Elliptic Curve Cryptography

Lecture by :
Dr. Anil Pinapati
Assistant Professor
Department of C.S.E
NIT Calicut

Introduction

- Cryptography is an analysis and design of mathematical model which establishes communication between two parties in the presence of malicious adversary.

Basic communications model.

Basic Terminology

- Plain text(P): Text that user can understand/readable
- Cipher Text(C): Text that is not readable
- Encryption(E): The process of converting plain text to cipher text.

  $C=E(P, k)=E_k(P)$
- Decryption(D): The process of converting cipher text to plain text.

  $P=D(C, k)=D_k(P)$ where k is secret key

  $P=D_k(E_k(P))$.

Basic Caesar Cipher

- En(m)=c=(m+n) mod 26.
- Dn(m)=m=(c-n) mod 26

Ex:- plain(m)=i am in lbs

cipher(c)=lxdpxlqxoev

Security Goals: CIAEN

The CIA Triad of confidentiality, integrity and availability is considered the core underpinning of information security.

- Confidentiality: Confidentiality measures protect information from unauthorized access and misuse.
- Integrity: Integrity measures protect information from unauthorized alteration.
- Availability : In order for an information system to be useful it must be available to authorized users.
- Entity authentication: Corroborating the identity of an entity—B should be convinced of the identity of the other communicating entity.
- Non-repudiation: Preventing an entity from denying previous commitments or actions

Cryptography

Symmetric/Private key cryptography:

- RC4,
- DES(Data Encryption Standard),
- AES(Advanced Encryption Standard)

Asymmetric/Public key cryptography:

- RSA,
- El-Gamal,
- Elliptic Curve Cryptography(ECC)

Conti…

- Symmetric Key Cryptography

Public Key Cryptography

RSA(Rivest, Shamir, Adelman)

1. Choose two primes p and q and let $n=pq$
2. Let $e \in Z$ be positive such that $\gcd(e, \varphi(n)) = 1$.
3. Compute a value for $d \in Z$ such that $de \equiv 1 \pmod{\varphi(n)}$.
4. Our public key is the pair (n, e) and our private key is the triple (p, q, d). For any non-zero integer $m<n$, encrypt mm using $c \equiv m^{\wedge}{}_{e} \pmod{n}$.
5. Decrypt c using $m \equiv c^{\wedge}{}_{d} \pmod{n}$.

<u>Example:</u>

1. Start with two prime numbers: $p = 7$, $q = 11$.
2. $n = p * q = 7 * 1 = 77$.
3. $\varphi(n) = (p-1)(q-1) = 6 * 10 = 60$.
4. If $e = 13$ then $d = 37$, since $13 * 37 = 481 \pmod{60} = 1$.

Public key = (e); private key (d)

$C = m^e \bmod n$,

$P = C^d \bmod n$.

RSA Key Generation

Algorithm 1.1 RSA key pair generation

INPUT: Security parameter l.

OUTPUT: RSA public key (n, e) and private key d.

1. Randomly select two primes p and q of the same bitlength $l/2$.
2. Compute $n = pq$ and $\phi = (p-1)(q-1)$.
3. Select an arbitrary integer e with $1 < e < \phi$ and $\gcd(e, \phi) = 1$.
4. Compute the integer d satisfying $1 < d < \phi$ and $ed \equiv 1 \pmod{\phi}$.
5. Return(n, e, d).

RSA Encryption and Decryption

Algorithm 1.2 Basic RSA encryption

INPUT: RSA public key (n, e), plaintext $m \in [0, n-1]$.
OUTPUT: Ciphertext c.

1. Compute $c = m^e \bmod n$.
2. Return(c).

Algorithm 1.3 Basic RSA decryption

INPUT: RSA public key (n, e), RSA private key d, ciphertext c.
OUTPUT: Plaintext m.

1. Compute $m = c^d \bmod n$.
2. Return(m).

Cont…

ENCRYPTION

Let m = 2,

1. $C = 2^{13} \bmod 77 = 8192 \bmod 77 = 30.$

DECRYPTION

1. $m = 30^{37} \bmod 77$

   $= 450283905890997363000000000000000000000000000000000000000 \bmod 77$

   $m = 2.$

Discrete Logarithm Problem

- public domain parameters (p, q, g). Here, p is a prime, q is a prime divisor of p–1, and g ∈ [1, p–1] has order q (i.e., t = q is the smallest positive integer satisfying g^t ≡ 1(mod p)).
- A private key is an integer x that is selected uniformly at random from the interval [1,q –1] (this operation is denoted $x \in_R [1,q-1]$), and the corresponding public key is y = g^x mod p.
- The problem of determining x given domain parameters (p, q, g) and y is the *Discrete Logarithm Problem* (DLP).

DL domain parameters and key pair generation

Algorithm 1.6 DL domain parameter generation

INPUT: Security parameters l, t.
OUTPUT: DL domain parameters (p, q, g).

1. Select a t-bit prime q and an l-bit prime p such that q divides $p - 1$.
2. Select an element g of order q:
   2.1 Select arbitrary $h \in [1, p-1]$ and compute $g = h^{(p-1)/q} \bmod p$.
   2.2 If $g = 1$ then go to step 2.1.
3. Return(p, q, g).

Algorithm 1.7 DL key pair generation

INPUT: DL domain parameters (p, q, g).
OUTPUT: Public key y and private key x.

1. Select $x \in_R [1, q-1]$.
2. Compute $y = g^x \bmod p$.
3. Return(y, x).

El-Gamal Encryption and Decryption

Algorithm 1.8 Basic ElGamal encryption

INPUT: DL domain parameters (p, q, g), public key y, plaintext $m \in [0, p-1]$.
OUTPUT: Ciphertext (c_1, c_2).

1. Select $k \in_R [1, q-1]$.
2. Compute $c_1 = g^k \bmod p$.
3. Compute $c_2 = m \cdot y^k \bmod p$.
4. Return(c_1, c_2).

Algorithm 1.9 Basic ElGamal decryption

INPUT: DL domain parameters (p, q, g), private key x, ciphertext (c_1, c_2).
OUTPUT: Plaintext m.

1. Compute $m = c_2 \cdot c_1^{-x} \bmod p$.
2. Return(m).

Elliptic curves in Cryptography

- Elliptic Curve (EC) systems as applied to cryptography were first proposed in 1985 independently by Neal Koblitz and Victor Miller.

- The Discrete Logarithm Problem(DLP) on elliptic curve groups is believed to be more difficult than the corresponding problem in (the multiplicative group of nonzero elements of) the underlying finite field.

Definition of Elliptic curves

- An elliptic curve over a field $\boldsymbol{K}$ is a nonsingular cubic curve in two variables, $f(x,y) = 0$ with a rational point (which may be a point at infinity).

- The field $\boldsymbol{K}$ is usually taken to be the complex numbers, reals, rationals, algebraic extensions of rationals, p-adic numbers, or a **finite field**.

- Elliptic curve groups for cryptography are examined with the underlying fields of $\boldsymbol{F_p}$ (*where* $p>3$ *is a prime*) and $\boldsymbol{F_{2^m}}$ (*a binary representation with* 2^m *elements*).

Advantages of ECC

- It provides all the benefits of the other cryptosystems like: confidentiality, integrity, authentication and non-repudiation.
- Shorter key lengths
  - Encryption, Decryption and Signature verification speed up
  - Storage and bandwidth savings

Table 1:- key comparison between private and public key cryptography.

Security bits	RSA	ElGamal	Elliptic
80	1024	1024	160
112	2048	2048	224
128	3074	3074	256
192	7680	7680	384
256	15360	15360	512

Applications of ECC

- Many devices are small and have limited storage and computational power
- Where can we apply ECC?
  - **Wireless communication devices**
  - Smart cards
  - Web servers that need to handle many encryption sessions
  - **Any application where security is needed but lacks the power, storage and computational power that is necessary for our current cryptosystems**

Security of ECC?

- Security of Public Key Cryptosystem: It relies on the evaluation of the computational difficulty of some families of mathematical problems and its complexity.
- How do we analyze Cryptosystems?
  - How difficult is the underlying problem that it is based upon
    - RSA – Integer Factorization(n=pq)
    - DH – Discrete Logarithms(y=a^x)
    - ECC - Elliptic Curve Discrete Logarithm problem(Q=sP)
  - How do we measure difficulty?
    - We examine the algorithms used to solve these problems

ECC operations: Hierarchy

Introduction to Elliptic Curves

Lets start with a small puzzle

- **What is the number of balls that may be piled as a square pyramid and also rearranged into a square array?**
- **Soln:** Let x be the height of the pyramid…

Thus, $$1^2 + 2^2 + 3^2 + \ldots + x^2 = \frac{x(x+1)(2x+1)}{6}$$

We also want this to be a square:

Hence, $$\boxed{y^2 = \frac{x(x+1)(2x+1)}{6}}$$

Graphical Representation

Curves of this nature are called ELLIPTIC CURVES

Method of Diophantus

- Uses a set of known points to produce new points
- (x1,y1)=(0, 0) and (x2,y2)=(1, 1) are two trivial solutions
- Equation of line through these points is y=x.

  y=m(x-x1)+y1
- Intersecting with the curve and rearranging terms:

$$y^2 = \frac{x(x+1)(2x+1)}{6} \quad => \quad x^3 - \frac{3}{2}x^2 + \frac{1}{2}x = 0$$

$$(x-a)(x-b)(x-c) = x^3 - (a+b+c)x^2 + (ab+ac+bc)x - abc.$$

- We know that from quad. eq i.e. a+b+c=-co.iff(x^2)

  ∴ 1 + 0 + x = 3/2 => x = ½ and y = ½
- Using symmetry of the curve we also have (1/2,-1/2) as another solution.

Diophantus' Method

- Consider the line through (1/2,-1/2) and (1,1), i.e., y=m(x-x1)+y1=>y=3(x-1/2)-1/2=>y=3x-2
- Intersecting with the curve we have:

$$x^3 - \frac{51}{2}x^2 + \ldots = 0$$

- Thus ½ + 1 + x = 51/2 or x = 24 and y=70

- Thus if we have **4900** balls we may arrange them in either way. (means Pyramid or Square)

Introduction about Groups

A group is a non-empty set G equipped with a binary operation * that satisfies the following properties for all a, b, c in G:

1. Closure : a, b in G, a. b is also in G.
2. Associativity: (a*b)*c = a*(b*c)
3. Identity: There exists an element e in G such that a*e =e*a = a. We call e the identity element of G.
4. Inverse: For each a in G, there exists an element d in G such that a*d = e = d*a. We call d is the inverse of a.

If a group G also satisfies the following property for all a, b in G

5. Commutativity: *a*b = b*a, we say G is an abelian group.*

The order of a group G is denoted by |G| is the number of elements in G.

Galois Field GF(P)

- It is a finite field and it consists of a set of integers {0,1,2,3….P-1} where P is a prime number. Additionally it satisfies the following arithmetic operations with P=29

(i) Addition: $17 + 20 = 8$ since $37 \bmod 29 = 8$.

(ii) Subtraction: $17 - 20 = 26$ since $-3 \bmod 29 = 26$.

(iii) Multiplication: $17 \cdot 20 = 21$ since $340 \bmod 29 = 21$.

(iv) Inversion: $17^{-1} = 12$ since $17 \cdot 12 \bmod 29 = 1$.

Galois Field GF(2^m)

- It is a finite field and is called binary finite field. It is a vector space of dimension m over GF(2) i.e. there exists a set of m elements $\{\alpha_{m-1}, \ldots, \alpha_1, \alpha_0\}$ each $\alpha_i \in \{0,1\}$ in GF(2^m) such that each $a \in$ GF(2^m)

$$a = \alpha_{m-1}x^{m-1} + \ldots + \alpha_1 x + \alpha_0$$

- Additionally it satisfies the following arithmetic operations :

For instance $a = \{a_{m-1},\ldots,a_1,a_0\}$ and $b = \{b_{m-1},\ldots,b_1,b_0\} \in$ GF(2^m)

- **<u>Addition :</u>** $a + b = c = \{c_{m-1},\ldots,c_1,c_0\}$ where $c_i = (a_i + b_i) \bmod 2$ finally $c \in$ GF(2^m) .
- **<u>Multiplication</u>** : $a . b = c = \{c_{m-1},\ldots,c_1,c_0\}$ where c is the remiander of the division of the polynomial a(x) . b(x) by an irreducible polynomial of degree m. $c \in$ GF(2^m)

Weierstrass Equation

- Generalized Weierstrass Equation of elliptic curves:

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$$

Here, a, b, x and y all belong to a field of say rational numbers, complex numbers, finite fields (F_p) or Galois Fields ($GF(2^n)$).

Types of Elliptic Curves

- An *elliptic curve* is a plane curve char(k)≠2,3

$$y^2 = x^3 + ax + b \text{ where } a, b \in F_p$$

- An *elliptic curve* with char(k)=2

$$y^2 + xy = x^3 + ax^2 + b \text{ where } a, b \in F_{2^m}$$

- Koblitz curves:

$$y^2 + xy = x^3 + ax^2 + b \text{ where } a \in \{0,1\}, b = 1$$

Point addition and Point doubling in graphical representation

P,Q,R are points on the curve.

(a) Addition: $P + Q = R$.

(b) Doubling: $P + P = R$.

Group laws of ECC where Char(p)>3

- P and Q be two points on $E_{(a,b)}(Fp)$ and O is the **point at infinity**.

  1. **Additive Identity** : P+O = O+P = P
  2. **Negation** : If $P = (x_1, y_1)$ then $(-P) = (x_1, -y_1)$ and $P + (-P) = O$.
  3. **Point addition :** If $P = (x_1, y_1)$ and $Q = (x_2, y_2)$, and P and Q are not O. then $P + Q = (x_3, y_3)$ where

     $x_3 = \lambda^2 - x_1 - x_2$

     $y_3 = \lambda(x_1 - x_3) - y_1$

     where $\lambda = (y_2 - y_1)/(x_2 - x_1)$ if $P \neq Q$
  4. **Point Doubling:** P+P=2P,P not equal to –P**.**

     $x_3 = \lambda^2 - 2x_1$,

     $y_3 = \lambda(x_1 - x_3) - y_1$

     Where $\lambda = (3x_1^2 + a)/ 2y_1$ if $P = Q$

Addition in Affine Co-ordinates

y=mx+c

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

$$P=(x_1,y_1), Q=(x_2,y_2)$$

$$R=(P+Q)=(x_3,y_3)$$

Let, P≠Q,

$$m=\frac{y_2-y_1}{x_2-x_1};$$

To find the intersection with E. we get

$$(\ mx+c\ )^2=x^3+Ax+B$$

$$or, 0=x^3-m^2x^2+\ldots$$

$$So, x_3=m^2-x_1-x_2$$

$$\Rightarrow y_3=m(x_1-x_3)-y_1$$

Doubling of a point

- Let P=Q, and take the derivative of Equation, derivation of y with respect to x.

$$y^2 = x^3 + ax + b$$

$$2y\frac{dy}{dx} = 3x^2 + A$$

$$\Rightarrow m = \frac{dy}{dx} = \frac{3x_1^2 + A}{2y_1}$$

$$If, y_1 \neq 0 \text{ (since then } P_1 + P_2 = \infty\text{):}$$

$$\therefore 0 = x^3 - m^2x^2 + \ldots$$

$$\Rightarrow x_3 = m^2 - 2x_1, y_3 = m(x_1 - x_3) - y_1$$

If Characteristics of field is not 2, 3:

$$y^2 = f(x) = x^3 + Ax + B$$

1. **Hence condition for no singularity is $4A^3+27B^2 \neq 0$**
2. **Generally, EC curves have no singularity**

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

For double roots,

$$x^3 + Ax + B = 3x^2 + A = 0$$

$$\Rightarrow x^2 = -A/3.$$

Also, $x^4 + Ax^2 + Bx = 0$,

$$\Rightarrow \frac{A^2}{9} - \frac{A^2}{3} + Bx = 0$$

$$\Rightarrow x = \frac{2A^2}{9B}$$

$$\Rightarrow 3\left(\frac{2A^2}{9B}\right)^2 + A = 0$$

$$\Rightarrow 4A^3 + 27B^2 = 0$$

For Instance Point Addition and Doubling

- Take an Elliptic Curve $E_{(4,20)}: y^2=x^3+4x+20 \in F_{29}$
- Set of points generated using the curve E is

O, (2,6) (4,19) (8,10) (13,23) (16,2) (19,16) (27,2),(0,7) (2,23) (5,7) (8,19) (14,6) (16,27) (20,3) (27,27), (0,22) (3,1) (5,22) (10,4) (14,23) (17,10) (20,26), (1,5) (3,28) (6,12) (10,25) (15,2) (17,19) (24,7), (1,24) (4,10) (6,17) (13,6) (15,27) (19,13) (24,22)

- Point addition :-where P=(5, 22), Q=(16, 27) , P+Q=(13,6)

  - $\lambda = (y_2-y_1)/(x_2-x_1) \bmod (29) = ((27-22)/(16-5)) \bmod 29=(5/11)\bmod 29$
    => (5*8) mod 29=11. (Because (11*8)=1mod 29)
    $x_3 = \lambda^2 - x_1 - x_2$ (29)=> ((11*11)-5-16)mod (29)=> 100 mod 29=13
    $y_3 = \lambda(x_1 - x_3) - y_1$ (29)=>(11(5-13)-22)mod 29 => -110 mod 29=6

- Point Doubling :- where P=(5,22) 2P=P+P=(14,6)

  - $\lambda = (3x_1^2+a)/ 2y_1$ (29) =>((3(5*5)+4)/(2*22))mod 29=>13
  - $x_3 = \lambda^2 - 2x_1$ (29)=> (13*13)-2*5 (29)=> 14
  - $y_3 = \lambda(x_1 - x_3) - y_1$ (29)=>(13(5-14)-22 (29) => 6.

Points on the Elliptic Curve (EC)

- Elliptic Curve over field L

$$E(L) = \{\infty\} \cup \{(x, y) \in L \times L \mid y^2 + \ldots = x^3 + \ldots\}$$

- It is useful to add the point at infinity
- The point is sitting at the top of the y-axis and any line is said to pass through the point when it is vertical
- It is both the top and at the bottom of the y-axis

The Points P,Q form Abelian Group

Given two points P,Q in *E(Fp)*, there is a third point, denoted by R=P+Q on *E(Fp)*, and the following relations hold for all P, Q, R in *E(Fp)*

- $P + Q = Q + P$ (*commutativity*)
- $(P + Q) + R = P + (Q + R)$ (*associativity*)
- $P + O = O + P = P$ (*existence of an identity element*)
- there exists $(-P)$ such that $-P + P = P + (-P) = O$ (*existence of inverses*)

Elliptic Curve on a finite set of Integers Z/5Z

- Consider $y^2 = x^3 + 2x + 3 \pmod 5$

$x = 0 \Rightarrow y^2 = 3 \Rightarrow$ no solution (mod 5)

$x = 1 \Rightarrow y^2 = 6 = 1 \Rightarrow y = 1,4 \pmod 5$

$x = 2 \Rightarrow y^2 = 15 = 0 \Rightarrow y = 0 \pmod 5$

$x = 3 \Rightarrow y^2 = 36 = 1 \Rightarrow y = 1,4 \pmod 5$

$x = 4 \Rightarrow y^2 = 75 = 0 \Rightarrow y = 0 \pmod 5$

- Then points on the elliptic curve are

(1,1) (1,4) (2,0) (3,1) (3,4) (4,0)

and the point at infinity: ∞

Using the finite fields we can form an Elliptic Curve Group where we also have a DLP problem which is harder to solve…

Continuation…

- How do we say that $y^2 = a(P)$ has solution with respect to F_p ?
- Using Legendre and Jacobi symbol.

$$\left(\frac{a}{p}\right) = \begin{cases} 1 & \text{if } a \text{ is a quadratic residue modulo } p \text{ and } a \not\equiv 0 \pmod{p}, \\ -1 & \text{if } a \text{ is a quadratic non-residue modulo } p, \\ 0 & \text{if } a \equiv 0 \pmod{p}. \end{cases}$$

(i) $\left(\frac{a}{p}\right) \equiv a^{(p-1)/2} \pmod{p}$. In particular, $\left(\frac{1}{p}\right) = 1$ and $\left(\frac{-1}{p}\right) = (-1)^{(p-1)/2}$. Hence $-1 \in Q_p$ if $p \equiv 1 \pmod 4$, and $-1 \in \overline{Q}_p$ if $p \equiv 3 \pmod 4$.

(ii) $\left(\frac{ab}{p}\right) = \left(\frac{a}{p}\right)\left(\frac{b}{p}\right)$. Hence if $a \in \mathbb{Z}_p^*$, then $\left(\frac{a^2}{p}\right) = 1$.

(iii) If $a \equiv b \pmod p$, then $\left(\frac{a}{p}\right) = \left(\frac{b}{p}\right)$.

(iv) $\left(\frac{2}{p}\right) = (-1)^{(p^2-1)/8}$. Hence $\left(\frac{2}{p}\right) = 1$ if $p \equiv 1$ or $7 \pmod 8$, and $\left(\frac{2}{p}\right) = -1$ if $p \equiv 3$ or $5 \pmod 8$.

(v) (*law of quadratic reciprocity*) If q is an odd prime distinct from p, then

$$\left(\frac{p}{q}\right) = \left(\frac{q}{p}\right)(-1)^{(p-1)(q-1)/4}.$$

In other words, $\left(\frac{p}{q}\right) = \left(\frac{q}{p}\right)$ unless both p and q are congruent to 3 modulo 4, in which case $\left(\frac{p}{q}\right) = -\left(\frac{q}{p}\right)$.

For Instance:

$$\left(\frac{158}{235}\right) = \left(\frac{2}{235}\right)\left(\frac{79}{235}\right) = (-1)\left(\frac{235}{79}\right)(-1)^{78\cdot 234/4} = \left(\frac{77}{79}\right)$$

$$= \left(\frac{79}{77}\right)(-1)^{76\cdot 78/4} = \left(\frac{2}{77}\right) = -1.$$

1. 4th property 235=3(8) yield -1
2. 5th property law of reciprocity
3. 5th property law of reciprocity
4. 4th property 77=5(8) yield -1

Implementation of scalar multiplication in ECC

kP=P+P+……+P; k times

k is scalar

Scalar Multiplication: MSB first

- Require $k=(k_{m-1},k_{m-2},\ldots,k_0)_2$, $k_{m-1}=1$
- Compute $Q=kP$
  - $Q=P$
  - For $i=m-2$ to 0
    - $Q=2Q$
    - If $k_i=1$ then
      - $Q=Q+P$
    - End if
  - End for
  - Return Q

Sequential Algorithm

Requires m point doublings and (m-1)/2 point additions on the average

Example

- **Compute 7P:**
  - $7=(111)_2$
  - 7P=2(2(P)+P)+P=> 2 iterations are required
  - Principle: First double and then add (accumulate)
- **Compute 6P:**
  - $6=(110)_2$
  - 6P=2(2(P)+P)

Scalar Multiplication: LSB first

- Require $k=(k_{m-1},k_{m-2},\ldots,k_0)_2$, $k_{m-1}=1$
- Compute $Q=kP$
  - $Q=0$, $R=P$
  - For $i=0$ to $m-1$
    - If $k_i=1$ then
      - $Q=Q+R$
    - End if
    - $R=2R$
  - End for
  - Return Q

Can **Parallelize**…

What you are doubling and what you are accumulating are different…

On the average m/2 point Additions and m/2 point doublings

Example for scalar multiplication

- **Compute 7P**, $7=(111)_2$, Q=0, R=P
  - Q=Q+R=0+P=P, R=2R=2P
  - Q=P+2P=3P, R=4P
  - Q=7P, R=8P
- **Compute 6P**, $6=(110)_2$, Q=0, R=P
  - Q=0, R=2R=2P
  - Q=0+2P=2P, R=4P
  - Q=2P+4P=6P, R=8P

Scalar multiplication using NAF

- In binary the remainders are {0,1}, but in NAF remainders are {0,1,-1},k is a scalar.

(i) k has a unique NAF denoted NAF(k).

(ii) NAF(k) has the fewest nonzero digits of any signed digit representation of k.

(iii) The length of NAF(k) is at most one more than the length of the binary representation of k.

(iv) If the length of NAF(k) is l, then $2^l/3 < k < 2^{l+1}/3$.

(v) The average density of nonzero digits among all NAFs of length l is approximately 1/3.

INPUT: A positive integer k.
OUTPUT: NAF(k).

1. $i \leftarrow 0$.
2. While $k \geq 1$ do
   - 2.1 If k is odd then: $k_i \leftarrow 2 - (k \bmod 4)$, $k \leftarrow k - k_i$;
   - 2.2 Else: $k_i \leftarrow 0$.
   - 2.3 $k \leftarrow k/2$, $i \leftarrow i + 1$.
3. Return$(k_{i-1}, k_{i-2}, \ldots, k_1, k_0)$.

Continuation…

- For instance k=31.

  (Binary) = (NAF)

  (11111) = (1 0 0 0 0 -1)

- For instance k=61

  (Binary) = (NAF)

  (111101) = (1 0 0 0 -1 0 1)

Scalar Multiplication in ECC (sP=P+P+……+P) is adding P to P, s number of times.

S.No	Method	Binary notation of "s" , P is Point	add-and-double	Operations
1	LSB	$31P=(11111)_2$	16P 8P 4P 2P P 1 1 1 1 1 **31P** 15P 7P 3P P	**4D+4A** **(n-1)D+(nz-1)A**
2	MSB	$10P=(1010)_2$	P 2P 4P 10P 1 0 1 0 P 2P 5P **10P**	**3D+1A** **(n-1)D+(nz-1)A**
3	**NAF**	$31P=(10000\text{-}1)_2$	32P 16P 8P 4P 2P P 1 0 0 0 0 -1 **31P** - P	**5D+1A** **(n-1)D+(nz-1)A**
4	**Montgomery Algorithm**	$10P=(1010)_2$	2P 3P 6P 11P 1 0 1 0 P 2P 5P **10P**	**4D+3A** **(n)D+(n-1)A**

A: Point addition, D: point doubling , n is length of binary notation of scalar s, nz: number of non-zeros in binary notation of "s".

Encode msg(M) to Msg point on the curve

M to P_M point conversion.

1. $X_j = 100 * M + j$; where $0 \leq M < (P/100)$, $0 \leq j < 100$.
2. $S_j = (X_j)^3 + aX_j + b \pmod P$, $(X_0, X_1, \ldots\ldots, X_{99})$.
3. Legendre(S_j, P)=1.

   Then $P \equiv 3 \bmod 4$. $Y_j = (S_j)^{(p+1)/4} \pmod P$, break.

   (Algo 2.143)(modular exponensation a^k mod p)

   else inc(j) , goto step 1.

output $P_M = (X_j, Y_j)$.

P_M to M convertion.

1. $M = \text{Floor}(X_j/100)$.

Encryption & Decryption Process using Elliptic Curve Cryptosystems (ECC)

Key Generation:

- Suppose Alice wants to send to Bob an encrypted message.
  - Both agree on a base point B. Alice and Bob create public/private keys.
  - n is order of G contains all points, n=#E(Fp), nB=O.
    - Alice
      - Private Key a ={1,2,……..n-1}
      - Public Key = P_A = a * B
    - Bob
      - Private Key b = {1,2,……..,n-1}
      - Public Key = P_B = b * B
  - Alice takes plaintext message M, and encodes it onto a point, P_M, from the elliptic group.

Encryption and Decryption (Cont…)

- Alice chooses another random integer, k=[1, p-1]
- The ciphertext is a pair of points

  $P_C = [\ C1=(kB),\ C2=(P_M + kP_B)\ ]$

- To decrypt P_M from C1,C2, Bob computes ,using private key, b

  $P_M = C2 - b * C1.$
- Bob then takes this product and subtracts it from the second point from P_C

  $(P_M + kP_B) - [b(kB)] = P_M + k(bB) - b(kB) = P_M$
- Bob then decodes P_M to get the message, M.

Summary of ECC

- "**Hard problem**" analogous to discrete log
  - **Q=kP, where Q, P belong to a prime curve**
    given k, P → "easy" to compute Q
    given Q, P → "hard" to find k
  - **known as the elliptic curve discrete logarithm problem**
    - **k must be large enough**
- ECC security relies on elliptic curve logarithm problem
  - compared to factoring, can use much smaller key sizes than with RSA etc

Topic wise Preferred Textbooks

- Group laws and scalar multiplication
  - Guide to ECC p.no: 80,96.
- Mathematical formulas, Legendre and jacobi symbol
  - Hand book of applied cryptography p.no: 73
- Massage point representation, El-Gamal Elliptic curve cryptography
  - Elliptic curve cryptography & number theory(pg no – 173,174,175)

- Implementation of ECC over $GF(2^m)$

Binary Field GF(2^m)

- Binary Field

$$\mathbb{F}_{2^m} = \{a_{m-1}z^{m-1} + a_{m-2}z^{m-2} + \cdots + a_2z^2 + a_1z + a_0 \ : \ a_i \in \{0, 1\}\}.$$

- Field elements of Binary field F_{2^4}

0	z^2	z^3	z^3+z^2
1	z^2+1	z^3+1	z^3+z^2+1
z	z^2+z	z^3+z	z^3+z^2+z
$z+1$	z^2+z+1	z^3+z+1	$z^3+z^2+z+1.$

Conti…

The following are some examples of arithmetic operations in $\mathbb{F}_{2^4}$ with reduction polynomial $f(z) = z^4 + z + 1$.

(i) Addition: $(z^3 + z^2 + 1) + (z^2 + z + 1) = z^3 + z$.

(ii) Subtraction: $(z^3 + z^2 + 1) - (z^2 + z + 1) = z^3 + z$. (Note that since $-1 = 1$ in $\mathbb{F}_2$, we have $-a = a$ for all $a \in \mathbb{F}_{2^m}$.)

(iii) Multiplication: $(z^3 + z^2 + 1) \cdot (z^2 + z + 1) = z^2 + 1$ since

$$(z^3 + z^2 + 1) \cdot (z^2 + z + 1) = z^5 + z + 1$$

and

$$(z^5 + z + 1) \bmod (z^4 + z + 1) = z^2 + 1.$$

(iv) Inversion: $(z^3 + z^2 + 1)|^{-1} = z^2$ since $(z^3 + z^2 + 1) \cdot z^2 \bmod (z^4 + z + 1) = 1$.

Conti…

- Binary multiplication:

INPUT: Binary polynomials $a(z)$ and $b(z)$ of degree at most $m-1$.
OUTPUT: $c(z) = a(z) \cdot b(z) \bmod f(z)$.

1. If $a_0 = 1$ then $c \leftarrow b$; else $c \leftarrow 0$.
2. For i from 1 to $m-1$ do
   2.1 $b \leftarrow b \cdot z \bmod f(z)$.
   2.2 If $a_i = 1$ then $c \leftarrow c + b$.
3. Return(c).

- Binary squaring:

$$a(z)^2 = a_{m-1}z^{2m-2} + \cdots + a_2 z^4 + a_1 z^2 + a_0.$$

Group laws with respect to 2^m

Group law for non-supersingular $E/\mathbb{F}_{2^m}$: $y^2 + xy = x^3 + ax^2 + b$

1. *Identity*. $P + \infty = \infty + P = P$ for all $P \in E(\mathbb{F}_{2^m})$.
2. *Negatives*. If $P = (x, y) \in E(\mathbb{F}_{2^m})$, then $(x, y) + (x, x + y) = \infty$. The point $(x, x + y)$ is denoted by $-P$ and is called the *negative* of P; note that $-P$ is indeed a point in $E(\mathbb{F}_{2^m})$. Also, $-\infty = \infty$.
3. *Point addition*. Let $P = (x_1, y_1) \in E(\mathbb{F}_{2^m})$ and $Q = (x_2, y_2) \in E(\mathbb{F}_{2^m})$, where $P \neq \pm Q$. Then $P + Q = (x_3, y_3)$, where

$$x_3 = \lambda^2 + \lambda + x_1 + x_2 + a \quad \text{and} \quad y_3 = \lambda(x_1 + x_3) + x_3 + y_1$$

   with $\lambda = (y_1 + y_2)/(x_1 + x_2)$.
4. *Point doubling*. Let $P = (x_1, y_1) \in E(\mathbb{F}_{2^m})$, where $P \neq -P$. Then $2P = (x_3, y_3)$, where

$$x_3 = \lambda^2 + \lambda + a = x_1^2 + \frac{b}{x_1^2} \quad \text{and} \quad y_3 = x_1^2 + \lambda x_3 + x_3$$

   with $\lambda = x_1 + y_1/x_1$.

Koblitz Curves

- Koblitz curves are very efficient and it doesn't require doubling.

$$y^2 + xy = x^3 + ax^2 + b \text{ where } a \in \{0,1\}, b = 1$$

$$E_a = E_1 = y^2 + xy = x^3 + x^2 + 1 =$$
$$E_0 = y^2 + xy = x^3 + 1$$

$$\textit{Frobenius map } \tau: E_a(F_{2^m}) \to E_a(F_{2^m}), \tau(\alpha) = \alpha, \tau(x,y) = (x^2, y^2)$$

$$(\tau^2 + 2)P = \mu\tau(P) \text{ for all } P \in E_a(\mathbb{F}_{2^m}),$$

$$\tau^2 - \mu\tau + 2 = 0, \text{with } \mu = (-1)^{1-a} \qquad \tau = \frac{-1 + \sqrt{-7}}{2}$$

$$2 = \tau - \tau^2, \text{with } a = 1, \mu = 1$$

$$7 = \tau^5 + \tau + 1 = 100011, \text{with } a = 1, \mu = 1$$

Scalar multiplication

- Multiplying koblitz curves . Time complexity (m/3)A.

INPUT: Integer $k \in [1, n-1]$, $P \in E(\mathbb{F}_{2^m})$ of order n.
OUTPUT: kP.

1. Use Algorithm 3.65 to compute $\rho' = k \text{ partmod } \delta$.
2. Use Algorithm 3.61 to compute $\text{TNAF}(\rho') = \sum_{i=0}^{l-1} u_i \tau^i$.
3. $Q \leftarrow \infty$.
4. For i from $l-1$ downto 0 do
   - 4.1 $Q \leftarrow \tau Q$.
   - 4.2 If $u_i = 1$ then $Q \leftarrow Q + P$.
   - 4.3 If $u_i = -1$ then $Q \leftarrow Q - P$.
5. Return(Q).

Research Directions

- Proposed algorithms will perform more efficient if we make parallel execution of statements.
- Trying to embed updated ECC on Koblitz curves in different places where older methods are using for security. For instance we are trying to embed ECC on **SIP Protocol in VOIP**.
- Increase the efficiency of ECC further using Hyper elliptic curves and Cab curves. Trying to increase the research to the maximum level which is useful for real time problems.
- Finding different methods to represent a scalar like partitioning scalar into 2 parts applying different ways to represent first and second part.
- $\boldsymbol{\tau}^n$ -NAF generalization where n=3, 4, …

References

1. J. Lopez and R. Dahab, “Fast Multiplication on Elliptic Curves over GF(2^m) without pre-computation”, CHES 1999
2. K. Fong etal, “Field Inversion and Point Halving Revisited”, IEEE Trans on Comp, 2004
3. Koblitz.N., “Elliptic Curve Cryptosystems,” Math. Computation, vol. 48, pp. 203-209, Jan. 1987.
4. Solinas. J.A., “Efficient Arithmetic on Koblitz Curves,” Design, Codes and Cryptography, vol. 19, pp. 195-249, Mar. 2000.
5. Avanzi.R.M., Ciet. M., and Sica.F. Faster Scalar Multiplication on Koblitz Curves combining Point Halving with the Frobenius Endomorphism. Proceedings of PKC 2004, LNCS 2947, Springer 2004.
6. Knudsen. E.W., Elliptic Scalar Multiplication Using Point Halving. In: Proceedings of ASIACRYPT 1999, LNCS 1716, pp. 135-149. Springer, 1999.
7. Sujoy Sinha Roy, Chester Rebeiro, Debdeep Mukhopadhyay, Junko Takahashi and Toshinori Fukunaga, Scalar multiplication on Koblitz curves using $\tau 2$ – NAF, 2009.
8. Fong. K., Hankerson. D., Lopez. J., and Menezes. A,”Field inversion and point halving revisited”. IEEE Transactions on Computers 53(8):1047-1059, 2004.

Books:

- [Slides borrowed from Prof. D. Mukhopadhyay, IIT Kharagpur]. Microsoft Powhttps://cse.iitkgp.ac.in/~debdeep/pres/TI/ecc.pdferPoint - ecc (iitkgp.ac.in).
- *Elliptic Curves: Number Theory and Cryptography, by Lawrence C. Washington*
- *Hand book of applied cryptography, Alfred J. Menezes*
- *Guide to Elliptic Curve Cryptography, Darrel R. Hankerson, A. Menezes and A. Vanstone*
- *http://cr.yp.to/ecdh.html (Daniel Bernstein)*

Thank You

Doubts ?