Journal of Physical Security 5(1), 78-82 (2011)

A Comparison of Cyber Attack Methods

Tyler J. Murphy
Lewis University
Romeoville, Illinois

Introduction

Have you ever seen the movie "Swordfish"? Do you remember when Hugh Grant was writing that "super worm" that was going to punch through the banks security systems and steal a whole bunch of money for John Travolta? What sticks out for me is that while he was writing his "super worm", there where graphical cubes floating around his 6 monitors, and every time something went wrong one of the cubes would shoot out of order. When he finished the hack, everything fit together like he was working on a jigsaw puzzle or something. Unfortunately, that is exactly how real world hacking doesn't happen.

There is an almost idealized view today of "hacking". In the media and in popular culture, they usually show a splash screen of some guy with earrings, or a bunch of donuts sitting in front of a computer staring at binary code, babbling about how he is going to bypass the firewall by cracking the encryption. Not very realistic!

While I was doing my undergraduate work at Lewis University, I wanted to do comparison of cyber attacks. In particular, I wanted to compare attack vectors. Which would be the best? Which would grant me access the fastest? So, I chose three attacks from the many different potential attack vectors—3 that typically receive much of the attention. The first is physical access. The second kind of attack is phishing or social engineering. The third attack is the famous attacking of the computer network.

All the attacks I explored and demonstrated were done on my own computer, or else on a computer and network with the full knowledge and consent of the owners (who offered me no assistance in actually executing the attacks).

If You Can Touch the Box, You Own the Box: Physical Attack

I setup a Windows XP system, a Macintosh OSX 10.5 system, and a Windows Vista Laptop for this experiment. The passwords were completely randomized and I had no idea what they were. As is well known, passwords do not represent a daunting hacker challenge. In the case of the Windows XP system, I used my own password cracker, but you can find them all over the Internet. Basically the cracker I used was a bootable ISO image. I simply rebooted the target system and dropped the ISO (in my case it was a UNIX BOOT OS) into the optical drive, and then followed the on-screen instructions. It has you mount the hard disk, select boot partition, etc. The actual cracking does not take place until the software locates the Windows SAM file. This file is where Windows stores a hashed version of your password. The problem is, the file is writeable. Most of these "crack disks" give you an option to change the password to something else. This, however, really isn't a good choice because the hacking victim is going find something is amiss when the password he/she has used for the past 6 months

doesn't work anymore. So the hacker should simply select "blank password" option. This basically deletes everything inside the same file effectively making the password blank.

The Macintosh 10.5 system similarly offered little resistance. In fact, all the hacker needs is a 10.5-operating disk. Macintosh supplies you with a password-reset utility, which basically does the same thing as the Windows "crack" disk. You can also boot the system in single sign-on mode and delete the initial set-up record. You will have to go through the annoying first time screens, but you get to create you very own admin account at the end.

Only slightly more difficult to hack was the BIOS password for the Vista laptop computer. The BIOS passwords (for most systems) are controlled by the C-MOS chip. To reset the password, all the hacker has to do is reset the chip. This can be done by removing the 3V lithium battery on the motherboard. The battery is easy to spot and is used in many other electric devices. The other way to reset it is a button located on the motherboard.

These attacks were based on having physical access to the computer systems. Even though there are many other ways to crack into an un-supervised computer, one of my favorites (which is often quite effective) is to look around the desk area for a sticky note with the password.

In my view, the best way to insure overall security against a physical attack is to deploy 2 pieces of technology that have been around for thousands of years, the door and the lock This is where physical and cyber security converge. In my experience, there has been a very intense focus on traditional cyber security measures, while physical security has been on the back burner since the 1980's. As a cyber security analyst, I have often seen a $50K Cyber Intrusion Detection System, or advanced firewall put into use in a room protected by an easy-top-defeat $2 lock. Time and time again, I have walked by server rooms where the door is wide open and nobody has even bothered to deploy the ineffective lock. This is a dangerous security practice that can lead to serious consequences. Physical Security is an integral part of Cyber Security: you cannot have a secure computer infrastructure without a physically secure facility to house it.

Gone Phishing and I Caught a Big One: Social Engineering Attack

Recently, social networking sites and their spin offs have become all the rage. This craze has lead to a dramatic increase in Social Engineering attacks. Social networking sites like Facebook, MySpace, and Twitter are making these types of attacks very easy to do. I view phishing attacks as a kind of network attack (with the exception of phone and mail scams) because they are operated from a remote location using the network as a conduit. On the other hand, phishing is ultimately about hacking a person.

For my social engineering attack, I decided to attack my brother (with his general knowledge and permission, but without his knowing any details of the attack). I lived in an apartment about 30 miles away from him, and had no internal knowledge of his network. Since many attackers like to target a specific person, I decided to go with a targeted phishing

attack. It's a bit like using a fish finder for real outdoors fishing. In my case, the fish finder was Facebook.

I did not create a fake Facebook account because doing so is a violation of Facebook's user agreement—not that a real hacker would much care about this!) I instead used a legitimate Facebook account of a friend of mine (with her knowledge and consent). It turns out that when a 17-year old boy (my brother, the target) gets an invitation from an attractive college-age blond female (my friend who's Facebook account I borrowed), to be his Facebook "friend", he is eager to accept.

Prior to this experiment, I had very little experience with Facebook, so I was surprised to learn how much information is just shot across the Internet. My brother had a privacy filter on, so only his friends could see details of his account, but out of what seems to be thousands of his Facebook friends, very few blocked anonymous users from viewing *their* content. I found the best bait for my brother was a free online game that he and his friends kept talking about. After figuring out the name of game, I did a quick recon on the games external website and got a feel for what the game was about. I even signed up for the free newsletter so that I could later simulate the look and feel of an email coming from the game site. After only about 20 minutes of detective work, I had a game plan.

I decided to spoof a fake e-mail address pretending to be the support staff of the game site. After that, I would slip him some code and presto I would have access to my brother's computer. Now the old adage is, "it is easier said than done". However, in this case, it was almost as easily done as said.

I thought about using an "smtp" hack in which I would brute force the password on a virtual "smtp" port, then spoof the index knowing he would not check. Alternately, I could use my already existent web-hosting client. I choose the web-hosting client. My fake email was decorated with the site's logo. and had the look and feel of the official newsletter.

The online gaming site was free, so I decided to attack my teenager brother where it hurts the most, his wallet. I wrote up an email claiming that the game was going to start charging for online services, however since he has been a loyal player he was going to be selected to receive the paid version for free. There was a catch, however: the fake email claimed that they wanted to run network tests and graphical tests to assess their users' computers. If he wanted to keep playing for free, he would have to install some software.

Now I could have written malicious code, and crafted it to his computer but I decided to go even simpler than that. I used totally legal and freely available software to take control of his system. I used a Virtual Network Controller (VNC) client network tunneling software and a few handcrafted batch files to shut off the pesky Windows firewall and start a background install. Within about 1 hour of sending off my bait to him, the fish bit down hard. As soon as he ran the batch files, the tunnel opened right up and I could use the VNC client to connect right to his system. Just like that, his system was under my control.

What I like most (as the hacker) and dislike most (as a computer security analyst) about this attack is its ease. It took little advanced knowledge of programming or network infrastructure. The attack was successfully completed with knowledge freely available on the Internet. If my brother had simply looked at the email address, or even the HTML index, he would have seen that it was coming from a bogus source. Unfortunately, he is not alone in this behavior. People and organizations fall victim to social engineering attacks all the time. In April of 2011, for example, The U.S. Department of Energy's Oak Ridge National Laboratory fell victim to the same type of attack as my 17-year-old brother did. (See, for example, Elizabeth Motalbano, "Phishing Attack Hits Oak Ridge National Laboratory", "http://www.informationweek.com/news/government/security/229402048).

Unfortunately for Network Administrators, there is no silver bullet to fix these kinds of phishing attacks. The only thing you can do is train your employees to be aware and vigilant. A good way to reduce the risk of becoming a target is to limit how much exposure you give yourself over social networking sites. As for the mass spam emails, remember these common sense rules: " NO ONE EVER GIVES AWAY ANYTHING FOR FREE" and "IF YOU DON'T REMEMBER SIGNING UP FOR THE SPANISH NATIONAL LOTTERY…. YOU PROBABLY DIDN'T".

Knock, Knock. Who's There?: Denial of Service (DOS) Attack

If I could go back in time to when I preformed this experiment and still retain the knowledge and experience I have now, I would do one thing differently: Only experiment with the first two attack vectors. The law got in my way more than any security feature did. DOS attacks are only effective if the hacker faces either attacking an extremely small target with limited bandwidth, or if he has a massive (illegal) bot net at his disposal to take on a larger network. Because I did not want to break any laws (though a real hacker might not be so constrained), I focused on attacking my parents network (again with their knowledge and consent).

I entered in through their wireless network, thankful as the hacker (but not pleased as the good son) that nobody had told my parents that WEP is not a secure protocol. So after some ARP relays, and a run through with KISMET, I was in. I used the network mapping tool NMAP to discover the location on the network. (Of course, any organization with minimal cyber savvy would black hole the ping sweep in a heart beat.) I found my target, my other brother who uses my parent's computer system.

Instead of using a crude ping bomb, I used an ARP bomb. Basically I simply asked his system about 10,000 times for his ARP tables. His system was so concerned with getting me this information that it locked up all other services. The systems network traffic came to a halt instantly and then he could not use any other network services. If you can imagine a large cluster of systems attacking at the same time, you can see how effective this method can be.

The DOS attack is quite easy to accomplish, but its effectiveness is limited to just being an annoyance. DOS attacks, however, are the majority of network-based attacks because they are

easy to do. Using IDS systems (or just about any other monitoring tool that tracks network traffic), administrators can prevent these attacks relatively easily.

Conclusion

All of these attacks are used every day out in the real world. The movies and media have painted the image of the hacker as someone staring at binary code or a scrolling through text file and spitting out random lines of code. The reality is much less romantic: hackers (or crackers) are going to use the path of least resistance. They are going to use the most effective attack at the lowest cost or level of effort. In this study, the cheapest and most reliable attack was the phishing attack. This is probably not surprising: social engineering is often the best way to compromise security.

Acknowledgement

Roger Johnston helped to edit this paper.