TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

RHEINISCH-WESTFÄLISCHE TECHNISCHE HOCHSCHULE RWTH AACHEN

Cryptography — Lecture notes

Mohamed Barakat and Timo Hanke

Version April 16, 2012

Department of mathematics, University of Kaiserslautern, 67653 Kaiserslautern, Germany *E-mail address:* barakat@mathematik.uni-kl.de

Lehrstuhl D für Mathematik, RWTH Aachen University, Templergraben 64, 52062 Aachen, Germany *E-mail address:* hanke@math.rwth-aachen.de

Preface

These lecture notes are based on the course "Kryptographie" given by Timo Hanke at RWTH Aachen University in the summer semester of 2010. They were amended and extended by several topics, as well as translated into English, by Mohammed Barakat for his course "Cryptography" at TU Kaiserslautern in the winter semester of 2010/11. Besides the literature given in the bibliography section, our sources include lectures notes of courses held by Michael Cuntz, Florian Hess, Gerhard Hiss and Jürgen Müller. We would like to thank them all.

Mohammed Barakat would also like to thank the audience of the course for their helpful remarks and questions. Special thanks to Henning Kopp for his numerous improvements suggestions. Also thanks to Jochen Kall who helped locating further errors and typos. Daniel Berger helped me with subtle formatting issues. Many thanks Daniel.

Contents

CHAPTER 1

General Concepts

For an overview see the slides (in German)
http://www.mathematik.uni-kl.de/~barakat/Lehre/WS10/Cryptography/material/Crypto_talk.pdf

Begin Lect. 2, last 30 min.

1. Algorithms and their runtime

DEFINITION 1.1. An algorithm is called **deterministic** if the output only depends on the input. Otherwise **probabilistic** (or **randomized**).

REMARK 1.2.

(1) The output of a deterministic algorithm is a function of the input.
(2) The steps of a probabilistic algorithm might depend on a **random source**.
(3) If the random source is regarded as an additional input, the probabilistic algorithm becomes deterministic.
(4) Probabilistic algorithms often enough supersede deterministic ones.

DEFINITION 1.3 (O-notation). Let $f : \mathbb{N} \to \mathbb{R}_{>0}$ be a function. Define

$$O(f) := \{h : \mathbb{N} \to \mathbb{R}_{\geq 0} \mid \exists c = c(h) \in \mathbb{R}, N = N(h) \in \mathbb{N} : h(n) \leq cf(n)\ \forall n \geq N\}.$$

O is called the big LANDAU O. Instead of $g \in O(f)$ one often writes $g = O(f)$.

REMARK 1.4. Let $f, g : \mathbb{N} \to \mathbb{R}_{\geq 0}$.

(1) $f \in O(f)$.
(2) $cO(f) = O(f)$ for all $c \in \mathbb{R}_{\geq 0}$.
(3) $O(f)O(g) = O(fg)$.

EXAMPLE 1.5.

(1) $O(1) = \{f : \mathbb{N} \to \mathbb{R}_{\geq 0} \mid f \text{ is bounded}\}$
(2) $O(5n^3 - 3n - 2) = O(n^3)$.
(3) $O(f) \subset O(g)$ for $f \leq g$.

DEFINITION 1.6. The **runtime**[1] $t_A(x)$ of an algorithm A for an input x is the number of (elementary) steps[2] (or operations) of the algorithm (when executed by a computer =

[1] German: Laufzeit
[2] ... including reading from the random source.

multitape TURING machine). The algorithm is said to **lie in** $O(f)$ for $f : \mathbb{N} \to \mathbb{R}_{\geq 0}$ if the runtime of the algorithm is bounded (above) by $f(s)$, where s is the "size"[3] of the input x.

DEFINITION 1.7. An algorithm is called a **polynomial (runtime) algorithm** if it lies in $O(n^k)$ for some $k \in \mathbb{N}_0$. Otherwise an **exponential (runtime) algorithm**.

Begin Lect. 3

EXAMPLE 1.8.

(1) Addition and subtraction of n-digit natural numbers lies in $O(n)$. Cannot be improved further.
(2) Multiplication and division of n-digit natural numbers lies in $O(n^2)$ (schoolbook algorithm). Can be improved: SCHÖNHAGE–STRASSEN multiplication algorithm lies in $O(n \log n \log\log n)$. Let $M(n)$ denote the runtime of the multiplication algorithm.
(3) Factorial of a (fixed) natural number m lies in $O(m^2 \log m)$. Can be improved!

2. Multi-valued maps

DEFINITION 2.1. A **multi-valued** map from M to N is a map $F : M \to \mathbf{2}^N$ with $F(m) \neq \emptyset$ for all $m \in M$, where $\mathbf{2}^N$ denotes the power set of N. We write $F : M \rightsquigarrow N$ and write $F(m) = n$ instead of $n \in F(m)$. Further:

(1) F is called **injective** if the sets $F(m)$ are pairwise *disjoint*.
(2) F is called **surjective** if $\bigcup_{m \in M} F(m) = N$.
(3) F is called **bijective** if it is injective and surjective.
(4) For a *surjective* $F : M \rightsquigarrow N$ define
$$F^{-1} : N \rightsquigarrow M, \quad F^{-1}(n) := \{m \in M \mid n \in F(m)\}.$$
F^{-1} is called the **(multi-valued) inverse** of F.
(5) For $F, F' : M \rightsquigarrow N$ we write $F \subset F'$ if $F(m) \subset F'(m)$ for all $m \in M$.
(6) A multi-valued map F defines a map $M \to N$ iff $|F(m)| = 1$ for all $m \in M$. We then say F *is* a map and denote the corresponding map $M \to N$ again by F.

EXERCISE 2.2.

(1) Let $F, F' : M \rightsquigarrow N$ be two multi-valued maps with $F \subset F'$. Then F' injective implies F injective.
(2) Let $F : M \rightsquigarrow N$ be surjective. Then
   (a) F^{-1} is surjective and $(F^{-1})^{-1} = F$.
   (b) F is injective (and hence bijective) iff F^{-1} is a (surjective) *map*.
(3) Each bijective multi-valued map $F : M \rightsquigarrow N$ is the multi-valued inverse g^{-1} of a surjective *map* $g : N \to M$ (viewed as a multi-valued map).

[3] E.g. the number of symbols needed to encode the value of x. The notion is suggestive although a bit ambiguous.

3. Alphabets and the word semi-group

DEFINITION 3.1. An **alphabet** A is a finite nonempty set. Its cardinality $|A|$ is called the **length** of the alphabet and its elements are called **letters**. Further:

(1) An element $w = (w_1, \dots, w_n) \in A^n$ is called a **word in A of length** $\ell(w) = n$. We write $w = w_1 \dots w_n$.
(2) Set $A^\bullet := \bigcup_{n\in\mathbb{N}_0} A^n$ with $A^0 := \{\varepsilon\}$, where ε is a symbol outside of the alphabet denoting the **empty word of length** 0.
(3) The **concatenation** of words is a binary operation $\cdot$ on $A^\bullet$ defined by $(v_1 \dots v_{\ell(v)}) \cdot (w_1 \dots w_{\ell(w)}) := v_1 \dots v_{\ell(v)} w_1 \dots w_{\ell(w)}$.

EXAMPLE 3.2.

(1) $A = \{\mathtt{a}, \dots, \mathtt{z}\}$, $\mathtt{crypto} \in A^\bullet$.
(2) $A = \{0, 1\}$, $1010010 \in A^\bullet$.

REMARK 3.3. The pair $(A^\bullet, \cdot)$ is a semi-group with neutral element ε. It is ABELIAN iff $|A| = 1$. Further $\ell(v \cdot w) = \ell(v) + \ell(w)$ for $v, w \in A^\bullet$, i.e., $\ell : (A^\bullet, \cdot) \to (\mathbb{Z}_{\geq 0}, +)$ is a semi-group homomorphism.

4. Cryptosystems

DEFINITION 4.1. A **cryptosystem** is a 5-tuple $(P \subset A_1^\bullet, C \subset A_2^\bullet, \kappa : K' \to K, \mathcal{E}, \mathcal{D})$ where

- A_1 and A_2 are alphabets,
- κ is bijective,
- $\mathcal{E} = (\mathcal{E}_e)_{e\in K}$ a family of multi-valued maps $\mathcal{E}_e : P \rightsquigarrow C$, and
- $\mathcal{D} = (\mathcal{D}_d)_{d\in K'}$ a family of surjective maps $\mathcal{D}_d : C \to P$,

such that

$$\mathcal{E}_{\kappa(d)} \subset \mathcal{D}_d^{-1} \quad \text{for all } d \in K'$$

(in the sense of Definition 2.1(5)). We further require that $\kappa, \mathcal{E}, \mathcal{D}$ are realized by polynomial runtime algorithms, where only $\mathcal{E}$ is allowed to be probabilistic. We call

- A_1 the **plaintext alphabet**,
- P the **set of plaintexts**,
- A_2 the **ciphertext alphabet**,
- C the **set of ciphertexts**,
- K the **encryption key space**,
- K' the **decryption key space**,
- κ the **key-correspondence**,
- $\mathcal{E}$ the **encryption algorithm**,
- $\mathcal{E}_e$ the **encryption algorithm with key** e (used by the sender),
- $\mathcal{D}$ the **decryption algorithm**, and
- $\mathcal{D}_d$ the **decryption algorithm with key** d (used by the receiver).

Often enough we take $A_1 = A_2 =: A$ and $P := A^\bullet$.

EXERCISE 4.2. The multi-valued map $\mathcal{E}_e$ is injective for all $e \in K$.

PRINCIPLE 4.3 (**KERCKHOFF's Principle**, 1883).
First formulation: The cryptographic strength of a cryptosystem should not depend on the secrecy of the cryptosystem but only on the secrecy of the decryption key d (see Remark 4.8 below).
Second formulation: The attacker knows the cryptosystem.

A simple justification of this principle is that it becomes increasingly difficult to keep an algorithm secret (**security by obscurity**) if it is used (by an eventually growing number of persons) over a long period of time. On the contrary: It is a lot easier to frequently change and exchange keys between two sides, use different keys for different communications, and destroy keys after usage. And for the same reason any cryptographic weakness of a public algorithm cannot remain secret for a long period of time.

REMARK 4.4.
(1) KERCKHOFF's Principle is nowadays a widely accepted principle.
(2) Major drawback: Your opponent/enemy[4] can use the same thoroughly tested and publicly trusted algorithm.

4.a. Stream ciphers.

DEFINITION 4.5. A cryptosystem is called a **stream cipher** if a word $p = v_1 \dots v_l \in A_1^l \cap P$ is encrypted into a word $\mathcal{E}_e(p) = c = c_0 \cdot w_1 \dots w_l \in C \subset A_2^\bullet$ with $c_0 \in C$ and where the letter w_i does not depend on $v_{i+1}, \dots, v_l$ (but only on e, the letters $v_1, \dots, v_i$, and the random source).

REMARK 4.6. This property of being a stream cipher can be relaxed to N-letter blocks simply by replacing A_1 by A_1^N. If N is "small" one still speaks about a stream cipher, where small means effectively enumerable in a "reasonable" amount of time. For example $\{0,1\}^{32}$ can still be regarded as an alphabet[5] but no longer[6] $\{0,1\}^{128}$.

Begin Lect. 4

4.b. Symmetric and asymmetric cryptosystems.

DEFINITION 4.7. A cryptosystem is called **symmetric** or a **secret key cryptosystem (SKC)** if computing images under κ is *feasible*[7], otherwise an **asymmetric** or a **public key cryptosystem (PKC)**. The corresponding key pairs (d, e) are called **symmetric** or **asymmetric**, respectively.

[4] A source of headache for ministries of interior and secret services.

[5] 32 bits = 4 bytes, the maximum in the UTF-encoding, which is (probably) enough to encode all known human alphabets.

[6] 128 bits = 16 bytes, the AES-block size.

[7] Requiring κ^{-1} to be realized by a polynomial runtime algorithm is not the correct concept as K and K' are finite sets in many relevant cryptosystems. In that case κ^{-1} is trivially computed by a polynomial runtime algorithm by testing the polynomial κ on the finite set K'.

REMARK 4.8.

(1) In many (and quite relevant) symmetric cryptosystems $K = K'$ and $\kappa = \mathrm{id}_K$. We then write $(P, C, K, \mathcal{E}, \mathcal{D})$. The most prominent example is the XOR-cryptosystem.
(2) Whereas the encryption key e of an *asymmetric* cryptosystem can be published (**public key**), e must be kept secret for a *symmetric* cryptosystem. d is in any case called the **secret key**.
(3) As algorithms implementing symmetric cryptosystem are typically more efficient than those of asymmetric ones, symmetric systems are used for almost all the cryptographic traffic, while asymmetric systems are used to exchange the needed symmetric keys.

4.c. Security properties.

DEFINITION 4.9. A cryptosystem is said to have the **security property**[8]

(1) **onewayness**[9] **(OW)** if it is unfeasible for the attacker to decrypt an arbitrary given ciphertext.
(2) **indistinguishability**[10] **(IND)** or **semantic security** if it is unfeasible for the attacker to associate to a given ciphertext one among several known plaintexts.
(3) **non-malleability**[11] **(NM)** if it is unfeasible for the attacker to modify a given ciphertext in such a way, that the corresponding plaintext is sensible.

REMARK 4.10. One can show that: NM $\implies$ IND $\implies$ OW.

4.d. Attacks.

DEFINITION 4.11. One distinguishes the following different **attack scenarios**[12]:

(1) **Ciphertext-only attack (COA)**: The attacker only receives ciphertexts.
(2) **Known-plaintext attack (KPA)**: The attacker receives pairs consisting of a plaintext and the corresponding ciphertext.
(3) **Chosen-plaintext attack (CPA)**: The attacker can *once* choose plaintexts and then receive their corresponding ciphertexts. "Once" in the sense that he is not allowed to alter his choice depending on what he receives.
(4) **Adaptive chosen-ciphertext attack (CCA2)**: The attacker is able to *adaptively* choose ciphertexts and receive their corresponding plaintexts. "Adaptive" means that he is allowed to alter his choice depending on what he receives. If he is challenged to decrypt a ciphertext he is of course not allowed to receive its plain text. But normally such attacks are intended to recover the decryption key d of the decryption algorithm $\mathcal{D}_d$.

[8]German: Sicherheitseigenschaft
[9]German: Einweg-Eigenschaft
[10]German: Nicht-Unterscheidbarkeit
[11]German: Nicht-Modifizierbarkeit
[12]German: Angriffsart

REMARK 4.12.

(1) CPA is trivial for public key systems.
(2) One can show that CCA2 $\succ$ CPA $\succ$ known-plaintext $\succ$ ciphertext-only attacks, where $\succ$ means "stronger than".

4.e. Security models.

DEFINITION 4.13. A **security model** is a security property together with an attack scenario.

REMARK 4.14. One can show that

$$\text{NM-CCA2} = \text{IND-CCA2}.$$

IND-CCA2, i.e., indistinguishability under chosen-ciphertext attack, is the strongest security model of an asymmetric probabilistic cryptosystem. To illustrate IND-CCA2 consider the following game between a challenger[13] H and an attacker A:

(1) H generates a secret key $d \in K'$ and publishes $e = \kappa(d)$.
(2) A has access to the decryption machine $\mathcal{D}_d$ (but not to the secret key d) and is able to perform arbitrary computations.
(3) A generates two different plaintexts $p_0, p_1 \in P$ and hands them to H.
(4) H chooses randomly an $i \in \{0, 1\}$ and sends $c = \mathcal{E}_e(p_i)$ back to A, challenging him to correctly guess i.
(5) A has access to the decryption machine $\mathcal{D}_d$ (but not to the secret key d) and is able to perform arbitrary computations, except deciphering c.
(6) A guesses which i was chosen by H, (only) depending on the computations he was able to do.

IND-CCA2 means that the probability of A correctly guessing i is not higher than $\frac{1}{2}$.

[13] German: Herausforderer

CHAPTER 2

Information Theory

1. Some probability theory

1.a. Probability spaces.

DEFINITION 1.1. Let Ω be a finite nonempty set and $\mu : \Omega \to [0,1]$ with $\sum_{x\in\Omega} \mu(x) = 1$. For $A \subset \Omega$ define $\mu(A) = \sum_{x\in A} \mu(x)$.

(1) (Ω, μ) is called a **finite probability space**[1].
(2) μ is called a **probability measure**[2] or **probability distribution**[3].
(3) A subset $A \subset \Omega$ is called an **event**[4], while an element $x \in \Omega$ an **elementary event**[5].
(4) The distribution $\bar{\mu}$ defined by $\bar{\mu}(x) := \frac{1}{|\Omega|}$ is called **the (discrete) uniform distribution**[6] on Ω.
(5) If $\mu(B) > 0$ define the **conditional probability**[7]
$$\mu(A \mid B) := \frac{\mu(A \cap B)}{\mu(B)},$$
the probability of A given the occurrence of B.
(6) The events A and B are called **(statistically) independent**[8] if
$$\mu(A \cap B) = \mu(A)\mu(B).$$

EXERCISE 1.2. Let (Ω, μ) be a finite probability space and A, B events in Ω.

(1) $\mu(\emptyset) = 0$, $\mu(\Omega) = 1$, $0 \le \mu(A) \le 1$, and $\mu(\Omega \setminus A) = 1 - \mu(A)$.
(2) $A \subset B \subset \Omega \implies \mu(A) \le \mu(B)$.
(3) $\mu(A \cap B) = \mu(A \mid B)\mu(B)$.
(4) **BAYES' formula**:
$$\mu(A \mid B) = \mu(B \mid A)\frac{\mu(A)}{\mu(B)}$$
if $\mu(A), \mu(B) > 0$.

[1] German: Wahrscheinlichkeitsraum
[2] German: Wahrscheinlichkeitsmaß
[3] German: Wahrscheinlichkeitsverteilung
[4] German: Ereignis
[5] German: Elementarereignis
[6] German: (diskrete) Gleichverteilung
[7] German: bedingte Wahrscheinlichkeit
[8] German: stochastisch unabhängig

(5) A and B are independent iff $\mu(B) = 0$ or $\mu(A \mid B) = \mu(A)$.
(6) For $\mu(A), \mu(B) > 0$: $\mu(A \mid B) = \mu(A)$ iff $\mu(B \mid A) = \mu(B)$.

1.b. Random variables.

DEFINITION 1.3. Let (Ω, μ) be a finite probability space.

(1) A map $X : \Omega \to M$ is called an **(M-valued discrete) random variable**[9] **on** Ω.
(2) The distribution μ_X defined by
$$\mu_X(m) := \mu_X(X = m) \text{ for } m \in M$$
is called the **distribution of** X, where $\{X = m\}$ or simply $X = m$ stands for the preimage set $X^{-1}(\{m\})$. It follows that $\mu_X(A) = \mu_X(X \in A)$ for $A \subset M$, where, again, $\{X \in A\}$ or simply $X \in A$ stands for the preimage set $X^{-1}(A)$.
(3) If M is a subset of $\mathbb{C}$ define the **expected value**[10]
$$E(X) := \sum_{x \in \Omega} X(x)\mu(x) \in \mathbb{C}.$$
(4) Let $X_i : \Omega \to M_i$, $i = 1, \dots n$ be random variables. For $m_i \in M$ define the **product probability measure** or **product distribution**
$$\mu_{X_1,\dots,X_n}(m_1, \dots, m_n) := \mu(X_1 = m_1, \dots, X_n = m_n) := \mu(\bigcap_{i=1}^{n} \{X_i = m_i\}).$$

Let $X : \Omega \to M$ and $Y : \Omega \to N$ be two random variables.

(5) X is called **uniformly distributed**[11] if $\mu_X(m) = \frac{1}{|M|}$ for all $m \in M$.
(6) For $\mu_Y(n) > 0$ define the **conditional probability**
$$\mu_{X|Y}(m \mid n) := \frac{\mu_{X,Y}(m, n)}{\mu_Y(n)},$$
the probability of $X = m$ given the occurrence of $Y = n$.
(7) X and Y are called **(statistically) independent** if
$$\mu_{X,Y}(m, n) = \mu_X(m)\mu_Y(n).$$

EXERCISE 1.4. Let (Ω, μ) be a finite probability space and $X : \Omega \to M$ and $Y : \Omega \to N$ be two random variables. Prove:

(1) **BAYES' formula**:
$$\mu_{X|Y}(m \mid n) = \mu_{Y|X}(n \mid m)\frac{\mu_X(m)}{\mu_Y(n)}$$
if $\mu_X(m), \mu_Y(n) > 0$. Or, equivalently:
$$\mu_{X|Y}(m \mid n)\mu_Y(n) = \mu_{Y|X}(n \mid m)\mu_X(m).$$

[9] German: Zufallsvariable
[10] German: Erwartungswert
[11] German: gleichverteilt

(2) X and Y are independent iff for all $m \in M$ and $n \in N$

$$\mu_Y(n) = 0 \text{ or } \mu_{X|Y}(m \mid n) = \mu_X(m).$$

EXERCISE 1.5. Let (Ω, μ) be a finite probability space and $X, Y : \Omega \to M := \mathbb{C}$ be two random variables. Define $X \overset{+}{\cdot} Y : \Omega \to \mathbb{C}$ by $(X \overset{+}{\cdot} Y)(x) = X(x) \overset{+}{\cdot} Y(x)$. Prove:

(1) $E(X) = \sum_{m \in M} m\mu_X(m)$.
(2) $E(X + Y) = E(X) + E(Y)$.
(3) $E(XY) = E(X)E(Y)$ if X and Y are independent. The converse[12] is false.

Begin Lect. 5

2. Perfect Secrecy

2.a. General assumptions. Let $\mathcal{K} := (P, C, K, \mathcal{E}, \mathcal{D})$ be a symmetric cryptosystem and μ_K a probability distribution on K (the probability distribution of choosing an encryption key). For the rest of the section we make the following assumptions:

(1) P, K, C are finite. We know that $|C| \geq |P|$ since $\mathcal{E}_e$ is injective.
(2) $\mu_K(e) > 0$ for all $e \in K$.
(3) All $\mathcal{E}_e$ are *maps*. Identify e with $\mathcal{E}_e$.
(4) $P \times K \to C,\ (p, e) \mapsto e(p)$ is surjective.
(5) Define $\Omega := P \times K$ to be a set of events: (p, e) is the elementary event, where the plain text $p \in P$ is encrypted using the key $e \in K$. Any probability distribution μ_P on P defines a distribution on Ω:

$$\mu(p, e) := \mu((p, e)) := \mu_P(p)\mu_K(e).$$

Conversely: μ_P, μ_K are then the probability distributions of the random variables[13] $P : \Omega \to P,\ (p, e) \mapsto p$ and $K : \Omega \to K, (p, e) \mapsto e$.
(6) The random variables P and K are independent, i.e., $\mu_{P,K} = \mu$ (in words: the choice of the encryption key is independent from the plaintext).

Recall that, by definition, the distribution of the random variable $C : \Omega \to C, (p, e) \mapsto e(p)$ is given by

$$\mu_C(c) = \sum_{\substack{(p,e) \in \Omega \\ e(p) = c}} \mu(p, e).$$

EXERCISE 2.1. Let $P = \{a, b\}$ with $\mu_P(a) = \frac{1}{4}$ and $\mu_P(b) = \frac{3}{4}$. Let $K := \{e_1, e_2, e_3\}$ with $\mu_K(e_1) = \frac{1}{2}$, $\mu_K(e_2) = \mu_K(e_3) = \frac{1}{4}$. Let $C := \{1, 2, 3, 4\}$ and $\mathcal{E}$ be given by the following **encryption matrix**:

$\mathcal{E}$	a	b
e_1	1	2
e_2	2	3
e_3	3	4

[12] German: Umkehrung

[13] Using P and K as names for the random variables is a massive but very useful abuse of language. We will do the same for C in a moment.

Compute the probability μ_C and the conditional probability $\mu_{P|C}$.

2.b. Perfect secrecy.

DEFINITION 2.2 (Shannon 1949). $\mathcal{K}$ is called **perfectly secret**[14] **for** μ_P (or simply **perfect for** μ_P) if P and C are independent, i.e.

$$\forall p \in P, c \in C : \mu_P(p) = 0 \text{ or } \mu_{C|P}(c \mid p) = \mu_C(c),$$

or, equivalently,

$$\forall p \in P, c \in C : \mu_C(c) = 0 \text{ or } \mu_{P|C}(p \mid c) = \mu_P(p).$$

$\mathcal{K}$ is called **perfectly secret** if it is perfectly secret for any probability μ_P.

EXERCISE 2.3. Is the cryptosystem $\mathcal{K}$ defined in Exercise 2.1 perfectly secret for the given μ_P?

REMARK 2.4.

(1) Perfect secrecy means that the knowledge of the ciphertext c does not yield any information on the plaintext p.
(2) Choosing μ_P
   - to be the natural (letter) distribution in a human language tests the security property OW.
   - with $\mu_P(p_0) = \mu_P(p_1) = \frac{1}{2}$ and $\mu_P(p) = 0$ for $p \in P \setminus \{p_0, p_1\}$ tests the security property IND.

2.c. Transitivity.

DEFINITION 2.5. We call $\mathcal{E}$ (or $\mathcal{K}$) **transitive** (**free**, **regular**) if for each pair $(p, c) \in P \times C$ there is one (at most one, exactly one) $e \in K$ with $e(p) = c$.

REMARK 2.6. Regarding each $p \in P$ as a map $p : K \to C,\ e \mapsto e(p)$ we have:

(1) $\mathcal{E}$ is transitive $\iff \forall p \in P : p$ surjective. This implies $|K| \geq |C|$.
(2) $\mathcal{E}$ is free $\iff \forall p \in P : p$ injective. This implies $|K| \leq |C|$.
(3) $\mathcal{E}$ is regular $\iff \forall p \in P : p$ bijective. This implies $|K| = |C|$.

REMARK 2.7.

(1) $|P| = |C|$ iff $e : P \to C$ is bijective for one (and hence for all) $e \in K$.
(2) Let $\mathcal{E}$ be free then: $|K| = |C|$ iff all $p : K \to C$ are bijective.

PROOF. The first statement follows simply from the injectivity of the maps $e : P \to C$. For the second statement again use the injectivitiy argument in Remark 2.6.(2). □

LEMMA 2.8. *The cryptosystem $\mathcal{K}$ is perfectly secret implies that $\mathcal{K}$ is transitive.*

[14] German: perfekt sicher (dies ist keine wörtliche Übersetzung)

PROOF. Assume that $\mathcal{E}$ is not transitive. So there exists a $p \in P$ with $p : K \to C$ is not surjective. Choose a $c \in C \setminus p(K)$. Then $\mu_{P|C}(p \mid c) = 0$ (by definition of μ_C). Since $P \times K \to C$ is surjective there exists a pair $(p', e) \in \Omega$ satisfying $e(p') = c$. Choose μ_P such that $\mu_P(p), \mu_P(p') > 0$. Since $\mu_K(e) > 0$ it follows that $\mu_C(c) > 0$. Hence $\mu_P(p) > 0$ and $\mu_{P|C}(p, c) = 0 < \mu_P(p)\mu_C(c)$, i.e., $\mathcal{K}$ is not perfectly secret. □

COROLLARY 2.9. *The cryptosystem $\mathcal{K}$ is perfectly secret and free implies that it is even* regular *and $|K| = |C|$.* □

EXAMPLE 2.10. These are examples of regular cryptosystems:

(1) $|P| = |C|$: Let G be a finite group and set $P = C = K := G$. Define $e(p) = ep$ (or $e(p) = pe$).
(2) $|P| = 2$: $P = \{p, p'\}$, $K = \{e_1, e_2, e_3, e_4\}$, $C = \{c_1, c_2, c_3, c_4\}$ and

$\mathcal{E}$	p	p'
e_1	c_1	c_2
e_2	c_2	c_1
e_3	c_3	c_4
e_4	c_4	c_3

EXAMPLE 2.11. This examples shows that μ_C might in general depend on μ_P and μ_K: Take $P = \{p_1, p_2\}$, $C = \{c_1, c_2\}$, $K = \{e_1, e_2\}$. Let $\mu_P(p_1)$ and $\mu_K(e_1)$ each take one of three possible values given by the right table (suffices to determine μ_P, μ_K, and μ_C):

$\mathcal{E}$	p	p'
e_1	c_1	c_2
e_2	c_2	c_1

		$\mu_C(c_1)$	$\mu_K(e_1)$		
			$\frac{1}{4}$	$\frac{1}{2}$	$\frac{3}{4}$
$\mu_P(p_1)$	$\frac{1}{4}$	$\frac{10}{16}$	$\frac{1}{2}$	$\frac{6}{16}$
	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
	$\frac{3}{4}$	$\frac{6}{16}$	$\frac{1}{2}$	$\frac{10}{16}$

REMARK 2.12. We can make the observation in the above right table precise:

(1) If $|P| = |C|$ then: μ_P uniformly distributed implies μ_C uniformly distributed.
(2) If $\mathcal{E}$ is regular then: μ_K uniformly distributed implies μ_C uniformly distributed.

PROOF. Keeping Remark 2.7 in mind:

(1) $|P| = |C|$ and μ_P constant implies that

$$\mu_C(c) = \sum_{e \in K} \mu(\mathcal{E}_e^{-1}(c), e) = \sum_{e \in K} \mu_P(\mathcal{E}_e^{-1}(c))\mu_K(e) = \text{const.}$$

(2) Since by the regularity assumption p is bijective for all $p \in P$ and μ_K is constant we conclude that

$$\mu_C(c) = \sum_{p \in P} \mu(p, p^{-1}(c)) = \sum_{p \in P} \mu_P(p)\mu_K(p^{-1}(c)) = \mu_K(p^{-1}(c)) = \text{const.}$$

□

2.d. Characterization. In the rest of the subsection assume $\mathcal{E}$ free. In particular $|K| \leq |C|$, there is no repetition in any column of the encryption matrix, and transitivity is equivalent to regularity.

LEMMA 2.13. *Let $\mathcal{E}$ be regular and μ_P arbitrary. $\mathcal{K}$ is perfectly secret for μ_P iff*

$$\forall e \in K, c \in C : \mu_{K,C}(e,c) = 0 \text{ or } \mu_K(e) = \mu_C(c).$$

PROOF. Recall: $\mathcal{K}$ perfectly secret for μ_P means that

$$\forall p \in P, c \in C : \mu_P(p) = 0 \text{ or } \mu_{C|P}(c \mid p) = \mu_C(c).$$

" $\Longrightarrow$ ": Assume $\mu_{K,C}(e,c) > 0$. Then there exists a $p \in P$ with $e(p) = c$ and $\mu_P(p) > 0$. This p is unique since e is injective. Moreover e is uniquely determined by p and c ($\mathcal{E}$ is free). Hence, the independence of P and K implies

$$\mu_P(p)\mu_K(e) = \mu_{P,K}(p,e) = \mu_{P,C}(p,c) = \mu_{C|P}(c \mid p)\mu_P(p). \tag{1}$$

From $\mu_P(p) > 0$ and the perfect secrecy of $\mathcal{K}$ we deduce that $\mu_K(e) = \mu_C(c)$.
"$\Longleftarrow$": Let $c \in K$ and $p \in P$ with $\mu_P(p) > 0$. The regularity states that there exists exactly one $e \in K$ with $e(p) = c$. The general assumption $\mu_K(e) > 0$ implies $\mu_{K,C}(e,c) > 0$ and hence $\mu_K(e) = \mu_C(c)$. Formula (1) implies $\mu_C(c) = \mu_{C|P}(c \mid p)$. □

THEOREM 2.14. *Let $\mathcal{E}$ be regular. Then $\mathcal{K}$ is perfectly secure for μ_P if μ_K is uniformly distributed.*

PROOF. Remark 2.12 implies that μ_C is uniformly distributed. From $|K| = |C|$ we deduce that $\mu_K(e) = \mu_C(c)$. Now apply Lemma 2.13. □

Begin Lect. 6

THEOREM 2.15. *Let $\mathcal{E}$ be regular (free would suffice) and μ_P arbitrary. If $\mathcal{K}$ is perfectly secure for μ_P and μ_C is uniformly distributed then μ_K is uniformly distributed.*

PROOF. Let $e \in K$. Choose $p \in P$ with $\mu(p) > 0$ and set $c := e(p)$. Then $\mu_{K,C}(e,c) > 0$. Hence $\mu_K(e) = \mu_C(c)$ by Lemma 2.13. (Freeness would suffice to prove " $\Longrightarrow$ " in Lemma 2.13.) □

THEOREM 2.16 (SHANNON, 1949). *Let $\mathcal{K}$ be regular and*[15] *$|P| = |C|$. The following statements are then equivalent:*

1. *$\mathcal{K}$ is perfectly secure for $\bar{\mu}_P$.*
2. *$\mathcal{K}$ is perfectly secure.*
3. *μ_K is uniformly distributed.*

PROOF.
(3) $\Longrightarrow$ (2): Theorem 2.14.
(2) $\Longrightarrow$ (1): Trivial.
(1) $\Longrightarrow$ (3): Let $\mu_P = \bar{\mu}_P \overset{2.12}{\Longrightarrow} \mu_C$ uniformly distributed $\overset{2.15}{\Longrightarrow} \mu_K$ uniformly distributed. □

[15] We will succeed in getting rid of the assumption $|P| = |C|$ later in Theorem 4.21.

EXAMPLE 2.17. The **VERNAM one-time pad (OTP)** introduced in 1917 is perfectly secure:

- $P = C = K = G = ((\mathbb{Z}/2\mathbb{Z})^n, +)$, i.e., bit-strings of length n.
- $e : p \mapsto p + e$, i.e., bitwise addition (a.k.a. XOR-addition).

EXERCISE 2.18. Construct an example showing that the converse of Theorem 2.14 is false and that the condition $|P| = |C|$ in SHANNON's Theorem 2.16 cannot be simply[16] omitted.

3. Entropy

Let $X : \Omega \to X$ be a **finite** random variable[17], i.e., with X finite, say of cardinality n.

3.a. Entropy.

DEFINITION 3.1. The **entropy** of X is defined as

$$H(X) := -\sum_{x \in X} \mu_X(x) \lg \mu_X(x),$$

where $\lg := \log_2$.

As we will see below, the entropy is an attempt to quantify (measure) the diversity of X, the ambiguity of X, our uncertainty or lack of knowledge about the outcome of the "experiment" X.

REMARK 3.2.

(1) Since $\lim_{a \to 0} a \lg a = 0$ we set $0 \lg 0 := 0$. Alternatively one can sum over all $x \in X$ with $\mu_X(x) > 0$.
(2) $H(X) = \sum_{x \in X} \mu_X(x) \lg \frac{1}{\mu_X(x)}$.
(3) $H(X) \geq 0$. $H(X) = 0$ iff $\mu_X(x) = 1$ for an $x \in X$.

PROOF. (3) $-a \lg a \geq 0$ for $a \in [0, 1]$ and $-a \lg a = 0$ iff $a = 0$ or $a = 1$. (The unique maximum in the interval $[0, 1]$ has the coordinates $(\frac{1}{e}, \frac{1}{e \ln(2)}) \approx (0.37, 0.53)$.) □

EXAMPLE 3.3.

(1) Throwing a coin with $\mu_X(0) = \frac{3}{4}$ and $\mu_X(1) = \frac{1}{4}$:

$$H(X) = \frac{3}{4} \lg \frac{4}{3} + \frac{1}{4} \lg 4 = \frac{3}{4}(2 - \lg 3) + \frac{1}{4} 2 = 2 - \frac{3}{4} \lg 3 \approx 0.81.$$

Let $\boxed{n := |X| < \infty}$ by the above general assumption.

[16] However, Theorem 4.21 shows that it can be replaced by the necessary condition of Corollary 2.9.
[17] We deliberately denote M by X as no confusion should occur!

(2) If X (i.e., μ_X) is uniformly distributed then

$$H(X) = \sum_{i=1}^{n} \frac{1}{n} \lg n = \lg n.$$

We will see later in Theorem 3.14 that $H(X) \leq \lg n$ and $H(X) = \lg n$ if and only if μ_X is uniformly distributed.

EXAMPLE 3.4. Let $X = \{x_1, x_2, x_3\}$ with $\mu_X(x_1) = \frac{1}{2}$, $\mu_X(x_2) = \mu_X(x_3) = \frac{1}{4}$. "Encode"[18] x_1 as 0, x_2 as 10, and x_3 as 11. The average bit-length of the encoding is

$$\mu_X(x_1) \cdot 1 + \mu_X(x_2) \cdot 2 + \mu_X(x_3) \cdot 2 = \frac{1}{2} + \frac{1}{2} + \frac{1}{2} = \frac{3}{2},$$

which in this case coincides with the entropy $H(X)$.

3.b. Encodings.

DEFINITION 3.5. A map $f : X \to \{0,1\}^\bullet$ is called a **encoding**[19] of X if the extension to $X^\bullet$ defined by

$$f : X^\bullet \to \{0,1\}^\bullet,\ x_1 \cdots x_n \mapsto f(x_1) \cdots f(x_n)$$

is an injective map.

EXAMPLE 3.6. Suppose $X = \{a, b, c, d\}$, and consider the following three different encoding candidates:

$f(a)$	$=$	1	$f(b)$	$=$	10	$f(c)$	$=$	100	$f(d)$	$=$	1000
$g(a)$	$=$	0	$g(b)$	$=$	10	$g(c)$	$=$	110	$g(d)$	$=$	111
$h(a)$	$=$	0	$h(b)$	$=$	01	$h(c)$	$=$	10	$h(d)$	$=$	11

f and g are encodings but h is not.

- An encoding using f can be decoded by starting at the end and moving backwards: every time 1 appears signals the end of the current element.
- An encoding using g can be decoded by starting at the beginning and moving forward in a simple sequential way by cutting off recognized bit-substrings. For example, the decoding of 10101110 is *bbda*.
- $h(ac) = 010 = h(ba)$.

For an encoding using f we could have started from the beginning. But to decide the end of an encoded substring we need to look one step forward. And decoding from the end forces us to use memory.

Maps like g that have the property of allowing a simple sequential encoding are called **prefix-free**: An encoding g is **prefix-free** if there do *not* exist two elements $x, y \in X$ and a string $z \in \{0,1\}^\bullet$ such that $g(x) = g(y)z$.

[18] See next definition.

[19] German: Kodierung. Do not confuse encoding with encryption.

Let $\ell : \{0,1\}^\bullet \to \mathbb{N}_0$ denote the length function (cf. Definition 3.1.(1)). Then $\ell \circ f \circ X$ is the random variable with expected value

$$\ell(f) := \sum_{x \in X} \mu_X(x)\ell(f(x)),$$

expressing the average length of the encoding f.

The idea is that the entropy of X should be $\ell(f)$, where f is the "most efficient" encoding of X. We would expect f to be most efficient if an event with probability $0 < a < 1$ should be encoded by a bit-string of "length" $-\lg a = \lg \frac{1}{a}$. In Example 3.4 we encoded an event with probability $\frac{1}{2^n}$ by a bit-string of length $n = -\lg \frac{1}{2^n}$.

THEOREM 3.7. *There exists an encoding f with $H(X) \leq \ell(f) \leq H(X) + 1$.*

PROOF. **HUFFMAN'S ALGORITHM** produces such an f. We illustrate it on the next example. □

EXAMPLE 3.8 (HUFFMAN'S ALGORITHM). Suppose $X := \{a, b, c, d, e\}$ has the following probability distribution: $\mu_X(a) = 0.05$, $\mu_X(b) = 0.10$, $\mu_X(c) = 0.12$, $\mu_X(d) = 0.13$, and $\mu_X(e) = 0.60$. View the points of X as the initial vertices of some graph. Take two vertices x, y with lowest probability $\mu_X(x), \mu_X(y)$ and connect them to a new vertex and label the two directed edges by $0, 1$ respectively. Assign to the new vertex the probability $\mu_X(x) + \mu_X(y)$. Repeat the process forgetting x and y until creating the edge assigned the probability 1.

This gives the following *prefix-free* encoding table:

x	$f(x)$
a	000
b	001
c	010
d	011
e	1

The average length of the encoding is

$$\ell(f) = 0.05 \cdot 3 + 0.10 \cdot 3 + 0.12 \cdot 3 + 0.13 \cdot 3 + 0.60 \cdot 1 = 1.8,$$

approximating the value of the entropy $H(X) \approx 1.74$ as described by the previous theorem.

3.c. Entropy of a natural language.

EXAMPLE 3.9. Let X be a random variable with values in $X = A = \{a, \ldots, z\}$.

(1) If μ_X is uniformly distributed then $H(X) = \lg 26 \approx 4.70$ (i.e., more than 4 bits and less than 5 bits).
(2) If μ_X is the distribution of the letters in the English language then $H(X) \approx 4.19$.

Begin Lect. 7

DEFINITION 3.10. Let A be an alphabet.

(1) If X is a random variable with $X \subset A^\ell$ then we call
$$R(X) := \lg n - H(X)$$
the **redundancy** of X. Since $0 \leq H(X) \leq \lg n$ we deduce that $0 \leq R(X) \leq \lg n$. By definition $H(X) + R(X) = \lg n$.

(2) Let $L_\ell \subset A^\ell$ be the random variable of ℓ-grams in a **(natural) language** $L \subset A^\bullet$. The **entropy of L (per letter)** is defined as
$$H_L := \lim_{\ell\to\infty} \frac{H(L_\ell)}{\ell}.$$
The **redundancy of L (per letter)** is defined as
$$R_L := \lg |A| - H_L = \lim_{\ell\to\infty} \frac{R(L_\ell)}{\ell}.$$

EXAMPLE 3.11. For $L = \text{English}$ we estimate $H(L_1) \approx 4.19$, $H(L_2) \approx 3.90$. Empirical data shows that
$$1.0 \leq H_L := H_{\text{English}} \leq 1.5.$$
For $H_L = 1.25 \approx 27\% \cdot \lg |A|$ the redundancy $R_L = R_{\text{English}} = 4.70 - 1.25 = 3.45 \approx 73\% \cdot \lg |A|$.
To understand what this means let us consider the following model for L: Assume $L \cap A^\ell$ contains exactly t_ℓ equally probable texts (or text beginnings), while all other texts have probability zero. Then from $H_L = \lim_{\ell\to\infty} \frac{\lg t_\ell}{\ell} = 1.25$ we conclude that $t_\ell \approx 2^{1.25\cdot\ell}$ for $\ell \gg 0$. For example, $t_{10} \approx 5793$ compared to the $|A^{10}| = 26^{10} \approx 1.41 \cdot 10^{14}$ possible 10-letter strings.

REMARK 3.12. A single text has no entropy. Entropy is only defined for a language.

3.d. Further properties.

DEFINITION 3.13. Let $X : \Omega \to X$ and $Y : \Omega \to Y$ be two finite random variables. Define

(1) the **joint entropy**[20]
$$H(X, Y) := -\sum_{x,y} \mu_{X,Y}(x, y) \lg \mu_{X,Y}(x, y).$$

(2) the **conditional entropy** or **equivocation**[21]
$$H(X \mid y) := -\sum_{x} \mu_{X|Y}(x \mid y) \lg \mu_{X|Y}(x \mid y)$$
and
$$H(X \mid Y) := \sum_{y} \mu_Y(y) H(X \mid y).$$

[20]German: Gemeinsame Entropie
[21]German: Äquivokation = Mehrdeutigkeit

(3) the **transinformation**[22]

$$I(X,Y) := H(X) - H(X \mid Y).$$

THEOREM 3.14.

(1) $H(X) \leq \lg n$. *Equality holds iff* μ_X *is uniformly distributed.*
(2) $H(X,Y) \leq H(X) + H(Y)$. *Equality holds iff* X, Y *are independent.*
(3) $H(X \mid Y) \leq H(X)$ *and equivalently* $I(X,Y) \geq 0$. *Equality holds iff* X, Y *are independent.*
(4) $H(X \mid Y) = H(X,Y) - H(Y)$.
(5) $H(X \mid Y) = H(Y \mid X) + H(X) - H(Y)$.
(6) $I(X,Y) = I(Y,X)$.

PROOF. (1) and (2) are exercise. For (2) use JENSEN's inequality (cf. Lemma A.1.1). (4) is a simple exercise. (3) follows from (2) and (4). (5) follows from (4) (since $H(X,Y) = H(Y,X)$, by definition) and (6) from (5). □

EXAMPLE 3.15. Let X be a random variable and X^n the random variable describing the n-fold independent repetition[23] of the "experiment" X. Then

$$H(X^n) = nH(X).$$

- If X describes throwing a perfect coin (i.e., μ_X is uniformly distributed) then $H(X^n) = H(\underbrace{X, \ldots, X}_{n}) = n$.
- If X describes throwing the coin of Example 3.3(1) then $H(X^n) \approx 0.81 \cdot n$.

4. Entropy in cryptosystems

For the rest of the chapter (course) let $\mathcal{K} = (P, C, K, \mathcal{E}, \mathcal{D})$ be a symmetric cryptosystem satisfying

(1) P, K, C are finite. In particular $|C| \geq |P|$ as $\mathcal{E}_e$ is injective by Exercise 1.4.2.
(2) $\mathcal{E}_e$ is a *map*.
(3) P and K are independent.

LEMMA 4.1. *The above assumptions on* $\mathcal{K}$ *imply:*

(1) $H(P,K) = H(K,C) = H(P,K,C)$
(2) $H(C) \geq H(C \mid K) = H(P \mid K) = H(P)$.
(3) $H(K \mid C) = H(P) + H(K) - H(C)$.
(4) $I(K,C) = H(C) - H(P) \geq 0$.

PROOF. $\mathcal{E}$ is injective and P, K are independent. □

DEFINITION 4.2. One calls

[22] German: Transinformation = gegenseitige Information

[23] If you are still in doubt of what this means then interpret X as the event space and define X^n as the product space with the product distribution.

- $H(K \mid C)$ the **key equivocation**[24].
- $I(K, C)$ the **key transinformation**.

REMARK 4.3.

- The statement $H(P) \leq H(C)$ is a generalization of Remark 2.12: If $|P| = |C|$ then μ_P uniformly distributed implies μ_C uniformly distributed.
- $H(P) < H(C)$ is possible, e.g., when $\mathcal{K}$ is perfectly secret, $|P| = |C|$, and P not uniformly distributed.

EXERCISE 4.4. Construct under the above assumption a cryptosystem with $H(K) < H(C)$.

DEFINITION 4.5. Denote by

$$R(P) := \lg |P| - H(P)$$

the **redundancy** of P.

THEOREM 4.6. *Let $|P| = |C|$. Then*

$$H(K) \geq H(K \mid C) \geq H(K) - R(P)$$

and

$$R(P) \geq I(K, C) \geq 0.$$

PROOF. Let $|P| = |C| = n$. From $H(C) \leq \lg n$ we deduce that

$$H(K \mid C) \geq H(K) + H(P) - \lg n = H(K) - R(P)$$

and

$$I(K, C) \leq \lg n - H(P) = R(P).$$

□

EXAMPLE 4.7. Reconsider Example 2.11 where $P = \{p_1, p_2\}$, $C = \{c_1, c_2\}$, $K = \{e_1, e_2\}$, and

$\mathcal{E}$	p	p'
e_1	c_1	c_2
e_2	c_2	c_1

Choose the distributions $\mu_P = (\frac{1}{4}, \frac{3}{4}), \mu_K = (\frac{1}{4}, \frac{3}{4})$. Then $\mu_C = (\frac{10}{16}, \frac{6}{16})$, $H(P) = H(K) \approx 0.81$, and $H(C) \approx 0.95$. Further $R(P) = 1 - H(P) \approx 0.19$ and $H(K) - R(P) \approx 0.62$. Hence

$$0.62 \leq H(K \mid C) \leq 0.81$$

and

$$0 \leq I(K, C) \leq 0.19.$$

[24]German: Schlüsseläquivokation bzw. -mehrdeutigkeit

Indeed,

$$\begin{aligned} H(K \mid C) &= H(P) + H(K) - H(C) \approx 0.67 \text{ and} \\ I(K, C) &= H(C) - H(P) \approx 0.14. \end{aligned}$$

REMARK 4.8. Interpreting Theorem 4.6:

- The redundancy of P is a (good) upper bound for the key transinformation.
- To get a nonnegative lower bound for the key equivocation $H(K \mid C) \geq H(K) - R(P)$ we need at least as much **key entropy** as redundancy in P.
- If P is uniformly distributed (e.g., random data) then $R(P) = 0$. It follows that $H(K \mid C) = H(K)$, i.e., $I(K, C) = 0$.

EXAMPLE 4.9. Let $P = C = A^n$ and $P = L_n$ for a language L with entropy H_L and redundancy R_L per letter. For n big enough we have

$$H(K \mid C) \geq H(K) - R(P) \approx H(K) - nR_L.$$

Interpretation: If the key entropy $H(K)$ is fixed and n is allowed to grow (e.g., repeated encryption with the same key) then as n increases the entropy of the key is **exhausted**[25].

DEFINITION 4.10. The number

$$n_0 := \lceil \frac{H(K)}{R_L} \rceil$$

is called the **unicity distance**[26].

REMARK 4.11. The higher the redundancy of the language the quicker a key is exhausted.

EXAMPLE 4.12. For $|A| = 26$ and $R_L = 3.45$ (as for the English language) one obtains:

type of the symmetric cryptosystem	$\lvert K \rvert$	$H(K)$	n_0
monoalphabetic substitution	$26! \approx 2^{88.4}$	≈ 88.4	26
permutation of 16-blocks	$16! \approx 2^{44.3}$	≈ 44.3	13
DES	2^{56}	56	17
AES	2^{128}	128	38

If we consider $n = 20$ for the monoalphabetic substitution then the key equivocation

$$H(K \mid C) \geq H(K) - R(P) \approx 88.4 - 20 \cdot 3.45 = 19.4$$

and $2^{19.4} \approx 691802$.

Begin Lect. 8

REMARK 4.13. There are several ways to increase the unicity distance despite short key lengths / small key entropies:

- Reduce the redundancy of P by compressing (zipping) the text.

[25] German: aufgebraucht
[26] German: Unizitätsmaß

Note that with $R_L \to 0$ imply $n_0 \to \infty$. We now estimate the maximum compression factor b where a text of length $n \mapsto$ a text of length $\frac{n}{b}$, $b \geq 1$. The "compressed" language L' has the entropy per letter:

$$H_{L'} := \lim_{n\to\infty} \frac{H(L_n)}{n/b} = b \cdot H_L \leq \lg |A|.$$

Hence $b \leq \frac{\lg |A|}{H_L}$. For L the English language this means that $b \leq \frac{4.70}{1.25} \approx 3.76$.
The following can be much more cheaper than compression:

- Find ways to "cover[27]" the redundancy of P against attackers with limited computing resources: Combination of substitution and FEISTEL ciphers (see [**Wik11c**] and [**MvOV97**, §7.4.1] and Chapter 4).
- Find ways to "bloat[28]" the key entropy against attackers with limited computing resources: Autokey cipher (Figure 4) and pseudo random sequences (see next chapter).

Plaintext:	`das`	`alphabet`	`wechselt`	`staendig`
Key:	`key`	`dasalpha`	`betwechs`	`eltstaen`
Ciphertext:	`neq`	`dlhhlqla`	`xivdwgsl`	`wetwgdmt`

FIGURE 1. Autokey variant of VIGENÈRE's cipher

4.a. Free systems.

DEFINITION 4.14. Analogously one calls

- $H(P \mid C)$ the **plaintext equivocation**[29].
- $I(P, C)$ the **plaintext transinformation**.

LEMMA 4.15. *Setting* $H_0(K) := H(K \mid PC)$. *Then*

(1) $H(P, K) = H(P, C) + H_0(K)$.
(2) $H(K) = H(C \mid P) + H_0(K)$.
(3) $H(K \mid C) = H(P \mid C) + H_0(K)$.

Further:

$$\mathcal{K} \text{ is free } \iff H_0(K) = 0 \iff I(K, PC) = H(K).$$

In particular: The key equivocation and the plaintext equivocation coincide in free *cryptosystems.*

REMARK 4.16. We interpret

(1) $H_0(K)$ as the **unused key entropy**.
(2) $I(K, PC)$ as the **used key entropy**.

[27] German: verschleiern
[28] German: aufblähen
[29] German: Klartextäquivokation bzw. -mehrdeutigkeit

PROOF OF LEMMA 4.15. Verify (1) as an exercise by a straightforward calculation using the definitions. (2) follows from subtracting $H(P)$ from (1) and use $H(K \mid P) = H(K)$, which is a consequence of the independence of P and K. To obtain (3) subtract $H(C)$ from (1) and use $H(P,K) = H(K,C)$ (Lemma 4.1.(1)). The equivalence is an exercise. □

THEOREM 4.17. *Let $\mathcal{K}$ be a* free *cryptosystem. Then:*

(1) $H(P \mid C) = H(K \mid C) = H(P) + H(K) - H(C)$.
(2) $I(P,C) = H(C) - H(K)$.
(3) $H(K) \leq H(C)$.

PROOF.
(1) follows from Lemma 4.15 and Lemma 4.1.(3). For the rest we verify that

$$0 \leq I(P,C) := H(P) - H(P \mid C) \overset{4.15.(3)}{=} H(P) - H(K \mid C) \overset{(1)}{=} H(C) - H(K).$$

□

REMARK 4.18. The statement $H(K) \leq H(C)$ is a generalization of Remark 2.12.(2), but without the regularity assumption:
If $\mathcal{K}$ is regular then: μ_K uniformly distributed implies μ_C uniformly distributed.

COROLLARY 4.19. *Let $\mathcal{K}$ be a free system.*

(1) *If $|P| = |C|$ then $H(P \mid C) \geq H(K) - R(P)$.*
(2) *If $|K| = |C|$ then $I(P,C) \leq R(K)$.*

PROOF. (1) follows from $H(P \mid C) = H(K \mid C)$ and Theorem 4.6. For (2) let $|K| = |C| =: n$. Then

$$I(P,C) = H(C) - H(K) \leq \lg n - H(K) = R(K).$$

□

EXAMPLE 4.20. For P, C, K of Exercise 2.1 we compute:

$$\begin{aligned} H(P) &= \frac{1}{4}\lg 4 + \frac{3}{4}\lg\frac{4}{3} = \frac{1}{4}\cdot 2 + \frac{3}{4}\cdot(2 - \lg 3) = 2 - \frac{3}{4}\lg 3 \approx 0.81 \text{ (maximum is 1).} \\ H(K) &= 1.5 \text{ (maximum is } \lg 3 \approx 1.58). \\ H(C) &\approx 1.85 \text{ (maximum is 2).} \end{aligned}$$

And since $\mathcal{K}$ is free:

$$\begin{aligned} H(P \mid C) &= H(P) + H(K) - H(C) \approx 0.46 \approx 57\% \cdot H(P), \\ I(P,C) &= H(C) - H(K) \approx 0.35 \approx 43\% \cdot H(P), \\ R(P) &\approx 1 - 0.81 \approx 0.19, \text{ and} \\ n_0 &= \lceil \frac{H(K)}{R(P)} \rceil = \lceil \frac{1.5}{0.19} \rceil = 8. \end{aligned}$$

If μ_P would have been uniformly distributed then we recompute:

$$\begin{aligned} H(P) &= 1, \\ H(C) &\approx 1.91, \\ H(P \mid C) &\approx 0.59, \\ I(P,C) &\approx 0.41, \end{aligned}$$

4.b. Perfect systems. The powerful notion of entropy offers us another characterization of perfect systems but *without* unnecessarily assuming that $|P| = |C|$ as in Theorem 2.16.

THEOREM 4.21. *The following statements are equivalent (under the general assumptions made at the beginning §4):*

(1) $\mathcal{K}$ *is perfect for* μ_P*, i.e.,* P *and* C *are independent.*
(2) $I(P,C) = 0$.
(3) $H(P,C) = H(P) + H(C)$.
(4) $H(C \mid P) = H(C)$.
(5) $H(P \mid C) = H(P)$.

In the case of freeness (and hence of regularity by Lemma 2.8) of the cryptosystem the list of equivalences additionally includes:

(6) $H(K) = H(C)$.
(7) $H(K \mid C) = H(P)$.

The last point implies that $H(K) \geq H(P)$.

PROOF. The equivalence of (1)-(5) is trivial. (6) is (4) and the equality $H(K) = H(C \mid P)$ (Lemma 4.15.(2)). (7) is (5) and the equality $H(K \mid C) = H(P \mid C)$ (Lemma 4.15.(3)). □

REMARK 4.22. Compare the statement $H(K) = H(C)$ with that of Lemma 2.13:
... $\mu(e) = \mu(c)$ for all $e \in K, c \in C$.

COROLLARY 4.23 (SHANNON). *Let* $\mathcal{K}$ *be a free cryptosystem. Then* $\mathcal{K}$ *is perfectly secret for* μ_P *and* μ_C *is uniformly distributed if and only if* $|K| = |C|$ *and* μ_K *is uniformly distributed (compare with Theorems 2.15 and 2.14).*

PROOF. $\Longrightarrow$: Set $n := |C|$. Since μ_C is uniformly distributed we know that $H(C) = \lg n$. Freeness of $\mathcal{K}$ implies that $|K| \leq |C| = n$. With the perfect secrecy of $\mathcal{K}$ for μ_P and the freeness of $\mathcal{K}$ we conclude that $H(K) = H(C) = \lg n$ by Theorem 4.21.(6). Hence μ_K is uniformly distributed and $|K| = |C|$ by Theorem 3.14.(1) (giving another proof of Corollary 2.9).
$\Longleftarrow$: Define $n := |K| = |C|$. μ_K uniformly distributed implies that $\lg n = H(K) \leq H(C) \leq \lg n$ by Theorem 4.17.(3). Hence $H(K) = H(C) = \lg n$ and $\mathcal{K}$ is perfectly secure for μ_P by Theorem 4.21.(6). □

CHAPTER 3

Pseudo-Random Sequences

1. Introduction

We want to distinguish random sequences from pseudo-random sequences, where we replace numbers by bits in the obvious way.

First we list what we expect from the word "random":

- Independence.
- Uniform distribution.
- Unpredictability.

Here is a small list of possible sources of random sequences of bits coming from a physical source:

- Throwing a perfect coin.
- Some quantum mechanical systems producing statistical randomness.
- Frequency irregularities of oscillators.
- Tiny vibrations on the surface of a hard disk.
- ... (see `/dev/random` on a LINUX system).

A **pseudo-random generator** is, roughly speaking, an algorithm that takes a usually short ***random* seed** and produces in a *deterministic* way a long pseudo-random sequence.

Now we list some advantages of a pseudo-random generator:

- Simpler than real randomness (produceable using software instead of hardware).
- Reconstructable if the seed is known (e.g., an exchange of a long key can be reduced to the exchange of a short random seed).

The disadvantages include:

- The seed must be random.
- Unpredictability is violated if the seed is known.

Possible applications:

- Test algorithms by simulating random input. An unpredictability would even be undesirable in this case as one would like to be able to reconstruct the input (for the sake of reconstructing a computation or an output).
- Cryptography: Generation of session keys, stream ciphers (the seed is part of the secret key), automatic generations of TANs and PINs, etc.

Begin Lect. 9

2. Linear recurrence equations and pseudo-random bit generators

Let K be a field, $\ell \in \mathbb{N}$, and $c = \begin{pmatrix} c_0 \\ \vdots \\ c_{\ell-1} \end{pmatrix} \in K^{\ell \times 1}$ with $c_0 \neq 0$.

DEFINITION 2.1. The **linear recurrence (or recursion) equation (LRE)** of **degree** $\ell \geq 1$

$$s_{n+\ell} = \begin{pmatrix} s_n & \cdots & s_{n+\ell-1} \end{pmatrix} \cdot \begin{pmatrix} c_0 \\ \vdots \\ c_{l-1} \end{pmatrix} \quad (n \geq 0) \tag{2}$$

defines[1] for the **initial value** $t = \begin{pmatrix} t_0 & \cdots & t_{\ell-1} \end{pmatrix} \in K^{1\times\ell}$ a sequence $s = (s_n)$ in K with $s_i = t_i$ for $i = 0, \ldots, \ell - 1$. We call $t^{(n)} := \begin{pmatrix} s_n & \cdots & s_{n+\ell-1} \end{pmatrix}$ the **n-th state vector** ($t^{(0)} = t$). We write $s = \langle c, t \rangle$.

EXAMPLE 2.2. Taking $K = \mathbb{F}_2$, $\ell = 4$, $c = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}$ and $t = \begin{pmatrix} 1 & 0 & 1 & 0 \end{pmatrix}$ we get:

$$s = \underline{1,0,1,0},1,1,1,1,0,0,0,1,0,0,1,|\,\underline{1,0,1,0},\ldots$$

REMARK 2.3. A linear recurrence equation over $K = \mathbb{F}_2$ (abbreviated by $\mathbb{F}_2$-LRE) of degree ℓ is nothing but an **ℓ-bit Linear Feedback Shift Register (LFSR)** [Wik10f]. It is an example of a *linear* **pseudo-random bit generator (PRBG)** (cf. Example 5.2). It is one among many pseudo-random bit generators [Wik10g].

DEFINITION 2.4.

(1) Define $\chi := \chi_c := x^\ell - c_{\ell-1}x^{\ell-1} - \cdots - c_1 x - c_0 \in K[x]$.
(2) s is called k-periodic ($k \in \mathbb{N}$) if $s_{i+k} = s_i$ for all $i \geq 0$, or equivalently, $t^{(i+k)} = t^{(i)}$ for all $i \geq 0$.
(3) c is called k-periodic ($k \in \mathbb{N}$) if $s = \langle c, t \rangle$ is k-periodic for all $t \in K^{1\times\ell}$.
(4) If s (resp. c) is k-periodic for some $k \in \mathbb{N}$ then denote by $\mathrm{per}(s)$ (resp. $\mathrm{per}(c)$) the smallest such number and call it the **period length**. If such a k does not exist then set $\mathrm{per}\, s := \infty$ (resp. $\mathrm{per}\, c := \infty$).

REMARK 2.5.

(1) $s_{n+\ell} = t^{(n)} \cdot c$.

[1] We identified the resulting 1×1 product matrix with its single entry.

(2) $t^{(n)} = t^{(n-1)} \cdot C = t \cdot C^n$ with

$$C := \begin{pmatrix} 0 & \cdots & \cdots & 0 & c_0 \\ 1 & 0 & \cdots & 0 & c_1 \\ 0 & \ddots & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & c_{\ell-2} \\ 0 & \cdots & 0 & 1 & c_{\ell-1} \end{pmatrix}.$$

(3) $\langle c, t\rangle$ is k-periodic if and only if $t \cdot C^k = t$.
(4) c is k-periodic if and only if $C^k = I_\ell$.
(5) $\operatorname{per}\langle c, t\rangle = \min\{k > 0 \mid t \cdot C^k = t\}$.
(6) $\operatorname{per} c = \min\{k > 0 \mid C^k = I_\ell\}$.
(7) $\langle c, t\rangle$ is k-periodic iff $\operatorname{per}\langle c, t\rangle \mid k$.
(8) $\operatorname{per} c = \operatorname{lcm}\{\operatorname{per}\langle c, t\rangle \mid t \in K^{1\times\ell}\}$.
(9) $\operatorname{per}\langle c, 0\rangle = 1$.
(10) There exists a (row) vector $t \in K^{1\times\ell}$ with $\operatorname{per}\langle c, t\rangle = \operatorname{per} c$.
(11) C is the companion matrix[2] of χ. Hence, χ is the minimal polynomial and therefore also the characteristic polynomial of C (as its degree is ℓ).
(12) C is a regular matrix since $c_0 \neq 0$, i.e., $C \in \mathrm{GL}_\ell(K)$.
(13) $\operatorname{per} c = \operatorname{ord} C$ in $\mathrm{GL}_\ell(K)$.
(14) GL_ℓ and its cyclic subgroup $\langle C\rangle$ generated by the matrix C both act on the vector space $K^{1\times\ell}$. The **orbit**[3] $t \cdot \langle C\rangle = \{t^{(i)} \mid i \geq 0\}$ of t is nothing but the set of all reachable state vectors.
(15) $\operatorname{per}\langle c, t\rangle = |t \cdot \langle C\rangle|$.

PROOF. (1)-(14) are trivial except maybe (10) which is an exercise. To see (15) note that the state vectors $t^{(i)}$ with $0 \leq i < \operatorname{per}\langle c, t\rangle$ are pairwise distinct: $t^{(i)} = t^{(j)}$ for $0 \leq i \leq j < \operatorname{per}\langle c, t\rangle$ means that $tC^i = tC^j$ and hence $tC^{j-i} = t$ with $j - i < \operatorname{per}\langle c, t\rangle$. Finally this implies that $j = i$. □

2.a. Linear algebra. Let K be a field, V a nontrivial finite dimensional K vector space, $\varphi \in \mathrm{End}_K(V)$, and $0 \neq v \in V$.

Recall, the **minimal polynomial** m_φ is the unique *monic*[4] generator of the principal ideal $I_\varphi := \{f \in K[x] \mid f(\varphi) = 0 \in \mathrm{End}_K(V)\}$, the so-called **vanishing ideal** of φ.

Analogously, the **minimal polynomial** $m_{\varphi,v}$ **with respect to** v is the unique *monic* generator of the principal ideal $I_{\varphi,v} := \{f \in K[x] \mid f(\varphi)v = 0 \in V\}$, the so-called **vanishing ideal of** φ **with respect to** v.

EXERCISE 2.6. For $0 \neq v \in V$ let $U_{\varphi,v} := \langle \varphi^i(v) \mid i \in \mathbb{N}_0\rangle \leq V$. Then
(1) $m_{\varphi,v} = m_{\varphi_{|U_{\varphi,v}}}$.
(2) $\dim_K U_{\varphi,v} = \min\{d \in \mathbb{N} \mid (v, \varphi(v), \ldots, \varphi^d(v)) \text{ are } K\text{-linearly dependent}\} \geq 1$.

[2]German: Begleitmatrix
[3]German: Bahn
[4]German: normiert

(3) $\deg m_{\varphi,v} = \dim_K U_{\varphi,v}$.
(4) $m_\varphi = \operatorname{lcm}\{m_{\varphi,v} \mid 0 \neq v \in V\}$. This gives an algorithm to compute the minimal polynomial of φ as the lcm of at most n minimal polynomials $m_{\varphi,v_1}, \ldots, m_{\varphi,v_\ell}$, where $\ell = \dim_K V$.
(5) $\alpha \in \operatorname{End}_K(V)$ is an automorphism if and only if $m_\alpha(0) \neq 0 \in K$. This gives an algorithm to compute the inverse of α.

DEFINITION 2.7. Let $0 \neq f \in K[x]$. If $f(0) \neq 0$ define

$$\operatorname{ord} f := \min\{k > 0 : f \mid x^k - 1\} \text{ or } \infty.$$

If $f(0) = 0$, then write $f = x^r \bar{f}$ with $\bar{f}(0) \neq 0$ and define

$$\operatorname{ord} f := \operatorname{ord} \bar{f}.$$

DEFINITION 2.8. Let $\alpha \in \operatorname{Aut}_K(V)$ and $\langle\alpha\rangle$ the cyclic subgroup of $\operatorname{Aut}_K(V)$ generated by α. By

$$\operatorname{ord} \alpha := |\langle\alpha\rangle|$$

denote the **order** of the group element α. For $v \in V$ denote the **orbit** of v under the action of the cyclic subgroup $\langle\alpha\rangle$ as usual by

$$\langle\alpha\rangle v := \{\alpha^i(v) \mid i \in \mathbb{N}_0\}.$$

PROPOSITION 2.9.

(1) $\operatorname{ord} m_\alpha = \operatorname{ord} \alpha$.
(2) $\operatorname{ord} m_{\alpha,v} = |\langle\alpha\rangle v|$ *for* $v \neq 0$.
(3) *If* m_α *is irreducible then* $|\langle\alpha\rangle v| = \operatorname{ord} m_\alpha$ *for all* $v \neq 0$.
(4) *If* K *is finite and* m_α *irreducible then* $\operatorname{ord} m_\alpha \mid |V| - 1$.
(5) *If there exists a vector* $v \in V$ *with* $\langle\alpha\rangle v = V \setminus \{0\}$ *then* m_α *is irreducible.*

PROOF. (1) follows from the equivalence

$$\alpha^k = \mathrm{id}_V \iff (x^k - 1)(\alpha) = 0 \iff m_\alpha \mid x^k - 1.$$

(2) If $|\langle\alpha\rangle v| < \infty$ then there exists $0 \leq i < j$ with $\alpha^i(v) = \alpha^j(v)$. Hence, $\alpha^{j-i}(v) = v$. Therefore (even if $|\langle\alpha\rangle v| = \infty$)

$$\begin{aligned} |\langle\alpha\rangle v| &= \min\{k > 0 \mid \alpha^k(v) = v\} \\ &= \min\{k > 0 \mid (\alpha^k - \mathrm{id}_V)(v) = 0 \in V\} \\ &= \min\{k > 0 : m_{\alpha,v} | x^k - 1\} \\ &= \operatorname{ord} m_{\alpha,v}. \end{aligned}$$

(3) If m_α is irreducible then the statements $m_{\alpha,v} \mid m_\alpha$ and $\deg m_{\alpha,v} > 0$ are equivalent to $m_{\alpha,v} = m_\alpha$. (2) completes the argument.
(4) follows from (3) which implies that the orbits of $\langle\alpha\rangle$ in $V \setminus \{0\}$ are all of the same length.
(5) First note that if $\langle\alpha\rangle v = V \setminus \{0\}$ for one $v \in V$ then also for all $v \in V \setminus \{0\}$ (being all elements of the orbit). In particular, $U_{\alpha,v} = V$ and hence $m_{\alpha,v} = m_\alpha$ for all $v \neq 0$.

If $m_\alpha = gh$ we want to prove that either $m_\alpha \mid g$ or $m_\alpha \mid h$: For $v \neq 0$ we obtain $0 = m_{\alpha,v}(\alpha)(v) = (gh)(\alpha)(v) = (g(\alpha) \circ h(\alpha))(v) = g(\alpha)(h(\alpha)(v))$. Hence, either $h(\alpha)(v) = 0$ or $v' := h(\alpha)(v) \neq 0$ and $g(\alpha)(v') = 0$. In other words, either $m_\alpha = m_{\alpha,v} \mid h$ or $m_\alpha = m_{\alpha,v'} \mid g$. □

2.b. Period length. Let $V = K^{1\times\ell}$ and $t \in V \setminus \{0\}$. Identify $\mathrm{Aut}_K(V)$ with $\mathrm{GL}_\ell(K)$ in the obvious way. Viewing the regular matrix C (of Remark 2.5.(2)) as an automorphism we get $\chi = m_C$. Define $\chi_t := m_{C,t}$.

COROLLARY 2.10. *From the last theorem we easily deduce*

(1) $\operatorname{per} c = \operatorname{ord} \chi$.
(2) $\operatorname{per}\langle c, t\rangle = \operatorname{ord} \chi_t$ *for* $t \neq 0$.
(3) *If* χ *is irreducible then* $\operatorname{per}\langle c, t\rangle = \operatorname{per} c$ *for all* $t \neq 0$.

And in case $K = \mathbb{F}_q$ *(i.e., finite with* q *elements):*

(4) *If* χ *is irreducible then* $\operatorname{per} c \mid q^\ell - 1$.
(5) *If* $\operatorname{per} c = q^\ell - 1$ *then* χ *is irreducible.*

EXAMPLE 2.11. Let $K = \mathbb{F}_2$.

(1) Consider the 3-bit LFSR (i.e., of degree $\ell = 3$) and maximum possible period length $q^\ell - 1 = 8 - 1 = 7$.

c^{tr}	χ	irred.	s	orbit lengths
$(1,0,0)$	x^3+1	false	100|100, 110|110, 1|111	$3+3+1$
$(1,1,0)$	x^3+x+1	true	1001011|100	7
$(1,0,1)$	x^3+x^2+1	true	1001110|100	7
$(1,1,1)$	x^3+x^2+x+1	false	1001|100, 01|010, 1|111	$4+2+1$

(2) Consider the 4-bit LFSR (i.e., of degree $\ell = 4$) and maximum possible period length $q^\ell - 1 = 16 - 1 = 15$.

c^{tr}	χ	irred.	s	orbit lengths
$(1,1,0,0)$	x^4+x+1	true	100010011010111|1000	15
$(1,0,0,1)$	x^4+x^3+1	true	100011110101100|1000	15
$(1,1,1,1)$	$x^4+x^3+x^2+x+1$	true	10001|1000, 01001|0100	$5+5+5$
			10100|1010	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

DEFINITION 2.12. We call a linear recurrence equation **irreducible** if χ is irreducible. If moreover $K = \mathbb{F}_q$ is a finite field then we call the LRE **transitive** if $\operatorname{per} c = q^\ell - 1$, where ℓ is its degree.

REMARK 2.13. There are faster ways to compute $\operatorname{per} c$ and to decide the transitivity of LREs. Consider for example $c = (1,1,0,0)^{\mathrm{tr}}$ with $\chi = x^4 + x + 1$ in the above table. Since χ is irreducible we know from the above corollary that $\operatorname{ord} \chi \mid 15$. It is obvious that $\chi \nmid x^k - 1$ for $k = 1, 3, 5$ (these are the divisors of 15). Hence $\operatorname{per} c = \operatorname{ord} \chi = 15$, the maximal possible period length, i.e., the corresponding LFSR is transitive.

EXERCISE 2.14. Classify all irreducible 4-bit LFSRs. How many of them are transitive?

REMARK 2.15. The **MERSENNE twister** [Wik10j] is a modern pseudo-random bit generator with an impressive period length.

Begin Lect. 10

3. Finite fields

3.a. Field extensions. Recall, if $K \leq L$ is a field extension, then L is a K-vector space. The degree of the field extension $K \leq L$ is defined as the dimension of L as a K-vector space:

$$[L : K] := \dim_K L.$$

For a 2-step field extension $K \leq L \leq M$ the **degree formula**

$$[M : K] = [M : L] \cdot [L : K]$$

is a direct consequence of the definition.

In what follows we only deal with **finite** field extensions $K \leq L$, i.e., where

$$d := [L : K] < \infty.$$

For any element $\alpha \in L$ the $d + 1$ elements $1, \alpha, \ldots, \alpha^d$ are always K-linearly dependent, which leads us to the next definition:

DEFINITION 3.1. Let $K \leq L$ be a finite field extension an $\alpha \in L$. The unique *monic* generator of the **vanishing (principal) ideal** $I_{\alpha,K} := \{f \in K[x] \mid f(\alpha) = 0\}$ is called the **minimal polynomial of** α **over the ground field** K, and denoted by $m_{\alpha,K}$, or simply m_α, if no confusion can occur about the ground field K.

REMARK 3.2. In the above definition the field L can be replaced by a (finite dimensional) K-algebra L. This gives a common generalization of the two definitions m_φ and $m_{\alpha,K}$ above, where in the former case $L = \mathrm{End}_K(V)$.

REMARK 3.3. Let $K \leq L$ be a finite field extension and $\alpha \in L$. The minimal polynomial $m_\alpha = m_{\alpha,K}$ satisfies the following properties:

(1) $f(\alpha) = 0 \iff m_\alpha \mid f$.
(2) m_α is irreducible in $K[x]$ and $1 \leq \deg m_\alpha \leq d$.
(3) If an irreducible monic polynomial $f \in K[x]$ satisfies $f(\alpha) = 0$ then $f = m_\alpha$.

We now recall **KRONECKER's construction** of field extensions:

PROPOSITION 3.4. *Let K be a field and $f \in K[x]$. The residue class K-algebra $L := K[x]/\langle f \rangle$ is a field if and only if f is irreducible. In this case $[L : K] = \deg f$ and $m_{\bar{x}} = m_{K,\bar{x}} = f$, where $\bar{x} := x + \langle f \rangle \in K[x]/\langle f \rangle$.*

EXAMPLE 3.5.

- Let $f := x - a$ for $a \in K$. Then $K[x]/\langle f \rangle \cong K$.

- Let K be a subfield of $\mathbb{R}$, e.g., $K = \mathbb{Q}$ or $K = \mathbb{R}$. Then $f = x^2 + 1$ is irreducible and the field $K[x]/\langle f\rangle$ is an extension of degree 2 over K with $(1, \bar{x})$ as a K-basis satisfying $\bar{x}^2 = -1$.
- Let $K = \mathbb{F}_2$ and $f = x^3 + x + 1$. The degree 3 polynomial f is irreducible since it has no roots in its field of definition $\mathbb{F}_2$. The field $L := \mathbb{F}_2[x]/\langle f\rangle$ is an extension of degree 3 over $\mathbb{F}_2$ with $(1, x, x^2)$ as an $\mathbb{F}_2$-basis[5] and elements

$$L = \{0, 1, x, x^2, 1 + x, x + x^2, 1 + x + x^2, 1 + x^2\}.$$

EXERCISE 3.6. Prove that the $\mathbb{F}_2$-algebra $L := \mathbb{F}_2[x]/\langle x^4+x^2+1\rangle$ is *not* a field. Find all nonzero noninvertible elements in L. Show that x is invertible in L. Determine the minimal polynomial $m_{\mathbb{F}_2,x}$ and show that it is reducible. Find all invertible elements different from 1 and invert them either by using their minimal polynomial (cf. Exercise 2.6) or by using the **extended EUCLIDian algorithm**.

3.b. Order of field elements.

REMARK 3.7. Let K be a field.

(1) Let $f \in K[x]$ be irreducible and $f \neq x$. Then $\bar{x} \neq 0$ in $L := K[x]/\langle f\rangle$ and $\operatorname{ord} f = \operatorname{ord} \bar{x}$ in the multiplicative group $L^* := (L \setminus \{0\}, \cdot)$.
(2) Let $K \leq L$ be a (finite) field extension and $\alpha \in L^*$. Then $\operatorname{ord} m_{\alpha,K} = \operatorname{ord} \alpha$ in the multiplicative group L^*.

PROOF. (1) $\bar{x}^k = 1 \iff f \mid x^k - 1$.
(2) $\alpha^k = 1 \iff \alpha$ is a root of $x^k - 1 \iff m_{\alpha,K} \mid x^k - 1$. □

COROLLARY 3.8. *Let $K = \mathbb{F}_q$ be the finite field with q elements and $f \in \mathbb{F}_q[x]$ of degree n. Then*

(1) *f irreducible $\implies \operatorname{ord} f \mid q^n - 1$.*
(2) *$\operatorname{ord} f = q^n - 1 \implies f$ irreducible.*

PROOF. We can assume that f is monic.
Case $f(0) \neq 0$:
Take $V = K^{1\times n}$ and $\beta \in \operatorname{Aut}_K(V)$ with $m_\beta = f$ (e.g., $\beta : t \mapsto t \cdot C$, where C is the companion matrix of f, which is due to $f(0) \neq 0$ regular). Now apply Proposition 2.9.(4),(5).
Case $f(0) = 0$:
(1) $f = x$ is the only irreducible monic polynomial with $f(0) = 0$. By definition $\operatorname{ord} x \overset{2.7}{:=} \operatorname{ord} 1 = 1$.
(2) Let $f = x^r \bar{f}$ with $\deg \bar{f} \leq n - 1$. From (1) we deduce that $\operatorname{ord} f = \operatorname{ord} \bar{f} \mid q^{n-1} - 1$. Hence $\operatorname{ord} f \neq q^n - 1$. □

DEFINITION 3.9. Let $K = \mathbb{F}_q$ be a *finite* field with q elements and L a finite field extension of $\mathbb{F}_q$. We call

- a degree n polynomial $f \in \mathbb{F}_q[x]$ **primitive** if $\operatorname{ord} f = q^n - 1$. Primitive polynomials are *irreducible* by the above corollary.

[5] We suppress the $\bar{\cdot}$.

- an element $\alpha \in L^*$ **primitive**[6] if $\operatorname{ord}\alpha = |L^*|$, i.e., if $L^* = \langle\alpha\rangle := \{\alpha^i \mid i \in \mathbb{N}_0\}$.

REMARK 3.10. Let $K = \mathbb{F}_q$ a finite field with q elements.

(1) Let $f \in \mathbb{F}_q[x]$ be a primitive polynomial and $L := \mathbb{F}_q[x]/\langle f\rangle$. Then $\bar{x}$ is a primitive element of L.
(2) If $\mathbb{F}_q \leq L$ is a finite field extension then $\alpha \in L^*$ is primitive iff $m_{\alpha,K}$ is a primitive polynomial (of degree $[L : K]$).

PROOF. (1) Define $n := \deg f$. Then $|L| = q^n$ and $|L^*| = q^n - 1$. Now use that $\operatorname{ord}\bar{x} \overset{3.7.(1)}{=} \operatorname{ord} f = q^n - 1$.
(2) Define $n := [L : K]$. If $L^* = \langle\alpha\rangle$ then $\operatorname{ord}\alpha = |L^*| = q^n - 1$. Set $f = m_{\alpha,K}$. Remark 3.7.(2) implies that $\operatorname{ord} f = \operatorname{ord}\alpha = q^n - 1$. Using that $\deg f \leq n$ and that $\operatorname{ord} f \mid q^{\deg f} - 1$ (Corollary 3.8.(1)) we conclude that $n = \deg f$ and finally the primitivity of $f = m_{\alpha,K}$. □

EXERCISE 3.11. Let $L := \mathbb{F}_2[x]/\langle f\rangle$ and

(1) $f := x^3 + x + 1$. Prove that f is a primitive polynomial, or equivalently, that $\bar{x} \in L$ is a primitive element, i.e., $L^* = \langle\bar{x}\rangle$.
(2) $f := x^4 + x^3 + x^2 + x + 1$. First prove that L is a field. Prove that f is an imprimitive polynomial, or equivalently, that $\bar{x} \in L$ is an imprimitive[7] element, i.e., $L^* \neq \langle\bar{x}\rangle$

3.c. Some field theory. Let K be a field. Recall:

- $K[x]$ is a **GAUSSian domain**[8]. A more suggestive name is **unique factorization domain (UFD)** or simply **factorial domain**[9].
- For $f \in K[x]$ the following holds:
  - $f(a) = 0 \iff (x - a) \mid f$.
  - $f(a) = f'(a) = 0 \iff (x - a)^2 \mid f$, where f' is the derivative of f w.r.t. x.
- The **characteristic** of K is defined as $\operatorname{char} K = \min\{c \in \mathbb{N} \mid c \cdot 1 = 0\}$ or 0. i.e., it is the unique nonnegative generator of the principal ideal $\ker(\mathbb{Z} \to K, c \mapsto c \cdot 1) \lhd \mathbb{Z}$. $\operatorname{char} K$ is either zero or a prime number.
- If $\operatorname{char} K = p > 0$ then $\mathbb{F}_p \leq K$. Else $\mathbb{Q} \leq K$. The fields $\mathbb{F}_p \cong \mathbb{Z}/p\mathbb{Z}$ resp. $\mathbb{Q} = \operatorname{Quot}(\mathbb{Z})$ are therefore called **prime fields**. Each field contains exactly one prime field as the smallest subfield.
- For a finite field extension $K \leq L$ define for an element $\alpha \in L$ the smallest subring of L containing K and α

$$K[\alpha] := \{\sum_{i=1}^{n} \lambda_i \alpha^i \mid n \in \mathbb{N}_0, \lambda_i \in K\}.$$

[6]Unfortunately, this name conflicts the notion of primitive elements of (algebraic) field extensions.
[7]Although $\bar{x}$ is a primitive element of the field extension $\mathbb{F}_2 \leq \mathbb{F}_2[\bar{x}] = \mathbb{F}_2[x]/\langle x^4 + x^3 + x^2 + x + 1\rangle$.
[8]German: GAUSSscher Bereich
[9]German: Faktorieller Bereich

- The vanishing ideal $I_{\alpha,K} = \langle m_{\alpha,K} \rangle$ is the kernel of the ring epimorphism $K[x] \to K[\alpha],\ x \mapsto \alpha$. Hence $K[\alpha] \cong K[x]/\langle m_{\alpha,K} \rangle$ as K-algebras and $K(\alpha) := K[\alpha]$ is a *field* with $[K(\alpha) : K] = \deg m_{\alpha,K}$. The field $K(\alpha)$ is called the **intermediate field**[10] generated by α.

Begin Lect. 11

EXAMPLE 3.12. Let $L := \mathbb{F}_2[x]/\langle x^4 + x^3 + x^2 + x + 1 \rangle$ as in Exercise 3.11.(2). The element $\alpha := \bar{x}^3 + \bar{x}^2 + 1$ satisfies $\alpha^2 = \bar{x}^3 + \bar{x}^2$. Hence $m_{\mathbb{F}_2,\alpha} = x^2 + x + 1$ and $\mathbb{F}_2(\alpha) = \mathbb{F}_2[\alpha]$ is an intermediate field of degree $[\mathbb{F}_2(\alpha) : \mathbb{F}_2] = 2$: $K(\alpha) = K + K\alpha = \{0, 1, \alpha, 1 + \alpha\}$.

PROPOSITION 3.13. *Let $K \leq L$ a field extension and $f \in K[x]$ a monic irreducible with $f(\alpha) = f(\alpha') = 0$ for two elements $\alpha, \alpha' \in L$. Then $K(\alpha) \cong K(\alpha')$ as K-algebras (or as fields* over *K).*

PROOF. $f(\alpha) = 0$ and f monic irreducible implies $f = m_{\alpha,K}$ by Remark 3.3.(3). The same ist true for α'. Hence $K(\alpha) \cong K[x]/\langle f \rangle \cong K(\alpha')$. □

Now we recall the notion of the splitting field of a polynomial $f \in K[x]$.

DEFINITION 3.14. Let $K \leq L$ and $f \in K[x]$. We say

- f **splits over**[11] L if f splits as a product of linear factors (when viewed as a polynomial) over $L[x]$.
- L is a **splitting field**[12] of f over K if f splits over L and L is minimal with this property.

REMARK 3.15. If f splits over L with roots $\alpha_1, \ldots, \alpha_n$ then $K(\alpha_1, \ldots, \alpha_n) = K[\alpha_1, \ldots, \alpha_n]$ is a splitting field of f contained in L.

THEOREM 3.16. *For each $f \in K[x]$ there exists a splitting field, unique up to K-isomorphism.*

PROOF. The above remark shows that it is enough to construct a field $M \geq K$ over which f splits. We may assume[13] that f is irreducible (otherwise we do the following for all factors). The field $L := K[x]/\langle f \rangle$ contains (at least) one root of f which is $\alpha := \bar{x}$. Hence $f = (x - \alpha)\bar{f} \in L[x]$. Proceed by induction on $\deg f$. □

We thus talk about *the* splitting field of f over K.

3.d. Finite fields. Recall, $\operatorname{char} K = p > 0$ means that $p\alpha = 0$ for any α in any field extension $L \geq K$.

LEMMA 3.17. *Let K be a field with $\operatorname{char} K = p > 0$. For each $i \in \mathbb{N}_0$ the map $\varphi_i : K \to K, x \mapsto x^{p^i}$ is an automorphism of K which fixes the prime field $\mathbb{F}_p$. It is called the i-**th** **FROBENIUS** **automorphism** of K.*

[10] German: Zwischenkörper
[11] German: zerfällt über
[12] German: Zerfällungskörper
[13] It is not clear how to make this step constructive. From the constructive point of view, we have just assumed that we have an algorithm to factor polynomials in irreducibles.

PROOF. Since $\varphi_i = \varphi_1^i$ it suffices to consider $i = 1$. Of course $1^p = 1$ and $(\alpha\beta)^p = \alpha^p\beta^p$ in any field of any characteristic. We therefore only need to prove the "characteristic p formula"

$$(\alpha \pm \beta)^p = \alpha^p \pm \beta^p.$$

Indeed the **binomial theorem** $(\alpha + \beta)^p = \sum_{k=0}^{p} \binom{p}{k} \alpha^k \beta^{p-k}$ implies the statement since $\binom{p}{0} = \binom{p}{p} = 1$ and $p \mid \binom{p}{k}$ for $0 < k < p$. Proving the bijectiveness is an exercise. □

THEOREM 3.18. *Let K be a field of prime characteristic p, $n \in \mathbb{N}$, and $q := p^n$. Consider $f := x^q - x \in \mathbb{F}_p[x]$.*

1. *f splits over K if and only if K contains exactly one subfield with q elements.*
2. *K is the splitting field of f if and only if $|K| = q$.*

PROOF. Set

$$N := \{\alpha \in K \mid f(\alpha) = 0\}.$$

Hence $|N| \leq q$. Since f has no multiple roots ($f' = -1 \implies \gcd(f, f') = 1$) we conclude that f splits of $K \iff |N| = q$. The previous lemma implies that N is an intermediate field of K: $\alpha, \beta \in N \implies (\alpha - \beta)^q = \alpha^q - \beta^q = \alpha - \beta$ and we are done with the forward implication in (1).
Now let $M \leq K$ be a subfield with q elements. Since $|M^*| = q - 1$ it follows that every $\alpha \in M^*$ is a root of $f = x(x^{q-1} - 1)$, hence $M \leq N$. From $|N| \leq q$ we conclude that $M = N$. This proves the uniqueness of $M = N$ and that f splits over a field K containing N.
(2) follows from (1) and the minimality of the splitting field. □

COROLLARY 3.19.

1. *If K is a finite field then* $\operatorname{char} K = p > 0$ *and $|K| = p^n$.*
2. *For each prime power $q = p^n$ there exists up to $\mathbb{F}_p$-isomorphism exactly one field with q elements.*

PROOF.

1. Since K is finite its characteristic $\operatorname{char} K = p > 0$ is prime. Hence, $\mathbb{F}_p$ is the prime field of K and, in particular, K is an $\mathbb{F}_p$-vector space. So $|K| = p^n$, where $n = [K : \mathbb{F}_p] := \dim_{\mathbb{F}_p} K$.
2. follows from Theorem 3.18.(2) and the uniqueness of the splitting field applied to the polynomial $f = x^q - x$.

□

We have been referring to this field as $\mathbb{F}_q$. Now we can say "the field $\mathbb{F}_q$".

COROLLARY 3.20. *The finite field $\mathbb{F}_q = \mathbb{F}_{p^n}$ contains the unique subfield (isomorphic to) $\mathbb{F}_{p^d}$ if and only if $d \mid n$. I.e.*

$$\mathbb{F}_{p^d} \leq \mathbb{F}_{p^n} \iff d \mid n.$$

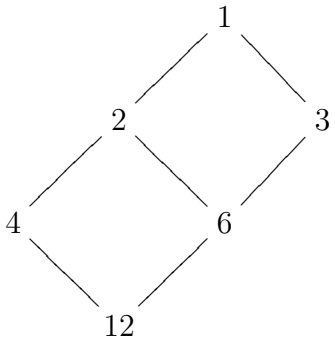
In other word, the **subfield lattice**[14] of $\mathbb{F}_{p^n}$ is isomorphic to the lattice of divisors of n (regardless of the prime number p).

PROOF. Let $K \leq \mathbb{F}_q = \mathbb{F}_p^n$. Then K has characteristic p and the prime field $\mathbb{F}_p$ of $\mathbb{F}_q$ is the prime field of K. Hence $K = \mathbb{F}_{p^d}$ for some $1 \leq d \leq n$. The degree formula $n = [\mathbb{F}_{p^n} : \mathbb{F}_p] = [\mathbb{F}_{p^n} : \mathbb{F}_{p^d}] \underbrace{[\mathbb{F}_{p^d} : \mathbb{F}_p]}_{d}$ implies that $d \mid n$.

Now we proof the converse. Let $d \mid n$. First note that $\alpha^{p^d} = \alpha$ implies $\alpha^{p^n} = \alpha$. In particular, the roots of $x^{p^d} - x$ are all roots of $x^{p^n} - x$. So $x^{p^d} - x$ splits over $\mathbb{F}_{p^n}$. Theorem 3.18.(1) then states that $\mathbb{F}_{p^n}$ contains the unique field with p^d elements. □

EXAMPLE 3.21.

- $\mathbb{F}_4 \not\leq \mathbb{F}_8$, but $\mathbb{F}_4 < \mathbb{F}_{16}$.
- The subfield lattice of $\mathbb{F}_{p^{12}}$ is isomorphic to the divisor lattice of 12



3.e. Irreducible polynomials over finite fields. We know that we can construct the finite field $\mathbb{F}_{p^n}$ as the splitting field of $x^{p^n} - x$. This would eventually involve iterated KRONECKER constructions. So it is natural to ask if we can get the job done with just one KRONECKER construction. This question is equivalent to asking if there exists an irreducible polynomial of degree n over $\mathbb{F}_p$.

EXERCISE 3.22. Let K be a field and $f \in K[x]$ with $f(0) \neq 0$. Then $\operatorname{ord} f \mid k \iff f \mid x^k - 1$.

COROLLARY 3.23. *Let $K = \mathbb{F}_q$ with $q = p^n$.*

(1) *Each irreducible polynomial $f \in \mathbb{F}_q[x]$ with $\deg f = n$ is* ***square free***[15][16] *and splits over $\mathbb{F}_{q^n}$.*
(2) *$\mathbb{F}_q[x]/\langle f \rangle \cong \mathbb{F}_{q^n}$ for all irreducible $f \in \mathbb{F}_q[x]$ with $\deg f = n$.*

PROOF.

[14] German: Zwischenkörperverband
[15] German: quadratfrei
[16] i.e., it has no multiple roots over its splitting field.

(1) Corollary 3.8.(1) states that $\operatorname{ord} f \mid q^n - 1$ which is equivalent to $f \mid x^{q^n} - x = x(x^{q^n-1} - 1)$ by the above exercise. But the polynomial $x^{q^n} - x$ splits with distinct roots over $\mathbb{F}_{q^n}$ by Theorem 3.18 and its proof (applied to q^n). The same holds for the divisor f.
(2) The statement follows from $|\mathbb{F}_q[x]/\langle f\rangle| = q^n$ and Theorem 3.18.(2).

□

Begin Lect. 12

DEFINITION 3.24. Set

$$A(d) = A(d,q) := |\{f \in \mathbb{F}_q[x] : f \text{ irreducible monic with } \deg f = d\}|.$$

THEOREM 3.25. *For $K = \mathbb{F}_q$ the numbers $A(d)$ satisfy*

$$(*) \qquad \sum_{d|n} dA(d) = q^n.$$

In particular, $A(1) = q$ and $A(d) = \frac{q^d - q}{d}$ if d is prime.

PROOF. Set $L := \mathbb{F}_{q^n}$. We know that $\mathbb{F}_{q^d} \leq L \iff d \mid n$. First note that a polynomial $f \in \mathbb{F}_q[x]$ with $\deg f \mid n$ is the minimal polynomial of d elements in L. This follows from Corollary 3.23.(1). f is then the minimal polynomial of all its d roots and of no other element in L. Now recall that $\deg m_{\alpha,\mathbb{F}_q} \mid n$ for all $\alpha \in L$ by the degree formula. This finishes the proof. □

EXAMPLE 3.26. Now we list all minimal polynomials (= irreducible monic polynomials) together with their degrees for the following fields:

- $K = \mathbb{F}_4$

$m_{\alpha,\mathbb{F}_2}$	x	$x+1$	x^2+x+1	
deg	1	1	2	$\sum = 4$

- $K = \mathbb{F}_{16}$

$m_{\alpha,\mathbb{F}_2}$	x	$x+1$	x^2+x+1	x^4+x+1	x^4+x^3+1	$x^4+x^3+x^2+x+1$	
deg	1	1	2	4	4	4	$\sum = 16$

REMARK 3.27. We list the following facts without proof:

(1) Asymptotically: $A(d) \sim \frac{q^d}{d}$.
(2) Since formula (*) is an inclusion-exclusion counting formula over a lattice one can use the **MÖBIUS function**

$$\mu(d) := \begin{cases} 1 & , \ d = 1 \\ 0 & , \ d \text{ is not square free} \\ (-1)^k & , \ d \text{ is the product of } k \text{ distinct primes} \end{cases}$$

to "invert" it:

$$A(n) = \frac{1}{n} \sum_{d|n} \mu(d) q^{\frac{n}{d}}.$$

EXAMPLE 3.28. $A(20,q) = \frac{1}{20}(q^{20} - q^{10} - q^4 + q^2)$.

3.f. Primitive polynomials. Counting primitive polynomials is much simpler.

PROPOSITION 3.29. *Let K be a field and U a* finite *subgroup of K^*. Then U is cyclic.*

PROOF. Exercise. □

COROLLARY 3.30. *Let φ denote* **EULER*'s totient function***[17]

(1) *There are exactly $\varphi(q-1)$ primitive elements in $\mathbb{F}_q$.*
(2) *There are exactly $\frac{\varphi(q^d-1)}{d}$ primitive monic polynomials of degree d in $\mathbb{F}_q[x]$.*

PROOF.

(1) Proposition 3.29 implies that the multiplicative group $\mathbb{F}_q^*$ is cyclic, in particular $\mathbb{F}_q^* = \{a^0, a^1, \dots, a^{q-2}\}$. a^i is primitive $\iff \gcd(i, q-1) = 1$.
(2) Every primitive $f \in \mathbb{F}_q[x]$ of degree d is the minimal polynomial of exactly d primitive elements in $L = \mathbb{F}_{q^d}$. This follows from the irreducibility of f (Corollary 3.8.(2)) and Remark 3.10 using (1) and the same argument as in the proof of Theorem 3.25.

□

EXAMPLE 3.31. In the exercises we will see how to use the FROBENIUS automorphisms to construct irreducible and primitive polynomials over finite fields. In the following two examples we mainly sum up computations we did before:

(1) $\mathbb{F}_4 = \mathbb{F}_{2^2} = \{0, 1, \omega, 1+\omega\}$ with $\omega^2 + \omega + 1 = 0$. ω and $1+\omega$ are all primitive elements of $\mathbb{F}_4$ and their minimal polynomial $x^2 + x + 1$ the only irreducible and primitive polynomial of degree 2 over $\mathbb{F}_2$.
(2) There are $\varphi(16-1) = 8$ primitive elements in $\mathbb{F}_{16} = \mathbb{F}_{2^4} = \mathbb{F}_{4^2}$. Hence there are $\frac{8}{2} = 4$ primitive polynomials

$$x^2 + x + \omega, \quad x^2 + x + \omega^2, \quad x^2 + \omega x + \omega, \quad x^2 + \omega^2 x + \omega^2$$

of degree 2 over $\mathbb{F}_4$ and $\frac{8}{4} = 2$ primitive polynomials

$$x^4 + x + 1, \quad x^4 + x^3 + 1$$

of degree 4 over $\mathbb{F}_2$. The polynomial $x^4 + x^3 + x^2 + x + 1 = \frac{x^5-1}{x-1}$ is the only irreducible imprimitive polynomial of degree 4 over $\mathbb{F}_2$.

EXAMPLE 3.32. We compare $A(d)$ and the number of primitive polynomials of degree d over $\mathbb{F}_2$:

d	1	2	3	4	5	6	7	8	9	10	11	16
$A(d,2)$	2	1	2	3	6	9	18	30	56	99	186	4080
primitive	2	1	2	2	6	6	18	16	48	60	176	2048

[17] German: EULERsche φ-Funktion

We will be using primitive polynomials χ over finite fields to construct pseudo-random sequences $s = \langle c, t\rangle$ of maximal possible period lengths (cf. Definition 2.4). Since $\chi = \chi_c$ will be part of the secret key, we need to know how to randomly choose primitive polynomials. The idea will be to randomly choose a polynomial and then to test its primitiveness.

4. Statistical tests

Let X_n and U_n denote random variables with values in $\{0,1\}^n$ where U_n is the *uniformly distributed* one. Note that any map $f : \{0,1\}^n \to \{0,1\}^m$ induces a random variable $Y_m := f \circ X_n$.

EXAMPLE 4.1. Define linear pseudo-random bit **generator** G as the map

$$G : \left\{ \begin{array}{ccc} \{0,1\}^{2\ell} & \to & \{0,1\}^\bullet \\ (c,t) & \mapsto & \langle c,t\rangle \end{array} \right. ,$$

where we consider the pair (c,t) as an ℓ-bit LFSR given by $c \in \mathbb{F}_2^{\ell\times 1}$ and initial value $t \in \mathbb{F}_2^{1\times\ell}$. By truncation to first n bits we get a map

$$G_n : \left\{ \begin{array}{ccc} \{0,1\}^{2\ell} & \to & \{0,1\}^n \\ (c,t) & \mapsto & \langle c,t\rangle_{i=0,\ldots,n-1} \end{array} \right.$$

Define the random variable

$$X := G \circ U_{2\ell}$$

In words, X is a (linear) pseudo-random bit **generator** with *random* seed. Define the finite random variable

$$X_n := G_n \circ U_{2\ell}$$

with values in $\{0,1\}^n$.

Our goal will be to compare X_n with U_n.

DEFINITION 4.2. A **(polynomial) statistical test** is a polynomial probabilistic algorithm

$$A : \left\{ \begin{array}{ccc} \{0,1\}^\bullet & \to & \{0,1\} \\ s & \mapsto & A(s) \end{array} \right. .$$

We say the $s \in \{0,1\}^n$ **passes** the test A only if $A(s) = 1$.

REMARK 4.3. The composition $A \circ X_n$ is a random variable with values in $\{0,1\}$ for any random variable X_n (with values in $\{0,1\}^n$). $\mu_{A\circ X_n}(1)$ is the probability that an $s \in \{0,1\}^n$ that was chosen according to X_n passes the test A.

The idea is to construct statistical tests A where $s \in \{0,1\}^n$ only passes A if it was chosen randomly, i.e., according to U_n.

4.a. Statistical randomness. The idea is to choose a **statistic**

$$S : \{0,1\}^\bullet \to \mathbb{R}$$

such that the distribution of $S \circ U_n$ converges to a *continuous* probability density f. Recall, a **continuous**[18] real valued random variable $X : \Omega \to \mathbb{R}$ has a **probability density** $f : \mathbb{R} \to \mathbb{R}_{\geq 0}$ satisfying the probability $P_X(I) = P(X \in I) = \int_I f(x)dx$ for any interval $I \subset \mathbb{R}$.

EXAMPLE 4.4. For $\alpha \in (0,1)$ choose an interval $I \subset \mathbb{R}$ (as small as possible) with $\int_I f(x)dx > 1-\alpha$, or, equivalently, $\int_{\mathbb{R}\setminus I} f(x)dx < \alpha$. Define the statistical test $A := A_{S,\alpha}$ **induced by the statistic** S by setting

$$A_{S,\alpha}(s) := \begin{cases} 1 & \text{if} \quad S(s) \in I \\ 0 & \text{if} \quad S(s) \notin I \end{cases} .$$

Then $\mu_{A\circ U_n}(1) > 1-\alpha$ and, equivalently, $\mu_{A\circ U_n}(0) < \alpha$. The real number α is called the **significance level**[19].

Recall that the expected value of the real valued random variable X with density f can be computed as $E(X) := \int_{x\in\mathbb{R}} xf(x)dx$. The **variance**[20] is defined by

$$\mathrm{Var}(X) := E((X - E(X))^2) = E(X^2) - E(X)^2.$$

Begin Lect. 13

REMARK 4.5. Let X, Y be two (finite) random variables and $a, b, c \in \mathbb{R}$.

(1) E is linear, i.e.
$$E(aX + bY) = aE(X) + bE(Y).$$
(2) If X and Y are *independent* then
$$\mathrm{Var}(aX + bY + c) = a^2 \mathrm{Var}(X) + b^2 \mathrm{Var}(Y).$$
(3) If Y is discrete and uniformly distributed and $(Y_1, \ldots, Y_n)$ the n-fold independent repetition of the experiment Y. Then $Z_n := \sum_{i=1}^n \left(\frac{Y_i - E(Y)}{\sqrt{n \mathrm{Var}(Y)}} \right) = \frac{\sum_{i=1}^n Y_i - nE(Y)}{\sqrt{n \mathrm{Var}(Y)}}$ converges to the **standard normal distribution**[21] $N(0,1)$ with expected value 0 and variance 1 (see Appendix 1.b).

PROOF. $E(Z_n) = 0$ and $\mathrm{Var}(Z_n) = \frac{n \mathrm{Var}(Y)}{n \mathrm{Var}(Y)} = 1$. □

EXAMPLE 4.6. We now give two examples of polynomial statistical tests.

(1) **Monobit (or balance) test**: Since $E(U_1) = \frac{1}{2}$ and $\mathrm{Var}(U_1) = \frac{1}{4}$ we define the statistic $S : \{0,1\}^\bullet \to \mathbb{R}$
$$S(s_0 s_1 \ldots s_{n-1}) := \frac{\sum_i s_i - \frac{n}{2}}{\sqrt{\frac{n}{4}}} = \frac{2\sum_i s_i - n}{\sqrt{n}}.$$

[18] German: stetig (hat zwei Bedeutungen!)
[19] German: Signifikanzniveau
[20] German: Varianz
[21] German: Standard-Normalverteilung

according to Remark 4.5.(3). Hence $S \circ U_n$ is an approximation of $N(0,1)$ for n large. For a given significance level $\alpha \in (0,1)$ choose $I = (-x, x)$ with $\operatorname{erfc}(\frac{x}{\sqrt{2}}) < \alpha$, i.e., $x < \sqrt{2}\operatorname{erfc}^{-1}(\alpha)$. Define

$$A : \begin{cases} \{0,1\}^n & \to & \{0,1\} \\ s & \mapsto & \begin{cases} 1 & \text{if } |S(s)| < x \\ 0 & \text{otherwise} \end{cases} \end{cases} = \begin{cases} 1 & \text{if } |\sum_i s_i - \frac{n}{2}| < d \\ 0 & \text{otherwise} \end{cases},$$

where $d := \sqrt{\frac{n}{2}}\operatorname{erfc}^{-1}(\alpha)$. Then $\mu_{A\circ U_n}(1) > 1-\alpha$ and, equivalently, $\mu_{A\circ U_n}(0) < \alpha$. For example, a bit sequence of length $n = 20000$ passes the monobit test with significance level $\alpha = 10^{-6}$ if the number of ones lies in the interval $(\frac{n}{2} - d, \frac{n}{2} + d) \approx (9654, 10346)$.

(2) **Autocorrelation test**: The autocorrelation test on the bit sequence $s = (s_i) \in \mathbb{F}_2^{\mathbb{N}_0}$ with **distance** d is the monobit test on the bit sequence $s' = (s_i + s_{i+d}) \in \mathbb{F}_2^{\mathbb{N}_0}$.

(3) There are much more such tests. See, for example, the so-called **runs test** [**Wik10i**].

REMARK 4.7. LFSRs have good statistical properties; in particular, their output passes all statistical tests in the above example. Indeed, if A is the monobit test then $\mu_{A\circ X_n}(1) \approx 1$ for $X_n = G_n \circ U_{2\ell}$ (cf. Example 4.1).

SKETCH OF PROOF. Each period of a primitive ℓ-bit LFSR (with maximal possible period length $2^\ell - 1$) consists of exactly $2^{\ell-1} - 1$ zeros and $2^{\ell-1}$ ones. □

4.b. Unpredictability. Passing all statistical randomness tests is not enough for a pseudo-random bit generators to be cryptographically secure. It must be "unpredictable" as well.

REMARK 4.8 (Predictability of an LFSR). Let $s = \langle c, t\rangle$ be the output of an ℓ-bit LFSR of which 2ℓ consecutive bits are known, say $s_0, \dots, s_{2\ell-1}$, w.l.o.g.[22]. Hence the ℓ consecutive vectors $t^{(0)}, \dots, t^{(\ell-1)} \in \mathbb{F}_2^{1\times\ell}$ are known. They satisfy the equation

$$\left(\begin{array}{c} t^{(0)} \\ \hline \vdots \\ \hline t^{(\ell-1)} \end{array}\right) \cdot c = \begin{pmatrix} s_\ell \\ \vdots \\ s_{2\ell-1} \end{pmatrix}.$$

This inhomogeneous linear system is solvable by our assumption on s. And there exists a unique solution for c if and only if $t^{(0)}, \dots, t^{(\ell-1)}$ are $\mathbb{F}_2$-linearly independent. In this case the next-bit $s_{2\ell} = (s_\ell \cdots s_{2\ell-1}) \cdot c$ can be predicted. This last condition is satisfied when the LFSR is irreducible and $t^{(0)} \neq 0$.

EXAMPLE 4.9. For $\ell = 4$ and $s = 10101111??\dots$ we solve

$$\begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix} \cdot c = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

[22] German: oBdA

and obtain the unique solution $c = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}$. The complete sequence is 101011110001001|1010 (cf. Example 2.11).

DEFINITION 4.10. Let P be a statistical test. The **next-bit test with predictability** P is the statistical test $A := A_P$ defined by

$$s = s_0 \dots s_n \mapsto A_P(s) = \begin{cases} 0 & \text{if} \quad P(s_0 \dots s_{n-1}) = s_n \\ 1 & \text{if} \quad P(s_0 \dots s_{n-1}) \neq s_n \end{cases}.$$

In words, 0 for correct prediction and 1 for incorrect prediction. Note that if P is polynomial then so is A_P.

Note that $\mu_{A_P \circ U_n}(1) = \mu_{A_P \circ U_n}(0) = \frac{1}{2}$, regardless of P.

EXAMPLE 4.11 (Linear predictability of an LFSR). Define the statistical test P by setting

$$P(s) := (s_\ell \cdots s_{2\ell-1}) \cdot c$$

where $s = s_0, \dots, s_{2\ell-1} \in \{0,1\}^{2\ell}$ and c computed as in Remark 4.8 (a not necessarily unique solution). Remark 4.8 implies[23] that $\mu_{A_P \circ X_{2\ell+1}}(1) = 0$ for $X_{2\ell+1} = G_{2\ell+1} \circ U_{2\ell}$ (cf. Example 4.1). An LFSR is (linearly predictable) and in its original form cryptographically insecure.

5. Cryptographically secure pseudo-random bit generators

Again, let X_n and U_n be random variables with values in $\{0,1\}^n$ where U_n is the uniformly distributed one.

DEFINITION 5.1.

(1) A polynomial *deterministic* algorithm $G : \{0,1\}^\bullet \to \{0,1\}^\bullet$ is called a **pseudo-random bit generator (PRBG)** if there is a function $n : \mathbb{N} \to \mathbb{N}$ with $n(k) > k$ and $G(\{0,1\}^k) \subset \{0,1\}^{n(k)}$ for all $k \in \mathbb{N}$. The function n is called the **stretch function of** G.

(2) A function $f : \mathbb{N} \to \mathbb{R}$ is called **negligible**[24] if for each positive polynomial p there exists a $k_0 \in \mathbb{N}$ such that $|f(k)| < \frac{1}{p(k)}$ for all $k \geq k_0$. The function $f(k) = e^{-k}$ is a prominent example of a negligible function.

(3) We say that G **passes** the statistical test A if

$$k \mapsto \mu_{A \circ G \circ U_k}(1) - \mu_{A \circ U_{n(k)}}(1)$$

is negligible.

[23] We skipped some details here. For example, since P is defined only for $|s|$ even, A_P is a priori only defined for $|s|$ odd. To define it for $|s| = 2r$ we set $A_P(s) := A_P(s_0 \dots s_{2r-2})$. The non-uniqueness of c has to be addressed as well (cf. [**Wik10c**].)

[24] German: vernachlässigbar

(4) G is called **cryptographically secure (CSPRBG)** if G passes *all* polynomial tests.

In words, G is cryptographically secure if an adversary with limited computational resources has no non-negligible advantage in predicting the next bit of the pseudo random sequence.

EXAMPLE 5.2. Let $n : \mathbb{N} \to \mathbb{N}$ be an arbitrary function with $n(k) > k$ for all $k \in \mathbb{N}$. Define G by

$$G(a_0 a_1 \dots a_{k-1}) := \langle a_0 \dots a_{\ell-1}, a_\ell \dots a_{2\ell-1} \rangle_{i=0,\dots,n(k)-1}, \quad \ell := \lfloor \frac{k}{2} \rfloor,$$

the PRBG corresponding to LFSRs. For P as in Example 4.11 we obtain $\mu_{A_P \circ G \circ U_k}(1) - \mu_{A_P \circ U_{n(k)}}(1) \equiv -\frac{1}{2}$, not negligible. Hence, LFSRs are *not* cryptographically secure (even for $n : k \mapsto k+1$).

We state without proof:

THEOREM 5.3 (YAO). *A PRBG is cryptographically secure iff it is* ***unpredictable****, i.e., iff it passes all polynomial next-bit tests.*

Begin Lect. 14

5.a. Empirical security. The problem with the LFSRs is basically their linearity. Here are some attempts to destroy this linearity.

(1) The first idea is to use the complete state vector $t^{(n)} = t^{(n-1)}C$ instead of simply returning its last entry $s_{n+\ell}$. For this use a non-linear "**filter function**" $f : \mathbb{F}_2^{1\times\ell} \to \mathbb{F}_2$, which will be regarded as part of the secret key:

EXAMPLE 5.4 (Knapsack[25] generator). Given a primitive ℓ-bit LFSR (i.e., with period $2^\ell - 1$), fix a natural number $k > \lg \ell$ and choose in some random way non-negative integers $a_0, \dots, a_{\ell-1}$. They build together with the initial vector the secret key. Define the filter function $f(u) :=$ (k-th last bit of $\sum_{u_i=1} a_i$) where $u = (u_0 \dots u_{\ell-1}) \in \mathbb{F}_2^{1\times\ell}$.

(2) The second idea is combine several LFSRs **clocked**[26] in different ways:

EXAMPLE 5.5 (Alternating step generator). Let R be an LFSR generating a sequence $r = (r_n)$ and S and S' two different LFSRs. Use R to reclock S and S' by resetting

$$\begin{aligned} s_{\ell+i} &:= t^{(i)} \cdot c \text{ and } s'_{\ell+i} := s'_{\ell+i-1}, \quad \text{if} \quad r_i = 0, \\ s'_{\ell+i} &:= t'^{(i)} \cdot c' \text{ and } s_{\ell+i} := s_{\ell+i-1}, \quad \text{if} \quad r_i = 1. \end{aligned}$$

Define the resulting sequence to be $(s_i + s'_i)$ (in the notation of Definition 2.1). For a C-implementation see [**Wik10b**].

(3) The third idea is that an LFSR throws away parts of another LFSR:

EXAMPLE 5.6 (Shrinking generator). Two LFSRs are running in parallel and produce the bit sequences s and s'. If $s'_i = 1$ the bit s_i is returned, otherwise it is discarded[27].

[25] German: Rucksack
[26] German: getaktet
[27] German: verworfen

5.b. Provable security. Let $f : \{0,1\}^\bullet \to \{0,1\}^\bullet$ be a polynomial deterministic algorithm. The question of whether there exists a polynomial algorithm that computes preimages[28] of f leads us to the next fundamental definition:

DEFINITION 5.7. Let $f : \{0,1\}^\bullet \to \{0,1\}^\bullet$ be a polynomial deterministic algorithm. For an arbitrary polynomial probabilistic algorithm $A : \{0,1\}^\bullet \to \{0,1\}^\bullet$ define

$$A_f : \{0,1\}^\bullet \to \{0,1\}, \quad x \mapsto \begin{cases} 1 & \text{if } f(A(f(x))) = f(x) \\ 0 & \text{otherwise} \end{cases}.$$

(1) f is called a **one-way function (OWF)**[29] if $k \mapsto \mu_{A_f \circ U_k}(1)$ is negligible for all A.

Let $b : \{0,1\}^\bullet \to \{0,1\}$ be a polynomial deterministic algorithm. For an arbitrary polynomial statistical test (i.e., a polynomial probabilistic algorithm) $A : \{0,1\}^\bullet \to \{0,1\}$ define

$$A_{f,b} : \{0,1\}^\bullet \to \{0,1\}, \quad x \mapsto \begin{cases} 1 & \text{if } A(f(x)) = b(x) \\ 0 & \text{otherwise} \end{cases}.$$

(2) b is called a **hardcore predicate**[30] (or **hardcore bit**, or **hidden bit**) **of** f if $k \mapsto \mu_{A_{f,b} \circ U_k}(1) - \frac{1}{2}$ is negligible for all A.

REMARK 5.8.

(1) If f is injective (in words, does not lose information) and has a hardcore predicate then f is a OWF.
(2) The existence of a hardcore predicate does not imply the injectivity of f. For example, the non-injective function f defined by $f(s_0 s_1 \dots s_n) = s_1 \dots s_n$ has the hardcore predicate $b(s_0 s_1 \dots s_n) = s_0$.

DEFINITION 5.9. A **one-way permutation (OWP)** is a bijective one way function which is length preserving, i.e., $f(\{0,1\}^n) \subset \{0,1\}^n$ and $f : \{0,1\}^n \to \{0,1\}^n$ is a permutation for all $n \in \mathbb{N}_0$.

THEOREM 5.10. *Let f be a OWP with hardcore predicate b and $n : \mathbb{N} \to \mathbb{N}$ an arbitrary function with $n(k) > k$ which is bounded by some polynomial and which is computable by a polynomial run-time algorithm. Then the function $G : \{0,1\}^\bullet \to \{0,1\}^\bullet$ defined by*

$$G(s) := b(s)b(f(s)) \dots b(f^{n(k)-1}(s))$$

is a CSPRBG with stretch function n.

PROOF. Consider

$$G'(s) := b(f^{n(k)-1}(s)) \dots b(f(s))b(s).$$

Assume G' is not cryptographically secure. Then YAO's Theorem would imply the existence of a next-bit test A_P which G' does not pass. But this contradicts b being a hardcore bit

[28] German: Urbilder
[29] German: Einwegfunktion
[30] German: Hardcore-Prädikat, oder Sicherheitsbit

of f. The proof is complete since cryptographic security does not depend on the order of the output. □

LEMMA 5.11. *Let f be a OWP. Then*

$$g : \{0,1\}^{2n} \to \{0,1\}^{2n},\ (x,y) \mapsto (f(x),y)$$

with $|x| = |y|$ defines a OWP with the **GOLDREICH-LEVIN** ***hardcore predicate*** *b given by $b(x,y) := \sum_{i=1}^{n} x_i y_i \in \mathbb{F}_2$.*

PROOF. This is a corollary of the GOLDREICH-LEVIN Theorem, see [**Tre05**]. □

COROLLARY 5.12. *The existence of a CSPRBG is equivalent to the existence of a OWP.*

PROOF. The backward implication follows from Theorem 5.10 and Lemma 5.11. The forward implication is an exercise. □

5.c. A CSPRBG based cryptosystem. We finish this chapter by constructing a cryptosystem based on a (public) CSPRBG G with stretch function n.

EXAMPLE 5.13. Define the *symmetric* cryptosystem $(P, C, K, \mathcal{E}, \mathcal{D})$ with $P = C = \{0,1\}^\bullet$, $K = \{0,1\}^k$ for some **security parameter** $k := \lceil \lg |K| \rceil \in \mathbb{N}$ (e.g., $k = 128$), and $\mathcal{E}$ as follows:
For each $p \in P$ choose randomly a key $e \in K$ and a seed $s \in K$ and compute $G(s) \in \{0,1\}^{n(k)}$. Set

$$c = \mathcal{E}_e(p) := (s+e) \cdot (p + G(s))_{0,\dots,|p|-1},$$

where $+$ is the bitwise addition and $\cdot$ the concatenation of bits. So $|c|$ is slightly bigger than $|p|$. If $|p| > n(k)$ then choose a new random seed. This cryptosystem has at least two advantages:

- After the receiver gets $s + e$ he can compute s and can start to computing $G(s)$.
- The receiver can decrypt c bitwise!

CHAPTER 4

AES and Block Ciphers

1. Block ciphers

DEFINITION 1.1. A **block cipher**[1] is a quadruple $(A, \ell, K, \mathcal{E})$ with finite sets A and K, an $\ell \in \mathbb{N}$, $B := A^\ell$, and $\mathcal{E} : B \times K \to B, (p, e) \mapsto \mathcal{E}_e(p)$, where $\mathcal{E}_e$ is a permutation for all $e \in K$. A is the **alphabet**, K the **key space**, and B the **blocks**.

In Example 1.5 we will see several ways to construct a symmetric cryptosystem out of a block cipher.

EXAMPLE 1.2. Let $A = F = \mathbb{F}_{2^n} = \mathbb{F}_2/\langle f \rangle$ be the finite field with 2^n elements. We list four different actions on $B := F^\ell$.

(1) `SubByte` or `S-box`: The inversion in the field F defines a permutation $^{-1} : a \mapsto a^{-1}$ for $a \in F^*$ and $0^{-1} := 0$. This permutation is *non-linear* but fixes $0, \pm 1$. Choose an $\mathbb{F}_2$-linear invertible map $g : F \to F$ and an element $t \in F$ such that $\sigma : F \to F,\ a \mapsto ga^{-1} + t$ is a *fixed-point-free* permutation (or **derangement**). Extend σ to a permutation $p = (a_1, \ldots, a_\ell) \mapsto (\sigma(a_1), \ldots, \sigma(a_\ell))$ on B.
(2) `ShiftRows`: A permutation $\pi \in S_\ell$ induces a *block permutation* on B defined by $p \mapsto (a_{\pi(1)}, \ldots, a_{\pi(\ell)})$.
(3) `MixColumns`: Choose an element $h \in F[x]$ of degree $m \mid \ell$ and an invertible element c in the residue class ring[2] $R := F[x]/\langle h \rangle$. Then $c \in R^*$ induces a permutation $c : F^m \to F^m, p \mapsto c \cdot p$, where $p = (a_1, \ldots, a_m)$ is identified with the polynomial $a_1 x^{m-1} + \cdots + a_{m-1}x + a_m$. Extend this permutation to a permutation on $B = F^\ell = (F^m)^{\frac{\ell}{m}}$ by $p = (p_1, \ldots, p_{\frac{\ell}{m}}) \mapsto (c \cdot p_1, \ldots, c \cdot p_{\frac{\ell}{m}})$.
(4) `AddRoundKey`: In case $K = B$ then the addition of a key e induces a permutation $p \mapsto p + e$ on $B = F^\ell$.

Note that (1) and (2) commute but (1) and (3) don't.

1.a. AES, the Advanced Encryption Standard.

EXAMPLE 1.3 (AES). The United States Government's **NIST**[3] announced on the 26th of November 2001, after a 5-year standardization process, the symmetric cryptosystem published under the name **Rijndael** as the **Advanced Encryption Standard (AES)**. This block cipher was developed by the two Belgian cryptographers **JOAN DAEMEN** and

[1] German: Blockchiffre
[2] Note that we do not require R to be a field.
[3] National Institute of Standards and Technology

Vincent Rijmen. The AES cryptosystem is now widely used. In the above example choose

- $n = 8$: The 256 elements in the field $F = \mathbb{F}_{2^8} = \mathbb{F}_{256}$ are considered as bytes (=8 bits) and represented by two hexadecimal digits $\texttt{00}, \texttt{01}, \ldots, \texttt{0F}, \texttt{10}, \ldots, \texttt{FF}$. As customary we write `0x` in front of hexadecimal numbers. So `0x63` is the hexadecimal representation of the decimal number 99. Its binary representation is 01100011.
- $\ell := 16$: $B = F^{16} \cong_{\mathbb{F}_2} \mathbb{F}_2^{128}$ which has more elements than atoms in the universe.
- $f := f_{\text{AES}} := x^8+x^4+x^3+x+1$: `0x63` corresponds the field element $\bar{x}^6+\bar{x}^5+\bar{x}+1 \in F$.
- $t :=$ `0x63` $\in F$ corresponding to the vector

$$t := \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} \text{ and choose } g := \begin{pmatrix} 1 & \cdot & \cdot & \cdot & 1 & 1 & 1 & 1 \\ 1 & 1 & \cdot & \cdot & \cdot & 1 & 1 & 1 \\ 1 & 1 & 1 & \cdot & \cdot & \cdot & 1 & 1 \\ 1 & 1 & 1 & 1 & \cdot & \cdot & \cdot & 1 \\ 1 & 1 & 1 & 1 & 1 & \cdot & \cdot & \cdot \\ \cdot & 1 & 1 & 1 & 1 & 1 & \cdot & \cdot \\ \cdot & \cdot & 1 & 1 & 1 & 1 & 1 & \cdot \\ \cdot & \cdot & \cdot & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \in \mathrm{GL}_8(\mathbb{F}_2)$$

For the lookup table of the permutation $F \to F,\ a \mapsto ga^{-1} + t$ see Figure 1.

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
00	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
10	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
20	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
30	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
40	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
50	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
60	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
70	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
80	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
90	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a0	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b0	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c0	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d0	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e0	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f0	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

FIGURE 1. Lookup table for the Rijndael S-box

- π to be the permutation inducing the following row-shifts

$$p := \begin{pmatrix} a_1 & a_5 & a_9 & \boxed{a_{13}} \\ a_2 & a_6 & a_{10} & \boxed{a_{14}} \\ a_3 & a_7 & a_{11} & \boxed{a_{15}} \\ a_4 & a_8 & a_{12} & \boxed{a_{16}} \end{pmatrix} \mapsto \begin{pmatrix} a_1 & a_5 & a_9 & \boxed{a_{13}} \\ a_6 & a_{10} & \boxed{a_{14}} & a_2 \\ a_{11} & \boxed{a_{15}} & a_3 & a_7 \\ \boxed{a_{16}} & a_4 & a_8 & a_{12} \end{pmatrix} \in B = F^{16} \equiv F^{4\times 4}.$$

- $m = 4$, $h := x^4 + 1 = (x+1)^4 \in F[x]$, and $c = \texttt{0x03} \cdot x^3 + x^2 + x + \texttt{0x02} \in R^*$. This corresponds to the matrix[4] multiplication

$$p := \begin{pmatrix} a_1 & a_5 & a_9 & a_{13} \\ a_2 & a_6 & a_{10} & a_{14} \\ a_3 & a_7 & a_{11} & a_{15} \\ a_4 & a_8 & a_{12} & a_{16} \end{pmatrix} \mapsto \underbrace{\begin{pmatrix} \texttt{02} & \texttt{03} & \texttt{01} & \texttt{01} \\ \texttt{01} & \texttt{02} & \texttt{03} & \texttt{01} \\ \texttt{01} & \texttt{01} & \texttt{02} & \texttt{03} \\ \texttt{03} & \texttt{01} & \texttt{01} & \texttt{02} \end{pmatrix}}_{\in F^{4\times 4}} \cdot p \in F^{4\times 4} \equiv F^{16} =: B$$

- $K \in \left\{ \underbrace{B = F^{16}}_{\text{128-bit keys}}, \underbrace{F^{24}}_{\text{192-bit keys}}, \underbrace{F^{32}}_{\text{256-bit keys}} \right\}$.

The ciphering algorithm $\mathcal{E}_e$ is composed out of $N+1$ **rounds**, where $N = 10, 12, 14$ depending on $K = F^{16}, F^{24}, F^{32}$. The Rijndael key schedule $K \to (F^{16})^{N+1}$, $e \mapsto (k_0, \ldots, k_N)$ produces the $N+1$ **round keys**[5] [Wik10h]. For details see [Wik10a] and the homepage of the course [Bar10].

Begin Lect. 15

REMARK 1.4. The iteration of the two steps `ShiftRows` and `MixColumns` produce what SHANNON called the **diffusion**:
Changing a bit in p would change each bit in $c = \mathcal{E}_e(p)$ with probability $\frac{1}{2}$.

1.b. Block cipher modes of operation.

EXAMPLE 1.5. Let $(A, \ell, K, \mathcal{E})$ be a block cipher, and $B := A^\ell$. We now list several ways to construct a symmetric cryptosystem $(P, C, K, \mathcal{E}, \mathcal{D})$ with $P = C = B^\bullet$. So for a (secret) key $e \in K$ define $\mathcal{E}_e(p_1 p_2 \ldots) = c_1 c_2 \ldots$ by setting

(1) $c_i := \mathcal{E}_e(p_i)$ (**ECB mode = electronic codebook mode**).

Assuming that $(A, +)$ is a group and B the ℓ-fold direct product group and that $c_{-1} \in B$ is an arbitrary (public) initial value:

(2) $c_i := \mathcal{E}_e(p_i + c_{i-1})$ for $i \geq 0$ (**CBC mode = cipher-block chaining**).
(3) $c_i := p_i + \mathcal{E}_e(c_{i-1})$ for $i \geq 0$ (**CFB mode = cipher feedback**).
(4) $c_i := p_i + s_i$, $s_i := \mathcal{E}_e(s_{i-1})$ for $i \geq 0$ (**OFB mode = output feedback**).

For diagrams see [Wik10d].

[4] The entries are hexadecimal numbers where we dropped the `0x`-prefix.
[5] German: Rundenschlüssel

EXAMPLE 1.6. Let $(A, \ell, K, \mathcal{E})$ be a block cipher with $A = \mathbb{F}_2^n$ and $B := A^\ell = \mathbb{F}_2^k$, where $k = n\ell$. The encryption algorithm $\mathcal{E}_e : B \to B$ can be used to construct a PRBG $G : \{0,1\}^k \to \{0,1\}^\bullet$ with fixes seed length k according to

$$G(s) = \mathcal{E}_e(s)\mathcal{E}_e^2(s)\dots$$

Compare this with the reversed situation in Example 5.13.

CHAPTER 5

Candidates of One-Way Functions

1. Complexity classes

DEFINITION 1.1. A problem Π lies in the complexity class

- **P** if Π is deterministically solvable in polynomial time.
- **BPP** if Π is probabilistically solvable in polynomial time.
- **BQP** if Π is solvable by a quantum computer in polynomial time.
- **NP** if a solution of Π can deterministically verified in polynomial time.
- **NPC** if any other problem in NP can be reduced to Π in polynomial time.
- **EXP** if Π deterministically solvable in exponential time.

BPP stands for "**bounded-error probabilistic polynomial** runtime" and BQP for "**bounded-error quantum polynomial** runtime".

REMARK 1.2. It is known that

- $\mathrm{P} \subset \mathrm{BPP} \subset \mathrm{NP} \subset \mathrm{EXP}$.
- $\mathrm{P} \subsetneq \mathrm{EXP}$ (e.g., evaluation of a chess move).
- the **traveling salesman problem** lies in NPC.
- the **factorization of natural numbers** (**FACTORING** problem) and the **discrete logarithm problem (DLP)** (see below) lie in $\mathrm{NP} \cap \mathrm{BQP}$.

It is conjectured[1] that

- $\mathrm{P} = \mathrm{BPP}$.
- $\mathrm{BPP} \neq \mathrm{NP}$, in particular $\mathrm{NPC} \cap \mathrm{BPP} = \emptyset$.
- FACTORING and DL do not lie in BPP.
- $\mathrm{BQP} \neq \mathrm{NP}$, in particular $\mathrm{NPC} \cap \mathrm{BQP} = \emptyset$.

It would be optimal for cryptography if there is a $\Pi \in NP \setminus BQP$.

DEFINITION 1.3. Let $G = \langle g \rangle$ be a finite cyclic group. For each $y \in G$, there is exactly one minimal $a \in \mathbb{N}_0$ with $g^a = y$. We call a the **discrete logarithm** of y with **basis** g. Computing $a =$"$\log_g y$" (given a and y) is called the **discrete logarithm problem (DLP)**

REMARK 1.4. In modern cryptography (2010) we make the following two standard assumptions:

(1) **DL assumption**: DLP $\notin$ BPP, i.e., the computation of **discrete logarithm** in the group $C_{p-1} \cong \mathbb{F}_p^*$ is not in BPP.

[1] December 2010

(2) **FACTORING assumption**: The factorization of natural numbers does not lie in BPP.

We can prove the existence of cryptographically useful one-way functions only under such assumptions.

EXAMPLE 1.5. For each prime p choose a fixed primitive element $a \in \mathbb{Z}/p\mathbb{Z}$. Assuming DL, the function

$$f : \{1, \ldots, p-1\} \to \{1, \ldots, p-1\}, x \mapsto a^x \bmod p.$$

is a OWF with hardcore predicate

$$b(x) = \begin{cases} 1 & \text{if } x < \frac{p}{2} \\ 0 & \text{if } x \geq \frac{p}{2} \end{cases} .$$

2. Squaring modulo n

Consider the squaring homomorphism

$$q_n : (\mathbb{Z}/n\mathbb{Z})^* \to (\mathbb{Z}/n\mathbb{Z})^*, x \mapsto x^2.$$

REMARK 2.1. If $n = p$ is a *prime* then

- $\ker(q_n) = \{\pm 1\}$.
- If $p > 2$ then there are exactly $\frac{p-1}{2}$ squares in $(\mathbb{Z}/p\mathbb{Z})^*$.

If $n = pq$, a product of two distinct odd primes p, q. Then

- $\ker(q_n)$ consists of four elements.
- There exists exactly $\frac{\varphi(n)}{4}$ squares in $(\mathbb{Z}/n\mathbb{Z})^*$.

EXAMPLE 2.2. Consider the following values of n:

- $n = 3$:

a	1	2
a^2	1	1

- $n = 5$:

a	1	2	3	4
a^2	1	4	4	1

- $n = 15$:

a	1	2	4	7	8	11	13	14
a^2	1	4	1	4	4	1	4	1

Now we want to study classical methods to identify squares and compute square roots.

2.a. Quadratic residues.

DEFINITION 2.3. For $a \in \mathbb{Z}$ and p prime define the **LEGENDRE symbol**

$$\left(\frac{a}{p}\right) := \begin{cases} 1 & \text{if} \quad a \equiv b^2 \bmod p, \\ 0 & \text{if} \quad p \mid a, \\ -1 & \text{otherwise} \end{cases} .$$

THEOREM 2.4 (EULER). *Let $p > 2$ be an odd prime. Then $\left(\frac{a}{p}\right) \equiv a^{\frac{p-1}{2}} \bmod p$*

PROOF. The case $p \mid a$ is clear. So assume $p \nmid a$. The group $\mathbb{F}_p^* = (\mathbb{Z}/p\mathbb{Z})^*$ is cyclic of order $p-1$ so $a^{p-1} \equiv 1 \bmod p$. Hence $a^{\frac{p-1}{2}}$ is a root of $x^2 - 1 \in \mathbb{F}_p[x]$ and the group homomorphism

$$h : (\mathbb{Z}/p\mathbb{Z})^* \to \{\pm 1\} \leq (\mathbb{Z}/p\mathbb{Z})^*, \ a \mapsto a^{\frac{p-1}{2}}$$

is surjective. The kernel of h thus consists of $\frac{p-1}{2}$ elements and contains $((\mathbb{Z}/p\mathbb{Z})^*)^2$, so it coincides with $((\mathbb{Z}/p\mathbb{Z})^*)^2$. □

EULER's theorem can be used to simplify the computation of the LEGENDRE symbol. For example

COROLLARY 2.5.

$$\left(\frac{-1}{p}\right) = (-1)^{\frac{p-1}{2}} = \begin{cases} 1 & \textit{if} \quad p \equiv 1 \bmod 4 \\ -1 & \textit{if} \quad p \equiv 3 \bmod 4 \end{cases}$$

EXERCISE 2.6. The map

$$\left(\frac{\cdot}{p}\right) : (\mathbb{Z}/p\mathbb{Z})^* \to \{\pm 1\}$$

is a group homomorphism.

DEFINITION 2.7. The elements in the kernel of $\left(\frac{\cdot}{p}\right)$ are called **quadratic residues** modulo p.

DEFINITION 2.8 (JACOBI symbol). Let $n = p_1^{a_1} \cdots p_r^{a_r} > 1$ be the decomposition of the natural number n as powers of distinct primes. For $a \in \mathbb{Z}$ set

$$\left(\frac{a}{n}\right) := \left(\frac{a}{p_1}\right)^{a_1} \cdots \left(\frac{a}{p_r}\right)^{a_r} \in \{-1, 0, 1\},$$

and for $n = 1$ set $\left(\frac{a}{1}\right) := 1$.

The JACOBI symbol can be computed without knowing an explicit factorization of n (cf. EZT Vorlesung).

COROLLARY 2.9. $\left(\frac{a}{n}\right) = -1$ *implies that a is not a square modulo n.*

DEFINITION 2.10. If $\left(\frac{a}{n}\right) = 1$ and a is not a square modulo n, then we call a a **pseudo-square modulo** n.

EXAMPLE 2.11. Consider the following values of n:

- $n = 3$:

a	1	2
a^2	1	1
$\left(\frac{a}{n}\right)$	1	-1

- $n = 5$:

a	1	2	3	4
a^2	1	4	4	1
$\left(\frac{a}{n}\right)$	1	-1	-1	1

- $n = 15$:

a	1	2	4	7	8	11	13	14
a^2	1	4	1	4	4	1	4	1
$\left(\frac{a}{n}\right)$	1	1	1	-1	1	-1	-1	-1

So 2 and 8 are pseudo-squares modulo 15.

DEFINITION 2.12. Define the set of

- squares modulo n:
$$Q_n := \{a \in (\mathbb{Z}/n\mathbb{Z})^* \mid \exists b \in \mathbb{Z}/n\mathbb{Z} : a = b^2\} = ((\mathbb{Z}/n\mathbb{Z})^*)^2.$$
- non-squares modulo n:
$$\overline{Q}_n := \{a \in (\mathbb{Z}/n\mathbb{Z})^* \mid \nexists b \in \mathbb{Z}/n\mathbb{Z} : a = b^2\} = (\mathbb{Z}/n\mathbb{Z})^* \setminus ((\mathbb{Z}/n\mathbb{Z})^*)^2.$$
- pseudo-squares modulo n:
$$\widetilde{Q}_n := \left\{a \in (\mathbb{Z}/n\mathbb{Z})^* \mid \left(\frac{a}{n}\right) = 1\right\} \setminus Q_n \subset \overline{Q}_n.$$

Begin Lect. 16

PROPOSITION 2.13. *Let p be a prime with $p \equiv 3 \bmod 4$. Then $a \in Q_p$ has exactly one square root in Q_p. We call it the* ***principal root of*** *a.*

PROOF. $\mathbb{Z}/p\mathbb{Z} \cong \mathbb{F}_p$ is a field, hence there are exactly two roots $\pm b$. By Corollary 2.5 and Exercise 2.6 we compute $\left(\frac{-b}{p}\right) = \left(\frac{-1}{p}\right)\left(\frac{b}{p}\right) = -\left(\frac{b}{p}\right)$. W.l.o.g. $\left(\frac{b}{p}\right) = 1$ and $\left(\frac{-b}{p}\right) = -1$, so $b \in Q_p$ and $-b \in \overline{Q}_p$. □

EXAMPLE 2.14.

- $Q_5 = \{1, 4\}$. The square roots of 4 are $2, 3 \in \overline{Q}_5$. The square roots of 1 are $1, 4 \in Q_5$.
- $Q_7 = \{1, 2, 4\}$. The square roots of 2 are $3 \in \overline{Q}_7$ and $4 \in Q_7$. The square roots of 4 are $2 \in Q_7$ and $5 \in \overline{Q}_7$.

DEFINITION 2.15. An $n \in \mathbb{N}$ is called a **BLUM number** if $n = p_1 p_2$ with p_i distinct primes and $p_i \equiv 3 \bmod 4$.

REMARK 2.16. The following holds for a BLUM number n:

(1) Each $a \in Q_n$ has exactly one square root in Q_n (again called the **principal root of** a), one in $\widetilde{Q}_n$, and two in $\overline{Q}_n \setminus \widetilde{Q}_n$.
(2) $-1 \in \widetilde{Q}_n$.

PROOF. (1) follows from Proposition 2.13 and the chinese remainder theorem. Details are left as an exercise. (2) follows from Corollary 2.5. □

2.b. Square roots.

DEFINITION 2.17. We list the following three fundamental problems:

FACTORING	Given an $n \in \mathbb{N}$ compute a prime factor.
SQROOT	Given an $n \in \mathbb{N}$ and a square $a \in Q_n$ compute a square root of a modulo n.
QRP	Given an $n \in \mathbb{N}$ and $a \in \mathbb{Z}$ with $\left(\frac{a}{n}\right) = 1$ decide whether a is a square or a pseudo-square.

THEOREM 2.18. *SQROOT for $n = p$ prime lies in BPP.*

PROOF. Let $n = p > 2$ a prime number and $a \in (\mathbb{Z}/p\mathbb{Z})^*$. The idea is to exploit that

$$a^m = 1 \text{ for } m \text{ odd implies that } \left(a^{\frac{m+1}{2}}\right)^2 = a^{m+1} = a. \tag{3}$$

Recall that

$$a \in Q_p \iff a^{\frac{p-1}{2}} \equiv 1 \bmod p,$$

by EULER's Theorem 2.4. So let a be a square.
Case 1: $\frac{p-1}{2}$ is odd, i.e., $p \equiv 3 \bmod 4$:
Using (3) with $m = \frac{p-1}{2}$ yields the square root $a^{\frac{m+1}{2}} = a^{\frac{p+1}{4}}$ of a.
Case 2: $\frac{p-1}{2}$ is even, i.e., $p \equiv 1 \bmod 4$:

$$a \in Q_p \iff a^{\frac{p-1}{2}} = 1 \iff a^{\frac{p-1}{4}} = \pm 1.$$

We now prove by induction that we can use the equation $a^m = \pm 1$ for $m \mid \frac{p-1}{4}$ to compute a square root of a. We start with $m = \frac{p-1}{4}$.
Case a): $a^m = 1$ and m even: Proceed with the equation $a^{\frac{m}{2}} = \pm 1$.
Case b): $a^m = 1$ and m odd: (3) yields a square root $a^{\frac{m+1}{2}}$ of a.
Case c): $a^m = -1$. *Choose* an arbitrary $b \in \overline{Q}_p$ and set $b' := b^{\frac{p-1}{4m}}$. Proceed with the equation

$$(ab'^2)^m = a^m b^{\frac{p-1}{2}} = (-1)^2 = 1.$$

Finally note that if c is a square root of ab'^2 then $a = (cb'^{-1})^2$.
This describes the probabilistic polynomial algorithm of **TONELLI–SHANKS** [Wik11f]. We omit the details. □

EXAMPLE 2.19. Let $p = 41$. We know that $a = -2 \in Q_{41}$ since $(-2)^{21} = -(2^7)^3 = -5^3 = -2$ so $(-2)^{20} = 1$. Now we want to use the above algorithm to compute a square root of a. Note that $\frac{p-1}{2} = 20$ is even and $\frac{p-1}{4} = 10$. Find an element $b \in \overline{Q}_{41}$ by randomly checking (probability of failure is $\frac{1}{2}$):

- $2^{20} = (-1)^{20} \cdot (-2)^{20} = 1$ ($\times$).
- $3^{20} = (3^4)^5 = (81)^5 = (-1)^5 = -1$ ($\surd$).

So choose $b = 3$:

	a	m	a^m	$\frac{p-1}{4m}$	$b' = b^{\frac{p-1}{4m}}$	$\frac{m+1}{2}$	b'^{-1}		$\sqrt{a}$
	-2	10	-1	1	$3^1 = 3$		14	$33 \cdot 14 =$	11
$-2 \cdot 3^2 =$	23	10	1						
	23	5	-1	2	$3^2 = 9$		32	$10 \cdot 32 =$	33
$23 \cdot 9^2 = -23 =$	18	5	1			3		$18^3 =$	10

LEMMA 2.20. *Let $n = p_1 p_2$ be a* BLUM *number ($p_i \equiv 3 \bmod 4$ will not be relevant). Any $x \in \ker q_n$ with $x \neq \pm 1$ yields a factorization of n.*

PROOF. Let $x \in \{m \in \mathbb{N} \mid 1 < m < n-1 \text{ and } m^2 \equiv 1 \bmod n\}$. Then $p_1 p_2 = n \mid x^2 - 1 = (x-1)(x+1)$. Since $n \nmid x \pm 1$ we conclude w.l.o.g. that $p_1 \mid x - 1$ and $p_2 \mid x + 1$. Now p_1 can be effectively computed as $p_1 = \gcd(x-1, n)$. □

THEOREM 2.21. *If SQROOT for* BLUM *numbers lies in BPP then FACTORING for* BLUM *numbers lies in BPP.*

PROOF. From a probabilistic polynomial algorithm A that solves SQROOT for BLUM numbers we construct the following probabilistic polynomial algorithm that solves FACTORING:
Choose an arbitrarily element $c \in (\mathbb{Z}/n\mathbb{Z})^*$ and compute $a := A(c^2)$. So $\frac{c}{a}$ is an element in $\ker q_n$ which is with probability $\frac{1}{2}$ different from ± 1. The rest is Lemma 2.20 □

2.c. One-way functions. We sharpen our
FACTORING assumption: FACTORING of BLUM numbers does not lie in BPP.
We also need the following
QR assumption: QRP for BLUM numbers does not lie in BPP.

THEOREM 2.22. *Let n be a* BLUM *number. Then $f := q_{n|Q_n} : Q_n \to Q_n$ is a permutation.*

(1) *f is a one-way permutation under the FACTORING assumption (with security parameter: $k := \lceil \lg |Q_n| \rceil = \lceil \lg \frac{\varphi(n)}{4} \rceil = \lceil \lg \varphi(n) \rceil - 2$).*
(2) *The so-called* ***parity (bit)***

$$\text{par} : (\mathbb{Z}/n\mathbb{Z})^* \mapsto \{0,1\},\ a \mapsto (\textit{smallest nonnegative representative of } a) \bmod 2$$

defines under the QR assumption a hardcore bit of f.

PROOF. f is a permutation by Remark 2.16.(1). (1) follows from Theorem 2.21. To prove (2) let $\left(\frac{a}{n}\right) = 1$, i.e., $a \in Q_n \dot\cup \widetilde{Q}_n$. For the principal root $w \in Q_n$ of a^2 we claim:

$$w = a \iff \operatorname{par}(w) = \operatorname{par}(a). \tag{par}$$

The forward implication of the claim is trivial. We now prove the backward implication: Since $-1 \in \widetilde{Q}_n$ by Remark 2.16.(2) and $w \in Q_n$ we deduce that $-w \in \widetilde{Q}_n$ (i.e., that $\widetilde{Q}_n = -Q_n$). So $a = w$ or $a = -w$. In other words: $a \neq w \implies a = -w \implies \operatorname{par}(w) \neq \operatorname{par}(a)$ (remember, n is odd). From an algorithm B for which $B(x)$ with $x = f(a) = a^2$ predicts $\operatorname{par}(a)$ we obtain an algorithm for QRP by returning:

$$\begin{cases} a \text{ is a square} & \text{if } \ B(a^2) = \operatorname{par}(a), \\ a \text{ is a pseudo-square} & \text{if } \ B(a^2) \neq \operatorname{par}(a). \end{cases}$$

□

DEFINITION 2.23. The function f is called the **RABIN function**. The PRBG G constructed according to Theorem 5.10 is called the **BLUM-BLUM-SHUB generator** (see [**Wik10e**]): For a BLUM number n and a **seed** $s \in (\mathbb{Z}/n\mathbb{Z})^*$ define $G(s) = x_0x_1x_2\ldots$ with $x_i = \operatorname{par}(s^{2^i})$. G is then a CSPRBG under the QR assumption for BLUM numbers.

Begin Lect. 17

2.d. Trapdoors. A OWP $f : \{0,1\}^\bullet \to \{0,1\}^\bullet$ can be viewed as a family of permutations $f_k : \{0,1\}^k \to \{0,1\}^k$.

To define a OWP with a **trapdoor**[2] we need the following ingredients:

- I, an infinite **index set**,
- $|\cdot| : I \to \mathbb{N}$, a **length function**.
- $I_k := \{i \in I : |i| \leq k\}$ (we call k the **security parameter**).
- X_i for all $i \in I$, a family of finite sets.
- $f = (f_i)_{i\in I} : \bigcup_{i\in I} X_i \to \bigcup_{i\in I} X_i$, a family of permutations $f_i : X_i \to X_i$.
- t_i for all $i \in I$, **trapdoor information** (see the examples below).
- $\mathcal{E}$, a polynomial algorithm with $\mathcal{E}(i,x) = \mathcal{E}_i(x) = f_i(x)$ for all $i \in I$ and $x \in X_i$.
- $\mathcal{D}$, a polynomial algorithm with $\mathcal{D}(i,t_i,f_i(x)) = \mathcal{D}_{(i,t_i)}(f_i(x)) = x$ for all $i \in I$ and $x \in X_i$.
- $S_k := \{(i,x) \mid i \in I_k, x \in X_i\}$, the possible inputs of $\mathcal{E}$ with security parameter k.

For a probabilistic algorithm $A : \bigcup_{i\in I} X_i \to \bigcup_{i\in I} X_i$ with output $A(i,y) \in X_i$ for all $i \in I$ and $y \in X_i$ define the probabilistic algorithm A_f by setting

$$A_f(i,x) = \begin{cases} 1 & \text{if } A(i,f_i(x)) = x, \\ 0 & \text{otherwise,} \end{cases}$$

for all $i \in I$ and $x \in X_i$.

As usual, let U_{S_k} denote the uniformly distributed random variable on S_k (i.e., first choose a random $i \in I$ and then a random $x \in X_i$). Then $\mu_{A_f \circ U_{S_k}}(1)$ is the probability of A correctly computing the preimage x of $y = f_i(x)$.

[2]German: wörtlich Falltür, man sagt aber im Deutschen Hintertür

DEFINITION 2.24. The permutation $f = (f_i)_{i\in I} : \bigcup_{i\in I} X_i \to \bigcup_{i\in I} X_i$ is called a **one-way permutation with trapdoor** $t = (t_i)_{i\in I}$ if $k \mapsto \mu_{A_f \circ U_{S_k}}(1)$ is negligible for all polynomial algorithms A as above (cf. Definition 5.7.(1)).

EXAMPLE 2.25 (RABIN function). Set $I := \{\text{BLUM numbers}\}$, $|n| = k = \lfloor \lg n \rfloor$, $X_n = Q_n$, $f_n = q_{n|Q_n} : x \mapsto x^2 \bmod n$, t_n the factorization of n, and $\mathcal{D}$ is the combination of the algorithm in the proof of Theorem 2.18 with the chinese remainder theorem. We obtain a one way permutation with trapdoor under the FACTORING assumption, which is equivalent to the "SQROOT $\notin$ BPP" assumption for BLUM numbers[3]. The parity bit $b :=$ par is a hardcore bit under the QR assumption (by Theorem 2.22.(2)).

EXERCISE 2.26. The knowledge of sufficiently many preimages of f_n leads to a polynomial runtime algorithm to determine t_n, i.e., to factor n.

2.e. The BLUM-GOLDWASSER construction. Given a OWP with trapdoor and hardcore bit b BLUM and GOLDWASSER constructed the following asymmetric *probabilistic*[4] cryptosystem $(P, C, \kappa, \mathcal{E}, \mathcal{D})$ with

- $P = C = \{0,1\}^\bullet$,
- $K = I$, $K' = \{(i, t_i) \mid i \in I\}$, $\kappa : K' \to K, (i, t_i) \mapsto i$,
- $\mathcal{E}_e : P \rightsquigarrow C$, $\mathcal{D}_d : C \to P$ as follows (compare with Example 5.13):
  Let $e \in K$ and $p \in \{0,1\}^\ell$. *Choose* an arbitrary **seed** $s \in X_e$ and compute the sequence
  $$r = b(s)b(f_e(s)) \dots b(f_e^{\ell-1}(s))$$
  together with $f_e^\ell(s) \in X_e$. Define
  $$\mathcal{E}_e(p) = f_e^\ell(s) \cdot (p + r),$$
  where, as customary, $+$ is the bitwise addition and $\cdot$ the concatenation[5] of bits.
  Let now $d = (e, t_e) \in K'$ and $c = s' \cdot c'$ with $c' \in \{0,1\}^\ell$. Use $s' = f_e^\ell(s)$ and the trapdoor information t_e to recursively compute $f_e^{\ell-1}(s), \dots, f_e(s), s$. Now compute $r = b(s)b(f_e(s)) \dots b(f_e^{\ell-1}(s))$ and return $\mathcal{D}_d(c) = c' + r$.

DEFINITION 2.27. The BLUM-GOLDWASSER construction applied to the RABIN function is called the **BLUM-GOLDWASSER cryptosystem**.

THEOREM 2.28. *The* BLUM-GOLDWASSER *cryptosystem is an asymmetric probabilistic cryptosystem where*

(1) *the FACTORING assumption implies* ***ASYMMETRY***[6] *(i.e., the secret key cannot be computed in polynomial time using the public key), and*

[3] The equivalence is Theorem 2.18 combined with the chinese remainder theorem and Theorem 2.21.

[4] Recall that for an asymmetric cryptosystem to satisfy IND it must be probabilistic with $\mathcal{E}$ multi-valued.

[5] $f^\ell(s)$ stands for its bit-coding.

[6] The negation of ASYMMETRY is called "total break". This property only makes sense for public key cryptosystems.

(2) *the QR assumption implies CPA-IND*[7].

PROOF. (1) By definition, computing $d = (n, t_n)$ means factoring the BLUM number n.

(2) Let $p_1, p_2 \in \{0,1\}^\ell$ and $c_i = \mathcal{E}_e(p_i)$. The QR assumption, Theorem 2.22.(2), and Theorem 5.10 imply that the construction of r defines a CSPRBG. Hence an attacker cannot distinguish between $p_1 + r_1$ and $p_2 + r_2$ (even if $f_e^\ell(s)$ is known, exercise).

□

[7]Cf. Definition 4.9 and Remarks 4.12 and 4.14

CHAPTER 6

Public Cryptosystems

1. RSA

REMARK 1.1. Let $n, e \in N$ with $\gcd(e, \varphi(n)) = 1$. Then

$$f_e : (\mathbb{Z}/n\mathbb{Z})^* \to (\mathbb{Z}/n\mathbb{Z})^*, a \mapsto a^e$$

is a permutation with inverse f_d, where $de \equiv 1 \bmod \varphi(n)$.

PROOF. By definition $\varphi(n) = |(\mathbb{Z}/n\mathbb{Z})^*|$. The extended EUCLIDian division algorithm yields the BÉZOUT identity $de + \lambda\varphi(n) = 1$. Since $a^{\varphi(n)} = 1$ by LAGRANGE's theorem we conclude that

$$a = a^1 = a^{de+\lambda\varphi(n)} = (a^e)^d.$$

□

For an n of the form $n = pq$ with p and q distinct primes it follows that $\varphi(n) = (p-1)(q-1)$.

EXAMPLE 1.2 (RSA function). Define

- $I := \{(n = pq, e) \mid p, q \text{ are distinct primes and } \gcd(e, (p-1)(q-1)) = 1\}$.

For $i = (n = pq, e) \in I$ define

- $|i| = k := \lfloor \lg n \rfloor$.
- $X_i = (\mathbb{Z}/n\mathbb{Z})^*$.
- $f_i : a \mapsto a^e \bmod n$.
- $t_i = d$ with $de \equiv 1 \bmod (p-1)(q-1)$.
- $\mathcal{D}(i, t_i, y) = \mathcal{D}_{(i,t_i)}(y) = y^d \bmod n$.

DEFINITION 1.3. The **RSA problem (RSAP)** is the problem of inverting the RSA function. The **RSA assumption** is that RSAP $\notin$ BPP.

REMARK 1.4.

- The RSAP reduces to FACTORING, i.e., the RSA assumption is stronger than the FACTORING assumption.
- Under the RSA assumption: The RSA function is a OWP with trapdoor and hard-core bit $b = \text{par}$ (without proof). The BLUM-GOLDWASSER construction yields, as for the RABIN function, a probabilistic asymmetric cryptosystem satisfying IND-CPA.

DEFINITION 1.5. The **RSA cryptosystem** is defined as follows:

- $P = \{0,1\}^k$, $C = \{0,1\}^{k+1}$ (e.g., $k = 1024$).
- $K = I$ as above.
- $K' =$
  $\{(n = pq, d, e) \mid p, q \text{ distinct primes}, \ |p|, |q| \approx \frac{k}{2}, \ |pq| \geq k, \ de \equiv 1 \bmod \varphi(n)\}$.
- $\kappa : (n, d, e) \mapsto (n, e) \in K$.
- $\mathcal{E}_{(n,e)}(x) = x^e \bmod n$.
- $\mathcal{D}_{(n,d,e)}(y) = y^d \bmod n$.

REMARK 1.6. We now list the security properties of the RSA cryptosystem (assuming a CPA attack, which is natural for public cryptosystems):

- p, q and $\varphi(n)$ must remain secret.
- RSA assumption $\implies$ OW $\implies$ ASYMMETRY.
- IND is not satisfied since the cryptosystem is deterministic.
- NM is not satisfied since the cryptosystem is *multiplicative*: $(ab)^e = a^e b^e$ (see below).

EXAMPLE 1.7. Let $p = 11$, $q = 23$, and $e = 3$. Then $n = pq = 253$, $k = \lfloor \lg n \rfloor = 7$, $\varphi(n) = (p-1)(q-1) = 10 \cdot 23 = 220$, and $d = 147$ with $ed = 441 \equiv 1 \bmod 220$. For $p = 0110100 = (52)_{10}$ we compute $c = \mathcal{E}_e(p) = 52^3 = 193 \bmod 253 = (1100001)_2$. $\mathcal{D}_{(253,147,3)}(c) = 193^{147} \equiv 52 \bmod 253 = p$.
Violating the NM: To shift p one position to the left we manipulate c to $c' = \mathcal{E}_e(2) \cdot c = 2^3 \cdot 193 = 26 \bmod 253$. Then $\mathcal{D}_{(253,147,3)}(c) = 26^{147} \equiv 104 \bmod 253 = (1101000)_2$.

Begin Lect. 18

In analogy with the trivial statement of Theorem 2.28.(1) for the BLUM-GOLDWASSER cryptosystem we prove:

THEOREM 1.8. *The FACTORING assumption implies the ASYMMETRY of the RSA cryptosystem, i.e., the secret key d cannot be computed in polynomial time using the public key (n, e).*

PROOF. Assume, a CPA adversary[1] can compute the secret key d. We need to show that he can then factor n using the knowledge of $(n, d, e) \in K'$:
The chinese remainder theorem provides an *isomorphism*

$$(\mathbb{Z}/n\mathbb{Z})^* \to (\mathbb{Z}/p\mathbb{Z})^* \times (\mathbb{Z}/q\mathbb{Z})^*, \ a \bmod n \mapsto (a \bmod p, a \bmod q).$$

In particular $\operatorname{ord}_{(\mathbb{Z}/n\mathbb{Z})^*}(a) = \operatorname{lcm}(\operatorname{ord}_{(\mathbb{Z}/p\mathbb{Z})^*}(a), \operatorname{ord}_{(\mathbb{Z}/q\mathbb{Z})^*}(a))$. The idea is to use the following trivial equivalence

$$c \equiv 1 \bmod p, \ c \not\equiv 1 \bmod q \iff p \mid c-1, \ q \nmid c-1 \iff (c-1, n) = p$$

to factor n. So our goal is to construct such an element c.
For any a we have that $\operatorname{ord}_{(\mathbb{Z}/p\mathbb{Z})^*}(a) \mid p-1$, $\operatorname{ord}_{(\mathbb{Z}/q\mathbb{Z})^*}(a) \mid q-1$, and $\operatorname{ord}_{(\mathbb{Z}/n\mathbb{Z})^*}(a) \mid (p-1)(q-1) = \varphi(n) \mid ed-1$. Write $ed - 1 = 2^s t$ with t odd. Then $(a^t)^{2^s} = 1$, hence $\operatorname{ord}_{(\mathbb{Z}/n\mathbb{Z})^*}(a^t) \mid 2^s$. Choose randomly an element $b \in ((\mathbb{Z}/n\mathbb{Z})^*)^t$, for instance $b = a^t$ with $a \in (\mathbb{Z}/n\mathbb{Z})^*$ randomly chosen. Then $\operatorname{ord}_{(\mathbb{Z}/p\mathbb{Z})^*}(b) = 2^i$ and $\operatorname{ord}_{(\mathbb{Z}/q\mathbb{Z})^*}(b) = 2^j$ with $i, j \leq s$.

[1] German: Gegner, Kontrahent

If $i \neq j$, or w.l.o.g. $i < j$, then $c := b^{2^i} \equiv 1 \bmod p$ and $c \not\equiv 1 \bmod q$ and we get the factorization $p = (c-1, n)$.
We now prove that $i \neq j$ for (at least) half of all $a \in (\mathbb{Z}/n\mathbb{Z})^*$ (recall, $b := a^t$). Recall that the choice of a primitive element $g \in (\mathbb{Z}/p\mathbb{Z})^*$ yields an isomorphism $(\mathbb{Z}/(p-1)\mathbb{Z}, +) \to (\mathbb{Z}/p\mathbb{Z})^*$, $x \mapsto g^x$. As above, $\operatorname{ord}_{(\mathbb{Z}/(p-1)\mathbb{Z},+)}(1) = p-1 \mid 2^s t$ and $\operatorname{ord}_{(\mathbb{Z}/(p-1)\mathbb{Z},+)}(t) = 2^s$. Using the identification $(\mathbb{Z}/n\mathbb{Z})^* \cong (\mathbb{Z}/p\mathbb{Z})^* \times (\mathbb{Z}/q\mathbb{Z})^* \cong (\mathbb{Z}/(p-1)\mathbb{Z}, +) \times (\mathbb{Z}/(q-1)\mathbb{Z}, +)$ it is equivalent to show that the inequality $\operatorname{ord}_{(\mathbb{Z}/(p-1)\mathbb{Z},+)}(xt) \neq \operatorname{ord}_{(\mathbb{Z}/(q-1)\mathbb{Z},+)}(yt)$ holds for (at least) half of all pairs $(x, y) \in (\mathbb{Z}/(p-1)\mathbb{Z}, +) \times (\mathbb{Z}/(q-1)\mathbb{Z}, +)$.
Proof of the inequality: Let $\operatorname{ord}_{(\mathbb{Z}/(p-1)\mathbb{Z},+)}(t) = 2^k$ and $\operatorname{ord}_{(\mathbb{Z}/(q-1)\mathbb{Z},+)}(t) = 2^\ell$. Note that $\operatorname{ord}_{(\mathbb{Z}/(p-1)\mathbb{Z},+)}(t) = \operatorname{ord}_{(\mathbb{Z}/(p-1)\mathbb{Z},+)}(xt)$ for all x odd (trivial). Now we distinguish to cases:

$k \neq \ell$: Again, w.l.o.g. let $\ell < k$. Then for all (x, y) with x odd we obtain:

$$\operatorname{ord}_{(\mathbb{Z}/(q-1)\mathbb{Z},+)}(yt) \leq \operatorname{ord}_{(\mathbb{Z}/(q-1)\mathbb{Z},+)}(t) = 2^\ell < 2^k = \operatorname{ord}_{(\mathbb{Z}/(p-1)\mathbb{Z},+)}(t) \operatorname{ord}_{(\mathbb{Z}/(p-1)\mathbb{Z},+)}(xt).$$

$k = \ell$: This case is left as an exercise.

□

EXAMPLE 1.9 (Example 1.7 continued). As above let $n = 253 = 11 \cdot 23$, $e = 3$, $d = 147$, $ed - 1 = 220 = 2^2 \cdot 55$. So $s = 2$ and $t = 55$. Try $a = 2$: $b = a^t = 2^{55} \equiv 208 \bmod 253$. Compute $(b^{2^i} - 1, n)$ for $i = 0, 1 < s = 2$: $(208 - 1, 253) = 23$.

Summing up, we get the following implications of security *assumptions*[2] for the RSA cryptosystem (under a CPA attack):

$$\begin{array}{ccc} \text{RSA} & \Longrightarrow & \text{FACTORING} \\ \Downarrow & & \Downarrow \\ \text{OW} & \Longrightarrow & \text{ASYMMETRY} \end{array}$$

2. ELGAMAL

Recall: Let p be a prime and g be a generator of $(\mathbb{Z}/p\mathbb{Z})^*$. The problem of inverting $\exp_g : (\mathbb{Z}/p\mathbb{Z})^* \to (\mathbb{Z}/p\mathbb{Z})^*$, $x \mapsto g^x$ is the discrete logarithm problem (DLP). $\exp_g$ is a OWP[3] under the DL assumption. We don't have candidates for a trapdoor.

DEFINITION 2.1. Let p be a prime number and $(\mathbb{Z}/p\mathbb{Z})^* = \langle g \rangle$.

- The problem of computing $g^{\alpha\beta}$ given g^α and g^β is called **DIFFIE-HELLMAN problem (DHP)**
- The **DIFFIE-HELLMAN or DH assumption** is that DHP $\notin$ BPP.

REMARK 2.2. The DHP reduces to the DLP, i.e., the DH *assumption* is stronger than the DL assumption. The equivalence is unknown.

DEFINITION 2.3. The **ELGAMAL cryptosystem** is defined by

- $P = \{0,1\}^k$ and $C = \{0,1\}^{2(k+1)}$.

[2] ... not the problems. For the "hardness of the problems" you have to invert the arrows.
[3] $g^0 = g^{p-1} = 1$.

- $K' = \{(p, g, a) \mid p \text{ prime}, \ \langle g \rangle = (\mathbb{Z}/p\mathbb{Z})^*, \ 2^k < p < 2^{k+1}, \ a \in \{0, \dots, p-2\}\}$.
- $\kappa : (p, g, a) \mapsto (p, g, g^a) \in K$.

We encode $\{0,1\}^k \subset (\mathbb{Z}/p\mathbb{Z})^* \subset \{0,1\}^{k+1}$. So we "replace" P by $(\mathbb{Z}/p\mathbb{Z})^*$ and C by $(\mathbb{Z}/p\mathbb{Z})^* \times (\mathbb{Z}/p\mathbb{Z})^*$. For $e = (p, g, A = g^a)$ and $d = (p, g, a)$ we define

- $\mathcal{E}_e(x) := (g^b, A^b x)$, where $b \in \{0, \dots, p-2\}$ is chosen randomly.
- $\mathcal{D}_d(y, z) := y^{-a} z$.

Of course, a has to be kept secret.

PROOF OF CORRECTNESS. $(g^b)^{-a} A^b x = g^{-ba+ab} x = x$. □

EXAMPLE 2.4. Take $p = 23$ and $g = 7$. For $a = 6$, i.e., $d = (23, 7, 6)$ compute $A = 7^6 \equiv 4 \bmod 23$. $e = \kappa(d) = (23, 7, 4)$. For $x = 7 \in (\mathbb{Z}/23\mathbb{Z})^* = P$ compute $\mathcal{E}_e(x)$ for different b's:

$b = 3$: $\mathcal{E}_e(x) = (7^3, 4^3 \cdot 7) = (21, 11) \in (\mathbb{Z}/23\mathbb{Z})^* \times (\mathbb{Z}/23\mathbb{Z})^* = C$.
$b = 2$: $\mathcal{E}_e(x) = (7^2, 4^2 \cdot 7) = (3, 20) \in (\mathbb{Z}/23\mathbb{Z})^* \times (\mathbb{Z}/23\mathbb{Z})^* = C$.

Now verify $\mathcal{D}_d(21, 11) = 21^{-6} \cdot 11 \equiv \boxed{7} \bmod 23 \equiv 3^{-6} \cdot 20 = \mathcal{D}_d(3, 20)$.

REMARK 2.5. The ELGAMAL cryptosystem is a probabilistic public key cryptosystem with multi-valued $\mathcal{E}$. It

- satisfies the IND-CPA security model under the DL assumption (without proof).
- does not satisfy NM because of its multiplicativity (like RSA, cf. Remark 1.6).

THEOREM 2.6. *Under a CPA attack the (probabilistic public key)* ELGAMAL *cryptosystem satisfies*

(1) *OW under the DH assumption.*
(2) *ASYMMETRY under the DL assumption.*

PROOF. (1) Assume the adversary can decrypt ciphertexts, i.e., from the public key information g^a and the ciphertext $(g^b, g^{ab}x)$ he can computer x. Then he can in particular decrypt $(g^b, 1)$ to obtain g^{-ab} and hence g^{ab}.
(2) If a CPA adversary (i.e., who has full access to $A = g^a$ and $\mathcal{E}_e$) can compute the secret key information a, then he has already solved the DLP.

□

Begin Lect. 19

3. The RABIN cryptosystem

DEFINITION 3.1. The **RABIN cryptosystem** is defined as follows:

- $P = \{0,1\}^k$ and $C = \{0,1\}^{k+1}$.
- $K' = \{(p, q) \mid p, q \text{ distinct primes}, \ p, q \equiv 3 \bmod 4, \ 2^k < pq < 2^{k+1}\}$.
- $\kappa : (p, q) \mapsto pq \in K = \{n \in \mathbb{N} \mid 2^k < n < 2^{k+1} \text{ a BLUM number}\}$.

We encode $\{0,1\}^k \subset (\mathbb{Z}/n\mathbb{Z})^* \subset \{0,1\}^{k+1}$. So we "replace" P and C by $(\mathbb{Z}/n\mathbb{Z})^*$. For $e = n$ and $d = (p, q)$ we define

- $\mathcal{E}_e(x) := x^2 \bmod n$ (not injective!)

- $\mathcal{D}_d(y) :=$ the four square roots of $x^2 \bmod n$ (not uniquely determined!) using the chinese remainder theorem and the simple case "$p \equiv 3 \bmod 4$" in the proof of Theorem 2.18.

EXAMPLE 3.2. Take $p = 3$ and $q = 7$. Then $n = 21$ is the public key of the RABIN cryptosystem. To encrypt the plain text $10 \in (\mathbb{Z}/21\mathbb{Z})^*$ we compute

$$c = 10^2 = 16 \bmod 21.$$

To decrypt $c = 16$ using the secret prime factors $p = 3$ and $q = 7$ we compute the four square roots of 16 modulo 21: We have

$$16^{\frac{p+1}{4}} = 16 \equiv \boxed{1} \bmod 3 \quad \text{and} \quad 16^{\frac{q+1}{4}} = 16^2 \equiv \boxed{4} \bmod 7.$$

Hence

- 1 and $-1 \equiv 2 \bmod 3$ are the square roots of 16 modulo 3.
- 4 and $-4 \equiv 3 \bmod 7$ are the square roots of 16 module 7.

With the chinese remainder theorem we get the four combinations

$$(1,4), \quad (1,3), \quad (2,4), \quad (2,3)$$

and finally the four square roots

$$4, \quad 10, \quad 11, \quad 17,$$

among which we search for the (hopefully unique) human readable "plain text" 10.

THEOREM 3.3. *For the* RABIN *cryptosystem the following implications of assumptions hold (under a CPA attack)*

$$\textit{FACTORING} \implies \textit{OW} \implies \textit{ASYMMETRY}.$$

PROOF. If an attacker can decrypt ciphertexts he can choose random elements $x \in (\mathbb{Z}/n\mathbb{Z})^*$ and compute square roots of x^2. By Theorem 2.21 he obtains a factorization of n, a total break. This proves the first implication. The second implication is trivial. □

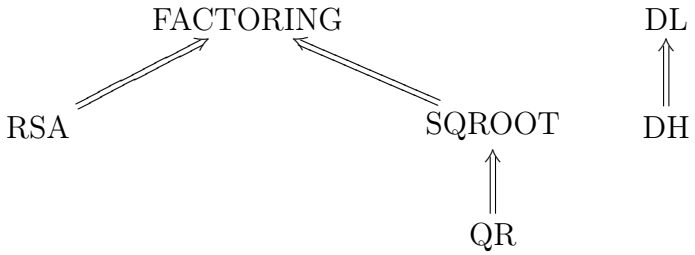
REMARK 3.4. Restricting P to Q_n does not eliminate the non-uniqueness issue: How to find an injective encoding $\{0,1\}^k \to Q_n$?.

4. Security models

REMARK 4.1. We know that the following security models are not fulfilled:

- The BLUM-GOLDWASSER does *not* satisfy the security model IND-CCA2 (exercise).
- The same reasoning as in the proof of Theorem 3.3 shows that the RABIN cryptosystem does *not* fulfill the security model ASYMMETRY-CCA2.

We have the following hierarchy of *assumptions*



REMARK 4.2. It is conjectured that all backward implications in the above diagram hold[4]. These conjectures would imply that the RSA, ELGAMAL, and BLUM-GOLDWASSER cryptosystems do *not* fulfill the security model ASYMMETRY-CCA2:

- RSA=FACTORING $\implies$ RSA $\notin$ ASYMMETRY-CCA2.
- DH=DL $\implies$ ELGAMAL $\notin$ ASYMMETRY-CCA2.
- QR=SQROOT $\implies$ BLUM-GOLDWASSER $\notin$ ASYMMETRY-CCA2 (exercise).

4.a. IND-CCA2. One can modify the BLUM-GOLDWASSER cryptosystem in such a way, that it fulfills IND-CCA2. For this we need the concept of a **one-way hash function**, which is, roughly speaking, a one-way function $H : \{0,1\}^\bullet \to \{0,1\}^k$ for some $k \in \mathbb{N}$.

REMARK 4.3. The modified BLUM-GOLDWASSER cryptosystem (cf. Definition 2.27)

$$\mathcal{E}_e(p) = f_e^\ell(s + H(y)) \cdot \underbrace{(p + r)}_{y}$$

now satisfies, under the QR assumption, the security model IND-CCA2.

4.b. OAEP. We now describe the so-called **optimal asymmetric encryption padding**[5] **(OAEP)** [Wik11d] which is often used to improve the security of public key cryptosystems by preprocessing plaintexts prior to the asymmetric encryption:

- Fix a security parameter $k \in \mathbb{N}$.
- Fix $k_0, k_1 \in \mathbb{N}$ with $\ell := k - k_0 - k_1 > 0$.
- Fix a CSPRBG
$$G : \{0,1\}^{k_0} \to \{0,1\}^{\ell + k_1}$$
.
- Fix a one-way hash function
$$H : \{0,1\}^{\ell + k_1} \to \{0,1\}^{k_0},$$
called the **compression function**.

[4]Recall that we already know that for BLUM numbers the FACTORING assumption $\implies$ SQROOT assumption (cf. Theorem 2.21).

[5]German: Polsterung

For an n-bit plaintext p and seed $s \in \{0,1\}^{k_0}$ return the k-bit concatenated string

$$p' = \underbrace{\left((p \cdot \underbrace{0 \ldots 0}_{k_1}) + G(s) \right)}_{=:y \in \{0,1\}^{n+k_1}} \cdot (s + H(y)).$$

Now one can apply the OWP $f : \{0,1\}^k \to \{0,1\}^k$ of public key cryptosystem to the padded message p'.

DEFINITION 4.4. The *probabilistic* public key cryptosystem obtained by applying the RSA function to an OAEP-preprocessed message p' is called the **RSA-OAEP cryptosystem**.

Assuming the existence of a so-called ideal compression function H one can prove that

THEOREM 4.5 (2001). *The RSA-OAEP cryptosystem satisfies the security model IND-CCA2 under the RSA assumption.*

CHAPTER 7

Primality tests

In this chapter we want to study various sorts of probabilistic and deterministic primality test. Let us denote by $\mathbb{P} \subset \mathbb{N}$ the set of prime numbers.

1. Probabilistic primality tests

1.a. Fermat test. Recall that for $p \in \mathbb{P}$ **Fermat's little theorem**[1] states that $a^{p-1} \equiv 1 \bmod p$ for all $a \in \mathbb{Z} \setminus p\mathbb{Z}$. This yields the so-called **Fermat test**, an elementary probabilistic test for primality, which lies in $O(\log^3 n)$:
If for a natural number n we succeed to find an $a \in (\mathbb{Z}/n\mathbb{Z})^*$ with $a^{n-1} \not\equiv 1 \bmod n$ then n is not a prime. Note that finding an $a \in N_n := ((\mathbb{Z}/n\mathbb{Z}) \setminus \{0\}) \setminus (\mathbb{Z}/n\mathbb{Z})^*$ is hopeless if n is the product of huge primes (compare $n = |\mathbb{Z}/n\mathbb{Z}|$ and $n - \varphi(n) = |N_n| + 1$).

Example 1.1. Let $n = 341$. For

$a = 2$: $2^{340} \equiv 1 \bmod 341$.
$a = 3$: $3^{340} \equiv 56 \bmod 341$.

Hence, 341 is a composite number and 3 is a witness.

Definition 1.2. Let $n \in \mathbb{N}$ and $a \in (\mathbb{Z}/n\mathbb{Z})^*$.

(1) n is called a **pseudoprime with Fermat nonwitness** a if
$$a^{n-1} \equiv 1 \bmod n,$$
i.e., the Fermat test of the primality of n passes for a.
(2) If n is a pseudoprime with Fermat nonwitness a but not prime then a is called a **Fermat liar**.
(3) If the Fermat test of the primality of n fails for a, i.e., if
$$a^{n-1} \not\equiv 1 \bmod n,$$
then a is called a **Fermat witness (for the compositeness) of** n.
(4) n is called a **Carmichael number** if n is a composite number without a Fermat witness, i.e., if all $a \in (\mathbb{Z}/n\mathbb{Z})^*$ are Fermat liars.

Of course, each prime is a pseudoprime for all $a \in (\mathbb{Z}/n\mathbb{Z})^*$. In 1994 it was proven that the set of Carmichael numbers is infinite:

$$561 = 3 \cdot 11 \cdot 17, \quad 1105 = 5 \cdot 13 \cdot 17, \quad 1729 = 7 \cdot 13 \cdot 19, \quad \ldots$$

Begin Lect. 20

[1] This is a special case of Euler's Theorem $a^{\varphi(n)} \equiv 1 \bmod n$, for the case $n = p \in \mathbb{P}$.

LEMMA 1.3. *Let n be a* CARMICHAEL *number and $p \in \mathbb{P}$. Then*

(1) *n is odd.*
(2) *n is square free.*
(3) *$p \mid n \implies p-1 \mid n-1$.*
(4) *n has at least 3 prime factors.*

PROOF. (1) n even $\implies$ $n-1$ odd $\underset{\text{CARMICHAEL}}{\overset{n}{\Longrightarrow}}$ $-1 = (-1)^{n-1} = 1 \in (\mathbb{Z}/n\mathbb{Z})^*$ $\implies$ $n = 2$ prime $\lightning$ (since 2 as a prime is not CARMICHAEL).

(2) Write $n = p^e \cdot n'$, where e is the maximal p-power. Then $\varphi(n) = \varphi(p^e)\varphi(n') = p^{e-1}(p-1)\varphi(n')$. $p^2 \mid n$ implies:
- $p \mid \varphi(n) = |(\mathbb{Z}/n\mathbb{Z})^*| \implies \exists a \in (\mathbb{Z}/n\mathbb{Z})^*$ with $\operatorname{ord}_{(\mathbb{Z}/n\mathbb{Z})^*}(a) = p$.
- $p \mid n \implies p \nmid n-1 \overset{\operatorname{ord}(a)=p}{\Longrightarrow} a^{n-1} \not\equiv 1 \bmod n \implies a$ is a FERMAT witness for $n \implies n$ not CARMICHAEL.

(3) Let p be a prime divisor of n. Since $a^{n-1} = 1$ in $(\mathbb{Z}/n\mathbb{Z})^*$ it follows that $a^{n-1} = 1$ in the factor group $(\mathbb{Z}/p\mathbb{Z})^*$, for all $a \in \mathbb{Z}$ with $(a, n) = 1$. Since $(\mathbb{Z}/p\mathbb{Z})^*$ is cyclic we deduce that $p - 1 = |(\mathbb{Z}/p\mathbb{Z})^*| \mid n - 1$ (choose a to be a primitive element, i.e., a generator of $(\mathbb{Z}/p\mathbb{Z})^*$).

(4) Exercise.

□

The existence of infinitely many CARMICHAEL numbers means that we cannot trust the FERMAT primality test (unless of course it produces a FERMAT witness).

1.b. MILLER-RABIN test. The MILLER-RABIN test makes use of the fact that the equation $a^2 = 1$ has exactly two solutions $a = \pm 1$ over $\mathbb{Z}/n\mathbb{Z}$ if n is a prime (since then $\mathbb{Z}/n\mathbb{Z}$ is a field).

LEMMA 1.4 (MILLER-RABIN). *Let $a \in (\mathbb{Z}/p\mathbb{Z})^*$. Write $p - 1 = 2^s t$ with t odd ($s \geq 0$). Then*

$$a^t \equiv \pm 1 \bmod p \quad \text{or}$$
$$a^{2^r t} \equiv -1 \bmod p \quad \text{for an } 0 < r < s.$$

PROOF. Let $0 \leq s_0 \leq s$ minimal with $a^{2^{s_0} t} = 1$ (recall $a^{p-1} = 1$). We distinguish two cases:

$s_0 = 0 : a^t = 1.$
$s_0 > 0 : a^{2^r t} = -1$ with $r = s_0 - 1 \in \{0, \ldots, s-1\}$.

□

DEFINITION 1.5. Let n be a composite number. Write $n - 1 = 2^s t$ with t odd (so $s \geq 0$). $a \in (\mathbb{Z}/n\mathbb{Z})^*$ is called a **MILLER-RABIN nonwitness** if

$$(\pm 1) \qquad a^t \equiv \pm 1 \bmod n \quad \underline{\text{or}}$$
$$(-1) \qquad a^{2^r t} \equiv -1 \bmod n \quad \text{for } \underline{\text{an}}\ 0 < r < s,$$

otherwise a **MILLER-RABIN witness (for the compositeness of n)**.

EXAMPLE 1.6. Consider $n = 561$: $n-1 = 560 = 2^4 \cdot 35$, so $s = 4$ and $t = 35$. For $a = 2$ we compute

$$\left.\begin{array}{rcr} 2^{35} & \equiv & 263 \bmod 561 \\ 2^{2\cdot 35} & \equiv & 166 \bmod 561 \\ 2^{4\cdot 35} & \equiv & 67 \bmod 561 \\ 2^{8\cdot 35} & \equiv & 1 \bmod 561 \end{array}\right\} \begin{array}{l} \not\equiv 1, -1 \bmod 561 \\ \\ \not\equiv -1 \bmod 561 \\ \\ \end{array}$$

So $a = 2$ is a MILLER-RABIN witness for CARMICHAEL number 561.

REMARK 1.7. If the **generalized RIEMANN hypothesis** holds, then p is a prime if one of the conditions (± 1) or (-1) is fulfilled for each $1 < a < 2\log^2 n$. This turns the probabilistic MILLER-RABIN test into a *deterministic* one. See Remark 1.12 below.

DEFINITION 1.8. For a fixed $n \in \mathbb{N}$ define

$$N := \{\text{MILLER-RABIN } \underline{\text{non}}\text{witness for } n\} \subset (\mathbb{Z}/n\mathbb{Z})^*.$$

The idea is to find a subgroup $U \leq (\mathbb{Z}/n\mathbb{Z})^*$ with $N \subset U$ and to bound the index $(\mathbb{Z}/n\mathbb{Z})^* : U$ from below away from 1. A natural candidate would be

$$U_0 := \{a \in (\mathbb{Z}/n\mathbb{Z})^* \mid a^{n-1} = 1\} = \{\text{FERMAT nonwitness}\} = \ker(x \mapsto x^{n-1}) \leq (\mathbb{Z}/n\mathbb{Z})^*.$$

But we know that the index $(\mathbb{Z}/n\mathbb{Z})^* : U_0$ might be 1:

$$U_0 = (\mathbb{Z}/n\mathbb{Z})^* \iff n \text{ is a prime or a CARMICHAEL number.}$$

LEMMA 1.9. *Let $n = p^\alpha$ for $\alpha \geq 2$. Then $(\mathbb{Z}/n\mathbb{Z})^* : U_0 \geq p$.*

PROOF. $p \mid p^{\alpha-1}(p-1) = \varphi(n) = |(\mathbb{Z}/n\mathbb{Z})^*|$. Then there exists an $a \in (\mathbb{Z}/n\mathbb{Z})^*$ with $\operatorname{ord}(a) = p$. On the other hand $p \mid n \implies p \nmid n-1 \implies a^{n-1} \not\equiv 1 \bmod n \implies a \notin U_0$. The same holds for $a^2, \ldots, a^{p-1}$. Hence $U_0, aU_0, a^2U_0, \ldots, a^{p-1}U_0 \in (\mathbb{Z}/n\mathbb{Z})^*/U_0$ are pairwise different and $(\mathbb{Z}/n\mathbb{Z})^* : U_0 \geq p$. □

THEOREM 1.10. *Let n be a composite odd number with $2, 3 \nmid n$. Then $|N| \leq \frac{\varphi(n)}{4} < \frac{n}{4}$.*

PROOF. Again write $n - 1 = 2^s t$ with t odd (so $s \geq 1$). Set $N_{-1} := \{a \in N \mid a^t \equiv 1 \bmod n\}$ and $N_i := \{a \in N \mid a^{2^i t} \equiv -1 \bmod n\}$ for $0 \leq i < s$. Then $N = \bigcup_{i=-1}^{s-1} N_i$ with $-1 \in N_0 \neq \emptyset$. Set $r := \max\{i \mid N_i \neq \emptyset\} \in \{0, \ldots, s-1\}$ and $m := 2^r t$. In particular, $m \mid \frac{n-1}{2}$ (because $r < s$). For all $a \in N$

$$a^m \equiv \begin{cases} -1 & \text{if } \ a \in N_r \\ 1 & \text{if } \ a \in N_i,\ i < r. \end{cases} \tag{4}$$

Consider the group endomorphism $f : (\mathbb{Z}/n\mathbb{Z})^* \to (\mathbb{Z}/n\mathbb{Z})^*,\ a \mapsto a^m$. Let $n = p_1^{\alpha_1} \cdots p_u^{\alpha_u}$ be the factorization of n as the product of prime powers. The chinese remainder theorem yields the isomorphism

$$(\mathbb{Z}/n\mathbb{Z})^* \cong (\mathbb{Z}/p^{\alpha_1}\mathbb{Z})^* \times \cdots \times (\mathbb{Z}/p^{\alpha_u}\mathbb{Z})^*,$$

identifying $a \in (\mathbb{Z}/n\mathbb{Z})^*$ with the u-tuple $(a \bmod p^{\alpha_1}, \ldots, a \bmod p^{\alpha_u})$. We make use this isomorphism to define the chain of subgroups

$$\begin{array}{lcll}
(\mathbb{Z}/n\mathbb{Z})^* & & & \\
\vee| & & & \\
U_0 & := & \ker(x \mapsto x^{n-1}) & = \{a \in (\mathbb{Z}/n\mathbb{Z})^* \mid a^{n-1} \equiv 1 \bmod n\}, \\
\vee| & & & \\
U_1 & := & f^{-1}(\{(\pm 1, \ldots, \pm 1)\}) & = \{a \in (\mathbb{Z}/n\mathbb{Z})^* \mid a^m \equiv \pm 1 \bmod p_i^{\alpha_i},\ 1 \le i \le u\} \\
\vee| & & & \\
U_2 & := & f^{-1}(\{\pm 1\}) & = \{a \in (\mathbb{Z}/n\mathbb{Z})^* \mid a^m \equiv \pm 1 \bmod n\}, \\
\vee| & & & \\
U_3 & := & f^{-1}(\{1\}) & = \{a \in (\mathbb{Z}/n\mathbb{Z})^* \mid a^m \equiv 1 \bmod n\}.
\end{array}$$

U_1 is a subgroup of U_0 since $m \mid \frac{n-1}{2}$ (see above). The remaining inclusions are obvious as the preimages of inclusions of a chain of subgroups. Since $N \subset U_2$ we want to bound the index $(\mathbb{Z}/n\mathbb{Z})^* : U_2$ from below away from 1. We claim that

$$(\mathbb{Z}/n\mathbb{Z})^* : U_2 \ge 4.$$

To this end we prove that the subgroup $\{(\pm 1, \ldots, \pm 1)\} \le \operatorname{im} f$:
Choose a $b \in N_r \neq \emptyset$, then $f(b) = b^m \equiv -1 \bmod n$, hence, $b^m \equiv -1 \bmod p_i^{\alpha_i}$ for all $i = 1, \ldots, u$. Now let y be an arbitrary element of the elementary ABELian subgroup[2] $\{(\pm 1, \ldots, \pm 1)\} \cong (\mathbb{F}_2^u, +)$, w.l.o.g. we can assume that $y = (1, \ldots, 1, -1, \ldots, -1)$. Then $x := (1, \ldots, 1, b, \ldots, b)$ is a preimage of y under f, i.e., $f(x) = y$. Summing up:

$$U_3 \underbrace{\underbrace{<}_{2} U_2 \underbrace{\le}_{2^{u-1}} U_1}_{2^u} \le U_0 \le (\mathbb{Z}/n\mathbb{Z})^*.$$

We now distinguish three cases:

$u \ge 3$: $U_2 \underbrace{<}_{\ge 4} U_1 \le U_0 \le (\mathbb{Z}/n\mathbb{Z})^*$.

$u = 2$: $U_2 \underbrace{\underbrace{<}_{2} U_1 \le U_0 \underbrace{<}_{\ge 2}}_{\ge 4} (\mathbb{Z}/n\mathbb{Z})^*$, by Lemma 1.3.(4).

$u = 1$: $U_2 \underbrace{=}_{1} U_1 \le U_0 \underbrace{<}_{\ge p \ge 5} (\mathbb{Z}/n\mathbb{Z})^*$, by Lemma 1.9 and the assumptions on n.

This finishes the proof. □

The proof provides a probabilistic primality test in $O(\log^3 n)$. If the MILLER-RABIN test passes for i randomly chosen different a's then the probability of n being prime is greater than $1 - \left(\frac{1}{4}\right)^i$.

EXAMPLE 1.11. Now we demonstrate the difference between the FERMAT test and the MILLER-RABIN test on a trival example. Let $n := 185$:

[2] $2 \nmid n \implies 1 \not\equiv -1 \bmod p_i^{\alpha_i}$ for all $i = 1, \ldots, u$.

a	1 $\in N_{-1}$	-1 $\in N_0$	43 $\in N_1 = N_r$	$N_2 = \emptyset$	36	6	2
a^t	1	-1	$\not\equiv \pm 1$		$\not\equiv \pm 1$	$\not\equiv \pm 1$	$\not\equiv \pm 1$
a^{2t}	1	1	-1		1	$\not\equiv -1$	$\not\equiv -1$
a^{4t}	1	1	1		1	1	$\not\equiv -1$
	MILLER-RABIN nonwitnesses				MILLER-RABIN witnesses		
a^{8t}	1	1	1		1	1	$\not\equiv 1$
	FERMAT nonwitnesses						FERMAT witnesses

REMARK 1.12. One can prove that an

$n < 2047$ is prime $\iff$ (± 1) or (-1) is fulfilled for $a = 2$.
$n < 1373653$ is prime $\iff$ (± 1) or (-1) is fulfilled $\forall a \in \{2, 3\}$.
$\vdots$
$n < \underbrace{341550071728321}_{>3.4\cdot 10^{14}}$ is prime $\iff$ (± 1) or (-1) is fulfilled $\forall a \in \{2, \dots, 17\}$,
i.e., for all of the first 7 primes.

For such n's the probabilistic MILLER-RABIN test becomes a deterministic one.

Begin Lect. 21

2. Deterministic primality tests

2.a. The AKS-algorithm. In this subsection we sketch the **AKS-test**, which was proposed by **A**GRAWAL and his master students **K**AYAL and **S**AXENA in 2002[3] as the first deterministic polynomial runtime primality test.

LEMMA 2.1. *Let $n \in \mathbb{N} \setminus \{1\}$ and $a \in \mathbb{Z}$ coprime to n. Then*[4]

$$n \text{ is prime} \iff (x + a)^n = x^n + a \bmod n.$$

PROOF. $\Longrightarrow$: Let $n \in \mathbb{P}$. The $n \mid \binom{n}{i}$ for all $0 < i < n$. Further, $a^n \equiv a \bmod n$ (recall, n is prime). Then $(x + a)^n = \sum_{i=0}^{n} \binom{n}{i} a^i x^{n-i} \equiv x^n + a^n \equiv x^n + a \bmod n$.
$\Longleftarrow$: Let $(x + a)^n = \sum_{i=0}^{n} \binom{n}{i} a^i x^{n-i} \equiv x^n + a \bmod n$ (*). Let p be a prime divisor of n. Then $\binom{n}{p} := \frac{n(n-1)\cdots(n-p+1)}{p(p-1)\cdots 1}$ is not divisible by n, since $p \mid n$ and $p \nmid (n-1), \dots, (n-p+1)$. Together with $(a, n) = 1$ this implies that $\binom{n}{p} a^p \not\equiv 0 \bmod n$. Hence $n = p$ by (*). □

The idea is to consider the equation

$$(x + a)^n \equiv x^n + a \bmod (n, x^r - 1),$$

for a fixed r, i.e., reduce the coefficients modulo n and the polynomial modulo $x^r - 1$.

We state without proof:

[3] Published 2004 in the Annals of Mathematics: *PRIMES is in P.*
[4] The right hand side is an identity of polynomials in x.

THEOREM 2.2 (AKS-criterion). *Let $2 < n \in \mathbb{N}$ and $r \in \mathbb{N}$ coprime to n. Further let $1 < s \in \mathbb{N}$ with $(a, n) = 1$ for all $a = 1, \dots, s$ and*

$$\text{(AKS)} \qquad \binom{\varphi(r) + s - 1}{s} > n^{2d\lfloor\sqrt{\frac{\varphi(r)}{d}}\rfloor} \quad \text{for all } d \mid \frac{\varphi(r)}{t},$$

where $t := |\langle n \rangle_{(\mathbb{Z}/r\mathbb{Z})^*}|$. If

$$(x + a)^n \equiv x^n + a \bmod (n, x^r - 1), \quad \text{for all } a = 1, \dots, s,$$

then n is a prime power.

To justify an early step in the AKS algorithm below we need a simple corollary of the following Lemma which we also state without proof:

LEMMA 2.3 (CHEBYSHEV[5]). *For $k \geq 2$*

$$\prod_{\substack{p \in \mathbb{P} \\ p \leq 2k}} p > 2^k.$$

COROLLARY 2.4. *Let $N \geq 2$ be a natural number of bit length $k := \lceil \lg N \rceil$. Then there exists a prime $p \leq 2k$ with $p \nmid N$.*

PROOF. $N < 2^k$, by definition of k. Now use the previous Lemma. □

The following version of the AKS-algorithm is due to LENSTRA and BERNSTEIN.

ALGORITHM 2.5. Let $n \in \mathbb{N} \setminus \{1\}$ be an odd number.

(1) Compute (the factors of) $N := 2n(n-1)(n^2-1)\cdots(n^{4\lceil \lg n \rceil^2} - 1)$ of bit length $k := \lceil \lg N \rceil$.
(2) Find the smallest prime $r \leq 2k$ with $r \nmid N$. If, before reaching the smallest prime r, you discover that
   (a) n is a prime $(n < r)$ then **return**: n prime.
   (b) a prime $p \mid n$ $(p < r)$ then **return**: n composite.
(3) If there is an element $a \in \{1, \dots, r\}$ with $(x + a)^n \not\equiv x^n + a \bmod (n, x^r - 1)$ then **return**: n composite.
(4) If there is an element $a \in \{1, \dots, \log_r n\}$ with $\sqrt[a]{n} \in \mathbb{N}$ then **return**: n composite.
(5) **return**: n prime.

THEOREM 2.6. *Algorithm 2.5 is correct and has polynomial runtime, i.e., it lies in $O(f(\ell))$, where f is a polynomial in $\ell := \lceil \lg n \rceil$.*

PROOF. (1) The factors $n - 1, n^2 - 1, \dots, n^{4\lceil \lg n \rceil^2} - 1$ can be computed with less than $4\ell^2 \lg(4\ell^2)$ multiplications. Further $\lg N \leq 1 + \lg n + (\lg n) \sum_{i=1}^{4\ell^2} i \leq 1 + \ell + \ell \frac{(4\ell^2+1)4\ell^2}{2}$, in particular, the bit length $k := \lceil \lg N \rceil$ is polynomial in ℓ.
(2) The runtime of listing all primes $\leq 2k$ is a polynomial in ℓ. If case (a) or (b) occur then the algorithm terminates.

[5]German: TSCHEBYSCHEFF

(3) Claim (i): $t := |\langle n\rangle_{(\mathbb{Z}/r\mathbb{Z})^*}| > 4\ell^2$. Proof: If not then there would exist an $i \in \{1,\dots,4\ell^2\}$ with $n^i \equiv 1 \bmod r \implies r \mid n^i - 1 \mid N \lightning$.
Consider the AKS-criterion for $s = r$. The previous steps guarantee that $(a, n) = 1$ for all $a = 1,\dots,r = s$ (recall, $r \nmid n$ since $n \mid N$).

Claim (ii): The (AKS) inequality is fulfilled. Proof: From $d \le \frac{\varphi(r)}{t} \overset{\text{Claim (i)}}{<} \frac{\varphi(r)}{4\ell^2}$ it follows that

$$(*) \qquad 2d\lfloor\sqrt{\frac{\varphi(r)}{d}}\rfloor \le 2d\sqrt{\frac{\varphi(r)}{d}} = \sqrt{4d\varphi(r)} \overset{\text{here}}{<} \frac{\varphi(r)}{\ell} \le \frac{\varphi(r)}{\lg n}.$$

Further $2 \mid N \implies r \ge 3 \implies \varphi(r) = r - 1 \ge 2 \implies$

$$\binom{\varphi(r)+s-1}{s} = \binom{\varphi(r)+r-1}{r} = \binom{2\varphi(r)}{\varphi(r)+1} \ge 2^{\varphi(r)} = n^{\frac{\varphi(r)}{\lg n}} \overset{(*)}{>} n^{2d\lfloor\sqrt{\frac{\varphi(r)}{d}}\rfloor}.$$

The AKS-criterion (Theorem 2.2) can now be applied proving the correctness of step (3). Note that exponentiating with n is polynomial in ℓ.

(4) If this step is reached then n is a prime power by the AKS-criterion. That this step is also polynomial in ℓ is an easy exercise.

□

CHAPTER 8

Integer Factorization

Consult [Wik11e] for the current state of the RSA factorization challenge. Be aware that n in RSA-n refers to the number decimal digits if $n \leq 500$ or $n = 617$, e.g., RSA-232, otherwise to the number of binary digits, e.g., RSA-768, which also has 232 decimal digits, and was successfully factored in December 12, 2009 [KAF+10]:

> The following effort was involved. We spent half a year on 80 processors on polynomial selection. This was about 3% of the main task, the sieving, which was done on many hundreds of machines and took almost two years. ...

There is no polynomial algorithm known up to date.

1. Pollard's $p-1$ method

Definition 1.1. Let $B \in \mathbb{N}$. An $n \in \mathbb{N}$ is called

(1) B-**smooth** if all its prime divisors are less than or equal to B.
(2) B-**powersmooth** if all its prime *power* divisors are less than or equal to B.

We now describe **Pollard's $p-1$ method** to factor an integer $n \in \mathbb{N}$ where p is a prime divisor of n:
If $(a, n) = 1$ for an $a \in \mathbb{Z}$ then $a^{p-1} \equiv 1 \bmod p$ (Fermat's little theorem). Assume $p-1$ is B-powersmooth for a "small" bound $B \in \mathbb{N}$. Then $p - 1 \mid \operatorname{lcm}\{1, \ldots, B\}$ and hence $a^{\operatorname{lcm}\{1,\ldots,B\}} \equiv 1 \bmod p$, or equivalently $p \mid a^{\operatorname{lcm}\{1,\ldots,B\}} - 1$. In particular:

$$(a^{\operatorname{lcm}\{1,\ldots,B\}} - 1, n) > 1$$

and we have found a divisor[1] of n.

A good heuristic value for B is $B \geq n^{\frac{1}{2}\left(1-\frac{1}{e}\right)} \approx n^{0.316}$. So for a fixed B the method should be able to cover all $n \leq B^{2\frac{e}{e-1}} \approx B^{3.164}$. Typically one chooses $B \approx 10^6$ which allows handling numbers $n \leq 10^{19}$.

Exercise 1.2. Describe a factorization algorithm using the above idea. Use your algorithm to factor 1633797455657959.

Begin Lect. 22

[1] Of course, the gcd might be n.

2. Pollard's ρ method

Let n be a composite number and $x_0, x_1, \ldots$ be a sequence in $\mathbb{Z}/n\mathbb{Z}$. For a prime divisor p of n set $y_k := x_k \bmod p$. Since $\mathbb{Z}/p\mathbb{Z}$ is finite two y's, say y_μ and $y_{\mu+\lambda}$ ($\mu \in \mathbb{Z}_{\geq 0}$ and $\lambda \in \mathbb{N}$), will eventually coincide: $y_{\mu+\lambda} = y_\mu \in \mathbb{Z}/p\mathbb{Z}$, or equivalently, $p \mid x_{\mu+\lambda} - x_\mu$. But then $d := (x_{\mu+\lambda} - x_\mu, n) > 1$ is factor of n. The trivial (improbable) case $d = n$ only occurs if already $x_{\mu+\lambda} = x_\mu \in \mathbb{Z}/n\mathbb{Z}$.

If the sequence $x_0, x_1, \ldots \in \mathbb{Z}/n\mathbb{Z}$ is chosen randomly then $y_0, y_1, \ldots$ will be a random sequence in $\mathbb{Z}/p\mathbb{Z}$, and the **birthday problem** [Wik11a] will imply that after approximately $\sqrt{p}$ random choices two y's, say y_μ and $y_{\mu+\lambda}$, will coincide with probability $\frac{1}{2}$.

To produce a pseudo-random sequence, Pollard suggested a recursion using a polynomial $f \in \mathbb{Z}[x]$. For an initial value $x_0 \in \mathbb{Z}/n\mathbb{Z}$ set $x_{k+1} := f(x_k) \bmod n$. For $y_k := x_k \bmod p$ still $y_{k+1} \equiv f(y_k) \bmod p$. One often uses the nonlinear polynomial $f := x^2 + c$ with $c \neq 0 \in \mathbb{Z}/n\mathbb{Z}$, typically $c = \pm 1$.

Recall that any recursive sequence in a finite set eventually becomes periodic, giving this method its name ρ. There are several **cycle-detection algorithms**. The two most prominent ones are **Floyd's tortoise**[2] **and hare**[3] **algorithm** and **Brent's algorithm** [Wik11b]. Their goal is to

(1) avoid too much comparisons.
(2) find the minimal μ and the period length λ.

Of course, only the first goal is relevant for us, where the comparison step in the cycle-detection algorithm has to be replaced by the gcd computation: $(y_{\mu+\lambda} - y_\mu, n)$.

The following version of the Pollard's ρ method is based on Floyd's algorithm:

Algorithm 2.1 (Pollard's ρ method). Given a composite number $n \in \mathbb{N}$ the following algorithm returns a nontrivial factor of n or fail.

(1) $x := 1$, $z := 1$, $d := 1$
(2) while $d = 1$ do
    - $x := f(x)$
    - $z := f(f(z))$
    - $d := (z - x, n)$
    - if $d = n$ then **return** fail
(3) **return** d

3. Fermat's method

Fermat's method for factoring a composite number n tries to write it as the difference of two squares $n = x^2 - y^2$, yielding the factorization $n = (x + y)(x - y)$. Indeed, for a composite *odd* number $n = ab$ such a representation always exists: Setting $x := \frac{a+b}{2}$ and $y := \frac{a-b}{2}$ we recover $a = x + y$ and $b = x - y$.

[2] German: Schildkröte
[3] German: Feldhase

EXAMPLE 3.1. Let $n = 7429$. $x = 227$ and $y = 210$ satisfy $x^2 - y^2 = n$ with $x - y = 17$ and $x + y = 437$. Hence $n = 17 \cdot 437$.

4. Dixon's method

Dixon's method for factoring a composite odd number n is a relaxation of FERMAT's one. It is based on the following fact: If x, y are integers with

$$x^2 \equiv y^2 \bmod n \quad \text{and } x \not\equiv \pm y \bmod n$$

then $(x - y, n)$ (and $(x + y, n)$) is a nontrivial divisor of n.

EXAMPLE 4.1. Let $n = 84923$. Taking $x = 20712$ and $y = 16800$ we compute $x^2 - y^2 = 1728 \cdot n$, $x - y = 3912$ (and $x + y = 37512$). Hence $(x - y, n) = 163$ and $n = 163 \cdot 521$.

ALGORITHM 4.2 (DIXON's algorithm). Given a composite number $n \in \mathbb{N}$ the following algorithm returns a nontrivial factor of n or fail.

(1) $F := \{p_1, \dots, p_k\} \subset \mathbb{P}$ be a set of k distinct "small" primes, where k is "small". We call F a **factor base**[4].
(2) Find $x_1, \dots, x_m \in \mathbb{N}$ $(m > k)$ such that $x_i^2 = p_1^{e_{1i}} \cdots p_k^{e_{ki}} \bmod n$.
(3) Set $v_i := ((e_{1i}, \dots, e_{ki}) \bmod 2) \in \mathbb{F}_2^k$ for $i = 1, \dots, m$. Solve the $\mathbb{F}_2$-linear system

$$\sum_{i=1}^{m} \varepsilon_i v_i = 0 \in \mathbb{F}_2^k.$$

(4) Set $(a_1, \dots, a_k) := \frac{1}{2} \sum_{i=1}^{m} \varepsilon_i (e_{1i}, \dots, e_{ki}) \in \mathbb{Z}_{\geq 0}^m$. Define

$$x := \prod_{i=1}^{m} x_i^{\varepsilon_i} \quad \text{and} \quad y := p_1^{a_1} \cdots p_k^{a_k}.$$

Then

$$\boxed{x^2} = \prod_{i=1}^{m} x_i^{2\varepsilon_i} \equiv \prod_{i=1}^{m} (p_1^{\varepsilon_i e_{1i}} \cdots p_k^{\varepsilon_i e_{ki}}) = p_1^{\sum_{i=1}^m \varepsilon_i e_{1i}} \cdots p_k^{\sum_{i=1}^m \varepsilon_i e_{ki}} = p_1^{2a_1} \cdots p_k^{2a_k} = \boxed{y^2} \bmod n.$$

(5) If $x \not\equiv y \bmod n$ then **return** $(x - y, n)$ else **return** fail.

EXAMPLE 4.3. Again let $n = 7429$. Take $F = \{2, 3, 5, 7\}$.

$$\begin{aligned} 87^2 &\equiv 2^2 \cdot 5 \cdot 7 \bmod 7429 \\ 88^2 &\equiv 3^2 \cdot 5 \cdot 7 \bmod 7429 \end{aligned}$$

So $v_1 = v_2 = (0, 0, 1, 1) \in \mathbb{F}_2^4$ and $\varepsilon_1 = \varepsilon_2 = 1$ since $v_1 + v_2 \in \mathbb{F}_2^4$. Hence $(a_1, \dots, a_4) = (1, 1, 1, 1) \in \mathbb{Z}_{\geq 0}^4$, $x = 87 \cdot 77 \equiv 227 \bmod n$ and $y = 2 \cdot 3 \cdot 5 \cdot 7 \equiv 210 \bmod n$. As we saw in Example 3.1 above $(x - y, n) = (17, n) = 17$.

EXERCISE 4.4. Treat Example 4.1 using DIXON's algorithm.
Hint: Try $x_1 := 513$ and $x_2 := 537$.

[4] German: Faktorbasis

5. The quadratic sieve

The **quadratic sieve (QS)**[5] of POMERANCE is an optimization of DIXON's method. The goal is to find x_i's close to the square root $\sqrt{n}$ such that x_i^2 is B-smooth mod n for a "small" bound $B \in \mathbb{N}$ (see Algorithm 4.2, step (2)).

As candidates for these smooth x_i^2 consider the quantities

$$Q(a) := (\lfloor\sqrt{n}\rfloor + a)^2 - n \in \mathbb{Z},$$

for a in some **sieve interval** $S := \{-s, \dots, s\} \subset \mathbb{Z}$ **with width** s.

As $Q(a)$ might be negative a slight modification of DIXON's method turns out to be useful:

EXERCISE 5.1. Describe a modified version of DIXON's method allowing the factor base F to include -1 (the "sign").

By definition, $Q(a)$ is a square mod n, that is $Q(a) \equiv x^2 \bmod n$ with $x = \lfloor\sqrt{n}\rfloor + a$. The key observation is the content of the following

REMARK 5.2. Let $q \in \mathbb{N} \setminus \{1\}$ and $x, a \in \mathbb{Z}$.

(1) $x^2 \equiv n \bmod q \iff q \mid Q(a)$ for $a = x - \lfloor\sqrt{n}\rfloor$.
(2) $q \mid Q(a) \implies q \mid Q(a + kq)$ for all $k \in \mathbb{Z}$.

PROOF. (1) is trivial by the definition of $Q(a)$.

(2) More generally, $Q(x + kq) \equiv Q(x) \bmod q$ since computing mod q is a ring homomorphism $\mathbb{Z}[x] \twoheadrightarrow \mathbb{Z}/q\mathbb{Z}[x]$.

□

In words: q is a divisor of $Q(a)$ for $a = x - \lfloor\sqrt{n}\rfloor$ iff the equation $x^2 \equiv n \bmod q$ is solvable. And if q is a divisor of $Q(a)$ the it is a divisor of $Q(a + kq)$ for all $k \in \mathbb{Z}$.

For the composite odd number n define

$$\mathbb{P}(n) := \left\{ p \in \mathbb{P} \mid p = 2 \text{ or } \left(\frac{n}{p}\right) = 1 \right\}.$$

This is the set of all primes for which the equation $x^2 \equiv n \bmod p$ is solvable and[6] $p \nmid n$.

ALGORITHM 5.3. Fix a bound $B \in \mathbb{N}$ and a factor base $F \subset \mathbb{P}(n)$. For a sieve interval $S := \{-s, \dots, s\} \subset \mathbb{Z}$ the following algorithm returns the list of those $Q(a)$ with $a \in S$ which are B-powersmooth with prime factors in[7] F.

(1) Set $L_a := Q(a)$ for all $a \in S$.
(2) For all $p \in F$:
  (a) Solve[8] the equation $x^2 \equiv n \bmod p$. (Hence, by Remark 5.2, $p \mid Q(a)$ for all $a \in \mathbb{Z}$ with $a \equiv x - \lfloor\sqrt{n}\rfloor \bmod p$.)

[5] German: Sieb

[6] Recall, $\left(\frac{n}{p}\right) = 0$ means that $p \mid n$ — so we have found a prime divisor of n and we are done.

[7] One says, "which factor over F".

[8] Cf. proof of Theorem 5.2.18.

(b) **Sieve**: For all $a \in S$ with $a \equiv \pm x - \lfloor\sqrt{n}\rfloor \bmod p$, where x is a solution of the equation $x^2 \equiv n \bmod p$: Replace L_a by the quotient $\frac{L_a}{p^e}$, where p^e is the maximal power dividing L_a which is $\leq B$.

(3) **return** the list of those $Q(a)$ with $a \in S$ for which $L_a = 1$.

EXERCISE 5.4. Let $n = 4417$. Compare the needed sieve width s for $F = \{-1, 2, 3, 7\}$ and $F = \{-1, 2, 3, 13\}$.

CHAPTER 9

Elliptic curves

Begin Lect. 23

1. The projective space

Let K be a field. The set

$$\mathbb{A}^n(K) = K^n = \{(x_1, \ldots, x_n) \mid x_i \in K\}$$

is called the **affine space** of dimension n over K. If K is clear from the context then we will simply write $\mathbb{A}^n$ instead.

Two distinct points $P, Q \in \mathbb{A}^n(K)$ uniquely determine an **(affine) line**

$$\overline{PQ} := P + K \cdot (Q - P) := \{P + k(Q - P) \mid k \in K\}.$$

containing both of them.

The **projective space** of dimension n over K is defined as the set

$$\mathbb{P}^n(K) := \left(K^{n+1} \setminus \{0\}\right) / K^* := \{K^* \cdot x \mid x \in K^{n+1} \setminus \{0\}\}.$$

Again we write $\mathbb{P}^n$ if the field K is clear from the context.

A point P in $\mathbb{P}^n(K)$ can thus be identified with a 1-dimensional subspace of K^{n+1}. More generally, define the **trace** of a subset $Z \subset \mathbb{P}^n(K)$ to be the subset

$$Z^* \subset K^{n+1} = \{x \in K^{n+1} \setminus \{0\} \mid K^* \cdot x \in Z\} \cup \{0\}.$$

This gives a one-to-one correspondence between subsets of $\mathbb{P}^n(K)$ and those subsets of the underlying vector space K^{n+1} which are unions of 1-dimensional subspaces.

EXAMPLE 1.1. A **(projective) line** in $\mathbb{P}^n(K)$ is the set of all 1-dimensional subspaces of a 2-dimensional subspace $L \leq K^{n+1}$. We identify the projective line with its trace L. Two distinct points $P, Q \in \mathbb{P}^n(K)$ determine a unique projective line $\overline{PQ} := P + Q$ passing through both of them. $P + Q$ is the 2-dimensional span of P, Q, both viewed as 1-dimensional subspaces of K^{n+1}.

1.a. Homogenous coordinates and affine charts. If $x = (x_0, \ldots, x_n) \in K^{n+1} \setminus \{0\}$ then for the point $P = K \cdot x$ we write

$$P = (x_0 : \ldots : x_n).$$

We call $x_0, \ldots, x_n \in K$ the **homogeneous coordinates** of P. They are uniquely determined by P up to a common nonzero factor:

$$(x_0 : \ldots : x_n) = (y_0 : \ldots : y_n) \iff (y_0, \ldots, y_n) = k \cdot (x_0, \ldots, x_n) \text{ for some } k \in K^*.$$

EXAMPLE 1.2. Fix a field K.

(1) $\mathbb{A}^1 = \{a | a \in K\}$ and $\mathbb{P}^1 = \{(x : y) \mid (x, y) \in K^2 \setminus \{0\}\}$. Identifying $\mathbb{A}^1$ with the affine subspaces $\{(1, y) \mid y \in K\} \subset K^2$ or $\{(x, 1) \mid x \in K\} \subset K^2$ defines two embeddings

$$\varphi_0 : \mathbb{A}^1 \to \mathbb{P}^1, y \mapsto (1 : y),$$
$$\varphi_1 : \mathbb{A}^1 \to \mathbb{P}^1, x \mapsto (x : 1).$$

Visualize the two "screens" and the "light rays" in $K^2 = \mathbb{R}^2$! These embeddings are also called (standard) **affine charts** of $\mathbb{P}^1$.
The elements of the image $\varphi_0(\mathbb{A}^1) \subset \mathbb{P}^1$ (resp. $\varphi_1(\mathbb{A}^1) \subset \mathbb{P}^1$) are called **affine points w.r.t. the chart** φ_0 (resp. φ_1). The point $(0 : 1) \in \mathbb{P}^1$, corresponding to the y-axis in K^2, is the only non-affine point w.r.t. φ_0. It is called the **point at infinity**[1] **w.r.t.** φ_0. Analogously for $(1 : 0)$ and φ_1. Summing up:

$$\mathbb{P}^1 = \underbrace{\varphi_0(\mathbb{A}^1)}_{\text{affine points}} \dot\cup \underbrace{\{(0 : 1)\}}_{\text{pt at } \infty} = \underbrace{\varphi_1(\mathbb{A}^1)}_{\text{affine points}} \dot\cup \underbrace{\{(1 : 0)\}}_{\text{pt at } \infty}.$$

The partial inverses are given by the "projections"

$$\varphi_0^{-1} : \mathbb{P}^1 \setminus \{(0 : 1)\} \to \mathbb{A}^1, (x : y) \mapsto \frac{y}{x},$$
$$\varphi_1^{-1} : \mathbb{P}^1 \setminus \{(1 : 0)\} \to \mathbb{A}^1, (x : y) \mapsto \frac{x}{y}.$$

(2) $\mathbb{A}^2 = \{(a, b) | a, b \in K\}$ and $\mathbb{P}^2 = \{(x : y : z) \mid (x, y, z) \in K^3 \setminus \{0\}\}$. We have three standard charts

$$\varphi_0 : \mathbb{A}^2 \to \mathbb{P}^2, (y, z) \mapsto (1 : y : z),$$
$$\varphi_1 : \mathbb{A}^2 \to \mathbb{P}^2, (x, z) \mapsto (x : 1 : z),$$
$$\varphi_2 : \mathbb{A}^2 \to \mathbb{P}^2, (x, y) \mapsto (x : y : 1).$$

We will usually identify $\mathbb{A}^2$ with its image under φ_2 and call its elements the **affine points (w.r.t.** φ_2**)**. The complementary set

$$U := \mathbb{P}^2 \setminus \varphi_2(\mathbb{A}^2) = \{(x : y : 0) \mid (x, y) \in K^2 \setminus \{0\}\} \subset \mathbb{P}^2$$

is a projective line, called the **line at infinity**[2] **(w.r.t. the chart** φ_2**)**. We will usually refer to φ_2. Visualize in $K^3 = \mathbb{R}^3$.

EXERCISE 1.3. Generalize to the n-dimensional case.

1.b. Algebraic sets and homogenization. The **vanishing set**

$$V(F) := \{(x, y) \in K^2 \mid F(x, y) = 0\}$$

of a polynomial $F \in K[x, y]$ is an example of a so-called **algebraic set**.

[1] German: unendlich ferner Punkt
[2] German: unendlich ferne Gerade

EXAMPLE 1.4. Visualize in $K^2 = \mathbb{R}^2$ the vanishing sets of the degree 2 polynomials

$$F = x^2 + y^2 - 1, \quad F = x^2 - y^2, \quad \text{and } F = x^2 - y.$$

For each F visualize the image $\varphi_2(V(F)) \subset \mathbb{P}^2(\mathbb{R})$ by its trace in $K^3 = \mathbb{R}^3$ and obtain special (singular) ruled surfaces[3].

DEFINITION 1.5. Let K be a field.

(1) A polynomial $F \in K[x_0, \dots, x_n]$ of degree d is called **homogeneous** if all its monomials are of degree d. It follows that $F(\lambda x_0, \dots, \lambda x_n) = \lambda^d F(x_0, \dots, x_n)$.
(2) For a polynomial $F \in K[x, y]$ define the **homogenization** $F^* \in K[x, y, z]$ **(w.r.t. φ_2)** by setting

$$F^*(x, y, z) := z^d F(\frac{x}{z}, \frac{y}{z}) \in K[x, y, z],$$

where $d = \deg F$. The homogenization is a homogeneous polynomial of degree d.

REMARK 1.6. Let $F \in K[x, y]$. The trace of the image $\varphi_2(V(F))$ coincides with the *affine* points of the vanishing set of the homogenized polynomial F^*:

$$\varphi_2(V(F)) = V(F^*) \setminus U.$$

EXAMPLE 1.7. Homogenizing the polynomials in Example 1.4 we get

$$F^* = x^2 + y^2 - z^2, \quad F^* = x^2 - y^2, \quad \text{and } F^* = x^2 - yz.$$

Visualize $V(F^*)$ in $K^3 = \mathbb{R}^3$.

(1) $V(x^2 + y^2 - z^2)$ does not intersect the line at infinity $U = \{z = 0\}$ if $K = \mathbb{R}$. What happens for K algebraically closed (e.g., $K = \mathbb{C}$)?
(2) $V(x^2 - y^2)$ has exactly two points at infinity, namely $(1 : 1 : 0)$ and $(1, -1 : 0)$.
(3) $V(x^2 - yz)$ meets U in the point $(0 : 1 : 0)$ (but with "multiplicity" 2). Visualize by a perspective drawing in $K^2 = \mathbb{R}^2$ with the line at infinity being the horizon.

EXERCISE 1.8. Discuss the points at infinity of $V(x^2 + y^2 - z^2)$, $V(x^2 - y^2)$, and $V(x^2 - yz)$ w.r.t. the two other standard charts φ_0 and φ_1.

In what follows we will often write $F \in K[x, y]$ and mean $V(F^*) \in \mathbb{P}^2$. An $L(x, y) = R(x, y)$ will stand for the polynomial $F := L - R$, so $V(L = R) := V(L - R)$.

1.c. Elliptic curves. Let K be a field.

DEFINITION 1.9. The equation

$$E^* : y^2 z + a_1 xyz + a_3 yz^2 = x^3 + a_2 x^2 z + a_4 xz^2 + a_6 z^3, \quad a_i \in K.$$

is called the **(homogeneous) WEIERSTRASS equation**. It is the homogenization of the **(affine) WEIERSTRASS equation**

$$E : y^2 + a_1 xy + a_3 y = x^3 + a_2 x^2 + a_4 x + a_6, \quad a_i \in K.$$

[3]German: Regelflächen

Denote the vanishing set of E^* by

$$E(K) := V(E^*) \subset \mathbb{P}^2.$$

The point $(0:1:0)$ is the only point of $E(K)$ at infinity, i.e., at $z=0$. It has "multiplicity" 3 since $E^*(x,y,0) : x^3 = 0$.

REMARK 1.10 (Normal forms). Depending on the characteristic of the field K one can transform the WEIERSTRASS equation into a simpler form by a coordinate change:

(1) If $\operatorname{char} K \neq 2$ then complete the square by substituting $y \to y - \frac{a_1 x + a_3}{2}$ to obtain the normal form
$$y^2 = x^3 + a_2' x^2 + a_4' x + a_6',$$
the right hand side being a cubical univariate polynomial (e.g., $a_2' = a_2 + \frac{a_1^4}{4}$).
(2) If $\operatorname{char} K \neq 2, 3$ then the substitution $x \to x - \frac{1}{3} a_2'$ finally yields
$$y^2 = x^3 + ax + b.$$

EXAMPLE 1.11. We fix $K = \mathbb{R}$, so let $f(x) = x^3 + ax + b \in \mathbb{R}[x]$. We distinguish three cases:

(1) $f(x)$ has exactly one real root. Visualize $V(y^2 = f(x))$ for $a > 0, a = 0, a < 0$.
(2) $f(x)$ has exactly three real roots. Visualize.
(3) $f(x)$ has one simple and one double root. Visualize.
(4) $f(x)$ has one triple root. Visualize.

EXAMPLE 1.12. The elliptic curve $E : y^2 = x^3 + 2x - 1$ over $K = \mathbb{F}_5$ has 6 points:

x	0	1	2	3	4
$f(x)$	4	2	1	2	1
y	2, 3	–	1, 4	–	1, 4

$E(\mathbb{F}_5) = \{(0,2), (0,3), (2,1), (2,4), (4,1), (4,4)\}$.

1.c.i. *Singularities.* Let K be a field, E a WEIERSTRASS equation, and

$$\begin{aligned} F &:= y^2 + a_1 xy + a_3 y - (x^3 + a_2 x^2 + a_4 x + a_6), \\ F^* &:= y^2 z + a_1 xyz + a_3 yz^2 - (x^3 + a_2 x^2 z + a_4 xz^2 + a_6 z^3) \end{aligned}$$

be the corresponding defining (affine resp. homogenous) polynomials.

DEFINITION 1.13. Let $P = (x_0 : y_0 : z_0) \in E(K)$ be a point on the elliptic curve. W.l.o.g. assume $x_0 \neq 0$.

- P is called a **singular point (of E)** or simply **singular** if
$$\frac{\partial F^*}{\partial y}(x_0, y_0, z_0) = \frac{\partial F^*}{\partial z}(x_0, y_0, z_0) = 0.$$
- $E(K)$ (or E) is called **singular** if there is a singular point $P \in E(K)$, otherwise **nonsingular** or **smooth**.

REMARK 1.14.

(1) $(0:1:0)$ is not a singular point: $\frac{\partial F^*}{\partial z}(0,1,0) = (y^2+a_1xy+2a_3yz-a_2x^2-2a_4xz-3a_6z^2)(0,1,0) = 1 \neq 0$.
(2) $\operatorname{char} K \neq 2,3$: $\operatorname{disc}(x^3+ax+b) = -(4a^3+27b^2)$.
(3) $\operatorname{char} K \neq 2$: $E : y^2 = f(x)$. Then E is singular $\iff$ $\operatorname{disc} f = 0$.

DEFINITION 1.15. E is called an **elliptic curve** if E is smooth.

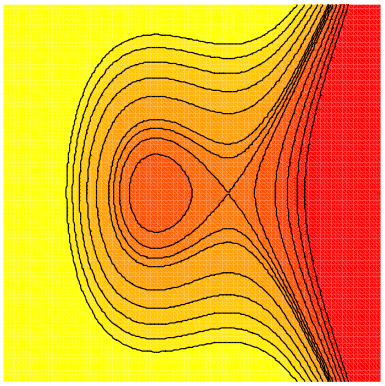


FIGURE 1. A family of elliptic curves.

Begin Lect. 24

2. The group structure $(E,+)$

Let K be a field, $\bar{K}$ its algebraic closure, and E an elliptic curve over K. In this section $\mathbb{P}^2$ refers to $\mathbb{P}^2 := \mathbb{P}^2(\bar{K})$ (the $\bar{K}$ points of E).

THEOREM 2.1. *Let $L \subset \mathbb{P}^2$ be a line. Then $|L \cap E(\bar{K})| = 3$, counted with multiplicity.*

PROOF. The idea is the following: Substituting a parametrization of the line yields an equation of degree 3 in one indeterminate (parameter the line). This has exactly three roots in $\bar{K}$ counted with multiplicity. It is not obvious how this argument takes care of points at infinity. So we give an elementary proof. Let

$$L = \{(x:y:z) \mid ax+by+cz = 0\} \in \mathbb{P}^2 \text{ with } (a,b,c) \neq (0,0,0).$$

Case 1: $a = b = 0$: $L = \{(x:y:0) \in \mathbb{P}^2\}$ is the line at infinity. To compute $L \cap E(\bar{K})$ set $z = 0$ in E to obtain $x^3 = 0$. The infinite far point $(0:1:0)$ is a root of multiplicity 3.
Case 2: $a \neq 0$ or $b \neq 0$: $L = \{(x,y) \mid ax+by = -c\} \cup \{(b:-a:0)\}$.
  Case a: $b \neq 0$: $(b:-a:0) \neq (0:1:0)$, hence $(b:-a:0) \notin E(\bar{K})$. Now we compute the affine points by substituting $y = -\frac{ax+c}{b}$ in E to obtain a cubic polynomial in x with 3 roots in $\bar{K}$ (counted with multiplicity).

Case b: $b = 0, a \neq 0$: $(0:1:0) \in E(\bar{K}) \cap L$. To determine the affine points substitute $x = -\frac{c}{a}$ in E and obtain a quadratic polynomial in y that has two roots in $\bar{K}$ (counted with multiplicity). This gives 3 points.

□

REMARK 2.2. BÉZOUT's theorem states two curves of degree n and m which do not have a common component intersect in nm points counting multiplicities. The previous Theorem is a special case of BÉZOUT's theorem. That two distinct lines intersect in exactly one points is also a special case.

2.a. Tangents. Let $F^* = y^2z + a_1xyz + a_3yz^2 - (x^3 + a_2x^2z + a_4xz^2 + a_6z^3)$ and E the corresponding elliptic curve.

DEFINITION 2.3. Let $P \in E(\bar{K})$. The line

$$T_P := \{(u:v:w) \in \mathbb{P}^2 \mid \frac{\partial F^*}{\partial x}(P) \cdot u + \frac{\partial F^*}{\partial y}(P) \cdot v + \frac{\partial F^*}{\partial z}(P) \cdot w = 0\}$$

is called the tangent of E at P. One can rewrite the defining equation as $\nabla F^*(P) \cdot \begin{pmatrix} u \\ v \\ w \end{pmatrix} = 0$, where $\nabla = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})$.

REMARK 2.4.

(1)
$$\frac{\partial F^*}{\partial x} = a_1yz - 3x^2 - 2a_2xz - a_4z^2.$$
$$\frac{\partial F^*}{\partial y} = 2yz + a_1xz + a_3z^2.$$
$$\frac{\partial F^*}{\partial z} = y^2 + a_1xy + 2a_3yz - a_2x^2 - 2a_4xz - 3a_6z^2.$$

(2) If $P = (0:1:0)$ then $\nabla F^*(P) = (0,0,1)$. Hence $T_P = \{(u:v:w) \mid w = 0\} = U$, the line at infinity.

(3) If $P = (x:y:1)$ then
$$\frac{\partial F^*}{\partial x}(P) = a_1y - 3x^2 - 2a_2x - a_4.$$
$$\frac{\partial F^*}{\partial y}(P) = 2y + a_1x + a_3.$$
$$\frac{\partial F^*}{\partial z}(P) = y^2 + a_1xy + 2a_3y - a_2x^2 - 2a_4x - 3a_6.$$

Verify that $\nabla F^*(P) \cdot \begin{pmatrix} u \\ v \\ w \end{pmatrix} = 3F(x,y)$.

From (2) and (3) we deduce that $P \in T_P$ for all $P \in E(\bar{K})$. On can prove that $P \in E(\bar{K})$ is a multiple intersection point of T_P and $E(\bar{K})$ wit multiplicity at least 2. We have verified this for infinite point $(0:1:0)$ which is an intersection point with multiplicity 3.

DEFINITION 2.5.

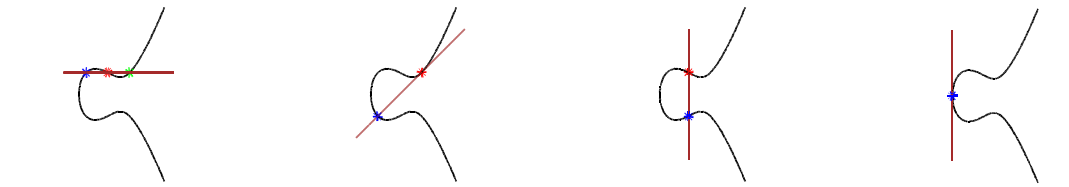
(1) Fix $O := (0:1:0) \in E(\bar{K})$ (O like origin).
(2) For $P, Q \in E(\bar{K})$ define $P * Q$ by $E(\bar{K}) \cap L = \{P, Q, P * Q\}$, where

$$L := \begin{cases} \overline{PQ} & \text{if} \quad P \neq Q \\ T_P & \text{if} \quad P = Q \end{cases}.$$

(3) Finally define the operation $+$ for $P, Q \in E(\bar{K})$ by

$$P + Q := (P * Q) * O.$$

Visualize!
Caution: This is different from the sum of traces $P + Q = \overline{PQ}$.



$P+Q+R=O$ $P+Q+Q=O$ $P+Q+O=O$ $P+P+O=O$

FIGURE 2. The group law on the $\mathbb{R}$-points of the elliptic curve $E : y^2 = x^3 - x + 1$

REMARK 2.6. Let P, Q, R be points on $E(\bar{K})$. Then

(1) $*$ and $+$ are commutative.
(2) $(P * Q) * P = Q$.
(3) $O * O = O$.
(4) Let $L \in \mathbb{P}^2$ be an arbitrary line, $E(\bar{K}) \cap L = \{P, Q, R\}$, then $(P + Q) + R = O$.
(5) $P + O = P$.
(6) $P + Q = O \iff P * Q = 0$.
(7) $+$ is associative (!)
(8) $(E(\bar{K}), +)$ is an ABELian group with neutral element O and $-P = P * O$.
(9) $E(K)$ is a subgroup of $E(\bar{K})$.

PROOF. (1) By construction.
(2) Definition of $*$.
(3) Remark 2.4.

(4) $(P+Q)+R := (((\underbrace{(P*Q)}_{R}*O)*R)*O \overset{(2)}{=} O*O \overset{(3)}{=} O.$

(5) $P+O = (P*O)*O = (O*P)*O = P.$

(6) $P+Q=O \iff (P*Q)*O = O \overset{(5)}{\iff} P*Q = O*O = O.$

(7) Without a further idea this leads to a lengthy case by case distinction. There exist wonderful geometric[4] ideas to prove the associativity. We will see one of them next semester in the seminar.

(8) Follows from (5) and (6)+(7)

(9) Let E be defined over K and $P,Q \in E(K)$. Then $L, L\cap E$ is defined over K. Then $P*Q$ is as the third root of $L\cap E(\bar{K})$ is also in K. The is a special case of the following simple fact:
If $f\in K[x]$ with $\deg f = r$ and if $r-1$ roots are in K then the last root is in K. □

2.b. A formula for $-P := P*O$ where $P\neq O$. Let $P=(x_0,y_0)=(x_0:y_0:1)$ (affine) and $O=(0:1:0)$. Next, we want to determine $\overline{PO}$. The following equivalences are immediate: $(x:y:1)\in\overline{PO} \iff \begin{pmatrix}x\\y\\1\end{pmatrix} \in \left\langle \begin{pmatrix}x_0\\y_0\\1\end{pmatrix}, \begin{pmatrix}0\\1\\0\end{pmatrix}\right\rangle \iff x=x_0$. So $\overline{PO} = \{(x_0:y:1)\mid y\in K\}\cup\{O\}$.
$(x_0:y:1)\in E(K) \iff F(x_0,y)=0 \iff y=y_0,y_1$ where $F(x_0,y)=(y-y_0)(y-y_1)$. Coefficient matching yields: $-y_0-y_1 = a_1x_0+a_3$, so $y_1=-y_0-a_1x_0-a_3$. Finally,

$$\boxed{-P = P*O = (x_0 : -y_0-a_1x_0-a_3 : 1).}$$

For the most important special case: $E: y^2=f(x)$, e.g. when $\operatorname{char} K\neq 2$. Then $a_1=a_3=0$ and

$$\boxed{-(x_0,y_0)=(x_0,-y_0).}$$

2.c. A formula for $P*Q$ where $P,Q\neq O$. Let $P=(x_1,y_1)=(x_1:y_1:1)$ and $Q=(x_2,y_2)=(x_2:y_2:1)$ be two affine points. By definition, $P*Q=O \iff P*O=Q \iff x_1=x_2$ and $y_1+y_2+a_1x_1+a_3=0$. For each line $L\subset\mathbb{P}^2$ we have: $O=(0:1:0)\in L \iff L\cap\mathbb{A}^2$ is parallel to the y-axis. Let without w.l.o.g. $P*Q\neq O$ (otherwise $Q=-P$). Set

$$L := \begin{cases} \overline{PQ} & \text{if } P\neq Q \\ T_P & \text{if } P=Q \end{cases}.$$

Then $L\cap\mathbb{A}^2 = \{(x,y)\mid y=\lambda x+\nu\}$ for some $\lambda,\nu\in K$.
Case $P\neq Q$: Then

$$\boxed{\lambda = \frac{y_2-y_1}{x_2-x_1} \text{ and } \nu = y_1-\lambda x_1.}$$

[4] The most elegant proof uses the theory of divisors [**Har77**, Chapter IV, Section 4].

Case $P = Q = (x_1, y_1) = (x_1 : y_1 : 1)$:

$$T_P \cap \mathbb{A}^2 = \{(x,y) \mid \frac{\partial F^*}{\partial x}(P)\cdot x + \frac{\partial F^*}{\partial y}(P)\cdot y + \frac{\partial F^*}{\partial z}(P) = 0\}$$

and

$$\frac{\partial F^*}{\partial x}(P) = a_1 y - 3x^2 - 2a_2 x - a_4.$$
$$\frac{\partial F^*}{\partial y}(P) = 2y + a_1 x + a_3.$$

Solving for y we recover the slope[5]

$$\boxed{\lambda = -\frac{\frac{\partial F^*}{\partial x}(P)}{\frac{\partial F^*}{\partial y}(P)} = \frac{a_1 y - 3x^2 - 2a_2 x - a_4}{2y + a_1 x + a_3}.}$$

Further $L \cap E(K) = L \cap \mathbb{A}^2 \cap E(K)$. Write $F(x, \lambda x + \nu) = -(x - x_1)(x - x_2)(x - x_3)$. Coefficient matching at x^2 yields: $x_1 + x_2 + x_3 = \lambda^2 + a_1 x - a_2$. Finally, $P * Q = (x_3, y_3)$ with

$$x_3 = \lambda^2 + a_1\lambda - a_2 - x_1 - x_2,$$
$$y_3 = \lambda x_3 + \nu = \lambda(x_3 - x_1) + y_1.$$

2.d. A formula for $P + Q$ where $P, Q \neq O$. We now put together the above computations for $P + Q = (P * Q) * O$

$$\boxed{P + Q = (\underbrace{\lambda^2 + a_1\lambda - a_2 - x_1 - x_2}_{=x_3}, \underbrace{-y_1 + \lambda(x_1 - x_3) - a_1 x_1 - a_3}_{=y_3}).}$$

For the most important special case: $y^2 = x^3 + ax + b$, i.e., $a_1 = a_2 = a_3 = 0$, $a_4 = a$, and $a_6 = b$. Then

$$\boxed{\lambda = \frac{y_2 - y_1}{x_2 - x_1} \text{ if } P \neq Q \quad \text{or} \quad \lambda = \frac{3x_1^2 + a}{2y_1} \text{ if } P = Q.}$$

Finally,

$$\boxed{\begin{aligned} x_3 &= \lambda^2 - x_1 - x_2, \\ y_3 &= -y_1 + \lambda(x_1 - x_3). \end{aligned}}$$

EXAMPLE 2.7. We take $K = \mathbb{F}_5 \cong \mathbb{Z}/5\mathbb{Z}$ and $y^2 = x^3 + 2x - 1$. Verify that

$$E(\mathbb{F}_5) = \{(0,2), (0,3), (2,1), (2,4), (4,1), (4,4)\} \cup \{O\}.$$

- $P = (0,2) \implies -P = (0,-2) = (0,3)$.
- $Q = (2,1) \implies P + Q = (2,4)$.
- $P + P = 2P = (4,1)$.
- $(0,3) + (2,1) = (4,1)$.

[5]German: Steigung

REMARK 2.8. The choice of $O := (0 : 1 : 0)$ was arbitrary but convenient for two reasons:

(1) It is the unique non-affine point w.r.t. the fixed coordinate system.
(2) It satisfies $O * O = O$.

Begin Lect. 25

3. Elliptic curves over finite fields

3.a. Squares in finite fields. Finding square roots in the finite field $\mathbb{F}_q$ is the first step to find $\mathbb{F}_q$-points (i.e., points in $E(\mathbb{F}_q)$) on an elliptic curve E.

REMARK 3.1. Let p be a prime and q a power of p.

(1) $\mathrm{sq} : \mathbb{F}_q^* \to \mathbb{F}_q^*, x \mapsto x^2$ has kernel $\ker \mathrm{sq} = \{\pm 1\}$. Hence

$$|(\mathbb{F}_q^*)^2| = \begin{cases} \frac{q-1}{2} & \text{for } q \text{ odd} \\ q-1 & \text{for } q \text{ even} \end{cases}.$$

(2) Define for q odd the **quadratic character**

$$\chi : \mathbb{F}_q^* \to \{\pm 1\} \subset \mathbb{F}_q^*, x \mapsto x^{\frac{q-1}{2}}.$$

That $\chi(a) = 1 \iff a \in (\mathbb{F}_q^*)^2$, easily follows from (1) generalizing Theorem 5.2.4 by EULER.

ALGORITHM 3.2. Given an *odd* prime power q and an $a \in \mathbb{F}_q^*$ with $\chi(a) = 1$.
return $b \in \mathbb{F}_q^*$ with $b^2 = a$ using the algorithm in the proof of Theorem 5.2.18, slightly generalized for prime powers.

3.b. Counting points. Let E be an elliptic curve over $\mathbb{F}_q$ and $N := |E(\mathbb{F}_q)|$.

THEOREM 3.3 (HASSE-WEIL). *Let $a := q+1-N$ and α, β be the roots of the quadratic polynomial $x^2 - ax + q$. Then*

$$|a| \leq 2\sqrt{q}.$$

Further,

$$|E(\mathbb{F}_{q^m})| = q^m + 1 - (\alpha^m + \beta^m)$$

for all $m \in \mathbb{N}$.

PROOF. A good reference is [**Was08**, Theorem 4.2]. We will prove this theorem in the seminar, next semester. □

REMARK 3.4. For $N = |E(\mathbb{F}_q)|$ the HASSE-WEIL theorem estimates

$$q + 1 - 2\sqrt{q} \leq N \leq q + 1 + 2\sqrt{q}.$$

If q is a *prime* one can show that each natural number in this interval occurs as the order of an elliptic curve $E(\mathbb{F}_q)$.

EXAMPLE 3.5. Let $E : y^2 = x^3 + 7x + 1$ be an elliptic curve of $\mathbb{F}_{101}$. It is possible to show that the point $(0, 1)$ has order 116 (see Algorithm 3.8 below), so $N = |E(\mathbb{F}_{101})|$ is a multiple of 116. But the HASSE-WEIL theorem says that

$$81 \leq 101 + 1 - 2\sqrt{101} \leq N \leq 101 + 1 + 2\sqrt{101} \leq 123,$$

and the only multiple of 116 in this range is 116. Hence, $E(\mathbb{F}_{101})$ is cyclic of order $N := |E(\mathbb{F}_{101})| = 116$, generated by $(0, 1)$.

EXAMPLE 3.6. With little effort we can determine all the points of $E : y^2 + xy = x^3 + 1$ over the small field $\mathbb{F}_2$:

$$E(\mathbb{F}_2) = \{O, (0, 1), (1, 0), (1, 1)\}.$$

Determining all $\mathbb{F}_{2^{101}}$-points is extremely hard (and of course useless). With $a = -1$ and $q = 2$ the HASSE-WEIL theorem counts for us the number of points:

$$\begin{aligned} |E(\mathbb{F}_{2^{101}})| &= 2^{101} + 1 - \left[\left(\frac{-1 + \sqrt{-7}}{2}\right)^{101} + \left(\frac{-1 - \sqrt{-7}}{2}\right)^{101}\right] \\ &= 2^{101} + 1 - 2969292210605269 \\ &= 2535301200456455833701195805484 \approx 2.5 \cdot 10^{30}. \end{aligned}$$

THEOREM 3.7 ([Was08, Theorem 4.3]). *Let p be a prime and $q = p^n$ and define $N := q + 1 - a$ for some $a \in \mathbb{Z}$ with $|a| \leq 2\sqrt{q}$. Then there is an elliptic curve E defined over $\mathbb{F}_q$ with $|E(\mathbb{F}_q)| = N$ if and only if a satisfies one of the following conditions:*

1. *$(a, p) = 1$.*
2. *n is even and $a = \pm 2\sqrt{q}$.*
3. *n is even, $p \not\equiv 1 \bmod 3$, and $a = \pm\sqrt{q}$.*
4. *n is odd, $p = 2$ or $p = 3$, and $a = \pm p^{\frac{n+1}{2}}$.*
5. *n is even, $p \not\equiv 1 \bmod 4$, and $a = 0$.*
6. *n is odd and $a = 0$.*

Let $P \in E(\mathbb{F}_q)$. We want to find the order of P as an element of the group $E(\mathbb{F}_q)$. We know that $NP = O$. Of course we don't know N yet, but we know that $q + 1 - 2\sqrt{2} \leq N \leq q + 1 + 2\sqrt{q}$. One could of course try all values in this interval. This would take $4\sqrt{q}$ steps. The following algorithm runs in about $4q^{\frac{1}{4}}$ steps:

ALGORITHM 3.8 (**Baby step, giant step**, [Was08, §4.3.4]). Given $P \in E(\mathbb{F}_q)$ compute its order.

1. Compute $Q = (q + 1)P$.
2. Choose an integer $m > q^{\frac{1}{4}}$. Compute and save the points jP for $j = 0, \ldots, m$ (baby steps).
3. Compute the points

$$Q + k(2mP) \text{ for } k = -m, \ldots, m$$

(giant steps) until $Q + k(2mP) = \pm jP$. Then $MP = O$ with $M := q + 1 + 2mk \mp j$.

(4) Factor $M = p_1^{e_1} \cdots p_r^{e_r}$. Compute $(M/p_i)P$ for $i = 1, \ldots, r$. If $(M/p_i)P = O$ for some i, replace M with M/p_i and repeat the step until $(M/p_i)P \neq O$ for all i. Then M is the order of P.

To determine $N = |E(\mathbb{F}_q)|$ continue as follows:

(5) Repeat the previous steps with randomly chosen points in $E(\mathbb{F}_q)$ until the least common multiple of the element order divides only one integer N in the HASSE-WEIL interval. It is then the group order N.

PROOF. Justifying the existence of k, j in (3) is an exercise. □

EXAMPLE 3.9. Let E be the elliptic curve $y^2 = x^3 - 10x + 21$ over $\mathbb{F}_{557}$ and $P = (2, 3) \in E(\mathbb{F}_{557})$.

(1) $Q = 558P = (418, 33)$.
(2) Let $m = 5 > 557^{\frac{1}{4}}$. The list of jP's ("baby steps") is
$$O,\ (2,3),\ (58,164),\ (44,294),\ (56,339),\ (132,364).$$
(3) For $k = 1$ we discover that $Q + k(2mP) = (2, 3)$ matches the list for $j = 1$ ("giant step"). Hence $(q + 1 + 2mk - j)P = 567P = O$.
(4) Factor $567 = 3^4 \cdot 7$. Compute $(567/3)P = 189P = O$. Factor $189 = 3^3 \cdot 7$. Compute $(189/3)P = (38, 535) \neq O$ and $(189/7)P = (136, 360) \neq O$. Therefore, 189 is the order of P.
(5) This suffices to conclude that $|E(\mathbb{F}_{557})| = 567$.

REMARK 3.10. There exists an algorithm due to **SCHOOF** which computes the number of points on an elliptic curves over finite fields $\mathbb{F}_q$ in about $\log^8 q$ steps (cf. [**Was08**, §4.5]).

3.c. Finding points.

ALGORITHM 3.11. Let q be a power of an odd prime and $E : y^2 = f(x)$ an elliptic curve of $\mathbb{F}_q$ (cf. Remark 1.10). The following algorithm returns an $\mathbb{F}_q$-point of E:

(1) Choose $x \in \mathbb{F}_q$ randomly until $f(x) \in (\mathbb{F}_q)^2$ (test $f(x) = 0$ or $f(x)^{\frac{q-1}{2}} = 1 \in \mathbb{F}_q$).
(2) Compute a square root y with $y^2 = f(x)$ using Algorithm 3.2.

REMARK 3.12.
For finding points on elliptic curves over $\mathbb{F}_{2^n}$ see [**Kob98**, Exercise 6.2.2, page 136].

3.d. The structure of the group $(E, +)$.

THEOREM 3.13 (Structure Theorem for finitely generated ABELian groups). *Let A be a finitely generated* ABEL*ian group. Then there exist $r, k \in \mathbb{N}_0$ and $n_1, \ldots, n_k \in \mathbb{N}$ with $n_i \mid n_{i+1}$ such that*
$$A \cong \mathbb{Z}^r \times \mathbb{Z}/n_1\mathbb{Z} \times \cdots \mathbb{Z}/n_k\mathbb{Z}.$$
*r is called the **rank**[6] of A and the n_i's are called the **determinantal divisors** of A.*

[6]Of course, A is finite if and only if its rank is 0.

THEOREM 3.14 (Structure Theorem for elliptic curves over finite fields). *There exists natural numbers n_1, n_2 with $n_1 \mid n_2$ such that*

$$E(\mathbb{F}_q) \cong \mathbb{Z}/n_1\mathbb{Z} \times \mathbb{Z}/n_2\mathbb{Z}.$$

PROOF. We will prove this in the seminar. The statement includes the case $(n_1, n_2) = (1, n)$ in which case $E(\mathbb{F}_q) \cong \mathbb{Z}/n\mathbb{Z}$. □

One can sharpen this further

THEOREM 3.15 (Structure Theorem for elliptic curves over finite fields (refined version)). *Let p, q, and N be as in Theorem 3.7. Write $N = p^e n_1 n_2$ with $p \nmid n_1 n_2$ and $n_1 \mid n_2$ (possibly $n_1 = 1$). There exists an elliptic curve E over $\mathbb{F}_q$ such that*

$$E(\mathbb{F}_q) \cong \mathbb{Z}/p^e\mathbb{Z} \times \mathbb{Z}/n_1\mathbb{Z} \times \mathbb{Z}/n_2\mathbb{Z}$$

if and only if

(1) *$n_1 \mid q - 1$ in the cases (1), (3), (4), (5), (6) of Theorem 3.7.*
(2) *$n_1 = n_2$ in casse (2) of Theorem 3.7.*

These are all groups that occur as $E(\mathbb{F}_q)$.

EXAMPLE 3.16. Here are the possible isomorphism types of elliptic curves $E(\mathbb{F}_5)$:

- HASSE-WEIL states that $2 \leq N = E(\mathbb{F}_q) \leq 10$. This leaves us with the following possibilities (according to Theorem 3.13):

  $$\mathbb{Z}/2,\ \mathbb{Z}/3,\ \mathbb{Z}/4,\ \mathbb{Z}/2 \times \mathbb{Z}/2,\ \mathbb{Z}/5,\ \mathbb{Z}/6 \cong \mathbb{Z}/2 \times \mathbb{Z}/3,\ \mathbb{Z}/7,\ \mathbb{Z}/8,$$
  $$\mathbb{Z}/2 \times \mathbb{Z}/4,\ \underline{\mathbb{Z}/2 \times \mathbb{Z}/2 \times \mathbb{Z}/2},\ \mathbb{Z}/9,\ \underline{\mathbb{Z}/3 \times \mathbb{Z}/3},\ \mathbb{Z}/10 \cong \mathbb{Z}/2 \times \mathbb{Z}/5.$$

- The above refined structure theorem 3.15 rules out the underlined groups.

Begin Lect. 26

4. LENSTRA'S FACTORIZATION METHOD

We now come to one of the amazing applications of elliptic curves. It can be viewed as an ingenious variation of POLLARD'S $p - 1$ method (see Section 8.1), but one that comes with an extra degree of freedom: Since it is based on elliptic curves, one can vary the used curve, and even run the algorithm for different curves in parallel.

Let n be the composite number that we want to factorize. LENSTRA'S method relies on the choice of random elliptic curves E_i over the ring $\mathbb{Z}/n\mathbb{Z}$ with random points on $E_i(\mathbb{Z}/n\mathbb{Z})$ (the group law of elliptic curves over rings is more involved [**Was08**, §2.11]). If one starts with the elliptic curve then finding a point involves finding a square root modulo n which, as we saw in Lemma 5.2.20, is computationally equivalent to factoring n. To overcome this problem the choice of the curve cannot be independent from the choice of the point:
Choose a random element a mod n and a random pair $P = (u, v)$ mod n. *Then* compute

$$b \equiv v^2 - u^3 - au \bmod n.$$

The random elliptic curve

$$y^2 = x^3 + ax + b$$

has the $\mathbb{Z}/n\mathbb{Z}$-point $P = (u, v)$.

ALGORITHM 4.1 (LENSTRA). The following algorithm takes a composite number n as its input and returns a factor of n or fail.

(1) Choose several[7] random elliptic curves $E_i : y^2 = x^3 + a_i x + b_i$ over $\mathbb{Z}/n\mathbb{Z}$ together with $\mathbb{Z}/n\mathbb{Z}$-points P_i (as above).
(2) Choose a bound B ($\approx 10^8$) and compute $\operatorname{lcm}\{1, \dots, B\}P_i$ (or $(B!)P_i$) on $E_i(\mathbb{Z}/n\mathbb{Z})$ for each i.
(3) If step (2) fails because some slope λ does not exist modulo n then we have found a factor of n. **return** this factor.
(4) **return** fail.

REMARK 4.2.

- One can use Remark 3.4 to explain why this method often yields a nontrivial factor. For details see [**Was08**, p. 193].
- The method is very effective in finding prime factors $< 10^{40}$. But in cryptographic applications ones uses prime numbers with at least 100 digits. In this range the quadratic sieve (QS) and the number field sieve methods (NFS) outperform LENSTRA's method. Nevertheless, it is still useful in intermediate steps of several attacks.

EXAMPLE 4.3 ([**Was08**, Example 7.1]). Let us demonstrate the method to factor $n = 4453$. Choose the elliptic curve $y^2 = x^3 + 10x - 2 \bmod n$ with $\mathbb{Z}/n\mathbb{Z}$-point $P = (1, 3)$. Try to compute $3P$. First compute $2P$. The slope of the tangent at P is

$$\frac{3x^2 + 10}{2y} = \frac{13}{6} \equiv 3713 \bmod 4453.$$

Hence $2P = (x, y)$ with

$$x \equiv 3713^2 - 2 \equiv 4332 \bmod 4453, \text{ and } y \equiv -3713(x - 1) - 3 \equiv 3230 \bmod 4453.$$

To compute $3P$ we add P to $2P$. The slope is

$$\frac{3230 - 3}{4332 - 1} = \frac{3227}{4331}.$$

But 4331 is not invertible modulo n since $(4331, 4453) = 61 \neq 1$, and we have found a factor of n. This gives the factorization $4453 = 61 \cdot 73$. For the elliptic curve this means that

$$E(\mathbb{Z}/4453\mathbb{Z}) = E(\mathbb{Z}/61\mathbb{Z}) \times E(\mathbb{Z}/73\mathbb{Z})$$

be the chinese remainder theorem.

The method worked since $\operatorname{ord}_{E(\mathbb{Z}/61\mathbb{Z})} P = 3$ while $\operatorname{ord}_{E(\mathbb{Z}/73\mathbb{Z})} = 64$. The improbable coincidence of these two orders would have produced 0 mod n as the denominator of the slope and the gcd would've been the trivial one $n = 4453$.

[7] ... depending on your computing resources.

5. Elliptic curves cryptography (ECC)

The hardness of solving the DLP in an ABELian group strongly depends on the way how the group is represented. For example $\mathbb{F}_p^* \cong (\mathbb{Z}/(p-1)\mathbb{Z}, +)$. The DLP is hard in the former and trivial in the latter.

The main usage of elliptic curves in cryptography is to provide an alternative realization of ABELian groups for which all known attacks on the DLP quickly lose their strength (see Chapter 10).

In principle, any cryptosystem or signature scheme which is based on the DLP in $\mathbb{F}_q^*$ can be used with elliptic curves. The DIFFIE-HELLMAN cryptosystem defined in 6.2.3 is a good example for this. There is even an elliptic curve analogue of RSA [**Was08**, §6.8] that was suggest by KOYAMA-MAURER-OKAMOTO-VANSTONE.

5.a. A coding function for elliptic curves. But in order to do this we need a way to encode messages by points on an elliptic curve:
Let $E : y^2 = x^3+ax+b$ be an elliptic curve over $\mathbb{F}_p$ with $p \gg 100$ an odd prime and let m be a message encoded as a number $0 \leq m < \frac{p}{100}$. For $j = 0, \ldots, 99$ compute $s_j = x_j^3 + ax_j + b$ with $x_j := 100m + j$ and check if s_j is a square (iff $s_j^{\frac{p-1}{2}} \equiv 1 \bmod p$). Stop at the first such j. Computing a square root y_j by Algorithm 3.11 yields an element $P = (x_j, y_j)$ on $E(\mathbb{F}_p)$. Recover the number m encoding the message as $\lfloor \frac{x_j}{100} \rfloor$. Since s_j is a random element of $\mathbb{F}_p^*$ the probability of lying in $(\mathbb{F}_p^*)^2$ is $\frac{1}{2}$. So the probability of failing to find a point P after 100 trials is 2^{-100}.

This method was proposed by NEAL KOBLITZ.

CHAPTER 10

Attacks on the discrete logarithm problem

In this chapter we list two attacks on the DLP. One is specific to the group $\mathbb{F}_q^*$ and the other is independent of the representation of the ABELian group.

1. Specific attacks

1.a. The index calculus. This attack is specific to the group $\mathbb{F}_q^*$. We will deal with the case $\mathbb{F}_p^*$ where p is an odd prime. The general case $\mathbb{F}_q^*$ requires more work.

So let p be an odd prime and g a generator of the cyclic group $\mathbb{F}_p^*$. The discrete logarithm (cf. Definition 1.3)

$$\log_g : \mathbb{F}_p^* \to \mathbb{Z}/(p-1)\mathbb{Z}$$

defined by $g^{\log_g y} = y \bmod p$ is an isomorphism of cyclic groups, in particular,

$$\log_g(y_1 y_2) = \log_g(y_1) + \log_g(y_2) \mod p-1. \tag{$\log_g$}$$

As we mentioned above, the DLP in the source group is hard, while the DLP in the range group is trivial.

The idea of the attack is to compute $\log_g(y)$ for "small" y's and then to use the identity ($\log_g$) to compute $\log_g$ for arbitrary y's. Note that

$$\log_g(-1) \equiv \frac{p-1}{2} \bmod p-1.$$

This is a reformulation of the equation $g^{\frac{p-1}{2}} \equiv -1 \mod p$.

ALGORITHM 1.1 (**index calculus**). The algorithm takes $y \in \mathbb{F}_p = \langle g \rangle$ as input and returns the discrete logarithm $\log_g y \in \mathbb{Z}/(p-1)\mathbb{Z}$ or fail.

(1) Choose a bound $B \in \mathbb{N}$ and the factor base $F(B) = \{p \in \mathbb{P} | p \le B\} \cup \{-1\} = \{-1, p_1, \dots, p_k\}$.

(2) Search for y_i's for which the **lift** $b_i \in \mathbb{Z}$ of $g^{y_i} \in \mathbb{F}_p^*$ can be factored over $F(B)$, i.e., $b_i \equiv (-1)^{e_{i0}} \prod_{j=1}^k p_j^{e_{ij}} \bmod p$. Do this until the set of equations

$$y_i \equiv \underbrace{\log_g\left((-1)^{e_{i0}}\right)}_{\equiv 0 \text{ or } \frac{p-1}{2}} + e_{i1}\log_g p_1 + \cdots + e_{ik}\log_g p_k \mod p-1$$

can be solved for the vector of k unknowns $(\log_g p_1, \dots, \log_g p_k)$. If the search fails **return** fail.

(3) Search for a j for which the lift $b \in \mathbb{Z}$ of $g^j \cdot y \in \mathbb{F}_p^*$ can be factored over $F(B)$, i.e., $b \equiv (-1)^{a_0} \prod_{j=1}^{k} p_j^{a_j} \bmod p$. If the search fails **return** fail.
(4) Solve the equation

$$j + \log y \equiv \underbrace{\log_g\left((-1)^{a_0}\right)}_{\equiv 0 \text{ or } \frac{p-1}{2}} + a_1 \log_g p_1 + \cdots + a_k \log_g p_k \mod p-1$$

and **return** $\log y$.

EXAMPLE 1.2 ([Was08, Example 5.1]). We will demonstrate the method by computing the discrete logarithm $\log_3 37 \bmod \underbrace{1217}_{p}$. Choosing $B = 13$ gives the factor base $F(B) = \{-1, 2, 3, 5, 7, 11, 13\}$. We compute

$$\begin{aligned} 3^1 &\equiv 3 \mod 1217 \\ 3^{24} &\equiv -2^2 \cdot 7 \cdot 13 \\ 3^{25} &\equiv 5^3 \\ 3^{30} &\equiv -2 \cdot 5^2 \\ 3^{54} &\equiv -5 \cdot 11 \\ 3^{87} &\equiv 13. \end{aligned}$$

This yields in particular: $\log_3 2 \equiv 216 \bmod \underbrace{1216}_{p-1}$, $\log_3 7 \equiv 113 \bmod 1216$, and $\log_3 11 \equiv 1059 \bmod 1216$. Finally $3^{16} \cdot 37 \equiv 2^3 \cdot 7 \cdot 11 \bmod 1217$. Therefore, $\log_3 37 \equiv 3\log_3 2 + \log_3 7 + \log_3 11 - 16 \equiv 588 \bmod 1216$.

REMARK 1.3. Several remarks are in order:

- The method was successfully used to compute discrete logarithms modulo a 120-digit prime.
- Finding the appropriate y_i's and the j can be done using a version of the quadratic sieve (QS) or the number field sieve (NFS) as in Section 8.5.
- In contrast to $\mathbb{F}_p^* = (\mathbb{Z}/p\mathbb{Z})^*$, elements of $E(\mathbb{F}_p)$ are rarely reductions of elements in $E(\mathbb{Z})$ (or $E(\mathbb{Q})$). It is thus widely believed that the index calculus cannot be adapted to solve the DLP for elliptic curves.

Begin Lect. 27

2. General attacks

By general attacks we mean attacks applicable for all representations of a finite ABELian group. POLLARD's ρ method can be modified to give a general *probabilistic* algorithm to solve the DLP [Was08, §5.2.2]. From the many known general attacks on the DLP we only treat a modified version of the baby-step-giant-step algorithm.

2.a. Baby step, giant step. Let G be an additively written ABELian group with (known) order N. W.l.o.g. we can assume that $G = \langle P \rangle$.

ALGORITHM 2.1 (SHANKS). The following *deterministic* algorithm takes as input an element $Q \in G$ and returns the discrete logarithm $k := \log_P Q$, i.e., the minimal $k \in \mathbb{N}$ with $kP = Q$.

(1) Fix an $m > \sqrt{N}$ and compute mP.
(2) Compute and save the list $L := (iP \mid 0 \leq i < m)$ ("baby steps").
(3) For $j = 0, \ldots, m-1$ compute the points $Q - jmP$ ("giant steps") until one matches an element in L, say $iP = Q - jmP$.
(4) Since $Q = kP$ with $k \equiv i + jm \bmod N$ **return** (the smallest such) k.

PROOF OF CORRECTNESS. Since $m^2 > N$ it follows that $0 \leq k < m^2$. Write $k = i + jm$ with $0 \leq i < m$ and $0 \leq j = \frac{k-i}{m} < m$. Then $Q - jmP = kP - jmP = iP$. □

EXAMPLE 2.2 ([Was08, Example 5.2]). Let $G = E(\mathbb{F}_{41})$, where $E : y^2 = x^3 + 2x + 1$. Let $P = (0, 1)$ and $Q = (30, 40)$. The group order N is at most 54 by HASSE-WEIL, so set $m = 8$. The list of baby steps iP for $0 \leq i < 8$ consists of the points

$$O,\ (0,1),\ (1,39),\ (8,23),\ (38,38),\ (23,23),\ (20,28),\ (26,9).$$

We start calculation the big steps for $j = 0, 1, 2, \ldots$

$$(30,40), (9,25), (26,9),$$

and stop since $(26, 9)$ matches $7P$ in the list. With $i = 7$ and $j = 2$ we compute the discrete logarithm $k = 7 + 2 \cdot 8 = 23$. Indeed $Q = 23P$.

CHAPTER 11

Digital signatures

Under a **digital signature** or **digital signature scheme (DSS)** we understand a mathematical scheme for ensuring the *authenticity* of data. A digital signature should lose its validity if anything in the signed data was altered. This is one of the major advantages compared with ink on paper signatures.

1. Definitions

DEFINITION 1.1. An **asymmetric signature** is a 5-tuple $(M, S, \kappa : K' \to K, \mathcal{S}, \mathcal{V})$ where

- M is called the set of **messages**,
- S the set of **signatures**,
- $\kappa : K' \to K, d \mapsto e$ a bijective map from the set of **secret signing keys** to the set of **public verification keys**,
- $\mathcal{S} = (\mathcal{S}_d : M \rightsquigarrow S)_{d \in K'}$ a family of multi-valued polynomial algorithms called the **signing algorithm**,
- $\mathcal{V} = (\mathcal{V}_e : M \times S \to \{0,1\})_{e \in K}$ a family of polynomial algorithms called the **signature verifications** satisfying $\mathcal{V}_{\kappa(d)}(m, \mathcal{S}_d(m)) = 0$.

Normally one signs only hash values of messages for performance reasons: "hash-then-sign". We will come to hash functions below.

DEFINITION 1.2. We list the following attacks models on a DSS:

(1) **Key-only attack**: The attacker only knows the public verification key of the signer.
(2) **Known-message attack (KMA)**: The attacker receives some messages (he didn't choose) and their corresponding signatures.
(3) **(Adaptive) chosen-message attack (CMA)**: The attacker is allowed to (adaptively) choose messages and receives the corresponding signatures.

DEFINITION 1.3. We list the following goals of an attack on a DSS:

(1) **Total break**: Recover the signing key.
(2) **Universal forgery**[1]: Forge signatures of any message.
(3) **Existential forgery**: Forge a signature for some message (without the ability to do this for any message).

[1]German: Fälschung

The strongest security model among the above combinations is the security against existential forgery under an adaptive chosen message attack.

2. Signatures using OWF with trapdoors

Let $f := (f_i)_{i\in I} : X \to X$ be a OWF with trapdoor information $(t_i)_{i\in I}$ as in Definition 5.2.24 (e.g., the RABIN function 5.2.25 or the RSA function 6.1.2). Set

- $M = S = X$,
- $K = I$, $K' = \{(i, t_i) \mid i \in I\}$, $\kappa : d := (i, t_i) \mapsto e := i$,
- $\mathcal{S}_d : M \to S,\ m \mapsto f_e^{-1}(m)$ (using the trapdoor information),
- $\mathcal{V}_e : M \times S \to \{0, 1\}, (m, s) \mapsto \begin{cases} 1 & \text{if} \quad f_e(s) = m \\ 0 & \text{if} \quad f_e(s) \neq m \end{cases}$.

REMARK 2.1.

(1) Existential forgery is always possible by first choosing the signature s then the message $m := f_e(s)$.
(2) If the OWF f is multiplicative (e.g., the RSA function: $(xy)^e = x^e y^e$) then the universal forgery under an adaptive chosen-message attack is possible: To sign m decompose it as $m = m_1 m_2$ with $m_1, m_2 \neq m$. Ask for the signatures of s_i of m_i (this is allowed since $m_i \neq m$). Compute $(m, s) = (m, s_1 s_2)$ by the multiplicativity of f.
(3) Another obvious drawback of this scheme is that the signature has the same length as the message.

We know give a variant of asymmetric signatures that, under certain assumptions, avoids the above mentioned drawbacks.

DEFINITION 2.2 (Hash-then-sign). This is a variant of Definition 1.1 with the following modifications (we are still using the notation of this section):

- $M = \{0, 1\}^\bullet$, $S = X$.
- $H : M \to X$ a map given by a polynomial algorithm, called the **hash function**.
- $\mathcal{S}_d : M \to S,\ m \mapsto f_e^{-1}(H(m))$.
- $\mathcal{V}_e : M \times S \to \{0, 1\},\ (m, s) \mapsto \begin{cases} 1 & \text{if} \quad f_e(s) = H(m) \\ 0 & \text{if} \quad f_e(s) \neq H(m) \end{cases}$.

To avoid the above attack scenarios the hash function H must be a one-way non-multiplicative function.

3. Hash functions

DEFINITION 3.1.

(1) A **hash function** is a function $H : \{0, 1\}^\bullet \to \{0, 1\}^\ell$ for some fixed $\ell \in \mathbb{N}$ given by a polynomial algorithm.

(2) H is called **collision resistant** if it is unfeasible to find distinct x_1, x_2 with $H(x_1) = H(x_2)$.

REMARK 3.2.

(1) A collision resistant hash function is a one-way function (since finding a preimage of $H(x_1)$ would lead to a collision).
(2) An "ideal" hash function behaves like a **random oracle (RO)**:
A random oracle would give to each $x \in \{0,1\}^\bullet$ a *random* answer $H(x) \in \{0,1\}^\ell$ *and* would store the answer internally as $H[x]$. If the oracle is given the same x again it will return the cached value $H[x]$.
(3) It is unknown if an "ideal" hash function exists.

EXAMPLE 3.3 (Hash functions from block ciphers). Let $(A, \ell, K, \mathcal{E})$ be a block cipher $(\mathcal{E} : B \times K \to B = A^\ell, (p, e) \mapsto \mathcal{E}_e(p))$. Let $A = \{0,1\}$ and $K = B$, in particular $\mathcal{E} : \{0,1\}^\ell \times \{0,1\}^\ell \to \{0,1\}^\ell$. Define $H(x_1, \ldots, x_r) \in \{0,1\}^\ell$ with $x_i \in B$ recursively by setting $H(\emptyset) = 0$ and

$$H(x_1, \ldots, x_r) = \mathcal{E}_h(x_r) + h, \text{ where } h = H(x_1, \ldots, x_{r-1}).$$

The widely used SHA1 : $\{0,1\}^\bullet \to \{0,1\}^{160}$ hash function is such an example. The details are too technical. For the SHA3 competition cf. [**NIS07**].

EXAMPLE 3.4. Let p be a prime number such that $q := \frac{p-1}{2}$ is also prime. Further let $b \in \mathbb{F}_p^* = \langle a \rangle$. Define the function

$$f : \{0, \ldots, q-1\} \times \{0, \ldots, q-1\} \to \mathbb{F}_p^*, \ (x, y) \mapsto a^x b^y.$$

As in the previous example, one can use f to construct a hash function. We claim that finding a collision of f implies computing the DL $\log_a b$.

PROOF. Let $a^x b^y = a^{x'} b^{y'}$ be a collision of f $((x, y) \neq (x', y'))$. If $y = y'$ then $a^x = a^{x'}$ and $x = x'$ ↯. So $y \neq y'$. Let $z := \log_a b$. $a^{x-x'} = b^{y'-y} \implies x - x' \equiv z(y' - y) \bmod p - 1$ can be solved to obtain z. □

4. Signatures using OWF without trapdoors

4.a. ELGAMAL signature scheme. Let $G = \langle g \rangle$ be a cyclic group of order N generated by g. Further let $f : G \to \{0,1\}^\bullet$ be a binary representation of the elements of G and $H : \{0,1\} \to \mathbb{Z}/N\mathbb{Z}$ a collision resistant hash function. The **ELGAMAL signature scheme** is defined by setting:

- $M = \{0,1\}^\bullet$.
- $S = G \times \mathbb{Z}/N\mathbb{Z}$.
- $K' = \{d = (g, a) \mid a \in \mathbb{Z}/N\mathbb{Z}\} \xrightarrow{\kappa} \{e = (g, y) \mid y \in G\} = K, \ (g, a) \mapsto (g, g^a)$.
- $\mathcal{S}_{(g,a)} : \{0,1\}^\bullet \rightsquigarrow G \times \mathbb{Z}/N\mathbb{Z}, \ m \mapsto \sigma$ with σ defined as follows:
  - Choose randomly an $k \in (\mathbb{Z}/N\mathbb{Z})^*$.
  - Set $r := g^k \in G$.
  - Set $s := k^{-1}(H(m) + aH(f(r)) \in \mathbb{Z}/N\mathbb{Z}$.

$-\ \sigma := (r, s)$.

- $\mathcal{V}_{(g,y)} : \{0,1\}^\bullet \times (G \times \mathbb{Z}/N\mathbb{Z}) \to \{0,1\},\ (m,(r,s)) \mapsto \begin{cases} 1 & \text{if } g^{H(m)}y^{H(f(r))} = r^s \\ 0 & \text{otherwise} \end{cases}$.

PROOF OF CORRECTNESS. $r^s = g^{H(m)}y^{H(f(r))} \iff ks = H(m) + aH(f(r)) \iff s = k^{-1}(H(m) + aH(f(r)))$. □

4.b. ECDSA. We end by describing the **elliptic curve** version of the **digital signature algorithms (ECDSA)**. Choose an an elliptic curve E over $\mathbb{F}_q$ with $E(\mathbb{F}_q) = \langle P \rangle$ of large prime order N (this assumption can be relaxed, see [**Was08**, §6.6]). Choose a *secret* random integer a and compute $Q = aP$ and publish $(E, \mathbb{F}_q, N, P, Q)$. To sign a message with hash value $m \in \mathbb{Z}/N\mathbb{Z}$:

- Choose a random integer $1 \leq k < N$ and compute $R = kP = (x, y)$.
- Compute $s = k^{-1}(m + ax) \bmod N$.
- The signature is (m, R, s).

To verify the signature do the following:

- Compute $u_1 = s^{-1}m \bmod N$ and $u_2 = s^{-1}x \bmod N$.
- Compute $V = u_1P + u_2Q$.
- The signature is valid if $V = R$.

PROOF OF CORRECTNESS. $V = u_1P + u_2Q = s^{-1}mP + s^{-1}xQ = s^{-1}(mP + xaP) = kP = R$. □

APPENDIX A

Some analysis

1. Real functions

1.a. Jensen's inequality.

LEMMA 1.1. JENSEN*'s inequality Let* $f : I \to \mathbb{R}$ *be a* ***strictly concave****, i.e.,* $\frac{f(x)+f(y)}{2} < f(\frac{x+y}{2})$ *for all* $x, y \in I$ *with* $x \neq y$*). Then for all* $a_i > 0$ *with* $\sum_i a_i = 1$ *and all* $x_i \in I$ $(i = 1, \ldots, n)$

$$\sum_i a_i f(x_i) \leq f(\sum_i a_i x_i).$$

Equality holds only if $x_1 = \ldots = x_n$. □

1.b. The normal distribution. Recall the **normal distribution** $N(\mu, \sigma)$ with expected value μ and variance σ is given by the GAUSSian density function

$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

$N(0, 1)$ is called the **standard** normal distribution.

If X is $N(0, 1)$ distributed and $x > 0$ then

$$\mu_X((-x, x)) = \operatorname{erf}(\frac{x}{\sqrt{2}}),$$

where

$$\operatorname{erf}(x) := \frac{2}{\sqrt{\pi}} \int_0^x e^{-s^2} ds$$

is the **GAUSSian error function**. The function $\operatorname{erfc} := 1 - \operatorname{erf}$ is called the **complementary GAUSSian error function**:

$$\mu_X(\mathbb{R} \setminus (-x, x)) = \operatorname{erfc}(\frac{x}{\sqrt{2}}).$$

Bibliography

[Bar10] Mohamed Barakat, *Cryptography — Lecture course held at the University of Kaiserslautern*, 2010, (http://www.mathematik.uni-kl.de/~barakat/Lehre/WS10/Cryptography/). 45

[Buc08] Johannes Buchmann, *Einführung in die kryptographie*, Springer-Verlag Berlin Heidelberg, 2008.

[GAP08] The GAP Group, *GAP – Groups, Algorithms, and Programming, Version 4.4.12*, 2008, (http://www.gap-system.org).

[Har77] Robin Hartshorne, *Algebraic geometry*, Springer-Verlag, New York, 1977, Graduate Texts in Mathematics, No. 52. MR MR0463157 (57 #3116) 86

[KAF+10] Thorsten Kleinjung, Kazumaro Aoki, Jens Franke, Arjen Lenstra, Emmanuel Thomé, Joppe Bos, Pierrick Gaudry, Alexander Kruppa, Peter Montgomery, Dag Arne Osvik, Herman te Riele, Andrey Timofeev, and Paul Zimmermann, *Factorization of a 768-bit rsa modulus*, Cryptology ePrint Archive, Report 2010/006, 2010, http://eprint.iacr.org/2010/006. 73

[KK10] Christian Karpfinger and Hubert Kiechle, *Kryptologie*, Vieweg + Teubner, Wiesbaden, 2010, Algebraische Methoden und Algorithmen. [Algebraic methods and algorithms]. MR 2650160

[Kob98] Neal Koblitz, *Algebraic aspects of cryptography*, Algorithms and Computation in Mathematics, vol. 3, Springer-Verlag, Berlin, 1998, With an appendix by Alfred J. Menezes, Yi-Hong Wu and Robert J. Zuccherato. MR 1610535 (2000a:94012) 90

[MvOV97] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone, *Handbook of applied cryptography*, CRC Press Series on Discrete Mathematics and its Applications, CRC Press, Boca Raton, FL, 1997, With a foreword by Ronald L. Rivest (http://www.cacr.math.uwaterloo.ca/hac/). MR 1412797 (99g:94015) 20

[NIS07] NIST, *Cryptographic hash algorithm competition of the national institute of standards and technology*, 2007, (http://csrc.nist.gov/groups/ST/hash/sha-3/index.html) [Online; accessed 06-February-2011]. 101

[Sti06] Douglas R. Stinson, *Cryptography*, third ed., Discrete Mathematics and its Applications (Boca Raton), Chapman & Hall/CRC, Boca Raton, FL, 2006, Theory and practice. MR 2182472 (2007f:94060)

[Tre05] Luca Trevisan, *Pseudorandomness and combinatorial constructions*, 2005, (http://www.cs.berkeley.edu/~luca/pacc/lecture08.pdf) [Online; accessed 12-December-2010]. 42

[Vau06] Serge Vaudenay, *A classical introduction to cryptography*, Springer, New York, 2006, Applications for communications security. MR 2171694 (2007b:94257)

[Was08] Lawrence C. Washington, *Elliptic curves*, second ed., Discrete Mathematics and its Applications (Boca Raton), Chapman & Hall/CRC, Boca Raton, FL, 2008, Number theory and cryptography. MR MR2404461 (2009b:11101) 88, 89, 90, 91, 92, 93, 96, 97, 102

[Wik10a] Wikipedia, *Advanced encryption standard — Wikipedia, the free encyclopedia*, 2010, (http://en.wikipedia.org/wiki/Advanced_Encryption_Standard) [Online; accessed 12-December-2010]. 45

[Wik10b] ———, *Alternating step generator — Wikipedia, the free encyclopedia*, 2010, (http://en.wikipedia.org/wiki/Alternating_step_generator) [Online; accessed 12-December-2010]. 40

[Wik10c] ———, *Berlekamp–massey algorithm — Wikipedia, the free encyclopedia*, 2010, (http://en.wikipedia.org/wiki/Berlekamp-Massey_algorithm) [Online; accessed 09-December-2010]. 39

[Wik10d] ———, *Block cipher modes of operation — Wikipedia, the free encyclopedia*, 2010, (http://en.wikipedia.org/wiki/Block_cipher_modes_of_operation) [Online; accessed 15-December-2010]. 45

[Wik10e] ———, *Blum blum shub — Wikipedia, the free encyclopedia*, 2010, (http://en.wikipedia.org/wiki/Blum_Blum_Shub) [Online; accessed 09-December-2010]. 53

[Wik10f] ———, *Linear feedback shift register — Wikipedia, the free encyclopedia*, 2010, (http://en.wikipedia.org/wiki/Linear_feedback_shift_register) [Online; accessed 24-November-2010]. 24

[Wik10g] ———, *List of random number generators — Wikipedia, the free encyclopedia*, 2010, (http://en.wikipedia.org/wiki/List_of_pseudorandom_number_generators) [Online; accessed 24-November-2010]. 24

[Wik10h] ———, *Rijndael key schedule — Wikipedia, the free encyclopedia*, 2010, (http://en.wikipedia.org/wiki/Rijndael_key_schedule) [Online; accessed 09-December-2010]. 45

[Wik10i] ———, *Run test — Wikipedia, the free encyclopedia*, 2010, (http://de.wikipedia.org/wiki/Run-Test) [Online; accessed 09-December-2010]. 38

[Wik10j] ———, Mersenne *twister — Wikipedia, the free encyclopedia*, 2010, (http://en.wikipedia.org/wiki/Mersenne_twister) [Online; accessed 24-November-2010]. 28

[Wik11a] ———, *Birthday problem — Wikipedia, the free encyclopedia*, 2011, (http://en.wikipedia.org/wiki/Birthday_problem) [Online; accessed 19-January-2011]. 74

[Wik11b] ———, *Cycle detection — Wikipedia, the free encyclopedia*, 2011, (http://en.wikipedia.org/wiki/Cycle_detection) [Online; accessed 19-January-2011]. 74

[Wik11c] ———, *Feistel cipher — Wikipedia, the free encyclopedia*, 2011, (http://en.wikipedia.org/wiki/Feistel_cipher) [Online; accessed 05-February-2011]. 20

[Wik11d] ———, *Optimal asymmetric encryption padding — Wikipedia, the free encyclopedia*, 2011, (http://en.wikipedia.org/wiki/Optimal_asymmetric_encryption_padding) [Online; accessed 13-January-2011]. 62

[Wik11e] ———, *Rsa numbers — Wikipedia, the free encyclopedia*, 2011, (http://en.wikipedia.org/wiki/RSA_numbers) [Online; accessed 19-January-2011]. 73

[Wik11f] ———, *Tonelli–shanks algorithm — Wikipedia, the free encyclopedia*, 2011, (http://en.wikipedia.org/wiki/Tonelli-Shanks_algorithm) [Online; accessed 05-February-2011]. 51

[GBM] Goldwasser, Bellare, and Mit, *Lecture notes on cryptography*, Script.

[Gol] Goldreich, *Pseusorandom generators*, Script.

[Gol05] Oded Goldreich, *Foundations of cryptography—a primer*, Found. Trends Theor. Comput. Sci. **1** (2005), no. 1, 1–116. MR 2379506 (2010d:94086)

[KK10] Christian Karpfinger and Hubert Kiechle, *Kryptologie*, Vieweg + Teubner, Wiesbaden, 2010, Algebraische Methoden und Algorithmen. [Algebraic methods and algorithms]. MR 2650160 (2011c:94001)

[Kob98] Neal Koblitz, *Algebraic aspects of cryptography*, Algorithms and Computation in Mathematics, vol. 3, Springer-Verlag, Berlin, 1998, With an appendix by Alfred J. Menezes, Yi-Hong Wu and Robert J. Zuccherato. MR 1610535 (2000a:94012)

[MvOV97] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone, *Handbook of applied cryptography*, CRC Press Series on Discrete Mathematics and its Applications, CRC Press, Boca Raton, FL, 1997, With a foreword by Ronald L. Rivest. MR 1412797 (99g:94015)

[BNS10] Albrecht Beutelspacher, Heike B. Neumann, and Thomas Schwarzpaul, *Kryptografie in Theorie und Praxis*, revised ed., Vieweg + Teubner, Wiesbaden, 2010, Mathematische Grundlagen für Internetsicherheit, Mobilfunk und elektronisches Geld. [Mathematical foundations for internet security, cellular phone networks and electronic money]. MR 2643186 (2011b:94027)

[Buc04] Johannes Buchmann, *Introduction to cryptography*, second ed., Undergraduate Texts in Mathematics, Springer-Verlag, New York, 2004. MR 2075209 (2005f:94084)

[DK07] Hans Delfs and Helmut Knebl, *Introduction to cryptography*, second ed., Information Security and Cryptography, Springer, Berlin, 2007, Principles and applications. MR 2345767 (2008e:94026)

[Gol05] Oded Goldreich, *Foundations of cryptography—a primer*, Found. Trends Theor. Comput. Sci. **1** (2005), no. 1, 1–116. MR 2379506 (2010d:94086)

[Gol10] ———, *A primer on pseudorandom generators*, University Lecture Series, vol. 55, American Mathematical Society, Providence, RI, 2010. MR 2677397 (2011h:68001)

[HT77] Robert V. Hogg and Elliot A. Tanis, *Probability and statistical inference*, Macmillan Publishing Co., New York, 1977. MR 0426223 (54 #14169)

[Kob87] Neal Koblitz, *Elliptic curve cryptosystems*, Math. Comp. **48** (1987), no. 177, 203–209. MR 866109 (88b:94017)

[Kob94] ———, *A course in number theory and cryptography*, second ed., Graduate Texts in Mathematics, vol. 114, Springer-Verlag, New York, 1994. MR 1302169 (95h:94023)

[LN97] Rudolf Lidl and Harald Niederreiter, *Finite fields*, second ed., Encyclopedia of Mathematics and its Applications, vol. 20, Cambridge University Press, Cambridge, 1997, With a foreword by P. M. Cohn. MR 1429394 (97i:11115)

[1]

[1]Wikipedia is a wonderful source of open and easily accessible knowledge. Nevertheless please treat it with care and consult other scientific sources. I cite wikipedia for background material and implementations of widely used algorithms. I do not cite it for theorems or proofs. If you spot a mistake on a page please contribute by correcting it.