Encryption

Leiria, September 5, 2006

Joachim Rosenthal

Rosenthal@math.unizh.ch

Department of Mathematics

University of Zürich

Outline of Talk:

1. Road Map to Cryptology and Historical Remarks

Outline of Talk:

1. Road Map to Cryptology and Historical Remarks
2. The Data Encryption Standard DES

Outline of Talk:

1. Road Map to Cryptology and Historical Remarks
2. The Data Encryption Standard DES
3. The Advanced Encryption Standard Rijndael

Outline of Talk:

1. Road Map to Cryptology and Historical Remarks
2. The Data Encryption Standard DES
3. The Advanced Encryption Standard Rijndael
4. Public Key Cryptography

Outline of Talk:

1. Road Map to Cryptology and Historical Remarks
2. The Data Encryption Standard DES
3. The Advanced Encryption Standard Rijndael
4. Public Key Cryptography
5. The RSA Public Key System

Outline of Talk:

1. Road Map to Cryptology and Historical Remarks
2. The Data Encryption Standard DES
3. The Advanced Encryption Standard Rijndael
4. Public Key Cryptography
5. The RSA Public Key System
6. The Discrete Logarithm Problem

Outline of Talk:

1. Road Map to Cryptology and Historical Remarks
2. The Data Encryption Standard DES
3. The Advanced Encryption Standard Rijndael
4. Public Key Cryptography
5. The RSA Public Key System
6. The Discrete Logarithm Problem
7. Systems Based on Group Actions

1. Road Map to Cryptology

Cryptology is the study of:

- *Cryptography*, the design of secret ciphers.

1. Road Map to Cryptology

Cryptology is the study of:

- *Cryptography*, the design of secret ciphers.
- *Cryptoanalysis*, the analysis of secret ciphers.

Cryptography

Cryptography is the study of mathematical techniques to aspects of

(i) Confidentiality during point to point communication.

Cryptography

Cryptography is the study of mathematical techniques to aspects of

(i) Confidentiality during point to point communication.

(ii) Data integrity (it can be verified that the data is the same as the original);

Cryptography

Cryptography is the study of mathematical techniques to aspects of

(i) Confidentiality during point to point communication.

(ii) Data integrity (it can be verified that the data is the same as the original);

(iii) Authentication (e.g. digital signature);

Cryptography

Cryptography is the study of mathematical techniques to aspects of

(i) Confidentiality during point to point communication.

(ii) Data integrity (it can be verified that the data is the same as the original);

(iii) Authentication (e.g. digital signature);

(iv) Access control (e.g. passwords, PIN numbers).

Historical remarks: Caesar ciphers

Caesar used to communicate with his generals using ‘cyclic substitution ciphers’:
For this identify the alphabet with set $\mathbb{Z}_{26}$:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26

A Caesar cipher is then of the form:

$$\begin{array}{rcl} \mathbb{Z}_{26} & \longrightarrow & \mathbb{Z}_{26} \\ x & \longmapsto & x+k \end{array}$$

where k is a secretly agreed number: $1 \leq k \leq 25$.

Example Caesar

Caesar sent a messenger to one of his generals in Gaul. The message was encrypted. It reads:
YWLPQNA WOPANET WJZ KXAHET DEY AP JQJY FQHEQO YWAOWN
A simple ‘brute force attack’ checking all 25 possibilities will reveal the text:

Example Caesar

Map	Resulting Plaintext
$x \mapsto x+1$	ZXMQROB ...

Example Caesar

Map	Resulting Plaintext
$x \mapsto x+1$	ZXMQROB ...
$x \mapsto x+2$	AYNRSPC ..

Example Caesar

Map	Resulting Plaintext
$x \mapsto x+1$	ZXMQROB ...
$x \mapsto x+2$	AYNRSPC ...
$x \mapsto x+3$	BZOSTQD ...

Example Caesar

Map	Resulting Plaintext
$x \mapsto x+1$	ZXMQROB ...
$x \mapsto x+2$	AYNRSPC ...
$x \mapsto x+3$	BZOSTQD ...
$x \mapsto x+4$	CAPTURE ASTERIX AND OBELIX ...

Example Caesar

Map	Resulting Plaintext
$x \mapsto x+1$	ZXMQROB ...
$x \mapsto x+2$	AYNRSPC ...
$x \mapsto x+3$	BZOSTQD ...
$x \mapsto x+4$	CAPTURE ASTERIX AND OBELIX ...

Alternatively it is possible to do a frequency analysis. In English, the most frequently used letters are, in decreasing order of frequency,

E, T, A, O, I, N, S, R, H, L, D, ...

Vigenère ciphers

Around the year 1600, Vigenère introduced the vector version of the substitution cipher. This involved a mapping of

$$(\mathbb{Z}_{26})^n \longrightarrow (\mathbb{Z}_{26})^n$$

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \longmapsto \begin{pmatrix} x_1 + k_1 \\ x_2 + k_2 \\ \vdots \\ x_n + k_n \end{pmatrix}$$

Vigenère ciphers can easily be broken with some frequency analysis.

Hill ciphers

In 1931, D. Hill introduced Hill Ciphers. A Hill Cipher utilizes an $n \times n$ matrix which is invertible over $\mathbb{Z}_{26}$. Similar to the vector version, the Hill Cipher is a map

$$(\mathbb{Z}_{26})^n \longrightarrow (\mathbb{Z}_{26})^n$$

defined by

$$x \longmapsto Ax + k = y.$$

So, in the Hill Cipher, the recipient receives y, and if they are told A^{-1} and k they can compute x.

Hill ciphers are weak because of plaintext attacks.

Example Hill cipher

Alice and Bob use a Hill cipher to exchange messages. Their cipher is of the form:

$$\begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} \longmapsto \begin{bmatrix} 4 & 18 & 21 \\ 24 & 3 & 7 \\ 11 & 0 & 3 \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} + \begin{bmatrix} 11 \\ 0 \\ 20 \end{bmatrix}.$$

Help to decipher Bob the following message he received from Alice:

$$\begin{bmatrix} 22 \\ 12 \\ 10 \end{bmatrix}, \begin{bmatrix} 16 \\ 12 \\ 20 \end{bmatrix}, \begin{bmatrix} 19 \\ 0 \\ 21 \end{bmatrix}, \begin{bmatrix} 11 \\ 9 \\ 19 \end{bmatrix}.$$

Example Hill cipher

To invert the Hill cipher, let $[x_1 x_2 x_3]$ denote the cipher-text. We then compute:

$$\begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} = A^{-1} \begin{bmatrix} x_1 - 11 \\ x_2 - 0 \\ x_3 - 20 \end{bmatrix}$$

Doing this, we obtain: “SEE” “YOU” “ATN” “OON’.

Kerckhoff's Principle

In 1883, Flemish linguist Auguste Kerckhoff published a groundbreaking article that is still widely cited because of the stated principle:

> The security of a cryptosystem must not depend on keeping secret the crypto algorithm. Instead the security should depend only on keeping the key secret.

No security by obscurity.

In the article Kerckhoff formulated six principles:

Kerckhoff's Principle

- The system must be practically, if not mathematically, indecipherable;

Kerckhoff's Principle

- The system must be practically, if not mathematically, indecipherable;
- It must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience;

Kerckhoff's Principle

- The system must be practically, if not mathematically, indecipherable;
- It must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience;
- Its key must be communicable and retainable without the help of written notes, and changeable or modifiable at the will of the correspondents;

Kerckhoff's Principle

- The system must be practically, if not mathematically, indecipherable;
- It must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience;
- Its key must be communicable and retainable without the help of written notes, and changeable or modifiable at the will of the correspondents;
- It must be applicable to telegraphic correspondence;

Kerckhoff's Principle

- The system must be practically, if not mathematically, indecipherable;
- It must not be required to be secret, and it must be able to fall into the hands of the enemy without inconvenience;
- Its key must be communicable and retainable without the help of written notes, and changeable or modifiable at the will of the correspondents;
- It must be applicable to telegraphic correspondence;
- It must be portable, and its usage and function must not require the concourse of several people;

Kerckhoff's Principle

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

JOURNAL
DES
SCIENCES MILITAIRES.

Janvier 1883.

LA CRYPTOGRAPHIE MILITAIRE.

« La cryptographie est un auxiliaire puissant de la tactique militaire. »
(Général LEWAL, *Études de guerre.*)

I.

LA CRYPTOGRAPHIE DANS L'ARMÉE

A. Notions historiques.

La *Cryptographie* ou l'*Art de chiffrer* est une science vieille comme le monde ; confondue à son origine avec la télégraphie militaire, elle a été cultivée, dès la plus haute antiquité, par les Chinois, les Perses, les Carthaginois ; elle a été enseignée dans les écoles tactiques de la Grèce, et tenue en haute estime par les plus illustres généraux romains [1].

Depuis la modeste scytale des Lacédémoniens et les *trucs* inventés ou rapportés par Æneas-le-Tacticien [2], jusqu'au fameux

[1] C'est sous la rubrique : *Stéganographie, chiffre* ou *écritures secrètes*, que certains dictionnaires encyclopédiques donnent les renseignements qui se rapportent à la cryptographie. Les anciens auteurs l'appellent plus ou moins correctement : *ars notarum, ars zipherarum, polygraphia, scotographia, cryptologia, steganologia, cryptonenytices*, etc. ; les Allemands disent aujourd'hui : *Geheimschrift* ou *Chiffreschrift* et les Anglais : *cryptography*.

[2] Lettres mises entre les semelles du messager, communications cachées dans un ulcère du porteur ou dans les pendants d'oreilles des femmes, dés percés de

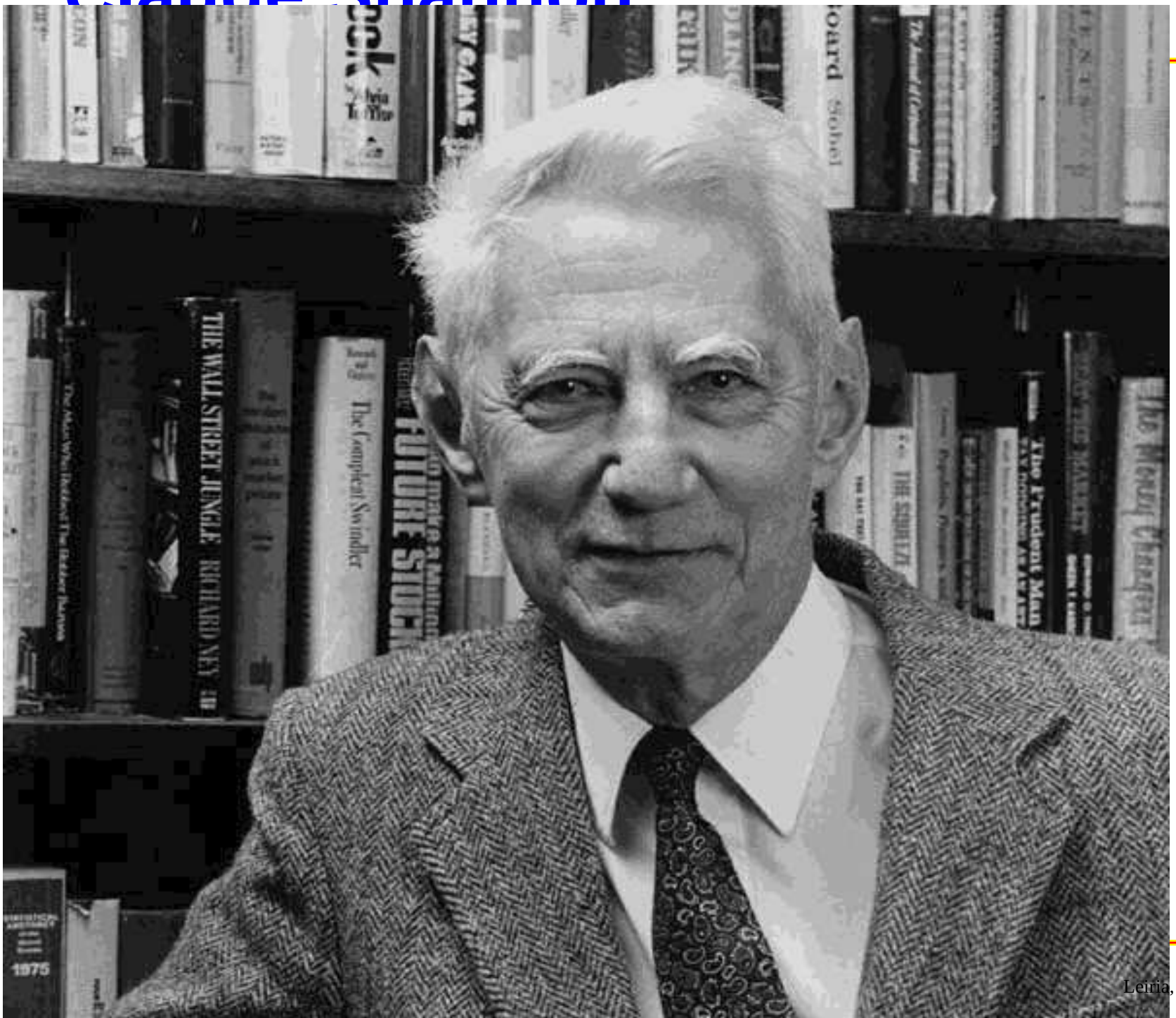
Provable security

In 1949 Claude Shannon [Sha49] published a fundamental result:

> *There exist unconditionally and provable secure cryptographic protocols.*

Practically this meant that there exist cryptographic protocols which cannot be broken even if somebody has unlimited computing power.

Claude Shannon



Perfect secrecy and Main Theorem

Definition 1 A crypto system has *perfect secrecy* or is *unconditionally secure* if

$$Pr(m \mid c) = Pr(m).$$

Theorem 2 *[Sha49] Let $|M| = |K|$ and assume $Pr(m) > 0$ for all $m \in M$. A secret key crypto system has perfect secrecy if and only if the random variable K is uniformly distributed and if for each message m and each cipher c there is a unique key k such that*

$$\varphi(m,k) = c.$$

Perfect secrect ciphers

A Consequence of Shannon's result is that in perfect secret system the uncertainty (=entropy) of the secret key has to be larger than the uncertainty of a message.

The ‘Vernam-One Time Pad’

Encryption:

Binary Text:	1	0	0	1	0	1	0	0	1	0
Secret Key:	0	0	1	0	1	1	1	1	0	0
Message:	1	0	1	1	1	0	1	1	1	0

The ‘Vernam-One Time Pad’

Encryption:

Binary Text:	1	0	0	1	0	1	0	0	1	0
Secret Key:	0	0	1	0	1	1	1	1	0	0
Message:	1	0	1	1	1	0	1	1	1	0

Decryption:

Message:	1	0	1	1	1	0	1	1	1	0
Secret Key:	0	0	1	0	1	1	1	1	0	0
Binary Text:	1	0	0	1	0	1	0	0	1	0

The ‘Vernam-One Time Pad’

Encryption:

Binary Text:	1	0	0	1	0	1	0	0	1	0
Secret Key:	0	0	1	0	1	1	1	1	0	0
Message:	1	0	1	1	1	0	1	1	1	0

Decryption:

Message:	1	0	1	1	1	0	1	1	1	0
Secret Key:	0	0	1	0	1	1	1	1	0	0
Binary Text:	1	0	0	1	0	1	0	0	1	0

Problem: The secret key is required to be as long as the message.

Recursive keys

One way to keep the secret key ‘small’ is through some (nonlinear) recurrence relation:

$$s_{n+d} = f(s_{n+d-1}, \ldots, s_n),\ n = 1, 2, \ldots$$

having initial conditions $s_1 = a_1, \ldots, s_d = a_d$.

Recursive keys

One way to keep the secret key 'small' is through some (nonlinear) recurrence relation:

$$s_{n+d} = f(s_{n+d-1}, \ldots, s_n),\ n = 1, 2, \ldots$$

having initial conditions $s_1 = a_1, \ldots, s_d = a_d$.

Example 3 Fibonacci sequence $s_{n+2} = s_{n+1} + s_n$ with initial condition $s_1 = 1,\ s_2 = 1$:

$$\text{over } \mathbb{F}_3\text{: } 1, 1, 2, 0, 2, 2, 1, 0, 1, 1$$

$$\text{over } \mathbb{F}_5\text{: } 1, 1, 2, 3, 0, 3, 3, 1, 4, 0, 4, 4, 3, 2, 0, 2, 2, 4, 1, 0, 1, 1$$

Linear Feedback Shift Register

If the recurrence relation is linear, i.e. if it has the form:

$$s_{n+d} = c_{d-1}s_{n+d-1} + \cdots + c_0 s_n,\ n = 1, 2, \ldots$$

then it is possible to implement the recurrence relation with a 'linear feedback shift register' (Almost Enigma).

Because of the Berlekamp-Massey algorithm [Mas69] linear feedback shift register turned out to be insecure.

Enigma

Enigma

Stream ciphers

Nonlinear recursive secret key systems, so called *stream ciphers* are still in use. Mathematically the key is generated through a (nonlinear) recurrence relation:

$$s_{n+d} = f(s_{n+d-1}, \ldots, s_n),\ n = 1, 2, \ldots$$

The most famous stream cipher is RC4 designed by Ron Rivest for RSA security.

2. The Data Encryption Standard DES

In the sequel let X, Y be arbitrary sets.

Definition 4 A one-way function is a map $\varphi : X \longrightarrow Y$ having the property that for all $x \in X$, $f(x)$ can be efficiently computed. In the same time it is practically not possible to compute $x \in \varphi^{-1}(y)$ for almost all $y \in Y$.

One-way functions are used e.g.:

- For password storage
- As 'hash functions'

One way functions with secret key

M: Message space.
K: Key space.
C: Cipher space.

Definition 5 A One way functions with secret key is a map
$\varphi : M \times K \longrightarrow C$
together with a map $\psi : C \times K \longrightarrow M$ such that:

1. $\psi(\varphi(m,k),k) = m$ for all $(m,k) \in M \times K$.
2. The induced maps $\varphi_m : K \longrightarrow C,\ k \longmapsto \varphi(m,k)$
   $\varphi_k : M \longrightarrow C,\ m \longmapsto \varphi(m,k)$
   are one way functions.

Data Encryption Standard DES

- 1973: National Institute of Standards asks for the construction of a one way function with secret key

Data Encryption Standard DES

- 1973: National Institute of Standards asks for the construction of a one way function with secret key
- 1975: IBM proposes ‘Lucipher DES’ which has a key size of 2^{128}.

Data Encryption Standard DES

- 1973: National Institute of Standards asks for the construction of a one way function with secret key
- 1975: IBM proposes ‘Lucipher DES’ which has a key size of 2^{128}.
- 1977: DES becomes the standard with a key size of 2^{56}.

DES

DES works with:

$$|M| = 2^{64}, \qquad |K| = 2^{56}, \qquad |C| = 2^{64}.$$

For a fixed $m \in M$ the one-way function $K \longrightarrow C, k \longmapsto \varphi(m,k)$ is used in the Unix system for password storage.

Nota Bene: The key size of K is less than 10^{17}, much too small for the current computers.

3. The Advanced Encryption Standard

In the mid 90th the National Institute of Standards asks for an 'Advanced Encryption Standard' (AES):

http://www.nist.gov/aes

August 9, 1999 - NIST Announces the AES Finalist Candidates for Round 2:

MARS, RC6, Rijndael, Serpent and Twofish

November 26, 2001 - NIST announces that **Rijndael** has been selected as the proposed AES

November 26, 2001

Announcing the

ADVANCED ENCRYPTION STANDARD (AES)

Federal Information Processing Standards Publications (FIPS PUBS) are issued by the National Institute of Standards and Technology (NIST) after approval by the Secretary of Commerce pursuant to Section 5131 of the Information Technology Management Reform Act of 1996 (Public Law 104-106) and the Computer Security Act of 1987 (Public Law 100-235).

1. Name of Standard. Advanced Encryption Standard (AES) (FIPS PUB 197).

2. Category of Standard. Computer Security Standard, Cryptography.

3. Explanation. The Advanced Encryption Standard (AES) specifies a FIPS-approved cryptographic algorithm that can be used to protect electronic data. The AES algorithm is a symmetric block cipher that can encrypt (encipher) and decrypt (decipher) information. Encryption converts data to an unintelligible form called ciphertext; decrypting the ciphertext converts the data back into its original form, called plaintext.

The AES algorithm is capable of using cryptographic keys of 128, 192, and 256 bits to encrypt and decrypt data in blocks of 128 bits.

4. Approving Authority. Secretary of Commerce.

5. Maintenance Agency. Department of Commerce, National Institute of Standards and Technology, Information Technology Laboratory (ITL).

6. Applicability. This standard may be used by Federal departments and agencies when an agency determines that sensitive (unclassified) information (as defined in P. L. 100-235) requires cryptographic protection.

Other FIPS-approved cryptographic algorithms may be used in addition to, or in lieu of, this standard. Federal agencies or departments that use cryptographic devices for protecting classified information can use those devices for protecting sensitive (unclassified) information in lieu of this standard.

In addition, this standard may be adopted and used by non-Federal Government organizations. Such use is encouraged when it provides the desired security for commercial and private organizations.

The Rijndael system

Compare with [Ros03]. Consider the irreducible polynomial

$$\mu(z) := z^8 + z^4 + z^3 + z + 1 \in \mathbb{Z}_2[z].$$

Let $\mathbb{F} := \mathbb{Z}_2[z]/ < \mu(z) >= \mathrm{GF}(256)$ be the Galois field of 2^8 elements and consider the ideal:

$$I :=< x^4 + 1, y^4 + 1, \mu(z) > \subset \mathbb{Z}_2[x,y,z].$$

We will describe the Rijndael algorithm through a sequence of polynomial manipulations inside the finite ring

$$R := \mathbb{Z}_2[x,y,z]/I = \mathbb{F}[x,y]/ < x^4 + 1, y^4 + 1 > . \qquad (1)$$

The Rijndael system

The monomials

$$\left\{x^i y^j z^k \mid 0 \leq i, j \leq 3,\ 0 \leq k \leq 7\right\}$$

form a $\mathbb{Z}_2$-basis of the ring (algebra) R. In particular $\dim_{\mathbb{Z}_2} R = 128$, i.e. $|R| = 2^{128}$. Whenever $r \in R$ is an element we will define elements $r_{i,j} \in \mathbb{F}$ and $r_i \in \mathbb{F}[y]/ < y^4 + 1 >$ through:

$$r = \sum_{i=0}^{3} \sum_{j=0}^{3} r_{i,j} x^i y^j = \sum_{i=0}^{3} r_i x^i. \tag{2}$$

For the Rijndael algorithm we define

$$K = M = C = R.$$

Rijndael system

Crucial for the description will be a fixed permutation polynomial:

$$\begin{aligned}\theta(u) := {} & \left(z^2+1\right)u^{254}+\left(z^3+1\right)u^{253}+\left(z^7+z^6+z^5+z^4+z^3+1\right)u^{251} \\ & +\left(z^5+z^2+1\right)u^{247}+\left(z^7+z^6+z^5+z^4+z^2\right)u^{239}+u^{223} \\ & +\left(z^7+z^5+z^4+z^2+1\right)u^{191}+\left(z^7+z^3+z^2+z+1\right)u^{127} \\ & +(z^6+z^5+z+1) \in \mathbb{F}[u]. \qquad (3)\end{aligned}$$

Assume Alice and Bob share a common secret key $k \in R$ and Alice wants to encrypt the message $m \in R$. In a first step both Alice and Bob do a *key expansion* which will result in 11 elements $k^{(t)} \in R\, t = 0, \ldots, 10$.

Rijndael key expansion:

Key expansion: Using the notation introduced in Equation (2), both Alice and Bob compute recursively 10 elements $k^{(t)} \in R,\ t = 0, \ldots, 9$ in the following way:

$$
\begin{aligned}
k^{(0)} &:= k \\
k_0^{(t+1)} &:= \left(\sum_{j=0}^{3} \theta(k_{3,j}^{(t)}) y^j \right) y^3 + z^t + k_0^{(t)} \text{ for } t = 0, \ldots, 9. \\
k_i^{(t+1)} &:= k_{i-1}^{(t+1)} + k_i^{(t)} \text{ for } t = 0, \ldots, 9,\ i = 1, 2, 3.
\end{aligned}
$$

In order to describe the actual encryption algorithm we define the polynomial:

$$\gamma := (z+1)y^3 + y^2 + y + z \in R.$$

Rijndael encryption algorithm:

Using the round keys $k^{(t)} \in R$ and starting with the message $m \in R$ Alice computes recursively:

$$
\begin{aligned}
m^{(0)} &:= m + k^{(0)} \\
m^{(t+1)} &:= \gamma \sum_{i=0}^{3} \sum_{j=0}^{3} \theta(m_{i,j}^{(t)}) x^{i+j} y^{j} + k^{(t+1)}, \; t = 0, \ldots, 8. \\
m^{(10)} &:= \sum_{i=0}^{3} \sum_{j=0}^{3} \theta(m_{i,j}^{(9)}) x^{i+j} y^{j} + k^{(10)}
\end{aligned}
$$

The cipher to be transmitted by Alice is $m^{(10)}$.

4. Public Key Cryptography

Fundamental Question:

How can a secure communication between two parties, say Alice and Bob, be established without having exchanged secretly a method of encryption?!

In 1976 W. Diffie, M. E. Hellmann and R. C. Merkle provided a mathematical formulation to this problem.

Illustration

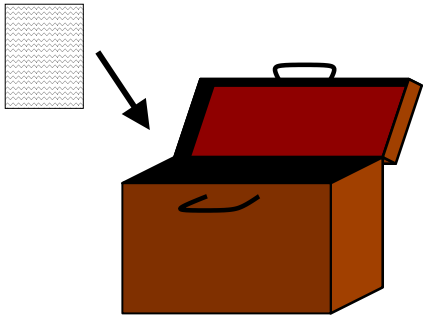
Alice

Bob

Illustration

Alice

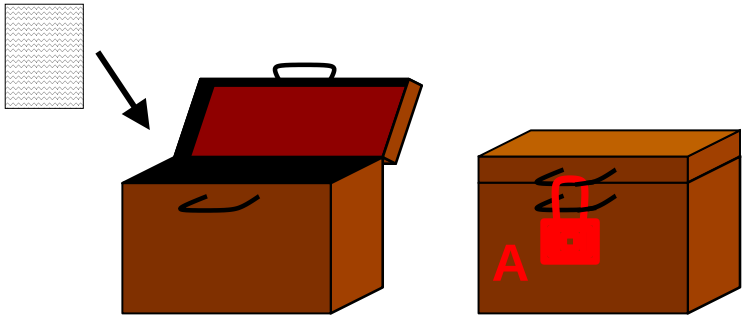
Bob



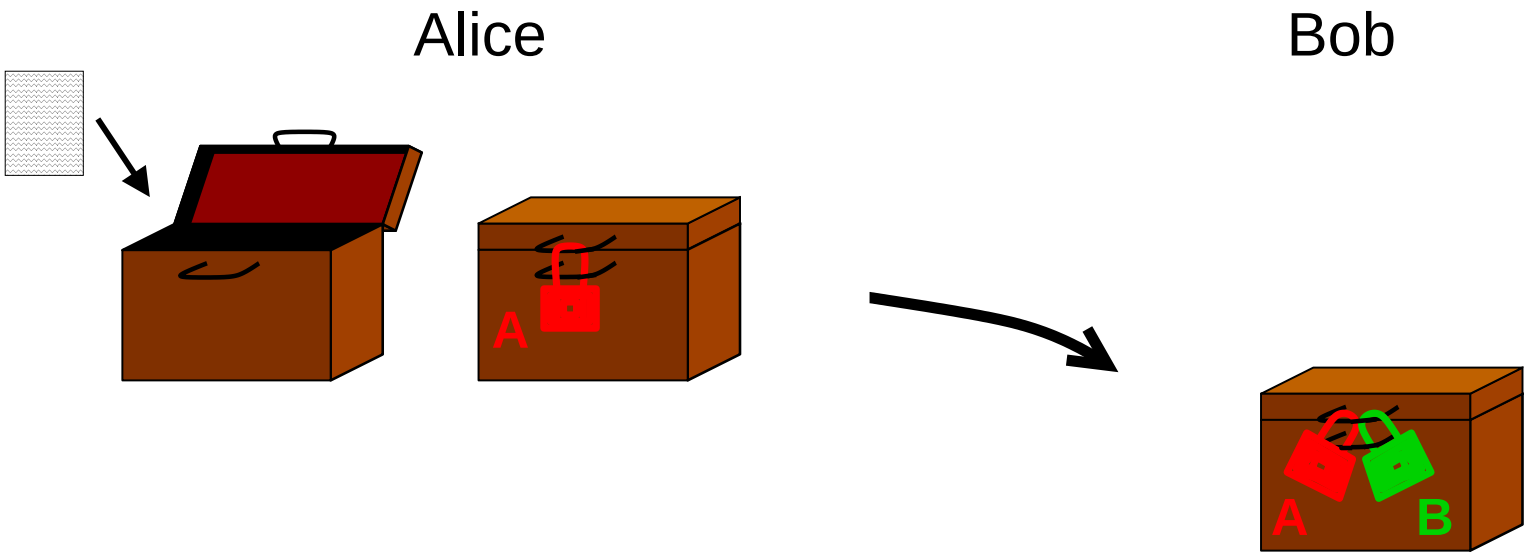
Illustration

Alice

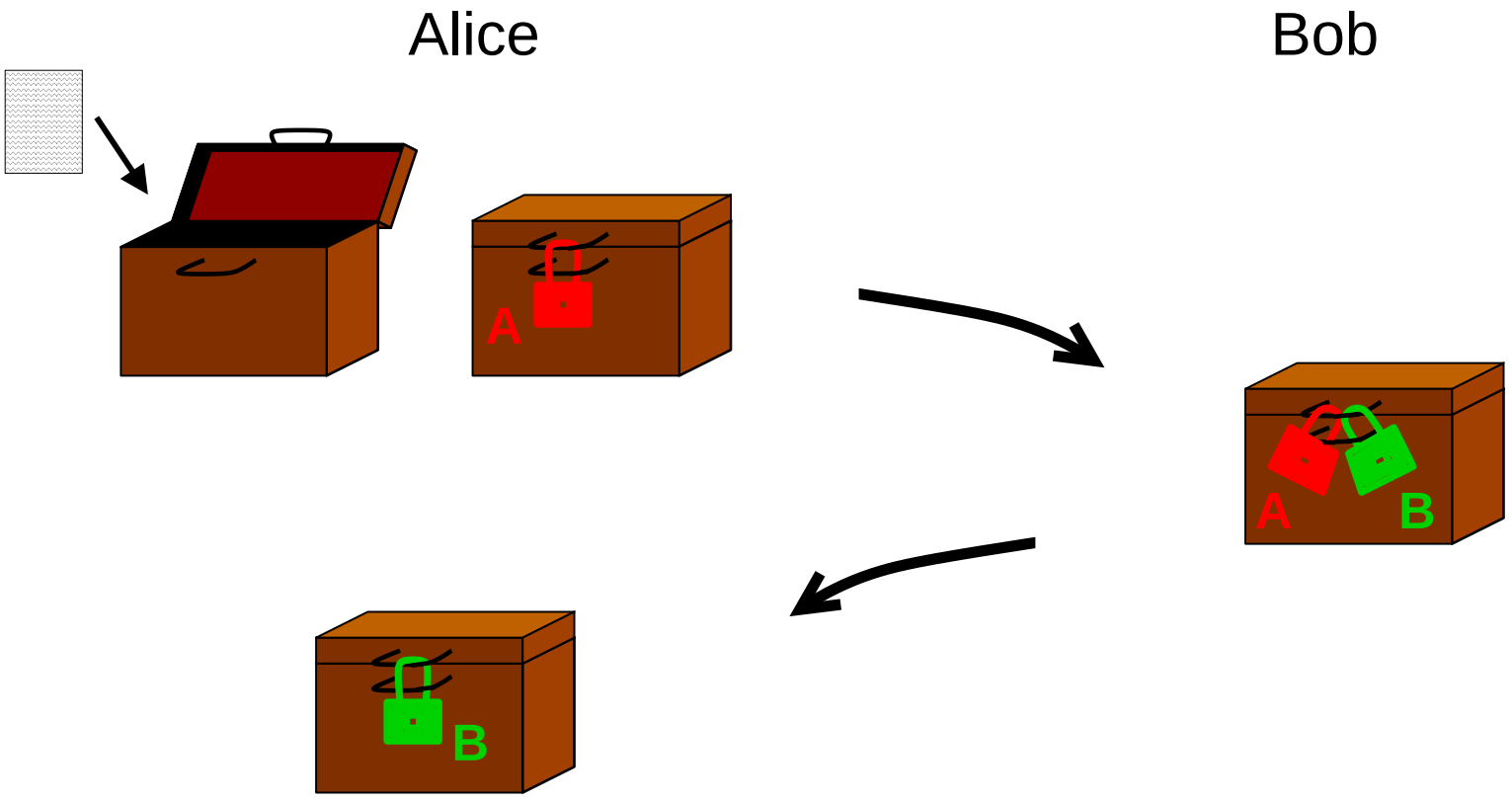
Bob



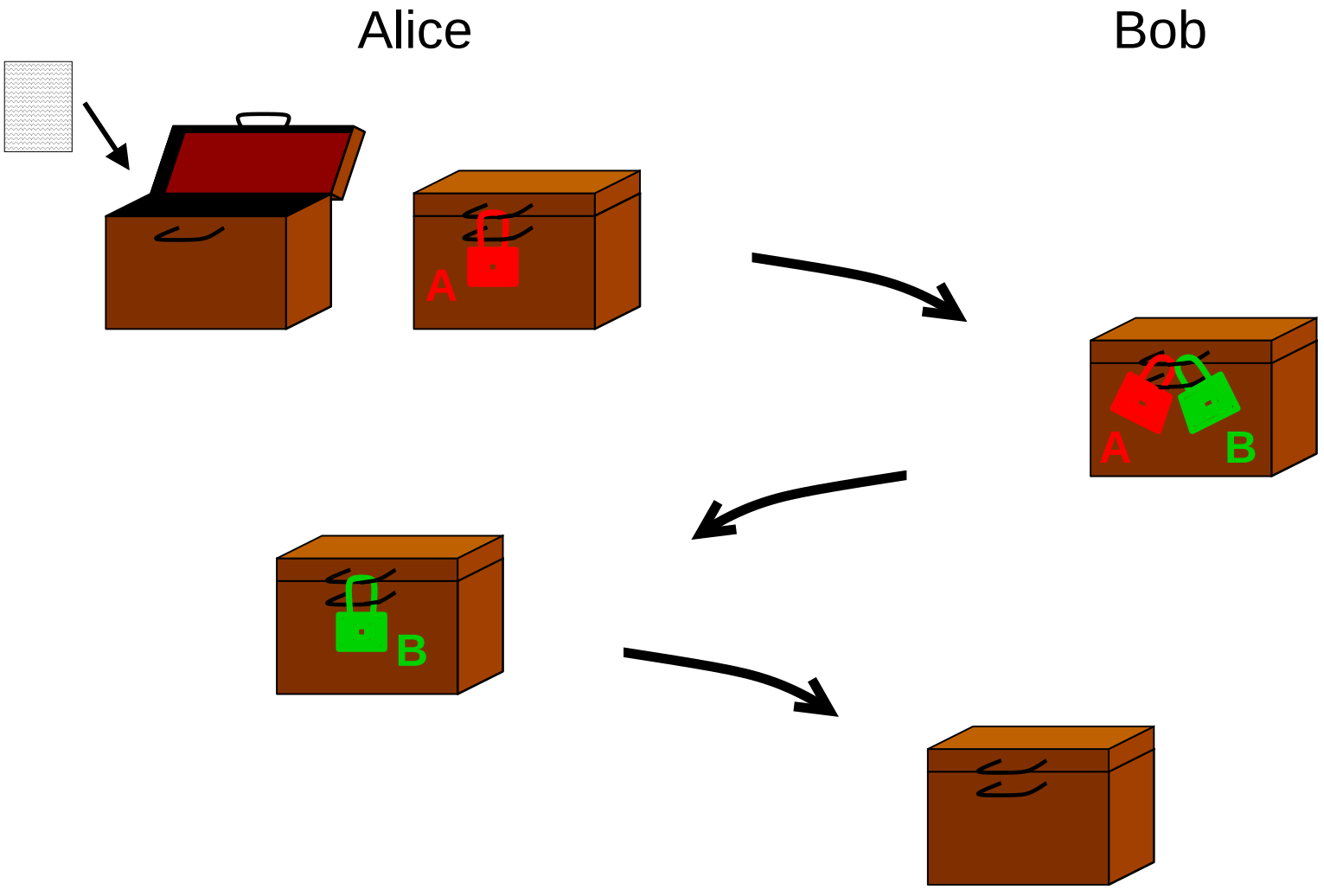
Illustration



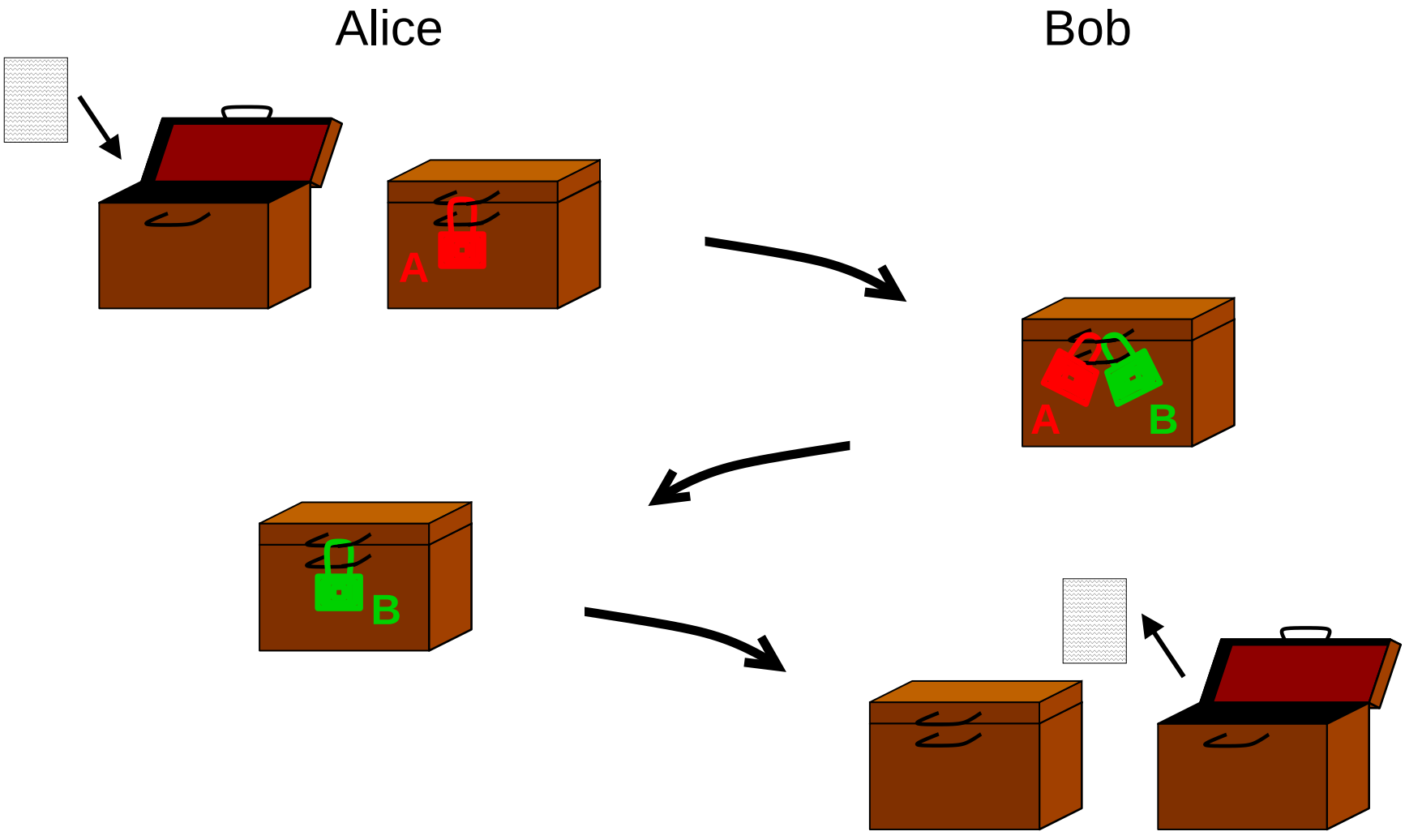
Illustration



Illustration



Illustration



Massey-Omura protocol

Alice and Bob use an elliptic curve $E(\mathbb{F}_q)$ of prime order for communication. This is 'the letter box'.
Alice wants to send the message $P \in E(\mathbb{F}_q)$ to Bob.

1. Alice sends to Bob aP, the integer $a \in \mathbb{Z}$ is the private key of Alice.

Massey-Omura protocol

Alice and Bob use an elliptic curve $E(\mathbb{F}_q)$ of prime order for communication. This is 'the letter box'.
Alice wants to send the message $P \in E(\mathbb{F}_q)$ to Bob.

1. Alice sends to Bob aP, the integer $a \in \mathbb{Z}$ is the private key of Alice.
2. Bob sends to Alice baP, the integer $b \in \mathbb{Z}$ is the private key of Bob.

Massey-Omura protocol

Alice and Bob use an elliptic curve $E(\mathbb{F}_q)$ of prime order for communication. This is 'the letter box'.
Alice wants to send the message $P \in E(\mathbb{F}_q)$ to Bob.

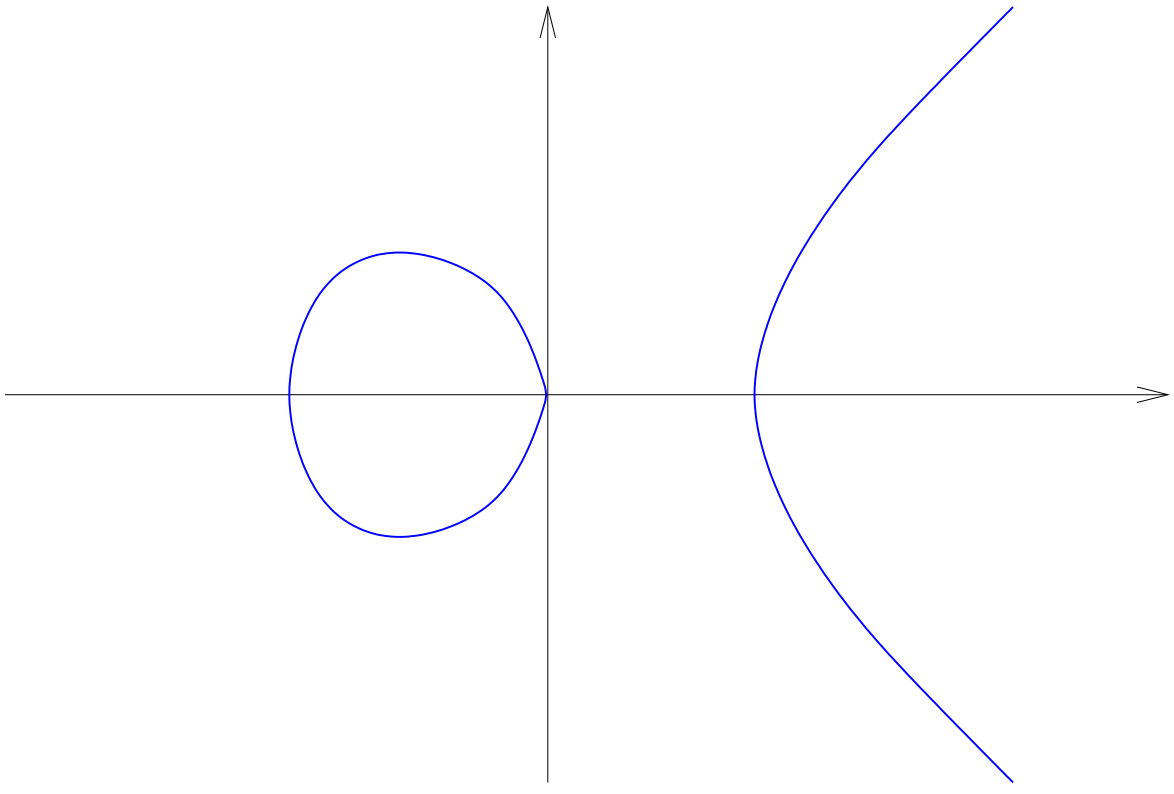
1. Alice sends to Bob aP, the integer $a \in \mathbb{Z}$ is the private key of Alice.
2. Bob sends to Alice baP, the integer $b \in \mathbb{Z}$ is the private key of Bob.
3. Alice sends to Bob $a^{-1}baP = bP$.

Massey-Omura protocol

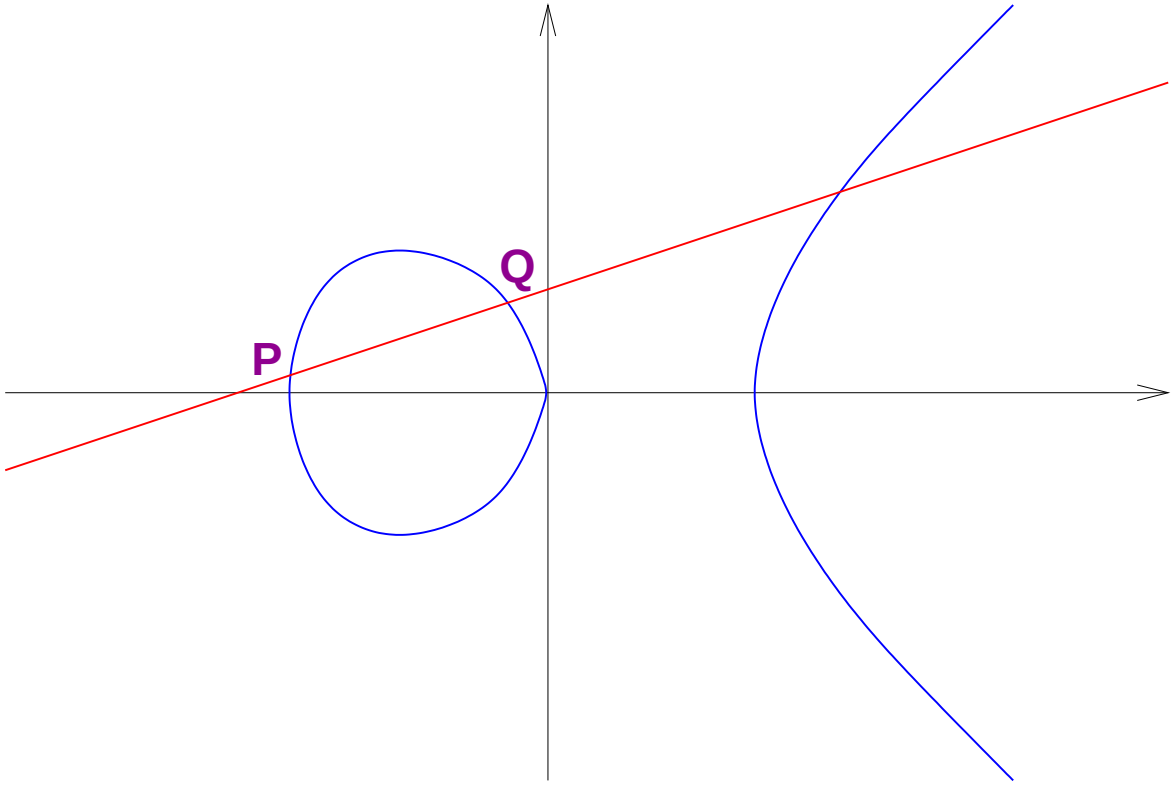
Alice and Bob use an elliptic curve $E(\mathbb{F}_q)$ of prime order for communication. This is 'the letter box'.
Alice wants to send the message $P \in E(\mathbb{F}_q)$ to Bob.

1. Alice sends to Bob aP, the integer $a \in \mathbb{Z}$ is the private key of Alice.
2. Bob sends to Alice baP, the integer $b \in \mathbb{Z}$ is the private key of Bob.
3. Alice sends to Bob $a^{-1}baP = bP$.
4. Bob computes $b^{-1}bP = P$ and reads the message.

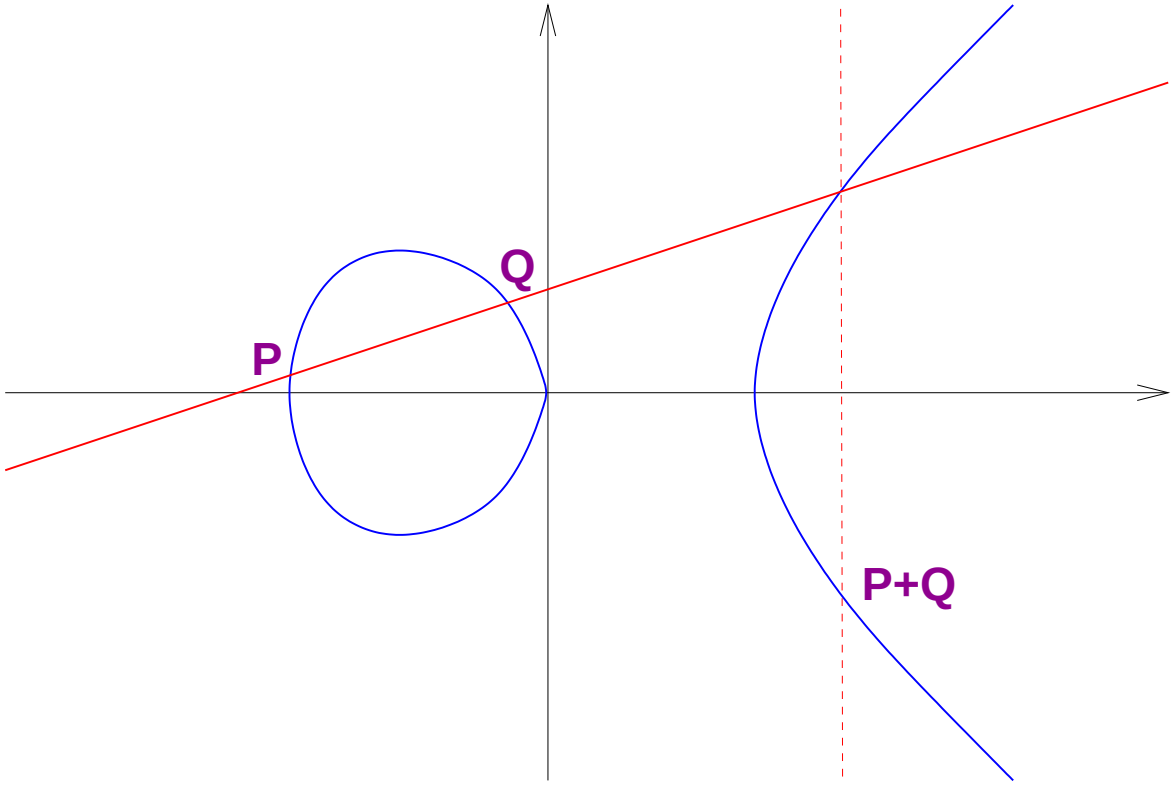
The elliptic curve group $E(\mathbb{F}_p)$



The elliptic curve group $E(\mathbb{F}_p)$



The elliptic curve group $E(\mathbb{F}_p)$



One way trapdoor functions

Definition 6 A one way trapdoor function is a one-way function $\varphi : X \longrightarrow Y$, which has the property:

1. φ is injective
2. With the help of a 'private key' it is possible to compute:

$$\varphi^{-1} : \varphi(X) \longrightarrow X.$$

Principle of public key cryptography

- Alice constructs a one-way trapdoor function $\varphi : X \longrightarrow Y$ and publishes it.

Principle of public key cryptography

- Alice constructs a one-way trapdoor function $\varphi : X \longrightarrow Y$ and publishes it.
- Bob wants to send to Alice the message $x \in X$. He computes $\varphi(x) \in Y$ and sends this to Alice.

Principle of public key cryptography

- Alice constructs a one-way trapdoor function $\varphi : X \longrightarrow Y$ and publishes it.
- Bob wants to send to Alice the message $x \in X$. He computes $\varphi(x) \in Y$ and sends this to Alice.
- Only Alice knows how to compute $x = \varphi^{-1}(\varphi(x))$.

Principle of public key cryptography

- Alice constructs a one-way trapdoor function $\varphi : X \longrightarrow Y$ and publishes it.
- Bob wants to send to Alice the message $x \in X$. He computes $\varphi(x) \in Y$ and sends this to Alice.
- Only Alice knows how to compute $x = \varphi^{-1}(\varphi(x))$.

Remark 6 In practices $x \in X$ represents often the key for some secret key system. The importance of one-way trapdoor functions was recognized by Diffie and Hellman in 1976.

Application: Digital signatures:

Alice wants to sign a document electronically. For this she has to deposit a one-way trapdoor function $\varphi : X \longrightarrow Y$ with a 'trusted party'.
She wants to sign the message:

$$y = \text{ Alice, Zürich, August 29, 2006}$$

She sends to Bob $\varphi^{-1}(y) = x$.
Bob verifies the signature by computing $\varphi(x) = y$.

Applications for one-way trapdoor func

One-way trapdoor functions are the building block of many interesting applications such as:

- Secret key exchange

Applications for one-way trapdoor func

One-way trapdoor functions are the building block of many interesting applications such as:

- Secret key exchange
- Digital signatures

Applications for one-way trapdoor func

One-way trapdoor functions are the building block of many interesting applications such as:

- Secret key exchange
- Digital signatures
- Authentication protocols

Applications for one-way trapdoor func

One-way trapdoor functions are the building block of many interesting applications such as:

- Secret key exchange
- Digital signatures
- Authentication protocols
- Digital Cash system

Applications for one-way trapdoor func

One-way trapdoor functions are the building block of many interesting applications such as:

- Secret key exchange
- Digital signatures
- Authentication protocols
- Digital Cash system
- Zero knowledge proofs

5. The RSA public key system

- Prime numbers in the range of 10^{100} can efficiently be computed with ‘Monte Carlo’ methods.

5. The RSA public key system

- Prime numbers in the range of 10^{100} can efficiently be computed with ‘Monte Carlo’ methods.
- If p, q are primes in the range of 10^{100} then it is computationally unknown how to factor $n = pq$.

5. The RSA public key system

- Prime numbers in the range of 10^{100} can efficiently be computed with ‘Monte Carlo’ methods.
- If p, q are primes in the range of 10^{100} then it is computationally unknown how to factor $n = pq$.
- The multiplicative group $(\mathbb{Z}_n)^*$ has $\phi(n) = (p-1)(q-1)$ elements.

5. The RSA public key system

- Prime numbers in the range of 10^{100} can efficiently be computed with ‘Monte Carlo’ methods.
- If p, q are primes in the range of 10^{100} then it is computationally unknown how to factor $n = pq$.
- The multiplicative group $(\mathbb{Z}_n)^*$ has $\phi(n) = (p-1)(q-1)$ elements.
- With the help of ‘consecutive squaring’ one can efficiently compute x^e inside $(\mathbb{Z}_n)^*$.

RSA one way trapdoor function

Alice constructs an integer $n = pq$ which has prime factors in the range of 10^{100}. She chooses a random number $e < n$, which is coprime to $\phi(n) = (p-1)(q-1)$. She publishes the one-way trapdoor function:

$$\begin{aligned} \varphi : \mathbb{Z}_n &\longrightarrow \mathbb{Z}_n \\ m &\longmapsto m^e = c \end{aligned} \tag{4}$$

RSA

If Bob wants to send to Alice a message m he sends to Alice m^e. Alice knows the group order of $(\mathbb{Z}_n)^*$ and this allows her to compute m. She applies Euclid's algorithm and computes $d, b \in \mathbb{Z}$ having the property that $de + b\phi(n) = 1$. The inverse function which only Alice knows is:

$$\begin{aligned} \mathbb{Z}_n &\longrightarrow \mathbb{Z}_n \\ c &\longmapsto c^d = m \end{aligned}$$

RSA

If Bob wants to send to Alice a message m he sends to Alice m^e. Alice knows the group order of $(\mathbb{Z}_n)^*$ and this allows her to compute m. She applies Euclid's algorithm and computes $d, b \in \mathbb{Z}$ having the property that $de + b\phi(n) = 1$. The inverse function which only Alice knows is:

$$\begin{array}{rcl} \mathbb{Z}_n & \longrightarrow & \mathbb{Z}_n \\ c & \longmapsto & c^d = m \end{array}$$

Verify:

$$c^d = (m^e)^d = m^{de} = m^{1-b\phi(n)} = m \cdot \left(m^{\phi(n)}\right)^{-b} = m$$

Remark

The RSA system is based on the fact that Alice knows the group order of the multiplicative group $(\mathbb{Z}_n)^*$ and an eavesdropper, say Eve, does not know this.
Whenever one can construct a group G where the designer knows the group order and the general public does not know it one has a potential one-way trapdoor function of the form:

$$\begin{array}{ccc} G & \longrightarrow & G \\ m & \longmapsto & m^e = c \end{array}$$

Security issues for RSA

- The modulus n has to have at least 1024 bits, the record for factoring a general number stands at 640 bits.

Security issues for RSA

- The modulus n has to have at least 1024 bits, the record for factoring a general number stands at 640 bits.
- The factors p, q of n have to be chosen 'sufficiently random'. E.g. all the numbers $p \pm 1$ and $q \pm 1$ should contain large prime factors.

Security issues for RSA

- The modulus n has to have at least 1024 bits, the record for factoring a general number stands at 640 bits.
- The factors p, q of n have to be chosen 'sufficiently random'. E.g. all the numbers $p \pm 1$ and $q \pm 1$ should contain large prime factors.
- The decryption exponent d has to be chosen 'large'.

Security issues for RSA

- The modulus n has to have at least 1024 bits, the record for factoring a general number stands at 640 bits.
- The factors p, q of n have to be chosen 'sufficiently random'. E.g. all the numbers $p \pm 1$ and $q \pm 1$ should contain large prime factors.
- The decryption exponent d has to be chosen 'large'.
- The submitted message has to be 'randomized' [BJN00] and acknowledgment of the server has to avoid the 'Bleichenbacher attack'

6. The Discrete Logarithm Problem

Definition 7 Let G be an arbitrary group, $\alpha \in G$ an arbitrary element and $H := <\alpha> \subset G$ the cyclic group generated by α. Assume $\beta \in H$ is an arbitrary element. The unique integer n having the property that $1 \leq n < |H|$ and $\alpha^n = \beta$ is called the discrete logarithm of β to the base α.

Notation 8 $\log_\alpha \beta = n.$

One has the usual computations:

$$\alpha^{(\log_\alpha \beta)} = \beta, \;\; \log_\alpha(\alpha^n) = n$$

$$\log_\alpha(\beta_1\beta_2) = \log_\alpha(\beta_1) + \log_\alpha(\beta_2) \mod |H|$$

Diffie-Hellman protocol [DH76]

Alice and Bob want to exchange a secret key over some insecure channel. In order to achieve this goal Alice and Bob agree on a group H and a common base $g \in H$. Alice chooses a random integer $a \in \mathbb{N}$ and Bob chooses a random integer $b \in \mathbb{N}$. Alice transmits to Bob g^a and Bob transmits to Alice g^b. Their common secret key is $k := g^{ab}$.

El Gamal one way trapdoor function:

Let $< \alpha >= H$ be a cyclic group, where it is known that the discrete logarithm problem is 'hard'. Let n be an integer $1 < n < |H|$ and compute $\beta := \alpha^n$.

Public Key: (α, β, G)

Encryption: $H \longrightarrow H \times H$

$$x \longmapsto (\alpha^k, x\beta^k) =: (c_1, c_2),$$

where k has been randomly chosen by Alice.

Bob, with the knowledge of n is able to compute x from the cipher text c_1, c_2:

$$x = c_2 \left((c_1)^n \right)^{-1}.$$

Groups where the DLP is interesting

Every finite cyclic group is isomorphic to $(\mathbb{Z}_n, +)$ and for this group the DLP is trivial.

The difficulty of the discrete logarithm problem in a finite cyclic group of order n is equivalent to finding an explicit isomorphism to the group $(\mathbb{Z}_n, +)$.

In practice the following groups were studied:

- $(\mathbb{F}_q)^*$, $(\mathbb{Z}_n)^*$.

Groups where the DLP is interesting

Every finite cyclic group is isomorphic to $(\mathbb{Z}_n, +)$ and for this group the DLP is trivial.

The difficulty of the discrete logarithm problem in a finite cyclic group of order n is equivalent to finding an explicit isomorphism to the group $(\mathbb{Z}_n, +)$.

In practice the following groups were studied:

- $(\mathbb{F}_q)^*$, $(\mathbb{Z}_n)^*$.
- $Gl_m(\mathbb{F}_q)$

Groups where the DLP is interesting

Every finite cyclic group is isomorphic to $(\mathbb{Z}_n, +)$ and for this group the DLP is trivial.

The difficulty of the discrete logarithm problem in a finite cyclic group of order n is equivalent to finding an explicit isomorphism to the group $(\mathbb{Z}_n, +)$.

In practice the following groups were studied:

- $(\mathbb{F}_q)^*$, $(\mathbb{Z}_n)^*$.
- $Gl_m(\mathbb{F}_q)$
- $E(\mathbb{F}_p)$, $E(\mathbb{F}_{2^n})$, where E is an elliptic curve over a finite field.

Groups where the DLP is interesting

Every finite cyclic group is isomorphic to $(\mathbb{Z}_n, +)$ and for this group the DLP is trivial.

The difficulty of the discrete logarithm problem in a finite cyclic group of order n is equivalent to finding an explicit isomorphism to the group $(\mathbb{Z}_n, +)$.

In practice the following groups were studied:

- $(\mathbb{F}_q)^*$, $(\mathbb{Z}_n)^*$.
- $Gl_m(\mathbb{F}_q)$
- $E(\mathbb{F}_p)$, $E(\mathbb{F}_{2^n})$, where E is an elliptic curve over a finite field.
- The Jacobian group $J_C(\mathbb{F}_q)$ over an elliptic curve and more general abelian varieties.

Remarks on Complexity of the DLP

$$L_n(\alpha, c) := O\left(e^{c(\log n)^{\alpha}(\log\log n)^{1-\alpha}}\right).$$

Remarks on Complexity of the DLP

$$L_n(\alpha, c) := O\left(e^{c(\log n)^{\alpha}(\log\log n)^{1-\alpha}}\right).$$

With $\alpha = 0$ this reduces to $O((\log n)^c)$, i. e. polynomial time.

Remarks on Complexity of the DLP

$$L_n(\alpha, c) := o\left(e^{c(\log n)^{\alpha}(\log\log n)^{1-\alpha}}\right).$$

With $\alpha = 0$ this reduces to $o\left((\log n)^c\right)$, i. e. polynomial time.

With $\alpha = 1$ this reduces to $o\left(n^c\right)$, i. e. exponential time.

Remarks on Complexity of the DLP

$$L_n(\alpha, c) := O\left(e^{c(\log n)^{\alpha}(\log\log n)^{1-\alpha}}\right).$$

With $\alpha = 0$ this reduces to $O((\log n)^c)$, i. e. polynomial time.

With $\alpha = 1$ this reduces to $O(n^c)$, i. e. exponential time.

Algorithms having a complexity $L_n(\alpha, c)$ with $0 < \alpha < 1$ are called *sub-exponential time algorithms*.

Remarks on Complexity of the DLP

$$L_n(\alpha, c) := O\left(e^{c(\log n)^{\alpha}(\log\log n)^{1-\alpha}}\right).$$

With $\alpha = 0$ this reduces to $O((\log n)^c)$, i. e. polynomial time.

With $\alpha = 1$ this reduces to $O(n^c)$, i. e. exponential time.

Algorithms having a complexity $L_n(\alpha, c)$ with $0 < \alpha < 1$ are called *sub-exponential time algorithms*.

The best known algorithms for factoring a number n or for solving a DLP in $\mathbb{F}_q^*$ is the *generalized number field sieve*, which has $L_n(\frac{1}{3}, c)$.

Remarks on Complexity of the DLP

$$L_n(\alpha, c) := O\left(e^{c(\log n)^{\alpha}(\log\log n)^{1-\alpha}}\right).$$

With $\alpha = 0$ this reduces to $O((\log n)^c)$, i. e. polynomial time.

With $\alpha = 1$ this reduces to $O(n^c)$, i. e. exponential time.

Algorithms having a complexity $L_n(\alpha, c)$ with $0 < \alpha < 1$ are called *sub-exponential time algorithms*.

The best known algorithms for factoring a number n or for solving a DLP in $\mathbb{F}_q^*$ is the *generalized number field sieve*, which has $L_n(\frac{1}{3}, c)$.

The best known algorithm for solving the discrete logarithm problem over an elliptic curve $E(\mathbb{F}_p)$ has exponential complexity.

Consequence

- Systems based on the hardness of factoring or the hardness of the DLP in $\mathbb{F}_q^*$ necessarily have to work with public keys of more than 1000 bits.

Consequence

- Systems based on the hardness of factoring or the hardness of the DLP in $\mathbb{F}_q^*$ necessarily have to work with public keys of more than 1000 bits.
- Systems based on the DLP over an elliptic curve $E(\mathbb{F}_p)$ are considered secure if the group size is more than 2^{160}.

Remarks on Quantum Computer

- The practically implemented public key crypto systems are based on the hardness of factorization integers and on the discrete logarithm problem.

Remarks on Quantum Computer

- The practically implemented public key crypto systems are based on the hardness of factorization integers and on the discrete logarithm problem.
- It has recently been shown by Shor [Sho99] that factorization of integers and the discrete logarithm problem are both polynomial time problems on a quantum computer. This means that if a quantum computer can ever be physically realized then most implemented public key protocols become insecure immediately.

7. Systems Based on Group Actions

Let G be an abelian semi-group, let S be a finite set and consider the action of G on S:

$$\begin{array}{rccc} \varphi: & G \times S & \longrightarrow & S \\ & (g,s) & \longmapsto & gs \end{array}$$

We will refer to this action as a G-action on the set S. By the definition of a group action we require that $(g \cdot h)s = g(hs)$ for all $g, h \in G$ and $s \in S$.

Generalized Diffie-Hellman protocol

Let S be a finite set, G an abelian semi-group and an action of G on S as defined above. The Extended Diffie-Hellman key exchange is the following protocol:

- Alice and Bob agree on an element $s \in S$.

Generalized Diffie-Hellman protocol

Let S be a finite set, G an abelian semi-group and an action of G on S as defined above. The Extended Diffie-Hellman key exchange is the following protocol:

- Alice and Bob agree on an element $s \in S$.
- Alice chooses $a \in G$ and computes as. Alice's secret key is a, her public key is as.

Generalized Diffie-Hellman protocol

Let S be a finite set, G an abelian semi-group and an action of G on S as defined above. The Extended Diffie-Hellman key exchange is the following protocol:

- Alice and Bob agree on an element $s \in S$.
- Alice chooses $a \in G$ and computes as. Alice's secret key is a, her public key is as.
- Bob chooses $b \in G$ and computes bs. Bob's secret key is b, his public key is bs.

Generalized Diffie-Hellman protocol

Let S be a finite set, G an abelian semi-group and an action of G on S as defined above. The Extended Diffie-Hellman key exchange is the following protocol:

- Alice and Bob agree on an element $s \in S$.
- Alice chooses $a \in G$ and computes as. Alice's secret key is a, her public key is as.
- Bob chooses $b \in G$ and computes bs. Bob's secret key is b, his public key is bs.
- Their common secret key is then

$$a(bs) = (a \cdot b)s = (b \cdot a)s = b(as)$$

Extended ElGamal public key system

If S is a group with respect to some operation $\circ$, then the Extended ElGamal public key system is the following protocol:

- Alice's public key is (s, as).

Extended ElGamal public key system

If S is a group with respect to some operation $\circ$, then the Extended ElGamal public key system is the following protocol:

- Alice's public key is (s, as).
- Bob chooses a random element $b \in G$ and encrypts a message m using the encryption function

$$(m, b) \longmapsto (bs, (b(as)) \circ m) = (c_1, c_2).$$

Extended ElGamal public key system

If S is a group with respect to some operation $\circ$, then the Extended ElGamal public key system is the following protocol:

- Alice's public key is (s, as).
- Bob chooses a random element $b \in G$ and encrypts a message m using the encryption function

$$(m, b) \longmapsto (bs, (b(as)) \circ m) = (c_1, c_2).$$

- Alice can decrypt the message using

$$m = (b(as))^{-1} \circ c_2 = (ac_1)^{-1} \circ c_2.$$

Example 9 Integers $\mathbb{Z}$ act on an abelian group H. This leads to the usual discrete logarithm problem.

Example 10 Any abelian group H comes with its ring of endomorphisms $\mathrm{End}\,H$ where addition is defined pointwise and multiplication via composition of maps. There is a natural action of $\mathrm{End}\,H$ on H as follows :

$$\begin{array}{rcl} \mathrm{End}\,H \times H & \longrightarrow & H \\ (\varphi, h) & \longmapsto & \varphi(h) \end{array}$$

For a given $\varphi \in \mathrm{End}\,H$, the subring $\mathbb{Z}[\varphi]$ of $\mathrm{End}\,H$ is commutative and yields to a Diffie-Hellman protocol.

Special situation

Let $\mathbb{F}_p$ be a prime finite field ($p > 3$), $\overline{\mathbb{F}_p}$ its algebraic closure and $E : y^2 = x^3 + ax + b$ an ordinary elliptic curve over $\mathbb{F}_p$ with complex multiplication. In this case, it is known that $\mathrm{End}\, E(\overline{\mathbb{F}_p}) \cong \mathbb{Z} \oplus \mathbb{Z}\varphi$, where φ is the Frobenius endomorphism:

$$\begin{aligned} \varphi : E(\overline{\mathbb{F}_p}) &\longrightarrow E(\overline{\mathbb{F}_p}) \\ (x, y) &\longrightarrow (x^p, y^p) \end{aligned}$$

Chebyshev action

Definition 11

$$T_n(x) = \cos(n \cos^{-1} x) = \sum_{k=0}^{\lfloor n/2 \rfloor} \binom{n}{2k} (-1)^k x^{n-2k} (1-x^2)^k$$

is called the nth Chebyshev polynomial.

Theorem 12 *$T_{nm}(x) = T_n(T_m(x))$ in $\mathbb{Z}[x]$. In particular if R is any finite semiring then $T_n(r)$ can be efficiently computed for any $r \in R$ and $n \in \mathbb{N}$.*

Actions on semi-modules

Let R be a semiring, not necessarily finite.
(Two operations '$+$' and '$\cdot$' which are distributive and associative. We assume also that '$+$' is commutative. No neutral elements assumed.)
Let M be a finite semi-module over R. With this we mean that M has the structure of a finite semi-group and there is an action $R \times M \longrightarrow M$ such that

$$\begin{aligned} r(sm) &= (rs)m, \\ (r+s)m &= rm+sm, \\ r(m+n) &= rm+rn. \end{aligned}$$

for all $r,s \in R$ and $m,n \in M$.

Let $Mat_{n\times n}(R)$ be the set of all $n \times n$ matrices with entries in R. The semi-ring structure on R induces a semi-ring structure on $Mat_{n\times n}(R)$. Moreover the semi-module structure on M lifts to a semi-module structure on M^n via the matrix multiplication:

$$\begin{aligned} Mat_{n\times n}(R) \times M^n &\longrightarrow M^n \\ (A, x) &\longmapsto Ax. \end{aligned} \tag{4}$$

One readily verifies that $Mat_{n\times n}(R) \times M^n \longrightarrow M^n$ is an action by a semi-group, indeed one readily computes that $A(Bg) = (AB)g$.

Commutative semi-groups

Let $R[t]$ be the polynomial ring in the indeterminant t and let $A \in Mat_{n\times n}(R)$ be a fixed matrix. Let $C \subset R$ be the center of R. If

$$p(t) = r_0 + r_1 t + \cdots + r_k t^k \in C[t]$$

then we define in the usual way $p(A) = r_0 I_n + r_1 A + \cdots + r_k A^k$, where $r_0 I_n$ is the $n \times n$ diagonal matrix with entry r_0 in each diagonal element.
Consider the semi-group

$$G := C[A] := \{p(A) \mid p(t) \in C[t]\}.$$

Clearly G has the structure of an abelian semi-group.

Diffie-Hellman protocol

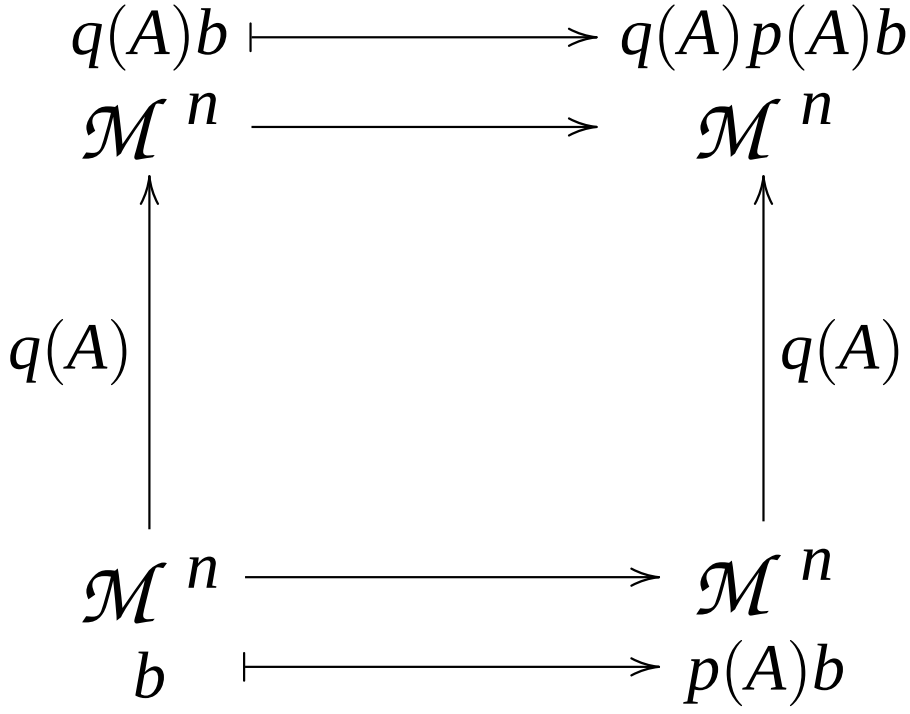
Alice and Bob agree on an R-module $\mathcal{M}$, an element $b \in \mathcal{M}^n$ and a matrix $A \in Mat_{n\times n}(R)$.
Alice chooses secretly $p(t) \in C[t]$ and computes $p(A)b$ and sends the result to Bob. Bob chooses secretly $q(t) \in C[t]$ and computes $q(A)b$ and sends the result to Alice.
As a common secret key serves $k := p(A)q(A)b$

Nota Bene:

It should be difficult to find $\tilde{p}(t) \in C[t]$ such that

$$\tilde{p}(A)b = p(A)b.$$

In Diagram:


$q(A)b$
$q(A)p(A)b$
$\mathcal{M}^n$
$\mathcal{M}^n$
$q(A)$
$q(A)$
$\mathcal{M}^n$
$\mathcal{M}^n$
b
$p(A)b$

Systems theory interpretation

Let $b \in \mathcal{M}^n$ and $A \in Mat_{n\times n}(R)$. Then the computation of $p(A)b$ can be iteratively done through the linear system:

$$x_{t+1} = Ax_t + bu_t,\ x_t \in \mathcal{M}^n,\ u_t \in R.$$

Eve is faced with the task of finding a control sequence $u_0, u_1, \ldots, u_n$ which steers the zero state $x_0 = 0$ to $p(A)b$.

When the semiring R is a field and the module $\mathcal{M}$ is vector space over this field then the problem is trivially solved by linear algebra. If R and $\mathcal{M}$ have less structure then the problem becomes computationally hard.

Special situation

$R = \mathbb{Z}$, the integers and as a module any finite abelian group $M = H$. The group H is a $\mathbb{Z}$ module and $Mat_{n\times n}(\mathbb{Z})$ operates on $S := H^n = H \times \ldots \times H$ via the formal multiplication:

$$\begin{bmatrix} g_1 \\ \vdots \\ g_n \end{bmatrix} \longmapsto \begin{bmatrix} a_{11} & \ldots & a_{1n} \\ \vdots & & \vdots \\ a_{n1} & \ldots & a_{nn} \end{bmatrix} \begin{bmatrix} g_1 \\ \vdots \\ g_n \end{bmatrix}. \tag{5}$$

How difficult is the reachability problem for the system:

$$x_{t+1} = Ax_t + bu_t,$$

$b, x_t \in H^n$, $u_t \in \mathbb{Z}$, and $A \in Mat_{n\times n}(\mathbb{Z})$.
E.g. When H is the abelian group over an elliptic curve then this is a control problem on the divisor group!

Computational difficulty

Example 13 $R = \mathbb{Z}$, the integers and $\mathcal{M}$ the abelian group over an elliptic curve. Even when $n = 1$ this is a very hard problem. When $n = 1$ the problem asks:

> Given two points P, Q on an elliptic curve. Find an integer $m \in \mathbb{Z}$ such that $Q = mP$.
> (This is a discrete logarithm problem).

Systems theory over simple semirings

Remark: Good simple semi-rings and semi-modules are needed. If a semi-ring has some proper subrings then a 'Pohlig-Hellman' type reduction is usually possible.

Example 14 Consider the discrete logarithm problem in the semi-group $G := Mat_{n\times n}(R)$, where $R = \mathbb{Z}_6$. Reduction modulo 2 and modulo 3 reduces the problem to two simpler instances which can be solved separately.

Simple semirings

Definition 15 A **congruence relation** on a semiring R is an equivalence relation $\sim$ that also satisfies

$$x_1 \sim x_2 \Rightarrow \begin{cases} c + x_1 & \sim & c + x_2, \\ x_1 + c & \sim & x_2 + c, \\ cx_1 & \sim & cx_2, \\ x_1 c & \sim & x_2 c, \end{cases}$$

for all $x_1, x_2, c \in R$. A semiring R that admits no congruence relations other than the trivial ones, id_R and $R \times R$, is said to be **congruence-simple**, or **c-simple**.

Simple semirings

Theorem 16 (Monico [Mon02]) *Let R be a finite, additively commutative, congruence-simple semiring. Then one of the following holds:*

1. $|R| = 2$.
2. $R \cong Mat_{n\times n}(\mathbb{F}_q)$ *for some finite field* $\mathbb{F}_q$ *and some* $n \geq 1$.
3. R *is a zero multiplication ring of prime order.*
4. R *is additively idempotent.*
5. *There is an infinite element* ∞ *such that* $R + R = \{\infty\}$. *(*∞ *is an element having the property that* $\infty r = r\infty = \infty + r = r + \infty = \infty$*).*

A simple semiring of order 2

$+$	0	1
0	0	1
1	1	1

$*$	0	1
0	0	0
1	1	1

A Simple Semiring of order 3

$+$	0	1	2
0	0	1	2
1	1	1	2
2	2	2	2

$*$	0	1	2
0	0	0	0
1	0	1	2
2	2	2	2

A simple semiring of order 6

$+$	0	1	2	3	4	5
0	0	1	2	3	4	5
1	1	1	1	1	1	5
2	2	1	2	1	2	5
3	3	1	1	3	3	5
4	4	1	2	3	4	5
5	5	5	5	5	5	5

$*$	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	1	2	3	4	5
2	0	2	2	0	0	5
3	0	3	4	3	4	3
4	0	4	4	0	0	3
5	0	5	2	5	2	5

Simple matrix rings

Theorem 17 (Maze [Maz03]) *When the semiring R is simple with $0,1$ then the matrix ring $Mat_{n\times n}(R)$ is simple.*

Lemma 18 *When the semiring R is simple with $0,1$ then $Mat_{n\times n}(R)$ contains elements of order*

$$g(n) = \max\{\mathrm{ord}\sigma \mid \sigma \in S_n\}$$
$$= \max\{\mathrm{lcm}\{a_1,\ldots,a_m\} \mid a_i > 0, a_1 + \cdots + a_m = n\}$$

In particular the order grows exponentially in n.

Example

Assume a matrix is given as:

```
0  2  0  0  0  0  0  0  0  3
2  0  3  0  0  0  1  0  0  0
0  0  0  0  2  0  0  0  0  0
0  0  2  0  0  0  0  0  0  0
0  0  0  2  0  5  0  0  0  0
0  0  0  0  4  0  0  0  0  2
0  0  3  0  0  2  0  0  0  0
0  0  0  0  0  0  2  0  0  0
0  0  0  0  0  0  0  2  0  0
0  3  0  0  0  0  0  0  2  0
```

Example

What power has the following matrix?

2	3	3	3	3	3	2	2	3	2
3	2	3	3	3	2	1	3	2	3
0	5	2	1	5	5	5	0	5	5
5	0	5	2	1	5	1	5	0	5
5	5	5	5	2	5	1	5	5	5
3	3	3	4	3	3	3	3	3	2
3	3	3	3	4	2	4	3	3	3
0	3	0	4	3	3	2	0	3	3
3	0	3	0	4	0	4	2	0	3
3	3	3	3	3	3	3	4	2	3

Systems theory questions

- How difficult is the reachability question over a semiring R for the linear system:

$$x_{t+1} = Ax_t + bu_t,$$

A, b, x_t and u_t all defined over R.

Systems theory questions

- How difficult is the reachability question over a semiring R for the linear system:

$$x_{t+1} = Ax_t + bu_t,$$

A, b, x_t and u_t all defined over R.

- How difficult is the nonlinear analogon?
For this assume that $f : R^{m+n} \longrightarrow R^n$ is a polynomial map.
Then consider the nonlinear system:

$$x_{t+1} = f(x_t, u_t),\ x_t \in R^n,\ u_t \in R^m.$$

Systems theory questions

- How difficult is the reachability question over a semiring R for the linear system:

$$x_{t+1} = Ax_t + bu_t,$$

A, b, x_t and u_t all defined over R.

- How difficult is the nonlinear analogon?
For this assume that $f : R^{m+n} \longrightarrow R^n$ is a polynomial map.
Then consider the nonlinear system:

$$x_{t+1} = f(x_t, u_t),\ x_t \in R^n,\ u_t \in R^m.$$

- Is it possible to identify a class of nonlinear systems where the control serves as trapdoor and a Diffie-Hellman exchange is possible.

An interesting semi-group action

Alice and Bob agree on a simple semi-ring R having center $C \subset R$ and agree on three matrices

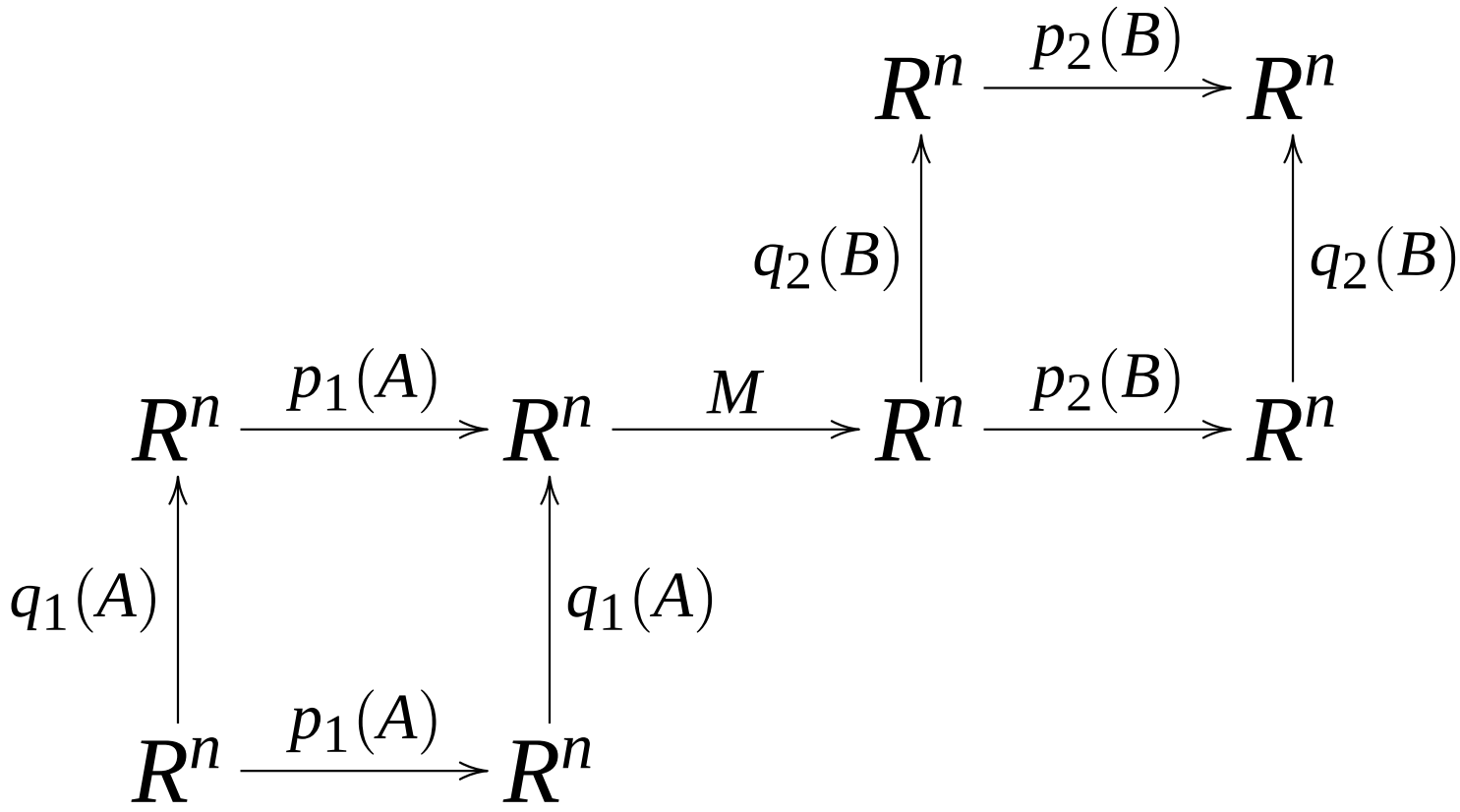
$$A, B, M \in Mat_{n \times n}(R).$$

Alice chooses secretly $p_1(t), p_2(t) \in C[t]$ and computes $p_1(A)Mp_2(B)$ and sends the result to Bob. Bob chooses secretly $q_1(t), q_2(t) \in C[t]$ and computes $q_1(A)Mq_2(B)$ and sends the result to Alice.
As a common secret key serves

$$k := p_1(A)q_1(A)Mq_2(B)p_2(B)$$

which both can easily compute.

In Diagram:



Example:

As a concrete choice let assume that $n = 20$. Consider the matrices

$$A = \begin{bmatrix}
1000000000000000000 \\
0010000000000000000 \\
0001000000000000000 \\
0000100000000000000 \\
0100000000000000000 \\
0000001000000000000 \\
0000000200000000010 \\
0000010000000000000 \\
0000000001000000000 \\
0000000001000000000 \\
0000000000200000000 \\
0000000000010000000 \\
0000000010000000000 \\
0000000000000100000 \\
0000000000000010000 \\
0000000000000001000 \\
0000000000000000100 \\
0000000000000000010 \\
0000000000000000001 \\
0000000000001000000
\end{bmatrix} \quad B = \begin{bmatrix}
0000000000000000010 \\
0000000000100000000 \\
0000001000000000000 \\
0010000000000000000 \\
0000000000000000004 \\
0000000000000010000 \\
0100000000000000000 \\
0000000000000000100 \\
0001000010000000000 \\
0000000000310000000 \\
0000000000000200000 \\
0001000000000000100 \\
0000000001000000000 \\
0000010000000000000 \\
0000000001000000000 \\
0000001000000000000 \\
1000000000000000000 \\
0000100000000000000 \\
0000000000000001000 \\
0000000000001000000
\end{bmatrix}$$

Example

$$M = \begin{bmatrix}
00200000000000000100 \\
01000000010001000000 \\
00000001000000000030 \\
20020000000010000000 \\
00000010000000001000 \\
00000005000100000001 \\
00000000200010000001 \\
01000000030000000003 \\
00000002000000010001 \\
01000100000010000000 \\
00000000000050100000 \\
00000000000004000000 \\
00000000000000100500 \\
00300000002000100000 \\
00001000000200001000 \\
00000002000000000100 \\
00002000001000000000 \\
00100000000100000000 \\
00020001000000000030 \\
10000001000010000001
\end{bmatrix} \qquad T = \begin{bmatrix}
02020000000204000200 \\
00111411002100241114 \\
30111011002000240134 \\
12000020020200202034 \\
22111424020100201110 \\
12222020022220222212 \\
11111014222124211122 \\
21111014222124222124 \\
00222020022022200200 \\
00002000022020220000 \\
00222020020000200200 \\
00000000022022200000 \\
00002000000000020200 \\
03333404021324040300 \\
02202420020020001010 \\
01111014000104040104 \\
32000020020220000034 \\
11111014020104211104 \\
31333424021124040334 \\
12202420020000211014
\end{bmatrix}.$$

The task of Eve will be to find $p_1(t), p_2(t) \in C[t]$ such that $p_1(A)Mp_2(B) = T$.

Questions:

1. Find all finite simple semirings of order smaller than 10.

Questions:

1. Find all finite simple semirings of order smaller than 10.

2. Develop a linear systems theory over semirings and semi-modules.

Questions:

1. Find all finite simple semirings of order smaller than 10.

2. Develop a linear systems theory over semirings and semi-modules.

3. Study 'how difficult it is' to solve linear algebra problems over a semi-ring such as the 'reachability problem for a system of the form $x_{t+1} = Ax_t + bu_t$.

Questions:

1. Find all finite simple semirings of order smaller than 10.

2. Develop a linear systems theory over semirings and semi-modules.

3. Study 'how difficult it is' to solve linear algebra problems over a semi-ring such as the 'reachability problem for a system of the form $x_{t+1} = Ax_t + bu_t$.

4. Come up with new semi-group actions on a finite set where the problem: Given as and s, find α such that $as = \alpha s$ is provable complex.

References

[BJN00] D. Boneh, A. Joux, and P. Q. Nguyen, *Why textbook ElGamal and RSA encryption are insecure (extended abstract)*, Advances in cryptology—ASIACRYPT 2000 (Kyoto), Lecture Notes in Comput. Sci., vol. 1976, Springer, Berlin, 2000, pp. 30–43.

[DH76] W. Diffie and M. E. Hellman, *New directions in cryptography*, IEEE Trans. Inform. Theory **IT-22** (1976), no. 6, 644–654.

[Mas69] J. L. Massey, *Shift-register synthesis and BCH decoding*, IEEE Trans. Inform. Theory **IT-15** (1969), 122–127.

[Maz03] G. Maze, *Algebraic methods for constructing one-way trapdoor functions*, Ph.D. thesis, University of Notre Dame, May 2003.

[Mon02] C. Monico, *Semirings and semigroup actions in public-key cryptography*, Ph.D. thesis, University of Notre Dame, May 2002, Available at http://www.nd.edu/~rosen/preprints.html.

[Ros03] J. Rosenthal, *A polynomial description of the Rijndael advanced encryption standard*, J. Algebra Appl. **2** (2003), no. 2, 223–236.

[Sha49] C. E. Shannon, *Communication theory of secrecy systems*, Bell System Tech. J. **28** (1949), 656–715.

[Sho99] P. W. Shor, *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM Rev. **41** (1999), no. 2, 303–332 (electronic).