Applications of Elliptic Curves in Cryptography

William King

What do these have in common?

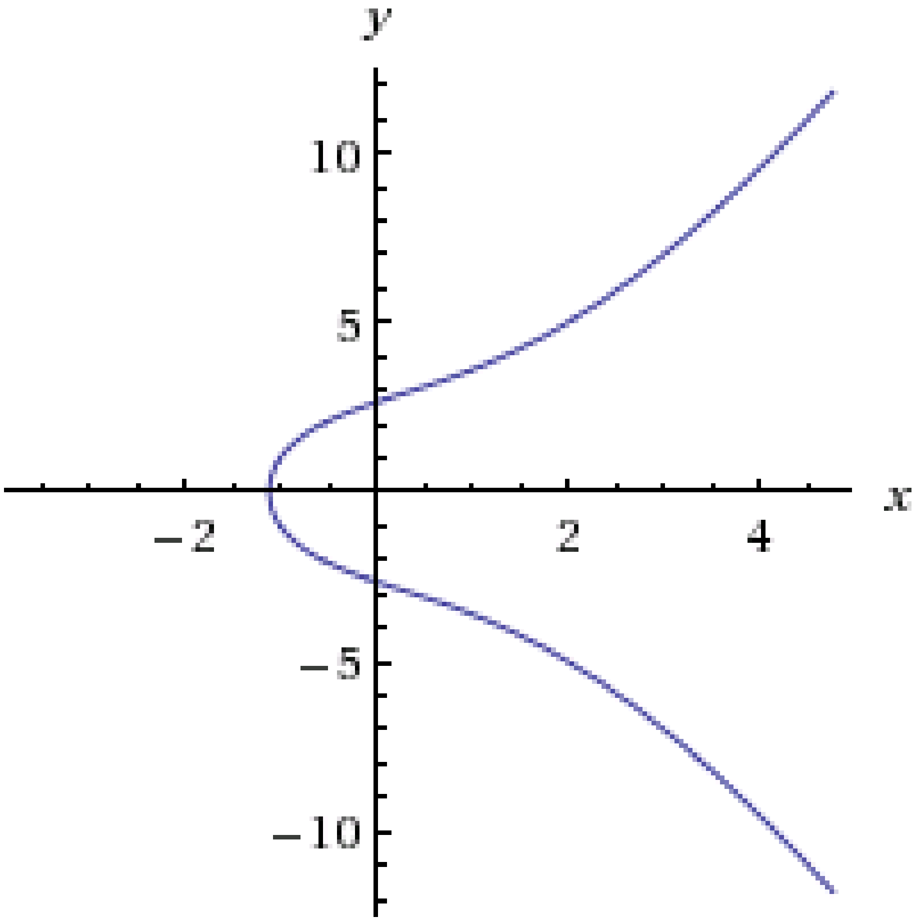
Gmail™
by Google™

PS3
PlayStation 3

Wii™

BlackBerry

What Are Elliptic Curves?



$y^2=x^3+5x+7$

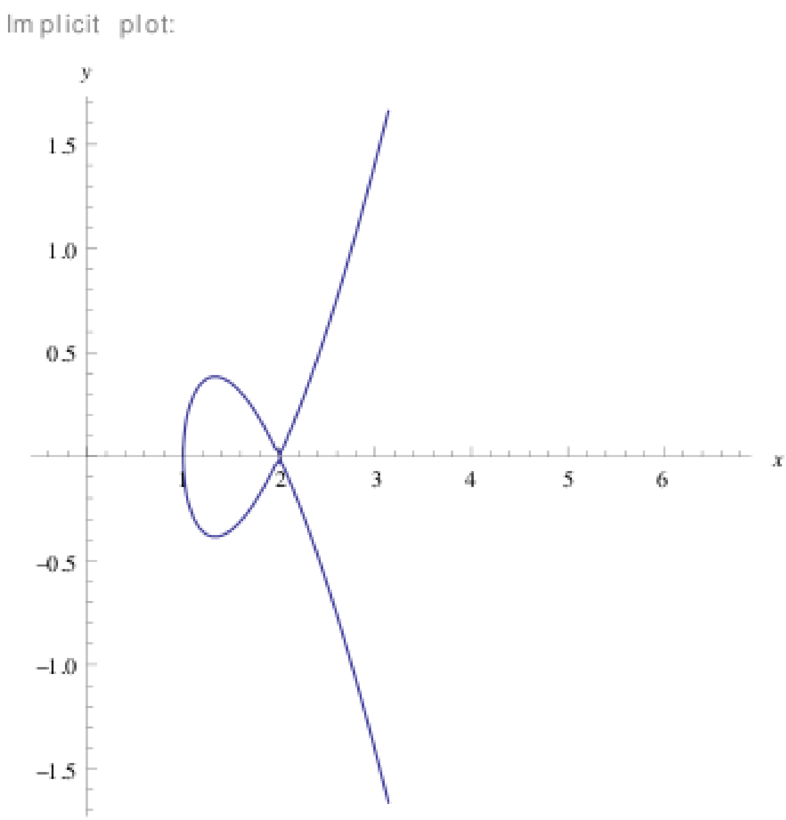
Equations of the form:

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

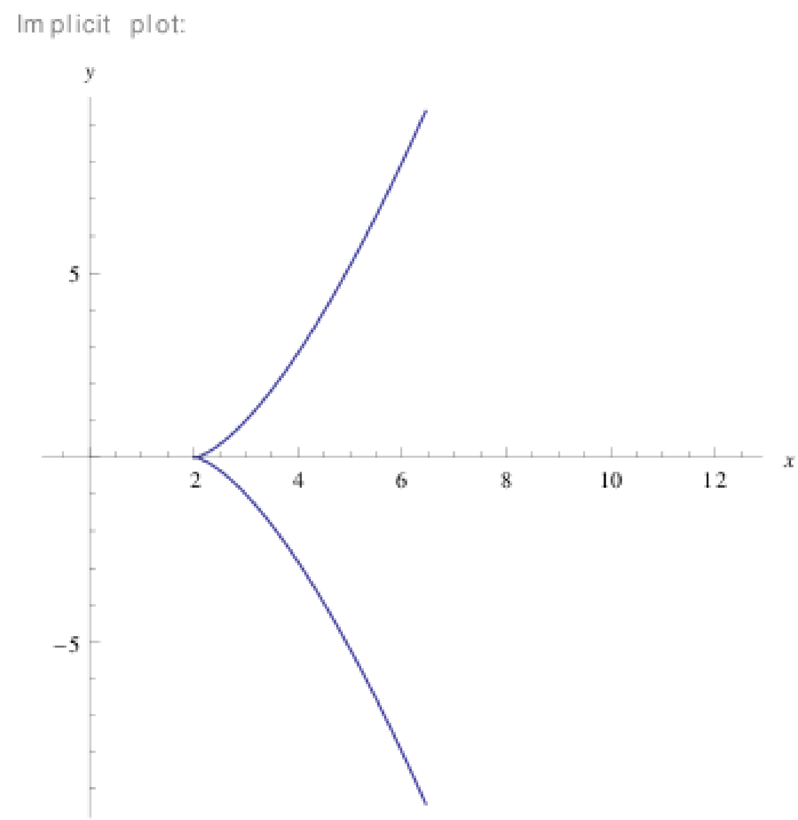
such that:

$4a^3+27b^2\neq 0$

$4a^3+27b^2\neq 0$

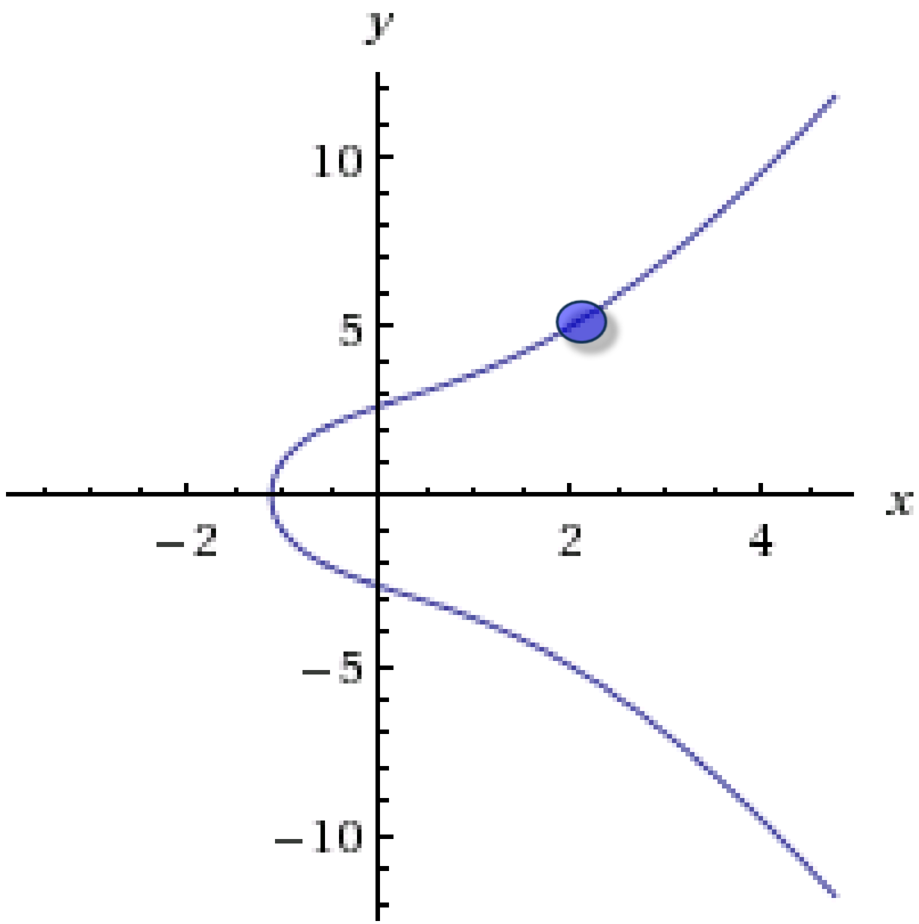


$y^2=(x-2)^2(x-1)$



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

Points on Elliptic Curves



$y^2=x^3+5x+7$

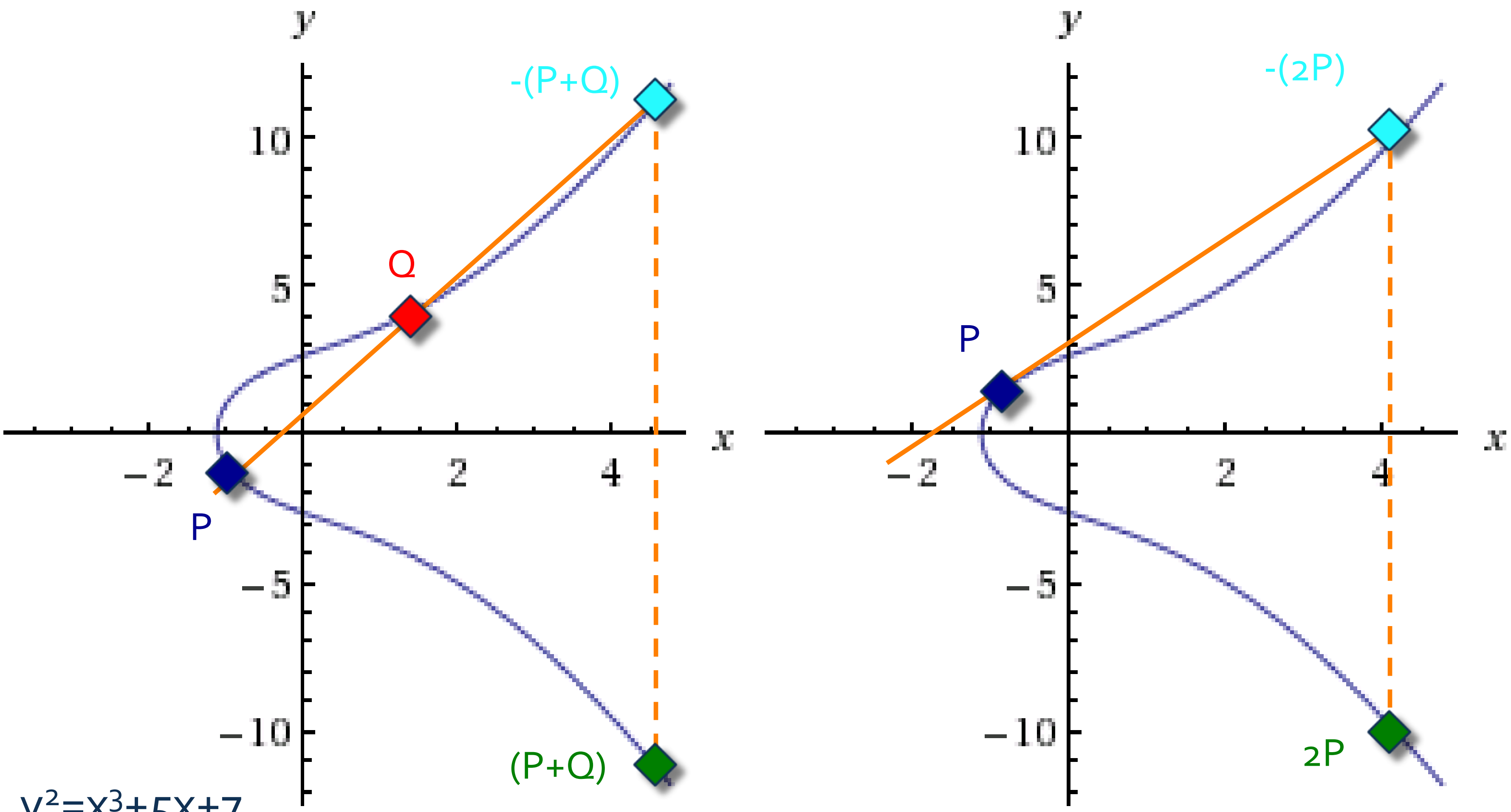
The set of all (x,y) such that:

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

For example: (2,5)

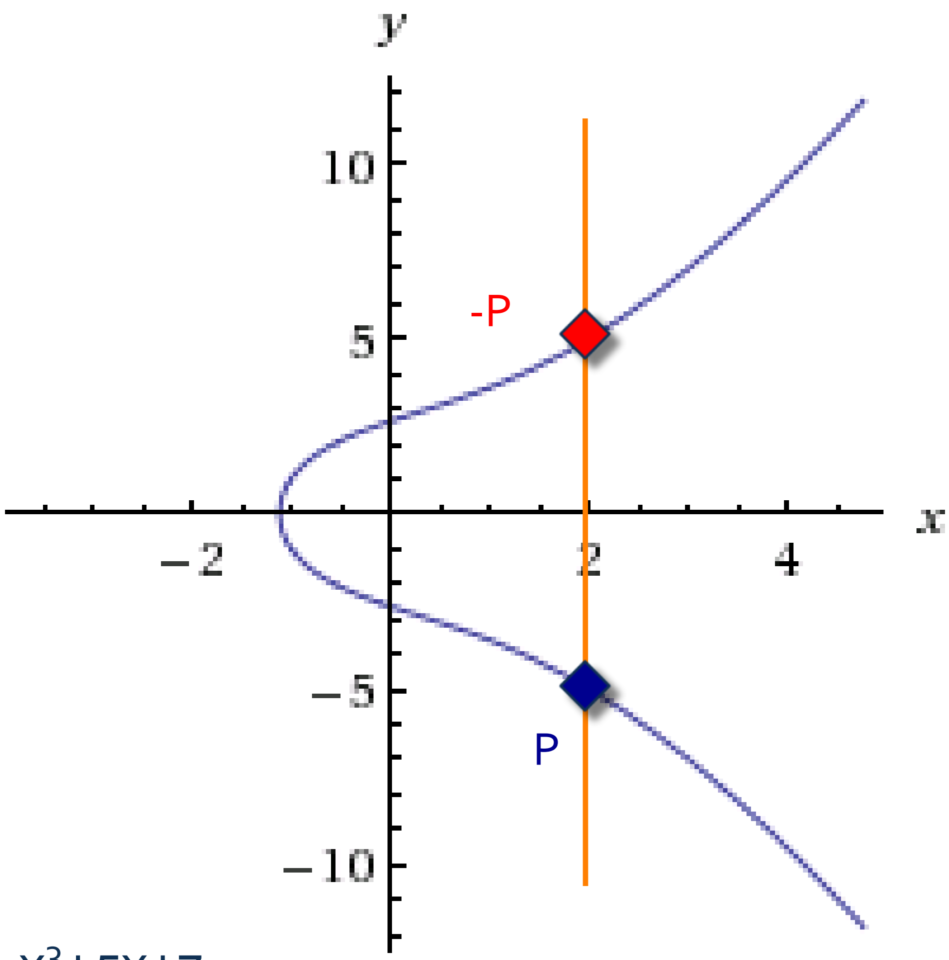
$5^2=2^3+5(2)+7$

Adding Points of Elliptic Curves!



$y^2=x^3+5x+7$

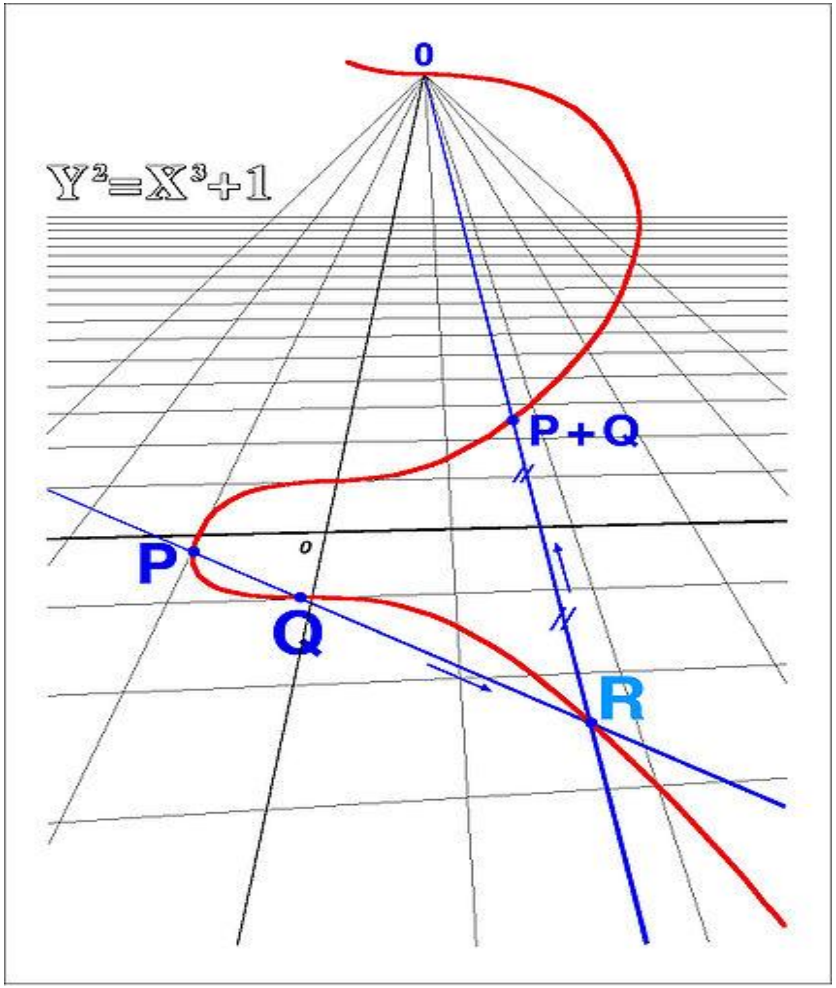
Point Addition (Continued)



$y^2=x^3+5x+7$

Where does the line intersect the curve?

The Point at Infinity



$P+(-P) = \infty$

We define ∞, the point at infinity, as the point where vertical lines meet.

We include the point at infinity with elliptic curves to achieve algebraic closure.

Point Addition: Algebraic Interpretation

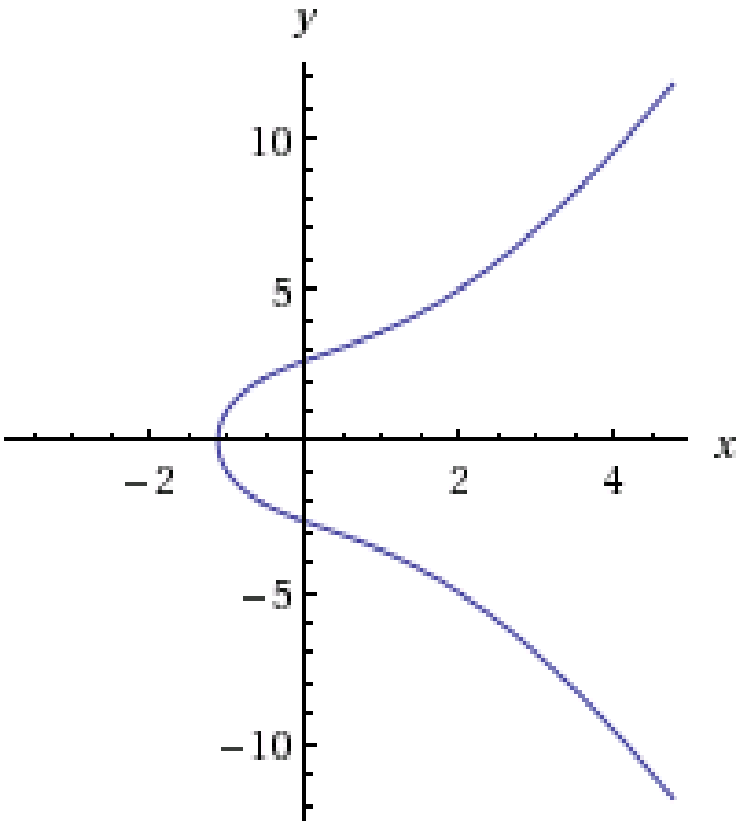
Four Cases:

1. For distinct points $P=(x_1, y_1)$, $Q=(x_2, y_2)$, such that Q is not the elliptic inverse of P, then $P+Q=(r, s)$ such that
   - $r = ((y_2 - y_1)(x_2 - x_1)^{-1})^2 - x_1 - x_2$
   - $s = ((y_2 - y_1)(x_2 - x_1)^{-1})(x_1 - r) - y_1$

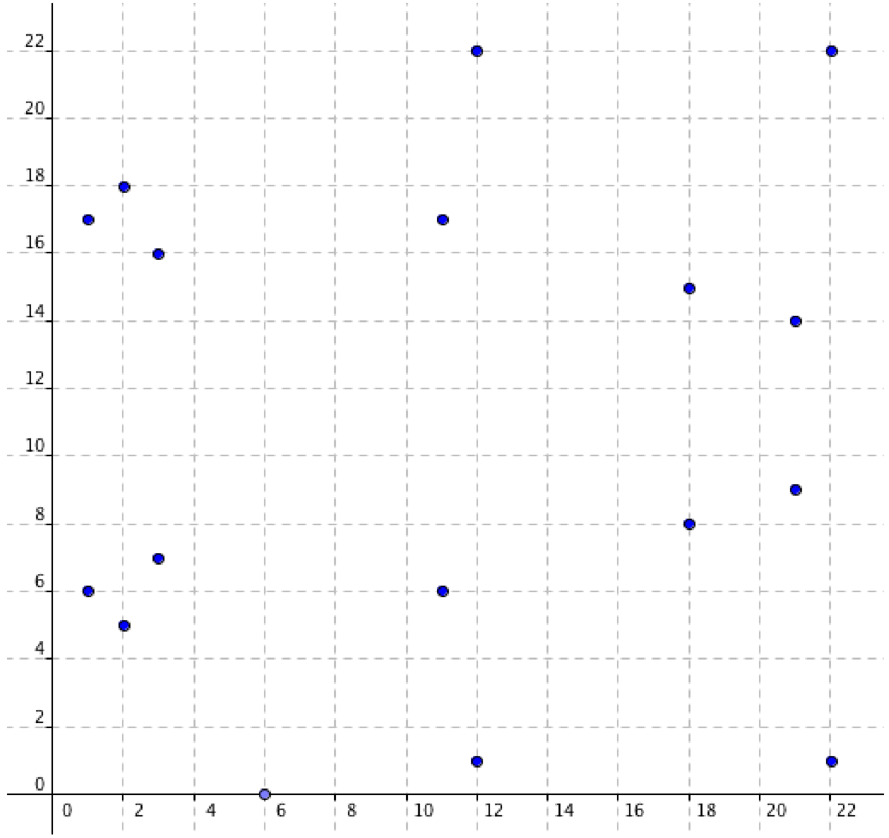
Point Addition: Algebraic Interpretation (Continued)

2. For a point, $P=(x_1, y_1)$, then $2P = (r, s)$ such that
   - $r = ((3x_1^2 + a)(2y_1)^{-1})^2 - 2x_1$
   - $s = ((3x_1^2 + a)(2y_1)^{-1})(x_1 - r) - y_1$
3. For elliptic inverses P and –P, $P+(-P) = \infty$
   - This relationship also allows us to define
   - $P+\infty = P$
4. For ∞, we define $\infty+\infty=\infty$

Elliptic Curves Over Finite Fields



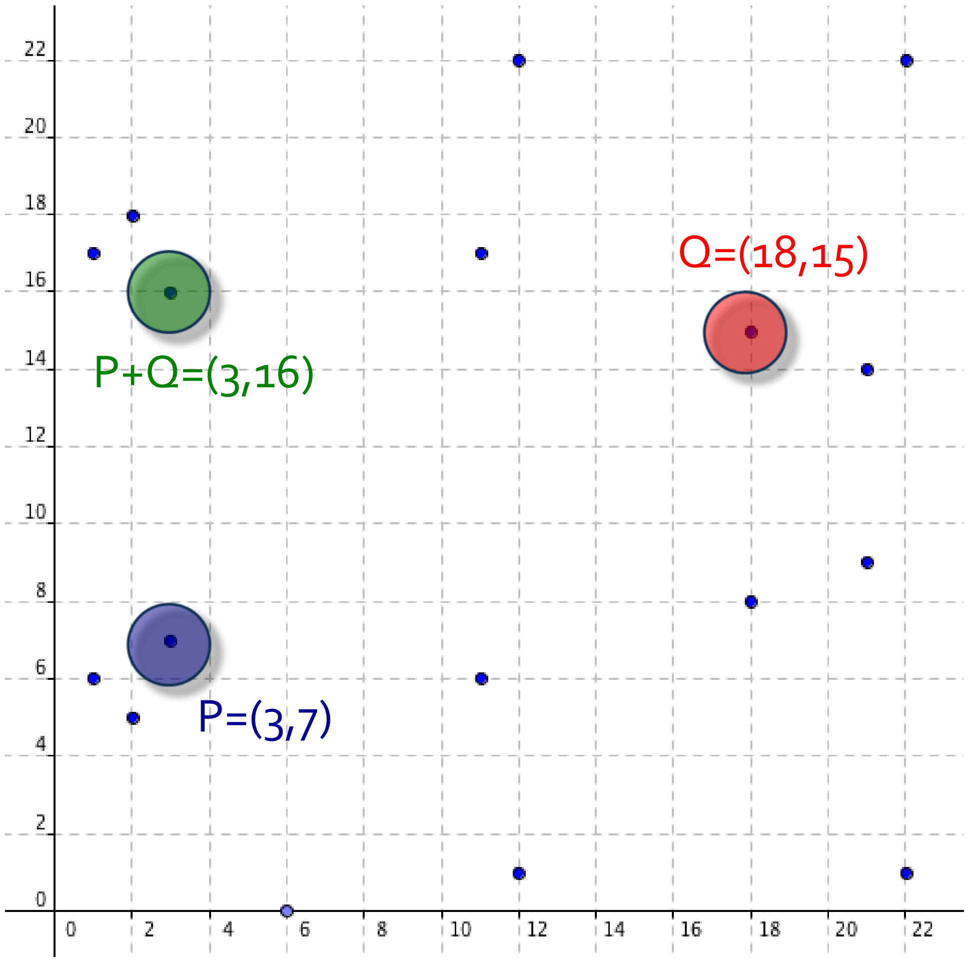
≡



$y^2=x^3+5x+7$

$y^2 \equiv x^3+5x+7 \pmod{23}$

Point Addition on Elliptic Curves over Finite Fields

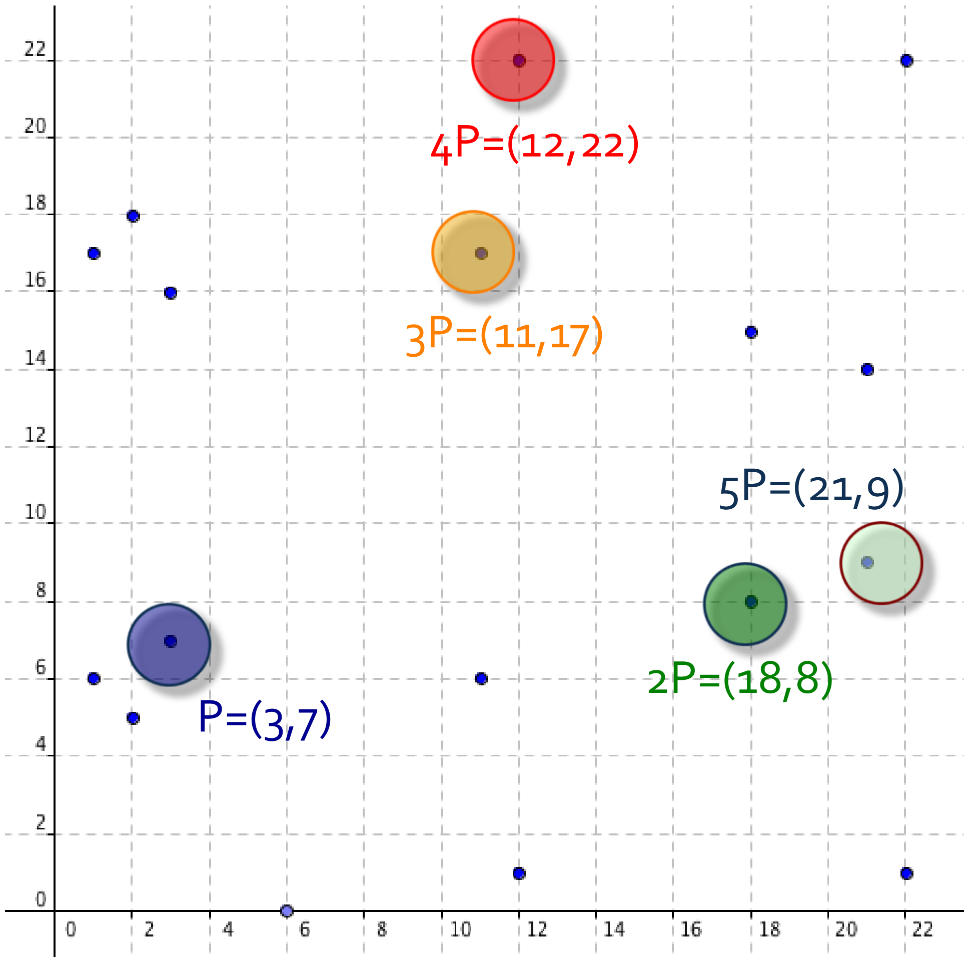


P+Q = (3, 7)+(18, 15) = (r, s)

$r = ((15-7)(18-3)^{-1})^2 - 3 - 18$
$=(8*(15)^{-1})^2 - 21 \pmod{23}$
$=22485 \pmod{23}$
$= 3$

$s = ((15-7)(18-3)^{-1})(3-3) - 7$
$=(8*(15)^{-1})(0) - 7 \pmod{23}$
$=0 - 7 \pmod{23}$
$= 16$

Point Addition on Elliptic Curves over Finite Fields



$2P = (3, 7)+(3, 7)= (r, s)$

$r = ((3(3)^2 +5)(2(7))^{-1})^2 - 2(3)$

$=((3(9)+5)(14)^{-1})^2 - 6 \pmod{23}$

$=((9)(5))^2 + 17 \pmod{23}$

$=501 \pmod{23}$

$=18$

$s =((3(3)^2 + 5)(2(7))^{-1})((3)-18)-7$

$=((3(9)+5)(14)^{-1})(8)+16$

$=(9*5)(8) + 16 \pmod{23}$

$=376 \pmod{23}$

$=8$

The Discrete Logarithm Problem (DLP)

Given:

- a prime integer *p*
- a cyclic group $Z_p = \{0, 1, 2, \ldots, p-1\}$
- a generator *g*, of Z_p
- a non-zero element of Z_p, *a*

This discrete logarithm d, of a to the base g is given by

$$a \equiv g^d \pmod{p}$$

DLP Example

Consider p = 23, then Z_{23} = {0,1,2,…,22}, and note that <11> = Z_{23}

Solve $15 \equiv 11^d \pmod{23}$ for d

Answer: 19

mod(seq(11^x,x,0,22),23) =
{1,11,6,20,13,5,9,7,8,19,2,22,12,17,3,10,18,14,16,15,4,21,1}
{0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22}

Elliptic Curve Discrete Logarithm Problem (ECDLP)

Given:

- an elliptic curve: $y^2 = x^3 + ax + b$
- a prime, *p*
- a field, F_p
- points P,Q on the elliptic curve such that Q is some multiple of P

This discrete logarithm k, of Q to the base P is given by

$$Q = kP$$

ECDLP Example

Consider the elliptic curve $y^2=x^3+9x+17$ over F_{23}

What is the discrete logarithm of Q=(4,5) to the base P=(16,5)? I.e., solve

(4,5) = k*(16,5) for *k*.

Answer: 9

1P=(16,5), 2P=(20,20), 3P=(14,14), 4P=(19,20), 5P=(13,10), 6P=(7,3), 7P=(8,7), 8P=(12,17), 9P=(4,5), …

SoOooOOoOoOoOOoOOOo

Given $Q=kP$ and P, it's difficult to find k; how does this relate to public key cryptography?

Elliptic Curve Cryptography! (ECC)

- Applications:
  - Asymmetric (Public) Key Cryptography
    - Digital Signatures
    - Secure Key Generation

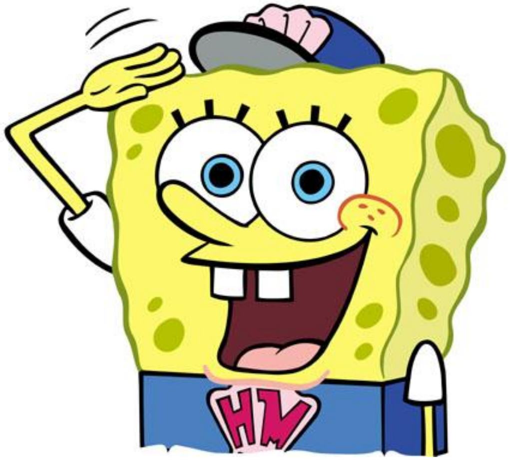
Elliptic Curve Cryptography Broadcast Parameters

$$(p, a, b, G, q)$$

Elliptic Curve Diffie-Hellman Key Exchange (ECDH)

Elliptic Curve Digital Signature Algorithm (ECDSA)

Meet the Players



Alice Bob Eve

Elliptic Curve Diffie-Hellman Key Exchange (ECDH)

Key Agreement Protocol

Elliptic Curve Diffie-Hellman Key Exchange (ECDH)

Alice randomly chooses an integer

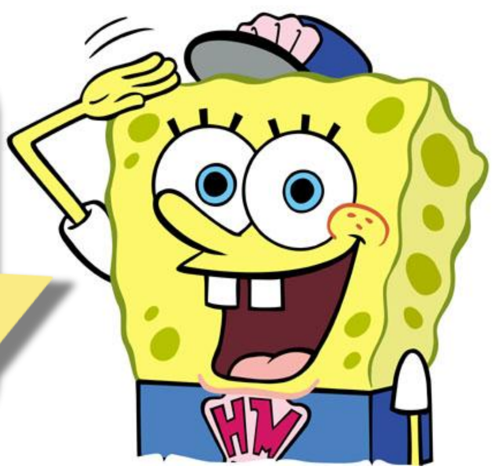
$k_A \in \{1,2,\ldots,q-1\}$

and keeps k_A secret.

Alice

Step 1

Bob



Bob randomly chooses an integer

$k_B \in \{1,2,\ldots,q-1\}$

and keeps k_B secret.

Elliptic Curve Diffie-Hellman Key Exchange (ECDH)

Alice

Alice computes A = $k_A G$

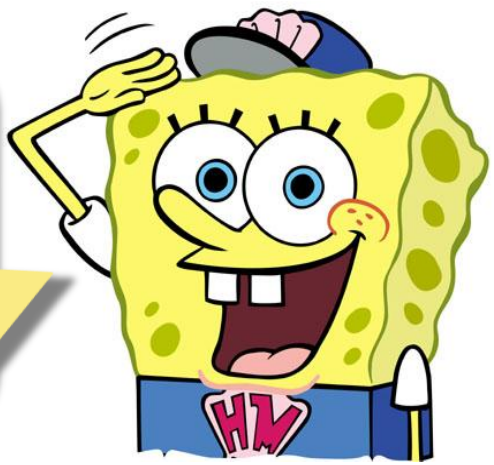
and sends A to Bob.

A

B

Step 2

Bob



Bob computes B = $k_B G$

and sends B to Alice.

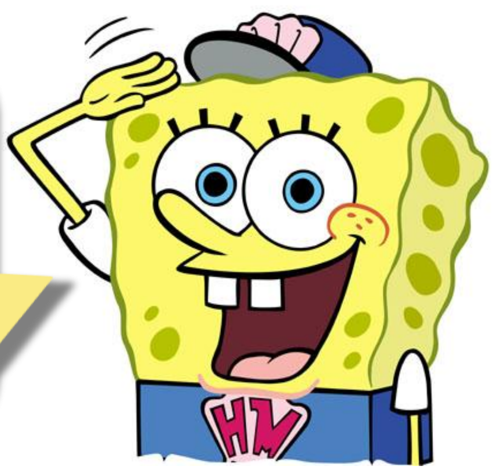
Elliptic Curve Diffie-Hellman Key Exchange (ECDH)

Alice computes $S_A = k_A B$

Alice

Step 3

Bob



Bob computes $S_B = k_B A$

ECDH Proof

Alice and Bob agree upon the same key because

$$S_A = k_A B = k_A(k_B G) = (k_A k_B)G = (k_B k_A)G$$

$$= k_B(k_A G) = k_B A = S_B$$

Elliptic Curve Digital Signature Algorithm (ECDSA)

Digital Signatures

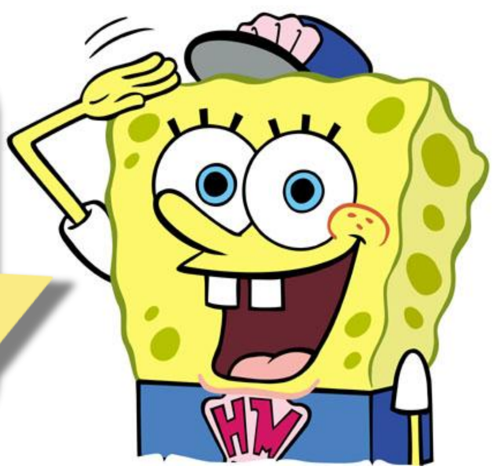
Elliptic Curve Digital Signature Algorithm (ECDSA)

Alice chooses a secret and random integer i, and computes A = iG and publishes A to all

Alice

Step 1

Bob



Bob waits patiently!

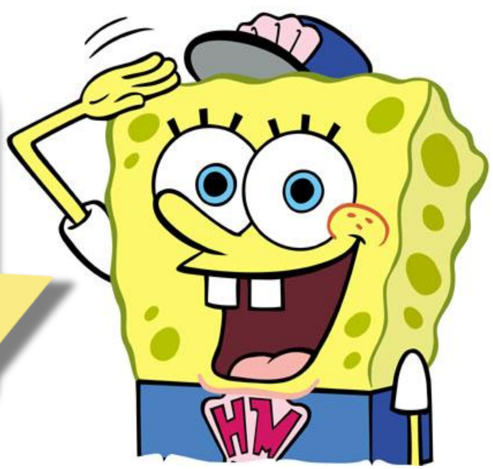
Elliptic Curve Digital Signature Algorithm (ECDSA)

Alice chooses another secret random integer $w \in \{1,2,\ldots,q-1\}$, and computes $Q = wG = (x_Q,y_Q)$

Alice

Step 2

Bob



Bob waits patiently!

Elliptic Curve Digital Signature Algorithm (ECDSA)

Alice

Alice computes h=hash(M) and $s \equiv w^{-1}(h+ix_Q) \pmod{q}$

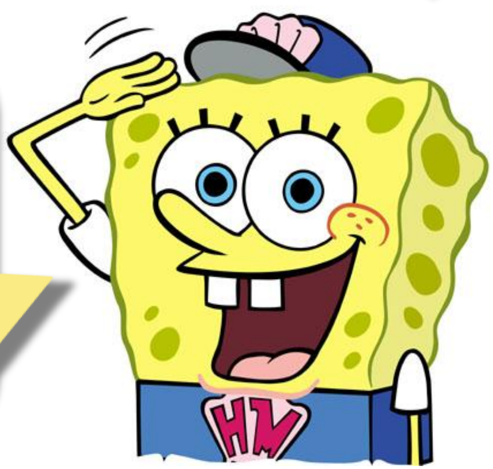
Alice sends (M, Q, s) to Bob

(M, Q, s)

Step 3

Bob

Bob waits patiently!



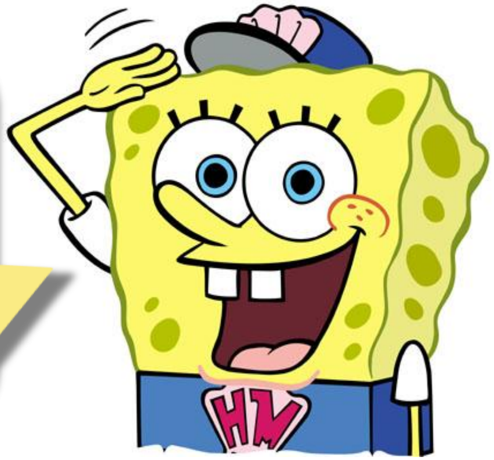
Elliptic Curve Digital Signature Algorithm (ECDSA)

Alice waits patiently!

Alice

Step 4

Bob



Bob computes h=hash(M) and
$z_1 \equiv s^{-1}(h) \pmod{q}$
$z_2 \equiv s^{-1}(x_Q) \pmod{q}$

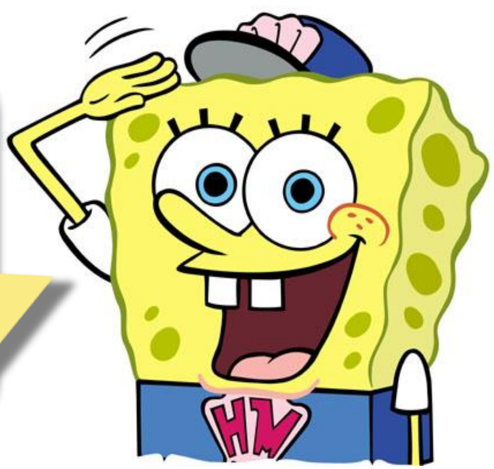
Elliptic Curve Digital Signature Algorithm (ECDSA)

Alice

Alice waits patiently!

Step 5

Bob



Bob computes $B=z_1G+z_2A$

Elliptic Curve Digital Signature Algorithm (ECDSA)

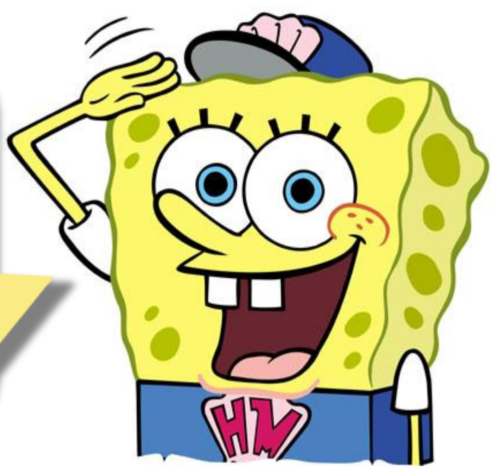
Alice waits patiently!

Alice

Step 6

Bob

If B = Q, then signature is valid, else the signature is invalid



ECDSA Proof

A bit more tricky, but…

Since $s \equiv w^{-1}(h+ix_Q)$

$$w \equiv s^{-1}(h+ix_Q) \equiv s^{-1}h+(s^{-1})ix_Q \equiv z_1+z_2 i \pmod{q}$$

then,

$$B=z_1G+z_2A=z_1G+z_2(iG)=(z_1+z_2i)G=wG=Q$$

Since, the integers *i*,*w* could have only come from Alice, the signature is valid.

Attacks on Elliptic Curve Systems

Solving the Elliptic Curve Discrete Logarithm Problem!

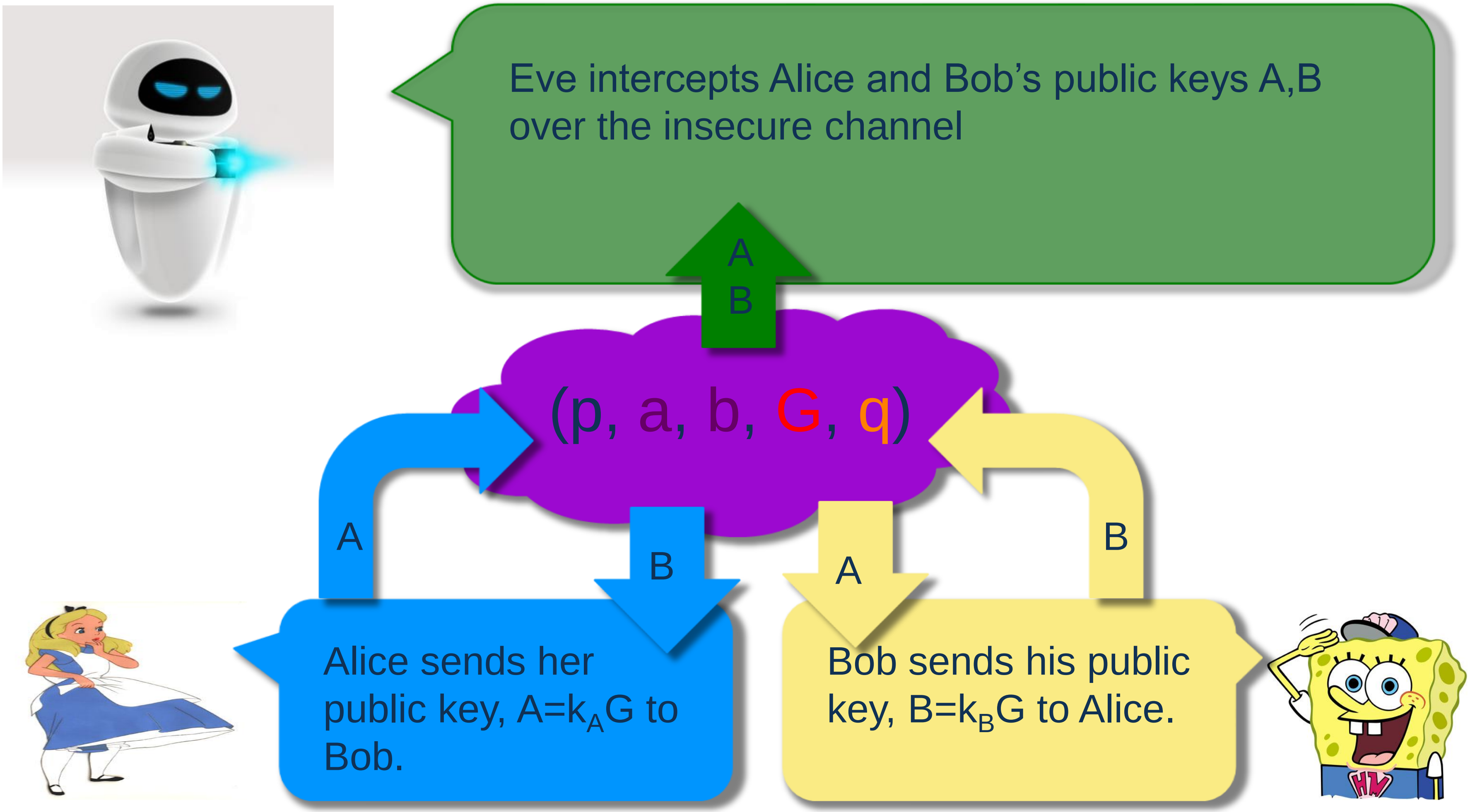
Eve, the Eavesdropper

Baby Step, Giant Step Method

Deterministic

$(q)^{1/2}$ steps & storage

Baby Step, Giant Step Method



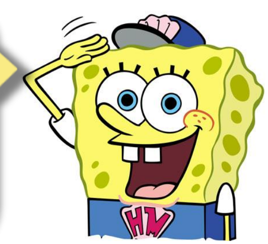
Baby Step, Giant Step Method

Eve chooses an integer $i \geq (q)^{1/2}$ and computes and stores all points jG such that $1 \leq j \leq i$

(p, a, b, G, q)

Alice computes
$S_A = k_A B$

Bob computes
$S_B = k_B A$



Baby Step, Giant Step Method

Eve computes A-(hi)G for consecutive integers h=0,1,2,…,i-1 until A-(hi)G=jG for some integer h and some j from the previous list

(p, a, b, G, q)

Alice and Bob have agreed on a shared key, $S_A=S_B$

Alice and Bob have agreed on a shared key, $S_A=S_B$

Baby Step, Giant Step Method

Eve has recovered Alice's private key, $k_A \equiv j+hi \pmod{q}$

(p, a, b, G, q)

Alice and Bob have agreed on a shared key, $S_A = S_B$

Alice and Bob have agreed on a shared key, $S_A = S_B$

Baby Step, Giant Step Method

Eve computes $S_A = k_A B$ and has arrived at the same shared secret key

(p, a, b, G, q)

Alice and Bob have agreed on a shared key, $S_A = S_B$

Alice and Bob have agreed on a shared key, $S_A = S_B$

Baby Step, Giant Step Method

Why does this work?

When jG=A-(hi)G

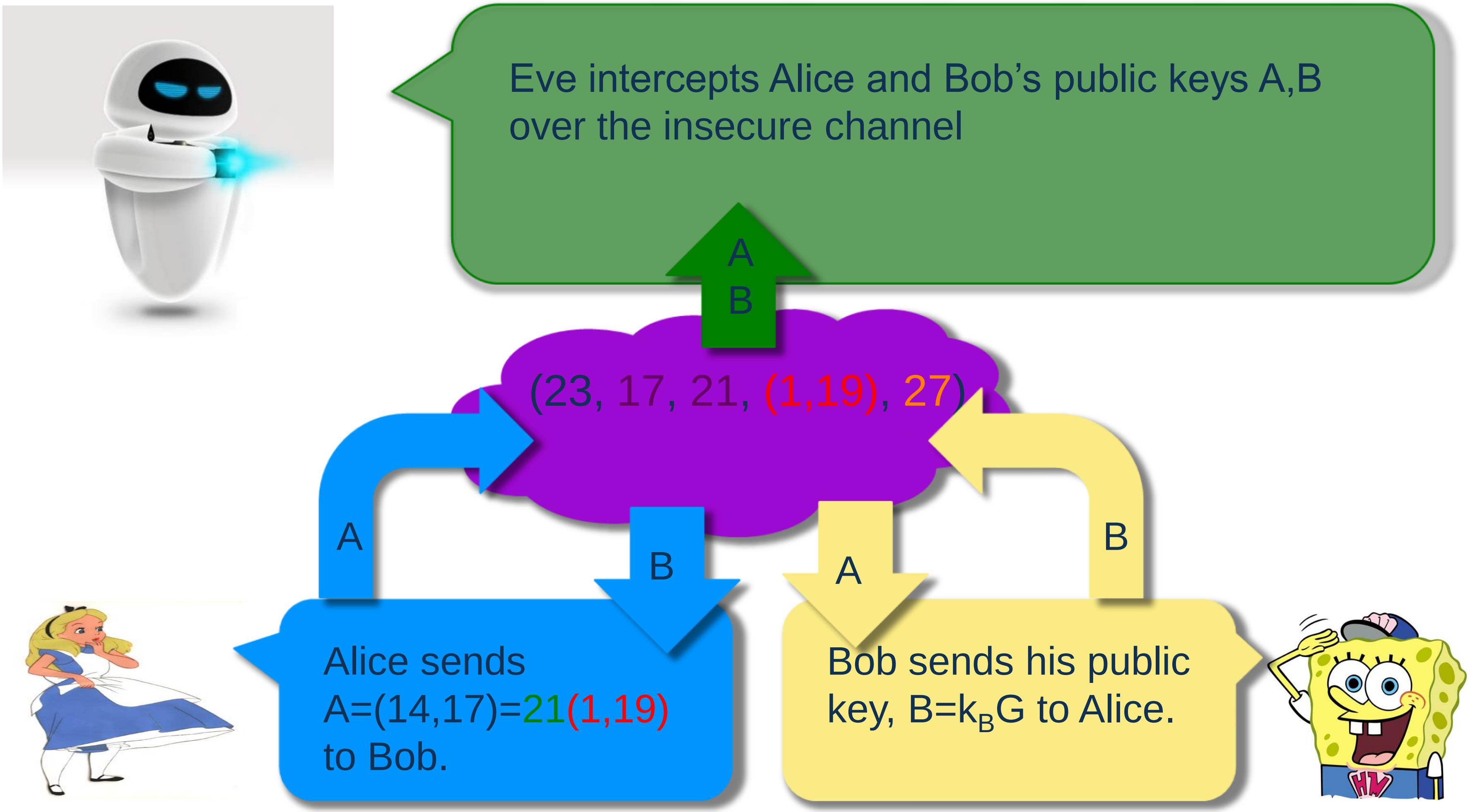
$$jG=A-(hi)G \Rightarrow jG+(hi)G = A-(hiG)+(hi)G$$

$$\Rightarrow (j+hi)G=A+\infty \Rightarrow (j+hi)G=A$$

$$\Rightarrow (j+hi)G = k_A G$$

$$\Rightarrow (j+hi) \equiv k_A$$

Baby Step, Giant Step Method: Example



Baby Step, Giant Step Method

Eve chooses an integer 6 ≥ $(27)^{1/2}$ and computes and stores all points jG such that 1≤ j ≤ 6 in list 1

j	LIST 1	jG
1	1(1,19)	(1,19)
2	2(1,19)	(10,15)
3	3(1,19)	(21,18)
4	4(1,19)	(19,21)
5	5(1,19)	(5,1)
6	6(1,19)	(20,9)

Baby Step, Giant Step Method

Eve computes (14,17)-(h6)(1,19) for consecutive integers h=0,1,2,…,5 Until (14,17)-(h6)G=jG for an integer h, and an integer j from the List 1

j	jG
1	(1,19)
2	(10,15)
3	(21,18)
4	(19,21)
5	(5,1)
6	(20,9)

h	(14,17)-(h6)(1,19)
0	(14,17)
1	(18,8)
2	(17,7)
3	(21,18)

Baby Step, Giant Step Method

Eve has recovered Alice's private key,

$k_A \equiv (3+3*6) \equiv 21 \pmod{27}$

(23, 17, 21, (1,19), 27)

Let's Put Things in Perspective

Windows DRM:

785963102379428822376694789446897396207498568951 ($\approx 7.86 \times 10^{47}$)

8.865×10^{23} steps/storage

NSA Recommends:

Primes larger than $2^{255} \approx 5.79 \times 10^{79}$

ECC Advantages

Security (Bits)	Symmetric encryption algorithm	Minimum Size (Bits) of Public Keys		
		DSA/DH	RSA	ECC
80	Skipjack	1024	1024	160
112	3DES	2048	2048	224
128	AES-128	3072	3072	256
192	AES-192	7680	7680	384
256	AES-256	15360	15360	512

http://www.design-reuse.com/articles/7409/ecc-holds-key-to-next-gen-cryptography.html

Conclusions

"Elliptic Curve Cryptography provides greater security and more efficient performance than the first generation public key techniques (RSA and Diffie-Hellman) now in use. As vendors look to upgrade their systems they should seriously consider the elliptic curve alternative for the computational and bandwidth advantages they offer at comparable security."