Conference

SANS Ransomware Summit 2022, Can You Detect This?

June 16, 2022

This report is a companion to the SANS Ransomware Summit 2022 "Can You Detect This" presentation today 6/16/22 @ 14:40 UTC (10:40 AM ET).

Slides: SANS Ransomware Summit 2022 – Can You Detect This
Recording: {should be available within 48 hours}

The 2021 Year In Review report provided insights into common MITRE ATT&CK techniques observed across our cases, and some opportunities for detection. In this report we will review a collection of actionable detections based on threat actor behavior in intrusions we have investigated over the past year.

Shout out to @_pete_0 and @yatinwad for presenting and writing this companion report.

As with all forms of detection content, its important to understand the nature of the detection and the target environment. This includes data sources, related configuration, and tuning for normal

behaviors in your environment. It is recommended any detection is tested prior to deployment on a production system.

Detection Methodology

The DFIR Report analysts use several approaches for developing detection content, our repo is https://github.com/The-DFIR-Report. This could be in Sigma for log source data, YARA for host based artifacts, or Snort/Suricata for network. Sigma provides a solution agnostic approach for expressing detection logic that can be translated into a taxonomy understood by SIEMs, data definition applications etc.

To validate our detection's, we either use the original case dataset, or develop representative data in our labs. We then replay this dataset into the likes of something like F-Secure's Chainsaw – if using Windows event source data. Chainsaw also supports Sigma, and can be customized via mappings. More details on Chainsaw here: https://github.com/countercept/chainsaw

We have a customized mapping for Chainsaw that supports Sysmon Event ID 22. The mapping file is available on our GitHub. You can find our detections repo here.

Intrusion Detections aligned to the MITRE ATT&CK framework

Initial Access

In the past year or so, we have observed malware variants such as IcedId and Trickbot being embedded into macro-based Office documents and delivered as email attachments.

In majority of those intrusions, the Office (Excel or Word) document spawns suspicious processes such as Living-of-the-land binaries (regsvr32.exe, rundll32.exe) and Windows Command Line shell (cmd.exe, powershell.exe).

Example:

Process Chain – XLSB file weaponized with Trickbot

The below Sigma rules can be used for creating detection rules:

Office Applications Spawning LOL Bins:

- https://github.com/SigmaHQ/sigma/blob/master/rules/windows/process_creation/proc_creation_win_lolbins_by_office_applications.yml

Office Applications Spawning WMI Command Line:

- https://github.com/SigmaHQ/sigma/blob/master/rules/windows/process_creation/proc_creation_win_office_applications_spawning_wmi_commandline.yml

Excel Proxy Executing Regsvr32 with Payload:

- https://github.com/SigmaHQ/sigma/blob/master/rules/windows/process_creation/proc_creation_win_office_from_proxy_executing_regsvr32_payload.yml
- https://github.com/SigmaHQ/sigma/blob/master/rules/windows/process_creation/proc_creation_win_office_from_proxy_executing_regsvr32_payload2.yml

Office Applications Spawning Windows Shell:

- https://github.com/SigmaHQ/sigma/blob/master/rules/windows/process_creation/proc_creation_win_office_shell.yml

Office Applications MS Office Product Spawning Exe in User Dir:

- https://github.com/SigmaHQ/sigma/blob/master/rules/windows/process_creation/proc_creation_win_office_spawn_exe_from_users_directory.yml

Recently, there has been an increase in the use of “ISO” image containers to deliver the initial payload to evade Mark-of-the-Web (MOTW).

- https://thedfirreport.com/2022/04/04/stolen-images-campaign-ends-in-conti-ransomware/
- https://thedfirreport.com/2022/04/25/quantum-ransomware/

The DFIR Report @TheDFIRReport · May 24

10 64 296

The DFIR Report
@TheDFIRReport

Max_Malyutin @Max_Mal_ · May 25

#Emotet LNK Infection
LNK > CMD > PowerShell > Regsvr32

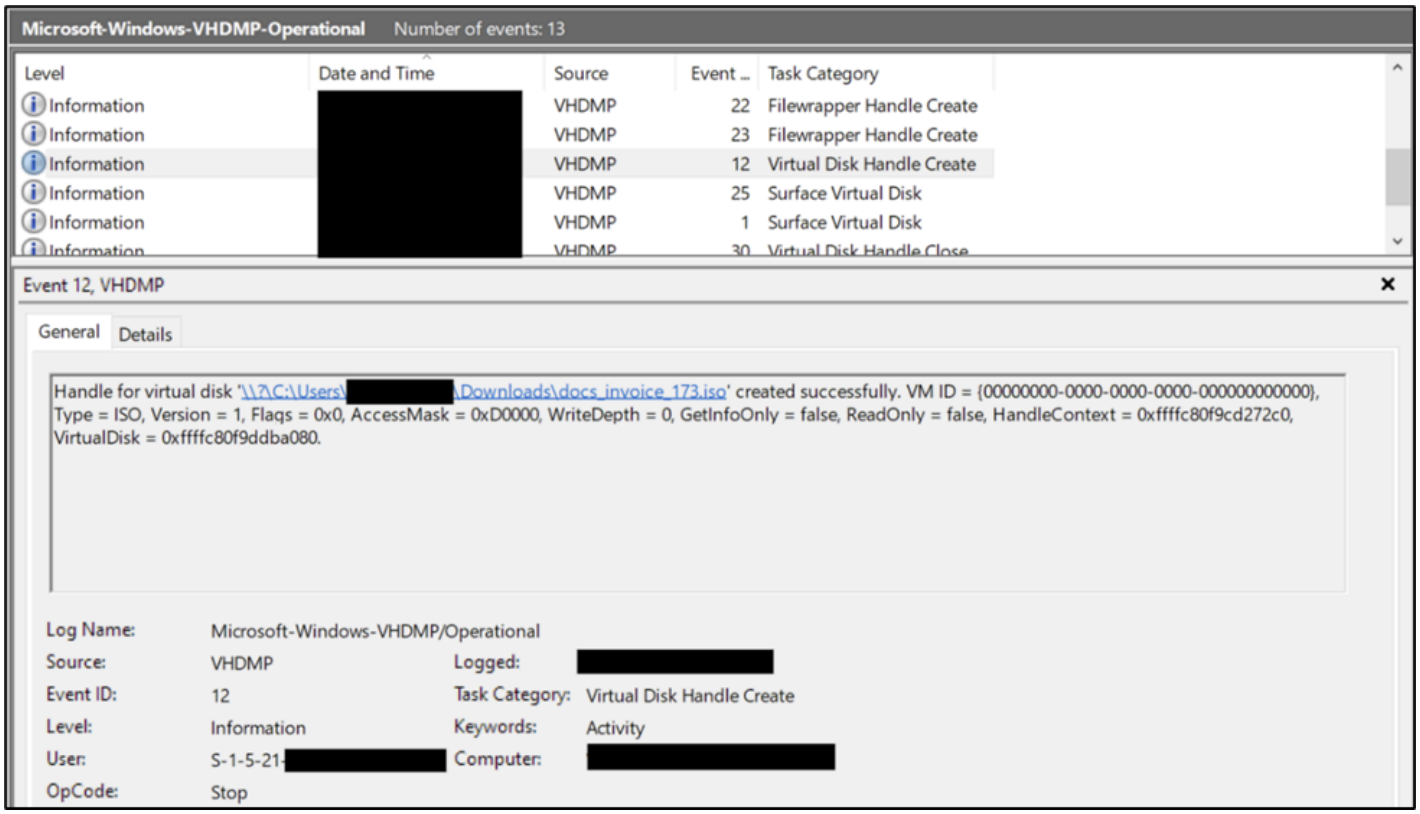
[+] Carrots; p^owershell.^e^xe 🥕
[+] Base64 split; $[Base64]=$[Base64_1]+$[Base64_2]
[+] X6 Distro URLs

Simple PS decode tip; Remove the IEX and execute (Stay safe and do it in VM!)
[screenshot 4]😉

Show this thread

For detecting mounting of ISO images, we can look into “Microsoft-Windows-VHDMP/Operational” log source.

Event IDs: 1 and 12



Sigma Rules:

ISO Image Mount:

- https://github.com/SigmaHQ/sigma/blob/04f72b9e78f196544f8f1331b4d9158df34d7ecf/rules/windows/builtin/security/win_iso_mount.yml

ISO or Image Mount Indicator in Recent Files:

- https://github.com/SigmaHQ/sigma/blob/d459483ef6bb889fb8da1baa17a713a4f1aa8897/rules/windows/file_event/file_event_win_iso_file_recent.yml

Rundll32 From Abnormal Drive

- https://github.com/SigmaHQ/sigma/blob/8bb3379b6807610d61d29db1d76f5af4840b8208/rules/windows/process_creation/proc_creation_win_rundll32_not_from_c_drive.yml

Persistence:

After establishing the initial foothold, threat actors deploy multiple techniques to maintain access to the victim environment. As highlighted in "2021 Year in Review", scheduled tasks were deployed by threat actors in more than 50% of the intrusions.

Scheduled Tasks:

Suspicious Scheduled Task Creation to execute LOLbins

- https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/suspicious_scheduled_task_creation_to_execute_lolbins.yaml

Rare Scheduled Task Creations

- https://github.com/SigmaHQ/sigma/blob/04f72b9e78f196544f8f1331b4d9158df34d7ecf/rules/windows/other/taskscheduler/win_rare_schtask_creation.yml

BITS Job:

In our "Diavol Ransomware" case, a new BITS job was created to execute the malicious DLL from mounted ISO image every 3 hours.

Web Shells*:*

In the “Exchange Exploit Leads to Domain Wide Ransomware” intrusion, multiple web shells were dropped on the infected Microsoft Exchange system. These shells were later used for performing discovery activity.

Sigma Rules:

Webshell Detection With Command Line Keywords

- https://github.com/SigmaHQ/sigma/blob/master/rules/windows/process_creation/proc_creation_win_webshell_detection.yml

Shells Spawned by Web Servers

- https://github.com/SigmaHQ/sigma/blob/master/rules/windows/process_creation/proc_creation_win_webshell_spawn.yml

Exchange Webshell creation

- https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/exchange_webshell_creation

Remote Access Tools:

A recent trend by the threat actor, once a foothold has been established, is to maintain long term persistence using third party remote services, such as those provided via Splashtop, AnyDesk,

NetSupport, etc.

From a recent case (illustration below), Splashtop was deployed, and provided the operator with access to the network using legitimate services. As this is a remote service, there will be a mix of process and network activity that should be very visible.

We've created two Sigma rules for Splashtop covering both process creation and network activity. Splashtop is a legitimate application, however it should be investigated if its not part of your software baseline.

Both process and network events will be detected:

Similarly, the AnyDesk Sigma rule also covers network activities.

Our SplashTop and AnyDesk Sigma rules can be found within our GitHub repo at:

AnyDesk Network:

- https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/win_network_anydesk.yml

SplashTop Network

- https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/win_network_splashtop.yml

SplashTop Process:

- https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/win_software_splashtop.yml

Privilege Escalation:

The threat actors want to elevate their privileges to follow through on their objectives. One technique used is Cobalt Strike's "GetSystem" functionality to achieve SYSTEM privileges.

There is a great article from Red Canary which talks about this functionality present in various offensive tools and hunting tips.

Credential Harvesting:

After escalation of privileges, threat actors target the LSASS process. The primary methods they deploy are:

Accessing the LSASS process using CS or initial malware:

Threat actors also commonly dump the LSASS process with the help of utilities such as Procdump and Task Manager.

Sigma Rules:

Suspicious Use of Procdump on LSASS

- https://github.com/SigmaHQ/sigma/blob/eebd0439e8f2373784c06b4b5fa4670171232c87/rules/windows/process_creation/proc_creation_win_susp_procdump_lsass.yml

LSASS credentials dump via Task Manager (file)

- https://github.com/mdecrevoisier/SIGMA-detection-rules/blob/3009577767f9eae4e9f63736e387b916c3819341/windows-os/win-os-LSASS%20credentials%20dump%20via%20Task%20Manager%20(file).yaml

LSASS Memory Dump

- https://github.com/NVISOsecurity/sigma-public/blob/master/rules/windows/sysmon/sysmon_lsass_memdump.yml

LSASS Memory Dump

- https://bradleyjkemp.dev/sigmadoc/rules/windows/process_access/proc_access_win_lsass_memdump.yml/

LSASS Process Memory Dump Files

- https://github.com/SigmaHQ/sigma/blob/master/rules/windows/file_event/file_event_win_lsass_dump.yml

Invocation of Ntdsutil:

Invocation of Active Directory Diagnostic Tool (ntdsutil.exe)

- https://github.com/NVISOsecurity/sigma-public/blob/master/rules/windows/process_creation/win_susp_ntdsutil.yml

Dumping of Registry Hives:

Grabbing Sensitive Hives via Reg Utility:

- https://github.com/SigmaHQ/sigma/blob/9f27ab5426a0b061f1f2787e3dc947d6d75ad8c0/rules/windows/process_creation/win_grabbing_sensitive_hives_via_reg.yml

Accessing of LSASS by Mimikatz:

- https://github.com/SigmaHQ/sigma/blob/b81839e3ce507df925d6e583e569e1ac3a3894ab/rules/windows/deprecated/sysmon_mimikatz_detection_lsass.yml

Defense Evasion:

While performing their activities, threat actors do take certain measures to avoid getting caught. Most common being disabling of Windows Defender AV.

Sigma Rules:

Microsoft Defender critical security components disabled (command)

- https://github.com/mdecrevoisier/SIGMA-detection-rules/blob/main/windows-defender/defender-critical%20security%20components%20disabled%20(command).yaml

Microsoft Defender critical security components disabled (PowerShell)

- https://github.com/mdecrevoisier/SIGMA-detection-rules/blob/main/windows-defender/defender-critical%20security%20components%20disabled%20(PowerShell).yaml

Windows Defender Threat Detection Disabled

- https://github.com/SigmaHQ/sigma/blob/f69868b5aa25f33c629208d8868994ed24b20b46/rules/windows/other/win_defender_disabled.yml

Discovery:

Threat actors rely on Windows built-in utilities for performing initial discovery activity.

Sigma Rules:

Domain Trust Discovery

- https://github.com/SigmaHQ/sigma/blob/8bb3379b6807610d61d29db1d76f5af4840b8208/rules/windows/process_creation/proc_creation_win_trust_discovery.yml

Suspicious Reconnaissance Activity

- https://github.com/SigmaHQ/sigma/blob/becf3baeb4f6313bf267f7e8d6e9808fc0fc059c/rules/windows/process_creation/proc_creation_win_susp_recon_activity.yml

Local Accounts Discovery

- https://github.com/SigmaHQ/sigma/blob/ff0f1a0222b5100120ae3e43df18593f904c69c0/rules/windows/process_creation/win_local_system_owner_account_discovery.yml

Net.exe Execution

- https://github.com/SigmaHQ/sigma/blob/08ca62cc8860f4660e945805d0dd615ce75258c1/rules/windows/process_creation/win_susp_net_execution.yml

Whoami Execution

- https://github.com/SigmaHQ/sigma/blob/08ca62cc8860f4660e945805d0dd615ce75258c1/rules/windows/process_creation/win_susp_whoami.yml

Windows Network Enumeration

- https://github.com/SigmaHQ/sigma/blob/ff0f1a0222b5100120ae3e43df18593f904c69c0/rules/windows/process_creation/win_net_enum.yml

CHCP CodePage Locale Lookup

- https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/CHCP%20CodePage%20Locale%20Lookup

They also deploy 3rd party tools such as Adfind for further enumeration.

Sigma Rules:

AdFind Discovery

- https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/adfind_discovery

Advanced IP Scanner:

- https://github.com/SigmaHQ/sigma/blob/master/rules/windows/process_creation/proc_creation_win_advanced_ip_scanner.yml

Lateral Movement:

To achieve their objectives, the threat actors pivot to multiple systems such as Domain Controllers, Backup Servers and File Shares using various tools/techniques:

- Sysinternals PsExec
- Post-Exploitation Framework, Cobalt Strike
- Remote WMI Execution

Sigma Rules

WMIC:

Suspicious WMI Execution

- https://github.com/SigmaHQ/sigma/blob/8bb3379b6807610d61d29db1d76f5af4840b8208/rules/windows/process_creation/proc_creation_win_susp_wmi_execution.yml

WMI Remote Command Execution

- https://github.com/SigmaHQ/sigma/blob/c5263039ae6e28a09192b4be2af40fea59a06b08/rules/windows/process_creation/proc_creation_win_wmic_remote_command.yml

PsExec:

PsExec Tool Execution

- https://github.com/SigmaHQ/sigma/blob/master/rules/windows/pipe_created/pipe_created_tool_psexec.yml

PsExec Tool Execution

- https://github.com/SigmaHQ/sigma/blob/master/rules/windows/file_event/file_event_win_tool_psexec.yml

PSEXEC Custom Named Service Binary

- https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/PSEXEC%20Custom%20Named%20Service%20Binary

Copy to Admin Shares:

- https://github.com/SigmaHQ/sigma/blob/master/rules/windows/process_creation/proc_creation_win_susp_copy_lateral_movement.yml

Cobalt Strike:

- Cobalt Strike, a Defender's Guide
- Cobalt Strike, a Defender's Guide – Part 2

Collection & Exfiltration:

Once the threat actors have pivoted to systems of their interest, they gather the data present on them and attempt to exfiltrate it out of the victim environment with the help of utilities such as Rclone, WinSCP and FileZilla.

Rclone:

Rclone being utilized to exfiltrate the data towards "MEGA" cloud storage services.

Sigma Rules:

Rclone Execution via Command Line or PowerShell

- https://gist.github.com/beardofbinary/fede0607e830aa1add8deda3d59d9a77#file-rclone_execution-yaml

Rclone config file creation

- https://gist.github.com/beardofbinary/b46b87b7a2d9b7c0bd3e7aa918338b75#file-rclone_config_creation-yaml

DNS Query for MEGA.io Upload Domain

- https://gist.github.com/beardofbinary/d46c3b4e37ba8b21a79a63fbf69c6411#file-mega_dns_lookup-yaml

A good reference on Rclone/Mega: https://redcanary.com/blog/rclone-mega-extortion/

In the past, we have seen Ransomware operators upload the LSASS dump file to ufile.io.

While uploading the file, a DNS request is made to "ufile.io" subdomain based on the geographical location.

Example:

Sigma Rule:

Operator Bloopers

We reported that its not uncommon for threat actors to make mistakes when conducting hands-on keyboard actions on endpoints. Whilst an operator will follow a playbook – that details the procedure to follow, sometimes mistakes will occur.

We have two Sigma detection rules that detect Cobalt Strike operator errors, where Cobalt Strike specific commands or modules are inadvertently entered directly on the host via the shell.

Both Sigma rules are based on popular Cobalt Strike commands or have observed operators utilizing in various cases.

For example:

If an operator entered commands, and this was captured on the endpoint via Sysmon for example, it would be possible to detect this action.

For example, if the operator entered 'av_query', (we have observed such mistakes in past cases), using the Sigma rule, we could observe the following output.

Our 'Operator Bloopers…' Sigma rules can be found within our GitHub repo as:

Operator Bloopers Cobalt Strike Modules

- https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/win_cobaltstrike_operator_bloopers_modules.yml

Operator Bloopers Cobalt Strike Commands

- https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/win_cobaltstrike_operator_bloopers_cmds.yml

Bring Your Own Tools (BYOT)

Custom Tools

In various intrusions we observe operators deploying custom tooling directly from endpoints or staging files locally.

Sometimes we see the same filenames for scripts, or very specific command-line parameters being utilized across cases. A frequent script, is ‘adf.bat’, that packages together a number of ADFind commands together.

The following Sigma rule detects several .BAT files we have observed being executed. Examples files are listed below – its not difficult to figure out the intent:

- adf.bat
- adfind.bat
- locker.bat
- kill.bat
- def.bat
- start.bat
- shadow.bat
- logdelete.bat
- closeapps.bat

The Sigma rule:

Our ‘BYOT’ Sigma rule can be found within our GitHub repo as:

Operator Bring Your Own Tools

- https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/win_software_byot.yml

Summary

The above provides a number of detections that could be deployed to detect malicious and suspicious activity that could be indicative of intruder actions.

Its important to note whilst the detections aim to detect actions, its not a substitute for preventing such activity from occurring in the first place. As mentioned in our previous report, following good practices such as hardening, maintaining software updates and provision of an anti-virus solution should stop initial access and establish foothold stages from materializing.

« WILL THE REAL MSIEXEC PLEASE STAND UP? EXPLOIT LEADS TO DATA EXFILTRATION

SELECT XMRIG FROM SQLSERVER »

Register For Our Next CTF

Reports

Threat Intelligence

Detection Rules

DFIR Labs

Mentoring and Coaching