Spoofing

Sources:

- *Spoofed/Forged Email*, CERT, www.cert.org/tech_tips/email_spoofing.html
- *Hackers Beware* by Eric Cole
- *Spoofing*, www.networkice.com
- *Web Spoofing – What are you looking at?* by Paul O'Brien, rr.sans.org/threats/web_spoof.php.
- *Introduction to IP Spoofing* by V. Velasco, rr.sans.org/threats/intro_spoofing.php

In *spoofing*, an attacker impersonates someone else.

This allows him to exploit their access privileges.

Spoofing can also make it harder to track down an attacker.

Types of spoofing:

- *IP spoofing*: Attacker uses IP address of another computer to acquire information or gain access
- *Email spoofing*: Attacker sends email but makes it appear to come from someone else
- *Web spoofing*: Attacker tricks web browser into communicating with a different web server than the user intended.
- *Non-technical spoofing*: Spoofing by “social engineering”

IP Spoofing

IP spoofing is the creation of TCP/IP packets with somebody else's IP address in the header.

Routers use the destination IP address to forward packets, but ignore the source IP address.

The source IP address is used only by the destination machine, when it responds back to the source.

When an attacker spoofs someone's IP address, the victim's reply goes back to that address.

Since the attacker does not receive packets back, this is called a *one-way attack* or *blind spoofing*.

To see the return packets, the attacker must *intercept* them.

IP spoofing is an integral part of many network attacks that do not need to see responses.

In many systems, access permissions and trusts have been based on IP addresses.

Attacks using IP spoofing [Velasco]:

- *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 from the original host to the hacker's host

- *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.

Basic types of IP spoofing attacks [Cole]:

- Basic address change
- Use of source routing to intercept packets
- Exploitation of trust relationships on UNIX machines

Steps in IP spoofing attack [Velasco]:

1. Selecting a target host (or victim).

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

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

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

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

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

An IP spoofing attack is usually made from the root account of the attacker against the root account of the target host.

Compromising the root account of the target gives the attacker administrative privileges.

Since an IP spoofing attack is “blind”, the attacker must know what the target has been sent and what response it expects.

Host Disabling

The attacker must first disable the trusted host and ensure that no network traffic gets to it.

The primary method for doing this is called *SYN flooding*.

A TCP connection request is initiated by a client by setting the SYN flag in the TCP header.

Normally a server will respond with SYN/ACK to the source address from 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 can begin.

TCP supports a limited number of concurrent SYN requests for a particular socket.

Additional connection requests will be discarded until pending connections have been dealt with.

In SYN flooding, the attacking host sends multiple SYN requests to the target (the trusted host) to fill the TCP queue with pending connections.

The source IP address on these requests must also be spoofed with the address of a currently unreachable host.

This prevents any host from receiving SYN/ACKs sent by the system under attack.

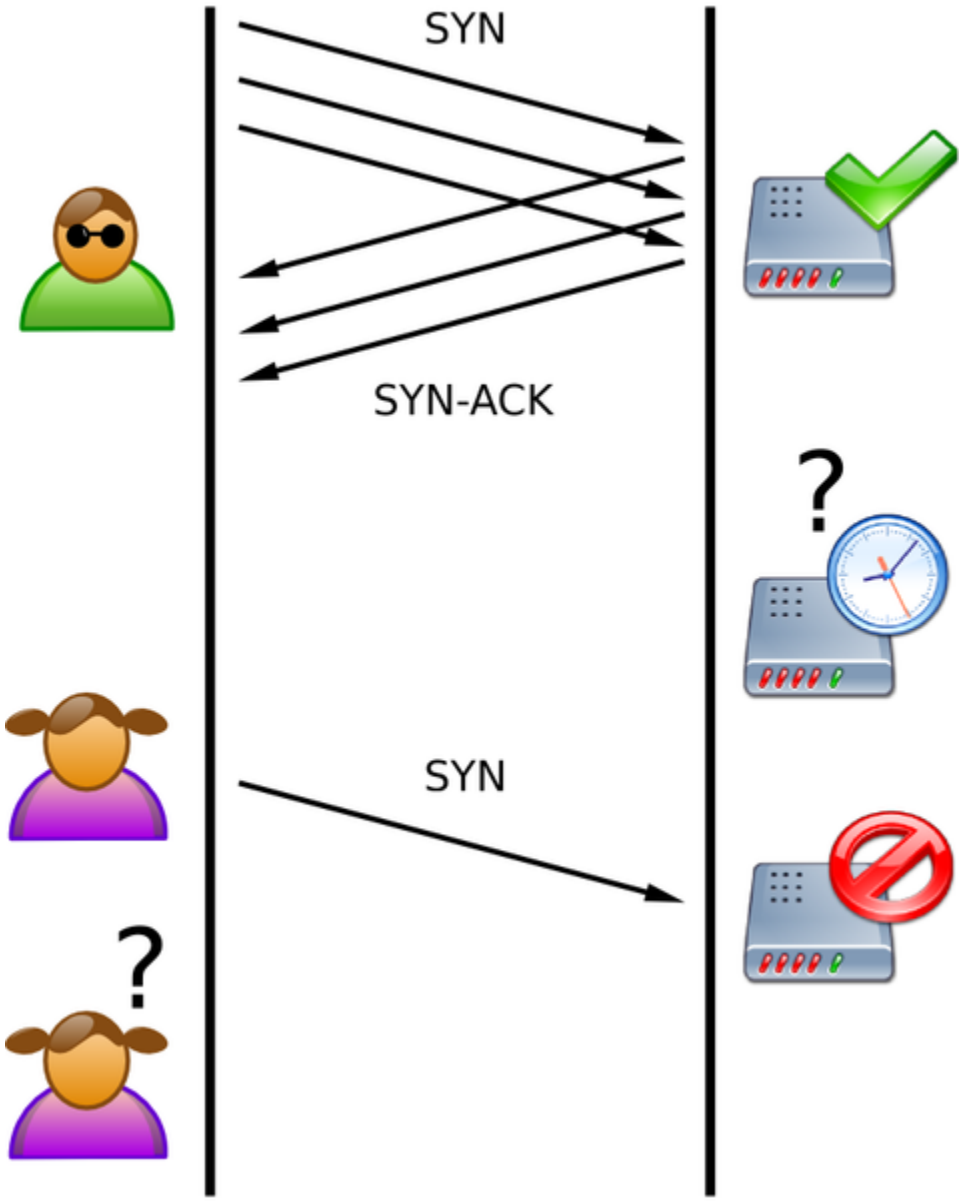
This would cause a RST to be sent back, spoiling the attack.

The target responds with SYN/ACKs to the spoofed IP address.

The attacker does not send any ACKs.

Once the TCP connection request queue is full, all other requests to this TCP port will be ignored.

This effectively disables the trusted host.



[en.wikipedia.org/wiki/File:Tcp_synflood.png]

Predicting Sequence Numbers

The attacker must then determine which sequence numbers the target will use.

Just prior to starting an attack, the attacker connects to a TCP port on the target (often SMTP) and completes a three-way handshake.

The attacker records the target's *initial sequence number* (*ISN*).

This process is repeated several times to estimate the *round trip time* (*RTT*).

The final ISN is retained.

The attacker knows the rate at which sequence numbers are incremented.

This allows him to predict the target's sequence numbers.

After initiating a spoofing attack, the attacker waits for the target's SYN/ACK to reach the trusted host, which cannot respond.

The attacker then sends an ACK to the target with the predicted sequence number plus one.

If the attacker's calculations are correct, the target will accept the ACK.

The target server has been compromised.

Most attackers will then install a backdoor to make it easier to get into the system in the future.

Countermeasures to IP Spoofing Attacks

- Don’t rely on address-based authentication.
- Limit access to configuration information on a machine.
- Use router filters to prevent packets from entering your network if they have a source address from inside it (*ingress filtering*).
- Use filters to prevent packets from leaving your network if they have a source address from outside it (*egress filtering*).
- Encrypt all network traffic.
- Use random initial sequence numbers.
- Use *SYN cookies*
  - Server sends back SYN+ACK response to the client but discards the SYN queue entry
  - If server receives a subsequent ACK from client, it reconstructs SYN queue entry using information encoded in TCP sequence number

Source Routing

One way for an attacker to see return traffic from a spoofing attack is for him to insert himself in the path the traffic would normally take.

Internet routing is normally *dynamic*.

Source routing can be used to guarantee that a packet follows a set path.

Types of source routing:

- *Loose source routing* (*LSR*): The sender specifies a list of some IP addresses that a packet must go through.

- *Strict source routing* (*SSR*): The sender specifies the exact path a packet must take.

Eight hops can be specified in an IP packet header.

(Many Internet routes require more hops than this.)

If the sender specifies source routing to the destination, the destination machine automatically uses source routing to get back to the sender.

The best way to protect against source routing spoofing is to disable source routing at your routers.

Email Spoofing

With email spoofing, someone receives email that appears to have originated from one source when it actually was sent from another source.

Purposes of email spoofing:

- Hiding sender's identity
- Impersonating someone
- Implicating someone
- Trick someone into making a damaging statement or releasing sensitive information

Note that anonymous email can be sent using an *anonymous remailer*.

Similar Email Addresses

One simple form of email spoofing is to create a plausible email address for someone and put their name in the *Alias* field of a counterfeit email.

Example:

```
From: Bill Clinton
[mailto: bubba@aol.com]
Sent: Friday, March 22, 2006
4:57 PM
To: Laura Bush
Subject: Romantic dinner?
```

To address this, digital signatures can be used.

Work email addresses should be used for work-related messages.

Modifying a Mail Client

When email is sent by a user, the *From*: address is not validated.

An attacker can use a mail client to specify whatever *From*: address he wants.

When the receiver replies, the reply goes to the *From*: address and not to the person spoofing it.

Email messages should be *logged* by mail servers, to permit the actual sender of a message to be determined.

Examination of the full email header will often reveal the actual sender.

Telnet to Port 25

Another way to perform spoofing is to *telnet* to the *Simple Mail Transfer Protocol* (*SMTP*) port (port 25) on a mail server.

Once connected, the spoofer types:

```
helo
mail from:spoofed-email-address
rcpt to: target-email-address
data
the message you want to send, followed by a period
```

In *mail relaying*, an attacker uses a mail server to send mail to someone in a different domain.

Newer mail servers don't allow mail relaying.

The most basic form of mail spoofing protection is the check that the recipient's domain is the same as the mail server's.

An attacker can also run his own mail server.

To protect against attacks against port 25:

- Make sure the latest patches for your mail server are installed.

- Make sure spoofing and relay filters are properly configured.

Web Spoofing

Web spoofing is tricking someone into visiting a web site other than the one they intend to and mimicking the intended site.

In this way, an attacker may obtain confidential information.

They can also provide false or misleading information.

One way to lure people to a malicious site is to give it a URL that is similar to that of a legitimate site, e.g.,

- `www.paypai.com`
- `wwwFirstNationalBank.com`

Another way is for the attacker to provide HTML with a mislabeled link to another page, e.g., in an email.

Example:

<a HREF="http://www.badhack.org"> American Red Cross

A sophisticated version is a man-in-the-middle attack in which links are rewritten dynamically [O'Brien]:

1. The victim requests a URL from their browser.

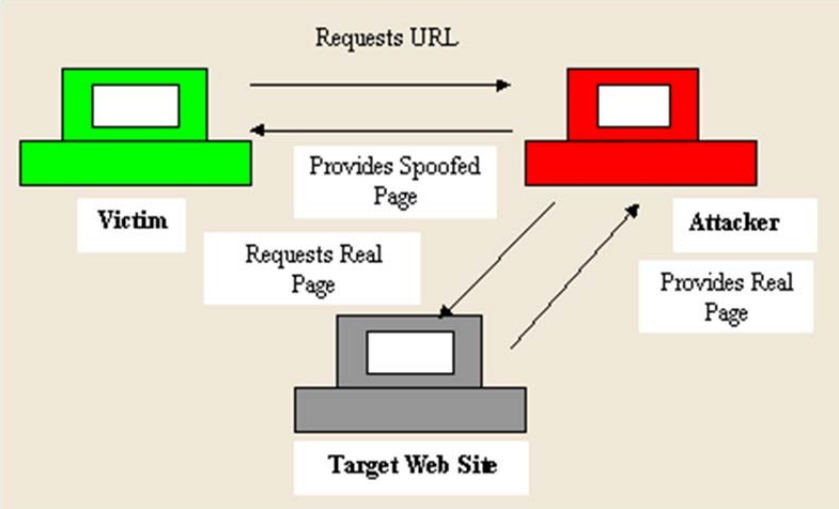
2. The attacker gets the real page requested from the World Wide Web.

3. The real server provides the page to the attacker's server.

4. The attacker then rewrites the page.

5. The attacker then provides the rewritten version to the victim.

Figure from [O’Brien]:



If the attacker's server is "www.attacker.com", he might rewrite URLs by adding this string to the front of the real URL.

If the victim follows a link on the new page, the page will again be fetched from the attacker's server.

A user can try to avoid being spoofed by checking their browser's status line before clicking on a link.

They can also check the location line after following a link to make sure the URL is what they expect.

JavaScript has been used to rewrite the status and location lines.

Web spoofing countermeasures:

- Examine the browser location line carefully.
- If in doubt, check location by other means (e.g., “Properties” in IE)
- Examine links in HTML source code.
- Disable “active” content (Java, JavaScript, Active X) in the browser.
- Disable plug-ins and helper applications.
- Ensure that your browser starts on a “secure page”.