ElGamal – Signaturen

Ziel: Für eine Nachricht m eine nicht fälschbare Unterschrift erzeugen, die von jedem verifiziert werden kann.

Setup: wie ElGamal-Verschlüsselung

(p, g, y, a) Schlüssel des Teilnehmers

$y \equiv g^a \bmod p$

Signieren einer Nachricht m

k Zufallszahl $\quad 2 \leq k \leq p-2 \wedge ggT(k, p-1) = 1$

$$r = g^k \bmod p$$

$$s = \frac{1}{k}(m - a \cdot r) \bmod p-1$$

$\uparrow ggT(k, p-1) = 1$

Signatur für m ist (r, s)

Verifizieren einer Signatur (r, s) für m

(*) Berechne $y^r \cdot r^s$ und prüfe $a^b \equiv g^m \bmod p$

$$y^r \cdot r^s \equiv (g^a)^r \cdot (g^k)^{\frac{1}{k} \cdot (m - ar)} \equiv g^m \bmod p$$

Falls (*) erfüllt, akzeptiere Signatur, sonst lehne ab.

Beispiel:

$p = 23$, $g = 5$, $a = 3$ (geheim), $m = 6$

$y \equiv 5^3 \equiv 10 \mod 23$

öffentlich $(p, g, y) = (23, 5, 10)$

Signieren von $m = 6$

$k = 5$, $ggT(k, 22) = 1$

$(r, s) = (20, 20)$

$y^r \cdot r^s \equiv 8 \equiv g^m \mod 23$ ✓

ElGamal $\longrightarrow$ DSA (p, g, a, y, q)

$q \mid p-1$

Bitlänge von q ist ≥ 256

DSA nachlesen

$\Rightarrow$ einzig bekannter Angriff ist Pollard in $O(\sqrt{q})$ Schritten $\sim \sqrt{2^{256}} = 2^{128}$