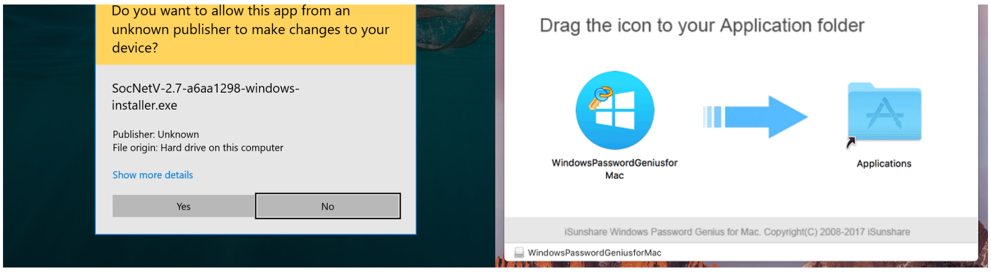
The PC-Mobile Security Divide

Captain, I'm receiving a transmission from an unknown alien source.
Spock, can you decipher the message?
Yes captain, it's an executable file named 'word.exe'. I tried running it but application is not starting.
Err, let me try running it as an administrator.
(The Enterprise's computer systems blue screen and power off due to a poor sense of operations security)

In the realm of personal computing, there exists a strange divide. On one hand, you have the mobile side, where permissions are explicitly granted to an application. *Maybe* that hilarious iBeer app, which turns your phone into a simulated pint glass, doesn't need access to your contacts... And, if any other app prompts you to access inappropriate permissions, you can simply deny it access. On the other hand, PC apps kind of just do whatever they feel like doing. Everything from your office programs to a funny purple monkey that dances around your desktop has essentially unfettered access to your user space.

Current Solutions

Applications that Windows and macOS users interact with nowadays are signed with a developer's private key when they are packaged, giving them the stamp of approval from Microsoft or Apple. And Microsoft and Apple are trustworthy, right? So... what's the issue?



Do you want to allow this app from an unknown publisher to make changes to your device?

SocNetV-2.7-a6aa1298-windows-installer.exe

Publisher: Unknown
File origin: Hard drive on this computer

Drag the icon to your Application folder

iSunshare Windows Password Genius for Mac. Copyright(C) 2008-2017 iSunshare

Well, as it turns out these two companies don't really review the contents of the executables you decide to run. And for good reason- with the insane amount of applications being created, it's impossible to review each one or probably even a majority of them. With that said, there are places like the Microsoft Store and the App Store where applications are reviewed before being published. However, even this is not foolproof plus you are always going to have some niche application that requires you to manually run or install an application outside of your curated app store.

But what about my anti-virus?

Anti-viruses are basically big tables of program signatures that try to identify malicious files. However, it isn't hard to make malicious applications that get missed by an antivirus, especially with the use of polymorphic code which allows an executable to change its appearance to anti-viruses.

So... what's the solution?

Well, to be clear the average user is probably smart enough to install applications from trustworthy sources. Or, maybe it is more accurate to say search engines have become more cognizant as to verifying the links to websites to download software. Maybe anti-viruses are actually getting better at detecting malware. However, it still stands that sticking to this paradigm of installing software is going to always be a cat and mouse game of the bad guys creating novel, malicious software and the good guys detecting it.

On the contrary,allowing users to clearly grant permissions to applications has proven itself to be a good solution in the mobile operating system space. This is not without its issues though; for example, if permissions aren't granular enough, a developer might be forced to add an overly broad permission. It also requires the OS to have a secure enough sandbox, lest the applications would grant themselves permissions. However, with enough improvements such an access control system can, and has proved in mobile operating systems, to be a great security layer.

As an aside, it would also be nice if the applications people commonly use were all open source and auditable... including the operating system... but maybe I'm a little off my gourd.

What is *your* solution?

I'm a Linux user, and for the uninitiated Linux, or more accurately GNU/Linux, is essentially a very customizable operating system due to its highly documented and open source nature. An application for Linux called Flatpak allows for the sand-boxing of user applications, while utilizing permission-based access similar to mobile applications. I have found it to be a wonderful way to install and use applications in a secure manner.

I'll explain in detail how I use flatpaks in my system in an upcoming blog post. For now, take a look at my dotfiles.

Happy computing!

Revision #6
Created 20 November 2024 15:47:07 by GT
Updated 2 January 2025 21:48:45 by GT