LOGPOINT

EMERGING THREAT:
RASPBERRY ROBIN, NOT A JUICY RASPBERRY YOU LOVE

Written by:

Swachchhanda Shrawan Poudel

www.logpoint.com

FOREWARD

In today's evolving climate of cybersecurity threats, the emergence of sophisticated malware offers significant challenges for businesses. Among these dangers is Raspberry Robin, a virus with worm characteristics that demands our attention
and vigilance.

Raspberry Robin, first discovered by Red Canary in 2021, started as a worm that propagated over USB devices, gaining access to afflicted PCs. However, as it evolved, it adopted more advanced approaches, such as using Discord to transmit malicious payloads and exploiting zero-day vulnerabilities like CVE-2023-36802 for local privilege escalation. The newest CheckPoint results, published on February 7, 2024, provide insight into Raspberry Robin's growing attack vectors, such as abuse of DLL sideloading.

Furthermore, Raspberry Robin has been used as a loader to deploy various malware versions, including ransomware and crypto-miners. IcedID, Bumblebee, and Truebot are some of the most notable malware deployed through Raspberry Robin. Its relationship with prominent hostile groups such as Evil Corp, Silence, FIN11, and TA505 emphasizes its importance in the threat scene.

In our investigation, we looked into the behavior of Raspberry Robin variations, discovering complex execution routes, including the dumping and proxy execution of these malicious DLL files. Variants detected launching processes, such as rundll32.exe and regsvr32.exe, demonstrate the complexities of Raspberry Robin operations.

As we explore Raspberry Robin's techniques, we must proactively build our defenses against such dangers. Understanding Raspberry Robin's mode of operation and developing techniques will allow us to better protect our systems and data from the threats presented by this and related malware variants.

Join us as we further explore Raspberry Robin's complexities, aiming to strengthen our cybersecurity defenses in the face of an ever-changing threat landscape.

LOGPOINT

CONTENTS

ABOUT THE AUTHOR

Swachchhanda Shrawan Poudel

Logpoint Security Research

Swachchhanda Shrawan Poudel is a cybersecurity enthusiast with a bachelor's degree in cybersecurity and certification as an ethical hacker. With an interest in both offensive and defensive security, he currently works as a Security Researcher at Logpoint, focusing on detection engineering, threat hunting, and remediation.

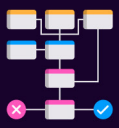
ABOUT EMERGING THREAT PROTECTION

The cybersecurity threat landscape continuously changes while new risks and threats are constantly discovered. Only some organizations have enough resources or the know-how to deal with evolving threats.

Emerging Threats Protection is a managed service provided by a Logpoint team of highly skilled security researchers who are experts in threat intelligence and incident response. Our team informs you of the latest threats and provides custom detection rules and tailor-made playbooks to help you Investigate and Respond to emerging threats.

****All new detection rules are available in Logpoint's latest release** and through the Logpoint Help Center. Customized investigation and response playbooks are open to all Logpoint Emerging Threats Protection customers.

1. Research for emerging threats such as malware families, threat actors and vulnerabilities
2. Data retrieval e.g., malware samples, IOCs, and TTP



1. Analysis of the collected data and malware and, tracking of threat actors' activities
2. Creation and update analytics and playbooks
3. Writing of ETP report

1. Publishing of report

1. Continuous monitoring for other emerging threats to create next ETP report

Below is a rundown of the incident, potential threats, and how to detect possible attacks and proactively defend using Logpoint Converged SIEM capabilities for detection, investigation, and response.

INTRODUCTION

Raspberry Robin is a malware with worm capabilities initially identified by Red Canary in 2021 but revealed in May 2022 through their report. In its early stages, primary sources of infection include **removable storage devices,** such as USB drives, to establish a foothold on infected systems. However, it has since evolved to leverage Discord to deliver malicious payloads and exploit n-days for more devastating effects.

Moreover, it has been employed as a loader malware to drop other malware variants, ranging from ransomware and stealers, but not limited to crypto-miners. The notable second-stage payloads that have been dropped through the usage of Raspberry Robin include IcedID, Bumblebee, Truebot, etc. Over time, this worm has continued to evolve, exhibiting noteworthy characteristics.

The usage of Raspberry Robin has been linked with highly notorious malicious groups like Evil Corp, Silence (aka Whisper Spider), FIN11, TA505, Clop, etc. However, the authors and maintainers remain unknown.

INFECTION CHAIN

When the Raspberry Robin first came into the limelight, it slowly gained popularity as a worm infected via a USB drive. These USB drives would contain malicious shortcut '.LNK' files masquerading as a thumb drive or a network share. According to Malwarebytes, Raspberry Robin affiliates would start the LNK file via autoruns and utilize social engineering to urge victims to click on it. When they click on the LNK file, cmd.exe launches the Windows Installer service msiexec.exe, which installs a malicious payload on infected QNAP network-attached storage (NAS) devices.

However, the latest report from CheckPoint, released on Feb 7, 2024, mentioned that the attack flow started from the archive downloaded from Discord as an attachment. The archive contains a legitimate Windows-signed binary with an unsigned malicious DLL file. That legitimate binary was used to load that unsigned malicious DLL file through the DLL side-loading technique. Further, the report elaborates on how the malware exploits **CVE-2023-36802** for local privilege escalation (LPE) even before the advisory on active exploitation of this vulnerability was revealed by Microsoft and CISA in Septemeber 2023. They believe the Raspberry Robin affiliates bought the exploit for **CVE-2023-36802** from Dark Web forums as it was on sale on Dark Web Forums in February 2023.

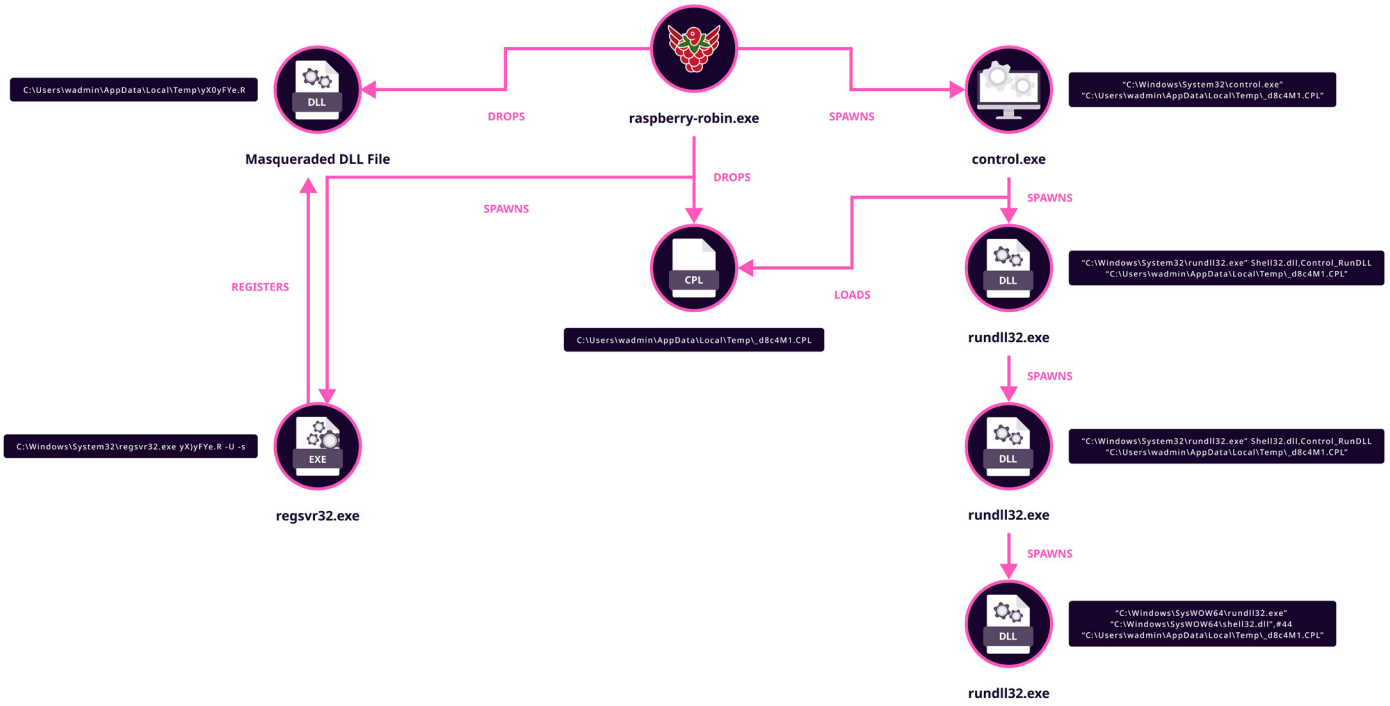
During our analysis, we examined an executable file, specifically a RarSfx wrapper malware. The high-level flow of this executable involves dropping a malicious DLL file upon execution. In certain variants, this DLL file was found in the form of a .cpl file, which is also a special kind of Windows DLL. It was loaded by control.exe, subsequently spawning a rundll32.exe process as a child. The .cpl file is then executed using the Control_RunDLL function from SHELL32.dll with the following command:

```
1  "C:\Windows\system32\rundll32.exe" Shell32.dll,Control_RunDLL "C:\Users\wadmin\AppData\Local\Temp\_d8c4M1.CPL"
```

In some samples, we observed the executable spawning regsvr32.exe, which registers the dropped malicious DLL file using the following command:

```
1  C:\Windows\System32\regsvr32.exe yXOyFYe.R -U -s
```

The high-level behavioral flow chart of analyzed raspberry-robin variants is given below.

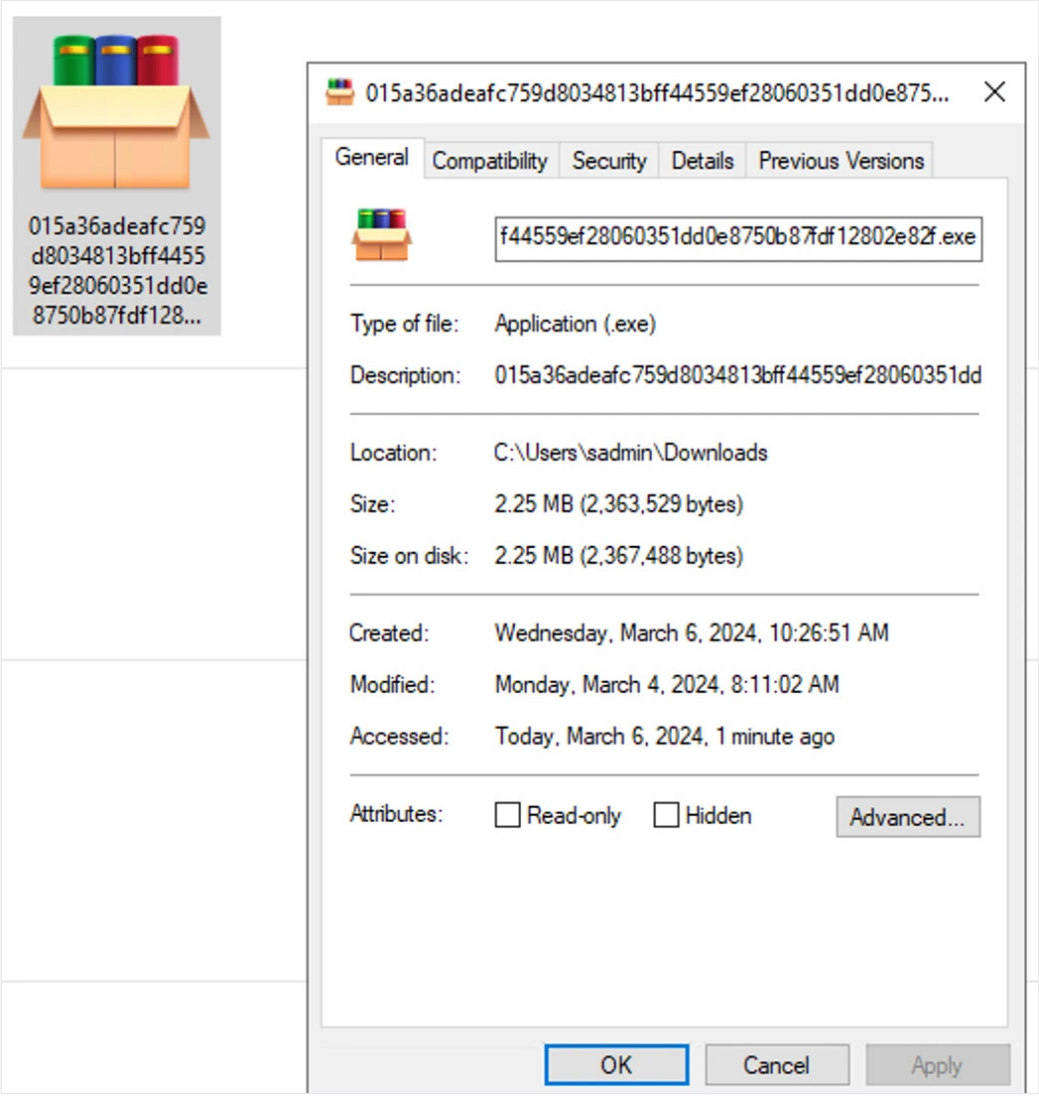


The detailed technical analysis of these malware variants can be found in the 'Technical Analysis Report' section of the report.

TECHNICAL ANALYSIS REPORT

This report highlights technical details of the latest Raspberry Robin sample downloaded from MalwareBaazar on March 6, 2024. The analysis commenced with static analysis, followed by dynamic analysis utilizing Logpoint's new plugin, "Process Tree." Detailed steps of the analysis are outlined below.

Static Analysis



At first look, the sample seemed to be a Windows executable (exe). However, malware can often manifest in various formats, such as wrappers or installers, prompting us to delve deeper into its nature. We then utilized the Sysinternals 'strings.exe' utility to extract and inspect the binary's string contents. Upon analyzing the strings, we noted the presence of '.zipx' and 'unzip,' which piqued our interest. This prompted further investigation into the possibility of it being a wrapper malware.

```
Please remove %s from %s folder. It is unsecure to run %s until it is done.
%s: %s
map/set too long
AES-0017
.zipx
z%s%02d
#+3;CScs
#+3;CScs
UnZip: Internal error 1
ARarHtmlClassName
Shell.Explorer
about:blank
```

Further examination of the strings file, we observed interesting other interesting strings such as 'RarSFX,' 'winrarsfxmappingfile.tmp,' 'sfxname,' etc. The identified strings strongly suggest that these executables are WinRAR self-extracting archives (RarSFX).

> RarSFX archives provide advanced functionality through extended SFX commands, enabling actions to be executed upon successful extraction. One such command allows specifying an executable to run after extraction completes. Unfortunately, this feature is often exploited by malicious actors who embed commands within SFX archives to execute harmful actions upon extraction. These actions may not necessarily involve embedding malware within the archive itself but instead leveraging native tools to carry out malicious commands as part of the extraction process.

It appears that this Raspberry Robin variant is also likely abusing this feature.

```
RarSFX
STATIC
unknown_folder
REPLACEFILEDLG
RENAMEDLG
%s %s
GETPASSWORD1
winrarsfxmappingfile.tmp
sfxname
%4d-%02d-%02d-%02d-%02d-%02d-%03d
sfxstime
STARTDLG
sfxcmd
```

Strings - Possible indication of RarSFX

```
WinRAR self-extracting archive
MS Shell Dlg 2
&Destination folder
Bro&wse...
hRichEdit20W
Installation progress
jmsctls_progress32
Install
Cancel
Confirm file replace
MS Shell Dlg 2
The following file already exists
Would you like to replace the existing file
with this one?
```

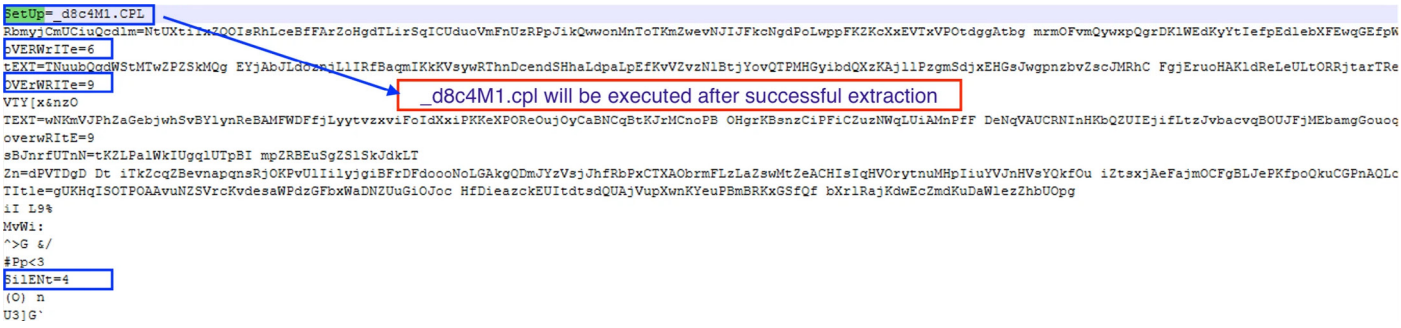
Additionally, in the content of the strings, we discovered XML data related to WinRAR SFX, further supporting our hypothesis.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<assembly xmlns="urn:schemas-microsoft-com:asm.v1" manifestVersion="1.0">
<assemblyIdentity
        version="1.0.0.0"
        processorArchitecture="*"
        name="WinRAR SFX"
        type="win32"/>
<description>WinRAR SFX module</description>
<trustInfo xmlns="urn:schemas-microsoft-com:asm.v2">
        <security>
                <requestedPrivileges>
                        <requestedExecutionLevel level="asInvoker"
                        uiAccess="false"/>
                </requestedPrivileges>
        </security>
</trustInfo>
<dependency>
        <dependentAssembly>
                <assemblyIdentity
                        type="win32"
                        name="Microsoft.Windows.Common-Controls"
                        version="6.0.0.0"
                        processorArchitecture="*"
                        publicKeyToken="6595b64144ccf1df"
                        language="*"/>
        </dependentAssembly>
</dependency>
<compatibility xmlns="urn:schemas-microsoft-com:compatibility.v1">
        <application>
                        <!--The ID below indicates application support for Windows Vista -->
                                <supportedOS Id="{e2011457-1546-43c5-a5fe-008deee3d3f0}"/>
                        <!--The ID below indicates application support for Windows 7 -->
                                <supportedOS Id="{35138b9a-5d96-4fbd-8e2d-a2440225f93a}"/>
                        <!--The ID below indicates application support for Windows 8 -->
                                <supportedOS Id="{4a2f28e3-53b9-4441-ba9c-d69d4a4a6e38}"/>
                        <!--The ID below indicates application support for Windows 8.1 -->
                                <supportedOS Id="{1f676c76-80e1-4239-95bb-83d0f6d0da78}"/>
                        <!--The ID below indicates application support for Windows 10 -->
                                <supportedOS Id="{8e0f7a12-bfb3-4fe8-b9a5-48fd50a15a9a}"/>
        </application>
</compatibility>
<asmv3:application xmlns:asmv3="urn:schemas-microsoft-com:asm.v3">
        <asmv3:windowsSettings xmlns="http://schemas.microsoft.com/SMI/2005/WindowsSettings">
                <dpiAware>true</dpiAware>
        </asmv3:windowsSettings>
</asmv3:application>
</assembly>
```

Further investigation led to the identification of another interesting string, '_d8c4M1.cpl'. When the main binary was executed, the SFX archiver seemed to execute this control panel file (.cpl) in the background.

> A CPL (Control Panel) file is a special type of binary file (special dll) in the Windows operating system that serves as a gateway to various system tools accessible through the control panel interface. These files, typically with a .cpl extension, are designed to open within the control panel and provide access to settings and configuration options for devices, applications, and system components.

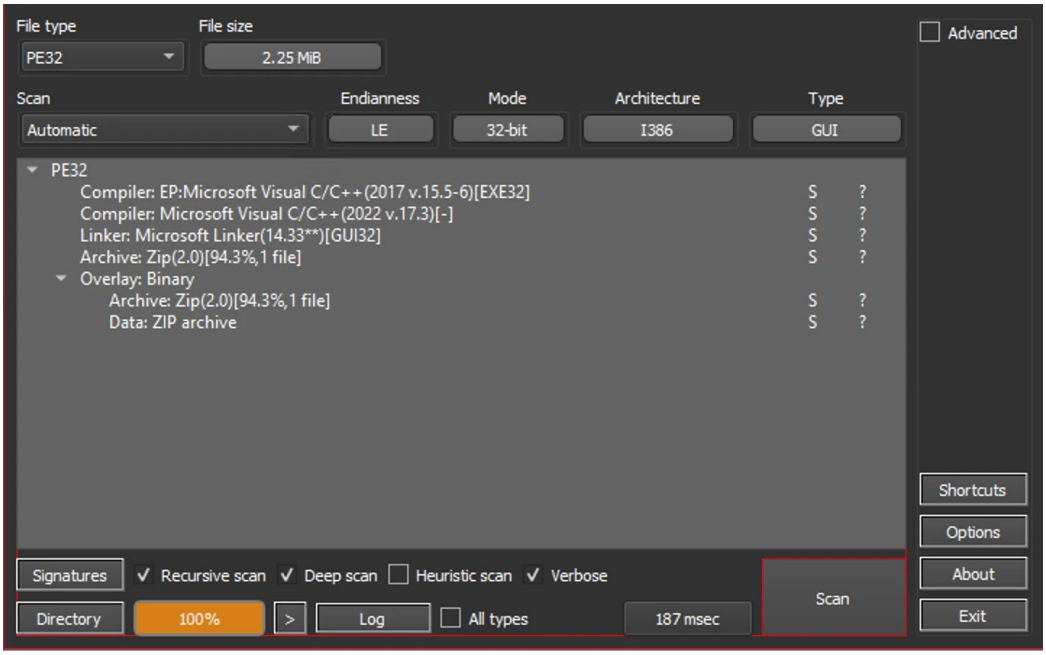
In RarSFX archives, there are predefined fields such as Setup, Overwrite, Text, Silent, Title, etc. These fields dictate the behavior of the self-extracting (SFX) functionality. Specifically, the Setup field typically specifies the program or code to execute after successful extraction, indicated by "Setup=<program>." During our analysis, we observed these fields in the provided screenshot. Upon examining the strings file, we found a noteworthy entry containing "Setup=_d8c4M1.cpl". This suggests that the CPL file "_d8c4M1.cpl" will be executed upon successful extraction.



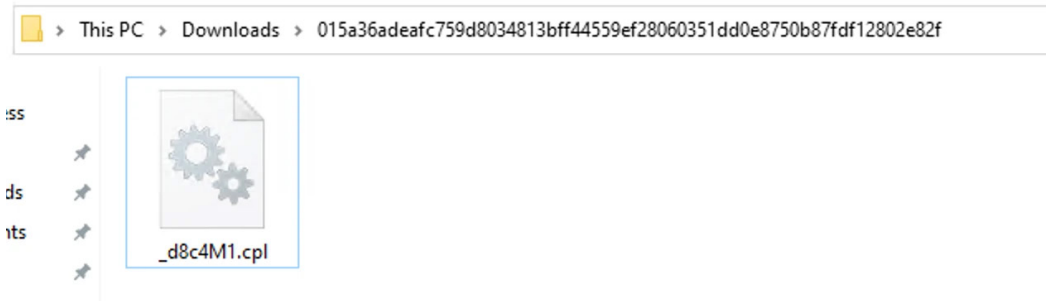
Additionally, it was noted that the execution path is set to the '%temp%' directory.

```
2xMFO
pi7p`
setUpcode
jpzVJEhkIeRlfTnYQf=psxaUkwVqHVcRzTKVf bShFuv DZetJhUDSRsLZZvzUwIre
path=%tEMP%
AZicXXyhIDBPlm=PADiSFmaQEcXfvdpIJtMSHehlSuaWxbmwdVvZMstVVUGxxLPkeG
TexT=OQtdCAewddSRwsNRjEUrbxPTbgFFEZlSUplpKRtrmhjcALFXXJjECOjOm kBK
TITLE=l FyRghjqTIIPdKmfO HLtvDZf
-2)woUkc
```

Analysis through DetectItEasy also hinted that the file is, in fact, a Zip Archive Installer.



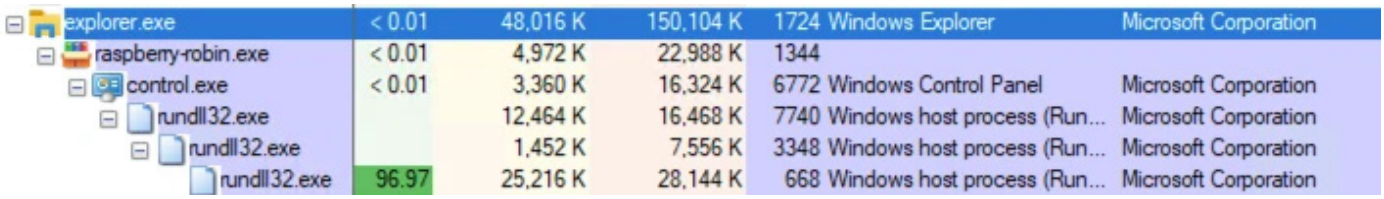
Following this revelation, an attempt was made to extract the executable file. We extracted it as how anyone would extract a normal zip file. Interestingly, the file '_d8c4M1.cpl', previously observed in the strings, was successfully extracted.



Dynamic Analysis

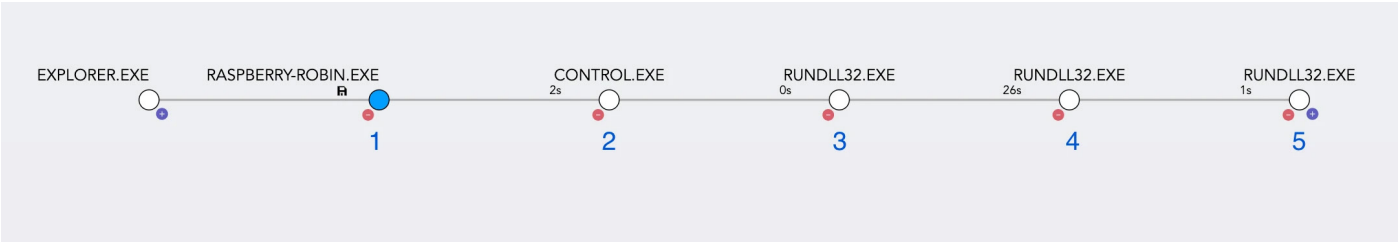
After the discovery, the malware was detonated to assess its dynamic behavior. Sysmon was used to offer thorough logging on the sandbox, allowing for effective analysis. Following that, all relevant logs were sent to Logpoint for further inspection. Utilizing the newly integrated Logpoint plugin "Process Tree," we visualized the parent-child process relationships along with additional information such as command-line parameters, disk operations, network connections, and registry-related activities.

We conducted a comparative analysis of Sysinternals' ProcMon, Process Explorer, and Logpoint's Process Tree features to showcase how Logpoint Process Tree, akin to these Sysinternals tools, provides valuable assistance in forensic investigations. The binary was renamed 'raspberry-robin.exe' throughout the examination and executed for further research.



explorer.exe	< 0.01	48,016 K	150,104 K	1724 Windows Explorer	Microsoft Corporation
raspberry-robin.exe	< 0.01	4,972 K	22,988 K	1344	
control.exe	< 0.01	3,360 K	16,324 K	6772 Windows Control Panel	Microsoft Corporation
rundll32.exe		12,464 K	16,468 K	7740 Windows host process (Run...	Microsoft Corporation
rundll32.exe		1,452 K	7,556 K	3348 Windows host process (Run...	Microsoft Corporation
rundll32.exe	96.97	25,216 K	28,144 K	668 Windows host process (Run...	Microsoft Corporation

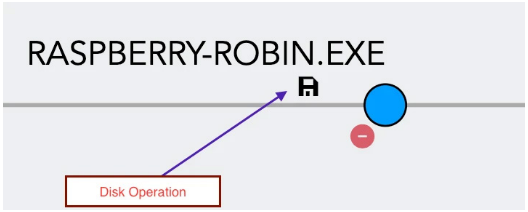
Using Process Explorer, we observed raspberry-robin.exe spawning control.exe as child processes, followed by rundll32.exe as child processes of control.exe. Subsequently, rundll32.exe spawned another instance of rundll32.exe. This identical process tree was also visualized through the Logpoint Process Tree.



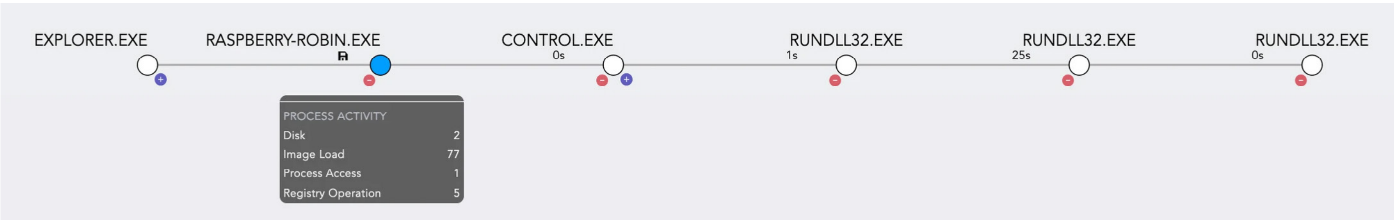
Let's break down each process node, beginning with process raspberry-robin.exe and numbering its child processes. Raspberry-robin.exe will be numbered as 1, control.exe as 2, and so on for the subsequent child processes.

1. raspberry-robin.exe

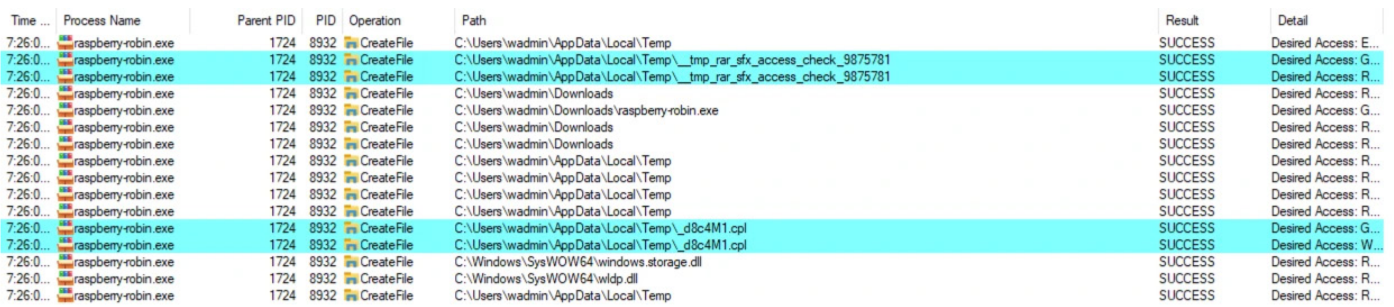
Upon initial inspection of the process tree, we noticed a disk operation indicator associated with the process node 'raspberry-robin.exe,' denoted by the memory sign. This sign suggests that the process has performed disk write operations, potentially indicating that files were dropped.



Hovering over this node, we observed a summary of process activity, including disk writes, image loads, process access, and registry operations, as depicted in the screenshot below. Full details could be observed by double-clicking on the node.

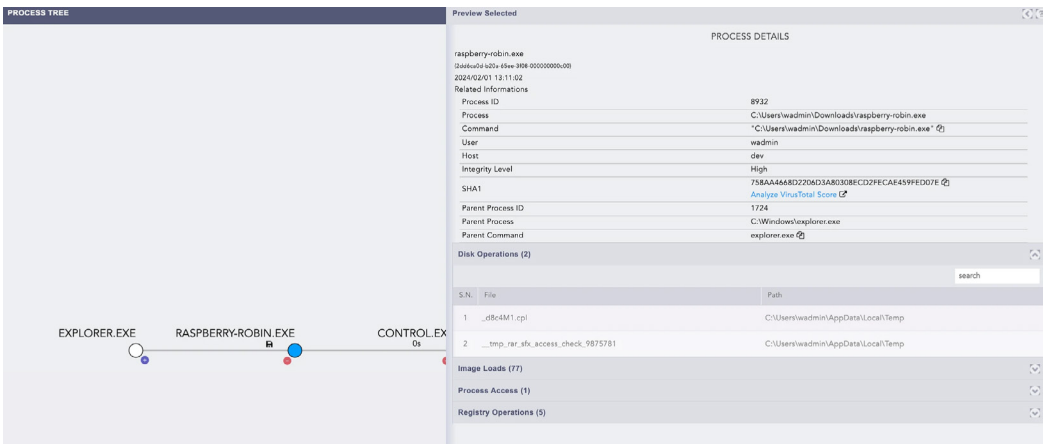


Through the file system activity captured by ProcMon, it was revealed that raspberry-robin.exe dropped two files, namely '__tmp_rar_sfx_access_check_9875781' and '_d8c4M1.cpl', into the temp folder.



Time ...	Process Name	Parent PID	PID	Operation	Path	Result	Detail
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Users\wadmin\AppData\Local\Temp	SUCCESS	Desired Access: E...
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Users\wadmin\AppData\Local\Temp__tmp_rar_sfx_access_check_9875781	SUCCESS	Desired Access: G...
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Users\wadmin\AppData\Local\Temp__tmp_rar_sfx_access_check_9875781	SUCCESS	Desired Access: R...
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Users\wadmin\Downloads	SUCCESS	Desired Access: R...
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Users\wadmin\Downloads\raspberry-robin.exe	SUCCESS	Desired Access: G...
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Users\wadmin\Downloads	SUCCESS	Desired Access: R...
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Users\wadmin\Downloads	SUCCESS	Desired Access: R...
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Users\wadmin\AppData\Local\Temp	SUCCESS	Desired Access: R...
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Users\wadmin\AppData\Local\Temp	SUCCESS	Desired Access: R...
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Users\wadmin\AppData\Local\Temp	SUCCESS	Desired Access: R...
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Users\wadmin\AppData\Local\Temp	SUCCESS	Desired Access: R...
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Users\wadmin\AppData\Local\Temp_d8c4M1.cpl	SUCCESS	Desired Access: G...
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Users\wadmin\AppData\Local\Temp_d8c4M1.cpl	SUCCESS	Desired Access: W...
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Windows\SysWOW64\windows.storage.dll	SUCCESS	Desired Access: R...
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Windows\SysWOW64\wldp.dll	SUCCESS	Desired Access: R...
7:26:0...	raspberry-robin.exe	1724	8932	CreateFile	C:\Users\wadmin\AppData\Local\Temp	SUCCESS	Desired Access: R...

The same information was also displayed via process details in the process tree.



As depicted in this screenshot, there were also events related to image loads and process access.

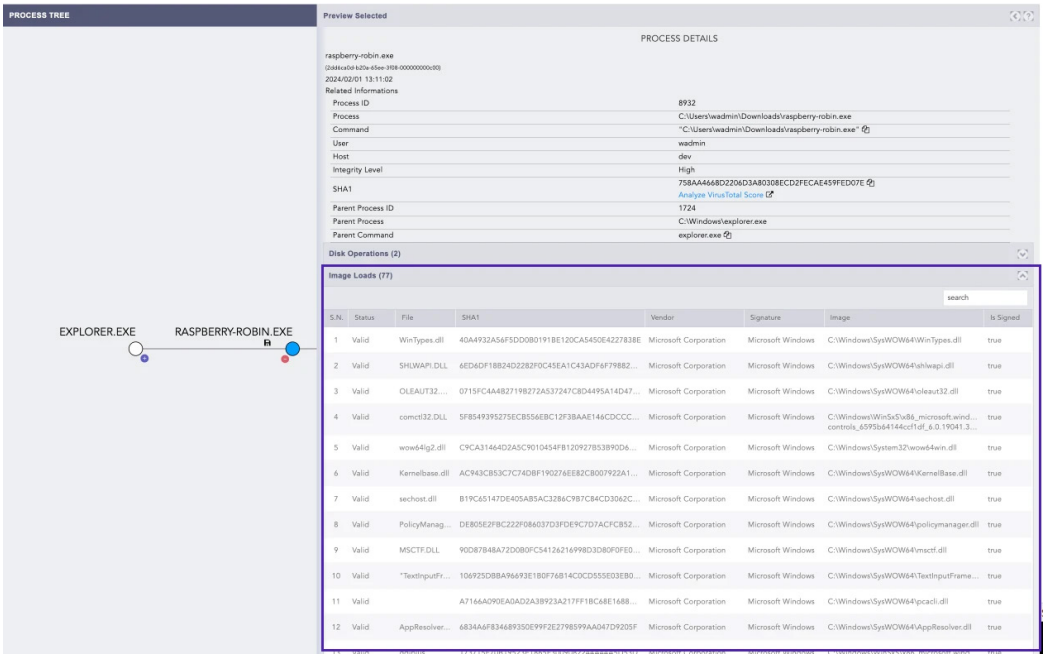
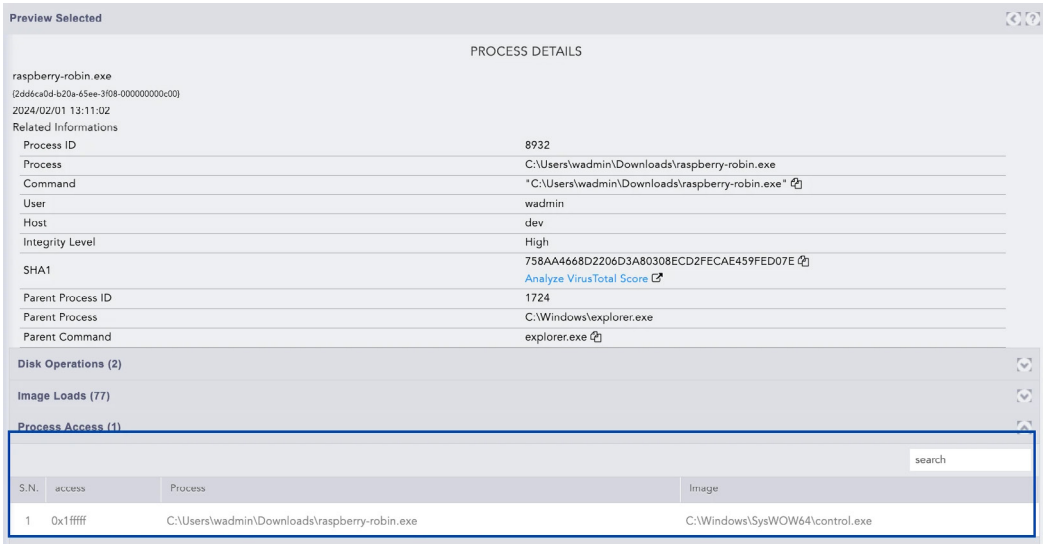


Image Loaded by raspberry-robin.exe

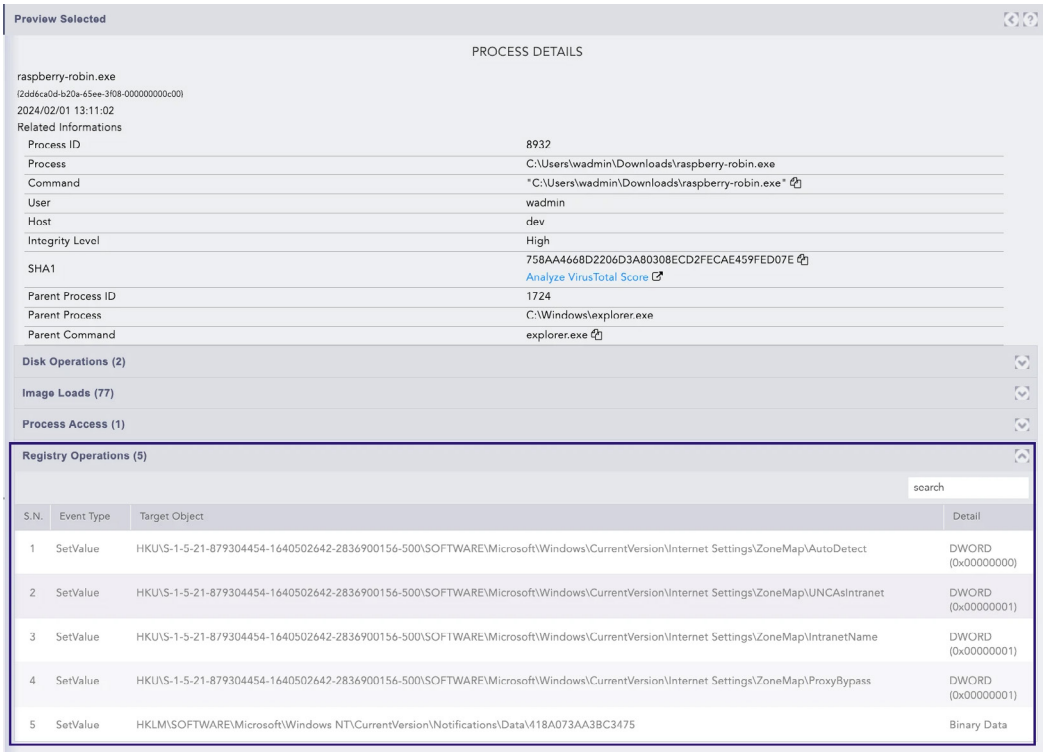
The process access event was interesting as raspberry-robin.exe accessed the process ‘control.exe’ with the access of ‘0x1fffff’. The access level ‘0x1fffff’ typically indicates full access rights to the ‘control.exe’ process, allowing raspberry-robin.exe to perform a wide range of operations and modifications within the context of ‘control.exe.’ This level of access could potentially enable raspberry-robin.exe to manipulate or control the behavior of the ‘control.exe’ process.As depicted in this screenshot, there were also events related to image loads and process access.



Furthermore, while observing registry activities in Procmon, the “RegSetValue” operation was filtered for the process “raspberry-robin.exe.” Some registry modifications related to the system’s proxy configuration were observed, as depicted in the screenshot.

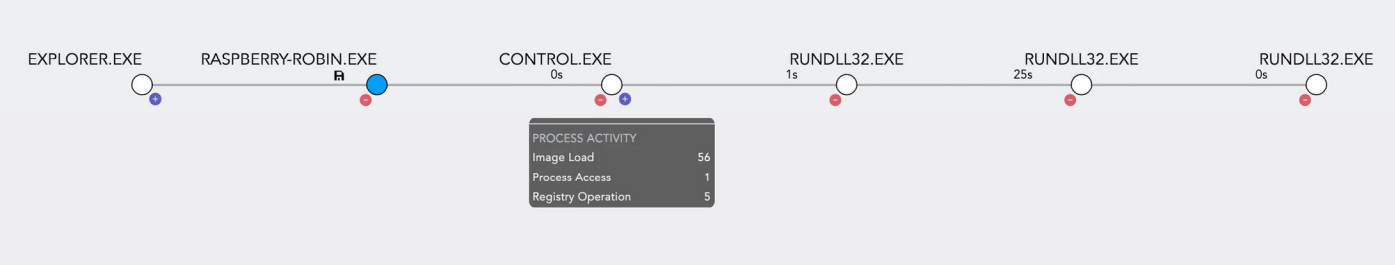
Time of Day	Process Name	Parent PID	PID	Operation	Path	Detail	Result
121327 AM	raspberry-robin.exe	1724	8932	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\FileExts\.cpl\OpenWithProgids\cplfile	Type: REG_NONE, Length: 0	SUCCESS
678142 AM	raspberry-robin.exe	1724	8932	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\ProxyBypass	Type: REG_DWORD, Length: 4, Data: 1	SUCCESS
678630 AM	raspberry-robin.exe	1724	8932	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\IntranetName	Type: REG_DWORD, Length: 4, Data: 1	SUCCESS
679011 AM	raspberry-robin.exe	1724	8932	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\UNCAsIntranet	Type: REG_DWORD, Length: 4, Data: 1	SUCCESS
679351 AM	raspberry-robin.exe	1724	8932	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\AutoDetect	Type: REG_DWORD, Length: 4, Data: 0	SUCCESS
698866 AM	raspberry-robin.exe	1724	8932	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\ProxyBypass	Type: REG_DWORD, Length: 4, Data: 1	SUCCESS
699476 AM	raspberry-robin.exe	1724	8932	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\IntranetName	Type: REG_DWORD, Length: 4, Data: 1	SUCCESS
699781 AM	raspberry-robin.exe	1724	8932	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\UNCAsIntranet	Type: REG_DWORD, Length: 4, Data: 1	SUCCESS
700060 AM	raspberry-robin.exe	1724	8932	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\AutoDetect	Type: REG_DWORD, Length: 4, Data: 0	SUCCESS
332752 AM	raspberry-robin.exe	1724	8932	RegSetValue	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Notifications\Data\418A073AA3BC3475	Type: REG_BINARY, Length: 130, Data: 86 02 00 00 00 00 00 00 04 00...	SUCCESS

Similar registry “set value” operations may also be visible through the Logpoint Process Tree. Raspberry Robin might have altered these proxy settings to circumvent security measures, ensuring uninterrupted connections with Command and Control servers, particularly if any proxy settings are blocking connections.



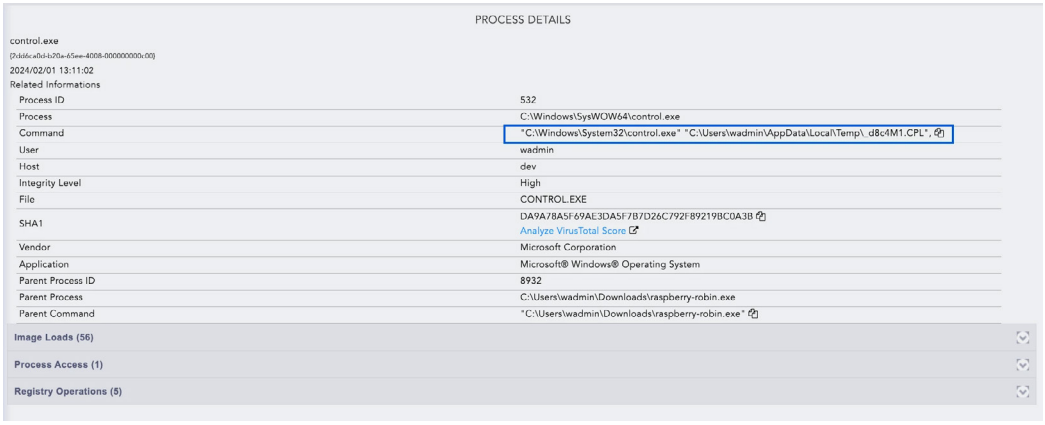
2. control.exe

Hovering over the “control.exe“ node displayed a similar process activity as raspberry-robin.exe. But no disk operation was performed, so no disk operation is shown.

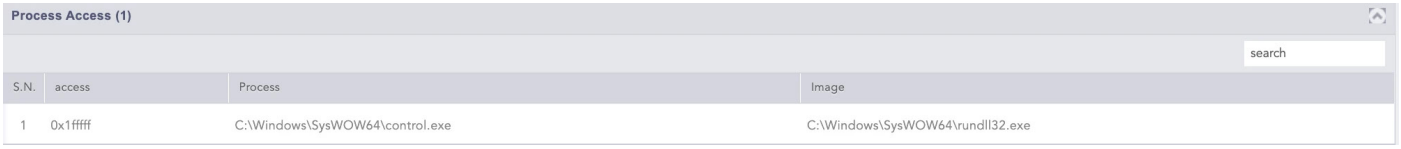


Upon further inspection, the command line of this process appeared suspicious. The control.exe was found executing the ‘_d8c4M1.cpl’ from the temp directory, which was earlier dropped by its parent_process “raspberry-robin.exe“.

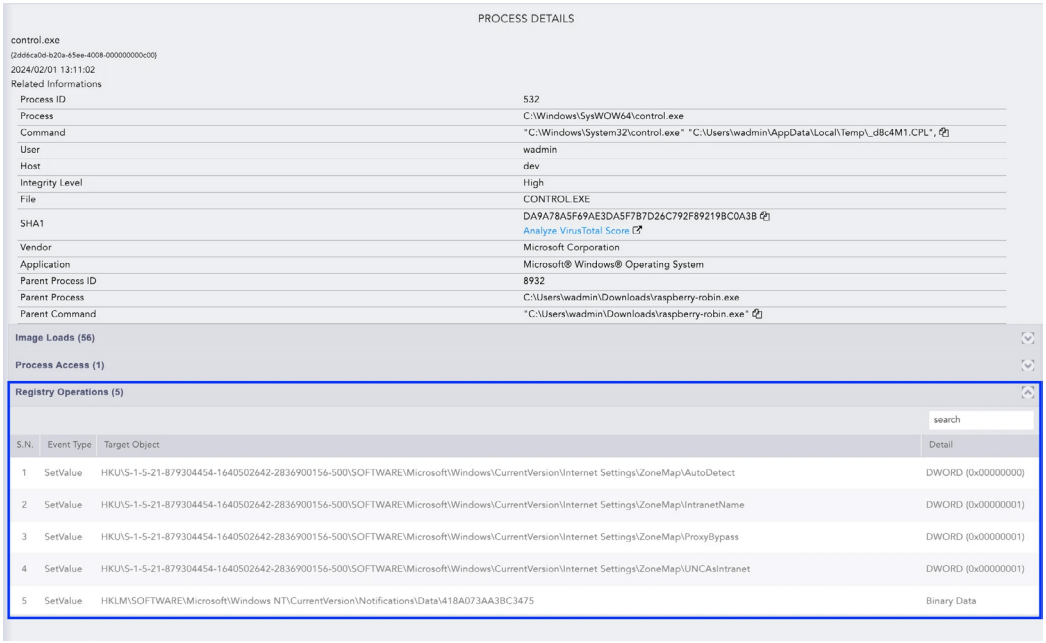
```
1    “C:\Windows\System32\control.exe” “C:\Users\wadmin\AppData\Local\Temp\_d8c4M1.CPL”,
```



Similar to raspberry-robin.exe, there were no suspicious image loads, but a suspicious process access event was observed. Interestingly, ‘control.exe’ accessed the process ‘rundll32.exe’ with the access level of ‘0x1fffff’.



The registry operations were the same as observed with its parent_process 'raspberry_robin.exe.'



PROCESS DETAILS

control.exe
{2dd6ca0d-b20a-65ee-4008-000000000c00}
2024/02/01 13:11:02
Related Informations

Process ID	532
Process	C:\Windows\SysWOW64\control.exe
Command	"C:\Windows\System32\control.exe" "C:\Users\wadmin\AppData\Local\Temp_d8c4M1.CPL",
User	wadmin
Host	dev
Integrity Level	High
File	CONTROL.EXE
SHA1	DA9A78A5F69AE3DA5F7B7D26C792F89219BC0A3B Analyze VirusTotal Score
Vendor	Microsoft Corporation
Application	Microsoft® Windows® Operating System
Parent Process ID	8932
Parent Process	C:\Users\wadmin\Downloads\raspberry-robin.exe
Parent Command	"C:\Users\wadmin\Downloads\raspberry-robin.exe"

Image Loads (56)

Process Access (1)

Registry Operations (5)

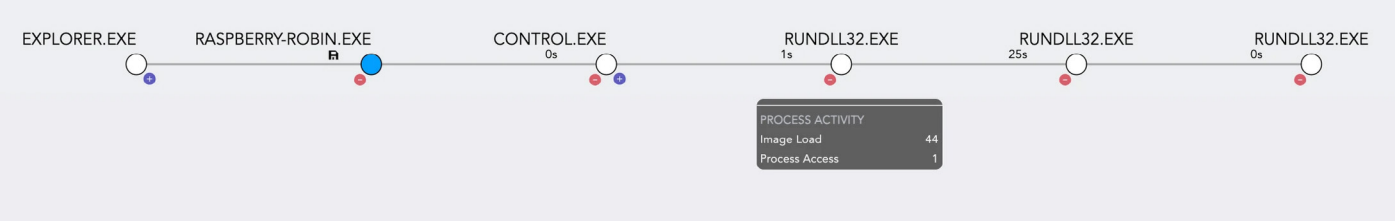
search

S.N.	Event Type	Target Object	Detail
1	SetValue	HKU\S-1-5-21-879304454-1640502642-2836900156-500\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\AutoDetect	DWORD (0x00000000)
2	SetValue	HKU\S-1-5-21-879304454-1640502642-2836900156-500\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\IntranetName	DWORD (0x00000001)
3	SetValue	HKU\S-1-5-21-879304454-1640502642-2836900156-500\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\ProxyBypass	DWORD (0x00000001)
4	SetValue	HKU\S-1-5-21-879304454-1640502642-2836900156-500\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\UNCAsIntranet	DWORD (0x00000001)
5	SetValue	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Notifications\Data\418A073AA3BC3475	Binary Data

ProcMon also recorded the same events.

Time of Day	Process Name	Parent PID	PID	Operation	Path	Detail	Result
9:31:12.5711957 AM	control.exe	1880	8364	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\ProxyBypass	Type: REG_DWORD, Length: 4, Data: 1	SUCCESS
9:31:12.5712532 AM	control.exe	1880	8364	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\IntranetName	Type: REG_DWORD, Length: 4, Data: 1	SUCCESS
9:31:12.5712856 AM	control.exe	1880	8364	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\UNCAsIntranet	Type: REG_DWORD, Length: 4, Data: 1	SUCCESS
9:31:12.5713143 AM	control.exe	1880	8364	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\AutoDetect	Type: REG_DWORD, Length: 4, Data: 0	SUCCESS
9:31:12.5739216 AM	control.exe	1880	8364	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\ProxyBypass	Type: REG_DWORD, Length: 4, Data: 1	SUCCESS
9:31:12.5739830 AM	control.exe	1880	8364	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\IntranetName	Type: REG_DWORD, Length: 4, Data: 1	SUCCESS
9:31:12.5740182 AM	control.exe	1880	8364	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\UNCAsIntranet	Type: REG_DWORD, Length: 4, Data: 1	SUCCESS
9:31:12.5740473 AM	control.exe	1880	8364	RegSetValue	HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\AutoDetect	Type: REG_DWORD, Length: 4, Data: 0	SUCCESS
9:32:13.1680389 AM	control.exe	1880	8364	RegSetValue	HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Notifications\Data\418A073AA3BC3475	Type: REG_BINARY, Length: 390, Data: BB 02 00 00 00 00 00 00 04 00...	SUCCESS

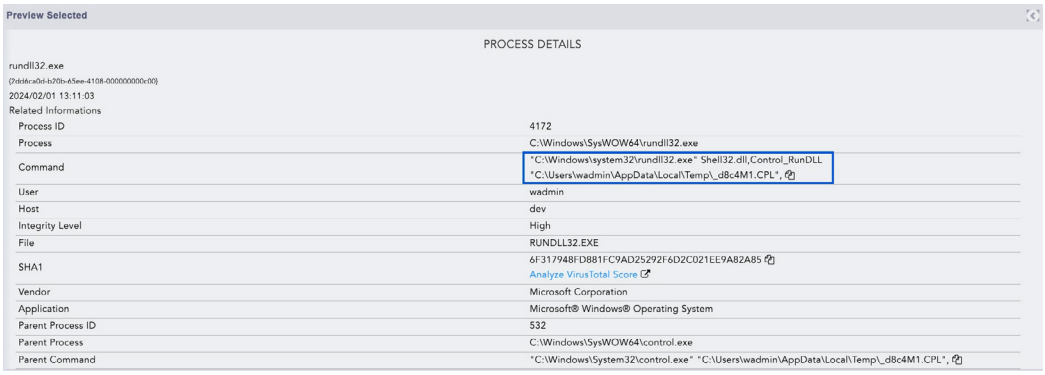
3. rundll32.exe



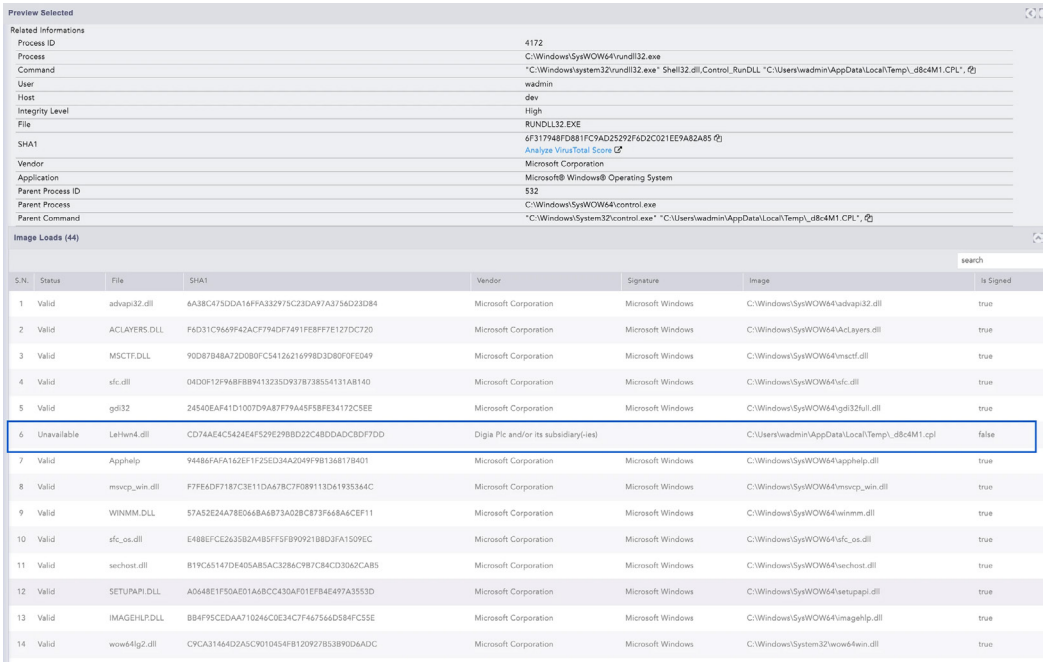
Hovering over the first rundll32.exe, it was observed that it had image load and process access events. In the process details, the command appeared suspicious. It seemed like a proxy execution of malicious code through rundll32. Rundll32.exe had been used to execute the Control_RunDLL function of Shell32.dll, with the malicious cpl file as the argument to the function, using the following command:

```
1  “C:\Windows\system32\rundll32.exe” Shell32.dll,Control_RunDLL “C:\Users\wadmin\AppData\Local\Temp\_d8c4M1.CPL”,
```

The .cpl file extension typically refers to Control Panel files, which are used to provide various configuration options in Windows. The Control_RunDLL function of Shell32.dll is a legitimate Windows function used to execute Control Panel applets. However, in this context, a malicious .cpl file had been executed via rundll32.exe, suggesting a potentially unauthorized or malicious activity.



It was further revealed to be loading an unsigned DLL file, d8c4M1.CPL (original file name LeHwn4.dll), which was earlier dropped by its predecessor process “raspberry-robin.exe“.



Preview Selected

Related Informations

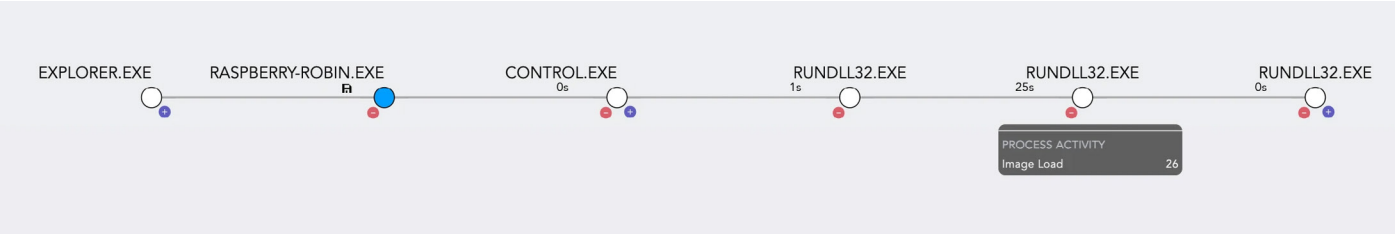
Process ID	4172
Process	C:\Windows\SysWOW64\rundll32.exe
Command	"C:\Windows\system32\rundll32.exe" Shell32.dll,Control_RunDLL "C:\Users\wadmin\AppData\Local\Temp_d8c4M1.CPL",
User	wadmin
Host	dev
Integrity Level	High
File	RUNDLL32.EXE
SHA1	6F3117948FD881FC9AD25292F6D2C021EE9A82A85 Analyze VirusTotal Score
Vendor	Microsoft Corporation
Application	Microsoft® Windows® Operating System
Parent Process ID	532
Parent Process	C:\Windows\SysWOW64\control.exe
Parent Command	"C:\Windows\System32\control.exe" "C:\Users\wadmin\AppData\Local\Temp_d8c4M1.CPL",

Image Loads (44)

search

S.N.	Status	File	SHA1	Vendor	Signature	Image	Is Signed
1	Valid	advapi32.dll	6A38C475DDA16FFA332975C23DA97A3756D23D84	Microsoft Corporation	Microsoft Windows	C:\Windows\SysWOW64\advapi32.dll	true
2	Valid	ACLAYERS.DLL	F6D31C9669F42ACF794DF7491FE8FF7E127DC720	Microsoft Corporation	Microsoft Windows	C:\Windows\SysWOW64\AcLayers.dll	true
3	Valid	MSCTF.DLL	90D87B48A72D0B0FC54126216998D3D80F0FE049	Microsoft Corporation	Microsoft Windows	C:\Windows\SysWOW64\msctf.dll	true
4	Valid	sfc.dll	04D0F12F968FBB8941323SD937B738554131AB140	Microsoft Corporation	Microsoft Windows	C:\Windows\SysWOW64\sfc.dll	true
5	Valid	gdi32	24540EAF41D1007D9A87F79A45F5BFE34172C5EE	Microsoft Corporation	Microsoft Windows	C:\Windows\SysWOW64\gdi32full.dll	true
6	Unavailable	LeHwn4.dll	CD74AE4C5424E4F529E29BBD22C48DDADCBDF7DD	Digia Plc and/or its subsidiary(-ies)		C:\Users\wadmin\AppData\Local\Temp_d8c4M1.cpl	false
7	Valid	Apphelp	94486FAFA162EF1F25ED34A2049F9B136817B401	Microsoft Corporation	Microsoft Windows	C:\Windows\SysWOW64\apphelp.dll	true
8	Valid	msvcp_win.dll	F7FE6DF7187C3E11DA67BC7F089113D61935364C	Microsoft Corporation	Microsoft Windows	C:\Windows\SysWOW64\msvcp_win.dll	true
9	Valid	WINMM.DLL	57A52E24A78E066BA6B73A02BC873F668A6CEF11	Microsoft Corporation	Microsoft Windows	C:\Windows\SysWOW64\winmm.dll	true
10	Valid	sfc_os.dll	E488EFCE2635B2A4B5FF5FB90921B8D3FA1509EC	Microsoft Corporation	Microsoft Windows	C:\Windows\SysWOW64\sfc_os.dll	true
11	Valid	sechost.dll	B19C65147DE405AB5AC3286C9B7C84CD3062CAB5	Microsoft Corporation	Microsoft Windows	C:\Windows\SysWOW64\sechost.dll	true
12	Valid	SETUPAPI.DLL	A0648E1F50AE01A6BCC430AF01EFB4E497A3553D	Microsoft Corporation	Microsoft Windows	C:\Windows\SysWOW64\setupapi.dll	true
13	Valid	IMAGEHLP.DLL	BB4F95CEDAA710246C0E34C7F467566D584FC55E	Microsoft Corporation	Microsoft Windows	C:\Windows\SysWOW64\imagehlp.dll	true
14	Valid	wow64lg2.dll	C9CA31464D2ASC9010454FB1209278538B90D6ADC	Microsoft Corporation	Microsoft Windows	C:\Windows\System32\wow64win.dll	true

4. rundll32.exe



As for the second rundll32.exe, hovering over it showed only image load-associated events. Its parent process was also rundll32.exe, and its command line was the same as the parent command. No significant process activity was observed, only some image-loading events.

PROCESS DETAILS

rundll32.exe
{2dd6ca0d-b224-65ee-4208-000000000c00}
2024/02/01 13:11:28
Related Informations

Process ID	8312
Process	C:\Windows\System32\rundll32.exe
Command	C:\Windows\system32\RunDll32.exe Shell32.dll,Control_RunDLL "C:\Users\wadmin\AppData\Local\Temp_d8c4M1.CPL",
User	wadmin
Host	dev
Integrity Level	High
File	RUNDLL32.EXE
SHA1	2576C63F45FBE13DBDC619C39124FADE94E002D0 Analyze VirusTotal Score
Vendor	Microsoft Corporation
Application	Microsoft® Windows® Operating System
Parent Process ID	4172
Parent Process	C:\Windows\SysWOW64\rundll32.exe
Parent Command	"C:\Windows\system32\rundll32.exe" Shell32.dll,Control_RunDLL "C:\Users\wadmin\AppData\Local\Temp_d8c4M1.CPL",

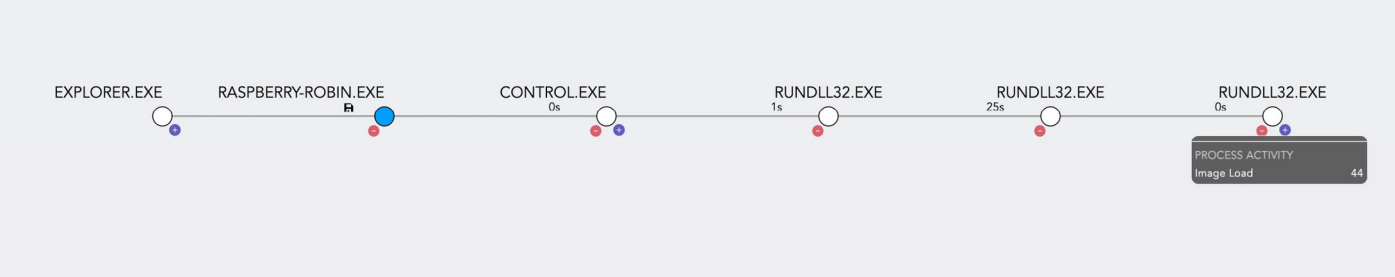
Image Loads (26)

search

S.N.	Status	File	SHA1	Vendor	Signature	Image	Is Signed
1	Valid	kernel32	2948B6BF7188304E62B7AE76E0C980911D…	Microsoft Corporation	Microsoft Windows	C:\Windows\System32\kernel32.dll	true
2	Valid	imm32	34C46FA33D7B582839E4C2FD9A80AC0C2…	Microsoft Corporation	Microsoft Windows	C:\Windows\System32\imm32.dll	true
3	Valid	sechost.dll	C7D5FAE3E827FB9C62CCF507720F4A750…	Microsoft Corporation	Microsoft Windows	C:\Windows\System32\sechost.dll	true
4	Valid	msvcp_w…	4C2A112E048FC75CD4C581836356E8815…	Microsoft Corporation	Microsoft Windows	C:\Windows\System32\msvcp_win…	true
5	Valid	Apphelp	27FB63606FC83372EB816D6EE225AB82A…	Microsoft Corporation	Microsoft Windows	C:\Windows\System32\apphelp.dll	true
6	Valid	gdi32	1977E858AB8423E3320BD56229ABC94AF…	Microsoft Corporation	Microsoft Windows	C:\Windows\System32\gdi32.dll	true
7	Valid	SHELL32…	2503E4A4E0863BA091EF4FDF7EB32DE18…	Microsoft Corporation	Microsoft Windows	C:\Windows\System32\shell32.dll	true
8	Valid	SHLWAP…	31EF5A400D452B967CCECD8D0EA7F8A2…	Microsoft Corporation	Microsoft Windows	C:\Windows\System32\shlwapi.dll	true
9	Valid	ucrtbase…	F156A272DBC6695CC170B6091EF8CD41D…	Microsoft Corporation	Microsoft Windows	C:\Windows\System32\ucrtbase.dll	true

5. rundll32.exe

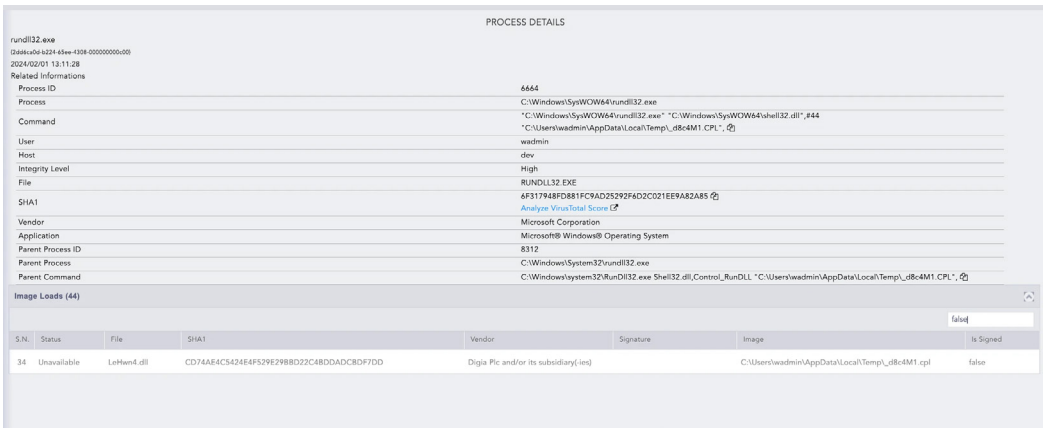
As its parent, it had no significant process activity, only some image-loading events.



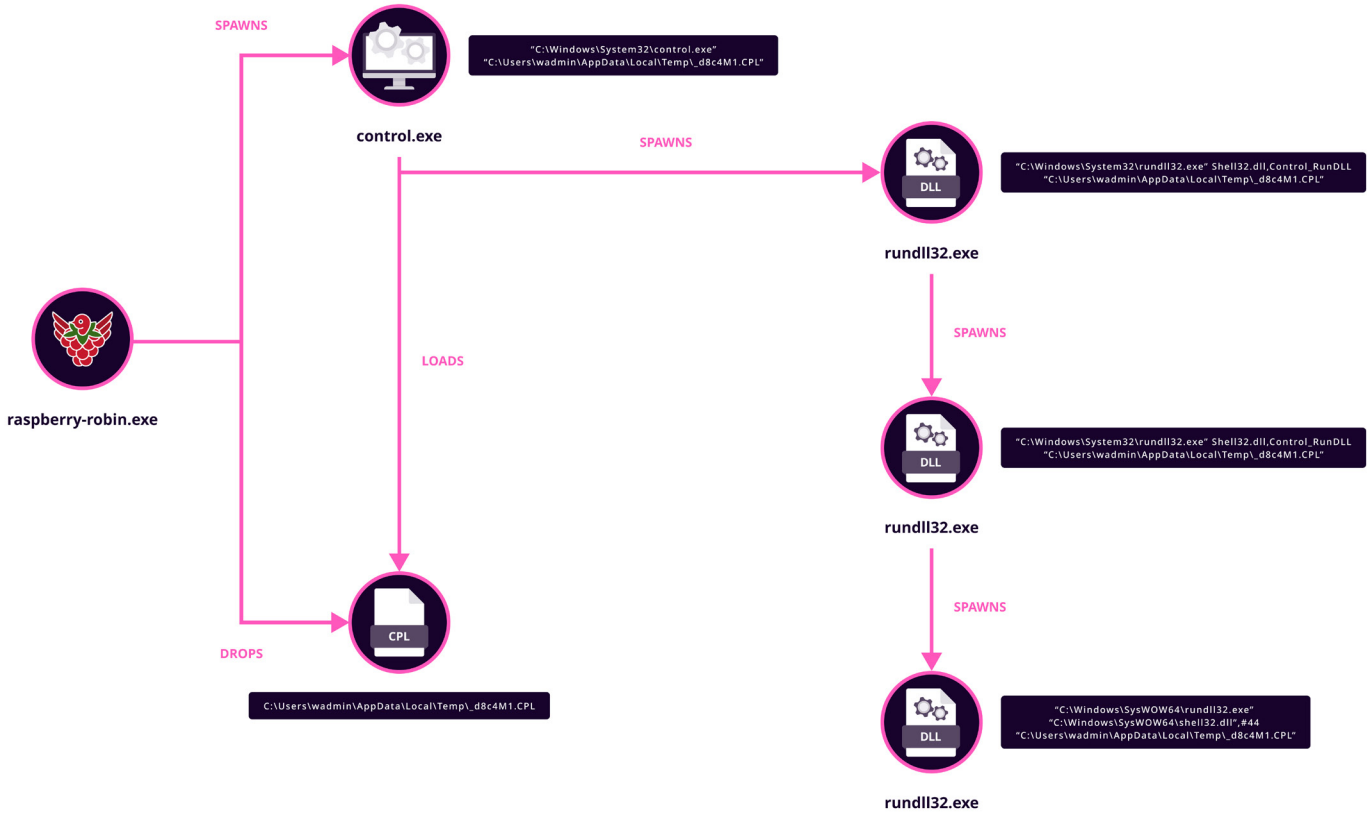
The command seemed to have been adjusted by as little as:

```
1  "C:\Windows\SysWOW64\rundll32.exe" "C:\Windows\SysWOW64\shell32.dll",#44 "C:\Users\
   wadmin\AppData\Local\Temp\_d8c4M1.CPL",
```

This command instructs the Windows operating system to execute a specific function within the shell32.dll file using the rundll32.exe. The function to be executed is identified by its ordinal number, #44, within the shell32.dll file. Additionally, the command specifies the path to a '.CPL' (Control Panel) file, '_d8c4M1.CPL', located in the temporary folder of the 'wadmin' user. This '.CPL' file is passed as an argument to the function within shell32.dll. Overall, this command is indicative of potential malicious activity, as it involves the execution of a function within a core Windows system file and the loading of a '.CPL' file from a temporary directory.



The breakdown of the entire execution flow of this variant is depicted in the flow chart provided below:



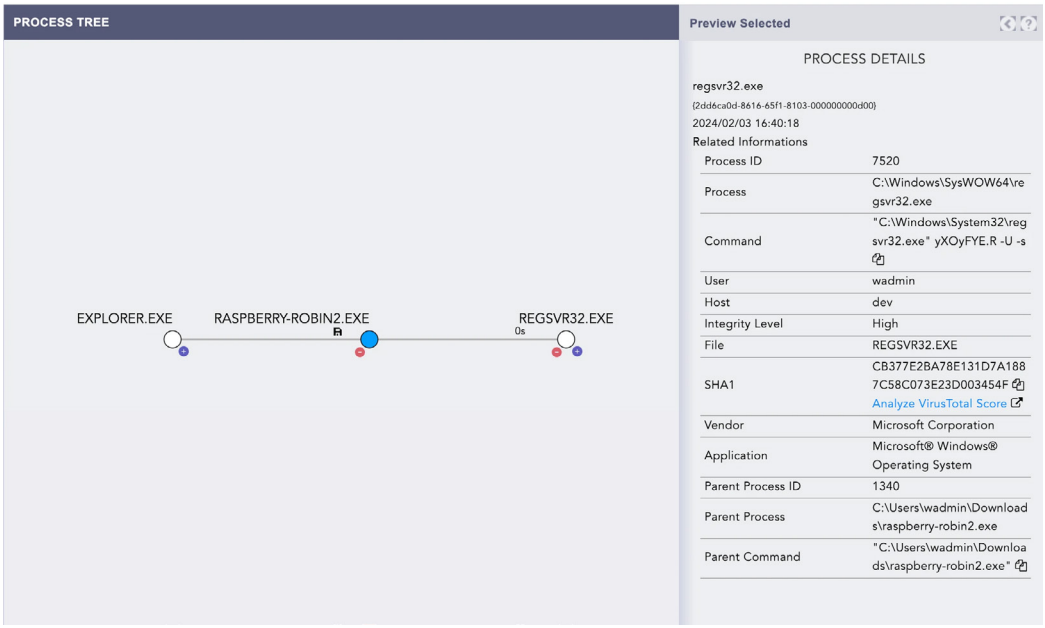
Interestingly, in another sample, we observed the same RarSFX installer, there was a certain change in command to be executed after the successful extraction of an executable file (i.e., RarSFX installer) as shown in the strings content of the executable below:

```
TExt=kRQUHyAHNpThsrCgr WuEkfjxbRuetuqOftuEQQDYSXhR:
jhLETkKFgHzftKgyQGt=dkVNMWRzqOYdkeJjNwHanbvWNUvItF
silEnt=3
Update=U
SETUP=regsvr32.exe    yXOyFYE.R -U   -s
upDaTE=u
$Of%
'L^"q
MRKII
  8lNW<2
LutuKQ=fiqAOVOZnxkPcLQoAapYWDtEi LAlNckNxYjtsuffDtI
```

The identical behavior was noted in dynamic analysis using the Logpoint Process Tree. The child process observed in this variant was regsvr32.exe, which registered the malicious DLL dropped by its parent process, the malware executable.

Command executed:

```
1  C:\Windows\System32\regsvr32.exe yXOyFYeR -U -s
```



DETECTION WITH LOGPOINT CONVERGED SIEM PLATFORM

Early discovery is critical for mitigating the possible consequences of Raspberry Robin's malicious activity. Organizations may use Logpoint's Converged SIEM platform, which includes powerful query capabilities, to identify and respond effectively. Security analysts may leverage this platform to develop targeted searches that detect critical Indicators of compromise and possible Raspberry Robin infections. We've compiled a collection of customized queries to help analysts track Raspberry Robin's nefarious activity on their network.

Log Sources Needed

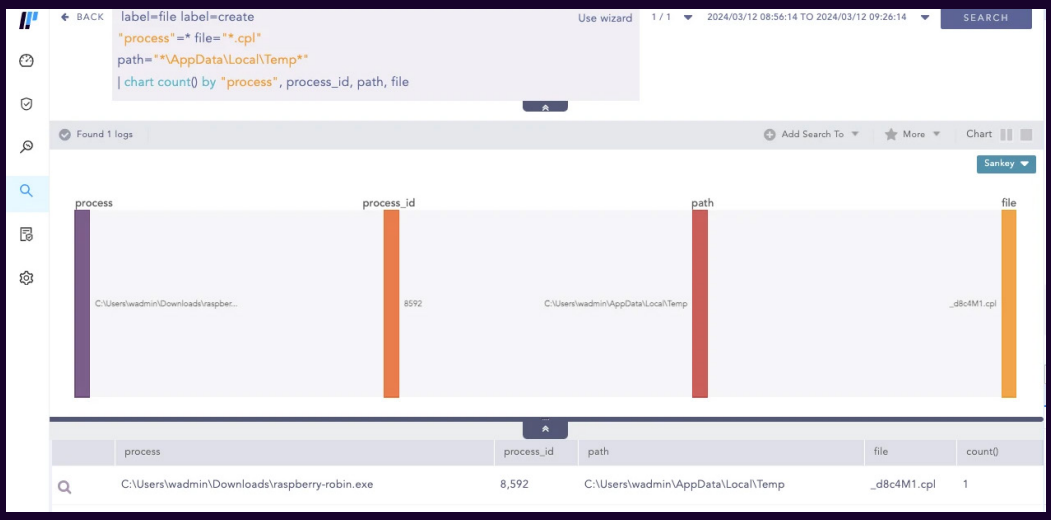
You must ensure you have the appropriate event logs from specified sources for the hunting queries to work. Some logs are logged by default, while others may need to be manually configured. The following log sources are required for effective detection.

1. Windows
    - Process Creation with command-line auditing should be enabled
2. Windows Sysmon

Investigation

The malware executable dropped few files upon execution. To detect the creation of a '.cpl' file in the temp directory, we can monitor for the presence of suspicious processes dropping such files using the following query:

```
1  label=file label=create
2  "process"=* file="*.cpl"
3  path="*\AppData\Local\Temp*"
```



Additionally, considering that threat actors often drop payloads in various writable directories, analysts can employ an extended query for comprehensive coverage:

```
label=file label=create
“process”=* file=”*.cpl”
path IN [“*\AppData\Local\Temp*”, “*\Windows\Temp*”, “*C:\Users\Public*”,
        “*C:\Users*\Downloads\*”, “*AppData\Local*”, “*AppData\Roaming*”]
```

The dynamic analysis detected various registry modification activities related to the system’s proxy configuration. The following query can be used to alert after detecting such registry modification activities.

```
norm_id=WindowsSysmon event_id=13
“process” IN [“*\AppData\Local\Temp\*”, “*\Downloads\*”, “*\Windows\Temp\*”, “*\Users\Public*”, “*AppData\Roaming*”]
target_object=”*\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap*”
(target_object IN [“*ProxyBypass”,”*IntranetName”, “*UNCAsIntranet”] detail=”DWORD (0x00000001)”)
OR
(target_object=”*AutoDetect” detail=”DWORD (0x00000000)”)
```

It focuses on processes originating from suspicious paths and suspicious processes targeting specific registry keys associated with proxy settings.

Rundll32.exe is a vital Microsoft Windows component that executes functions within DLL files. Despite its legitimate purpose, it’s frequently exploited by malware and threat actors due to its ability to load and execute code from DLLs. This executable, signed by Microsoft, is often used for “Living off the Land” attacks, where legitimate system tools are employed for malicious activities, making detection more challenging. As mentioned, rundll32 has been abused by a Raspberry-robin sample to execute a malicious ‘.cpl’ file through the **Control_RunDLL** function of Shell32.dll.

Shell32.dll is a Windows dynamic link library containing functions for managing the Windows Shell. Control_RunDLL is a function within Shell32.dll used to execute Control Panel items. It allows convenient access to Control Panel applets from the command line or programmatically.

The following query can be used to raise an alert after detecting such suspicious activity.

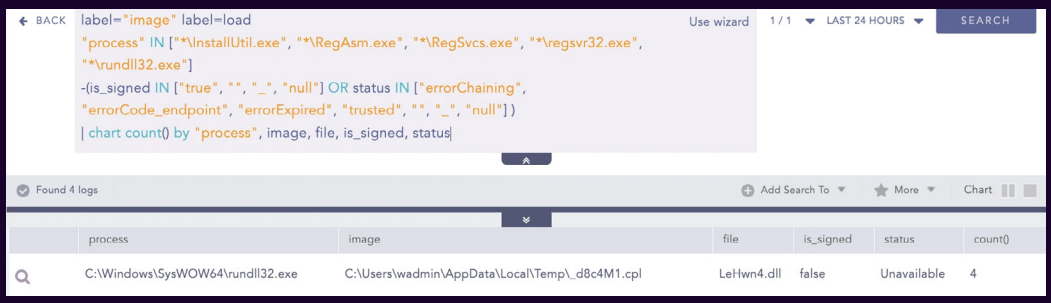
```
label="process" label=create
"process"="*\rundll32.exe"
command="*shell32.dll*" command="*Control_RunDLL*" command="*.CPL*"
command="*\AppData\Local\Temp*"
```

This query can be brittle as adversary often changes their TTPs, and they can drop their payload on other publicly writable folders. For better visibility, this modified query can be leveraged.

```
label="process" label=create
"process"="*\rundll32.exe"
command="*shell32.dll*" command="*Control_RunDLL*" command="*.CPL*"
command IN ["*\AppData\Local\Temp*", "*\Windows\Temp*", "*C:\Users\Public*",
        "*C:\Users*\Downloads\*", "*AppData\Local*", "*AppData\Roaming*"]
```

DLLs are commonly exploited in malicious attacks for injection, hijacking, or side-loading, allowing attackers to execute arbitrary code or evade detection by abusing legitimate system components. The DLL needs to be loaded by some program for its execution. Often, these malicious DLLs are unsigned. Therefore, searching for indicators of Windows utilities like **rundll32** , **regsvr32** , etc., and loading unsigned DLLs can be a valuable detection technique for defenders to identify potentially malicious activity within their enterprise systems.

```
label="image" label=load
"process" IN ["*\InstallUtil.exe", "*\RegAsm.exe", "*\RegSvcs.exe", "*\regsvr32.exe", "*\rundll32
.exe"]
-(is_signed IN ["true", "", "_", "null"] OR status IN ["errorChaining", "errorCode_endpoint",
"errorExpired", "trusted", "", "_", "null"] )
```



In one variant, we observed regsvr32.exe registering a malicious DLL with an uncommon DLL extension. We can use this sigma rule to hunt for such behaviors.

```
label="process" label=create command=*
-(command" IN ["*.ax*", "*.cpl*", "*.dll*", "*.ocx*", "*.ppl*", "*.bav*", "*null*"])
```

It might also be a good idea to check if tools like rundll32.exe and regsvr32.exe are making network connections without empty command line parameters.

```
[norm_id=WindowsSysmon event_id=1
“process” IN [“*\rundll32.exe”, “*\regsvr32.exe”]
-command=*] as s1
followed by
[norm_id=WindowsSysmon event_id=3
“process” IN [“*\rundll32.exe”, “*\regsvr32.exe”]] as s2 within 5 minute
on s1.process_guid=s2.process_guid and s1.user=s2.user
```

These queries, while valuable, are not exhaustive for effective threat hunting and alerting. Below are additional relevant alerts that can be further useful for investigation purposes.

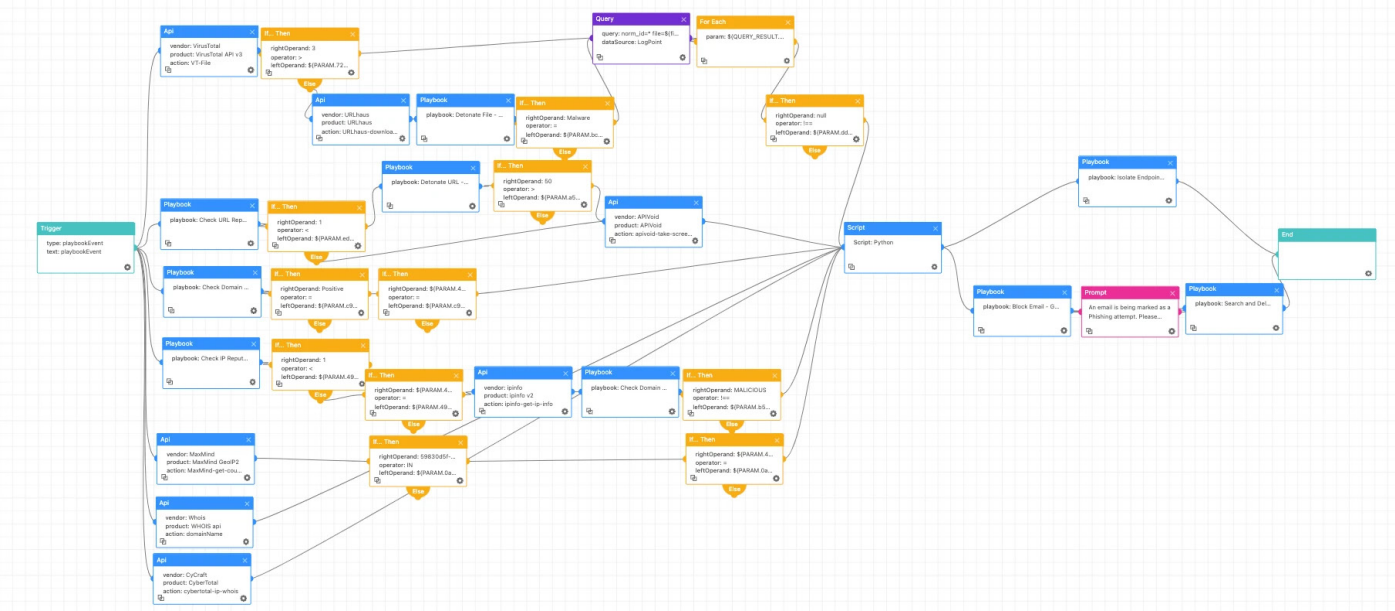
1. LP_Windows Shell Spawning Suspicious Program
2. LP_Rundll32 Internet Connection Detected
3. LP_Suspicious Control Panel DLL Load Detected
4. LP_Suspicious Rundll32 Activity Detected
5. LP_Suspicious Process Execution Without DLL
6. LP_Unsigned DLLs loaded by RunDLL32 or RegSvr32

INVESTIGATION AND RESPONSE WITH LOGPOINT CONVERGED SIEM

Logpoint Converged SIEM is pre-loaded with SOAR features, including several playbooks for streamlining and automating incident response and investigation operations. These playbooks cover a wide range of real-time use cases for forensic investigation and remediation, increasing efficiency and effectiveness in security incident management. With Agentx, Logpoint New Agent with EDR capabilities bolstered with SOAR, proactive detection and remediation are now easier and faster than ever.

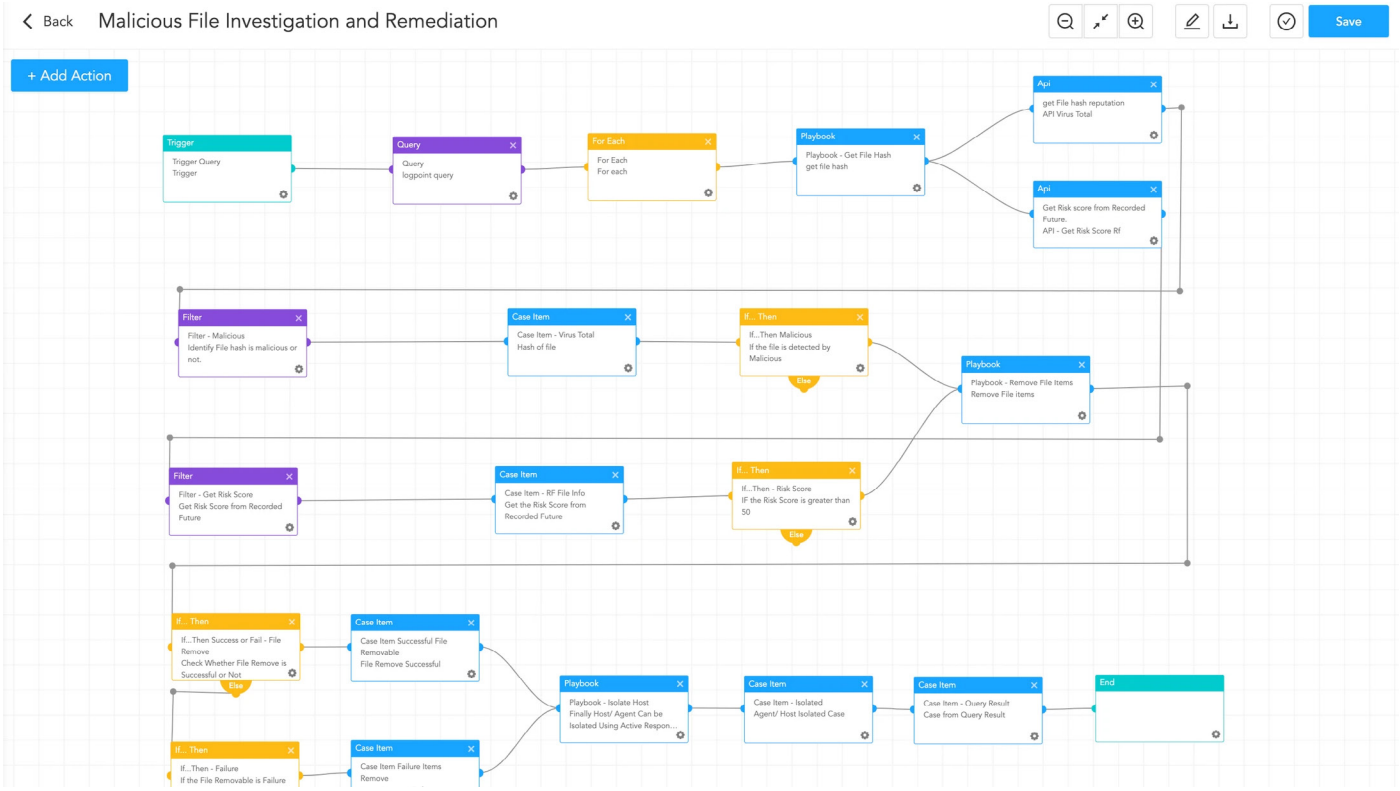
Phishing Investigation and Response

Social engineering, especially phishing, is prevalent among threats like Raspberry Robin. Considering the widespread use of phishing as a primary attack vector, this playbook ensures that all suspicious phishing incidents are thoroughly investigated and promptly addressed, significantly reducing response time and minimizing human error.

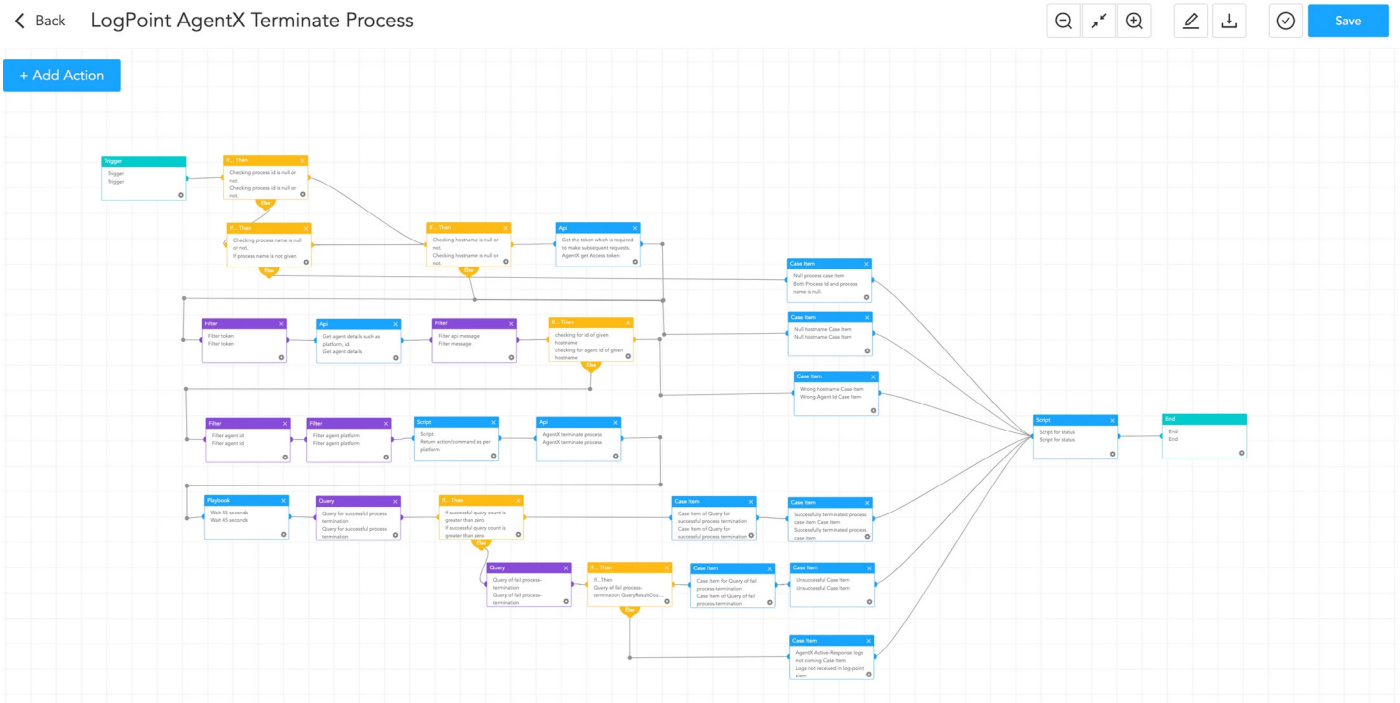


Malicious File Investigation and Remediation

Many malware delivery campaigns utilize weaponized attachments and sophisticated social engineering techniques to trick victims into executing them. Additionally, contemporary attacks often employ multi-staged tactics for payload delivery. This playbook focuses on investigating and containing malicious binaries dropped on the system. It verifies the hash of the dumped file against threat intelligence sources, and if identified as dangerous, it terminates the associated processes and removes the file accordingly.



Furthermore, this playbook searches the identified hash across other endpoints to pinpoint potentially infected machines. The playbook outlines precise steps to address the situation if such machines are discovered. To execute these activities seamlessly, the playbook leverages the functionality of the “AgentX Terminate Process” and “AgentX Remove Item” playbooks. This integration empowers analysts to efficiently terminate malicious processes and eradicate malicious files from infected machines.



SECURITY BEST PRACTICES RECOMMENDATIONS

The recommendations for organizations to avoid the infection of Pikabot include:

1. **Patch and Update Regularly:**
   - Ensure that your operating systems and applications receive security patches and updates regularly. This proactive method greatly reduces the number of software vulnerabilities that the Raspberry Robin malware may exploit.
   - Regularly assess vulnerabilities for their potential impact and exploitability. Consider the impacted system, the risk of exploitation, and the probable repercussions. Give top priority to vulnerabilities with high severity ratings. These represent the greatest risk and must be handled immediately.
2. **Implement Security Awareness Training:**
   - Conduct frequent security awareness training programs to teach users/employees how to detect phishing, malicious activities, possible ransomware, malware attacks, and related risks.
   - Encourage a security-conscious culture and provide training on detecting and reporting suspicious conduct.
3. **Monitor Network Traffic:**
   - Consistently monitor network traffic for any signs of unusual activity and unexpected spikes in requests.
   - To stay ahead of evolving threats, it's essential to assess and update firewall rules periodically.
4. **Network Segmentation:**
   - Enhance network security by employing segmentation to isolate critical systems from less secure areas
   - Enforce strict communication restrictions between segments to mitigate potential threats.
5. **Use Cybersecurity Solutions:**
   - Install cybersecurity solutions like firewalls, intrusion detection systems, and DDoS protection tools to prevent unauthorized visits and detect botnet activities.
   - An Endpoint Protection Platform for host-level security is also required.
6. **Regular Backups:**
   - Regularly backup critical data, ensuring backups are secure, offline, and thoroughly tested for reliable restoration.
   - This practice safeguards against data loss and facilitates swift recovery in the event of a ransomware attack.
7. **Incident Response Plan:**
   - Develop and continue to implement an incident response plan to handle security incidents swiftly and effectively.
   - Conduct simulations and exercises regularly to test the incident response plan.
8. **Regular Audits and Compliance Adherence:**
   - Regularly conduct security audits, including penetration tests and vulnerability assessments, to proactively identify and address vulnerabilities and weaknesses within the network and systems. This helps fortify defenses and mitigate potential security risks.
   - Ensure compliance with relevant data protection and cybersecurity standards while staying abreast of evolving laws and regulations. This enables adjustment of security measures to maintain robust protection and alignment with legal requirements.

CONCLUSION

As the Raspberry Robin infection rate rises due to threat actor affiliates deploying additional payloads such as ransomware, crypto-miners, and data exfiltration tools, traditional security tools struggle to effectively detect and mitigate this multifaceted malware. In this research, we attempted to shed light on Raspberry Robin's capabilities and behavioral patterns, underlining the significance of using sophisticated detection and remediation solutions as the Logpoint Converged SIEM platform.

The Logpoint Converged SIEM platform provides increased Endpoint Detection and Response (EDR) capabilities via its native agent, AgentX, which simplifies the transmission of logs and telemetry from endpoints to the SIEM. Converged SIEM, powered by Security Orchestration, Automation, and Response (SOAR) capabilities, provides automated threat analysis and remediation, allowing enterprises to quickly identify, investigate, and respond to a wide range of cyber threats, including Raspberry Robin.

Converged SIEM reduces manual procedures and allows for quick reactions to threats like Raspberry Robin by leveraging out-of-the-box alerts, playbooks, threat information, orchestration, and automated actions. In an ever-evolving threat landscape, enterprises must deploy advanced security operations platforms to defend against complex cyber attacks and secure digital assets proactively.
In conclusion, the research underscores the significance of leveraging cutting-edge technologies like Converged SIEM to bolster cybersecurity posture and effectively combat the pervasive threat posed by Raspberry Robin and its counterparts in the cyber realm.

ABOUT LOGPOINT

Logpoint is the creator of a reliable, innovatie cybersecurity operations platform - empowering organisations worldwide to thrive in a world of evolving threats.

By combining sophisticated technology and a profound understaning of customer challenges, Logpoint bolsters security teams' capabilities while helping them combat current and future threats.

Logpoint offers SIEM, UEBA, and SOAR technologies in a complete platform that efficiently detects threats, minimizes false positives, autonomously prioritizes risks, responds to incidents, and much more.

Headquartered in Copenhagen, Denmark, with offices around the world, Logpoint is a multinational, multicultural, and inclusive company.

For more information visit www.logpoint.com

www.logpoint.com