Chapter 11

Lecture 11 - Security

Last session

- Buffer overflow attacks
- PkZip attack
- DVDs and the CSS
- SSH and SSL
- PGPfone
This session

- Design principles
- Biometrics
- IPSec
- Formal methods
- Formal evaluation
- Exam

Design principles

Paper by Saltzer and Schroeder, summarized below:

- **Economy of mechanism**: Keep the design as simple and small as possible. (identd assumption)

- **Fail-safe defaults**: Base access decisions on permission rather than exclusion. This is conservative design. (mail server - mail only access)

- **Complete mediation**: Every access to every object must be checked for authority. (DNS cache poisoning)
Design principles

- **Open design**: The design should not be secret. (DVDs, Microsoft SAM hashes...)

- **Separation of privilege**: Two keys are better than one. No single event can compromise the system. (su - password and *wheel* group)

- **Least privilege**: Every program and every user of the system should operate using the least set of privileges necessary to complete the job. (Military need-to-know)

- **Least common mechanism**: Minimize the amount of mechanism common to more than one user and depended on by all users. (supervisor or library).

- **Psychological acceptability**: Human interface easy to use.

In the textbook there are examples of the use of each of these design principles.
Biometrics is the use of human physical characteristics to support authentication.
Biometrics - eyes

Minimal hardware biometrics

✔ **Voices** - Record and process voice leading to either speaker verification or recognition.

✔ **Faces** - Capture either a static or moving image of a face.

✔ **Keystrokes** - Capture a sequence of keystrokes, recording timing.

Combinations of characteristics may be used, but in general biometric techniques are not reliable on their own. Good second key for *separation of privilege*. 
This session

- Design principles
- Biometrics
- IPSec
- Formal methods
- Formal evaluation
- Exam

CS3235 - Hugh Anderson's notes.

Page number: 521

IPSec

✔ IPSec is a **set of standards** intended to support communication security between networked computers, particularly in the newer IPv6 (IP Next-Generation) network.

✔ IPSec software is available in Windows2000, Linux, and on routers on the Internet.

    http://www.faqs.org/rfcs/rfc2401.html

✔ IPSec may be used in a range of ways.
IPSec VPN

IPSec point-to-point
There are two types of header, one used for authentication, and the other used for encryption:

1. **AH** - the Authentication Header for data integrity, anti-replay and authentication

2. **ESP** - the Encapsulating Security Payload header, for confidentiality. ESP can also provide AH services.

Communicating parties agree on a Security Association (SA), one SA for each direction, and one SA for each type of communication.
Modes of operation

- An end-to-end SA - Transport mode

<table>
<thead>
<tr>
<th>Original IPv6 hdr</th>
<th>AH</th>
<th>Transport segment</th>
</tr>
</thead>
<tbody>
<tr>
<td>authenticated</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Original IPv6 hdr</th>
<th>ESP</th>
<th>Transport segment</th>
<th>ESP</th>
</tr>
</thead>
<tbody>
<tr>
<td>encrypted</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>authenticated</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- An SA between security gateways - Tunnel mode

<table>
<thead>
<tr>
<th>New IPv6 hdr</th>
<th>AH</th>
<th>Original IPv6 hdr</th>
<th>Transport segment</th>
</tr>
</thead>
<tbody>
<tr>
<td>authenticated</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>New IPv6 hdr</th>
<th>ESP</th>
<th>Original IPv6 hdr</th>
<th>Transport segment</th>
<th>ESP</th>
</tr>
</thead>
<tbody>
<tr>
<td>encrypted</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>authenticated</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SAs form a kind of distributed database.
This session

- Design principles
- Biometrics
- IPSec
- Formal methods
- Formal evaluation
- Exam

Formal methods

- FM encompasses a wide range of techniques...
- Model checking:
  - constructing formal models, with
  - appropriate formal specifications.
- Example is Promela and Spin.
Promela and spin

✔ The **language** Promela is 'C' like, with an initialization procedure. It can model asynchronous or synchronous, deterministic or non-deterministic systems

✔ Spin is the **checker** for Promela models

✔ Assertions to test correctness of model:

\[
\text{assert(some\_boolean\_condition)};
\]

✔ If condition not TRUE then assertion violated.

Temporal claims

✔ *We got here again without making any progress!*

✔ The support for temporal claims takes the form of:

✔ **Endstate** labels - for determining valid endstates
✔ **Progress** labels - claim no non-progress cycles
✔ **Never** claims - impossible temporal assertions
Simple example

Promela example

```promela
init
{
  chan AtoB = [1] of { mtype,byte }
  chan BtoA = [1] of { mtype,byte }
  chan Ain = [2] of { mtype,byte }
  chan Bin = [2] of { mtype,byte }
  chan Aout = [2] of { mtype,byte }
  chan Bout = [2] of { mtype,byte }
  atomic {
    run application( Ain,Aout );
    run transfer( Aout,Ain,BtoA,AtoB );
    run transfer( Bout,Bin,AtoB,BtoA );
    run application( Bin,Bout )
  };
  AtoB!err(0)
}
```
Promela example

```c
#define MAX 10
mtype = { ack, nak, err, next, accept }
proctype transfer( chan in, out, chin, chout )
{
    byte o,i;
in?next(o);
do
    :: chin?nak(i) -> out!accept(i); chout!ack(o)
    :: chin?ack(i) -> out!accept(i); in?next(o); chout!ack(o)
    :: chin?err(i) -> chout!nak(o)
    od
}
```

Promela example

```c
proctype application( chan in, out )
{
    int i=0, j=0, last.i=0;
do
    :: in?accept(i) -> assert( i==last.i );
    if
      :: (last.i!=MAX) -> last.i = last.i+1
      :: (last.i==MAX)
    fi
    :: out!next(j) ->
    if
      :: (j!=MAX) -> j=j+1
      :: (j==MAX)
    fi
    od
}
```
Spin simulation

This session

- Design principles
- Biometrics
- IPSec
- Formal methods
- Formal evaluation
- Exam
TCSEC (The Orange book) was the first rating system for the security of products. It defined six different evaluation classes. The classes are:

- **C1** - For same-level security access. Not currently used.
- **C2** - Controlled access protection - users are individually accountable for their actions. Most OS manufacturers have C2 versions of the OS.
- **B1** - Mandatory BLP policies - for more secure systems handling classified data.
- **B2** - structured protection - mandatory access control for all objects in the system. Formal models.
- **B3** - security domains - more controls, minimal complexity, provable consistency of model.
- **A1** - Verified design - consistency proofs between model and specification.
Formal evaluation - ITSEC

✔ From Dutch, English, French and German national security evaluation criteria.

✔ Adaptable.

✔ Sponsor determines operational requirements, threats and security objectives.

✔ ITSEC specifies the interactions and documents between the sponsor and the evaluator.

ITSEC

✔ Again there are various levels of evaluation: E0..E6, with E6 giving the highest level of assurance - it requires two independant formal verifications.

✔ First certification of a smart-card system under E6.
  ✔ The smart-cards are electronic purses - that is they carry value,
  ✔ Forgery must be impossible.
  ✔ The certification encompassed the communication with the card, as well as the software within the card, and at the bank.
This session

- Design principles
- Biometrics
- IPSec
- Formal methods
- Formal evaluation
- Exam

Exam

You can expect 12 pages - write on paper. Marks/50.

- 10 short answer questions worth 1 mark each
- Longer questions on...
  - Encryption
  - Information
  - Models
  - Key systems
Exam

✔ Modulo, primes, Fermat, Euler: general & specific
✔ Symmetric cryptosystems: IC, DES, general & specific
✔ Physical limits: general
✔ Information theory: general & security-specific
✔ Models: BLP, Biba - general & specific
✔ Key systems: RSA, Kerberos, specific

Real World Applications of Network/Computer Security

Abstract: The lecture is on practical applications of network and computer security technology. Examples include virtual private networks, security solutions for e-banking, fair exchange of digital valuables over the Internet (e.g., electronic contract signing over a network and certified e-mail delivery) and techniques for user privacy protection in cyberspace.
Finally

This is my last lecture, so...

✔ **Good luck** with the exam, and

✔ **Thanks** for your attention

    Good luck!