The Zero Day Tipping Point

What I learned in one day for $13 should scare the hell out of you.

I'm a digital marketer by trade. A PR guy doing all things digital. I'm somewhat technical and above all else, white hat.

Well, maybe light gray hat, but that's not really the point.

I work for an International MSP Marketing company. A really good one with a fully distributed workforce, spanning multiple countries.

It's pretty sweet. But that's not my point either. I'm getting there, just need to set it all up properly.

My MSP handles only IT companies, which is awesome, because I dig it and have conversations about cyber security with major IT MSPs every day. For example, we have one client that made the CRN 500 eight years in a row.

So when they speak, I listen.

But I wonder...

Is all the hype on security warranted, or is it self promoting hyperbole? What is the state of this fight? How could I, a digital marketer with various skills, determine these answers for myself?

Hacker boot camp.

Saturday I woke up and decided to see what aspects of hacking could I learn in one day. Initially I thought it would be just a few choice things. But I was so wrong.

So very wrong.

Because I learned what I'd consider to be most everything. I know that's a bold statement, stay with me.

It was largely free and not that hard.

And that's not good for you.

Because it's not about me, I'm not responsible for the liability of a network, any employee activities or a plethora of other dangers like compliance, remote workers, medical records, financials and intellectual property. Typical things on a computer somewhere that's accessible via the Internet.

And if that's the case, then you're in trouble. Because what I learned, is that it isn't even really about technology.

So who am I? I'm a vivid example of being your worst nightmare.

I'm educated, have both time and money, I love computers and I've been writing some code since doing some Basic on the TRS-80 way back when. Former “script-kiddie,” I do know html, css, some JavaScript, Liquid, some PHP and I can edit most any language...

That's really not that impressive but it sets the stage of how to value what I'm going to say.

That what I learned in one day should scare the hell out of you.

It's not because I'm some super genius. It's because all the tools I could ever need were provided for me (for free), with detailed instructions on how to quickly use them.

So yes, I truly believe it's Zero Day. There's no question in my mind.

If I could learn what I did in one day, then anyone can. Anyone that wants it and is willing to try.

I bought an app. I'm not going to say the name of it but I will just say that it was about learning to hack, as a “white-hat, ethical hacker.” Stumbled on it in a major app store.

I was initially pretty skeptical, but I decided to check it out. First chapter was free, what's not to like?

And so it began.

Kali Linux

The start was initially daunting when I realized I needed to be able to operate in Linux, which I had never done. I can finally thank early Windows versions for something, as I was pretty comfortable with the command line, And with that, in 30 minutes had found a list of Linux commands, installed an emulator on my PC and booted up Linux.

Kali Linux to be more specific. You know, the version made by hackers, for hackers? The one that comes loaded with a ton of free, useful hacking programs? For free?

And having the stage set, I began.

Linux Scripting

In under 30 minutes, I had written a bash script which I even pretty much understood. My first script ever. Wrote an IP Sweeper script that I used to ping all the IP addresses in a given range, filter out the ones that responded, and save them in a text file. Had a little trouble at first, but with a little research got it working nicely.

A now I had a list of what I could target. The open ports.

Being fairly impressed with myself at this point, I paid the $13 and went to lesson two.

Changing MAC Addresses

Next I learned how to spoof a MAC Address. Why?

Come to find out, a MAC Address (Media Access Control) is a unique identifier assigned to every internet connected machine, that allows it to be identified when connected to a network. It ensures that the physical address is unique. It is used within a network to identify devices and transfer data between devices. It consists of a source MAC and a destination MAC.

The concept of a unique identifier sounded great, until I learned how to change the factory-assigned MAC address of anything at will. On both Windows and on Linux. Windows is ridiculously easy.

Why is that a big deal?

Because now I am you. The you who has already been authorized into a network, based on your “unique” MAC address.

Or should I say, our MAC address.

This “spoofing” is very similar to taking over someone's identity and performing actions by impersonating them.

As the key to hacking is getting network access, that ain't good. Either a wireless card or wired Ethernet card, I could see what they were, and in a few minutes I changed my MAC address and had full access.

Just like that.

It also made me anonymous and I entered my target network as an authorized user, took over a computer's identity and was fully authorized to do most anything. Oh the things I could have done.

All of this was in an hour and a half. I went from knowing a little, to having hacked a network in an hour and a half.

And then I learned about Steganography. The ability to hide embedded, secret messages in a text, image, audio or video file. That literally took about 5 minutes.

At this point, I was shown how I could begin thinking about hacking. How to structure my preparation and strategy into five stages.

1. Footprinting - Surveillance and Recon
2. Scanning – Information Gathering
3. Getting Access – Putting that information to work
4. Access Maintenance – Ensuring future access
5. Covering my Tracks – Ensuring I don't get caught!

Again I began to move forward, learning how to keep myself safe.

I learned a VPN gives me anonymity and hides my IP address from my ISP and the government. It lets me bypass geo-restrictions, access blocked websites, gives me protection from cyber attacks. Also free, if you know where to look

God bless open source developers.

This is when I began to see the real problem. That it's not the technology.

It's human nature.

The technology and the fight to crack it are the Yen and Yang of the situation. Been like that forever. Each side occasionally makes small strides, the other catches up. So I do believe we're doing everything we can. But every action has an equal and opposite reaction. Up until this point, I was still rooting for the IT company.

But then I realized why the fight will never stop. Because we're not fighting technology. We're fighting ourselves. We're fighting ego and fear. We're fighting vanity, hatred, injustice and every other kind of human weakness.

And that can't be controlled by technology. It has to come from change. And that's asking a lot.

It's Called Social Engineering

Social Engineering is a problem. There's no hardware or software that can help someone not run their mouth. Or throw sensitive information into the trash. Or not to be tricked. Or not to get drunk and misspeak. Or lose a laptop, phone or file folder. Or put up a lot of social media photos in a particular sports jersey. If an avid enough team fan, surely that team has something to do with their password. Right? Doesn't yours? Mine does. Or did anyway.

Not into sports? Fine, how about your kids? Pets? Your address? Family member birthday? Is your ATM PIN is based on something like that? Strange hybrid of these things, possibly with numbers and special characters?

It's human nature.

Phishing

Phishing is a well-known example of Social Engineering. It's using found information about you from public sources like social media, interactions, email, phone calls, trash surfing, whatever it takes. Phishing is one thing, Whaling is another. Whales are CEOs or CFOs, those with access to company bank accounts or intellectual property. Higher level target.

This Phishing/Whaling concept in simple terms is a type of identity theft that uses technical tricks and social engineering to deceive users into revealing sensitive personal information. It's been said that 91% of all attacks begin and end with some form of social engineering.

Sounds high but maybe it is accurate, couldn't really say.

Program that came with my Kali Linux install was Zphisher, which comes with 37 phishing templates that are ready to go for Facebook, Twitter, Paypal... you name it. It also has 4 port forwarding tools and requires no graphic or web design knowledge, programming or anything like that.

Click a few buttons and you're ready to send is a mirror email template of big site emails along with a reason someone should have concern and go access their account. Things like:

“We've noticed some strange activity on your account,
click here to change your information.”

“To continue, please enter your Facebook password.”

“Please verify your bank account details.”

“Please verify you are not a robot.”

You are sending them dangerous links directing them to your mirror web page of a major site. They try to login, send their information to the hacker who shows a login error and redirects them to the real site. They think it was a minor glitch, login for real and never give it another thought.

There are ways to spot this, if you're at all savvy. But, consider how many people are not. Like parents, grandparents, kids... Think of how handy the use of shortened URLs would be here.

Ok, enough. Phishing bad. Like real, real bad.

Keyloggers

Up next I learned how easy it is to make and deploy a keyboard spying program and why it's one of the most used and dangerous attacks a hacker can use. Keylogger. I thought a keylogger was just a software program designed to monitor and record all your keystrokes.

But I was wrong. It can also be a device. And that's scary. Because a device doesn't really comprise any threat to your system and isn't detectable with most anti-virus programs. Once activated, it sends logins, passwords, banking information, PIN codes and everything else you type directly to the hacker.

I clicked one button that downloaded BeeLogger and installed everything I needed (Wine, Python, Pywin32) and a variety of other things that allowed me to quickly make one. Export options were web link, PDF, Powerpoint, Excel, Word Doc, tainted USB stick, you name it.

These things are then sent to you using social engineering techniques, which aren't really that hard. In my opinion. They can come through email, chats, P2P networks, text messages and even social networks.

Personally, if it were me, I'd infect a bunch of really, really nice USB drives and then leave them all over the place. Places frequented by high worth individuals, because human nature would lead some to pocket them. Then use them and be infected without ever knowing it.

That's social engineering. Using human nature against someone.

Wifi Hacking

As if all of this wasn't enough, the next section taught me how to crack various wifi networks, first with WEP, which is incredibly easy and next with WPA and WPA2. As I live in a populated city with lots of neighbor houses, finding networks was not a problem. Cracked WPA in 15 minutes, cracked a WPA wifi in 25 minutes and then eventually a WPA2 secured network after about three hours. Really not that hard.

Network Spying

Up next? What I could do with all my new accesses. Network Spying was first, learned what a Man in the Middle attack was, ARP poisoning and spoofing, spying techniques and how to bypass https which come to find out, isn't really that secure! Unfortunately I wouldn't do any of these things to my neighbors or anybody else, so I had to stop.

Light gray remember?

Database Hacking/SQL Injection

Moving on to database hacking, I learned about SQL injection, which was amazingly effective. Luckily there are some purposed servers with vulnerabilities for the white hats to practice on. I learned how to generate SQL commands and how to inject them into a URL string or data structure on PHP and ASP.NET platforms. And I was able to do some devastating damage like dump the whole database of a system, to modify the content inside and performed different queries that are not allowed.

The trick here is to get the application to not validate the inputs properly before passing them on. This can be done at any point of entry like address bars, search fields and data fields. Even login and

password areas.

Android Hacking

Next I was taught how vulnerable Android systems are and how manufactures don't really do a great job on updates and patches. Which of course, is the kiss of death and exactly what I need to make and deliver you an APK file., which is commonly used for the distribution and installation of mobile apps, games and middleware.

Android has a 70% market share, making it top target for hackers and where most effort is placed. Safest mobile phone? Windows. Not because of any technology, it's because it has only a 10% market share, so why would anybody bother?

And that is how the game is played.

Using various payload techniques, I can easily manipulate a reverse TCP attack and upload you a file. Then use, you guessed it, social engineering techniques to get the running exploit installed. It's not easy because I know anything, it's easy because I clicked a few buttons following instructions, ran a few scripts I was given and could then hack contacts, cameras and more. I used my own phone, so I didn't need to trick myself into anything, which made it fast and easy.

Password Hacking

Hacking passwords is the process of attempting to gain unauthorized access to restricted systems using common passwords or algorithms that guess passwords. These include dictionary attacks, brute force attacks, rainbow tables attacks, just plain guessing and a spidering attack.

Dictionary attacks use lists of words while Brute force exhaustively does the same thing, taking many more variations into play, including special characters and numbers. It will literally try every variation possible. Good news is you don't have to watch it, you can just run it, come back later, see how you did.

Rainbow table approach was interesting. This uses previously computed hashes, typically stored as md5 hashes. Using a different database filled with md5 hashes of commonly used passwords, we compare the two and if we have a match, then we have a password.

Just plain guessing is surprisingly effective, when knowing default install patterns. Admin, qwerty, password and god top the list. If never changed or if the user is careless, they can be easily compromised.

Spidering is interesting because most companies use passwords that contain company information so they make sense and are easy to remember. This information can typically be found on company websites and social media. Spiders gather this information from these sources and then generate tailored word lists, used in both dictionary and Brute Force attacks and can be turned into hashes and compared to rainbow tables.

Why these things exist

I should say the app did make an effort to constantly say white hat, or ethical hacker. And to even offer suggestions as how to thwart some of these things. It also gave examples of reasons these very techniques are used in “good” ways, like the boss of an organization using keylogging on employees.

And they all say the same thing... That the break/fix model is done and that it's about IT MSP stacks, per seat pricing and the parent companies ability to offload the entire thing. To know that everything is covered. Tempting.

Managed **S**ervice **P**rovider. Think of it like an all-inclusive pass to technology. Where all you have to do is clearly state what you need.

And then you get it.