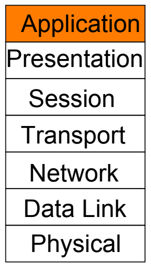
CSC458

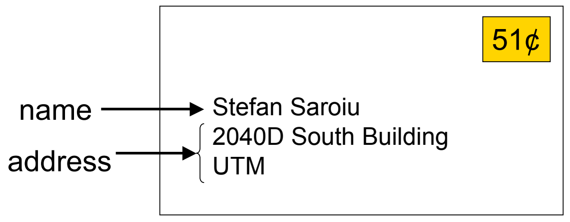
Naming and the DNS
Network Security

This Lecture

- Naming
- Focus
  - How do we name hosts etc.?
- Topics
  - Domain Name System (DNS)
  - Email/URLs



Names and Addresses



- Names are identifiers for objects/services (high level)
- Addresses are locators for objects/services (low level)
- Binding is the process of associating a name with an address
- Resolution is the process of looking up an address given a name
- But, addresses are really lower-level names; many levels used

Naming in Systems

- Ubiquitous
  - Files in filesystem, processes in OS, pages on the web, …
- Decouple identifier for object/service from location
  - Hostnames provide a level of indirection for IP addresses
- Naming greatly impacts system capabilities and performance
  - Ethernet addresses are a flat 48 bits
    - flat → any address anywhere but large forwarding tables
  - IP addresses are hierarchical 32/128 bits
    - hierarchy → smaller routing tables but constrained locations

Internet Hostnames

- Hostnames are human-readable identifiers for end-systems based on an administrative hierarchy
  - cleo.slup.cs.toronto.edu is my desktop machine
- IP addresses are a fixed-length binary encoding for end-systems based on their position in the network
  - 192.12.174.140 is cleo's IP address

- Original name resolution: HOSTS.TXT
- Current name resolution: Domain Name System
- Future name resolution: ?

Original Hostname System

- When the Internet was really young …

- Flat namespace
  - Simple (host, address) pairs

- Centralized management
  - Updates via a single master file called HOSTS.TXT
  - Manually coordinated by the Network Information Center (NIC)

- Resolution process
  - Look up hostname in the HOSTS.TXT file

Scaling Problems

- Coordination
  - Between all users to avoid conflicts

- Inconsistencies
  - Between update and distribution of new version

- Reliability
  - Single point of failure

- Performance
  - Competition for centralized resources

Domain Name System (DNS)

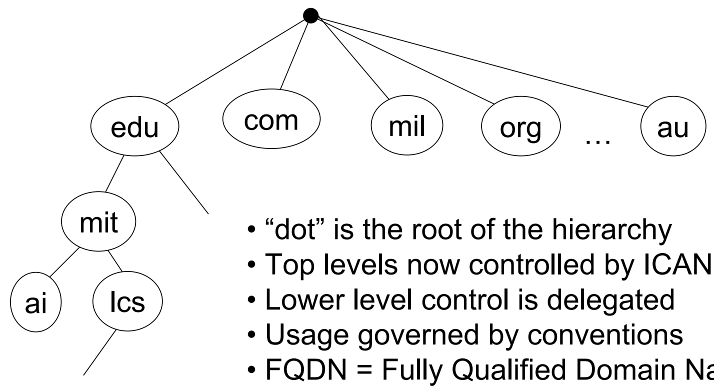
- Designed by Mockapetris and Dunlap in the mid 80s

- Namespace is hierarchical
  - Allows much better scaling of data structures
  - e.g., cleo.slup.cs.toronto.edu

- Namespace is distributed
  - Decentralized administration and access
  - e.g., *.toronto.edu managed by CSC

- Resolution is by query/response
  - With replicated servers for redundancy
  - With heavy use of caching for performance

DNS Hierarchy



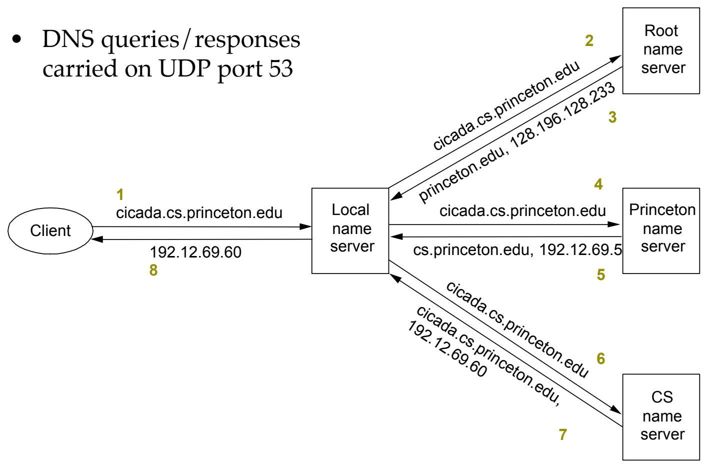
- "dot" is the root of the hierarchy
- Top levels now controlled by ICANN
- Lower level control is delegated
- Usage governed by conventions
- FQDN = Fully Qualified Domain Name

DNS Distribution

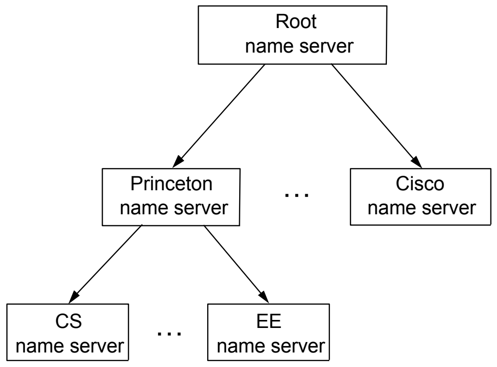
- Data managed by zones that contain resource records
  - Zone is a complete description of a portion of the namespace
  - e.g., all hosts and addresses for machines in toronto.edu with pointers to subdomains like cs.toronto.edu
- One or more nameservers manage each zone
  - Zone transfers performed between nameservers for consistency
  - Multiple nameservers provide redundancy
- Client resolvers query nameservers for specified records
  - Multiple messages may be exchanged per DNS lookup to navigate the name hierarchy (coming soon)

DNS Lookups/Resolution

- DNS queries/responses carried on UDP port 53



Hierarchy of Nameservers



Caching

- Servers and clients cache results of DNS lookups
  - Cache partial results too (e.g., server for princeton.edu)
  - Greatly improves system performance; lookups the rare case
- Cache using time-to-live (TTL) value from provider
  - higher TTL means less traffic, lower TTL means less stale info
- Negative caching is used too!
  - errors can cause repeated queries for non-existent data

DNS Bootstrapping

- Need to know IP addresses of root servers before we can make any queries
- Addresses for 13 root servers ([a-m].root-servers.net) handled via initial configuration (named.ca file)

Building on the DNS

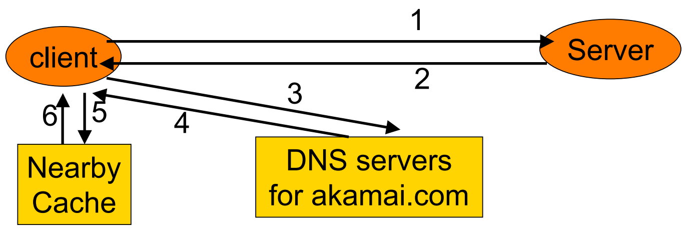
- Other naming designs leverage the DNS
- Email:
  - e.g., stefan@cs.toronto.edu is stefan in the domain cs.toronto.edu
- Uniform Resource Locators (URLs) name for Web pages
  - e.g., www.cs.toronto.edu/~stefan
  - Use domain name to identify a Web server
  - Use "/" separated string to name path to page (like files)

Future Evolution of the DNS

- Design constrains us in two major ways that are increasingly less appropriate
- Static host to IP mapping
  - What about mobility (Mobile IP) and dynamic address assignment (DHCP)
- Location-insensitive queries
  - What if I don't care what server a Web page comes from, as long as it's the right page?
  - e.g., a yahoo page might be replicated

Akamai

- Use the DNS to effect selection of a nearby Web cache



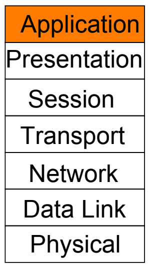
- Leverage separation of static/dynamic content
- Beware DNS caching

Key Concepts

- The design of names, addresses and resolution has a significant impact on system capabilities
- Hierarchy, decentralization and caching allow the DNS to scale
  - These are general techniques!

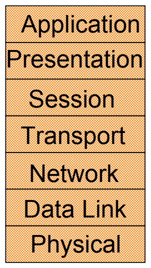
Last Time

- Naming
- Focus
  - How do we name hosts etc.?
- Topics
  - Domain Name System (DNS)



This Time

- Network security
- Focus
  - How do we secure distributed systems?
- Topics
  - Privacy, integrity, authenticity
  - Cryptography
  - Practical security



What do we mean by "Security"?

- Networks are fundamentally shared
  - Need means to protect messages sent by legitimate participants from others with access to the network

- Privacy: messages can't be eavesdropped
- Integrity: messages can't be tampered with
- Authenticity: messages were sent by the right party

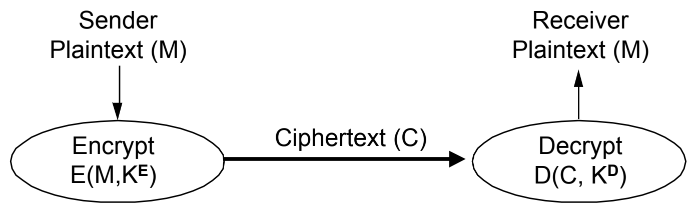
- These are in addition to the need to protect networked systems from intrusions and compromise by attackers

Approaches at 10,000 ft

- Physical security
  - Tackle the problem of sharing directly
- "Security through obscurity"
  - Hope no-one will find out what you're doing!
- Throw math at the problem
  - Cryptography

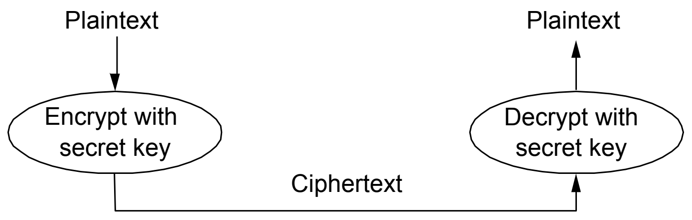
- Why is security difficult?
  - It's a negative goal: can you be sure there are no flaws?
  - Often assumptions turn out to be invalid, esp. randomness

Basic Encryption for Privacy



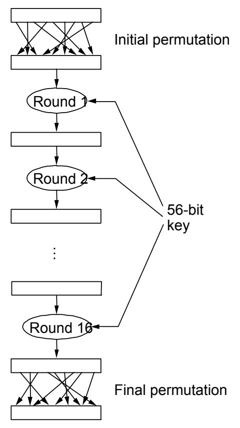
- Cryptographer chooses functions E, D and keys K^E, K^D
  - Mathematical basis
- Cryptanalyst try to "break" the system
  - Depends on what is known: E and D, M and C?

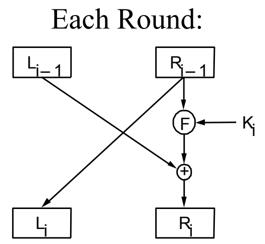
Secret Key Functions (DES, IDEA)



- Single key (symmetric) is shared between parties
  - Often chosen randomly, but must be communicated

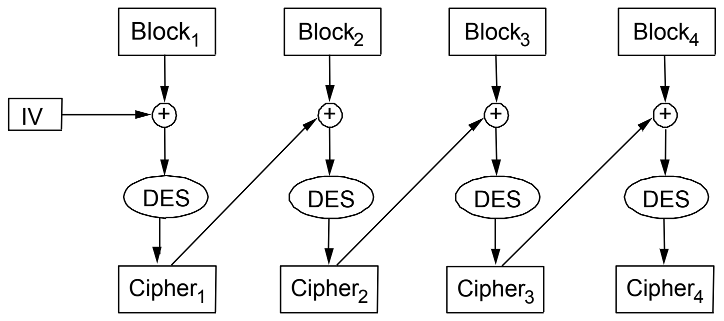
Basics of DES





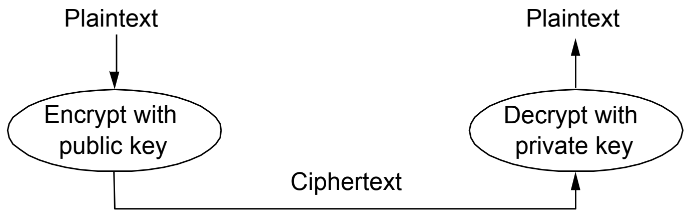
DES uses a 64 bit key (56 + 8)
Message encrypted 64 bits at a time
16 rounds in the encryption
Each round scrambles 64 bits

DES (cont.)



- Repeat process for larger messages with "chaining"

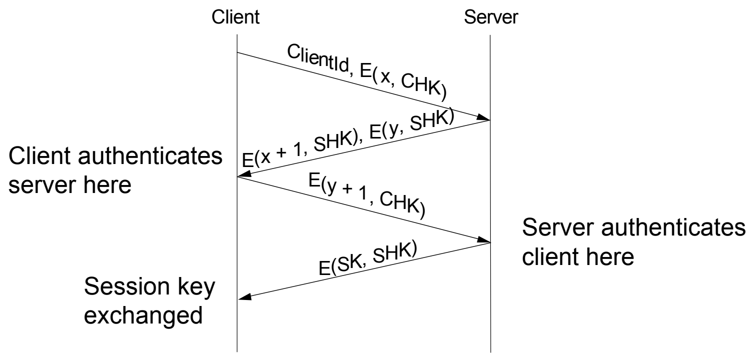
Public Key Functions (RSA)



- Public and private key related mathematically
  - Public key can be published; private is a secret

Authentication Protocols

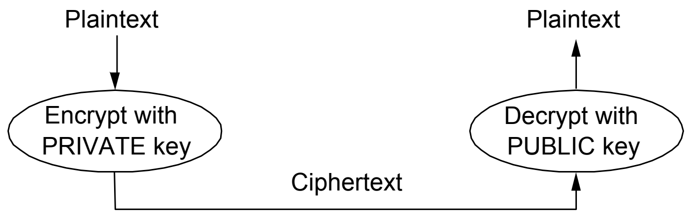
- Three-way handshake for mutual authentication
  - Client and server share secrets, e.g., login password



Authenticity and Integrity

- Sometimes we care about knowing messages authentic, but don't care about privacy.
- If only sender and receiver knew the keys we would be done … but that's often not the case
  - A pair of keys for each pair of communicating parties?
- In public key (RSA) systems the "encryption" key is potentially known by everyone
  - anyone could have sent us a confidential message by encrypting with our public key

RSA Digital Signature



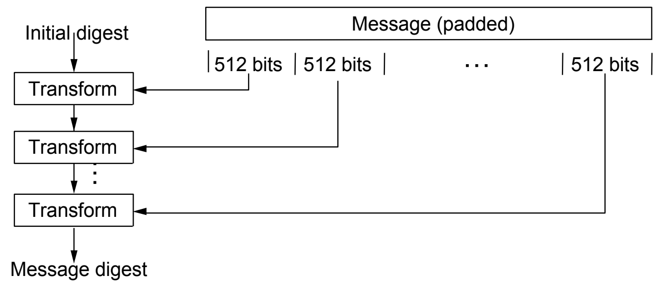
- Notice that we reversed the role of the keys (and the math just works out) so only one party can send the message but anyone can check it's authenticity

A Faster "RSA Signature"

- Encryption can be expensive, e.g., RSA 1Kbps
- To speed up, let's sign just the checksum instead!
  - Check that the encrypted bit is a signature of the checksum
- Problem: Easy to alter data without altering checksum
- Answer: Cryptographically strong "checksums" called message digests where it's computationally difficult to choose data with a given checksum
  - But they still run much more quickly than encryption
  - MD5 (128 bits) is the most common example

Message Digests (MD5, SHA)

- Act as a cryptographic checksum or hash
  - Typically small compared to message (MD5 128 bits)
  - "One-way": infeasible to find two messages with same digest



Cryptography in Protocols

- These techniques can be applied at different levels:
  - IP packets (IPSEC)
  - Web transfers or other transports (SSL /TLS, Secure HTTP)
  - Email (PGP)

Practical issues

- In practice, systems are not that secure
  - hackers can go after weakest link
    - any system with bugs is vulnerable
  - vulnerability often not anticipated
    - usually not a brute force attack against encryption system
  - often can't tell if system is compromised
    - hackers hide their tracks
  - can be hard to re-secure system after breakin
    - hackers can leave hard-to-detect backdoors

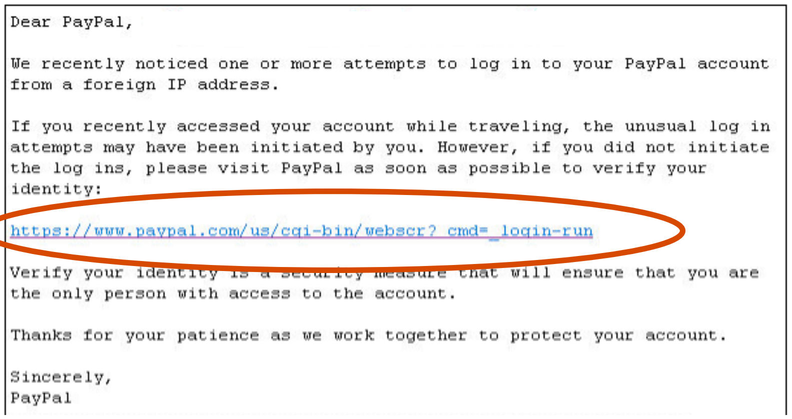
Password dictionary attacks

- Moore's law: brute force attacks get cheaper with time
- UNIX passwords:
  - time to check all 5 letter passwords (lower case)?
    - 26^5 =~ 10 million passwords
    - 1975: 1 day
    - 1992: 10 seconds
    - 2002: 0.01 seconds
  - how about six letters, requiring upper, lower, number, and control character?
    - 70^6 ~ 600 billion passwords
    - 1992: 6 days
    - 2002: with 100 PC's in parallel, <60 seconds (!!!)

What do you trust? Why?

- Can you trust your login prompt?
  - how do you know the person before you really logged out?
- Can you trust your web browser?
  - what if somebody modified the installed version to capture your passwords and bank account numbers?
  - did you download the browser over the web? How do you know it wasn't modified at the source, or in flight?
  - does your browser have vulnerabilities? How do you know the web sites you've visited haven't exploited them?
- Can you trust your email?
  - how do you know the sender sent the mail, and that it wasn't modified in flight?

The lure: an email message



Dear PayPal,

We recently noticed one or more attempts to log in to your PayPal account from a foreign IP address.

If you recently accessed your account while traveling, the unusual log in attempts may have been initiated by you. However, if you did not initiate the log ins, please visit PayPal as soon as possible to verify your identity:

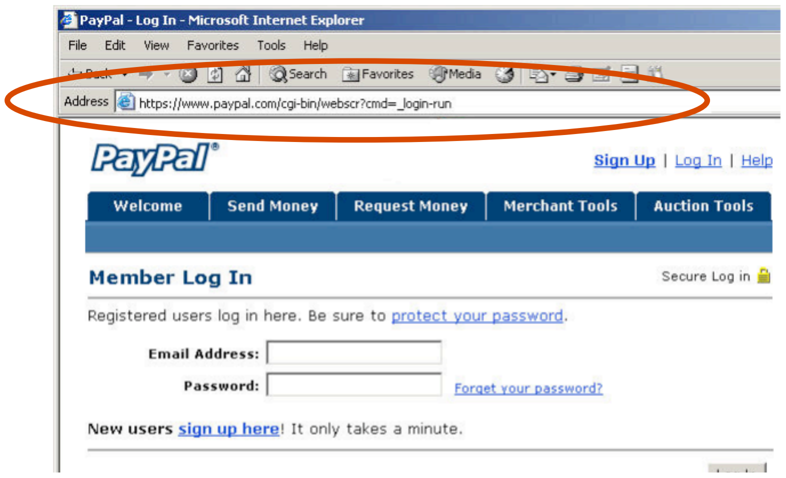
https://www.paypal.com/us/cgi-bin/webscr?cmd=_login-run

Verify your identity is a security measure that will ensure that you are the only person with access to the account.

Thanks for your patience as we work together to protect your account.

Sincerely,
PayPal

The mimicked website…



https://www.paypal.com/cgi-bin/webscr?cmd=_login-run

Member Log In Secure Log in

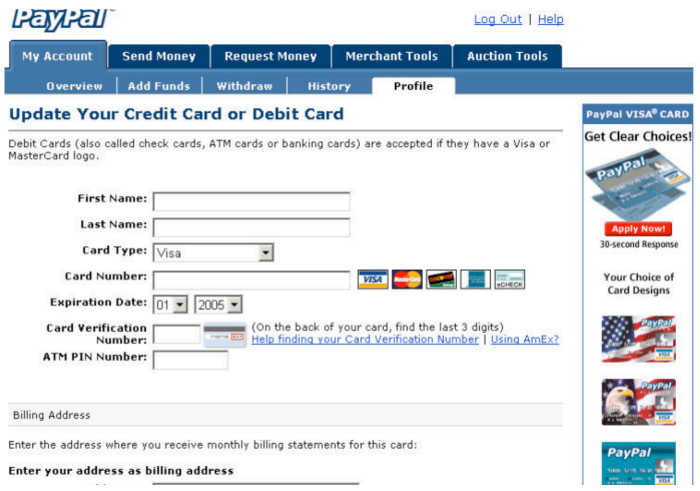
Registered users log in here. Be sure to protect your password.

Email Address:

Password: Forget your password?

New users sign up here! It only takes a minute.

The sinker….



Update Your Credit Card or Debit Card

Debit Cards (also called check cards, ATM cards or banking cards) are accepted if they have a Visa or MasterCard logo.

First Name:

Last Name:

Card Type: Visa

Card Number:

Expiration Date: 01 2005

Card Verification Number: (On the back of your card, find the last 3 digits) Help finding your Card Verification Number | Using AmEx?

ATM PIN Number:

Billing Address

Enter the address where you receive monthly billing statements for this card:

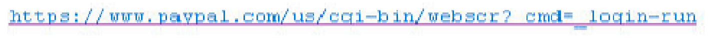
Enter your address as billing address

How did this work? (1/3)

- The email message itself is spam
  - sent to hundreds of millions of destination addresses
  - attacker only needs to harvest tiny fraction
- Spam is typically transmitted through "relays"
  - compromised PCs forced to run relay software
  - makes it harder to trace and shut down attacker

How did this work? (2/3)

- The link in the email message is really an image
  - like the web, email can contain hyperlinked images
    - clicking on the image takes you to the linked web page
  - the image is:


https://www.paypal.com/us/cgi-bin/webscr?_cmd=_login-run

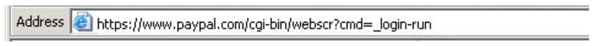
  - the link takes you to:

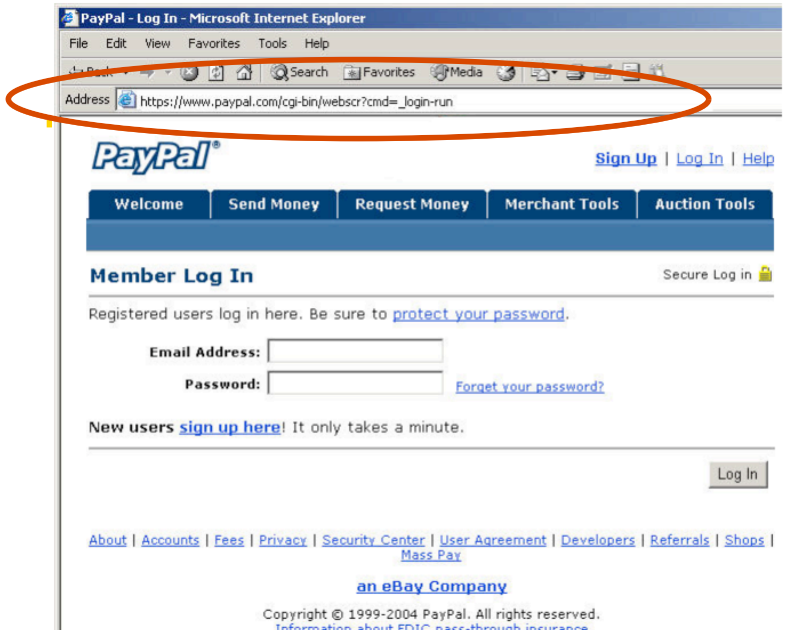
http://218.246.224.203/icons/.cgi-bin/paypal/cgi-bin/webscrcmd_login.php

  - 218.246.224.203 is some machine in China
    - most likely a compromised PC

How did this work? (3/3)

- The web page contains content that…
  - instructs IE not to show the real address bar
    - hides "http://218.246.224.203/icons/…" from user
  - displays images and text that spoofs an address bar containing a falsified URL


Address https://www.paypal.com/cgi-bin/webscr?cmd=_login-run


PayPal - Log In - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address https://www.paypal.com/cgi-bin/webscr?cmd=_login-run

Sign Up | Log In | Help

Welcome | Send Money | Request Money | Merchant Tools | Auction Tools

Member Log In Secure Log in

Registered users log in here. Be sure to protect your password.

Email Address:

Password: Forget your password?

New users sign up here! It only takes a minute.

Log In

About | Accounts | Fees | Privacy | Security Center | User Agreement | Developers | Referrals | Shops | Mass Pay

an eBay Company

Copyright © 1999-2004 PayPal. All rights reserved.

Information about FDIC pass-through insurance

Internet worms

- Worm performs the following steps:

```
while(1) {
    pick random IP address;
    scan IP address for known remote vulnerability;
    if is vulnerable {
        exploit vulnerability and copy self to remote host;
    }
}
```

- Deadly, but can do much better
  - non-random scanning, multiple vulnerabilities, etc.

Why are worms bad?

- They cause damage to victims
  - worms can carry “payloads”
    - e.g., install spyware
    - e.g., mount coordinated attack on a Web site
- They cause damage to the Internet
  - probing for victims and spreading causes Internet traffic
  - a fast-spreading worm can overwhelm Internet links

Famous examples

- Code Red v2 [2002]
  - attacked Microsoft IIS web servers
  - infected 500,000 victims within 10 hours
  - doubled in size every 37 minutes
- Sapphire [2003]
  - attacked Microsoft SQL server
  - infected 75,000 victims within 10 minutes
  - doubled in size every 8.5 seconds

Worse case scenario

- Hypothetical “hitlist” worm
  - probe for potential victims before releasing worm
  - attack these susceptible victims first
  - avoids “random probe” that most worms perform
- In principle, would infect millions within seconds

Ping of Death

- IP packets can be fragmented and reordered in flight
- Reassembly done at remote host
  - can get fragments out of order, so host OS much allocate a buffer to hold fragments
- Malformed IP fragment is possible
  - offset + length > max packet size
  - Windows didn’t check this
  - could overflow buffer in Windows kernel
- Was used for denial of service (crash Windows)
  - but could have been used for worm propagation

DNS cache poisoning

- DNS queries/responses are unauthenticated
  - no encryption used
- Many attacks possible as a result
- DNS cache poisoning:
  - attacker monitors network for a DNS query flowing by
    - e.g., for www.google.com
  - attacker spoofs a reply to "poison" the cache of whomever asked the query
    - spoofed response points to server of attacker's choosing
  - Imagine if UT's DNS servers are poisoned…

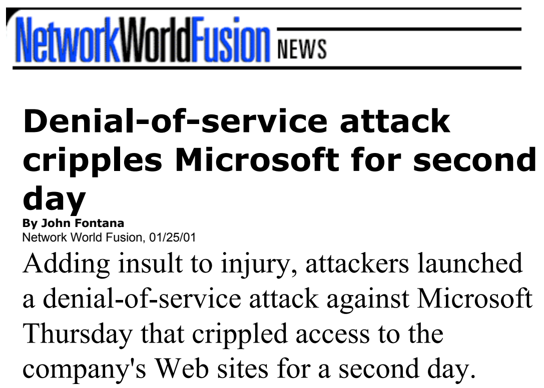
Browser hacks

- Netscape used to use time of day, process ID to seed random number generator
  - random number used to pick conversation key
  - easy to predict, and therefore break
- Netscape used to be downloaded without encryption
  - four byte change to executable made it use attacker's key
- Plenty of browser bugs
  - drive-by download: web server exploits bug to 0wn client
  - phishing attacks: attack web site looks like authoritative site
    - often combined with homograph attack:
      - www.goog1e.com

Social engineering

- Con person into giving out information
- Phone secretary, say:
  - "Hi. I'm your company's IT administrator. Your boss is currently traveling, and I can't reach them. I need their password to verify their account hasn't been broken into. This is really urgent."
- Somebody phones you, and says:
  - "Hi. I'm with the Bank of America credit card fraud division. We've detected suspicious activity on your account, and we want to ensure you haven't become a victim of identity theft. Before we start, I need to verify your identity. What is your bank account number? SSN?"
- Often far more effective than technical attack
  - requires all people with access to sensitive information to be conscious of security issues

Denial of Service in the News



NetworkWorldFusion NEWS

Denial-of-service attack cripples Microsoft for second day

By John Fontana
Network World Fusion, 01/25/01

Adding insult to injury, attackers launched a denial-of-service attack against Microsoft Thursday that crippled access to the company's Web sites for a second day.

What is Denial of Service?

- Attacker can deny service to legitimate users if they can overwhelm the system providing the service
  - System is full of bugs … just send it packets that trigger them
  - System has limited bandwidth, CPU, memory, etc. … just sent it too many packets to handle
- Big issue in practice and lack of effective solutions
  - Today, patch as found (CERT) or build implementation to tolerate DOS
  - Tomorrow, design protocols to withstand, possibly network support for shutting down attack?
- Two broad classes:
  - Nasty packets trigger implementation bugs, e.g., Ping of Death
  - Packet floods target bandwidth, CPU, memory, e.g., SYN flood

Complication: Spoofed Addresses

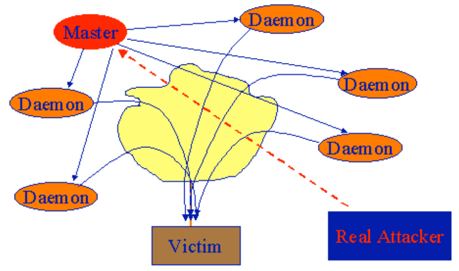
- Why reveal your real address? Instead, “spoof” it.
  - Can implicate others and appear to be many hosts
- Solution?
  - Ingress filtering (ISPs check validity of source addresses) helps, but has poor incentive patterns and is not a complete solution
- Opportunity: “backscatter analysis”
  - host reponds to spoofed packet, sends response packet to essentially random IP
  - if you have a large number of unused IPs, just listen and you’ll hear the backscatter -- can measure DOS attacks!

Complication: Reflectors & Amplifiers

- Some packets arriving “out of the blue” trigger a reply
  - Use this with spoofing to launder attack traffic (e.g., DNS)
  - Use with broadcast addresses to amplify attack (e.g., Smurf)

Distributed DOS (DDOS)

- Use automated tools to set up a network of zombies
  - Trin00, TFN, mstream, Stacheldraht, …



Lessons

- Encryption is powerful tool
  - strong mathematical properties
  - used to provide integrity, authenticity, privacy
  - must be used correctly
- Many other security issues in practice
  - non-mathematical, “best practice” based
  - easy to get wrong
- In the end, people are the weak link
  - social engineering