Why didn't Windows XP auto-elevate programs beyond those named setup.exe?

devblogs.microsoft.com/oldnewthing/20100726-00

July 26, 2010

Raymond Chen

Commenter J-F has a friend who wonders why Windows XP didn't auto-elevate all installers but rather only the ones named setup.exe. (Perhaps that friend's name is Josh, who repeated the question twelve days later.) Remember what the starting point was. In Windows 2000, *nothing* was auto-elevated. Before adding a feature, you have to know what problem the feature is trying to solve. The problem is improving the experience for non-administrators who want to install software. When they try to install a program and forget to use the *Run as* feature, then instead of proceeding halfway through the installer and then getting an *Access denied* error, do the *Run as* for them automatically. Knowing whether the user is running an installer that requires elevation requires a degree of semantic analysis beyond what you want to add to the `CreateProcess` code path. Hey, here's a program called `PRONT4.EXE`. Is it an installer? Turns out that it is. And then there are the programs that might be installers, depending on what other command line switches you provide. Given that you're reduced to a heuristic, you have to decide what the acceptable rates of false positives and false negatives will be. If you guess wrong and think a program requires administrator privileges when it doesn't, then you've screwed over all the non-administrators who want to use the program. "I used to be able to run this program, but now when I try, I'm asked for the administrator password, which I do not know. Windows broke my program." The effect of a false positive is *My program stops working*. On the other hand, if you fail to detect a program that requires being run with administrator privileges, the behavior is the same as before: The user gets an *Access denied* error. The effect of a false negative is *No change*. Given that the cost of a false positive is huge and the cost of a false negative is zero, you can see that the math says to use a conservative heuristic. The heuristic is that a program named `setup.exe` will be treated as an installation program, and nothing else. Windows was under no obligation to auto-detect installation programs. Indeed, according to the strict interpretation of operating system design, it *shouldn't* do this. If the user says to run this program at the current privilege level, then you darned well better run the program with the current privilege level. The treatment of programs named `setup.exe` is really just a compatibility hack, a courtesy to make your life a little bit easier. It's a case of giving somebody five dollars and being asked why you didn't give them ten.

Starting in Windows Vista, applications can specify via a manifest whether they want to run at the privilege level the user requested (`requestedExecutionLevel level="asInvoker"`) or always to elevate to administrator (requestedExecutionLevel level=”require-Administrator”). Hopefully, all new applications will specify their elevation requirements explicitly, and the heuristic will be necessary only for old programs. e>

Raymond Chen

Follow