Kryptografia z elementami algebry, wykład 2

Maciej Grześkowiak

18 października 2021

Elementy algebry

Kryptografia, złożoność obliczeniowa

Co to jest kryptografia?

KRYPTOLOGIA

Co to jest kryptografia?

KRYPTOLOGIA

KRYPTOGRAFIA

Co to jest kryptografia?

KRYPTOLOGIA

KRYPTOGRAFIA

KRYPTOANALIZA

Co to jest kryptografia?

KRYPTOLOGIA

KRYPTOGRAFIA

Projektowanie:

Protokoły

Algorytmy

KRYPTOANALIZA

Co to jest kryptografia?

KRYPTOLOGIA

KRYPTOGRAFIA
Projektowanie:
Protokoły
Algorytmy

KRYPTOANALIZA
Łamanie:
Protokoły
Algorytmy

Protokoły kryptograficzne

Definicja Protokół jest szeregiem kroków, obejmujących dwie lub wiecej stron, które należy wykonać w celu realizacji zadania.

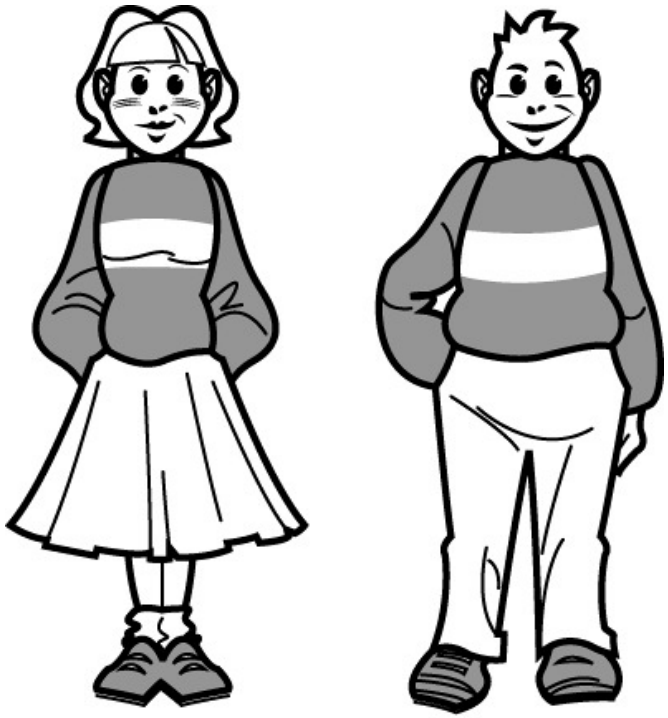
Własności protokołów

1. Każdy użytkownik musi go znać i kolejno wykonywać wszystkie kroki,
2. Każdy użytkownik musi zgodzić się na jego stosowanie,
3. Protokół musi być nie mylący. Każdy krok powinien być dobrze zdefiniowany i nie może wystapić szansa na jakiekolwiek nieporozumienie,
4. Protokół musi być kompletny. Dla każdej możliwej sytuacji musi być podany odpowiedni sposób postępowania

Protokoły kryptograficzne

Definicja Protokół kryptograficzny jest protokołem, który wykorzystuje kryptografię. Strony uczestniczące w protokole mogą być bezwarunkowo ufającymi sobie osobami, albo być całkowicie nieufającymi sobie adwersarzami.

Uczestnicy protokołu: Alice i Bob



Alice Bob

Uczestnicy protokołu: Mallet

Algorytmy kryptograficzne

- Przestrzeń wiadomości $\mathcal{M}$,
- Przestrzeń szyfrogramów $\mathcal{C}$
- Przestrzeń kluczy $\mathcal{K}$,

Algorytmy kryptograficzne

- Przestrzeń wiadomości $\mathcal{M}$,
- Przestrzeń szyfrogramów $\mathcal{C}$
- Przestrzeń kluczy $\mathcal{K}$,

- Algorytm generowania klucza G,
- Algorytm szyfrowania E,
- Algorytm deszyfrowania D

Algorytmy kryptograficzne

- Przestrzeń wiadomości $\mathcal{M}$,
- Przestrzeń szyfrogramów $\mathcal{C}$
- Przestrzeń kluczy $\mathcal{K}$,

- Algorytm generowania klucza G,
- Algorytm szyfrowania E,
- Algorytm deszyfrowania D

$$E_k(m) = c, \qquad D_k(c) = m$$

Kryptografia może zapewnić poufność

Symetryczny Protokół Szyfrowania

ALICE, BOB

cel: Poufne przekazywanie informacji

1. Uzgadniają jawnie (E, D)
2. Uzgadniają w bezpieczny sposób K
3. Alice tworzy M
4. Alice oblicza $E_K(M) = C$
5. Alice wysyła do Boba C
6. Bob oblicza $D_K(C) = M$

Symetryczny protokół wymiany klucza, tajne klucze k_A i k_B

Cel: Bezpieczna dystrybucja klucza sesyjnego.

1. Alice, aby połączyć się z Bobem kieruje zamównie na klucz sesyjny do centrali dystrybucji kluczy KDC. (Key distribution center)
2. KDC generuje klucz sesyjny k i szyfruje k otrzymując $c_A = E_{k_A}(k)$ oraz $c_B = E_{k_B}(k)$. Ponadto szyfruje kluczem k_B informację o tożsamości Alice m otrzymując $c = E_{k_B}(m)$.
3. KDC wysyła c_A, c_B do Alice.
4. Alice deszyfruje c_A i w ten sposób otrzymuje k.
5. Alice wysyła c_B, c do Boba
6. Bob deszyfruje c_B oraz c i w ten sposób otrzymuje k. Wykorzystuje m do określenia kim jest Alice.
7. Alice i Bob wykorzystują k do bezpiecznej komunikacji.

Bezpieczeństwo doskonałe

Definicja 1 Schemat szyfrowania (G, E, D) nad $\mathcal{M}$ jest doskonały, jeśli dla każdego rozkładu prawdopodobieństwa nad $\mathcal{M}$, dla każdego $m \in \mathcal{M}$, i każdego $c \in \mathcal{C}$, $Pr[C = c] > 0$ mamy

$$Pr[M = m|C = c] = Pr[M = m].$$

Bezpieczeństwo doskonałe

Definicja 1 Schemat szyfrowania (G, E, D) nad $\mathcal{M}$ jest doskonały, jeśli dla każdego rozkładu prawdopodobieństwa nad $\mathcal{M}$, dla każdego $m \in \mathcal{M}$, i każdego $c \in \mathcal{C}$, $Pr[C = c] > 0$ mamy

$$Pr[M = m|C = c] = Pr[M = m].$$

Lemat Schemat szyfrowania (G, E, D) nad $\mathcal{M}$ jest doskonały wtedy i tylko wtedy, gdy dla każdego rozkładu prawdopodobieństwa nad $\mathcal{M}$, dla każdego $m \in \mathcal{M}$, i każdego $c \in \mathcal{C}$, $Pr[C = c] > 0$ mamy

$$Pr[M = m|C = c] = Pr[C = c].$$

One-time pad

Ustalmy $l > 0$. Niech $\mathcal{M}, \mathcal{C}, \mathcal{K} \in \{0,1\}^l$. Niech $\oplus$ oznacza pobitowy xor.

1. G generuje klucz $k \in \mathcal{K}$ w sposób jednostajny tj. $P[K = k] = \frac{1}{2^l}$.
2. E : Dla danych $m \in \mathcal{M}$, $k \in \mathcal{K}$ algorytm oblicza $c \in \mathcal{C}$, gdzie $c = m \oplus k$.
3. D : Dla danych $c \in \mathcal{M}$, $k \in \mathcal{K}$ algorytm oblicza $m \in \mathcal{C}$, gdzie $m = c \oplus k$.

Bezpieczeństwo One-time pad

Twierdzenie: Schemat szyfrowania One-time pad jest doskonały.

Dowód: Ustalmy rozkład prawdopodobieństwa na $\mathcal{M}$ oraz $m \in \mathcal{M}$, $c \in \mathcal{C}$. Mamy,

$$Pr[C = c|M = m] = Pr[M \oplus K = c|M = m] =$$
$$= Pr[m \oplus K = c] = Pr[K = m \oplus k] = \frac{1}{2^l}$$

Powyższe obliczenia są prawdziwe dla każdego $m \in \mathcal{M}$ oraz każdego $c \in \mathcal{C}$. W szczególności dla $m_0, m_1 \in \mathcal{M}$ mamy,

$$Pr[C = c|M = m_0] = \frac{1}{2^l} = Pr[C = c|M = m_1].$$

Z Lematu 2 dostajemy, że One-Time Pad jest doskonały.

Bezpieczeństwo doskonałe

Twierdzenie: Niech schemat szyfrowania (G, E, D) nad $\mathcal{M}$ będzie doskonały. Niech $\mathcal{K}$ będzie przestrzenią kluczy generowaną przez G. Wtedy $|\mathcal{K}| \geq |\mathcal{M}|$.

Operacje elementarne na bitach

p	0	0	0	0	1	1	1	1
r_1	0	0	1	1	0	0	1	1
r_2	0	1	0	1	0	1	0	1
w	0	1	1	0	1	0	0	1
np	0	0	0	1	0	1	1	1

Notacja wielkie-O

Definicja Niech $f, g : \mathbb{N} \mapsto \mathbb{R}$. Mówimy, że

$$f(n) = O(g(n))$$

jeśli istnieje stała $c > 0$ taka, że dla każdego $n \geq n_0$ mamy

$$|f(n)| \leq cg(n).$$

Algorytm czasu wielomianowego i wykładniczego

Definicja: Mówimy, że algorytm $\mathcal{A}$ działający na danych o liczbie bitów k jest czasu wielomianowego (wykładniczego), jeżeli istnieje stała $c > 0$ taka, że liczba operacji elementarnych na bitach potrzebnych do wykonania tego algorytmu jest rzędu $O(k^c)$ ($O(e^{ck})$).

Operacje elementarne na bitach, przykład

Zadanie: Niech $a, b \in \mathbb{N}$, gdzie $a \geq b$. Ile elementarnych operacji na bitach potrzeba do obliczenia $S(a, b) = a + b$?

Zadanie: Niech $a, b \in \mathbb{N}$, gdzie $a \geq b$. Ile elementarnych operacji na bitach potrzeba do obliczenia $I(a, b) = ab$?

Operacje elementarne na bitach, przykład

Przykład: Niech $a = (1011)_2, b = (1001)_2$, to

$$\begin{array}{cccccc} & 1 & 0 & 1 & 1 \\ + & 1 & 0 & 0 & 1 \\ \hline 1 & 0 & 1 & 0 & 0 \end{array}$$

Stąd, wykonujemy $O(\log a)$ operacji elementarnych na bitach.

Operacje elementarne na bitach, przykład

Przykład: Niech $a = (1011)_2, b = (1001)_2$, to

$$
\begin{array}{cccccccc}
 & & & & 1 & 0 & 1 & 1 \\
 & & & * & 1 & 0 & 0 & 1 \\
\hline
 & & & & 1 & 0 & 1 & 1 \\
 & & & 0 & 0 & 0 & 0 & \\
 & & 0 & 0 & 0 & 0 & & \\
+ & 1 & 0 & 1 & 1 & & & \\
\hline
\end{array}
$$

Stąd, wykonujemy $O(\log a)O(\log b) = O(\log^2 a)$ operacji elementarnych na bitach.

Asymetryczny Protokół Szyfrowania

ALICE, (Generowania kluczy)

1. Wybiera jawnie (E, D)
2. Generuje klucze (K_A, k_A) do (E, D)
3. Upublicznia K_A i zachowuje w sekrecie k_A

Asymetryczny Protokół Szyfrowania

BOB, (Szyfrowanie)

1. Pobiera K_A
2. Ustala M
3. Oblicza $E_{K_A}(M) = C$
4. Wysyła C do Alice

Asymetryczny Protokół Szyfrowania

Alice, (Deszyfrowanie)

1. Pobiera C
2. Oblicza $D_{k_A}(C) = M$