CSE331: Introduction to Networks and Security

Lecture 11
Fall 2004
Announcements

• HW2 Graded:
  – Average: 76  //  Median 73  //  Std. Dev. 13

• HW3 Due on Wednesday

• Project 1 Due on Friday

• Midterm #1 will be held in class on Monday
  – Will cover all material so far
  – Similar format to homeworks
Protocol Stack Revisited

Application

Presentation

Session

Transport

Network

Data Link

Physical

SMTP, HTTP, SNMP, FTP, …

So far…
Common Features

• SMTP, HTTP, SNMP, …
  – Request/Reply protocols built on TCP or UDP
  – Designed to handle a fixed set of messages
  – Companion data format
  – Many applications

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Data Format</th>
<th>Programs</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMTP</td>
<td>RFC 822 and MIME</td>
<td>Pine, NSMail, Eudora, Outlook,…</td>
</tr>
<tr>
<td>HTTP</td>
<td>HTML</td>
<td>Explorer, Netscape, Opera,…</td>
</tr>
<tr>
<td>SNMP</td>
<td>MIB</td>
<td>snmpget, snmpset,…</td>
</tr>
</tbody>
</table>
SMTP: Simple Mail Transfer Protocol

- Data format RFC822
  - [http://www.faqs.org/rfcs/rfc822.html](http://www.faqs.org/rfcs/rfc822.html)
  - ASCII text
  - Header and Body

- MIME: Multipurpose Internet Mail Extensions
  - Mail systems assume ASCII
    - Only 64 valid characters A-Z, a-z, 0-9, +, /
  - Some datatypes include arbitrary binary data (e.g. JPEG)
  - Base64 encoding
    - 3 bytes of data map to 4 ASCII Characters
    - A=0,B=1,...
RFC822 Headers

- <CRLF>-terminated lines containing pairs of form **type: value**
- Many valid Header types
- Some headers filled out by client
  - To: stevez@cis.upenn.edu
  - Subject: CSE331
- Others filled out by mail delivery system
  - Date:
  - Received:
  - From:
From: Steve Zdancewic <stevez@cis.upenn.edu>
MIME-Version: 1.0
To: stevez@cis.upenn.edu
Subject: Example Mail
Content-Type: multipart/mixed; boundary="------------020307000708030506070607"

This is a multi-part message in MIME format.
------------020307000708030506070607
Content-Type: text/plain; charset=us-ascii; format=flowed
Content-Transfer-Encoding: 7bit

This is the body.

------------020307000708030506070607
Content-Type: text/plain; name="example.txt"
Content-Transfer-Encoding: 7bit
Content-Disposition: inline; filename="example.txt"

Hello

------------020307000708030506070607
Content-Type: image/jpeg; name="doc.jpg"
Content-Transfer-Encoding: base64
Content-Disposition: inline; filename="doc.jpg"

/9j/4AAQSkZJRgABAQEASABIAAD//gAXQ3JlYXRlZCB3aXRoIFRoZSBHSU1Q/9sAQwAIBgYHBgUIBwcHCQkICgwUDQwLCwwZEhMPFB0aHx4dGhwICQyAiLCMcHCg3KSwwMTQ0NB8nOT04...
SMTP

- Mail Reader
  - User edits/reads/search e-mail

- Mail Daemon
  - Process running on each host (port 27)
  - Uses SMTP/TCP to transmit mail to daemons on other machines
  - Most daemons based on Berkley’s sendmail

- Mail Gateways
  - Store and forward e-mail (much like IP router)
  - Buffers on disk
  - Attempts to resend
SMTP Mail Gateways

- No need for explicit host in e-mail address
  - User can receive mail at different machines
- Recipient’s machine may not be up
  - Mail gateway can hold message for later
SMTP Dialogs

• Client posts commands
  – HELO, MAIL, RCPT, DATA, QUIT
• Server responds with code and human-readable explanation
Example SMTP Dialog

HELO seas.upenn.edu
250 Hello daemon@smtpauth.seas.upenn.edu [158.130.12.180]

MAIL FROM:<stevez@seas.upenn.edu>
250 OK

RCPT TO:<billg@microsoft.com>
250 OK

RCPT TO:<ted@microsoft.com>
550 No such user here

DATA
354 Start mail input; end with <CRLF>.
Blah blah blah...
<CRLF>.
250 OK

QUIT
221 Closing Connection
Network Vulnerabilities

- **Anonymity**
  - Attacker is remote, origin can be disguised
  - Authentication

- **Many points of attack**
  - Attacker only needs to find weakest link
  - Attacker can mount attacks from many machines

- **Sharing**
  - Many, many users sharing resources

- **Complexity**
  - Distributed systems are large and heterogeneous

- **Unknown perimeter**
- **Unknown attack paths**
Syn Flood Attack

• Recall TCP’s 3-way handshake:
  – SYN    --- SYN+ACK  --- ACK

• Receiver must maintain a queue of partially open TCP connections
  – Called SYN_RECV connections
  – Finite resource (often small: e.g. 20 entries)
  – Timeouts for queue entries are about 1 minute.

• Attacker
  – Floods a machine with SYN requests
  – Never ACKs them
  – Spoofs the sending address (Why? Two reasons!)
Reflected denial of service

• Broadcast a ping request
  – For sender’s address put target’s address
  – All hosts reply to ping, flooding the target with responses

• Hard to trace

• Hard to prevent
  – Turn off ping? (Makes legitimate use impossible)
  – Limit with network configuration by restricting scope of broadcast messages
(Distributed) Denial of Service

• Coordinate multiple subverted machines to attack
• Flood a server with bogus requests
  – TCP SYN packet flood
  – > 600,000 packets per second
• Detection & Assessment?
  – 12,800 attacks at 5000 hosts! (in 3 week period during 2001)
  – IP Spoofing (forged source IP address)
• Prevention?
  – Filtering?
  – Decentralized file storage?
Network Address Translation

- Idea: Break the invariant that IP addresses are globally unique
NAT Behavior

• NAT maintains a table of the form:
  <client IP> <client port> <NAT ID>

• Outgoing packets (on non-NAT port):
  – Look for client IP address, client port in the mapping table
  – If found, replace client port with previously allocated NAT ID (same size as PORT #)
  – If not found, allocate a new unique NAT ID and replace source port with NAT ID
  – Replace source address with NAT address
NAT Behavior

• Incoming Packets (on NAT port)
  – Look up destination port number as NAT ID in port mapping table
  – If found, replace destination address and port with client entries from the mapping table
  – If not found, the packet is not for us and should be rejected

• Table entries expire after 2-3 minutes to allow them to be garbage collected
Benefits of NAT

• Only allows connections to the outside that are established from *inside*.
  – Hosts from outside can only contact internal hosts that appear in the mapping table, and they’re only added when they establish the connection
  – Some NATs support firewall-like configurability

• Can simplify network administration
  – Divide network into smaller chunks
  – Consolidate configuration data

• Traffic logging
Drawbacks of NAT

- Rewriting IP addresses isn’t so easy:
  - Must also look for IP addresses in other locations and rewrite them (may have to be protocol-aware)
  - Potentially changes sequence number information
  - Must validate/recalculate checksums

- Hinder throughput

- May not work with all protocols
  - Clients may have to be aware that NAT translation is going on

- Slow the adoption of IPv6?