☆ ECC Simulating Elgamal

i.e. ECC-Elgamal

Key Generation (Say By Bob)

(1) Select $E_p(a, b)$

(2) Select $e_1 = (x_1, y_1)$

(3) Select d

(4) Calculate $e_2 = d \times e_1 = (x_2, y_2)$

Public key: (e_1, e_2, E_p)

Private key: d

★ Encryption (Say by Alice)

Plain Text : M

① Select random integer 'r'

② $C_1 = r \times e_1$

③ $C_2 = M + r \times e_2$

Cipher Text : C_1, C_2

★ Decryption (Say by Bob)

$M = C_2 - (d \times C_1)$

Verification RHS = $C_2 - d \times C_1$

$= M + r \times e_2 - d \times r \times e_1$

$= M + r \times d \times e_1 - d \times r \times e_1$

$= M + 0$

$= M$

$=$ LHS

Ex: Key generation:

Part A

① $E_{13}(1,1)$

② $e_1 = (1,4)$

③ $d = 4$

④ $e_2 = d \times e_1$

$= 4 e_1$

$= 4 * (1,4)$

$= (11,11)$ (See Ex. 2, Ex. 3)

Public Key: $e_1 = (1,4)$

$e_2 = (11,11)$

$E_{13}(1,1)$

Private key: $d = 4$

Part B Plaintext $M : (12,5) \in E_{13}(1,1)$

$r = 1$

$C_1 = r \times e_1$

$= 1 \times e_1$

$= e_1$

$= (1,4)$

$C_2 = M + r \times e_2$

$= (12,5) + 1 \times (11,11)$

$= (12,5) + (11,11)$

$\lambda = [(11-5)/(11-12)] \text{ MOD } 13$

$= -6 \text{ MOD } 13$

$= 7$

$x_3 = (\lambda^2 - x_1 - x_2) \% 13$

$= (49 - 12 - 11) \% 13$

$= 0$

$$y_3 = (\lambda(x_1 - x_3) - y_1) \% 13$$
$$= (7(12-0) - 5) \% 13$$
$$= 79 \% 13$$
$$= 1$$

$$\therefore \boxed{C_2 = (x_3, y_3) = (0,1)}$$
$$\boxed{C_1 = (1,4)}$$

Part C Decryption

$$C_2 - d \times C_1$$
$$= (0,1) - 4 \times (1,4)$$
$$= (0,1) - (11,11)$$
$$= (0,1) + -(11,11)$$
$$= (0,1) + (11,-11)$$ (Add Inverse of (11,11))

$$\lambda = \left(\frac{-11-1}{11-0}\right) \text{MOD } 13$$
$$= (-12 \times 11^{-1}) \text{MOD } 13$$
$$= (-12 \times 6) \% 13$$
$$= -72 \% 13$$
$$= 6$$

$$\therefore \boxed{\lambda = 6}$$

$$x_3 = (\lambda^2 - x_1 - x_2) \% 13$$
$$= (36 - 0 - 11) \% 13$$
$$= 25 \% 13$$
$$= 12$$

$$y_3 = (\lambda(x_1 - x_3) - y_1) \% 13$$
$$= (6(0 - 12) - 1) \% 13$$
$$= -73 \% 13$$
$$= 5$$

$$13 \overline{)\,-73\,} \quad -6$$
$$-78$$
$$+$$
$$5$$

∴ Decrypted ~~text~~'s Point
(x_3, y_3)
$= (12, 5)$
$= M$

Additional Exercise

Take $E_{67}(2,3)$

Create ~~few~~ points over the curve.

Take $e_1 = (2,22)$, $d = 4$, $e_2 = (13,45)$

PlainText $M = (24,26)$

Encrypt M and Decrypt C_1, C_2 to verify whether it is equals to M or not.

Issue: Challenge is to map any Arbitrary plaintext to a point on the Elliptic curve.

→ Alice needs to find a one to one Correspondence between symbols and the points on the curve.

Security of ECC:

(a) If Eve knows r, she can use $M = C_2 - (r \times e_2)$ to find M.

→ But to knao r, Eve needs to solve

$$C_1 = r \times e_1$$

C_1: C.T. point; r: Integer; e_1: Public key (point)

Given C_1, e_1, we need to get r which is known as "Elliptic Curve Logarithm problem"

If r is large, this problem is very Hard to Solve.

(b) If Eve knows d, she can use

$M = C_2 - (d \times C_1)$ to find M.

But to find 'd', one needs to solve

$$e_2 = d \times e_1$$

(e_2: Point, d: Int, e_1: Point)

Given e_2, e_1, to find d is known as / Elliptic curve Log problem

This is hard when n is large.

★ For the same level of security (computational effort), the modulus n, can be smaller than that in RSA.

E.g. ECC over $GF(2^n)$ with n of 160 bits can provide the same level of security as RSA with n of 1024 bits.

Date
Page

☆ Comparison between Elgamal and ECC-Elgamal

Elgamal	ECC - Elgamal
(1) Uses a multiplicative group	(1) ECC - elgamal uses an elliptic group
(2) Exponents are numbers in the multiplicative group	(2) Two multiplies in this are points on the Elliptic curve
(3) Private key is integer.	(3) Private key is integer.
(4) Secret number chosen i.e. k is integer.	(4) Secret number chosen i.e. k is integer.
(5) Exponentiation is used.	(5) Multiplication of a point by a constant is used
(6) Multiplicative Inverse (of the Mult. group) is used.	(6) Additive Inverse of a point is used
(7) Exponentiation is comparatively more harder than multiplication.	(7) Calculation is usually easier because Multiplication is simpler
(8) Finding Inverse requires Extended Euclidean algo.	(8) Finding inverse of a point here is much simpler.