemprox.dll",
                "wmiutils.dll", "wbemsvc.dll", "fastprox.dll")
```

### Contributors

- David French

### References

- <https://medium.com/threatpunter/detecting-adversary-tradecraft-with-image-load-event-logging-8de93338c16>



## 1.2.128 WMI Execution with Command Line Redirection

Identifies command execution via WMI with redirected output. WMI provides a method to execute a process on a local or remote host, but does not expose a way to read any console output. To get around this restriction, some administrators or attackers will execute `cmd.exe` with output redirection to a file. Then the file can be retrieved to read program output.

**id** 7c7f3114-7bdd-4477-a4e0-b5105b6babd8

**categories** detect

**confidence** medium

**os** windows

**created** 12/04/2019

**updated** 12/04/2019

### MITRE ATT&CK™ Mapping

**tactics** Collection

**techniques** T1074 Data Staged

### Query

```
sequence with maxspan=5s by unique_pid
  [process where subtype.create and process_name == "cmd.exe" and command_line == "*>*
  ↳" and
    descendant of [process where process_name == "wmiprivse.exe"]]
  [file where subtype.create and wildcard(file_name, "*.txt", "*.log")]
```

### Contributors

- Daniel Stepanic

### References

- <https://www.elastic.co/blog/embracing-offensive-tooling-building-detections-against-koadic-using-eql>

Analytic	Contributors	Updated	Tactics	Techniques
<i>Access of Outlook Email Archives</i>	Endgame	7/26/2019	Collection	T1114 Email Collection
<i>Account Discovery via Built-In Tools</i>	Endgame	7/26/2019	Discovery	T1087 Account Discovery
<i>AD Dumping via Ntdsutil.exe</i>	Tony Lambert	01/07/2019	Credential Access	T1003 Credential Dumping
<i>Adding the Hidden File Attribute with via attrib.exe</i>	Endgame	7/26/2019	Defense Evasion Persistence	T1158 Hidden Files and Directories
<i>AppCert DLLs Registry Modification</i>	Endgame	7/26/2019	Privilege Escalation Persistence	T1182 AppCert DLLs

Continued on next page

Table 1 – continued from previous page

Analytic	Contributors	Updated	Tactics	Techniques
<i>Audio Capture via PowerShell</i>	Endgame	11/30/2018	Collection	T1123 Audio Capture
<i>Audio Capture via SoundRecorder</i>	Endgame	11/30/2018	Collection	T1123 Audio Capture
<i>Bypass UAC via CMSTP</i>	Endgame	11/30/2018	Defense Evasion Execution	T1191 CMSTP T1088 Bypass User Account Control
<i>Bypass UAC via CompMgmt-Launcher</i>	Daniel Stepanic	12/04/2019	Privilege Escalation	T1088 Bypass User Account Control
<i>Bypass UAC via Fodhelper.exe</i>	Tony Lambert	05/17/2019	Privilege Escalation	T1088 Bypass User Account Control
<i>Bypass UAC via Fodhelper.exe</i>	Tony Lambert	05/17/2019	Privilege Escalation	T1088 Bypass User Account Control
<i>Bypass UAC via WSReset.exe</i>	Tony Lambert	05/17/2019	Privilege Escalation	T1088 Bypass User Account Control
<i>Change Default File Association</i>	Endgame	11/30/2018	Persistence	T1042 Change Default File Association
<i>Clearing Windows Event Logs with wevtutil</i>	Endgame	11/30/2018	Defense Evasion	T1070 Indicator Removal on Host
<i>COM Hijack via Script Object</i>	Endgame	11/30/2018	Persistence Defense Evasion	T1122 Component Object Model Hijacking
<i>Command-Line Creation of a RAR file</i>	Endgame	11/30/2018	Exfiltration	T1002 Data Compressed
<i>Control Panel Items</i>	Endgame	7/26/2019	Defense Evasion Execution	T1196 Control Panel Items
<i>Creation of an Archive with Common Archivers</i>	Endgame	7/26/2019	Collection	T1074 Data Staged
<i>Creation of Kernel Module</i>	Endgame	7/26/2019	Persistence	T1215 Kernel Modules and Extensions
<i>Creation of Scheduled Task with schtasks.exe</i>	Endgame	7/26/2019	Privilege Escalation Execution Persistence	T1053 Scheduled Task
<i>Creation or Modification of Systemd Service</i>	Endgame	7/26/2019	Persistence	T1501 Systemd Service
<i>Credential Enumeration via Credential Vault CLI</i>	David French	8/16/2019	Credential Access	T1003 Credential Dumping
<i>Delete Volume USN Journal with fsutil</i>	Endgame	11/30/2018	Defense Evasion	T1070 Indicator Removal on Host
<i>Disconnecting from Network Shares with net.exe</i>	Endgame	7/26/2019	Defense Evasion	T1126 Network Share Connection Removal

Continued on next page

Table 1 – continued from previous page

Analytic	Contributors	Updated	Tactics	Techniques
<i>Discovery and Enumeration of System Information via Rundll32</i>	Daniel Stepanic	12/04/2019	Discovery	T1087 Account Discovery T1096 NTFS File Attributes T1033 System Owner/User Discovery
<i>Discovery of a Remote System's Time</i>	Endgame	11/30/2018	Discovery	T1124 System Time Discovery
<i>Discovery of Domain Groups</i>	Endgame	7/26/2019	Discovery	T1069 Permission Groups Discovery
<i>Discovery of Network Environment via Built-in Tools</i>	Endgame	7/26/2019	Discovery	T1016 System Network Configuration Discovery
<i>Discovery of Network Environment via Built-in Tools</i>	Endgame	7/26/2019	Discovery	T1016 System Network Configuration Discovery
<i>DLL Search Order Hijacking with known programs</i>	Endgame	7/26/2019	Privilege Escalation Defense Evasion Persistence	T1038 DLL Search Order Hijacking
<i>Domain Trust Discovery</i>	Endgame	7/26/2019	Discovery	T1482 Domain Trust Discovery
<i>Domain Trust Discovery via Nltest.exe</i>	Tony Lambert	05/17/2019	Discovery	T1482 Domain Trust Discovery
<i>Encoding or Decoding Files via CertUtil</i>	Endgame	11/30/2018	Defense Evasion	T1140 Deobfuscate/Decode Files or Information
<i>Enumeration of Local Shares</i>	Endgame	11/30/2018	Discovery	T1135 Network Share Discovery
<i>Enumeration of Mounted Shares</i>	Endgame	11/30/2018	Discovery	T1049 System Network Connections Discovery
<i>Enumeration of Remote Shares</i>	Endgame	11/30/2018	Discovery	T1135 Network Share Discovery
<i>Enumeration of System Information</i>	Endgame	7/26/2019	Discovery	T1082 System Information Discovery
<i>Enumeration of System Information</i>	Endgame	7/26/2019	Discovery	T1082 System Information Discovery
<i>Executable Written and Executed by Microsoft Office Applications</i>	Daniel Stepanic	12/04/2019	Execution	T1204 User Execution T1173 Dynamic Data Exchange
<i>Execution of a Command via a SYSTEM Service</i>	Endgame	11/30/2018	Privilege Escalation	T1035 Service Execution T1050 New Service

Continued on next page

Table 1 – continued from previous page

Analytic	Contributors	Updated	Tactics	Techniques
<i>Execution of Existing Service via Command</i>	Endgame	7/26/2019	Execution	T1035 Service Execution
<i>Execution via cmstp.exe</i>	Endgame	7/26/2019	Defense Evasion Execution	T1191 CMSTP
<i>HH.exe execution</i>	Dan Beavin	09/26/2019	Defense Evasion Execution	T1223 Compiled HTML File
<i>Host Artifact Deletion</i>	Endgame	7/26/2019	Defense Evasion	T1070 Indicator Removal on Host
<i>Image Debuggers for Accessibility Features</i>	Endgame	11/30/2018	Persistence Privilege Escalation Defense Evasion	T1015 Accessibility Features T1183 Image File Execution Options Injection
<i>Incoming Remote PowerShell Sessions</i>	Endgame	7/26/2019	Lateral Movement Execution	T1028 Windows Remote Management
<i>Indirect Command Execution</i>	Endgame	11/30/2018	Defense Evasion	T1202 Indirect Command Execution
<i>Installation of Port Monitor</i>	Endgame	7/26/2019	Privilege Escalation Persistence	T1013 Port Monitors
<i>Installation of Security Support Provider</i>	Endgame	7/26/2019	Persistence	T1101 Security Support Provider
<i>Installation of Time Providers</i>	Endgame	7/26/2019	Persistence	T1209 Time Providers
<i>Installing Custom Shim Databases</i>	Endgame	11/30/2018	Persistence Privilege Escalation	T1138 Application Shimming
<i>InstallUtil Execution</i>	Endgame	7/26/2019	Execution Defense Evasion	T1118 InstallUtil
<i>Interactive AT Job</i>	Endgame	11/30/2018	Privilege Escalation	T1053 Scheduled Task
<i>Launch Daemon Persistence</i>	Endgame	7/26/2019	Privilege Escalation Persistence	T1160 Launch Daemon
<i>Loading Kernel Modules with kextload</i>	Endgame	7/26/2019	Persistence	T1215 Kernel Modules and Extensions
<i>Local Job Scheduling Paths</i>	Endgame	7/26/2019	Execution Persistence	T1168 Local Job Scheduling
<i>Local Job Scheduling Process</i>	Endgame	7/26/2019	Execution Persistence	T1168 Local Job Scheduling
<i>Logon Scripts with UserInitMprLogon-Script</i>	Endgame	11/30/2018	Persistence	T1037 Logon Scripts
<i>LSA Authentication Package</i>	Endgame	7/26/2019	Persistence	T1131 Authentication Package
<i>LSASS Memory Dumping</i>	Tony Lambert	01/07/2019	Credential Access	T1003 Credential Dumping

Continued on next page

Table 1 – continued from previous page

Analytic	Contributors	Updated	Tactics	Techniques
<i>LSASS Memory Dumping via Proc-Dump.exe</i>	Tony Lambert	01/07/2019	Credential Access	T1003 Credential Dumping
<i>Modification of Boot Configuration</i>	Endgame	05/17/2019	Impact	T1490 Inhibit System Recovery
<i>Modification of ld.so.preload</i>	Tony Lambert	05/17/2019	Defense Evasion	T1055 Process Injection
<i>Modification of Logon Scripts from Registry</i>	Endgame	7/26/2019	Lateral Movement Persistence	T1037 Logon Scripts
<i>Modification of rc.common Script</i>	Endgame	7/26/2019	Persistence	T1163 Rc.common
<i>Modifications of .bash_profile and .bashrc</i>	Tony Lambert	01/10/2019	Persistence	T1156 .bash_profile and .bashrc
<i>Mounting Hidden Shares</i>	Endgame	11/30/2018	Lateral Movement	T1077 Windows Admin Shares
<i>Mounting Windows Hidden Shares with net.exe</i>	Endgame	7/26/2019	Lateral Movement	T1077 Windows Admin Shares
<i>MS Office Template Injection</i>	Daniel Stepanic	02/12/2020	Defense Evasion	T1221 Template Injection
<i>Mshsta Descendant of Microsoft Office</i>	Daniel Stepanic	12/04/2019	Execution Defense Evasion Command and Control	T1170 Mshsta
<i>Mshsta Network Connections</i>	Endgame	11/30/2018	Execution Defense Evasion Command and Control	T1170 Mshsta
<i>Network Service Scanning via Port</i>	Endgame	7/26/2019	Discovery	T1046 Network Service Scanning
<i>Non-browser processes making DNS requests to Dynamic DNS Providers</i>	Daniel Stepanic	02/12/2020	Command and Control	T1071 Standard Application Layer Protocol
<i>Office Application Startup via Template File Modification</i>	Endgame	7/26/2019	Persistence	T1137 Office Application Startup
<i>Office Application Startup via Template Registry Modification</i>	Endgame	7/26/2019	Persistence	T1137 Office Application Startup
<i>Password Policy Enumeration</i>	Endgame	7/26/2019	Discovery	T1201 Password Policy Discovery
<i>Persistence via AppInit DLL</i>	Endgame	11/30/2018	Persistence Privilege Escalation	T1103 AppInit DLLs
<i>Persistence via NetSh Key</i>	Endgame	11/30/2018	Persistence	T1128 Netsh Helper DLL

Continued on next page

Table 1 – continued from previous page

Analytic	Contributors	Updated	Tactics	Techniques
<i>Persistence via Screensaver</i>	Endgame	11/30/2018	Persistence	T1180 Screensaver
<i>Persistent process via Launch Agent</i>	Endgame	7/26/2019	Persistence	T1159 Launch Agent
<i>Plist Modification</i>	Endgame	7/26/2019	Privilege Escalation Defense Evasion Persistence	T1150 Plist Modification
<i>Potential Gatekeeper Bypass</i>	Endgame	7/26/2019	Defense Evasion	T1144 Gatekeeper Bypass
<i>Process Discovery via Built-In Applications</i>	Endgame	7/26/2019	Discovery	T1057 Process Discovery T1063 Security Software Discovery
<i>Process Discovery via Windows Tools</i>	Endgame	7/26/2019	Discovery	T1057 Process Discovery T1063 Security Software Discovery
<i>Processes Running with Unusual Extensions</i>	Endgame	7/26/2019	Defense Evasion	T1036 Masquerading
<i>Processes with Trailing Spaces</i>	Endgame	7/26/2019	Defense Evasion Execution	T1151 Space after Filename
<i>Proxied Execution via Signed Scripts</i>	Endgame	7/26/2019	Defense Evasion Execution	T1216 Signed Script Proxy Execution
<i>Reading the Clipboard with pbpaste</i>	Endgame	7/26/2019	Collection	T1115 Clipboard Data
<i>Registration of a Password Filter DLL</i>	Endgame	7/26/2019	Credential Access	T1174 Password Filter DLL
<i>Registration of Winlogon Helper DLL</i>	Endgame	7/26/2019	Persistence	T1004 Winlogon Helper DLL
<i>Registry Persistence via Run Keys</i>	Endgame	7/26/2019	Persistence	T1060 Registry Run Keys / Startup Folder
<i>Registry Persistence via Shell Folders</i>	Endgame	7/22/2019	Persistence	T1060 Registry Run Keys / Startup Folder
<i>Registry Preparation of Event Viewer UAC Bypass</i>	Endgame	11/30/2018	Privilege Escalation	T1088 Bypass User Account Control
<i>RegSvr32 Scriptlet Execution</i>	Endgame	11/30/2018	Execution	T1117 Regsvr32
<i>Remote Desktop Protocol Hijack</i>	Endgame	7/26/2019	Lateral Movement	T1076 Remote Desktop Protocol
<i>Remote Execution via WMIC</i>	Endgame	11/30/2018	Lateral Movement Execution	T1047 Windows Management Instrumentation
<i>Remote System Discovery Commands</i>	Endgame	7/26/2019	Discovery	T1018 Remote System Discovery

Continued on next page

Table 1 – continued from previous page

Analytic	Contributors	Updated	Tactics	Techniques
<i>Remote Terminal Sessions</i>	Endgame	7/26/2019	Lateral Movement	T1021 Remote Services
<i>Resumed Application on Reboot</i>	Endgame	7/26/2019	Persistence	T1164 Re-opened Applications
<i>Root Certificate Install</i>	Endgame	7/26/2019	Defense Evasion	T1130 Install Root Certificate
<i>SAM Dumping via Reg.exe</i>	Endgame	11/30/2018	Credential Access	T1003 Credential Dumping
<i>Scheduled Task Creation via Microsoft Office Application</i>	David French	8/16/2019	Persistence	T1053 Scheduled Task
<i>Searching for Passwords in Files</i>	Endgame	7/26/2019	Credential Access	T1081 Credentials in Files
<i>Searching for Passwords in Files</i>	Endgame	7/26/2019	Credential Access	T1081 Credentials in Files
<i>Service Path Modification with sc.exe</i>	Endgame	7/26/2019	Persistence	T1031 Modify Existing Service
<i>Service Stop or Disable with sc.exe</i>	Endgame	7/26/2019	Impact	T1489 Service Stop
<i>Startup Folder Execution via VBScript</i>	Daniel Stepanic	02/12/2020	Persistence	T1060 Registry Run Keys / Startup Folder
<i>Startup Folder Persistence with Shortcut/VBScript Files</i>	Daniel Stepanic	02/12/2020	Persistence	T1060 Registry Run Keys / Startup Folder
<i>Stopping Services with net.exe</i>	Endgame	7/26/2019	Impact	T1489 Service Stop
<i>Suspicious ADS File Creation</i>	Endgame	11/30/2018	Defense Evasion	T1096 NTFS File Attributes
<i>Suspicious Bit-admin Job via bitsadmin.exe</i>	Endgame	11/30/2018	Defense Evasion Persistence	T1197 BITS Jobs
<i>Suspicious Bit-admin Job via PowerShell</i>	Endgame	11/30/2018	Defense Evasion Persistence	T1197 BITS Jobs
<i>Suspicious File Creation via Browser Extensions</i>	Endgame	7/26/2019	Persistence	T1176 Browser Extensions
<i>Suspicious MS Office Registry Modifications</i>	Daniel Stepanic	02/12/2020	Defense Evasion	T1112 Modify Registry
<i>Suspicious Process Loading Credential Vault DLL</i>	David French	8/16/2019	Credential Access	T1003 Credential Dumping
<i>Suspicious Script Object Execution</i>	Endgame	11/30/2018	Defense Evasion Execution	T1117 Regsvr32
<i>System Information Discovery</i>	Endgame	11/30/2018	Discovery	T1082 System Information Discovery

Continued on next page



Table 1 – continued from previous page

Analytic	Contributors	Updated	Tactics	Techniques
<i>System Network Connections Discovery</i>	Endgame	7/26/2019	Discovery	T1049 System Network Connections Discovery
<i>System Owner and User Discovery</i>	Endgame	7/26/2019	Discovery	T1033 System Owner/User Discovery
<i>Trap Signals Usage</i>	Endgame	7/26/2019	Execution Persistence	T1154 Trap
<i>Unload Sysmon Filter Driver with fltmc.exe</i>	Endgame	11/30/2018	Defense Evasion	T1089 Disabling Security Tools
<i>Unusual Child Process</i>	Endgame	11/30/2018	Defense Evasion Execution	T1093 Process Hollowing T1055 Process Injection
<i>User Account Creation</i>	Endgame	11/30/2018	Persistence Credential Access	T1136 Create Account
<i>Volume Shadow Copy Deletion via VssAdmin</i>	Endgame	05/17/2019	Impact	T1490 Inhibit System Recovery
<i>Volume Shadow Copy Deletion via WMIC</i>	Endgame	05/17/2019	Impact	T1490 Inhibit System Recovery
<i>Windows File Permissions Modification</i>	Endgame	7/26/2019	Defense Evasion	T1222 File Permissions Modification
<i>Windows Network Enumeration</i>	Endgame	11/30/2018	Discovery	T1018 Remote System Discovery
<i>WMI Execution via Microsoft Office Application</i>	David French	8/16/2019	Execution	T1047 Windows Management Instrumentation
<i>WMI Execution with Command Line Redirection</i>	Daniel Stepanic	12/04/2019	Collection	T1074 Data Staged

### 1.3 Atomic Blue Detections

Analytic	Contributors	Updated	Tactics	Techniques
<i>AD Dumping via Ntdsutil.exe</i>	Tony Lambert	01/07/2019	Credential Access	T1003 Credential Dumping
<i>Audio Capture via PowerShell</i>	Endgame	11/30/2018	Collection	T1123 Audio Capture
<i>Audio Capture via SoundRecorder</i>	Endgame	11/30/2018	Collection	T1123 Audio Capture
<i>Bypass UAC via CMSTP</i>	Endgame	11/30/2018	Defense Evasion Execution	T1191 CMSTP T1088 Bypass User Account Control

Continued on next page



Table 2 – continued from previous page

Analytic	Contributors	Updated	Tactics	Techniques
<i>Bypass UAC via Fodhelper.exe</i>	Tony Lambert	05/17/2019	Privilege Escalation	T1088 Bypass User Account Control
<i>Bypass UAC via Fodhelper.exe</i>	Tony Lambert	05/17/2019	Privilege Escalation	T1088 Bypass User Account Control
<i>Bypass UAC via WSReset.exe</i>	Tony Lambert	05/17/2019	Privilege Escalation	T1088 Bypass User Account Control
<i>Change Default File Association</i>	Endgame	11/30/2018	Persistence	T1042 Change Default File Association
<i>Clearing Windows Event Logs with wevtutil</i>	Endgame	11/30/2018	Defense Evasion	T1070 Indicator Removal on Host
<i>COM Hijack via Script Object</i>	Endgame	11/30/2018	Persistence Defense Evasion	T1122 Component Object Model Hijacking
<i>Command-Line Creation of a RAR file</i>	Endgame	11/30/2018	Exfiltration	T1002 Data Compressed
<i>Delete Volume USN Journal with fsutil</i>	Endgame	11/30/2018	Defense Evasion	T1070 Indicator Removal on Host
<i>Discovery of a Remote System's Time</i>	Endgame	11/30/2018	Discovery	T1124 System Time Discovery
<i>Domain Trust Discovery via Nltest.exe</i>	Tony Lambert	05/17/2019	Discovery	T1482 Domain Trust Discovery
<i>Encoding or Decoding Files via CertUtil</i>	Endgame	11/30/2018	Defense Evasion	T1140 Deobfuscate/Decode Files or Information
<i>Enumeration of Mounted Shares</i>	Endgame	11/30/2018	Discovery	T1049 System Network Connections Discovery
<i>Enumeration of Remote Shares</i>	Endgame	11/30/2018	Discovery	T1135 Network Share Discovery
<i>Execution of a Command via a SYSTEM Service</i>	Endgame	11/30/2018	Privilege Escalation	T1035 Service Execution T1050 New Service
<i>HH.exe execution</i>	Dan Beavin	09/26/2019	Defense Evasion Execution	T1223 Compiled HTML File
<i>Image Debuggers for Accessibility Features</i>	Endgame	11/30/2018	Persistence Privilege Escalation Defense Evasion	T1015 Accessibility Features T1183 Image File Execution Options Injection
<i>Indirect Command Execution</i>	Endgame	11/30/2018	Defense Evasion	T1202 Indirect Command Execution
<i>Installing Custom Shim Databases</i>	Endgame	11/30/2018	Persistence Privilege Escalation	T1138 Application Shimming
<i>Interactive AT Job</i>	Endgame	11/30/2018	Privilege Escalation	T1053 Scheduled Task

Continued on next page

Table 2 – continued from previous page

Analytic	Contributors	Updated	Tactics	Techniques
<i>Logon Scripts with UserInitMprLogon-Script</i>	Endgame	11/30/2018	Persistence	T1037 Logon Scripts
<i>LSASS Memory Dumping</i>	Tony Lambert	01/07/2019	Credential Access	T1003 Credential Dumping
<i>LSASS Memory Dumping via Proc-Dump.exe</i>	Tony Lambert	01/07/2019	Credential Access	T1003 Credential Dumping
<i>Modification of Boot Configuration</i>	Endgame	05/17/2019	Impact	T1490 Inhibit System Recovery
<i>Modification of ld.so.preload</i>	Tony Lambert	05/17/2019	Defense Evasion	T1055 Process Injection
<i>Modifications of .bash_profile and .bashrc</i>	Tony Lambert	01/10/2019	Persistence	T1156 .bash_profile and .bashrc
<i>Mounting Hidden Shares</i>	Endgame	11/30/2018	Lateral Movement	T1077 Windows Admin Shares
<i>Mshata Network Connections</i>	Endgame	11/30/2018	Execution Defense Evasion Command and Control	T1170 Mshata
<i>Persistence via AppInit DLL</i>	Endgame	11/30/2018	Persistence Privilege Escalation	T1103 AppInit DLLs
<i>Persistence via NetSh Key</i>	Endgame	11/30/2018	Persistence	T1128 Netsh Helper DLL
<i>Persistence via Screensaver</i>	Endgame	11/30/2018	Persistence	T1180 Screensaver
<i>Registry Preparation of Event Viewer UAC Bypass</i>	Endgame	11/30/2018	Privilege Escalation	T1088 Bypass User Account Control
<i>RegSvr32 Scriptlet Execution</i>	Endgame	11/30/2018	Execution	T1117 Regsvr32
<i>SAM Dumping via Reg.exe</i>	Endgame	11/30/2018	Credential Access	T1003 Credential Dumping
<i>Suspicious ADS File Creation</i>	Endgame	11/30/2018	Defense Evasion	T1096 NTFS File Attributes
<i>Suspicious Bit-admin Job via bitsadmin.exe</i>	Endgame	11/30/2018	Defense Evasion Persistence	T1197 BITS Jobs
<i>Suspicious Bit-admin Job via PowerShell</i>	Endgame	11/30/2018	Defense Evasion Persistence	T1197 BITS Jobs
<i>Suspicious Script Object Execution</i>	Endgame	11/30/2018	Defense Evasion Execution	T1117 Regsvr32
<i>System Information Discovery</i>	Endgame	11/30/2018	Discovery	T1082 System Information Discovery
<i>Unload Sysmon Filter Driver with fltmc.exe</i>	Endgame	11/30/2018	Defense Evasion	T1089 Disabling Security Tools

Continued on next page

Table 2 – continued from previous page

Analytic	Contributors	Updated	Tactics	Techniques
<i>User Account Creation</i>	Endgame	11/30/2018	Persistence Credential Access	T1136 Create Account
<i>Volume Shadow Copy Deletion via VssAdmin</i>	Endgame	05/17/2019	Impact	T1490 Inhibit System Recovery
<i>Volume Shadow Copy Deletion via WMIC</i>	Endgame	05/17/2019	Impact	T1490 Inhibit System Recovery
<i>Windows Network Enumeration</i>	Endgame	11/30/2018	Discovery	T1018 Remote System Discovery

## 1.4 Enterprise ATT&CK Matrix

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
Drive-by Compromise	AppleScript	bash_profile and .bashrc	Exploitation for Privilege Escalation	Access Token Manipulation	Account Manipulation	Account Discovery	Application Deployment Software	Audio Capture	Automated Exfiltration	Commonly Used Port	Data Destruction
		<ul style="list-style-type: none"> <li>• Modification of .bash_profile and .bashrc (l, m)</li> </ul>	<ul style="list-style-type: none"> <li>• Exploitation</li> </ul>			<ul style="list-style-type: none"> <li>• Account Discovery via Built-In Tools (l, m, w)</li> <li>• Discovery and Enumeration of System Information via Rundll32 (w)</li> </ul>		<ul style="list-style-type: none"> <li>• Audio Capture via Power-Shell (w)</li> <li>• Audio Capture via SoundRecorder (w)</li> </ul>			

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
Exploit Public-Facing Application	Command Line Interface	Accessibility Features	Image File Execution Options Injection for Accessibility Features (w)	BITS Jobs • Image Debugger for Accessibility Features (w)	Bash History <i>Suspicious Bit-sad-min Job via bit-sad-min.exe (w)</i> • <i>Suspicious Bit-sad-min Job via Power-Shell (w)</i>	Application Window Discovery	Distributed Component Object Model	Automated Collection	Data Compressed • <i>Command-Line Creation of a RAR file (w)</i>	Communication Through Remote Media	Data Encrypted for Impact
Hardware Additions	Dynamic Data Exchange	App DLLs	Cert History Injection <i>AppCert DLLs Registry Modification (w)</i>	Binary Padding	Brute Force	Browser Bookmark Discovery	Exploitation of Remote Services	Clipboard Data • <i>Reading the Clipboard with pb-paste (m)</i>	Data Encrypted	Connection Proxy	Defacement

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
Spearphishing Attachment	Execution through API	AppInit DLLs •	Setuid and Setgid <i>Persistence via AppInit DLL (w)</i>	Bypass User Account Control • <i>Bypass UAC via CM-STP (w)</i>	Credential Dumping • <i>Credential Vault DLL (w)</i> • <i>LSASS Memory Dumping (w)</i> • <i>AD Dumping via Ntdsutil.exe (w)</i> • <i>LSASS Memory Dumping via ProcDump.exe (w)</i>	Domain Trust Discovery • <i>Domain Trust Discovery (w)</i>	Logon Scripts • <i>Logon Scripts from Registry (w)</i>	Data Staged • <i>Data Staged from Registry (w)</i>	Data Transfer <i>WMI Size Ex-Limits with Command Line Redirection (w)</i> • <i>Creation of an Archive with Common Archivers (l, m)</i>	Custom Command and Control Protocol	Disk Content Wipe
106									Chapter 1. Next Steps		

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
Spearphishing Link	Execution through Module Load	Application Shim-ming <ul style="list-style-type: none"> <li>Installing Custom Shim Databases (w)</li> </ul>	Sudo	CMSTP <ul style="list-style-type: none"> <li>Execution via cm-stp.exe (w)</li> <li>Bypass UAC via CM-STP (w)</li> </ul>	Credentials in Files <ul style="list-style-type: none"> <li>Searching for passwords in Files (l, m)</li> <li>Searching for Passwords in Files (w)</li> </ul>	File and Directory Disabling	Pass the Hash	Data from Information Repositories	Exfiltration Over Alternative Protocol	Custom Cryptographic Protocol	Disk Structure Wipe
Spearphishing via Service	Exploitation for Client Execution	Authentication Package <ul style="list-style-type: none"> <li>LSA Authentication Package (w)</li> </ul>	Sudo Caching	Clear Command History	Credentials in Registry	Network Service Scanning <ul style="list-style-type: none"> <li>Network Service Scanning via Port (l, m, w)</li> </ul>	Pass the Ticket	Data from Local System	Exfiltration Over Command and Control Channel	Data Encoding	Endpoint Denial of Service

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
Supply Chain Compromise	Graphical User Interface	Bootkit		Code Signing	Exploitation for Credential Access	Network Share Discovery	Remote Desktop Protocol	Data from Network Shared Drive	Exfiltration Over Other Network Medium	Data Obfuscation	Firmware Corruption
						<ul style="list-style-type: none"> <li>• Enumeration of Local Shares (w)</li> <li>• Enumeration of Remote Shares (w)</li> </ul>	<ul style="list-style-type: none"> <li>• Remote Desktop Protocol Hijack (w)</li> </ul>				

Continued on next page



Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
Trusted Relationship	LSASS Driver	Browser Extensions <ul style="list-style-type: none"> <li>• <i>Suspicious File Creation via Browser Extensions (m, w)</i></li> </ul>		Compile After Delivery	Forced Authentication	Password Policy Discovery <ul style="list-style-type: none"> <li>• <i>Password Policy Enumeration (l)</i></li> </ul>	Remote Services <ul style="list-style-type: none"> <li>• <i>Remote Services (l, w)</i></li> </ul>	Data from Removal <ul style="list-style-type: none"> <li>• <i>Removable Media</i></li> </ul>	Exfiltration Over Physical Medium	Domain Fronting	Inhibit System Recovery <ul style="list-style-type: none"> <li>• <i>Modification of Boot Configuration (w)</i></li> <li>• <i>Volume Shadow Copy Deletion via VssAdmin (w)</i></li> <li>• <i>Volume Shadow Copy Deletion via WMIC (w)</i></li> </ul>

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
	PowerShell	Change Default File Association		Compiled HTML File	Input Prompt	Peripheral Device Discovery	Replication Through Removable Media	Email Collection	Scheduled Transfer	Domain Generation Algorithms	Network Denial of Service
		<ul style="list-style-type: none"> <li>Change Default File Association (w)</li> </ul>		<ul style="list-style-type: none"> <li>HH.exe execution (w)</li> </ul>				<ul style="list-style-type: none"> <li>Access of Outlook Email Archives (w)</li> </ul>			
	Scheduled Task	Create Account		Component Firmware	Kerberos	Permissions Groups Discovery	SSH Hijacking	Input Capture		Fallback Channels	Resource Hijacking
	<ul style="list-style-type: none"> <li>Creation of Scheduled Task with schtasks.exe (w)</li> </ul>	<ul style="list-style-type: none"> <li>User Account Creation (w)</li> </ul>				<ul style="list-style-type: none"> <li>Discovery of Domain Groups (l, m)</li> </ul>					

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
	Service Execution <ul style="list-style-type: none"> <li>• Execution of Existing Service via Command (w)</li> </ul>	DLL Search Order Hijacking <ul style="list-style-type: none"> <li>• DLL Search Order Hijacking with known programs (w)</li> </ul>		Component Object Model Hijacking <ul style="list-style-type: none"> <li>• COM Hijack via Script Object (w)</li> </ul>	Keychain Access	Process Discovery <ul style="list-style-type: none"> <li>• Process Discovery via Built-In Applications (l, m)</li> <li>• Process Discovery via Windows Tools (w)</li> </ul>	Shared Web-root Man in the Browser			Multi-Stage Channels	Runtime Data Manipulation
	Source	Dylib Hijacking		Control Panel Items <ul style="list-style-type: none"> <li>• Control Panel Items (w)</li> </ul>	LLMNR/NS Poisoning Relay	Key Registry Taint Shared Content	Screen Capture		Multi-hop Proxy	Service Stop <ul style="list-style-type: none"> <li>• Service Stop or Disable with sc.exe (w)</li> <li>• Stopping Services with net.exe (w)</li> </ul>	

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
	Third-party Software	External Remote Services		DCShadow	Network Sniffing	Remote System Discovery	Windows Admin Shares	Video Capture		Multiband Communication	Stored Data Manipulation
						<ul style="list-style-type: none"> <li>• <i>Mounting Hidden Windows Network Shares (w)</i></li> <li>• <i>Enumeration (w) Mounting Windows Hidden Shares with net.exe (w)</i></li> <li>• <i>Remote System Discovery Commands (w)</i></li> </ul>					

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
	<ul style="list-style-type: none"> <li>Trap</li> </ul>	<ul style="list-style-type: none"> <li>File System</li> <li>Trap</li> <li>Personal Mis-Us-sions</li> <li>Weak-ness (l, m)</li> </ul>		<ul style="list-style-type: none"> <li>DLL Side-Loading</li> </ul>	<ul style="list-style-type: none"> <li>Password Filter</li> <li>DLL</li> </ul>	<ul style="list-style-type: none"> <li>Security Software Dis-covery</li> <li>Registration of a Password Filter DLL (w)</li> <li>Process Discovery via Built-In Applications (l, m)</li> <li>Process Discovery via Windows Tools (w)</li> </ul>				<ul style="list-style-type: none"> <li>Multilayer Encryption</li> </ul>	<ul style="list-style-type: none"> <li>Transmitted Data Manipulation</li> </ul>

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
	User Execution <ul style="list-style-type: none"> <li>• Executable Written and Executed by Microsoft Office Applications (w)</li> </ul>	Hooking		Deobfuscation or Information <ul style="list-style-type: none"> <li>• Encoding or Decoding Files via CertUtil (w)</li> </ul>	Privileges <ul style="list-style-type: none"> <li>• Keys</li> </ul>	System Information Discovery <ul style="list-style-type: none"> <li>• Enumeration of System Information (l)</li> <li>• Enumeration of System Information (w)</li> <li>• System Information Discovery (w)</li> </ul>				Remote Access Tools	

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
	<p>Windows Management Instrumentation</p> <ul style="list-style-type: none"> <li>WMI Execution via Microsoft Office Application (w)</li> <li>Remote Execution via WMIC (w)</li> </ul>	Hypervisor		<p>Disabling Security Tools</p> <ul style="list-style-type: none"> <li>Unload Sysmon Filter Driver with fltmc.exe (w)</li> </ul>	<p>Security Memory</p>	<p>System Network Configuration Discovery</p> <ul style="list-style-type: none"> <li>Discovery of Network Environment via Built-in Tools (l, m)</li> <li>Discovery of Network Environment via Built-in Tools (w)</li> </ul>				Remote File Copy	

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
	Windows Remote Management • <i>Incoming Remote Power-Shell Sessions (w)</i>	Kernel Modules and Extensions • <i>Creation of Kernel Module (l)</i> • <i>Loading Kernel Modules with next-load (m)</i>		Execution Guardrails	Two-Factor Authentication Interception	System Network Connections Discovery • <i>Enumeration of Mounted Shares (w)</i> • <i>System Network Connections Discovery (l, m)</i>				Standard Application Layer Protocol • <i>Non-browser processes making DNS requests to Dynamic DNS Providers (w)</i>	

Continued on next page



Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		LC_LOAD_DYLIB Addition		Exploitation for Defense Evasion		System Owner/User Discovery <ul style="list-style-type: none"> <li>System Owner and User Discovery (w)</li> <li>Discovery and Enumeration of System Information via Rundll32 (w)</li> </ul>				Standard Cryptographic Protocol	
		Launch Agent <ul style="list-style-type: none"> <li>Persistent process via Launch Agent (m)</li> </ul>		Extra Window Memory Injection		System Service Discovery				Standard Non-Application Layer Protocol	

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		<ul style="list-style-type: none"> <li>Launch Daemon</li> <li>• <i>Launch Daemon Persistence (m)</i></li> </ul>		File Deletion		<ul style="list-style-type: none"> <li>System Time Discovery</li> <li>• <i>Discovery of a Remote System's Time (w)</i></li> </ul>				Uncommonly Used Port	
		<ul style="list-style-type: none"> <li>Local Job Scheduling</li> <li>• <i>Local Job Scheduling Paths (l, m)</i></li> <li>• <i>Local Job Scheduling Process (l, m)</i></li> </ul>		<ul style="list-style-type: none"> <li>File Permissions Modification</li> <li>• <i>Windows File Permissions Modification (w)</i></li> </ul>					Web Service		
		Login Item		File System Logical Offsets							

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		Modify Existing Service <ul style="list-style-type: none"> <li>Service Path Modification with sc.exe (w)</li> </ul>		Gatekeeper Bypass <ul style="list-style-type: none"> <li>Potential Gatekeeper Bypass (m)</li> </ul>							
		Netsh Helper DLL <ul style="list-style-type: none"> <li>Persistence via NetSh Key (w)</li> </ul>		Group Policy Modification							
		New Service		HISTCONTROL							

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		Office Application Startup <ul style="list-style-type: none"> <li>Office Application Startup via Template File Modification (w)</li> <li>Office Application Startup via Template Registry Modification (w)</li> </ul>		Hidden Files and Directories <ul style="list-style-type: none"> <li>Adding the Hidden File Attribute with <i>attrib.exe</i> (w)</li> </ul>							
		Path Interception		Hidden Users							

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		<ul style="list-style-type: none"> <li>Port Monitors</li> </ul>	<ul style="list-style-type: none"> <li><i>Installation of Port Monitor (w)</i></li> </ul>	Hidden Window							
		<ul style="list-style-type: none"> <li>rc.common</li> </ul>	<ul style="list-style-type: none"> <li><i>Modification of rc.common Script (m)</i></li> </ul>	Indicator Blocking							
		<ul style="list-style-type: none"> <li>Re-opened Applications</li> </ul>	<ul style="list-style-type: none"> <li><i>Resumed Application on Reboot (m)</i></li> </ul>	Indicator Removal from Tools							

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		<ul style="list-style-type: none"> <li>Registry Run Keys / Startup Folder</li> <li>• <i>Startup Folder Execution via VBScript (w)</i></li> <li>• <i>Startup Folder Persistence with Shortcut/VBScript Files (w)</i></li> <li>• <i>Registry Persistence via Run Keys (w)</i></li> <li>• <i>Registry Persistence via Shell Folders (w)</i></li> </ul>		<ul style="list-style-type: none"> <li>Indicator Removal on Host</li> <li>• <i>Delete Volume USN Journal with fsutil (w)</i></li> <li>• <i>Host Artifact Deletion (w)</i></li> <li>• <i>Clearing Windows Event Logs with wevtutil (w)</i></li> </ul>							

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		Screensaver <ul style="list-style-type: none"> <li>• <i>Persistence via Screensaver (w)</i></li> </ul>		Indirect Command Execution <ul style="list-style-type: none"> <li>• <i>Indirect Command Execution (w)</i></li> </ul>							
		Security Support Provider <ul style="list-style-type: none"> <li>• <i>Installation of Security Support Provider (w)</i></li> </ul>		Install Root Certificate <ul style="list-style-type: none"> <li>• <i>Root Certificate Install (w)</i></li> </ul>							
		Service Registry Permissions Weakness		InstallUtil <ul style="list-style-type: none"> <li>• <i>InstallUtil Execution (w)</i></li> </ul>							
		Shortcut Modification		LC_MAIN Hijacking							
		Startup Items		Launchctl							

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		System Firmware		Masquerading <ul style="list-style-type: none"> <li>Processes Running with Unusual Extensions (w)</li> </ul>							
		System Service <ul style="list-style-type: none"> <li>Creation or Modification of Systemd Service (l)</li> </ul>		Modify Registry <ul style="list-style-type: none"> <li>Suspicious MS Office Registry Modifications (w)</li> </ul>							

Continued on next page



Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		<ul style="list-style-type: none"> <li>Time Providers</li> <li>Installation of Time Providers (w)</li> </ul>		Mshta <ul style="list-style-type: none"> <li>Mshta Descendant of Microsoft Office (w)</li> <li>Mshta Network Connections (w)</li> </ul>							
		Web Shell		NTFS File Attributes <ul style="list-style-type: none"> <li>Suspicious ADS File Creation (w)</li> </ul>							

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		Windows Management Instrumentation Event Subscription		Network Share Connection Removal <ul style="list-style-type: none"> <li>• <i>Disconnecting from Network Shares with net.exe (w)</i></li> </ul>							
		Winlogon Helper DLL <ul style="list-style-type: none"> <li>• <i>Registration of Winlogon Helper DLL (w)</i></li> </ul>		Obfuscated Files or Information							
				Plist Modification <ul style="list-style-type: none"> <li>• <i>Plist Modification (m)</i></li> </ul>							
				Port Knocking							

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
				Process Doppelgänger							
				Process Hollowing <ul style="list-style-type: none"> <li>• <i>Unusual Child Process (w)</i></li> </ul>							
				Process Injection <ul style="list-style-type: none"> <li>• <i>Modification of ld.so.preload (l)</i></li> <li>• <i>Unusual Child Process (w)</i></li> </ul>							
				Redundant Access							
				Regsvcs/Regasm							
				Regsvr32 <ul style="list-style-type: none"> <li>• <i>Suspicious Script Object Execution (w)</i></li> </ul>							
				Rootkit							

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
				Rundll32							
				SIP and Trust Provider Hijacking							
				Scripting							
				Signed Binary Proxy Execution							
				Signed Script Proxy Execution <ul style="list-style-type: none"> <li>• <i>Proxied Execution via Signed Scripts (w)</i></li> </ul>							
				Software Packing							
				Space after File-name <ul style="list-style-type: none"> <li>• <i>Processes with Trailing Spaces (l, m)</i></li> </ul>							

Continued on next page

Table 3 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
				Template Injection <ul style="list-style-type: none"> <li>MS Office Template Injection (w)</li> </ul>							
				Timestomp							
				Trusted Developer Utilities							
				Valid Accounts							
				Virtualization/Sandbox Evasion							
				XSL Script Processing							



### 1.4.1 Linux

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
Drive-by Compromise	Command Line Interface	bash_profile and .bashrc <ul style="list-style-type: none"> <li>• <i>Modifications of .bash_profile and .bashrc</i></li> </ul>	Exploitation for Privilege Escalation	Binary Padding	Bash History	Account Discovery <ul style="list-style-type: none"> <li>• <i>Account Discovery via Built-In Tools</i></li> </ul>	Application Deployment Software	Audio Capture	Automated Exfiltration	Commonly Used Port	Data Destruction
Exploit Public-Facing Application	Exploitation for Client Execution	Bootkit	Setuid and Setgid	Clear Command History	Brute Force	Browser Bookmark Discovery	Exploitation of Remote Services	Automated Collection	Data Compressed	Communication Through Removable Media	Data Encrypted for Impact
Hardware Additions	Graphical User Interface	Browser Extensions	Sudo	Compile After Delivery	Credential Dumping	File and Directory Discovery	Remote Services <ul style="list-style-type: none"> <li>• <i>Remote Terminal Sessions</i></li> </ul>	Clipboard Data	Data Encrypted	Connection Proxy	Defacement
Spearphishing Attachment	Single Source	Create Account	Sudo Caching	Disabling Security Tools	Credential in Files <ul style="list-style-type: none"> <li>• <i>Searching for Passwords in Files</i></li> </ul>	Network Service Scanning	SSH Hijacking	Data Staged <ul style="list-style-type: none"> <li>• <i>Creation of an Archive with Common Archivers</i></li> </ul>	Data Transfer Limits	Custom Command and Control Protocol	Disk Content Wipe
Spearphishing Link	Third-party Software	Kernel Modules and Drivers		Execution Guardrails	Exploitation for Credential Access	Password Policy Discovery <ul style="list-style-type: none"> <li>• <i>Password Policy</i></li> </ul>		Data from Information Repositories	Exfiltration Over Alternative Protocol	Custom Cryptographic Protocol	Disk Structure Wipe
<b>1.4. Enterprise ATT&amp;CK Matrix</b>											
											<b>131</b>





### 1.4.2 macOS

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
Drive-by Compromise	AppleScript	bash_profile and .bashrc <ul style="list-style-type: none"> <li>• <i>Modifications of .bash_profile and .bashrc</i></li> </ul>	Exploitation for Privilege Escalation	Binary Padding	Bash History	Account Discovery <ul style="list-style-type: none"> <li>• <i>Account Discovery via Built-In Tools</i></li> </ul>	Application Deployment Software	Audio Capture	Automated Exfiltration	Commonly Used Port	Data Destruction
Exploit Public-Facing Application	Command Line Interface	Browser Extensions <ul style="list-style-type: none"> <li>• <i>Suspicious File Creation via Browser Extensions</i></li> </ul>	Setuid and Setgid	Clear Command History	Brute Force	Application Window Discovery	Exploitation of Remote Services	Automated Collection	Data Compressed	Communication Through Removable Media	Data Encrypted for Impact
Hardware Additions	Exploitation for Client Execution	Create Account	Sudo	Code Signing	Credential Dumping	Browser Bookmark Discovery	Logon Scripts	Clipboard Data <ul style="list-style-type: none"> <li>• <i>Reading the Clipboard with pb-paste</i></li> </ul>	Data Encrypted	Connection Proxy	Defacement
Spearphishing Attachment	Graphical User Interface	Dylib Hijacking	Sudo Caching	Compile After Delivery	Credentials in Files <ul style="list-style-type: none"> <li>• <i>Searching for Passwords in Files</i></li> </ul>	File and Directory Discovery	Remote Services <ul style="list-style-type: none"> <li>• <i>Remote Terminal Sessions</i></li> </ul>	Data Staged <ul style="list-style-type: none"> <li>• <i>Creation of an Archive with Common Archivers</i></li> </ul>	Data Transfer <ul style="list-style-type: none"> <li>• <i>Limits of an Archive with Common Archivers</i></li> </ul>	Custom Command and Control Protocol	Disk Content Wipe
<b>1.4. Enterprise ATT&amp;CK Matrix</b>											
Spearphishing Link	Kernel Mod-			Disabling Security	Exploitation for	Network Services	SSH Hijacking	Data from	Exfiltration Over	Custom Crypt-	Disk Struc-

### 1.4.3 Windows

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
Drive-by Compromise	Command Line Interface	Accessibility Features <ul style="list-style-type: none"> <li>• <i>ImageScalability Features</i></li> </ul>	Exploitation for Privilege Escalation	Access Token Manipulation	Account Manipulation	Account Discovery <ul style="list-style-type: none"> <li>• <i>Account Discovery via Built-In Tools</i></li> <li>• <i>Discovery and Enumeration of System Information via Rundll32</i></li> </ul>	Application Deployment Software	Audio Capture <ul style="list-style-type: none"> <li>• <i>Audio Capture via PowerShell</i></li> <li>• <i>Audio Capture via SoundRecorder</i></li> </ul>	Automated Exfiltration	Commonly Used Port	Data Destruction

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
Exploit Public-Facing Application	Dynamic Data Exchange	App DLLs	Certificate File Execution	BITS Jobs	Brute Force	Application Window Discovery	Distributed Component Object Model	Automated Collection	Data Compressed	Communication Through Remote Media	Data Encrypted for Impact
	• Executable Writ-ten and Ex-ecuted by Mi-crosoft Of-fice Ap-pli-ca-tions	• Application DLLs	Regions Injec-tion	Image De-bug-gers for Ac-ces-si-bil-ity Fea-tures	• Suspicious Bit-sad-min Job via bit-sad-min.exe				• Com-bined Lin-ear Cre-ation of a RAR file		

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
Hardware Additions	Execution through API	AppInit DLLs <ul style="list-style-type: none"> <li>• <i>Persistence via AppInit DLL</i></li> </ul>	SID-History Injection	Binary Padding	Credential Dumping <ul style="list-style-type: none"> <li>• <i>Suspicious Process Loading Credential Vault DLL</i></li> <li>• <i>Credential Enumeration via Credential Vault CLI</i></li> <li>• <i>LSASS Memory Dumping</i></li> <li>• <i>AD Dumping via Ntdsutil.exe</i></li> <li>• <i>LSASS Memory Dumping via ProcDump.exe</i></li> <li>• <i>SAM Dumping via Reg.exe</i></li> </ul>	Browser Bookmarks <ul style="list-style-type: none"> <li>• <i>Suspicious Proxy</i></li> </ul>	Exploitation of Remote Services	Clipboard Data	Data Encrypted	Connection Proxy	Defacement
136									Chapter 1. Next Steps		

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
Spearphishing Attachment	Execution through Module Load	Application Shim- ming • <i>Installing Custom Shim Databases</i>		Bypass User Account Control • <i>Bypass UAC via CM-STP</i>	Credentials in Files • <i>Searching for Passwords in Files</i>	Domain Trust Discovery • <i>Domain Trust Discovery</i>	Logon Scripts • <i>Modification of Logon Scripts from Registry</i>	Data Staged • <i>WMI Size Ex-Limits</i>	Data Transfer <i>with Command Line Redirection</i>	Custom Command and Control Protocol	Disk Content Wipe
Spearphishing Link	Exploitation for Client Execution	Authentication Package • <i>LSA Authentication Package</i>		CMSTP • <i>Execution via cm-stp.exe</i> • <i>Bypass UAC via CM-STP</i>	Credentials in Registry <i>Registry</i>	File and Directory Discovery	Pass the Hash	Data from Information Repositories	Exfiltration Over Alternative Protocol	Custom Cryptographic Protocol	Disk Structure Wipe
Spearphishing via Service	Impersonation User Interface	Bootkit		Code Signing	Exploitation for Credential Access	Network Service Scanning • <i>Network Service Scanning via Port</i>	Pass the Ticket	Data from Local System	Exfiltration Over Command and Control Channel	Data Encoding	Endpoint Denial of Service

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
Supply Chain Compromise	LSASS Driver	Browser Extensions • <i>Suspicious File Creation via Browser Extensions</i>		Compile After Delivery	Forced Authentication	Network Share Discovery • <i>Enumeration of Local Shares</i> • <i>Enumeration of Remote Shares</i>	Remote Desktop Protocol • <i>Remote Desktop Protocol in jack</i>	Data from Network Shared Drive	Exfiltration Over Other Network Medium	Data Obfuscation	Firmware Corruption

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
Trusted Relationship	PowerShell	Change Default File Association		Compiled HTML File	Input Prompt	Password Policy Discovery	Remote Services	Data from Removal	Exfiltration Over Physical Medium	Domain Fronting	Inhibit System Recovery
		<ul style="list-style-type: none"> <li>Change Default File Association</li> </ul>		<ul style="list-style-type: none"> <li>HH.exe execution</li> </ul>			<ul style="list-style-type: none"> <li>Remote Terminate Sessions</li> </ul>				<ul style="list-style-type: none"> <li>Modification of Boot Configuration</li> <li>Volume Shadow Copy Deletion via VssAdmin</li> <li>Volume Shadow Copy Deletion via WMIC</li> </ul>
	Scheduled Task	Create Account		Component Firmware	Kerberos	Peripheral Device Discovery	Replication Through Removable Media	Email Collection	Scheduled Transfer	Domain Generation Algorithms	Network Denial of Service
	<ul style="list-style-type: none"> <li>Creation of Scheduled Task with schtasks.exe</li> </ul>	<ul style="list-style-type: none"> <li>User Account Creation</li> </ul>					<ul style="list-style-type: none"> <li>Access of Outlook Email Archives</li> </ul>				

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
	Service Execution <ul style="list-style-type: none"> <li>• <i>Execution of Existing Service via Command</i></li> </ul>	DLL Search Order Hijacking <ul style="list-style-type: none"> <li>• <i>DLL Search Order Hijacking with known programs</i></li> </ul>		Component Object Model Hijacking <ul style="list-style-type: none"> <li>• <i>COM Hijack via Script Object</i></li> </ul>	Local MNR/NTLMS Poisoning and Relay	RBAC Groups Discovery	Shared Web-root	Input Capture		Fallback Channels	Resource Hijacking
	Third-party Software	External Remote Services		Control Panel Items <ul style="list-style-type: none"> <li>• <i>Control Panel Items</i></li> </ul>	Network Sniffing	Process Discovery <ul style="list-style-type: none"> <li>• <i>Process Discovery via Windows Tools</i></li> </ul>	Taint Shared Content	Man in the Browser		Multi-Stage Channels	Runtime Data Manipulation

Continued on next page



Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
	User Execution <ul style="list-style-type: none"> <li>• Executable Writings Weakness Executed by Microsoft Office Applications</li> </ul>	File System Permissions		DCShadow	Password Filter DLL <ul style="list-style-type: none"> <li>• Registration of a Password Filter DLL</li> </ul>	Query Registry	Windows Admin Shares <ul style="list-style-type: none"> <li>• Mounting Hidden Shares</li> <li>• Mounting Windows Hidden Shares with net.exe</li> </ul>	Screen Capture		Multi-hop Proxy	Service Stop <ul style="list-style-type: none"> <li>• Service Stop or Disable with sc.exe</li> <li>• Stopping Services with net.exe</li> </ul>

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
	Windows Management Instrumentation <ul style="list-style-type: none"> <li>• <i>WMI Execution via Microsoft Office Application</i></li> <li>• <i>Remote Execution via WMIC</i></li> </ul>	Hooking		DLL Side-Loading	Private Keys	Remote System Discovery <ul style="list-style-type: none"> <li>• <i>Windows Network Enumeration</i></li> <li>• <i>Remote System Discovery Commands</i></li> </ul>		Video Capture		Multiband Communication	Stored Data Manipulation

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
	Windows Remote Management	Hypervisor		Deobfuscation Files or Information	Two-Factor Authentication Interception	Security Software Discovery				Multilayer Encryption	Transmitted Data Manipulation
	<ul style="list-style-type: none"> <li><i>Incoming Remote Power-Shell Sessions</i></li> </ul>			<ul style="list-style-type: none"> <li><i>Encoding or Decoding Files via CertUtil</i></li> </ul>		<ul style="list-style-type: none"> <li><i>Process Discovery via Windows Tools</i></li> </ul>					
		<ul style="list-style-type: none"> <li>Modify Existing Service</li> </ul>		<ul style="list-style-type: none"> <li>Disabling Security Tools</li> </ul>		<ul style="list-style-type: none"> <li>System Information Discovery</li> </ul>				Remote Access Tools	
		<ul style="list-style-type: none"> <li><i>Service Path Modification with sc.exe</i></li> </ul>		<ul style="list-style-type: none"> <li><i>Unload Sysmon Filter Driver with fltmc.exe</i></li> </ul>		<ul style="list-style-type: none"> <li><i>Enumeration of System Information</i></li> <li><i>System Information Discovery</i></li> </ul>					

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		Netsh Helper DLL <ul style="list-style-type: none"> <li>• <i>Persistence via NetSh Key</i></li> </ul>		Execution Guardrails		System Network Configuration Discovery <ul style="list-style-type: none"> <li>• <i>Discovery of Network Environment via Built-in Tools</i></li> </ul>				Remote File Copy	
		New Service		Exploitation for Defense Evasion		System Network Connections Discovery <ul style="list-style-type: none"> <li>• <i>Enumeration of Mounted Shares</i></li> </ul>				Standard Application Layer Protocol <ul style="list-style-type: none"> <li>• <i>Non-browser processes making DNS requests to Dynamic DNS Providers</i></li> </ul>	

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		Office Application Startup <ul style="list-style-type: none"> <li>Office Application Startup via Template File Modification</li> <li>Office Application Startup via Template Registry Modification</li> </ul>		Extra Window Memory Injection		System Owner/User Discovery <ul style="list-style-type: none"> <li>System Owner and User Discovery</li> <li>Discovery and Enumeration of System Information via Rundll32</li> </ul>				Standard Cryptographic Protocol	
		Path Interception		File Deletion		System Service Discovery				Standard Non-Application Layer Protocol	

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		Port Monitors <ul style="list-style-type: none"> <li>• <i>Installation of Port Monitor</i></li> </ul>		File Permissions Modification <ul style="list-style-type: none"> <li>• <i>Windows File Permissions Modification</i></li> </ul>		System Time Discovery <ul style="list-style-type: none"> <li>• <i>Discovery of a Remote System's Time</i></li> </ul>				Uncommonly Used Port	

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		Registry Run Keys / Startup Folder <ul style="list-style-type: none"> <li>• Startup Folder Execution via VBScript</li> <li>• Startup Folder Persistence with Shortcut/VBScript Files</li> <li>• Registry Persistence via Run Keys</li> <li>• Registry Persistence via Shell Folders</li> </ul>		File System Logical Offsets						Web Service	

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		<ul style="list-style-type: none"> <li>Screensaver</li> </ul>	<ul style="list-style-type: none"> <li><i>Persistence via Screensaver</i></li> </ul>	<ul style="list-style-type: none"> <li>Group Policy Modification</li> </ul>							
		<ul style="list-style-type: none"> <li>Security Support Provider</li> </ul>	<ul style="list-style-type: none"> <li><i>Installation of Security Support Provider</i></li> </ul>	<ul style="list-style-type: none"> <li>Hidden Files and Directories</li> </ul>	<ul style="list-style-type: none"> <li><i>Adding the Hidden File Attribute with via attrib.exe</i></li> </ul>						
		<ul style="list-style-type: none"> <li>Service Registry Permissions Weakness</li> </ul>		<ul style="list-style-type: none"> <li>Indicator Blocking</li> </ul>							
		<ul style="list-style-type: none"> <li>Shortcut Modification</li> </ul>		<ul style="list-style-type: none"> <li>Indicator Removal from Tools</li> </ul>							

Continued on next page



Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		System Firmware		Indicator Removal on Host <ul style="list-style-type: none"> <li>• <i>Delete Volume USN Journal with fsutil</i></li> <li>• <i>Host Artifact Deletion</i></li> <li>• <i>Clearing Windows Event Logs with wevtutil</i></li> </ul>							
		Time Providers <ul style="list-style-type: none"> <li>• <i>Installation of Time Providers</i></li> </ul>		Indirect Command Execution <ul style="list-style-type: none"> <li>• <i>Indirect Command Execution</i></li> </ul>							

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
		Web Shell		Install Root Certificate <ul style="list-style-type: none"> <li>• <i>Root Certificate Install</i></li> </ul>							
		Windows Management Instrumentation Event Subscription		InstallUtil <ul style="list-style-type: none"> <li>• <i>InstallUtil Execution</i></li> </ul>							
		Winlogon Helper DLL <ul style="list-style-type: none"> <li>• <i>Registration of Winlogon Helper DLL</i></li> </ul>		Masquerading <ul style="list-style-type: none"> <li>• <i>Processes Running with Unusual Extensions</i></li> </ul>							

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
				Modify Registry <ul style="list-style-type: none"> <li>• <i>Suspicious MS Office Registry Modifications</i></li> </ul>							
				Mshta <ul style="list-style-type: none"> <li>• <i>Mshta Descendant of Microsoft Office</i></li> <li>• <i>Mshta Network Connections</i></li> </ul>							
				NTFS File Attributes <ul style="list-style-type: none"> <li>• <i>Suspicious ADS File Creation</i></li> </ul>							

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
				Network Share Connection Removal <ul style="list-style-type: none"> <li><i>Disconnecting from Network Shares with net.exe</i></li> </ul>							
				Obfuscated Files or Information							
				Process Doppelgänger-ing							
				Process Hollowing <ul style="list-style-type: none"> <li><i>Unusual Child Process</i></li> </ul>							
				Process Injection <ul style="list-style-type: none"> <li><i>Unusual Child Process</i></li> </ul>							

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
				Redundant Access							
				Regsvcs/Regasm							
				Regsvr32							
				<ul style="list-style-type: none"> <li>Suspicious Script Object Execution</li> </ul>							
				Rootkit							
				Rundll32							
				SIP and Trust Provider Hijacking							
				Scripting							
				Signed Binary Proxy Execution							
				Signed Script Proxy Execution <ul style="list-style-type: none"> <li>Proxied Execution via Signed Scripts</li> </ul>							

Continued on next page

Table 4 – continued from previous page

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Exfiltration	Command and Control	Impact
				Software Packing							
				Template Injection <ul style="list-style-type: none"> <li>MS Office Template Injection</li> </ul>							
				Timestamp							
				Trusted Developer Utilities							
				Valid Accounts							
				Virtualization/Sandbox Evasion							
				XSL Script Processing							

## 1.5 Schemas

### 1.5.1 Microsoft Sysmon

This is the mapping from Microsoft Sysmon native fields to the *security schema*.

#### Timestamp

**field** UtcTime

**format** %Y-%m-%d %H:%M:%S.%f

## Globally provided mapping

```

hostname split(ComputerName, ".", 0)
pid number(ProcessId)
process_name baseName(Image)
process_path Image
unique_pid ProcessGuid
user User
user_domain split(User, "\\", 0)
user_name split(User, "\\", 1)

```

## Event specific mappings

### file

```
EventId in (11, 15)
```

#### fields

```

file_name baseName(TargetFilename)
file_path TargetFilename

```

### image\_load

```
EventId == 7
```

#### fields

```

image_name baseName(ImageLoaded)
image_path ImageLoaded

```

### network

```
EventId == 3
```

#### subtype mapping

```

incoming Initiated == 'false'
outgoing Initiated == 'true'

```

#### fields

```

destination_address DestinationIp
destination_port DestinationPort
protocol Protocol
source_address SourceIp
source_port SourcePort

```

### process

EventId in (1, 5)

#### subtype mapping

```
create EventId == 1
terminate EventId == 5
```

#### fields

```
command_line CommandLine
logon_id number(LogonId)
original_file_name OriginalFileName
parent_process_name baseName(ParentImage)
parent_process_path ParentImage
ppid number(ParentProcessId)
unique_ppid ParentProcessGuid
```

### registry

EventId in (12, 13, 14)

#### hive mapping

```
hklm TargetObject == "HKLM\\*"
hku TargetObject == "HKU\\*"
```

#### fields

```
registry_key dirName(TargetObject)
registry_path TargetObject
registry_value baseName(TargetObject)
```

## 1.5.2 MITRE Cyber Analytics Repository

This is the mapping from MITRE Cyber Analytics Repository native fields to the *security schema*.

### Timestamp

```
field @timestamp
format %Y-%m-%dT%H:%M:%S.%fZ
```

### Globally provided mapping

```
hostname hostname
pid pid
process_name exe
```



```
process_path image_path
unique_pid process_guid
user user
user_domain split(user, "\\", 0)
user_name split(user, "\\", 1)
```

## Event specific mappings

### file

```
data_model.object = 'file'
```

#### subtype mapping

```
create arrayContains(data_model.actions, "create")
delete arrayContains(data_model.actions, "delete")
modify arrayContains(data_model.actions, "modify")
```

#### fields

```
file_name file_name
file_path file_path
```

### network

```
data_model.object == 'flow'
```

#### subtype mapping

```
incoming not initiated
outgoing initiated
```

#### fields

```
destination_address dest_ip
destination_port dest_port
protocol transport
source_address src_ip
source_port src_port
```

### process

```
data_model.object = 'process'
```

#### subtype mapping

```
create arrayContains(data_model.action, 'create')
terminate arrayContains(data_model.action, 'terminate')
```

#### fields

```
command_line command_line
parent_process_name parent_exe
parent_process_path parent_image_path
ppid ppid
unique_ppid parent_process_guid
```

### registry

```
data_model.object == "registry" and not arrayContains(data_model.actions,
"remove")
```

#### registry\_type mapping

```
binary type == "REG_BINARY"
dword type = "REG_DWORD"
expand_string type = "REG_EXPAND_SZ"
multi_string type = "REG_MULTI_SZ"
qword type = "REG_QWORD"
string type = "REG_SZ"
```

#### hive mapping

```
hklm hive == "HKEY_LOCAL_MACHINE"
hku hive == "HKEY_USERS"
```

#### fields

```
registry_data data
registry_key key
registry_path key
registry_value value
```

## 1.5.3 Security Events

This is the primary schema used for normalizing across data sources. Queries are written to match this schema, and data sources are converted to this schema. This unifies sources to a unified by a common language and a common data model, so analytics can be written generically and are easy shareable.

#### Globally provided fields

- hostname
- pid
- process\_name
- process\_path
- unique\_pid
- user
- user\_domain

- user\_name
- user\_sid

## **file**

### **subtype** options

- create
- modify
- delete

### **fields**

- file\_name
- file\_path

## **image\_load**

### **fields**

- image\_name
- image\_path

## **network**

### **subtype** options

- incoming
- outgoing
- disconnect

### **fields**

- destination\_address
- destination\_port
- protocol
- source\_address
- source\_port
- total\_in\_bytes
- total\_out\_bytes

## **process**

### **subtype** options

- create
- terminate

### **fields**

- `command_line`
- `logon_id`
- `original_file_name`
- `parent_process_name`
- `parent_process_path`
- `ppid`
- `unique_ppid`

### registry

#### hive options

- `hku`
- `hklm`

#### registry\_type options

- `dword`
- `qword`
- `string`
- `expand_string`
- `multi_string`
- `binary`

#### fields

- `registry_data`
- `registry_key`
- `registry_path`
- `registry_value`

## 1.6 Resources

### 1.6.1 Blogs

- [EQL Threat Hunting](#)
- [Ransomware, interrupted: Sodinokibi and the supply chain](#)
- [Detecting Adversary Tradecraft with Image Load Event Logging and EQL](#)
- [EQL's Highway to Shell](#)
- [Getting Started with EQL](#)
- [EQL For the Masses](#)
- [Introducing EQL](#)

## 1.6.2 Presentations

- BSides DFW 2019: ATT&CKing Koadic with EQL (slides)
- BlackHat 2019: Fantastic Red-Team Attacks and How to Find Them (slides, blog)
- BSides SATX 2019: The Hunter Games: How to Find the Adversary with EQL (slides)
- Circle City Con 2019: The Hunter Games: How to Find the Adversary with EQL (slides)
- Atomic Friday: Endgame on EQL (slides, notebook)
- MITRE ATT&CKcon: From Technique to Detection

## 1.6.3 Additional Resources

- Atomic Red Team
- Microsoft Sysmon
- MITRE ATT&CK™
- Event Query Language (docs, code, twitter)
- EQL Analytics Library (docs, code)

## 1.7 License

MIT License

Copyright (c) 2018 Endgame, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

---

**Note:** The [Event Query Language](#) has an [AGPL License](#)

---



## Symbols

-file, -f

- convert-data command line option, 4
- query command line option, 5
- survey command line option, 6

-format

- convert-data command line option, 4
- query command line option, 5
- survey command line option, 6

-c

- survey command line option, 6

-e <encoding>

- convert-data command line option, 4
- query command line option, 5
- survey command line option, 6

-h

- convert-data command line option, 4
- convert-query command line option, 5
- query command line option, 5
- survey command line option, 6

-s <data-source>, -source <data-source>

- convert-data command line option, 4
- convert-query command line option, 5
- query command line option, 5
- survey command line option, 6

## A

analytic-path [analytic-path, ...]

- survey command line option, 5

## C

convert-data command line option

- file, -f, 4
- format, 4
- e <encoding>, 4
- h, 4
- s <data-source>, -source <data-source>, 4
- output-json-file, 4

convert-query command line option

-h, 5

- s <data-source>, -source <data-source>, 5
- eql-query, 4

## E

eql-query

- convert-query command line option, 4

## I

input-query

- query command line option, 5

## O

output-json-file

- convert-data command line option, 4

## Q

query command line option

- file, -f, 5
- format, 5
- e <encoding>, 5
- h, 5
- s <data-source>, -source <data-source>, 5
- input-query, 5

## S

survey command line option

- file, -f, 6
- format, 6
- c, 6
- e <encoding>, 6
- h, 6
- s <data-source>, -source <data-source>, 6
- analytic-path [analytic-path, ...], 5