Mitnick's Christmas: Understanding the Perils of Source Address-Based Authentication

Summary

On December 25, 1994, Kevin Mitnick launched an attack on a computer owned by Tsutomu Shimomura at the San Diego Supercomputer Center (SDSC). The two main techniques Mitnick used to exploit Shimomura's systems were: 1) IP source address spoofing, to appear as a trusted machine on the network, and 2) TCP sequence counter prediction, to validate forged ACK packets sent under the spoofed source IP. Mitnick used these forged packets to connect to one of Shimomura's machines on a privileged port, establishing a one-way connection that enabled him to steal proprietary cellular network software via file transfer.

Understanding the Attack

Mitnick’s attack on Shimomura exposed vulnerabilities in privileged communication that relies on address-based authentication. While a legitimate way to configure inter-computer communication, this case study highlights multiple mechanisms that allowed Mitnick to access restricted systems. Some vulnerabilities stem from the TCP protocol, such as the three-way handshake and sequence counters, while others are based on external software handling TCP connections, like connection queues. This study explores the two primary mechanisms that enabled Mitnick’s successful attack: IP source address forgery and TCP sequence counter predictability.

Mitnick’s attack targeted cellular software, as shown by subsequent intellectual property thefts where similar proprietary cellular software was stolen. In the 1990s, cellular networks were rapidly expanding, and Mitnick recognized that obtaining advanced software would give him an edge in the evolving security landscape. His attack targeted companies like Motorola, which held significant influence over the cellular market. However, by publishing his findings, Shimomura’s analysis enabled other security engineers to respond to similar threats.

Shimomura's analysis revealed the relationship between the machines Mitnick used to access his files. Generally, this type of attack requires a target machine (A), a machine with privileged access to A (call it B), and a forged IP address for the LAN connecting A and B. Although Shimomura's topology differed, the attack's principles were similar. Mitnick targeted the server instance, probing its relationship with other machines through commands:

```
14:09:32 toad.com# finger -l @target       // display login info about
"target"
14:10:21 toad.com# finger -l @server       // display login info about
"server"
14:10:50 toad.com# finger -l root@server   // see if "server" is
privileged
14:11:07 toad.com# finger -l @x-terminal
14:11:38 toad.com# showmount -e x-terminal  // show directories of "x-
terminal"
14:11:49 toad.com# rpcinfo -p x-terminal    // get port info of remote
procedures
14:12:05 toad.com# finger -l root@x-terminal // check x-term privileges
```

Mitnick used these initial probes to discover a login service hosted on port 513 (a privileged port) of the server machine. After six minutes, he sent SYN packets from a forged IP address to this service, saturating the connection queue. The server, attempting to complete the TCP handshake with SYN-ACK packets, waited indefinitely for connections from a non-existent client, isolating it from all other connections. This enabled Mitnick to send spoofed packets with reduced risk of legitimate SYN-ACK or RST packets reaching the server and interrupting his spoofed session.

With this foothold, Mitnick established a session with the X-terminal instance by imitating the server. His assumption was that the X-terminal would trust packets from the server's IP and port 513. Mitnick needed to forge a SYN request from the server, but needed to predict the correct TCP sequence number so his packets would not be discarded. These sequence numbers are critical in TCP for reliable data transmission. To establish the connection, the sequence number must be exactly one increment higher than the previously received packet.

To understand how TCP sequence numbers were generated, Mitnick sent SYN packets to the X-terminal, observing responses to find that sequence numbers were incremented by 128,000.

```
14:18:34.885071 x-terminal.shell > apollo.it.luc.edu.983: S
2024000000:2024000000(0) ack 1382727008 win 4096
// NOTICE: 2024000000

14:18:35.395723 x-terminal.shell > apollo.it.luc.edu.982: S
2024128000:2024128000(0) ack 1382727009 win 4096
// NOTICE: 2024128000
```

This was the last piece of information Mitnick needed to complete the spoofing attack. By controlling all traffic in and out of the server machine, he monitored the X-terminal's sequence counter in the handshake, knowing no other connections would alter the sequence numbers. Though he could not see the sequence number returned by his initial SYN packet to the X-terminal, he calculated it from his data to create a valid ACK response.

```
14:18:36.245045 server.login > x-terminal.shell: S
1382727010:1382727010(0) win 4096
14:18:36.755522 server.login > x-terminal.shell: . ack 2024384001 win
4096
```

Here, we see a one-way connection established from the server to the X-terminal, although it was Mitnick impersonating the server. The forged packet succeeded.

In the final attack stages, Mitnick used his one-way connection to send three more spoofed packets with calculated sequence numbers to execute `"echo ++ >>/.rhosts"`, adding a wildcard to the list of known hosts using `rsh` or `rlogin`. This allowed him to connect to the X-terminal from any IP address for post-exploitation. Mitnick then closed the forged connection and sent an RST flag to all half-open connections in the server's queue. Using his new access, he connected to Shimomura's X-terminal to install a kernel module, enabling him to steal proprietary cellular network software.

While future implementations can avoid some mechanisms that enabled Mitnick's attack, certain TCP design elements, like sequence numbers, pose inherent challenges. Even with random seeding, sequence prediction remains possible, as TCP requires predictable sequences for reliable data transmission. Thus, eliminating these vulnerabilities entirely from TCP systems is difficult.

Steven Bellovin of Pacific Bell noted that while multiple defenses exist, each has limitations based on protocol and implementation. For example, using network topology to block packets with local addresses from entering through outside interfaces is effective if only local machines are trusted, but it fails if external trust is granted or if attackers, like Mitnick, identify trust patterns.

Another solution is to prevent external connections using firewalls. However, firewalls that rely on address-based authentication are vulnerable if attackers reverse-engineer firewall rules. Bellovin cautioned that address-based authentication remains exploitable, even if attackers cannot see responses. Bellovin ultimately advised against address-based authentication, recommending cryptographic methods as a more secure alternative.

If address-based authentication must be used, a multi-layered approach with robust logging is a strong option. External connections could be secured with a VPN, requiring authentication before interaction, while internal and external firewalls could monitor and control trusted and public traffic. This limits address-based authentication to internal networks and applies Bellovin's topological strategy, while cryptographic authentication safeguards against external threats.

Sources:

https://www.giac.org/paper/gsec/1929/kevin-mitnick-hacking/100826

https://www.crime-research.org/library/cybercrime2.html

https://www.cs.columbia.edu/~smb/papers/acsac-ipext.pdf

http://www.takedown.com/coverage/tsu-post.html

https://yarchive.com/risks/ip_spoofing.html