

Kryptologia (nie)stosowana

Andrzej GRZESIK, Kraków

Artykuł będzie dotyczył kryptologii niestosowanej, czyli takiej, która wyszła już z użycia oraz takiej, która jeszcze nie weszła do użycia. Postaramy się też sprawdzić, czy aby na pewno te historyczne metody nie są już stosowane. . .

Słowa *kryptografia* i *kryptologia* często używane są zamiennie, a tak naprawdę ich znaczenia są różne (jak *geografia* i *geologia*, czy *topografia* i *topologia*). Formalnie kryptologia to nauka o przekazywaniu informacji w sposób zabezpieczony przed niepowołanym dostępem. Dzielimy ją na kryptografię (naukę o układaniu systemów kryptograficznych) oraz kryptoanalizę (naukę o ich łamaniu). Jak wskazuje tytuł artykułu – tym także się zajmujemy.

1. Z życia wzięte. . .

Kryptografia jest stosowana już od wielu wieków, między innymi Juliusz Cezar szyfrował wiadomości militarne. Niebagatelną rolę w dziejach historii odegrała też kryptoanaliza. Historię złamania szyfru Enigmy przez polskich matematyków zna chyba każdy, więc może przytoczę inny przykład: podczas I wojny światowej Zimmermann wysłał telegram do Meksyku, w którym Niemcy proponowali sojusz w przypadku, gdyby Stany Zjednoczone włączyły się do wojny. Telegram rozszyfrowano i jego treść tak Amerykanów zdenerwowała, że. . . włączyli się do wojny.

Ale też i w życiu codziennym przydaje się kryptografia. Zaświadcza o tym między innymi... *Kamasutra*, czyli indyjska księga o sztuce miłości. Pozycja numer 45 w spisie 64 sztuk, których poznanie zaleca się kobietom, to *mlecchita-vikalpa*, czyli sztuka posługiwania się tajnym pismem. Według *Kamasutry* umiejętność ta przydaje się na równi z innymi, takimi jak gotowanie, śpiewanie czy haftowanie.

2. Szyfry monoalfabetyczne

Tyle słowem wstępu – przejdźmy do konkretów, czyli szyfrów. Historycznie najstarsze, ale też i najprostsze, są tzw. **szyfry monoalfabetyczne**. Polegają na tym, że szyfrując, każdą literę tekstu jawnego zastępujemy w sposób jednoznaczny jedną literą, co więcej – za każdym razem w taki sam sposób. Tzn. litera A jest zawsze szyfrowana tak samo, niezależnie od jej pozycji w tekście czy liter poprzedzających. Takim szyfrem jest np. tzw. **szyfr Cezara** (tak, to ten używany przez Juliusza Cezara), w którym:

$$A \rightarrow D, \quad B \rightarrow E, \quad C \rightarrow F, \quad \dots, \quad Z \rightarrow C.$$

Widać regułę? Szyfr ten spełniał swoją rolę, ponieważ zdecydowana większość ludzi nie znała alfabetu.

Inne przykłady takich szyfrów to **ROT-13**:

$$A \rightarrow N, \quad B \rightarrow O, \quad C \rightarrow P, \quad \dots, \quad Z \rightarrow M$$

oraz **AtBash**:

$$A \rightarrow Z, \quad B \rightarrow Y, \quad C \rightarrow X, \quad \dots, \quad M \rightarrow N.$$

Ich zaletą było to, że deszyfrowało się tak samo jak szyfrowało, więc było 2 razy mniej do zapamiętania. :-)

Jak złamać taki kod?

Najczęstsza i zarazem najefektywniejsza metoda to **analiza częstości występowania liter** w szyfrogramie. Metoda korzysta z tego, że litery w tekście występują z różną częstością. W języku polskim wygląda to następująco:

8,78%	E	4,17%	S	2,74%	P	1,11%	Ą
8,65%	I	3,85%	Y	2,19%	L	0,93%	H
8,61%	A	3,82%	C	2,05%	U	0,84%	Ś
7,34%	O	3,72%	R	1,49%	B	0,79%	Ó
5,67%	Z	3,35%	M	1,40%	Ę	0,56%	Ć
5,60%	N	3,33%	D	1,40%	G	0,36%	F
4,30%	T	2,95%	K	1,36%	Ł	0,12%	Ń
4,25%	W	2,81%	J	1,24%	Ż	0,06%	Ź

Natomiast w języku angielskim:

11.16%	E	5.74%	S	3.00%	H	1.10%	K
8.50%	A	5.45%	L	2.47%	G	1.00%	V
7.58%	R	4.54%	C	2.07%	B	0.29%	X
7.55%	I	3.63%	U	1.81%	F	0.27%	Z
7.16%	O	3.39%	D	1.78%	Y	0.20%	J
6.95%	T	3.17%	P	1.29%	W	0.19%	Q
6.66%	N	3.01%	M				

W praktyce — mamy zaszyfrowany tekst, np.

PXPXKXENVDRUXVTNLXHYMXGMAXYKXJNXGVRFXMAHWGXXWLEHGZXXKBIAXXMXQM

i sądzimy, że jest to tekst w języku angielskim zakodowany przez pewne przesunięcie w alfabecie. Szukamy więc najczęściej występującej litery – w tym przypadku jest to X – i zakładamy, że zastosowano przesunięcie $E \rightarrow X$. Zatem przyjmując do deszyfracji przesunięcie o 7 (by X przeszedł w E), dostajemy:

WEWERELUCKYBECAUSEOFTENTHEFREQUENCYMETHODNEEDSLONGERCIPHERTEXT.

3. Szyfry polialfabetyczne

Za łatwo poszło. Trudniejszym kodowaniem jest **szyfr polialfabetyczny**. Polega on na tym, że, tak jak wcześniej, jednej literze odpowiada jedna litera, ale tym razem nie zawsze ta sama – jest ona zależna od pozycji litery w tekście. Przykładem jest szyfr Vigenère’a, w którym najpierw wybieramy słowo-klucz, np. MSN, a następnie szyfrujemy robiąc cykliczne przesunięcie w alfabecie, ale o tyle, ile nam wskazuje słowo-klucz. W przypadku klucza MSN przesuwamy pierwszą literę o 13 (bo M jest 13 literą w alfabecie), drugą o 19 (bo S jest 19), trzecią o 14 (N jest 14), a dalej cyklicznie, czyli czwartą o 13, piątą o 19, itd. W ten sposób słowo

GRZEGORZEWICE

zakodujemy przez

TKNRZCESSJBQR.

A jak złamać ten kod?

Ewidentnie jest on trudniejszy do złamania. Niemniej, zauważmy, że jedynym problemem jest odkrycie długości klucza. Znając ją, możemy stosować poprzednią metodę dla liter położonych w odległościach będących wielokrotnościami długości klucza. A jak znaleźć długość klucza? Na to mamy aż 3 metody.

Pierwsza (tzw. **metoda Kasiskiego**) polega na szukaniu w zakodowanym tekście powtarzających się ciągów znaków. Przykładowo, w języku angielskim bardzo często występuje trójka **the**, więc poszukajmy jej w kodzie. Załóżmy, że w anglojęzycznym tekście zakodowanym znaleźliśmy 8 wystąpień trójki **sm**, a odstęp między nimi wynosi odpowiednio 24, 32, 20, 56, 37, 43 i 92. Możemy zauważyć, że większość z nich podzielna jest przez 4 – zatem możemy przyjąć, że długość klucza wynosi 2 lub 4. Wartości 37 i 43 mogły być związane z przypadkowym zaszyfrowaniem innego tekstu z innymi przesunięciami i w ten sposób otrzymaliśmy jedną „falszywą” trójkę **sm**.

Druga metoda (tzw. **indeks koincydencji**) opiera się na tym, że im tekst jest dłuższy, tym częstość występowania liter jest bliższa losowej. To odchylenie od rozkładu równomiernego możemy odczytać ze wzoru:

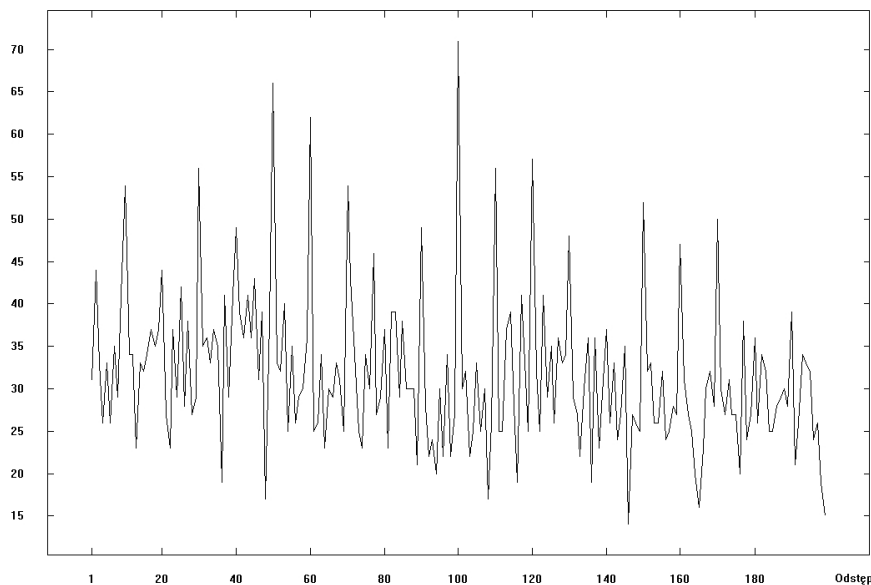
$$IC = \frac{\sum_{a \in \mathcal{A}} F_a(F_a - 1)}{N(N - 1)},$$

gdzie F_a – liczba wystąpień litery a z alfabetu \mathcal{A} w tekście o długości N . Doświadczalnie sprawdzono, jak kształtuje się długość klucza w zależności od IC , dzięki czemu możemy ją odczytać (z pewnym błędem) z tabelki:

długość klucza	IC
1	0,066
2	0,052
3	0,047
4	0,045
5	0,044
10	0,041
bardzo duża	0,038

Trzecia metoda – **autokorelacja** – polega na tym, że porównujemy tekst z jego przesuniętą kopią i zliczamy pokrywające się znaki. Okazuje się, że jeśli przesuniemy o długość klucza (lub jego wielokrotność), to pokrywających się znaków jest dużo więcej. Na rysunku 3 znajduje się wykres przedstawiający liczbę pokrywających się znaków w zależności od przesunięcia tekstu względem siebie. Jaka jest długość klucza?

Liczba znaków, które się zgadzają.



Rys. 1. Ilustracja metody autokorelacji. Jaka jest długość klucza?

4. Szyfry poligramowe

Jak widać, taki szyfr też da się łatwo złamać. Kolejnym utrudnieniem jest szyfrowanie po kilka znaków, a nie – jak wcześniej – po jednym. Takie szyfry nazywamy **szyframi poligramowymi**. Ich przewaga polega na tym, że dana litera jest szyfrowana w różny sposób (zależny od sąsiednich liter). Przykładem takiego systemu jest tzw. **szyfr Playfair**. Aby szyfrować, najpierw należy ustalić klucz (np. SZKOŁA MATEMATYKI POGLĄDOWEJ), usunąć z niego powtarzające się litery, uzupełnić brakującymi literami alfabetu i całość ustawić w kwadrat 5×5 . W alfabecie łacińskim jest 26 liter, dlatego dwie się utożsamia (zazwyczaj I z J). W naszym przykładzie dostajemy:

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

Szyfrowanie polega na zamianie dwóch liter w dwie nowe (w przypadku, gdy mamy nieparzystą liczbę liter, dopisujemy ustaloną literę np. X) w następujący sposób:

- Jeśli litery, które chcemy zakodować, znajdują się w różnych wierszach i kolumnach, to zastępowane są znakami znajdującymi się na drugiej przekątnej, stworzonego przez te znaki prostokąta.

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

Szyfrując litery **EB** otrzymamy **AH**. Ważna jest kolejność podawania liter do szyfrowania – szyfrując **BE** otrzymamy **HA**.

- Jeżeli szyfrowane znaki znajdują się w jednej kolumnie (lub wierszu) macierzy szyfrowania, wówczas zastępowane są znakami następującymi po nich w tej kolumnie (wierszu).

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

Szyfrując litery **MY** otrzymamy **TA**. Podobnie, szyfrując **AS** dostaniemy **IA**.

Innym sposobem szyfrowania poligramowego jest zastosowanie **macierzy szyfrującej**. Aby kodować tą metodą, najpierw cały tekst zapisujemy w macierzy $2 \times n$, gdzie każdą literę zastępujemy liczbą oznaczającą jej pozycję w alfabecie ($A \rightarrow 0, B \rightarrow 1, C \rightarrow 2, \dots, Z \rightarrow 25$). Przykładowo:

$$\text{MATEMATYKA} = \begin{pmatrix} 12 & 19 & 12 & 19 & 10 \\ 0 & 4 & 0 & 24 & 0 \end{pmatrix}.$$

Kodowanie polega na mnożeniu modulo 26 odwracalnej macierzy 2×2 przez macierz reprezentującą tekst. Niech macierzą-kluczem będzie $A = \begin{pmatrix} 2 & 3 \\ 7 & 8 \end{pmatrix}$. Jest odwracalna (wyznacznik niezerowy i względnie pierwszy z 26). Słowo **MATEMATYKA** zakodowane tą macierzą, to

$$\begin{pmatrix} 2 & 3 \\ 7 & 8 \end{pmatrix} \begin{pmatrix} 12 & 19 & 12 & 19 & 10 \\ 0 & 4 & 0 & 24 & 0 \end{pmatrix} = \begin{pmatrix} 24 & 24 & 24 & 6 & 20 \\ 22 & 9 & 22 & 13 & 18 \end{pmatrix} = \text{YWYJYWGNUS}.$$

Dekodowanie odbywa się przez mnożenie macierzy odwrotnej do klucza $A^{-1} = \begin{pmatrix} 14 & 11 \\ 17 & 10 \end{pmatrix}$ przez macierz zakodowanego tekstu.

5. Kryptografia z kluczem publicznym

Wszystkie dotychczas zaprezentowane metody kodowania miały dużą wadę – znając sposób szyfrowania, od razu znano sposób deszyfrowania. Gdyby takie coś obowiązywało np. w banku, to mielibyśmy spory problem. Ponadto, by móc kodować wiadomości, musimy wcześniej ustalić w bezpieczny sposób klucz, co też nie zawsze jest możliwe. Rozwiązaniem tego problemu jest kryptografia z kluczem publicznym oparta na funkcjach jednokierunkowych.

Definicja 1. *Funkcja jednokierunkowa* jest to funkcja wzajemnie jednoznaczna $f : X \rightarrow Y$ taka, że dla danego $x \in X$ łatwo jest obliczyć $f(x)$, ale wyliczenie $f^{-1}(y)$ dla przypadkowo wybranego y jest bardzo trudne.

W kryptografii używane są funkcje, które pozostają jednokierunkowe pod warunkiem, że pewna informacja (klucz prywatny) jest utrzymana w tajemnicy. Oznacza to, że każdy może przesłać zaszyfrowaną wiadomość do danego użytkownika korzystając z publicznego klucza (funkcji szyfrującej). Nie ma potrzeby żadnego tajnego porozumiewania się między nadawcą i odbiorcą.

Przykładem funkcji jednokierunkowej jest potęgowanie w ciele \mathbb{Z}_p^* . Odwrócenie tej funkcji to tzw. problem logarytmu dyskretnego:

Definicja 2. *Problemem logarytmu dyskretnego* w ciele \mathbb{Z}_p^* przy podstawie $g \in \mathbb{Z}_p^*$ nazywamy zadanie wyznaczenia dla danego $y \in \mathbb{Z}_p^*$ takiej liczby naturalnej x , że $y \equiv g^x \pmod{p}$.

Nie znamy jeszcze metod pozwalających szybko rozwiązywać ten problem.

6. Kryptosystem RSA

Szyfr RSA jest najczęściej używanym obecnie kryptosystemem. Wprawdzie artykuł miał być o kryptologii niestosowanej, ale o szyfrze RSA też warto wspomnieć.

Pierwsza osoba (nazwijmy ją Apolonią) wybiera dwie dość duże liczby pierwsze p i q tak, żeby ich iloczyn równy n był większy od długości szyfrowanego tekstu. Ponadto wybiera losowy wykładnik e , który jest względnie pierwszy zarówno z $p - 1$, jak i z $q - 1$. Kluczem publicznym Apolonii będzie para (n, e) .

Natomiast jej kluczem prywatnym niech będzie dowolna liczba d , dla której jednocześnie

$$de \equiv 1 \pmod{p-1} \quad \text{oraz} \quad de \equiv 1 \pmod{q-1}.$$

Taką liczbę Apolonia może łatwo wyliczyć korzystając z algorytmu Euklidesa dla e i $\text{NWW}(p-1, q-1)$.

Gdy drugi użytkownik (nazwijmy go Boguś) pragnie przesłać Apolonii wiadomość w , wysyła jej $s \equiv w^e \pmod{n}$.

Aby odszyfrować wiadomość, Apolonia oblicza resztę z dzielenia s^d przez n , ponieważ

$$s^d \equiv w^{de} \equiv w \pmod{n}.$$

Co powstrzymuje nieuprawnioną osobę (niech to będzie Czesiek) od rozszyfrowania wiadomości?

Znając n i e (klucz publiczny Apolonii) oraz przechytując wiadomość zaszyfrowaną Czesiek nie jest w stanie odwrócić operacji $w \rightarrow w^e \pmod{n}$ bez znajomości klucza deszyfrującego d .

Wydaje się, że nie ma innej metody na znalezienie wykładnika deszyfrującego niż odnalezienie w pierw liczb p i q . Piszemy „wydaje się”, gdyż to stwierdzenie nie jest udowodnione. Możemy jedynie powiedzieć, że złamanie szyfru RSA jest prawdopodobnie tak trudne jak rozkład n na czynniki pierwsze.

7. System „Polly Cracker”

Przejdźmy teraz do metod, które jeszcze nie weszły do powszechnego użycia. Bardzo ciekawy schemat kryptosystemu z kluczem publicznym (zwany **systemem „Polly Cracker”**) został opisany przez Fellowsa.

Niech \mathbb{F} będzie ciałem skończonym i niech $T = \{t_i\}_{i=1}^n$ będzie zbiorem zmiennych. Kluczem prywatnym Apolonii jest losowy wektor $y \in \mathbb{F}^n$, a jej klucz publiczny stanowi pewien podzbiór $B = \{q_j\}$ zbioru wielomianów $\mathbb{F}[T]$ o tej własności, że

$$q_j(y) = 0 \quad \text{dla wszystkich } j.$$

Chcąc wysłać wiadomość w , Boguś generuje element

$$p = \sum h_j q_j,$$

będący kombinacją liniową elementów z B i wysyła Apolonii wielomian

$$c = p + w.$$

Apolonia, otrzymawszy szyfrogram w postaci wielomianu c , deszyfruje wiadomość w przez wyliczenie wartości tego wielomianu w punkcie y , tzn.

$$c(y) = p(y) + w = w.$$

Aby taki system dobrze funkcjonował, znalezienie zera zbioru wielomianów musi być bardzo trudne (NP-trudne).

Zobaczmy na przykładzie, jak to funkcjonuje.

Zbudujemy system kryptograficzny korzystający z tego, że problem trójkolorowania grafu jest NP-trudny.

Kluczem publicznym niech będzie dowolny graf $G = (V, E)$, zbiór zmiennych $\{t_{v,i} : v \in V, 1 \leq i \leq 3\}$ i baza wielomianów $B = B_1 \cup B_2 \cup B_3$, gdzie

$$B_1 = \{t_{v,1} + t_{v,2} + t_{v,3} - 1 : v \in V\};$$

$$B_2 = \{t_{v,i}t_{v,j} : v \in V, 1 \leq i, j \leq 3\};$$

$$B_3 = \{t_{u,i}t_{v,i} : uv \in E, 1 \leq i \leq 3\}.$$

Kluczem prywatnym jest właściwe trójkolorowanie grafu G , tzn. odwzorowanie $v \mapsto i_v \in \{1, 2, 3\}$ określone na wierzchołkach $v \in V$ takie, że jeśli $uv \in E$, to $i_u \neq i_v$.

Zauważmy, że jeśli znamy klucz prywatny, to możemy otrzymać punkt zerowy wszystkich wielomianów z B przyjmując $t_{v,i}$ równe 1, gdy wierzchołek v ma kolor i oraz 0 w przeciwnym przypadku. Łatwo też zauważyć, że aby znaleźć wspólne miejsce zerowe tej grupy wielomianów, trzeba znaleźć właściwe trójkolorowanie grafu.

Od razu nasuwa się pytanie – czy inne problemy NP-trudne też można do tego użyć? Okazuje się, że tak – nawet wszystkie:

Twierdzenie 3 (Fellows, Koblitz). *Dowolnego problemu \mathcal{P} klasy NP można użyć do konstruowania kryptosystemu analogicznego do poprzedniego, tzn. dla każdego przypadku problemu \mathcal{P} można skonstruować zbiór B (o rozmiarze wielomianowym), że znajomość zera zbioru B jest wielomianowo równoważna znajomości rozwiązania tego problemu.*

8. Kryptosystemy eliptyczne

W ostatnich latach bujny rozwój przeżywa też **kryptografia na krzywych eliptycznych**. Prawdopodobnie znacznie być szerzej stosowana, więc warto ją tu zaprezentować.

Definicja 4. *Krzywą eliptyczną E nad ciałem \mathbb{F} nazywamy gładką krzywą zadaną równaniem postaci*

$$Y^2 + a_1XY + a_3Y = X^3 + a_2X^2 + a_4X + a_6, \quad a_i \in \mathbb{F}.$$

Jeśli $\text{char } \mathbb{F} \neq 2, 3$, to przez liniową zamianę zmiennych równanie krzywej eliptycznej można sprowadzić do postaci

$$Y^2 = X^3 + aX + b, \quad a, b \in \mathbb{F}.$$

Przez E oznaczmy zbiór złożony z punktów $(x, y) \in \mathbb{F}^2$, których współrzędne spełniają to równanie, oraz punktu „w nieskończoności” oznaczanego przez O .

Punktem przeciwnym do punktu $P = (x, y)$ leżącego na krzywej eliptycznej nazywamy punkt $-P = (x, -y)$, który także należy do tej krzywej eliptycznej.

Natomiast sumę punktów P i Q leżących na krzywej eliptycznej E definiujemy następująco:

- Jeśli $Q = O$, to $P + Q = P$;

- Jeśli P i Q mają takie same współrzędne x , to $P + Q = O$;
- Jeśli P i Q mają różne współrzędne x , to prosta przechodząca przez te punkty przecina krzywą E w jeszcze jednym, jedynym punkcie R i wówczas przyjmujemy $P + Q = -R$;
- Jeśli $P = Q$, to prowadzimy styczną w tym punkcie do krzywej E , przecina ona krzywą w jeszcze jednym punkcie R , wtedy przyjmujemy $P + Q = -R$.

Można to streścić w jednym zdaniu:

Suma trzech punktów leżących na jednej prostej wynosi 0.

Taka definicja sumy zadaje na zbiorze punktów krzywej eliptycznej strukturę grupy abelowej. Mamy tu też analogon problemu logarytmu dyskretnego:

Definicja 5. *Problem logarytmu dyskretnego* na E o podstawie $Q \in E$ jest problemem znalezienia dla danego $P \in E$ liczby całkowitej n , dla której $P = nQ$, o ile taka liczba istnieje.

Korzystając z tego, że na krzywej eliptycznej jeszcze trudniej o jego rozwiązanie, możemy stworzyć system kryptograficzny. Weźmy krzywą eliptyczną E nad ciałem skończonym i ustalmy na niej punkt Q .

Wymiana kluczy:

Apolonia wybiera losową liczbę k_A , wylicza punkt $k_A Q$ i wysyła Bogusiowi. Boguś analogicznie: wybiera losową liczbę k_B , wylicza $k_B Q$ i wysyła Apolonii. Wspólnym kluczem jest $P = k_A k_B Q$. Apolonia i Boguś mogą go wyznaczyć mnożąc swoją sekretną liczbę przez otrzymaną od drugiej osoby.

Podsluchujący Czesiek musi wyznaczyć $P = k_A k_B Q$ znając tylko Q , $k_A Q$ i $k_B Q$ – czego nie może zrobić bez znajomości k_A i k_B .

Przesyłanie informacji:

Jeśli Boguś chce wysłać Apolonii wiadomość $M \in E$, to wybiera losową liczbę l i wysyła parę punktów $(lQ, M + lk_A Q)$.

Aby odszyfrować wiadomość, Apolonia wymnaża pierwszy punkt przez swoją tajną liczbę k_A i odejmuje wynik od drugiego punktu:

$$M + lk_A Q - k_A \cdot lQ = M.$$

Czesiek, aby przeczytać wiadomość, nadal musi rozwiązać problem logarytmu dyskretnego...

9. W praktyce

Na koniec polecam, żeby samemu sprawdzić, czy aby na pewno metody te nie są stosowane. Zwłaszcza te proste, historyczne.

Jak to sprawdzić?

Można włączyć jakiś program, np. komunikator gadu-gadu. Następnie ustawić proste hasło (np. `abcdefg`) i zobaczyć, jak wygląda jego zaszyfrowana wersja w pliku przechowującym dane programu. Życzę miłej zabawy. :-)