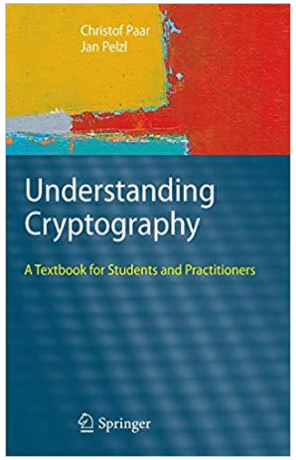
COM/BLM 426
Information Security

Asst. Prof. Dr. Bulent TUGRUL
btugrul@eng.ankara.edu.tr

Slides are mainly based on:

Understanding Cryptography: A Textbook for Students and Practitioners



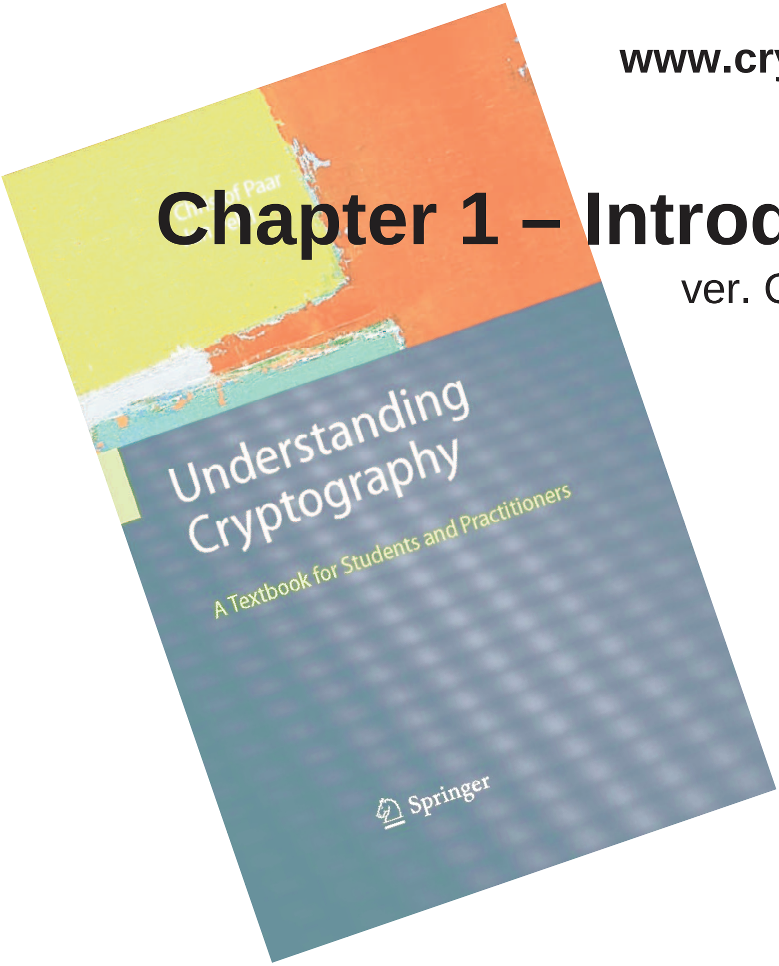
by Christof Paar and Jan Pelzl

Springer, 1st Edition, 2010

Understanding Cryptography – A Textbook for Students and Practitioners

by Christof Paar and Jan Pelzl

www.crypto-textbook.com



Chapter 1 – Introduction to Cryptography

ver. October 28, 2010

These slides were prepared by Christof Paar and Jan Pelzl

Some legal stuff (sorry): Terms of Use

Content of this Chapter

- Overview on the field of cryptology
- Basics of symmetric cryptography
- Cryptanalysis
- Substitution Cipher
- Modular arithmetic
- Shift (or Caesar) Cipher and Affine Cipher

Content of this Chapter

- **Overview on the field of cryptology**
- Basics of symmetric cryptography
- Cryptanalysis
- Substitution Cipher
- Modular arithmetic
- Shift (or Caesar) Cipher and Affine Cipher

Further Reading and Information

Addition to *Understanding Cryptography* .

- A.Menezes, P. van Oorschot, S. Vanstone, *Handbook of Applied Cryptography*. CRC Press, October 1996.
- H.v.Tilborg (ed.), *Encyclopedia of Cryptography and Security*, Springer, 2005

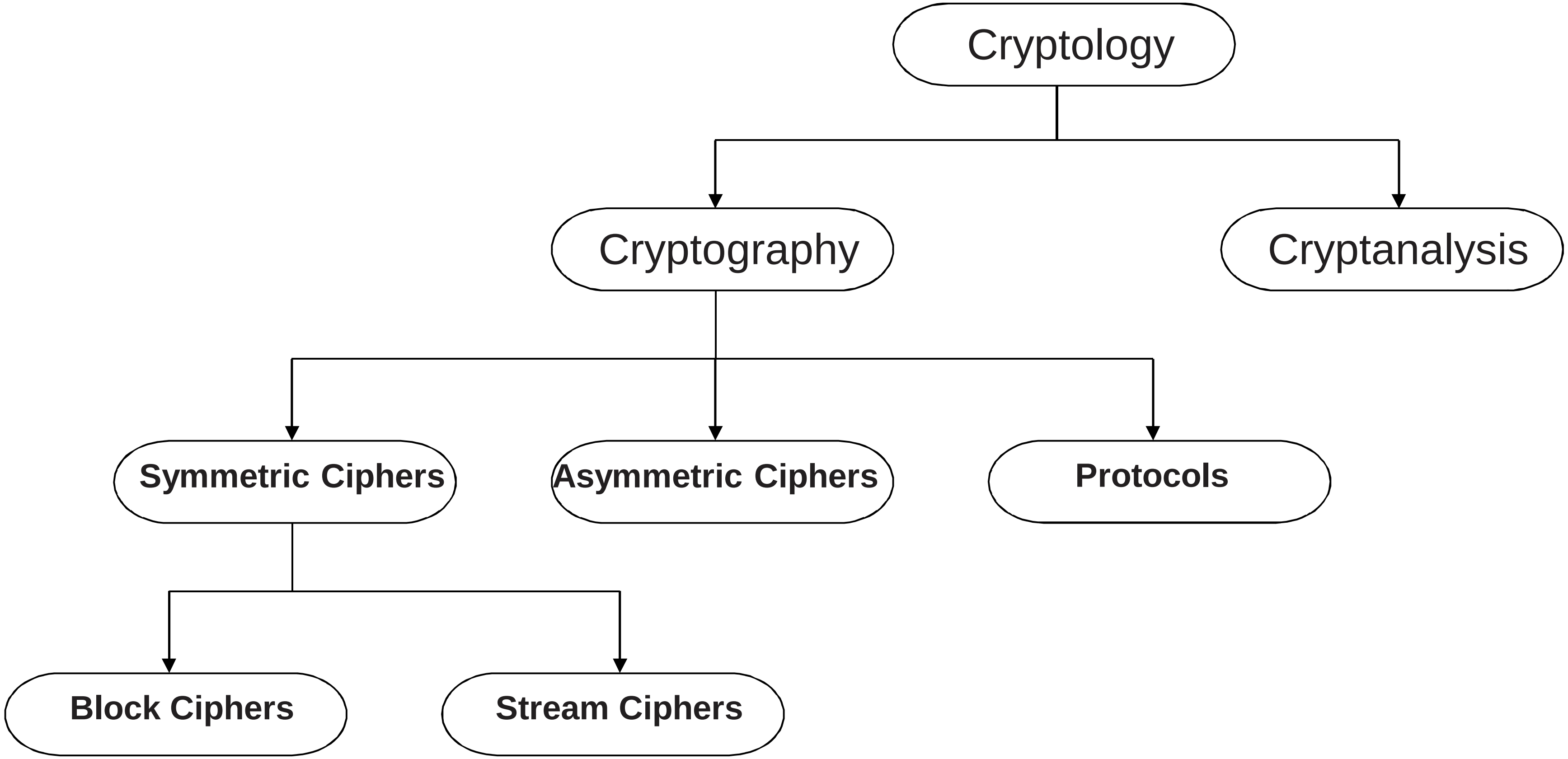
History of Cryptography (great bedtime reading)

- S. Singh, *The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography,* Anchor, 2000.
- D. Kahn, *The Codebreakers: The Comprehensive History of Secret Communication from Ancient Times to the Internet*. 2nd edition, Scribner, 1996.

Software (excellent demonstration of many ancient and modern ciphers)

- *Cryptool, http://www.cryptool.de*

Classification of the Field of Cryptology



Some Basic Facts

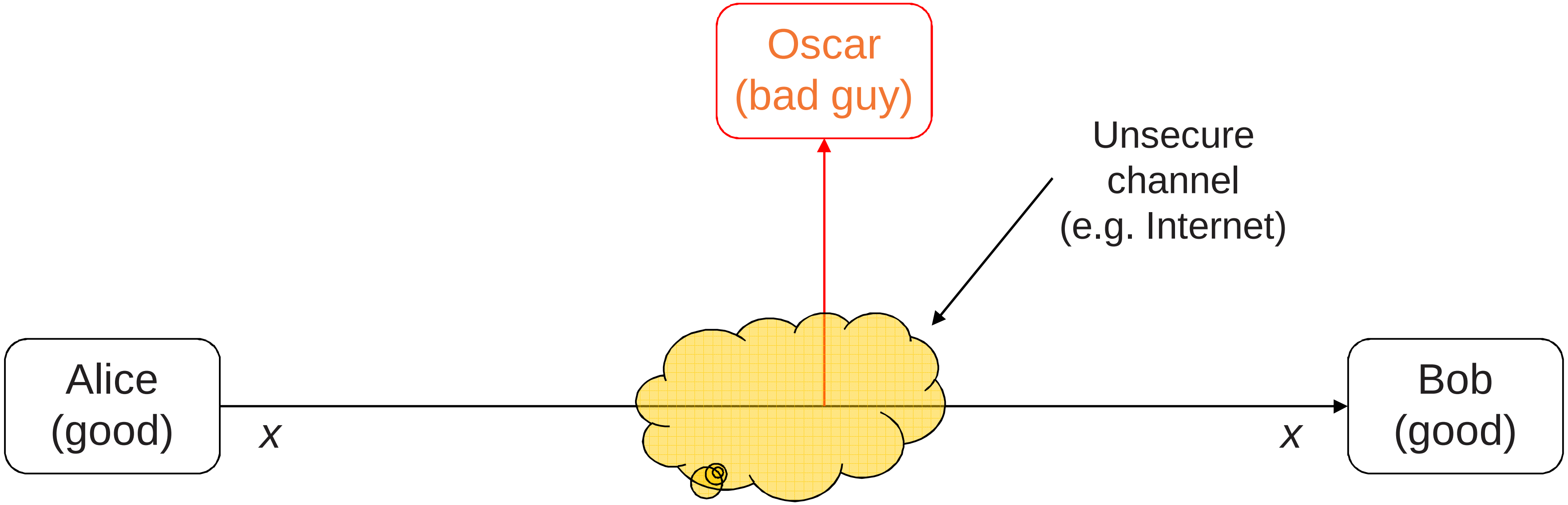
- **Ancient Crypto:** Early signs of encryption in Eqypt in ca. 2000 B.C. Letter-based encryption schemes (e.g., Caesar cipher) popular ever since.
- **Symmetric ciphers:** All encryption schemes from ancient times until 1976 were symmetric ones.
- **Asymmetric ciphers:** In 1976 public-key (or asymmetric) cryptography was openly proposed by Diffie, Hellman and Merkle.
- **Hybrid Schemes:** The majority of today‘s protocols are hybrid schemes, i.e., the use both
  - symmteric ciphers (e.g., for encryption and message authentication) and
  - asymmetric ciphers (e.g., for key exchange and digital signature).

Content of this Chapter

- Overview on the field of cryptology
- **Basics of symmetric cryptography**
- Cryptanalysis
- Substitution Cipher
- Modular arithmetic
- Shift (or Caesar) Cipher and Affine Cipher

Symmetric Cryptography

- Alternative names: **private-key**, **single-key** or **secret-key** cryptography.

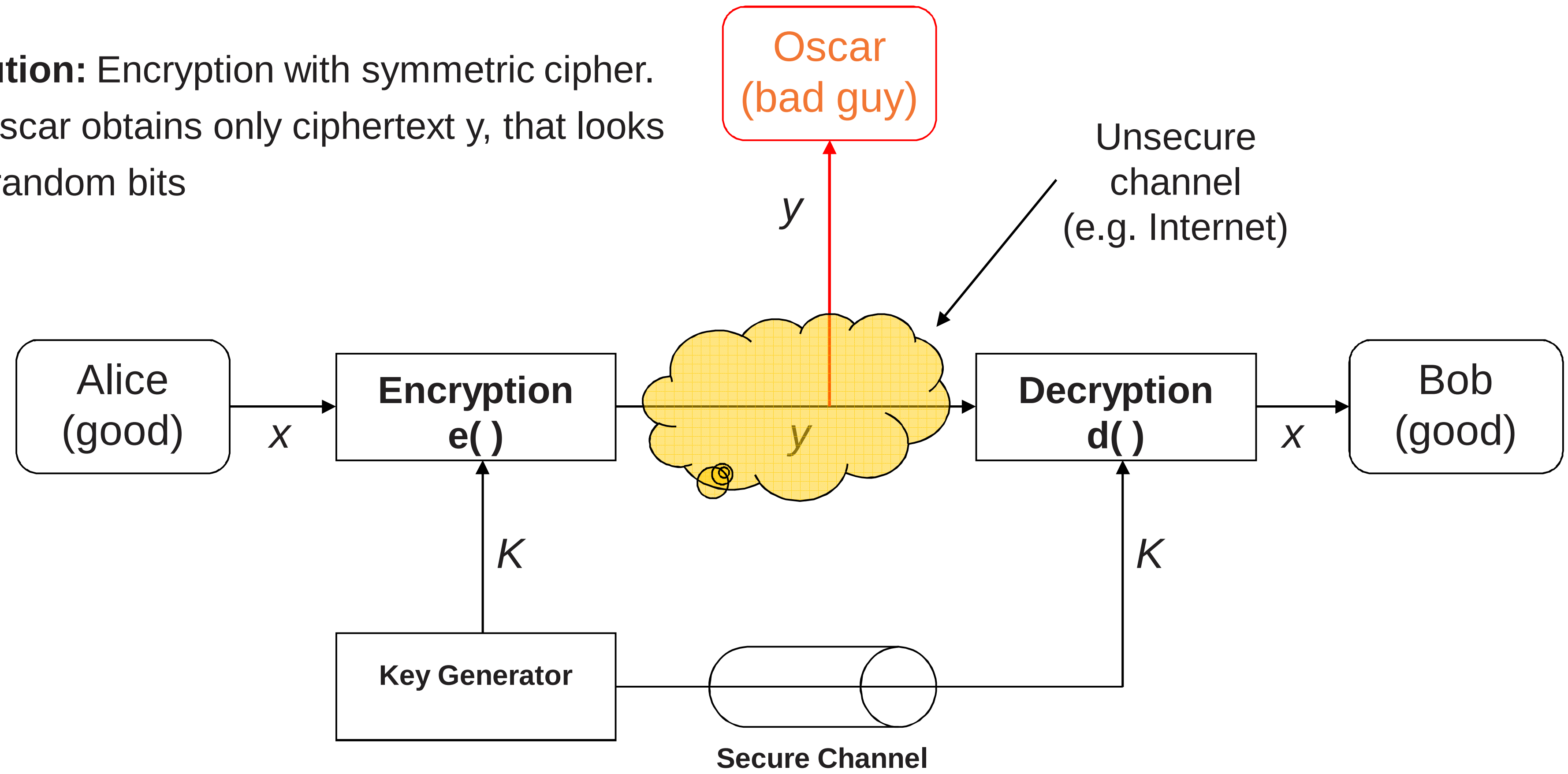


- **Problem Statement:**
  1) Alice and Bob would like to communicate via an unsecure channel (e.g., WLAN or Internet).
  2) A malicious third party Oscar (the bad guy) has channel access but should not be able to understand the communication.

Symmetric Cryptography

Solution: Encryption with symmetric cipher.

⇒ Oscar obtains only ciphertext y, that looks like random bits



- x is the. **plaintext**
- y is the **ciphertext**
- *K* is the **key**
- Set of all keys {*K*1, *K*2, ...,*Kn*} is the **key space**

Symmetric Cryptography

- Encryption equation $\mathbf{y = e_K(x)}$
- Decryption equation $\mathbf{x = d_K(y)}$

- Encryption and decryption are inverse operations if the same key K is used on both sides:

$$d_K(y) = d_K(e_K(x)) = x$$

- Important: The key must be transmitted via a **secure channel** between Alice and Bob.
- The secure channel can be realized, e.g., by manually installing the key for the Wi-Fi Protected Access (WPA) protocol or a human courier.
- However, the system is only secure if an attacker does not learn the key K!

⇒ **The problem of secure communication is reduced to secure transmission and storage of the key K.**

Content of this Chapter

- Overview on the field of cryptology
- Basics of symmetric cryptography
- **Cryptanalysis**
- Substitution Cipher
- Modular arithmetic
- Shift (or Caesar) Cipher and Affine Cipher

Why do we need Cryptanalysis?

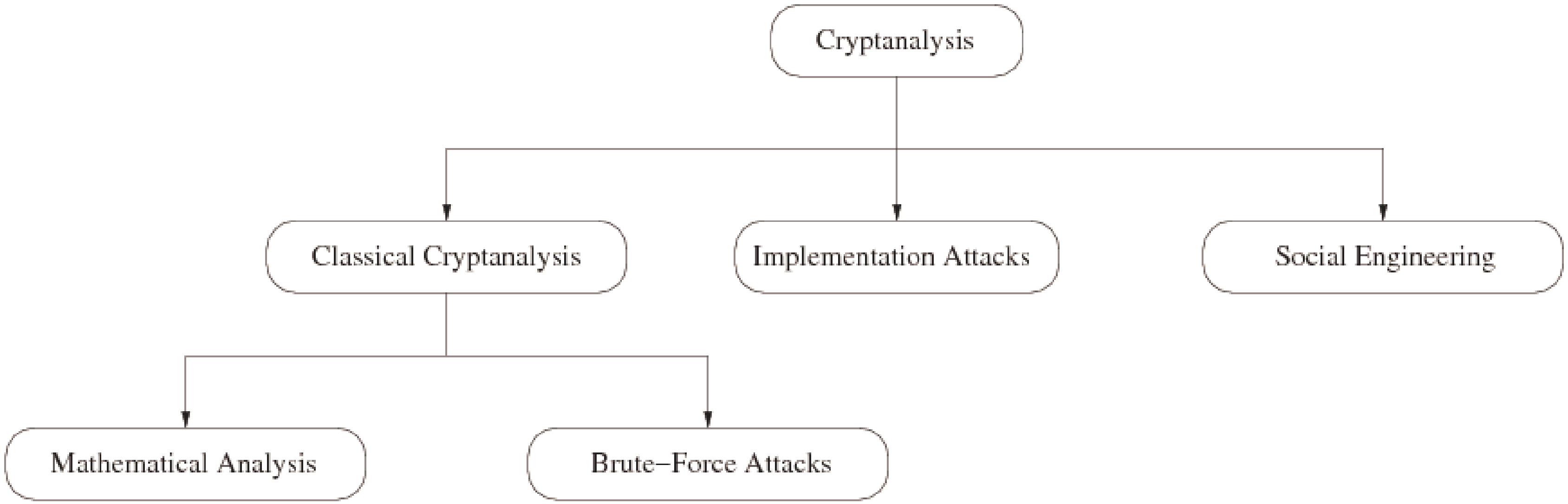
- There is no *mathematical proof of security* for any practial cipher
- The only way to have assurance that a cipher is secure is to try to break it (and fail) !

Kerckhoff‘s Principle is paramount in modern cryptography:

> A cryptosystem should be secure even if the attacker (Oscar) knows all details about the system, with the exception of the secret key.

- In order to achieve Kerckhoff‘s Principle in practice:
  Only use widely known ciphers that have been cryptanalyzed for several years by good cryptographers! (*Understanding Cryptography* only treats such ciphers)
- **Remark:** It is tempting to assume that a cipher is „more secure“ if its details are kept secret. However, history has shown time and again that secret ciphers can almost always been broken once they have been reversed engineered. (Example: Content Scrambling System (CSS) for DVD content protection.)

Cryptanalysis: Attacking Cryptosystems



- **Classical Attacks**
  - Mathematical Analysis
  - Brute-Force Attack
- **Implementation Attack**: Try to extract key through reverese engineering or power measurement, e.g., for a banking smart card.
- **Social Engineering**: E.g., trick a user into giving up her password

Brute-Force Attack (or Exhaustive Key Search) against Symmetric Ciphers

- Treats the cipher as a black box
- Requires (at least) 1 plaintext-ciphertext pair (x_0, y_0)
- Check all possible keys until condition is fulfilled:

$$d_K(y_0) \stackrel{?}{=} x_0$$

- How many keys to we need ?

Key length in bit	Key space	Security life time (assuming brute-force as best possible attack)
64	2^{64}	**Short term** (few days or less)
128	2^{128}	**Long-term** (several decades in the absence of quantum computers)
256	2^{256}	**Long-term** (also resistant against quantum computers – note that QC do not exist at the moment and might never exist)

Important: An adversary only needs to succeed with **one** attack. Thus, a long key space does not help if other attacks (e.g., social engineering) are possible..

Content of this Chapter

- Overview on the field of cryptology
- Basics of symmetric cryptography
- Cryptanalysis
- **Substitution Cipher**
- Modular arithmetic
- Shift (or Caesar) Cipher and Affine Cipher

Substitution Cipher

- Historical cipher
- Great tool for understanding brute-force vs. analytical attacks
- Encrypts letters rather than bits (like all ciphers until after WW II)

Idea: replace each plaintext letter by a fixed other letter.

Plaintext		Ciphertext
A	$\rightarrow$	k
B	$\rightarrow$	d
C	$\rightarrow$	w
	....	

for instance, ABBA would be encrypted as `kddk`

- Example (ciphertext):

```
iq ifcc vqqr fb rdq vfllcq na rdq cfjwhwz hr bnnb  hcc
hwwhbsqvqbre hwq vhlq
```

- How secure is the Substitution Cipher? Let‘s look at attacks…

Attacks against the Substitution Cipher

1. Attack: Exhaustive Key Search (Brute-Force Attack)

- Simply try every possible subsititution table until an intelligent plaintext appears (note that each substitution table is a key)..
- How many substitution tables (= keys) are there?

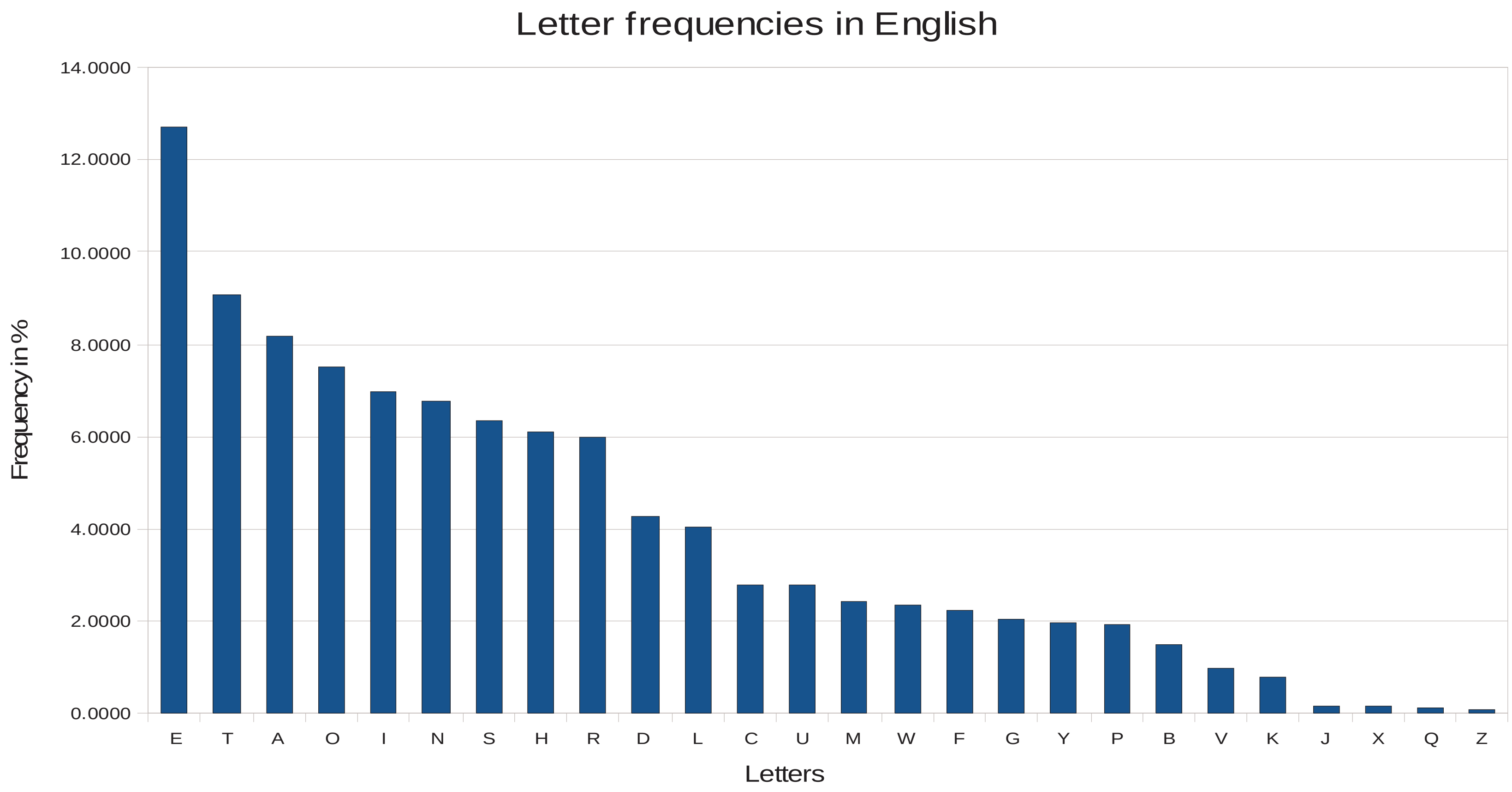
$$26 \times 25 \times \ldots \times 3 \times 2 \times 1 = 26! \approx 2^{88}$$

Search through 2^{88} keys is completely infeasible with today‘s computers!
(cf. earlier table on key lengths)

- Q: Can we now conclude that the substitution cipher is secure since a brute-forece attack is not feasible?
- A: No! We have to protect against **all** possible attacks…

2. Attack: Letter Frequency Analysis (Brute-Force Attack)

- Letters have very different frequencies in the English language
- Moreover: the frequency of plaintext letters is preserved in the ciphertext.
- For instanc, „e“ is the most common letter in English; almost 13% of all letters in a typical English text are „e“.
- The next most common one is „t“ with about 9%.



Breaking the Substitution Cipher with Letter Frequency Attack

- Let‘s retun to our example and identify the most frequent letter:

  `iq ifcc vqqr fb rdq vfllcq na rdq cfjwhwz hr bnnb hcc hwwhbsqvqbre hwq vhlq`

- We replace the ciphertext letter q by E and obtain:

  `iE ifcc vEEr fb rdE vfllcE na rdE cfjwhwz hr bnnb  hcc hwwhbsEvEbre hwE vhlE`

- By further guessing based on the frequency of the remaining letters we obtain the plaintext:

  `WE WILL MEET IN THE MIDDLE OF THE LIBRARY AT NOON ALL ARRANGEMENTS ARE MADE`

Breaking the Substitution Cipher with Letter Frequency Attack

- In practice, not only frequencies of individual letters can be used for an attack, but also the frequency of letter pairs (i.e., „th“ is very common in English), letter triples, etc.
- cf. Problem 1.1 in *Understanding Cryptography* for a longer ciphertext you can try to break!

Important lesson: Even though the substitution cipher has a sufficiently large key space of appr. 2^{88}, it can easily be defeated with analytical methods. This is an excellent example that an encryption scheme must withstand all types of attacks.

Content of this Chapter

- Overview on the field of cryptology
- Basics of symmetric cryptography
- Attacking crypto schemes
- Substitution Cipher
- **Modular arithmetic**
- Shift (or Caesar) Cipher and Affine Cipher

Short Introduction to Modular Arithmetic

Why do we need to study modular arithmetic?

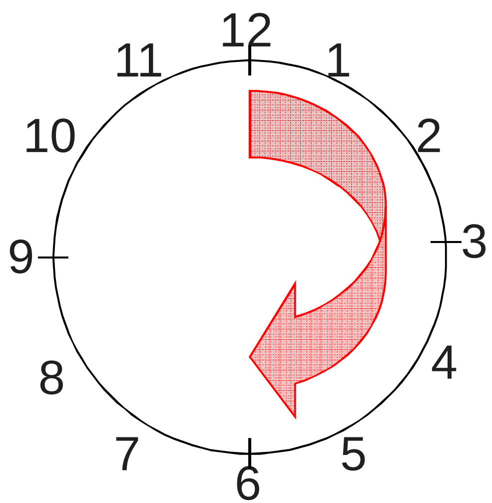
- Extremely important for asymmetric cryptography (RSA, elliptic curves etc.)
- Some historical ciphers can be elegantly described with modular arithmetic (cf. Caesar and affine cipher later on).

Short Introduction to Modular Arithmetic

Generally speaking, most cryptosytems are based on **sets of numbers** that are

1. **discrete** (sets with integers are particularly useful)
2. **finite** (i.e., if we only compute with a finiely many numbers)

Seems too abstract? --- Let‘s look at a finite set with discrete numbers we are quite familiar with: a clock.



Interestingly, even though the numbers are incremented every hour we never leave the set of integers:

1, 2, 3, … 11, 12, 1, 2, 3, … 11, 12, 1, 2, 3, …:

Short Introduction to Modular Arithmetic

- We develop now an arithmetic system which allows us to **compute** in finite sets of integers like the 12 integers we find on a clock (1,2,3, … ,12).
- It is crucial to have an operation which „keeps the numbers within limits“, i.e., after addition and multiplication they should never leave the set (i.e., never larger than 12).

Definition: Modulus Operation

Let a, r, m be integers and $m > 0$. We write

$$a \equiv r \bmod m$$

if (r-a) *is divisible by m.*

- "m" is called the **modulus**
- "r" is called the **remainder**

Examples for modular reduction.

- Let a= 12 and m= 9 : $12 \equiv 3 \bmod 9$
- Let a= 37 and m= 9: $34 \equiv 7 \bmod 9$
- Let a= -7 and m= 9: $-7 \equiv 2 \bmod 9$

(you should check whether the condition „m divides (r-a)“holds in each of the 3 cases)

Properties of Modular Arithmetic (1)

- **The remainder is not unique**

It is somewhat surprising that for every given modulus *m* and number *a*, there are (infinitely) many valid remainders.

Example:

- $12 \equiv 3 \bmod 9$ → 3 is a valid remainder since 9 divides (3-12)
- $12 \equiv 21 \bmod 9$ → 21 is a valid remainder since 9 divides (21-12)
- $12 \equiv -6 \bmod 9$ → -6 is a valid remainder since 9 divides (-6-12)

Properties of Modular Arithmetic (2)

- **Which remainder do we choose?**

By convention, we usually agree on the **smallest positive integer *r*** as remainder. This integer can be computed as

$$a = \underbrace{q}_{\text{quotient}} \; m + \underbrace{r}_{\text{remainder}} \qquad \text{where } 0 \le r \le m-1$$

- Example: $a=12$ and $m=9$

$$12 = 1 \times 9 + 3 \qquad \rightarrow \; r = 3$$

Remark: This is just a convention. Algorithmically we are free to choose any other valid remainder to compute our crypto functions.

Properties of Modular Arithmetic (3)

- **How do we perform modular division?**

First, note that rather than performing a division, we prefer to multiply by the inverse. Ex:

$$b / a \equiv b \times a^{-1} \bmod m$$

The inverse a^{-1} of a number a is defined such that:

$$a\, a^{-1} \equiv 1 \bmod m$$

Ex: What is $5 / 7 \bmod 9$?

The inverse of $7 \bmod 9$ is 4 since $7 \times 4 \equiv 28 \equiv 1 \bmod 9$, hence:

$$5 / 7 \equiv 5 \times 4 = 20 \equiv 2 \bmod 9$$

- **How is the inverse compute?**

The inverse of a number $a \bmod m$ only exists if and only if:

$$\gcd(a, m) = 1$$

(note that in the example above gcd(5, 9) = 1, so that the inverse of 5 exists modulo 9)

For now, the best way of computing the inverse is to use exhaustive search. In Chapter 6 of *Understanding Cryptography* we will learn the powerful Euclidean Algorithm which actually computes an inverse for a given number and modulus.

Properties of Modular Arithmetic (4)

- **Modular reduction can be performed at any point during a calculation**

Let's look first at an example. We want to compute $3^8 \mod 7$ (note that exponentiation is extremely important in public-key cryptography).

1. Approach: Exponentiation followed by modular reduction

$$3^8 = 6561 \equiv \mathbf{2} \mod 7$$

Note that we have the intermediate result 6561 even though we know that the final result can't be larger than 6.

2. Approach: Exponentiation with intermediate modular reduction

$$3^8 = 3^4\, 3^4 = 81 \times 81$$

At this point we reduce the intermediate results 81 modulo 7:

$$3^8 = 81 \times 81 \equiv 4 \times 4 \mod 7$$

$$4 \times 4 = 16 \equiv \mathbf{2} \mod 7$$

Note that we can perform all these multiplications without pocket calculator, whereas mentally computing $3^8 = 6561$ is a bit challenging for most of us.

General rule: For most algorithms it is advantageous to reduce intermediate results as soon as possible.

An Algebraic View on Modulo Arithmetic: The Ring Z_m (1)

We can view modular arithmetic in terms of sets and operations in the set. By doing arithmetic modulo m we obtain **the integer ring Z_m** .with the following properties:

- **Closure**: We can add and multiply any two numbers and the result is always in the ring.
- Addition and multiplication are **associative**, i.e., for all $a,b,c \in Z_m$

  $a + (b + c) = (a + b) + c$

  $a \times (b \times c) = (a \times b) \times c$

  and addition is **commutative**: $a + b = b + a$
- The **distributive law** holds: $a \times (b+c) = (a \times b)+(a \times c)$ for all $a,b,c \in Z_m$
- There is the **neutral element *0* with respect to addition**, i.e., for all $a \in Z_m$

  $a + 0 \equiv a \bmod m$
- For all $a \in Z_m$, there is always an **additive inverse element *–a*** such that

  $a + (-a) \equiv 0 \bmod m$
- There is the **neutral element 1 with respect to multiplication**, i.e., for all $a \in Z_m$

  $a \times 1 \equiv a \bmod m$
- The **multiplicative inverse a^{-1}**

  $a \times a^{-1} \equiv 1 \bmod m$

  exists only for some, but not for all, elements in Z_m.

An Algebraic View on Modulo Arithmetic: The Ring Z_m (2)

Roughly speaking, a ring is a structure in which we can always add, subtract and multiply, but we can only divide by certain elements (namely by those for which a multiplicative inverse exists).

- We recall from above that an element $a \in Z_m$ has a multiplicative inverse only if:

  gcd $(a, m) = 1$

  We say that *a* is **coprime** or **relatively prime** to *m*.

- Ex: We consider the ring $Z_9 = \{0,1,2,3,4,5,6,7,8\}$

  The elements 0, 3, and 6 do not have inverses since they are not coprime to 9.

  The inverses of the other elements 1, 2, 4, 5, 7, and 8 are:

  $1^{-1} \equiv 1 \bmod 9$ $\quad 2^{-1} \equiv 5 \bmod 9$ $\quad 4^{-1} \equiv 7 \bmod 9$

  $5^{-1} \equiv 2 \bmod 9$ $\quad 7^{-1} \equiv 4 \bmod 9$ $\quad 8^{-1} \equiv 8 \bmod 9$

Content of this Chapter

- Overview on the field of cryptology
- Basics of symmetric cryptography
- Attacking crypto schemes
- Substitution Cipher
- Modular arithmetic
- **Shift (or Caesar) Cipher and Affine Cipher**

Shift (or Caesar) Cipher (1)

- Ancient cipher, allegedly used by Julius Caesar
- Replaces each plaintext letter by another one.
- Replacement rule is very simple: Take letter that follows after *k* positions in the alphabet

Needs mapping from letters → numbers:

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

- Example for $k = 7$

Plaintext = ATTACK = 0, 19, 19, 0, 2, 10

Ciphertext = `haahr` = 7, 0, 0, 7, 17

Note that the letters ”wrap around” at the end of the alphabet, which can be mathematically be expressed as reduction modulo 26, e.g., $19 + 7 = 26 \equiv 0 \mod 26$

Shift (or Caesar) Cipher (2)

- Elegant mathematical description of the cipher.

Let k, x, y ε {0,1, …, 25}

- Encryption: $y = e_k(x) \equiv x + k \bmod 26$
- Decryption: $x = d_k(x) \equiv y - k \bmod 26$

- Q; Is the shift cipher secure?
- A: No! several attacks are possible, including:
  - Exhaustive key search (key space is only 26!)
  - Letter frequency analysis, similar to attack against substitution cipher

Affine Cipher (1)

- Extension of the shift cipher: rather than just adding the key to the plaintext, we also multiply by the key
- We use for this a key consisting of two parts: k = (a, b)

Let k, x, y ε {0,1, …, 25}

- Encryption: $y = e_k(x) \equiv a\,x + b \bmod 26$
- Decryption: $x = d_k(x) \equiv a^{-1}(\,y - b) \bmod 26$

- Since the inverse of *a* is needed for inversion, we can only use values for *a* for which:

  $$\gcd(a, 26) = 1$$

  There are 12 values for *a* that fulfill this condition.
- From this follows that the key space is only 12 x 26 = 312 (cf. Sec 1.4 in *Understanding Cryptography*)
- Again, several attacks are possible, including:
  - Exhaustive key search and letter frequency analysis, similar to the attack against the substitution cipher

Lessons Learned

- Never ever develop your own crypto algorithm unless you have a team of experienced cryptanalysts checking your design.
- Do not use unproven crypto algorithms or unproven protocols.
- Attackers always look for the weakest point of a cryptosystem. For instance, a large key space by itself is no guarantee for a cipher being secure; the cipher might still be vulnerable against analytical attacks.
- Key lengths for symmetric algorithms in order to thwart exhaustive key-search attacks:
  - 64 bit: insecure except for data with extremely short-term value
  - 128 bit: long-term security of several decades, unless quantum computers become available (quantum computers do not exist and perhaps never will)
  - 256 bit: as above, but probably secure against attacks by quantum computers.
- Modular arithmetic is a tool for expressing historical encryption schemes, such as the affine cipher, in a mathematically elegant way.