ELEC 473 Cryptography and Network Security

6. Public Key Cryptography

Instructor: Dr. Jianbing Ni

Fall 2021

Review of C5 Topic 3

- Z_p^* is a cyclic group, that is $\exists\ g \in Z_p^*$ such that $\{1, g, g^2, g^3, \ldots, g^{p-2}\} = Z_p^*$, g is a generator of Z_p^*
- The Repeated Squaring Algorithm
- Intractable Problems with Primes: In G, $Dlog_g (g^x) = x$, where x in $\{0,\cdots, q-1\}$.
- DLog is hard in G if for all efficient alg. A:

  $Pr_{g \leftarrow G,\ x \leftarrow Z_q} [A(G, q, g, g^x) = x] < \text{negligible}$

  (1) Z_p^* for large p, (2) Elliptic curve groups mod p
- Intractable Problems with Composites:
  - Factor a random N that $N = p \cdot q$ where p,q are n-bit primes
  - Find x in Z_N s.t. $f(x) = 0$ in Z_N, where $N = p \cdot q$ where p,q are n-bit primes

Outline

- RSA Cryptosystem
- ElGamal Cryptosystem
- Elliptic Curve Cryptography (ECC)

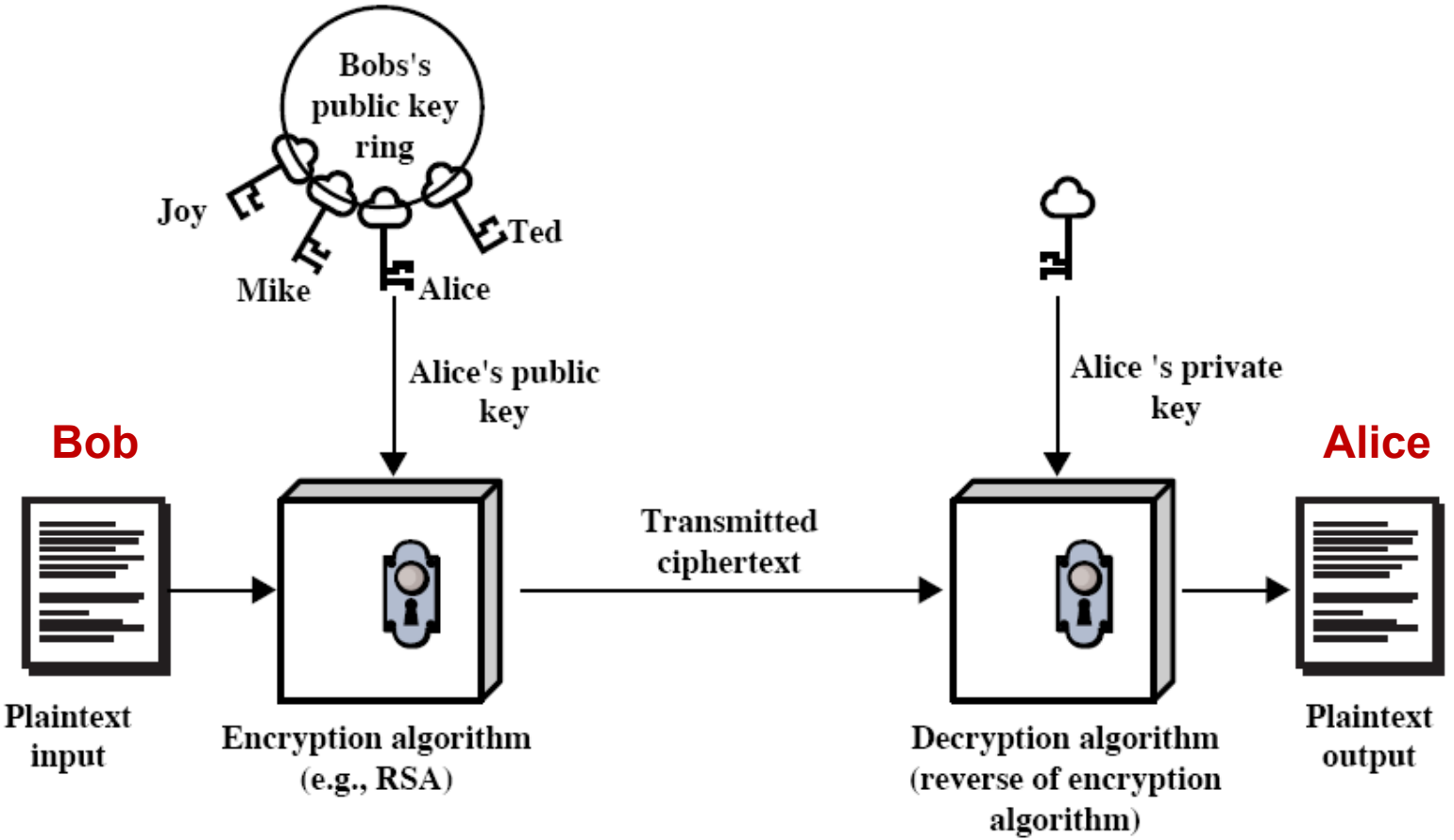
Public Key Cryptography

- Symmetric-key cryptography requires the prior communication of the key K between Alice and Bob, using a secure channel, before any ciphertext is transmitted.
- Asymmetric-key cryptography.
- Invented by Whitfield Diffie & Martin Hellman 1976.
- Developed to address two main issues:
  - key distribution
  - Authentication

Public Key Cryptography

- The advent of asymmetric-key cryptography DOES NOT eliminate the need for symmetric-key cryptography.
- Each user has a pair of keys: a public key and a private key.
- The public key is used for encryption.
  - The key is known to the public.
- The private key is used for decryption.
  - The key is only known to the owner.

Public Key Cryptography



Trapdoor One-way Function

The main idea behind asymmetric-key cryptography is the concept of the trapdoor one-way function.

Easy: $x \xrightarrow{f} y$

Hard: $x \xleftarrow{f^{-1}} y$

Easy: $x \xleftarrow[\text{trapdoor}]{f^{-1}} y$

Use trapdoor as the private key.

- Many public-key cryptosystems are based on trapdoor one-way functions.

RSA Cryptosystem

Based on the one-way function of *e*'th root

$f : x \rightarrow y = x^e \bmod n$ (easy)

$f^{-1} : y \rightarrow x = \sqrt[e]{y} \bmod n$ (hard)

Let n be a composite number and $e>1$

$n = pq$

When does $\sqrt[e]{y}$ **in** Z_n^* exist? Can we compute it efficiently?

RSA Cryptosystem

欧拉函数φ(n)是小于等于n的正整数中与n互质的数的数目

When does $y^{1/e}$ in Z_n^* exist? Can we compute it efficiently?

Suppose gcd(e , φ(n)) = 1, Then for all y in Z_n^*: $y^{1/e}$ exists in Z_n^* and is easy to find.

Proof: let $d = e^{-1} \bmod \varphi(n) \Rightarrow$ $d \cdot e = 1 \bmod \varphi(n) \Rightarrow$

$\exists\, k \in Z_{\varphi(n)} : d \cdot e = k \cdot \varphi(n) + 1 \Rightarrow y^{d \cdot e} = y^{k \cdot \varphi(n) + 1} = y$ in Z_n^*

Then $y^{1/e} = y^d$ in Z_n^*

So, computing $\sqrt[e]{y} \bmod n$ needs to know $\varphi(n)$, which requires the factorization of n

RSA Cryptosystem

The first public key cryptography (RSA) is designed by Rivest, Shamir and Adleman in 1977.

Used for data encryption and digital signature.

The security of RSA is based on the hard problems of large number factorization

A user wishing to set up an RSA cryptosystem will:

- Choose a pair of public/private keys: (PK, SK).
- Publish the public (encryption) key.
- Keep secret the private (decryption) key.

RSA Setup

- Select two large primes p and q at random
- Compute n=pq. Note $\varphi(n)=(p-1)(q-1)$.
- Select an encryption key e satisfying $1<e<\varphi(n)$ and $\gcd(e, \varphi(n))=1$. (i.e., $e \in Z_{\varphi(n)}^*$, $e \neq 1$.)
- Compute the decryption key: $d=e^{-1} \bmod \varphi(n)$.
  - $ed \equiv 1 \bmod \varphi(n)$
  - d is the inverse of e mod $\varphi(n)$
- Public key PK=(n, e); private key SK=d
- Important: p, q, and $\varphi(n)$ must be kept secret.
- p and q can be deleted after generating $\varphi(n)$.

RSA Encryption and Decryption

- Suppose Bob is to send a secret message m to Alice.
- To encrypt, Bob will
  - Obtain Alice's public key $PK_{Alice}=(n, e)$
  - Encrypt m as $c=m^e \bmod n$.
  - Note: $m \in Z_n^*$
- To decrypt the ciphertext c, Alice will compute
  - $m=c^d \bmod n$, using her private key $SK_{Alice} = d$
- Which key will Alice use to encrypt her reply to Bob?

$m = c^d \bmod n$

$= (m^e \bmod n)^d \bmod n$

$= m^{ed} \bmod n = m^{k\varphi(n)+1} \bmod n$

- $ed \equiv 1 \bmod (\varphi(n))$

$ed \bmod \varphi(n) = 1 \bmod \varphi(n)$

$= k\varphi(n)+1$

RSA Question

- Select two primes: p=17, q=11
- Compute the modulus n=pq=187
- Compute φ(n)=(p-1)(q-1)=160
- Select e between 1 and 160 such that gcd(e, 160)=1, let e=7
- Compute $d=e^{-1} \bmod \varphi(n)=7^{-1} \bmod 160 = 23$
- Public Key: PK=(n, e)=(187, 7)
- Private key: SK=d=23

- Suppose m=88
- Encryption: $c=m^e \bmod n = 887 \bmod 187 = 11$
- Decryption: $m=c^d \bmod n = 1123 \bmod 187 = 88$

Topic 1 RSA Cryptosystem

RSA Cryptosystem Security

Instructor: Dr. Jianbing Ni

Fall 2021

RSA Cryptosystem Security

RSA Hard Problem: It is hard to compute m from $m^e \bmod n$

- If an adversary can factor $n=pq$, then?

If the adversary knows p and q, he can compute $\varphi(n)=(p-1)(q-1)$. Thus, he can further computer $d=e^{-1} \bmod \varphi(n)$ using extended Euclidean algorithm.

$ax+by=\gcd(a,b)$

- If an adversary cannot factor $n=pq$, then?

If the RSA hard problem can be solved, can the adversary factor n? This is an open problem, but we believe it is true generally. Therefore, the security of RSA depends on the difficulty of factoring the large integer n.

RSA Cryptosystem Security

Is there other approaches in which the integer factoring is unnecessary?

Proof: to compute *φ(n)* from *n* ⇒ factoring *n*.

Suppose *n=pq*, *p>q*， and *φ(n)=(p-1)(q-1)*， then,

p+q=n-φ(n)+1

and

$$p-q=\sqrt{(p+q)^2-4n}=\sqrt{[n-\varphi(n)+1]^2-4n}$$

$$p=\frac{1}{2}[(p+q)+(p-q)]$$

$$q=\frac{1}{2}[(p+q)-(p-q)]$$

Therefore, computing φ(n) without factoring n is not easier than factoring n.

RSA Cryptosystem Security

Best known alg. (NFS): run time exp($\tilde{O}(\sqrt[3]{n})$) for n-bit integer

- RSA – 129 (428bit) is factored in April 1994
- RSA – 130 is factored in April 1996
- RSA – 140 is factored in February 1999
- RSA – 155 (512bit) is factored in August 1999
- RSA – 174 (576bit) is factored in December 2003
- Current world record: **RSA-768** (232 decimal digits) in December 12, 2009

Work: two years on hundreds of machines

Factoring a 1024-bit integer ($100,000): about 1000 times harder

⇒ likely possible before 2020

Choosing p and q

To ensure the security of RSA, the following requirements should be met:

(1) $|p-q|$ should be large

(2) The length of p and q should be proximate

(3) $p-1$ and $q-1$ should have large prime factor

(4) $\gcd(p-1, q-1)$ should be small

Summary

- Model of Public Key Cryptography
- Trapdoor One-Way Function
- RSA Cryptosystem
- RSA Cryptosystem Security

Practice Question

A common way to speed up RSA decryption incorporates the Chinese Remainder Theorem, as follows. Suppose that $d_k(y)=y^d \bmod n$ and $n=pq$. Define $d_p=d \bmod (p-1)$ and $d_q=d \bmod (q-1)$; and let $M_p=q^{-1} \bmod p$ and $M_q=p^{-1} \bmod q$. Then, consider the following algorithm:

Algorithm 1. CRT-Optimized RSA Decryption $(n, d_p, d_q, M_p, M_q, y)$

$x_p \leftarrow y^{dp} \bmod p$

$x_q \leftarrow y^{dq} \bmod q$

$x \leftarrow M_p q x_p + M_q p x_q \bmod n$

return (x)

The Algorithm 1 replaces an exponentiation modulo n by modular exponentiations modulo p and q. If p and q are l-bit integers and exponentiation modulo an l-bit integer takes time cl^3, then the time to perform the required exponentiation is reduced from $c(2l)^3$ to $2cl^3$, a saving of 75%. The final step, involving the Chinese Remainder Theorem, require $O(l^2)$ if d_p, d_q, M_p, and M_q have been pre-computed.

a. Prove that the value x returned by Algorithm 1 is, in fact, $y^d \bmod n$.

b. Given that p=1511, q=2003, and d=1234577, compute d_p, d_q, M_p, and M_q.

c. Given the above values of p, q, and d, decrypt the ciphertext y=152702 using Algorithm 1.

Thanks

ELEC 473 Cryptography and Network Security

6. Public Key Cryptography

Instructor: Dr. Jianbing Ni

Fall 2021

Review of C6 Topic 1

- Each user has a pair of keys: a public key and a private key.
- The public key is used for encryption, known to the public.
- The private key is used for decryption, only known to the owner.
- The main idea behind asymmetric-key cryptography is trapdoor one-way function
- RSA cryptosystem is based on one-way property of e'th root

$$f : x \rightarrow y = x^e \bmod n \qquad \text{(easy)}$$

$$f^{-1} : y \rightarrow x = y^{1/e} \bmod n \qquad \text{(hard)}$$

- Suppose $\gcd(e, \varphi(n)) = 1$, for all y in Z_n^*: $y^{1/e}$ exists in Z_n^* and is easy to find.
- RSA Cryptosystem: Setup (Key Generation), Encryption and Decryption
- RSA Hard Problem: It is hard to compute m from $m^e \bmod n$
- The relationship between RSA hard problem and the prime factorization of n.
- Computing $\varphi(n)$ without factoring n is not easier than factoring n.

Outline

- RSA Cryptosystem
- ElGamal Cryptosystem
- Elliptic Curve Cryptography (ECC)

The Diffie-Hellman Protocol (1976)

Invited by Whitfield Diffie & Martin Hellman 1976 in their paper

"New Directions in Cryptography"

The first public-key cryptosystem based on DLog hard problem.

Agree a secret key between Alice and Bob (Key Agreement)

The Diffie-Hellman Protocol (1976)

Fix a finite cyclic group G (e.g $G = Z_p^*$) of order q

Fix a generator g in G (i.e. $G = \{1, g, g^2, g^3, \ldots, g^{q-1}\}$)

Alice **Bob**

choose random **a** in {1,…,q} choose random **b** in {1,…,q}

$A = g^a$

$B = g^b$

$B^a = (g^b)^a = k_{AB} = g^{ab} = (g^a)^b = A^b$

The Diffie-Hellman Protocol Security

Computing g^{ab} from (g, g^a, g^b) is strongly related to the DLog hard problem. If computing the DLog (base g) in G were easy, then it is easy to compute g^{ab} from (g, g^a, g^b):

Given:

$(g, g^a\ g^b)$

One can efficiently compute g^{ab} in the following way:

- Compute a by taking the discrete log of g^a to base g
- Compute g^{ab} by exponentiation $g^{ab}=(g^b)^a$

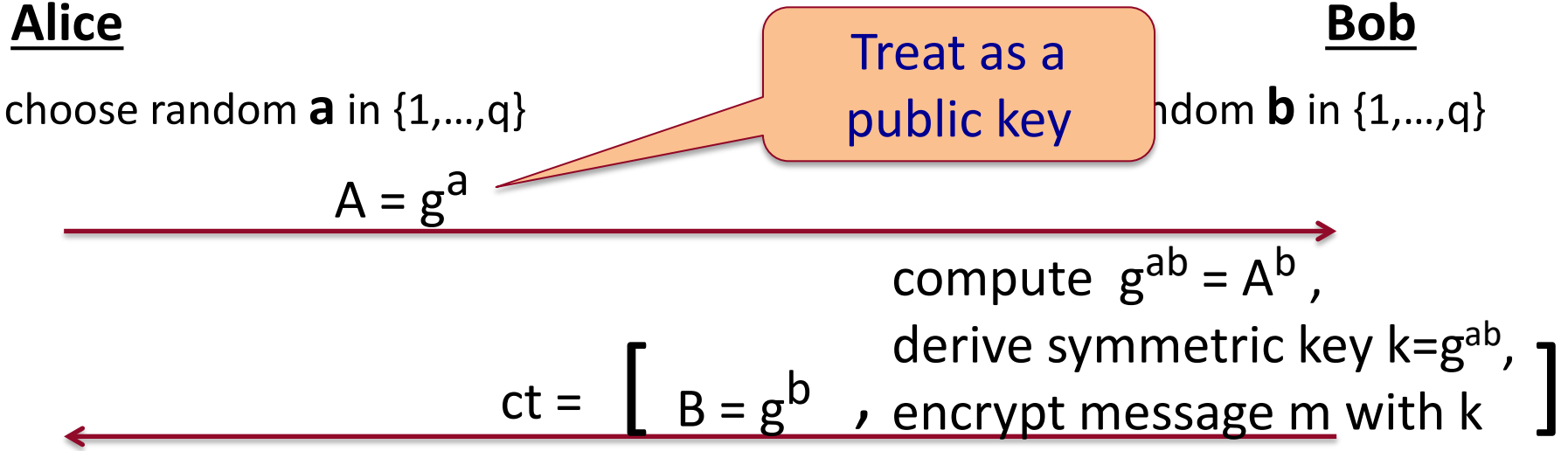
Computing DLog is the only known method for computing g^{ab} from (g, g^a, g^b) .

It is an open problem to determine whether the DLog problem is equivalent to the problem of computing g^{ab} from (g, g^a, g^b).

ElGamal: Converting to Pub-Key Enc. (1984)

Fix a finite cyclic group G (e.g $G = Z_p^*$) of order q

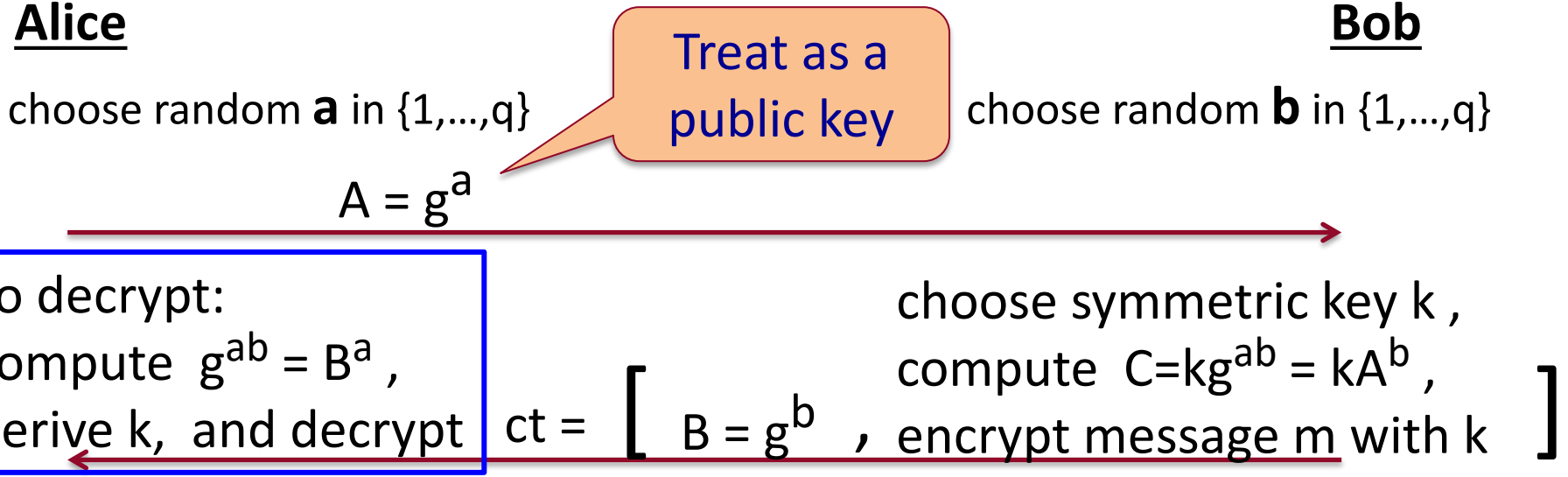
Fix a generator g in G (i.e. $G = \{1, g, g^2, g^3, \ldots, g^{q-1}\}$)



ElGamal: Converting to Pub-Key Enc. (1984)

Fix a finite cyclic group G (e.g $G = Z_p^*$) of order q

Fix a generator g in G (i.e. $G = \{1, g, g^2, g^3, \ldots, g^{q-1}\}$)



ElGamal Encryption (1984)

- G: finite cyclic group of order q
- $(sk_A, pk_A)=(a, g^a)$: Alice's secret-public key pair

Bob:

$E(pk=(g, pk_A), m)$:

$b \xleftarrow{R} Z_q$, $u \leftarrow g^b$,

$v \leftarrow m \cdot pk_A^{\,b}$,

output (u, v)

Alice:

$D(sk=a, (u, v))$:

$m \leftarrow v \cdot u^{-a}$

output m

$m \cdot pk_A^{\,b} \cdot (g^b)^{-a}$

$m \cdot g^{ab} \cdot g^{-ab}$

ElGamal Encryption Question

Let p = **23,** select a generator g = **11**

Choose a private key x = **6**

a. Compute the public key

b. Use the public key to encrypt m = 10

c. Use the secret key to decrypt the obtained ciphertext in b.

a. $pk = g^{\overset{\text{private}}{a}} \bmod p = 11^6 \bmod 23 = 9$

b. $q = p - 1 = 22$ $\quad b \xleftarrow{R} Z_q \Rightarrow b = 3 \Rightarrow u = g^b = 11^3 \bmod 23 = 20$

$V = m \cdot pk_A{}^b = 10 \cdot 9^3 \bmod 23 = 22$

c. $m = V u^{-a} = 22 \cdot 20^{-6} \bmod 23$

ElGamal Security

The security of the ElGamal cryptosystem is based on DLog problem.

- If DLog problem can be solved, ElGamal cryptosystem is insecure
- If DLog problem is hard, ElGamal cryptosystem is secure

Def: We say that **DLog is hard in G** if for all efficient alg. A:

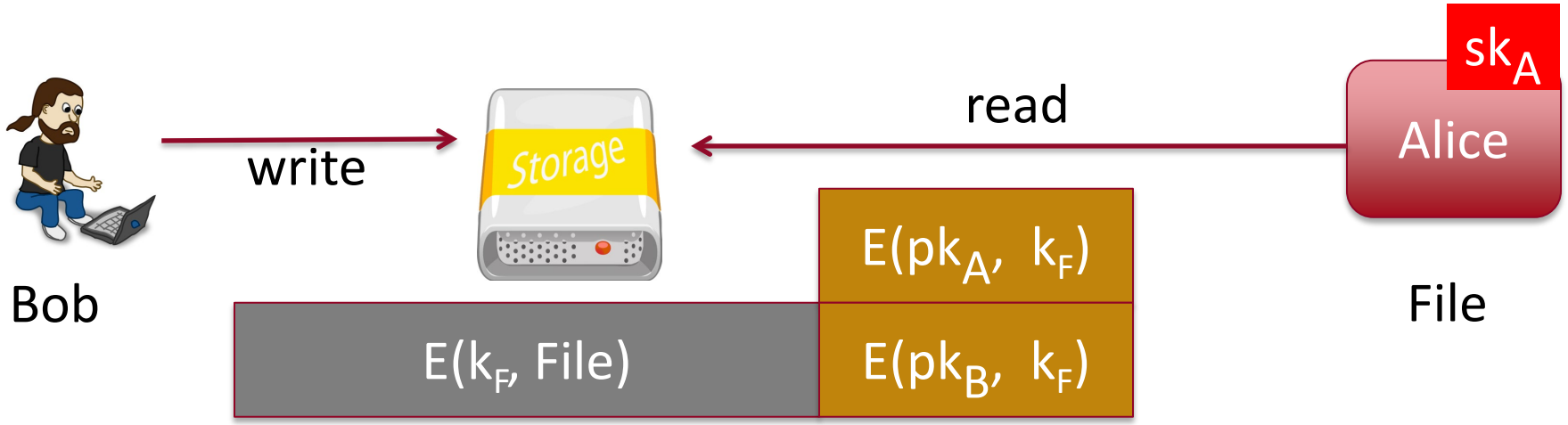
$$\Pr_{g \leftarrow G,\, x \leftarrow Z_q} [A(G, q, g, g^x) = x] < \text{negligible}$$

Public-key Encryption Applications

Key exchange (e.g. in HTTPS)

Encryption in non-interactive settings:

- Secure Email: Bob has Alice's pub-key and sends her an email
- Encrypted File Systems

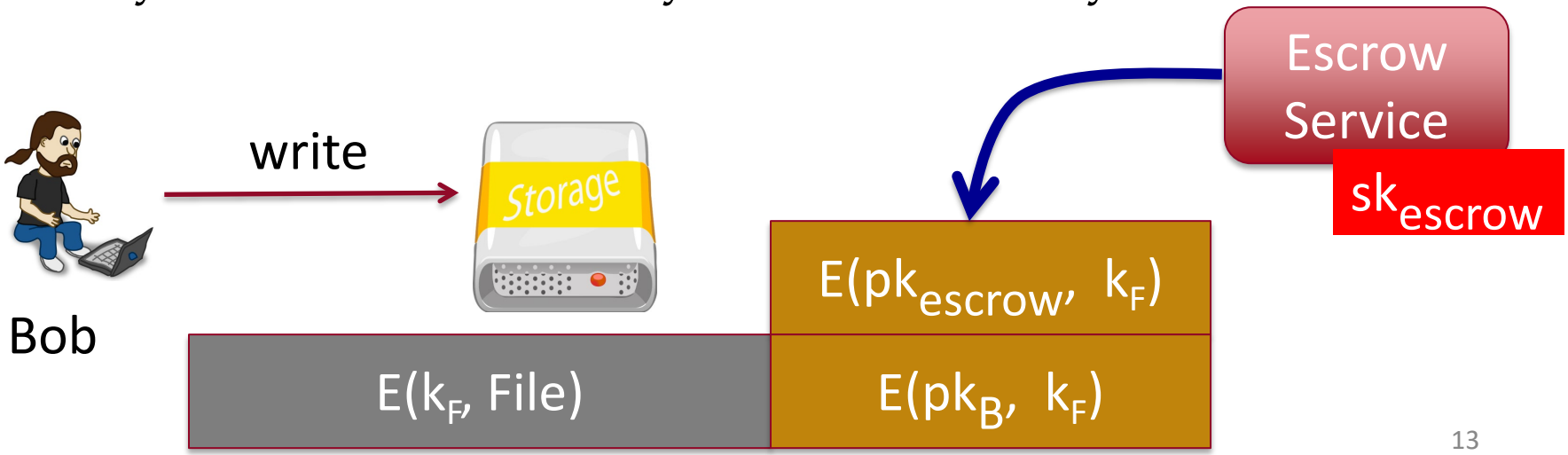


Public-key Encryption Applications

Key exchange (e.g. in HTTPS)

Encryption in non-interactive settings:

- Secure Email: Bob has Alice's pub-key and sends her an email
- Encrypted File Systems
- Key escrow: data recovery without Bob's key



Summary

- ElGamal Cryptosystem
- ElGamal Security
- Public Key Encryption Applications

Assignment

Consider an ElGamal scheme with a common prime q=71 and a generator g=7.

a. If B has public key Y_B=3 and A chose the random integer k=2, what is the ciphertecxt of M=30?

b. If A now chooses a different value of k, so that the encoding of M=30 is C=(59, C_2), what is the integer C_2?

Thanks

ELEC 473 Cryptography and Network Security

6. Public Key Cryptography

Instructor: Dr. Jianbing Ni

Fall 2021

Review of C6 Topic 2

- Applications of Public-key Cryptosystems
  - Key exchange (e.g. in HTTPS)
  - Secure Email: Bob has Alice's pub-key and sends her an email
  - Encrypted File Systems
  - Key escrow: data recovery without Bob's key
- ElGamal Cryptosystem
  - G: finite cyclic group of order q
  - $(sk_A, pk_A)=(a, g^a)$: Alice's private-public key pair

E(pk=(g, pk_A), m) :

$b \leftarrow Z_q$, $u \leftarrow g^b$,

$v \leftarrow m \cdot pk_A^{\,b}$,

output (u, v)

D(sk=a, (u, v)) :

$m \leftarrow v \cdot u^{-a}$

output m

- The security of the ElGamal cryptosystem is based on DLog problem

Outline

- RSA Cryptosystem
- ElGamal Cryptosystem
- Elliptic Curve Cryptography (ECC)

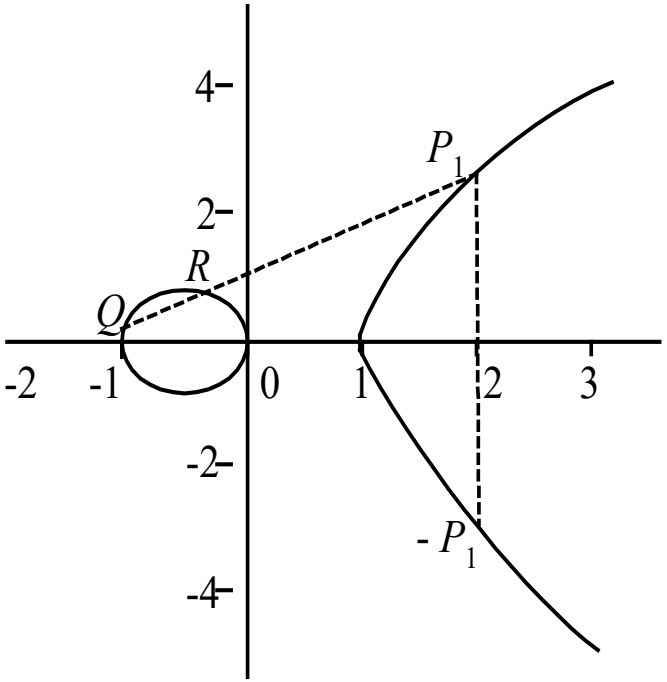
Elliptic Curves Over R

Let a and b be real numbers. An *elliptic curve* E over the field of real numbers **R** is the set of points (x,y) with x and y in **R** that satisfy the equation

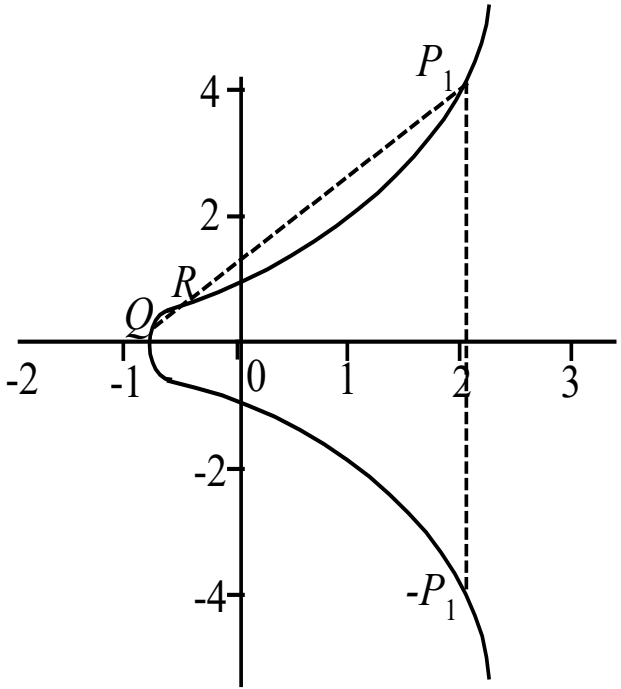
$$E = \left\{ (x, y) \in \boldsymbol{R} \times \boldsymbol{R} \middle| y^2 = x^3 + ax + b \right\} \cup \{ \mathcal{O} \}$$

Where $a, b \in \boldsymbol{R}$, $4a^3 + 27b^2 \neq 0$ and $\mathcal{O}$ is called the *point at infinity*.

Examples



(a) $y^2=x^3-x$



(b) $y^2=x^3+x+1$

Group Operation + Over E

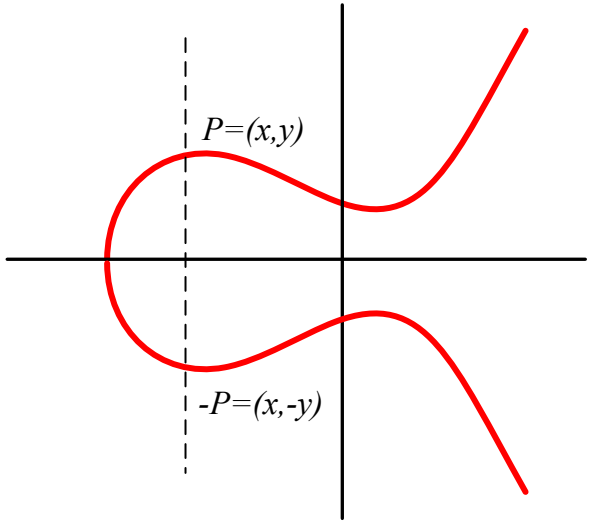
- (E, +) is an abelian group
- The point of infinity, O, will be the identity element

Given $P, Q \in E, P = (x_1, y_1), Q = (x_2, y_2)$

$$P + \mathrm{O} = \mathrm{O} + P$$

If $x_1 = x_2$, and $y_1 = -y_2$, then $P + Q = \mathrm{O}$

$(\text{i.e.} - P = -(x_1, y_1) = (x_1, -y_1))$



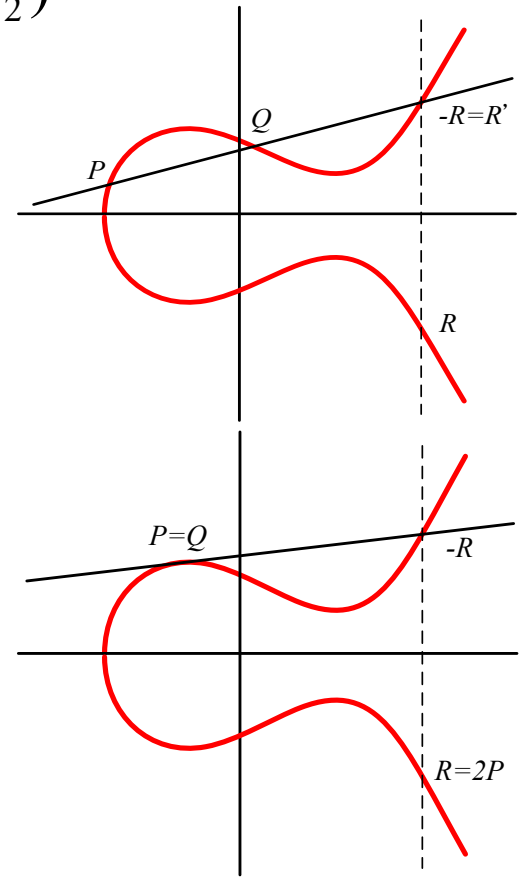
Group Operation + Over E

- Given $P, Q \in E, P = (x_1, y_1), Q = (x_2, y_2)$
  Compute $R = P + Q = (x_3, y_3)$
  - Addition $(P \neq Q)$

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1}$$
$$x_3 = \lambda^2 - x_1 - x_2$$
$$y_3 = (x_1 - x_3)\lambda - y_1$$

  - Doubling $(P = Q)$

$$\lambda = \frac{3x_1^{\,2} + a}{2y_1}$$
$$x_3 = \lambda^2 - 2x_1$$
$$y_3 = (x_1 - x_3)\lambda - y_1$$



Example: Addition

$$E : y^2 = x^3 - 25x$$

$$P = (x_1, y_1) = (0,0),\ Q = (x_2, y_2) = (-5,0),\ P + Q = (x_3, y_3)$$

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1} = \frac{0 - 0}{-5 - 0} = 0$$

$$x_3 = \lambda^2 - x_1 - x_2 = 0^2 - 0 - (-5) = 5$$

$$y_3 = (x_1 - x_3)\lambda - y_1 = (0 - 5) \times 0 - 0 = 0$$

Example: Doubling

$$E : y^2 = x^3 - 25x$$

$$P = (x_1, y_1) = (-4,6),\ 2P = (x_2, y_2)$$

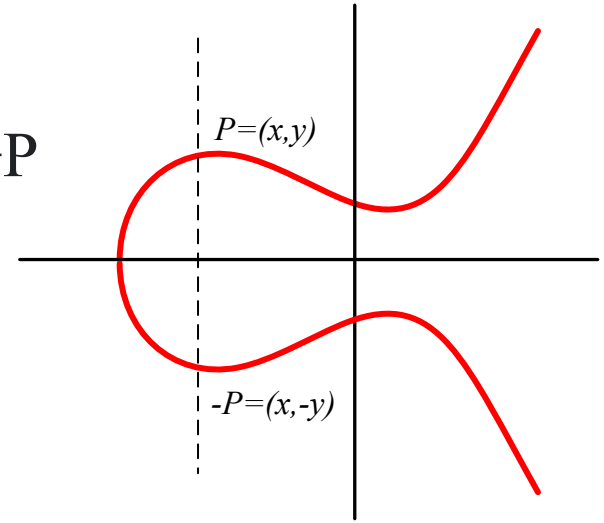
$$\lambda = \frac{3{x_1}^2 + a}{2y_1} = \frac{3(-4)^2 - 25}{2 \times 6} = \frac{23}{12}$$

$$x_2 = \lambda^2 - 2x_1 = \left(\frac{23}{12}\right)^2 - 2 \times (-4) = \frac{1681}{144}$$

$$y_2 = (x_1 - x_2)\lambda - y_1 = \left(-4 - \frac{1681}{144}\right) \times \frac{23}{12} - 6 = -\frac{62279}{1728}$$

Group Operation + Over E

- (E, +) is an abelian group
- Closure: Given P, Q ∈ E, P+Q ∈ E
- Associative Law: Given P, Q, R ∈ E, (P+Q)+E=P+(Q+E)
- Identity: O
- Inverse: -P for P ∈ E
- Commutative Law: Given P, Q ∈ E, P+Q=Q+P



Topic 3 Elliptic Curve Cryptography (ECC)

Elliptic Curves Modulo a Prime

Instructor: Dr. Jianbing Ni

Fall 2021

Elliptic Curves Modulo a Prime

Let $p > 3$ be prime. An *elliptic curve* E over Z_p is the set of points $(x, y) \in Z_p \times Z_p$ to the congruence

$$y^2 \equiv x^3 + ax + b \mod p,$$

where $a, b \in Z_p$ are constants such that $4a^3 + 27b^2 \not\equiv 0 \mod p$, together with a special point $\mathcal{O}$ called the *point at infinity*.

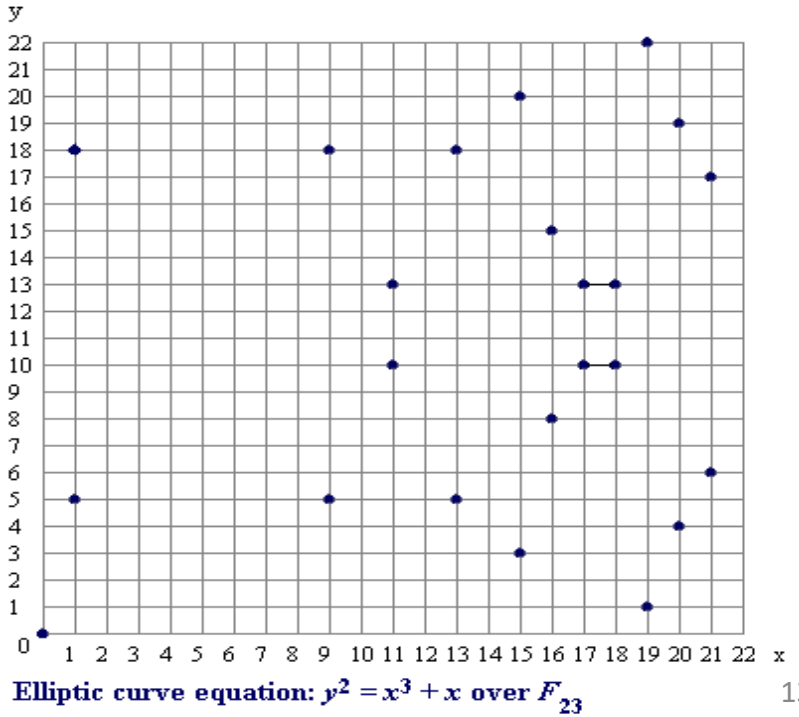
Elliptic Curves Modulo a Prime

Let $p > 3, a, b \in Z_p,\ 4a^3 + 27b^2 \neq 0 \pmod{p}$

$$E = \left\{ (x, y) \in Z_p \times Z_p \,\middle|\, y^2 \equiv x^3 + ax + b \pmod{p} \right\} \cup \{ O \}$$

Example:

$$E : y^2 = x^3 + x \text{ over } Z_{23}$$



Elliptic curve equation: $y^2 = x^3 + x$ over F_{23}

Example

$E : y^2 = x^3 + x + 6$ over Z_{11}

Find all *(x, y)* and O:

Fix *x* and determine *y*

O is point at infinity

12 *(x, y)* pairs plus O,

and have *#E=13*

$y^2 = 5 \bmod 11$

$= 5$

$\equiv 4^2 \bmod 11$

$\equiv 7^2 \bmod 11$

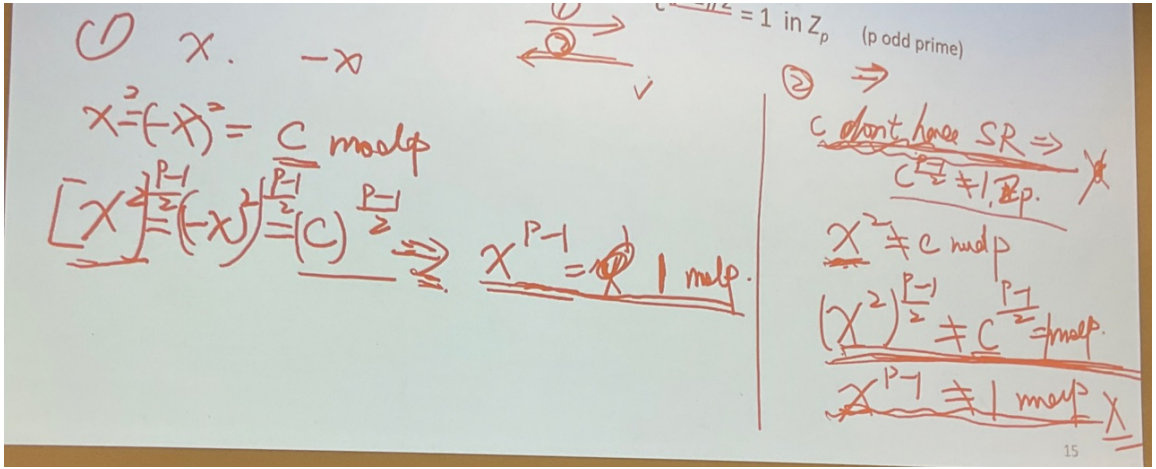
x	$x^3 + x + 6$	quad res?	y
0	6	*no*	
1	8	*no*	
2	5	*yes*	4,7
3	3	*yes*	5,6
4	8	*no*	
5	4	*yes*	2,9
6	8	*no*	
7	4	*yes*	2,9
8	9	*yes*	3,8
9	7	*no*	
10	4	*yes*	2,9

Quadratic Residue

$x^2 \equiv c \bmod p \Rightarrow c$ is Q.R

Def: c in Z_p is a **quadratic residue** (Q.R.) modulo p if it has a square root in Z_p (p is odd prime).

Euler's theorem: c in $(Z_p)^*$ is a Q.R. $\Leftrightarrow$ $c^{(p-1)/2} = 1$ in Z_p (p odd prime)



a.p互质

Base on little Theorem ($a^{p-1} \equiv 1 \bmod p$)

Assume $p = 2q+1$

$\therefore a^{2q} \equiv 1 \bmod p$

$(a^{q-1})(a^{q+1}) \equiv 0 \bmod p$

$\therefore a^q \equiv \pm 1 \bmod p$

$\therefore a^{\frac{p-1}{2}} \equiv \pm 1 \bmod p$

Quadratic Residue

Lemma: When $p = 3 \mod 4$, if $c \in Z_p^*$ is Q.R. then $\sqrt{c} = c^{(p+1)/4}$ in Z_p

When $p = 1 \pmod 4$, can also be done efficiently, but a bit harder

run time $\approx O(\log^3 p)$

$$(\sqrt{c})^2 = c^{\frac{p+1}{2}} = c^{\frac{p-1+2}{2}} = c^{\frac{p-1}{2}+1} = c^{\frac{p-1}{2}} \cdot c$$

Legendre 符号:

$$\left(\frac{a}{p}\right) = \begin{cases} 0 & a \equiv 0 \mod p \\ 1 & a \not\equiv 0 \mod p, \exists x \text{ s.t. } x^2 \equiv a \pmod p \\ -1 & \nexists x \text{ s.t. } x^2 \equiv a \pmod p \end{cases} \Rightarrow$$

Euler 准则:

对 $\forall a \in \mathbb{Z}$, $p \nmid a$:

$$\left(\frac{a}{p}\right) \equiv a^{\frac{p-1}{2}} \pmod p$$

(E, +) over Z_p is a cyclic abelian group

(E,+) over Z_p is an abelian group.

- Closure: Given P, Q ∈ E, P+Q ∈ E
- Associative Law: Given P, Q, R ∈ E, (P+Q)+E=P+(Q+E)
- Identity: O
- Inverse: -P for P ∈ E
- Commutative Law: Given P, Q ∈ E, P+Q=Q+P

$$E : y^2 = x^3 + x \text{ over } Z_{23}$$

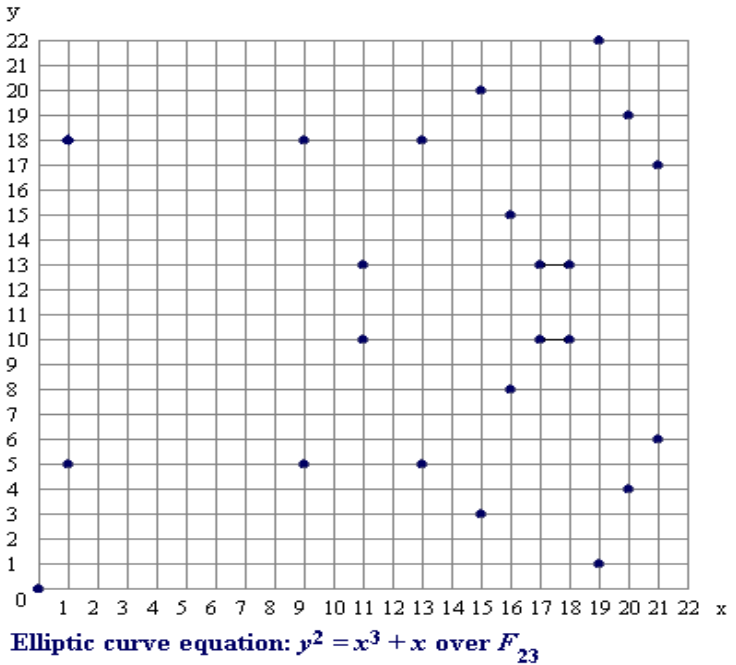
P=(1,5), Q=(11,10)
P+Q= (17, 10)

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1}$$

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

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

ex. $\frac{10-5}{11-1} = \frac{5}{10} = 5 \cdot 10^{-1}$



Elliptic curve equation: $y^2 = x^3 + x$ over F_{23}

(E, +) over Z_p is a cyclic abelian group

(E, +) is a cyclic group.

$E : y^2 = x^3 + x + 6 \text{ over } Z_{11}$

There are 13 points on the group $E(Z_{11})$ and so any non-identity point (i.e. not the point at infinity O) is a generator of $E(Z_{11})$.

Choose a generator

Compute $\alpha = (2,7)$

$$2\alpha = (x_2, y_2)$$

$$\lambda = \frac{3x_1^2 + a}{2y_1} = \frac{3(2)^2 + 1}{2 \times 7} = \frac{13}{14} = 2 \times 3^{-1} = 2 \times 4 = 8 \mod 11$$

$$x_2 = \lambda^2 - 2x_1 = (8)^2 - 2 \times (2) = 5 \mod 11$$

$$y_2 = (x_1 - x_2)\lambda - y_1 = (2 - 5) \times 8 - 7 = 2 \mod 11$$

Example (Cont.)

Compute $3\alpha = (x_3, y_3)$

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1} = \frac{2-7}{5-2} = 2 \mod 11$$

$$x_3 = \lambda^2 - x_1 - x_2 = 2^2 - 2 - 5 = 8 \mod 11$$

$$y_3 = (x_1 - x_3)\lambda - y_1 = (2-8)\times 2 - 7 = 3 \mod 11$$

So, we can compute

$$\alpha = (2,7) \quad 2\alpha = (5,2) \quad 3\alpha = (8,3)$$

$$4\alpha = (10,2) \quad 5\alpha = (3,6) \quad 6\alpha = (7,9)$$

$$7\alpha = (7,2) \quad 8\alpha = (3,5) \quad 9\alpha = (10,9)$$

$$10\alpha = (8,8) \quad 11\alpha = (5,9) \quad 12\alpha = (2,4)$$

Topic 3 Elliptic Curve Cryptography (ECC)

ElGamal on Elliptic Curve

Instructor: Dr. Jianbing Ni

Fall 2021

Elliptic Curve ElGamal

Suppose Elliptic Curve $E(Z_p)$ with an order q. α is a generator of $E(Z_p)$.

KeyGen: Bob's private key is $a \in Z_q$. Bob's public key is $\beta=a\alpha$.

Encryption: To encrypt a message x (which is a point on $E(Z_p)$), Alice randomly selects $k \in Z_q$, and uses Bob's public key β to compute:

$$C=E(\beta, x)=(y_1,y_2)=(k\alpha, x+k\beta).$$

Decryption: To decrypt C, Bob uses the private key a to compute:

$$x=D(a, C)=y_2-ay_1.$$

Elliptic Curve ElGamal

Let's modify ElGamal encryption by using the elliptic curve $E(Z_{11})$. The order of $E(Z_{11})$ is 13.

Suppose that $\alpha = (2,7)$ and Bob's private key is a=7, so $\beta = 7\alpha = (7,2)$

Suppose that Alice wishes to encrypt the plaintext x = (10,9).

If she chooses the random value k = 3, then

$$y_1 = 3(2,7) = (8,3) \text{ and}$$

$$y_2 = (10,9) + 3(7,2) = (10,9) + (3,5) = (10,2)$$

Hence $y = \big((8,3),(10,2)\big)$. Now, if Bob receives the ciphertext y, he decrypts:

$$x = (10,2) - 7(8,3) = (10,2) - (3,5)$$

$$= (10,2) + (3,6) = (10,9)$$

Elliptic Curve DLP

Basic computation of ECC

$$Q = kP = \underbrace{P + P + \ldots + P}_{k \text{ times}}$$

where P is a curve point, k is an integer

Elliptic Curve Discrete Logarithm Problem (ECDLP)

- Given curve, the point P, and kP, it is hard to recover k

We say that **ECDLog is hard in E over Z_p** if for all efficient alg. A:

$$\Pr_{P \leftarrow E,\ k \leftarrow Z_p}[A(E, p, P, kP) = k] < \text{negligible}$$

Security of ECC vs. RSA/ElGamal

- Elliptic curve cryptosystems give the most security per bit of any known public-key schemes.
- The ECDLP problem appears to be much more difficult than the integer factorisation problem and the discrete logarithm problem of Z_p^*.
- The strength of elliptic curve cryptosystems grows much faster with the key size increases than does the strength of RSA.

Security of ECC vs. RSA/ElGamal

Queen's UNIVERSITY

Symmetric Key Size (bits)	RSA and Diffie-Hellman Key Size (bits)	Elliptic Curve Key Size (bits)
80	1024	160
112	2048	224
128	3072	256
192	7680	384
256	15360	521

NIST Recommended Key Sizes

Summary

- Elliptic Curve over R
- Elliptic Curve Modulo a Prime
- Elliptic Curve ElGamal
- Security of ECC vs. RSA/ElGamal

Practice Question

1. On the elliptic curve over the real numbers $y^2 = x^3 - 36x$, let P=(-3.5, 9.5) and Q=(-2.5, 8.5). Find $P + Q$ and $2P$.

2. Consider the elliptic curve over Z_{23}: $y^2=x^3+x+1$ mod 23. Please list all the non-negative points in the quadrant from (0, 0) through (22, 22) that satisfy the equation mod 23.

Practice Question of C6T2

Consider an ElGamal scheme with a common prime q=71 and a generator g=7.

a. If B has public key Y_B=3 and A chose the random integer k=2, what is the ciphertecxt of M=30?

b. If A now chooses a different value of k, so that the encoding of M=30 is C=(59, C_2), what is the integer C_2?

E(pk=(g, pk_A), m) :
b ← Z_q, u ← g^b,
v ← m·$pk_A{}^b$,
output (u, v)

D(sk=a, (u, v)) :
m ← v·u^{-a}
output m

a. $C_1 = g^k \bmod q = 7^2 \bmod 71 = 49$

$C_2 = M \cdot Y_B^k \bmod q = 30 \times 3^2 \bmod 71 = 57$

a. $u = g^b = 7^2 \bmod 71 = 49$

$V = m \cdot pk_A{}^b = 30 \cdot 3^2 \bmod 71 = 57$

b. $59 = g^k \bmod 71$. $k = 3$.

b. $C_2 = m \cdot pk_A{}^b = 30 \cdot 3^b \bmod 71$

$C_1 = g^b = 7^b \bmod 71 = 59$

$\Rightarrow k = 3$

$C_2 = 30 \cdot 3^3 \bmod 71 = 289$

$C_2 = M Y_B^k \bmod q = 30 \cdot 3^3 \bmod 71 = 29$.

Thanks

Let's work out an algebraic formula to compute R. First, the equation of $\mathcal{L}$ is $y = \lambda x + \nu$, where the slope of $\mathcal{L}$ is

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1},$$

and

$$\nu = y_1 - \lambda x_1 = y_2 - \lambda x_2.$$

In order to find the points in $\mathcal{E} \cap \mathcal{L}$, we substitute $y = \lambda x + \nu$ into the equation for $\mathcal{E}$, obtaining the following:

$$(\lambda x + \nu)^2 = x^3 + ax + b,$$

which is the same as

$$x^3 - \lambda^2 x^2 + (a - 2\lambda\nu)x + b - \nu^2 = 0. \tag{7.9}$$

The roots of equation (7.9) are the x-co-ordinates of the points in $\mathcal{E} \cap \mathcal{L}$. We already know two points in $\mathcal{E} \cap \mathcal{L}$, namely, P and Q. Hence x_1 and x_2 are two roots of equation (7.9).

Since equation (7.9) is a cubic equation over the reals having two real roots, the third root, say x_3, must also be real. The sum of the three roots must be the negative of the coefficient of the quadratic term, or λ^2. Therefore

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

x_3 is the x-co-ordinate of the point R'. We will denote the y-co-ordinate of R' by

$-y_3$, so the y-co-ordinate of R will be y_3. An easy way to compute y_3 is to use the fact that the slope of $\mathcal{L}$, namely λ, is determined by any two points on $\mathcal{L}$. If we use the points (x_1, y_1) and $(x_3, -y_3)$ to compute this slope, we get

$$\lambda = \frac{-y_3 - y_1}{x_3 - x_1},$$

or

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

Therefore we have derived a formula for $P + Q$ in case 1: if $x_1 \neq x_2$, then $(x_1, y_1) + (x_2, y_2) = (x_3, y_3)$, where

$$\begin{aligned} x_3 &= \lambda^2 - x_1 - x_2, \\ y_3 &= \lambda(x_1 - x_3) - y_1, \quad \text{and} \\ \lambda &= \frac{y_2 - y_1}{x_2 - x_1}. \end{aligned}$$