Threat Hunter Playbook

Search this book...

KNOWLEDGE LIBRARY

Windows

PRE-HUNT ACTIVITIES

Data Management

GUIDED HUNTS

Windows

LSASS Memory Read Access

DLL Process Injection via CreateRemoteThread and LoadLibrary

Active Directory Object Access via Replication Services

Active Directory Root Domain Modification for Replication Services

Registry Modification to Enable Remote Desktop Conections

Local PowerShell Execution

WDigest Downgrade

PowerShell Remote Session

Alternate PowerShell Hosts

Domain DPAPI Backup Key Extraction

SysKey Registry Keys Access

SAM Registry Hive Handle Request

WMI Win32_Process Class and Create Method for Remote Execution

WMI Eventing

WMI Module Load

Local Service Installation

Remote Service creation

Remote Service Control Manager Handle

Remote Interactive Task Manager LSASS Dump

Registry Modification for Extended NetNTLM Downgrade

Access to Microphone Device

Remote WMI ActiveScriptEventConsumers

Remote DCOM IErtUtil DLL Hijack

Remote WMI Wbemcomn DLL Hijack

SMB Create Remote File

WuauClt CreateRemoteThread Execution

TUTORIALS

Jupyter Notebooks

SAM Registry Hive Handle Request

Hypothesis

Adversaries might be getting a handle to the SAM database to extract credentials in my environment

Technical Context

Every computer that runs Windows has its own local domain; that is, it has an account database for accounts that are specific to that computer. Conceptually,this is an account database like any other with accounts, groups, SIDs, and so on. These are referred to as local accounts, local groups, and so on. Because computers typically do not trust each other for account information, these identities stay local to the computer on which they were created.

Offensive Tradecraft

Adversaries might use tools like Mimikatz with lsadump::sam commands or scripts such as Invoke-PowerDump to get the SysKey to decrypt Security Account Mannager (SAM) database entries (from registry or hive) and get NTLM, and sometimes LM hashes of local accounts passwords.

In addition, adversaries can use the built-in Reg.exe utility to dump the SAM hive in order to crack it offline.

Additional reading

- https://github.com/OTRF/ThreatHunter-Playbook/tree/master/docs/library/windows/security_account_manager_database.md
- https://github.com/OTRF/ThreatHunter-Playbook/tree/master/docs/library/windows/syskey.md

Pre-Recorded Security Datasets

Metadata	Value
docs	https://securitydatasets.com/notebooks/atomic/windows/credential_access/SDWIN-190625103712.html
link	https://raw.githubusercontent.com/OTRF/Security-Datasets/master/datasets/atomic/windows/credential_access/host/empire_mimikatz_sam_access.zip

Download Dataset

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

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

Read Dataset

```
import pandas as pd
from pandas.io import json

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

Analytics

A few initial ideas to explore your data and validate your detection logic:

Analytic I

Monitor for any handle requested for the SAM registry hive.

Data source	Event Provider	Relationship	Event
Windows registry	Microsoft-Windows-Security-Auditing	Process requested access Windows registry key	4656
Windows registry	Microsoft-Windows-Security-Auditing	User requested access Windows registry key	4656

Logic

```
SELECT `@timestamp`, Hostname, SubjectUserName, ProcessName, ObjectName, AccessMa
FROM dataTable
WHERE LOWER(Channel) = "security"
    AND EventID = 4656
    AND ObjectType = "Key"
    AND lower(ObjectName) LIKE "%sam"
```

Pandas Query

```
(
df[['@timestamp','Hostname','SubjectUserName','ProcessName','ObjectName','AccessM

[(df['Channel'].str.lower() == 'security')
    & (df['EventID'] == 4656)
    & (df['ObjectType'] == 'Key')
    & (df['ObjectName'].str.lower().str.endswith('sam', na=False))
]
.head()
)
```

Known Bypasses

False Positives

Hunter Notes

Hunt Output

Type	Link
Sigma Rule	https://github.com/SigmaHQ/sigma/blob/master/rules/windows/builtin/security/win_sam_registry_hive_handle_request.yml
Sigma Rule	https://github.com/SigmaHQ/sigma/blob/master/rules/windows/process_creation/win_grabbing_sensitive_hives_via_reg.yml

References

- http://www.harmj0y.net/blog/activedirectory/remote-hash-extraction-on-demand-via-host-security-descriptor-modification/
- https://github.com/gentilkiwi/mimikatz/wiki/module-~-lsadump
- https://adsecurity.org/?page_id=1821#LSADUMPSAM

Previous
SysKey Registry Keys Access

Next
WMI Win32_Process Class and Create Method for Remote Execution

By Roberto Rodriguez @Cyb3rWard0g