**CERT-EU Security Advisory 2018-024**

# Windows Task Scheduler – Privileges Escalation Vulnerability

***August 29, 2018 — v1.0***

*History:*

- *29/08/2018 — v1.0: Initial publication*

## Summary

On August 27th, a tweet from a researcher with a nick *SandboxEscaper* announced an unpatched local privileges escalation vulnerability in Windows. This flaw is affecting the way Task Scheduler uses Advanced Local Procedure Call (ALPC) to read and set permissions. This allows a user with read access to an object to change his rights on it. Eventually, this vulnerability allows a user to run code with SYSTEM privileges. It is important to notice that a POC has been already published on Internet and there is no available patch yet [1, 2].

## Technical Details

The POC has been distributed with a text document giving some technical details about the exploit. This advisory is based on that [2].

The Windows Task Scheduler uses an ALPC method - `SchRpcSetSecurity` - to deal with the rights of the created tasks. As side effect, this function checks if a `.job` file exists under `c:\windows\tasks` and tries to set the permissions there too. Since all users have write permission there, an attacker can create a hard link in this folder in order to rewrite the privileges he has over other objects of the system. After that he can modify the object ending up with his code running as SYSTEM. This is the path followed by the POC with a DLL hijacking as outcome. It is important to note that this technique for rights rewriting could be used in other unforeseen ways.

## Products Affected

So far we have checked the POC in a Windows 10 64-bits machine. The POC might need some patching to work in 32 bits, because it has some hard-coded paths. It is supposed also to affect Windows Server 2016.

## Recommendations

At the date of issuing this advisory there is no patch available for this vulnerability from Microsoft.

### Workarounds

Since there is no patch available, CERT-EU has been searching for detection methods that could be deployed through different infrastructures.

It is important to note that to get benefits from this method, the proper policies - such as enabling audit on file change [5] - have to be set in the final systems.

As before mentioned, in order to use this exploit, a `.job` file in the folder `c:\windows\tasks` needs to be created. This event cannot be easily monitored with event log [6]. Anyway, there are other methods that can be used such as Sysmon event 11, a PowerShell script, or specific programs [7, 8].

Subsequently, we have observed that a security event 4664 was recorded in the Windows event log when the hard link is created to link the `.job` file with the targeted object. As can be seen:

```
4664

An attempt was made to create a hard link.

Subject:
   Account Name:  DESKTOP\username
   Account Name:  username
   Account Domain:  DESKTOP
   Logon ID:  0x2190D

Link Information:
   File Name: C:\Windows\System32\DriverStore\FileRepository\prnms003.inf_amd64_be4393c143e46548\
     Amd64\PrintConfig.dll
   Link Name: C:\Windows\Tasks\UpdateTask.job
   Transaction ID: {00000000-0000-0000-0000-000000000000}
```

Finally, we have observed that a security event 4670 was created when rights are changed on the `.job`, being changed also in the same way via the hard link in the targeted object. As can be seen:

```
4670

Permissions on an object were changed.

Subject:
    Security ID:  SYSTEM
    Account Name:  DESKTOP$
    Account Domain:  WORKGROUP
    Logon ID:  0x3E7

Object:
    Object Server: Security
    Object Type: File
    Object Name: C:\Windows\Tasks\UpdateTask.job
    Handle ID: 0x198c
```

```
Process:
    Process ID: 0x3f4
    Process Name: C:\Windows\System32\svchost.exe

Permissions Change:
    Original Security Descriptor: D:PAI(A;;FA;;;S-1-5-80-956008885-3418522649-1831038044-185329263
      1-2271478464)(A;;0x1200a9;;;BA)(A;;FA;;;SY)(A;;0x1200a9;;;BU)(A;;0x1200a9;;;AC)
    New Security Descriptor: D:ARAI(A;;FA;;;BA)(A;;FA;;;SY)(A;;0x1301bf;;;AU)(A;;0x1200a9;;;BU)(A;
      ID;FA;;;BA)(A;ID;FA;;;SY)(A;ID;FA;;;S-1-5-80-956008885-3418522649-1831038044-1853292631-2271
      478464)
```

Although, some of these events might be noisy, targeted searches through the events collected in a SIEM taking into account the path for for the `.job` files and the time window for the events can lead in a straightforward detection process. CERT-EU is implementing these searches across the logs available.

**Note:** This is a temporary workaround. The final solution of course will be patching properly the system once the patch is available from Microsoft.

## Exploits

As already mentioned there is an PoC exploit available in the wild [2].

## References


[1] https://twitter.com/SandboxEscaper/status/1034125195148255235?ref_src=twsrc%5Etfw

[2] https://github.com/SandboxEscaper/randomrepo/blob/master/PoC-LPE.rar

[3] https://cwiki.apache.org/confluence/display/WW/S2-057

[4] https://doublepulsar.com/task-scheduler-alpc-exploit-high-level-analysis-ff08cda6ad4f

[5] https://www.ultimatewindowssecurity.com/blog/default.aspx?p=64d64d65-de9d-47d1-ac83-c7fb9fa0ebb2

[6] https://blog.varonis.com/windows-file-activity-monitoring-event-log/

[7] https://superuser.com/questions/226828/how-to-monitor-a-folder-and-trigger-a-command-line-action-when-a-file-is-created

[8] https://www.nirsoft.net/utils/folder_changes_view.html