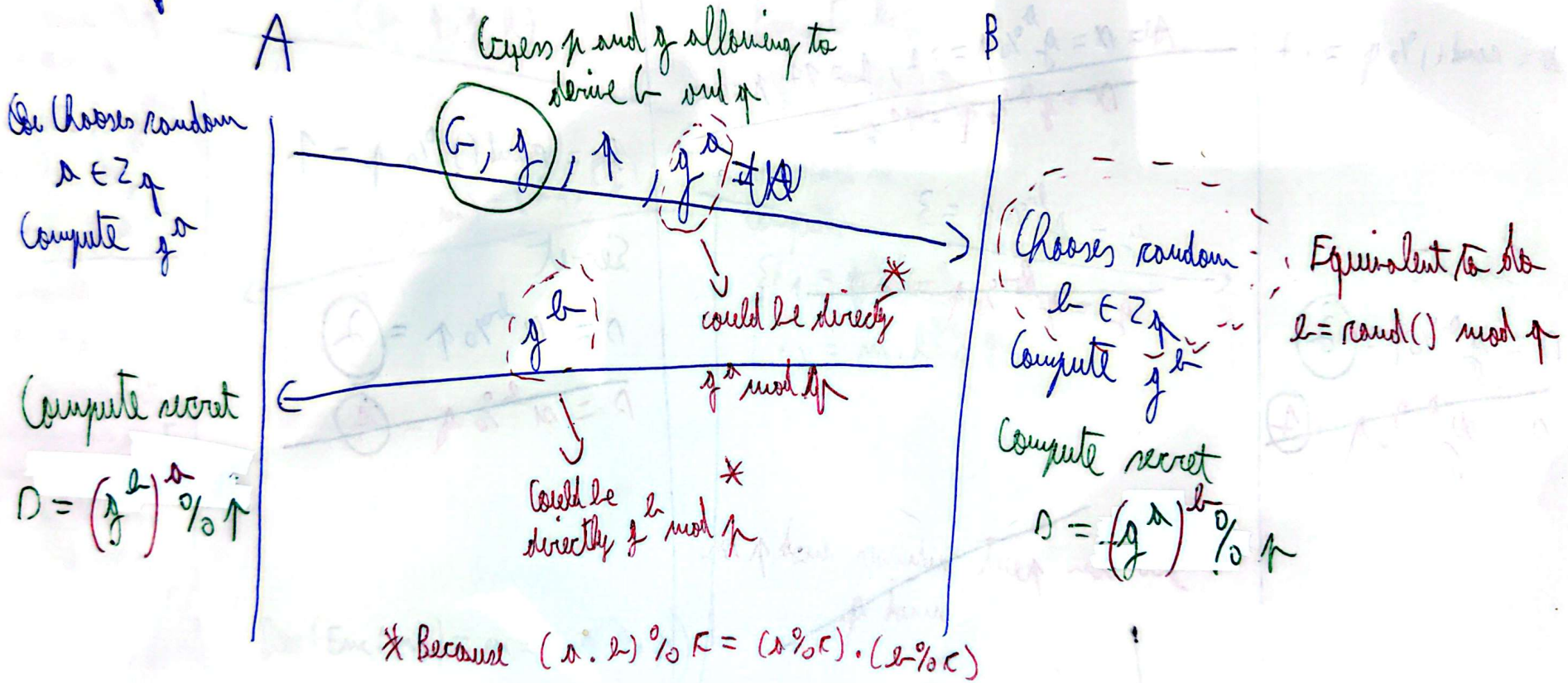
DIFFIE-HELLMAN

Cyclic group G of order p

$|G| = p$; $G = \langle g \rangle$; G generated as before with p and g

$\mathbb{Z}_p = \{0; 1; \dots; p-1\}$



EXAMPLE

7

$p = 47$ $q = 17$

As before $G = [17, 7, \dots, 30, 1] = \langle 17 \rangle$ (1, 2, ..., 22, 23)

$|G| = n = 23$; $\mathbb{Z}_n = \{0; 1; \dots; 22\}$

A

$a = \text{rand}() \% n = 14$

Secret

$b = y^a \% n = 2$

~~$a = x^b \% n = 1$~~

$x = g^a \% n = 37$, $y = 17$, $n = 47$

~~$x = g^a \% n = 14$~~

$y = g^b \% n = 3$

~~$y = g^b \% n = 20$~~

Ici on peut utiliser mod p ou mod q

↳ En fait non car alors on est plus dans le groupe G d'ordre q des coup comme $q < p$ on réduit les possibilités donc on affaiblit l'algorithme.

B

$b = \text{rand}() \% n = 7$

Secret

$a = x^b \% n = 2$

~~$a = x^b \% n = 1$~~

We have $G, g, \uparrow, \uparrow$ $|G| = \uparrow$ as before

EL GAMAL

A

Private key
$x = \text{rand}() \%_0 \uparrow$
so $x \in \mathbb{Z}_\uparrow$
Compute $h = g^{x} \%_0 \uparrow$
Public key :
$(G, \uparrow, g, h)$

Compute
$\begin{cases} r = c_1^{x} \%_0 \uparrow \\ \hookrightarrow \text{Si on appliquait pas de modulo avant avec } c_1 \text{ on aurait } r = g^{x \cdot y} \text{ mais le fait d'après revient au même} \\ m = c_2 / r \%_0 \uparrow \end{cases}$

$\left(\text{Dec}(\text{Enc}(m)) = m = c_2 \cdot r^{-1} \%_0 \uparrow \right)$

$\Leftrightarrow$
$m = c_2 \cdot e^{-1} \%_0 \uparrow$
we have to find the modular inverse of r

$(G, \uparrow, g, h)$ →

← $\text{Enc}(m) = (c_1, c_2)$

B

~~Private key~~
$y = \text{rand}() \%_0 \uparrow$
so $y \in \mathbb{Z}_\uparrow$
message m
Computes :
$\begin{cases} c_1 = g^{y} \%_0 \uparrow \\ c_2 = m \cdot h^{y} \%_0 \uparrow \end{cases}$

ElGamal Example

$p = 47$, $g = 17$, $q = 23$ ≡ Same as before

A

Random $\alpha \in \mathbb{Z}_q$

$\alpha = 6$ Private key

Compute $h = g^{\alpha} \% p$

$\Rightarrow h = 17^6 \% 47 = 14$

Public key pub = (G, p, g, h)

(G, p, g, h) →

B

random $\beta \in \mathbb{Z}_q$

$\beta = 22$ until $q-1$ so 23 not allowed

Message $m = \text{"Hello"} = 448378203247$

! m est plus large que p donc ça marche pas

Compute $\begin{cases} r_1 = g^{\beta} \% p = 17^{22} \% 47 = 35 \\ r_2 = m \cdot h^{\beta} \% p = 41 \end{cases}$

Plutôt pose $m = \text{"\$"} = 35$ in $\mathbb{Z}_p^*$

Compute $\begin{cases} r_1 = g^{\beta} \% p = 17^{22} \% 47 = 35 \\ r_2 = m \cdot h^{\beta} \% p = 35 \cdot 14^{22} \% 47 = 15 \end{cases}$

← $Enc(m) = (r_1, r_2)$

Compute $Dec(Enc(m))$:

$D = r_1^{\alpha} \% p$

$= 35^6 \% 47 = 37$

$D^{-1} = 14$ Trouvé en python

$(37 \cdot 14) \% 47 = 1$

$m = r_2 \cdot D^{-1} \% p$

$= (41 \cdot 14) \% 47 = 10$ ≠ m

=

← $Enc(m) = (r_1, r_2)$

$D = r_1^{\alpha} \% p =$

$35^6 \% 47 = 37$

$D^{-1} = 14$ (same)

$m = r_2 \cdot D^{-1} \% p$

$= 15 \cdot 14 \% 47$

$= 35 = \text{'\$'}$

I got my message

we have $G, g, \Lambda, |G| = \Lambda$ as before

$G \Rightarrow \Lambda$ includes Λ and g

EXPONENTIAL ELGAMAL

A

& N. voters

Secret $\alpha \in \mathbb{Z}_\Lambda$

Compute and publishes $h = g^\alpha \%_0 \Lambda$

(G^*, Λ, g, h)

Compute random $(r) \in \mathbb{Z}_\Lambda$

$\begin{cases} r_1 = g^r \%_0 \Lambda \\ r_2 = m' \cdot h^r \%_0 \Lambda \end{cases}$

Compute $m' = g^m \%_0 \Lambda \mid m \in \{0;1\}$

$Enc(m') = (r_1, r_2)$

Receive N votes encrypted and multiply them

$\prod_1^N Enc(m'_i) = Enc\left(\prod_1^N m'_i\right)$

$= Enc\left(\prod_1^N g^{m_i}\right)$

$= Enc\left(g^{\sum_i^N m_i}\right)$

In other words $\begin{cases} r'_1 = \prod_1 r_{1i} \\ r'_2 = \prod_1 r_{2i} \end{cases}$

$\Delta = (r'_1)^\alpha \%_0 \Lambda$

$m'' = r'_2 / \Delta \%_0 \Lambda = r'_2 \Delta^{-1} \%_0 \Lambda$

$m'' = g^{\sum_i m'_i}$

$\sum_i m'_i = \log_g(m'') \%_0 \Lambda$

→ Thanks to the exponential system where sum of encryptions gives the encrypted exponential exponent the sum of votes we have the right only one decryption for N voters.

→ But if the manager wanted, he could decrypt one by one and determines what each voter voted so there is no security improvement only performance.

EXPO-EG EXAMPLE 5.

Entity

3 voters

$p = 47 \wedge g = 17 \wedge q = 23$

Compute $x \in \mathbb{Z}_q$: $x = 14$

Compute and publishes

$h = g^x \% p = 17^{14} \% 47 = 9$

$(h = 9, p = 47, g = 17, q = 23)$

(r_{11}, r_{21})

(r_{12}, r_{22})

(r_{13}, r_{23})

$r_1' = \prod_i r_{1i} \% p = r_{11} \cdot r_{12} \cdot r_{13} \% p$
$= (3 \cdot 24 \cdot 36) \% 47$
$= 4$

$r_2' = \prod_i r_{2i} \% p = r_{21} \cdot r_{22} \cdot r_{23} \% p$
$= (3 \cdot 42 \cdot 28) \% 47$
$= 3$

$D = (r_1')^x \% p = 4^{14} \% 47 = 34$

D^{-1} in module $p = 47 = 18$

$m'' = r_2' \cdot D^{-1} \% p = 3 \cdot 18 \% 47$
$= 4$

$\sum_i m_i = \log_g(m'') \% p = \log_{17}(4) \% 47$
$= 2 = m_1 + m_2 + m_3 = 1 + 0 + 1$

-> only one decryption for 3 voters

V_1 votes $m = 1 \Rightarrow$ { random $r \in \mathbb{Z}_q$: $r = 7$; $m' = g^m \% p = 17$; $r_1 = g^r \% p = 3$; $r_2 = m' \cdot h^r \% p = 3$

V_2 votes $m = 0 \Rightarrow$ { $r = 19$; $m' = 1$; $r_1 = 24$; $r_2 = 42$

V_3 votes $m = 1 \Rightarrow$ { $r = 22$; $m' = 17$; $r_1 = 36$; $r_2 = 28$

DISTRIBUTED KEY GENERATION FOR ELGAMAL

p, g, q, G are shared

External sender wants to share $m \in G$

$\forall i :: T_i$ — Round (R) $\in \mathbb{Z}_q$ ∧ $c_1 = g^R \% p$ ∧ $c_2 = m \cdot h^R \% p$ — $\forall j \mid j \neq i :: T_j$

* h can be computed even by external parties

Choose secret $x_i \in \mathbb{Z}_q$ $(0 \leq x_i < q)$

Compute and publishes h

$h_i = g^{x_i} \% p$

Broadcast (c_1, c_2) to all trustees

h_i → (SAME)

$\forall j \mid j \neq i : h_j$ ←

Compute $h = \prod_i h_i \% p$

Compute $d_i = c_1^{x_i} \% p$

d_i → (SAME)

∧ Publishes it

$\forall j \mid j \neq i : d_j$ ←

Compute

$d = \prod_i d_i \% p$

Compute

$m = \frac{c_2}{d} \% p = c_2 \cdot d^{-1} \% p$

$d^{-1} \equiv$ inverse of d in modulo p

↳ Even external parties can see it and so decrypt the message

So what is the utility?

→ Trustees together authorize decryption by publishing d_i.

→ Once they do, the message becomes public to all observers.

$p = 47$, $g = 17$, $|G| = q = 23$ (ms + P1)

EXAMPLE 4
DKG – EG

t=1	t1	t2	t3	S
	Choose $\alpha_1 = 14 \in \mathbb{Z}_q$	Choose $\alpha_2 = 15 \in \mathbb{Z}_q$	Choose $\alpha_3 = 3 \in \mathbb{Z}_q$	
	Compute $h_1 = g^{\alpha_1} \% p = 17^{14} \% 47 = 9$	$h_2 = g^{\alpha_2} \% p = 17^{15} \% 47 = 15$	$h_3 = g^{\alpha_3} \% p = 17^{3} \% 47 = 25$	
	Publishes h_1	Publishes h_2	Publishes h_3	
	Compute $h = \prod_i h_i \% p$ $= (9 \cdot 15 \cdot 25) \% 47$ $= 28$	Compute $h = 28$	Compute $h = 28$	could compute h because infos are public
				wants to send $m = 42 \in \mathbb{Z}_p$
				Compute and publish $r \in \mathbb{Z}_p$: $r = 7$
				$c_1 = g^{r} \% p = 17^{7} \% 47 = 3$
				$c_2 = m \cdot h^{r} \% p = 42 \cdot 28^{7} \% 47 = 9$
				could compute d and decrypt
				Every external can decrypt m
	After receiving (c_1, c_2) from S Compute and publishes $d_1 = c_1^{\alpha_1} \% p = 3^{14} \% 47 = 14$	$d_2 = c_1^{\alpha_2} \% p = 3^{15} \% 47 = 32$	$d_3 = c_1^{\alpha_3} \% p = 3^{3} \% 47 = 27$	
now info is made public since all trustees have allowed it.	Compute $d = \prod_i d_i \% p = (14 \cdot 32 \cdot 27) \% 47$ $= 14$	Compute $d = 14$	Compute $d = 14$	
$(14 \cdot 36 \% 47 = 1)$ *	Compute $m = \frac{c_2}{d} \% p = c_2 \cdot d^{-1} \% p$ $= 9 \cdot 14^{-1} \% 47 = 9 \cdot 36 \% 47 = 42$ *	Compute $m = 42$	Compute $m = 42$	

p, g shared $\Rightarrow G$, p is before

$|G| = p$? $A \in G$ (or otherwise)

P

Knows secret $x \in \mathbb{Z}_p$

and $\begin{cases} a = g^x \bmod p \\ b = g'^x \bmod p \end{cases}$

Picks random $r \in \mathbb{Z}_p$

Computes $\begin{cases} t_1 = g^r \bmod p \\ t_2 = g'^r \bmod p \end{cases}$

(t_1, t_2)

c

Computes $s = (r + c \cdot x) \bmod p$

s

V

$g' \neq g^n \Rightarrow g', g \in G$

They are two bases

$\Rightarrow g^p \bmod p = 1 \wedge g'^p \bmod p = 1 \wedge$

$\forall 1 \leq k < p :: g^k \bmod p \neq 1 \wedge g'^k \bmod p \neq 1$

In general $g' = g^\tau \bmod p$ | τ

(p is the prime, p is the order) random $\tau \in \mathbb{Z}_p^*$

Picks random $c \in \mathbb{Z}_p$

Knows a et b (public keys)

Verify

$\begin{cases} g^s = t_1 \cdot a^c \pmod p \iff g^s \bmod p = (t_1 \cdot a^c) \bmod p \\ g'^s = t_2 \cdot b^c \pmod p \iff g'^s \bmod p = (t_2 \cdot b^c) \bmod p \end{cases}$

This protocol permits to prove to V knowing he has a valid public key $b = g'^x \bmod p$ that P is the holder of the associated private key x before sending datas encrypted from V to P with EL-GAMAL. Even if a malicious P can't decrypt EL-GAMAL, it's still talking to an imposter.

EXAMPLE 9

As before $n = 44$ $q = 11$ $\Rightarrow$ $G = [1, 4, \dots, 39, 1] = \langle 4 \rangle$ $\Rightarrow$ $q = |G| = 23$

Known secret $x = 11 \in \mathbb{Z}_q$

Known:

$$\begin{cases} h_1 = g = 17 \\ h_2 = g^{r} \% n = 17^{9} \% 44 = 2 \end{cases}$$

$(r = 4 \in \mathbb{Z}_q^*$, random$)$

$$\begin{cases} k_1 = g_1^{r} \% n = 17^{11} \% 44 = 5 \\ k_2 = g_2^{r} \% n = 2^{11} \% 44 = 37 \end{cases}$$

Compute:

and $r \in \mathbb{Z}_q \Rightarrow r = 19$

$$\begin{cases} a_1 = g_1^{r} \% n = 17^{19} \% 44 \\ \quad \Leftrightarrow a_1 = 24 \\ a_2 = g_2^{r} \% n = 2^{19} \% 44 \\ \quad \Leftrightarrow a_2 = 3 \end{cases}$$

Compute: (l est un exposant de mod q)

$f = (r + l \cdot x) \% q$

$\Rightarrow f = (19 + 41308 \cdot 11) \% 23$

$f = 19$

P $\rightarrow$ V: $(a_1, a_2) = (24, 3)$

V: Knows h_1 and h_2 + g_1 and g_2

Compute random $l \in \mathbb{Z}_2^n$

ex $l = 41308$

V $\rightarrow$ P: l

P $\rightarrow$ V: $f = 19$

Checks:

$$\begin{cases} g_1^{f} \% n = (a_1 \cdot h_1^{l}) \% n \\ g_2^{f} \% n = (a_2 \cdot h_2^{l}) \% n \end{cases}$$

$$\begin{cases} 17^{19} \% 44 = (24 \cdot 5^{41308}) \% 44 \\ 2^{19} \% 44 = (3 \cdot 37^{41308}) \% 44 \end{cases}$$

$$\begin{cases} 24 = 24 \\ 3 = 3 \end{cases}$$