

# Podstawy Kryptografii

## Mechanizmy zabezpieczające Kryptografia

**Kryptologia** to nauka o pismach szyfrowanych, sposobach ich tworzenia i rozwiązywania. W jej skład wchodzą:

- **Kryptografia** - zajmuje się zapisywaniem tekstu w sposób utajniony (szyfrowanie i deszyfrowanie)
- **Kryptoanaliza** - zajmuje się zagadnieniami związanymi z łamaniem szyfru, czyli próbą znalezienia klucza szyfru lub odczytaniu tekstu jawnego na podstawie kryptogramu, bez znajomości klucza

## Rodzaje systemów kryptograficznych

Systemy kryptograficzne rozróżniamy używając trzech niezależnych kryteriów:

- Rodzaj operacji stosowanej do przekształcenia tekstu jawnego w tekst zaszyfrowany (podział „klasyczny”)
- Liczba używanych kluczów
- Sposób przetwarzania tekstu jawnego

## Rodzaje SK - Operacja

### ■ Metoda podstawienia

każdy element tekstu jawnego (bit, znak, litera) jest odwzorowywany na inny element.

### ■ Metoda transpozycji

przestawienie kolejności elementów tekstu jawnego.

## Rodzaje SK - Liczba kluczów

■ **Szyfrowanie z jednym kluczem**  
(konwencjonalne, symetryczne, z tajnym kluczem)

■ **Szyfrowanie z dwoma kluczami**  
(asymetryczne, z kluczem jawnym)

## Rodzaje SK - Przetwarzanie

### ■ **Szyfr blokowy**

przetwarza po kolei każdy blok tekstu wejściowego, produkując jeden blok wyjściowy na każdy blok wejściowy

### ■ **Szyfr strumieniowy**

przetwarza elementy wejściowe w sposób ciągły, produkując jednocześnie ciąg wyjściowy

## Wymagania

- Bezstratność
- Odwracalność

## Bezpieczeństwo bezwarunkowe

- Schemat szyfrujący jest **bezwarkunkowo bezpieczny**, jeżeli generowany tekst zaszyfrowany nie zawiera wystarczająco dużo informacji, by jednoznacznie określić odpowiadający mu tekst jawny, niezależnie od ilości dostępnego tekstu zaszyfrowanego

## Bezpieczeństwo obliczeniowe

- Schemat szyfrujący jest **obliczeniowo bezpieczny**, jeżeli koszt złamania szyfru przewyższa wartość informacji zaszyfrowanej oraz/lub czas potrzebny do złamania szyfru przekracza użyteczny „czas życia” informacji.

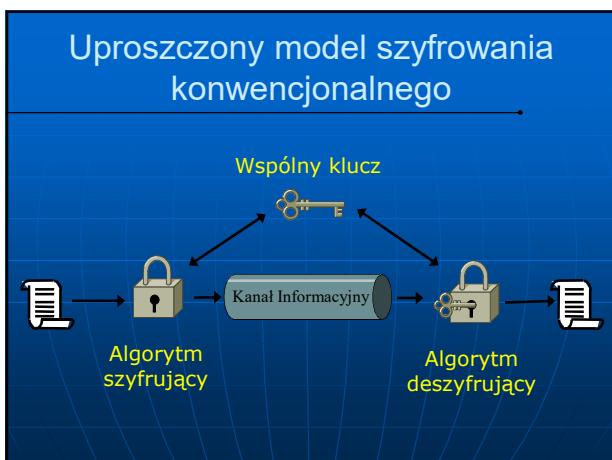
## Systemy kryptograficzne

- **Symetryczne**
  - **Klasyczne** (Cezara, PlayFair, Płot Wieloalfabetowy, Viegenera)
  - **Współczesne** (DES, IDEA, AES, RC4)
- **Asymetryczne**
  - RSA, LUC, El-Gamal

## Szyfrowanie konwencjonalne (symetryczne)

### Szyfrowanie symetryczne

- Obie strony posiadają ten sam klucz
- Algorytmy do szyfrowania i deszyfrowania są bardzo podobne lub identyczne
- Wiadomość zaszyfrowana tajnym kluczem może być odszyfrowana wyłącznie tym samym tajnym kluczem
- Klucz nie może być ujawniony osobie trzeciej (tajny klucz) !!!



### Szyfr Cezara

Polega na zastąpieniu każdej litery alfabetu literą znajdującej się o określona liczbę pozycji dalej

Np. dla przesunięcia o 3 pozycje:

$$C = E(p) = (p+3) \bmod (26)$$

gdzie  $C$  to litera tekstu zaszyfrowanego odpowiadająca literze tekstu jawnego o indeksie  $p$ .

### Szyfr Cezara

Przesunięcie o 3 pozycje

szyfr juliusza cezara  
VCBIU MXOLXVCD FHCDUD

### Szyfr Cezara

- Przesunięcie może mieć wielkość dowolną, ogólna postać algorytmu:
$$C = E(p) = (p+k) \bmod (26), 0 < k < 26$$
- Algorytm deszyfrujący ma postać:
$$p = D(C) = (C-k) \bmod (26)$$
- Dla szyfru Cezara łatwo można przeprowadzić kryptoanalizę metodą brutalną polegającą na wypróbowaniu 25 możliwych kluczy  $k$ .

### Szyfry jednoalfabetowe

- Szyfr Cezara z kluczem 7:

abcdefghijklmnopqrstuvwxyz  
HIJKLMNOPQRSTUVWXYZABCDEG

- Przyporządkowanie 'dowolne':

abcdefghijklmnopqrstuvwxyz  
NJTRUHSLWGBKVIZOEYDZMAPCQF

**26! = 4 \* 10<sup>26</sup> możliwych kluczy**

## Kryptoanaliza szyfrów jednoalfabetowe

- Metoda brutalna możliwa, ale długotrwała
  - Zastosowanie metody brutalnej wymaga algorytmów analizy treści
- Możliwa kryptoanaliza statystyczna na podstawie regularności języka - częstości występowania liter (dwuznaków) w języku

## Częstość względna występowania liter w tekście angielskim

Litera	Częstość (%)	Litera	Częstość (%)	Litera	Częstość (%)
E	12,75	L	3,75	W	1,50
T	9,25	H	3,50	V	1,50
R	8,50	C	3,50	B	1,25
N	7,75	F	3,00	K	0,50
I	7,75	U	3,00	X	0,50
O	7,50	M	2,75	Q	0,50
A	7,25	P	2,75	Z	0,25
D	4,25	G	2,00		

## Szyfr Playfair

- Jednostką (blokiem) są dwuznaki tekstu jawnego. Szyfr tłumaczy je na dwuznaki zaszyfrowane.
- Algorytm używa matrycy 5x5, zbudowanej przy użyciu słowa kluczowego.

## Matryca Playfair

Słowo kluczowe: **MONARCHY**

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

## Szyfrowanie Playfair

### Krok 1

Powtarzające się litery tekstu jawnego oddzielić literą wypełniającą np. X.  
Czyli *flood* zamienia się na *floxod*.

## Szyfrowanie Playfair – zasada 1

Gdy litery pary liter tekstu jawnego występują w tym samym wierszu matrycy, następuje się każdą z nich literą o jedną pozycję w prawo

ar	->	RM		
ps	->	QT		
M	O	N	A	R
C	H	Y	B	D
E	F	G	I/ J	K
L	P	Q	S	T
U	V	W	X	Z

## Szyfrowanie Playfair – zasada 2

Gdy litery pary liter tekstu jawnego występują w tej samej kolumnie matrycy, zastępuje się każdą z nich literą o jedną pozycję w dół

**mu** -> CM  
**yq** -> GW

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

## Szyfrowanie Playfair – zasada 3

W innym wypadku, każdą literę zastępuje się literą leżącą w tym samym, co ona rzędzie, lecz w kolumnie, w której leży druga litera

**hs** -> BP  
**ea** -> IM (JM)

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

## Szyfrowanie Playfair

the Playfair is good  
th eP la yf ai ri sg ox od  
PD FL MS HG BS AK IQ AV RH

PDFILM**SHGBSAKIQAVRH**

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

## Szyfry wieloalfabetowe

- Stosowanie różnych podstawień jednoalfabetowych podczas szyfrowania jednego komunikatu.
- abcdefghijklmnoprstuvwxyz  
HIJKLMNOPQRSTUVWXYZABCDEFC**
- abcdefghijklmnoprstuvwxyz  
NJTRUHSLWGBKVIZOEYDZMAPCQF**
- szyfry  
ZFFHYQ**

## Szyfr Vigenere'a

- Stosuje się 26 alfabetów - szyfrów Cezara z przesunięciami od 0 do 25.
- Stosuje się cyklicznie powtarzany klucz.
- Litera klucza wyznacza wiersz, a litera tekstu jawnego kolumnę w tablicy.

## Szyfr Vigenere'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	
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	
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	
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	
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	
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	
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	
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	
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	
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	
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	

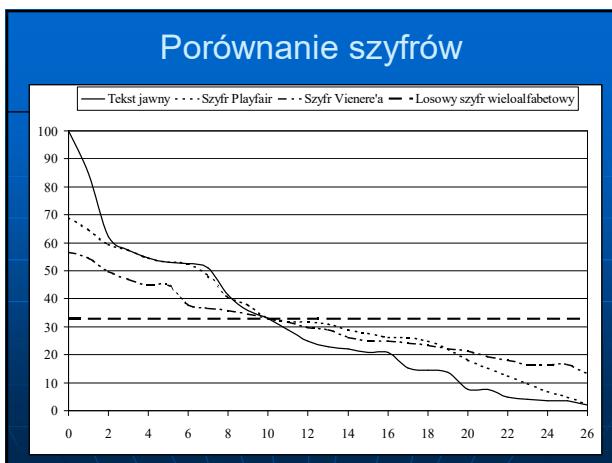
### Szyfr Vigenere'a

	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>a</b>	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</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	
<b>c</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	
<b>d</b>	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	
<b>e</b>	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	

*deceptive*deceptive*deceptive*deceptive  
*wearediscoverededsaveyourself*  
ZICVTWQNGRZGVTWAVZHCQYGLMGJ

### Losowe szyfry wieloalfabetowe

- Klucz jednorazowy, losowy o długości równej długości komunikatu
- Algorytm nie do złamania !!!



### Szyfry transpozycyjne

- permutacja znaków tekstu jawnego
- technika płotu: tekst jawny zapisuje się jako ciąg kolumn, a odczytuje jako ciąg wierszy

### Technika transpozycyjna – ‘plot’

Tekst jawny zapisuje się jako ciąg kolumn, a odczytuje jako ciąg wierszy

**SZYFROWANIE TECHNIKĄ PŁOTU**

S	F	W	I	E	N	O	
Z	R	A	E	C	I	P	T
Y	O	N	T	H	K	Ł	U

**SFWIENAOZRAECEPTYONTHKLU**

### Wielokrotne szyfrowanie

- Można zastosować kolejno wiele podstawień i transpozycji

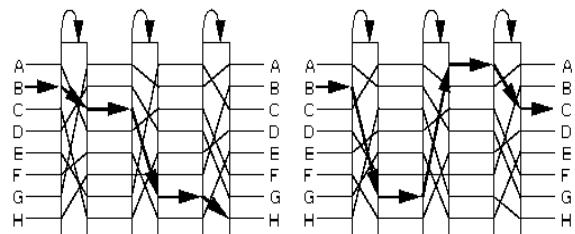
## Maszyny rotorowe

- Enigma, Purple
- Składają się z ruchomych cylindrów, przez które przepływają impulsy elektryczne. Każdy cylinder składa się z pewnej liczby styków wejściowych i wyjściowych, połączonych między sobą wewnętrznymi przewodami.
- Pojedynczy cylinder realizuje szyfr wieloalfabetowy (np. 26)

## Maszyny rotorowe

- Zastosowanie wielu cylindrów zwiększa liczbę wykorzystywanych alfabetów:
  - 3 →  $26^3 = 17576$
  - 4 → 456976
  - 5 → 11881376
- Zasada działania maszyn rotorowych dała podstawy do stworzenia szyfru DES
- Ciekawy symulator maszyny rotorowej:  
<http://enigmaco.de/enigma/enigma.swf>

## Maszyny rotorowe



## Dlaczego długość klucza jest ważna ?

- Od długości klucza zależy liczba różnych kluczy  
Klucz 10-bitowy:  $2^{10} = 1024$  różnych kluczy  
Klucz 20-bitowy:  $2^{20} = 1\ 048\ 576$  kluczy
- Najprostsza (i zawsze skuteczna) metoda złamania szyfru polega na sprawdzeniu wszystkich możliwych kluczy (tzw. **metoda brutalna**)

## Kryptografia współczesna

Systemy symetryczne

## Algorytmy produktowe

- Większość współczesnych algorytmów symetrycznych to **algorytmy produktowe**
- Wykonywane są proste, stosunkowo mało bezpieczne kroki szyfrujące zwane rundami.
- Dzięki stosowaniu wielu rund bezpieczeństwo algorytmu znacząco rośnie.

## Sieci Feistela

- Każdy blok dzielony jest na dwie równe połówki L (lewa połówka) oraz R (prawa połówka)
- Ponieważ algorytm składa się wielu rund, postać każdej połówki w  $i$ -tej rundzie oznaczamy  $L_i$  oraz  $R_i$

## Sieci Feistela

- Połówki do następnej rundy są wyznaczane następująco:

$$L_{i+1} = R_i$$

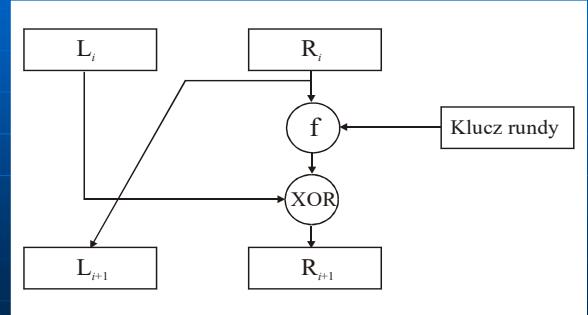
$$R_{i+1} = L_i \text{ XOR } f_{S,i}(R_i)$$

- Funkcja  $f_{S,i}$  wprowadza klucz rundy

## Sieci Feistela – klucz rundy

- Obliczany na podstawie klucza głównego.
- Zazwyczaj każda runda ma inny klucz rundy.
- Funkcje  $f_{S,i}$  nie muszą być odwracalne.

## Sieci Feistela



## Funkcja XOR

X	Y	X xor Y
1	1	0
1	0	1
0	1	1
0	0	0

$$X \text{ xor } Y \text{ xor } Y = X \text{ xor } (Y \text{ xor } Y) = \\ X \text{ xor } 0 = X$$

## Deszyfrowanie w sieciach Feistela

- Znajomość funkcji  $f_{S,i}$  dla wszystkich rund  $i$  umożliwia łatwe deszyfrowanie:

$$\begin{aligned} L_i &= \\ L_i \text{ XOR } f_{S,i}(R_i) \text{ XOR } f_{S,i}(R_i) &= \\ (L_i \text{ XOR } f_{S,i}(R_i)) \text{ XOR } f_{S,i}(R_i) &= \\ R_{i+1} \text{ XOR } f_{S,i}(R_i) & \end{aligned}$$

## Deszyfrowanie w sieciach Feistela

$$R_{n-1} = L_n$$

$$L_{n-1} = R_n \text{ XOR } f_{S,n-1}(R_{n-1})$$

...

$$R_0 = L_1$$

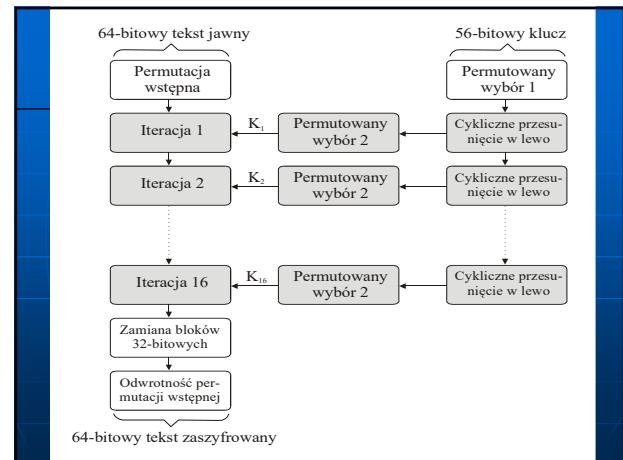
$$L_0 = R_1 \text{ XOR } f_{S,0}(R_0)$$

## DES

### Data Encryption Standard

## DES

- Zaakceptowany w 1977 r. Przez Narodowe Biuro Standardów USA.
- Dane są szyfrowane w 64-bitowych blokach z zastosowaniem 56 bitowego klucza.
- Algorytm w serii etapów przetwarza 64-bitowe dane wejściowe na 64-bitowy wynik.
- W celu odwrócenia operacji stosuje się te same etapy i ten sam klucz.

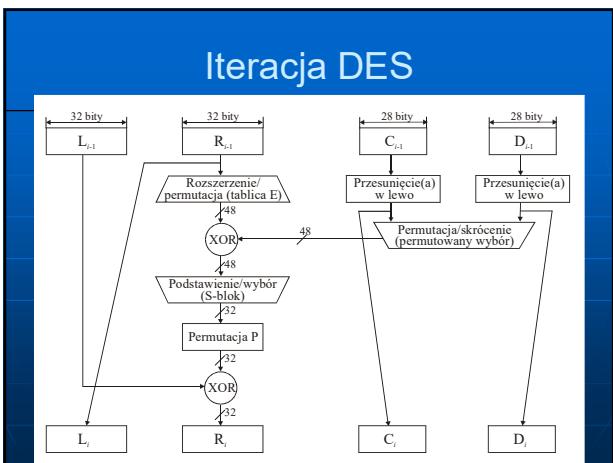


## DES - Permutacja wstępna

- Permutacja wstępna IP (*Initial Permutation*) i jej odwrotność  $IP^{-1}$
- Obie funkcje są odwrotne względem siebie.
- Jeżeli szyfrujemy blok  $M$  to tekst po permutacji wygląda następująco  $X=IP(M)$ .
- Po permutacji odwrotnej otrzymujemy  $Y=IP^{-1}(X)=IP^{-1}(IP(M))=M$ .

## DES - Permutacja wstępna

Bit wyjściowy	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Z bitu wejśc.	58	50	42	34	26	18	10	2	60	52	44	36	28	20	12	4
Bit wyjściowy	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Z bitu wejśc.	62	54	46	38	30	22	14	6	64	56	48	40	32	24	16	8
Bit wyjściowy	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
Z bitu wejśc.	57	49	41	33	25	17	9	1	59	51	43	35	27	19	11	3
Bit wyjściowy	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
Z bitu wejśc.	61	53	45	37	29	21	13	5	63	55	47	39	31	23	15	7



### Permutacja rozszerzająca

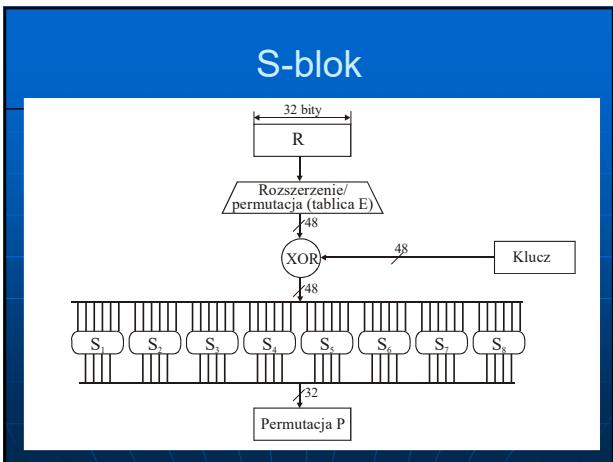
Bit wyjściowy	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Z bitu wejsc.	32	1	2	3	4	5	4	5	6	7	8	9	8	9	10	11

Bit wyjściowy	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Z bitu wejsc.	12	13	12	13	14	15	16	17	16	17	18	19	20	21	20	21

Bit wyjściowy	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
Z bitu wejsc.	22	23	24	25	24	25	26	27	28	29	28	29	30	31	32	1



### S-blok

- Pierwszy i ostatni bit wejścia S-bloku wskazuje odpowiedni wiersz tabeli S-bloku. Środkowe 4 bity określają kolumnę.

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

- wejście **011011** (wiersz **01** (**1**)), kolumna **1101** (**13**)) daje na wyjściu **5** (**0101**).

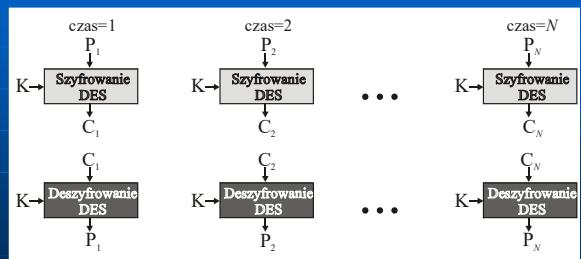
- ### Efekt lawinowy
- Intensywniejsze niż dla dyfuzji rozmarzanie tekstu jawnego w tekście zaszyfrowanym.
  - Dla zmiany pojedynczego bitu tekstu jawnego lub klucza, każdy bit tekstu zaszyfrowanego powinien zmienić swoją wartość z prawdopodobieństwem równym dokładnie 50%.
  - Kryptoanaliza różnicowa wykorzystuje nawet niewielkie odstępstwo od powyższej zasady.

- ### Efekt lawinowy w DES
- Szyfrując dwa bloki różniące się jednym bitem przeciętnie już w czwartej/piątej iteracji otrzymujemy różnicę na ponad 32 bitach szyfrogramu.

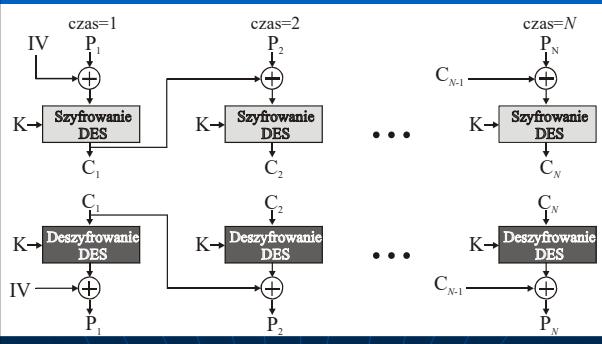
## Tryby pracy DES

- Elektronicznej książki kodowej ECB (*electronic codebook*).
  - Wiązania bloków zaszyfrowanych CBC (*cipher block chaining*).
- Tryby strumieniowe:
- Ze sprzężeniem zwrotnym - CFB (*cipher feedback*).
  - Sprzężenia zwrotnego wyjściowego - OFB (*output feedback*).
- Podobny do CFB, bardziej odporny na błędy w transmisji, mniej odporny na złamanie.

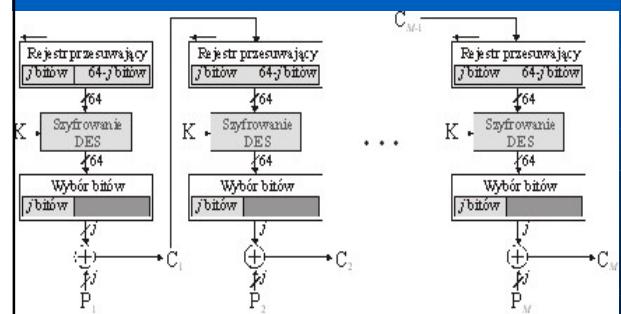
## Electronic Codebook



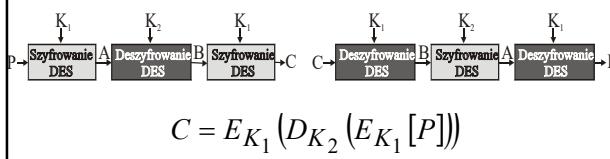
## Cipher Block Chaining



## Cipher Feedback



## Potrójny DES z dwoma kluczami



## IDEA

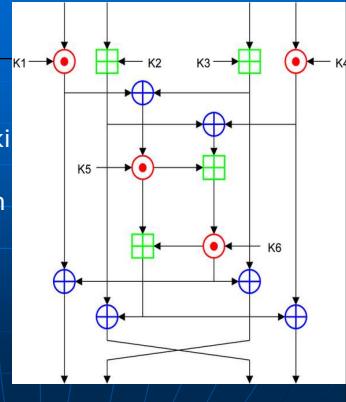
- IDEA to algorytm szyfrowania blokowego stworzony przez Xuejia Lai i Jamesa Massey w 1990 roku.
- Jest uznawany za następcę DES

## Założenia projektowe algorytmu IDEA

- Szyfrowania danych w blokach po 64 bity.
- 128-bitowy klucz.
- Zastosowanie prostych operacji arytmetycznych.
- Podobieństwo między szyfrowaniem i deszyfrowaniem.
- Regularna struktura ułatwiająca realizację w technice VLSI

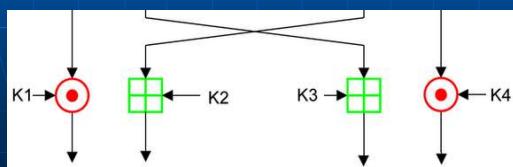
## Iteracja Idea

- 64-bitowy blok dzielony jest na 16-bitowe podbloki
- W każdej iteracji wykorzystywanych jest 6 podkluczy wygenerowanych na podstawie 128 bitowego klucza.



## Algorytm Idea

- Proces szyfrowania składa się z 9 etapów: 8 iteracji (każda z zastosowaniem 6 podkluczy) oraz przekształcenia końcowego (4 podklucze).



## IDEA - Generowanie podkluczy

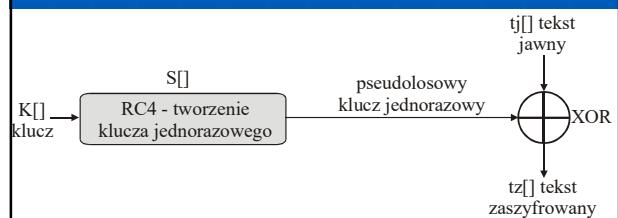
- Podklucze o długości 16 bitów są generowane na podstawie głównego 128 bitowego klucza. Pierwsze 8 podkluczy jest generowany bezpośrednio z klucza. Następnie cyklicznie przesuwamy klucz w lewo o 25 bitów i generujemy kolejne 8 podkluczy.

K (128 bitów)  
 ← K1 → K2 → K3 → K4 → K5 → K6 → K7 → K8 →  
 K15 → K16 → K9 → K10 → K11 → K12 → K13 → K14 →  
 → K22 → K23 → K24 → K17 → K18 → K19 → K20 → K21  
 K28 → K29 → K30 → K31 → K32 → K25 → K26 → K27 →  
 → K35 → K36 → K37 → K38 → K39 → K40 → K33 → K34  
 K41 → K42 → K43 → K44 → K45 → K46 → K47 → K48 →  
 ← K49 → K50 → K51 → K52 →

## RC4

- opracowany w 1987 przez Rona Rivesta
- długość klucza: 40 lub 128 bitów
- algorytm strumieniowy: w zależności od długości klucza tworzony jest ciąg bajtów, który wykorzystuje się jako klucz jednorazowy
- tekst zaszyfrowany jest rezultatem operacji XOR tekstu jawnego i stworzonego jednorazowego klucza

## RC4 - szyfrowanie



## Tworzenie pseudolosowego klucza

- Klucz RC4 (dowolnej długości) służy do wygenerowania klucza wewnętrznego
- Klucz wewnętrzny to dwa bajty oraz permutacja liczb 0, ..., 255
- Liczba możliwych kluczy wewnętrznych:  $256! * 256^2 \approx 2^{1700}$
- Na podstawie klucza wewnętrznego generowany jest pseudolosowy klucz jednorazowy

## Tworzenie pseudolosowego klucza

- Generowanie klucza wewnętrznego  
Dane: klucz o długości m bajtów:  $S_0, \dots, S_{m-1}$   
 $i=0, j=0, P_k=k \ (k=0, \dots, 255)$   

```
for (a=0; a<256; a++) {
    j=(j+P_i+S_i) mod 256
    zamień (P_i, P_j)
    i=(i+1) mod m
}
```

Uzyskujemy wartości i, j oraz permutację P

## Tworzenie pseudolosowego klucza

- Tworzenie pojedynczego bajtu klucza K  
 $i=(i+1) \text{ mod } 256$   
 $j=(j+P_i) \text{ mod } 256$   
 $\text{zamień}(P_i, P_j)$   
 $t=(P_i+P_j) \text{ mod } 256$   
 $K = P_t$
- Szyfrowanie  
 $C[i] = P[i] \oplus K[i]$
- Deszyfrowanie  
 $P[i] = C[i] \oplus K[i]$

## Konkurs AES

- W 1997 roku agencja NIST (*National Institute of Standards and Technology*) rozpisała konkurs na nowy standard szyfrowania, który miał otrzymać nazwę AES (*Advanced Encryption Standard*)

## Finaliści AES

- **Rijndael** - opracowany przez naukowców belgijskich dr Joan Daemen oraz dr Vincent Rijmen
- RC6
- Mars
- Serpent
- Twofish

## AES (Rijndael)

- Algorytm symetryczny blokowy – bloki o długości 128, 192 lub 256 bitów
- Klucz o długości 128, 192 lub 256 bitów
- Został zatwierdzony jako następca algorytmu DES
- Obecnie jest to zalecany algorytm symetryczny

## AES (Rijndael)

Proces szyfrowania podlega iteracjom, przy czym rozróżnia się:

- rundę wstępna,
- pewną liczbę rund standardowych (ich liczba zależy od długości klucza i bloku i wynosi 10, 12 lub 14), z których każda posiada 4 transformacje,
- rundę końcową.

## Zalety AES

- Spełnia 3 główne założenia postawione przez twórców algorytmu: odporność na wszystkie znane ataki, szybkość pracy i zwartość kodu na różnych platformach, łatwość implementacji.
- Nie jest chroniony żadnymi zastrzeżeniami patentowymi, nie wymaga opłat licencyjnych.
- Aktualny stan wiedzy w zakresie kryptoanalizy nie pozwala na skuteczny atak.

## Algorytm Rijndael

- **Runda** (*round*) to odpowiednik standardowego etapu obliczeń mającym jako parametr klucz rundy (*Round Key*). Runda jest superpozycją, co najmniej 2 bijekcji.
- **Stan** (*state*) – bajtowa macierz prostokątna stanowiąca wynik pośredni kolejnych obliczeń podczas realizacji algorytmu. Macierz Stanu ma 4 wiersze i  $N_b$  kolumn ( $N_b$  to długość bloku podzielona przez 32),  $N_b=4$ , 6 lub 8.

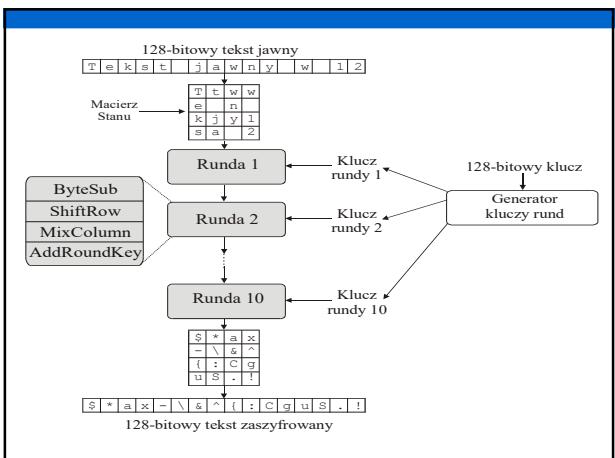
## Rozmiar klucza i bloku AES

- Klucz jest reprezentowany jako macierz o 4 wierszach i  $N_k$  kolumnach ( $N_k$  to długość klucza podzielona przez 32),  $N_k=4$ , 6 lub 8.
- Długość klucza i bloku czyli  $N_k$  i  $N_b$  możemy zmieniać niezależnie. Liczba rund  $N_r$  stosowana w algorytmie zależy od  $N_b$  i  $N_k$ .

## Liczba rund AES

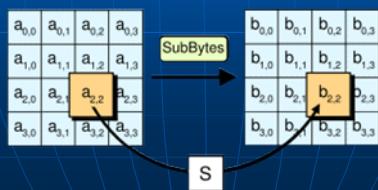
N <sub>b</sub>	N <sub>k</sub>	N <sub>r</sub>	N <sub>b</sub>	N <sub>k</sub>	N <sub>r</sub>	N <sub>b</sub>	N <sub>k</sub>	N <sub>r</sub>
4	4	10	4	6	12	4	8	14
6	4	12	6	6	12	6	8	14
8	4	14	8	6	14	8	8	14

- Na początku przekształceń **Stan** to tekst jawnny.
- W wyniku kolejnych przekształceń w ramach rundy i kolejnych rund Stan ulega zmianie i po ostatniej rundzie zawiera tekst zaszyfrowany.



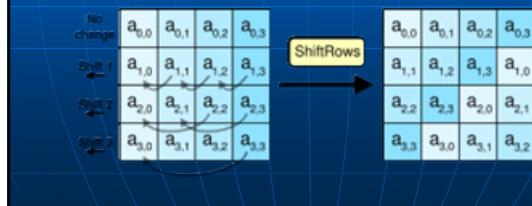
## Przekształcenie ByteSub

- Każdy bajt przechodzi transformację, którą ze względów historycznych nazwano S-Boxem. W fazie tej wykonuje się jedynie operacje na bajtach, a zatem jest to łatwe nawet w procesorach 8-bitowych.



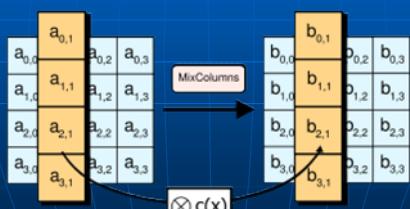
## Przekształcenie ShiftRow

- Przesuwa cyklicznie kolejne wiersze macierzy o odpowiednią liczbę pozycji. Wartości przesunięcia zależą od wielkości bloku i klucza.



## Przekształcenie MixColumn

- Przekształca (miesza) wartości w kolumnie.

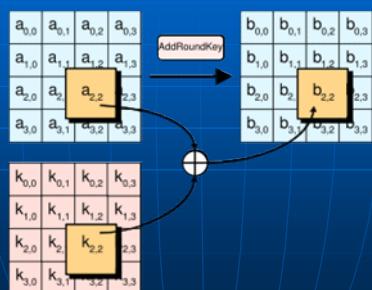


## Przekształcenie MixColumn

- Kolumny Stanu są traktowane jako wielomiany w  $GF(2^8)$  i są mnożone przez  $c(x) = '03'x^3 + '01'x^2 + '01'x + '02'$  modulo  $(x^4 + 1)$ . Można to zapisać jako mnożenie macierzy, gdzie  $b(x) = c(x) \otimes a(x)$ :

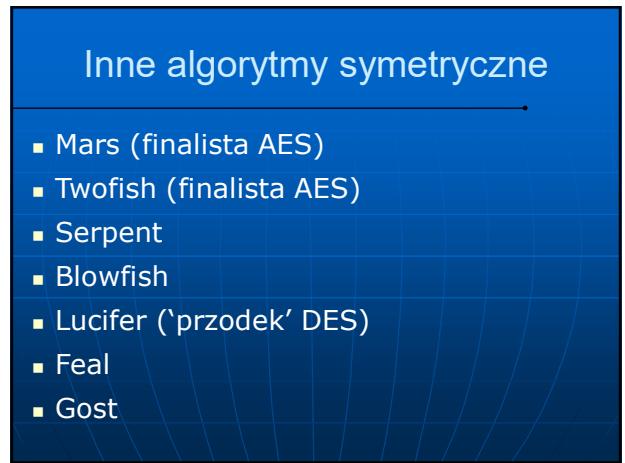
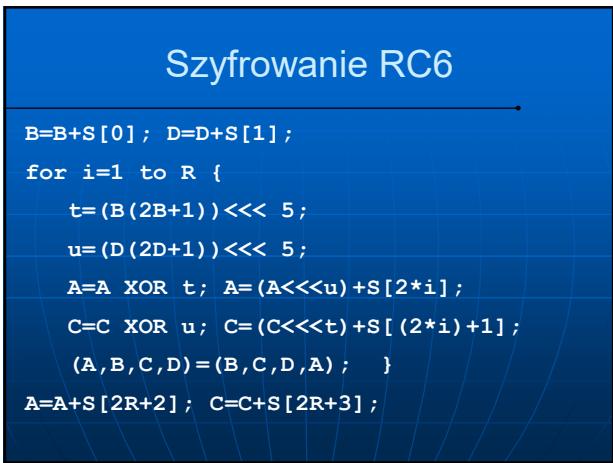
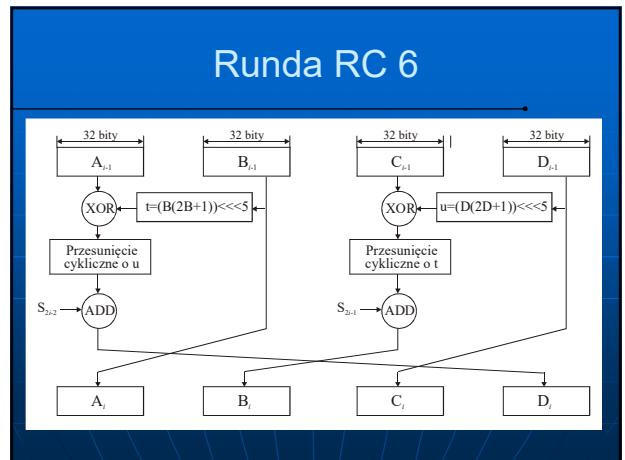
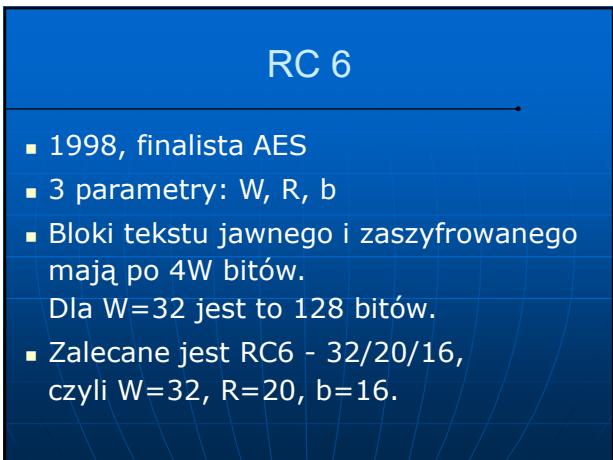
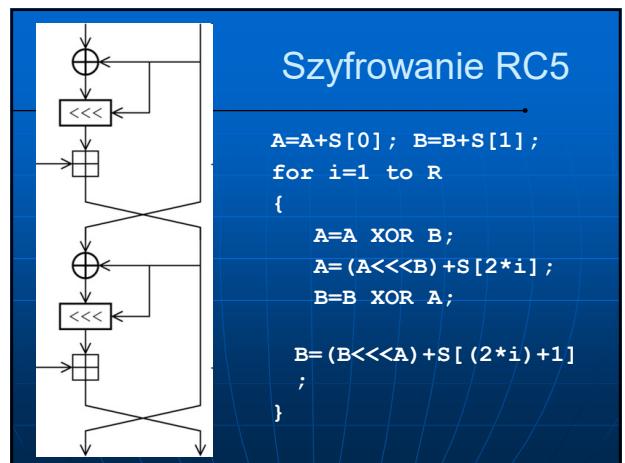
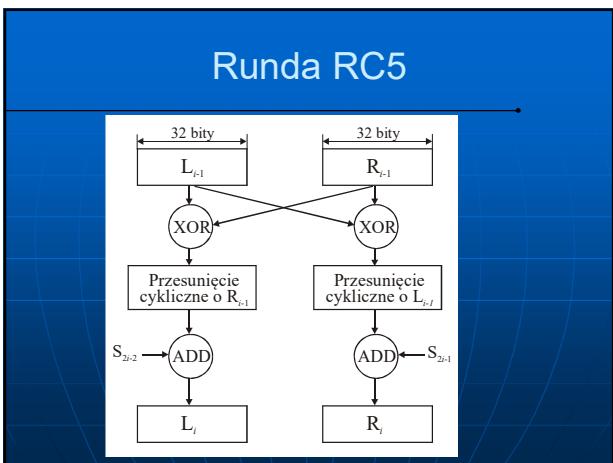
$$\begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \cdot \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}$$

## Dodawanie klucza rundy



## RC5

- Symetryczny, blokowy, 1994
- 3 parametry:
  - długość słowa (W). Bloki tekstu jawnego i zaszyfrowanego mają po  $2W$  bitów.  
Dla  $W=32$  blok ma 64 bity.
  - liczba rund (R).
  - długość klucza (b bajtów,  $b=0, 1, \dots, 255$ ).
- Zalecane jest RC5-32/12/16, czyli  $W=32$ ,  $R=12$ ,  $b=16$ .



## Szyfrowanie symetryczne

Które z **usług ochrony** są realizowane przez system szyfrujący symetryczny?

- Poufność TAK
- Integralność TAK
- Uwierzytelnianie TAK
- Niezaprzeczalność NIE

## Współczesne algorytmy symetryczne

- W większości są to algorytmy blokowe, zorientowane bitowo
- Klucze są traktowane jako ciągi bitów
- Większość polega na wykonywaniu wielu prostych przekształceń: podstawień i transpozycji
- Składają się z pewnej liczby identycznych rund (iteracji)

## Szyfrowanie symetryczne - algorytmy

Nazwa	Rodzaj	Rozm. bloku	Dług. klucza
DES	Blokowy	64	56
3DES	Blokowy	64	112
IDEA	Blokowy	64	128
AES	Blokowy	128, 192, 256	128, 192, 256
Blowfish	Blokowy	64	Do 448
RC5, RC6	Blokowy	32, 64, 128	Do 2048
RC4	Strumieniowy	-	Permut. 256 + 16 bitów

## Kryptoanaliza

### Algorytmy symetryczne

## Metody kryptoanalizy

- Atak z tekstem zaszyfrowanym (*ciphertext-only*)
- Atak ze znanyim tekstem jawnym (*known plaintext*)
- Atak z wybranym tekstem jawnym (*chosen plaintext*)
- Atak z wybranym tekstem zaszyfrowanym (*chosen ciphertext*)

## Kryptoanaliza różnicowa

- Biham, Shamir, 1990 rok
- Atak z wybranym tekstem jawnym
  1. Wybieramy 2 różniące się (na ogólnie minimalnie) teksty jawne
  2. Porównujemy kryptogramy

## Kryptoanaliza liniowa

- Mitsuru Matsui, 1993 rok
- Działanie urządzenia szyfrującego opisywane liniową aproksymacją
- DES złamany na kilku komputerach w 50 dni