Part A: Block Ciphers

1. Introduction
   - DES
   - AES
   - Modes of operation & security proofs

2. Differential cryptanalysis
   - Basics
   - Design theories

3. Differential cryptanalysis in practice

4. Linear cryptanalysis, variations on differential cryptanalysis

The setting

Using cryptography

Principles
- Kerckhoffs’ principle:
  Algorithm is public, except for 1 parameter: the key
- Key generation, distribution, management:
  - Different problem

Goals of Cryptography
- Confidentiality
- Integrity
- Authentication
- Anonymity
- Non-repudiation (origin, delivery)
- Time stamping
- Key escrow
### Symmetric cryptography
- Sender and receiver use the same key
  - Or keys that can easily be derived from one another
- Sender and receiver are equivalent
- By far the oldest type of cryptography
- Best performance
- Highest security standards
- Only disadvantage: difficult key management

### Practical cryptography
- Short key is used to encrypt long messages
- Perfect secrecy is not possible
- Complexity-theoretic security
  - No satisfactory results thus far
- Practical security
  - Resistance against cryptanalysis
  - "Human ignorance" model

### Academic attacks and real attacks
- Academic attack = primitive behaves suboptimal
- Real attack: can be broken in practice

- Example:
  1. Encryption algorithm with 40-bit key
     - Best attack is to try out all $2^{40}$ keys
     - Practical attack
  2. Encryption algorithm with 256-bit key
     - Key can be recovered with a method that has a complexity equivalent to $2^{200}$ encryptions
     - Academic attack

### Assumptions on the attacker
- Ciphertext-only attack
  - Most modern encryption systems are resistant
- Known-plaintext attack
  - Known headers, formatting, …
  - Can be statistical information
- Chosen-plaintext attack
  - Surprisingly, often quite realistic
- Related-key attack

### Simple substitution cipher
- Permutation of the alphabet

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>...</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q</td>
<td>W</td>
<td>E</td>
<td>...</td>
<td>M</td>
</tr>
</tbody>
</table>
- $26!$ possibilities (keys)
- Frequency-analysis

### Advanced substitution cipher
- Permutation on block of characters

<table>
<thead>
<tr>
<th>AAAA</th>
<th>AAAB</th>
<th>AAAC</th>
<th>...</th>
<th>ZZZZ</th>
</tr>
</thead>
<tbody>
<tr>
<td>QAQZ</td>
<td>WIJT</td>
<td>ENTO</td>
<td>...</td>
<td>MIHB</td>
</tr>
</tbody>
</table>
- "code book"
- Even more keys
- If blocks large enough, then frequency analysis impossible (infeasible)
Block cipher

- Avoid transport & storage of huge table
- Introduce computation rule to compute table elements:
  \[ T[X] = f(X, \text{key}) \]
- Design 'good' rule \( f \):
  - Secure
  - Efficient

Block cipher formally

- Family of permutations
- Every value of the key selects one permutation
- Block length \( n \): \( 2^n \approx 2^{(n-1)}2^n \) permutations
- Key length \( k \): \( 2^k \) selectable permutations
- Library of code books

Shannon's view on block cipher security

- Short key
  - Conditional security
- Key determined by equations
  - Derived from message \( X \), ciphertext \( Y \), algorithm \( B \)
  - Should be difficult to solve
- Without key, impossible (infeasible) to
  - Decrypt (encrypt)
  - Derive statistical information about the message

Shannon's principles

1. Confusion: equations in the key should be
   - Complicated (non-linear)
   - Involve many variables
2. Diffusion: redundancy in message should be dissipated over large structures in ciphertext

Design principles

- Shannon: product ciphers
  \[ B = T \circ M \circ S \]
  - \( M \): mixing transformation (known)
  - \( S, T \): simple substitution ciphers (keyed)
- Iterative ciphers:
  \[ B = S_1 \circ M \circ S_2 \circ M \circ S_3 \circ \ldots \circ M \circ S_r \]
  - Round transformation, round: \( (S_i \circ M) \)
  - Often: \( S_i = S \circ \text{AddKey} \)

AddKey: key addition

- Injection of key material
  - Addition of key to intermediate variable
  - Use of key-dependent transformations
- Key schedule
  - Input: cipher key
  - Output: round keys
Iterative cipher

What is nonlinearity?

- Distance to linear functions
  - $\mathcal{D}$ = how difficult to approximate by a linear function
  - $\neq$ nonlinear degree

- Example:
  - $f(a,b,c,d) = abcd$
    - $abcd \approx 0$
    - Nonlinearity($f$) = $d(f,0) = 1/16$
  - $g(a,b,c,d) = ab + cd$
    - Nonlinearity($g$) = $6/16$

Importance of nonlinearity

- Linear cryptanalysis
- Linear approximations of the cipher
- Differential cryptanalysis
- Non-uniformity of first order derivative

Mixing

- Boolean equations in a small number of variables are always easy to solve
- Mixing needs to ensure strong dependencies between sub-systems
- Easiest to measure for linear transformations (usually)

Practical constraints

- Hardware/software
- Key agility

- Typically
  - Small substitution elements
  - Mixing by means of interconnection
Feistel ciphers and SP-networks

- Round transformation is an involution
- Encryption and decryption only differ in the order of the round keys
  - Saves hardware area/code size

Feistel encryption

\[
\begin{align*}
L &\xrightarrow{F} R \\
R &\xrightarrow{K_1} L+F(R,K_1) \\
F &\xrightarrow{K_2} L+F(R,K_1) \\
\end{align*}
\]

Feistel decryption

\[
\begin{align*}
R &\xrightarrow{F} L+F(R,K_1) \\
L &\xrightarrow{K_1} R+F(L+F(R,K_1), K_2) \\
F &\xrightarrow{K_2} R+F(L+F(R,K_1), K_2) \\
\end{align*}
\]

Block cipher research

- Majority of designs uses Feistel structure or uniform structure
- Designs concentrate on selection of nonlinear elements
  - Small elements to reduce cost
  - Connection

Data Encryption Standard (1977)

- 1970: need for a commercial-grade encryption standard
- 1973-1977: Development of a block cipher DES
  - IBM together with NBS
- Encrypts blocks of 64 bits
- Effective key length of 56 bits
- Structure:
  - Initial bit shuffle
  - 16 iterations of a round transformation (Feistel)
  - Inverse bit shuffle
The DES round function

S-box 1

S-box design criteria
- Surrounded with mystery (“No need to know”)
- Apparently, largest S-box that would make DES fit on a single chip (in 1974)
- S-box input bits
  - 2 row selection bits, 4 column selection bits
  - 2 middle bits, 2 times 2 end bits
- Every row is a permutation
- End bits are shared between neighbouring S-boxes

Bit permutation P criteria
1. For every S-box, two outputs go to middle input bits, and two outputs go to end bits
2. Outputs of every S-box affect 6 S-boxes
3. If output of one S-box affects middle of another S-box, then not vice versa

Rise of the DES
- Design criteria classified
  - Design rationale remained unclear until 1990
- Modifications by NSA
  - Trapdoors?
- Short key length
  - Exhaustive key search
- World-wide adoption: the only commercial standard
- Also used for data authentication mechanisms
Fall of the DES

- Designed for 1970 technology
- No use of nifty processor features
- 1991, 1993: academic attacks + design of a DES cracker machine
- 1998: exhaustive key search performed in practice (EFF)
- Temporary solution: 3-DES

Multiple encryption

- DES is not a group:
  - In general, we can’t find a $k_3 = f(k_1, k_2)$ such that $E_{k_2}(E_{k_1}(x)) = E_{k_3}(x)$
- Hence, multiple encryption is not equivalent to single encryption
  - Can be used to increase the key space
  - Double encryption is not sufficient

Block ciphers and cryptographic hash functions

3-DES: triple encryption

- E-E-E or E-D-E
  - E-D-E easier for backwards compatibility
  - Triple key or double key: $E_{k_1}(E_{k_2}(E_{k_1}(x)))$
    - Triple key offers more practical security
  - Slow
- Alternative: XDES (‘triple-key DES’)
  - $y = k_3 + E_{k_2}(k_1 + x)$

Advanced Encryption Standard

- 1997: public call for submission
- Encrypt blocks of 128 bits
- Key of lengths 128, 192, 256
- To be available royalty-free
- August 1998: first AES conference

Public evaluation

- Only public comments taken into account
- Decisions by NIST, motivated by public reports
- Most analysis done by the public
- NSA had the right to veto NIST’s decision
Evaluation criteria

- Security
- Efficiency
- Intellectual Property issues
- Flexibility
- Elegance, ability to prove absence of trapdoors, ...

Design trade-off

- Luke O’Connor (IBM):
  “Most ciphers are secure after sufficiently many rounds”

- James L. Massey (ETH Zuerich):
  “Most ciphers are too slow after sufficiently many rounds”

Science or Engineering?

- Practical security can be achieved easily if we don’t worry about performance

- It is not sufficient to prove that a secure block cipher exists
  - We have to construct it

- Design challenge:
  - security AND performance
  - provability

Rijndael

- Based on the dissertations of Joan Daemen (1995) and Vincent Rijmen (1997)

- Not a Feistel cipher (finally!)

- Influenced by experience with chip card based practical systems

Rijndael: Iterated Block Cipher

- 10/12/14 times applying the same round transformation
- Uniform round transformation
- Composed of 4 steps, each its own purpose:
  - SubBytes: non-linearity
  - ShiftRows: inter-column diffusion
  - MixColumns: inter-byte diffusion within columns
  - AddRoundKey

Round step 1: SubBytes

- Bytes are transformed by invertible S-box.
- One S-box (lookup table) for complete cipher:
  - High non-linearity: multiplicative inverse in GF(2^8)
  - Complex algebraic expression: additional linear transformation
Round step 3: MixColumns

- Columns transformed by matrix over GF(2^8)
- High intra-column diffusion:
  - based on theory of error-correcting (MDS) codes

Round step 2: ShiftRows

- Rows are shifted over 4 different offsets
- High diffusion over multiple rounds:
  - Interaction with MixColumns
  - Bits flip in minimum 25 active S-boxes per 4 rounds

Round step 4: Key addition

- Makes round function key-dependent
- Round keys derived in a simple way from the master key

Modes of operation

- How to encrypt data that is not exactly one block?
  - Integer number of blocks
  - Fractions of blocks
- Using block ciphers for other goals than encryption
  - MACing
  - Hashing
- Consequence of popularity of the DES

Electronic Code Book
Problem of ECB

Properties

- Patterns are hidden
- Even repeated encryption of the same message not detectable (by changing IV)
- Last ciphertext block depends on all plaintext blocks
- Not true for decryption direction: each plaintext block depends on only two ciphertext blocks
- Favourite encryption mode (definitely in the past)

Birthday attack

- Encrypt $2^{nd/2}$ blocks under the same key
- With high probability:
  $\exists i, j$ such that $C_i = C_j$
  $C_{i+1} \oplus P_i = C_{j+1} \oplus P_j$
  $P_i \oplus P_j = C_{i+1} \oplus C_{j+1}$
- Information on plaintext revealed
- Encrypting slightly more blocks leads to many more collisions
- Main reason why AES has block length 128

Counter Mode

Properties

- Counter should start at values sufficiently far away from one another
  - Never same inputs to block cipher
- Parallel
  - Pipelining
  - Random access (hard disks)
- Block cipher is used to build a stream cipher
Message Authentication Code (MAC)

- Cryptographic check sum
- Allows to detect malicious modifications to messages
- Sender and receiver use the same key
  - Not a digital signature

Authenticated encryption modes

- Combine encryption and authentication
- Less errors
  - Order of encryption and authentication
  - Different keys or the same
- Faster
  - One pass over the data
  - Not true for unpatented schemes
- Security proofs

Security proofs for modes

- Concrete
  - For one or more given block ciphers
- Standard model
  - Block cipher is a Pseudo-Random Permutation (PRP)
- Random Oracle Model – Ideal cipher model

CBC-MAC (Simple MAC, S-MAC)

- Block ciphers and cryptographic hash functions

Authenticated encryption modes

- Combine encryption and authentication
- Less errors
- Faster
- Security proofs

Security proofs

- ‘But that’s not security,’ said Alice, ‘security means something else.’
- ‘Security means what I choose it to mean,’ said the queen.

Alice in Wonderland

Pseudo-Random Permutation (PRP)

- Function indistinguishable from random permutation
- There are $2^n$ permutations from n bits to n bits
- Denote by $R$ the set of all n-bit permutations
- Random permutation: randomly selected element of $R$

Further definition:
- Oracle: black box: for each input, it gives the output of the function it implements
**Distinguishing**

Game: for \( r \in \mathbb{R}, f \in \mathbb{F} \)

When given two oracles, one for \( r \), one for \( f \)

Say which is which

- Average probability of success – 0.5 = Advantage
- Advantage depends on
  - Number of oracle accesses (queries)
  - Computational power (usually: not limited)
  - Size \( n \)

**Indistinguishable**

- We look at what happens when \( n \) grows
- Advantage = \( f(q,n) \)
- A primitive is called *indistinguishable from random* if
  - \( f \) decreases as an exponential function of \( n \)
  - Even if \( q \) grows as a polynomial function of \( n \)

**Block cipher as Pseudo-Random Permutation**

- Block cipher is family of permutations
  - One for each key
- We know constructions to build block ciphers that are PRPs
  - Luby-Rackoff
- Security proofs for applications: if the block cipher is a PRP, then ...

**Luby-Rackoff construction**

If \( f_1, f_2, f_3, f_4 \) are pseudo-random functions, then this is a PRP

Note that we can't really build this in practice

**PRP**

- A PRP can have:
  - Weak keys
  - Equivalent keys
  - Output the key upon receipt of a special plaintext

Because the model considers only the ‘average case’

(On average, pedestrians walk in the middle of the road)

- A PRP can further have
  - Weaknesses only apparent if you consider more keys (related keys)

Because the model doesn’t consider this

**Ideal Cipher Model**

- The attacker is not allowed to look at the block cipher
- Should help to concentrate on the security of the mode

**Argument pro**

- Allows to prove security where the standard model doesn’t
  - Block cipher based hash function
  - Anything where key input is not random

**Argument contra**

- ‘.prove security’ means here: define security as the property that you can prove
Use of security proofs

- Definitely, don’t use a mode of operation proven insecure
- Is it better to have a proof of security than to have no proof?
  - Yes, if everything else is equal
- We don’t know how to build block ciphers that can be proven to be PRP, are efficient and use a short key
- There is no idea how to measure whether a block cipher is close to ideal

Secure mode of operation

- Submit $q$ queries of length $n$, $2n$, $3n$, ...
- Try to distinguish
  - Mode M with block cipher replaced by ideal cipher
  - Large ideal cipher (with variable block length)
- Advantage = $f(q,n)$
  - $f$ decreases as an exponential function of $n$
  - Even if $q$ grows as a polynomial function of $n$

ECB is insecure

- Submit $(P,P)$
- Oracle answers $(C_1, C_2)$
- For ECB: $C_1 = C_2$ always
- For ideal cipher with block length $2n$:
  - $C_1 \neq C_2$ with probability $1 - 2^{-n}$

CBC is secure

- But need to use a new, unpredictable IV every time
- Otherwise, submit $P_1$ and $(P_1, P_2)$
- What about the birthday attack?
  - $q$ grows exponentially
  - Not allowed

CBC-MAC is secure

- But only if all messages have the same length!
- Let $T_1 = \text{MAC}(X_1)$, $T_2 = \text{MAC}(T_1)$
- Then $\text{MAC}(X_1, 0) = T_2$
- (Can be fixed easily)

Offset Code Book (OCB)
OCB start and stop

- **Whitening values** $Z_i$
  - $\gamma_i$: gray code counter
  - $Z_i = \gamma_i \times E[0] + E[\text{Nonce} + E[0]]$

- **Final values (tags)**
  - $C_{n+1} = E[\text{Length}(P) \times x^i \times Z_{n+1}] + \text{Length}(P)$
  - $C_{n+2} = E[\sum_i P_i + Z_{n+1}]$

- **Provably secure against**
  - Distinguishing attacks
  - Forgery attacks

Tweakable block cipher

- **Idea**: introduce additional variability: the tweak parameter
- **Known to the attacker**

\[
\begin{align*}
P & \to B \to C \\
K \to B & \to B# \\
T \to C
\end{align*}
\]

Provable security

- If a secure tweakable block cipher exists, then also a secure block cipher exists (obviously)
- If a secure block cipher exists, then also constructions for secure tweakable block ciphers exist

- **Tweakable block ciphers simplify (proofs of) modes**
  - OCB is close to ECB with tweakable block cipher

Conclusion

- **Practical block ciphers**, DES, AES
- **Shannon’s ideas on practical designs**
- **Modes of operation**
- **(Security) proofs**