Short Description

Race condition in Windows Error Reporting Service’s wersvc.dll: UtilGetTempFile 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

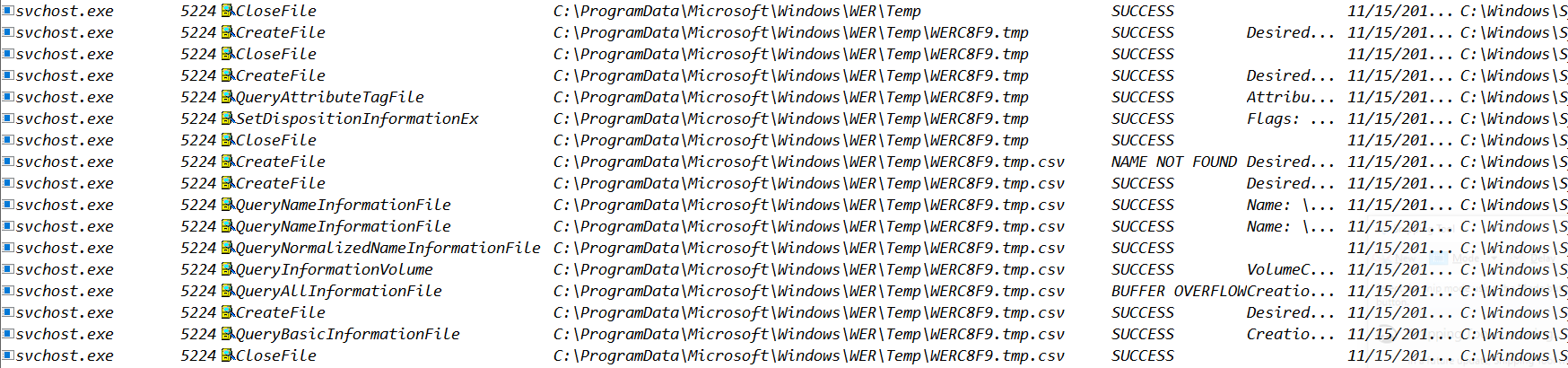
When a process crashes, Windows Error Reporting service will try to collect process information, a series of temp files will be created under C:\ProgramData\Microsoft\Windows\WER\Temp, which is a standard user writable directory.

WerSvc uses GetTempFileNameW to generate unique temp file name during process,

the procedure of getting a temp file is encapsulated in wersvc.dll: UtilGetTempFile, which has a race condition bug, a standard user may win the race and turn the generated temp file into a symbolic link, which leads to cause arbitrary file deletion.

Description

When processing temp files, WerSvc uses GetTempFileNameW to generate a temp file name, then delete it, and write extra information using the file name as a prefix, the following pic shows the operation:



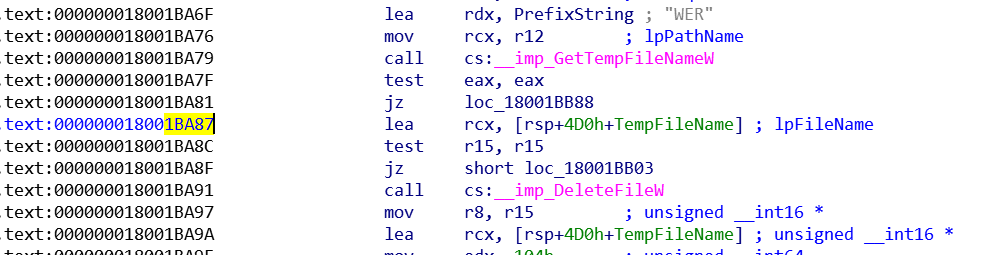
Pic 1

As Pic 1 states, WerSvc uses UtilGetTempFile:: GetTempFileNameW to get a temp file name, then close and delete it. But during the process of GetTempFileNameW and DeleteFileW, there’s a race window which gives a standard user the chance to alter the only generated temp file into a symbolic link, this will lead to arbitrary file deletion on the following DeleteFileW op. Although the race window is not a big one, but on a modern multi-core processor, it’s applicable to win the race with a chance.

Reproduce

In order to reproduce this issue, 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 + 0x1BA87, as Pic 2 states:



Pic 2

Then triggers a crash, WerSvc will stop right after the GetTempFileName call and before the delete op, turn Temp file into a symbolic link and target file will be deleted.

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.

Additional information on case 54956:

Inorder to achieve arbitrary file deletion, we need to let GetTempFileNameW's generated file pointing at fixed

position. This can be achieved by placing 65535 DosDeviceSymlink in advance, all targeting the same file we want to

remove. Thus the race can be exploited as long as Temp directory is a MountPointLink when UtilGetTempFile::GetTempFileNameW

is invoked.

Acknowledgement

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