Paskaita 15

Elektroniniai parašas

Reikia pasirašyti įvairius elektroninius dokumentus:

a) kontraktus, b) banko transakcijos

c) elektroninio pašto laiškus su patvirtinimu

pvz. balsavimo dokumentus

Bendra schema:

1. Alice turi pasirašyti pranešimą m. Ji naudoja (kieno nors) viešojo rakto algoritmą ir su savo privačiu raktu d pasirašo dokumentą

$$s = S(d, m).$$

2. Bobas panaudodamas viešą raktą e gali patikrinti parašo teisingumą

$\tilde{m} = E(e, s)$ ar $\tilde{m} = m$?

Niekas negali generuoti s, neišnaudodamas privataus Alice rakto d.

RSA realizacija.

$n = pq$, e - viešasis raktas, (n, e)
d - privatus raktas

1. $S = m^d \bmod n$ - Alice pasirašė dokumentą

2. $m' = s^e \bmod n = m^{ed} \bmod n = m$.
Bobas gali patikrinti ar $m' = m$.

Pvz $p = 11$, $q = 23$, $e = 3$
$n = 253$, $d = 147$
$(n, e) = (253, 3)$

$m = 111 \Rightarrow S = 111^{147} \bmod 253 = 89$

$m = s^3 \bmod 253 = 89^3 \bmod 253 = 111.$

Atakos prieš elektroninį parašą.

1. Pakeisti (e, n) savo paskraktu (e', n')
Tada Bobas galvos, kad dokumentą pasirašo Alice, bet tai bus Oneteis parengtas ir pasirašytas dokumentas

2. Kiekvienam $s \in (0, 1, \dots, n-1)$ yra potencialus galiojantis parašas kažkokiam m. (Existential forgery)

Multiplikatyvumo savybė

Je leidžia gauti galiojančius (teisingus) parašus, (dokumentų, kurių turinio nežinome).

Imkime du Alice pasirašytus dokumentus

$$s_1 = m_1^{d} \bmod n \quad , \quad s_2 = m_2^{d} \bmod n$$

Tada nagrinėkime parašą

$$s = s_1 \cdot s_2 \bmod n.$$

Lichauų aritmetikoje

$$s = (m_1 m_2)^{d} \bmod n$$

Todėl, faip gavome parašą dokumento

$$m = m_1 m_2,$$

kurio Alice nepasirašinėjo.

Kaip pasinaudoti šia parašo savybe:

Sakykime Ignas turi vieną dokumentą, jau pasirašytą Alice (tai gali būti „nekaltas" verslo dokumentas)

(m_1, s_1).

Bet jam reikia gauti dokumento

m parašą s.

Jis Euklido algoritmu išsprendžia lygtį

$$m_1 m_2 = m \mod n.$$

Tada paprašo Alice pasirašyti dokumentą (m_2, s_2)

Turėdamas s_2 jis pats pasirašo jam reikalingą dokumentą

$$s = s_1 s_2 \mod n.$$

P.S. Aš pasirašinėjau dokumentus, kuriuos parengdavo įvairūs katedros darbuot. detaliai jų neskaitydavau

Parašas su pertekline informacija
(signature with redundancy)

Pasirenkame perteklines informacijos f-ją (redundancy function)

$$R: \{0,1\}^* \longrightarrow \{0,1\} \quad w \longrightarrow R(w)$$

(Nagrinėjame binarinę pranešimo formą)

pvz 1. $w \longrightarrow w \circ w$ (pakartojame pranešimą)

pvz. 2 $w \longrightarrow w \circ (\underbrace{0,0,\dots 0}_{k})$. k-bitų 0.

Imkime, kaip pavyzdį,

$m = w \circ w$. Alice ir pasirašo dokumentą m, nors tikrasis dokumentas yra w.

1) Tada jau labai sunku parinkti tokį $s \in \{0, ..., n-1\}$, kad $m = s^e \mod n$ būtų leistinos formos

2) Jeigu pasirenkame m_1 ir m_2
$m_1 = w_1 \circ w_1$, $m_2 = w_2 \circ w_2$
tai labai maža tikimybė, kad
$$m_1 m_2 = w_3 \circ w_3.$$

RSA šifravimo blokinis algoritmas

$\Sigma = \mathbb{Z}_N = \{0, 1, .., N-1\}$ alfabetas

$N = 26$ abecėlė

$n = pq$, p ir q yra pirminiai skaičiai

$$k = \lfloor \log_N n \rfloor$$

Imkime pranešimą
$$m = \sum_{i=1}^{k} m_i N^{k-i}, \quad m_i \in \mathbb{Z}_N$$

Patikrinkime, kad $0 \le m < n$, t.y. $m \in \mathbb{Z}_N$.

$$m = \sum_{i=1}^{k} m_i N^{k-i} \le (N-1) \sum_{i=1}^{k} N^{k-i}$$

$$= (N-1) \frac{N^k - 1}{N-1} = N^k - 1 < n,$$

nes $k = \lfloor \log_N n \rfloor$.

Blokas $m_1 m_2 \dots m_k \in \sum^k$ susideja-mas su atitinkamu sveiku skaičiumi m.

Šifruojame panaudodami RSA algor.

$$\boxed{c = m^e \bmod n}$$

Gautą skaičių $0 \le c < n$ užrašome N-skaičiavimo sistemoje. Kadangi

$$0 \le c < n < N^{k+1}$$ tai šio skleidinio ilgis yra nedidesnis už $\underline{k+1}$.

$$c = \sum_{i=0}^{k} c_i N^{k-i}, \quad c_i \in \sum, \ i = 0, \dots, k.$$

$$m \in \Sigma^{k} \longrightarrow c \in \Sigma^{k+1}$$

atvaizdavimas yra injekcija ($\forall$ vaizdas c turi tik vieną pirmvaizdį, ir atvaizdis yra apibrėžtas $\forall m \in \Sigma^{k}$).

Taigi formaliai tai nėra blokinis algoritmas, nes pradinio (plain) teksto bloko ilgis nesutampa su šifruoto teksto c (ciphertext) bloko ilgiu. Vadinasi reikia truputi šiek tiek modifikuoti ECB ir CBC algoritmus.

Pvz. $\Sigma = \{a, b, c, d\}$, atvaizduojame lentele $N = 4$

a	b	c	d
0	1	2	3

$$n = 253 = \underset{p}{11} \cdot \underset{q}{23}$$

$4^3 = 64$

$4^4 = 256$

$k = \lfloor \log_4 253 \rfloor = 3$

Taigi nešifruoto pranešimo bloko ilgis yra 3 ženklai.

Šifruoto pranešimo bloko ilgis yra $3+1 = 4$ ženklai.

Blokas bcc

Suskaičiuojame atitinkamą skaičių

$m = 1 \cdot 4^2 + 2 \cdot 4 + 2 \cdot 4^0 = 26$

Šifruotas tekstas $(e = 3)$

$C = 26^3 \bmod 253 = 119$

$\gcd(e, (p-1)(q-1)) = 1.$

$\gcd(3, 220) = 1.$

$119 = 1 \cdot 4^3 + 3 \cdot 4^2 + 1 \cdot 4 + 3 \cdot 1$

Taigi šifruoto teksto blokas

bdbd