Two-Factor Authentication

The last lesson talked about passwords and why they are so important. But, something that is just as important is *two-factor authentication or 2FA*, which...as it sounds, is a second form of authentication required to access an account. *This could be verifying a code sent to your phone number after you've put in your password, it can be a second password requirement, a code from an app like Authy or Google Authenticator*, a USB key, or security questions can work as well.

We'll cover more after this short message...

Promotional Spot

The idea with 2FA, is even if your password fails you, you are still protected. Maybe a service you use gets hacked and someone gets your password. Maybe you're tricked into handing your password over through social engineering, a scammer, a phishing site, or maybe your password is brute-forced, this stuff can happen to any of us. But, even though someone has your password, which assuming you followed your *OPSEC* is only being used on that one service, they still can't gain access to that account because additional authentication is required. *It's similar to movies, where in order to authorize access, you need to have two people turn the key at the same time to unlock the door.*

This is all great in theory, but not every form of 2FA is created equal. The most common form of 2FA is an *SMS text message sent to your phone with a code. The site asks you what the code is, and you simply type it in. This seems extremely secure. Well, not really.* Lots of cell phone providers are known to have weak security when it comes to social engineering, something we'll talk about down the road in the course. Someone can call your cell phone provider pretending to be you, and *forward all the SMS messages sent to your number--to theirs. This actually happened to Linus Tech Tips, leading to the compromise of their Twitter page and their website, as well as many other YouTubers including Boogie2988.*

On top of this, *Kevin Mitnick in his book The Art of Invisibility discusses a pretty easy social engineering technique that exploits SMS two-factor authentication. Here's what he said: "Say I want to take over your email account and don't know your password. I do know your cell phone number because you're easy to find through google. I can go to the reset page for your email service and request a password reset, which, because you enabled 2FA, will result in an SMS code sent to your phone." Mitnick then directly social engineerings the user by texting the person a non-suspicious text "from Google" saying "Google has detected unusual activity on your account. Please respond with the code sent to your mobile device*

to stop unauthorized activity" So he's impersonating Google here to get a user to send over the two-factor authentication code. Now that Mitnick has the code, he has all the information he needs to reset the account password and gain access. On top of all of this, SMS is unencrypted, opening up the potential for SMS sniffing; we're going to talk more about that in lesson 3.15: safe communication.

Because of all of these SMS risks, I highly recommend avoiding SMS 2FA, unless it's the only option available, which unfortunately does happen. If your account only gives you the option to use text verification, I encourage you to speak your voice and push them to be more secure. But, do remember that having text 2FA is better than nothing.

So what is proper two-factor authentication? *Well, one solution is an app which generates keys for you. The reason these are so much more secure is most never touch the internet or any communication protocols; they locally store keys generated on your device, making it extremely difficult for anyone to get these keys outside someone with physical access to your device.*

So what authenticator app should you use? *There are tons of apps that give you this functionality, and to be perfectly honest, this is one of the few areas of this course where I'm going to tell you it doesn't matter too much which one you go with,* it's mostly personal preference. These are simple apps, and all they do is generate new keys for you every 30 seconds, that's it. *The way it works is you scan a QR code for the service you're signing up for, and now it'll generate codes for you to use.* Probably the most popular *FOSS authenticator is Authy,* now don't get me wrong, Authy is great...but it offers cloud syncing and key backups which is something we don't want. We want all of this to be stored locally for better security. You can use Authy, just make sure to avoid the features inside of it. Alternatively, I'd recommend *FreeOTP for iOS, and andOTP for Android.* These are both FOSS, much more limited than Authy, and get the job done just as well.

The last topic I want to cover is physical two-factor authentication, and this comes in many different shapes and sizes, so there are no specific rules or information to follow since everybody's digital life is set up differently. You can require a password in addition to a **** show USB **** USB key to get into your operating systems, so the USB key is functioning as an additional authentication requirement. This can work with full-disk encryption as well. Something like *YubiKey is great for this, or you can turn any flash drive into a device as well.* You can use a *TPM module if your computer supports it, so that your drives can only be booted from a specified computer, and there are hundreds of different products out there that have their own proprietary version of physical 2FA*. Like I said, there's nothing in particular I'm going to recommend, but just be aware that these products exist.

Queue Outro Promos

That's everything I have to say about 2 factor authentication. You can have the world's strongest password, but you're as strong as your weakest link..don't let the lack of 2FA be your weak link. Thank you for watching, and I'll see you in the next *lesson: Search Engines.*