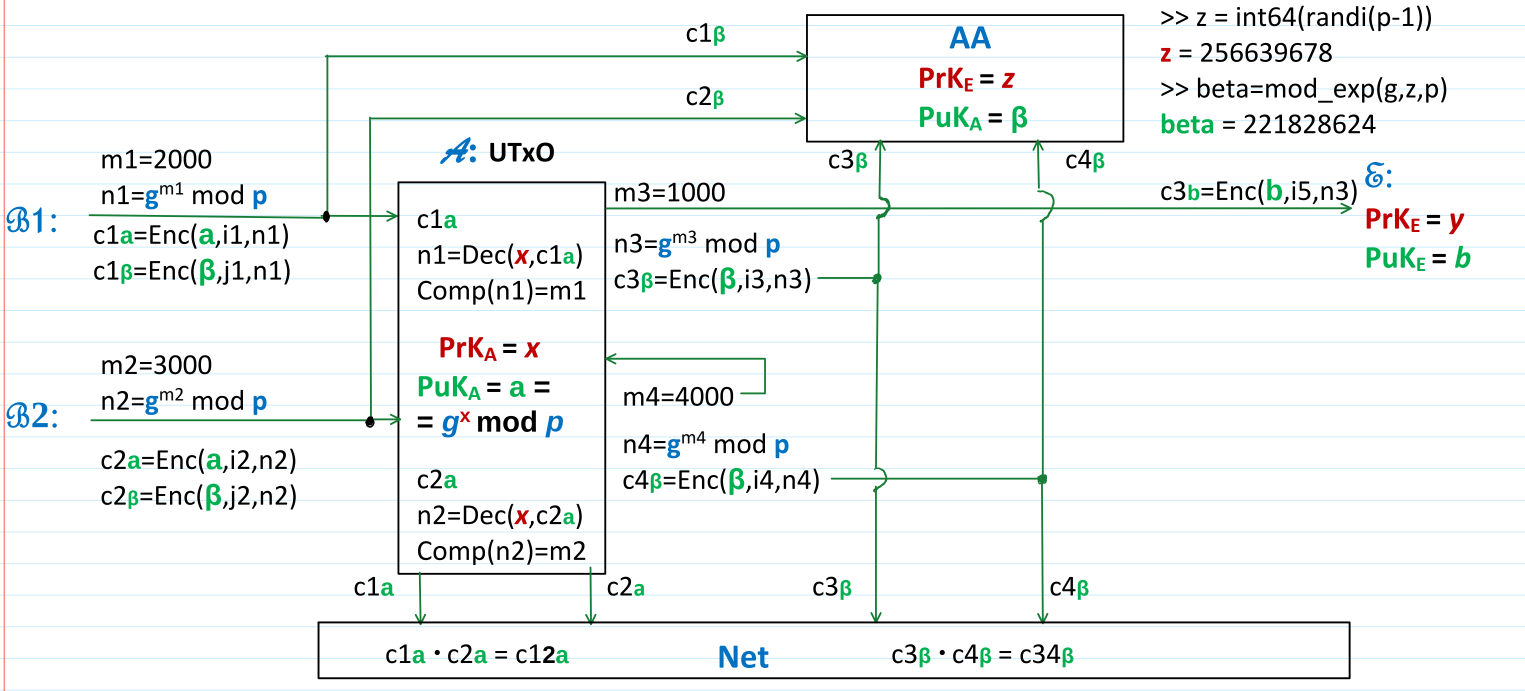
Confidential Verifiable Transactions - 3. Anonymization $PP = (p, g)$.



$c12a = Enc(a, i1+i2, n1 \cdot n2)$

$\quad = Enc(a, i1+i2, \mathbf{n12})$

$n12 = n1.n2 \bmod p$

$\quad = g^{(m1+m2) \bmod (p-1)} \bmod p$

$c34\beta = Enc(\beta, i3+i4, n3 \cdot n4)$

$\quad = Enc(\beta, i3+i4, \mathbf{n34})$

$n34 = n3.n4 \bmod p$

$\quad = g^{(m3+m4) \bmod (p-1)} \bmod p$

Since $n1=g^{m1} \bmod p$ and $n2=g^{m2} \bmod p$, and $n3=g^{m3} \bmod p$ and $n4=g^{m4} \bmod p$ then
if $m1+m2 = m3+m4$ then $\mathbf{n12} \bmod p = n1 \cdot n2 \bmod p = n3 \cdot n4 \bmod p = \mathbf{n34} \bmod p$.

Fig. 1. Private and verifiable transactions using ElGamal Cryptosystem.

ElGamal Encryption:

$\mathcal{B}: \; i \leftarrow randi(\mathcal{I}_p^*)$

$$\left.\begin{array}{l} E = m \cdot a^i \bmod p \\ D = g^i \bmod p \end{array}\right\} c = (E, D)$$

ElGamal Decryption:

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

$m = ED^{-x} \bmod p;$

Let us compare the number of most resources consuming operations required to realize the private and verifiable transactions using ElGamal Cryptosystem (EGC) and Elliptic Curve Cryptosystem (ECC) schemes for only 1 sender **Bob1** and 1 receiver-sender **Alice**.

In ElGamal Cryptosystem such operation is Discrete Exponent Function (DEF), e.g. of the form $\mathbf{a} = g^x \bmod p$ or exponentiation.

In Elliptic Curve Cryptosystem (ECC) such operation is Elliptic Curve point G multiplying by integer z, e.g. $A = z*G$ which we name as EC exponentiation: encryption is replaced by Pedersen Commitment.

We assume that these operations are almost equivalent.

1. **EGC operations**.

1.1. **Bob1** performs 2 encryptions: 1 for Alice and 1 for AA. Hence the number of exponentiations is 4.
We do not take into account the exponentiation for computing n1 since the number m1 is considerable small, and Comp(n1)=m1 since Alice knows the approximate sum of m1.

1.2. **Alice** performs 1 decryption for income from **Bob1** - 1 exponentiations and 2 encryptions for expenses: 1 for Ema and 1 for AA requiring 4 exponentiations. Hence **Alice** performs 5 exponentiations.

1.3. To proof the equivalence of ciphertexts $\mathbf{c}_{12\alpha}$ and $\mathbf{c}_{34\beta}$ it is required to perform 4 exponentiations.

In total it is required to perform **9 exponentiations** for **Alice**.

Alice computations to prove that transaction is honest, i.e. that 2 ciphertexts are obtained by encryption the same sum of incomes and expenses by different public keys α and β are equivalent.

The following commitments $\{t_1, t_2, t_3\}$ are computed:

$$t_1 = g^u \mod p$$

$$t_2 = g^v \mod p$$

$$t_3 = (D_{12\alpha})^u \cdot \beta^{-v} \mod p$$

This requires to perform 4 exponentiations.

Net verifies transaction correctness by verifying the following identities

$$g^r = a^h \cdot t_1 \mod p$$ // A proves that she knows her PrK = x

$$g^s = (D_{34\beta})^h \cdot t_2 \mod p$$ // A proves that she knows her random parameter i_{34} used for encryption

$$(E_{34\beta})^h \cdot (E_{12\alpha})^{-h} \cdot (D_{12\alpha})^r \cdot \beta^{-s} = t_3 \mod p$$

A proves that based on her knowledge of x and i_{34}, the ciphertexts $c_{12\alpha}$ and $c_{34\beta}$ are equivalent.

In 2024.11 Donal Trump declared America to be a Bitcoin country.
Possibly inspired by Elon Musk.

Anonymity in Blockchain

Let **Alice** opened her Bitcoin account with Bitcoin Address by generating her private key **PrK**=x and public key **PuK**=α.
We assume that **PuK**=α are linked to **Alice** Aaddress in Bitcoin.

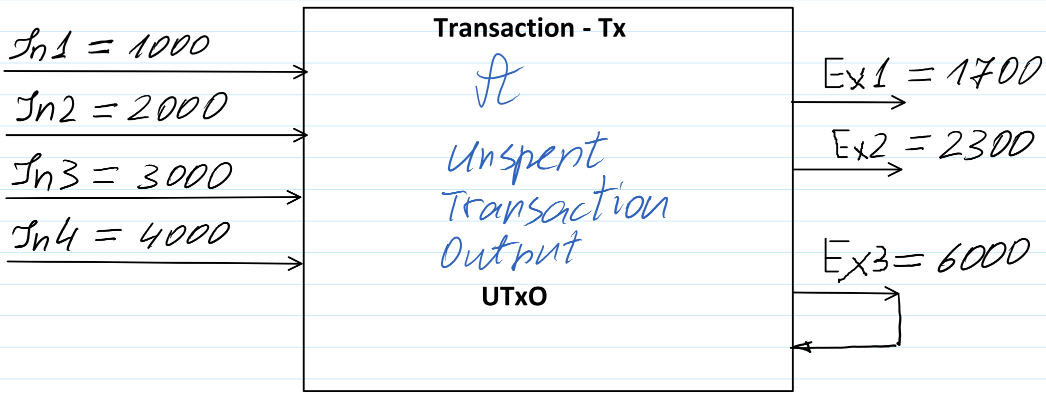
In Bitcoin and other Blockchains the Address is computed as a function of user's public key:
$\mathbf{Addr_A}$ = **F**(**PuK**) and consist of several dozens of decimal numbers.

Cryptocurrency transaction

No.	Pajamos-Incomes	Išlaidos-Expenses	Likutis-Balance
In1.	Client1: 1000 Sat		1000 Sat
In2.	Client2: 2000 Sat	Out1. Firm 5: 1700 Sat	1300 Sat
In3.	Client3: 3000 Sat	Out2.t Firm 6: 2300 Sa	2000 Sat
In4.	Client4: 4000 Sat	Out3. Firm 7:	6000 Sat
Total	**10 000 Sat**	**4000 Sat**	**6000 Sat**

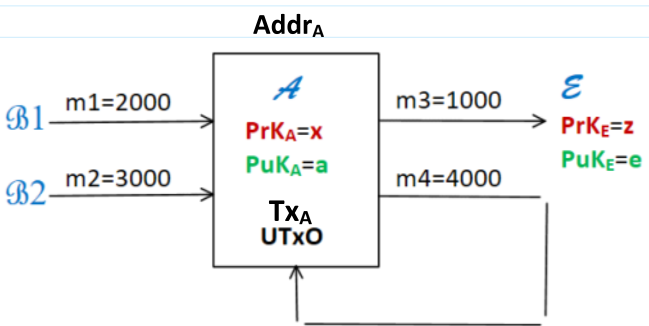
Sum of Inputs =
= Sum of Outputs
Divisibility

Unspent Transaction Output - **UTxO** paradigm



Transaction (Tx) information in simplified form consist of the following information:

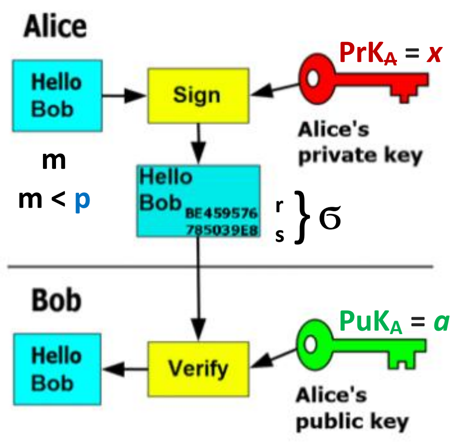
1. The address of Tx creator.
2. The sums of Incomes and addresses of senders.
3. The sums of Expenses and addresses of receivers.



Alice has a certificate **Cert$_A$** for her PuK$_A$ = a **= g^x mod p**.

Schnorr Signature

In the case of Schnorr cryptosystem our simulation is performed with Public Parameters:
PP = (p, g); **p=268435019; g=2;** p=int64(268435019)
By having PP private key PrK and public key PuK are generated:
PrK = x <-- randi(p-1)
PuK = a = g^x **mod** p.



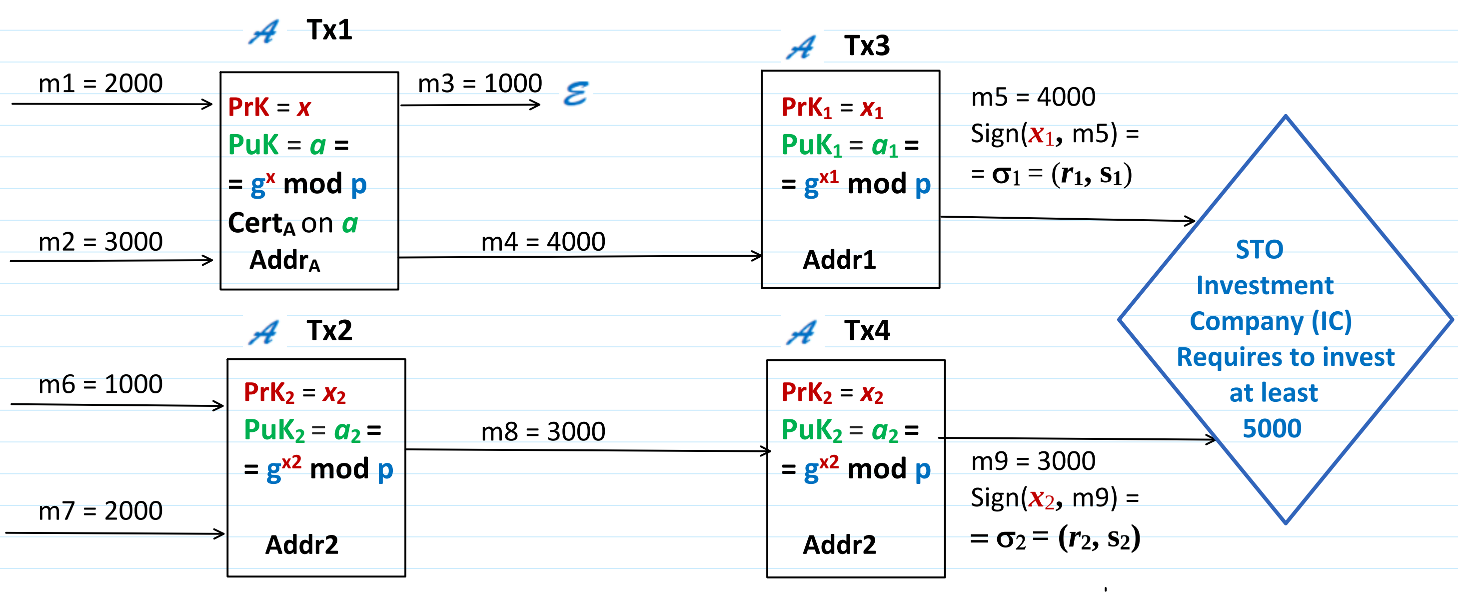
u <-- randi(p-1).
$r = g^u \bmod p$.
$h = H(M||r)$. >> con=concat(M,r)
>> h=hd28(con)
$s = u + xh \bmod (p-1)$. (*) >> s=mod(u+x*h,p-1)
Alice's signature on h is $\sigma = (r, s)$.

Notice that it is infeasible to find x from (*), when
s and h are given, since there is 1 equation (*) and 2 unknowns u and x.

Signature is valid if: $\underbrace{g^s \bmod p}_{V1} = \underbrace{r a^h \bmod p}_{V2}$. (Eq.1)

But **Alice** do not want that all her incomes belonging to her Address were known and therefore and she prefers to be anonymous to the **Net**.
Then she creates a set of Addresses by generating a set of private keys $\{PrK_i = x_i\}$ and a set of public keys $\{PuK_i = a_i\}$, where i=1, 2, ..., N.

But! There are the situations when **Alice** must prove some subjects that she possesses some amount of money distributed among a lot of her accounts and transactions with different addresses.
For example, she could pretend to tax concessions - (mokesčiu lengvatos) (according to the law) and she must prove to certain Investment Company that she possesses sufficient amount of money.
In this case she must prove that she controls some accounts with this sufficient amount of money for investment.
In this case **Alice** must prove that her transactions are authentic (i.e. are created by her) by proving that **PuK**=a belongs to her, e.g. using Certificate issued by Certificate Authority for **PuK**=a, but at the same time she remains **anonymous** for other part of the **Net**.



In Monero blockchain for anonymization Alice is using Ring Signature, instead procedure presented above.
It is interresting to compare the realization effectivity of procedure presented above and procedure based on Ring Signature.

Compare realization effectivity of DEF Schnorr multisignature with ECC ring signature computing the number of Discrete Exponent Function Operations - DEFO: $a = g^u \bmod p$

Schnorr-Multi-Signature
Anonymization in BlockChain

Anonymous Group of Signers (**GoS**) must sign on different transactions with different private keys.
In this case the group consist of 2 anonymous addresses **Addr1** and **Addr2** belonging to **Alice**.
Let the **GoS** is: $\{S_1; S_2\}$.
All members of **GoS** have their private and public keys:

S_1;	S_2;
$\mathbf{PrK_1}=x_1$, $\mathbf{PuK_1}=a_1$;	$\mathbf{PrK_2}=x_2$, $\mathbf{PuK_2}=a_2$;
u_1 <-- randi(p-1);	u_2 <-- randi(p-1);
$r_1=g^{u_1}$ **mod** p;	$r_2=g^{u_2}$ **mod** p;
$h_1=\mathrm{H}(Tx3\|\|r_1)$;	$h_2=\mathrm{H}(Tx4\|\|r_2)$;
$s_1=u_1+x_1h_1 \bmod (p-1)$;	$s_2=u_2+x_2h_2 \bmod (p-1)$;

$\sigma_1 = (r_1, s_1)$. $\qquad\qquad$ $\sigma_2 = (r_2, s_2)$.

How to join signatures $\sigma_1 = (r_1, s_1)$ and $\sigma_2 = (r_2, s_2)$ to the one signature $\sigma_P = (r_P, s_P)$.
Schnorr multisignature solves this problem.
Individual Schnorr signatures are multiplied by the special multiplication operation.

$\sigma_{12} = \sigma_1 * \sigma_2 = (r_1, s_1)*(r_2, s_2) = (R_{12}, S_{12})$.
$R_{12} = r_1 * r_2 \bmod p = g^{u_1} * g^{u_2} \bmod p = g^{u_1+u_2 \bmod (p-1)} \bmod p$.
$S_{12} = s_1+s_2 \bmod (p-1) = [(s_1=u_1+x_1h_1) + (s_2=u_2+x_2h_2)] \bmod (p-1) = u_1 + x_1h_1 + u_2 + x_2h_2 \bmod (p-1)$.

GoS signature verification:

$$\underbrace{g^{S_{12}} \bmod p}_{V1} = R_{12} * \underbrace{(a_1)^{h_1} * (a_2)^{h_2} \bmod p}_{V2}. \quad \text{(Eq.2)}$$

Compare it with a single Schnorr signature verification in (Eq. 1)

$$\underbrace{g^s \bmod p}_{V1} = \underbrace{r a^h \bmod p}_{V2}. \quad \text{(Eq.1)}$$

Correctness:

$g^{S_{12}} \bmod p = g^{(s_1+s_2) \bmod (p-1)} \bmod p = g^{s_1 \bmod (p-1)} * g^{s_2 \bmod (p-1)} \bmod p = g^{(u_1+x_1 h_1) \bmod (p-1)} * g^{(u_2+x_2 h_2) \bmod (p-1)}$
mod p =

$= r_1 * (a_1)^{h_1} * r_2 * (a_2)^{h_2} \bmod p =$
$= r_1 * r_2 * (a_1)^{h_1} * (a_2)^{h_2} \bmod p =$
$= R12 * a_1^{h_1} * a_2^{h_2} \bmod p$.

Compare with Pedersen Commitment.

Till this place

$g^r = g^{xh+u} = g^{xh} \cdot g^u = (g^x)^h \cdot g^u = a^h \cdot t_1 \bmod p$; $\qquad$ $a = g^x \bmod p$

$g^r = g^{xh + u} = g^{xh} \cdot g^u = (g^x)^h \cdot g^u = a^h \cdot t_1 \bmod p$;

$g^s = g^{i34*h + v} = g^{i34*h} \cdot g^v = (g^{i34})^h \cdot g^v = (D_{34\beta})^h \cdot t_2 \bmod p$;

$a = g^x \bmod p$

$(\underbrace{n_{34} \cdot \beta^{i_{34}}}_{E_{34\beta}}, \underbrace{g^{i_{34}}}_{D_{34\beta}}) = c_{34\beta}$

$(E_{34\beta})^h = (n34 \cdot \beta^{i34})^h = (n34)^h \cdot (D_{34\beta})^h \bmod p$.

$(E_{12a})^{-h} = (n12 \cdot a^{i12})^{-h} = (n12)^{-h} \cdot a^{-(i12*h)} \bmod p$;

$(D_{12a})^r = (g^{i12})^r = (g^{i12*x*h + i12*u}) = (g^x)^{i12*h} \cdot (g^{i12})^u = a^{h*i12} \cdot (g^{i12})^u = a^{i12*h} \cdot (D_{12a})^u \bmod p$;

$(E_{12a}, D_{12a}) = (n_{12} \cdot a^{i_{12}}, g^{i_{12}}) = c_{12a}$

$r = (x \cdot h + u) \bmod (p-1)$

$s = (i_{34} \cdot h + v) \bmod (p-1)$

$\beta^{-s} = \beta^{-i34*h - v} = \beta^{-i34*h} \cdot \beta^{-v} = (D_{34\beta})^{-h} \cdot \beta^{-v} \bmod p$;

$$(E_{34\beta})^h \cdot (E_{12a})^{-h} \cdot (D_{12a})^r \cdot \beta^{-s} \bmod p ===$$

$$=== \overbrace{(n34)^h \cdot (D_{34\beta})^h}^{(E_{34\beta})^h} \cdot \overbrace{(n12)^{-h} \cdot a^{-(i12*h)}}^{(E_{12a})^{-h}} \cdot \overbrace{a^{i12*h} \cdot (D_{12a})^u}^{(D_{12a})^r} \cdot \overbrace{(D_{34\beta})^{-h} \cdot \beta^{-v}}^{\beta^{-s}} \bmod p ===$$

If balance equation is valid, then $n34 = n12 = n \bmod p$ then $(n34)^h = (n12)^{-h} = n^{-h} \bmod p$ and $(n34)^h \cdot (n12)^{-h} = n \cdot n^{-h} = 1 \bmod p$.

$$=== \underbrace{(n34)^h \cdot (n12)^{-h}}_{1} \cdot (D_{12a})^u \cdot \beta^{-v} \bmod p ===$$

$$=== 1 \cdot (D_{12a})^u \cdot \beta^{-v} === (D_{12a})^u \cdot \beta^{-v} = \boldsymbol{t_3}.$$

The correctness of (30), (31) is proved by the following identities:

$$g^r = g^{xh + u} = g^{xh} \cdot g^u = (g^x)^h \cdot g^u = a^h \cdot t_1; \quad (33)$$

$$g^s = g^{lh + v} = g^{lh} \cdot g^v = (g^l)^h \cdot g^v = (\delta_{\beta,E})^h \cdot t_2. \quad (34)$$

The correctness of (32) is proved by considering every multiplier separately:

$$(\varepsilon_{\beta,E})^h = (E \cdot \beta^l)^h = E^h \cdot \beta^{lh}; \quad (35)$$

$$(\varepsilon_{a,I})^{-h} = (I \cdot a^k)^{-h} = I^{-h} \cdot a^{-kh}; \quad (36)$$

$$(\delta_{a,I})^r = (g^k)^r = (g^{kxh + ku}) = (g^x)^{hk} \cdot (g^k)^u = a^{hk} \cdot (g^k)^{\mathrm{u}} = a^{hk} \cdot (\delta_{a,I})^u; \quad (37)$$

$$\beta^{-s} = \beta^{-lh - v} = \beta^{-lh} \cdot \beta^{-v}. \quad (38)$$

Notice that k is not known to Alice and is included in $(\delta_{a,I})$. If the transaction is honest, then the transaction balance (1) is satisfied and $I=E$ since. Then $E^h \cdot I^{-h} = 1 \bmod p$, and putting it all together, we obtain:

$$E^h \cdot \beta^{lh} \cdot I^{-h} \cdot a^{-kh} \cdot a^{hk} \cdot (\delta_{a,I})^u \cdot \beta^{-lh} \cdot \beta^{-v} = (\delta_{a,I})^u \cdot \beta^{-v} = t_3. \quad (39)$$

This is the proof to the Net that the balance equation (1) is valid.