System Security

Attack Techniques

Any information system that is exposed to the Internet or network is vulnerable to attack. Attacks come in all shapes and sizes and an exact definition can prove problematic. For instance, Is guessing someones password an attack? What if you then proceed to read their files? Moreover what if you then delete them? Each of these could constitute an attack given the correct circumstances. Taking a leaf from Intrusion Detection, an attack can be seen as:

…actions that attempt to compromise the confidentiality, integrity or availability of a resource.

There are various different types of attack each with their own nuances and subtleties. Attacks can be classified in terms of their type, target and/or goal. But other aspects can be considered.

This chapter will provide an overview of an attacker (Intruder) before looking at the malicious software they can use to setup the infrastructure needed to perform network based attacks. The remainder of this chapter will then look at three types of network based attacks Scan-based, DNS-based and Botnet-based that they, the intruder can perform utilizing the infrastructure. Concluding this chapter is a short overview of other network attacks that can occur.

Other attacks that utilize malicious software, such as Worms, Viruses and other similar software shall also be addressed.

1.1. Intruder

The Intruder is an important part of the attacking process, they initiate it. They are the ones who press the button, type in the commands and reap the benefits. They can be classed as either a: Masquerader — those not authorized to access a resource but do so anyway; Misfeasor — a legitimate user that access resources out with their remit and/or abuses their privileges; and Clandestine User — an individual that seizes supervisory control and uses this control to hide their actions. Such individuals can be an insider or an outsider. An attackers motivation can

range from the most benign where they simply want to explore what is out their to the more serious, when they wish to turn a profit from their actions.

The two most important tools that attackers utilize are a sophisticated knowledge of intrusion techniques and persistence. The former implies a willingness to spend countless hours investigating systems to identify weak spots. While the latter concentrates on the know how to guide their actions and perform them.

As it is commonly known there are two levels of hackers, the script kiddies, the ones who simply use the tools and may not know the underlying principles and technology at length. The others are the true hackers, those who posses the wherewithal to actual build the tools, maintain them and ultimately profit from them.

1.2. Malicious Software

Malicious software is perhaps on of the more well known and oldest form of threat that can threaten a information system. These software programs can be used on their own or together as part of the aforementioned network based attacks. More often than not malicious software is used as a means to set up and establish the infrastructure needed to perform the network based attacks.

1.2.1. Trapdoors

Trapdoors are both a godsend and a nuisance for administrators. They are secret entry points into the information system that allows the user who is aware of such trapdoors to bypass any and all security mechanisms. They are primarily used during the testing and development of software, as it quickens certain actions. However if left in the actual production version of the code, an intruder can use them to gain access.

1.2.2. Logic Bomb

One of the oldest types of program threat. A Logic Bomb is code that when activated will perform some action such as file deletion, file encryption (for ransom purposes) or halt the system. In order to have gained access to a system an attacker will havedropped a logic bomb through the trapdoor.

1.2.3. Trojan Horses

Trojan Horses owe their name to Greek mythology and the sacking of Troy by the Greeks. A Trojan Horse is a innocuous piece of software or file that normally adds little to no extra functionality with its presence i.e. 10 minute games, email attachments, but contained within the program is code that is launched when the software has been activated. On launch this new code will perform some unwanted or harmful action.

1.2.4. Zombie

A Zombie is a host on a network that is under the control of an intruder. Most system owners will normally not be aware that their system is a zombie as their actions are normally hidden from the system owner. Computers are normally infected through use of a Trojan Horse, Worm or Virus which install a Rootkit that allows the attacker to control the computer. Zombies are used to co-ordinated distributed attacks.

1.2.5. Viruses

Computer Viruses are programs that distribute themselves by attaching copies of themselves to mobile software that is distributed from machine to machine. Once the virus has infected a machine it will then perform some action. Many of the above types of malicious software are used by viruses to perform their distinct phases: Dormant — the virus is idle; Propagation — distribution; Trigger — activation; and Execution — perform damage. Viruses always have code that detects whether or not a host is infected. Viruses can be classified into several categories. A Parasitic virus is one with the aim of self-propagation.Memory-Resident viruses reside in main memory and infects programs when the program is executed. Boot Sector viruses infects the boot records of a system as propagates on disk start up. A Stealth virus constantly covers its tracks from anti-virus software. The final type is that of Polymorphic viruses, such viruses constantly mutate their structure thus hindering the development of a signature.

1.2.6. Worms

Similar to a virus in that they distributed themselves from machine to machine, a Worm is a program that actively seeks outs new hosts to infect and each newly infected machine seeks out more machines to infect. It can be said that worms are

viruses at host level and not program level, as such they share the same life cycle. Worms propagate themselves through network connections and has a multitude of was to achieve this. For instance a worm can take advantage of Email, Remote Login and/or Remote Execution. The typical sequence of actions performed by a worm to propagate itself are:

- Find other infected hosts.
- Establish remote connection.
- Copy itself.

Some worm propagation techniques are discussed in Appendix app:worm:prop.

1.3. Scans

Network Scans are performed by viruses, worms and attackers in order to determine the services offered by a host. It is a form of information gathering, as together with knowledge of vulnerabilities in the services offered a malicious entity can then tailor their attack. They can also be used to determine the network topology. Scans usually send probes to specific TCP/UDP ports of a host and wait for a response, the probes themselves can even be the source of an attack. A common network tool for this isnmap.

1.4. TCP Port Scanning

Regular Scan

A Regular TCP port scan will attempt to establish a regular TCP connection with a host. The target machine then has the option to, accept, deny or ignore the connection attempt. This is an easy to implement method but is also slow.

SYN Scan

A faster variant is that of a SYN scan, like a regular scan the attacker will attempt to connect to the target if the target does not ignore or deny the connection i.e. accepts it, in that it sends a SYN/ACK. The attacker reply with a RST, thus a connection will not be established. This is much quicker than a regular scan in that there is no overhead in relation to the setting up and tearing down of a connection. However this does involve more in depth coding.

1.4.1. UDP Port Scanning

Given that UDP is connectionless there are two approaches one can take, based upon the type of response, to scan.

Negative Answer

An attacker can wait for a negative answer to be sent i.e. ICMP Port Unreachable.

Positive Answer

An attack can also wait for a positive answer to be sent back i.e. a response from a DNS query.

1.4.2. Types

Typically a network is scanned by first going through the available IP addresses associated with the network, often calledHorizontal scan. Individual hosts are then subjected to a port scan, Vertical Scan. A clever attacker will often limit the ranges in terms of scanned IP addresses and Ports used, this is a Block Scan.

1.4.3. Hiding

Scanning activity can be detected through analysis of suitable logs such as those produced by SSH and last. Thus, techniques have been developed to hide the scanning process, this can simply take the form of slowing down the scan process Slow Scanning, using multiple hosts to perform the scan in a coordinated process Distributed Scan or even using Indirect methods such as anIdle Scan.

1.4.4. Idle Scanning

Idle scans consist of an attacker spoofing the IP address of a zombie host to sent the probe i.e. SYN, to a target and asking the zombie to report on the response i.e. SYN/ACK. The problem lies in how to ask the zombie, who did not originate the request to not only identify the response by to ask in the first place. TCP Sequence numbers are abused (IPID). Idle scans consist of two stages:

First Stage

The spoofed SYN packet is sent to the target, who responds with a SYN/ACK to the zombie. At the same time the attacker will send an IPID Probe (SYN/ACK) to the zombie who responds with a RST packet that contains the IPID number.

Second Stage

As the zombie did not know that it sent the spoofed packet, it will then send a bogus RST packet to the target. This will increment the IPID. The attacker will then simply send another IPID probe to the zombie to learn the new IPID. If it has incremented then the attack knows that the port is open.

1.5. Attacking and Abusing DNS

DNS is a tool used to resolve host names to IP addresses. The goal of DNS based attacks include making a DNS server unusable for regular users i.e. DoS attacks, compromising the integrity of the information stored and also misuse DNS for some purpose. The attacks mentioned here can be adapted to other services.

1.5.1. DNS Queries

A DNS query consists of a simple request response mechanisms. A host will send the name to the DNS server using UDP on port 53 and the server will respond with the appropriate IP address. If the server does not know the address it will consult another DNS server who is aware of the corresponding IP address (authoritative). This is a recursive query.

1.5.2. Distributed Denial-Of-Service

The aim of a Denial of Service (DoS) attack is to overwhelm and crash the targeted service. This involves the sending of multiple queries to the service simultaneously. In the case of DNS this simply implies the sending of multiple DNS requests. However a single host may not have enough resources i.e. CPU and bandwith, to complete the attack. Moreover a clever defend who has recognised that a DoS is being performed will block the attackers IP address.

To counter this problem attackers will utilise a botnet to launch a Distributed DoS (DDoS). This involves the use of multiple hosts to perform the attack, thus spreading the load.

1.5.3. Cache Poisoning

The main aim of Cache Poisoning is to compromise the DNS information and make the DNS servers response to a request wrong. DNS improves the efficiency of its operation, by letting DNS clients and servers to cache the responses. Moreover DNS can contain additional entries. Unfortunately, some DNS servers do not validate the authority of the responder, who as it will be shown can corrupt this information.

Variant One

One way to poison the cache is to act as an authoritative DNS Server and in response (valid) to a request, place in the additional section wrong resolving information for alternate hosts. The figure above illustrates this attack. With this attack it is possible to perform it for entire domains and on alternate ones.

Variant Two

Another way in which to poison the cache is to intercept the response of an authoritative DNS server i.e. at a normal DNS server operation, and modify the response . This is illustrated in the figure above. This variant is not so easy. DNS uses query IDs, responses and queries carry a random ID, thus the response ID must match the query ID. As such the attack has to guess this query ID, two approaches are to use Brute Force and send hundreds of responses with different IDs or attempt to predicate the ID. Some DNS servers use(d) a flawed RNG.

Note Brute force attacks work. Some DNS servers always use the same source port to query other servers. The solution is to randomise this port. This the attacker has to guess the ID and source port.

1.5.4. Reflected DoS Attack

A variant of DDoS is that of a Reflected DDoS RDDoS or DrDoS. This is when the attack sends a DNS query using a spoofed source IP address to a DNS server. The response is then sent to host whose IP address was spoofed. Such an attack can

utilise multiple zombie computers to launch the attack on multiple DNS servers. DrDoS is a powerful attack due to the amplification of messages.

An initial DNS query is 60 bytes in length, the response is 512 bytes. This is 8.5 times bigger. A variant of DNS EDNS allows larger answers. Through the combination of different response types, answers larger than 4000 bytes are possible that is roughly Garon) studied DrDoS attacks with up to 140, 000 DNS servers the resulting bandwidth used was 10 Gb/s

When performing these attacks it is usually better to use a public DNS server for various reasons. Firstly, Availability, while a private or DNS server is typical used solely by its owners for their own ends, a public server is accessible by anyone and is a public service. As such, they are more readily available and accessible to malicious entities when compared to private DNS servers. Furthermore, the availability in terms of up time is also guaranteed by the importance that DNS plays in the use of the Internet.

Secondly, Accessibility, similarly to availability public DNS servers, by default are accessible by anyone. Thus not only are they usable by anyone, their numbers are greater than when compared to other services that are more transitory in nature i.e. anonymous ftp.

Finally, Amplification, a DNS response is greater in size than that of a DNS response. Hence this attack is highly economical due to this difference and that the victimized ip address will be inundated with data as a result of a DoS attacked using public DNS service, while the perpetrator only needs to utilize a smaller amount of bandwidth.

1.5.5. DNS Tunneling

DNS Tunnelling, is the use of DNS to connect to services and host using DNS as the transport mechanism. Mostly this is performed to avoid network use charges i.e. in airports. In order to tunnel information using DNS information must be sentupstream and downstream to a specified DNS server own by the user sending the query.

Upstream

A DNS query is sent to resolve a private URL on your domain i.e. helloworld.mydomain.nl, that is ultimately resolved using your own DNS server. The ISP's DNS server will be forced to consult you own DNS, it is in the query that the outgoing information is stored. With a DNS query it can store up to 252 characters, though the set is restricted it is not case sensitive. Thus with each character being around 5bits, each query can contain around 110 bytes.

At DNS Server

The private DNS server will take this information and perform the actions mentioned in the query and send the response in a DNS Response.

Downstream

The information is then sent back as a standard DNS response, to you host.

The main limitation with this is that the response must be less than 512 bytes to avoid fragmentation. Moreover the response must be in txt and this the character set is restricted to 7 bit ASCII and thus with 6 bits/character this results in 220 bytes. It is possible to do this with both MX-records and A-records but it is much more complicated.

1.6. Botnets: Zombie Armies

As mentioned with many of the attacks they require that a Zombie Army is available to perform various distributed attacks such as DrDoS. The problem is how do you assemble such an army and organise their efforts. Botnets classify a collection of zombie machines that are controlled by a Botmaster. The advantages of using such armies is that the zombies are the result of a worm that has no border restrictions, thus they are located around the globe. The botmaster's identity is usually hidden from both the zombie and the target. Moreover zombies and botnets are used for spamming, click-fraud, as proxies for other illegal activities and much, much more. In May 2009 58% of all spam emails were sent by botnets.In September 2009, the Zeus botnet used 3.6 million zombies and koobface utilised 2.9 million zombies,

1.6.1. Zombie Recruitment

There are three ways in which a Zombie army can be raised. The first is manually, that is when a Bot Master attacks selected host machines to gain root access. Second is through user interaction, in which the user downloads some malicious software such as a virus, trojan horse or email attachment. The third and final way is automatically, in that Worms loaded with the botnet client, are used to infect the host machines. Usually a worm manually started by the bot master and they themselves infect Patient Zero.

1.6.2. Control

The thing with worms is that they infect random hosts, thus the bot master does not know who their minions are. Thus zombies in a botnet usually report to a server and wait for commands, this could be send spam, attack or perform an update. Often this server is a public service, IRC, P2P and specialised web servers have been utilised. A popular one was/is Command& Conquer (C& C) servers. With a hacked web server stenography is used to hide the instructions in a specially created image or other file. Which the zombie knows how to read. Through the use of public servers the bot master does not reveal his identify and that the behaviour of the clients are not seen as suspicious. Figure fig:attack:botnet:software contains a screen dump of example control software.

1.7. Other Network Based Attacks

The attacks mentioned here are but a small drop in the ocean in terms of available attacks. For instance the is SQL injection, XSS cross site scripting, CSRF cross site request forgery etc.

1.8. Further Reading

The information in this chapter has been based upon the following material:

- Network Security Essentials, Chapters 9,10,11 by Stallings
- The Internet
- DNS Open Recursion - Team Cymru Research NFP White Paper
- DNS Amplifcation Attacks - from isotf.org Preliminary Version

- The Strange Tale of the Denial of Service Attacks Against GRC.COM Gibson Research Corporation Whitepaper
- DrDoS (Distributed Reflection Denial of Service): Description and analysis of a potent, increasingly prevalent, and worrisome Internet attack - Gibson Research Corporation Whitepaper

Source: http://jfdm.host.cs.st-andrews.ac.uk/notes/netsec/#_security_attacks