PEDERSON COMMITMENT

A cryptographic commitment scheme

We have two generators g and h of group G $|G| = q$

$G = \{ g^i \% q \mid i \in [0; q-1] \} = \{ h^i \% q \mid i \in [0; q-1] \}$ (Rappel q and p are large primes.)

Only order differs but it doesn't matter for sets.

We have $C = g^r \cdot h^\pi \% q \mid C \in \mathbb{Z}_q \wedge r, \pi \in \mathbb{Z}_q$ (on gros % q sur r, π ici ils dépassent q)

$C \equiv$ Commitment

$\pi \equiv$ Random Binding Factor $\Rightarrow$ Perfect Hiding : Thanks to random π (doesn't reveal r nothing about

$\Rightarrow$ Computationally binding : can find another π', r' such that $C = g^{r'} \cdot h^{\pi'} \% q$ without DL

$r \equiv$ Committed value

$g, h \equiv$ Generators of group G $\Rightarrow$ If Alice is only g could say $h = g^\pi \% q$ $\Rightarrow$ In this context it's most used as a public key

Friendly Building Block : $C_1 = g^{r_1} \cdot h^{\pi_1} \% q \wedge C_2 = g^{r_2} \cdot h^{\pi_2} \% q$

$\Rightarrow C_1 \cdot C_2 = g^{r_1 + r_2} \cdot h^{\pi_1 + \pi_2} \% q$

Same thing but commit several values.

MULTI-PEDERSON COMMITMENT ↑

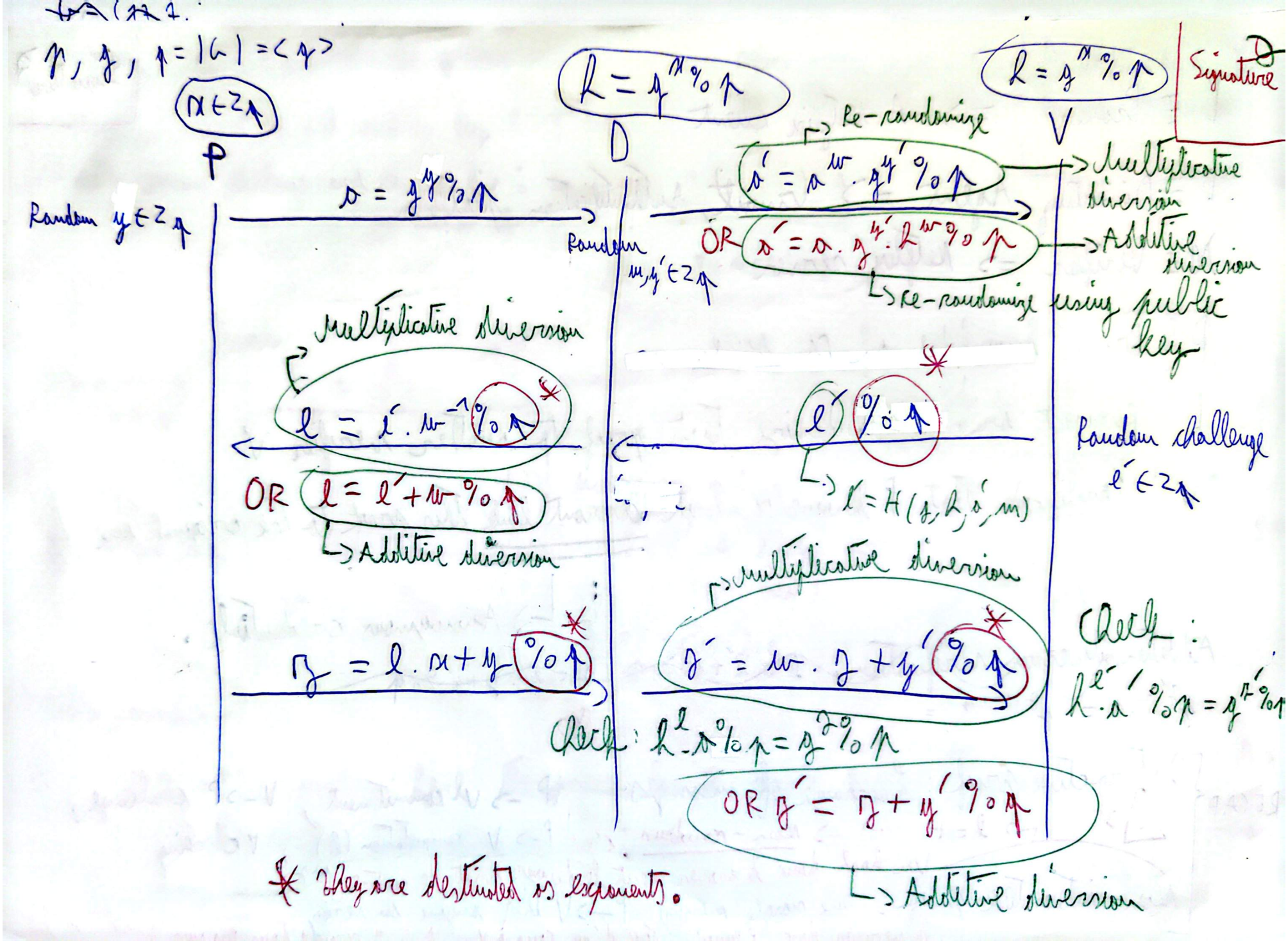
$r_1, \ldots, r_n \in \mathbb{Z}_q$

random $(r) \in \mathbb{Z}_q$

$n+1$ generators : $\forall i,j \mid i \neq j :: g_i = g_j^{\alpha} \bmod p \;\wedge\; g_i = h^{\alpha} \bmod p$

↳ $g_1, \ldots, g_n, h$ can be generated by any generator

$$C = \left(\prod_i g_i^{r_i}\right) \cdot h^{r} \bmod p = \left(g_1^{r_1} \cdot \ldots \cdot g_n^{r_n}\right) \cdot h^{r} \bmod p$$


$p, g, q = |G| = \langle g \rangle$
$x \in \mathbb{Z}_q$
P
$h = g^x \bmod p$
D
$h = g^x \bmod p$
V
Signature
Random $y \in \mathbb{Z}_q$
$a = g^y \bmod p$
Random $w, y' \in \mathbb{Z}_q$
Re-randomize
$a' = a^w \cdot g^{y'} \bmod p$
Multiplicative diversion
OR $a' = a \cdot g^{y'} \cdot h^w \bmod p$
Additive diversion
Re-randomize using public key
Multiplicative diversion
$l = i' \cdot w^{-1} \bmod q$ *
OR $l = l' + w \bmod q$
Additive diversion
$l \bmod q$ *
Random challenge $i' \in \mathbb{Z}_q$
$i' = H(g, h, a', m)$
Multiplicative diversion
$z = l \cdot x + y \bmod q$ *
$z' = w \cdot z + y' \bmod q$ *
Check: $h^{i'} \cdot a' \bmod p = g^{z'} \bmod p$
Check: $h^l \cdot a \bmod p = g^z \bmod p$
OR $z' = z + y' \bmod q$
Additive diversion
* They are destinated as exponents.

Signature ?

P ≡ Prover => A Netflix client

D ≡ Diverting Agent => Gmail, Authentification platform

V ≡ Verifier => Netflix server

P : Proves knowledge of x once

D : Redirect and re-randomize that proof to another verifier V

V : Convinced that P knows x, but cannot link this proof to the original one

↳ Anonymous credentials.

Additive diversion verification : $l = l' + w \cdot r$; $z' = w \cdot z + z' \cdot q$

$g^{z'} y^{r} = g^{z \cdot w + z'} =$

RECAP

- Interactive proof : Exchange of messages : P -> V commitment, V -> P challenge, P -> V computation (f), V checking
  - ⚠ -> $l = H(\dots)$ -> non-random, c'est un hash donc le prover peut le deviner avant de contacter
- Non-interactive proof : One single message P -> V that anyone can verify. le vérifier donc P envoit tout d'un coup à V. ex : l serait tous envoyer ensemble en 1 message

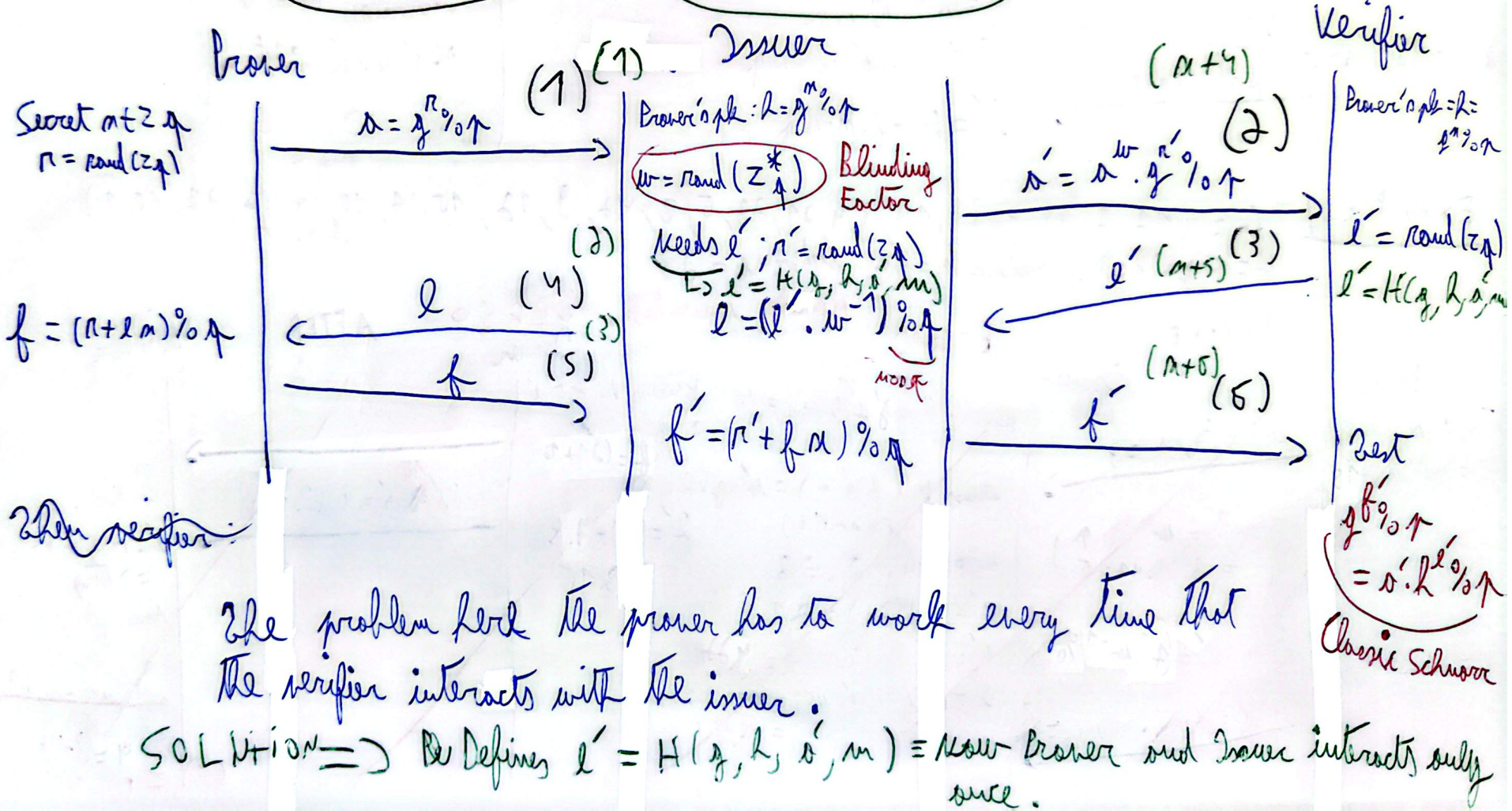
- Group $G = \langle g \rangle$ (mod q) of order $q = |G|$
- Prover's secret x and public key $h = g^x \% q$
- All Schnorr challenges and responses live in $\mathbb{Z}_q$. $\Rightarrow x, r, l \in \mathbb{Z}_q$ (not element of G)

Before error best must demonstrate reduced to mod q implicitly.

DIVERTIBLE BLIND SCHNORR PROOF

TIMELINE

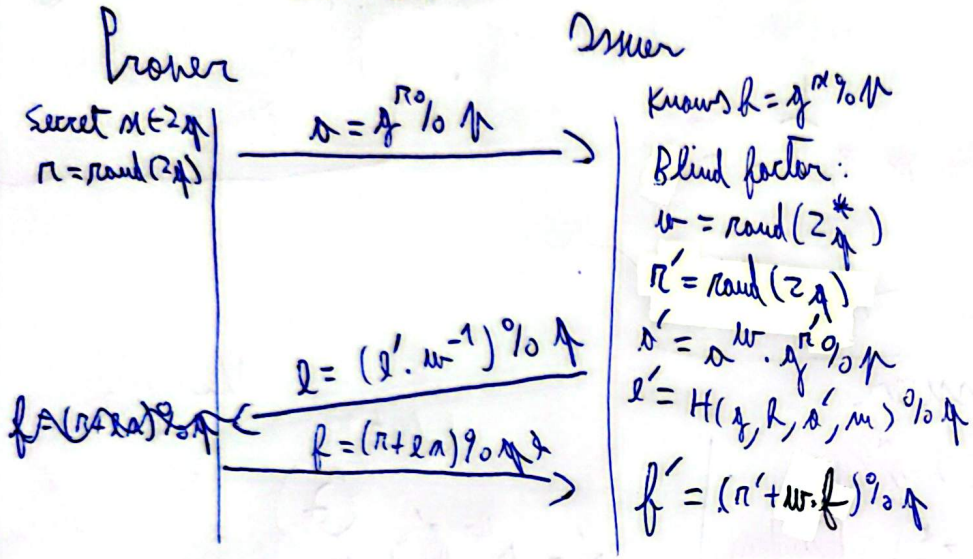
NEW TIMELINE



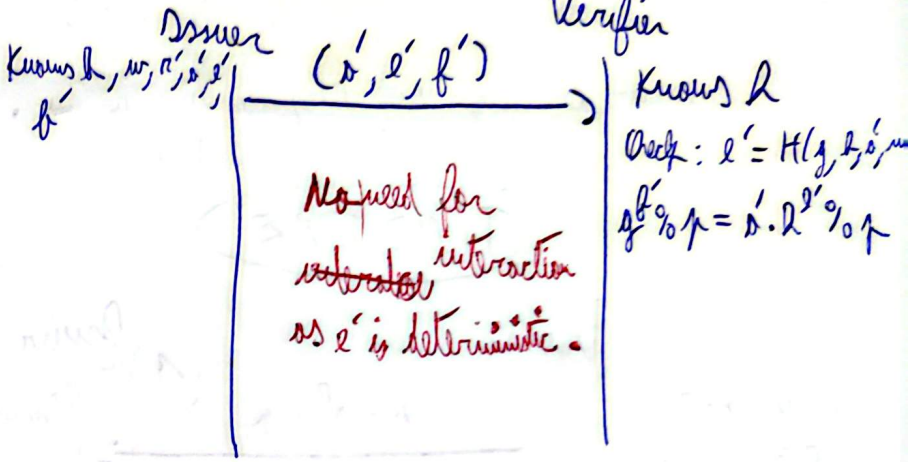
The problem here the prover has to work every time that the verifier interacts with the issuer.

SOLUTION $\Rightarrow$ Re Defines $l' = H(g, h, a', m) \equiv$ now Prover and Issuer interacts only once.

BEFORE



AFTER

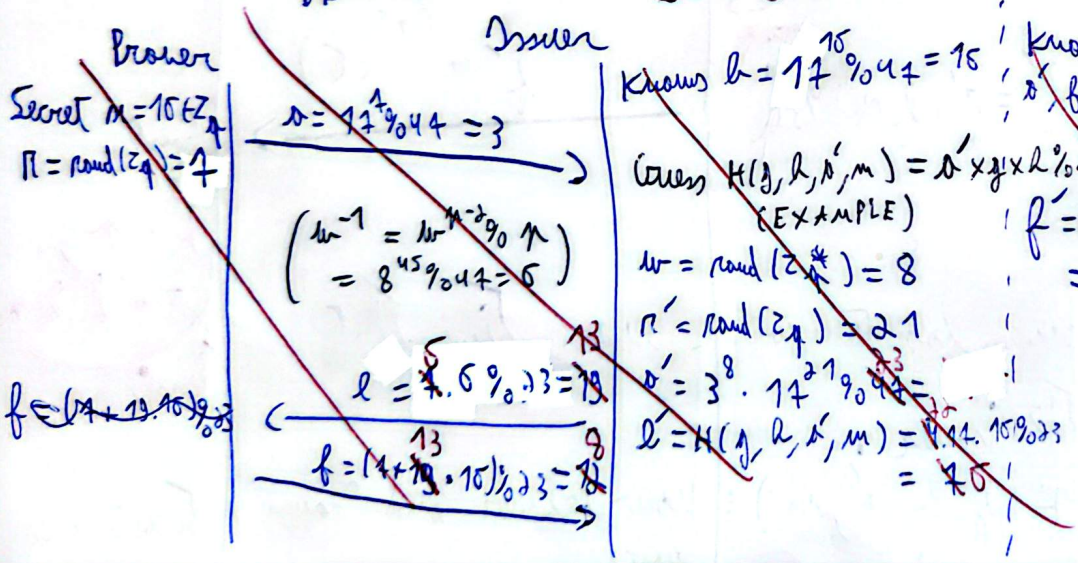


EXAMPLE:- $G = (17, 4, 25, 2, 34, 14, 3, 4, 21, 28, 5, 8, 42, 9, 12, 15, 24, 18, 24, 32, 27, 35, 1)$

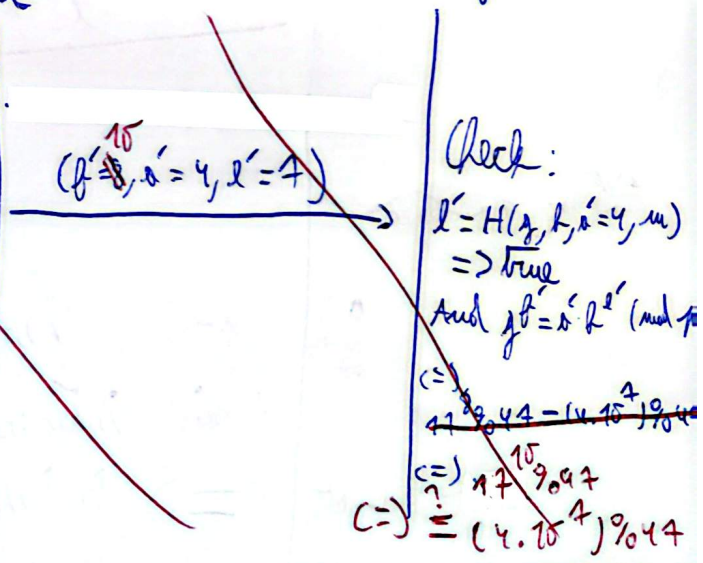
$G = \langle g = 17 \rangle$, modulo $p = 44$, order $q = 23$

modulus mismatch VERSO

BEFORE



AFTER



~~b = (11, 7)~~

BEFORE

Prover		Issuer
Secret $x = 15 \in \mathbb{Z}_q$		Knows $k = 11^{15} \% 44 = 15$
$r = rand(\mathbb{Z}_q) = 1$	$a = 11^7 \% 44 = 3$ →	Knows $H(y, k, a', m) = (y \cdot k \cdot a') \% q$
	($w \in \mathbb{Z}_q^*$, $w^{-1} = w^{q-2} \% q = 8^{21} \% 23 = 3$)	$w = rand(\mathbb{Z}_q^*) = 8$
		$r' = rand(\mathbb{Z}_q) = 21$
		$a' = (3^8 \cdot 11^{21}) \% 44 = 4$
	← $l = (3 \cdot 7) \% 23 = 21$	$l' = (11 \cdot 15 \cdot 4) \% 23 = 7$
	$f = (7 + 21 \cdot 15) \% 23 = 21$ →	$f' = (21 + 8 \cdot 21) \% 23 = 5$

AFTER

ALWAYS PAYING ATTENTION TO MODULOS, EASY SOURCE OF MISTAKES !!

Issuer		Verifier
Knows k, l', a', f'	(a', l', f') →	Knows $k = 15$
		Check : $l' = H(y, k, a', m)$ = Of course true
		Check :
		$11^5 \% 44 = (4 \cdot 15^7) \% 44$
		$\Leftrightarrow$ $34 = 34$

$\Rightarrow$ The issuer doesn't need the prover anymore.

B

DIVERTIBLE CHAUM PEDERSON for SIGNATURES

$G = \langle g, \ldots, u_0, \ldots, u_k \rangle$ Γ is also a base of G

↳ Known and agree by everyone

BEFORE

Prover | Issuer

$\pi = \text{rand}(Z_q)$

Commitment: $(m_1, \ldots, m_k)$

$\Gamma = u_1^{m_1} \cdot \ldots \cdot u_k^{m_k} \cdot u_0^{\pi} \% q$

$\Pi \equiv$ a set of proofs proving knowledge of m_i

$\pi' = \text{rand}(Z_q)$

$\Gamma' = \Gamma \cdot u_0^{\pi'} \% q$

$d' = d \cdot u_0^{\pi'} \% q$

$w = \text{rand}(Z_q)$, $R' = \text{rand}(Z_q)$

$s' = s^{w} g^{R'} \% q$

$h' = (h \cdot d^{\pi'})^{w} \cdot (\Gamma \cdot u_0^{\pi'})^{R'} \% q$

$c' = H(g, h, \Gamma', d', s', h')$

$(x' \in Z_q)$

$f' = (R' + w f) \% q$

c, Π →

Secret $x \in Z_q$

Public key: $h = g^{x} \% q$

$d = \Gamma^{x} \% q$, $u_0 = u_0^{x}$

$R = \text{rand}(Z_q) \Rightarrow s = g^{R} \% q$, $h = \Gamma^{R} \% q$, $s_0 = u_0^{R} \% q$

← d, s, h, s_0, u_0

$c = c' \cdot w^{-1} \% q$ →

$f = (R + c \cdot x) \% q$

← f

Prover has finaly signature:

$\sigma = (\Gamma', d', s', h', f')$

AFTER

Prover | Verifier

Signature
$G = (c', d', \hat{s}, \hat{e}', t')$ —— G (ONLY) ——>

Check : Recompute challenge $\hat{e}'$

→ $\hat{e}' = H(g, h, t', d', \hat{s}, \hat{e}')$

Checks signature / proof of knowledge of α

→ $\begin{cases} g^{\hat{s}} \% n \stackrel{?}{=} a' h^{\hat{e}'} \% n \\ (t')^{\hat{s}} \% n \stackrel{?}{=} \hat{e}' \cdot (d')^{\hat{e}'} \% n \quad (*) \end{cases}$

=> If true it says someone who knows α did the proof

Check if commitment t' (re-randomization of t) is associated to the signature

→ $\begin{cases} t' \text{ inside hash } \hat{e}' \\ t' \text{ as base in } (*) \\ d' \text{ satisfies } d' = (t')^{\alpha} \end{cases}$ —> Enough, cannot compute that on verifier

Then executes homomorphic functions on t' to get conclusions on values without revealing real values.

Une attribute $m_1 \equiv$ AGE $n = 47$, $g = 17$, $q = 23$ $\quad u_0 = 42$, $u_1 = 34$ See VERSO

$C = (17, 4, 25, 2, 34, 14, 3, 4, 21, 28, 5, 8, 42, 9, 12, 15, 37, 18, 24, 32, 27, 35, 1)$ 10

Issuer BEFORE (ONCE) | **Prover** | AFTER (AS MANY WE WANT) | **Verifier**

Secret $x = 17 \in \mathbb{Z}_4$

Sect Π ..., compute:

$d = 21^{17} \% 47 = 12$

$R = \text{rand}(\mathbb{Z}_4) = 3$

$A = 17^3 \% 47 = 25$

$b = 21^3 \% 47 = 2$

$A_0 = 42^{30} \% 47 = 15$

$r_0 = 42^{12} \% 47 = 9$

$f = (9 \cdot 14 + 3) \% 23 = 18$

$\pi = \text{rand}(\mathbb{Z}_4) = 2$

$m_1 \in \mathbb{Z}_4 = 22$

$c = 34^{22} \cdot 42^2 \% 47 = 27$

Π = Proof that prover knows m_1 (VERSO)

$\xleftarrow{c \text{ and } \Pi}$

$\xrightarrow{d, A, b, A_0, r_0}$

Re-randomize: $\pi' = \text{rand}(\mathbb{Z}_4) = 2$

$c' = 21 \cdot 42^{12} \% 47 = 2$

$m = \text{rand}(\mathbb{Z}_4^*) = 11$

$R' = \text{rand}(\mathbb{Z}_4) = 5$

$A' = 25^{11} \cdot 17^5 = 15$

$b' = (2 \cdot 15^{12})^{11} \cdot (2 \cdot 42^{12})^5 \% 47 = 18$

$\xleftarrow{l}$

$l' = H(g, R, r', d', A', b') \% 4 = 1$ (arbitrary choice)

$m^{-1} = 11^{23-2} \% 23 = 21$

$l = (1 \cdot 21) \% 23 = 9$

$\xrightarrow{f}$

Re-randomize:

$f' = (11 \cdot 18 + 5) \% 23 = 20$

$\sigma = (r', d', s', b', f')$

$d' = 12 \cdot 9^{12} \% 47 = 35$

σ sent

$l = 17^{17} \% 47 = 34$

$l' = H(g, R, r', d', s', b') = 1$

Check signature validity

$17^{20} \% 47 \overset{?}{=} (15 \cdot 32^7) \% 47$

$32 = 32$

Check commitment validity

$2^{20} \% 47 \overset{?}{=} (18 \cdot 35^7) \% 47$

$5 = 5$

$u_0 = g^{r_0} \% q$ $\quad y_i = \text{rand}(\mathbb{Z}_q) \mid \gcd(y_i, q) = 1$

$u_1 = g^{r_1} \% q$ $\quad y_0 = 13, y_1 = 17 \Rightarrow$ OK

So $u_0 = 4^{13} \% 47 = 42$

$u_1 = 17^{17} \% 47 = 37$

→ Test with two new bases give exact or swift order ?
(order doesn't matter)

Π = Prove knowledge of u_1 $\Rightarrow$
↳ it could be something more complicated as test if $u_1 > k$ ....

Prover

$r_1 = \text{rand}(\mathbb{Z}_q)$

$A_1 = u_1^{r_1} \% q$

$E_1 = H(g, h, A_1) \% q$

$F_1 = (r_1 + E_1 \cdot w_1) \% q$

(A_1, E_1, F_1)

Verifier

$k = g^{r'} \% q$

Non-interactive Schnorr proof to prove our commitment

Prover

$r = \text{rand}(\mathbb{Z}_q)$

$t_1 = u_1^{w_1} \cdot u_0^{r} \% q$

$r_0 = \text{rand}(\mathbb{Z}_q)$

$r_1 = \text{rand}(\mathbb{Z}_q)$

$A = u_0^{r_0} \cdot u_1^{r_1} \% q$

$E = H(h, g, A, \dots)$

$F_0 = (r_0 + r \cdot E) \% q$

$F_1 = (r_1 + w_1 \cdot E) \% q$

$\Pi = (A, F_0, F_1), t_1$

Verifier

$E = H(h, g, A, \dots)$

Test:

$u_0^{F_0} \cdot u_1^{F_1} \% q$

$= A \cdot t_1^{E} \% q$

— TO SEND E —

Check: $\begin{cases} E_1 = H(\dots) \\ u_1^{F_1} \% q = A_1 \cdot u_1^{E_1} \% q \end{cases}$ → This check is sufficient

$\Rightarrow$ By doing so, for all u_i we can send all proof at the same time