$$\Rightarrow \quad \beta = \alpha^x = \alpha^{x_g m + x_b}$$

$$\Rightarrow \quad \beta \cdot (\alpha^{-m})^{x_g} = \alpha^{x_b} \qquad (1)$$

L'idée est de chercher une solution (x_g, x_b) de (1)

Déroulement : On calcule et on stocke α^{x_b}, $(0 \leq x_b < m)$

avec un temps $O(m) = O(\sqrt{|G|})$ temps

$O(\sqrt{|G|})$ valeurs stockées en mémoire.

(Etape bébé).

Etape. Ensuite, on calcule si (1) est vérifiée pour $0 \leq x_g < m$ pour chacune valeur x_g stockée.

$\Rightarrow$ { total $O(\sqrt{|G|})$ étapes calcul.
$O(\sqrt{|G|})$ espace.

Donc une attaque d'ordre 2^{80}, exige $|G| \geq 2^{160}$.

A3. <u>Méthode de Pollard</u> : $O(\sqrt{|G|})$ calcul,

basé sur le paradoxe des anniversaires :

On génère aléatoirement $\alpha^i \cdot \beta^j$ dans G. On continue jusqu'à collision :

$$\alpha^{i_1} \cdot \beta^{j_1} = \alpha^{i_2} \cdot \beta^{j_2} \qquad (2)$$

Si on prend $\beta = \alpha^x$, on remplace dans (2), on a

$$i_1 + j_1 x \equiv i_2 + x j_2 \Rightarrow x = \log_\alpha \beta \equiv \frac{i_2 - i_1}{j_1 - j_2} \pmod{|G|}$$

Dans ECC, le groupe est de taille $|G| > 2^{160}$.