



中山大學
SUN YAT-SEN UNIVERSITY

Module II. Internet Security

Chapter 5

Network Attack and Defence

Web Security: Theory & Applications

School of Data & Computer Science, Sun Yat-sen University

Outline

❑ 5.1 Introduction

- ◆ Network Security Crisis
- ◆ Hacking & Hackers
- ◆ Network Threats
- ◆ Steps of Network Attack
- ◆ Methods of Network Defense

❑ 5.2 Network Attacks

- ◆ Computer Network Attack
- ◆ Common Types of Network Attack
- ◆ Port Scan
- ◆ Idle Scan

❑ 5.3 Password Cracking

- ◆ The Vulnerability of Passwords
- ◆ Password Selection Strategies
- ◆ Password Cracking
- ◆ Password Cracking Tools

Outline

❑ 5.4 Buffer Overflow

- ◆ Background
- ◆ Classification
- ◆ Practicalities
- ◆ Protection

❑ 5.5 Spoofing Attack

- ◆ ARP Cache Poisoning
- ◆ DNS Spoofing
- ◆ Web Spoofing
- ◆ IP Spoofing

5.5 Spoofing Attack

5.5.1 ARP Cache Poisoning

❑ MITM Attacks

- ♦ MITM (man-in-the-middle) attacks are one of the most prevalent (流行) network attacks used against individuals and large organizations. MITM works by establishing connections to victim machines and relaying messages between them.
- ♦ In cases like these, one victim believes it is communicating directly with another victim, when in reality the communication flows through the attacking host. The attacking host can not only intercept (侦听) sensitive data, but can also inject (注入) and manipulate (操纵) a data stream to gain further control of its victims.

5.5 Spoofing Attack

5.5.1 ARP Cache Poisoning

□ ARP Cache Poisoning (ARP 缓存污染)

- ◆ ARP (Ethernet Address Resolution Protocol, 以太网地址解析协议) cache poisoning (or ARP Poison Routing) is one of the oldest forms of modern MITM attack. It allows an attacker, **on the same subnet as its victims**, to eavesdrop on all network traffic between the victims.
- ◆ It is one of the simplest to execute but is considered one of the most effective once implemented by attackers.

5.5 Spoofing Attack

5.5.1 ARP Cache Poisoning

□ ARP Cache Poisoning

- ◆ The ARP protocol was designed out of necessity to facilitate the translation of addresses between the second layer (Data Link Layer, DLL) and the third layer (Network Layer) of the OSI model (ISO/IEC 7498).
- ◆ DLL uses MAC addresses so that hardware devices can communicate to each other directly on a small scale. Network Layer uses IP addresses (most commonly) to create large scalable networks that devices are directly connected AND indirectly connected across the globe. Each layer has its own addressing scheme, and they must work together in order to make network communication happen.
- ◆ For this very reason, ARP was created with [RFC 826](#), “An Ethernet Address Resolution Protocol”.
- ◆ The ARP operation is centered around two packets, an ARP request and an ARP reply. The purpose of the request and reply are to locate the hardware MAC address associated with a given IP address so that traffic can reach its destination on a network.

5.5 Spoofing Attack

5.5.1 ARP Cache Poisoning

□ ARP Cache Poisoning

- ◆ The principal packet structure of ARP packets is shown in the following table which illustrates the case of IPv4 networks running on Ethernet. In this scenario, the packet has 48-bit fields for the sender hardware address (SHA) and target hardware address (THA), and 32-bit fields for the corresponding sender and target protocol addresses (SPA and TPA). The ARP packet size in this case is 28 bytes.
- ◆ Structure of Ethernet frame

MAC dest.	MAC source	Protocol	Payload	Padding	FCS
-----------	------------	----------	---------	---------	-----

- ◆ MAC dest.: 6 octets, Destination address
- ◆ MAC source: 6 octets, Source address
- ◆ Protocol: 2 octets, Protocol type encapsulated in payload data (Ethernet II if greater than 1534) or length (IEEE 802.3 if smaller than or equal to 1500)
- ◆ Payload: 46-1500 octets (e.g. an ARP packet)
- ◆ FCS: 4 octets, Frame Check Sequence (32-bit CRC)

5.5 Spoofing Attack

5.5.1 ARP Cache Poisoning

□ ARP Cache Poisoning

- ◆ Structure of ARP packet

HTYPE	PTYPE	HLEN	PLEN	OPER	SHA	SPA	THA	TPA
-------	-------	------	------	------	-----	-----	-----	-----

- ✧ HTYPE: 2 octets, Hardware type (e.g. 0x0001 for Ethernet)
- ✧ PTYPE: 2 octets, Protocol type (e.g. 0x0800 for IPv4)
- ✧ HLEN: 1 octet, Hardware address length (e.g. 6 for Ethernet)
- ✧ PLEN: 1 octet, Protocol address length (e.g. 4 for IPv4)
- ✧ OPER: 2 octets, Operation (1 for request, 2 for reply)
- ✧ SHA: 6 octets, Sender hardware address
- ✧ SPA: 4 octets, Sender protocol address
- ✧ THA: 6 octets, Target hardware address
- ✧ TPA: 4 octets, Target protocol address

5.5 Spoofing Attack

5.5.1 ARP Cache Poisoning

❑ ARP Cache Poisoning

- ◆ The request packet is sent to every device on the network segment and says

“Hey, my IP address is XX.XX.XX.XX, and my MAC address is XX:XX:XX:XX:XX:XX. I need to send something to whoever has the IP address XX.XX.XX.XX, but I don’t know what their hardware address is. Will whoever has this IP address please respond back with their MAC address?”

- ◆ The response would come in the ARP reply packet and effectively provide this answer,

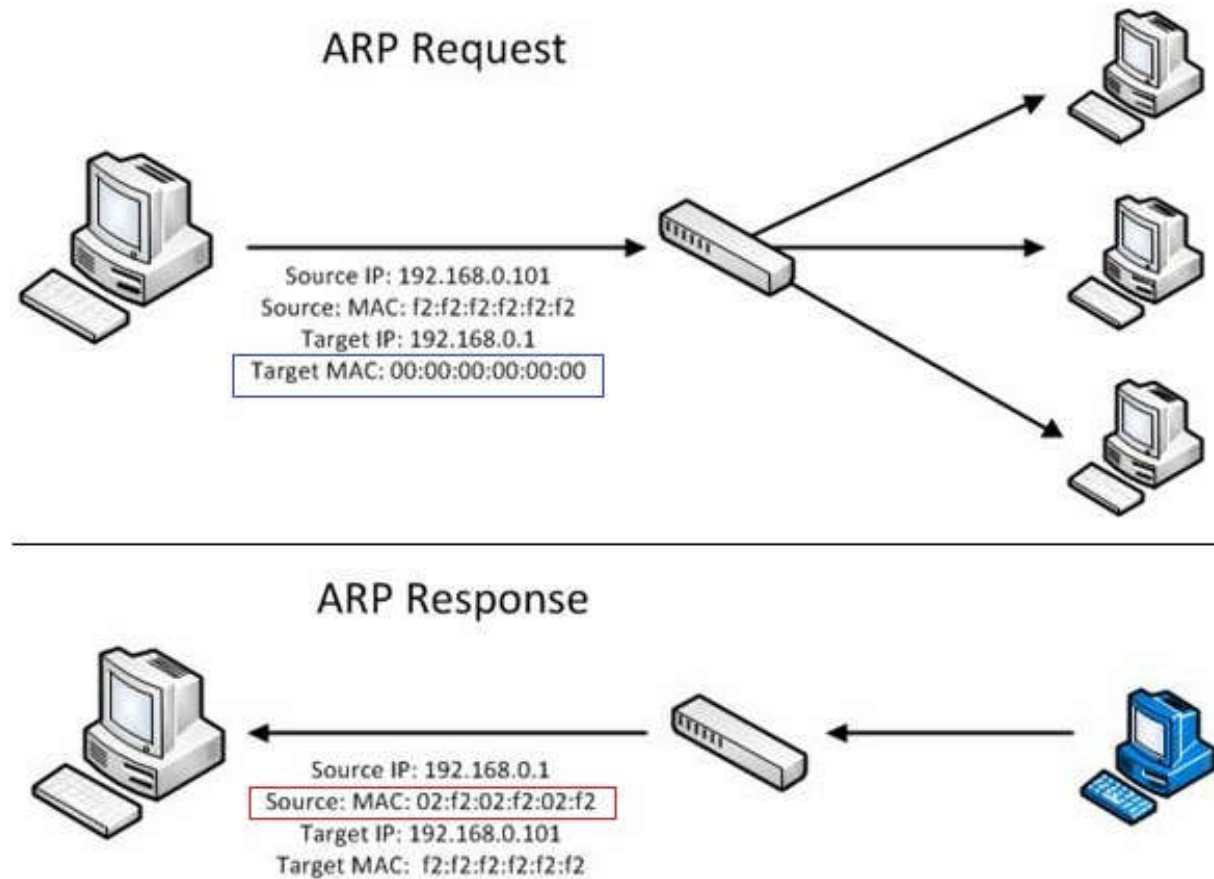
“Hey transmitting device. I am who you are looking for with the IP address of XX.XX.XX.XX. My MAC address is XX:XX:XX:XX:XX:XX.”

- ◆ Once this is completed the transmitting device will update its ARP cache table and the devices are able to communicate with one another.

5.5 Spoofing Attack

5.5.1 ARP Cache Poisoning

□ ARP Cache Poisoning



5.5 Spoofing Attack

5.5.1 ARP Cache Poisoning

❑ ARP Cache Poisoning

- ♦ *ARP Cache Poisoning* takes advantage of the insecure nature of the ARP protocol. Unlike protocols such as DNS that can be configured to only accept secured dynamic updates, devices using ARP will accept updates at any time. This means that any device can send an **ARP reply packet** to another host and force that host to update its ARP cache with the new value. Sending an ARP reply when no request has been generated is called sending a gratuitous (免费) ARP. When malicious intent is present the result of a few well placed gratuitous ARP packets used in this manner can result in hosts who think they are communicating with one host, but in reality are communicating with a listening attacker.

5.5 Spoofing Attack

5.5.1 ARP Cache Poisoning

□ ARP Cache Poisoning

- ◆ **Practice.**
 - ✧ Try to install and activate *Cain & Abel's* ARP cache poisoning features and allow your analyzing system to be the middleman for all communications between some two victims.

5.5 Spoofing Attack

5.5.1 ARP Cache Poisoning

□ ARP Cache Poisoning

- ◆ ARP Cache Poisoning is only a viable (切实可行的) attack technique when attempting to intercept traffic between two hosts on the same local area network. The only reason we would have to fear this is if a local device on our network has been compromised, a trusted user has malicious intent, or someone has managed to plug an un-trusted device into the network. Although we too often focus the entirety of our security efforts on the network perimeter (网络边界), defending against internal threats and having a good internal security posture (态度) can help eliminate the fear of the attack.

5.5 Spoofing Attack

5.5.2 DNS Spoofing

□ DNS Spoofing

- ◆ DNS spoofing is a MITM technique based on the deliberate (蓄意的) misassociation of IP addresses and DNS names.
 - ✧ It is used to supply false DNS information to a host so that when they attempt to browse, for example, www.bankofamerica.com at the IP address [XXX.XX.XX.XX](#) they are actually sent to a fake www.bankofamerica.com residing at an different IP address [YYY.YY.YY.YY](#) which an attacker has created in order to steal online banking credentials and account information from unsuspecting users.
- ◆ We are interested in how it works, how it is done, and how to defend against it.

5.5 Spoofing Attack

5.5.2 DNS Spoofing

□ DNS Spoofing

- ◆ The DNS protocol
 - ✧ The DNS protocol, defined in [RFC 1034/1035](#), is one of the most important protocols in use by the Internet. This is because DNS is the proverbial molasses that holds the bread together (DNS 是众所周知的将各个部分有效结合的机制). In a nutshell (简而言之), whenever typing in a web address such as <http://www.google.com> into the browser, a DNS request is made to a DNS server in order to find out what IP address that name resolves to. This is because routers and the devices that interconnect the Internet do not understand CHAR string “[google.com](#)”, they only understand addresses such as [74.125.95.103](#).
 - By Chris Sanders,
<http://www.windowsecurity.com/articles/understanding-man-in-the-middle-attacks-arp-part2.html>

5.5 Spoofing Attack

5.5.2 DNS Spoofing

□ DNS Spoofing

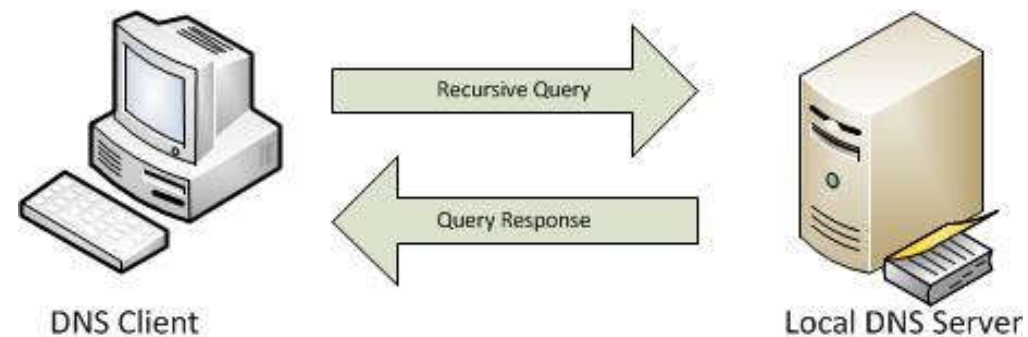
- ◆ DNS transaction
 - ✧ A DNS server itself works by storing a database of entries (called resource records) of IP address to DNS name mappings, communicating those resource records to clients or other DNS servers. A basic DNS transaction is shown in next slide.
- ◆ DNS functions in a query/response type format
 - ✧ A client wishing to resolve a DNS name to an IP address sends a query to a DNS server, and the server sends the requested information in its response. From the clients' perspective, the only two packets that are seen are this query and response.

5.5 Spoofing Attack

5.5.2 DNS Spoofing

□ DNS Spoofing

- ◆ A DNS Query & Response



5.5 Spoofing Attack

5.5.2 DNS Spoofing

□ DNS Spoofing

- ◆ DNS Recursion (DNS 递归查询)

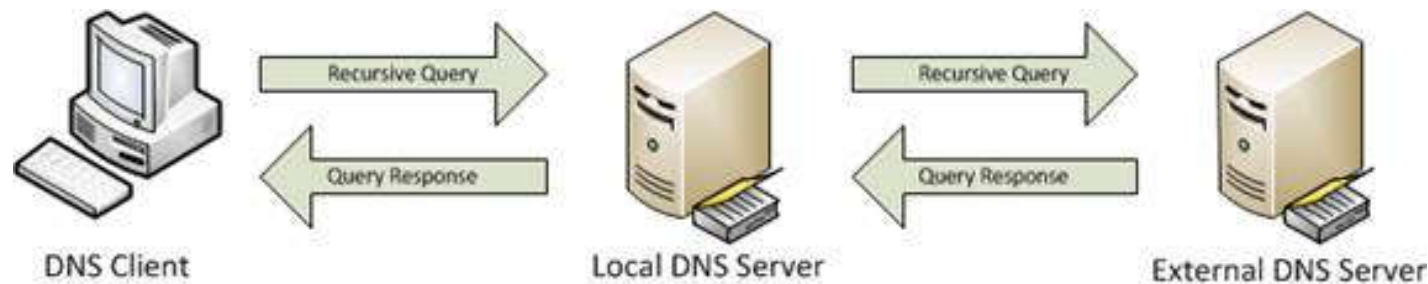
- ✧ A bit more complicated scenario is about DNS recursion. Due to the hierarchical nature of the DNS structure of the Internet, DNS servers need the ability to communicate with each other in order to locate answers for the queries submitted by clients. After all, it might be fair to expect our internal DNS server to know the name to IP address mapping of our local intranet server, but we can't expect it to know the IP address correlated with Google or Dell. This is where recursion comes into play. Recursion is when one DNS server queries another DNS server on behalf of a client who has made a request. Basically, this turns a DNS server into a client itself, seen the Figure in next slide.

5.5 Spoofing Attack

5.5.2 DNS Spoofing

□ DNS Spoofing

- ◆ A DNS Recursion



5.5 Spoofing Attack

5.5.2 DNS Spoofing

□ DNS Spoofing

- ◆ DNS ID Spoofing
 - ✧ Every DNS query that is sent out over the network contains a uniquely generated identification number to identify queries and responses and tie them together. This means that if our attacking computer can intercept a DNS query sent out from a target device, all we have to do is create a fake packet that contains that identification number in order for that packet to be accepted by that target.

5.5 Spoofing Attack

5.5.2 DNS Spoofing

□ DNS Spoofing

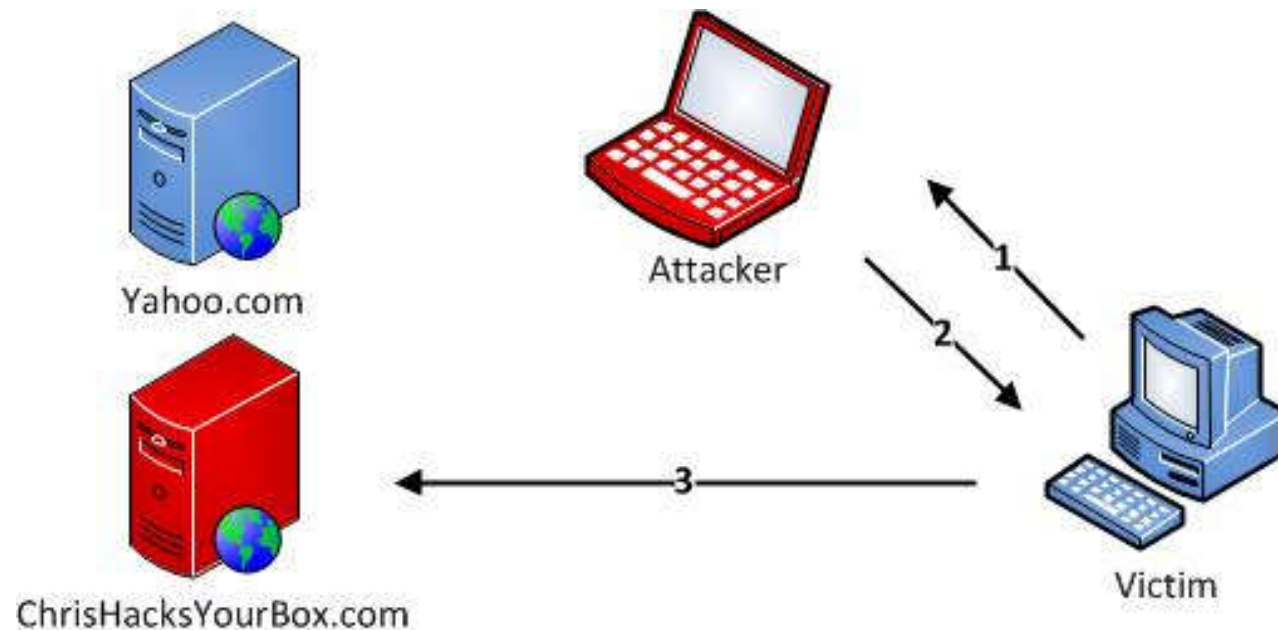
- ◆ DNS ID Spoofing
 - ✧ We will complete this process doing two steps with a single tool.
 - First, we ARP cache poison the target device to reroute its traffic through our attacking host so that we can intercept the DNS request, and then we actually send the spoofed packet. The goal of this scenario is to get users on the target network to visit our malicious website rather than the website they are attempting to access.
 - A depiction of this attack is shown in next slide.

5.5 Spoofing Attack

5.5.2 DNS Spoofing

□ DNS Spoofing

- ◆ DNS ID Spoofing



1. Legitimate DNS Request Destined for DNS Server
2. Fake DNS Reply from Listening Attacker
3. Victim begins communicating with malicious site as a result

5.5 Spoofing Attack

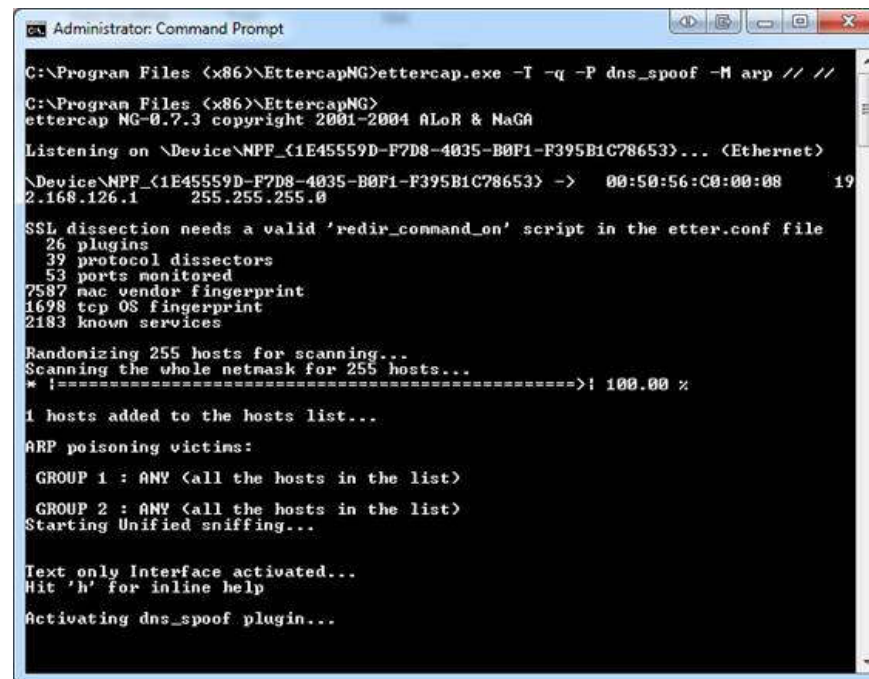
5.5.2 DNS Spoofing

❑ DNS Spoofing

- ◆ DNS ID Spoofing

 - ✧ Practice.

 - Applying DNS ID Spoofing: install and activate Ettercap.
<http://ettercap.sourceforge.net/>



```
Administrator: Command Prompt

C:\Program Files (x86)\EttercapNG>ettercap.exe -I -q -P dns_spoof -M arp // //
C:\Program Files (x86)\EttercapNG>
ettercap NG-0.7.3 copyright 2001-2004 ALOR & NaGA

Listening on \Device\NPF_{1E45559D-F7D8-4035-B0F1-F395B1C78653}... (Ethernet)
\Device\NPF_{1E45559D-F7D8-4035-B0F1-F395B1C78653} -> 00:50:56:C0:00:08 19
2.168.126.1 255.255.255.0

SSL dissection needs a valid 'redir_command_on' script in the etter.conf file
 26 plugins
 39 protocol dissectors
 53 ports monitored
7587 mac vendor fingerprint
1698 tcp OS fingerprint
2183 known services

Randomizing 255 hosts for scanning...
Scanning the whole netmask for 255 hosts...
* !----->! 100.00 %

1 hosts added to the hosts list...

ARP poisoning victims:
GROUP 1 : ANY <all the hosts in the list>
GROUP 2 : ANY <all the hosts in the list>
Starting Unified sniffing...

Text only Interface activated...
Hit 'h' for inline help

Activating dns_spoof plugin...
```

5.5 Spoofing Attack

5.5.2 DNS Spoofing

❑ Defending against DNS Spoofing

- ◆ DNS spoofing is difficult to defend against due to the attacks being mostly passive by nature. Typically, we will never know our DNS is being spoofed until it has happened. What we get is a webpage that is different than what we are expecting. In very targeted attacks it is very possible that we may never know that we have been tricked into enter our credentials into a false site until we receive a call from our bank.
- ◆ There are still a few things that can be done to defend against these types of attacks:
 - ✧ Secure our internal machines
 - ✧ Don't rely on DNS for secure systems
 - ✧ Use IDS
 - ✧ Use DNSSEC

5.5 Spoofing Attack

5.5.2 DNS Spoofing

❑ Defending against DNS Spoofing

- ◆ Secure our internal machines
 - ✧ Attacks like these are most commonly executed from inside the network.
 - ✧ Having a good internal security posture (态度) is good for defending against internal threats.
 - ✧ Secure our network devices to make less of a chance of those compromised hosts being used to launch a spoofing attack.
- ◆ Don't rely on DNS for secure systems
 - ✧ On highly sensitive and secure systems, use local hosts file for sensitive name resolution data.
- ◆ Use IDS
 - ✧ An IDS (Intrusion Detection System), when placed and deployed correctly, can typically pick up on most forms of ARP cache poisoning and DNS spoofing.

5.5 Spoofing Attack

5.5.2 DNS Spoofing

❑ Defending against DNS Spoofing

- ◆ Use DNSSEC
 - ✧ DNSSEC (*Domain Name System Security Extensions*, DNS 安全扩展, [RFC2535](#), 1999) and DNSSEC specifications (named DNSSEC-bis, DNS Security Introduction and Requirements, [RFC4033](#), 2005) are newer alternatives to DNS that use digitally signed DNS records to ensure the validity of a query response. The Extensions add data origin authentication and data integrity to DNS through the use of cryptographic digital signatures. DNSSEC is not yet in wide deployment but has been widely accepted as “the future of DNS”. This is so much so that the United States Department of Defence has mandated that all MIL and GOV domains begin using DNSSEC. Other TLDs (top level domains) such as .edu, .net, and .com implemented DNSSEC in 2010/2011.
 - ✧ Google Public DNS (2013) is a freely provided, public DNS service, fully supporting DNSSEC.

5.5 Spoofing Attack

5.5.3 Web Spoofing

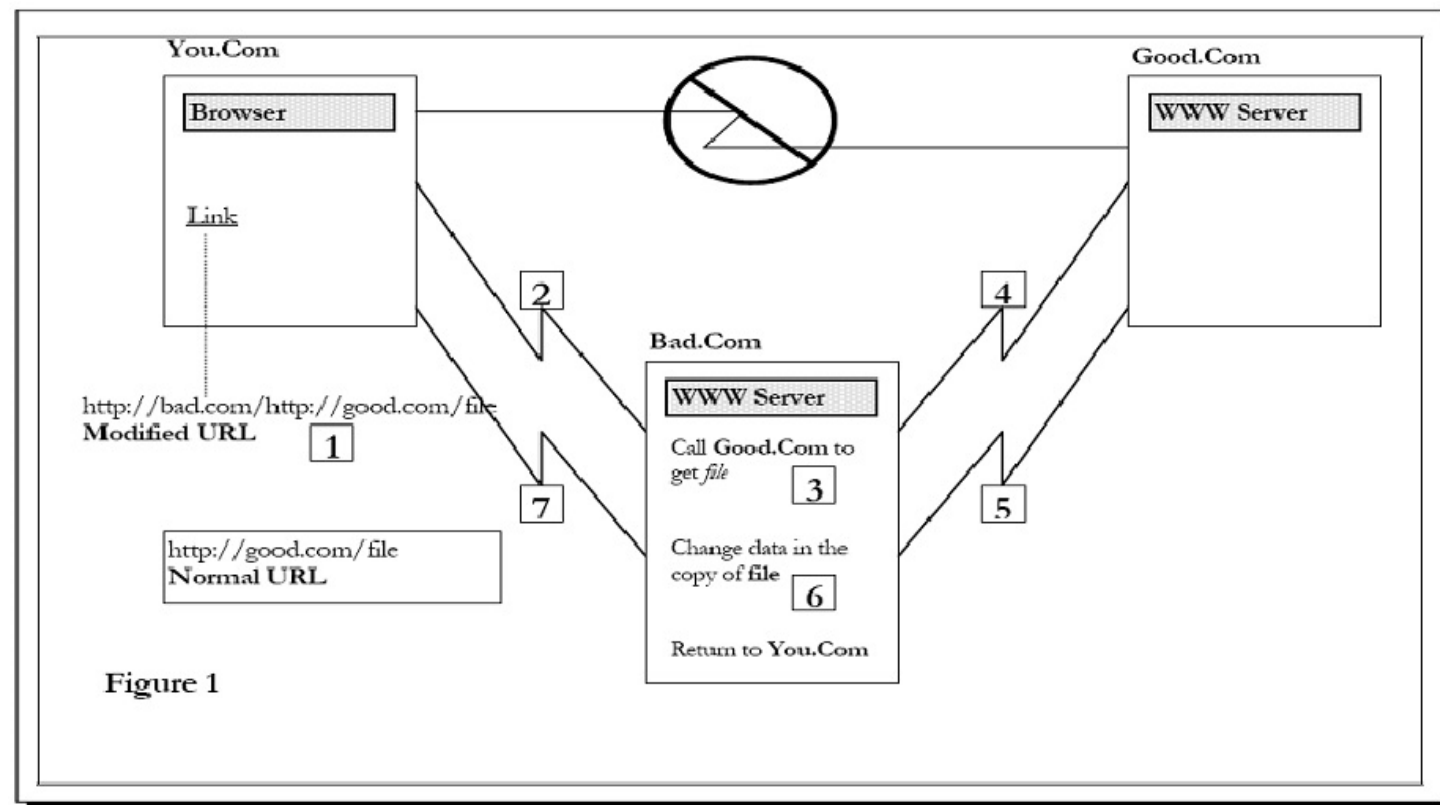
□ What is Web Spoofing

- ◆ Aka *Phishing Attack* (网络钓鱼攻击)
- ◆ Pretending to be a legitimate site
 - ✧ Attacker creates convincing but false copy of the legitimate site.
 - Fake Web looks and feels like the real one.
 - ✧ Attacker controls the false web by surveillance.
- ◆ Stealing personal information such as login ID, password, credit card credential, bank account, and much more.
- ◆ Modifying integrity of the data from the victims.

5.5 Spoofing Attack

5.5.3 Web Spoofing

□ How Web Spoofing Works



5.5 Spoofing Attack

5.5.3 Web Spoofing

❑ Different Types of Web Spoofing

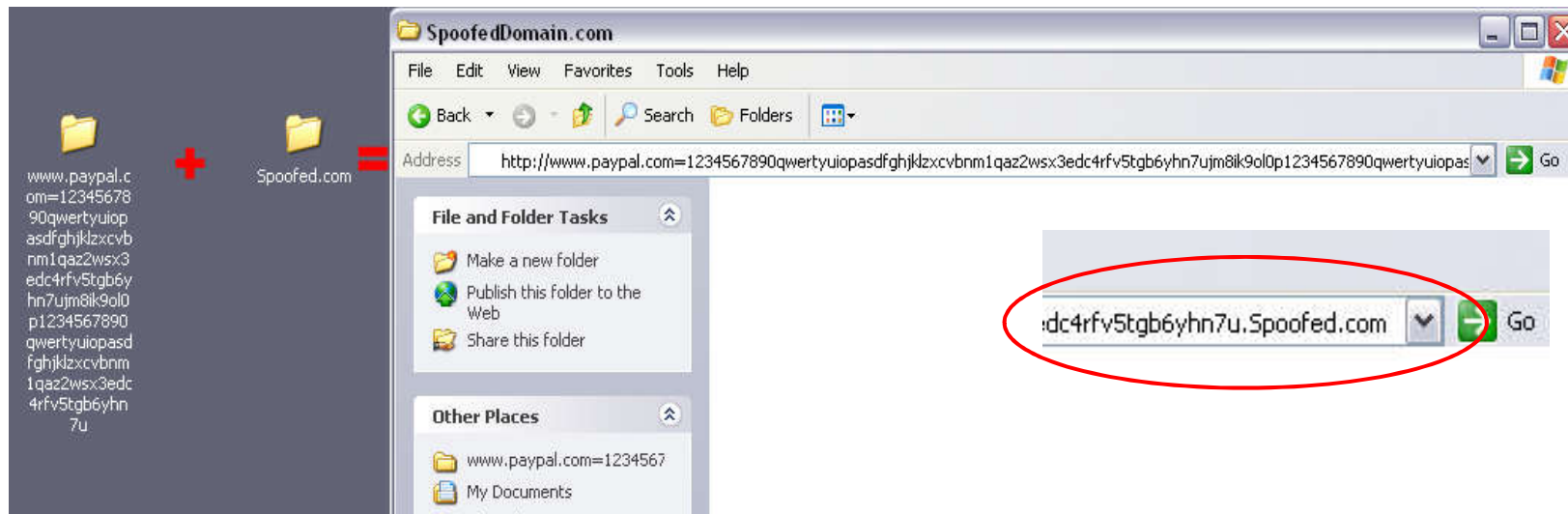
- ◆ DNS server spoofing attack
 - ✧ One of the most complex types of attack
 - ✧ Alter a domain name to point to different IP address
 - ✧ Redirect to a different server hosting a spoofed site
- ◆ Content theft
 - ✧ A copy of a site can be created from the original by saving all the publicly accessible pages, images, and scripts from a site to another server.
 - ✧ Can be done automated by using programs called “spiders”

5.5 Spoofing Attack

5.5.3 Web Spoofing

❑ Different Types of Web Spoofing

- ◆ Subdomain Spoofing
 - ✧ Tricking Internet user that they are on the correct URL
 - ✧ Make the URL long enough so that the user cannot see the entire URL (IE: 2803; Firefox: 65536; Chrome: 8182)
 - Normal subdomain: <http://subdomain.domain.com>



5.5 Spoofing Attack

5.5.3 Web Spoofing

□ Different Types of Web Spoofing

- ◆ And more...
 - ✧ IP Address as URL
 - ✧ Email with HTML attached
 - ✧ Frameless Pop-up
 - ✧ ...

5.5 Spoofing Attack

5.5.3 Web Spoofing

❑ How to Spot a Spoofed Webpage

- ◆ URL (the easiest way to detect the attack)
 - ✧ Triple check the spelling of the URL
 - ✧ Look for small differences such as a hyphen (-) or an underscore (e.g. sun_trust.com vs. sun-trust.com)
- ◆ Mouse over message
 - ✧ Be careful: this can be spoofed too
- ◆ Beware of pages that use server scripting such as PHP. That will make it easy to obtain your information
- ◆ Beware of javascripting as well
- ◆ Beware of longer than average load times

5.5 Spoofing Attack

5.5.3 Web Spoofing

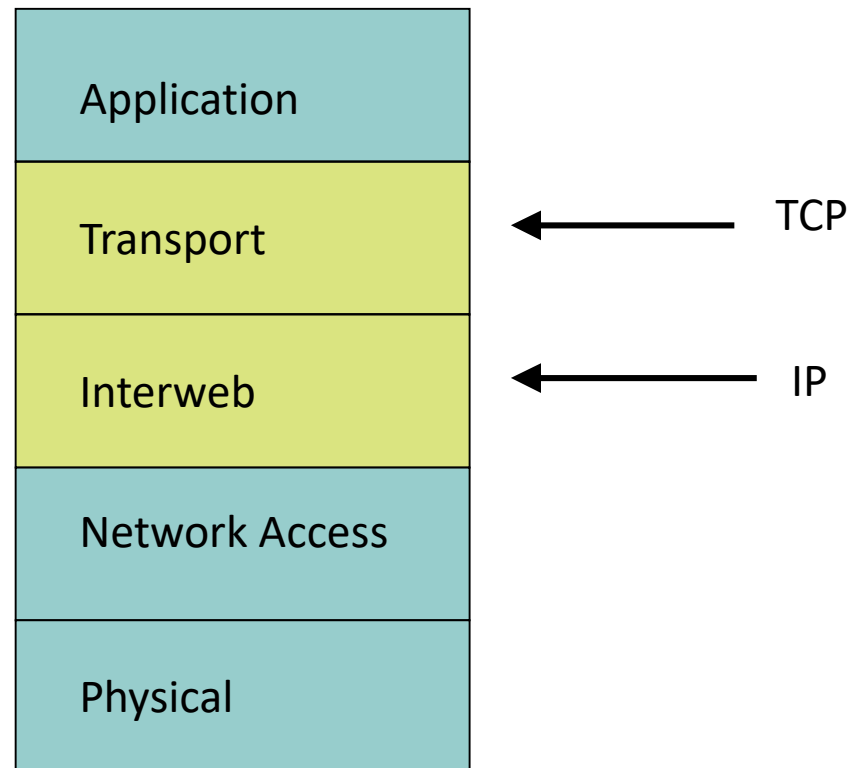
❑ How to Spot a Spoofed Webpage

- ◆ APWG: Anti-Phishing Working Group
 - ✧ An international consortium (国际联盟) that brings together businesses affected by phishing attacks, security products and services companies, law enforcement agencies, government agencies, trade association, regional international treaty organizations (区域性的国际条约组织) and communications companies.
 - ✧ Founded in 2003 by *David Jevans*, the APWG has more than 3200+ members from more than 1700 companies and agencies worldwide. Member companies include leading security companies such as BitDefender, McAfee, Symantec, VeriSign, IronKey and Internet Identity. Financial Industry members include the ING Group, VISA, Mastercard and the American Bankers Association.
<http://www.antiphishing.org/resources.html#apwg>

5.5 Spoofing Attack

5.5.4 IP Spoofing

□ TCP/IP – in brief



5.5 Spoofing Attack

5.5.4 IP Spoofing

□ TCP/IP – in brief

- ◆ IP is the Internet layer protocol.
- ◆ It does not guarantee delivery or ordering, only **does its best** to move packets from a source address to a destination address.
- ◆ IP addresses are used to express the source and destination.
- ◆ IP assumes that each address is unique within the network.

5.5 Spoofing Attack

5.5.4 IP Spoofing

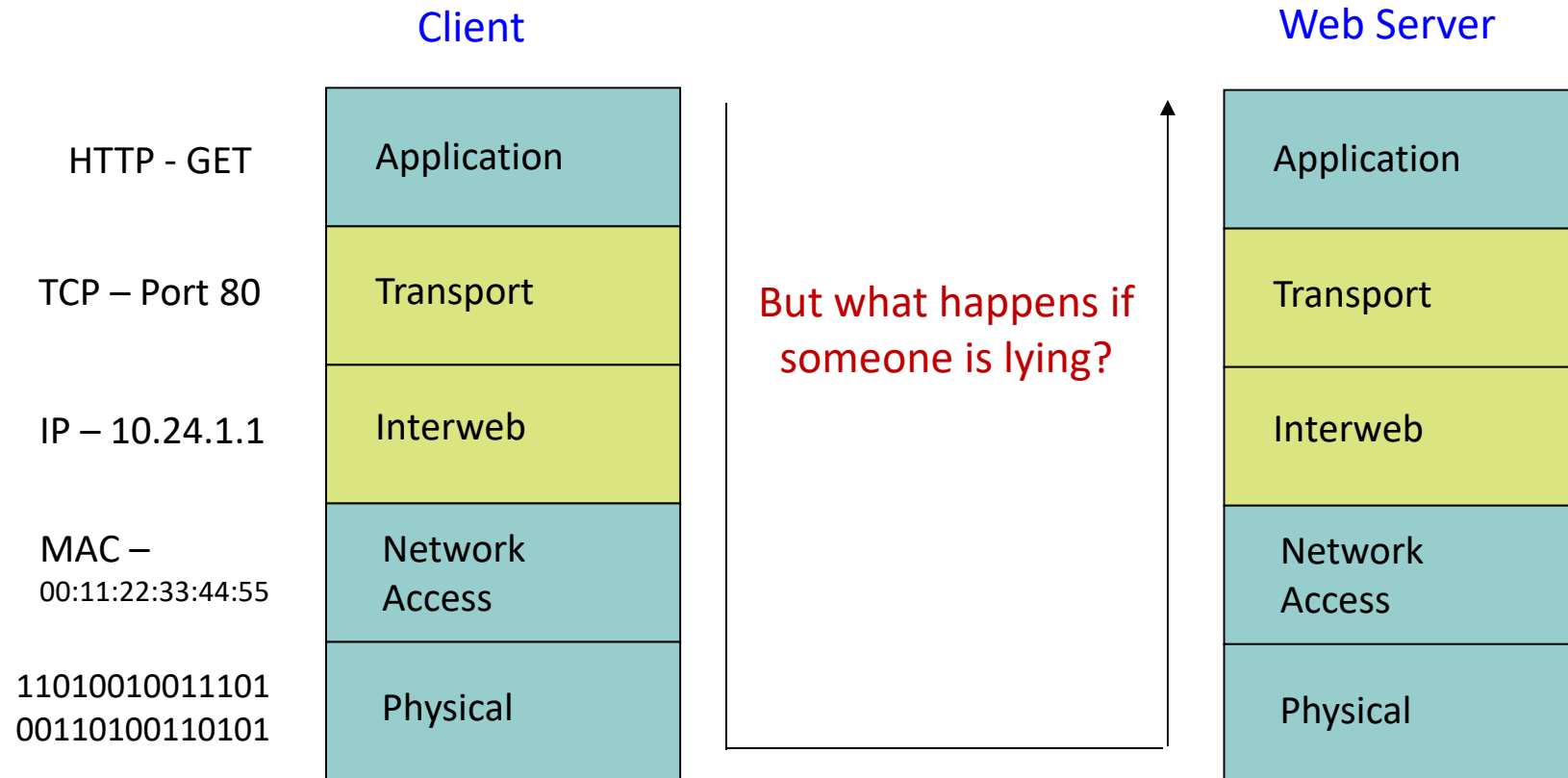
❑ TCP/IP – in brief

- ◆ TCP is the transport layer protocol.
- ◆ It guarantees delivery and ordering, but **relies upon IP** to move packets to proper destination.
- ◆ Port numbers are used to express source and destination.
- ◆ Destination Port is assumed to be awaiting packets of data.

5.5 Spoofing Attack

5.5.4 IP Spoofing

□ TCP/IP – in brief



5.5 Spoofing Attack

5.5.4 IP Spoofing

❑ IP Spoofing

◆ Overview

- ✧ Basically, IP spoofing is lying about an **IP address**.
- ✧ Normally, the source address is incorrect.
- ✧ Lying about the source address lets an attacker assume a new identity.
- ✧ Because the source address is not the same as the attacker's address, any replies generated by the destination will not be sent to the attacker.
- ✧ Attacker must have an alternate way to spy on traffic/predict responses.
- ✧ To maintain a connection, attacker must adhere to protocol requirements

5.5 Spoofing Attack

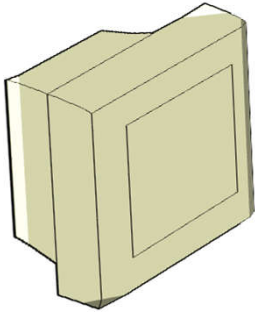
5.5.4 IP Spoofing

□ IP Spoofing

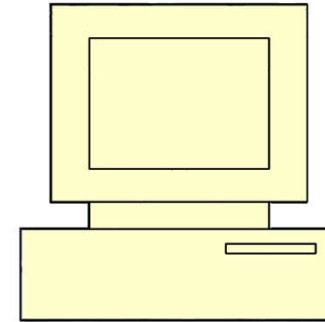
- ◆ Difficulties for attackers
 - ✧ TCP sequence numbers
 - ✧ One way communication
 - ✧ Adherence to protocols for other layers

5.5 Spoofing Attack

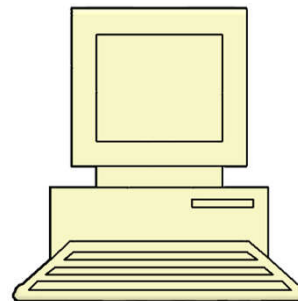
❑ IP Spoofing



Sucker - *Alice*



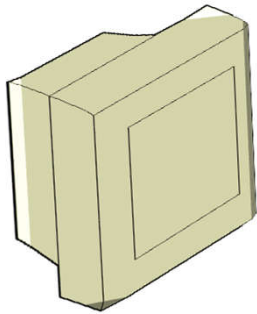
Victim - *Bob*



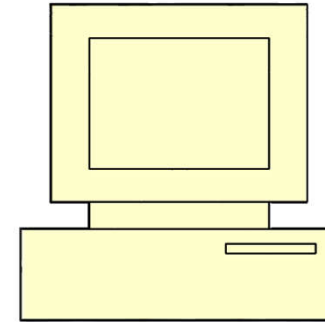
Attacker - *Eve*

5.5 Spoofing Attack

❑ IP Spoofing

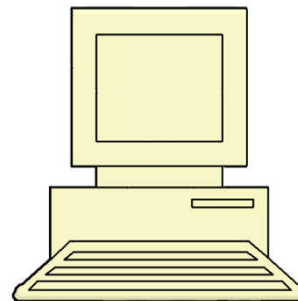


Sucker - *Alice*



Victim - *Bob*

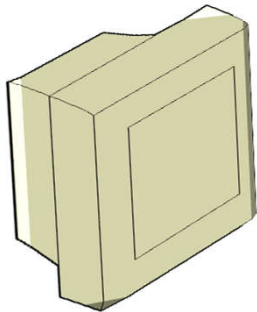
1. SYN – I'm
Bob. Let's have
a conversation



Attacker - *Eve*

5.5 Spoofing Attack

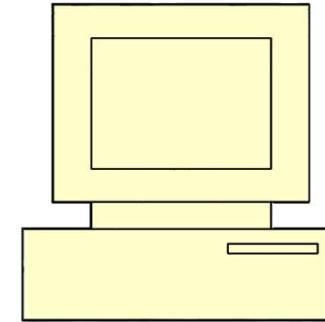
❑ IP Spoofing



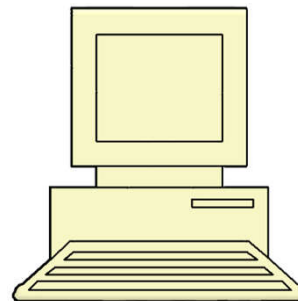
Sucker - *Alice*



2. SYN ACK – Sure,
what do you want
to talk about?



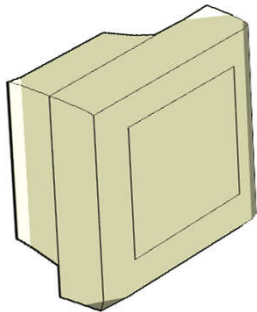
Victim - *Bob*



Attacker - *Eve*

5.5 Spoofing Attack

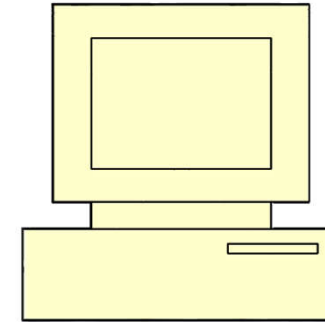
❑ IP Spoofing



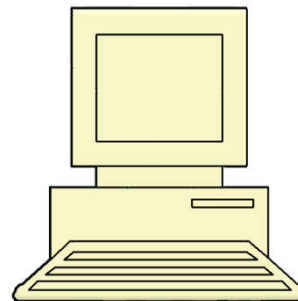
Sucker - *Alice*



3. RESET – Umm.. I
have no idea why you
are talking to me



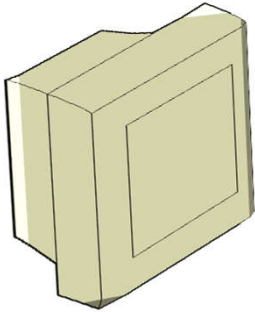
Victim - *Bob*



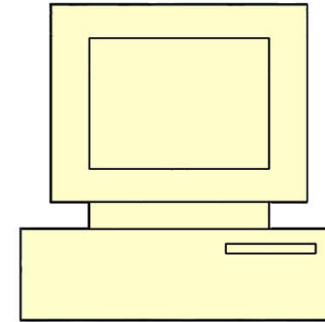
Attacker - *Eve*

5.5 Spoofing Attack

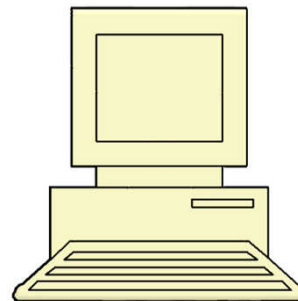
❑ IP Spoofing



Sucker - *Alice*



Victim - *Bob*



Attacker - *Eve*

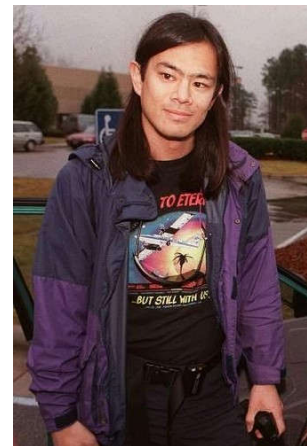
4. No connection – Guess
I need to take Bob out of
the picture...

5.5 Spoofing Attack

5.5.4 IP Spoofing

❑ IP Spoofing

- ◆ The attack – IP spoofing and abuse (濫用) of trust relationships between a diskless terminal and login server.
- ◆ *Kevin David Mitnick* hacked a diskless workstation on December 25th, 1994.
- ◆ The victim is *Tsutomu Shimomura* (下村勉).

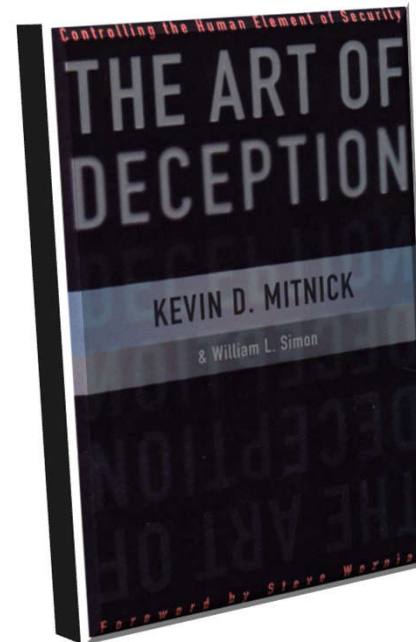
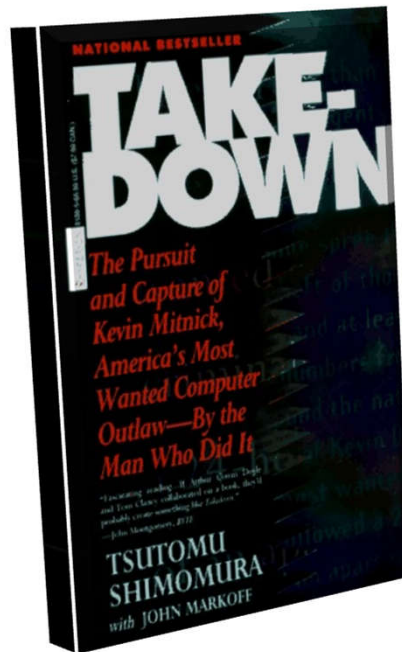


5.5 Spoofing Attack

5.5.4 IP Spoofing

□ IP Spoofing

- ◆ *Tsutomu Shimomura, John Markoff. Takedown: The Pursuit and Capture of America's Most Wanted Computer Outlaw, 1995. 抓住他：追捕美国头号计算机窃贼*
- ◆ *Kevin Mitnick. The Art of Deception, 2002. 欺骗的艺术*



5.5 Spoofing Attack

5.5.4 IP Spoofing

❑ IP Spoofing

- ♦ The film *Takedown* (2000, aka *Track Down*) is also worth watching.

Takedown (2000) [SEE RANK](#)

R 96 min - Crime | Drama | Thriller - 15 March 2000 (France)

 **Your rating:** ★★★★★★ -/10
Ratings: **6.3**/10 from 5,487 users
Reviews: 46 user | 14 critic

This film is based on the story of the capture of computer hacker "Kevin Mitnick".

Director: Joe Chappelle

Writers: Tsutomu Shimomura (book), John Markoff (book), 4 more credits »

Stars: Skeet Ulrich, Russell Wong, Angela Featherstone | See full cast and crew »

[+ Watchlist](#) [Share...](#) [Own the rights? Add a poster »](#)

[Contact the Filmmakers on IMDbPro »](#)



5.5 Spoofing Attack

5.5.4 IP Spoofing

❑ IP Spoofing

- ◆ Another film *Blackhat* (2015)

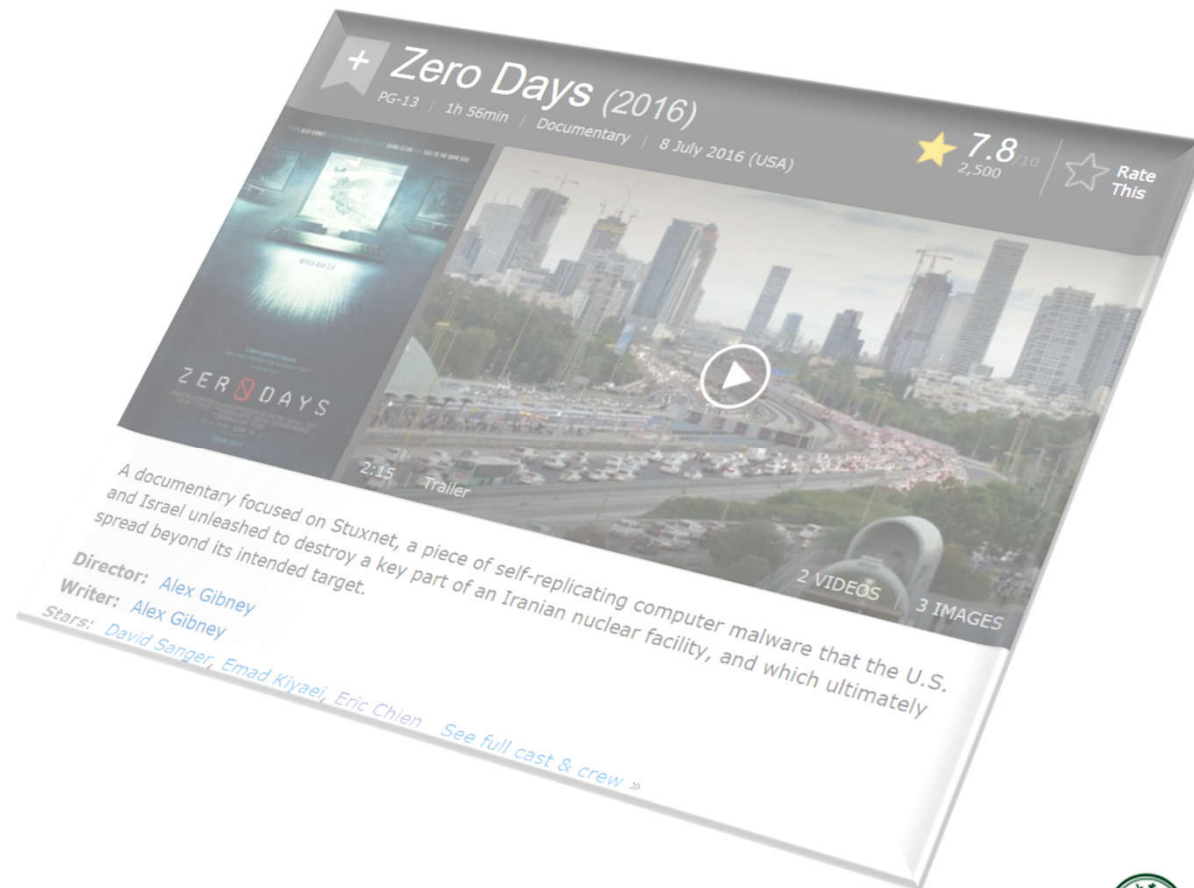


5.5 Spoofing Attack

5.5.4 IP Spoofing

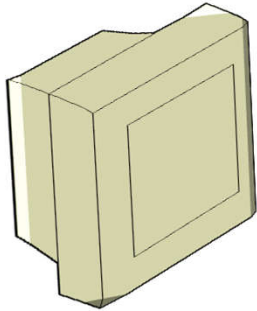
□ IP Spoofing

- ◆ *Another film *Zero Days* (2016) about *Stuxnet*

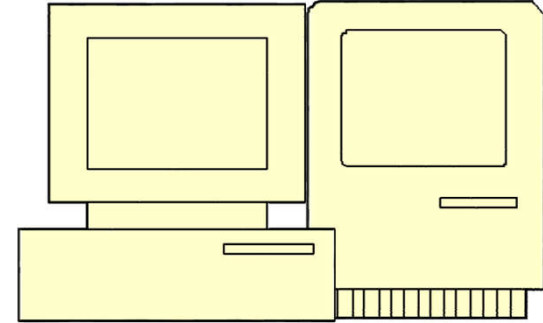


5.5 Spoofing Attack

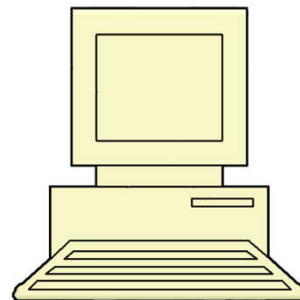
❑ IP Spoofing - *Mitnick*



Workstation



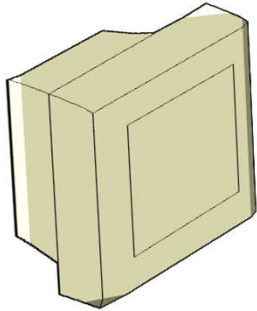
Server



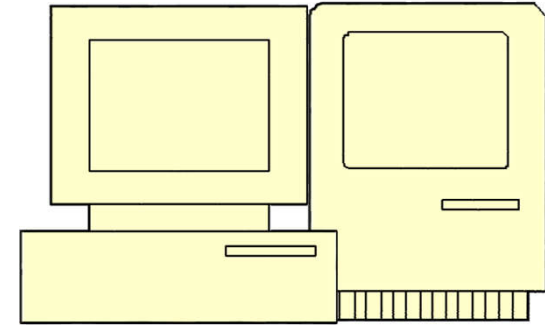
Kevin Mitnick

5.5 Spoofing Attack

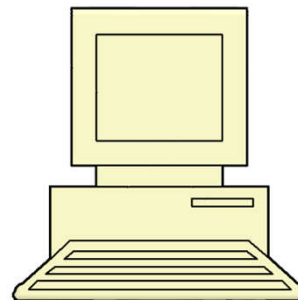
❑ IP Spoofing - *Mitnick*



Workstation



Server

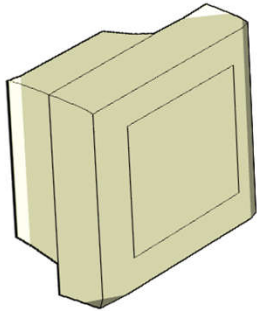


Kevin Mitnick

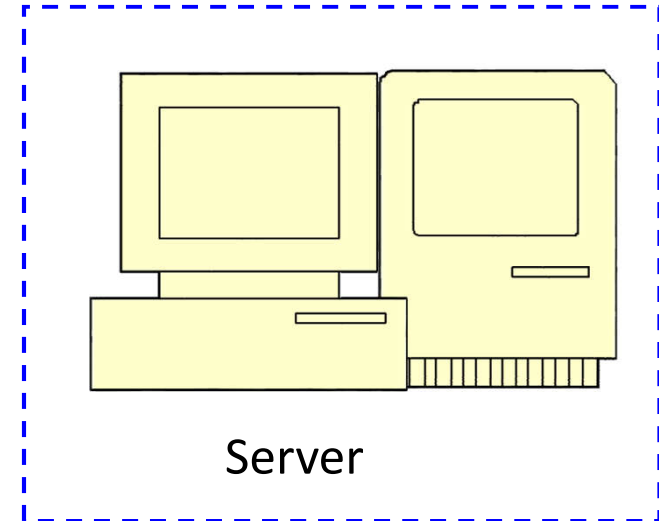
1. *Mitnick* Flood's the Server's login port so it can no longer respond.

5.5 Spoofing Attack

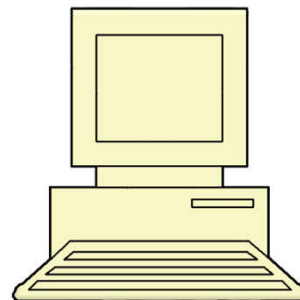
❑ IP Spoofing - *Mitnick*



Workstation



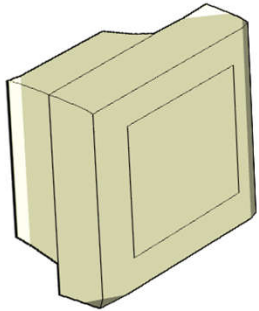
Server



Kevin Mitnick

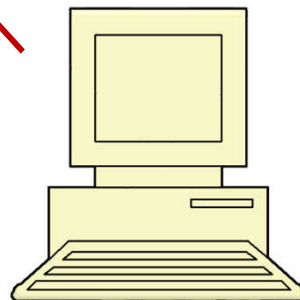
5.5 Spoofing Attack

❑ IP Spoofing - *Mitnick*

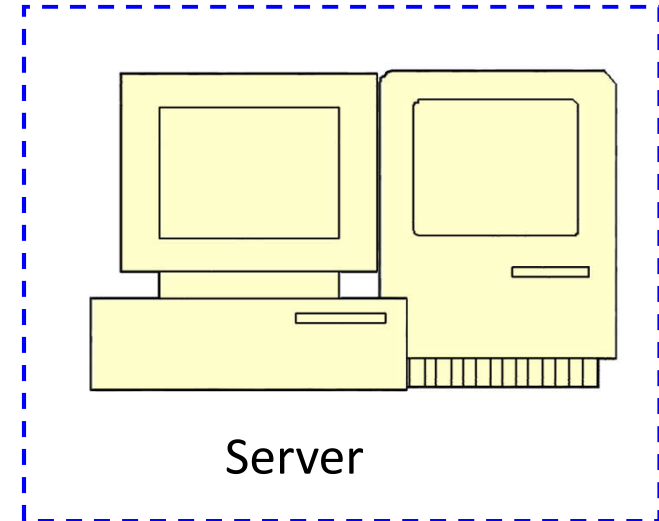


Workstation

2. *Mitnick* Probes the Workstation to determine the behavior of its TCP sequence number generator.



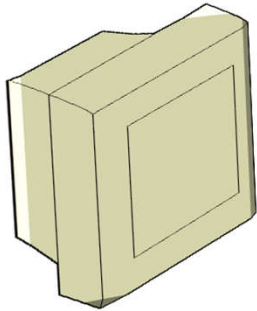
Kevin Mitnick



Server

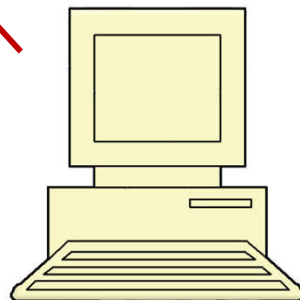
5.5 Spoofing Attack

❑ IP Spoofing - *Mitnick*

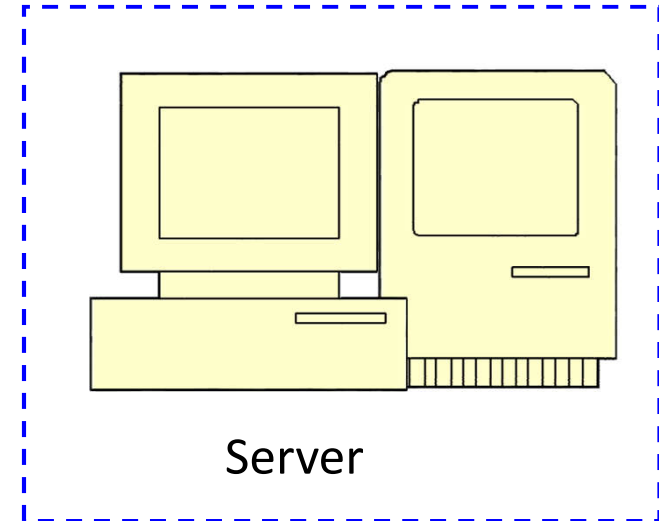


Workstation

3. *Mitnick* discovers that the TCP sequence number is incremented by 128000 each new connection.



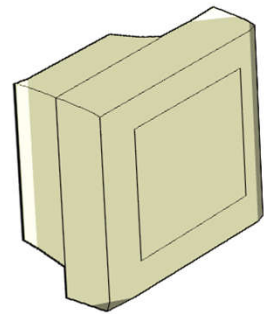
Kevin Mitnick



Server

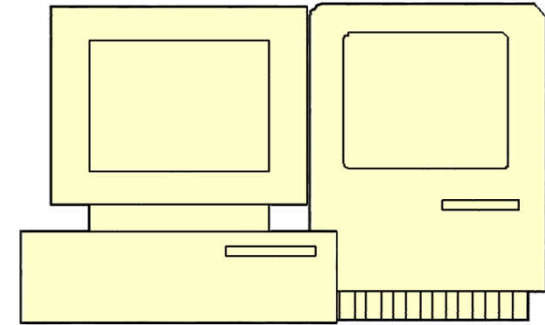
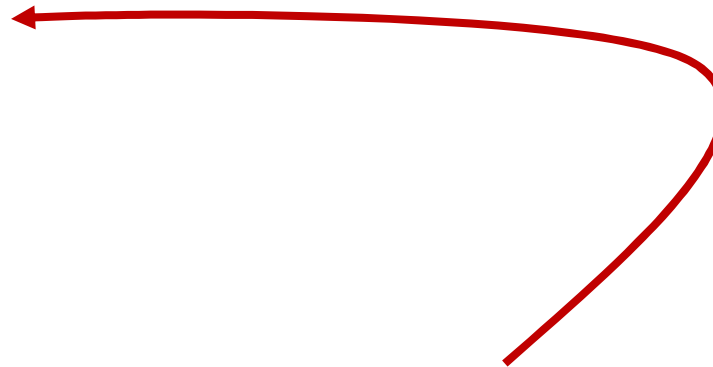
5.5 Spoofing Attack

❑ IP Spoofing - *Mitnick*

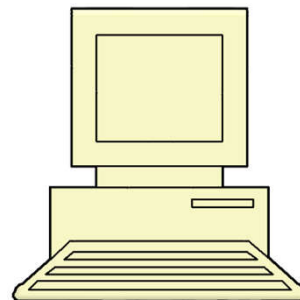


Workstation

4. *Mitnick* forges a SYN from the Server to the Workstation.



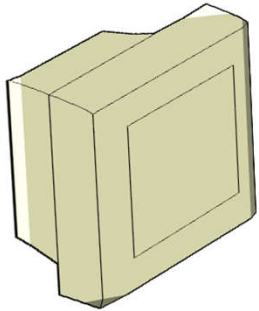
Server



Kevin Mitnick

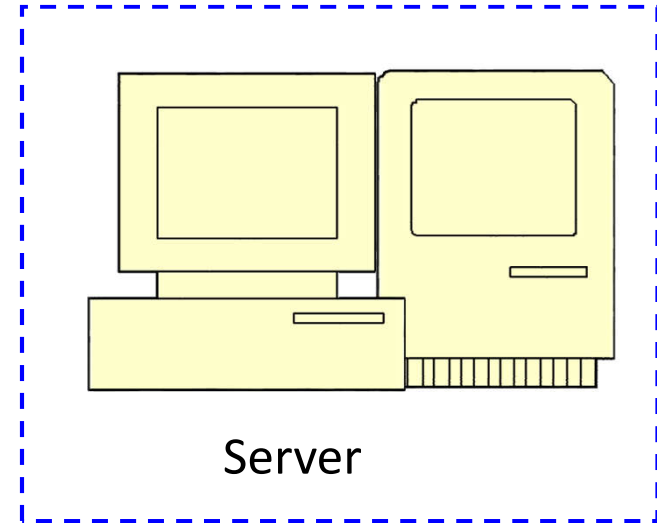
5.5 Spoofing Attack

❑ IP Spoofing - *Mitnick*

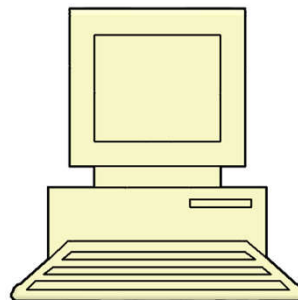


Workstation

5. The Workstation responds with an ACK, which is ignored by the flooded port of the Server (and not visible to *Mitnick*).



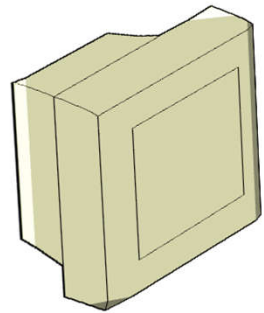
Server



Kevin Mitnick

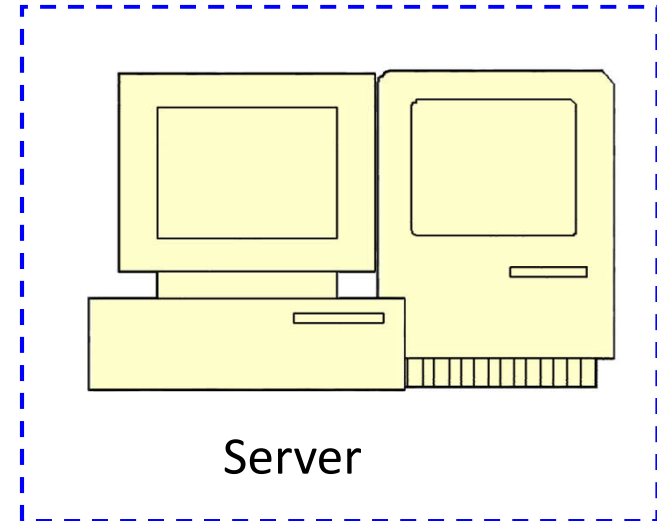
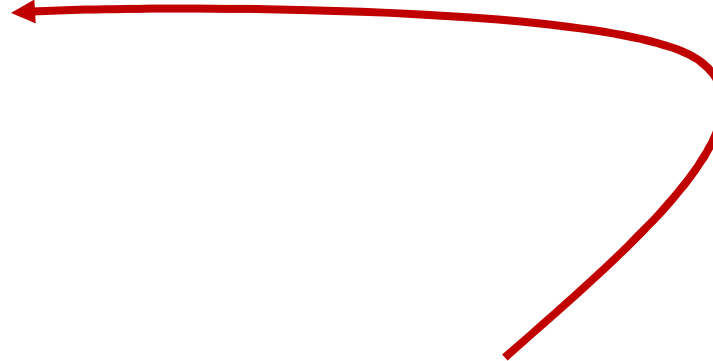
5.5 Spoofing Attack

❑ IP Spoofing - *Mitnick*

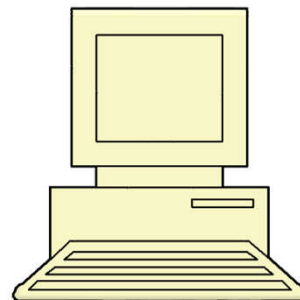


Workstation

6. *Mitnick* fakes the ACK from the Server, using the proper TCP sequence number.



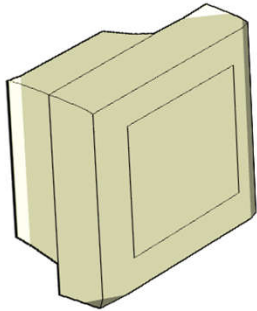
Server



Kevin Mitnick

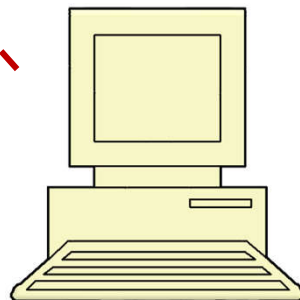
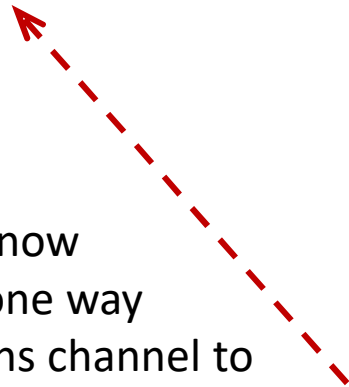
5.5 Spoofing Attack

❑ IP Spoofing - Mitnick

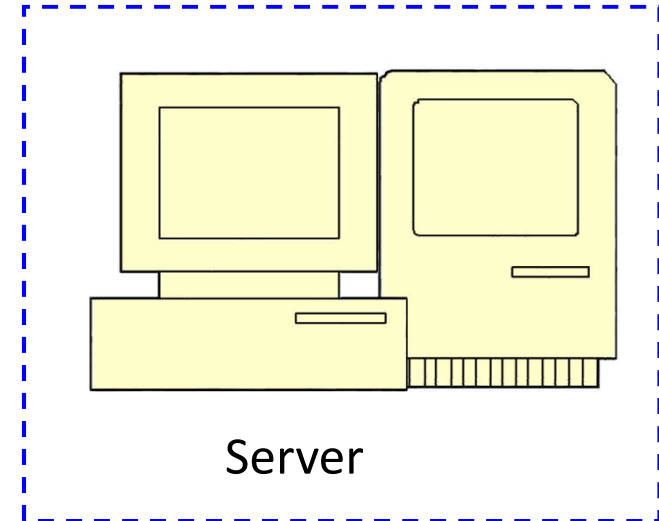


Workstation

7. *Mitnick* has now established a one way communications channel to the Workstation.



Kevin Mitnick



Server

5.5 Spoofing Attack

5.5.4 IP Spoofing

❑ IP Spoofing

- ◆ *Mitnick attack*

- ✧ *Mitnick* abused the trust relationship between the Server and the Workstation
- ✧ He flooded the Server to prevent communication between the Server and the Workstation
- ✧ Used math skills to determine the TCP sequence number algorithm (i.e., add 128000)
- ✧ This allowed *Mitnick* to open a connection without seeing the Workstations outgoing sequence numbers and without the Server interrupting his attack
- ✧ Refer to the books in detail.

5.5 Spoofing Attack

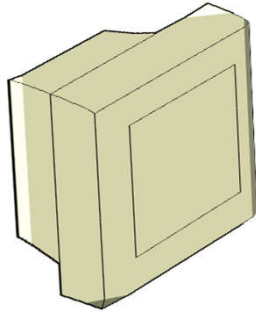
5.5.4 IP Spoofing

❑ IP Spoofing

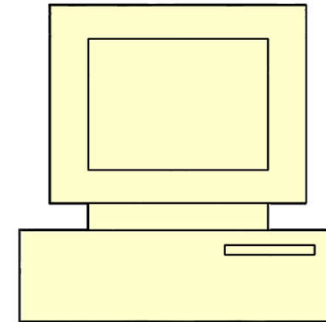
- ◆ Session Hijack
 - ✧ IP spoofing used to eavesdrop/take control of a session.
 - ✧ Attacker normally within a LAN/on the communication path between server and client.
 - ✧ Not blind, since the attacker can see traffic from both server and client.

5.5 Spoofing Attack

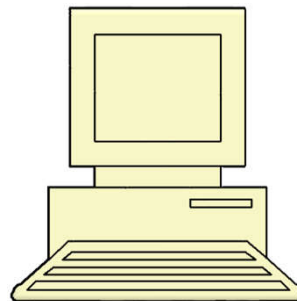
❑ IP Spoofing - Session Hijack



Alice



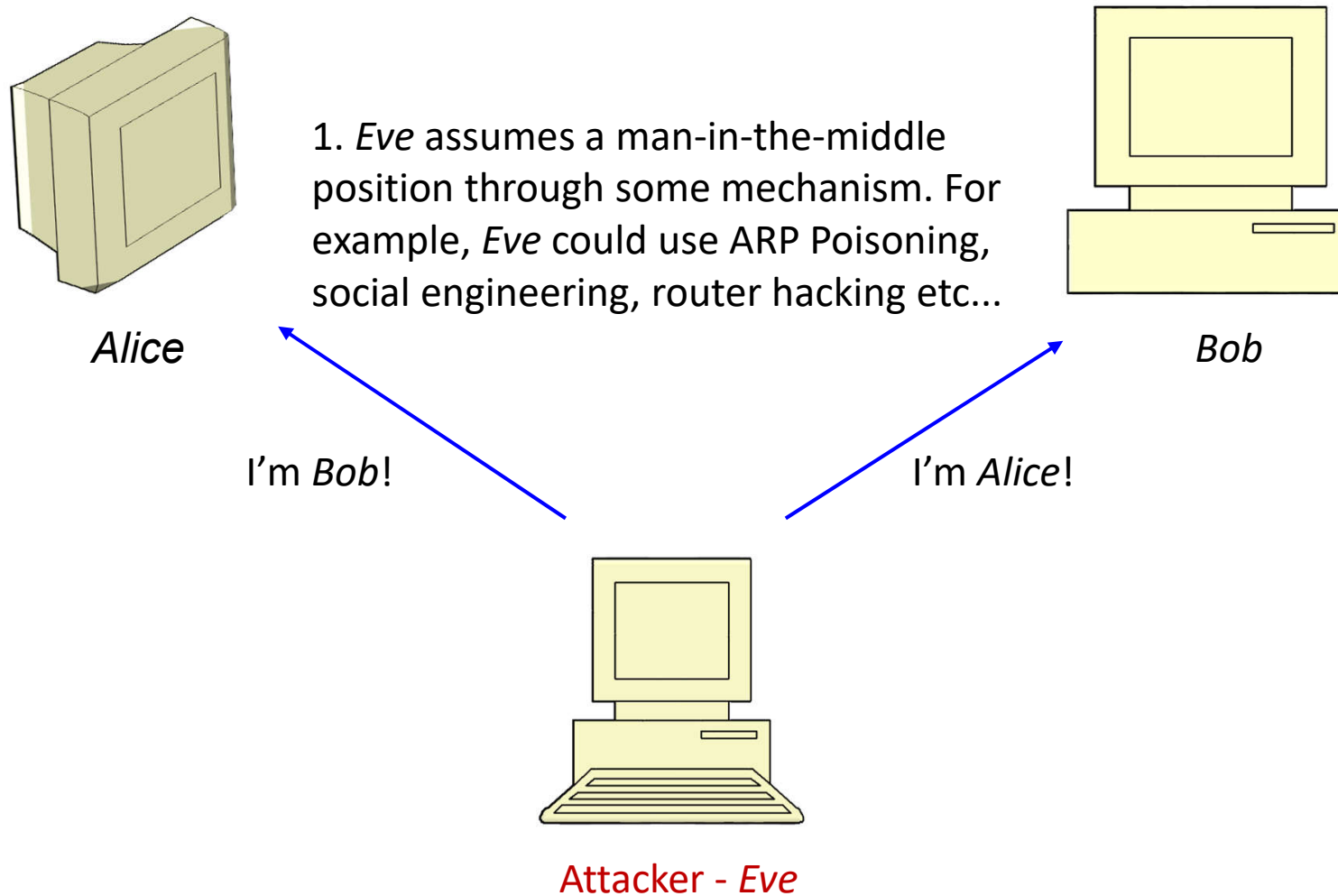
Bob



Attacker - Eve

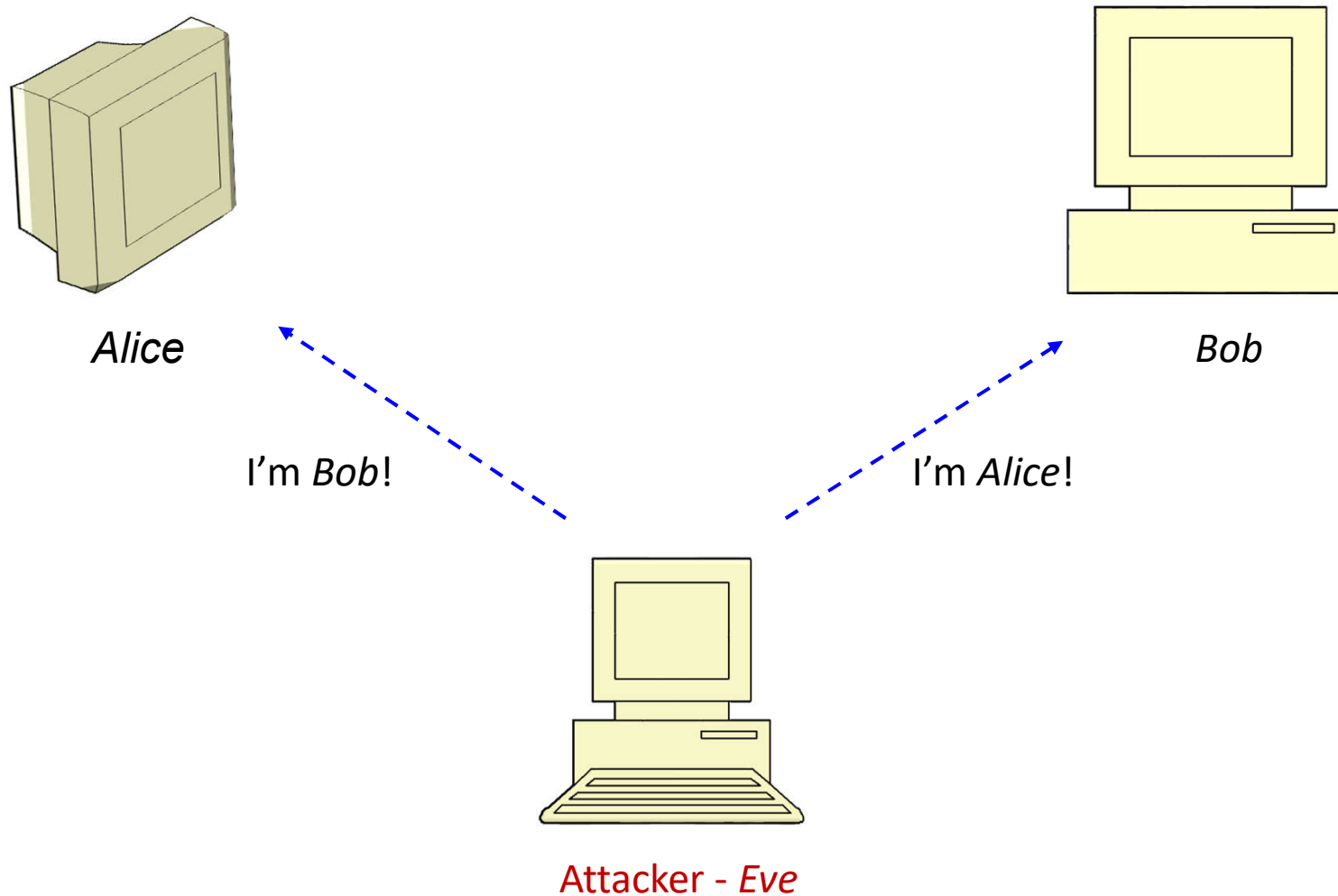
5.5 Spoofing Attack

❑ IP Spoofing - Session Hijack



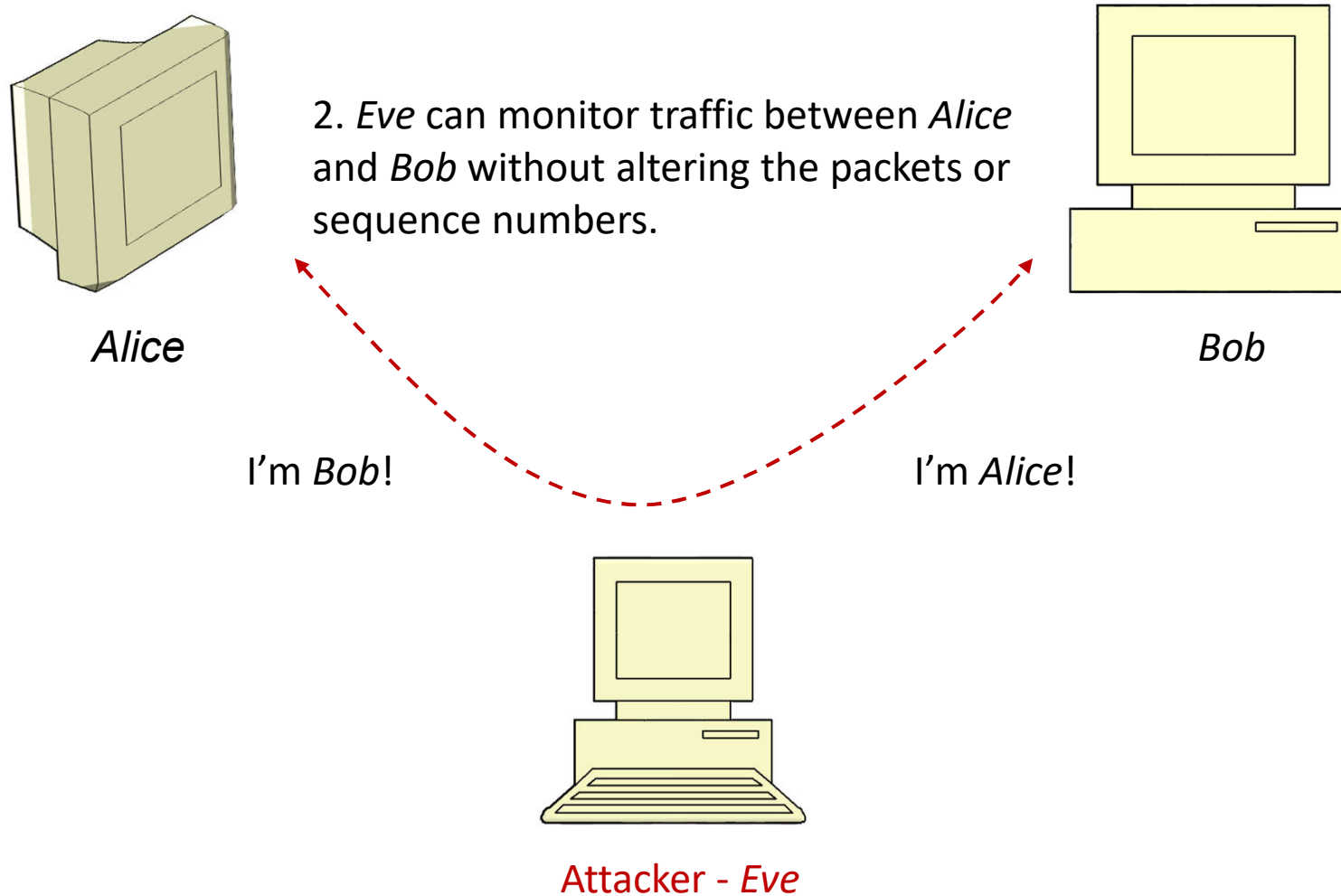
5.5 Spoofing Attack

❑ IP Spoofing - Session Hijack



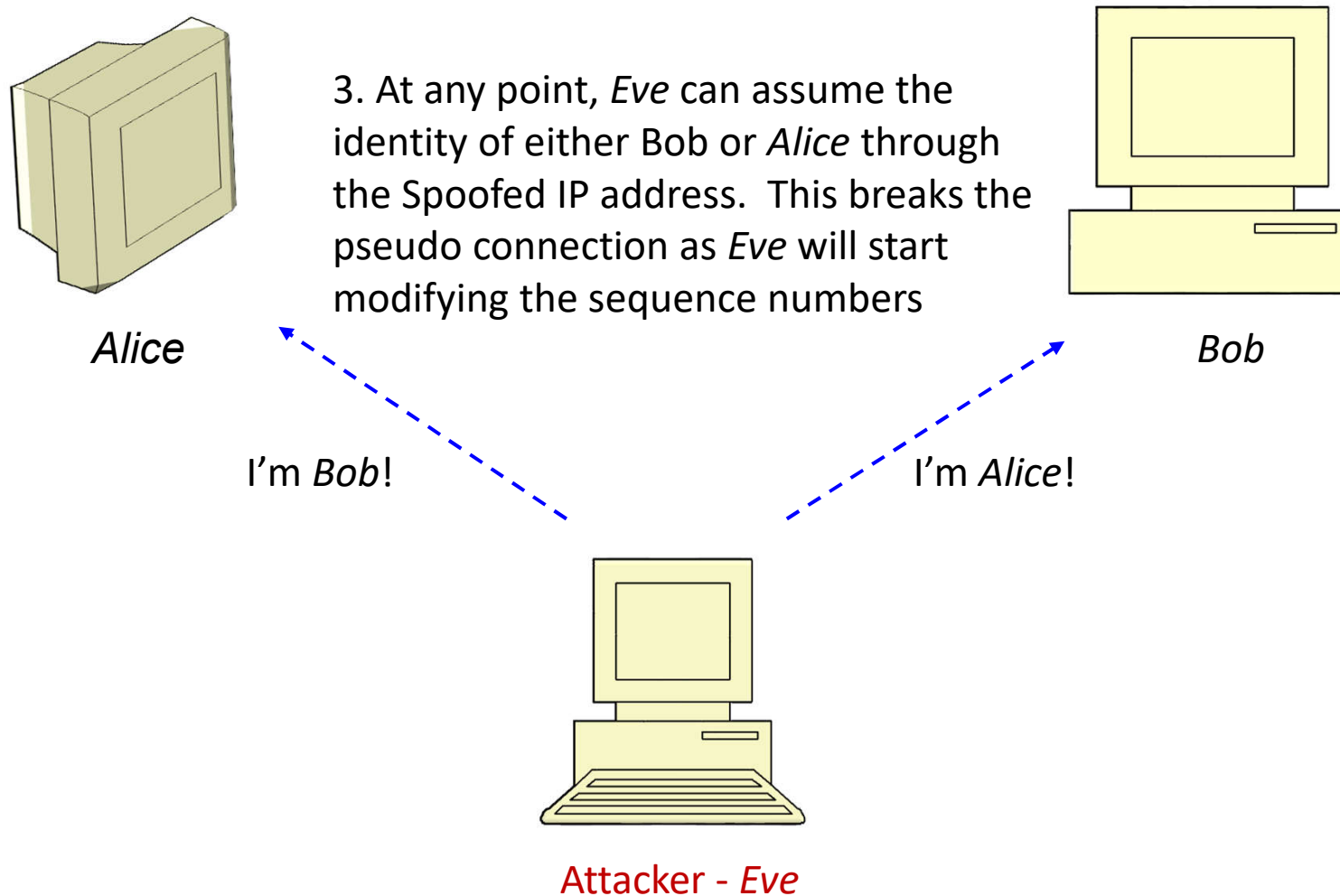
5.5 Spoofing Attack

❑ IP Spoofing - Session Hijack



5.5 Spoofing Attack

❑ IP Spoofing - Session Hijack



5.5 Spoofing Attack

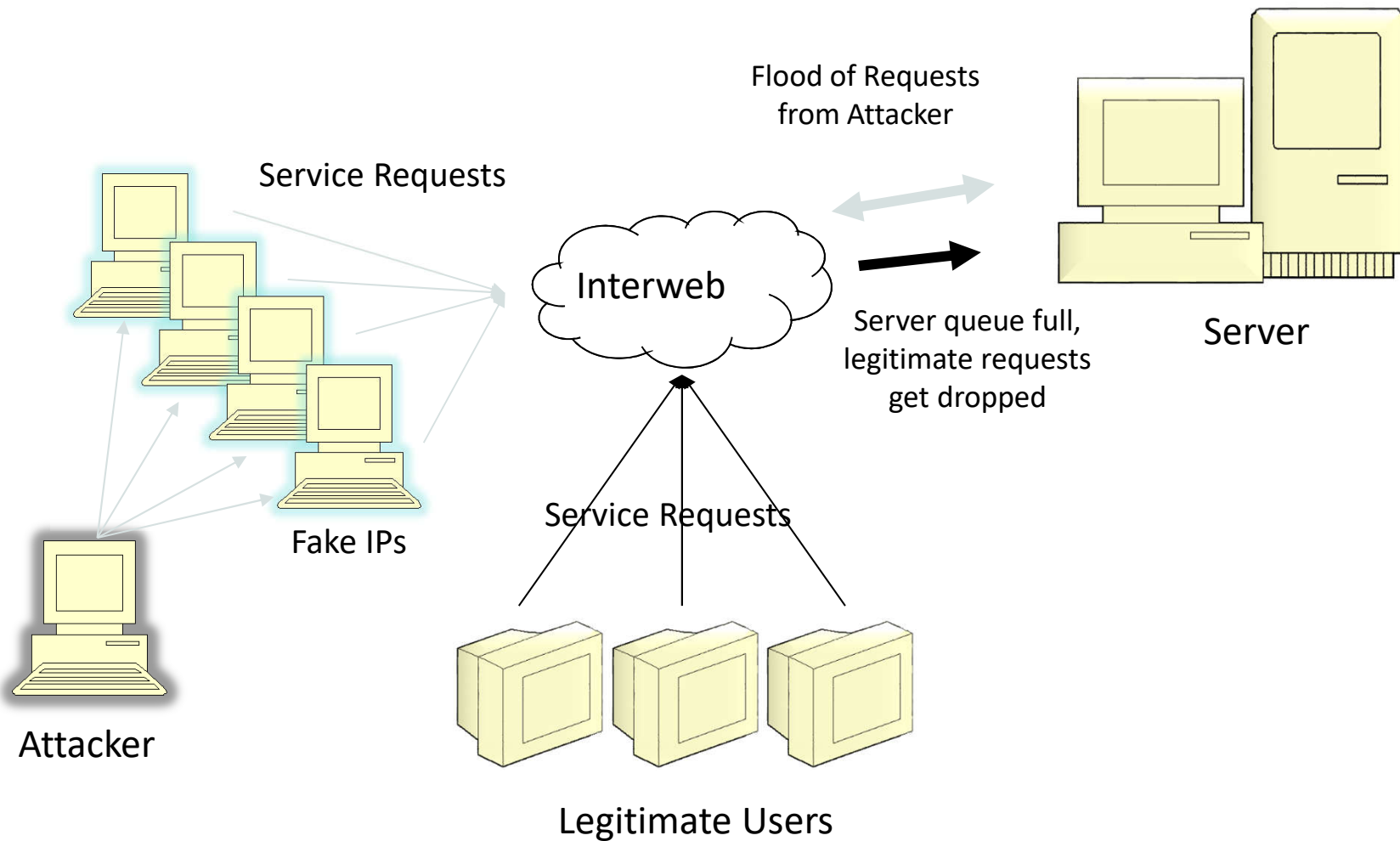
5.5.4 IP Spoofing

□ IP Spoofing

- ◆ DoS / DDoS
 - ✧ Denial of Service (DoS) and Distributed Denial of Service (DDoS) are attacks aimed at preventing clients from accessing a service.
 - ✧ IP Spoofing can be used to create DoS attacks
 - The attacker spoofs a large number of requests from various IP addresses to fill a services queue.
 - With the services queue filled, legitimate user's cannot use the service.

5.5 Spoofing Attack

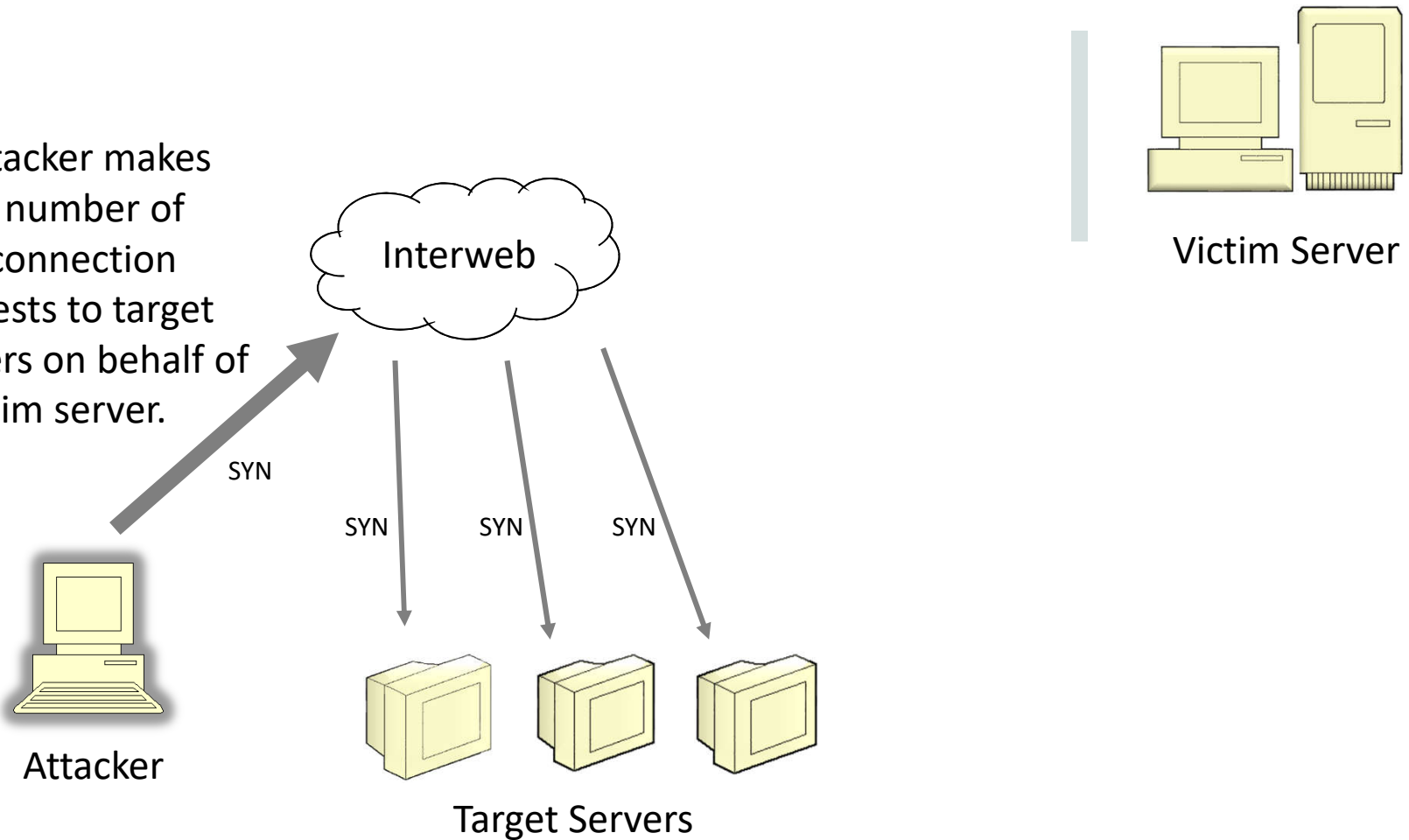
❑ IP Spoofing – DoS / DDoS



5.5 Spoofing Attack

❑ IP Spoofing – DoS

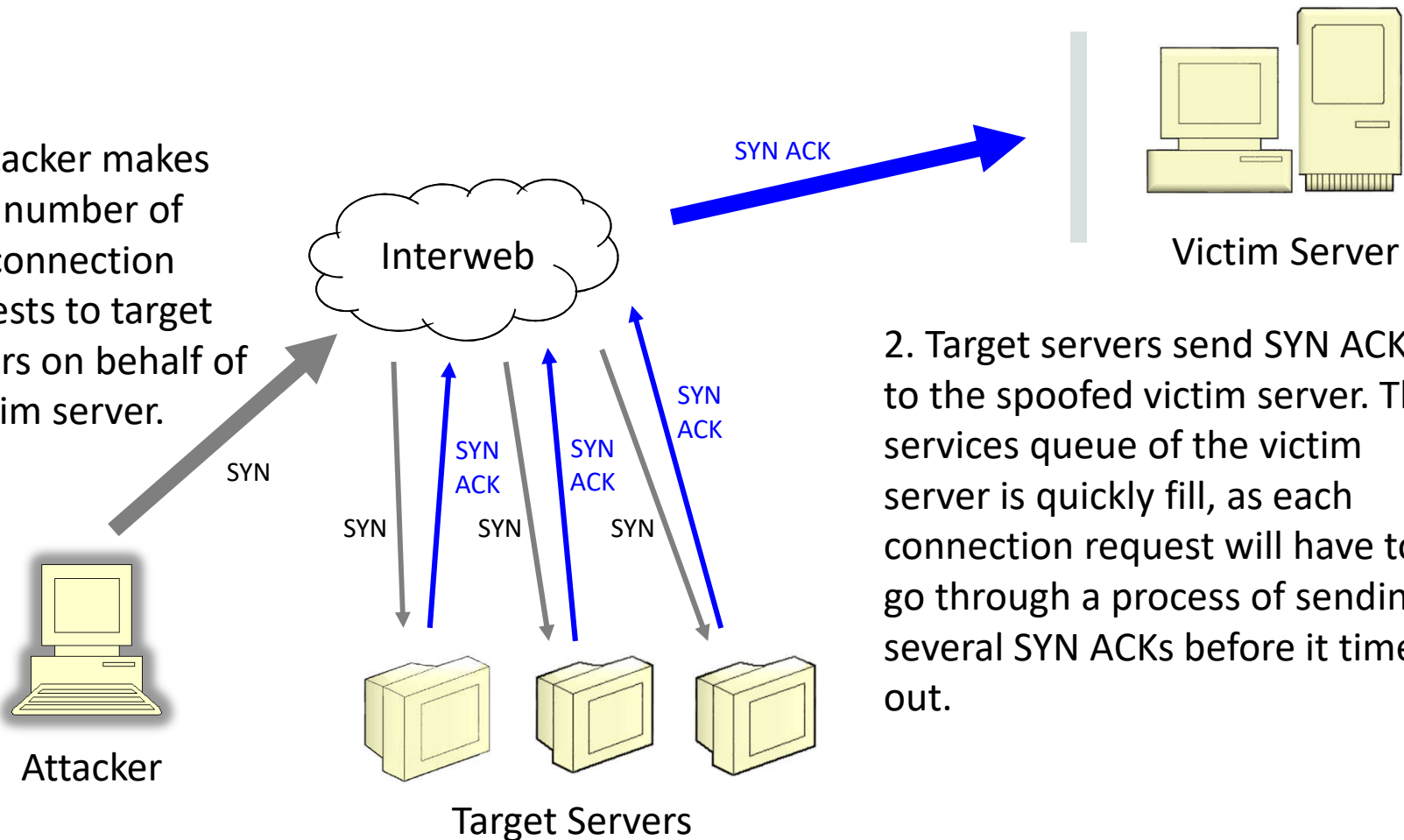
1. Attacker makes large number of SYN connection requests to target servers on behalf of a victim server.



5.5 Spoofing Attack

❑ IP Spoofing – DoS

1. Attacker makes large number of SYN connection requests to target servers on behalf of a victim server.

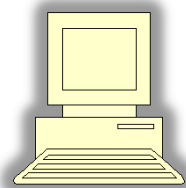


2. Target servers send SYN ACK to the spoofed victim server. The services queue of the victim server is quickly fill, as each connection request will have to go through a process of sending several SYN ACKs before it times out.

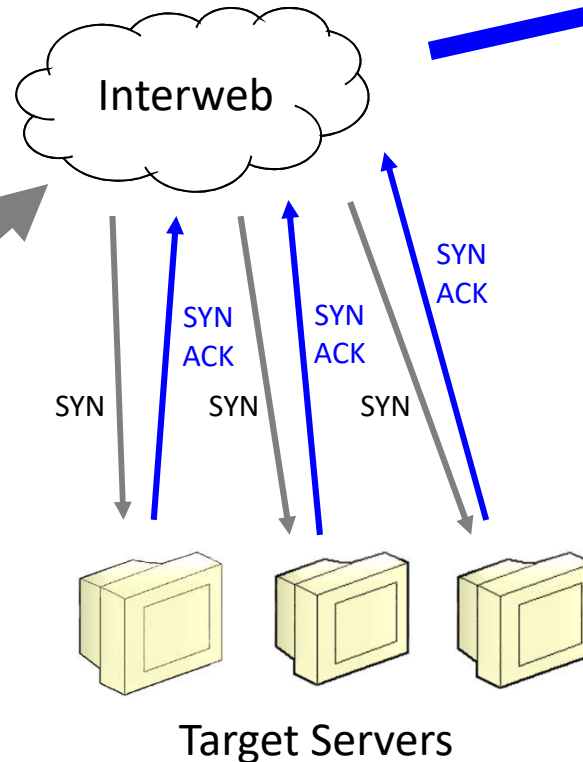
5.5 Spoofing Attack

❑ IP Spoofing – DoS

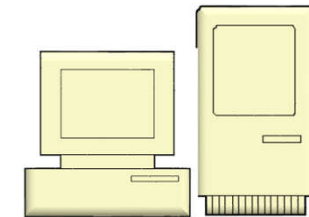
1. Attacker makes large number of SYN connection requests to target servers on behalf of a victim server.



Attacker



Queue Full (already DoS'd)

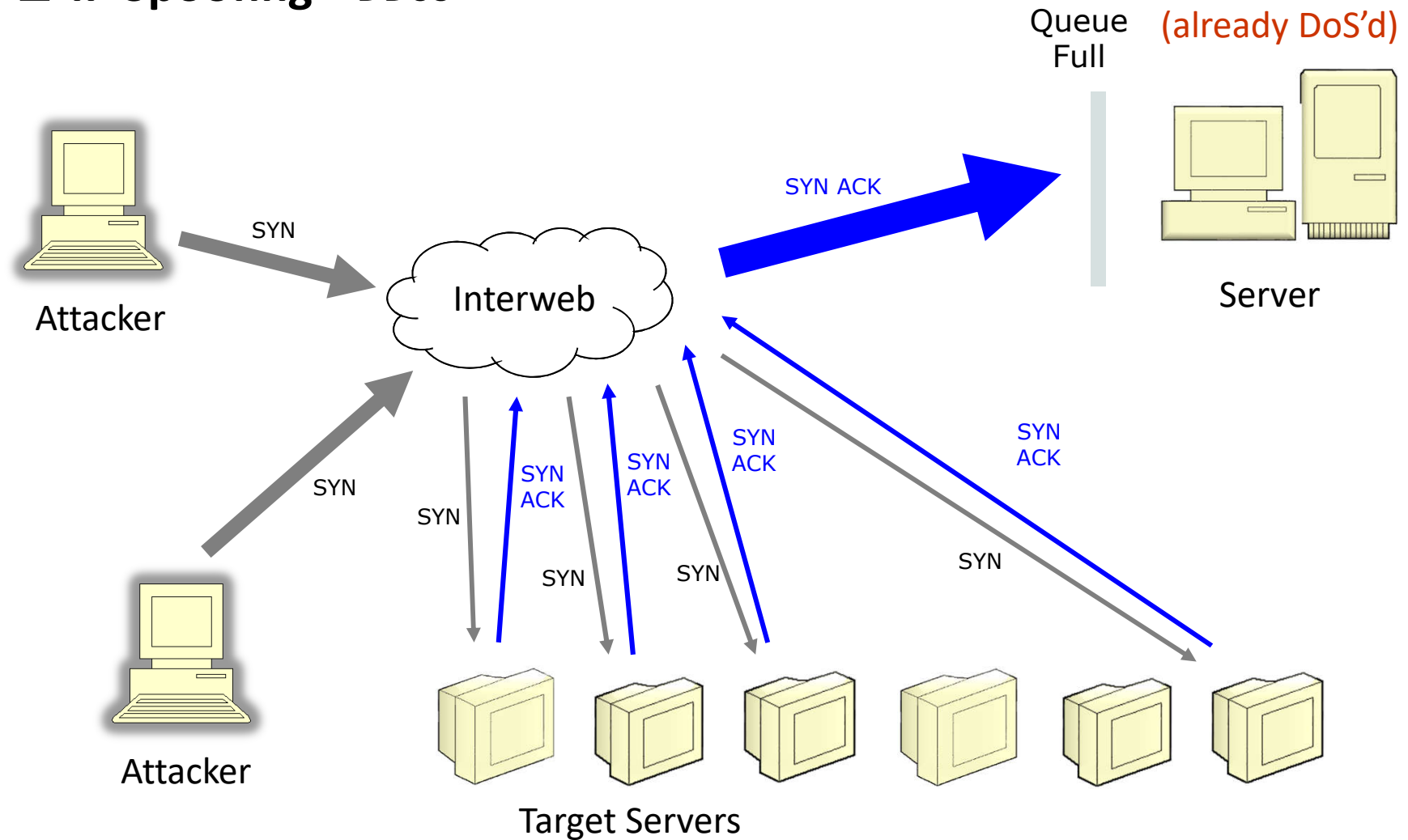


Victim Server

2. Target servers send SYN ACK to the spoofed victim server. The services queue of the victim server is quickly fill, as each connection request will have to go through a process of sending several SYN ACKs before it times out.

5.5 Spoofing Attack

❑ IP Spoofing – DDoS



5.5 Spoofing Attack

5.5.4 IP Spoofing

□ IP Spoofing

- ◆ DoS/DDoS
 - ✧ Many other types of DDoS are possible.
 - ✧ DoS becomes more dangerous if spread to multiple computers.

5.5 Spoofing Attack

5.5.4 IP Spoofing

❑ Defending against the IP Spoofing Threat

- ◆ IP spoofing can be defended against in some ways:
 - ✧ Other protocols in the Architectural model may reveal spoofing.
 - TCP sequence numbers are often used in this manner.
 - New generators for sequence numbers are a lot more complicated than "add 128000".
 - Makes it difficult to guess proper sequence numbers if the attacker is blind.
 - ✧ "Smart" routers can detect IP addresses that are outside its domain.
 - ✧ "Smart" servers can block IP ranges that appear to be conducting a DoS.

5.5 Spoofing Attack

5.5.4 IP Spoofing

❑ Continuous Evolution

- ♦ IP spoofing is still possible today, but has to evolve in the face of growing security.
- ♦ New issue of Phrack (www.phrack.org) includes a method of using IP spoofing to perform remote scans and determine TCP sequence numbers.
- ♦ This allows a session hijack attack even if the attacker is blind.

References

1. *E. W. Felten, D. Balfanz, D. Dean and D. Wallach*. Web Spoofing: An Internet Con Game. Technical Report 54-96 (revised). Department of Computer Science, Princeton University, 1997.
2. *Pierre-Alain Fayolle, Vincent Glaume*. A Buffer Overflow Study Attacks & Defenses, ENSEIRB, 2002.
3. *William E. Byrd*. Stack Smashing Defense: A Buffer Overflow Lab Exercise, Feb 14, 2004.
4. *James F. Kurose, Keith W. Ross*, Computer Networking: A Top-Down Approach Featuring the Internet
5. *Larry L. Peterson, Bruce S. Davie*, Computer Networks: A Systems Approach, 4th Edition (The Morgan Kaufmann Series in Networking).
6. *Behrouz Forouzan*, Cryptography & Network Security. McGraw-Hill
7. *Jon Erickson*, Hacking: The Art of Exploitation.
8. *Chris Sanders*, Understanding Man-in-the-Middle Attacks, <http://www.windowsecurity.com>



References

9. <http://www.millersmiles.co.uk/archives.php>
10. <http://www.antiphishing.org/phishReportsArchive.html>
11. <http://www.securityfocus.com/infocus/1674>
12. <http://www.gulker.com/ra/hack/tsattack.html>
13. <http://www.phrack.org/>
14. <http://technet.microsoft.com/en-us/library/cc959354.aspx#mainSection>
15. http://www.softpanorama.org/DNS/Security/dns_spoofing.shtml
21. http://en.wikipedia.org/wiki/Buffer_overflow
22. <http://www.princeton.edu/~unix/Solaris/troubleshoot/vm.html>
23. <http://www.ibm.com/developerworks/cn/linux/l-overflow/>
24. <http://www.ibm.com/developerworks/cn/linux/l-cn-gccstack/>



End of Chapter 5

