Dubious security vulnerability: Copying a program and running the copy

devblogs.microsoft.com/oldnewthing/20150311-00

March 11, 2015

Raymond Chen

This wasn't an actual security vulnerability report, but it was inspired by one. "If you take the program `XYZ.EXE` and you rename it or copy it to a new name that contains the letters `XYX`, then you can trigger a buffer overflow in the renamed/copied version of `XYZ.EXE` due to a bug in the way it parses its own file name in order to generate the names of its auxiliary files."

While that's a bug, and thanks for pointing it out, it is not a security issue because there is no elevation of privilege. Sure, you could rename or copy the program and run it, but if you have permission to do that, you may as well do it the easy way: Instead of copying `XYZ.EXE` and running it, just copy `pwnz0rd.exe` and run it! Either way, it's just a case of you attacking yourself. You did not gain any privileges.

Renaming or copying a file requires `FILE_ADD_FILE` permission in the destination directory, and if you have permission to add files to a directory, why stop at just adding files that are copies of existing files? You can add entirely new files!

In other words, instead of `copy XYZ.EXE XYX.EXE`, just do `copy pwnz0rd.exe XYX.EXE`.

This is a variation of the dubious vulnerability known as *Code execution results in code execution*.

Now, this would be an actual vulnerability if you could somehow redirect attempts by other people to run `XYZ.EXE` from the original to your alternate `XYX.EXE` instead. But that would be attacking the redirection code, not attacking `XYZ.EXE` itself. Because if you can fool somebody into running `XYX.EXE` instead of `XYZ.EXE`, then you may as well fool them into running `pwnz0rd.exe`. It's not like the `CreateProcess` function performs a hard drive scan looking for a program whose name is similar to the one you requested and running that other program instead.

Raymond Chen

Follow