Belarusian State Technological University
Department of Information Systems and Technology

Pavel Urbanovich

INFORMATION PROTECTION

Part 7: BASIC CRYPTOGRAPHIC ALGORITHMS AND STANDARDS

pav.urb@yandex.by, p.urbanovich@belstu.by

Encryption/Decryption with symmetric key

Definition 1. Let the encryption scheme consist of sets of encryption and decryption functions, $\{E_e: e \in K\}$ and $\{D_d: d \in K\}$ respectively where K is the set of keys. Then such a scheme is called a symmetric key scheme, if for each attached key pair (e, d), it is easy to designate the key d, knowing only the key e, and vice versa: in most symmetric key schemes $e = d$.

Classical encryption techniques with symmetric key:

- ***Substitution***: each character in the text is replaced by another character of the same or different alphabet,
- ***Permutation (Transposition)***: the order, but not the value, of the characters in the text is changed,
- ***A combination of the Substitution and Permutation (S&P)***.

Simple Substitution: cyclic permutation
Caesar's cipher

- Replace each character in plaintext with the character **3** positions forward in the alphabet. If the end of the alphabet is reached, start over in the alphabet.
- Julius Caesar in his work "Galilean War" describes the first cipher based on the substitution method.
- Caesar describes how he sent a letter to Cicero. This cipher consists in assigning each of the letters of the Latin alphabet corresponding to the digit in the range [0,25] (a = 0, b = 1, ..., z = 25) and then for each letter denoted by ***x*** of the execution of the operation
y = (x + 3) mod 26. Decryption consists in executing the action
x = (y - 3) mod 26, for the letter denoted by ***y***.

0	1	2	3	4	5					10														25
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
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

Example. The famous message of Caesar ‘VENI VIDI VICI’ (*Came*, *Seen*, *Won*)
Encrypted: ‘YHQL YLGL YLFL’

Applying simultaneously addition and multiplication modulo ***n*** over elements of the set (indexes of letters of the alphabet), we can obtain a system of substitutions, which is called **the affine system of Caesar's substitutions (Affine cipher).**

The Affine cipher is a special case of the more general monoalphabetic substitution cipher. The 'key' for the Affine cipher consists of 2 numbers, we'll call them *a* and *b*.

We define the transformation (encryption) in such system:

$$y = ax + b \ (mod\ n),$$

where *a,b* – integers, $0 \le a, bn < n$, $GCD\ (a,b) = 1$.

Example. Let $n = 26$, $a = 3$, $b = 5$.

x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
3x+5	5	8	11	14	17	20	23	0	3	6	9	12	15	18	21	24	1	4	7	10	13	16	19	22	25	2

Two using alphabets:

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
F	I	L	O	R	U	X	A	D	G	J	M	P	S	V	Y	B	E	H	K	N	Q	T	Q	Z	C

The decryption function is:

$$x = a^{-1} (y-b) \pmod{n},$$

where a^{-1} is the multiplicative inverse of a in the group of integers modulo n.
To find a multiplicative inverse, we need to find a number x such that:

$$az = 1 \bmod n.$$

If we find the number z such that the equation is true,
then z is the inverse of a, and we call it a^{-1}.

To find a^{-1} we use the the extended Euclidean algorithm (see sl.33-35, part 6).

The easiest way to solve this equation is to search each of the numbers 1 to 25,
and see which one satisfies the equation.

For our **example**: Believe that x=1 (character ‘b’), than y=8 (character ‘i’;encryption).
Decryption: we need to solve the equation:

$$x = 3^{-1} (8-5) \pmod{26},$$

to find a number z such that $3z = 1 \bmod 26$ the extended Euclidean algorithm and get $z = 9$; it means that $a^{-1} = 9$.

And finally
$x = a^{-1} (y-b) \pmod{n} = 9*3 \bmod 26 = 1$ (character ‘b’).

Cryptanalysis of the Affine Cipher

- The Affine cipher is a very insecure cipher, with the Caesar cipher possibly being the only easier cipher to crack.
- The Affine cipher is a monoalphabetic substitution cipher, so all the methods that are used to cryptanalyse substitution ciphers can be used for the affine cipher.
- Affine ciphers can also be cracked if any 2 characters are known.

Caesar's cipher with a key word

- It is desirable that all the letters of the keyword are different.
- A special feature of this system is using a keyword for the displacement and change the order of characters in the alphabet substitution and a certain number of a, $0 \le a < n$.

Example. Let keyword is 'DIPLOMAT', a=5.

Write the keyword (does not contain duplicate characters) with a=5:

0	1	2	3	4	5				10					15					20				25
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
					D	I	P	L	O	M	A	T												

The remaining letters of the substitution alphabet are written after the keyword in alphabetical order:

					5																			
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
V	W	X	Y	Z	**D**	**I**	**P**	**L**	**O**	**M**	**A**	**T**	B	C	E	F	G	H	J	K	N	Q	R	S	U

If the open message is BELSTU, then WZAHJK is encrypted message.

Vigenere cipher

As we know, in a Caesar cipher, each letter of the alphabet is shifted along some number of places. The Vigenere cipher has several Caesar ciphers in sequence with different shift values.

The question arises: what happens if you take several keys while encrypting/decrypting one message?

Then there will be a system known as the Vigenere'a cipher
(*he French diplomat Blaise Vigenère presented a description of a simple, but resistant (to breaking) cipher before the commission of Henry III in France in 1586*).

- The keys are created based on the sequence of cyclic shifts of the original alphabet.
- A table (Vigenere table or Vigenere square) of size *N* * *N* is created (*N* is the number of characters in the alphabet used).
- These characters can include not only letters, but also, for example, a space or other characters.

•The first line of the table records the entire alphabet used.

Each successive line is obtained from the previous one by a cyclic shift of the last one character to the left.

•Thus, with the capacity ***N*** of the alphabet (English language) equal to 26, it is necessary to execute successively 25 shifts to form the entire table.

•The form of such a table (the Vigenere square) is presented below.

•Encryption occurs using

$$y = x + k \pmod{N}.$$

The Vigenere square

	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
A	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
B	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	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	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

•The choice of a substitution character when encrypting each character of the message (*plaintext*; selected from the *left (before line)* column of the table) is determined by the intersection of this symbol and the corresponding *key symbol* (with the same index) of the key sequence (selected from the *top row (before line)* of the table).

Example. Let the encrypted message has the form: "**BELSTU**". As in Caesar's encryption system with the keyword, in our case it is necessary to select the key. Let the key be the word "**TIR**". Let's write down the message, the key and the ciphertext in the form of a table:

Message	B	E	L	S	T	U
Key	T	I	R	T	I	R
Ciphertext	U	M	C	L	B	L

Even from this simple example, it can be seen that the same letters in the ciphertext (L) correspond to different letters of the message (S, U). This is important for using the **cryptanalysis frequency method**.

Task. To study the features of the cipher machine **Enigma** and the breaking of this cipher.

Cryptanalysis

It is always assumed that the legitimate parties choose a common key at random from the set of all keys.

A.Kerckhoffs' assumption.

The attack knows all details of the enciphering process and deciphering process except for the value of the secret key.

Next we consider the types of attack on encryption systems.

- Ciphertext only attack.

The attacker intercepts a set of ciphertexts.

- Known plaintext attack.

The attacker obtains a set of s plaintexts m1, m2, ..., ms and the corresponding cipher-texts c1, c2, ..., cs. That is, the attacker has no control over the pairs of plaintexts and ciphertexts available to him.

The types of attack

- Chosen plaintext attack.

The attacker chooses a priori a set of s plaintexts $m_1, m_2, \ldots, m_s$ and obtains in someway the corresponding ciphertexts $c_1, c_2, \ldots, c_s$.

- Adaptively chosen plaintext attack.

The attacker chooses a set of plaintexts $m_1, m_2, \ldots, m_s$ interactively as he obtains the corresponding ciphertexts $c_1, c_2, \ldots, c_s$.
That is, the attacker chooses m_1, obtains c_1,then chooses m_2 etc.

- Chosen ciphertext attacks.

These are similar to the chosen plaintext attacks and adaptively chosen plaintext attacks, where the roles of plaintext and ciphertext are swapped.

- Adaptively chosen plaintext attack.

The attacker chooses a set of plaintexts $m_1, m_2, \ldots, m_s$ interactively as he obtains the corresponding ciphertexts $c_1, c_2, \ldots, c_s$.
That is, the attacker chooses m_1, obtains c_1, then chooses m_2 etc.

- Chosen ciphertext attacks.

These are similar to the chosen plaintext attacks and adaptively chosen plaintext attacks, where the roles of plaintext and ciphertext are swapped.

Cryptanalysis of simple substitution ciphers

•All natural languages (e.g., English) contain redundancy. Some letters occur more often than others and also some sequences of two and three letters occur more often than others.

•For (typical) English text the probabilities of occurrence of the 26 letters are given in Table:

a	b	c	d	e	f	g	h	i
.082	.015	.028	.043	.126	.022	.020	.061	.070
j	k	l	m	n	o	p	q	r
.002	.008	.040	.024	.067	.075	.019	.001	.060
s	t	u	v	w	x	y	z	
.063	.091	.028	.010	.023	.001	.020	.001	

•Some common sequences of two letters are “th”, “he”, and “in”, and “the”, “ing”, and “and” are common sequences of three letters.

The idea behind the Vigenere cipher, like all other polyalphabetic ciphers, is to disguise the plaintext letter frequency to interfere with a straightforward application of frequency analysis.

For instance, if P is the most frequent letter in a ciphertext whose plaintext is in English, one might suspect that P corresponds to E since E is the most frequently used letter in English.

By using the Vigenere cipher, E can be enciphered as different ciphertext letters at different points in the message, which defeats simple frequency analysis.

Example.

•For an English text we expect that the two characters that appear the most is “e” and “t”.

•The encryption function is $e_k(m) = am + b \pmod{26}$ where a and b are unknown.

We use Affine cipher.

•With the two guesses for “e” and “t” we get two equations with two unknowns. If our guess is correct, we can solve the equations. We then decrypt the ciphertext and check whether the plaintext is meaningful. If we fail, then we start over with other guesses.

Assume that we have intercepted the following ciphertext:

QMLRLBQMLYVHTMBODVODVOQMLRXYWCQXCBJRLHMXZWCCXHXFLQMVODUBHQ.

1.First count the frequencies of the letters in the ciphertext:

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
0	4	4	3	0	1	0	4	0	1	0	6	6	0	4	0	6	3	0	1	1	4	2	5	2	1

2.It follows that three letters occur the most, each six times. The most frequent two letters in an English text are “e” and “t”. We guess that “L” is the encryption of “e” and replace all occurrences of “L” by an “e”:

-	-	e	-	e	-	-	e	-	-	-	-	-	-	-	-	-	-	-	e	-	-	-	-	-	-	-	-	-	-	-	-	-	e	-	-	-	-	-	-	-	-	-	-	-	e	-	-	-	-	-	-	-	-	-
Q	M	L	R	L	B	Q	M	L	Y	V	H	T	M	B	O	D	V	O	D	V	O	Q	M	L	R	X	Y	W	C	Q	X	C	B	J	R	L	H	M	X	Z	W	C	C	X	H	X	F	L	Q	M	V	O	D	U	B	H	Q

3. Next we guess that “Q” is the encryption of “t” and we get:

t	-	e	-	e	-	t	-	e	-	-	-	-	-	-	-	-	-	-	t	-	e	-	-	-	-	t	-	-	-	-	-	e	-	-	-	-	-	-	-	-	-	-	-	-	e	t	-	-	-	-	-	-	-	t
Q	M	L	R	L	B	Q	M	L	Y	V	H	T	M	B	O	D	V	O	D	V	O	Q	M	L	R	X	Y	W	C	Q	X	C	B	J	R	L	H	M	X	Z	W	C	C	X	H	X	F	L	Q	M	V	O	D	U	B	H	Q

4. Note that we have not been guessing completely at random. We now have a situation where “t” and “e” occur three times with the same third ciphertext letter, “M” between them, thus indicating that “M” is the encryption of “h” (since “the” is a common three-letter sequence).
We arrive at:

t	h	e	-	e	-	t	h	e	-	-	-	-	h	-	-	-	-	-	-	-	t	h	e	-	-	-	-	t	-	-	-	-	e	-	h	-	-	-	-	-	-	-	-	e	t	h	-	-	-	-	-	-	t
Q	M	L	R	L	B	Q	M	L	Y	V	H	T	M	B	O	D	V	O	D	V	O	Q	M	L	R	X	Y	W	C	Q	X	C	B	J	R	L	H	M	X	Z	W	C	C	X	H	X	F	L	Q	M	V	O	D	U	B	H	Q

5. One possible way to continue from above is:

t h e w e a t h e r - - - h a - - - - - - - t h e w - r - - t - - a - w e - h - - - - - - - - - e t h - - - - a - t

Q M L R L B Q M L Y V H T M B O D V O D V O Q M L R X Y W C Q X C B J R L H M X Z W C C X H X F L Q M V O D U B H Q

6. Believe „H” is „s”, „X” is „o”:

t h e w e a t h e r i s c h a n g i n g i n t h e w o r l d t o d a y w e s h o u l d d o s o m e t h i n g f a s t

Q M L R L B Q M L Y V H T M B O D V O D V O Q M L R X Y W C Q X C B J R L H M X Z W C C X H X F L Q M V O D U B H Q

or theweatherischangingintheworldtodayweshoulddosomethingfast

or THE WEATHER IS CHANGING IN THE WORLD TODAY WE SHOULD DO SOMETHING FAST

7. Let believe „V” is „i”:

t h e w e a t h e r i s - h a - - i - - i - t h e w o r - - t o - a - w e s h o - - - - - - o - e t h i - - - a - t

Q M L R L B Q M L Y V H T M B O D V O D V O Q M L R X Y W C Q X C B J R L H M X Z W C C X H X F L Q M V O D U B H Q

.....

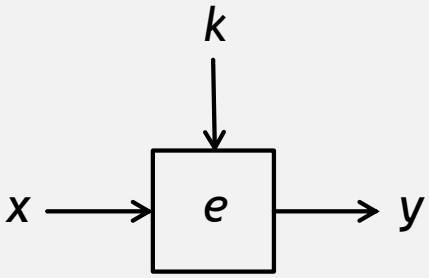
At some point when we have identified enough plaintext characters we can simply derive the rest of them from the context in the text:

t h e w e a t h e r i s - h a - - i - - i - t h e w o r - - t o - a - w e s h o - - - - - - o - e t h i - - - a - t

Q M L R L B Q M L Y V H T M B O D V O D V O Q M L R X Y W C Q X C B J R L H M X Z W C C X H X F L Q M V O D U B H Q

Modern symmetric block ciphers

- These ciphers are called product ciphers.



*A block cipher encrypts block **x** of **n** bits into block **y** of **n** bits using key **k** of **ℓ** bits.*

- Most of the block ciphers in use today are product ciphers. These ciphers often use severallayers of substitutions and transpositions.
- A product cipher is called an iterated cipher if theciphertext is computed by iteratively applying a layer of small ciphers a number of times.
- One iteration is also called a round.
- The two most prominent modern symmetric encryption systems are: the Data Encryption Standard (**DES**) and the Advanced Encryption Standard (**AES**).

System	Year	Block size *(n)*	Key size *(ℓ)*	Rounds
DES	1977	64	56	16
AES	2002	128	128, 192 or 256	10, 12 or 14

Cipher **DES** - Data Encryption Standard

Since **11.21.1976** - the national standard of the United States, since **1981** - the encryption standard for individuals (**ANSI X3.92; DEA** - D.E. Algorithm).

DES can be seen as a special implementation of a Feistel cipher:

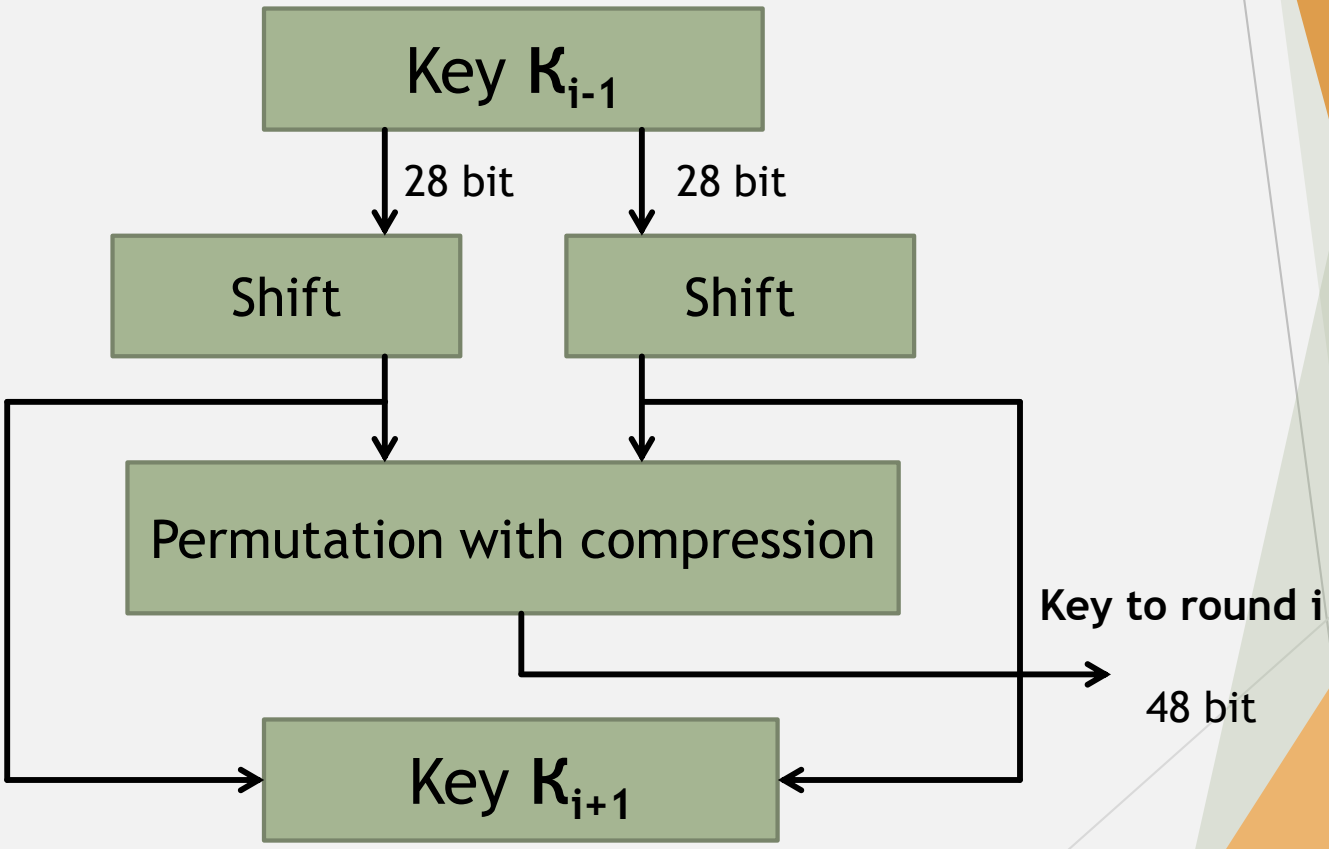
$$L_i = R_{i-1},$$
$$R_i = L_{i-1} + F(R_{i-1}, K_{i-1}),$$

where L_i, R_i – respectively left and right parts of the encrypted data block, K – key

- The 64-bit plaintext, x, is split into two halves, L_0 and R_0 of 32 bits each. The right half is input to a function F together with a ***subkey***. The left half of the plaintext is added bitwise modulo 2 to the output of F.
- These operations constitute one round of DES.
- These operations are now repeated but with a new subkey in every iteration.
- DES runs for 16 rounds, so it needs sixteen subkeys of 48 bits each.

Key to DES

•The input key to DES is only 56 bits, and so the sixteen subkeys are all derived from the 56-bit key in a so-called *key-schedule* (every eighth bit of 64 bits of the key - parity bits - discarded).



Initial Key Preparation Matrix

(compressive permutation before the first round)

57 49 41 33 25 17 09
01 58 50 42 34 26 18 *28 bit*
10 02 59 51 43 35 27
19 11 03 60 52 44 36
63 55 47 39 31 23 15
07 62 54 46 38 30 22 *28 bit*
14 06 61 53 45 3729
21 13 05 28 20 12 04

without 8, 16, 24, 32, 40, 48, 56, 64 bits

Table of shifts for calculating the key

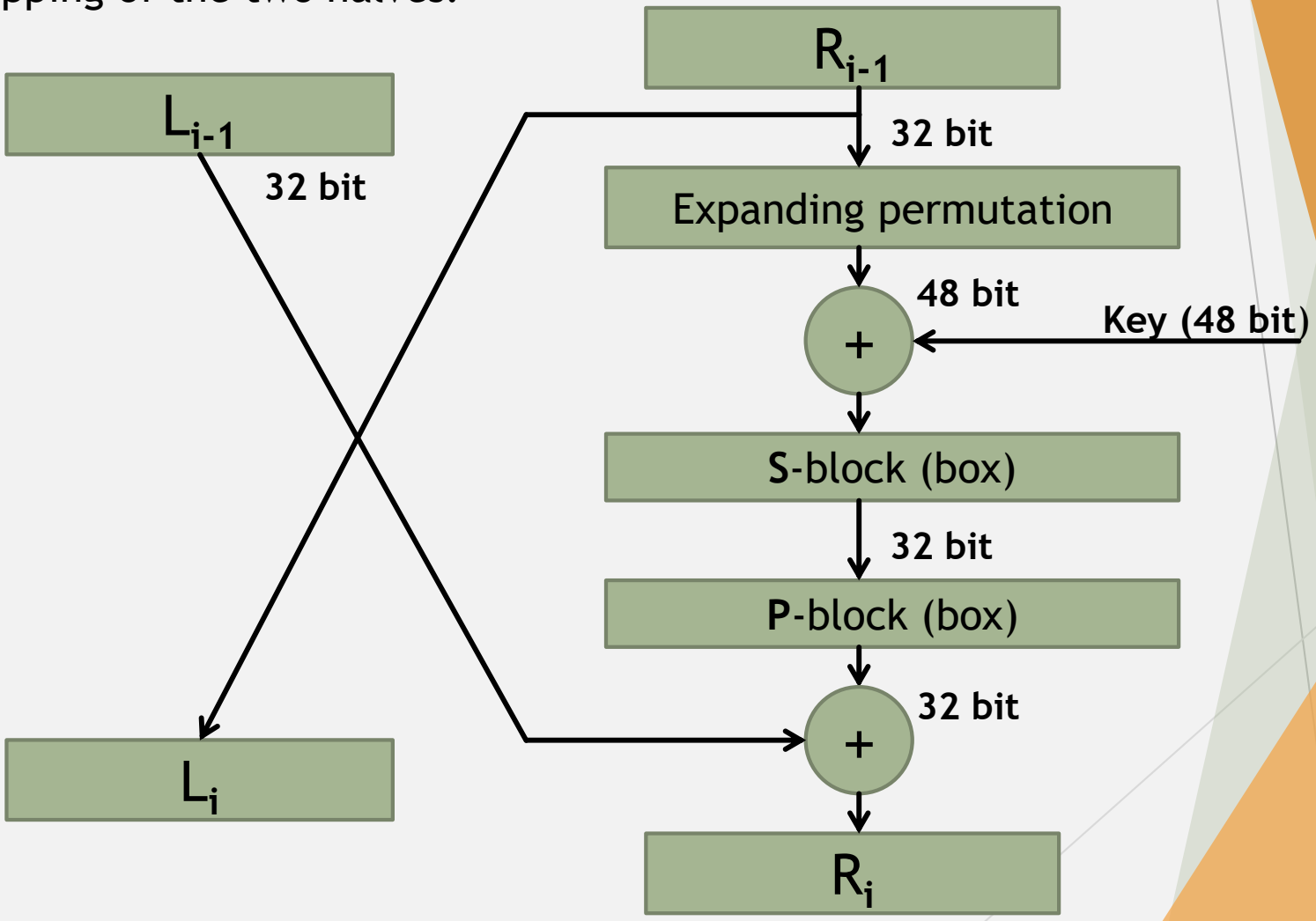
Round number	Shift left (bit)
01	1
02	1
03	2
04	2
05	2
06	2
07	2
08	2
09	1
10	2
11	2
12	2
13	2
14	2
15	2
16	1

Compressive permutation of the key before the first round

14 17 11 24 01 05
03 28 15 06 21 10
23 19 12 04 26 08
16 07 27 20 13 02
41 52 31 37 47 55
30 40 51 45 33 48
44 49 39 56 34 53
46 42 50 36 29 32

A round of DES

The round function of DES consists of the function F plus the swapping of the two halves.



Expanding permutation

The goal of expansion – **avalanche effect**.

- The meaning of such a transformation (extension) is that **the influence of one bit of the message on the overall result increases.**

- The avalanche effect **increases the cryptographic resistance** of the system.

some characters are repeated

1	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
1	13	14	15	16	17
2	17	18	19	20	21
3	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

S-box (Substitution box) and P-box (Permutation box)

• After 48-bit subkey (derived in the key-schedule) is exclusive-ored to the text the result is divided into **eight chunks** of **six bits each** (8 * 6 = 48 bits).

•Each **chunk of six bits is input to an of 8 S-boxes (substitution boxes) which returns four bits** (8 * 4 = 32 bits).

•**S-box** is a **matrix** of **4 rows** and **16 columns**.

•Let $\boldsymbol{a}$ = (a_5, a_4, a_3, a_2, a1, a_0) be a six-bit input:

a_5, a_0 - row address (00, 01, 10, 11),

a_4, a_3, a_2, a_1 - column address (0000, 0001, 0010, ..., 1111);

Example. Consider S-box S1.

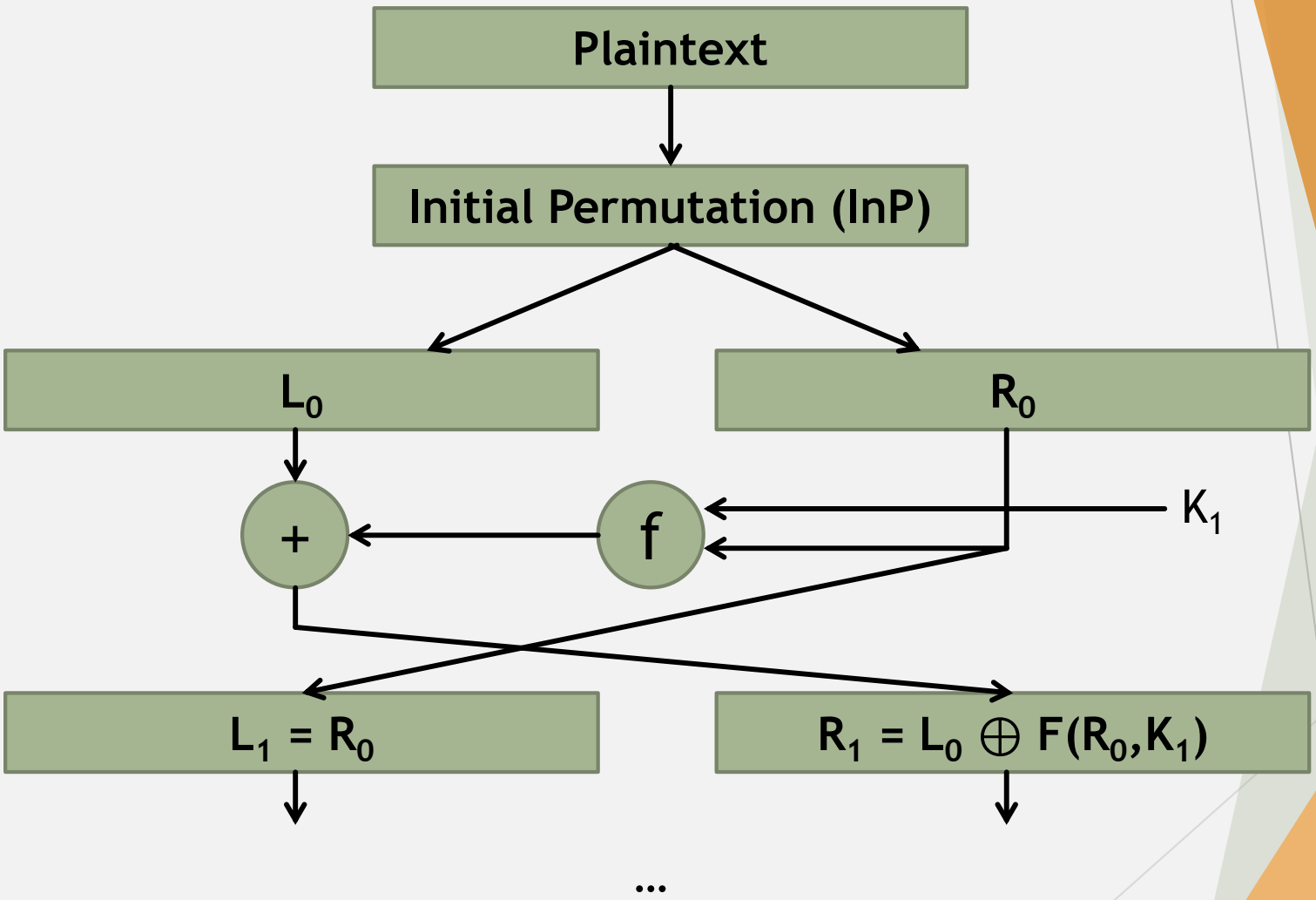
14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

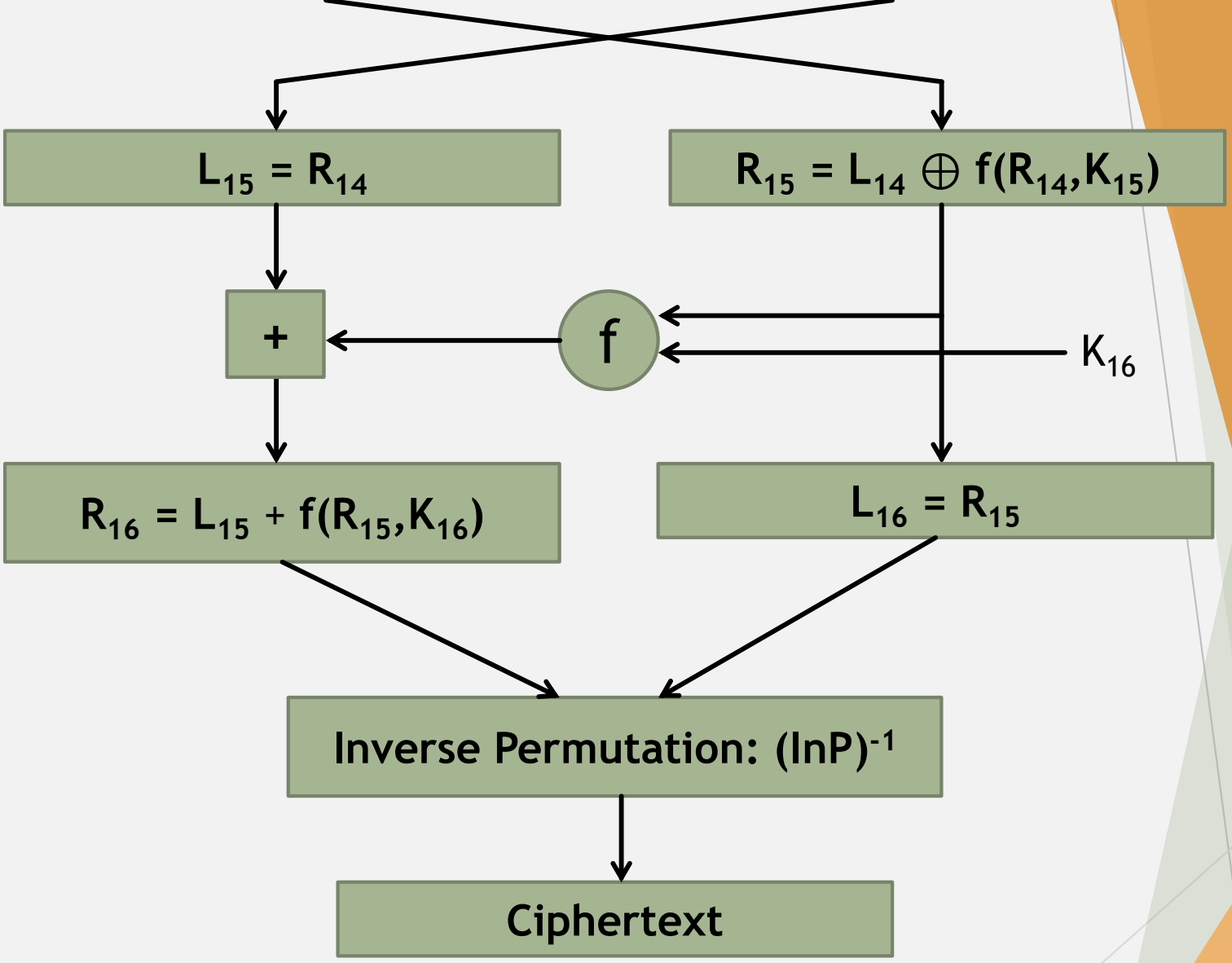
If input six bits are 011110, then row address is 00 (red) and column address is 1111 (green); box **returns four bits: 0111 (digit 7).**

Example of Permutation table:

1	7	20	21	29	12	28	17	16	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9	19	13	30	6	22	11	4	25

Flowchart of DES





Decryption is an identical algorithm, but the keys are in the reverse order.

Features of DES

•DES works with binary numbers: 0 and 1. Each group of 4 bits is converted to a hexadecimal number: binary "0001" equals hexadecimal "1", "1000" - "8", "1111" is equal to "F".

•DES encrypts groups of 64-bit messages (16 hexadecimal numbers).

•DES uses keys (16 hex or 64 bits, each 8th key bit is ignored by the DES algorithm, so the effective key size is 56 bits).

Example.

M = 8787878787878787 (length - 64 bits),

K = 0E329232EA6D0D73,

Encryption DES: **C = 0000000000000000**.

Deciphering with the same key: **M = 8787878787878787**.

Example. **M** = 0123456789ABCDEF (**M** – hex form).

M in binary form(64 bits):

M = 0000 0001 0010 0011 0100 0101 0110 0111 1000 1001 1010 1011 1100 1101 1110 1111

L = 0000 0001 0010 0011 0100 0101 0110 0111

R = 1000 1001 1010 1011 1100 1101 1110 1111

K = 133457799BBCDFF1

or K = 0001 0011 00110100 01010111 01111001 10011011 10111100 11011111 11110001

every eighth bit is ignored by DES according to the table

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

We obtain the 56-bit permuted key:

K+ = 1111000 0110011 0010101 0101111

0101010 1011001 1001111 0001111

Example. **M** = «*Your lips are smoother than vaseline*» - text message with a length of 36 bytes (72 hex numbers).

M_{16} = 596F7572206C6970732061726520736D 6F6F746865722074 68616E2076617365 6C696E650D0A.

72 hex numbers represent a string in English, «0D» - this hex for transferring the carriage to the next line and «0A» - a new line).

The message must be complemented with bytes (the length is a multiple of 8 bytes (or 16 hex numbers or 64 bits), usually „0”: complement **M** with zeros at the end to receive the length of 80 hex numbers:

596F7572206C6970 732061726520736D 6F6F746865722074 68616E2076617365 6C696E650D0A0000.

Use key: 0E329232EA6D0D73.

Encryption: ***C*** = C0999FDDE378D7ED 727DA00BCA5A84EE 47F269A4D6438190 9DD52F78F5358499 828AC9B453E0E653.

$C = c_1, \ldots c_{10}$

$c_1 = C0999FDD$,

$c_2 = E378D7ED$,

…..

$c_{10} = 53E0E653$.

By decryption the added zeros are discarded.

The **Kerberos** (Cerberus) **protocol is based on DES** : it assumes a highly reliable server that stores the original copies of the keys to interact with each user on the network.

Cryptographic resistance of DES

• The key size of DES is only **56 bits, which is too short today**.

•Finding a 56-bit key can be done in relatively short time. This was demonstrated in **1997**. In a US led initiative a group of Internet users joint efforts in a distributed project to break a DES key. After about **90 days** of computation, the key was found.

•Another project was started in January **1998** and found a DES key in **39 days**.

• In **1998 a special-purpose DES key search machine** was built at the cost of US dollars 250,000. This machine found a DES key in less than **three days** .

• In 1999 in a joint effort with almost 100,000 PCs connected to the Internet the key search machine found a DES key in only 22 hours.

3DES

•Introduced in 1998, **3DES**, also known as **Triple DES**, **Triple DEA**, TDEA, or the **Triple Data Encryption Algorithm**.

• Formally Triple DES specifies the use of ***three distinct DES keys***, *for a total key length of **168 bits***.

• Let $\mathbf{E_k(\cdot)}$ and $\mathbf{D_k(\cdot)}$ denote encryption and decryption using **DES**. In **3DES** encryption with three independent keys **k1**, **k2**, and **k3**, the ***E***ncryption and ***D***ecryption operations are:

$$3DES_k(m) = E_{k3}(E_{k2}(E_{k1}(m))),$$
$$3DES^{-1}{}_k\,(c) = D_{k1}(D_{k2}(D_{k3}(m))).$$

•**3DES operates in three steps: Encrypt-Decrypt-Encrypt (EDE).** It works by taking three 56-bit keys (K1, K2 and K3), and encrypting first with K1, decrypting next with K2 and encrypting a last time with K3.

•With 3DES, therefore, **each of the three rounds can be run in either direction: encrypt or decrypt** - using the DES algorithm.

This results in some different possible modes for 3DES:

EDD - Encrypt - Decrypt - Decrypt ,

EDE - Encrypt - Decrypt - Encrypt ,

EED - Encrypt - Encrypt - Decrypt ,

EEE - Encrypt - Encrypt - Encrypt .

•The reason for going through this multiple encryption exercise is to build a **composite cipher that is stronger than Single DES**.

•Because of **meet-in-the-middle attacks**, the **effective cryptographic resistance 3DES is only 112 bits.**

•Two-key 3DES (which is no longer approved for encryption due to its susceptibility to **brute force attacks**) thus has 112 bits of strength (**56 multiplied by two**).

•The **speed** of 3DES is **3 times lower** than that of DES.

A public-key (asymmetric) cryptosystem

•In symmetric cryptosystems the sender and receiver share the same secret information, the secret key.

• In public-key cryptosystem the key is "split into two halves", the private key and the public key.

•A public-key cryptosystem is a function which is easy to compute one way (encryption), but hard to compute the other way (decryption).

•In **part 6** of our course we analyzed two approaches in asymmetric cryptography:

-functions based on the **problem of factoring integers ,**

-functions based on the so-called **discrete logarithm problem.**

The factoring problem.

Given $n = pq$, find p and q, where p and q are different primes.

It is clear that if n is small, it is easy to find the factors p and q.

However, if both p and q are large, different primes, then it is difficult to find the factors, at least when p and q are randomly chosen (or appear to be so).

It is relatively easy to generate large prime numbers, and the function behind the factoring problem is then a candidate for a trapdoor *one-way function*.

The discrete logarithm problem.

Given a, n, and $y = a^x \bmod n$, find x.

For the discrete logarithm problem to be hard, the numbers involved must be relatively large.

Often n is chosen to be a prime and x is chosen at random. The value of a can almost be chosen at random, but there are some bad values that one should avoid, e.g., $a = 1$.

Diffie-Hellman algorithm
(Diffie–Hellman key exchange)

•This idea of a public-key cryptosystem - by W. Diffie and M. Hellman (1976).

•**DH** - is a method of securely exchanging cryptographic keys over a public channel and was one of the first public-key protocols.

• DH Algorithm

Two parties, **A** (Anna) and **B**(Borys).

The implementation of the protocol uses the multiplicative group of integers modulo ***n***, where ***n*** is prime, and a is a primitive root modulo ***n***.

1. **A** and **B** agree to use a modulus $n = 23$ and base $a = 5$ (which is a primitive root **modulo 23**).

2. **A** chooses a **secret integer** $x_A = 4$, then sends **B** number $Y_A = a^{x_A} \bmod n$:

$$Y_A = 5^4 \bmod 23 = 4,$$

B chooses a secret integer $x_B = 3$, then sends A number $Y_B = a^{x_B} \bmod n$:

$$Y_B = 5^3 \bmod 23 = 10.$$

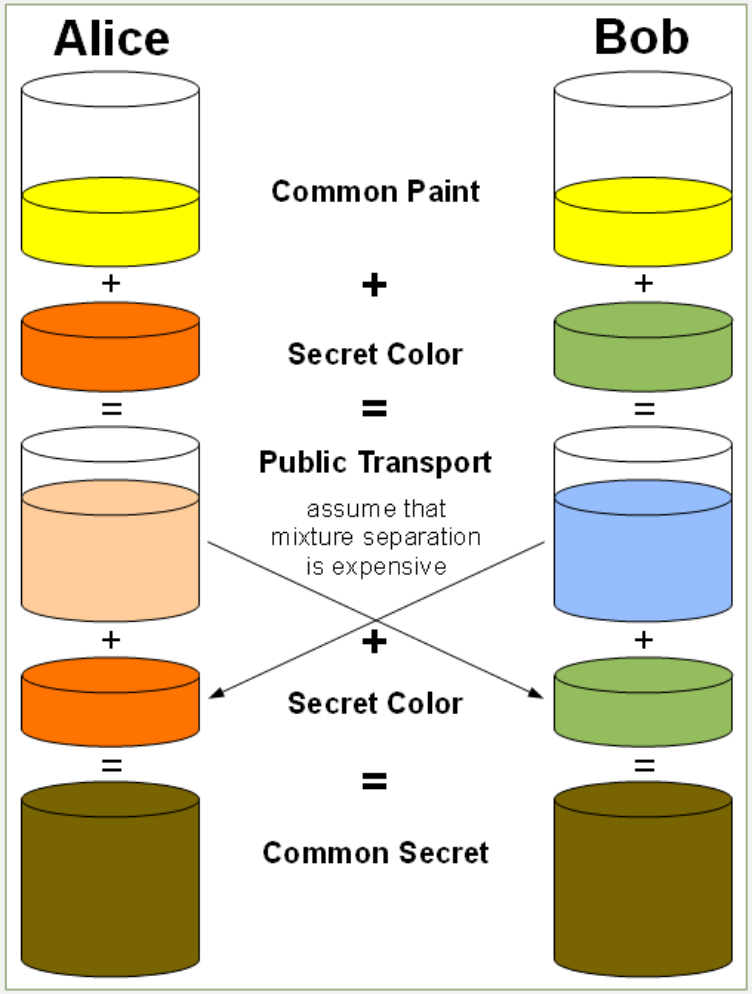
3. **A** computes $K_A = (Y_B)^{x_A} \bmod n$: $K_A = 10^4 \bmod 23 = 18$,

B computes $K_B = (Y_A)^{x_B} \bmod n$: $K_B = 4^3 \bmod 23 = 18$

Anna and Borys now share a secret key (the number 18).

Note, that only x_A, x_B, and ($a^{x_A x_B} \bmod n = a^{x_B x_A} \bmod n$) are kept secret.

Illustration of the idea of Diffie-Hellman key exchange by color



Source: https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange

The RSA cryptosystem

W. Diffie and M. Hellman came up with the idea of a public-key cryptosystem, but they didn't actually find a system for doing it.

In 1977 R.Rivest, A.Shamir, and L.Adleman published such a system, soon named **RSA** after the initials of the designers.

To set up the RSA public-key encryption system, the user (or group of users) does the following.

I. Key creation:

1. Find two big, different primes p and q and compute $n = pq$.

2. Find an integer e, such that $GCD(e, \varphi(n)) = 1$, where $\varphi(n) = (p - 1)(q - 1)$ and where φ (.) is Euler's phi-function.

3. Compute the multiplicative inverse, d, of $e \bmod \varphi(n)$, such that $ed \equiv de \equiv 1 \bmod \varphi(n)$.

Public key: (n, e), where $n = pq$,

Private key: (n, d), where $ed \equiv 1 \bmod \varphi(n)$;

numbers p, q also need to be kept secret.

Example.

1. Choose two random prime numbers
2. :$p = 61$ and $q = 53$;Compute $n = pq$
3. :$n = 61 * 53 = 3233$
4. Compute the totient $\phi(n) = (p-1)(q-1)$
5. : $\phi(n) = (61-1)(53-1) = 3120$
6. Choose $e > 1$ coprime to 3120
7. :$e = 17$
8. Choose d to satisfy $de \equiv 1 \pmod{\phi(n)}$
9. :$d = 2753$
10. : $17 * 2753 = 46801 = 1 + 15 * 3120$.

Public key: *(n, **e**) = (**3233**, **17**)*
Private key: *(n, **d**) = (3233, **2753**)*

II. Using Key to encryption/decryption.

To do this, both the sender and receiver of the message use the **receiver's keys**: the sender encrypts the message (***m***) with a public key (***e***, ***n***), and the receiver decrypts the cryptogram (***c***) with own secret key (***d***, ***n***):

Encryption. The encryption of ***m***:

$$c(m) = m^e \bmod n.$$

Decryption. The decryption of ***c***:

$$m(c) = c^d \bmod n.$$

Example.

To encrypt ***m*** = **123**, we calculate:

$$c = 123^{17} \bmod 3233 = 855;$$

To decrypt ***c*** = **855**, we calculate:

$$m = 855^{2753} \bmod 3233 = 123.$$

Security of RSA

Assume that the attacker has access to the public key (n, e), where $n = pq$ for secret primes p and q.
Here are some of the opportunities to break the system.

I.

If an attacker can find the factors of n, then he can break the system:
if an attacker can find factor n and thereby get p and q, he can find the decryption exponent in exactly the same way as was done in the setup of the system.

II.

If an attacker can find $\varphi(n)$, then he can find factor n and break the system:

recall that: $\varphi(n) = (p - 1)(q - 1)$,

so with $\varphi(n)$ an attacker can set up a system of two equations with two unknowns, since he knows also that $n = pq$.

Expanding the expression in $\varphi(n)$ gives:

$$\varphi(n) = n - p - q + 1 \Rightarrow p = n - \varphi(n) - q + 1.$$

Since p = n/q we get: $n/q = n - \varphi(n) - q + 1$,

which leads to the degree-two equation:

$$0 = q^2 + q\,(\varphi(n) - n - 1) + n.$$

The two solutions to this equation are the primes p and q.

Example. Let n = 7454108611.

Assume that:

$\varphi(n)$ = 7453207872 is known to attacker.

Then he computes:

$$0 = q^2 + q(\varphi(n) - n - 1) + n$$

and

$$0 = q^2 - 900740\, q + 7454108611.$$

The square root of the discriminant of this system is 884034, so solving for q gives **two solutions: 8353 and 892387 .**

III.

If an attacker can find the decryption exponent ***d***, then he can find factor ***n***.

In this case, it is necessary to perform a more complex analysis than in the previous cases.

Details can be found, for example, in:

- Knudsen L. R., Robshaw M., The Block Cipher Companion, *Springer*, 2011.
- Jason Hinek M., Cryptanalysis of RSA and its Variants, *CRC Press*, 2010.
- Abdulaziz Alrasheed and Fatima, *RSA Attacks*, https://www.utc.edu/center-information-security-assurance/pdfs/course-paper-5600-rsa.pdf

For the RSA cryptosystem to be secure, it is recommended (in year 2011) that the modulus in RSA (the integer ***n***) is a number of 1024 bits or more.

Here is an **example** of a **1024-bit RSA modulus:**

135066410865995223349603216278805969938881475605667027524485143851526510604859533833940287150571909441798207282164471551373680419703964191743046496589274256239341020864383202110372958725762358509643110564073501508187510676594629205563685529475213500852879416377328533906109750544334999811150056977236890927563

RSA today

Example. An example of ***n*** = 2048-bit:

25195908475657893494027183240048398571429282126204032027771378360436
62020707595556264018525880784406918290641249515082189298559149176184 5
02808489120072844992687392807287776735971418347270261896375014971824 6
91165077613379859095700097330459748808428401797429100642458691817195 1
18746121515172654632282216869987549182422433637259085141865462043576 7
98423387184774447920739934236584823824281198163815010674810451660377 3
06056201619676256133844143603833904414952634432190114657544454178424 0
20924616515723350778707749817125772467962926386356373289912154831438 1
67899885040445364023527381951378636564391212010397122822120720357

http://en.wikipedia.org/wiki/RSA_Factoring_Challenge

The lengths of symmetric and asymmetric keys with the same degree of resistance to *brute-force attacks*.

Sym key, bit	Asym key, bit
56	384
64	512
80	768
112	1792
128	2304

Exercises.

- Consider an RSA system with $p = 3$ and $q = 11$.

Argue why encryption exponent $e = 7$ is allowed.

1. Find the corresponding decryption exponent d.
2. Find Encrypt $m = 11$.
3. Find Decrypt c.

- Consider an RSA system with the same parameters.

Let M = «Borys».

1. Find the encryption (C) of the message M for m1='B' is number **1**, m2='o' is number **15**… (use eng. alf.).
2. Find the decryption of the ciphertext C.

El-Gamal Public-Key Encryption

- Diffie and Hellman used the **discrete logarithm problem** (DLP) in the mid 1970s.
- Only in 1985 that a public-key cryptosystem based on this problem was presented by ***Taher El-Gamal***.

The El-Gamalpublic-key cryptosystem

Public key: (***n***, ***a***, ***y***),chosen such that DLP (***n***) is difficult, and where ***n*** is an odd prime, ***a*** is such that $a^{\varphi(n)} \bmod n = 1$.

$$y = a^x \bmod n.$$

Private key: x, $1 < x < n$.

Encryption.

For message ***m***:

1. Choose random ***k*** (keep ***k*** secret), ***1< k < n-1***,
2. Calculate (y_1, y_2), where:

$$y_1 = a^k \bmod n,$$
$$y_2 = m\, y^k \bmod n.$$

Decryption.

For $y = (y_1, y_2)$:

$$m = y_2((y_1)^x)^{-1} \bmod n.$$

Exercises.

•Consider the El-Gamal public-key encryption system with the parameters n = 167, a = 5, y = $a^{29} \equiv 55 \bmod n$.

1. Find the encryption of the message m = 10 using k = 79.
2. Find the decryption of the ciphertext (87, 149).

•Consider the El-Gamal public-key encryption system with the parameters n = 167, a = 5, x = 3 and ***M*** = «Borys».

1. Find the encryption (***C***) of the message ***M*** for m1= ‘B’ is number **1**, m2 = ‘o’ is number **15**… (use eng. alf.).
2. Find the decryption of the ciphertext ***C***.

References:

1. J. Pieprzyk, T. Harjono, J. Seberry. Fundamentals of Computer Security, Springer-Verlag, 2003

2. B. Schneier. Applied Cryptography, Second Edition, John Wiley & Sons, 1996

3. Урбанович, П. П. Защита информации методами криптографии, стеганографии и обфускации: учеб.-метод. пособие для студ.. - Минск : БГТУ, 2016. - 220 с.
(URL: http://elib.belstu.by/handle/123456789/23763)

4. Affine Cipher, [Electronic Resource], URL: http://practicalcryptography.com/ ciphers/affine-cipher/

5. Homophonic Substitution Cipher, [Electronic Resource],
URL: http://practicalcryptography.com/ciphers/ substitution-category/homophonic-substitution/

6. Cryptanalysis of the Affine Cipher, [Electronic Resource],
URL: http://practicalcryptography.com/cryptanalysis/stochastic-searching/cryptanalysis-affine-cipher/

7. Cryptography and Cryptanalysis, [Electronic Resource],
URL: https://ocw.mit.edu/courses/find-by-topic/#cat=engineering&subcat=computerscience&spec=cryptography

8.Diffie–Hellman key exchange, [Electronic Resource],
URL:https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange

9. RSA (cryptosystem), [Electronic Resource], URL:https://en.wikipedia.org/wiki/RSA_(cryptosystem)

10. Rivest, R., Shamir A., Adleman L. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, Communications of the ACM, 1978, no.21 (2), pp. 120–126.

11. Knudsen L. R., Robshaw M.. The Block Cipher Companion, Springer, 2011.

12. Jason Hinek M.. Cryptanalysis of RSA and its Variants, CRC Press, 2010.

13. Abdulaziz Alrasheed and Fatima. RSA Attacks, [Electronic Resource], URL:https://www.utc.edu/center-information-security-assurance/pdfs/course-paper-5600-rsa.pdf