Discrete logarithm problem (DLP) & ECDSA

Many slides are from Rong-Jaye Chen@NCTU

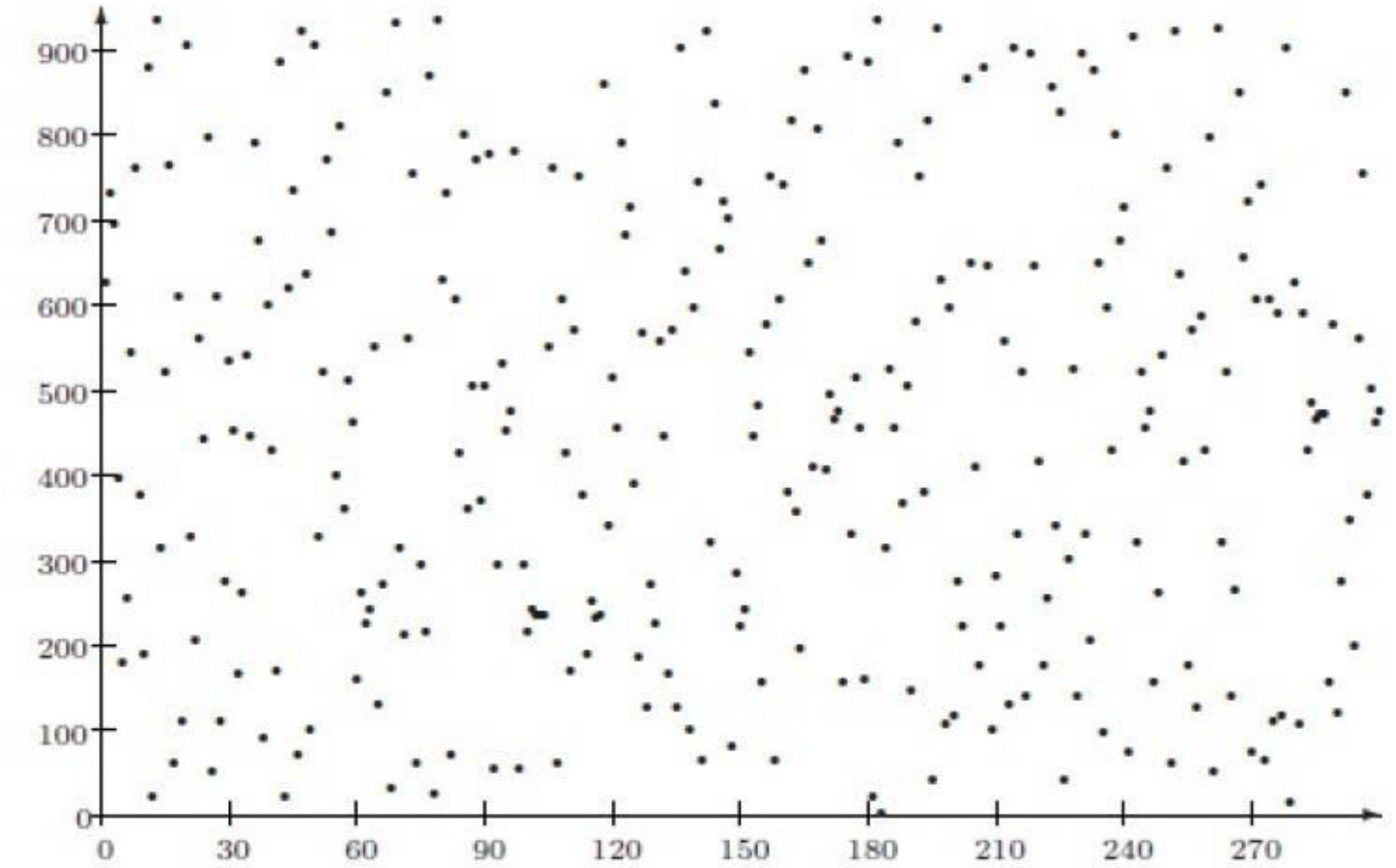
Discrete Logarithm

- Fix a prime p. Let a, b be nonzero integers (mod p)
- The problem of finding x such that $a^x \equiv b \pmod{p}$ is called the discrete logarithm problem (DLP)
- We denote $x=L_a(b)$, and call it the discrete log of b w.r.t. a (mod p)
- Ex: $p=11$, $a=2$, $b=9$, then $x=L_2(9)=6$

In some references, $x=\mathrm{Log}_a(b)$

Discrete logarithm: plotting

A graph of $f(x) = 627^x \bmod 941$ for $x = 1, 2, 3, \ldots$



Source: Kaafarani@Oxford Univ.

Discrete Logarithms

- In the Diffie-Hellman and ElGamal methods, the difficulty of solving the discrete logarithm problem yields good cryptosystems
- Given p, a, b, solve $a^x \equiv b \pmod{p}$
- If $\{a^x : 0 \leq x \leq p-2\} = \{1,2,3,\ldots\ p-1\}$, a is called a primitive root mod p
  - a is aka a generator

What a generator is doing

- The group of positive integers modulo prime p
  $Z_p^* \equiv \{1, 2, 3, \ldots, p-1\}$
- By Fermat's little theorem: $a^{(p-1)} = 1 \pmod{p}$
- Example of Z_7^*

x	x^2	x^3	x^4	x^5	x^6
1	1	1	1	1	1
2	4	1	2	4	1
3	**2**	**6**	**4**	**5**	**1**
4	2	1	4	2	1
5	**4**	**6**	**2**	**3**	**1**
6	1	6	1	6	1

Generators

For all p, the group is cyclic. A cyclic group is a group whose elements are generated by a single element (the generator)

Discrete Logarithms

- Discrete log problem
  - Given $Z_p^* = \langle\alpha\rangle$
  - $\text{Log}_\alpha(y) = x$, if $y = \alpha^x$.

Set of elements which are generated by the exponentiation of α

- Example
  - $Z_{13}^* = \langle 2\rangle$; $2^1=2$, $2^2=4$, $2^3=8$, $2^4=3$, $2^5=6$, $2^6=12$, $2^7=11$, $2^8=9$, $2^9=5$, $2^{10}=10$, $2^{11}=7$, $2^{12}=1$
  - $\text{Log}_2(5) = 9$.

Algorithms that solves DLP

- Some are of sub-exponential complexity

That’s why we need ECC

- Elliptic Curve Cryptography (ECC)
  - Similar to DLP
    - Called ECDLP
  - No efficient algorithms yet

Bitcoin uses ECDSA

- Elliptic Curve Digital Signature Algorithm
- curve used is `secp256k1`
- set of points $(x,y) \in \{F_p \times F_p \mid y^2 = x^3 + 7 \pmod p\}$
- $p = 2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1$
- Forms a group E, $|E| = q \approx p \approx 2^{256}$

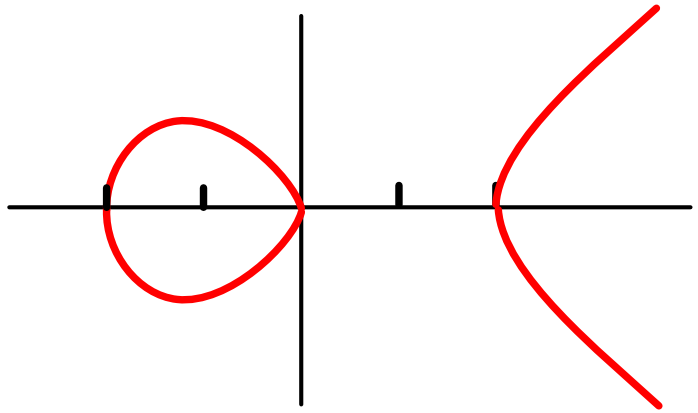
	range	format	size (bits)
sk	Z_q	random	256
pk	E	$sk \cdot G$	512/257*
m	Z_q	H(message)	256
sig	$Z_q \times Z_q$	(r, s)	512

Elliptic curves over Real (R)

- Definition $a, b \in \boldsymbol{R},\ 4a^3 + 27b^2 \neq 0$

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

- Example: $E : y^2 = x^3 - 4x$



"Adding" two points in an elliptic curve

- Group operation +

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 $(P = Q)$

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

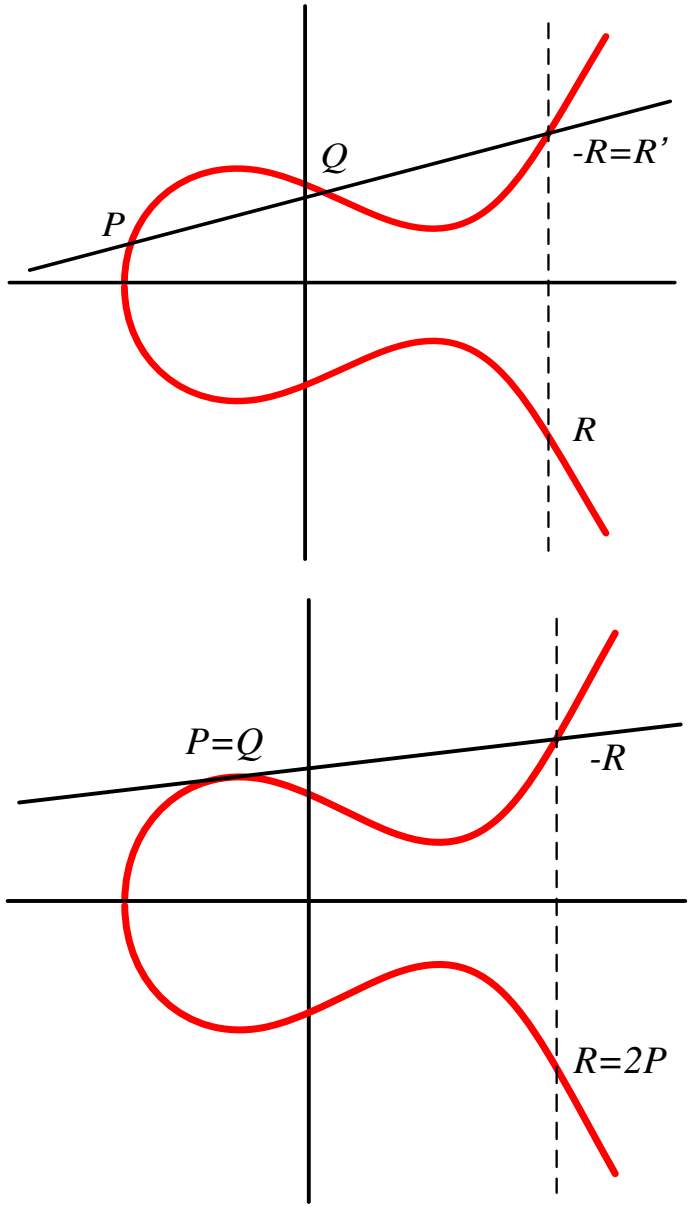
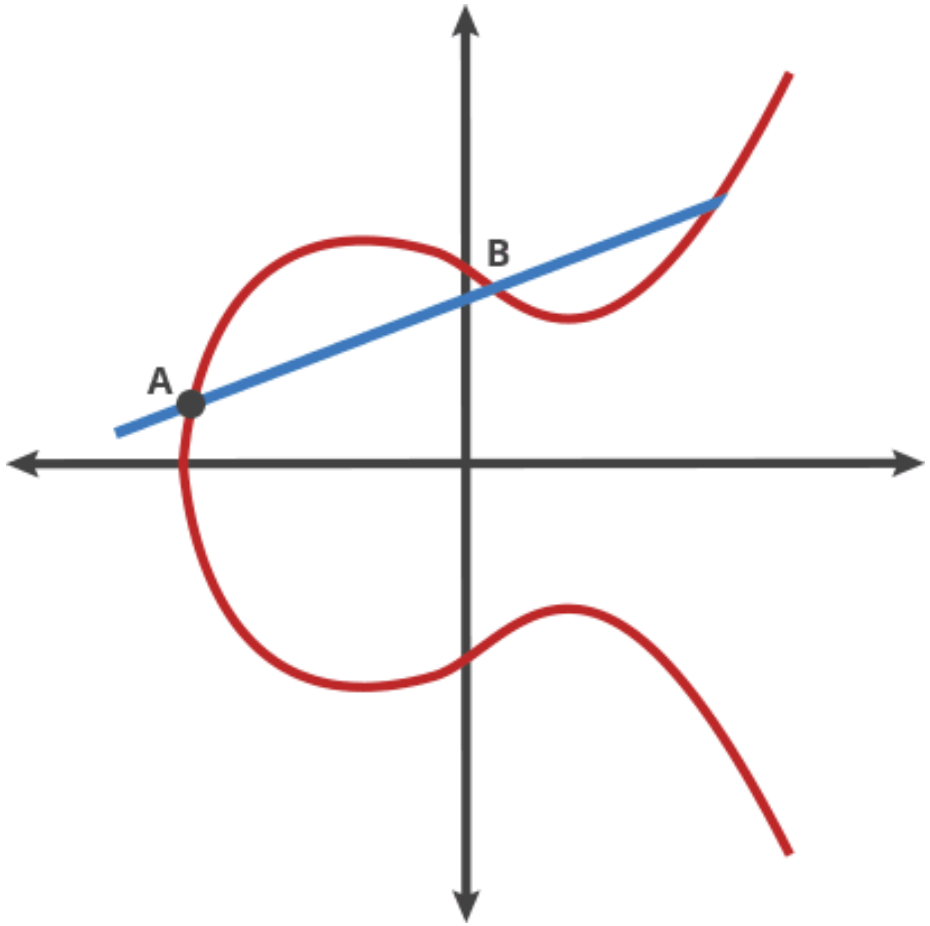


Illustration of Addition in an elliptic curve



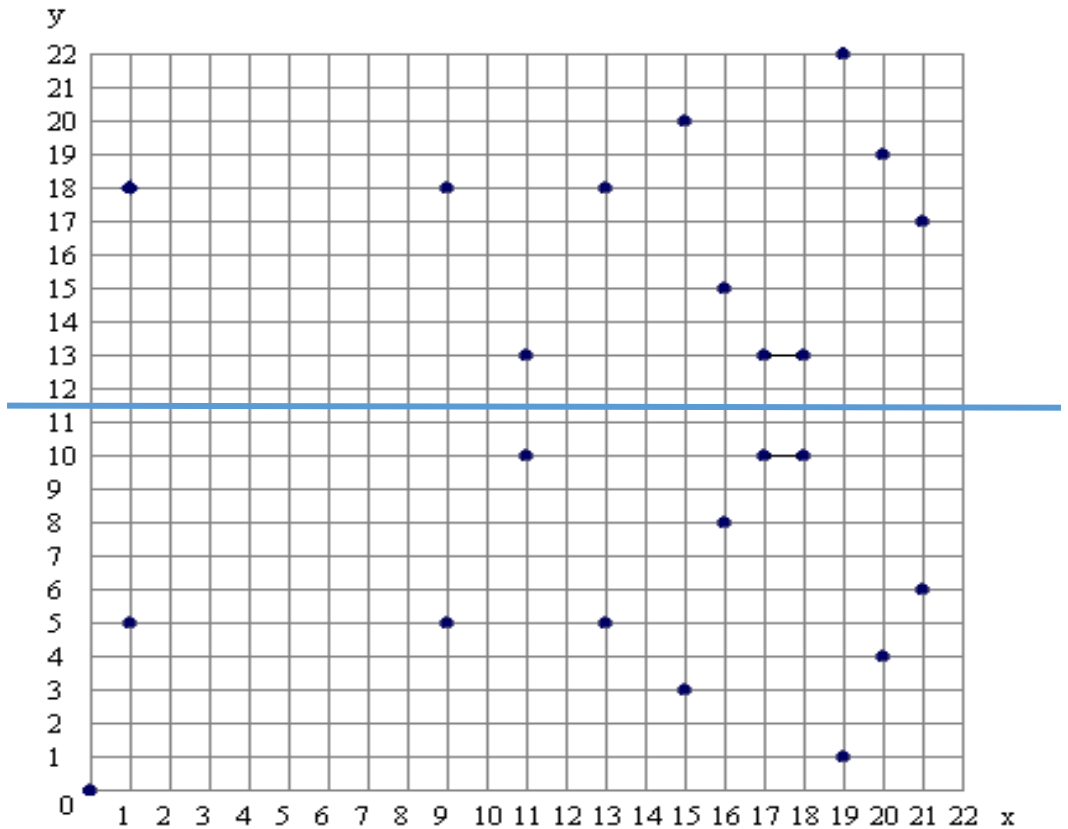
Elliptic Curves over GF(p)

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

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

- Example:

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



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

An illustration of an ECDLP

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

Find all (x, y) and O:

- Fix *x* and determine *y*
- O is an artificial point

12 (x, y) pairs plus O,
and have *#E=13*

Cardinality, q in ECDSA table

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

Integer y is called a quadratic residue modulo n if there exists x s.t. $x^2 = y \bmod n$

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

  Choose generator $\alpha = (2,7)$

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

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

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

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

- Example (continue):

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

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

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

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

along this line, we can compute

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

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

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

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

We will first see ElGamal DLP
Then ECC version of ElGamal, which is ECDLP

Setting up ElGamal

- Let p be a large prime
  - By "large" we mean here a prime of thousands bits
- Select a special number g
  - The number g must be a primitive element modulo p.
- Choose a private key x
  - This can be any number bigger than 1 and smaller than p-1
- Compute public key y (from x, p and g)
  - The public key y is g raised to the power of the private key x modulo p. In other words:

$$y = g^x \bmod p$$

- Publicize p, g, y

ElGamal encryption

The first job is to represent the plaintext M as a series of numbers modulo p. Then:

1. Generate a random number k (ephemeral key)
   - Only sender knows k
2. Compute two values C_1 and C_2, where

   $$\mathbf{C_1 = g^k \bmod p} \quad \text{and} \quad \mathbf{C_2 = My^k \bmod p}$$

3. Send the ciphertext C, which consists of the two separate values C_1 and C_2.

ElGamal decryption

$$C_1 = g^k \bmod p \quad C_2 = My^k \bmod p$$

1 - The receiver begins by using their private key **x** to transform C_1 into something more useful:

$$C_1^x = (g^k)^x \bmod p$$

NOTE: $$C_1^x = (g^k)^x = (g^x)^k = (y)^k = y^k \bmod p$$

2 - This is a very useful quantity because if you divide C_2 by it, you get **M**. In other words:

$$C_2 / C_1^x = C_2 / y^k = (My^k) / y^k = M \bmod p$$

Come back to ECDLP

x is the message (x_1,x_2) and k is the ephemeral key

- Example (continue):
  Let's modify ElGamal encryption by using the elliptic curve $E(Z_{11})$.
  Suppose that generator $\alpha = (2,7)$ and Bob's private key is 7, so

$$\beta = 7\alpha = (7,2)$$

Thus the encryption operation is

$$e_K(x,k) = \big(k(2,7), x + k(7,2)\big),$$

where $x \in E$ and $0 \le k \le 12$, and the decryption operation is

$$d_K(y_1, y_2) = y_2 - 7y_1. = x + k(7,2) - 7k(2,7)$$
$$= x + k(7,2) - k(7,2)$$

- Example (continue):
  Suppose that Alice wishes to encrypt the plaintext x = (10,9) (x ∈ E). 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 it as follows:

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

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

Calculate public key from private key: elliptic curve cryptography (ECC)

- Example:

  Compute 3895P

$$3895P = \underbrace{P + P + \cdots + P}_{\text{3894 additionsneeded}}$$

$$= (111100110111)_2 P$$

$$= 2(2(2(2(2(2(2(2(2(2(2P + P) + P) + P))) + P) + P)) + P) + P) + P$$

$$\rightarrow 11\,\text{doublings and}\ 8\,\text{additions needed}$$

$$= (1000(-1)0100(-1)00(-1))_2 P$$

$$= 2(2(2(2(2(2(2(2(2(2(2P))) - P)) + P))) - P))) - P$$

$$\rightarrow 12\,\text{doublings and}\ 4\,(\text{additions or subtractio ns})\ \text{needed}$$

Elliptic Curve Discrete Logarithm Problem (ECDLP)

- Basic computation of ECC
  - $Q = kP = \underbrace{P + P + \ldots + P}_{k\ times}$
    where P is a curve point, k is an integer

- Strength of ECC
  - Given curve, the point P, and kP
    It is hard to recover k
    - Elliptic Curve Discrete Logarithm Problem (ECDLP)

Security of ECC versus RSA/ElGamal

- Elliptic curve cryptosystems give the most security per bit of any known public-key scheme
- The ECDLP problem appears to be much more difficult than the integer factorization problem and the discrete logarithm problem of Z_p (no index calculus algo!)
- The strength of elliptic curve cryptosystems grows much faster with the key size increases than does the strength of RSA

Security level of Elliptic Curve Cryptography (ECC)

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

Before going into DSA & ECDSA,
we need to know digital signature & (sub)group

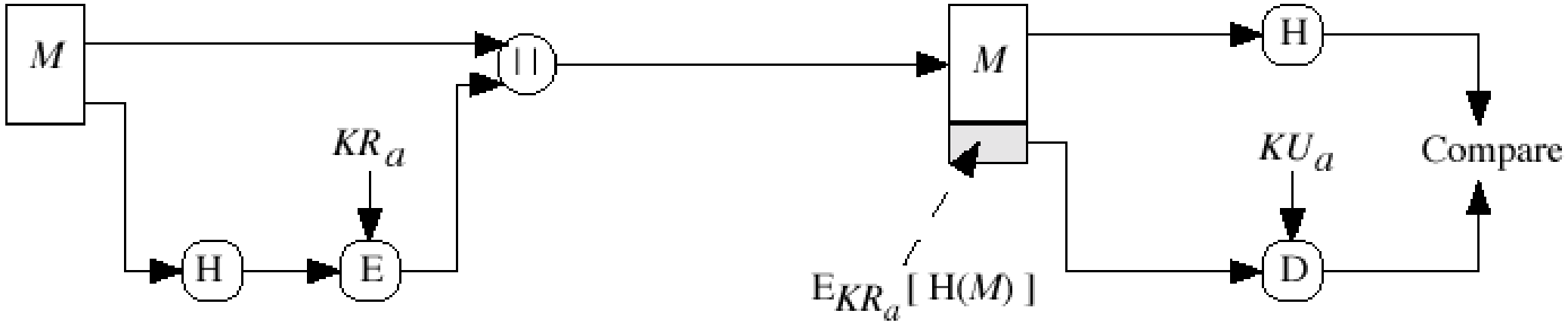
Digital Signatures

- data integrity, non-repudiation, authentication
- Basic idea
  - use private key on the message to generate a piece of information that can be generated only by yourself
    - because you are the only person who knows your private key
  - public key can be used to verify the signature
    - so everybody can verify
- Generally signatures are created and verified over the hash of the message
  - Not over the original message. Why?

Digital Signature

Sender Alice

Receiver



M: message to be signed

H: Hash function

E: RSA Private Key Operation

KR_a: Sender's Private Key

D: RSA Public Key Operation

KU_a: Sender's Public Key

$E_{KR\_a}[H(M)]$ Signature of sender Alice over hash of M

Background for finite cyclic groups

- Order
  - order of a set S: |S|, # of elements in S
  - Order of an element x: ord(x), the least $n \geq 1$ s.t. $x^n \equiv 1 \pmod{p}$
- Cyclic group G
  - A group that can be generated by exponentiating a generator g
    - G = <g>
  - Group order |G| = Z_p^* = n, s.t. $g^n \equiv 1 \pmod{p}$
- Subgroup (≠ subset) of a cyclic group G
  - Every subgroup of G is cyclic, and satisfies all the properties of group
  - if order of G is |G|, each subgroup S has the form $<g^d>$, where d is a positive divisor of |G|
    - each subgroup has size dividing the size of the group
    - |S|*k=|G|, $k \in N$

A cyclic group (from modular exponentiation)

- mod 7 case, $Z_7^* = \{1,2,3,4,5,6\}$

$1^1 \equiv 1$	$1^2 \equiv 1$	$1^3 \equiv 1$	$1^4 \equiv 1$	$1^5 \equiv 1$	$1^6 \equiv 1$
$2^1 \equiv 2$	$2^2 \equiv 4$	$2^3 \equiv 1$	$2^4 \equiv 2$	$2^5 \equiv 4$	$2^6 \equiv 1$
$3^1 \equiv 3$	$3^2 \equiv 2$	$3^3 \equiv 6$	$3^4 \equiv 4$	$3^5 \equiv 5$	$3^6 \equiv 1$
$4^1 \equiv 4$	$4^2 \equiv 2$	$4^3 \equiv 1$	$4^4 \equiv 4$	$4^5 \equiv 2$	$4^6 \equiv 1$
$5^1 \equiv 5$	$5^2 \equiv 4$	$5^3 \equiv 6$	$5^4 \equiv 2$	$5^5 \equiv 3$	$5^6 \equiv 1$
$6^1 \equiv 6$	$6^2 \equiv 1$	$6^3 \equiv 6$	$6^4 \equiv 1$	$6^5 \equiv 6$	$6^6 \equiv 1$

Look at rows this time!

Group Z_7^* or G has two generators: G = <3> = <5>
|G| = 6, its unique divisors are {1,2,3,6},
for each unique divisor d, $<g^d>$ generates a subgroup
$< 3^1 >$ = {1,2,3,4,5,6} = G
$< 3^2 >$ = <2> = {1,2,4}
$< 3^3 >$ = <6> = {1,6}
$< 3^6 >$ = <1> = {1}

Zoom in to a subgroup

- $\langle 3 \rangle = \langle 2 \rangle = \{1,2,4\} = S$
- 2 is the generator g of a subgroup S; S is cyclic
- the order of g is the size of the subgroup $\langle g \rangle = S$
  - $2^{|S|} \equiv 1 \pmod{p}$
- $|S| = 3 = q$
- For $\forall x \in S$, $x^q \equiv 1 \pmod{p}$
  - Then x is the q-th root of 1 (mod p)
  - How many roots for $x^q = 1 \pmod{p}$?
    - q
  - S consists of the q-th roots of 1 (mod p)

Digital Signature Algorithm (DSA)

$L=0 \bmod 64,\ 512 \le L \le 1024$

- Let p be a L-bit prime such that the DLP in Z_p^* is intractable
- Let q be a 160-bit prime that divides p-1.
- Let α be a q_{th} root of 1 modulo p

$$p = qr+1$$
$$h^r = \alpha$$

Define $K=\{ (p,q,\alpha,a,\beta): \beta=\alpha^a \bmod p \}$

p,q,α,β are the public key, a is private key

- For a (secret) random number k, define
  $sig_K(x,k)=(\gamma,\delta)$, where
  $\gamma=(\alpha^k \bmod p) \bmod q$ and
  $\delta=(H(x)+a\gamma)k^{-1} \bmod q$

x is the message to be signed

- For a message $(x,(\gamma,\delta))$, verification is done by performing the following computations:

$$e_1=H(x)*\delta^{-1} \bmod q$$
$$e_2=\gamma*\delta^{-1} \bmod q$$

verify$(x,(\gamma,\delta))$=true iff $(\alpha^{e_1}\beta^{e_2} \bmod p) \bmod q=\gamma$

Elliptic Curve DSA (ECDSA)

- Let p be a prime, and let E be an elliptic curve defined over F_p.

- Let A be a point on E having prime order q, such that DL problem in <A> is infeasible.

  Define K={ (p,q,E,A,m,B): B=mA }

  p,q,E,A,B are the public key, m is private

- For a (secret) random number k, define $sig_K(x,k)=(r,s)$, where $kA=(u,v)$, $r=u \bmod q$ and $s=k^{-1}(H(x)+mr) \bmod q$

- For a message {x,(r,s)}, verification is done by performing the following computations:

  $i=H(x)*s^{-1} \bmod q$

  $j=r*s^{-1} \bmod q$

  $(u,v)=iA+jB$

  verify(x,(r,s))=true iff $u \bmod q=r$