Dėl koliokviumo, noriu atšaukti savo komentarą paskaitos įraše, atsiprašau.
Koliokviumo rezultatai jau yra įvesti į AIS.

Kas dar nepasirinkote, prašau pasirinkti Stendinio Pranešimo temą:
https://docs.google.com/document/d/1Bp8NW2e4AbIsZ5UuBqa8BxkK6py01rjV/edit?usp=sharing&ouid=111502255533491874828&rtpof=true&sd=true

Stendinio Pranešimo reikalavimai pateikti čia:
https://docs.google.com/document/d/1IPzwEVVmvnObQoJPRP9GDUiHccEGWgMw/edit?usp=sharing&ouid=111502255533491874828&rtpof=true&sd=true

Coin Flipping

Coin flipping, **coin tossing**, either **heads** or **tails** is the practice of throwing a coin in the air and checking which side is showing when it lands, in order to choose between two alternatives, heads or tails, sometimes used to resolve a dispute between two parties.
It is a form of sortition which inherently has two possible outcomes.
The party who calls the side that the coin lands on wins.
From <https://en.wikipedia.org/wiki/Coin_flipping>







Dice throwing



Card game - Poker



A: $PrK_A = x$, $PuK_A = a$; b, e; $a = g^x \bmod p$

B: $PrK_B = y$, $PuK_B = b$; a, e; $b = g^y \bmod p$;

E: $PrK_C = z$, $PuK_C = e$; a, b; e = ---

ElGamal encryption

PP=(***p***, ***g***) **>> p = 268 435 019**; % 2^28 -1 --> >> int64(2^28-1)

% ans = **268 435 455**

>> g=2;

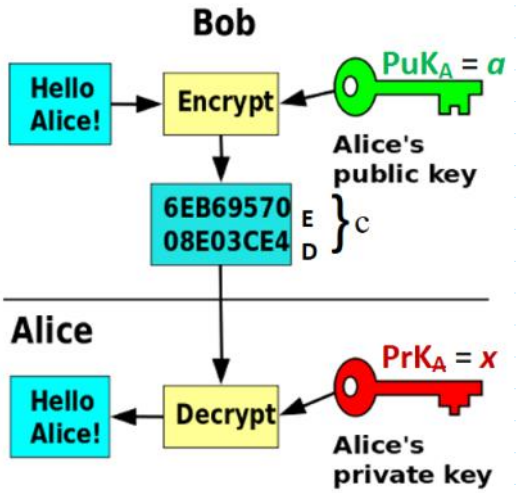
$m \in \mathcal{L}_p^* = \{1, 2, 3, \dots, p-1\}$; * mod p

message to be encrypted

$i \leftarrow$ randi ; $i \in \mathcal{L}_{p-1} = \{0, 1, 2, \dots, p-2\}$

$c = Enc(a, i, m) =$

$= (E, D) = (m a^i \bmod p, \underbrace{g^i \bmod p}_{D})$



$Dec(x, c) = E \cdot D^{-x} \bmod p = \frac{E}{D^x} \bmod p =$

$= \frac{m a^i \bmod p}{(g^i)^x} = \frac{m (g^x)^i \bmod p}{g^{ix}} = m \bmod p = m$ ← if $m < p$

D^{-x} mod ***p*** computation using Fermat theorem:

If ***p*** is prime, then for any integer ***a*** holds $a^{p-1} = 1 \bmod p$.

$$D^{-x} \bmod (p-1) \bmod p = D^{p-1-x} \bmod p$$

$D^{-x} \bmod p$ computation

a) D^{-1} computation : >> D_m1=mulinv(D, p)

b) D^{-x} computation : $(D^{-1})^x = D^{-x}$ >> D_mx=mod_exp(D_m1,x, p)

A : $PrK_A = x$; $PuK_A = a$; $PuK_B = b$;

$m_i \in \{1, 2\}$

Coin flipping scheme: ***Alice*** before coin flipping assigns possible results to variables m_1=1 and m_2=2.

1. ***Alice*** after coin flipp assigns result *m* either to *m*=1 or *m*=2.
2. ***Alice*** encrypts m_1=1 and m_2=2 by her **PuK=*a*** using random generated numbers *i*1 and *i*2 computing ciphertexts c_{1A} and c_{2A} respectively:

$m_i \in \{1, 2\}$

$i_1, i_2 \leftarrow randi(\mathcal{L}_{p-1})$

$c_{1A} = Enc(a, i_1, m_1) = (E_{1A}, D_{1A})$

$c_{2A} = Enc(a, i_2, m_2) = (E_{2A}, D_{2A})$

$E_{1A} = m_1 \cdot a^{i_1} \bmod p$; $D_{1A} = g^{i_1} \bmod p$
$E_{2A} = m_2 \cdot a^{i_2} \bmod p$; $D_{2A} = g^{i_2} \bmod p$

$\xrightarrow{C_{1A}, C_{2A}}$ B

A:

B: $PrK_B = y$; $PuK_B = b$.

$C_{2A} \leftarrow rand\{C_{1A}, C_{2A}\}$; $C_{iA} = C_{2A}$

$i_3 \leftarrow randi(\mathbb{Z}_{p-1})$

$Enc(b, i_3, E_{2A}) = (E_{2AB}, D_{2AB}) = C_{2AB}$

$= (E_{2A} \cdot b^{i_3} \bmod p,\ g^{i_3} \bmod p)$

E_{2AB} , D_{2AB}

$\xleftarrow{E_{2AB}}$ C_{2AB}

$Dec(x, C_{2AB}) = \frac{E_{2AB}}{(D_{2A})^x} = E_{2AB} \cdot (D_{2A})^{-x} =$

$= \frac{E_{2A} \cdot b^{i_3}}{(g^{i_2})^x} = \frac{m_2 \cdot a^{i_2} \cdot b^{i_3}}{g^{i_2 x}} =$

$= \frac{m_2 \cdot a^{i_2} \cdot b^{i_3}}{g^{i_2 x}} = \frac{m_2 \cdot g^{x i_2} \cdot b^{i_3}}{g^{i_2 x}} =$

$= m_2 \cdot b^{i_3} = E_{2ABA}$ $\xrightarrow{E_{2ABA}}$ B: $C_{2ABA} = (E_{2ABA}, D_{2AB})$

① Let B guessed that A tossed C_{2A}

$Dec(y, C_{2ABA}) = E_{2ABA} \cdot (D_{2AB})^{-y} =$

$= \frac{E_{2ABA}}{(D_{2AB})^y} = \frac{m_2 \cdot b^{i_3}}{(g^{i_3})^y} = \frac{m_2 \cdot (g^y)^{i_3}}{g^{i_3 y}} =$

$= \frac{m_2 \cdot g^{y i_3}}{g^{i_3 y}} = m_2$

g^{i_3}

m_2, i_3

$$m = E_{2ABA} \cdot (b)^{-i_3} \bmod p =$$

$$m_2 \cdot b^{i_3} \cdot b^{-i_3} = m_2 \cdot b^{i_3 - i_3} =$$

$$= m_2 \cdot b^0 = m_2 \cdot 1 = m_2$$

$m_2 \bmod p = m_2$ if $m_2 < p$

$i_2, \{1, 2\}$ $\mathcal{B}$: $C_{2A} = (E_{2A}, D_{2A})$

$$E_{2A} \cdot a^{-i_2} \bmod p =$$

$$= m_2 \cdot a^{i_2} \cdot a^{-i_2} \bmod p =$$

$$= m_2 \cdot a^{i_2 - i_2} = m_2 \cdot a^0 = m_2$$

---

(2) Let $\mathcal{B}$ choosed that $\mathcal{A}$ tossed $C_{1A} = (m_1 \cdot a^{i_1}, g^{i_1})$

$\mathcal{B}$ did not guess the toss.

$i_3 \leftarrow \text{randi}(\mathbb{Z}_{p-1})$

$$Enc(b, i_3, E_{1A}) = (E_{1AB}, D_{1AB}) = C_{1AB}$$

$$= (\underbrace{E_{1A} \cdot b^{i_3} \bmod p}_{E_{1AB}}, \underbrace{g^{i_3} \bmod p}_{D_{1AB}})$$

E_{1AB}

C_{1AB}

$$Dec(x, C_{2AB}) = \frac{E_{1AB}}{(D_{2A})^x} =$$

$$= \frac{E_{1A} \cdot b^{i_3}}{(g^{i_2})^x} = \frac{m_1 \cdot a^{i_1} \cdot b^{i_3}}{g^{i_2 x}} =$$

$$= \frac{m_1 \cdot a^{i_1} \cdot b^{i_3}}{g^{i_2 x}} = \frac{m_1 \cdot g^{x i_1} \cdot b^{i_3}}{g^{i_2 x}} =$$

E_{1AB}

$\mathcal{B}$:

g^{i_2} ^ g^{i_2} ^

$= E_{1ABA}$ $\xrightarrow{E_{1ABA}}$ B:

$Dec(y, c_{1ABA}) =$

$= \frac{E_{1ABA}}{(D_{1AB})^y} = m'$

$E_{1ABA} \cdot (b)^{-i_3} \bmod p = m''$ $\xleftarrow{m', i_3}$

$\xrightarrow{m'', i_2 \{1,2\}}$ B: $Dec(y, m'')$

$E_{1A} \cdot a^{-i_2} \bmod p = u \notin \{1,2\}$

If random generated number i in ElGamal encryption is revealed then ciphertext can be decrypted.

$Enc(a, i, m) = (E, D) = (m \cdot a^i \bmod p, g^i \bmod p) = c$

Decryption without knowledge $Pr.K = x$ but having i:

$E \cdot a^{-i} \bmod p = m \cancel{a^i} \cancel{a^{-i}} \bmod p = m \bmod p = m.$

m mod p = m
if m < p

Dice throwing

Poker

3 x 2

$\rightarrow \equiv 21$

$m \in \{1, 2, 3, 4, 5, 6, \dots$

$r_1 \leftarrow randi, \dots, r_6 \leftarrow randi$

$c_i = Enc(a, r_i, m_i), i = 1, 2, 3, \dots$

$c_1 \equiv 1;\ c_2 \equiv 2;\ c_3 \equiv 3; \dots c_6 = \dots$ $\xrightarrow{}$ B: $c_i \leftarrow rand\{c_i\}$ $\xrightarrow{}$ C: $c_i = c_6$

$c_{ij} = Enc(a, r_{ij}, m_{ij})$ $c_{ij} \leftarrow rand\{c_{ij}\}$

$i =$ 1, 2, 3, ... kauliuko reikšmės

$j = \overline{1,6}$ kauliuko numeris

Card game - Poker

52 kortos & 4 mostis

1 kortos šifrav.

$c_i = Enc(a, r_i, m_i)$; $i = \overline{1,4}$.

$c_{ij} = Enc(a, r_{ij}, m_{ij})$; $j = \overline{1,52}$.

Ring-Signature introduction

Rivest, Ronald L., Adi Shamir, and Yael Tauman. "How to leak a secret." *International Conference on the Theory and Application of Cryptology and Information Security*. Springer, Berlin, Heidelberg, 2001.

The general notion of a **group signature** scheme was introduced in 1991 by Chaum and van Heyst [2].
In such a scheme, a trusted group ***manager*** predefines certain groups of users and distributes specially designed keys to their members.
Individual members can then use these keys to anonymously sign messages on behalf of their group.
The signatures produced by different group members look indistinguishable to their verifiers, but not to the group ***manager*** who can revoke the anonymity of misbehaving signers.
In this paper we formalize the related notion of **ring signature** schemes.
These are *simplified group signature* schemes which have only users and **no managers**
(we call such signatures "***ring signatures***" instead of "***group signatures***" since
rings is some uniform community and no center).
Group signatures are useful when the members want to cooperate, while **ring signatures** are useful when the members do not want to cooperate.
Both **group signatures** and **ring signatures** are signer-ambiguous, but in a **ring signature** scheme there are no **prearranged groups of users**, there are **no procedures for setting**, changing, or deleting groups, there is **no way to distribute specialized keys**, and there is **no way to revoke the anonymity of the actual signer** (unless he decides to expose himself).
Our only assumption is that each member is already associated with the **public key** of some standard signature scheme such as RSA or ECDSA.
To produce a **ring signature**, the actual signer declares an arbitrary set of possible signers that includes himself, and computes the signature entirely by himself using only his **private key** and the others' **public keys**.
In particular, the other possible signers have their **private keys** and can to conduct e-commerce over the internet, and may be completely unaware that their **public keys** are used by a stranger to produce such a **ring signature** on a message they have never seen and would not wish to sign.

Terminology: We call a set of possible signers a **ring**. We call the ring member who produces the actual signature the ***signer*** and each of the other ring members a ***non-signer***.

A **ring signature** scheme is set-up free: The ***signer*** does not need the knowledge, consent, or assistance of the other ring members to put them in the ring - all he needs is knowledge of their regular **public keys**.
Different members can use different independent **public key** signature schemes, with different key and signature sizes. *Size of **ring** signature depends of the number of **ring members**.*
Verification must satisfy the usual soundness and completeness conditions, but in addition we want the signatures to be signer-ambiguous in the sense that the verifier should be ***unable*** to determine the ***identity*** of the actual ***signer*** in a ring of size ***r*** with probability greater than 1/***r***.
This limited anonymity can be either computational or unconditional.
Our main construction provides unconditional anonymity in the sense that even an infinitely powerful adversary with access to an unbounded number of chosen-message signatures produced by the same ring

member cannot guess his identity with any advantage, and cannot link additional signatures to the same signer.