CS 475 : Network Attacks and Defenses

Rachel Greenstadt
Reminders

- Project 2 due today
- Thursday - about Project 3
# Network Review

## OSI Reference Model

<table>
<thead>
<tr>
<th>Layer</th>
<th>Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Layer</td>
<td>System Independent data</td>
</tr>
<tr>
<td>Presentation Layer</td>
<td>Basically Ignored</td>
</tr>
<tr>
<td>Session Layer</td>
<td>Reliable Streams</td>
</tr>
<tr>
<td>Transport Layer</td>
<td>Routing</td>
</tr>
<tr>
<td>Network Layer</td>
<td>Packets</td>
</tr>
<tr>
<td>Logical Link Layer</td>
<td>Unstructured bits</td>
</tr>
<tr>
<td>Physical Layer</td>
<td></td>
</tr>
</tbody>
</table>

Tuesday, May 19, 15
Network Review

OSI Reference Model
Network Desired Properties (Assets)

- Availability
- Integrity (Consistency)
- Authentication
- Confidentiality
Types of Attack (Adversaries)

• Passive attack
  • Eavesdrop but do not modify

• Active attack
  • Transmit, replay, modify, delete messages from network, covert channels

• Local vs remote attacks
Basic Problems (Threats)

- Network protocols have no integrity or confidentiality
- Why?
  - “It was a more innocent time”
- Export controls

  *Mary had a little key
  (It’s all she could export)*

  *And all the email that she sent
  Was opened at the fort.*

  -- Ron Rivest

  (via Kaufman, Perlman, and Speciner)
Basic Problems (Threats)

- Network protocols have no integrity or confidentiality
- Vulnerabilities in network services enable remote exploits
- End-to-end argument - dumb network
  - If you want security (or anything else) then get it
  - But what if the ends are incompetent? Or what if only “one end” supports it?
What network layer should you secure?

- Layer 4 (transport/TCP) and below in OS
- Above that is user level process
- Easier to deploy if you don’t need to change the OS (SSL/TLS and SSH)
- But if you secure layer 3, security happens automagically without application mods (IPSec)
- Plus, SSL can’t tell TCP that it’s integrity check failed, so SSL will discard bad data, then TCP will discard resent good data as duplicates.
On the other hand...

- IPSec can’t tell layers above it anything else that which IP something was sent from (not which user, even if it knows)

- Same as IPSec between two firewalls
  - Encrypt Traffic - eavesdropper protection
  - Policies can allow/deny IP addresses/ports
  - Address-based authentication
  - No other authentication
Physical Layer Attacks

- Wire taps (or wire cutting)
- Electronic emanation
breaking secrecy of the ballot with a radio scanner

10 October 2006
Link Layer (LANs)

- **ARP is protocol for finding the link layer (MAC) address from IP address on local network**
  - **ARP Request.** Computer A asks the network, "Who has this IP address?"
  - **ARP Reply.** Computer B tells Computer A, "I have that IP. My MAC address is [whatever it is]."
- **Reverse ARP Request (RARP).** Same concept as ARP Request, but Computer A asks, "Who has this MAC address?"
  - **RARP Reply.** Computer B tells Computer A, "I have that MAC. My IP address is [whatever it is]."
- Replies can be sent without requests, results are cached...
- Any problems with this?
Abusing ARP

• Can send ARP messages claiming to be another computer on the LAN, bad result cached
  • Traffic gets sent to attacker instead
  • Snooping, modifying, or DOS (associate a nonexistent MAC address with them)
Abusing TCP/IP

- Three way handshake produces many opportunities for denial of service
SYN Flood

- Send many SYN packets, never acknowledge the replies
- Too many open connections overload machine
- SYNcookie defense: choose Y to be E(X), no need to keep state about open connection
- Can reconstruct queue if needed
Stateless cookie protocol

Initiator

I want to talk

\[ c = \text{hash (IP addr, secret)} \]

\[ c, \text{ start rest of protocol} \]

Bob

verify \( c \) before continuing

Tuesday, May 19, 15
Syncookies

- Cookie (syn ack) is timestamp $t$ (5 bits) + MSS + hash(client/server IP + port, $t$)

- When receive ack, subtract 1 to get the cookie.

  - Check $t$ to see if expired

  - Reconstruct hash to see if valid cookie

  - Decode MSS to reconstruct queue
Smurf Attacks

- Broadcast IP addresses allow you to send messages to a whole subnet at once
- Send ICMP echo requests (ping) to broadcast address but spoof the return address as victim
- The subnet will now flood the victim with pings
- Fix - don’t answer pings to a broadcast address and shun those who do
Distributed DOS (DDOS)

• Gather a botnet of machines
• Have them all send traffic to target
• More traffic, harder to track down or block
Types of DDoS

• Bandwidth exhaustion - flood network with more traffic than they can consume

• Resource exhaustion - use up client resources (a web server that can handle only 4000 simultaneous sessions)

• Application exploitation - weaknesses in the actual application (account lock out after three log-in attempts...lock out users)
Further Spoofing

- Email spoofing (change from, reply, reply-to)
- DNS spoofing
  - return the wrong address for a page by guessing TXN ID and poison cache (summer 2008)
- Session hijacking - Mallory wants to pretend to be Alice to Bob, DOS Alice, guess TCP sequence number (so sequence numbers should be random)
DNS Vulnerabilities

• maps names to ip addresses

• www.drexel.edu \rightarrow 144.118.31.11

• distributed: root server delegates to .edu server
  delegates to drexel.edu server

• don’t want badhacker.drexel.edu answering for
  www.drexel.edu

• supposed to be solved by transaction ID (# btwn 0
  and 65535 that real server knows, others don’t)
DNS Attack

- Once an answer is received it is cached for TTL (usually one day)
- 1 day * 65,536 lookups / 2 = 84.5 years for 50% chance (not exactly)
- Four observations by Dan Kaminsky
  - Bad guy doesn’t have to look anything up, so replies first (if right TXID)
  - Bad guy can try numbers until good guy returns (maybe 100?)
  - TTL only stops lookups for www.foo.com, not random other names like name1.foo.com, name2.foo.com, etc
  - name83.foo.com can win www.foo.com by delegating his answer to www.foo.com at some wrong address (6.6.6.6)
The Stopgap fix

- Many ways to force a lookup, do this attack
- So, add another 16 bits of randomness, via source port
- Before: 65536 to 1 odds
- After: Between 163,840,000 to 1 and 2,147,483,648 to 1 odds
- This is an improvement
  - That’s a lot of traffic to go unnoticed
  - Not necessarily too much
  - Long term solution?
I want to know the shortest path between SRC and DST.

So, the routers must exchange local information!
Routing Attacks

- Routers assume computers are honest and rely on them to tell them about paths through the Internet
- So the attacker lies
- Easy to black hole traffic (advertise a short route to nowhere)
- Pakistan did this to youtube
BGP Eavesdropping

- Border Gateway Protocol used to advertise paths between ASes on Internet
- Intercept traffic to target addresses
- Routers listen to the most specific advertisement (smallest set of IP addresses) so attacker advertise a narrower chunk to the wrong place
- AS-path prepending to cause select routers to reject advertisement and forward traffic to real destination (so no one notices)
Encrypting Network Traffic

- Before 90s, most attacks were network attacks
- Watch the traffic, grab a password
- SSH made a big difference
Network Sniffing Attacks

• Unencrypted wifi connections
  • http vs https, cookie stealing
  • Firesheep, cookie cadger

• Rogue access points
Secure Network Configurations

• Best way to secure a network, secure machines in the network

• Keep them patched, don’t run insecure services, teach users about security

• This is expensive and difficult
Firewalls

• Most commercially successful network security product
Types of Firewalls

• Ingress filtering
  • Block addresses and port numbers coming in
  • Block connections based on TCP headers

• Application proxies - serve as intermediaries for mail, web, etc...strip out bad stuff (spam, active web content)

• Egress filtering - block packets leaving network (classified information, attacks from within)

• DMZ - space between multiple firewalls
Firewall weaknesses

*Firewalls are obsolete now that we have users behind them*

- Hard to block bad things without blocking good things
- Oppresses sophisticated users
- Firewalls have holes
- One internal machine can spread attack
VPNs

- Encrypt and integrity-protect traffic between firewalls
- Privacy and security while traversing Internet
Intrusion Detection

- Signature detection
  - Prevent known attack vectors
  - Won’t catch new attacks
- Anomaly detection
  - Block weird traffic - false positive problem
  - Receiving operator characteristic (ROC curve)
ROC curves

- Plot detection probability vs false alarm probability
- Dominant curves (above and to the left)

Figure 1: Example of a ROC curve. The red line represents the trade-off between TPR and FPR. The intersection between the red and the green line is where we find the Equal Error Rate.
Network Threat Modeling
Using Microsoft STRIDE model

- **Spoofing** describes any threat that allows an attacker (or accidentally causes a user) to pretend to be someone or something else.

- **Tampering** describes any threat that allows an attacker (or accidentally causes a user) to alter or destroy data which the application has not allowed them to.

- **Repudiation** describes any threat that allows an attacker to deny that they have taken an action that your application allowed.

- **Information Disclosure** describes any threat that allows an attacker (or accidentally causes a user) to see data which the application should not allow them to.

- **Denial of Service** describes any threat that allows an attacker (or accidentally causes a user/service) to prevent, or reduce, legitimate access to services or data which the application should be providing.

- **Elevation of Privilege** describes any threat that allows an attacker (or accidentally causes a user) to have access to data or functionality which the application should not allow them to.