The present need for security products far exceeds the number of individuals capable of designing secure systems. Consequently, industry has resorted to employing folks and purchasing “solutions” from vendors that shouldn’t be let near a project involving securing a system. -- Lucky Green

**Introduction**

- some basic concepts and terms
- examples for threats on the Internet
- classification of network security services and mechanisms

**Attack, threat, and vulnerability**

- security is about how to prevent attacks, or -- if prevention is not possible -- how to detect attacks and recover from them
- **attack**
  - a deliberate attempt to compromise a system
  - exploits vulnerabilities
- **vulnerability**
  - a flaw or weakness in a system's design, implementation, or operation and management
  - most systems have vulnerabilities
  - not every vulnerability is exploited
  - whether a vulnerability is likely to be exploited depends on
    - the difficulty of the attack
    - the perceived benefit of the attacker
- **threat**
  - a possible way to exploit vulnerabilities
  - a potential attack
Types of system compromises

- incorrect status of some system resources (static char.)
  - examples:
    - loss of confidentiality of sensitive data (e.g., passwords)
    - inappropriately set file access rights
    - incorrect configuration files

- incorrect behavior of some system components (dynamic char.)
  - examples:
    - malfunctioning devices, programs, services, ...

- decreased overall system dependability
  - the system works but the quality of service provided is not acceptable

Potential locations for attacks

- can be on any link or in control of any machine

[Diagram showing various network components such as hosts, subnets, and gateways under attacker's control.]
Passive vs. active attacks

- **passive attacks**
  - attempts to learn or make use of information from the system but does not affect system resources
  - examples:
    - eavesdropping message contents
    - traffic analysis
      - gaining knowledge of data by observing the characteristics of communications that carry the data
      - even if message contents is encrypted, an attacker can still
        - determine the identity and the location of the communicating parties
        - observe the frequency and length of the messages being exchanged
        - guess the nature of the communication
  - difficult to detect, should be prevented

- **active attacks**
  - attempts to alter system resources or affect their operation
  - examples:
    - masquerade (spoofing)
      - an entity pretends to be a different entity
    - replay
      - capture and subsequent retransmission of data
    - modification (substitution, insertion, destruction)
      - (some parts of the) legitimate messages are altered or deleted, or fake messages are generated
      - if done in real time, then it needs a "man in the middle"
    - denial of service
      - normal use or management of the system is prevented or inhibited
      - e.g., a server is flooded by fake requests so that it cannot reply normal requests
  - difficult to prevent, should be detected
Internet protocols and threats

ARP – Address Resolution Protocol

- mapping from IP addresses to MAC addresses

**Request**

```
arp req | target IP: 140.252.13.5 | target eth: ?
```

**Reply**

```
arp rep | sender IP: 140.252.13.5 | sender eth: 00:00:C2:9B:26
```
**ARP spoofing**

- an ARP request can be responded by another host

**Request**

```
1 2 3 4 5
08:00:20:03:F6:42 00:00:C0:C2:9B:26
```

```
fram req | target IP: 140.252.13.5 | target eth: ?
```

**Reply**

```
1 2 3 4 5
08:00:20:03:F6:42 00:34:CD:C2:9F:A0 00:00:00:00:00:00
```

```
fram rep | sender IP: 140.252.13.5 | sender eth: 00:34:CD:C2:9F:A0
```

**IP - Internet Protocol**

- provides an unreliable, connectionless datagram delivery service to the upper layers
- its main function is routing
- it is implemented in both end systems and intermediate systems (routers)
- routers maintain routing tables that define the next hop router towards a given destination (host or network)
- IP routing uses the routing table and the information in the IP header (e.g., the destination IP address) to route a packet
IP security problems

- user data in IP packets is not protected in any way
  - anyone who has access to a router can read and modify the user data in the packets
- IP packets are not authenticated
  - it is fairly easy to generate an IP packet with an arbitrary source IP address
- traffic analysis
  - even if user data was encrypted, one could easily determine who is communicating with whom by just observing the addressing information in the IP headers
- information exchanged between routers to maintain their routing tables is not authenticated
  - correct routing table updates can be modified or fake ones can be disseminated
  - this may screw up routing completely leading to loops or partitions
  - it may also facilitate eavesdropping, modification, and monitoring of traffic
  - it may cause congestion of links or routers (i.e., denial of service)

TCP – Transmission Control Protocol

- provides a connection oriented, reliable, byte stream service to the upper layers
- connection oriented:
  - connection establishment phase prior to data transfer
  - state information (sequence numbers, window size, etc.) is maintained at both ends
- reliable:
  - positive acknowledgement scheme (unacknowledged bytes are retransmitted after a timeout)
  - checksum on both header and data
  - reordering of segments that are out of order
  - detection of duplicate segments
  - flow control (sliding window mechanism)
**TCP connection establishment**

3 way handshake

- sequence numbers are 32 bit long
- the sequence number in a data segment identifies the first byte in the segment
- sequence numbers are initialized with a "random" value during connection setup
- the RFC suggests that the ISN is incremented by one at least every 4 µs

**TCP SYN attack**

- in Berkeley implementations, the ISN is incremented by a constant amount (64000)
  - once per 0.5 second, and
  - each time a connection is initiated
- it is not hopeless to guess the next ISN to be used by a server
- an attacker can impersonate a trusted host (e.g., in case of "r" commands, authentication is based on source IP address solely)
TCP SYN attack – How to guess ISN₀?

- ISN₀’ depends on ISN₀ and Δt
- Δt can be estimated from the round trip time
- assume Δt can be estimated with 10 ms precision
- the attacker has an uncertainty of 2500 in the possible value for ISN₀’
- assume each trial takes 5 s
- the attacker will likely succeed in 5x2500 = 12500 s = 3.5 hours

FTP – File Transfer Protocol

- typical FTP commands:
  - RETR filename - retrieve (get) a file from the server
  - STOR filename - store (put) a file on the server
  - TYPE type - specify file type (e.g., A for ASCII)
  - USER username - username on server
  - PASS password - password on server
FTP security problems

- neither the control nor the data connection is protected
  - passwords can be eavesdropped
    - FTP is a text(ASCII) based protocol, which makes password sniffing even easier
  - files transmitted over the data connection can be intercepted and modified

```
ftp ftp.epfl.ch
Connected to ftp.epfl.ch.
Name: buttyan
Password: kiskacsa
220 ftp.epfl.ch FTP server (version 5.60) ready.
USER buttyan
331 Password required for user buttyan.
PASS kiskacsa
230 User buttyan logged in.
```

Telnet

- provides remote login service to users
- text (ASCII) based protocol
Telnet security problems

- passwords are sent in clear

\[
\text{Client} \rightarrow \text{Server} \\
\text{Login: } b \rightarrow \text{"b"} \\
\text{Login: } bu \rightarrow \text{"u"} \\
\text{Login: } buttyan \rightarrow \text{"n"} \\
\text{Password: } k \rightarrow \text{"k"} \\
\text{Password: } kiskacsa \rightarrow \text{"a"} \\
\text{OS greetings and shell prompt, e.g., "%"}
\]

SMTP - Simple Mail Transfer Protocol

- User agent sends email to local MTA
- Local MTA sends email to SMTP relay MTA
- SMTP relay MTA receives email from local MTA
- SMTP relay MTA sends email to remote MTA
- Remote MTA delivers email to user mailbox
SMTP cont’d

- SMTP is used by MTAs to talk to each other
- SMTP is a text (ASCII) based protocol

sending MTA (rivest.hit.bme.hu) receiving MTA (shamir.hit.bme.hu)

`TCP connection establishment to port 25`

```
“HELO rivest.hit.bme.hu.”
“250 shamir.hit.bme.hu Hello rivest.hit.bme.hu, pleased to meet you”
“MAIL from: buttyan@rivest.hit.bme.hu”
“250 buttyan@rivest.hit.bme.hu... Sender ok”
“RCPT to: hubaux@lca.epfl.ch”
“250 hubaux@lca.epfl.ch... Recipient ok”
“DATA”
“354 Enter mail, end with a ‘.’ on a line by itself”
<message to be sent>.
“250 Mail accepted”
“QUIT”
“221 shamir.hit.bme.hu delivering mail”
```

SMTP security problems

- SMTP does not provide any protection of e-mail messages
  - messages can be read and modified by any of the MTAs involved
  - fake messages can easily be generated (e-mail forgery)
- Example:

  ```
  % telnet frogstar.hit.bme.hu 25
  Trying...
  Connected to frogstar.hit.bme.hu.
  Escape character is “^]“.
  220 frogstar.hit.bme.hu ESMTP Sendmail 8.11.6/8.11.6;
  Mon, 10 Feb 2003 14:23:21 +0100
  helo abcd.bme.hu
  250 frogstar.hit.bme.hu Hello [152.66.249.32], pleased to meet you
  mail from: bill.gates@microsoft.com
  250 2.1.0 bill.gates@microsoft.com... Sender ok
  rcpt to: buttyan@ebizlab.hit.bme.hu
  250 2.1.5 buttyan@ebizlab.hit.bme.hu... Recipient ok
  data
  354 Enter mail, end with “.” on a line by itself
  Your fake message goes here.
  .
  250 2.0.0 h1ADO5e21330 Message accepted for delivery
  quit
  221 frogstar.hit.bme.hu closing connection
  Connection closed by foreign host.
  %
  ```
Be careful, though!

HTTP - Hypertext Transfer Protocol

- HTTP is the protocol used by web servers and browsers
- interactive web sites are based on forms and scripts
  - the user fills the form and clicks on a button to submit it
  - this creates a request to the server that contains the data typed in by the user
  - the request launches a script on the server that processes the data supplied by the user
- if pure HTTP is used, then the form data are sent in clear
  - sensitive information can be eavesdropped and/or modified
  - examples for sensitive information:
    - passwords
    - credit card numbers
    - personal data
Security services

- processing or communication services that are provided by a system to give a specific kind of protection to system resources
- implement security policies -> closely related to general security objectives
- implemented by security mechanisms

X800 (OSI security architecture) security services:
- authentication
- access control
- confidentiality
- integrity
- non-repudiation

+ availability is treated as a property
  it’s not always obvious how to achieve it!
Authentication

- aims to detect masquerade (spoofing)
- provides assurance that a communicating entity is the one that it claims to be

peer entity authentication
- provides for the corroboration of the identity of a peer entity in an association (logical connection)
- can be performed at the establishment of, or at times during the lifetime of the connection

data-origin authentication
- provides assurance that the source of data received in a connectionless transfer is as claimed

Access control

- prevention of unauthorized access to a resource
  - who can have access to a resource?
  - under what conditions access can occur?
  - what is allowed to do with the resource?
Confidentiality

- protection of data from unauthorized disclosure
  
  **connection confidentiality**
  - confidentiality protection of all data transferred via a connection
  
  **connectionless confidentiality**
  - confidentiality protection of data in a single message
  
  **selective-field confidentiality**
  - confidentiality protection of selective fields within a single message or messages in a connection
  
  **traffic flow confidentiality**
  - protection of information that might be derived from observation of traffic flows

Integrity

- aims to detect modification and replay
- provides assurance that data received are exactly as sent by the sender
  
  **connection integrity**
  - provides for the integrity of a stream of messages (all data on a connection)
  - ensures that messages are received as sent, with no duplication, modification, insertion, deletion, reordering, or replays
  
  **connectionless integrity**
  - provides protection against modification of a single message
  - may provide limited forms of replay detection
  
  **selective field integrity**
  - provides for the integrity of selective fields within a single message or messages in a connection
Non-repudiation

- provides protection against denial by one entity involved in a communication of having participated in all or part of the communication

  non-repudiation of origin
  - provides proof that a message was sent by a specified party

  non-repudiation of delivery
  - provides proof that a message was received by a specified party

Specific security mechanisms

- encryption
  - symmetric, asymmetric
- digital signature
- access control schemes
  - access control lists, capabilities, security labels, ...
- data integrity mechanisms
  - message authentication code, sequence numbering, time stamping, cryptographic chaining
- authentication protocols
  - passwords, cryptographic techniques, biometrics
- traffic padding
- routing control
  - selection of physically secure routes
- notarization
  - e.g., time stamping, conflict resolution
**Relationship between services and mechanisms**

<table>
<thead>
<tr>
<th></th>
<th>encryption</th>
<th>digital signature</th>
<th>access control schemes</th>
<th>data integrity</th>
<th>authentication protocols</th>
<th>traffic padding</th>
<th>notarization</th>
</tr>
</thead>
<tbody>
<tr>
<td>peer entity</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>authentication</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>data origin</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>authentication</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>access control</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>confidentiality</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>traffic flow</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>confidentiality</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>data integrity</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>non-repudiation</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Placement of security services**

- some services can more naturally be implemented at the application layer (e.g., non-repudiation, access control)
- some services better fit in the link layer (e.g., traffic flow confidentiality)
- but many services can be provided at any layer (e.g., authentication, confidentiality, integrity)
  - lower layer:
    - services are generic, can be used by many applications
    - transparency to the user
  - higher layer:
    - services are more application specific
    - user awareness
**Link-oriented security**

- Provides security (e.g., integrity and confidentiality) for PDUs passing over a communication link between two nodes, regardless of the ultimate source and destination of the PDUs.
- Independent (different) keys are used on each link.
- **Pros**
  - Both protocol control information and user data can be encrypted.
  - Frequency and length patterns can be masked if a continuous stream of ciphertext bits is maintained on the link.
  - Very effective for defending against traffic analysis.
- **Cons**
  - All intermediate nodes between the source and the destination must be trusted.
  - Intermediate nodes are more complex (expensive).
  - Works well with point-to-point links only.

**End-to-end security**

- Protects PDUs on an association from source to destination without relying on communication link security.
- **Pros**
  - Intermediate nodes need not be trusted.
  - Can be used in a broader class of networks (e.g., packet broadcast) than link-oriented security.
  - Different communicating pairs can employ different end-to-end measures without affecting others.
- **Cons**
  - Doesn't protect against traffic analysis.
Summary

- basic concepts
  - vulnerability, threat, attack, security service, security mechanism
  - passive vs. active attacks
  - eavesdropping, traffic analysis, masquerade (spoofing), modification, replay, denial of service
  - authentication, access control, confidentiality, integrity, non-repudiation, availability

- real world examples
  - ARP spoofing, TCP SYN attack, e-mail forgery, eavesdropping Telnet and FTP passwords, DNS spoofing, denial of service, etc.

What’s next?

- cryptography and cryptographic protocols
- why?
  - many security mechanisms are based on cryptography (e.g., encryption, digital signature, some data integrity mechanisms, some authentication schemes, etc.)

- but be cautious:
  "If you think cryptography is going to solve your problem, you don’t understand cryptography and you don’t understand your problem."
  -- Bruce Schneier

- security is like a chain: it will break at the weakest link
  - other important aspects include
    - physical protection
    - procedural rules
      - choosing a hard to guess password and changing it regularly
      - installing patches
      - ...

Recommended reading

- RFC 2828: Internet Security Glossary