Kryptografia — systemy z kluczem tajnym

Krótkie vademecum (słabego) szyfranta

Podstawowe pojęcia:

- *tekst jawny (otwarty)* $\Longrightarrow$ *tekst zaszyfrowany (kryptogram)*
- *alfabet* obu tekstów (zwykle różny)
- *jednostki tekstu*: na przykład *pojedyncza litera*, *digram*, *trigram*, ..., *polygram*.
- *Funkcja (przekształcenie) szyfrująca (-e)* f każdej jednostce tekstu otwartego przyporządkowuje jednostkę tekstu zakodowanego
- *Funkcja (przekształcenie) deszyfrująca (-e)* f^{-1} z tekstu zakodowanego odtwarza tekstu otwarty

$$\mathcal{P} \overset{f}{\Longrightarrow} \mathcal{C} \overset{f^{-1}}{\Longrightarrow} \mathcal{P}.$$

System kryptograficzny

Podstawowe pojęcia, przykłady:

- *pojedyncze* jednostki tekstu jawnego (i zaszyfrowanego) – a więc litery „przekładamy” na język cyfr.
- Digramy – litery A-Z i odstęp (27 znaków). System pozycyjny o podstawie 27.
- Jednostki tekstu k-literowe w alfabecie z N liter numerujemy ...
- Funkcją f – na przykład dla pojedynczych liter – ...

26-literowy alfabet (A–Z)numerujemy liczbami $0, 1, 2, \ldots, 25$ — zamiast A mamy 0, zamiast B – 1, zamiast ... X=23, itd.

Podstawowe pojęcia, przykłady:

- *pojedyncze* jednostki tekstu jawnego (i zaszyfrowanego) – a więc litery „przekładamy” na język cyfr.
- Digramy – litery A-Z i odstęp (27 znaków). System pozycyjny o podstawie 27.
- Jednostki tekstu k-literowe w alfabecie z N liter numerujemy ...
- Funkcją f – na przykład dla pojedynczych liter – ...

odstęp $= 26$; digramowi xy, gdzie $x, y \in \{0, 1, 2, \ldots, 26\}$ przyporządkowujemy numer

$$27 \cdot x + y \in \{0, 1, 2, \ldots, 728\}.$$

np. słowo „NO” to $27 \cdot 13 + 14 = 365$.

Podstawowe pojęcia, przykłady:

- *pojedyncze* jednostki tekstu jawnego (i zaszyfrowanego) – a więc litery „przekładamy” na język cyfr.
- Digramy – litery A-Z i odstęp (27 znaków). System pozycyjny o podstawie 27.
- Jednostki tekstu k-literowe w alfabecie z N liter numerujemy . . .
- Funkcją f – na przykład dla pojedynczych liter – . . .

. . . liczbami od 0 do $N^k - 1$; — np. dla trigramów;
$xyz = 27^2 \cdot x + 27 \cdot y + z$ — $x, y, z \in \{0, 1, \ldots, 27^3 - 1 = 19\,682\}$.

Podstawowe pojęcia, przykłady:

- *pojedyncze* jednostki tekstu jawnego (i zaszyfrowanego) – a więc litery „przekładamy” na język cyfr.
- Digramy – litery A-Z i odstęp (27 znaków). System pozycyjny o podstawie 27.
- Jednostki tekstu k-literowe w alfabecie z N liter numerujemy ...
- Funkcją f – na przykład dla pojedynczych liter – ...

... może być

$$f(\mathcal{P}) = \mathcal{P} + 3 \pmod{26}.$$

Jest to znany (od przeszło dwóch tysięcy lat!) kod Juliusza Cezara i stanowi szczególny przypadek *kodu szyfrowania*, którym jest *przesunięcie*

$$\mathcal{C} = f(\mathcal{P}) \equiv \mathcal{P} + b \pmod{N}, \qquad \mathcal{P} = f^{-1}(\mathcal{C}) \equiv \mathcal{C} - b \pmod{N}.$$

Podstawowe pojęcia, przykłady, c.d.

Kod *przesunięcia* jest bardzo łatwy do złamania – pod warunkiem, że odgadniemy (albo mamy dodatkowe informacje) *strukturę kodu.* (przesunięcie o dane b). Aby znaleźć b możemy postępować heurystycznie – albo (bardziej profesjonalnie) dokonać *analizy częstości.*

Lepszym systemem jest tzw. *przekształcenie afiniczne*:

$$\mathcal{C} = f(\mathcal{P}) \equiv a\mathcal{P} + b \pmod{N}, \qquad \mathcal{P} \equiv a^{-1}\mathcal{C} + b' \pmod{N},$$

gdzie $aa^{-1} \equiv 1 \pmod{N}$, $b' = -a^{-1} \cdot b$; dodatkowo $a \perp N$.

Ale i ten system jest łatwo złamać. Na przykład – alfabet 26 znakowy, jednostka litera – z analizy częstości wynika, że najczęściej spotykaną literą jest „K”, a drugą w kolejności – „D”. Odpowiada to (w tekście angielskim) literom „E” i „T” – tak więc musimy tylko rozwiązać układ kongruencji:

– tak więc musimy tylko rozwiązać układ kongruencji:
$10a' + b' \equiv 4 \pmod{26}$, $3a' + b' \equiv 19 \pmod{26}$.
Odejmując stronami: $7a' \equiv 11 \pmod{26}$ – skąd $a' \equiv 9 \pmod{26}$.
Podstawiamy: $b' \equiv 4 - 10a' \pmod{26} \equiv 18 \pmod{26}$.

Mamy $\mathcal{P} \equiv 9\mathcal{C} + 18 \pmod{26}$.

Nie zawsze sprawy są aż tak proste: na przykład – alfabet 28-znakowy: 26 liter (0-25), spacja (26) i ?(28), – z analizy częstości wynika, że najczęściej spotykaną literą jest „B”, a drugą w kolejności – „?”, co odpowiada spacji i „E”.
układ kongruencji: $a' + b' \equiv 26 \pmod{28}$, $27a' + b' \equiv 4 \pmod{28}$.
Po odjęciu (i podzieleniu przez 2): $a' \equiv 11 \pmod{14}$ – a to oznacza, że $a \equiv 11$ lub $25 \pmod{28}$. – odpowiednio $b \equiv 15$ lub $1 \pmod{28}$ (obie możliwości realizują to samo odwzorowanie odwrotne).
Albo heurystyka, albo . . . trzecia kongruencja, np. jeżeli „I” jest trzecią co do częstości literą w kryptogramie („T” w tekście jawnym) mamy $8a' + b' \equiv 19 \pmod{28}$.
To wystarcza aby wybrać właściwą (pierwszą) część alternatywy.

Szyfrowanie digramów

Każdą kombinację dwóch liter (łącznie z ewentualnym odstępem, lub ewentualnym uzupełnieniem nietypową literą) traktujemy jako liczbę zapisaną w postaci $xN + y$, gdzie $x, y, \in \{0, 1, \ldots, N-1\}$. Następnie liczbę $\mathcal{P}$ poddajemy transformacji afinicznej:

$$\mathcal{C} \equiv a\mathcal{P} + b \pmod{N^2}, \quad a \perp N.$$

Złamanie zaszyfrowanego digramu jest trochę bardziej skomplikowane niż złamanie szyfru jedno-literowego (przesunięcie, lub transformacja afiniczna). Znowu podstawa jest analiza częstości – w języku angielskim najczęstsze digramy to „TH” i „HE”.

Szyfrowanie (digramów) przy pomocy transformacji macierzowych

To dla nas sprawa dość oczywista: digram kojarzymy z wektorem o dwóch współrzędnych: $\binom{x}{y}$, gdzie liczby x i y są określone modulo N.

Na przykład w 26-literowym alfabecie A-Z wektorem digramu „NO” jest $\binom{13}{14}$.

Aby zaszyfrować naszą wiadomość poddajemy wektor-digram transformacji

$$\mathcal{C} = \mathcal{A}\mathcal{P}, \quad \text{to jest} \quad \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}.$$

Liczymy modulo N (liczba znaków alfabetu). Dlatego wyznacznik macierzy $\mathcal{A}$ – liczba $D = ad - cb$ nie tylko musi być różny od zera (aby istniała macierz odwrotna), ale dodatkowo $D \perp N$. Tylko wtedy, będziemy mogli deszyfrować wiadomość:

$$\mathcal{P} = \mathcal{A}^{-1}\mathcal{C}, \quad \text{to jest} \quad \begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{D} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix}.$$

W praktyce szyfrujemy nie jeden ale cały ciąg digramów –
na przykład „NOANSWER” to $\binom{13}{14}\binom{0}{13}\binom{18}{22}\binom{4}{17}$,
a macierz szyfrująca to $\mathcal{A} = \begin{pmatrix} 2 & 3 \\ 7 & 8 \end{pmatrix}$

$$\begin{aligned} \mathcal{C} &= \mathcal{AP} = \begin{pmatrix} 2 & 3 \\ 7 & 8 \end{pmatrix}\begin{pmatrix} 13 & 0 & 18 & 4 \\ 14 & 13 & 22 & 17 \end{pmatrix} \\ &= \begin{pmatrix} 68 & 39 & 102 & 59 \\ 203 & 104 & 302 & 164 \end{pmatrix} \equiv \begin{pmatrix} 16 & 13 & 24 & 7 \\ 21 & 0 & 16 & 8 \end{pmatrix}. \end{aligned}$$

(oczywiście liczymy modulo 26) Zaszyfrowany tekst to „QVNAYQHI”.

Transformacje macierzowe – łamanie kodu

Zakładamy, że kodujący używa macierzy 2×2; litery A-Z mają zwykłe odpowiedniki (0-25); spacja – 26, „?” – 27 i „!” – 28.
Dostaliśmy komunikat „GFPYJPX?UYXSTLADPLW”, ale – uwaga – wiemy, że uprzejma korespondentka podpisała się: KARLA.
Innymi słowy: „AR”$\to$„DP” i „LA”$\to$„LW” (piąta litera jest nieprzydatna). To już wystarcza do znalezienia macierzy $\mathcal{A}$, bo mamy:

$$\mathcal{A}\begin{pmatrix}0 & 11\\ 17 & 0\end{pmatrix} = \begin{pmatrix}3 & 11\\ 15 & 22\end{pmatrix}, \quad \mathcal{A}^{-1} = \begin{pmatrix}0 & 11\\ 17 & 0\end{pmatrix}\begin{pmatrix}3 & 11\\ 15 & 22\end{pmatrix}^{-1} = \begin{pmatrix}21 & 19\\ 22 & 18\end{pmatrix}$$

$$\mathcal{P} =$$

$$= \mathcal{A}^{-1}\mathcal{C} = \begin{pmatrix}21 & 19\\ 22 & 18\end{pmatrix}\begin{pmatrix}6 & 15 & 9 & 26 & 27 & 24 & 18 & 11 & 3 & 11\\ 5 & 24 & 15 & 23 & 20 & 23 & 19 & 0 & 15 & 22\end{pmatrix}$$

$$= \begin{pmatrix}18 & 17 & 10 & 26 & 19 & 13 & 14 & 28 & 0 & 1\\ 19 & 8 & 4 & 0 & 26 & 14 & 13 & 10 & 17 & 0\end{pmatrix}$$

czyli — „STRIKE AT NOON!KARLA”

Transformacje macierzowe – łamanie kodu

Zakładamy, że kodujący używa macierzy 2×2; litery A-Z mają zwykłe odpowiedniki (0-25); spacja – 26, „?” – 27 i „!” – 28.
Dostaliśmy komunikat „GFPYJPX?UYXSTLADPLW”, ale – uwaga – wiemy, że uprzejma korespondentka podpisała się: KARLA.
Innymi słowy: „AR”$\to$„DP” i „LA”$\to$„LW” (piąta litera jest nieprzydatna). To już wystarcza do znalezienia macierzy $\mathcal{A}$, bo mamy:

$$\mathcal{A}\begin{pmatrix}0 & 11\\ 17 & 0\end{pmatrix} = \begin{pmatrix}3 & 11\\ 15 & 22\end{pmatrix}, \quad \mathcal{A}^{-1} = \begin{pmatrix}0 & 11\\ 17 & 0\end{pmatrix}\begin{pmatrix}3 & 11\\ 15 & 22\end{pmatrix}^{-1} = \begin{pmatrix}21 & 19\\ 22 & 18\end{pmatrix}$$

$$\mathcal{P} =$$

$$= \mathcal{A}^{-1}\mathcal{C} = \begin{pmatrix}21 & 19\\ 22 & 18\end{pmatrix}\begin{pmatrix}6 & 15 & 9 & 26 & 27 & 24 & 18 & 11 & 3 & 11\\ 5 & 24 & 15 & 23 & 20 & 23 & 19 & 0 & 15 & 22\end{pmatrix}$$

$$= \begin{pmatrix}18 & 17 & 10 & 26 & 19 & 13 & 14 & 28 & 0 & 1\\ 19 & 8 & 4 & 0 & 26 & 14 & 13 & 10 & 17 & 0\end{pmatrix}$$

czyli — „STRIKE AT NOON!KARLA”

Warunek: $(D, N) = 1$!

Jeżeli warunek $D \perp N$ nie jest spełniony to już nie jest tak łatwo. Na przykład: 26-literowy alfabet, tekst zaszyfrowany „WKNCCHSSJH” i wiemy, że pierwsze 4 litery to słowo„GIVE”. Podobnie jak w poprzednim przykładzie:

$$\mathcal{P} = \text{„GIVE”} = \begin{pmatrix} 6 & 21 \\ 8 & 4 \end{pmatrix},$$

$$\mathcal{C} = \text{„WKNC”} = \begin{pmatrix} 22 & 13 \\ 10 & 2 \end{pmatrix} \qquad \mathcal{A}^{-1} = \mathcal{P}\mathcal{C}^{-1}.$$

Wyznacznik macierzy $\mathcal{C}$, $\det \mathcal{C} = 18$, a więc wyznacznik macierzy odwrotnej będzie miał wspólny dzielnik z $N = 26$ — $(18, 26) = 2$. Możemy *zredukować* wszystkie macierze do modulo 13:
$\mathcal{A} \to \bar{\mathcal{A}}$, $\mathcal{P} \to \bar{\mathcal{P}}$, $\mathcal{C} \to \bar{\mathcal{C}}$,

$$\text{Mamy } \bar{\mathcal{A}}^{-1} = \bar{\mathcal{P}}\bar{\mathcal{C}}^{-1} = \begin{pmatrix} 6 & 8 \\ 8 & 4 \end{pmatrix} \begin{pmatrix} 9 & 0 \\ 10 & 2 \end{pmatrix} = \begin{pmatrix} 2 & 4 \\ 3 & 2 \end{pmatrix}$$

Aby powrócić do modulo 26, zauważmy że do każdego wyrazu macierzy $\bar{\mathcal{A}}^{-1}$ możemy dodać 0 lub 13, czyli

$$\mathcal{A}^{-1} = \begin{pmatrix} 2 & 4 \\ 3 & 2 \end{pmatrix} + 13\mathcal{A}\mathcal{A},$$

gdzie macierz $\mathcal{A}\mathcal{A}$ składa się wyłącznie z zer lub jedynek (16 możliwości).

Wyznacznik macierzy $\mathcal{A}^{-1}$ musi być jednak liczbą *nieparzystą* (ze względu na jego względną pierwszość z 26)
– to już redukuje liczbę możliwości do sześciu; dalsze próby z równaniem

$$\mathcal{A}^{-1}\begin{pmatrix}22 & 13\\ 10 & 2\end{pmatrix} = \begin{pmatrix}6 & 21\\ 8 & 4\end{pmatrix}$$

gdzie za $\mathcal{A}^{-1}$ podstawiamy $\begin{pmatrix}2 & 4\\ 3 & 2\end{pmatrix} + 13\mathcal{A}\mathcal{A}$ pokazują, że:

$$\mathcal{A}\mathcal{A} = \begin{pmatrix}1 & 0\\ 1 & 1\end{pmatrix} \quad \text{lub} \quad \mathcal{A}\mathcal{A} = \begin{pmatrix}1 & 1\\ 1 & 1\end{pmatrix}$$

$$\text{albo} \quad \mathcal{A}^{-1} = \begin{pmatrix}15 & 4\\ 16 & 15\end{pmatrix} \quad \text{lub} \quad \mathcal{A}^{-1} = \begin{pmatrix}15 & 17\\ 16 & 15\end{pmatrix}$$

Ta druga macierz daje sensowny tekst – ‚GIVETHEMUP”; (pierwsza daje „GIVEGHEMHP”).

Szyfrowanie (digramów) przy pomocy transformacji macierzowych

Szyfrowanie przy pomocy macierzy istnieje także w wersji afinicznej – digram-wektor $\mathcal{P}$ mnożymy przez macierz szyfrującą $\mathcal{A}$, a następnie dodajemy pewien stały wektor $\mathcal{B}$:

$$\mathcal{C} = \mathcal{A} \cdot \mathcal{P} + \mathcal{B},$$

lub *explicite*

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix} = \begin{pmatrix} ax + by + e \\ cx + dy + f \end{pmatrix}.$$

Przekształcenie odwrotne to oczywiście

$$\mathcal{P} = \mathcal{A}^{-1}\mathcal{C} - \mathcal{A}^{-1}\mathcal{B}.$$

1976 – Hellman, Pohlig;

1. Szyfr „digitalizujemy”, np.

␣	A	B	C	D	E	F	G	H	I	J	K	L	M
↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕
00	01	02	03	04	05	06	07	08	09	10	11	12	13

N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	↕	
14	15	16	17	18	19	20	21	22	23	24	25	26	

2. wybieramy: liczbę pierwszą p, wykładnik e; $NWD(e, p-1) = 1$ i długość bloku tekstu $P_i < p$. Dzielimy tekst na bloki każdy blok poddajemy transformacji szyfrującej
$$C_i = f(P_i) \equiv {P_i}^e \pmod{p}$$
3. Transformacja deszyfrująca to

$$P_i = f^{-1}(C_i) \equiv ({P_i}^e)^d \equiv P_i \pmod{p}, \quad e \cdot d \equiv 1 \pmod{p-1}.$$

Szyfry wykładnicze — przykład (Song Y. Yan)

tekst

$P =$ ENCRYPTION REGULATION MOVES TO A STEP CLOSER,

05	14	03	18	25	16	20	09	15	14	00	18	05	07	21
12	01	20	09	15	14	00	13	15	22	05	19	00	20	15
00	0l	00	19	20	05	16	00	03	12	15	19	05	18	

0514	0318	2516	2009	1514	0018	0507	2112	0120	0915	1400
1315	2205	1900	2015	000l	0019	2005	1600	0312	1519	0518

$p = 7951$; $e = 91$. Sprawdzamy: NWM$(7950, 91) = 1$.

0514	0318	2516	2009	1514	0018	0507	2112	0120	0915	1400
1315	2205	1900	2015	000l	0019	2005	1600	0312	1519	0518

$C_1 = 0514^{91} \pmod{7951} = 2174$, $C_2 = 0318^{91} \pmod{7951} = 4468$,
$C_3 = 2516^{91} \pmod{7951} = 7889$, $C_4 = 2009^{91} \pmod{7951} = 6582$,
$C_5 = 1514^{91} \pmod{7951} = 924$, $C_6 = 0018^{91} \pmod{7951} = 5460$,
$C_7 = 0507^{91} \pmod{7951} = 7868$, $C_8 = 2112^{91} \pmod{7951} = 7319$,
$C_9 = 0120^{91} \pmod{7951} = 726$, $C_{10} = 915^{91} \pmod{7951} = 2890$,
$C_{11} = 1400^{91} \pmod{7951} = 7114$, $C_{12} = 1315^{91} \pmod{7951} = 5463$,
$C_{13} = 2205^{91} \pmod{7951} = 5000$, $C_{14} = 1900^{91} \pmod{7951} = 438$,
$C_{15} = 2015^{91} \pmod{7951} = 2300$, $C_{16} = 0001^{91} \pmod{7951} = 1$,
$C_{17} = 0019^{91} \pmod{7951} = 1607$, $C_{18} = 2005^{91} \pmod{7951} = 3509$,
$C_{19} = 1600^{91} \pmod{7951} = 7143$, $C_{20} = 0312^{91} \pmod{7951} = 5648$,
$C_{21} = 1519^{91} \pmod{7951} = 3937$, $C_{22} = 0518^{91} \pmod{7951} = 4736$.

Szyfry wykładnicze — przykład (Song Y. Yan), c.d

tekst

$P =$ ENCRYPTION REGULATION MOVES TO A STEP CLOSER,

0514	0318	2516	2009	1514	0018	0507	2112	0120	0915	1400
1315	2205	1900	2015	000l	0019	2005	1600	0312	1519	0518

$\downarrow$

2174	4468	7889	6582	0924	5460	7868	7319	0726	2890	7114
5463	5000	0438	2300	0001	1607	3509	7143	5648	3937	5064

Transformacja deszyfrująca to

$$P_i = f^{-1}(C_i) \equiv (P_i{}^e)^d \equiv P_i \pmod{p}, \quad e \cdot d \equiv 1 \pmod{p-1}.$$

$$e \cdot d \equiv 1 \pmod{7951-1} \quad \rightarrow \quad d \equiv 961 \pmod{7950}.$$

2174	4468	7889	6582	0924	5460	7868	7319	0726	2890	7114
5463	5000	0438	2300	0001	1607	3509	7143	5648	3937	5064

$$
\begin{aligned}
P_1 &= 2174^{961} \pmod{7951} = 514, & P_2 &= 4468^{961} \pmod{7951} = 318,\\
P_3 &= 7889^{961} \pmod{7951} = 2516, & P_4 &= 6582^{961} \pmod{7951} = 2009,\\
P_5 &= 924^{961} \pmod{7951} = 1514, & P_6 &= 5460^{961} \pmod{7951} = 18,\\
P_7 &= 7868^{961} \pmod{7951} = 507, & P_8 &= 7319^{961} \pmod{7951} = 2112,\\
P_9 &= 726^{961} \pmod{7951} = 120, & P_{10} &= 2890^{961} \pmod{7951} = 915,\\
P_{11} &= 7114^{961} \pmod{7951} = 1400, & P_{12} &= 5463^{961} \pmod{7951} = 1315,\\
P_{13} &= 5000^{961}. \pmod{7951} = 2205, & P_{14} &= 438^{961} \pmod{7951} = 1900,\\
P_{15} &= 2300^{961} \pmod{7951} = 2015, & P_{16} &= 1^{961} \pmod{7951} = 1,\\
P_{17} &= 1607^{961} \pmod{7951} = 19, & P_{18} &= 3509^{961} \pmod{7951} = 2005,\\
P_{19} &= 7143^{961} \pmod{7951} = 1600, & P_{20} &= 5648^{961} \pmod{7951} = 312,\\
P_{21} &= 3937^{961} \pmod{7951} = 1519, & P_{22} &= 4736^{961} \pmod{7951} = 518
\end{aligned}
$$

0514	0318	2516	2009	1514	0018	0507	2112	0120	0915	1400
1315	2205	1900	2015	000l	0019	2005	1600	0312	1519	0518