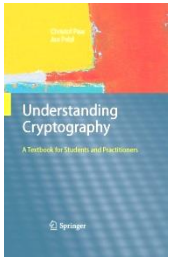
CM3110 Security

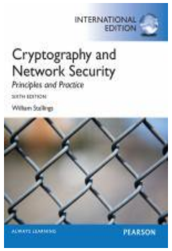
George Theodorakopoulos - TheodorakopoulosG@cardiff.ac.uk

Textbooks

- Christof Paar and Jan Pelzl, "Understanding Cryptography," Springer, 2010.
- William Stallings, "Cryptography and Network Security: Principles and Practice," 6th ed., Prentice Hall, 2014.
- David Kahn, "The Codebreakers: The story of Secret Writing," Scribner, 1996.
- Alfred J. Menezes, Paul C. Van Oorschot, and Scott A. Vanstone, "Handbook of applied cryptography," CRC press, 2010.



Main reference (+slides)

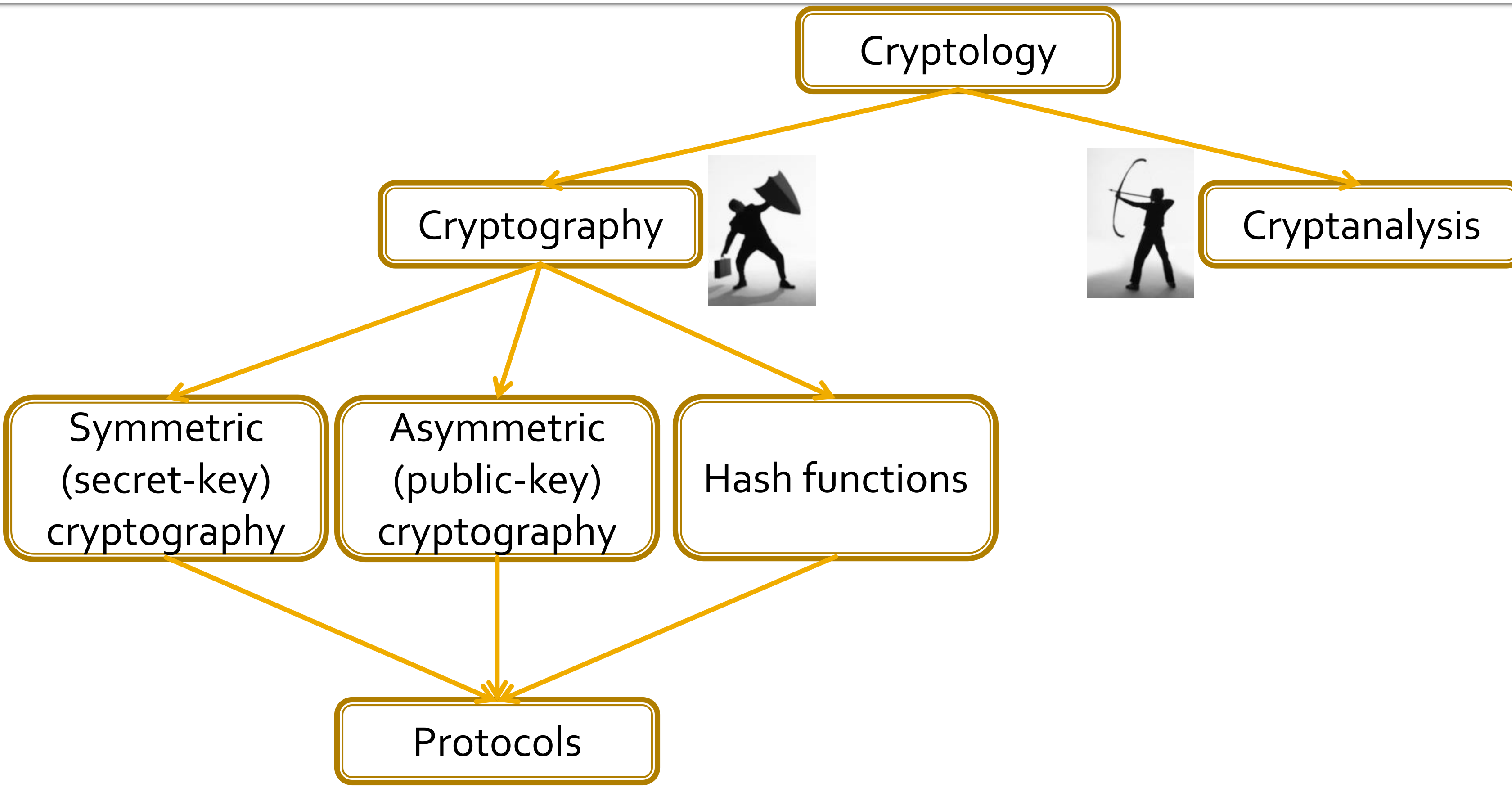


Similar material (incl. TLS)

Crypto History

Mathematics (free online)

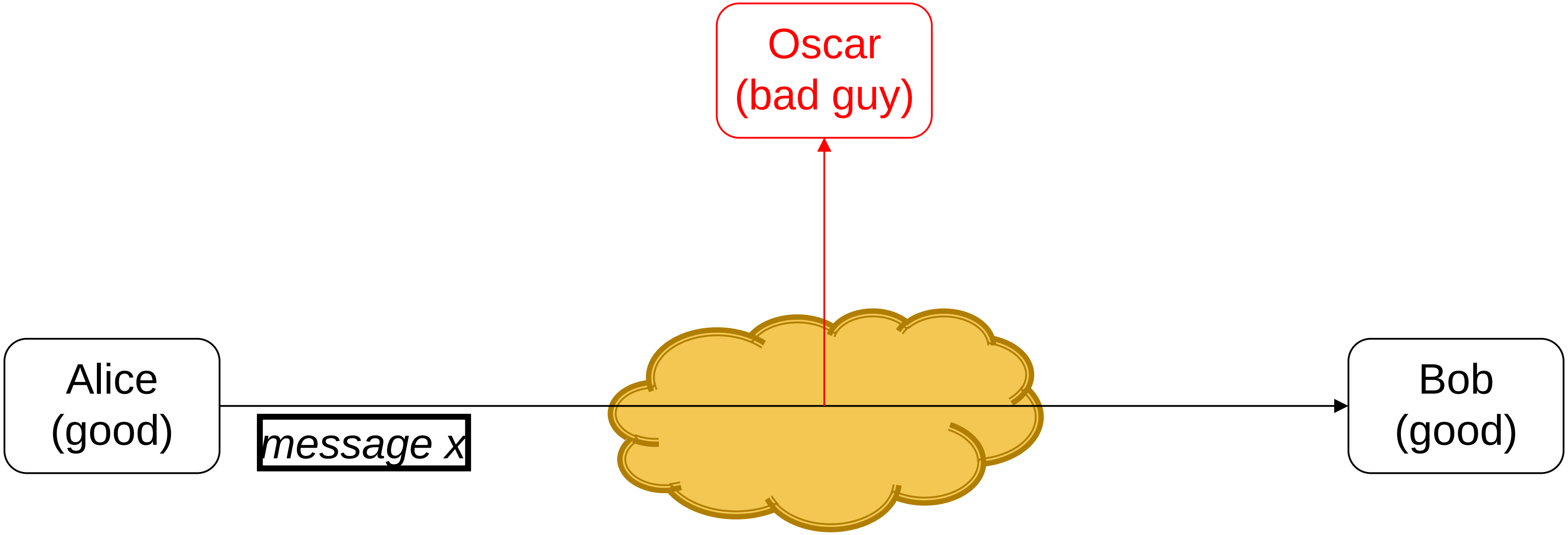
Outline


Cryptology
Cryptography
Cryptanalysis
Symmetric (secret-key) cryptography
Asymmetric (public-key) cryptography
Hash functions
Protocols

Symmetric Cryptography

- Terminology and basic scenario
- Intro to Cryptanalysis
  - Substitution Cipher
  - Brute-force attack and Frequency analysis
- Modular Arithmetic
  - Caesar's Cipher
  - Affine Cipher
- Modern Symmetric ciphers
  - Stream Ciphers
  - Block Ciphers (AES)
  - Modes of operation (ECB, CBC, CTR)

Our Basic Scenario



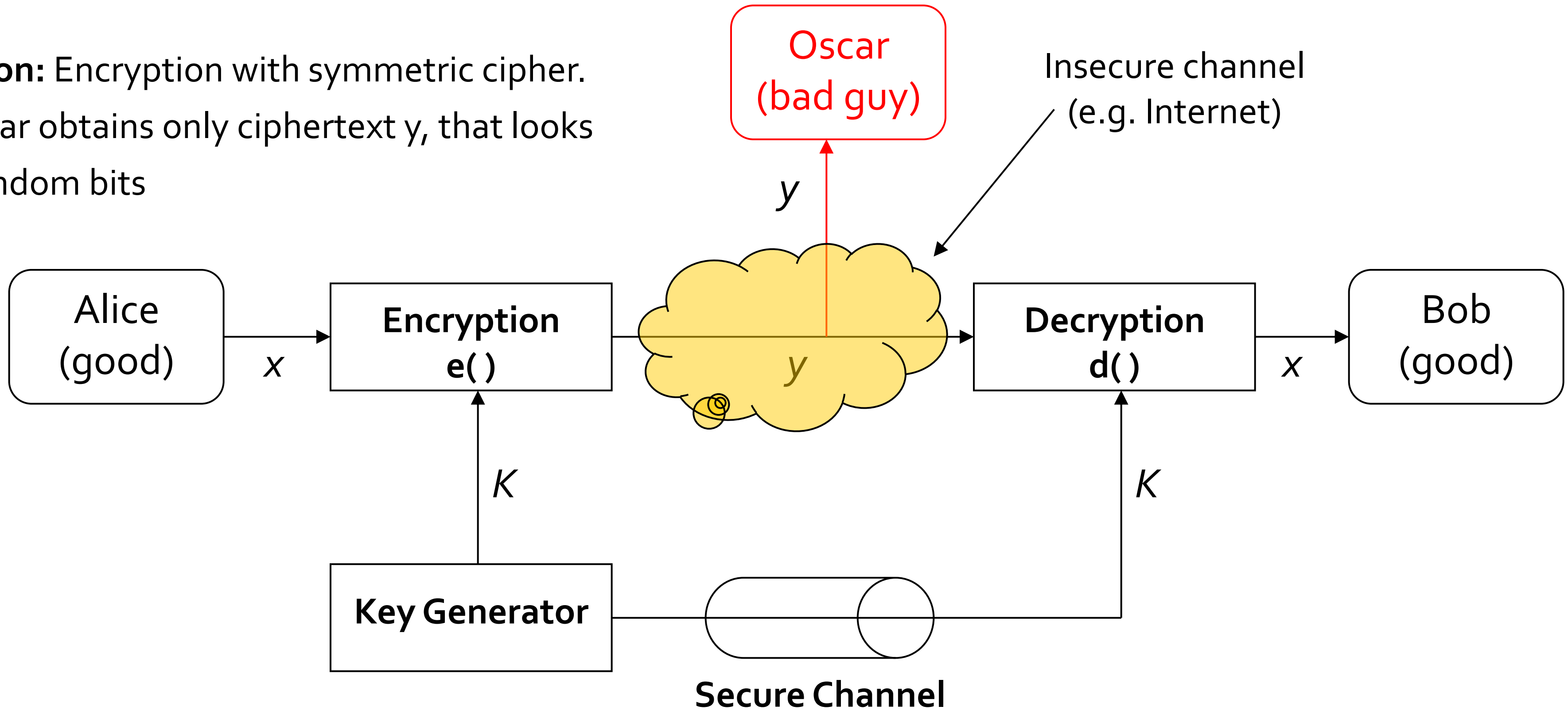
Problem Statement:

1) Alice and Bob want to communicate via an insecure channel (e.g., WLAN or Internet).

2) A malicious third party Oscar (the bad guy) can read the data transmitted through the channel, but he should not be able to understand the conversation between Alice and Bob.

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 possible 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 by sending the key with a human courier.
- However, the system is only secure if an attacker does not learn the key K!

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

Terminology

plaintext	- original message
ciphertext	- encrypted message
cipher	- algorithm for transforming plaintext to ciphertext
key	- info used in cipher known only to sender/receiver
encipher (encrypt)	- converting plaintext to ciphertext
decipher (decrypt)	- recovering ciphertext from plaintext
cryptography	- study of encryption principles/methods
cryptanalysis	- study of principles/ methods of deciphering ciphertext *without* knowing key
cryptology	- field of both cryptography and cryptanalysis

Symmetric Cryptography

- Terminology and basic scenario
- **Intro to Cryptanalysis**
  - Substitution Cipher
  - Brute-force attack and Frequency analysis
- Modular Arithmetic
  - Caesar's Cipher
  - Affine Cipher
- Modern Symmetric ciphers
  - Stream Ciphers
  - Block Ciphers (AES)
  - Modes of operation (ECB, CBC, CTR)

Substitution cipher

- Encrypts letters rather than bits (same as all ciphers up to WW II)
- **IDEA:** Replace each occurrence of a plaintext letter with the same ciphertext letter.

Substitution (Encryption) Table

If plain letter is	`ABCDEFGHIJKLMNOPQRSTUVWXYZ`
replace it with	`DKVQFIBJWPESCXHTMYAUOLRGZN`

EXAMPLE: `CARDIFF` is encrypted to `VDYQWII`

Substitution cipher

- **EXAMPLE:** Ciphertext

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

- Let's try to break this…

Cryptanalysis of Substitution Ciphers

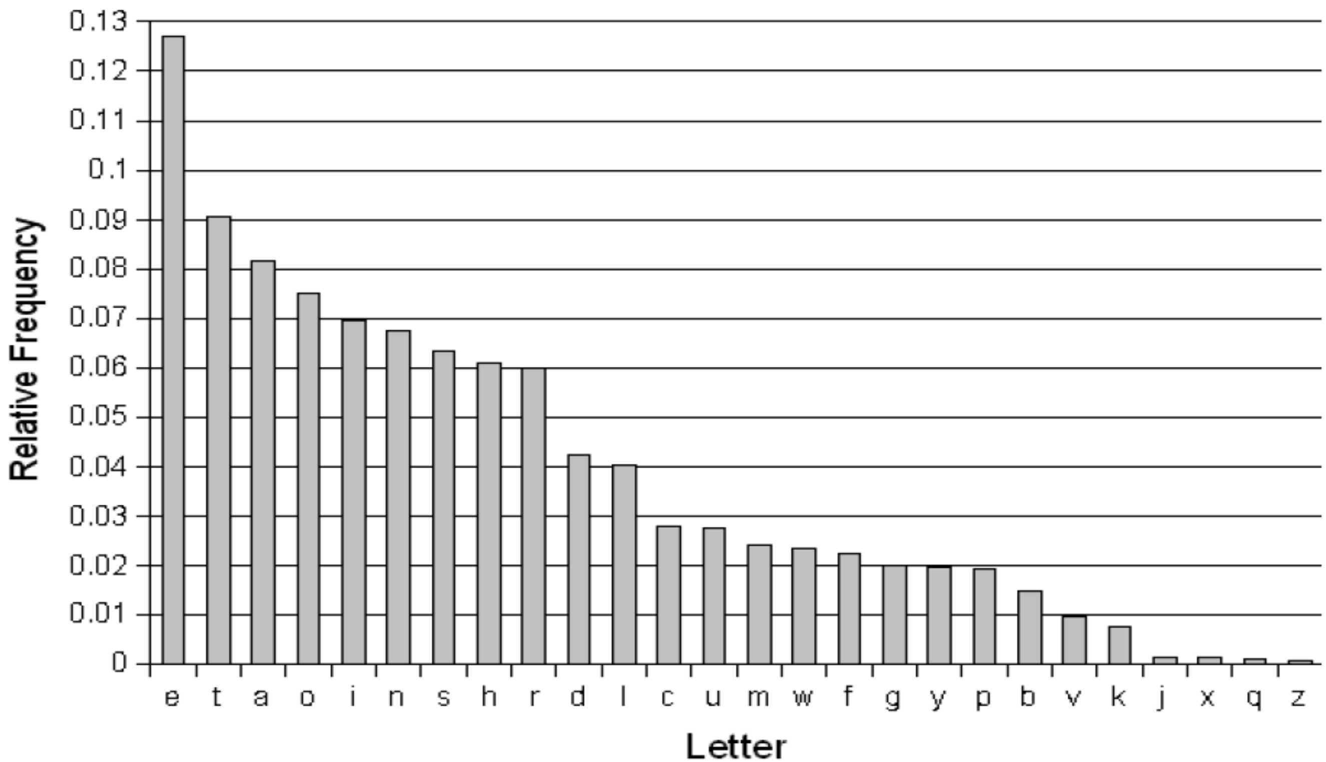
- **Attack #1: Brute force – Exhaustive key search**
- What is the key in a substitution cipher?
- How many possible substitution tables are there?

$$26 \times 25 \times \ldots \times 3 \times 2 \times 1 = 26! = 4.03 \times 10^{26} \approx 2^{88}$$

- Equivalent to an 88-bit key, which is secure enough against today's computers
- So this attack does not work…
- But the key space is not the problem…

Cryptanalysis of Substitution Ciphers

- **Attack #2: Frequency analysis**
- Plaintext letter always replaced by the same ciphertext letter
- Plaintext letter frequencies are not identical



Breaking the Substitution Cipher with Frequency Analysis

- Let's return 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

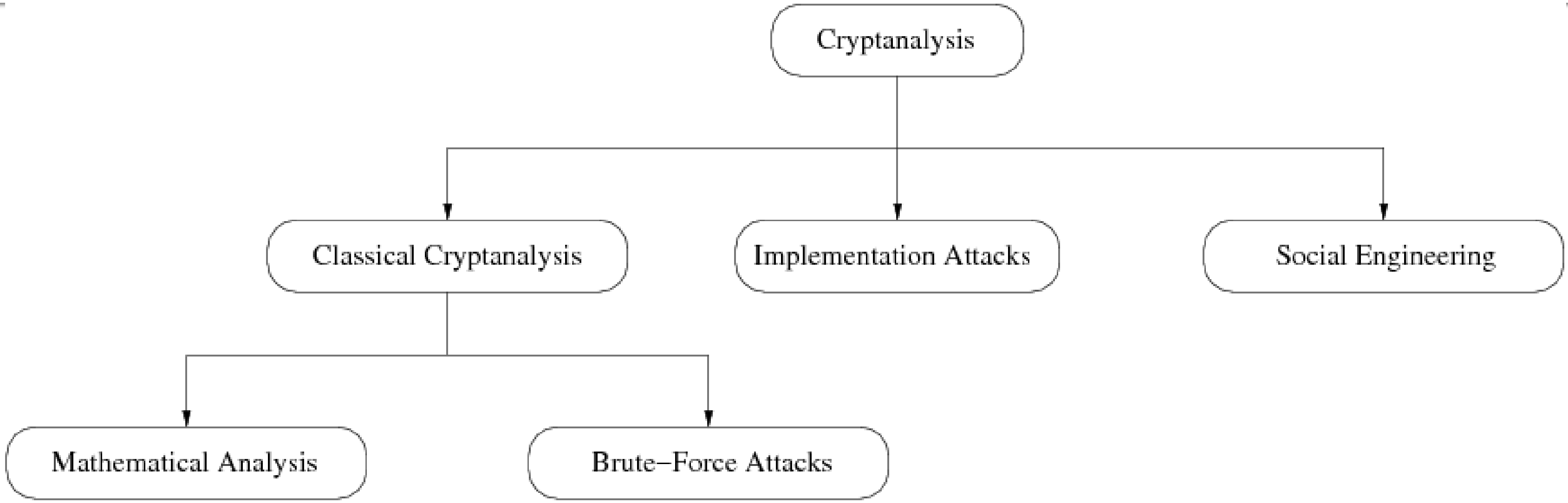
Frequency Analysis: Conclusion

- We can also use frequencies of letter **pairs** (e.g. ‘th’ is very common in English), letter **triples**, etc.

- **EXERCISE:** Try to break ciphertext in Problem 1.1 (main textbook).

- **LESSON:**
  Even though the substitution cipher has a sufficiently large key space of appr. 2^{88}, it can easily be defeated with analytical methods.
  So, encryption schemes must withstand **all types of attacks**.

Cryptanalysis (Attacks)



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

Cryptanalysis (Attacks)

- No **mathematical proofs** of security.
- Security ≈ repeated failures to break cipher

- Kerckhoffs' Principle:

A cryptosystem should be secure even if everything about the system, **except the key**, is public knowledge.

Key size and Brute Force attack

- IDEA: The key space is finite
- **Brute Force attack**:
  1. Get hold of a plaintext-ciphertext pair (x_0, y_0)
  2. Decrypt with all possible keys
  3. If $\mathbf{d_K(y_0) = x_0,}$ **SUCCESS!**

- Nothing we can do against this attack

- AES software decryption takes 352 clock cycles for a 128-bit plaintext.
- If key size is 40bits (number of keys = 2^40) → 2 days in a 2Ghz Pentium.
- If key is 50bits → six years in a 2Ghz Pentium.
- If key is 128bits → 10^24 years in a 2GHz Pentium.

Symmetric Cryptography

- Terminology and basic scenario
- Intro to Cryptanalysis
  - Substitution Cipher
  - Brute-force attack and Frequency analysis
- **Modular Arithmetic**
  - Caesar's Cipher
  - Affine Cipher
- Modern Symmetric ciphers
  - Stream Ciphers
  - Block Ciphers (AES)
  - Modes of operation (ECB, CBC, CTR)

Shift (or Caesar's) Cipher

- 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 $k = 8$

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

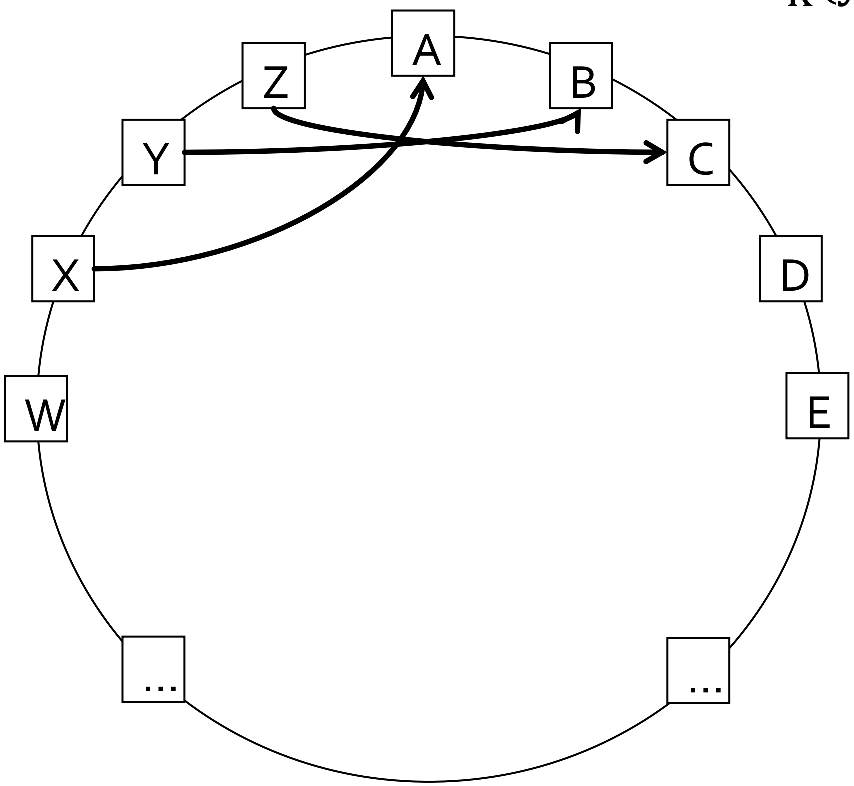
Ciphertext = `ibbiks` = 8, 1, 1, 8, 10, 18

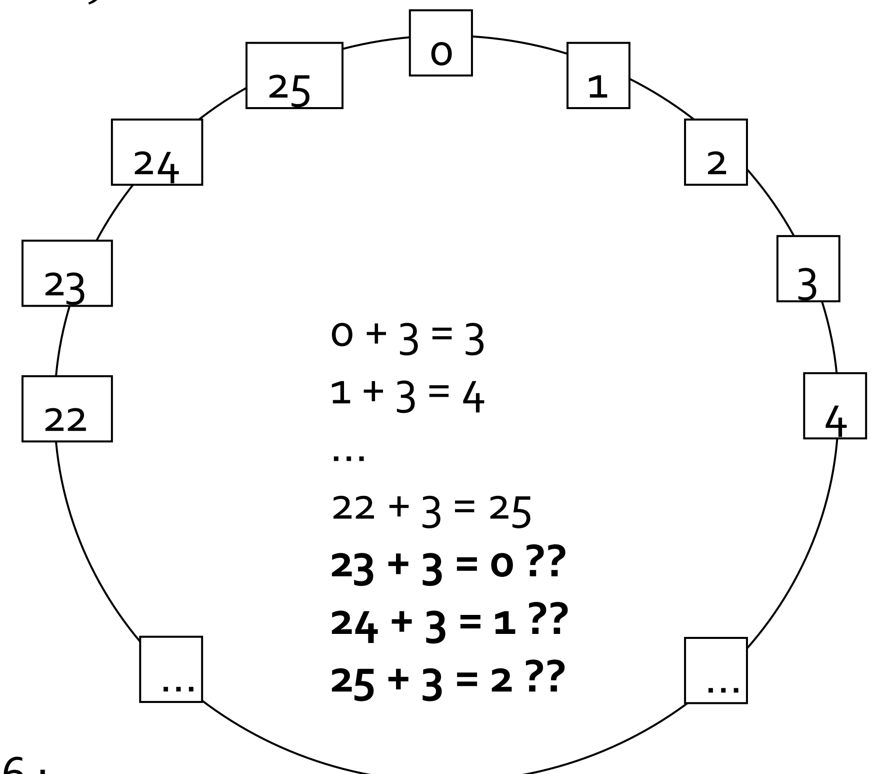
Note that the letters "wrap around" at the end of the alphabet, which can be mathematically be expressed as reduction modulo 26, e.g., $19 + 8 = 27 \equiv 1 \bmod 26$

Wrap around

- Another way of looking at shift ciphers

$$y = e_K(x) = (x + K) \bmod 26$$
$$x = d_K(y) = (y - K) \bmod 26$$





mod 26 :
(positive) remainder when dividing by 26

Shift (or Caesar) Cipher

- 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

Modular Arithmetic

- Why do we care about it?
  - Caesar's cipher is not that important today…
  - BUT: Asymmetric cryptography (RSA and elliptic curve crypto) uses modular arithmetic extensively

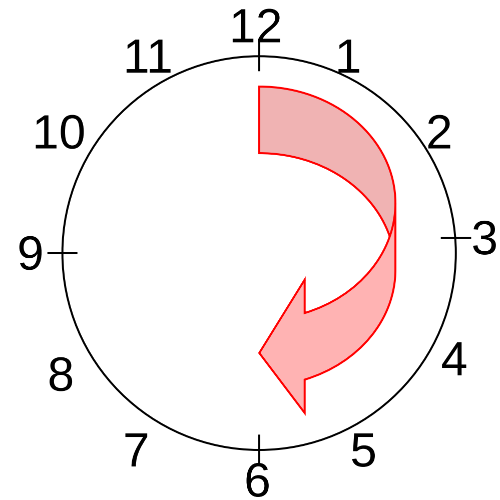
Modular Arithmetic

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

1. **discrete** (sets with integers are particularly useful)
2. **finite** (i.e., we only compute with finitely 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, …

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**

Section 1.4 of *Understanding Cryptography*

Modular Arithmetic

Examples for modular reduction

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

Does the condition „*(r-a) is divisible by m*" hold 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)

As the remainder is not unique, which one do we choose?

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

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

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

$$12 = 1 \times 9 + 3 \qquad \longrightarrow \quad r = 3$$

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, 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 as follows:

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

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:

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

Properties of Modular Arithmetic (4)

How is the inverse computed?

- The inverse of a number *a mod m* exists if and only if:

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

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

- For now, the best way of computing the inverse is to use **exhaustive search**.
- The Euclidean Algorithm is a more efficient way to compute inverses (Chapter 6 of *Understanding Cryptography*)
- Finding modular inverses is very important in public-key cryptography.

Properties of Modular Arithmetic (5)

Modular reduction can be performed at any point during a calculation

Let's look first at an example. We want to compute $3^8 \bmod 7$

(note that exponentiation is extremely important in public-key cryptography).

1st Approach: Exponentiation followed by modular reduction

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

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

Properties of Modular Arithmetic (6)

2nd 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:

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

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

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

General rule: Reduce intermediate results as soon as possible.

Algebra and Modular Arithmetic: The Ring Z_m (1)

We can view modular arithmetic in terms of sets and operations in the set.

The **integer ring Z_m** is the set {0, …, m-1} together with addition and multiplication.

- **Closure**: The sum/product of two numbers is always another number in the ring.
- Addition and multiplication are **associative**, i.e., for all $a,b,c \in Z_m$
  $$a + (b + c) = (a + b) + c$$
  $$a \times (b \times c) = (a \times b) \times c$$
- 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.

Section 1.4 of *Understanding Cryptography*

Algebra and Modular Arithmetic: The Ring Z_m (2)

In a ring, we can always add, subtract, and multiply, but we cannot always divide (we can only divide by the elements that have a multiplicative inverse).

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

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

  We say that a is **coprime** to m 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:

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

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

Affine Cipher

- Extension of the shift cipher: rather than just adding the key to the plaintext, we also multiply by the key
- The key consists 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: $x = d_k(x) \equiv a^{-1}(y - b) \bmod 26$

Question: How large is the key space? (Hint: Which values of *a* are allowed?)

Question: Recall the attacks we have seen. Are they feasible against the affine cipher?

Also: 1.13 and 1.14 from the main textbook