






# Can you track processes accessing the camera and microphone?


 svch0st · Follow  
3 min read · Jun 7, 2020

 --

 2







In certain investigations, it may arise that you need to find the following:

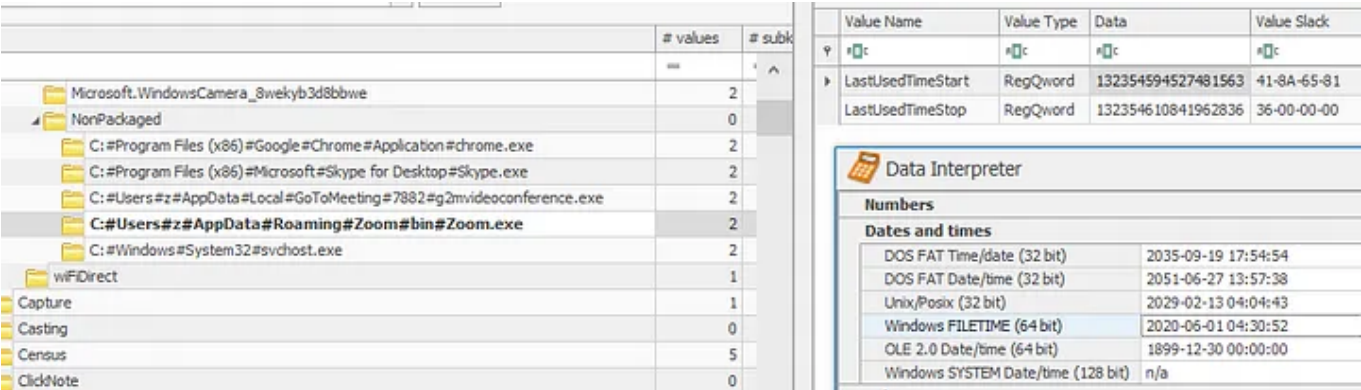
- What process was using the camera or microphone?
- When was the last session?
- How long was that session?

Using the contents of the following reg keys, you can to determine when and how long a process had access to privacy protected resources. These resources include the microphone, webcam, bluetooth, location, contacts and more. For this blog, I will focus on the microphone and webcam as an example.

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\CapabilityAccessManager\ConsentStore\webcam\  
  
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\CapabilityAccessManager\ConsentStore\microphone\  
  
HKEY_USERS\*\SOFTWARE\Microsoft\Windows\CurrentVersion\CapabilityAccessManager\ConsentStore\webcam\  
  
HKEY_USERS\*\SOFTWARE\Microsoft\Windows\CurrentVersion\CapabilityAccessManager\ConsentStore\microphone\
```

*EDIT 2022/01/08: Some further testing was done by Phill Moore and observed the keys were seen in the user hives instead of SOFTWARE on Win10 20H2 - <https://thinkdfir.com/2022/01/04/i-can-see-and-hear-you-seeing-and-hearing-me/>*

Below is an example of the typical entries in the `webcam` directory. There are several entries including Microsoft and non-Microsoft applications



Microsoft applications are stored in as child keys but non-Microsoft applications (which are of the most interest) are stored in the `NonPackaged` child key.

Within the `NonPackaged` directory, you can see that the name of the keys are the full path of an executable with `#` replacing `\`.

Each entry has two values, `LastUsedTimeStart` and `LastUsedTimeStop`, with the timestamps in FILETIME format.

From the example above, you are able to determine, **Zoom.exe** had access to **my webcam for 27.2 minutes** (between 2020/06/01 04:30:52 UTC and 2020/06/01 04:58:04 UTC).

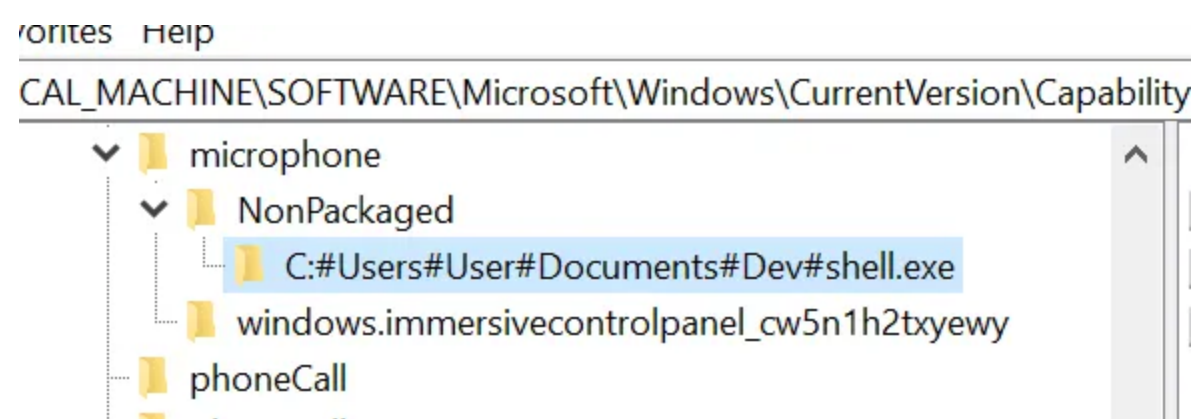
Whether you are looking at what processes had access to a webcam or even trying to prove long a user’s conversation may have been, this is a great source of information.

### Testing RAT-like behaviour

I needed to test if this also applied to more malicious methods of accessing the microphone. I used a meterpreter post-exploit module to record audio from Windows VM.

```
msf5 post(multi/manage/record_mic) > run
[*] 192.168.171.130 - 20% ...
[*] 192.168.171.130 - 40% ...
[*] 192.168.171.130 - 60% ...
[*] 192.168.171.130 - 80% ...
[*] 192.168.171.130 - 100% ...
[*] 192.168.171.130 - Audio size: (55169 bytes)
[+] 192.168.171.130 - Audio recording saved: /root/.msf4/loot/20200608024520_de
[*] Post module execution completed
```

As soon as I ran the recording command, a new entry was populated from where my meterpreter shell was executed. Pretty cool!



## Monitoring

If we wanted to track all sessions (not just the last), it is easy with Sysmon. If you are running something like the Swift on Security configuration, you will need to add an inclusion line for event id 12,13 and 14 (Registry modification):

```
<TargetObject
condition="contains">SOFTWARE\Microsoft\Windows\CurrentVersion\Cap
abilityAccessManager\ConsentStore\</TargetObject> <!-- When a
process accesses bluetooth, location, webcam, microphone etc, the
timestamps of last access are updated here. HKLM and HKCU -->
```

After updating your configuration, a Sysmon event will now be created when the registry keys are created or updated. Below is the LastUsedTime key being updated for Skype.exe accessing my microphone in the Sysmon event log.

Type	Date	Time	Event	Source	Category	User
Information	7/06/2020	12:09:03 PM	13	Microsoft-Windows-Sy	Registry value set (rule: RegistryEvent)	\SYSTEM
Information	7/06/2020	12:08:53 PM	13	Microsoft-Windows-Sy	Registry value set (rule: RegistryEvent)	\SYSTEM
Information	7/06/2020	12:08:50 PM	13	Microsoft-Windows-Sy	Registry value set (rule: RegistryEvent)	\SYSTEM
Information	7/06/2020	12:08:50 PM	13	Microsoft-Windows-Sy	Registry value set (rule: RegistryEvent)	\SYSTEM
Information	7/06/2020	12:08:50 PM	13	Microsoft-Windows-Sy	Registry value set (rule: RegistryEvent)	\SYSTEM
Information	7/06/2020	12:08:50 PM	13	Microsoft-Windows-Sy	Registry value set (rule: RegistryEvent)	\SYSTEM

Description

Registry value set:  
RuleName:  
EventType: SetValue  
UtcTime: 2020-06-07 02:08:50.730  
ProcessGuid: {ED0FE286-2FC7-5EDC-0000-001039650200}  
ProcessId: 3068  
Image: C:\windows\system32\svchost.exe  
TargetObject: HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\CapabilityAccessManager\ConsentStore\microphone\NonPackaged\C:#Program Files (x86)\Microsoft#Skype for Desktop#Skype.exe\LastUsedTimeStart  
Details: QWORD (0x01d63c70-0x96294c6e)

The timestamp in the log are still in hex which needs to be converted to decimal then to a human readable timestamp, however the timestamp of the event itself is also very accurate.

## Conclusion

What spurred this off is when I came across this page in the settings, and it got me thinking on where this data is stored.

It will be interesting if there are other places that track historical sessions without the use of monitoring. This would be more valuable to forensic analysts that don't always have nice logs.

Further research also could be done to identify which device the process is accessing (front camera, USB camera etc). I would also like to explore if this method catches more covert RAT malware.

Thanks for reading,

Zach

- Forensics
- Dfir
- Incident Response

 --  2



Written by svch0st

318 Followers

Follow

