Short Description

When processing temp files, Windows Error Reporting Service

wer.dll’s Temp directory check can be bypassed due to multiple race condition bugs, leading to arbitrary file deletion.

Vulnerability type

Elevation of privilege

Platform

Tested on Windows 10, Version 1909.18363.476, will also work on Windows Insider.

Summary

When a process crashes, Windows Error Reporting service will try to collect process information, a series of temp files will be created under C:\ProgramData\Microsoft\Windows\WER\Temp, which is a normal user writable directory.

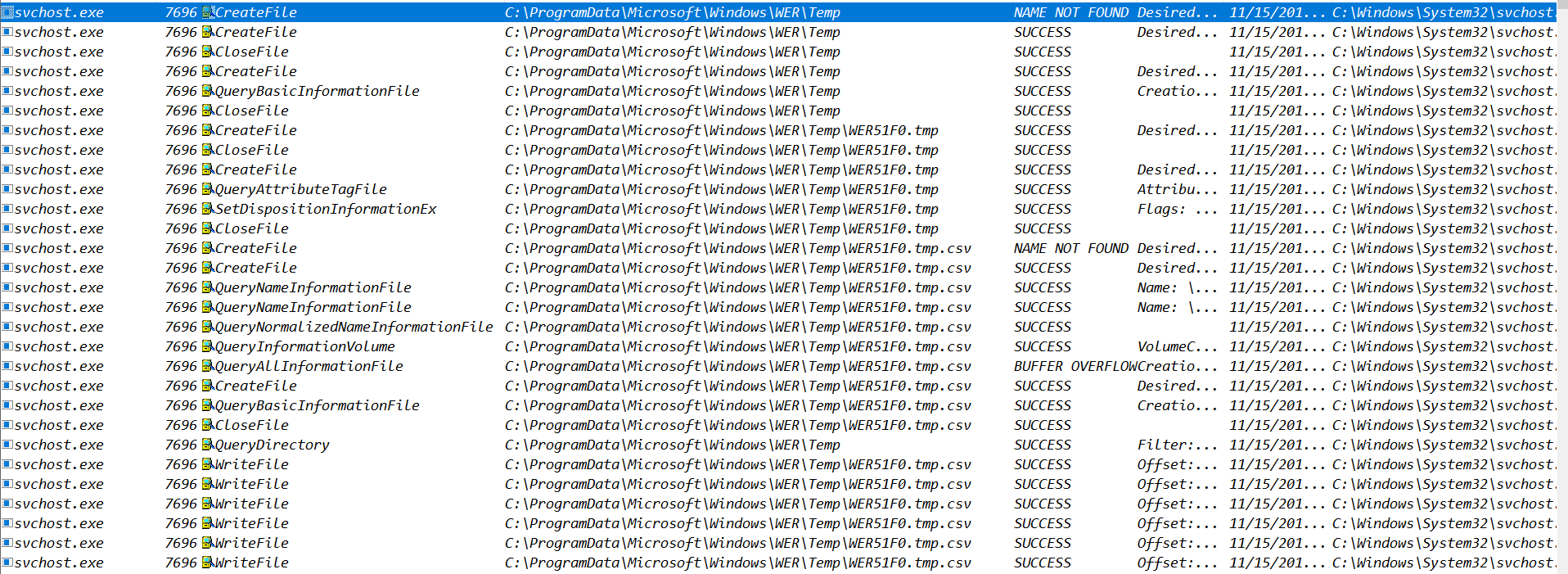
During the November 2019 Security Updates, the service is enhanced by a forehead checking before writing into the Temp directory, it mainly checks if the Temp Directory is part of a Symbolic link. If so, further ops will be prohibited.

But the check has serveral race condition bugs which gives a standard user the chance to turn the Temp Directory into a symbolic link after the check, which bypasses the check and this will lead to arbitrary file deletion.

Description

Race1.

If a standard user attempts to delete Temp directory, WerSvc will try to recreate the Temp Directory when any process crashes, because the directory is created there will be no extra check, here’s a screenshot of Process Monitor of this condition:



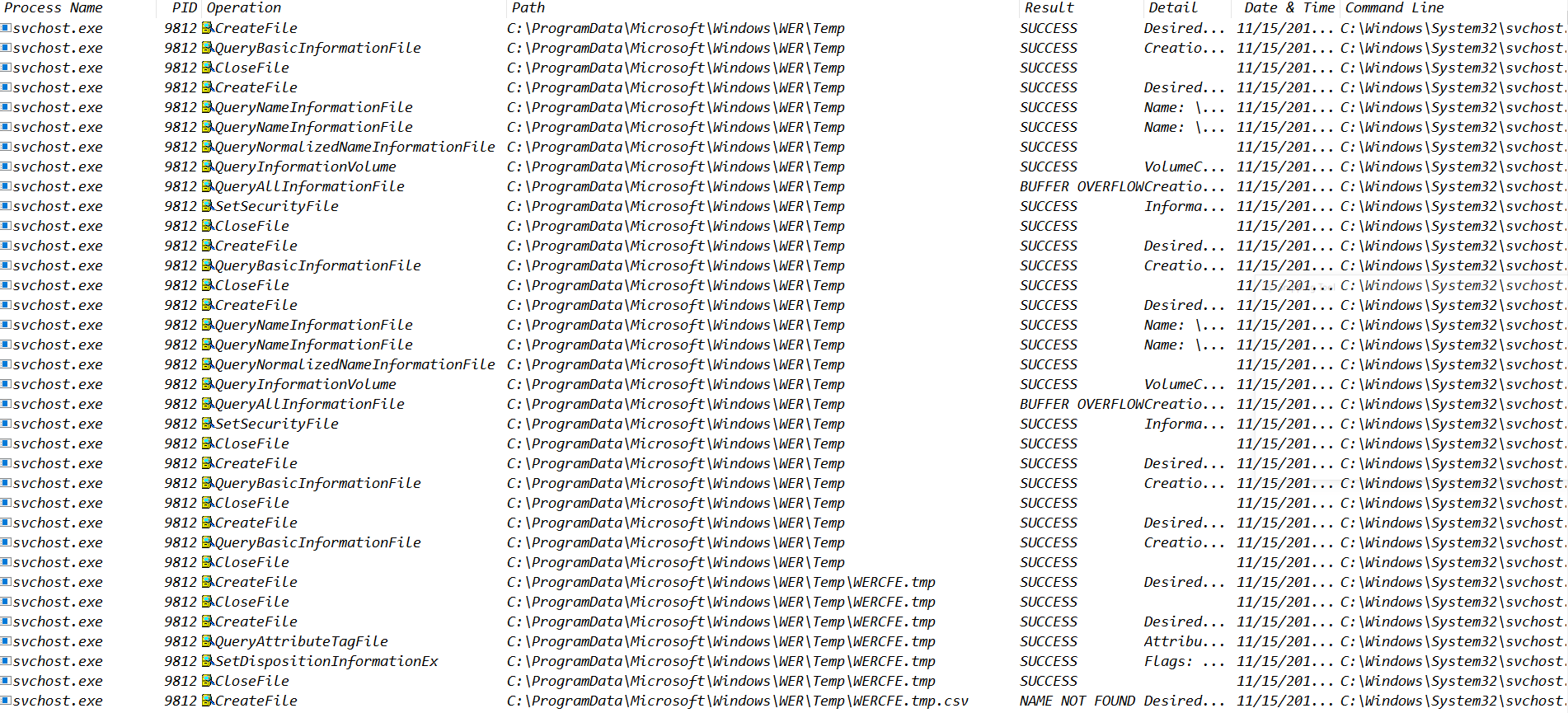
Pic 1

As Pic 1 states, WerSvc detects Temp Directory’s missing, it creates the directory and moves to GetTempFileName call without any checking. The race chance is good in this situation, a standard user can try setting an oplock on Temp Directory repeatedly, since no extra handle is opened after creation and before the GetTempFileName call, once oplock is set successfully, the user can turn Temp into a symbolic link which will lead to the same bug stated in case CVE-2019-1374, which is actually leading to arbitrary file deletion.

Race2.

The second race condition bug is after check happens and before the GetTempFileName call.

When a standard user triggers a crash, WerSvc processes Temp Directory as follow Pic 2:



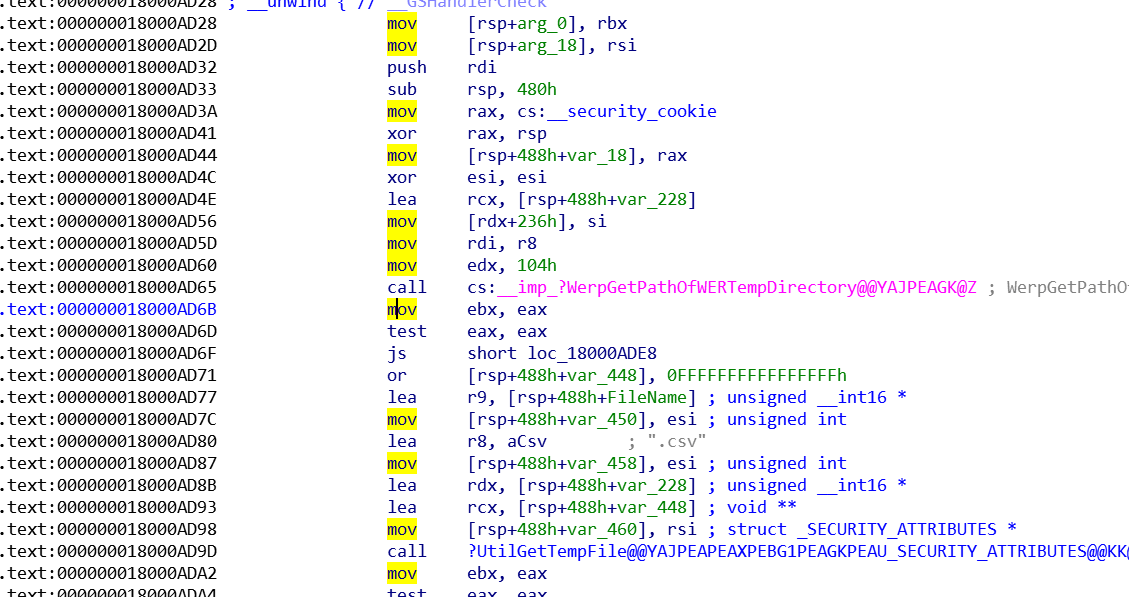
Pic 2

WerSvc do check if Temp Directory is a legal directory but after the check and before the GetTempFileName call, no extra handles is opened. Thus a standard user can race to get a handle before GetTempFileName, and use it to alter Temp directory to a symbolic link, under this situation, there’re no more checks. This leads to the same bug stated in case CVE-2019-1374.

Reproduce

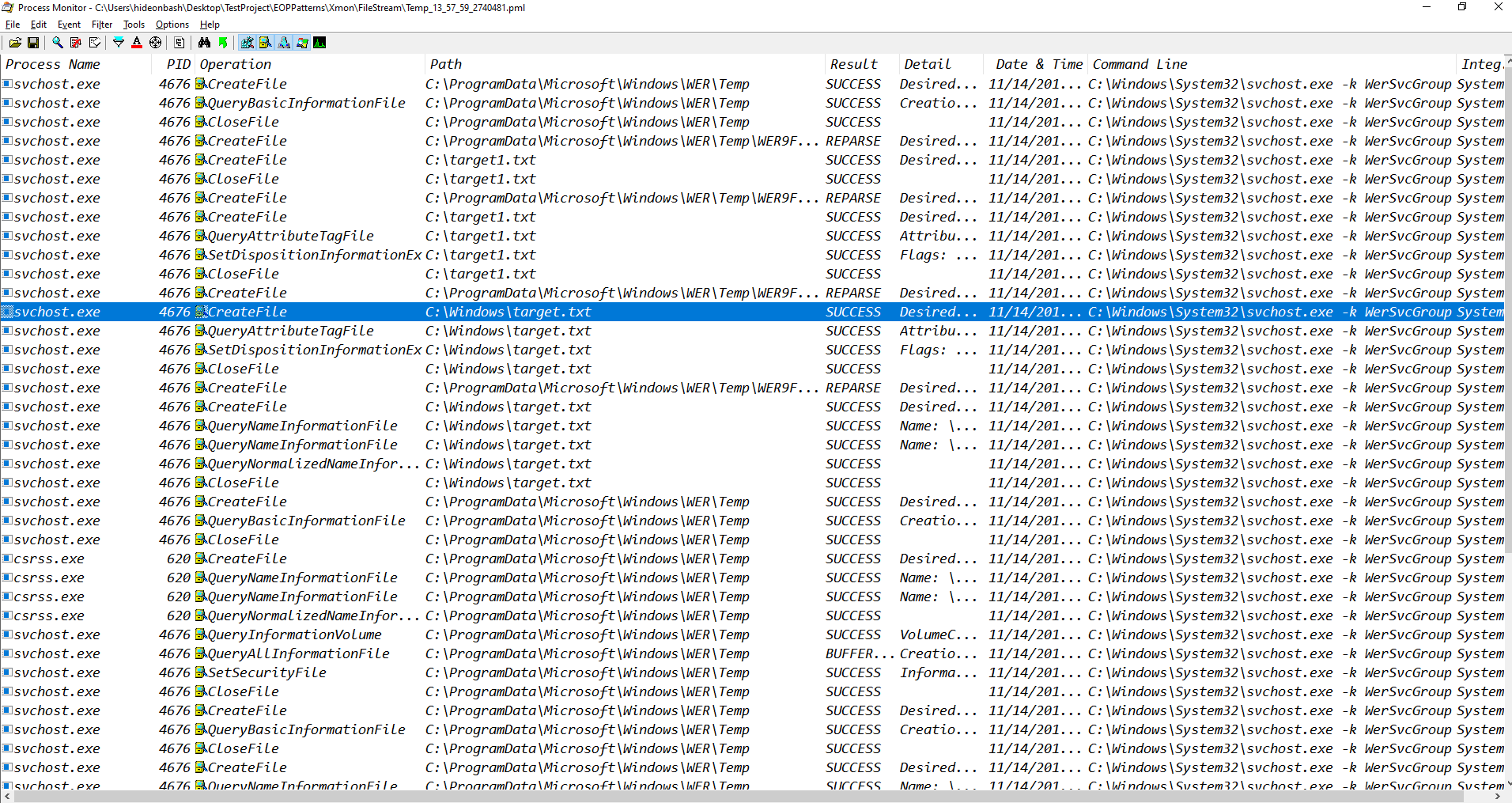
In order to reproduce this 2 issues, it’s applicable to write a race process with oplock to test, while it’s also a convenient way to use windbg.

By setting a break at wersvc.dll + 0xAD6B, as Pic 3 states:



Pic 3

Then triggers a crash, WerSvc will stop right after the check and before GetTempFileName, turn Temp directory to a symbolic link and target file will be modified. This manner should reproduce the above two race bugs like in Pic 4 below:



Pic 4 Race Condition to delete C:\Windows\target.txt

Annotation

This issue is one of the two bugs we found in this service, the other one is in the other report and they’re different bugs.

Acknowledgement

Zhiniang Peng (@edwardzpeng) of Qihoo 360 Core security and Fangming Gu (@afang5472)