CS641

Modern Cryptology

Lecture 14

ElGamal Cryptosystems

- Proposed by Taher ElGamal in 1985.
- Generic scheme based on finite groups.
- Leads to multiple cryptosystems depending on specific group chosen.

Key Generation

- Let G be a finite group under operation '$\cdot$'.
- Let $g \in G$ be an element of large order, say t.
- Pick a random e, $1 < e < t$.
- Encryption or public-key: (g, g^e, t)
- Decryption or private-key: $t - e$

Encryption

- Plaintext block m is viewed as an element of G.
- Pick a random r, $1 < r < t$.
- Compute g^r and $m \cdot g^{er}$.
- Output $c = (g^r, m \cdot g^{er})$.

Decryption

- Let $c = (h, \hat{m})$ be the ciphertext block.
- Compute h^{t-e} and output $\hat{m} \cdot h^{t-e}$.
- If $h = g^r$ and $\hat{m} = m \cdot g^{er}$, then

$$\hat{m} \cdot h^{t-e} = m \cdot g^{er} \cdot g^{r(t-e)} = m \cdot g^{rt} = m.$$

Efficiency

- Key generation, encryption and decryption all require computing a large power of an element of G.
- If the group operation can be carried out efficiently, then all of them can be executed efficiently.

Security

- Given public-key (g, g^e, t), computing g^{t-e} is equivalent to computing g^e.
- Computing g^e is exactly the Discrete Log problem in group G.
- So if solving Discrete Log in G is hard, computing private key is hard.
- Given $(g^r, m \cdot g^{er}, g, g^e, t)$, computing m is equivalent to computing g^{er}.
- This seems to require computing either r or e which again reduces to solving Discrete Log problem.

El Gamal System Based on F_p^*

- Let p be a large Sophie Germain prime.
- Let $G = F_p^*$ and g a generator of F_p^*.
- Discrete Log problem in F_p^* is believed to be hard.
- The fastest known algorithm takes time $2^{O((\log p)^{1/3}(\log\log p)^{2/3})}$ as already noted.
- This requires a key size of 1024 bits for security.
- Is there a group with harder Discrete Log problem?

Elliptic Curves

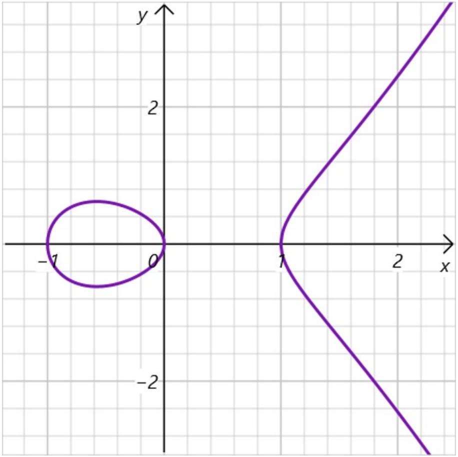
- Elliptic curves over $\mathbb{R}$ are given by equation:

$$y^2 = x^3 + Ax + B,$$

  with $4A^3 + 27B^2 \neq 0$.
- The condition $4A^3 + 27B^2 \neq 0$ ensures that $x^3 + Ax + B$ does not have repeated roots.

Example Curve: $y^2 = x^3 - x$

- Roots of $x^3 = x$ are -1, 0, and 1.



Elliptic Curve Group

- Let C represent the equation of an elliptic curve, and F a field.
- Define
$$E(C,F) = \{(x,y) \in F^2 \mid C(x,y) = 0\} \cup \{O\},$$
where O is point at infinity.
- It is assumed that any line parallel to y-axis meets O.
- We now define an addition operation on points in $E(C,F)$.

Elliptic Curve Group over $\mathbb{R}$

- First consider $E(C, \mathbb{R})$.
- Given $P, Q \in E(C, \mathbb{R})$, define $P + Q = R$ where R is obtained as follows.
  - If $P = O$ then $R = Q$, and if $Q = O$ then $R = P$.
  - Otherwise, let $P = (x_1, y_1)$ and $Q = (x_2, y_2)$. If $x_1 \neq x_2$, then draw a line passing through P and Q. This line will intersect the curve at a third point, say (x_3, y_3). Then, $R = (x_3, -y_3)$.
  - If $x_1 = x_2$ and $y_1 = -y_2$, then $R = O$.
  - If $x_1 = x_2$ and $y_1 = y_2$, then draw a tangent on C passing through P, let (x_3, y_3) be the second point of intersection with C, and set $R = (x_3, -y_3)$.
- The point $R \in E(C, \mathbb{R})$ since $(a, b) \in E(C, \mathbb{R})$ iff $(a, -b) \in E(C, \mathbb{R})$.

Elliptic Curve Group over $\mathbb{R}$

- Addition can be viewed as drawing a line through two points and reflecting the third point of intersection wrt x-axis.
  - Line through $P = (x_1, y_1)$ and O is parallel to y-axis by assumption, which intersects the curve at $(x_1, -y_1)$. Reflected wrt x-axis, we get point P.
  - When $x_1 = x_2$ and $y_1 = -y_2$, line through the points is again parallel to y-axis and meets O at infinity. Reflecting wrt x-axis is still point at infinity.
  - When $x_1 = x_2$ and $y_1 = y_2$, tangent at P is the limit of taking a point on C close to P, drawing a line through the two, and then reducing the distance between them.

Elliptic Curve Group over $\mathbb{R}$

Theorem

$E(C, \mathbb{R})$ is a group under addition.

- Closure is already shown.
- Point O is identity since $P + O = P$ for any P.
- Inverse of $P = (x, y)$ is $(x, -y)$ since $P + (x, -y) = O$.
- We write $-P$ for $(x, -y)$.
- Associativity is hard to prove, so not shown.

General Elliptic Curve Group

- $E(C, F)$ can be shown to be a group for any field F under suitably defined addition.
- Instead of geometric, we use algebraic definitions:
  - $(x_1, y_1) + (x_2, y_2) = (x_3, y_3)$ with $x_3 = m^2 - x_1 - x_2$, and $y_3 = y_1 + m(m^2 - 2x_1 - x_2)$ where $m = (y_2 - y_1)/(x_2 - x_1)$.
- $E(C, \mathbb{C})$ and $E(C, \mathbb{Q})$ have been intensely studied:
  - $E(C, \mathbb{C})$ is shaped like a donut.
  - $E(C, \mathbb{Q})$ is used in proof of Fermat's Last Theorem.
- We will use $E(C, F_p)$, where p is prime.

Elliptic Curve Group over F_p

Hasse's Theorem

$p + 1 - 2\sqrt{p} \leq |E(C, F_p)| \leq p + 1 + 2\sqrt{p}$.

- The group $E(C, F_p)$ is either cyclic or is a product of two cyclic groups, depending on the curve C.

Elliptic Curve Cryptography (ECC)

- Choose a prime of size 160 bits.
- Choose a curve C such that $E(C, F_p)$ is cyclic with generator P and size n.
- Public key is (C, p, P, eP) and private key is $n - e$ where $1 < e < n$.
- For encryption, plaintext block m is mapped to a point P_m on the curve whose x-coordinate is defined by m.
- Group addition can be carried out efficiently.

Security of ECC

- Discrete Log problem for $E(C, F_p)$ has no known efficient algorithms.
- The fastest known algorithm takes time $2^{O(\log p)}$.
- This makes it significantly more difficult that solving Discrete Log for F_q^* or factoring n.
- Therefore, security provided by 160-bit prime p is roughly same as security provided by 1024-bit RSA.
- This makes encryption and decryption significantly faster for ECC than RSA.

Quantum Computers

- Quantum computers use quantum superposition to carry out certain computations much faster than classical computers.
- Peter Shor showed that both integer factoring and discrete log problems can be efficiently solved using quantum computers.
- This breaks the security of both RSA and ECC.
- Since it is expected that quantum computers will be build in near future, a new public-key encryption algorithm that is secure against quantum computers is required.