A Brief History of the World

BRINGING CIVILIZATION TO ITS KNEES...

Goths

Vandals

Huns

Geeks

HACK
HACK

HACK
HACK

HACK
HACK

HACK
HACK

HACK
HACK
Network Security

Week 7
Why and Who Attack Networks?

- Challenge: Hackers
- Money: Espionage
- Money: Organized Crime
- Ideology: Hacktivists/Cyberterrorists
- Revenge: Insiders
Intrusion Techniques

- Reconnaissance
- Eavesdropping and Wiretapping
- Impersonation
- Message confidentiality threats
- Web site vulnerabilities
- DOS and DDOS
Reconnaissance

- Port scan
  - For a given address find which ports respond
- OS and application fingerprinting
  - Certain features reveal OS/apps manufacturer and versions
  - Nmap: guess the OS and version, what services are offered

![Port scan result]

Starting nmap 3.75 http://www.insecure.org/nmap/ at 2004-10-26 11:31 PDT
Interesting ports on scanme.nmap.org (205.217.155.55):
(The 1638 ports scanned but not shown below are in state: filtered)

<table>
<thead>
<tr>
<th>PORT</th>
<th>STATE</th>
<th>SERVICE VERSION</th>
</tr>
</thead>
<tbody>
<tr>
<td>22/tcp open ssh</td>
<td>OpenSSH 3.1p1 (protocol 1.99)</td>
<td></td>
</tr>
<tr>
<td>25/tcp open smtp</td>
<td>email-smtp</td>
<td></td>
</tr>
<tr>
<td>80/tcp open http</td>
<td>Apache/2.0.39 (Fedora Core 2) mod_perl/1.98-07-dev Perl/5.6.1</td>
<td></td>
</tr>
<tr>
<td>111/tcp closed auth</td>
<td>Device type: general purpose</td>
<td></td>
</tr>
<tr>
<td>69/udp closed</td>
<td>Running: Linux-2.4.17-4.EL #1 SMP</td>
<td>OS details: Linux 2.4.17-4.EL #1 SMP, Linux kernel 2.4.17-4.EL, Uptime 176.575 min (since Sun May 30 22:08:10 2004)</td>
</tr>
</tbody>
</table>

Nmap run completed -- 1 IP address (1 host up) scanned in 27.008 seconds

Felix/W
Reconnaissance (cont’d)

- Social engineering
  - Use social skills
  - Pretend to be someone else and ask for details
  - Run `ipconfig - all`

- Intelligence
  - Dumpster diving
  - Eavesdropping
  - Blackmail

- Bulletin boards and Chats
Social Engineering

- People can be just as dangerous as unprotected computer systems
  - People can be lied to, manipulated, bribed, threatened, harmed, tortured, etc. to give up valuable information
Social Engineering

- Pretexting
- Phishing
- Baiting
- Quid Pro Quo
- Tailgating
Pretexting

- Example 1:
  - “Hi, I’m your AT&T rep, I’m stuck on a pole. I need you to punch a bunch of buttons for me”
Pretexting

- **Example 2:** Call in the middle of the night
  - “Have you been calling Egypt for the last six hours?”
  - “No”
  - “Well, we have a call that’s actually active right now, it’s on your calling card and it’s to Egypt and as a matter of fact, you’ve got about $2000 worth of charges on your card and ... **read off your AT&T card number and PIN** and then I’ll get rid of the charge for you”
Phishing

- E-mail
  - Appears to come from a legitimate business
  - Requests "verification" of information
    - Home address
    - Password, PIN, SSN, credit card number
  - Dire consequences if not provided
  - Contains a link to a fraudulent web page that seems legitimate—with company logos and content
Baiting

- Physical world Trojan horse
- Attacker leaves a malware infected CD, flash drive in public space
- Write something appealing on front
  - "Executive Salary Summary Q1 2016"
- Exploit finder curiosity
Intrusion Techniques

- Reconnaissance
- Eavesdropping and Wiretapping
- Impersonation
- Message confidentiality threats
- Web site vulnerabilities
- DOS and DDOS
Wiretapping

- Cable
  - Packet sniffers
  - Inductance/radiation emitted, Cutting the cable
- Satellite
  - Easily intercepted over large areas
- Optical fiber
  - Harder to wiretap
  - Repeaters, splices and taps are vulnerable
- Wireless
  - Easy to intercept, steal service and disrupt/interfere
Packet Sniffing

- Recall how Ethernet works ...
- When someone wants to send a packet to someone else
  - Put the bits on the wire with the destination MAC address
- Other hosts are listening on the wire to detect for collisions ...
- It couldn’t get any easier to figure out what data is being transmitted over the network!
Packet Sniffing (cont’d)

- This works for wireless too!
  - In fact, it works for any broadcast-based medium
- What kinds of data is of interest
- Answer:
  - Anything in plain text
  - Passwords
Intrusion Techniques

- Reconnaissance
- Eavesdropping and Wiretapping
- Impersonation
- Message confidentiality threats
- Web site vulnerabilities
- DOS and DDOS
Impersonation

- Access the system by pretending to be authenticated user
  - Password guessing/capture
  - Spoofing
Password Guessing

- Very common attack
- Attacker knows a login (from email/web page etc)
- Attempts to guess password for it
  - Defaults, short passwords, common word searches
  - User info (variations on names, birthday, phone, common words/interests)
    - Exhaustively searching all possible passwords
- Check by login or against stolen password file
- Success depends on password chosen by user
  - *Surveys show many users choose poorly*
Password Capture

- Watch over shoulder as password is entered
- Use key logger to collect
- Monitor an insecure network login
  - E.g. telnet, FTP, web, email
Password Capture using Sniffing

- Monitor an insecure network login
- **Example**: Microsoft LAN Manager
  - Hash of passwd was transmitted, not passwd
  - At most 14 characters
  - *Split in blocks of 7 chars, each with a different hash!*
  - If 7 chars or less, second hash is of nulls
  - If 8 chars, second hash is of single char
  - Vulnerable to brute force attacks
Password Collection Protection

- **SSH, not Telnet**
  - Many people still use Telnet and send their password in the clear (use PuTTY instead!)
  - Now that I have told you this, please do not exploit this information
  - Packet sniffing is, by the way, prohibited by Computing Services
- **HTTP over SSL**
  - Especially when making purchases with credit cards!
- **SFTP, not FTP**
  - Unless you *really* don’t care about the password or data
- **IPSec**
  - Provides network-layer confidentiality
Spoofing

- Pretend to be someone else
  - Masquerade
  - Session Hijacking
  - Man-In-the-Middle-Attack
Masquarade

- One host pretends to be someone else
- Easy to confuse names or mistype
- Example: BlueBank vs Blue-Bank (masquerade)
  1. Blue-Bank copies web page of BlueBank
  2. Attracts customers of BlueBank
     - Phishing, Ads, Spam, etc ...
  3. Ask customer to enter account name and passwd
  4. Optional: redirect connection to BlueBank
- Try http://www.sonicwall.com/furl/phishing/ to test your phishing nose
Session Hijack vs. MitMA

- Intercept and carry on session begun by another entity

  Example:
  - Administrator uses telnet to login to privileged account
  - Attacker intrudes in the communication and passes commands as if on behalf of admin

- Man-In-The-Middle Attack
  - Similar, but...
  - Attacker needs to participate since session start
Intrusion Techniques

- Reconnaissance
- Eavesdropping and Wiretapping
- Impersonation
- Message confidentiality threats
- Web site vulnerabilities
- DOS and DDOS
Message Confidentiality Threats

- Misdelivery
  - Mistyping the destination address

- Exposure
  - Packets are exposed over wires and in buffers at
    - Switches, gateways, routers, ...

- Traffic Flow Analysis
  - The existence of communication leaks information
Intrusion Techniques

- Reconnaissance
- Eavesdropping and Wiretapping
- Impersonation
- Message confidentiality threats
- Web site vulnerabilities
- DOS and DDOS
Anyone has access to the code of a web page
  - Also the order in which pages are accessed
Example vulnerabilities:
  - Web site defacement
  - Buffer overflows
Intrusion Techniques

- Reconnaissance
- Eavesdropping and Wiretapping
- Impersonation
- Message confidentiality threats
- Web site vulnerabilities
- DOS and DDOS
Denial of Service

- Make a network service unusable, usually by overloading the server or network
- Many different kinds of DoS attacks
  - SYN flooding
  - SMURF
  - Distributed attacks
TCP Three Way Handshake

- **SYN**: Client sends a SYN to the server
  - The segment sequence number is a random value A
- **SYN-ACK**: Server replies with a SYN-ACK
  - The acknowledgment number is set to one more than the received sequence number (A + 1)
  - Sequence number that the server chooses for the packet is another random number B
- **ACK**: Client sends an ACK back to the server
  - The acknowledgement number is set to one more than the received sequence number B + 1
  - Sequence number is set to the received acknowledgement value A + 1
**SYN Flooding Attack**

- **Send SYN packets with bogus source address**
  - Why?
- Server responds with SYN+ACK and keeps state about TCP half-open connection
- Eventually, server memory exhausted with state
- **Solution:** use “SYN cookies”
SYN Cookies

- In response to a SYN, create a special “cookie” for the connection, and forget everything else

- Let:
  - \( t = \text{timestamp} \)
  - \( m = \text{maximum segment size (MSS) value that the server would have stored in the SYN queue entry} \)
  - \( s = H_K(t, IP_{srv}, port_{srv}, IP_{cli}, port_{cli}) \)

- SYN Cookie: initial sequence number B
  - First 5 bits: \( t \mod 32 \)
  - Next 3 bits: an encoded value representing \( m \)
  - Final 24 bits: \( s \mod (\text{some prime of 24 bits}) \)
SYN Cookies

- **ACK**: Client sends an ACK back to the server.
  - The acknowledgement number is set to one more than the received sequence number \( N = B + 1 \)

- The server performs the following operations:
  - Break \( N-1 \) into \( t, m, s \) fields (by length)
  - Check the value \( t \) against the current time to see if the connection is expired
  - Compare \( s = H_K(t, \text{IP}_{srv}, \text{port}_{srv}, \text{IP}_{cli}, \text{port}_{cli}) \) ?
  - Decode \( m \) from the 3-bit encoding in the SYN Cookie
    - Reconstruct the SYN queue entry
Smurf Attack

Honey! I think our network is having another Smurf attack!
Smurf Attack

- **ICMP echo request (ping) traffic to IP broadcast address**
  - Source IP address of a broadcast ping is spoofed - victim
  - Large number of machines respond back to victim, overloading it
Smurf Attack - ICMP

- **ICMP echo** (spoofed source address of victim)
  - Sent to IP broadcast address
- **ICMP echo reply**
Smurf Attack Defenses

1. Configure individual hosts and routers not to respond to ping requests or broadcasts.

2. Configure routers not to forward packets directed to broadcast addresses.
Distributed Denial of Service (DDoD)

- Same as regular DoS, but on a larger scale
- Example: Sub7Server Trojan and IRC bots
  - Infect a large number of machines with a “zombie” program
  - Zombie program logs into an IRC (Internet Relay Chat) channel and awaits commands
    - Bot command: `!p4 207.71.92.193`
    - Result: runs `ping.exe 207.71.92.193 -l 65500 -n 10000`
    - Sends 10,000 64k packets to the host (655MB!)
Mini Case Study – Code Red

- **July 19, 2001**: over 359,000 computers infected with Code-Red in less than 14 hours
- Used a recently known buffer exploit in Microsoft IIS
- Damages estimated in excess of $2.6 billion
- Launched a DDOS attack against [www1.whitehouse.gov](http://www1.whitehouse.gov) from the 20th to the 28th of every month!
- Spent the rest of its time infecting other hosts
Defenses against DDoS

- Intrusion Detection
- Blacklisting and Firewalls
- CloudFlare
No CloudFlare

When visitor types allen.com
- Browser contacts DNS
- Gets back 1.1.1.1
- Sends request to 1.1.1.1

Without CloudFlare

Allen.com server IP: 1.1.1.1
With CloudFlare

CloudFlare: sits between the visitor and the website it protects
CloudFlare

- Has (collaborates with) data centers around the world
- For the initial DNS request: route the request to the data center closest to visitor
  - The result: IP in the CloudFlare data center closest to visitor
  - Not 1.1.1.1, but 99.99.99.99
  - Visitor makes request to 99.99.99.99 (not 1.1.1.1)
CloudFlare

- CloudFlare's edge servers on IP 99.99.99.99 address receive the request for the protected website.
- Analyze the traffic before sending to protected website.
- Verify if the visitor appears to be a threat based on:
  - The visitor's IP address (blacklisting/firewall)
  - Requested resources
  - Payload posted (malware, buffer overflow, SQL injection, etc)
  - Frequency of requests
CloudFlare Caching

- Speed up the response time
- Cache parts of websites that are static in CloudFlare servers
  - Images, CSS, and JavaScript
  - Do not cache HTML (to not mess up dynamic pages)
CloudFlare Request Handling

- If the visitor is not a threat
- Front server checks the request against the cache
- Serve from cache if found
- Otherwise, request page (from IP 99.99.99.99 to the original webpage (1.1.1.1))
CloudFlare Advantage

- Only CloudFlare knows the IP of webserver (1.1.1.1)
- CloudFlare protects multiple clients (webservers)
- Sees many attacks and attackers
- Can build more efficient blacklists
- Can use machine learning to detect existing and new attacks (similar to intrusion detection systems)