Important Terms

computers interconnected by communication network

= computer network (of the first type)

computers providing switching in communication network

= computer network (of the second type)

distributed system

  spatial
  control and implementation structure

open system ≠ public system ≠ open source system

service integrated system

digital system
threats: example: medical information system

1) unauthorized access to information
   computer company receives medical files

2) unauthorized modification of information
   undetected change of medication

3) unauthorized withholding of
   information or resources
   detected failure of system

no classification, but pragmatically useful
example: unauthorized modification of a program

protection goals:

confidentiality

integrity

availability for authorized users

\[ \geq \text{total correctness} \]
\[ \cong \text{partial correctness} \]

1) cannot be detected, but can be prevented; cannot be reversed
2)+3) cannot be prevented, but can be detected; can be reversed
Threats and corresponding protection goals

**threats:**

1) unauthorized access to information
   - example: medical information system
   - computer company receives medical files

2) unauthorized modification of information
   - undetected change of medication

3) unauthorized withholding of information or resources
   - detected failure of system

**protection goals:**

- confidentiality
- integrity
- availability for authorized users

no classification, but pragmatically useful

example: unauthorized modification of a program

1) cannot be detected, but can be prevented;
   cannot be reversed

2)+3) cannot be prevented, but can be detected;
   can be reversed
Definitions of the protection goals

**confidentiality**

Only **authorized users** get the **information**.

**integrity**

**Information** are **correct, complete, and current** or this is detectably not the case.

**availability**

**Information** and resources are accessible where and when the **authorized user** needs them.

- **subsume**: data, programs, hardware structure

- **it has to be clear, who is authorized to do what in which situation**

- **it can only refer to the inside of a system**
Protection against whom?

Laws and forces of nature
- components are growing old
- excess voltage (lightning, EMP)
- voltage loss
- flooding (storm tide, break of water pipe, heavy rain)
- change of temperature ...

Human beings
- outsider
- user of the system
- operator of the system
- service and maintenance
- producer of the system
- designer of the system
- producer of the tools to design and produce
- designer of the tools to design and produce
- producer of the tools to design and produce
  the tools to design and produce
- designer ...

fault tolerance

Trojan horse
- universal
- transitive

includes user, operator, service and maintenance ... of the system used
attacker model

It’s not possible to protect against an omnipotent attacker.

- roles of the attacker (outsider, user, operator, service and maintenance, producer, designer …), also combined
- area of physical control of the attacker
- behavior of the attacker
  - passive / active
  - observing / modifying (with regard to the agreed rules)
- stupid / intelligent
  - computing capacity:
    - not restricted: computationally unrestricted
    - restricted: computationally restricted
Observing vs. modifying attacker

observing attacker

modifying attacker

acting according to the agreed rules

possibly breaking the agreed rules
Strength of the attacker (model)

Attacker (model) $A$ is stronger than attacker (model) $B$, iff $A$ is stronger than $B$ in at least one respect and not weaker in any other respect.

Stronger means:
- set of roles of $A \supset$ set of roles of $B$,
- area of physical control of $A \supset$ area of physical control of $B$,
- behavior of the attacker
  - active is stronger than passive
  - modifying is stronger than observing
- intelligent is stronger than stupid
  - computing capacity: not restricted is stronger than restricted
- more money means stronger
- more time means stronger

Defines partial order of attacker (models).
Realistic protection goals/attacker models: Technical solution possible?
Security in computer networks

**confidentiality**
- message content is confidential
- place: sender / recipient anonymous

**integrity**
- detect forgery
- recipient can prove transmission
- sender can prove transmission
- ensure payment for service

**availability**
- enable communication

---

end-to-end encryption
mechanisms to protect traffic data

authentication system(s)
sign messages
receipt
during service by digital payment systems

diverse networks;
fair sharing of resources
Multilateral security

- Each party has its particular protection goals.
- Each party can formulate its protection goals.
- Security conflicts are recognized and compromises negotiated.
- Each party can enforce its protection goals within the agreed compromise.

Security with minimal assumptions about others
Multilateral security (2nd version)

- Each party has its particular goals.

- Each party can formulate its protection goals.

- Security conflicts are recognized and compromises negotiated.

- Each party can enforce its protection goals within the agreed compromise.

Security with minimal assumptions about others
Multilateral security (3rd version)

• Each party has its particular goals.

• Each party can formulate its protection goals.

• Security conflicts are recognized and compromises negotiated.

• Each party can enforce its protection goals within the agreed compromise. As far as limitations of this cannot be avoided, they equally apply to all parties.

Security with minimal assumptions about others
### Protection Goals: Sorting

<table>
<thead>
<tr>
<th>Content</th>
<th>Circumstances</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Prevent the unintended</strong></td>
<td><strong>Confidentiality</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Hiding</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Anonymity</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Unobservability</strong></td>
</tr>
<tr>
<td><strong>Achieve the intended</strong></td>
<td><strong>Integrity</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Accountability</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Availability</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Reachability</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Legal Enforceability</strong></td>
</tr>
</tbody>
</table>
Protection Goals: Definitions

Confidentiality ensures that nobody apart from the communicants can discover the content of the communication.

Hiding ensures the confidentiality of the transfer of confidential user data. This means that nobody apart from the communicants can discover the existence of confidential communication.

Anonymity ensures that a user can use a resource or service without disclosing his/her identity. Not even the communicants can discover the identity of each other.

Unobservability ensures that a user can use a resource or service without others being able to observe that the resource or service is being used. Parties not involved in the communication can observe neither the sending nor the receiving of messages.

Unlinkability ensures that an attacker cannot sufficiently distinguish whether two or more items of interest (subjects, messages, actions, …) are related or not.

Integrity ensures that modifications of communicated content (including the sender’s name, if one is provided) are detected by the recipient(s).

Accountability ensures that sender and recipients of information cannot successfully deny having sent or received the information. This means that communication takes place in a provable way.

Availability ensures that communicated messages are available when the user wants to use them.

Reachability ensures that a peer entity (user, machine, etc.) either can or cannot be contacted depending on user interests.

Legal enforceability ensures that a user can be held liable to fulfill his/her legal responsibilities within a reasonable period of time.
Transparency ensures that the data collection and data processing operations can be planned, reproduced, checked and evaluated with reasonable efforts.

Intervenability ensures that the user is able to exercise his or her entitled rights within a reasonable period of time.
Correlations between protection goals

Confidentiality

Hiding

Anonymity

Unobservability

Integrity

Availability

Accountability

Reachability

Legal Enforceability

implies

+ strengthens

– weakens
Correlations between protection goals

Confidentiality → Hiding

Anonymity ← Unobservability

Integrity ← Availability

Accountability ← Reachability ← Legal Enforceability

Transitive closure to be added

implies

+ strengthens

- weakens
Physical security assumptions

Each technical security measure needs a physical “anchoring” in a part of the system which the attacker has neither read access nor modifying access to.

Range from “computer centre X” to “smart card Y”

What can be expected at best?

**Availability** of a locally concentrated part of the system cannot be provided against *realistic* attackers

→ **physically distributed system**

... hope the attacker cannot be at many places at the same time.

Distribution makes **confidentiality** and **integrity** more difficult. But physical measures concerning confidentiality and integrity are more efficient: Protection against *all realistic* attackers seems feasible. If so, physical distribution is quite ok.
Key exchange using symmetric encryption systems

Key exchange centers

NSA:
- Key Escrow
- Key Recovery

participant A

$k_{AX}(k_1) k_{AY}(k_2) k_{AZ}(k_3)$

$k_{BX}(k_1) k_{BY}(k_2) k_{BZ}(k_3)$

key $k = k_1 + k_2 + k_3$

$k(messages)$

participant B
Needham-Schroeder-Protocol using Symmetric encryption

• from 1978

• goals:
  – key freshness:
    • key is „fresh“, i.e. a newly generated one
  – key authentication:
    • key is only known to Alice and Bob (and maybe some trusted third party)

• preconditions:
  – a trusted third party $T$
  – shared term secret keys between Alice (resp. Bob) and the trusted third party:
    • $k_{AT}$, $k_{BT}$
Needham-Schroeder-Protocol using Symmetric encryption

key exchange center

1. A, B, N_A

2. k_AT(N_A, B, k_AB, k_BT(k_AB, A))

3. k_BT(k_AB, A)

4. k_AB(N_B)

5. k_AB(N_B-1)

participant A

participant B

Problem:
- no key freshness / authentication for B, if old k_AB was compromised
- attack:
  - replay ③
  - decrypt ④
  - modify ⑤
Asymmetric encryption system

key generation
encryption
decryption
ciphertext
encryption key, publicly known
decryption key, kept secret
random number
Domain of trust
Area of attack
Opaque box with spring lock; 1 key

more detailed notation

Domain of trust

plaintext
$r'$ random number
$S := \text{enc}(c, x, r')$

$c(x)$
$c := \text{gen}(r)$
$(c, d) := \text{gen}(r)$

$r$
$S$
$\text{dec}(d, S) = \text{dec}(d, \text{enc}(c, x, r'))$
$x := \text{dec}(d, S)$
$x := \text{dec}(d, \text{enc}(c, x, r'))$
Needham-Schroeder-Protocol using Asymmetric encryption

• from 1978

• goals:
  – key freshness:
    • key is „fresh“, i.e. a newly generated one
  – key authentication:
    • key is only known to Alice and Bob

• preconditions:
  – public encryption keys of Alice $c_A$ and Bob $c_B$ known to each other
Needham-Schroeder-Protocol using Asymmetric encryption

\[ c_B(N_A, A) \]

\[ c_A(N_A, N_B) \]

\[ c_B(N_B) \]

\[ k_{AB} = \text{KDF}(N_A, N_B) \]

\[ k_{AB}(\text{messages}) \]

- Problem:
  - \( B \) does not know if he really talks to \( A \)
Attack on asymmetric Needham-Schroeder-Protocol

[Loewe 1996!]

\[ k_{AB} = \text{KDF}(N_A, N_B) \]

- **Solution:**
  - \( B \) has to include his identity in his message \( \circlearrowleft \)
Attack on asymmetric Needham-Schroeder-Protocol

Participant $A$ sends $c_M(N_A, A)$ to participant $M$.

Participant $M$ sends $c_A(N_A, N_B, B)$ to participant $A$.

Participant $B$ sends $c_B(N_A, A)$ to participant $M$.

Participant $M$ sends $c_A(N_A, N_B, B)$ to participant $B$.

Note:
- encryption has to be non-malleable
One-Time-Pad mod 4

participant A

c=m+k mod 4

participant B

c=11_2=3_{10}

• Problem:
  - invert last bit of plain-text

m=c-k mod 4

<table>
<thead>
<tr>
<th>possible Keys</th>
<th>Plain-text</th>
<th>manipulated Plain-text</th>
<th>manipulated Cipher-text</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>3=11_2</td>
<td>10_2=2</td>
<td>2=10_2</td>
</tr>
<tr>
<td>1</td>
<td>2=10_2</td>
<td>11_2=3</td>
<td>0=00_2</td>
</tr>
<tr>
<td>2</td>
<td>1=01_2</td>
<td>00_2=0</td>
<td>2=10_2</td>
</tr>
<tr>
<td>3</td>
<td>0=00_2</td>
<td>01_2=1</td>
<td>0=00_2</td>
</tr>
</tbody>
</table>

• Problem: k=3, c=2 \rightarrow m=3=11_2
Cipher Block Chaining (CBC)

All lines transmit as many characters as a block comprises

⊕ Addition mod appropriately chosen modulus
⊗ Subtraction mod appropriately chosen modulus

If error on the line: Resynchronization after 2 blocks, but block borders have to be recognizable

self synchronizing
Cipher Block Chaining (CBC) (2)

All lines transmit as many characters as a block comprises

- Addition mod appropriately chosen modulus
- Subtraction mod appropriately chosen modulus

1 modified plaintext bit
⇒ from there on completely different ciphertext

memory for ciphertext block $n-1$

memory for ciphertext block $n-1$

bit error

key

encryption

ciphertext block $n$

key

decryption

plaintext block $n$

useable for authentication ⇒ use last block as MAC

plaintext block $n+1$
CBC for authentication

- Key
- Memory for ciphertext block \( n-1 \)
- Encryption
- Memory for ciphertext block \( n \)
- Comparison
- Last block
- Plaintext block \( n \)
- Plaintext
Why CBC IV should be random?

\[ c = k(IV_A \oplus m) \]
\[ c_M = k(IV_M \oplus m_M) \]
\[ c_M = k(IV_M \oplus IV \oplus YES) \]
\[ c_M = k(IV_M \oplus IV_M \oplus IV_A \oplus YES) \]
\[ c_M = k(IV_A \oplus YES) \]

\[ c_M = c? \]
CBC for Confidentiality & Integrity

plaintext

CBC-Encryption & MAC-Generation (last block)

ciphertext, MAC

CBC-Decryption

CBC-MAC-Generation

MAC (last block)

comparison

ok?
Whole Disk Encryption – Requirements

• The data on the disk should remain confidential
• Manipulations on the data should be detectable
• Data retrieval and storage should both be fast operations, no matter where on the disk the data is stored.
• The encryption method should not waste disk space (i.e., the amount of storage used for encrypted data should not be significantly larger than the size of plaintext)

• Attacker model:
  – they can read the raw contents of the disk at any time
  – they can request the disk to encrypt and store arbitrary files of their choosing
  – and they can modify unused sectors on the disk and then request their decryption
Watermarking Attack on Whole Disk Encryption

- **Goal:** Detect stored files
- **Assumptions regarding Attacker:**
  - they can read the raw contents of the disk at any time
  - they can request the disk to encrypt and store arbitrary files of their choosing
- **Assumptions regarding Encryption & Storage:**
  - CBC (IV, k, m) $\rightarrow$ CBC (sector number, k, m)
  - Remember first block CBC: $\text{Enc}(k, m[0] \oplus \text{IV})$
  - (parts of) larger files stored at consecutive sectors
    - $\text{SN}_x$, $\text{SN}_{x+1}$, $\text{SN}_{x+2}$, ..., $\text{SN}_{x+y}$
    - where will be an $x$ where the $t$ least significant bits are all 0 and $y \geq 2^t$
      - $t=3 \rightarrow \text{SN}_x$: $\text{zzzzzz000}$, ..., $\text{SN}_{x+7}$: $\text{zzzzzz111}$
- **Attack:**
  - Create plaintext such that the first plaintext-blocks stored in each sector differ only in the LSB
Watermarking Attack on Whole Disk Encryption

Sector Number (IV) | Plaintext blocks stored in the sectors
--- | ---
zzz000 | bbbb$A_2$ ...
zzz001 | bbbb$\bar{A}_2$ ...
zzz010 | bbbb$A_2$ ...
zzz011 | bbbb$\bar{A}_2$ ...
zzz100 | bbbb$A_2$ ...
zzz101 | bbbb$\bar{A}_2$ ...
## Watermarking Attack on Whole Disk Encryption

<table>
<thead>
<tr>
<th>Sector Number (IV)</th>
<th>First Plaintext block stored in the sectors</th>
<th>First ciphertext block stored in the sectors</th>
</tr>
</thead>
<tbody>
<tr>
<td>zzz000</td>
<td>$\text{bbbb}A_2$</td>
<td>$\text{Enc(}\text{bbbb}A_2 \oplus \text{zzz000}) = \text{Enc(}p_1) = c_1$</td>
</tr>
<tr>
<td>zzz001</td>
<td>$\text{bbbb}\bar{A}_2$</td>
<td>$\text{Enc(}\text{bbbb}\bar{A}_2 \oplus \text{zzz001}) = \text{Enc(}p_1) = c_1$</td>
</tr>
<tr>
<td>zzz010</td>
<td>$\text{bbbb}A_2$</td>
<td>$\text{Enc(}\text{bbbb}A_2 \oplus \text{zzz010}) = \text{Enc(}p_2) = c_2$</td>
</tr>
<tr>
<td>zzz011</td>
<td>$\text{bbbb}\bar{A}_2$</td>
<td>$\text{Enc(}\text{bbbb}\bar{A}_2 \oplus \text{zzz011}) = \text{Enc(}p_2) = c_2$</td>
</tr>
<tr>
<td>zzz100</td>
<td>$\text{bbbb}A_2$</td>
<td>$\text{Enc(}\text{bbbb}A_2 \oplus \text{zzz100}) = \text{Enc(}p_3) = c_3$</td>
</tr>
<tr>
<td>zzz101</td>
<td>$\text{bbbb}\bar{A}_2$</td>
<td>$\text{Enc(}\text{bbbb}\bar{A}_2 \oplus \text{zzz101}) = \text{Enc(}p_3) = c_3$</td>
</tr>
</tbody>
</table>
Watermarking Attack on Whole Disk Encryption

<table>
<thead>
<tr>
<th>Sector Number (IV)</th>
<th>First Plaintext block stored in the sectors</th>
<th>First ciphertext block stored in the sectors</th>
</tr>
</thead>
<tbody>
<tr>
<td>zzz000</td>
<td>bbbbA₂</td>
<td>c₁</td>
</tr>
<tr>
<td>zzz001</td>
<td>bbbbĀ₂</td>
<td>c₁</td>
</tr>
<tr>
<td>zzz010</td>
<td>bbbbA₂</td>
<td>c₂</td>
</tr>
<tr>
<td>zzz011</td>
<td>bbbbĀ₂</td>
<td>c₂</td>
</tr>
<tr>
<td>zzz100</td>
<td>bbbbA₂</td>
<td>c₃</td>
</tr>
<tr>
<td>zzz101</td>
<td>bbbbĀ₂</td>
<td>c₃</td>
</tr>
</tbody>
</table>

Watermark!
Watermarking Attack on Whole Disk Encryption

• Solution: unpredictable IVs

• Construction:
  – Encrypted salt-sector initialization vector (ESSIV)
  – \( IV(SN) = Enc(\ Hash(k), SN ) \)
Probability Exercise

Message $m$
- $P(\text{"Yes"}) = 0.7$
- $P(\text{"No"}) = 0.3$

Key $k$
- $P(0) = 0.4$
- $P(1) = 0.6$

$c = k \oplus m$

Eve

Bob

$P(\text{"Yes"} | c=1) =$?
Message $m$
- $P(\text{“Yes”}) = 0.7$
- $P(\text{“No”}) = 0.3$

Key $k$
- $P(0) = 0.4$
- $P(1) = 0.6$

Eve

Bob

$c = k \oplus m$

<table>
<thead>
<tr>
<th>$m$</th>
<th>$k=0$</th>
<th>$k=1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>$c=1$</td>
<td>$c=0$</td>
</tr>
<tr>
<td>No</td>
<td>$c=0$</td>
<td>$c=1$</td>
</tr>
</tbody>
</table>

- For $m=\text{“Yes”}$, $P(\text{“Yes”} | c=1) = \frac{0.28}{0.28 + 0.18} = \frac{0.28}{0.46} \approx 0.61$
- For $m=\text{“No”}$, $P(\text{“Yes”} | c=1) = \frac{0.12}{0.12 + 0.18} = \frac{0.12}{0.30} = 0.40$

$\sum P = 1.0$
Probability Exercise

- Remember: \( P(A_2 | A_1) = \frac{P(A_1 \cap A_2)}{P(A_1)} \)

- \( P(m|c) = \frac{P(c \cap m)}{P(c)} \)

- \( P(c=0) = P(\text{"Yes"}) \cdot P(k=1) + P(\text{"No"}) \cdot P(k=0) = 0.54 \)

- \( P(c=0 \text{ and } m=\text{"Yes"}) = P(\text{"yes"}) \cdot P(k=1) = 0.42 \)

- \( P(\text{"Yes"} | c=0) \approx 0.77 \quad P(\text{"Yes"} | c=1) \approx 0.61 \)
One-way functions – cryptographic hash functions

**One-way function** $f$:
- calculating $f(x)=y$ is easy
- calculating $f^{-1}(y)=x$ is hard
  - computation / storage
- open question: Do one-way functions exist?

**Cryptographic hash function** $h$
- might have different properties depending on the use case
  - collision resistance:
    - it is hard to find $x, y$ with $h(y)=h(x)$ and $y\neq x$
    - note: $h$ is usually not collision free, because $|h(x)| \ll |x|$
  - preimage resistance / one-way function / secrecy
    - given $h(x)$ it is hard to find $x$
  - second-preimage resistance / weak collision resistance / binding
    - given $x$, $h(x)$ it is hard to find $y$ with $h(y)=h(x)$ and $y\neq x$
- Note:
  - $h$ is not necessarily a “random extractor”
  - only one of “secrecy” and “binding” can be information theoretic secure
Symmetric authentication system

Show-case with lock; 2 identical keys

more detailed notation

Domain of trust

secret area

Domain of trust

random number \( r \)

\( k := \text{gen}(r) \)

key generation \( \text{gen} \)

secret key

Encode:

plaintext \( x \)

\( \text{MAC} := \text{code}(k, x) \)

plaintext with authenticator \( x, k(x) \)

\( \text{MAC} = \text{code}(k, x) \)

test:

\( \text{MAC} = \text{code}(k, x) \) ?

"pass" or "fail"

Area of attack

Domain of trust

Area of attack
Calculating with and without $p,q$ (2)

$Z_n^*$: multiplicative group

$a \in Z_n^* \iff \gcd (a,n) = 1$

- Inverting is fast (extended Euclidean Algorithm)
  - Determine to $a,n$ the values $u,v$ with
    \[ a \cdot u + n \cdot v = 1 \]
  - Then: $u \equiv a^{-1} \mod n$

Example: $3^{-1} \mod 11$ ?

\[
\begin{align*}
11 &= 3 \cdot 3 + 2 \\
3 &= 1 \cdot 2 + 1
\end{align*}
\]

\[
\begin{align*}
11 &= -11 + 4 \cdot 3 \\
1 &= 1 \cdot 3 - 1 \cdot (11 - 3 \cdot 3)
\end{align*}
\]

$\Rightarrow 3^{-1} \equiv 4 \mod 11$
Visual Cryptography Scheme by Naor and Shamir (simplified)

<table>
<thead>
<tr>
<th>Plaintext-Pixel</th>
<th>Key</th>
<th>Ciphertext</th>
<th>Superposition</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Case 1</td>
<td>Pixelblock Slide 1</td>
<td>Pixelblock Slide 2</td>
</tr>
<tr>
<td></td>
<td>Case 2</td>
<td>Pixelblock Slide 1</td>
<td>Pixelblock Slide 2</td>
</tr>
<tr>
<td></td>
<td>Case 1</td>
<td>Pixelblock Slide 1</td>
<td>Pixelblock Slide 2</td>
</tr>
<tr>
<td></td>
<td>Case 2</td>
<td>Pixelblock Slide 1</td>
<td>Pixelblock Slide 2</td>
</tr>
</tbody>
</table>
Hallo
ausprobieren:  http://www-sec.uni-regensburg.de/vc/
Plausible Deniability

cipher text

Key 1

Key 2
Do not reuse keys!

Key

Ciphertext 1

Ciphertext 2