Announcements

• Project 4
  ‣ Due Thursday at 5pm

• That’s it!
  ‣ ... except for the final exam
Chapter 8 roadmap

8.1 What is network security?
8.2 Principles of cryptography
8.3 Message integrity
8.4 End point authentication
8.5 Securing e-mail
8.6 Securing TCP connections: SSL
8.7 Network layer security: IPsec
8.8 Securing wireless LANs
8.9 Operational security: firewalls and IDS
IPsec: Network Layer Security

- network-layer secrecy:
  - sending host encrypts the data in IP datagram
  - TCP and UDP segments; ICMP and SNMP messages.
- network-layer authentication
  - destination host can authenticate source IP address
- two principal protocols:
  - authentication header (AH) protocol
  - encapsulation security payload (ESP) protocol

  - for both AH and ESP, source, destination handshake:
    - create network-layer logical channel called a security association (SA)
- each SA unidirectional.
- uniquely determined by:
  - security protocol (AH or ESP)
  - source IP address
  - 32-bit connection ID: Security Parameter Index (SPI)
Authentication Header (AH) Protocol

- provides source authentication, data integrity, no confidentiality
- AH header inserted between IP header, data field.
- protocol field: 51
- intermediate routers process datagrams as usual

AH header includes:
- connection identifier (SPI)
- authentication data: source- signed message digest calculated over original IP datagram.
- next header field: specifies type of data (e.g., TCP, UDP, ICMP)

---

IP header | AH header | data (e.g., TCP, UDP segment)
ESP Protocol

- provides secrecy, host authentication, data integrity.
- data, ESP trailer encrypted.
- next header field is in ESP trailer.

- ESP authentication field is similar to AH authentication field.
- Protocol = 50.
Chapter 8 roadmap

8.1 What is network security?
8.2 Principles of cryptography
8.3 Message integrity
8.4 End point authentication
8.5 Securing e-mail
8.6 Securing TCP connections: SSL
8.7 Network layer security: IPsec
8.8 Securing wireless LANs
8.9 Operational security: firewalls and IDS
IEEE 802.11 security

• **war-driving:** drive around Bay area, see what 802.11 networks available?
  ‣ More than 9000 accessible from public roadways
  ‣ 85% use no encryption/authentication
  ‣ packet-sniffing and various attacks easy!

• securing **802.11**
  ‣ encryption, authentication
  ‣ first attempt at 802.11 security: Wired Equivalent Privacy (WEP): a failure
  ‣ current attempt: 802.11i
Wired Equivalent Privacy (WEP):

- authentication as in protocol ap4.0
  - host requests authentication from access point
  - access point sends 128 bit nonce
  - host encrypts nonce using shared symmetric key
  - access point decrypts nonce, authenticates host
- no key distribution mechanism
- authentication: knowing the shared key is enough
WEP data encryption

- Host/AP share 40 bit symmetric key (semi-permanent)
- Host appends 24-bit initialization vector (IV) to create 64-bit key
- 64 bit key used to generate stream of keys, $k_i^{IV}$
- $k_i^{IV}$ used to encrypt $i$th byte, $d_i$, in frame:
  \[ c_i = d_i \oplus k_i^{IV} \]
- IV and encrypted bytes, $c_i$, sent in frame
Breaking 802.11 WEP encryption

security hole:
• 24-bit IV, one IV per frame, -> IV’s eventually reused
• IV transmitted in plaintext -> IV reuse detected
• attack:
  ‣ Trudy causes Alice to encrypt known plaintext \( d_1 \ d_2 \ d_3 \ d_4 \ldots \)
  ‣ Trudy sees: \( c_i = d_i \ XOR \ k_i^{IV} \)
  ‣ Trudy knows \( c_i \ d_i \), so can compute \( k_i^{IV} \)
  ‣ Trudy knows encrypting key sequence \( k_1^{IV} \ k_2^{IV} \ k_3^{IV} \ldots \)
  ‣ Next time IV is used, Trudy can decrypt!
802.11i: improved security

- numerous (stronger) forms of encryption possible
- provides key distribution
- uses authentication server separate from access point
Chapter 8 roadmap

8.1 What is network security?
8.2 Principles of cryptography
8.3 Message integrity
8.4 End point authentication
8.5 Securing e-mail
8.6 Securing TCP connections: SSL
8.7 Network layer security: IPsec
8.8 Securing wireless LANs
8.9 Operational security: firewalls and IDS
Firewalls

isolates organization’s internal net from larger Internet, allowing some packets to pass, blocking others.
Firewalls: Why

prevent denial of service attacks:
- SYN flooding: attacker establishes many bogus TCP connections, no resources left for “real” connections

prevent illegal modification/access of internal data.
- e.g., attacker replaces CIA’s homepage with something else

allow only authorized access to inside network (set of authenticated users/hosts)

three types of firewalls:
- stateless packet filters
- stateful packet filters
- application gateways
Stateless packet filtering

• internal network connected to Internet via router firewall
• router filters packet-by-packet, decision to forward/drop packet based on:
  ‣ source IP address, destination IP address
  ‣ TCP/UDP source and destination port numbers
  ‣ ICMP message type
  ‣ TCP SYN and ACK bits

Should arriving packet be allowed in? Departing packet let out?
Stateless packet filtering: example

- **Example 1**: block incoming and outgoing datagrams with IP “Protocol field” = 17 and with either source or dest port = 23.
  - all incoming, outgoing UDP flows and telnet connections are blocked.

- **Example 2**: Block inbound TCP segments with ACK=0.
  - prevents external clients from making TCP connections with internal clients, but allows internal clients to connect to outside.
<table>
<thead>
<tr>
<th><strong>Policy</strong></th>
<th><strong>Firewall Setting</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>No outside Web access.</td>
<td>Drop all outgoing packets to any IP address, port 80</td>
</tr>
<tr>
<td>No incoming TCP connections, except those for institution’s public Web server only.</td>
<td>Drop all incoming TCP SYN packets to any IP except 130.207.244.203, port 80</td>
</tr>
<tr>
<td>Prevent Web-radios from eating up the available bandwidth.</td>
<td>Drop all incoming UDP packets - except DNS and router broadcasts.</td>
</tr>
<tr>
<td>Prevent your network from being used for a smurf DoS attack.</td>
<td>Drop all ICMP packets going to a “broadcast” address (eg 130.207.255.255).</td>
</tr>
<tr>
<td>Prevent your network from being tracerouted</td>
<td>Drop all outgoing ICMP TTL expired traffic</td>
</tr>
</tbody>
</table>
Access Control Lists

- **ACL**: table of rules, applied top to bottom to incoming packets: (action, condition) pairs

<table>
<thead>
<tr>
<th>action</th>
<th>source address</th>
<th>dest address</th>
<th>protocol</th>
<th>source port</th>
<th>dest port</th>
<th>flag bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>allow</td>
<td>222.22/16</td>
<td>outside of 222.22/16</td>
<td>TCP</td>
<td>&gt; 1023</td>
<td>80</td>
<td>any</td>
</tr>
<tr>
<td>allow</td>
<td>outside of 222.22/16</td>
<td>222.22/16</td>
<td>TCP</td>
<td>80</td>
<td>&gt; 1023</td>
<td>ACK</td>
</tr>
<tr>
<td>allow</td>
<td>222.22/16</td>
<td>outside of 222.22/16</td>
<td>UDP</td>
<td>&gt; 1023</td>
<td>53</td>
<td>---</td>
</tr>
<tr>
<td>allow</td>
<td>outside of 222.22/16</td>
<td>222.22/16</td>
<td>UDP</td>
<td>53</td>
<td>&gt; 1023</td>
<td>----</td>
</tr>
<tr>
<td>deny</td>
<td>all</td>
<td>all</td>
<td>all</td>
<td>all</td>
<td>all</td>
<td>all</td>
</tr>
</tbody>
</table>
Stateful packet filtering

- stateless packet filter: heavy handed tool
  - admits packets that “make no sense,” e.g., dest port = 80, ACK bit set, even though no TCP connection established:

<table>
<thead>
<tr>
<th>action</th>
<th>source address</th>
<th>dest address</th>
<th>protocol</th>
<th>source port</th>
<th>dest port</th>
<th>flag bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>allow</td>
<td>outside of 222.22/16</td>
<td>222.22/16</td>
<td>TCP</td>
<td>80</td>
<td>&gt; 1023</td>
<td>ACK</td>
</tr>
</tbody>
</table>

- stateful packet filter: track status of every TCP connection
  - track connection setup (SYN), teardown (FIN): can determine whether incoming, outgoing packets “makes sense”
  - timeout inactive connections at firewall: no longer admit packets
## Stateful packet filtering

- ACL augmented to indicate need to check connection state table before admitting packet

<table>
<thead>
<tr>
<th>action</th>
<th>source address</th>
<th>dest address</th>
<th>proto</th>
<th>source port</th>
<th>dest port</th>
<th>flag bit</th>
<th>check conxion</th>
</tr>
</thead>
<tbody>
<tr>
<td>allow</td>
<td>222.22/16</td>
<td>outside of 222.22/16</td>
<td>TCP</td>
<td>&gt; 1023</td>
<td>80</td>
<td>any</td>
<td></td>
</tr>
<tr>
<td>allow</td>
<td>outside of 222.22/16</td>
<td>222.22/16</td>
<td>TCP</td>
<td>80</td>
<td>&gt; 1023</td>
<td>ACK</td>
<td>❌</td>
</tr>
<tr>
<td>allow</td>
<td>222.22/16</td>
<td>outside of 222.22/16</td>
<td>UDP</td>
<td>&gt; 1023</td>
<td>53</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>allow</td>
<td>outside of 222.22/16</td>
<td>222.22/16</td>
<td>UDP</td>
<td>53</td>
<td>&gt; 1023</td>
<td>---</td>
<td>❌</td>
</tr>
<tr>
<td>deny</td>
<td>all</td>
<td>all</td>
<td>all</td>
<td>all</td>
<td>all</td>
<td>all</td>
<td>all</td>
</tr>
</tbody>
</table>
Application gateways

- filters packets on application data as well as on IP/TCP/UDP fields.
- **example:** allow select internal users to telnet outside.

1. require all telnet users to telnet through gateway.
2. for authorized users, gateway sets up telnet connection to dest host. Gateway relays data between 2 connections
3. router filter blocks all telnet connections not originating from gateway.
Limitations of firewalls and gateways

- **IP spoofing**: router can’t know if data “really” comes from claimed source.
- If multiple app’s. need special treatment, each has own app. gateway.
- Client software must know how to contact gateway.
  - E.g., must set IP address of proxy in Web browser.
- Filters often use all or nothing policy for UDP.
- Tradeoff: degree of communication with outside world, level of security.
- Many highly protected sites still suffer from attacks.
Intrusion detection systems

• packet filtering:
  ‣ operates on TCP/IP headers only
  ‣ no correlation check among sessions

• **IDS: intrusion detection system**
  ‣ *deep packet inspection*: look at packet contents (e.g., check character strings in packet against database of known virus, attack strings)
  ‣ examine correlation among multiple packets
    • port scanning
    • network mapping
    • DoS attack
Intrusion detection systems

- multiple IDSs: different types of checking at different locations
Network Security (summary)

Basic techniques......

- cryptography (symmetric and public)
- message integrity
- end-point authentication

.... used in many different security scenarios

- secure email
- secure transport (SSL)
- IP sec
- 802.11

Operational Security: firewalls and IDS