Global Information Assurance Certification Paper

Introduction to IP Spoofing
Victor Velasco
November 21, 2000

Introduction

This paper describes the use of IP spoofing as a method of attacking a network in order to gain unauthorized access. The attack is based on the fact that Internet communication between distant computers is routinely handled by routers which find the best route by examining the destination address, but generally ignore the origination address. The origination address is only used by the destination machine when it responds back to the source.

In a spoofing attack, the intruder sends messages to a computer indicating that the message has come from a trusted system. To be successful, the intruder must first determine the IP address of a trusted system, and then modify the packet headers to that it appears that the packets are coming from the trusted system.

In essence, the attacker is fooling (spoofing) the distant computer into believing that they are a legitimate member of the network The goal of the attack is to establish a connection that will allow the attacker to gain root access to the host, allowing the creation of a backdoor entry path into the target system.

Brief History of IP Spoofing

In the April 1989 article entitled: "*Security Problems in the TCP/IP Protocol Suite*" , author S. M Bellovin of AT & T Bell labs was among the first to identify IP spoofing as a real risk to computer networks. Bellovin describes how Robert Morris, creator of the now infamous Internet Worm, figured out how TCP created sequence numbers and forged a TCP packet sequence. This TCP packet included the destination address of his "victim" and using an IP spoofing attack Morris was able to obtain root access to his targeted system without a User ID or password.

A common misconception is that "IP spoofing" can be used to hide your IP address while surfing the Internet, chatting on-line, sending e-mail, and so forth. This is generally not true. Forging the source IP address causes the responses to be misdirected, meaning you cannot create a normal network connection. However, IP spoofing is an integral part of many network attacks that do not need to see responses (blind spoofing).

Recent Attacks using IP Spoofing

Since the initial Internet worm, a number if attacks have been made using this vulnerability samples include;

- Man–in-the-middle: packet sniffs on link between the two endpoints, and can pretend to be one end of the connection

- Routing re-direct : redirects routing information form the original host to the hacker's host (a variation on the man-in the-middle attack)

- Source routing: redirects individual packets by the hacker's host

- Blind spoofing: predicts responses from a host, allowing commands to be sent, but does not get immediate feedback

- Flooding; SYN flood fills up the receive queue from random source addresses; smurf/fraggle spoofs victims address, causing everyone to respond to the victim.

Details of an Attack

IP spoofing in brief consists of several interim steps;

- Selecting a target host (or victim).

- The trust relationships are reviewed to identify a host that has a "trust" relationship with the target host.

- The trusted host is then disabled and the target's TCP sequence numbers are sampled.

- The trusted host is then impersonated, the sequence numbers forged (after being calculated) .

- A connection attempt is made to a service that only requires address-based authentication (no user id or password).

- If a successful connection is made, the attacker executes a simple command to leave a backdoor.

Attack directed against Root

The attack is generally made from the root account of the attacker against the root account of the target host. The reason being that gaining root access to the target will allow the attacker to fully manipulate the system. This would include the loading of Trojan horses, backdoors and possible modification of data. Going through all this effort to only gain user access is less than value added for a malicious attacker.

IP Spoofing is a Blind attack

An IP spoofing attack is made in the "blind", meaning that the attacker will be assuming the identity of a "trusted" host. From the perspective of the target host, it is simply carrying on a "normal" conversation with a trusted host. In truth, they are conversing with an attacker who is busy forging IP –address packets. The IP datagrams containing

the forged IP addresses will reach the target intact, IP being a connectionless-oriented protocol which requires no handshaking. Each datagram is sent without concern for the other end).

However, the datagrams that the target sends back (destined for the trusted host) will end up in the bit bucket, the attacker will never see them. The routers between the target and attacker know the destination address of the datagrams, that being the "trusted" host, since this is where they originally came from and where they should be returned. Once the datagrams are routed there, and the information is demultiplexed on its way up the protocol stack, and once it reaches TCP, it will be discarded.

The reason for this is that a TCP connection request is initiated by a client via a SYN flag toggled on within the TCP header. Normally a server will respond to this request via the SYN/ACK to the 32 bit source address located within the IP header. Upon receipt of the SYN/ACK, the client sends an ACK to the server (completing the three way handshake) and data transfer in the form of datagrams can commence. TCP will only support a limited number of concurrent SYN requests for a particular socket. This limit applies to both complete and incomplete connections. If this backlog limit is reached, TCP will silently dump all incoming SYN requests until the pending connections can be dealt with.

So an attacker must be very smart and 'know" what the target has been sent and "know" what type of response the server is looking for. The attacker cannot "see" what the target host sends, but based on the handshaking procedure, an attacker can predict what the target host will send in response. Knowing both what has been sent and what the response will be eliminates the need to actually "see" the response. This allows the attacker to work in the "blind" and manipulate the system.

Host disabling

To impersonate the trusted host, the attacker must first disable and make certain that no network traffic gets to the trusted host. The primary method used is called SYN flooding. As described in the previous section, TCP will silently dump all incoming SYN requests until the pending connections can be dealt with.

The attacking host sends multiple SYN requests to the target (in this instance the trusted host) to load up the TCP queue with pending connections. The attacking host must also ensure that the source IP-address is spoofed and select a different, currently unreachable host, as this is where the target TCP will be sending it's response. The reason that it must be unreachable is to prevent any host from receiving the SYN/ACKS sent by the system under attack. This would result in a RST (resend) being sent back to the system under attack, foiling the attack.

The target responds with SYN/ACKS to the spoofed IP address and once the queue limit is reached, all other requests to this TCP port will be ignored. This effectively disables the "trusted host" and allows the attacker to proceed with impersonating the "trusted host".

Packet Sequence Sampling and Prediction

The attacker must next determine where in the 32 bit sequence number space the targets TCP is located. The attacker then connects to a TCP port on the target (quite often SMTP) just prior to starting an attack and completes the three-way handshake, making sure that the initial sequence number (ISN) is recorded. This process is repeated several times to determine the Round Trip Time (RTT) and the final ISN retained. The RTT is necessary to predict the next ISN.

The attacker uses the baseline ISN (from the last connect) and knows that the sequence numbers are incremented 128,000/second and 64,000 per connection. The attacker can average the time to travel to the host (½ the RTT) and then proceed on to the next phase of the attack, sending a packet with a spoofed ISN.

When the spoofed segment reaches the target, three separate actions may be taken, based on the accuracy of the prediction

- If the sequence number is exactly where TCP expects it do be, the incoming data will be placed on the next available slot in the receive buffer.

- If the sequence number is less that expected number the byte is treated as a re-transmission and the packet is discarded.

- If the sequence is greater than expected but within the bounds of the receive window, it is held by TCP pending arrival of the missing bytes.

- If the sequence number is greater than expected and out of the bounds of the receive window the segment is dropped and TCP responds with a segment that contains the expected sequence number.

Impersonating the Trusted Host

If everything goes according to the plan, the SYN/ACK will be dropped by the incapacitated "trusted" host. The attacker must then wait to give the "trusted" host (under attack) time to send the SYN/ACK (remember that the attacker cannot see this segment). Then the attacker sends an ACK to the target server with the predicted sequence number (plus one, to accommodate the ACK). If the calculations are correct, the target server will accept the ACK. The target server has then been compromised and data transfer can start.

System Compromise

After initial compromise, most attackers will install a backdoor to make it much easier to get into the system in the future. Once compromised the attacker can use it to mount additional attacks or extract data and other information.

Defense

The simplest solution is to not rely upon address-based authentication. By disabling all the r* commands and by removing all .rhosts files and clearing out the /etc/hosts.equiv file on Unix systems. This makes remote users use other type of remote connection such as telnet, ssh, or skey.

Another possible solution is encrypting all network traffic to avoid source and host destinations from being compromised.

The final recommended solution, one proposed by Bellovin in 1989 was to use random initial sequence numbering. This solution has been adopted by a number of Unix based operating systems in response to the increasing number of these type attacks during the past decade.

Conclusion

IP spoofing is less of a threat today due to the patches to the Unix Operating system and the widespread use of random sequence numbering. Many security experts are predicting a shift from IP spoofing attacks to application-related spoofing in which hackers can exploit a weakness in a particular service to send and receive information under false identities. Sendmail is one example, that when not properly configured allows anyone to send mail as president@whitehouse.gov.

As Security professionals, we must remain current with the Operating Systems that we use in our day to day activities. A steady stream of changes and new challenges is assured as the hacker community continues to seek out vulnerabilities and weaknesses in our systems and our networks.

References

Bellovin, S. M. (1989, April). Security Problems in the TCP/IP Protocol. Computer Communication Review, Vol 19, No. 2, 32-48.
[On Line], Available; http://www.ja.net/CERT/Bellovin/TCP-IP_Security_Problems.html

Computer Incident Advisory Committee (CIAC) (1995). Advisory Notice F-08 Internet Spoofing and Hijacked Session Attacks.
[On-line], Available: http://ciac.llnl.gov/ciac/bulletins/f-08/shtml

Daemon9. (1996, June).IP Spoofing Demystified. Phrack Magazine Review, Vol 7, No. 48, 48-14.
[On Line], Available; http://www.fc.net/phrack/files/p48/p48-14.html

Donkers, A. (1998, July). Are You really Who You Say You Are? System Administrator ,Vol 7, No. 7, 69-71.
[On-line], Obtained from the University of Phoenix Student Library (Requires access code)

Nice Network article (2000). Underground:Hacking:Methods:Technical:Spoofing
[On-line], Available:
http://advice.networkice.com/Advice/Underground/Hacking/Methods/Technical/Spoofing/default.htm

Microsoft Press (1997). Internetworking with TCP/IP on Windows NT 4.0 Redmond: Microsoft Press.