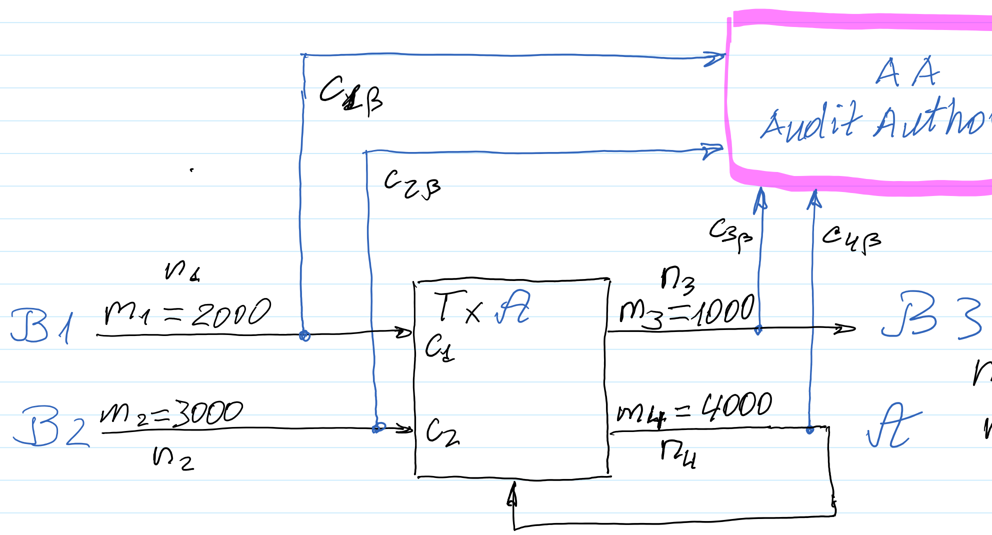
Agregate BLS signatures

New approach



$PrK = z$

$PuK = \beta$

$m_1 + m_2 = m_3 + m_4 = 5000$

$n_1 \cdot n_2 \bmod p = n_3 \cdot n_4 \bmod p$

1. Taxes declaration to AA.
2. To prove to the Net, that transaction is honest.

B_1: $Enc_\beta(l_1, n_1) = C_{1\beta} = (E_{1\beta}, D_{1\beta})$

B_2: $Enc_\beta(l_2, n_2) = C_{2\beta} = (E_{2\beta}, D_{2\beta})$

$\to$ AA

$Enc_a(r_1, n_1) = C_1 = (E_1, D_1)$

$Enc_a(r_2, n_2) = C_2 = (E_2, D_2)$

$\to$ A

$Dec_z(C_{1\beta}) = n_1 \to$ computes m_1

$Dec_z(C_{2\beta}) = n_2 \to$ computes m_2

$Dec_x(C_1) = n_1 \to$ computes m_1

$Dec_x(C_2) = n_2 \to$ computes m_2

A: 1) $C_{12} = C_1 \cdot C_2 \longrightarrow$ Net

2) encrypts $C_{3\beta} = Enc_\beta(r_3, n_3) = (E_{\beta 3}, D_{\beta 3})$

encrypts $C_{4\beta} = Enc_\beta(r_4, n_4) = (E_{\beta 4}, D_{\beta 4})$

$C_{34\beta} = C_{3\beta} \cdot C_{4\beta} \longrightarrow$ Net

C_{12} and $C_{34\beta}$ encrypted the same data: $n_{12} = n_1 \cdot n_2 \bmod p$

$n_{34} = n_3 \cdot n_4 \bmod p$

but with the different PuK, namely n_{12} with $PuK_A = a$

n_{34} with $PuK_{AA} = \beta$.

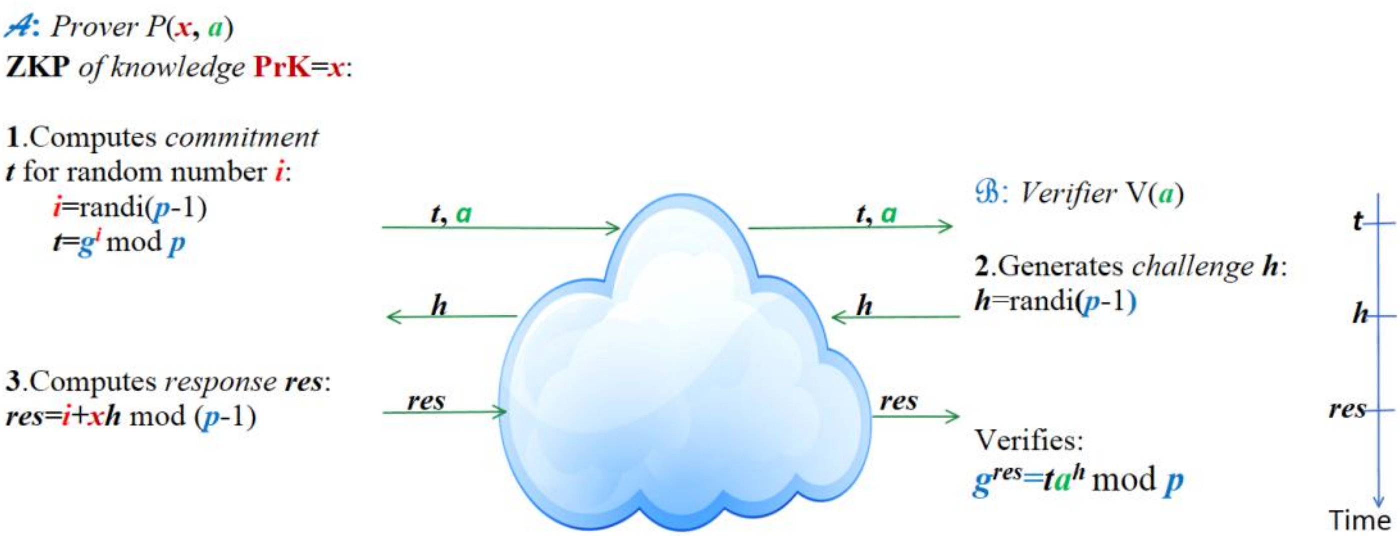
Therefore $c_{12} \neq c_{34\beta}$

A must prove that ciphertexts c_{12} and $c_{34\beta}$ encrypted the same value $n_{Bal} = n_{12} = n_{34}$ → Net.

Proof of two ciphertexts equivalency.

Schnorr Identification: Zero Knowledge Proof - ZKP **PP** = (p, g).

Schnorr Id Scenario: **Alice** wants to prove **Net** that she knows her Private Key - **PrK_A** = x which corresponds to her Public Key - **PuK_A**= a = g^x **mod** p not revealing **PrK_A**= x.



Non-Interactive Zero Knowledge Proof - NIZKP **PP** = (p, g).

NIZKP Scenario: **Alice** wants to prove **Net** that she knows her Private Key - **PrK_A** = x which corresponds to her Public Key - **PuK_A**= a = g^x **mod** p not revealing **PrK_A**= x and using non-interactive protocol.

Alice chooses at random u, $1<u<p-1$ and computes number r:

$$r=g^u \bmod p. \qquad (2.19)$$

Alice computes H-function value h of the number r:

$$h=H(r), \qquad (2.20)$$

Alice computes value s:

$$s=u+xh \bmod (p-1). \qquad (2.21)$$

Alice declares the values $\pi=(r, s)$ to the **Net**.

Net according to (2.20) computes h and verifies if:

$$\underbrace{g^s \bmod p}_{V1} = \underbrace{ra^h \bmod p}_{V2}. \qquad (2.22)$$

Symbolically this verification function we denote by

$$\mathbf{Ver}(a,\pi,h)=V\in\{True, False\}\equiv\{1, 0\}. \qquad (2.23)$$

This function yields ***True*** if (2.22) is valid and if: $\mathbf{PuK_A} = a = F(\mathbf{PrK_A}) = g^x \ \mathbf{mod}\ p$.

Correctness:

$g^s \bmod p = g^{u+xh \bmod (p-1)} \bmod p = g^u g^{xh} \bmod p = r(g^x)^h \bmod p = r a^h \bmod p$.