KROLL

How to use KAPE and SQLECmd with EventTranscript.db

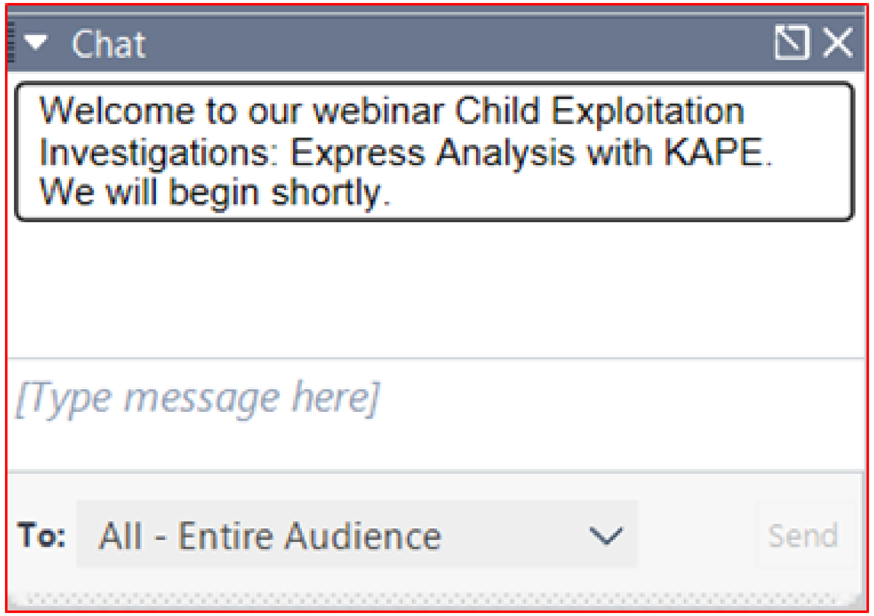
How to Use KAPE and SQLECmd with EventTranscript.db

By: Andrew Rathbun and Josh Mitchell

September 21, 2021

Notes

- Session is being recorded, You'll receive access to the recording in a couple days
- Ask questions via chat >
- We'll try to answer as many questions as possible



Welcome to our webinar Child Exploitation Investigations: Express Analysis with KAPE. We will begin shortly.

Upcoming KAPE Intensive Training and Certification Sessions

- Virtual live sessions
- Max 25 students

Full Calendar Available here:

bit.ly/KAPE2021

SCHEDULE	INSTRUCTORS
September 28, 2021 10:00 a.m. - 7:00 p.m. ET	Eric Zimmerman Sean Straw Scott Zuberbuehler Andrew Rathbun
October 7, 2021 8:00 a.m. - 5:00 p.m. GMT	James Thoburn Paul Wells Guillermo Roman
October 20, 2021 9:00 a.m. - 6:00 p.m. HKT	Paul Jackson David Klopp Rob Phillips

Table of Contents

1. EventTranscript.db Introduction
2. DiagTrack Service Overview
3. Data Collection Control Mechanisms
4. Introduction to SQLECmd
5. Introduction to KAPE
6. Automating SQLECmd with KAPE
7. Questions

Introduction

Who are we?

- Andrew Rathbun
  - Senior Associate, Kroll Cyber Risk
  - KAPE Instructor
  - Former Federal LE (HHS OIG)
  - Former Local LE (MSUPD)
  - Former US Military (USMC – 0311)
  - Digital Forensics Discord Server Administrator
  - AboutDFIR Contributor
  - GitHub Enthusiast

- Josh Mitchell
  - Senior Vice President, Kroll Cyber Risk
  - Software Reverse Engineering
  - Malware Analysis
  - Former US Military (USAF – 1N5)
  - Background in Vulnerability Discovery and Exploit Development

EventTranscript.db Introduction

EventTranscript.db

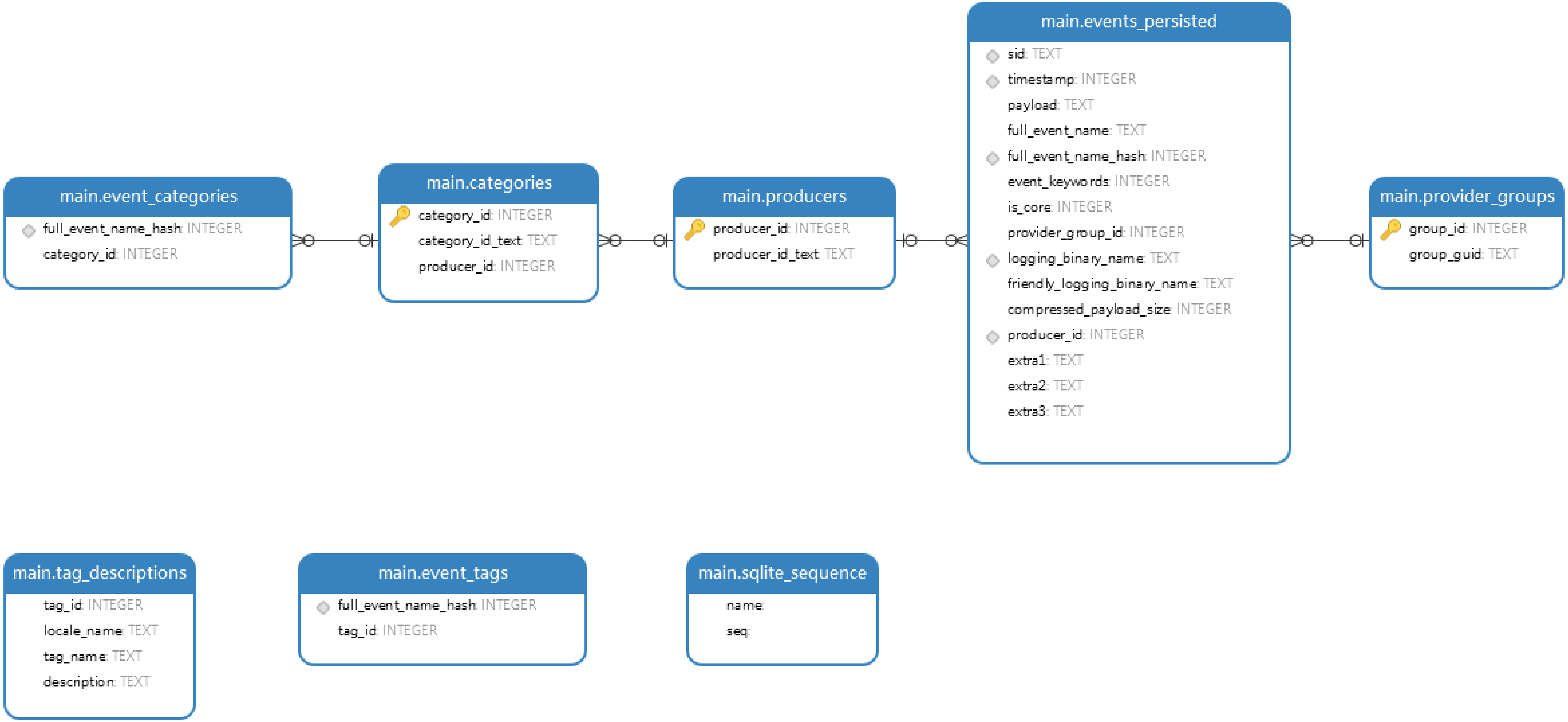
An Introduction to a New DFIR Artifact

Forensically Unpacking EventTranscript.db: An Investigative Series

View full research at: kroll.com/eventtranscript

- Has existed since Windows 10 version 1709 and exists through most current build of Windows 11 (KB5006050)
- Relates to telemetry and diagnostic data tracking
- Multiple levels of diagnostic data tracking
- Plenty of documentation exists for Diagnostic Data and Telemetry, but nothing exists prior to our research about EventTranscript.db as a forensic artifact
- DiagTrack.dll controls the recording of Diagnostic Data events to EventTranscript.db

EventTranscript.db Schema



main.event_categories
full_event_name_hash: INTEGER
category_id: INTEGER

main.categories
category_id: INTEGER
category_id_text: TEXT
producer_id: INTEGER

main.producers
producer_id: INTEGER
producer_id_text: TEXT

main.events_persisted
sid: TEXT
timestamp: INTEGER
payload: TEXT
full_event_name: TEXT
full_event_name_hash: INTEGER
event_keywords: INTEGER
is_core: INTEGER
provider_group_id: INTEGER
logging_binary_name: TEXT
friendly_logging_binary_name: TEXT
compressed_payload_size: INTEGER
producer_id: INTEGER
extra1: TEXT
extra2: TEXT
extra3: TEXT

main.provider_groups
group_id: INTEGER
group_guid: TEXT

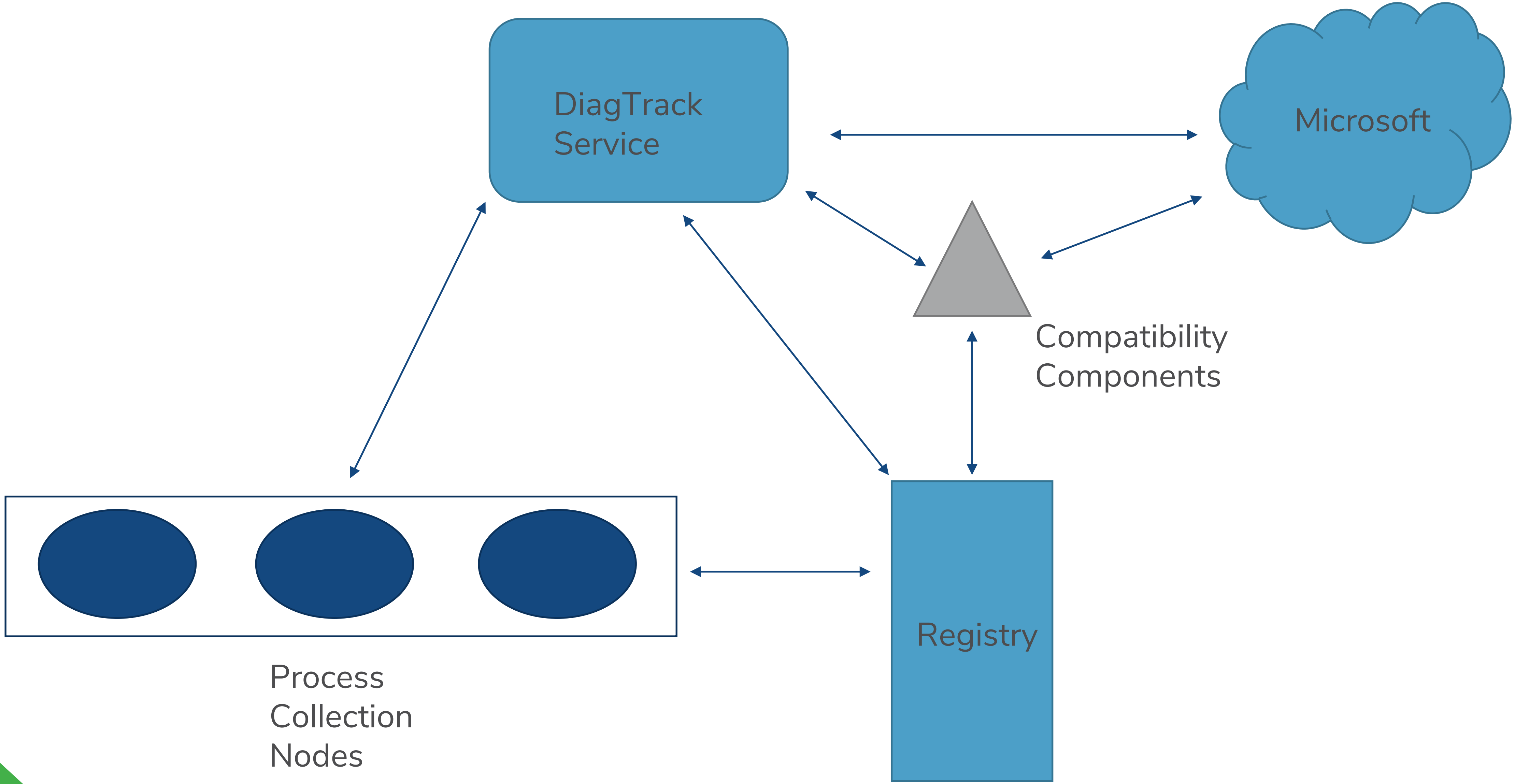
main.tag_descriptions
tag_id: INTEGER
locale_name: TEXT
tag_name: TEXT
description: TEXT

main.event_tags
full_event_name_hash: INTEGER
tag_id: INTEGER

main.sqlite_sequence
name:
seq:

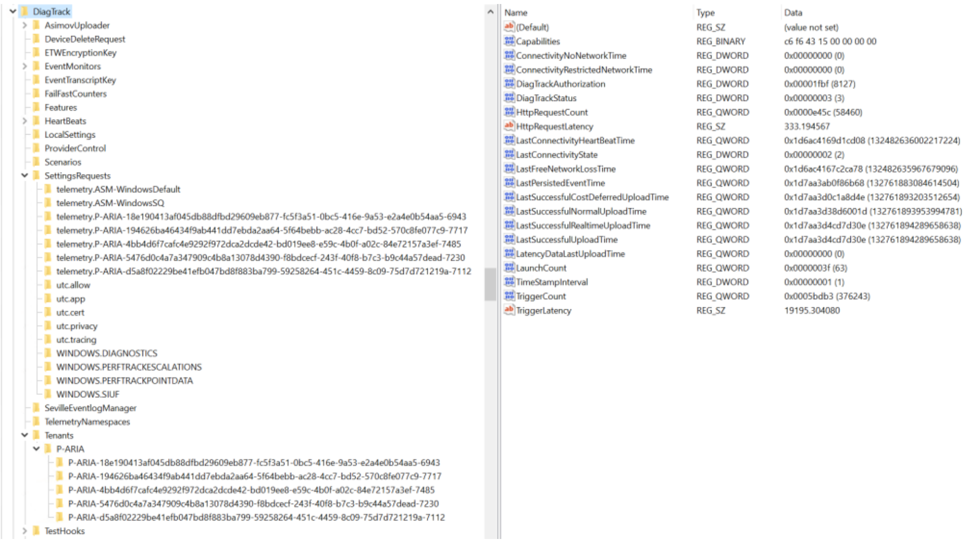
DiagTrack Service Overview

Diagnostics Overview



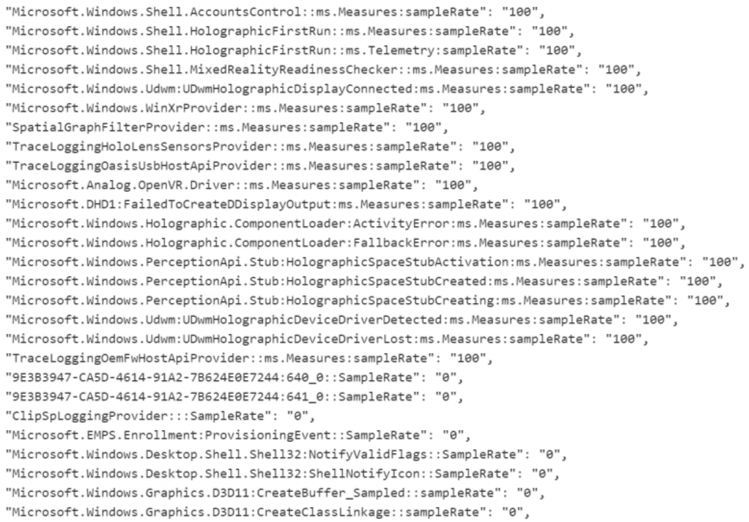
DiagTrack Service Registry

- Components listed under HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Diagnostics
- Tenants list enabled telemetry collection packages
- SettingsRequests include settings for the collection packages (or where to get the settings)
- JSON and XML files appear to define the collected items (data sampling)

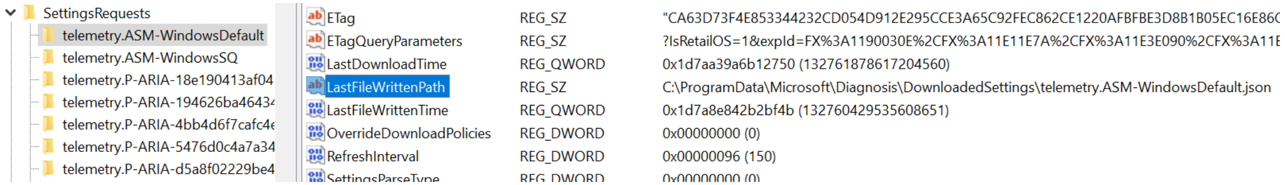


```
DiagTrack
  AsimovUploader
  DeviceDeleteRequest
  ETWEncryptionKey
  EventMonitors
  EventTranscriptKey
  FailFastCounters
  Features
  HeartBeats
  LocalSettings
  ProviderControl
  Scenarios
  SettingsRequests
    telemetry.ASM-WindowsDefault
    telemetry.ASM-WindowsSQ
    telemetry.P-ARIA-18e190413af045db88dfbd29609eb877-fc5f3a51-0bc5-416e-9a53-e2a4e0b54aa5-6943
    telemetry.P-ARIA-194626ba46434f9ab441dd7ebda2aa64-5f64bebb-ac28-4cc7-bd52-570c8fe077c9-7717
    telemetry.P-ARIA-4bb4d6f7cafc4e9292f972dca2dcde42-bd019ee8-e59c-4b0f-a02c-84e72157a3ef-7485
    telemetry.P-ARIA-5476d0c4a7a347909c4b8a13078d4390-f8bdcecf-243f-40f8-b7c3-b9c44a57dead-7230
    telemetry.P-ARIA-d5a8f02229be41efb047bd8f883ba799-59258264-451c-4459-8c09-75d7d721219a-7112
    utc.allow
    utc.app
    utc.cert
    utc.privacy
    utc.tracing
    WINDOWS.DIAGNOSTICS
    WINDOWS.PERFTRACKESCALATIONS
    WINDOWS.PERFTRACKPOINTDATA
    WINDOWS.SIUF
  SevilleEventlogManager
  TelemetryNamespaces
  Tenants
    P-ARIA
      P-ARIA-18e190413af045db88dfbd29609eb877-fc5f3a51-0bc5-416e-9a53-e2a4e0b54aa5-6943
      P-ARIA-194626ba46434f9ab441dd7ebda2aa64-5f64bebb-ac28-4cc7-bd52-570c8fe077c9-7717
      P-ARIA-4bb4d6f7cafc4e9292f972dca2dcde42-bd019ee8-e59c-4b0f-a02c-84e72157a3ef-7485
      P-ARIA-5476d0c4a7a347909c4b8a13078d4390-f8bdcecf-243f-40f8-b7c3-b9c44a57dead-7230
      P-ARIA-d5a8f02229be41efb047bd8f883ba799-59258264-451c-4459-8c09-75d7d721219a-7112
  TestHooks
```

Name	Type	Data
(Default)	REG_SZ	(value not set)
Capabilities	REG_BINARY	c6 f6 43 15 00 00 00 00
ConnectivityNoNetworkTime	REG_DWORD	0x00000000 (0)
ConnectivityRestrictedNetworkTime	REG_DWORD	0x00000000 (0)
DiagTrackAuthorization	REG_DWORD	0x00001fbf (8127)
DiagTrackStatus	REG_DWORD	0x00000003 (3)
HttpRequestCount	REG_QWORD	0x0000e45c (58460)
HttpRequestLatency	REG_SZ	333.194567
LastConnectivityHeartBeatTime	REG_QWORD	0x1d6ac4169d1cd08 (132482636002217224)
LastConnectivityState	REG_DWORD	0x00000002 (2)
LastFreeNetworkLossTime	REG_QWORD	0x1d6ac4167c2ca78 (132482635967679096)
LastPersistedEventTime	REG_QWORD	0x1d7aa3ab0f86b68 (132761883084614504)
LastSuccessfulCostDeferredUploadTime	REG_QWORD	0x1d7aa3d0c1a8d4e (132761893203512654)
LastSuccessfulNormalUploadTime	REG_QWORD	0x1d7aa3d38d6001d (132761893953994781)
LastSuccessfulRealtimeUploadTime	REG_QWORD	0x1d7aa3d4cd7d30e (132761894289658638)
LastSuccessfulUploadTime	REG_QWORD	0x1d7aa3d4cd7d30e (132761894289658638)
LatencyDataLastUploadTime	REG_QWORD	0x00000000 (0)
LaunchCount	REG_QWORD	0x0000003f (63)
TimeStampInterval	REG_DWORD	0x00000001 (1)
TriggerCount	REG_QWORD	0x0005bdb3 (376243)
TriggerLatency	REG_SZ	19195.304080



```
"Microsoft.Windows.Shell.AccountsControl::ms.Measures:sampleRate": "100",
"Microsoft.Windows.Shell.HolographicFirstRun::ms.Measures:sampleRate": "100",
"Microsoft.Windows.Shell.HolographicFirstRun::ms.Telemetry:sampleRate": "100",
"Microsoft.Windows.Shell.MixedRealityReadinessChecker::ms.Measures:sampleRate": "100",
"Microsoft.Windows.Udwm:UDwmHolographicDisplayConnected:ms.Measures:sampleRate": "100",
"Microsoft.Windows.WinXrProvider::ms.Measures:sampleRate": "100",
"SpatialGraphFilterProvider::ms.Measures:sampleRate": "100",
"TraceLoggingHoloLensSensorsProvider::ms.Measures:sampleRate": "100",
"TraceLoggingOasisUsbHostApiProvider::ms.Measures:sampleRate": "100",
"Microsoft.Analog.OpenVR.Driver::ms.Measures:sampleRate": "100",
"Microsoft.DHD1:FailedToCreateDDisplayOutput:ms.Measures:sampleRate": "100",
"Microsoft.Windows.Holographic.ComponentLoader:ActivityError:ms.Measures:sampleRate": "100",
"Microsoft.Windows.Holographic.ComponentLoader:FallbackError:ms.Measures:sampleRate": "100",
"Microsoft.Windows.PerceptionApi.Stub:HolographicSpaceStubActivation:ms.Measures:sampleRate": "100",
"Microsoft.Windows.PerceptionApi.Stub:HolographicSpaceStubCreated:ms.Measures:sampleRate": "100",
"Microsoft.Windows.PerceptionApi.Stub:HolographicSpaceStubCreating:ms.Measures:sampleRate": "100",
"Microsoft.Windows.Udwm:UDwmHolographicDeviceDriverDetected:ms.Measures:sampleRate": "100",
"Microsoft.Windows.Udwm:UDwmHolographicDeviceDriverLost:ms.Measures:sampleRate": "100",
"TraceLoggingOemFwHostApiProvider::ms.Measures:sampleRate": "100",
"9E3B3947-CA5D-4614-91A2-7B624E0E7244:640_0::SampleRate": "0",
"9E3B3947-CA5D-4614-91A2-7B624E0E7244:641_0::SampleRate": "0",
"ClipSpLoggingProvider:::SampleRate": "0",
"Microsoft.EMPS.Enrollment:ProvisioningEvent::SampleRate": "0",
"Microsoft.Windows.Desktop.Shell.Shell32:NotifyValidFlags::SampleRate": "0",
"Microsoft.Windows.Desktop.Shell.Shell32:ShellNotifyIcon::SampleRate": "0",
"Microsoft.Windows.Graphics.D3D11:CreateBuffer_Sampled::sampleRate": "0",
"Microsoft.Windows.Graphics.D3D11:CreateClassLinkage::sampleRate": "0",
```



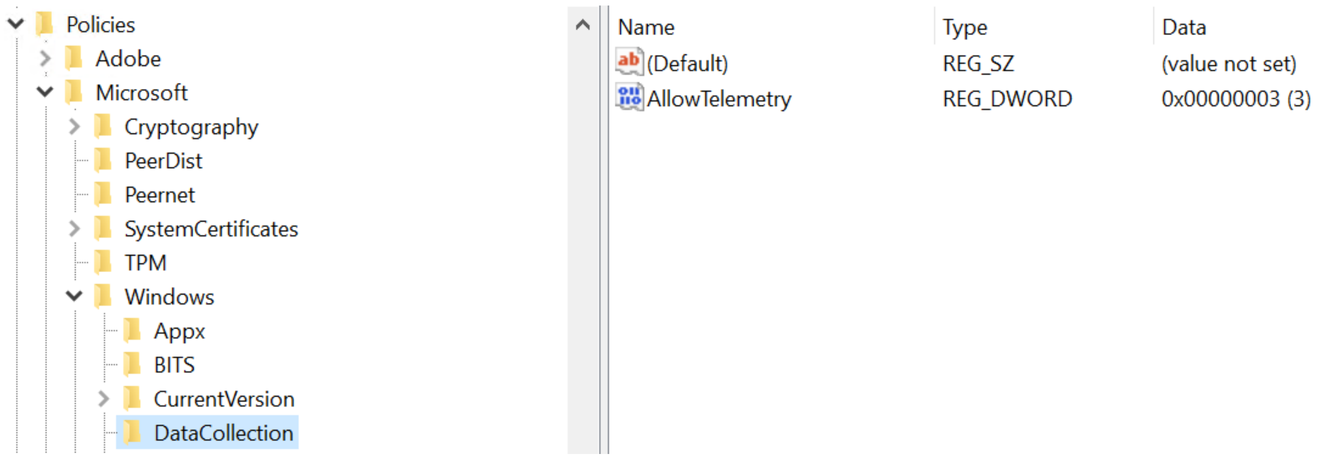
```
SettingsRequests
  telemetry.ASM-WindowsDefault
  telemetry.ASM-WindowsSQ
  telemetry.P-ARIA-18e190413af04
  telemetry.P-ARIA-194626ba4643
  telemetry.P-ARIA-4bb4d6f7cafc4
  telemetry.P-ARIA-5476d0c4a7a34
  telemetry.P-ARIA-d5a8f02229be4
```

ETag	REG_SZ	"CA63D73F4E853344232CD054D912E295CCE3A65C92FEC862CE1220AFBFBE3D8B1B05EC16E86
ETagQueryParameters	REG_SZ	?IsRetailOS=1&expId=FX%3A1190030E%2CFX%3A11E11E7A%2CFX%3A11E3E090%2CFX%3A11E
LastDownloadTime	REG_QWORD	0x1d7aa39a6b12750 (132761878617204560)
LastFileWrittenPath	REG_SZ	C:\ProgramData\Microsoft\Diagnosis\DownloadedSettings\telemetry.ASM-WindowsDefault.json
LastFileWrittenTime	REG_QWORD	0x1d7a8e842b2bf4b (132760429535608651)
OverrideDownloadPolicies	REG_DWORD	0x00000000 (0)
RefreshInterval	REG_DWORD	0x00000096 (150)
SettingsParseType	REG_DWORD	0x00000000 (0)

Data Collection Control Mechanisms

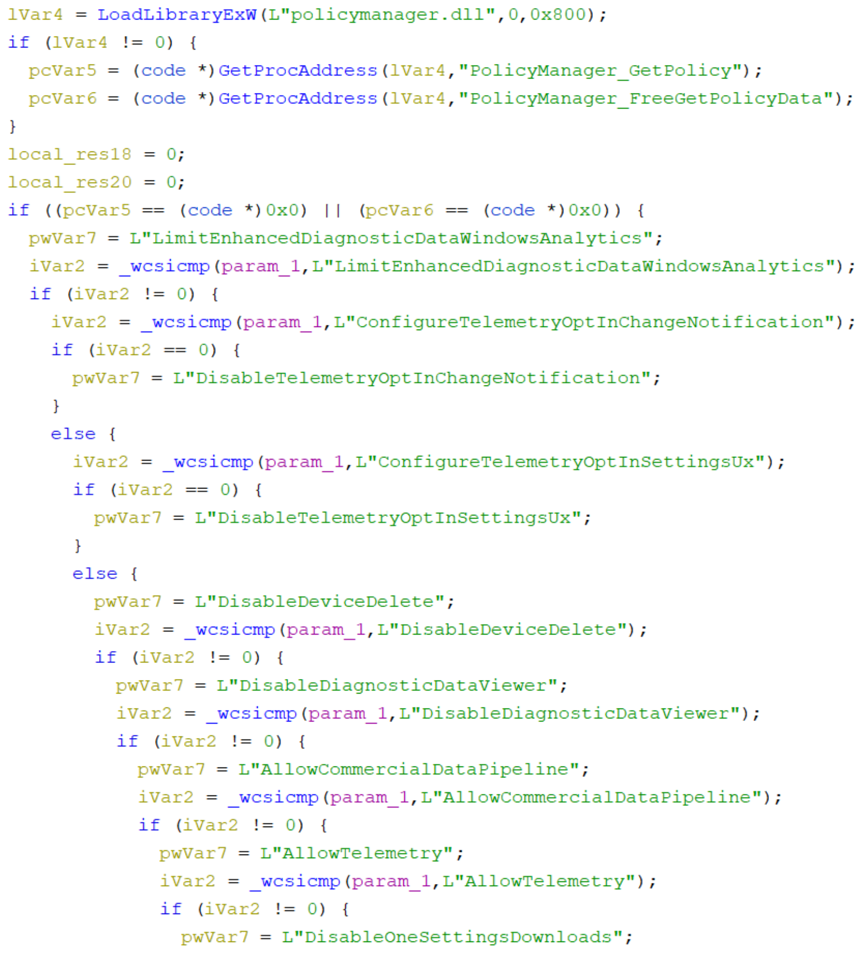
Data Collection Control Mechanisms

- From the DiagTrack service FlightSettings.dll used policymanager.dll and HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\Data Collection appear to dictate collection
- From Edge Windows.System.Profile.PlatformDiagnosticsAndUsageDataSettings.dll performs a similar function



```
Policies
  Adobe
  Microsoft
    Cryptography
    PeerDist
    Peernet
    SystemCertificates
    TPM
    Windows
      Appx
      BITS
      CurrentVersion
      DataCollection
```

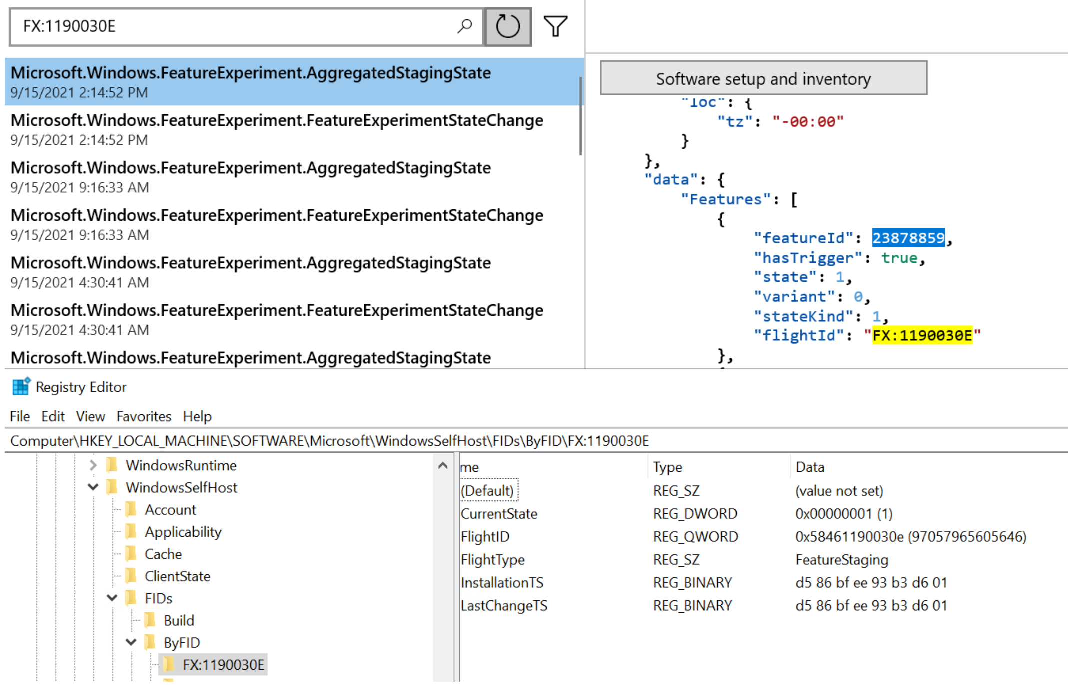
Name	Type	Data
(Default)	REG_SZ	(value not set)
AllowTelemetry	REG_DWORD	0x00000003 (3)



```
lVar4 = LoadLibraryExW(L"policymanager.dll",0,0x800);
if (lVar4 != 0) {
  pcVar5 = (code *)GetProcAddress(lVar4,"PolicyManager_GetPolicy");
  pcVar6 = (code *)GetProcAddress(lVar4,"PolicyManager_FreeGetPolicyData");
}
local_res18 = 0;
local_res20 = 0;
if ((pcVar5 == (code *)0x0) || (pcVar6 == (code *)0x0)) {
  pwVar7 = L"LimitEnhancedDiagnosticDataWindowsAnalytics";
  iVar2 = _wcsicmp(param_1,L"LimitEnhancedDiagnosticDataWindowsAnalytics");
  if (iVar2 != 0) {
    iVar2 = _wcsicmp(param_1,L"ConfigureTelemetryOptInChangeNotification");
    if (iVar2 == 0) {
      pwVar7 = L"DisableTelemetryOptInChangeNotification";
    }
    else {
      iVar2 = _wcsicmp(param_1,L"ConfigureTelemetryOptInSettingsUx");
      if (iVar2 == 0) {
        pwVar7 = L"DisableTelemetryOptInSettingsUx";
      }
      else {
        pwVar7 = L"DisableDeviceDelete";
        iVar2 = _wcsicmp(param_1,L"DisableDeviceDelete");
        if (iVar2 != 0) {
          pwVar7 = L"DisableDiagnosticDataViewer";
          iVar2 = _wcsicmp(param_1,L"DisableDiagnosticDataViewer");
          if (iVar2 != 0) {
            pwVar7 = L"AllowCommercialDataPipeline";
            iVar2 = _wcsicmp(param_1,L"AllowCommercialDataPipeline");
            if (iVar2 != 0) {
              pwVar7 = L"AllowTelemetry";
              iVar2 = _wcsicmp(param_1,L"AllowTelemetry");
              if (iVar2 != 0) {
                pwVar7 = L"DisableOneSettingsDownloads";
```

Data Collection Control Mechanisms Cont.

- FlightSettings.dll has additional Registry keys at HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows SelfHost which appear related to update mechanisms/experimentation



Microsoft.Windows.FeatureExperiment.AggregatedStagingState
9/15/2021 2:14:52 PM

Microsoft.Windows.FeatureExperiment.FeatureExperimentStateChange
9/15/2021 2:14:52 PM

Microsoft.Windows.FeatureExperiment.AggregatedStagingState
9/15/2021 9:16:33 AM

Microsoft.Windows.FeatureExperiment.FeatureExperimentStateChange
9/15/2021 9:16:33 AM

Microsoft.Windows.FeatureExperiment.AggregatedStagingState
9/15/2021 4:30:41 AM

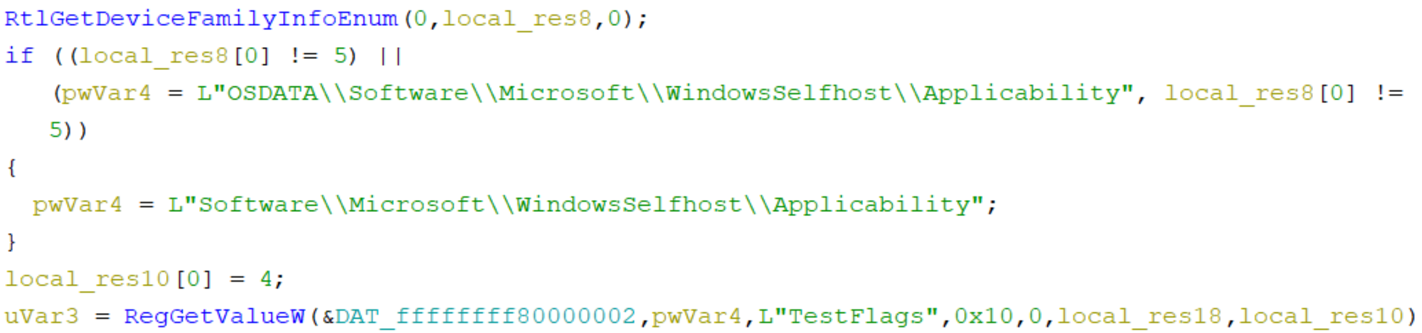
Microsoft.Windows.FeatureExperiment.FeatureExperimentStateChange
9/15/2021 4:30:41 AM

Microsoft.Windows.FeatureExperiment.AggregatedStagingState

```
        "loc": {
            "tz": "-00:00"
        }
    },
    "data": {
        "Features": [
            {
                "featureId": 23878859,
                "hasTrigger": true,
                "state": 1,
                "variant": 0,
                "stateKind": 1,
                "flightId": "FX:1190030E"
            },
```

Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\WindowsSelfHost\FIDs\ByFID\FX:1190030E

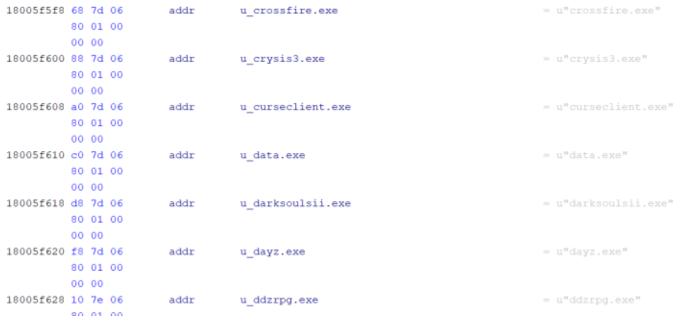
me	Type	Data
(Default)	REG_SZ	(value not set)
CurrentState	REG_DWORD	0x00000001 (1)
FlightID	REG_QWORD	0x58461190030e (97057965605646)
FlightType	REG_SZ	FeatureStaging
InstallationTS	REG_BINARY	d5 86 bf ee 93 b3 d6 01
LastChangeTS	REG_BINARY	d5 86 bf ee 93 b3 d6 01



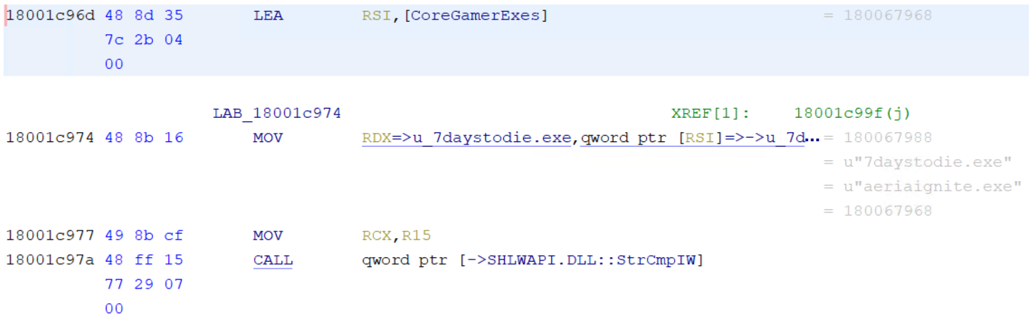
```
RtlGetDeviceFamilyInfoEnum(0,local_res8,0);
if ((local_res8[0] != 5) ||
   (pwVar4 = L"OSDATA\\Software\\Microsoft\\WindowsSelfhost\\Applicability", local_res8[0] !=
   5))
{
  pwVar4 = L"Software\\Microsoft\\WindowsSelfhost\\Applicability";
}
local_res10[0] = 4;
uVar3 = RegGetValueW(&DAT_ffffffff80000002,pwVar4,L"TestFlags",0x10,0,local_res18,local_res10)
```

Executables Involved

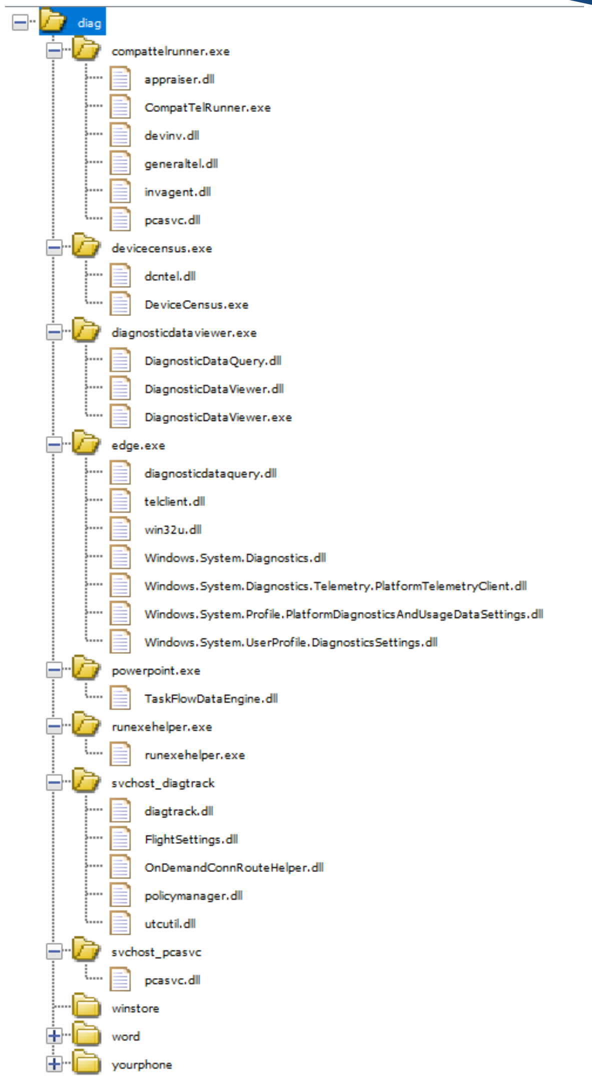
- We have just begun to scratch the surface
- More time is required to understand messaging system (ETW, COM, RPC), scheduled tasks, HTTP/HTTPS communication, filesystem storage locations, and more…
- Like how is onecore involved?
- Or why does generaltel.dll look for games?



```
18005f5f8 68 7d 06     addr     u_crossfire.exe                  = u"crossfire.exe"
          80 01 00
          00 00
18005f600 88 7d 06     addr     u_crysis3.exe                    = u"crysis3.exe"
          80 01 00
          00 00
18005f608 a0 7d 06     addr     u_curseclient.exe                = u"curseclient.exe"
          80 01 00
          00 00
18005f610 c0 7d 06     addr     u_data.exe                       = u"data.exe"
          80 01 00
          00 00
18005f618 d8 7d 06     addr     u_darksoulsii.exe                = u"darksoulsii.exe"
          80 01 00
          00 00
18005f620 f8 7d 06     addr     u_dayz.exe                       = u"dayz.exe"
          80 01 00
          00 00
18005f628 10 7e 06     addr     u_ddzrpg.exe                     = u"ddzrpg.exe"
          80 01 00
```



```
18001c96d 48 8d 35     LEA      RSI,[CoreGamerExes]                        = 180067968
          7c 2b 04
          00

                   LAB_18001c974                        XREF[1]:     18001c99f(j)
18001c974 48 8b 16     MOV      RDX=>u_7daystodie.exe,qword ptr [RSI]=>->u_7d… = 180067988
                                                                           = u"7daystodie.exe"
                                                                           = u"aeriaignite.exe"
                                                                           = 180067968

18001c977 49 8b cf     MOV      RCX,R15
18001c97a 48 ff 15     CALL     qword ptr [->SHLWAPI.DLL::StrCmpIW]
          77 29 07
          00
```



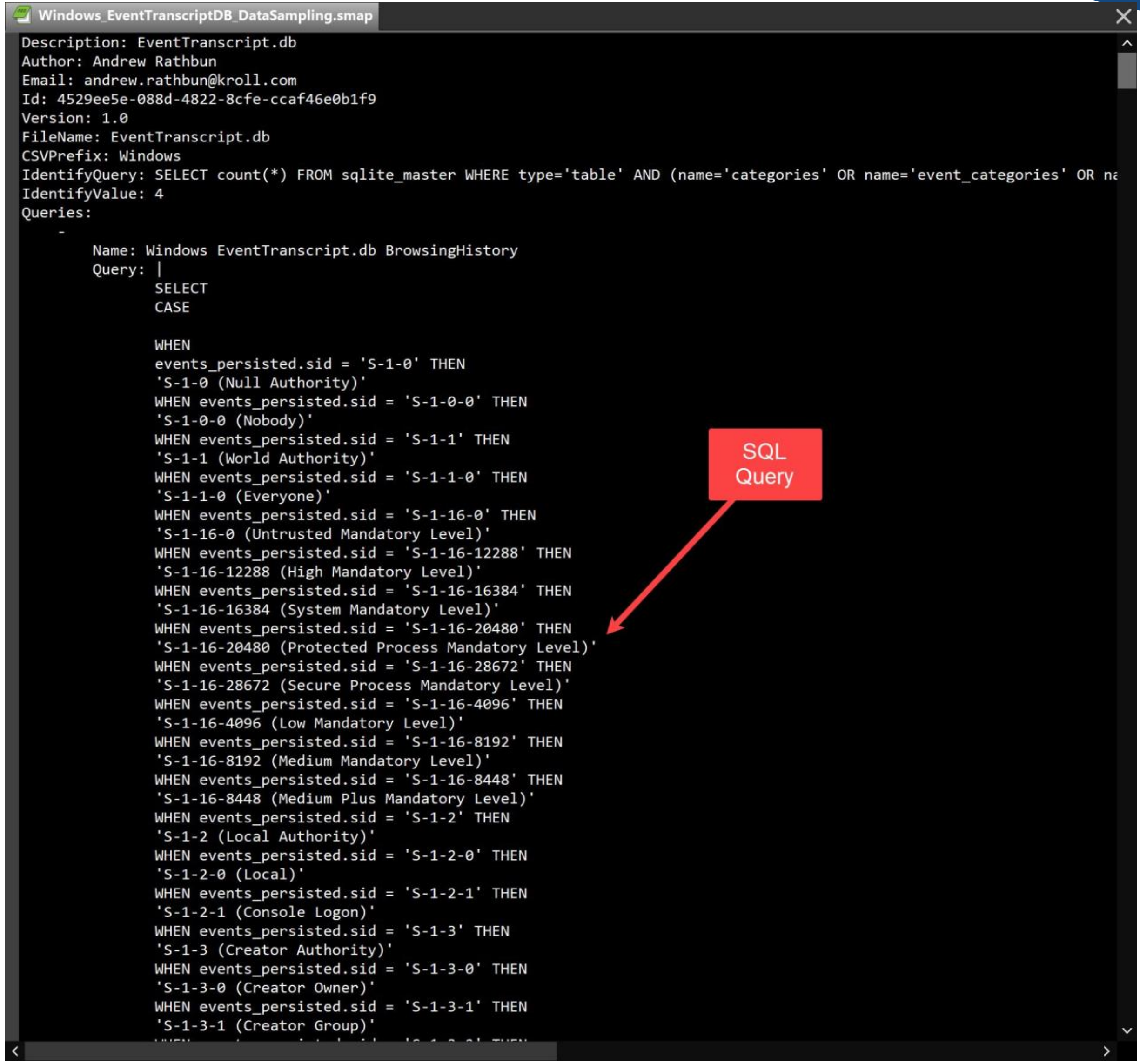
```
diag
  compattelrunner.exe
    appraiser.dll
    CompatTelRunner.exe
    devinv.dll
    generaltel.dll
    invagent.dll
    pcasvc.dll
  devicecensus.exe
    dcntel.dll
    DeviceCensus.exe
  diagnosticdataviewer.exe
    DiagnosticDataQuery.dll
    DiagnosticDataViewer.dll
    DiagnosticDataViewer.exe
  edge.exe
    diagnosticdataquery.dll
    telclient.dll
    win32u.dll
    Windows.System.Diagnostics.dll
    Windows.System.Diagnostics.Telemetry.PlatformTelemetryClient.dll
    Windows.System.Profile.PlatformDiagnosticsAndUsageDataSettings.dll
    Windows.System.UserProfile.DiagnosticsSettings.dll
  powerpoint.exe
    TaskFlowDataEngine.dll
  runexehelper.exe
    runexehelper.exe
  svchost_diagtrack
    diagtrack.dll
    FlightSettings.dll
    OnDemandConnRouteHelper.dll
    policymanager.dll
    utcutil.dll
  svchost_pcasvc
    pcasvc.dll
  winstore
  word
  yourphone
```

What is SQLECmd?

SQLECmd

High level overview

- A tool by Eric Zimmerman to parse SQLite databases quickly, regardless of where they came from (any OS!)
- Utilizes Maps (similar to EvtxECmd) to influence CSV output
  - Maps consist of parameters for which SQLECmd looks to determine if a Map matches an SQLite database to be parsed
  - If the Map matches the database schema, SQLECmd will execute the SQLite query within the Map against the SQLite DB to produce output
- Anyone can create Maps for any DB in existence

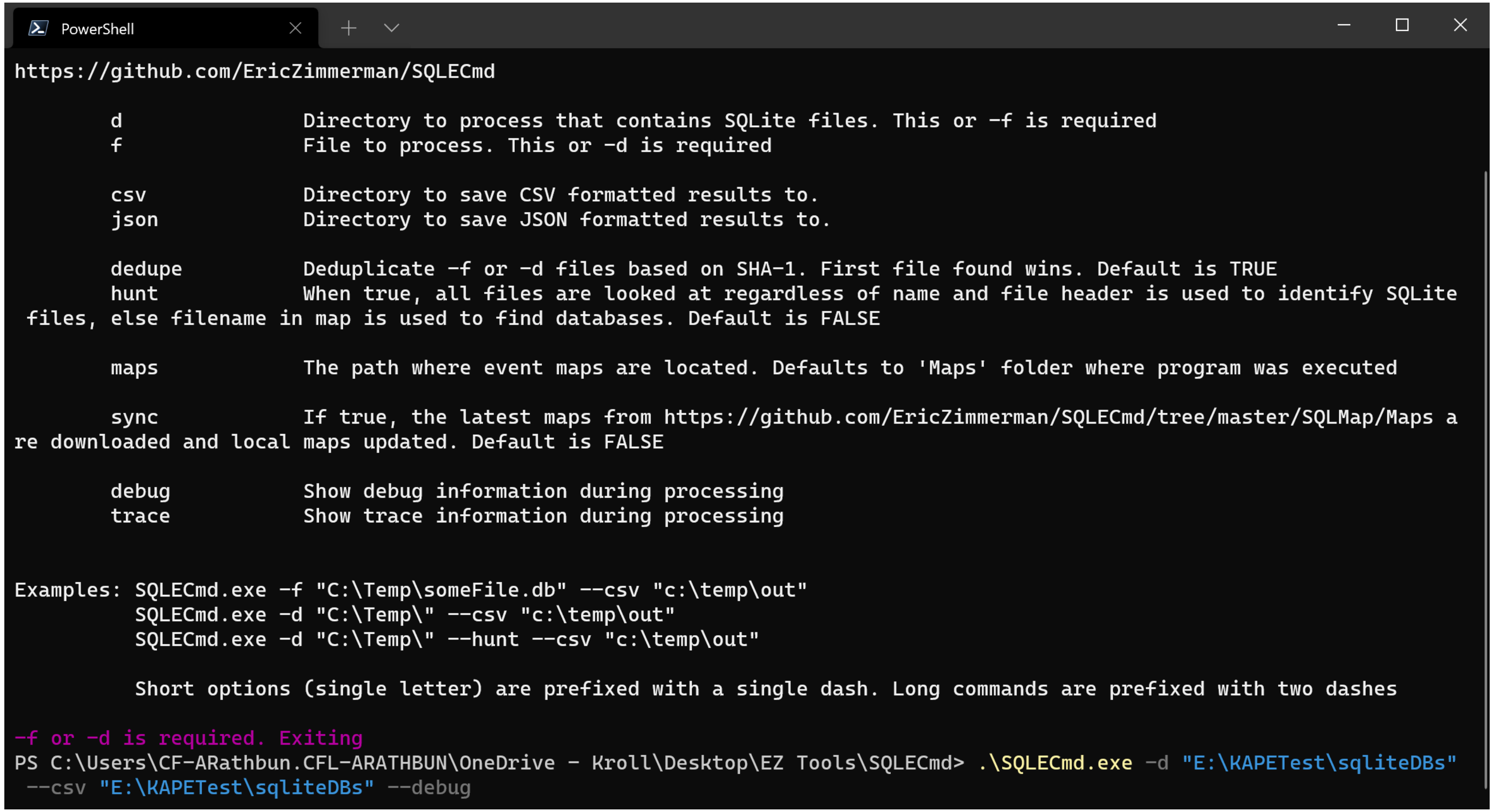


```
Description: EventTranscript.db
Author: Andrew Rathbun
Email: andrew.rathbun@kroll.com
Id: 4529ee5e-088d-4822-8cfe-ccaf46e0b1f9
Version: 1.0
FileName: EventTranscript.db
CSVPrefix: Windows
IdentifyQuery: SELECT count(*) FROM sqlite_master WHERE type='table' AND (name='categories' OR name='event_categories' OR na
IdentifyValue: 4
Queries:
    -
        Name: Windows EventTranscript.db BrowsingHistory
        Query: |
               SELECT
               CASE

               WHEN
               events_persisted.sid = 'S-1-0' THEN
               'S-1-0 (Null Authority)'
               WHEN events_persisted.sid = 'S-1-0-0' THEN
               'S-1-0-0 (Nobody)'
               WHEN events_persisted.sid = 'S-1-1' THEN
               'S-1-1 (World Authority)'
               WHEN events_persisted.sid = 'S-1-1-0' THEN
               'S-1-1-0 (Everyone)'
               WHEN events_persisted.sid = 'S-1-16-0' THEN
               'S-1-16-0 (Untrusted Mandatory Level)'
               WHEN events_persisted.sid = 'S-1-16-12288' THEN
               'S-1-16-12288 (High Mandatory Level)'
               WHEN events_persisted.sid = 'S-1-16-16384' THEN
               'S-1-16-16384 (System Mandatory Level)'
               WHEN events_persisted.sid = 'S-1-16-20480' THEN
               'S-1-16-20480 (Protected Process Mandatory Level)'
               WHEN events_persisted.sid = 'S-1-16-28672' THEN
               'S-1-16-28672 (Secure Process Mandatory Level)'
               WHEN events_persisted.sid = 'S-1-16-4096' THEN
               'S-1-16-4096 (Low Mandatory Level)'
               WHEN events_persisted.sid = 'S-1-16-8192' THEN
               'S-1-16-8192 (Medium Mandatory Level)'
               WHEN events_persisted.sid = 'S-1-16-8448' THEN
               'S-1-16-8448 (Medium Plus Mandatory Level)'
               WHEN events_persisted.sid = 'S-1-2' THEN
               'S-1-2 (Local Authority)'
               WHEN events_persisted.sid = 'S-1-2-0' THEN
               'S-1-2-0 (Local)'
               WHEN events_persisted.sid = 'S-1-2-1' THEN
               'S-1-2-1 (Console Logon)'
               WHEN events_persisted.sid = 'S-1-3' THEN
               'S-1-3 (Creator Authority)'
               WHEN events_persisted.sid = 'S-1-3-0' THEN
               'S-1-3-0 (Creator Owner)'
               WHEN events_persisted.sid = 'S-1-3-1' THEN
               'S-1-3-1 (Creator Group)'
```

Parsing with SQLECmd

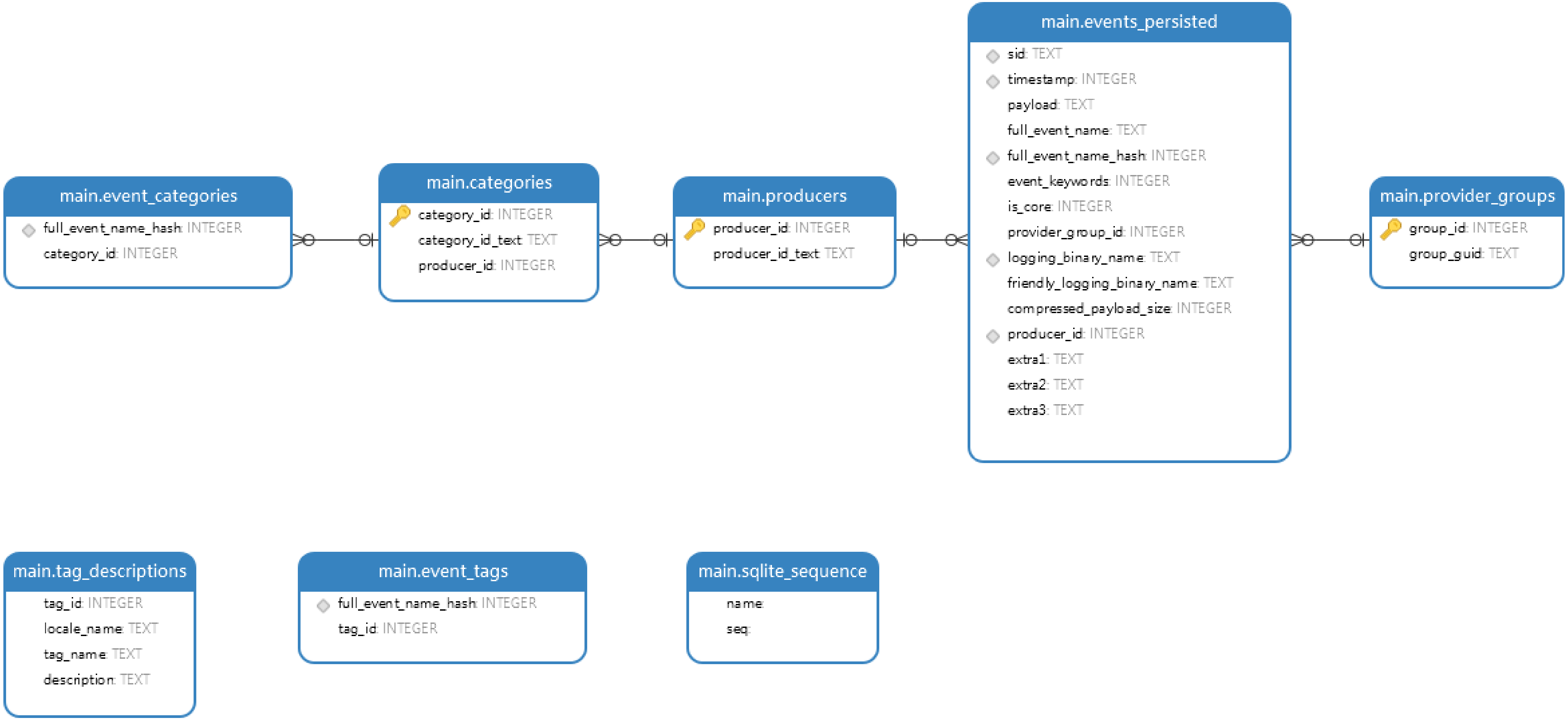
Using SQLECmd outside of KAPE to parse EventTranscript.db

sqlecmd.exe -d path\to\file.db --csv path\to\csv\output --debug



```
https://github.com/EricZimmerman/SQLECmd

        d               Directory to process that contains SQLite files. This or -f is required
        f               File to process. This or -d is required

        csv             Directory to save CSV formatted results to.
        json            Directory to save JSON formatted results to.

        dedupe          Deduplicate -f or -d files based on SHA-1. First file found wins. Default is TRUE
        hunt            When true, all files are looked at regardless of name and file header is used to identify SQLite
 files, else filename in map is used to find databases. Default is FALSE

        maps            The path where event maps are located. Defaults to 'Maps' folder where program was executed

        sync            If true, the latest maps from https://github.com/EricZimmerman/SQLECmd/tree/master/SQLMap/Maps a
re downloaded and local maps updated. Default is FALSE

        debug           Show debug information during processing
        trace           Show trace information during processing


Examples: SQLECmd.exe -f "C:\Temp\someFile.db" --csv "c:\temp\out"
          SQLECmd.exe -d "C:\Temp\" --csv "c:\temp\out"
          SQLECmd.exe -d "C:\Temp\" --hunt --csv "c:\temp\out"

          Short options (single letter) are prefixed with a single dash. Long commands are prefixed with two dashes

-f or -d is required. Exiting
PS C:\Users\CF-ARathbun.CFL-ARATHBUN\OneDrive - Kroll\Desktop\EZ Tools\SQLECmd> .\SQLECmd.exe -d "E:\KAPETest\sqliteDBs"
 --csv "E:\KAPETest\sqliteDBs" --debug
```

EventTranscript.db Schema (Refresher)



main.event_categories

- full_event_name_hash: INTEGER
- category_id: INTEGER

main.categories

- category_id: INTEGER
- category_id_text: TEXT
- producer_id: INTEGER

main.producers

- producer_id: INTEGER
- producer_id_text: TEXT

main.events_persisted

- sid: TEXT
- timestamp: INTEGER
- payload: TEXT
- full_event_name: TEXT
- full_event_name_hash: INTEGER
- event_keywords: INTEGER
- is_core: INTEGER
- provider_group_id: INTEGER
- logging_binary_name: TEXT
- friendly_logging_binary_name: TEXT
- compressed_payload_size: INTEGER
- producer_id: INTEGER
- extra1: TEXT
- extra2: TEXT
- extra3: TEXT

main.provider_groups

- group_id: INTEGER
- group_guid: TEXT

main.tag_descriptions

- tag_id: INTEGER
- locale_name: TEXT
- tag_name: TEXT
- description: TEXT

main.event_tags

- full_event_name_hash: INTEGER
- tag_id: INTEGER

main.sqlite_sequence

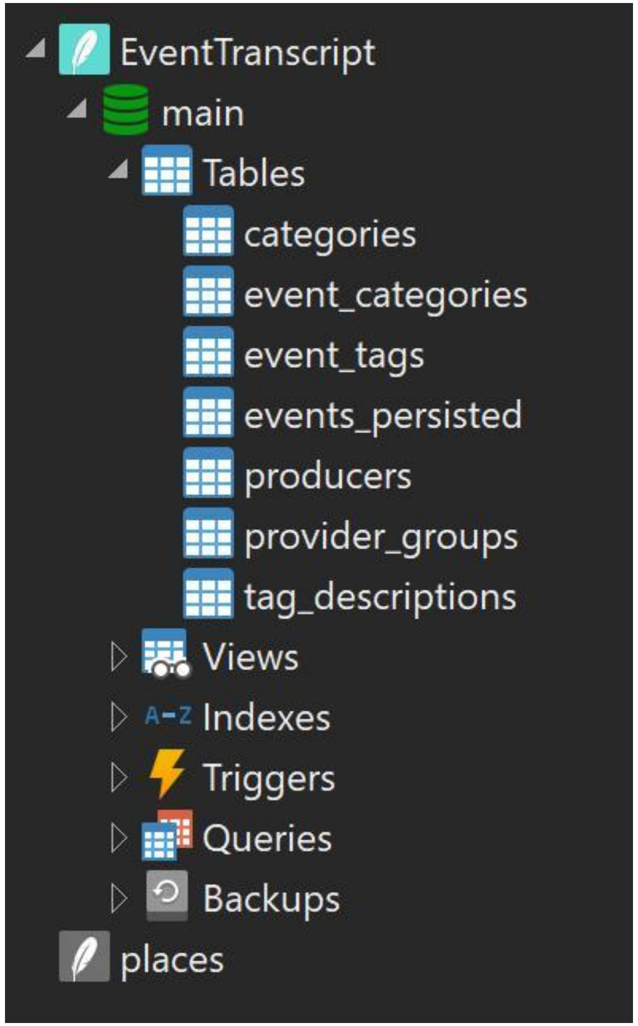
- name:
- seq:

Parsing with SQLECmd – Understanding Maps

Matching table names specified within Map to Tables that exist in a given DB

- **IdentifyQuery**: SELECT count(*) FROM sqlite_master WHERE type='table' AND (name='**categories**' OR name='**event_categories**' OR name='**event_tags**' OR name='**events_persisted**' OR name='**producers**' OR name='**provider_groups**' OR name=**'tag_descriptions**');
- **IdentifyValue**: 4

- **IdentifyValue** is only 4 because I've seen only 4 of these present within this database during my testing
- So long as **IdentifyQuery** and **IdentifyValue** are valid, and the DB filename matches, SQLECmd will parse the DB and provide output according to the Query within the Map.



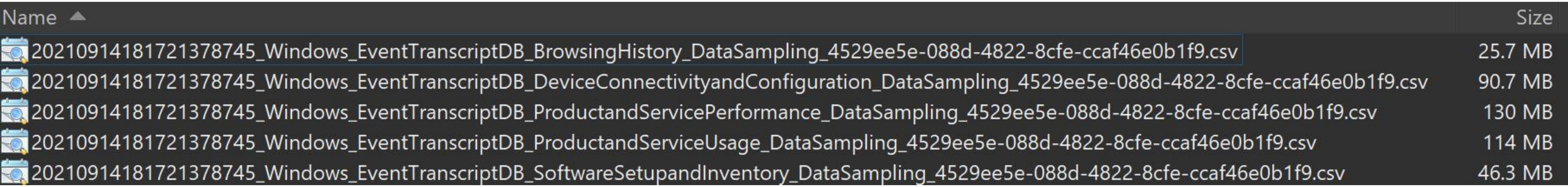
```
◢ EventTranscript
   ◢ main
      ◢ Tables
           categories
           event_categories
           event_tags
           events_persisted
           producers
           provider_groups
           tag_descriptions
      ▷ Views
      ▷ Indexes
      ▷ Triggers
      ▷ Queries
      ▷ Backups
  places
```

EventTranscript.db SQLECmd Map

High level overview

- Outputs 6 CSVs (One for each Tag_Description):
  - EventTranscript.db_BrowsingHistory
  - EventTranscript.db_Device Connectivity and Configuration
  - EventTranscript.db_Inking Typing and Speech Utterance
  - EventTranscript.db_ProductandServicePerformance
  - EventTranscript.db_Product and Service Usage
  - EventTranscript.db_Software Setup and Inventory

- This helps reduce the size of the output when dealing with data sampling where DBs have been seen to be 1GB+ in size with similar size CSV output
- Attempts to label presence of **DataSampling** vs **NoDataSampling** based on current research and understanding of the artifact

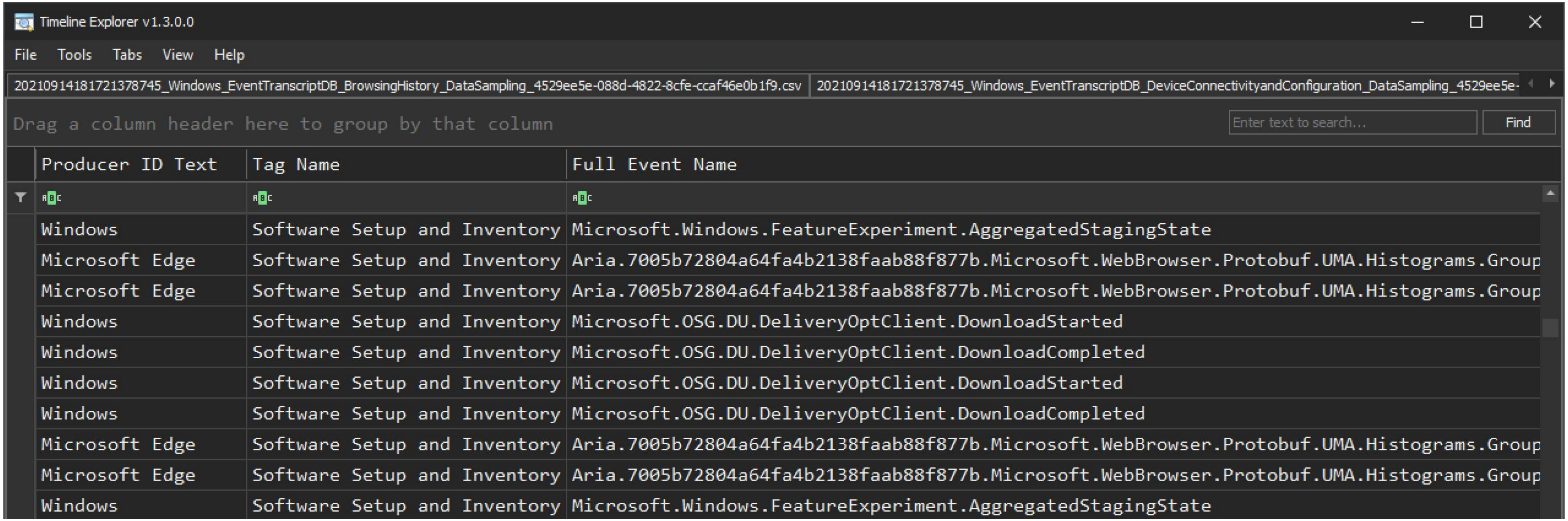


Name	Size
20210914181721378745_Windows_EventTranscriptDB_BrowsingHistory_DataSampling_4529ee5e-088d-4822-8cfe-ccaf46e0b1f9.csv	25.7 MB
20210914181721378745_Windows_EventTranscriptDB_DeviceConnectivityandConfiguration_DataSampling_4529ee5e-088d-4822-8cfe-ccaf46e0b1f9.csv	90.7 MB
20210914181721378745_Windows_EventTranscriptDB_ProductandServicePerformance_DataSampling_4529ee5e-088d-4822-8cfe-ccaf46e0b1f9.csv	130 MB
20210914181721378745_Windows_EventTranscriptDB_ProductandServiceUsage_DataSampling_4529ee5e-088d-4822-8cfe-ccaf46e0b1f9.csv	114 MB
20210914181721378745_Windows_EventTranscriptDB_SoftwareSetupandInventory_DataSampling_4529ee5e-088d-4822-8cfe-ccaf46e0b1f9.csv	46.3 MB

^ No Inking Typing and Speech Utterance data parsed ^

Analyzing SQLECmd Output

Use Timeline Explorer to view CSV output


Timeline Explorer v1.3.0.0
File
Tools
Tabs
View
Help
Drag a column header here to group by that column
Enter text to search...
Find

20210914181721378745_Windows_EventTranscriptDB_BrowsingHistory_DataSampling_4529ee5e-088d-4822-8cfe-ccaf46e0b1f9.csv

20210914181721378745_Windows_EventTranscriptDB_DeviceConnectivityandConfiguration_DataSampling_4529ee5e-

Producer ID Text	Tag Name	Full Event Name
Windows	Software Setup and Inventory	Microsoft.Windows.FeatureExperiment.AggregatedStagingState
Microsoft Edge	Software Setup and Inventory	Aria.7005b72804a64fa4b2138faab88f877b.Microsoft.WebBrowser.Protobuf.UMA.Histograms.Group
Microsoft Edge	Software Setup and Inventory	Aria.7005b72804a64fa4b2138faab88f877b.Microsoft.WebBrowser.Protobuf.UMA.Histograms.Group
Windows	Software Setup and Inventory	Microsoft.OSG.DU.DeliveryOptClient.DownloadStarted
Windows	Software Setup and Inventory	Microsoft.OSG.DU.DeliveryOptClient.DownloadCompleted
Windows	Software Setup and Inventory	Microsoft.OSG.DU.DeliveryOptClient.DownloadStarted
Windows	Software Setup and Inventory	Microsoft.OSG.DU.DeliveryOptClient.DownloadCompleted
Microsoft Edge	Software Setup and Inventory	Aria.7005b72804a64fa4b2138faab88f877b.Microsoft.WebBrowser.Protobuf.UMA.Histograms.Group
Microsoft Edge	Software Setup and Inventory	Aria.7005b72804a64fa4b2138faab88f877b.Microsoft.WebBrowser.Protobuf.UMA.Histograms.Group
Windows	Software Setup and Inventory	Microsoft.Windows.FeatureExperiment.AggregatedStagingState

Important Notes

Tips and Tricks

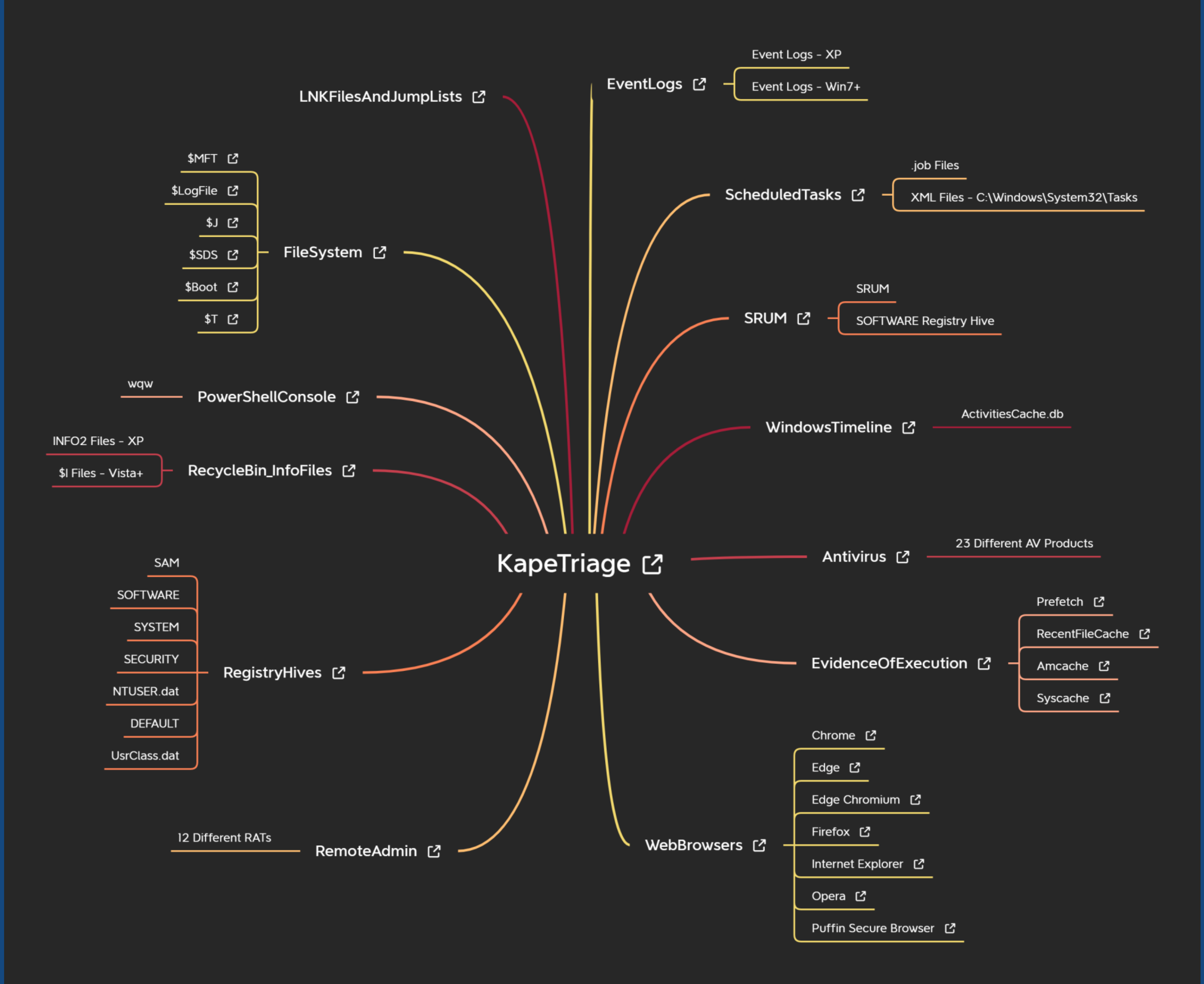
- Always be sure to run a sync with SQLECmd if it's been a while since you last did:
  - sqlecmd.exe --sync
  - Sync_SQLECmd.mkape Module in KAPE
- You can use the --hunt switch to locate SQLite DBs
  - Useful since not every SQLite DB has a file extension nor is every .db* file an SQLite database
  - SQLECmd-Hunt.mkape Module automates this process!
- There is no right or wrong way to locate SQLite DBs. They are EVERYWHERE!
  - Also, be hungry for output beyond what tools provide you. No tool can parse EVERYTHING out there.
- **Navicat for SQLite** is a very useful tool (but not free) for making SQL queries (you don't need to know SQL; the Query Builder will give you a crash course)
- If you need help building queries or finding DBs, contact me on either Discord, Twitter, or LinkedIn and let's solve the world's problems together!

Introduction to KAPE

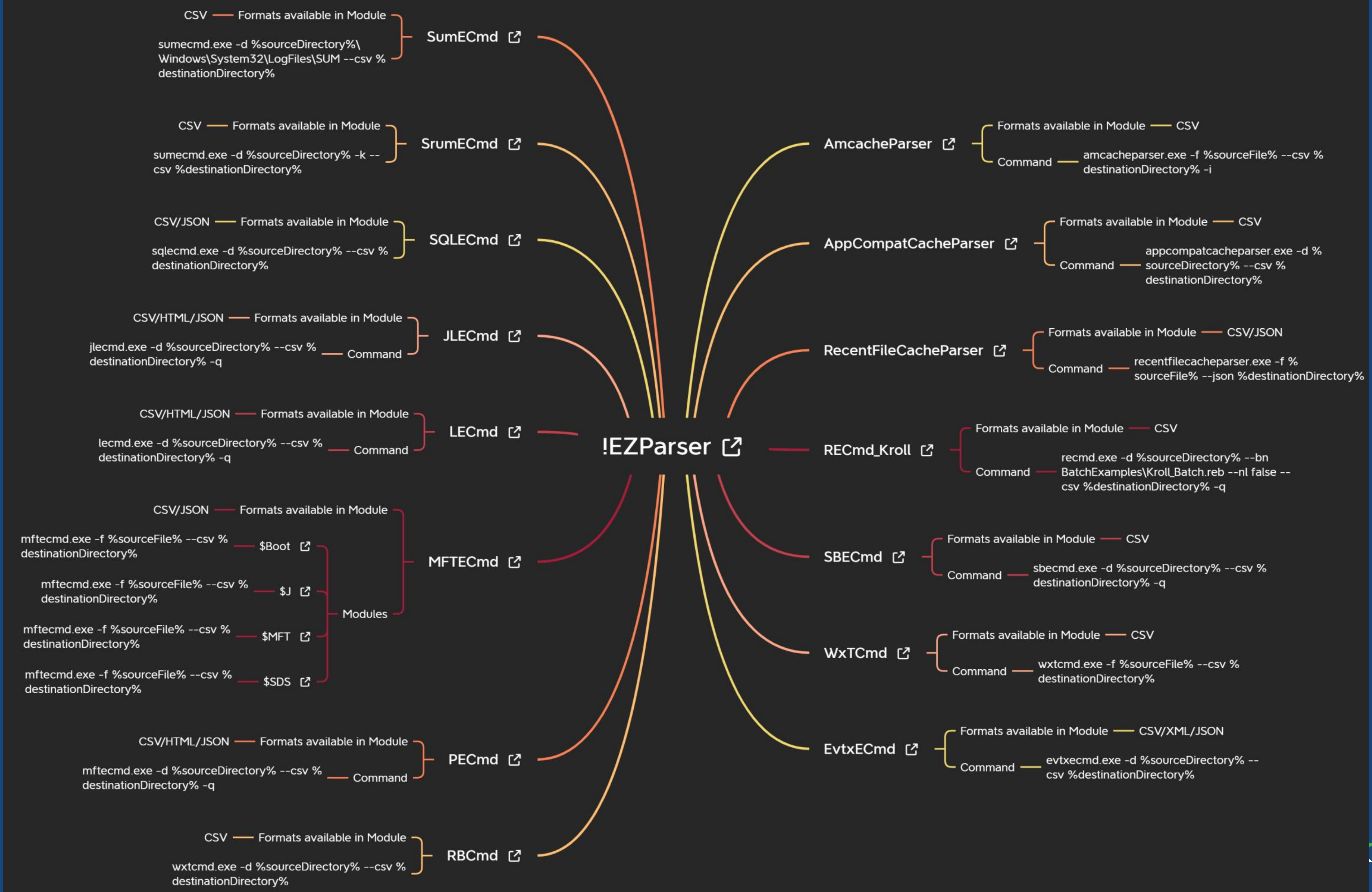
Introduction to KAPE

High level overview

- Kroll Artifact Parser and Extractor (KAPE) is primarily a triage program
- It targets a device or storage location to:
  - Find the most forensically relevant artifacts (based on your needs) using **Targets**
  - Parse them within a few minutes using EZ Tools/your other favorite CLI tool using **Modules**
- KAPE can be used to collect the most critical artifacts prior to the start of imaging
  - While the imaging completes, the data generated by KAPE can be reviewed for leads, building timelines, etc.
- Common IR Workflow
  - **KapeTriage** Target -> **!EZParser** Module



- KapeTriage
  - LNKFilesAndJumpLists
  - EventLogs
    - Event Logs - XP
    - Event Logs - Win7+
  - FileSystem
    - $MFT
    - $LogFile
    - $J
    - $SDS
    - $Boot
    - $T
  - ScheduledTasks
    - .job Files
    - XML Files - C:\Windows\System32\Tasks
  - SRUM
    - SRUM
    - SOFTWARE Registry Hive
  - PowerShellConsole
    - wqw
  - WindowsTimeline
    - ActivitiesCache.db
  - RecycleBin_InfoFiles
    - INFO2 Files - XP
    - $I Files - Vista+
  - Antivirus
    - 23 Different AV Products
  - RegistryHives
    - SAM
    - SOFTWARE
    - SYSTEM
    - SECURITY
    - NTUSER.dat
    - DEFAULT
    - UsrClass.dat
  - EvidenceOfExecution
    - Prefetch
    - RecentFileCache
    - Amcache
    - Syscache
  - RemoteAdmin
    - 12 Different RATs
  - WebBrowsers
    - Chrome
    - Edge
    - Edge Chromium
    - Firefox
    - Internet Explorer
    - Opera
    - Puffin Secure Browser



!EZParser

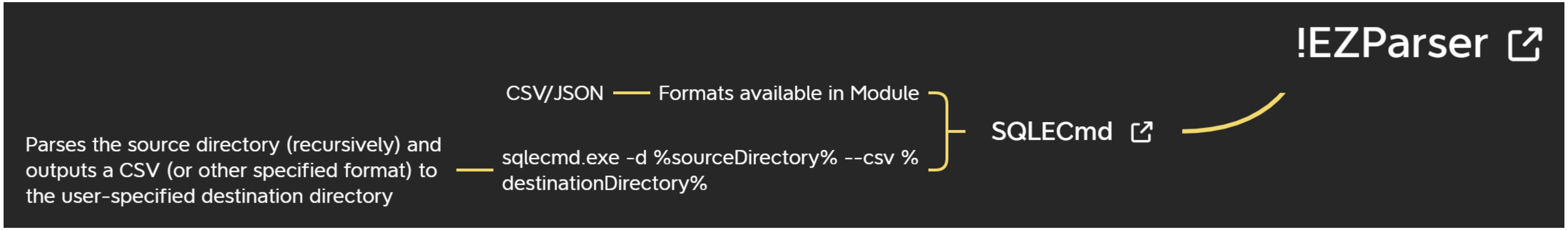
- SumECmd
  - Formats available in Module — CSV
  - sumecmd.exe -d %sourceDirectory%\Windows\System32\LogFiles\SUM --csv %destinationDirectory%
- SrumECmd
  - Formats available in Module — CSV
  - sumecmd.exe -d %sourceDirectory% -k --csv %destinationDirectory%
- SQLECmd
  - Formats available in Module — CSV/JSON
  - sqlecmd.exe -d %sourceDirectory% --csv %destinationDirectory%
- JLECmd
  - Formats available in Module — CSV/HTML/JSON
  - Command — jlecmd.exe -d %sourceDirectory% --csv %destinationDirectory% -q
- LECmd
  - Formats available in Module — CSV/HTML/JSON
  - Command — lecmd.exe -d %sourceDirectory% --csv %destinationDirectory% -q
- MFTECmd
  - Formats available in Module — CSV/JSON
  - Modules
    - $Boot — mftecmd.exe -f %sourceFile% --csv %destinationDirectory%
    - $J — mftecmd.exe -f %sourceFile% --csv %destinationDirectory%
    - $MFT — mftecmd.exe -f %sourceFile% --csv %destinationDirectory%
    - $SDS — mftecmd.exe -f %sourceFile% --csv %destinationDirectory%
- PECmd
  - Formats available in Module — CSV/HTML/JSON
  - Command — mftecmd.exe -d %sourceDirectory% --csv %destinationDirectory% -q
- RBCmd
  - Formats available in Module — CSV
  - wxtcmd.exe -d %sourceDirectory% --csv %destinationDirectory%
- AmcacheParser
  - Formats available in Module — CSV
  - Command — amcacheparser.exe -f %sourceFile% --csv %destinationDirectory% -i
- AppCompatCacheParser
  - Formats available in Module — CSV
  - Command — appcompatcacheparser.exe -d %sourceDirectory% --csv %destinationDirectory%
- RecentFileCacheParser
  - Formats available in Module — CSV/JSON
  - Command — recentfilecacheparser.exe -f %sourceFile% --json %destinationDirectory%
- RECmd_Kroll
  - Formats available in Module — CSV
  - Command — recmd.exe -d %sourceDirectory% --bn BatchExamples\Kroll_Batch.reb --nl false --csv %destinationDirectory% -q
- SBECmd
  - Formats available in Module — CSV
  - Command — sbecmd.exe -d %sourceDirectory% --csv %destinationDirectory% -q
- WxTCmd
  - Formats available in Module — CSV
  - Command — wxtcmd.exe -f %sourceFile% --csv %destinationDirectory%
- EvtxECmd
  - Formats available in Module — CSV/XML/JSON
  - Command — evtxecmd.exe -d %sourceDirectory% --csv %destinationDirectory%

Automating SQLECmd with KAPE

SQLECmd Module in KAPE

Automating SQLECmd with KAPE

- Automates the most common use case for SQLECmd with KAPE
  – Attempts to match SQLite databases with SQLECmd Maps based on DB filename and the DB Schema
- Provides CSV output in accordance with the query within the respective Map
  – Maps are only as good as the author made them to be
  – If you don't like the output, the queries are open sourced, and you can improve them!
- Outputs into SQLDatabases folder within your specified --mdest (Module Destination) directory

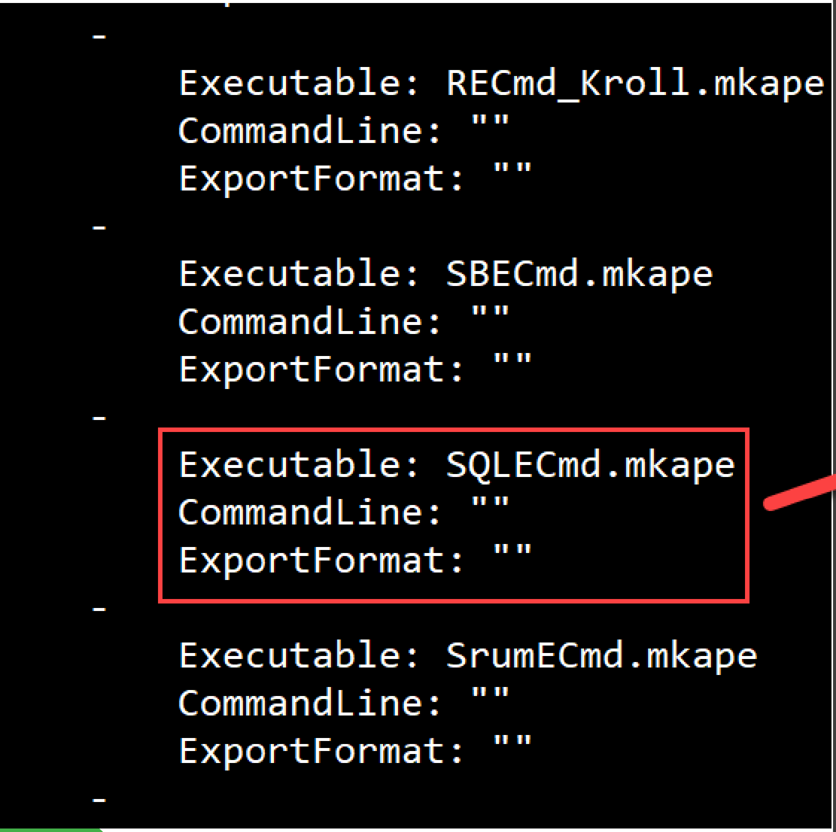


Parses the source directory (recursively) and outputs a CSV (or other specified format) to the user-specified destination directory

```
sqlecmd.exe -d %sourceDirectory% --csv %
destinationDirectory%
```

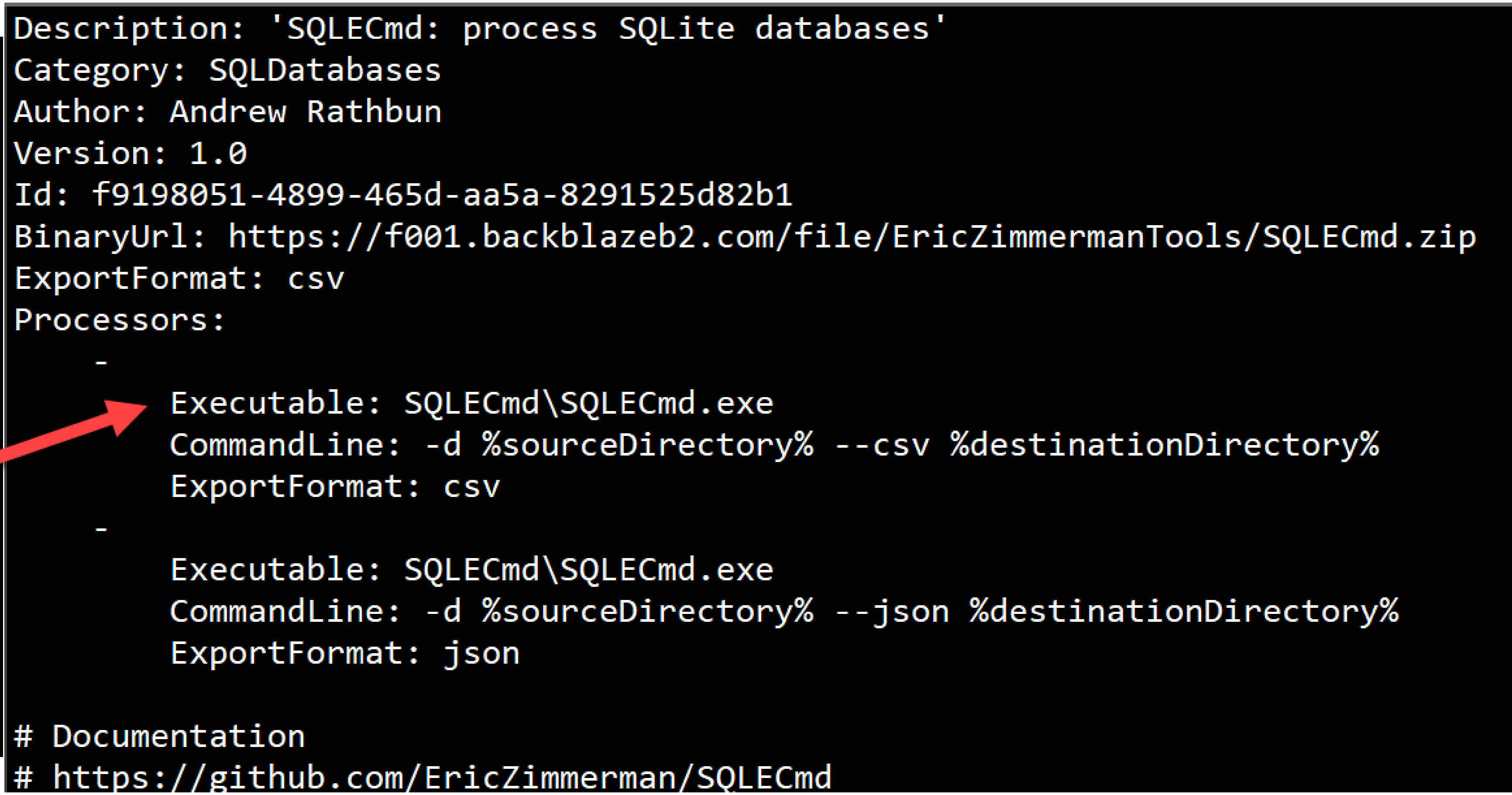
SQLECmd Module in KAPE

Running the !EZParser Module will run the SQLECmd Module, as seen here:



```
    -
        Executable: RECmd_Kroll.mkape
        CommandLine: ""
        ExportFormat: ""
    -
        Executable: SBECmd.mkape
        CommandLine: ""
        ExportFormat: ""
    -
        Executable: SQLECmd.mkape
        CommandLine: ""
        ExportFormat: ""
    -
        Executable: SrumECmd.mkape
        CommandLine: ""
        ExportFormat: ""
    -
```

^ !EZParser ^

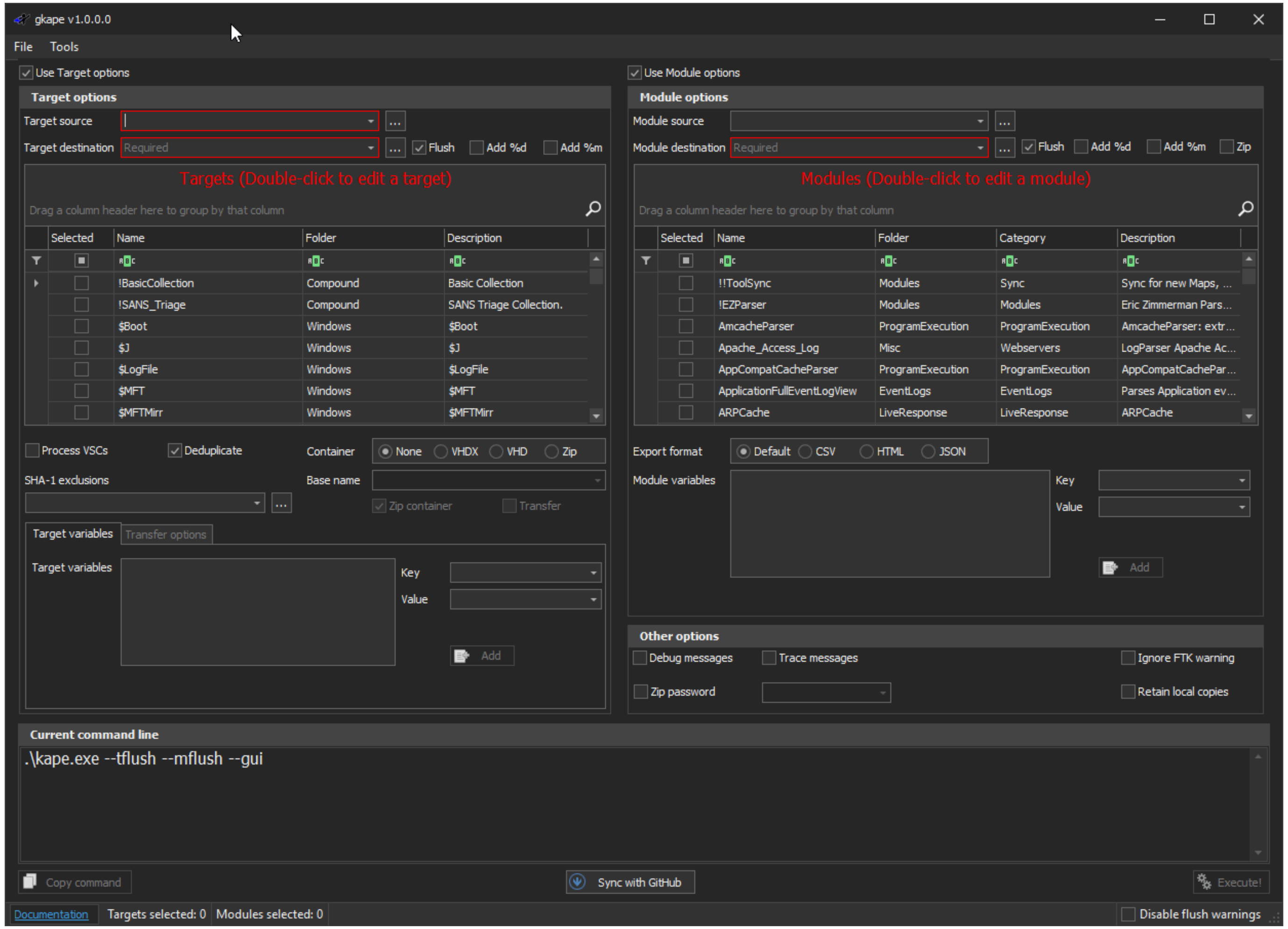


```
Description: 'SQLECmd: process SQLite databases'
Category: SQLDatabases
Author: Andrew Rathbun
Version: 1.0
Id: f9198051-4899-465d-aa5a-8291525d82b1
BinaryUrl: https://f001.backblazeb2.com/file/EricZimmermanTools/SQLECmd.zip
ExportFormat: csv
Processors:
    -
        Executable: SQLECmd\SQLECmd.exe
        CommandLine: -d %sourceDirectory% --csv %destinationDirectory%
        ExportFormat: csv
    -
        Executable: SQLECmd\SQLECmd.exe
        CommandLine: -d %sourceDirectory% --json %destinationDirectory%
        ExportFormat: json

# Documentation
# https://github.com/EricZimmerman/SQLECmd
```

Using KAPE

gKape GUI: KapeTriage -> !EZParser Workflow

- Acquires data with Targets (**KapeTriage**) from specified **--tsource** and places a copy within specified **--tdest** directory
- Processes acquired data with Modules (**!EZParser**) and places parsed output within specified **--mdest** directory
- CSV for export format of parsed output
- Debug messages help for troubleshooting should anything not work as intended
- Ingest into Timeline Explorer, Modern CSV, Excel, or any other CSV viewer tool for analysis



☑ Use Target options

Target options

Target source:
Target destination: Required
☑ Flush □ Add %d □ Add %m

Targets (Double-click to edit a target)

Drag a column header here to group by that column

Selected	Name	Folder	Description
□	!BasicCollection	Compound	Basic Collection
□	!SANS_Triage	Compound	SANS Triage Collection.
□	$Boot	Windows	$Boot
□	$J	Windows	$J
□	$LogFile	Windows	$LogFile
□	$MFT	Windows	$MFT
□	$MFTMirr	Windows	$MFTMirr

□ Process VSCs ☑ Deduplicate

Container: ● None ○ VHDX ○ VHD ○ Zip

SHA-1 exclusions

Base name:

☑ Zip container □ Transfer

Target variables

Key:
Value:

☑ Use Module options

Module options

Module source:
Module destination: Required
☑ Flush □ Add %d □ Add %m □ Zip

Modules (Double-click to edit a module)

Drag a column header here to group by that column

Selected	Name	Folder	Category	Description
□	!!ToolSync	Modules	Sync	Sync for new Maps, ...
□	!EZParser	Modules	Modules	Eric Zimmerman Pars...
□	AmcacheParser	ProgramExecution	ProgramExecution	AmcacheParser: extr...
□	Apache_Access_Log	Misc	Webservers	LogParser Apache Ac...
□	AppCompatCacheParser	ProgramExecution	ProgramExecution	AppCompatCachePar...
□	ApplicationFullEventLogView	EventLogs	EventLogs	Parses Application ev...
□	ARPCache	LiveResponse	LiveResponse	ARPCache

Export format: ● Default ○ CSV ○ HTML ○ JSON

Module variables

Key:
Value:

Other options

□ Debug messages □ Trace messages □ Ignore FTK warning

□ Zip password □ Retain local copies

Current command line

```
.\kape.exe --tflush --mflush --gui
```

Targets selected: 0 Modules selected: 0 □ Disable flush warnings

Final Thoughts

- Kroll speculates that Diagnostic Data and Telemetry within Windows is here to stay
  - It’s reasonable to speculate that the level of logging should only increase over time given the value of the data to Microsoft as it relates to Windows and its future development
  - If that holds true, this artifact should only become more prevalent over time!
- Regardless, if this database exists without data sampling, it can still serve as a redundant source for information commonly available within the Windows Registry, Event Logs, etc.
- EventTranscript.db persists conventional anti-forensics methods such as Event Log clearing, timestomping, etc
- It will take a long time to fully research this database’s 2500+ events to identify which ones are the most fruitful for DFIR examiners
  - Explore the DB yourself. Report your findings! Blog about it. Pivot on Event Names and try to identify those that provide quick wins for the DFIR community and SHARE your findings.
- Discuss or provide any new findings on the EventTranscript.db Research GitHub repo

Questions?

Resources

- Kroll Blog
  - Forensically Unpacking EventTranscript.db: An Investigative Series (kroll.com)
- GitHub Repos
  - KapeFiles: EricZimmerman/KapeFiles: This repository serves as a place for community created Targets and Modules for use with KAPE. (github.com)
  - SQLECmd: EricZimmerman/SQLECmd (github.com)
- Andrew's GitHub
  - EventTranscript.db Research: rathbuna/EventTranscript.db-Research: A repo for centralizing ongoing research on the new Windows 10/11 DFIR artifact, EventTranscript.db. (github.com)
  - Awesome-KAPE: rathbuna/Awesome-KAPE: A curated list of KAPE-related resources (github.com)

For More KAPE:

Intensive Training and Certification Sessions

- Virtual live sessions
- Max 25 students

Full Calendar Available here:

bit.ly/KAPE2021

SCHEDULE	INSTRUCTORS
September 28, 2021 10:00 a.m. - 7:00 p.m. ET	Eric Zimmerman Sean Straw Scott Zuberbuehler Andrew Rathbun
October 7, 2021 8:00 a.m. - 5:00 p.m. GMT	James Thoburn Paul Wells Guillermo Roman
October 20, 2021 9:00 a.m. - 6:00 p.m. HKT	Paul Jackson David Klopp Rob Phillips

For more information, please contact:

KAPE@Kroll.com

About Kroll

Kroll is the world's premier provider of services and digital products related to governance, risk and transparency. We work with clients across diverse sectors in the areas of valuation, expert services, investigations, cyber security, corporate finance, restructuring, legal and business solutions, data analytics and regulatory compliance. Our firm has nearly 5,000 professionals in 30 countries and territories around the world. For more information, visit www.kroll.com.

M&A advisory, capital raising and secondary market advisory services in the United States are provided by Duff & Phelps Securities, LLC. Member FINRA/SIPC. Pagemill Partners is a Division of Duff & Phelps Securities, LLC. M&A advisory, capital raising and secondary market advisory services in the United Kingdom are provided by Duff & Phelps Securities Ltd. (DPSL), which is authorized and regulated by the Financial Conduct Authority. Valuation Advisory Services in India are provided by Duff & Phelps India Private Limited under a category 1 merchant banker license issued by the Securities and Exchange Board of India.