August, 2020

SRUM

Another Windows Time Machine

LIFARS
your digital world, **secured**

Contents

OVERVIEW

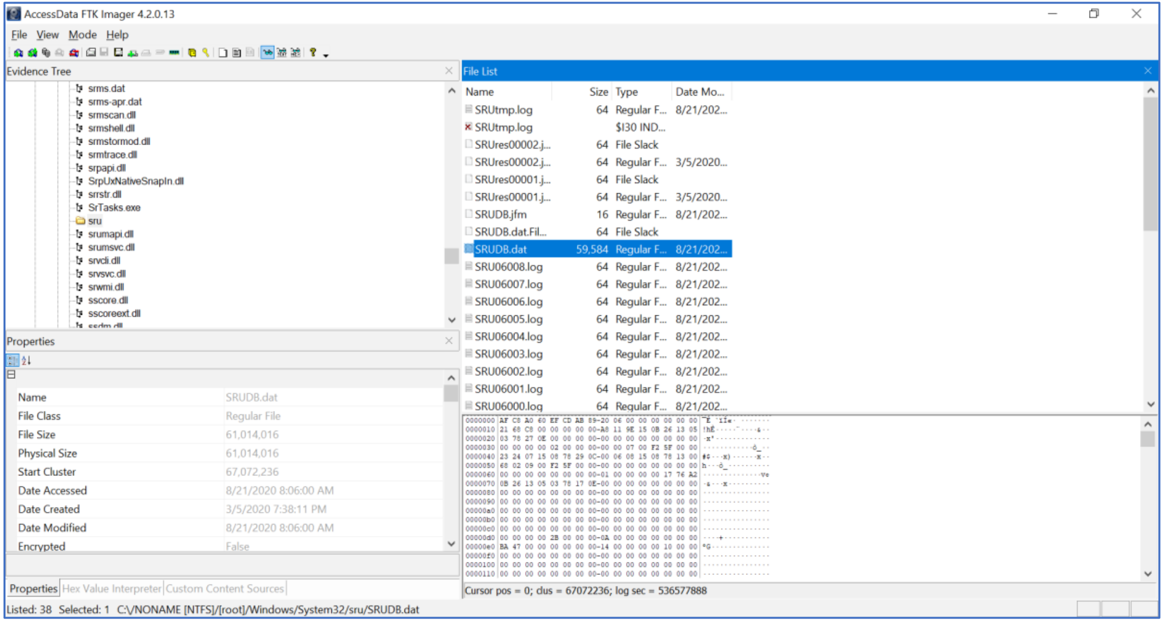
In standard forensics investigations, sooner or later arises the need to find and extract evidence of program execution on the victim system. We are looking for traces of malware that has been running on the system, or for indication that a benign application could have been misused for adversary purposes.

There are plenty of ways to answer these questions: Prefetch files, ShimCache and Amcache, user registry keys storing lists of last executed programs. However, in multiple recent investigations we faced issues with gathering more information about activities of specific programs. More precisely, we needed to determine if the executables communicated over the network or if any data were transmitted, indicating data exfiltration. These questions can be easily solved if network traffic captures are in your possession, or if you have been able to perform live analysis of the investigated environment. What if no such information is available? We need to rely on evidence present on "dead" system. Fortunately, SRUM, which stands for System Resource Usage Monitor, can give some of the answers.

WHAT, WHERE, WHY

INTRO

System Resource Usage Monitor was first spotted on Windows 8 operating system. It is a component of Diagnostic Policy Service – DPS, which enables problem detection, troubleshooting, and resolution for components in the Windows operating system. SRUM monitors services, desktop application programs, windows applications, and network connections on the host – and stores collected information in a database. This database file, SRUMDB.dat, is located in the Windows\System32\sru directory.



Evidence Tree:

- srms.dat
- srms-apr.dat
- srmscan.dll
- srmshell.dll
- srmstormod.dll
- srmtrace.dll
- srpapi.dll
- SrpUxNativeSnapIn.dll
- srrstr.dll
- SrTasks.exe
- sru
- srumapi.dll
- srumsvc.dll
- srvcli.dll
- srvsvc.dll
- srwmi.dll
- sscore.dll
- sscoreext.dll
- ssdm.dll

File List:

Name	Size	Type	Date Mo...
SRUtmp.log	64	Regular F...	8/21/202...
SRUtmp.log		$I30 IND...	
SRUres00002.j...	64	File Slack	
SRUres00002.j...	64	Regular F...	3/5/2020...
SRUres00001.j...	64	File Slack	
SRUres00001.j...	64	Regular F...	3/5/2020...
SRUDB.jfm	16	Regular F...	8/21/202...
SRUDB.dat.Fil...	64	File Slack	
SRUDB.dat	59,584	Regular F...	8/21/202...
SRU06008.log	64	Regular F...	8/21/202...
SRU06007.log	64	Regular F...	8/21/202...
SRU06006.log	64	Regular F...	8/21/202...
SRU06005.log	64	Regular F...	8/21/202...
SRU06004.log	64	Regular F...	8/21/202...
SRU06003.log	64	Regular F...	8/21/202...
SRU06002.log	64	Regular F...	8/21/202...
SRU06001.log	64	Regular F...	8/21/202...
SRU06000.log	64	Regular F...	8/21/202...

Properties:

Name	SRUDB.dat
File Class	Regular File
File Size	61,014,016
Physical Size	61,014,016
Start Cluster	67,072,236
Date Accessed	8/21/2020 8:06:00 AM
Date Created	3/5/2020 7:38:11 PM
Date Modified	8/21/2020 8:06:00 AM
Encrypted	False

```
0000000 AF C8 A0 60 EF CD AB 89-20 06 00 00 00 00 00 00  ¯È `ïÍ«‰ ·······
0000010 21 68 C8 00 00 00 00 00-A8 11 9E 15 0B 26 13 05  !hÈ·····¨·ž··&··
0000020 03 78 27 0E 00 00 00 00-00 00 00 00 00 00 00 00  ·x'·············
0000030 00 00 00 00 02 00 00 00-00 00 07 00 F2 5F 00 00  ············ò_··
0000040 23 24 07 15 08 78 29 0C-00 06 08 15 08 78 13 00  #$···x)······x··
0000050 68 02 09 00 F2 5F 00 00-00 00 00 00 00 00 00 00  h···ò_··········
0000060 00 00 00 00 00 00 00 00-01 00 00 00 00 17 76 A2  ··············v¢
0000070 0B 26 13 05 03 78 17 0E-00 00 00 00 00 00 00 00  ·&···x··········
0000080 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ················
0000090 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ················
00000a0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ················
00000b0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ················
00000c0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ················
00000d0 00 00 00 00 2B 00 00 00-0A 00 00 00 00 00 00 00  ····+···········
00000e0 BA 47 00 00 00 00 00 00-14 00 00 00 00 10 00 00  ºG··············
00000f0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ················
0000100 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ················
0000110 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00  ················
```

Cursor pos = 0; clus = 67072236; log sec = 536577888

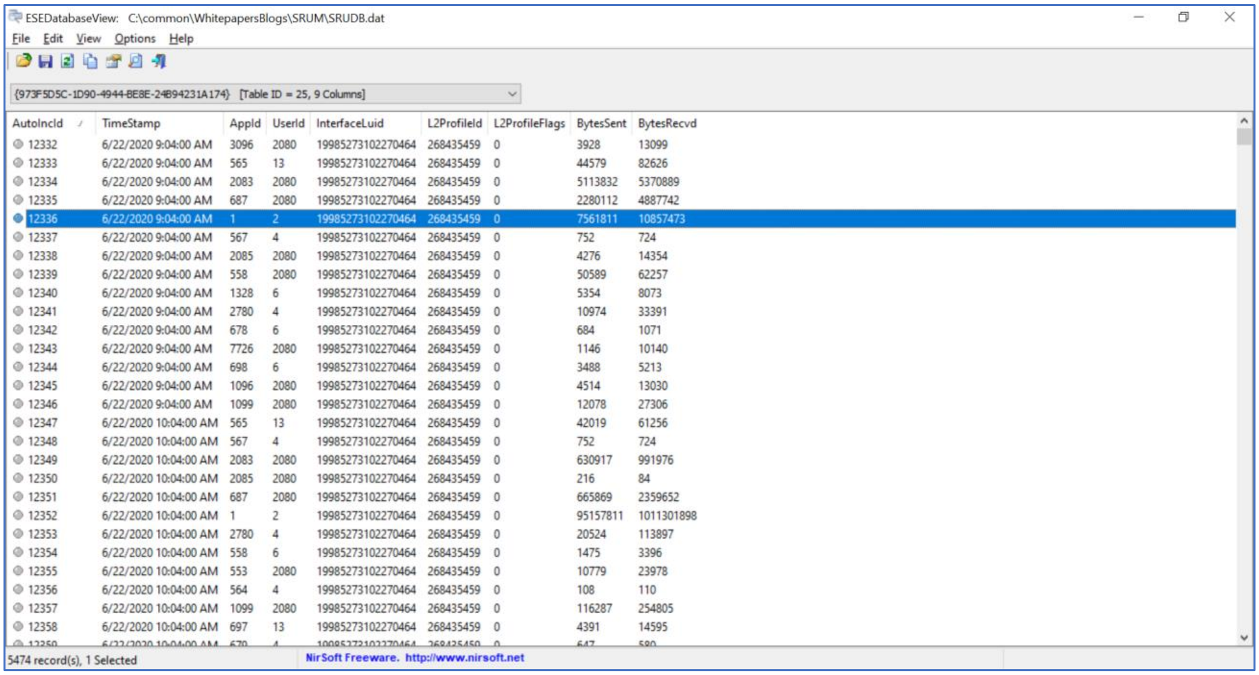
Listed: 38 Selected: 1 C:\/NONAME [NTFS]/[root]/Windows/System32/sru/SRUDB.dat

Figure 1: Location of SRUDB.dat

STORAGE FORMAT

As the extension suggests, SRUM data are stored in a database file. It is ESE DB – Extensible Storage Engine format, proprietary Microsoft technology, which is core of MS Exchange, Windows Search or Active Directory operations, to name a few.

To open ESE database we can use the NirSoft tool called ESEDatabaseView.



ESEDatabaseView: C:\common\WhitepapersBlogs\SRUM\SRUDB.dat
File Edit View Options Help
{973F5D5C-1D90-4944-BE8E-24B94231A174} [Table ID = 25, 9 Columns]

AutoIncId	TimeStamp	AppId	UserId	InterfaceLuid	L2ProfileId	L2ProfileFlags	BytesSent	BytesRecvd
12332	6/22/2020 9:04:00 AM	3096	2080	19985273102270464	268435459	0	3928	13099
12333	6/22/2020 9:04:00 AM	565	13	19985273102270464	268435459	0	44579	82626
12334	6/22/2020 9:04:00 AM	2083	2080	19985273102270464	268435459	0	5113832	5370889
12335	6/22/2020 9:04:00 AM	687	2080	19985273102270464	268435459	0	2280112	4887742
12336	6/22/2020 9:04:00 AM	1	2	19985273102270464	268435459	0	7561811	10857473
12337	6/22/2020 9:04:00 AM	567	4	19985273102270464	268435459	0	752	724
12338	6/22/2020 9:04:00 AM	2085	2080	19985273102270464	268435459	0	4276	14354
12339	6/22/2020 9:04:00 AM	558	2080	19985273102270464	268435459	0	50589	62257
12340	6/22/2020 9:04:00 AM	1328	6	19985273102270464	268435459	0	5354	8073
12341	6/22/2020 9:04:00 AM	2780	4	19985273102270464	268435459	0	10974	33391
12342	6/22/2020 9:04:00 AM	678	6	19985273102270464	268435459	0	684	1071
12343	6/22/2020 9:04:00 AM	7726	2080	19985273102270464	268435459	0	1146	10140
12344	6/22/2020 9:04:00 AM	698	6	19985273102270464	268435459	0	3488	5213
12345	6/22/2020 9:04:00 AM	1096	2080	19985273102270464	268435459	0	4514	13030
12346	6/22/2020 9:04:00 AM	1099	2080	19985273102270464	268435459	0	12078	27306
12347	6/22/2020 10:04:00 AM	565	13	19985273102270464	268435459	0	42019	61256
12348	6/22/2020 10:04:00 AM	567	4	19985273102270464	268435459	0	752	724
12349	6/22/2020 10:04:00 AM	2083	2080	19985273102270464	268435459	0	630917	991976
12350	6/22/2020 10:04:00 AM	2085	2080	19985273102270464	268435459	0	216	84
12351	6/22/2020 10:04:00 AM	687	2080	19985273102270464	268435459	0	665869	2359652
12352	6/22/2020 10:04:00 AM	1	2	19985273102270464	268435459	0	95157811	1011301898
12353	6/22/2020 10:04:00 AM	2780	4	19985273102270464	268435459	0	20524	113897
12354	6/22/2020 10:04:00 AM	558	6	19985273102270464	268435459	0	1475	3396
12355	6/22/2020 10:04:00 AM	553	2080	19985273102270464	268435459	0	10779	23978
12356	6/22/2020 10:04:00 AM	564	4	19985273102270464	268435459	0	108	110
12357	6/22/2020 10:04:00 AM	1099	2080	19985273102270464	268435459	0	116287	254805
12358	6/22/2020 10:04:00 AM	697	13	19985273102270464	268435459	0	4391	14595
12359	6/22/2020 10:04:00 AM	670	4	19985273102270464	268435459	0	647	580

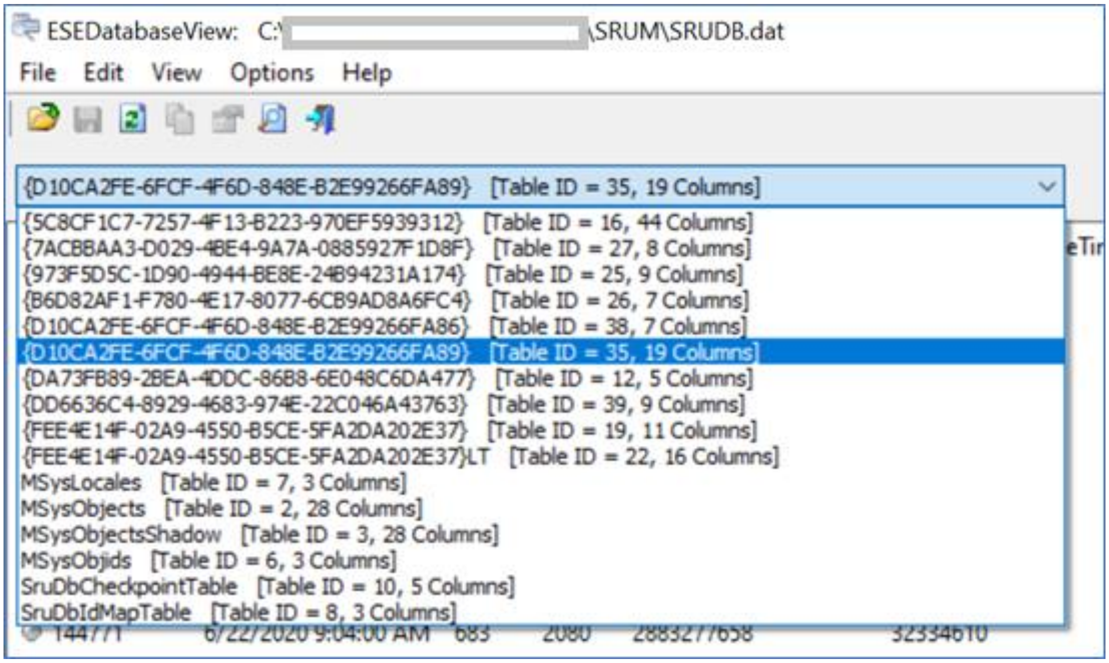
5474 record(s), 1 Selected

NirSoft Freeware. http://www.nirsoft.net

Figure 2: SRUMBD.dat opened in ESEDBView.exe.

In Figure 2, we have opened one of the database tables. Its name – {973F5D5C-…} – has the structure of Windows GUID, Globally Unique IDentifier. When it comes to SRUM, tables are also known by their ID, which is 35 in this case. Column names suggest that this table holds data related to networking activity – BytesSent, BytesReceived – but so far, we do not know the application which produced each of the entries. More parsing to come!

In Figure 3, see other tables stored in SRUM database.



ESEDatabaseView: C:\[redacted]\SRUM\SRUDB.dat
File Edit View Options Help

```
{D10CA2FE-6FCF-4F6D-848E-B2E99266FA89}   [Table ID = 35, 19 Columns]
{5C8CF1C7-7257-4F13-B223-970EF5939312}   [Table ID = 16, 44 Columns]
{7ACBBAA3-D029-4BE4-9A7A-0885927F1D8F}   [Table ID = 27, 8 Columns]
{973F5D5C-1D90-4944-BE8E-24B94231A174}   [Table ID = 25, 9 Columns]
{B6D82AF1-F780-4E17-8077-6CB9AD8A6FC4}   [Table ID = 26, 7 Columns]
{D10CA2FE-6FCF-4F6D-848E-B2E99266FA86}   [Table ID = 38, 7 Columns]
{D10CA2FE-6FCF-4F6D-848E-B2E99266FA89}   [Table ID = 35, 19 Columns]
{DA73FB89-2BEA-4DDC-86B8-6E048C6DA477}   [Table ID = 12, 5 Columns]
{DD6636C4-8929-4683-974E-22C046A43763}   [Table ID = 39, 9 Columns]
{FEE4E14F-02A9-4550-B5CE-5FA2DA202E37}   [Table ID = 19, 11 Columns]
{FEE4E14F-02A9-4550-B5CE-5FA2DA202E37}LT   [Table ID = 22, 16 Columns]
MSysLocales   [Table ID = 7, 3 Columns]
MSysObjects   [Table ID = 2, 28 Columns]
MSysObjectsShadow   [Table ID = 3, 28 Columns]
MSysObjids   [Table ID = 6, 3 Columns]
SruDbCheckpointTable   [Table ID = 10, 5 Columns]
SruDbIdMapTable   [Table ID = 8, 3 Columns]
144771   6/22/2020 9:04:00 AM   683   2080   2883277658   32334610
```

Figure 3: Detail of tables in SRUMDB.dat.

FORENSIC VALUE

Without going too deep, let us introduce the following tables which can be of the greatest use in forensic investigations:

1. Network Connectivity - {DD6636C4-8929-4683-974E-22C046A43763}

In this table, information about network connectivity is stored. Valuable data to get from here are

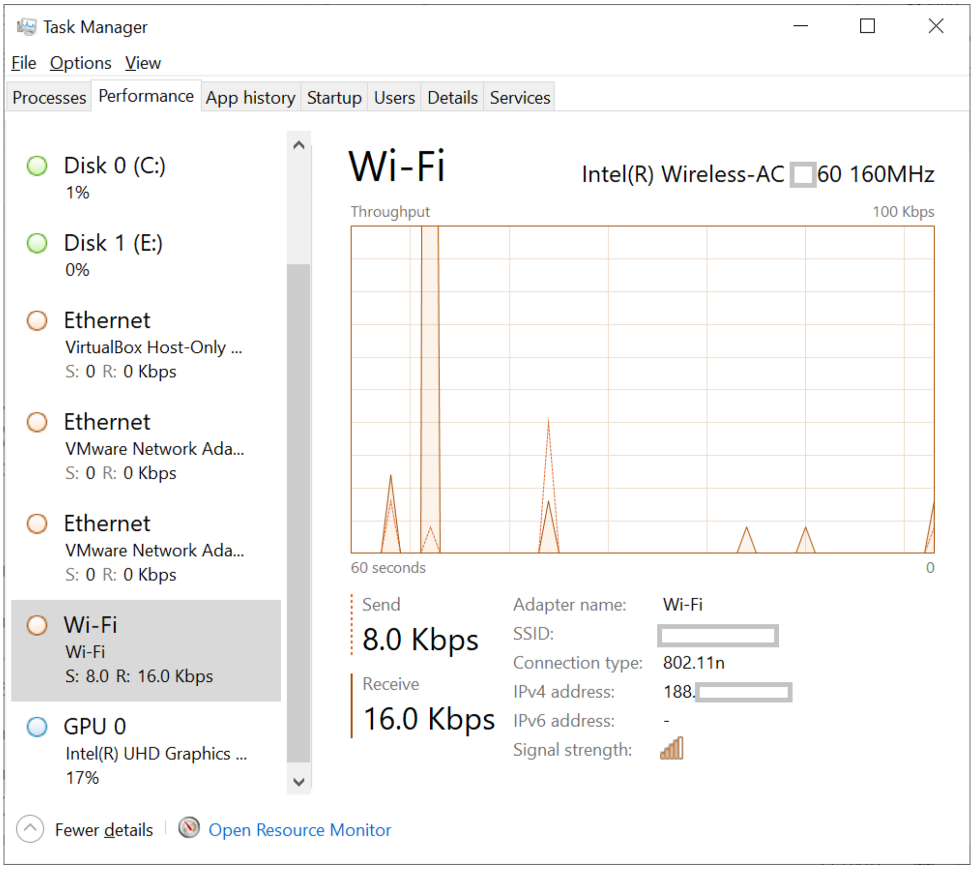
- Interface Type & ID
- Network Profile ID
- Time when the connection was established
- Duration of connection – how long was interface connected to specific network

2. Network Data usage {973F5D5C-1D90-4944-BE8E-24B94231A174}

More precise data about which application and user were using network can be gathered from here:

- Application/Service/App consuming data (User SID)
- Bytes Uploaded & Downloaded
- Interface Type & ID
- Network Profile ID

However, do not expect to see any endpoint information (outbound IP address or port numbers) or any details of what data have been transferred (what was exfiltrated or downloaded).



Task Manager

File Options View

Processes | Performance | App history | Startup | Users | Details | Services

Disk 0 (C:)
1%

Disk 1 (E:)
0%

Ethernet
VirtualBox Host-Only ...
S: 0 R: 0 Kbps

Ethernet
VMware Network Ada...
S: 0 R: 0 Kbps

Ethernet
VMware Network Ada...
S: 0 R: 0 Kbps

Wi-Fi
Wi-Fi
S: 8.0 R: 16.0 Kbps

GPU 0
Intel(R) UHD Graphics ...
17%

Wi-Fi

Intel(R) Wireless-AC [redacted]60 160MHz

Send
8.0 Kbps

Receive
16.0 Kbps

Adapter name: Wi-Fi
SSID: [redacted]
Connection type: 802.11n
IPv4 address: 188.[redacted]
IPv6 address: -
Signal strength:

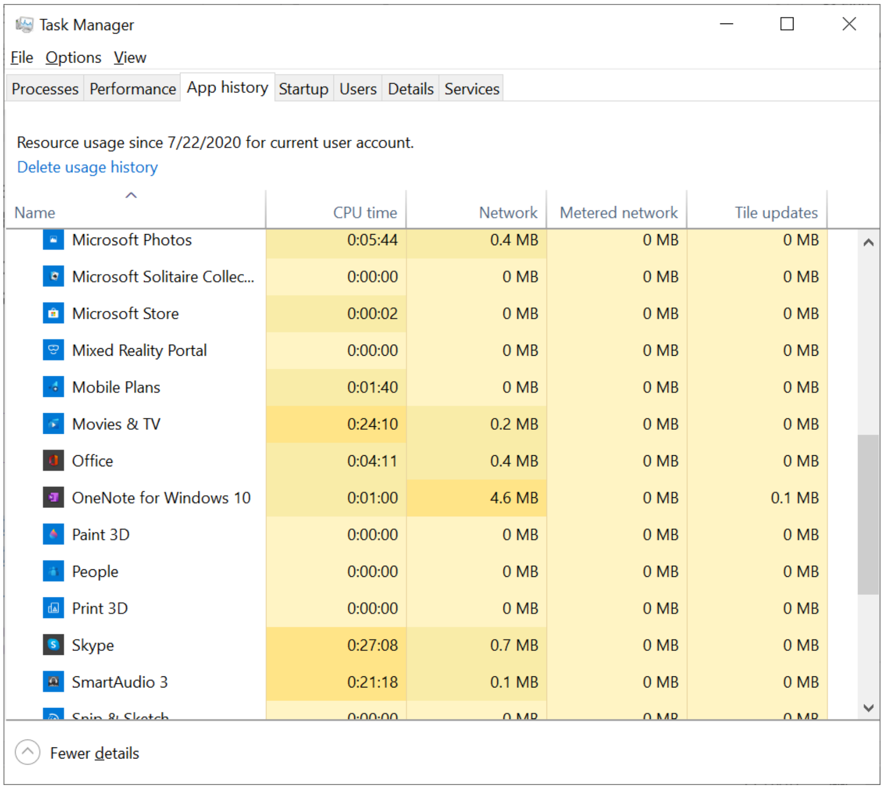
Fewer details | Open Resource Monitor

Figure 4: Networking in TaskManager. Data are stored in SRUMDB.dat.

3. Application Resource usage - {D10CA2FE-6FCF-4F6D-848E-B2E99266FA89}

Here we come to that tab of Task Manager which lists all those nice apps running on your PC, along with details on what resources they consume from your device. What's more, it allows for mapping user SID to the program running on the system:

- CPU cycles
- Context switches
- I/O bytes read/written
- Number of read operations
- Number of write operations
- Number of flushes
- Time the app was in foreground/background
- SID of user who launched program


Task Manager
File Options View
Processes Performance App history Startup Users Details Services

Resource usage since 7/22/2020 for current user account.
Delete usage history

Name	CPU time	Network	Metered network	Tile updates
Microsoft Photos	0:05:44	0.4 MB	0 MB	0 MB
Microsoft Solitaire Collec...	0:00:00	0 MB	0 MB	0 MB
Microsoft Store	0:00:02	0 MB	0 MB	0 MB
Mixed Reality Portal	0:00:00	0 MB	0 MB	0 MB
Mobile Plans	0:01:40	0 MB	0 MB	0 MB
Movies & TV	0:24:10	0.2 MB	0 MB	0 MB
Office	0:04:11	0.4 MB	0 MB	0 MB
OneNote for Windows 10	0:01:00	4.6 MB	0 MB	0.1 MB
Paint 3D	0:00:00	0 MB	0 MB	0 MB
People	0:00:00	0 MB	0 MB	0 MB
Print 3D	0:00:00	0 MB	0 MB	0 MB
Skype	0:27:08	0.7 MB	0 MB	0 MB
SmartAudio 3	0:21:18	0.1 MB	0 MB	0 MB
Snip & Sketch	0:00:00	0 MB	0 MB	0 MB

Figure 5: Task Manager and data from SRUM.DB

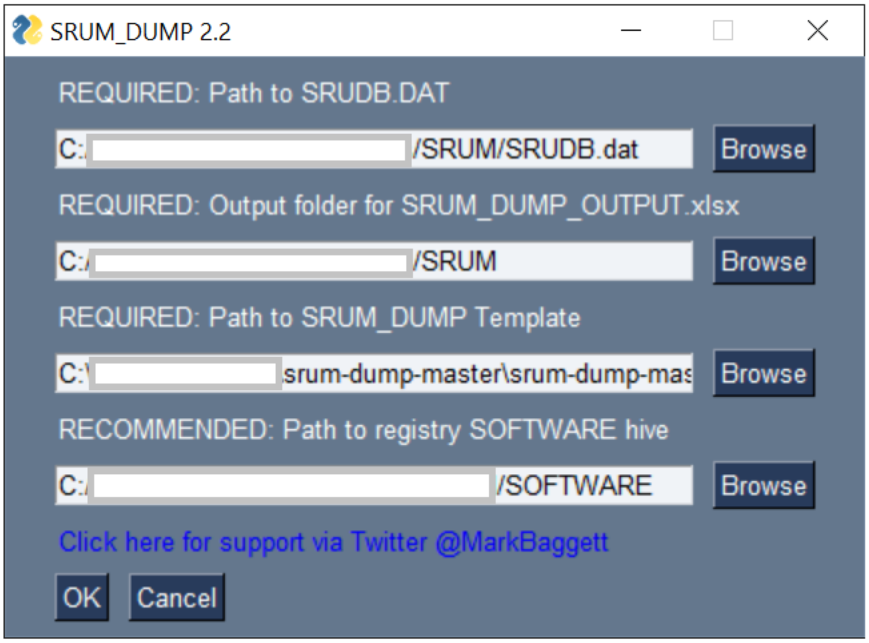
Other well-known tables that we will not detail in this article are

4. Windows Push Notification data - {D10CA2FE-6FCF-4F6D-848E-B2E99266FA86}
5. Energy usage data - {FEE4E14F-02A9-4550-B5CE-5FA2DA202E37} {FEE4E14F-02A9-4550-B5CE-5FA2DA202E37} LT

PARSING OPTIONS

There are multiple freeware and commercial forensics suits providing the ability to get data from SRUM DB. One of well-known and free tools to perform this task is srum-dump (see Resources for GitHub link).

Download the utility from Github and analyze either a live system (run with administrative privileges in that case) or point to an extracted DB file. To parse data, it is advised to also point the tool to the SOFTWARE registry hive (stored in Windows\System32\config directory).


SRUM_DUMP 2.2

REQUIRED: Path to SRUDB.DAT

C:[redacted]/SRUM/SRUDB.dat

REQUIRED: Output folder for SRUM_DUMP_OUTPUT.xlsx

C:/[redacted]/SRUM

REQUIRED: Path to SRUM_DUMP Template

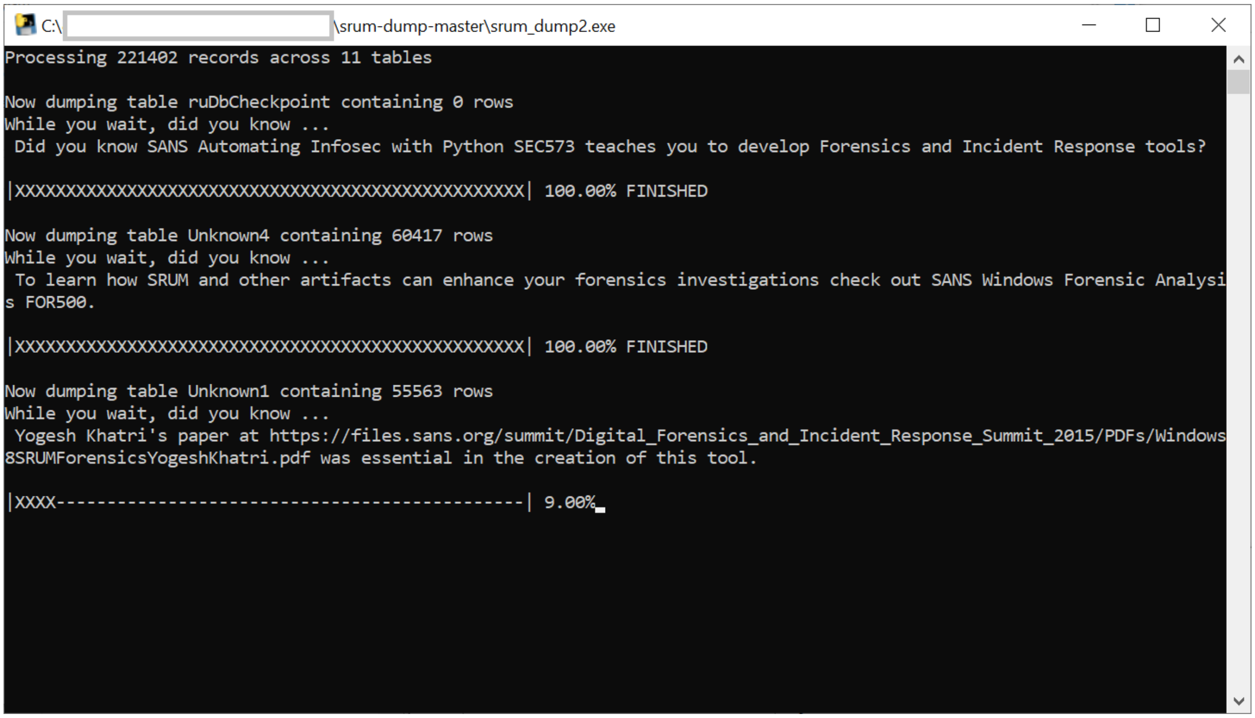
C:\[redacted]srum-dump-master\srum-dump-mas

RECOMMENDED: Path to registry SOFTWARE hive

C:/[redacted]/SOFTWARE

Figure 6: srum-dump ready to run.

After all is set, press Enter and enjoy tips which are showed on the screen while parsing is in progress 😊

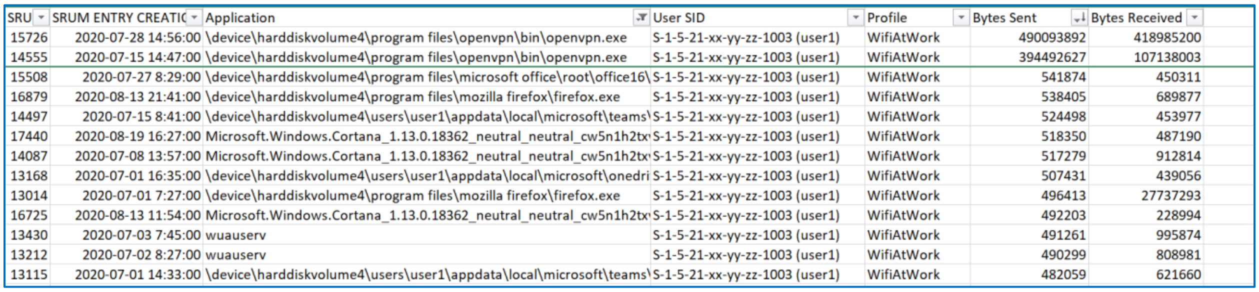

C:\[redacted]\srum-dump-master\srum_dump2.exe

```
Processing 221402 records across 11 tables

Now dumping table ruDbCheckpoint containing 0 rows
While you wait, did you know ...
 Did you know SANS Automating Infosec with Python SEC573 teaches you to develop Forensics and Incident Response tools?

|XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX| 100.00% FINISHED

Now dumping table Unknown4 containing 60417 rows
While you wait, did you know ...
 To learn how SRUM and other artifacts can enhance your forensics investigations check out SANS Windows Forensic Analysi
s FOR500.

|XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX| 100.00% FINISHED

Now dumping table Unknown1 containing 55563 rows
While you wait, did you know ...
 Yogesh Khatri's paper at https://files.sans.org/summit/Digital_Forensics_and_Incident_Response_Summit_2015/PDFs/Windows
8SRUMForensicsYogeshKhatri.pdf was essential in the creation of this tool.

|XXXX----------------------------------------------| 9.00%_
```

Figure 7: srum-dump in action.

After the processing is complete and an Excel spreadsheet with parsed data is written to the desired output location, we can review the results. In the example below, we sorted Network Usage table entries based on bytes sent by each program. Selected top-senders are VPN service, which is expected while working from home (new standard of these days 😉). Other applications do not seem to be out of place either – Teams, OneDrive, Updates service. However, if we were investigating data exfiltration, we could focus on the suspected time period and check if there was any exceptional data flow or unusual application causing large data transfer.

In other cases, evidence of a user accessing an unknown network can be crucial for investigation. Profile column gives us the SSID of the connected WiFi network or wired network identifier. User SIDs are mapped to usernames, allowing for immediate identification of the user spawning the process with corresponding network usage.



SRU	SRUM ENTRY CREATIC	Application	User SID	Profile	Bytes Sent	Bytes Received
15726	2020-07-28 14:56:00	\device\harddiskvolume4\program files\openvpn\bin\openvpn.exe	S-1-5-21-xx-yy-zz-1003 (user1)	WifiAtWork	490093892	418985200
14555	2020-07-15 14:47:00	\device\harddiskvolume4\program files\openvpn\bin\openvpn.exe	S-1-5-21-xx-yy-zz-1003 (user1)	WifiAtWork	394492627	107138003
15508	2020-07-27 8:29:00	\device\harddiskvolume4\program files\microsoft office\root\office16\	S-1-5-21-xx-yy-zz-1003 (user1)	WifiAtWork	541874	450311
16879	2020-08-13 21:41:00	\device\harddiskvolume4\program files\mozilla firefox\firefox.exe	S-1-5-21-xx-yy-zz-1003 (user1)	WifiAtWork	538405	689877
14497	2020-07-15 8:41:00	\device\harddiskvolume4\users\user1\appdata\local\microsoft\teams\	S-1-5-21-xx-yy-zz-1003 (user1)	WifiAtWork	524498	453977
17440	2020-08-19 16:27:00	Microsoft.Windows.Cortana_1.13.0.18362_neutral_neutral_cw5n1h2tx	S-1-5-21-xx-yy-zz-1003 (user1)	WifiAtWork	518350	487190
14087	2020-07-08 13:57:00	Microsoft.Windows.Cortana_1.13.0.18362_neutral_neutral_cw5n1h2tx	S-1-5-21-xx-yy-zz-1003 (user1)	WifiAtWork	517279	912814
13168	2020-07-01 16:35:00	\device\harddiskvolume4\users\user1\appdata\local\microsoft\onedri	S-1-5-21-xx-yy-zz-1003 (user1)	WifiAtWork	507431	439056
13014	2020-07-01 7:27:00	\device\harddiskvolume4\program files\mozilla firefox\firefox.exe	S-1-5-21-xx-yy-zz-1003 (user1)	WifiAtWork	496413	27737293
16725	2020-08-13 11:54:00	Microsoft.Windows.Cortana_1.13.0.18362_neutral_neutral_cw5n1h2tx	S-1-5-21-xx-yy-zz-1003 (user1)	WifiAtWork	492203	228994
13430	2020-07-03 7:45:00	wuauserv	S-1-5-21-xx-yy-zz-1003 (user1)	WifiAtWork	491261	995874
13212	2020-07-02 8:27:00	wuauserv	S-1-5-21-xx-yy-zz-1003 (user1)	WifiAtWork	490299	808981
13115	2020-07-01 14:33:00	\device\harddiskvolume4\users\user1\appdata\local\microsoft\teams\	S-1-5-21-xx-yy-zz-1003 (user1)	WifiAtWork	482059	621660

Figure 8: Table Network Usage parsed.

LIMITATIONS

As all (or most??) artifacts, SRUMDB.dat also has its limits that may hinder our investigation.

First of all, the database only stores information from the past 30 days. In the case that your investigation goes further back, SRUM will not yield useful information, although there is still a chance that we can find older SRUM databases in volume shadow copies and stretch the covered time period.

Another limitation to keep in mind is that the database gets updated every 60 minutes – all entries from a 1h period will have the same timestamp, or the time when the system was properly shut down. In case that 'dirty' shutdown occurred, you may need to repair ESE database with Microsoft built-in utilities – esentutl. To determine if the database is healthy, issue command esentutl /mh SRUMDB.dat before attempting to parse it.

RESOURCES

1. https://www.sciencedirect.com/science/article/abs/pii/S1742287615000031
2. https://troopers.de/downloads/troopers19/TROOPERS19_AD_Beyond_Windows_Forensics.pdf
3. https://www.hecfblog.com/2019/01/daily-blog-595-solution-saturday-11219.html
4. https://github.com/MarkBaggett/srum-dump