Symmetric Cryptography

René Rydhof Hansen

Informatics and Mathematical Modelling
Technical University of Denmark

E05-02230

Today

Goals

- Designing a Crypto-system
- The one-time pad
- The DES
- Modes of Operation
- Stream vs. Block Ciphers

Reading

[PP] 2.1–2.6, 10.1–10.2

Cryptography—The Algebraic Perspective

Symmetric

- *One* key: K
- Encrypt: $E_K(M)$ / Decrypt: $D_K(M)$
- Such that:

Asymmetric

- *Pair* of keys (public/private): K^+ and K^-
- Encrypt: / Decrypt:
- Such that: $D_{K^-}(E_{K^+}(M)) = M$

The Rest

- Signature (sign/verify: $V_{K^+}(S_{K^-}(M)) = M$)
- Hash functions: $H(M)$

Cryptography—The Algebraic Perspective

Symmetric

- *One* key: K
- Encrypt: $E_K(M)$ / Decrypt: $D_K(M)$
- Such that: $D_K(E_K(M)) = M$

Asymmetric

- *Pair* of keys (public/private): K^+ and K^-
- Encrypt: / Decrypt:
- Such that: $D_{K^-}(E_{K^+}(M)) = M$

The Rest

- Signature (sign/verify: $V_{K^+}(S_{K^-}(M)) = M$)
- Hash functions: $H(M)$

Cryptography—The Algebraic Perspective

Symmetric

- *One* key: K
- Encrypt: $E_K(M)$ / Decrypt: $D_K(M)$
- Such that: $D_K(E_K(M)) = M$

Asymmetric

- *Pair* of keys (public/private): K^+ and K^-
- Encrypt: $E_{K^+}(M)$ / Decrypt: $D_{K^-}(M)$
- Such that: $D_{K^-}(E_{K^+}(M)) = M$

The Rest

- Signature (sign/verify: $V_{K^+}(S_{K^-}(M)) = M$)
- Hash functions: $H(M)$

Designing a Crypto-System

Kerckhoffs' Design Principles (ca. 1883)

1. The system must be practically, if not mathematically, indecipherable;
2. It must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience;
3. Its key must be communicable and retainable without the help of written notes, and changeable or modifiable at the will of the correspondents;
4. It must be applicable to telegraphic correspondence;
5. It must be portable, and its usage and function must not require the concourse of several people;
6. Finally, it is necessary, given the circumstances that command its application, that the system be easy to use, requiring neither mental strain nor the knowledge of a long series of rules to observe.

Designing a Crypto-System

Kerckhoffs' Design Principles (ca. 1883)

1. The system must be practically, if not mathematically, indecipherable;
2. It must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience;
3. Its key must be communicable and retainable without the help of written notes, and changeable or modifiable at the will of the correspondents;
4. It must be applicable to telegraphic correspondence;
5. It must be portable, and its usage and function must not require the concourse of several people;
6. Finally, it is necessary, given the circumstances that command its application, that the system be easy to use, requiring neither mental strain nor the knowledge of a long series of rules to observe.

Modes of Attack

Kerckhoffs' Law

Attacker *always* knows encryption algorithm: $E\_(\_)$

Attacking the ciphertext: $E_K(M)$

- Ciphertext only: $E_K(M)$
- Known plaintext: $(M_1, E_K(M_1)), \ldots, (M_n, E_K(M_n))$
- Chosen plaintext: $(M_1, E_K(M_1)), \ldots, (M_n, E_K(M_n))$ (M_i's chosen by attacker)
- Chosen ciphertext: $(M_1, E_K(M_1)), \ldots, (M_n, E_K(M_n))$ ($E_K(M_i)$'s chosen by attacker)
- Chosen text: $E_K(M)$, $(M_1, E_K(M_1)), \ldots, (M_n, E_K(M_n))$ (M_i's and $E_K(M_i)$'s chosen by attacker)

Substitution Ciphers

Definition (Substitution Cipher)

Substituting a character (or symbol) for each character of the plaintext message.

Example (Caesar Cipher)

Shift alphabet three (3) places: $A \mapsto d$, $B \mapsto e$, ...

Key:	3												
Plaintext:	D	A	T	A	S	I	K	K	E	R	H	E	D
Ciphertext:													

Example (A Very Simple Cipher)

Key:	123												
Plaintext:	D	A	T	A	S	I	K	K	E	R	H	E	D
Ciphertext:													

Substitution Ciphers

Definition (Substitution Cipher)

Substituting a character (or symbol) for each character of the plaintext message.

Example (Caesar Cipher)

Shift alphabet three (3) places: $A \mapsto d$, $B \mapsto e$, ...

Key:	3												
Plaintext:	D	A	T	A	S	I	K	K	E	R	H	E	D
Ciphertext:	g	d	w	d	v	l	n	n	h	u	k	h	g

Example (A Very Simple Cipher)

Key:	123												
Plaintext:	D	A	T	A	S	I	K	K	E	R	H	E	D
Ciphertext:													

Substitution Ciphers

Definition (Substitution Cipher)

Substituting a character (or symbol) for each character of the plaintext message.

Example (Caesar Cipher)

Shift alphabet three (3) places: $A \mapsto d$, $B \mapsto e$, ...

Key:	3											
Plaintext:	D	A	T	A	S	I	K	K	E	R	H	E	D
Ciphertext:	g	d	w	d	v	l	n	n	h	u	k	h	g

Example (A Very Simple Cipher)

Key:	123											
Plaintext:	D	A	T	A	S	I	K	K	E	R	H	E	D
Ciphertext:	e	c	x	b	u	l	l	m	h	s	j	h	e

Substitution Ciphers

The One-Time Pad

- Key as long as message
- Key must be *absolutely* random
- Keys must *never* be re-used (cf. Venona)
- Guarantees perfect security
- Key management *very* hard

Computers and One-Time Pads

Computer generated "random" number sequences are (usually) *not* good enough

Transpositions

Definition

Rearranging the characters of a message.

Example

Key: 3

Plaintext: D A T A S I K K E R H E D

Transposed:

Ciphertext:

Transpositions

Definition

Rearranging the characters of a message.

Example

Key:	3												
Plaintext:	D	A	T	A	S	I	K	K	E	R	H	E	D
Transposed:	D	A	K	R	D								
	A	S	K	H									
	T	I	E	E									
Ciphertext:	D	A	K	R	D	A	S	K	H	T	I	E	E

Combining Substitution and Transposition: Product Ciphers*

Good Cipher = Confusion + Diffusion

- *Confusion*: Message not readily recognisable, interceptor unable to predict change in ciphertext from given change in plaintext
- *Diffusion*: Information from plaintext should be spread all over the ciphertext, change in plaintext implies many changes in ciphertext

Product Cipher

- Combining substitution with transposition
- Substitution $\approx$?
- Transpositions $\approx$?

Combining Substitution and Transposition: Product Ciphers*

Good Cipher = Confusion + Diffusion

- *Confusion*: Message not readily recognisable, interceptor unable to predict change in ciphertext from given change in plaintext
- *Diffusion*: Information from plaintext should be spread all over the ciphertext, change in plaintext implies many changes in ciphertext

Product Cipher

- Combining substitution with transposition
- Substitution $\approx$ confusion
- Transpositions $\approx$ diffusion

The Data Encryption Standard (DES)*

Overview

- Created by IBM (and NSA) in the 1970's
- A product cipher using *substitution* and *permutation*
- 16 cycles (repetitions)
- Keys are 64 bit (only 56 bits are used)
- Works on words of size 64 bit
- Uses only standard arithmetic and logical operations
  - Easy to implement efficiently
- Superseded by the AES (Rijndael)

Details of DES

Initialisation

- Split input into 64 bit blocks
- Reduce key from 64 to 56 bits
- *Initial Permutation*
- Begin cycles

Cycle*

- Break 64 bits into two halves
- Shift and permute the key
- Apply key-driven substitution (S-box) and permutation to right half
- Combine transformed right half with left half (= new right half)
- Right half = new left half
- Repeat 16 times

Details of DES

A Cycle

- Expansion (32 to 48 bits)
- Combination with sub-key
- Substitution, permuted choice (S-box)
- Permutation and combination

Details of DES

Substitution by S-box*

- Substitution is table-driven
- Substitutes 6 bits $b_1 b_2 b_3 b_4 b_5 b_6$ with 4 bits $b'_1 b'_2 b'_3 b'_4$
- Bits $b_1 b_6$ is used to look up row
- Bits $b_2 b_3 b_4 b_5$ used to look up column
- 8 S-boxes

Example

- Consider 010011
- Row $= 01 = 1$
- Column $= 1001 = 9$

	...	8	9	10	...
0	...	3	10	6	...
1	...	10	6	12	...
2	...	15	12	9	...
3	...	5	11	3	...

Details of DES

Substitution by S-box*

- Substitution is table-driven
- Substitutes 6 bits $b_1 b_2 b_3 b_4 b_5 b_6$ with 4 bits $b'_1 b'_2 b'_3 b'_4$
- Bits $b_1 b_6$ is used to look up row
- Bits $b_2 b_3 b_4 b_5$ used to look up column
- 8 S-boxes

Example

- Consider 010011
- Row $= 01 = 1$
- Column $= 1001 = 9$

	...	8	9	10	...
0	...	3	10	6	...
1	...	10	6	12	...
2	...	15	12	9	...
3	...	5	11	3	...

Details of DES

Substitution by S-box*

- Substitution is table-driven
- Substitutes 6 bits $b_1 b_2 b_3 b_4 b_5 b_6$ with 4 bits $b'_1 b'_2 b'_3 b'_4$
- Bits $b_1 b_6$ is used to look up row
- Bits $b_2 b_3 b_4 b_5$ used to look up column
- 8 S-boxes

Example

- Consider 010011
- Row $= 01 = 1$
- Column $= 1001 = 9$

	...	8	9	10	...
0	...	3	10	6	...
1	...	10	6	12	...
2	...	15	12	9	...
3	...	5	11	3	...

Weaknesses of DES

Things to look out for

- Complements (one's complement):
  $C = E_K(M) \quad \Rightarrow \quad \neg C = E_{\neg K}(\neg M)$
- Weak keys: $C = E_K(M) \qquad \text{and} \qquad M = E_K(C)$
- Semiweak keys: same as weak keys, but for specific key pairs
- Design weaknesses: S-boxes may leak information
- Key clustering: $E_{K_1}(M) = C = E_{K_2}(M)$

Attacking DES

No fatal flaws... but possible to brute force (1998: 112 hours, \$130.000)

3DES

- Use DES three times with different keys

$$E^{3\mathrm{DES}}_{k_0,k_1,k_2}(M) = E_{k_0}(D_{k_1}(E_{k_2}(M)))$$

- Provably stronger than DES
- Backwards compatible with DES (when $k_0 = k_1 = k_2$)
- Why not 2DES?
  - No more secure than DES

Modern Algorithms

Advanced Encryption Standard (AES)

- Chosen as DES' replacement
- Evaluated against many criteria
  - General security
  - Software Implementations
  - Restricted Space Environments
  - Hardware Implementations
  - Attacks on Implementations
  - Encryption vs. Decryption
  - Key Agility
  - Veratile and Flexible (different key sizes)
  - Potential for Instruction-Level Parallelism

Other Algorithms

MARS, Twofish, Blowfish...

Modes of Operation*

Electronic Code Book (ECB)

- Encryption is point-wise $E_K(P_1|\ldots|P_n) = E_K(P_1)|\ldots|E_K(P_n)$

Cipher Block Chaining Mode

- Encryption is point-wise, but previous result is xor'ed in:

$$E_K(P_1|\ldots|P_n) = E_K(IV \oplus P1) \,|\, E_K(E_K(IV \oplus P1) \oplus P_2) \,|\, \ldots$$

- Requires Initialisation Vector (IV)

Counter Mode

- Inspired by one-time pad:

$$E_K(P_1|\ldots|P_n) = E_K(ctr) \oplus P_1 \,|\, E_K(ctr+1) \oplus P_2 \,|\, \ldots$$

Stream vs. Block Ciphers*

Stream Cipher

- Transforms one symbol into another

Advantages	Disadvantages
Speed	Low-diffusion
Low-error propagation	Susceptible to insertions

Block Cipher

- Works on *blocks* of symbols, e.g., 64 bits

Advantages	Disadvantages
High-diffusion	Speed (lack of)
Resistant to insertion attacks	Error propagation

Using Block Ciphers as Stream Ciphers (special Modes)

Output FeedBack Mode

- Encoding an s bit symbol using a 64 bit block cipher
- Requires IV:

$$E_K(P_1|P_2|\ldots) = \underbrace{|\underbrace{E_K(IV)}_{64}|_s}_{s} \oplus \underbrace{P_1}_{s} \quad | \quad |E_K(|E_K(IV)|_s)|_s \oplus P_2$$

Cipher FeedBack Mode

- Like OFB but using the encrypted output:

$$\underbrace{|\underbrace{E_K(IV)}_{64}|_s}_{s} \oplus \underbrace{P_1}_{s} \quad | \quad \underbrace{|E_K(\underbrace{|\underbrace{E_K(IV)}_{64}|_s}_{64} \oplus P_1)|_s}_{s} \oplus \underbrace{P_2}_{s}$$

Suggested Reading

- Historical
  - David Kahn "The Codebreakers"
  - David Kahn "Seizing the Enigma"
- Handbook of Applied Cryptography
  - `http://www.cacr.math.uwaterloo.ca/hac/`