Saturday, 2 March 2019

Threat Hunting #25 - Scheduled Tasks for Persistence and/or Remote Execution

The Task Scheduler enables you to automatically perform routine tasks on a chosen computer. The Task Scheduler does this by monitoring whatever criteria you choose to initiate the tasks (referred to as triggers) and then executing the tasks (Action) when the criteria is met (user logon, system startup, event log triggered, fixed execution time reached etc.).

Attackers (ab)uses Task Scheduler to guarantee persistence and/or remote execution. In this post we will be covering some of the suspicious scheduled tasks related behaviors that you can start hunting for:

A) Scheduled Task running programs from suspicious locations or scripting utilities:

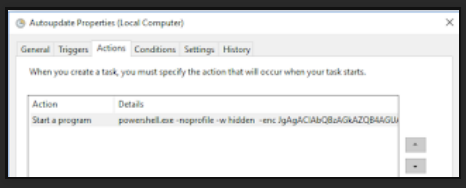
Tasks running scripts or programs from temp directories or insecure location (writable by any user) are a good indicator for initial (malware just landed) execution/persistence via scheduled tasks, includes but not limited to the following locations:

- c:\users*
- c:\programdata*
- c:\windows\temp*

For scripting utilities pay attention to tasks with action set to one of the following (inspect the arguments if they point to the above insecure commonly used paths):

- cscript.exe
- wscript.exe
- rundll32.exe
- regsvr32.exe
- wmic.exe
- cmd.exe
- mshta.exe
- powershell.exe

Example of similar malicious entry using powerlshell.exe and obfuscated arguments:



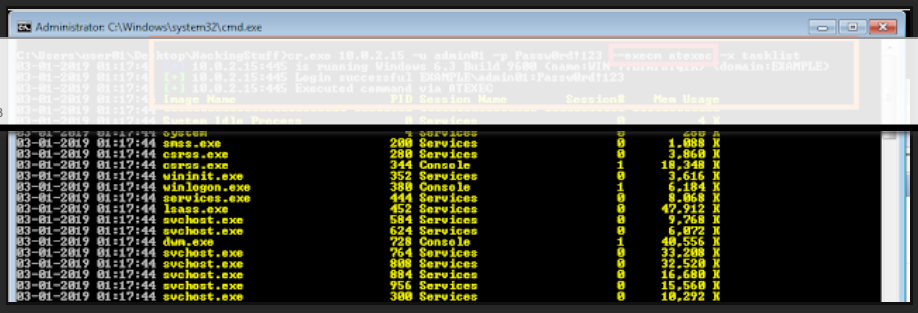
B) Remote Task creation using ATSVC named pipe or the deprecated AT.exe cmdlet:

Using At.exe command or directly interacting with the ATSVC named API to create remote scheduled Job will leave several traces (Events 106, 4698, file write to c:\windows\tasks\At*), but all of those indicators apply also to a local scheduled task, in this case we are more interested by the remote one.

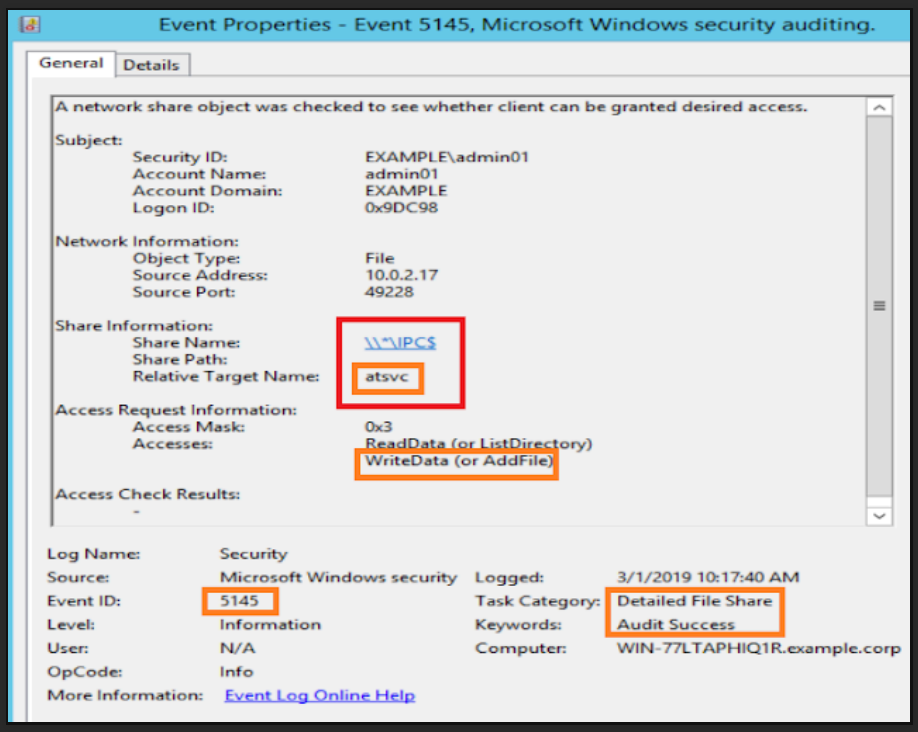
Just as an example, we will be using crackmap (post exploitation toolkit, very powerful hacking tool) and opt for ATEXEC as a remote execution method (which interact with ATSVC named pipe):

Blog Archive

- ► 2022 (2)
- ► 2021 (3)
- ► 2020 (4)
- ▼ 2019 (39)
  - ► November (2)
  - ► July (1)
  - ► April (3)
  - ▼ March (7)
    - [Initial Access & execution] - Evidences for files...
    - An overview of Windows EventID 4648 - Logon with e...
    - Initial Access & Execution - Windows default trace...
    - Brute-forcing Password Protected Office Files - Fo...
    - How to hunt for processes starting from Run RunOnc...
    - Threat Hunting #26 - Remote Windows Service Creati...
    - Threat Hunting #25 - Scheduled Tasks for Persisten...
  - ► February (26)



This results in the following key indicator:



As you can see above, we can hunt for it using only EventId 5145 and ShareName:*\IPC$ and RelativeTargetName equal to atsvc named pipe, below a SIGMA rule example:

win_atsvc_share_access.yml

```yaml
title: Remote Task Creation via ATSVC named pipe
description: Detects remote task creation via at.exe or API interacting with ATSVC namedpipe
tags:
    - attack.lateral_movement
    - attack.persistence
    - attack.T1053
status: experimental
author: Samir Bousseaden
logsource:
    product: windows
    service: security
    description: 'The advanced audit policy setting "Object Access > Audit Detailed File Share" must
detection:
    selection:
        EventID: 5145
        ShareName: \\*\IPC$
        RelativeTargetName: atsvc
    condition: selection
falsepositives:
    - pentesting
level: medium
```

And an example of a Splunk query:

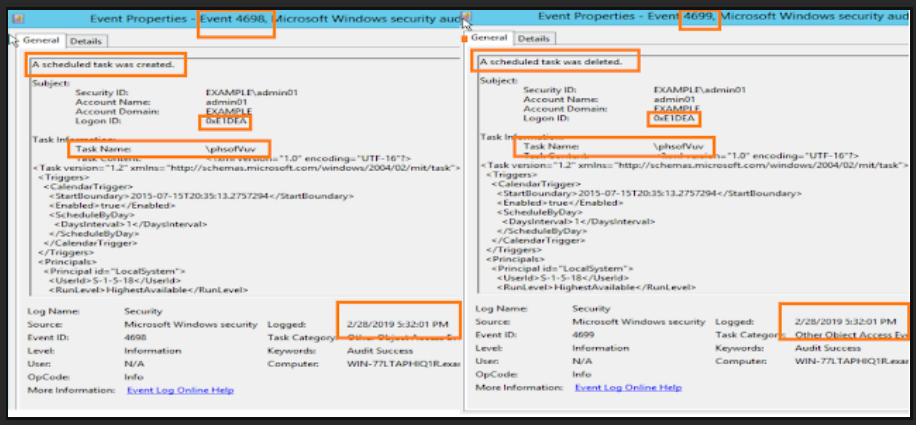
(EventID="5145" ShareName="*\\IPC$" RelativeTargetName="atsvc")

C) Tasks with Short LifeTime:

the task scheduler. We will need the following two events:

- 4698 - A Scheduled Task was created
- 4699 - A Scheduled Task was deleted

Below an example of malicious task with less than 1 min life time:



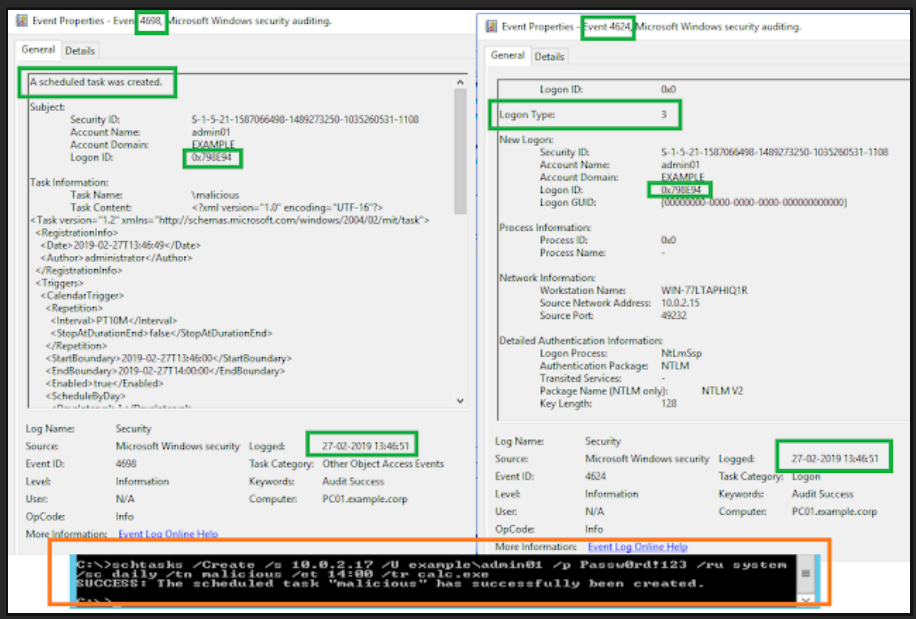
Detection Logic: If 4698 followed by 4699 with same LogonID and TaskName within 1min -> alert("Suspicious Scheduled Task - Short Life Time")

D) Remote Task Creation:

Remote scheduled tasks are not necessarily malicious, but it's worth checking and verifying their legitimacy. For this use case we will need two security events from the target machine:

- 4624 - An account was successfully logged on (with Logon Type =3 -> Network)
- 4698 - A scheduled task was created

Below an example of the observed logs:



Detection Logic:

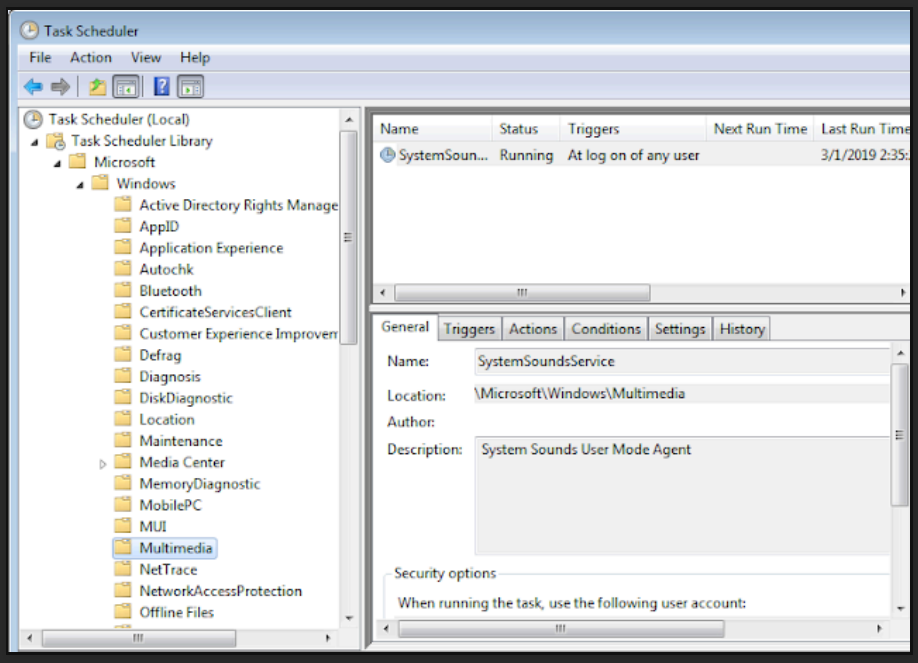
If (event.id=4624 and event.logontype=3 followed by event.id=4698) and same event.logonid within 1min --> Alert ("Remote Scheduled Task Created")

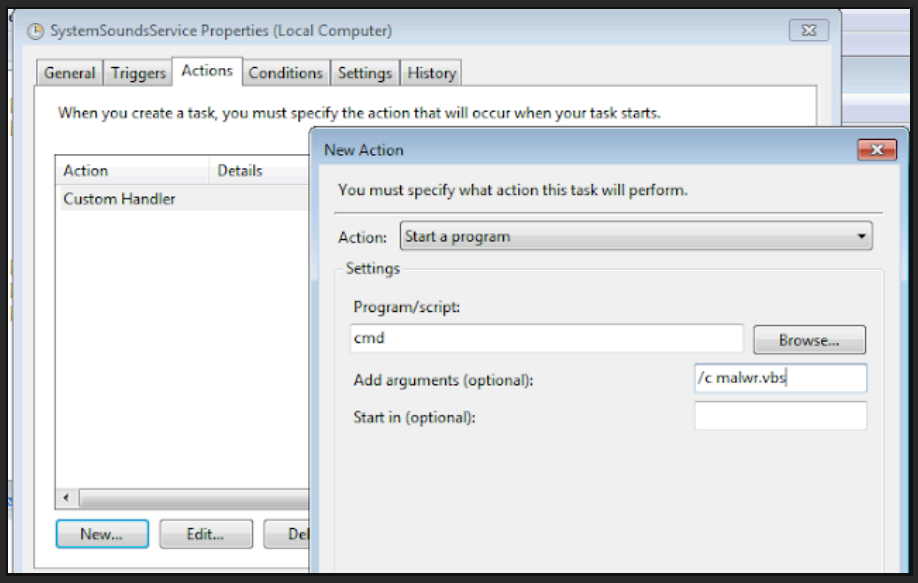
E) Modification of an existing Windows Default Scheduled Task:

From a malicious actor perspective, adding an extra action to an existing windows default scheduled task (as shown below) has the following advantages:

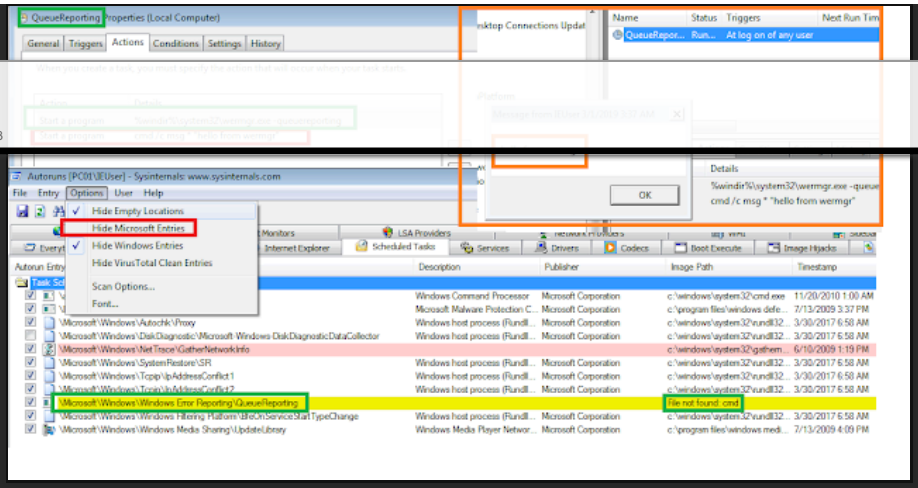
- No New Scheduled Task Creation Event is triggered (EventIDs: 106 & 4698)
- Rogue task mixes with default windows task name and triggers (less suspicion)

Any (including the ones with Action set to custom handler) Windows default scheduled task that runs for example at any user logon and with status ready can be abused by adding an extra action:

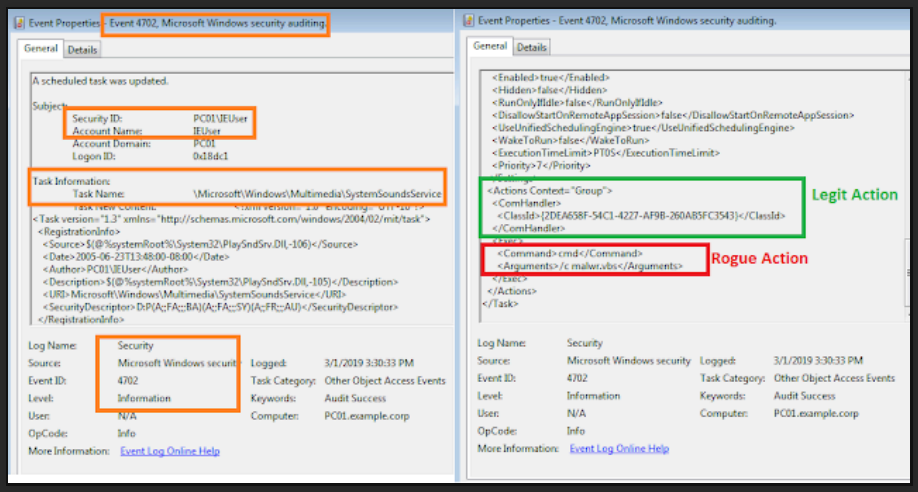




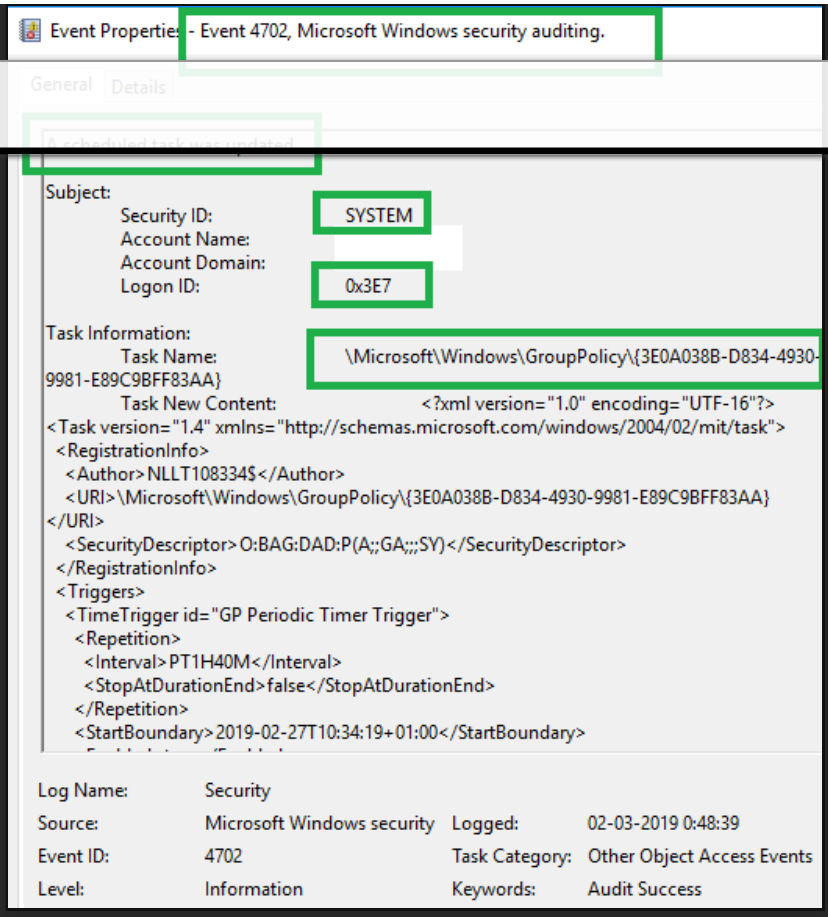
And when checking with Autorunsc, this is what you will see:



The only relevant indicator we've observed is event **4702 "Task Updated"** indicating the update of a Microsoft Windows Task and source account name is different than the local System account (which is abnormal):



For normal Windows default tasks updates 4702 you will see something like this:



F) Modification of the program run by a Windows Default Scheduled Task:

Files of interest for hijacking existing default windows 7 and 10 scheduled tasks (Action trigger is set to system startup or any user logon or every day at working hours):

- %SystemRoot%\System32\aitagent.exe
- %windir%\system32\compattel\DiagTrackRunner.exe
- %windir%\system32\CompatTelRunner.exe
- acproxy.dll
- %SystemRoot%\System32\wsqmcons.exe
- %windir%\system32\lpremove.exe
- srrstr.dll,ExecuteScheduledSPPCreation
- %windir%\system32\wermgr.exe
- %systemroot%\System32\sdclt.exe
- %windir%\system32\appidcertstorecheck.exe
- %windir%\system32\AppHostRegistrationVerifier.exe
- "C:\Windows\System32\MicTray64.exe"
- %systemroot%\system32\usoclient.exe
- %SystemRoot%\System32\dsregcmd.exe
- %systemroot%\System32\sihclient.exe

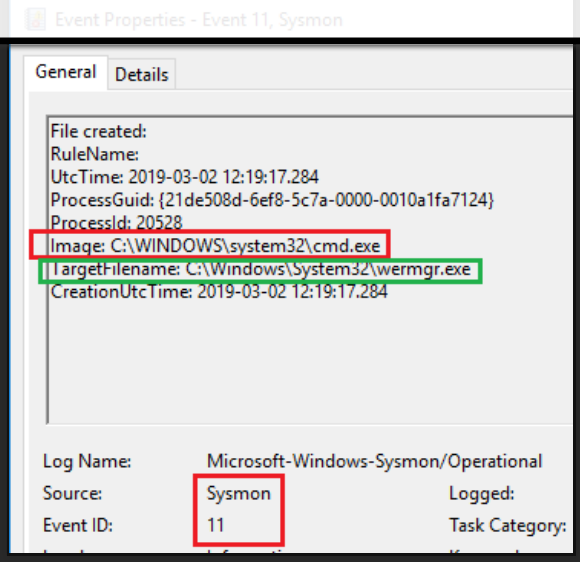
Action set to Custom Handler and triggered at user logon or system startup :

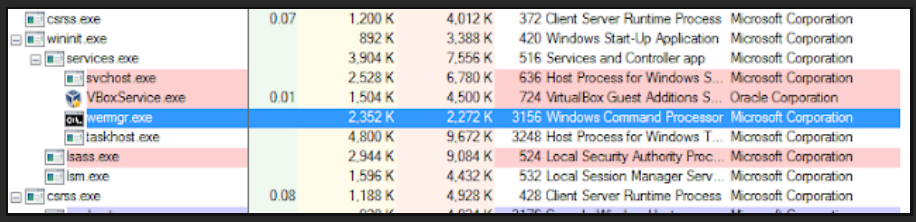
- system32\dimsjob.dll
- Racengn.dll
- HotstartUserAgent.dll
- MsCtfMonitor.dll
- PlaySndSrv.dll

Common third party tasks's programs that are of interest:

C:\Program Files (x86)\Common Files\Adobe\ARM\1.0\AdobeARM.exe
C:\Program Files (x86)\Google\Update\GoogleUpdate.exe

Any modification to those files must be reviewed using Sysmon EID 11 (FileCreate - include them in your sysmonconfig) or EDR (filemod) or similar.

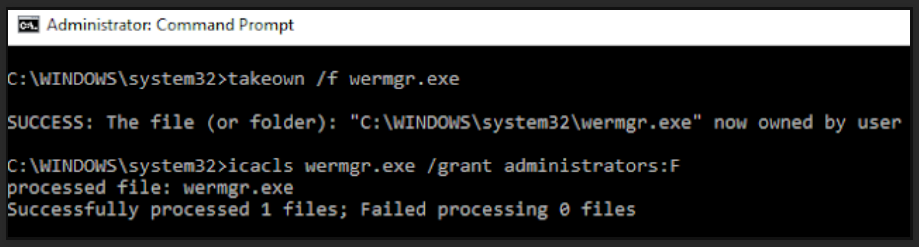




Example of CarbonBlack Query:

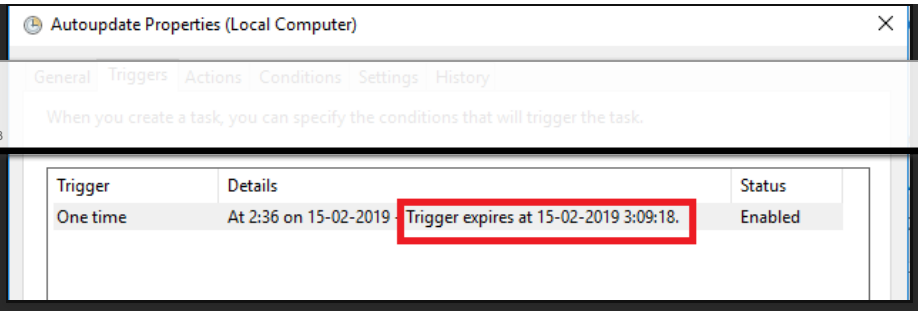
filemod:aitagent.exe or filemod:DiagTrackRunner.exe or filemod:CompatTelRunner.exe or filemod:acproxy.dll or filemod:wsqmcons.exe or filemod:lpremove.exe or filemod:srrstr.dll or filemod:wermgr.exe or filemod:sdclt.exe or filemod:appidcertstorecheck.exe or filemod:AppHostRegistrationVerifier.exe or filemod:MicTray64.exe or filemod:usoclient.exe or filemod:dsregcmd.exe or filemod:sihclient.exe or filemod:dimsjob.dll or filemodLracengn.dll or filemod:HotstartUserAgent.dll or filemod:MsCtfMonitor.dll or filemod:PlaySndSrv.dll or filemod:AdobeARM.exe or filemod:GoogleUpdate.exe

N.B. changing files in protected system directories will require from the attacker to change file owner and then grant himself or a group Full access rights, windows builtin utilities to do that are takeown.exe and icacls.exe (include them in your watchlist, may come renamed, use IMPHASH in your sysmon configuration or File description or Hashes).

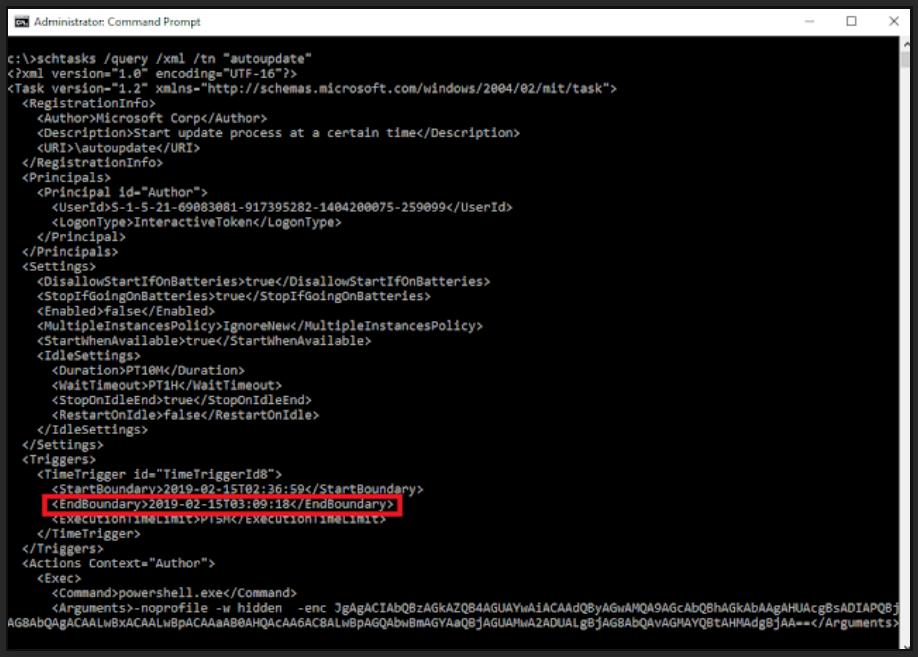


G) Scheduled Task set to run only once (weird):

Example of only once scheduled tasks can be seen below:



XML config of the same:



Detection Logic:

if event.id=4698 and event.payload regxp-matches "(?i)(.*TimeTrigger.+EndBoundary.*)" -> Alert ("One Time Exec Scheduled Task Detected")

Posted by MENASEC at 14:07

Labels: persistence, remote execution, task scheduler

No comments:

Post a Comment

To leave a comment, click the button below to sign in with Google.

SIGN IN WITH GOOGLE

Newer Post Home Older Post

Subscribe to: Post Comments (Atom)