The Art of Building a Remote Attack

Rhett Green

Abstract—The invention of the internet during a 1969 Stanford experiment has changed the world in more ways than anyone could have imagined. Some would call it the next frontier and as with any new frontier it brought new resources anytime there are resources to be had there will be attackers attempting to steal them. There are many different methods used to steal users private information and the purpose of this paper is to focus on the process used to create one of these attacks. Remote attacks are built around the premise of accessing your personal information using any advantage your system or hardware allows that a knowledgeable attacker can manipulate. There are limitless numbers of attacks, but for the purpose of this study we will focus on building an exploit to attack a standard macbook air. The first stage of any attack is reconnaissance. This process involves gathering data about the target machine that can then be utilized to discover its protections, weaknesses, and security vulnerabilities. After you have gathered all the information you can you move onto the next phase scanning, this process involves three main types of port scanning, vulnerability scanning, and network mapping. Now that you know the weaknesses of the system and it is possible vulnerable entry points its finally time to construct your attack and gain access to the system. For the purpose of this paper and the attack performed, we will discuss these stages in depth and the methods and programs used to gain access. It must be noted that due to time constraints and a changing environment the experiment had some unexpected changes but the theory remains the same.

Index Terms—Remote Access, Reconnaissance, Scanning, FatRat

1 Introduction

In the 1950s the US Air Force had a need for a vast network to keep control of its fleet if a nuclear attack ever happened. US engineer Paul Baran proposed a communication network with no central command point, he called it a distributed network.[1] This didn't immediately result in the internet we know and depend on so much today the invention of the internet was and still is today a continous collection of computer scientist's bringing their technology together to create the impressive vast network we know and recognize today. In 1974 two American computer scientist's Bob Kahn and Vint Cerf decided there needed to be an agreed set of rules for handling data and together they created a new method for sending data packets in digital envelopes. The idea was that these datagrams can be read by any type of computer but only the host machine is allowed to open it and read its contents. What Kahn and Cerf had created is know today as transmission-control protocol or TCP and is used everyday in modern data transmissions. Another vital form of identification for information in computers is DNS or Domain Name System. It acts as a phone book for the internet converting confusing IP addresses into simple names. Everyday users connect to public networks without a second thought unaware that these open networks represent a modern battlefield for hackers vying for users personal information. Attackers use a multitude of methods to try and gain access to this information to then be able to exploit loopholes in the computers system to view files they should not have access to. It makes you wonder if when people like Paul Baran were hard at work on the infrastructure of the internet in the fifties if they foresaw the future battlefield they were creating. In fact its common sense to think that the first computer virus must have been created long after the creation of computers but in fact mathematician John von Neumann was the first to consider the idea as early as 1949. He suggested that a self-replicating automatic entity could work within a computer to attain its own means. However it wouldn't be until 1971 that John von Neumann's prediction would come to fruition when DEC PDP-10 computers working on the TENEX operating system started displaying the message, "I'm the creeper, catch me if you can!" and although this message wasn't malicious it was a proof of concept and more importantly foreshadowing for the future. Hacking has become such a problem in modern societies that entire organizations dedicated to the craft have begun popping up, the most famous of which Anonymous first burst on the scene in 2003. They are famous for being the "Robin Hood" of the cybersecurity scene performing cyberattacks against targets that are generally seen as "bad". With no specific leader the group is a collection of unknown hackers who have carried out attacks on major corporations and even the Church of Scientology. Their calling card is the use of the Guy Fawkes mask to hide the identity of there members and let the world know their affiliation with the group. Not all hacks are illegal either, there is a whole new demand for penetration testers. This is a security exercise where a cyber-security expert does there best to find and exploit vulnerabilities in a computer system in the hopes that this simulated attack will identify any weak spots in a system's defenses which attackers could take advantage of. The modern world is filled with possible attackers and its difficult to try and combat them but the best method is to consider there point of attack and try and build defenses. This requires understanding the attack methods, so for this study we will focus on remote attack methods used by attackers to access unsuspecting victims devices.

- *E-mail: green.rhett@ou.edu*
- *University of Oklahoma.*

1.1 Remote Attacks

The premise of a remote attack is to use some method to gain unauthorized access to a victims system with the objective of stealing data or performing some other form of malicious activity. There are two main types of attacks, they can be either passive or active in nature. In a passive attack the attacker can monitor or steal information but without making any change to the data. In an active attack the attacker not only gains unauthorized access but also modifies the data either deleting, encrypting or otherwise harming it. The attackers goal is to penetrate the network perimeter and then the hardware's security measures allowing them to gain access to its internal system, this is usually accomplished through multiple attack methods working in unison. For the purpose of this study we will focus on the process used by hackers and penetration testers to determine which of these methods is right for there attack focusing on the three main stages of hacking which are reconnaissance, scanning, and finally gaining access to the victim computer.

1.2 Reconnaissance

The first step of any attack we will be considering is the reconnaissance phase. Network engineers use tools to view network data to solve issues and track data however hackers have found that its is possible to use these tools to look for people on the network. The process includes gathering data about the target machine that can then be utilized to discover its protection, weaknesses, and security vulnerabilities. This process is not unlike the work a detective does while trying to solve a murder gathering data and information in the hopes to better understand there victim and eventually the attacker. The two different types of reconnaissance are active and passive. In active reconnaissance you aquire your data by connecting directly to the framework you are attempting to access. This can be very efficient at providing in depth information of your target host but its very likely that your IP address will be know by the target which can lead them back to you. The second type passive reconnaissance involves gathering data without interfacing with the victims framework. This can be accomplished by looking at web indexes or any other freely available reports and while you may have to work harder to gather information it is highly unlikely that the victim will know your IP address or that they are being targeted. There are many tools used for reconnaissance and below are some of the most well known.[4]

1.2.1 Nmap Nmap one of the most well known tools for network reconnaissance, it is a network scanner designed to find details about a system and the programs running on it. It accomplishes this by using raw IP packets to determine what hosts are available on a network, what services they are running, what OS system they are running, and what type of packet filters/firewall and other characteristics. An attacker can launch scans against a system or a range of IP addresses under a target's control and can learn a significant amount of information about the target network. [2]

1.2.2 Scanrand Another form of reconnaissance tool is Scanrand. There are many scanning tools but not many as quick or effective as Scanrand. It does this in two ways, first it contains a process that sends multiple queries at once and another process that receives the response and integrates them. The two processes are separated from each other and they then use a hash based method to show you the valid responses it receives from scanning.[5]

1.2.3 Paratrace A simple command that many developers have to use is Traceroute it allows the user to see where a packet is going and how long it takes to get there. However effective firewalls block these packets for security reasons because an attacker can use this information to create a map of your network layout, and try to identify where your firewalls and routers are situated. Well Paratrace is essentially the same method however it is able to circumvent the firewalls by listening for outbound connections leaving the network and quickly inserting a few TCP segments with an incrementing TTL value starting at 1, this causes the routers to legally respond back along the path with ICMP TTL Exceeded.[6]

1.3 Scanning

Now that the reconnaissance stage has been completed and you've identified a potential victim its time for the next phase in the attack, scanning. Computers have physical docking points built into their system that are used by protocols for operation of network applications. Ports in the range of 0 to 1023 are considered ″well-known″ or ″system ports″ and use by system process for widely use network services and are vital for most network attacks. The range 49152-65536 contains dynamic ports also known as private ports which are used for a large range of private or customized services and are often used temporarily. These can be used for generating all types of tools and come into play in our attack. The purpose of scanning is to probe systems for open ports its like walking down a hallway at a hotel looking for open doors the doors being the computers ports. When looking at each door there are five possible states the door can be in open, closed, filtered, unfiltered. If a port is open that means an application is currently accepting TCP connections, UDP datagrams or SCTP associations on this port. This is what the attacker is looking for in our hallway analogy this would be the equivalent of a wide open door through which if the conditions are right the attacker can enter. A closed port however means the port receives and responds to the scanners probe packets but there isn't an application running on it so it isn't what were looking for. The fact that it is responding though means its accessible and it should continue to be scanned in case it opens later in the future. A filtered packet is exactly what it sounds like the scanner can't determine if the port is open or closed because the port hows some sort of packet filtering which prevents the probes from reaching the port. This can be caused by anything from firewalls to router rules. These filters delay the attacker because of the lack of information returned and the scanner usually has to try several times in case the probe was dropped due to network congestion which slows down the speed of the scan. So now that you understand the states of ports it's time to start scanning to see if there are any vulnerable ports, below are some of the different types of scanners. Knowing which ports are open is vital to creating a line of communication with the victim. [8]

1.3.1 Ping Scan

One of the most basic but vital scans is the ping scan. Ping scans can be used to sweep either an entire network or a single target to check if it is alive. It does this by sending an ICMP echo request to the target if the target responds back with an ICMP reply of its own its alive or active. A simple way to test this is to ping a known website like google it should ping back the relative ICMP reply packets. Unfortunately it is becoming more and more commonplace for firewalls and routers to block pings so they wont always work.[8]

1.3.2 TCP Half-Open

The process of creating a TCP connection involves first sending a packet with the SYN, synchronize flag set to the destination. The destination acknowledges the SYN with a SYN-ACK (synchronize-acknowledge) flag set. The final step is the sender acknowledges the arrival of the SYN-ACK response by sending the destination a packet with the ACK flag set establishing a connection. For TCH Half-Open scans the final ACK packet isn't sent so a connection is not established however you still know if the port is available and listening.[8]

1.3.3 TCP Connect

As the name suggests TCP Connect is a similar process to TCP Half-Open. They are in fact the same exact process except the handshake process is completed because the final ACK packet is sent. This enable good data retrieval but at the cost of speed because it takes more packets to complete the process.[8]

1.3.4 UDP

UDP scans are used mostly to detect DNS, SNMP, and DHCP services by sending a packet to each port. It then uses the response packets to determine the status of the port. If the target responds with an ICMP unreachable error (type3) packet the port is closed, if it sends back a ICMP unreachable error packet with other codes it is considered filtered. If no response is received the port is either open or filtered. UDP is an unreliable scanner however since it doesn't establish a connection or synchronize packets like TCP scanners do so the scanner can be slow because it might wait for a packet that's never coming.[8]

1.3.5 Stealth Scanning

Stealth scans create packet flags that try and create a handshake without actually establishing a connection. These scans are most useful because they are good at eliciting response packets without detection. They're very useful but not without faults as they do not work on Microsoft systems which can help you narrow down the victims operating system. If you use a stealth scan and receive a response of an open port you know the victim cannot be running a Microsoft based operating system.[8]

1.4 Gaining Access

Now that reconnaissance and port scanning have been performed it is time to implement an exploit of the system built off of the data collected. For this paper we will be discussing an attack I built to attack my own 2017 Macbook Pro with a 2.8 GHz Intel Core i7 processor running macOS Mojave version 10.14.6. Lets start with the first phase the reconnaissance phase.

1.4.1 Experiment Reconnaissance

For this experiment I utilized NMAP. More specifically I used a GUI Zenmap that runs NMAP commands just like you would over terminal but is better at sorting and displaying the data. I had to first find the IP address of the victim to run in Zenmap. I accomplished this by creating an IP logger link using iplogger.org. This website takes in a file of your choosing and generates two links one for collecting statistics and another for viewing them after the link was created. I used an image that I got the link for and input into iplogger.org resulting in my two links. The link was then sent to the victim which in this case was me after clicking on the image it pulled up the image normally so the victim would be unaware there information is being retrieved. As soon as the link was clicked the IP address, location, operating system, and even browser being used were returned to the second link where I could view and record them. I quickly double checked that the IP address was active with a simple ping resulting in a stream of return packets alerting me that the computer was active. With this information I was ready to begin my reconnaissance by inputting the IP address into Zenmap the only requirement for this is both the host and victim device must share the same network. I ran an intense scan with the command nmap -T4 -A -v *IP address* resulting in a succesful scan that gave me lots of information about the victim system. It confirmed the operating system with an estimated accuracy it scanned 1000 ports 996 of which were closed with 1 filtered port and 3 opened ports. Another feature that is really useful on Zenmap is the Topology feature which provides a GUI representation of the network scanned. It shows the active devices on the network there connection to each other and any firewalls it detects. In this case it detected two computers connected to the local host both using active firewalls the other system being my roommates computer which I didn't even have to scan it was found because it was on the same network. This kind of scan would work best on a large network with many devices that I could then view using topology telling me which ones are protected using a firewall providing a list of possible victims but for the purpose of this study I wanted to attack a specific device. The results of the scan informed that I had an open port at 2000, 8291, 8292, and a filtered port at 8383. It also disclosed the service being ran on that port, unfortunately these ports are not ones I can use to access the system because they run programs that don't allow enough data for me to formulate an attack. The type of ports I was hoping for are file transfer ports like port 22 or 23. These are the ssh and telnet ports and would have provided an ideal entry point for an exploit for a real world exploit using the ports it would be more realistic to create a bot scanner that scans networks for ports like these that are open. This sort of scanner could then be used to scan vast networks and find suitable victims. Due to the secure nature of Mac OS Mojave, there were no vulnerable ports available for me to manipulate so I had to look for alternative entry methods.

1.4.2 FatRat

Since I couldn't access the system using its ports, I began looking into other access methods and I started researching remote access tools. Their purpose is to provide system administrators remote access to the system, but when used for malicious purposes they are known as a Remote Access Trojan or RAT. Essentially the RAT consists of a client and server program with the client server executable being stored on the victims device and starting up a server between the victim and host upon opening of the executable. RATS hide their true nature from firewalls by changing their name, size, and even behavior or encryption methods. The RAT I decided to implement was TheFatRat a large public exploiting tool used for generating backdoors. The tool generates a variety of different payloads for different operating systems that can be executed on windows, android, linux and mac with the ability to bypass most AV software. There was on small issue because TheFatRat doesn't like running on OS operating systems so I had to utilize a virtual box that I then installed kali linux on. After I got the virtual box up and running I was able to download TheFatRat by cloning it from its public github. The process from there was rather straight forward after opening TheFatRat I was prompted with a home page that gave me multiple payload types. I chose the msfvenom payload that utilizes metasploit a Ruby based open source framework for penetration testing. The main reason I ran kali linux is because it has metasploit framework built in so all I had to install was TheFatRat and then create the payload. After choosing the exploit type I have to input a LHOST which is just my local IP address providing the address for the server to return to. Next I input a LPORT which is just choosing a port for the servers to communicate on for this I selected 6666 and gave the payload a name. The file was created and saved to my system now I just had to create the listener on my system to listen for the executable to be opened. This process is simple I return to the main menu of TheFatRat and select the "create auto listeners" option where I can then choose listeners according to operating system. For this case I selected the linux option. The LPORT and LHOST are the same essentially this is telling the listener what port to listen on for the server once it's executed. All that remains is opening the listener and executing the payload on the victim computer as soon as I opened the payload, a meterpreter session opened on the listener granting me access to the victim.[10]

1.4.3 Meterpreter

Due to time constraints I was unable to experiment with the backdoor like I wanted to but with meterpreter connection there are multiple commands I could use to traverse and further infiltrate the system. Cd and pwd commands work normally so it is easy to traverse the system but I prefer the search command which allows you to efficiently search for specific files or repositories on the victims system. If your only goal is to wipe the victims system you can do so with a clearev command that clears the Application, System, and Security logs on any Windows system. One of the most important features for an attacker is the download command that enables you to download files from the remote machine this could allow you to implant more malware to gain more access or perform any task you can imagine with ease. This allows for more aggressive attacks of many different types. My favorite command however is the webcamlist command that enables you to display all current available webcams on the target host. If the host contains available webcams you can use webcamsnap to grab a picture from a connected web cam on the target system, and saves it to your host disc as a JPEG image. This is not only terrifying but simple to do with as few as two commands, this shows just how dangerous an attack with a meterpreter connection can be.[11]

1.4.4 Steganography

Due to time constraints I simply executed the payload normally on my laptop as a proof of concept. However in a real world scenario its hardly realistic to expect the victim to willingly execute the payload, so I researched methods of hiding executable. The method I would implement in this attack given an extended timetable would be Steganography, which is the art of hiding information by embedding messages within types of media. It can be done simply enough in the terminal by adding the executable to a zip and then using " copy /b picturename.jpg + zipfile.rar outputfilename.jpg". I would hide the executable in an unsuspecting image relatable to the victim like for instance a meme so when they click the picture to expand it the server would be created.[13]

2 Conclusion

Computers are a vital part of normal everyday life and corporations use large complex networks connected to multiple endpoints to house there business operations and make workflow easier to contain. This flexible nature is vital for the companies ability to process large quantities of data effectively and efficiently between many users. This flexible nature however is exactly want an attacker wants as well because the flexibility built into the network means if a malicious attacker gains access to the network, they are free to move around and cause damage, often without anyone's knowledge. The purpose of this study is to target a single user on a small open network but the ideas behind this attack knows no scale as Google found out in 2017 when they were hit with the largest ever recorded DDos attack. Google measures these such attacks by bits per second (bps) for attacks targeting DNS servers and network devices. The attack in 2017 was part of a six month campaign that supposedly used multiple methods of attack but at its peak it reach 2.5 Tbps, an enormous figure. These sort of attacks aren't exclusive to corporation's either entire countries can be targeted as we found out in 2020 with what is being called the SolarWinds breech which originated in Russia and accessed approximately 18,000 US government and private computer networks. Attacks of this nature and size will only increase which also means the need for individuals to monitor and fight them will as well. While the scale of these attacks is beyond this study the nature of them is not. Like the attacks stated above they revolve around tricking computers into giving authorized access to those who do not have it. Whether its over the network or through more physical means like a RAT every attack is tailor made for the victim there is no ”Universal Hack” that can hack any system at anytime regardless of hardware, operating system, and other variables. I began this process with a plan of attack but I quickly learned the most important quality in Remote Access Attack is the ability to adapt to changing environment's and build a purpose built attack specific for the victim.[9]

References

[1] National Science and Media Museum. 2021. A short history of the internet — National Science and Media Museum. [online] Available at: ¡https://www.scienceandmediamuseum.org.uk/objectsand-stories/short-history-internet: :text=The

[2] freeCodeCamp.org. 2021. What is Nmap and How to Use it A Tutorial for the Greatest Scanning Tool of All Time. [online] Available at: ¡https://www.freecodecamp.org/news/what-isnmap-and-how-to-use-it-a-tutorial-for-the-greatest-scanning-toolof-all-time/¿ [Accessed 31 March 2021].

[3] Tripwire, I., 2021. 3 Types of Network Attacks to Watch Out For. [online] The State of Security. Available at: ¡https://www.tripwire.com/state-of-security/vulnerabilitymanagement/3-types-of-network-attacks/: :text=Some

[4] “Greycampus,” Ethical Hacking. [Online]. Available: https://www.greycampus.com/opencampus/ethical-hacking/phases-of-hacking. [Accessed: 13-May-2021].

[5] Vulnerabilityassessment.co.uk. 2021. VulnerabilityAssessment.co.uk. [online] Available at: ¡http://www.vulnerabilityassessment.co.uk/scanrand.htm¿ [Accessed 1 April 2021].

[6] Adeptus-mechanicus.com. 2021. Paratrace - Sneaky Tracerouting. [online] Available at: ¡http://www.adeptusmechanicus.com/codex/paratrc/paratrc.php¿ [Accessed 1 April 2021].

[7] Deorg, IP Logger URL Shortener - Log and Track IP addresses. [Online]. Available: https://iplogger.org/. [Accessed: 13-May-2021].

[8] “Port Scanning Basics: Nmap Network Scanning,” Port Scanning Basics — Nmap Network Scanning. [Online]. Available: https://nmap.org/book/man-port-scanning-basics.html. [Accessed: 13-May-2021].

[9] Securityweek.com. 2021. Google Targeted in Record-Breaking 2.5 Tbps DDoS Attack in 2017 — SecurityWeek.Com. [online] Available at: ¡https://www.securityweek.com/google-targeted-recordbreaking-25-tbps-ddos-attack-2017¿ [Accessed 29 March 2021].

[10] Screetsec, “Screetsec/TheFatRat,” GitHub. [Online]. Available: https://github.com/Screetsec/TheFatRat. [Accessed: 13-May-2021].

[11] “Meterpreter Basic Commands,” Offensive Security. [Online]. Available: https://www.offensive-security.com/metasploit-unleashed/meterpreter-basics/. [Accessed: 13-May-2021].

[12] J. P. U. 3/29/2020, “What is Metasploit? The Beginner's Guide - Varonis,” Inside Out Security, 30-Mar-2020. [Online]. Available: https://www.varonis.com/blog/what-is-metasploit/. [Accessed: 13-May-2021].

[13] “Home,” Learn Ethical Hacking and Penetration Testing Online. [Online]. Available: https://www.hackingloops.com/steganography-hide-exe-within-the-jpeg-image-file/. [Accessed: 13-May-2021].