Practical Cryptographic Systems

Elliptic Curve Cryptography

Some Housekeeping

- Weekly HW#2 coming out shortly
- Start looking for a project group (proposal due in 1.5 weeks 10/6)!

Last time: Key Strength

Level	Protection	Symmetric	Asymmetric	Discrete Logarithm Key	Discrete Logarithm Group	Elliptic Curve	Hash
1	Attacks in "real-time" by individuals *Only acceptable for authentication tag size*	32	-	-	-	-	-
2	Very short-term protection against small organizations *Should not be used for confidentiality in new systems*	64	816	128	816	128	128
3	Short-term protection against medium organizations, medium-term protection against small organizations	72	1008	144	1008	144	144
4	Very short-term protection against agencies, long-term protection against small organizations *Smallest general-purpose level,* *Use of 2-key 3DES restricted to 2^{40} plaintext/ciphertexts,* *protection from 2009 to 2011*	80	1248	160	1248	160	160
5	Legacy standard level *Use of 2-key 3DES restricted to 10^{6} plaintext/ciphertexts,* *protection from 2009 to 2018*	96	1776	192	1776	192	192
6	Medium-term protection *Use of 3-key 3DES,* *protection from 2009 to 2028*	112	2432	224	2432	224	224
7	Long-term protection *Generic application-independent recommendation,* *protection from 2009 to 2038*	128	3248	256	3248	256	256
8	"Foreseeable future" *Good protection against quantum computers*	256	15424	512	15424	512	512

Source: www.keystrength.com (BlueKrypt). Based on ECrypt recommendations.

Why Elliptic Curves

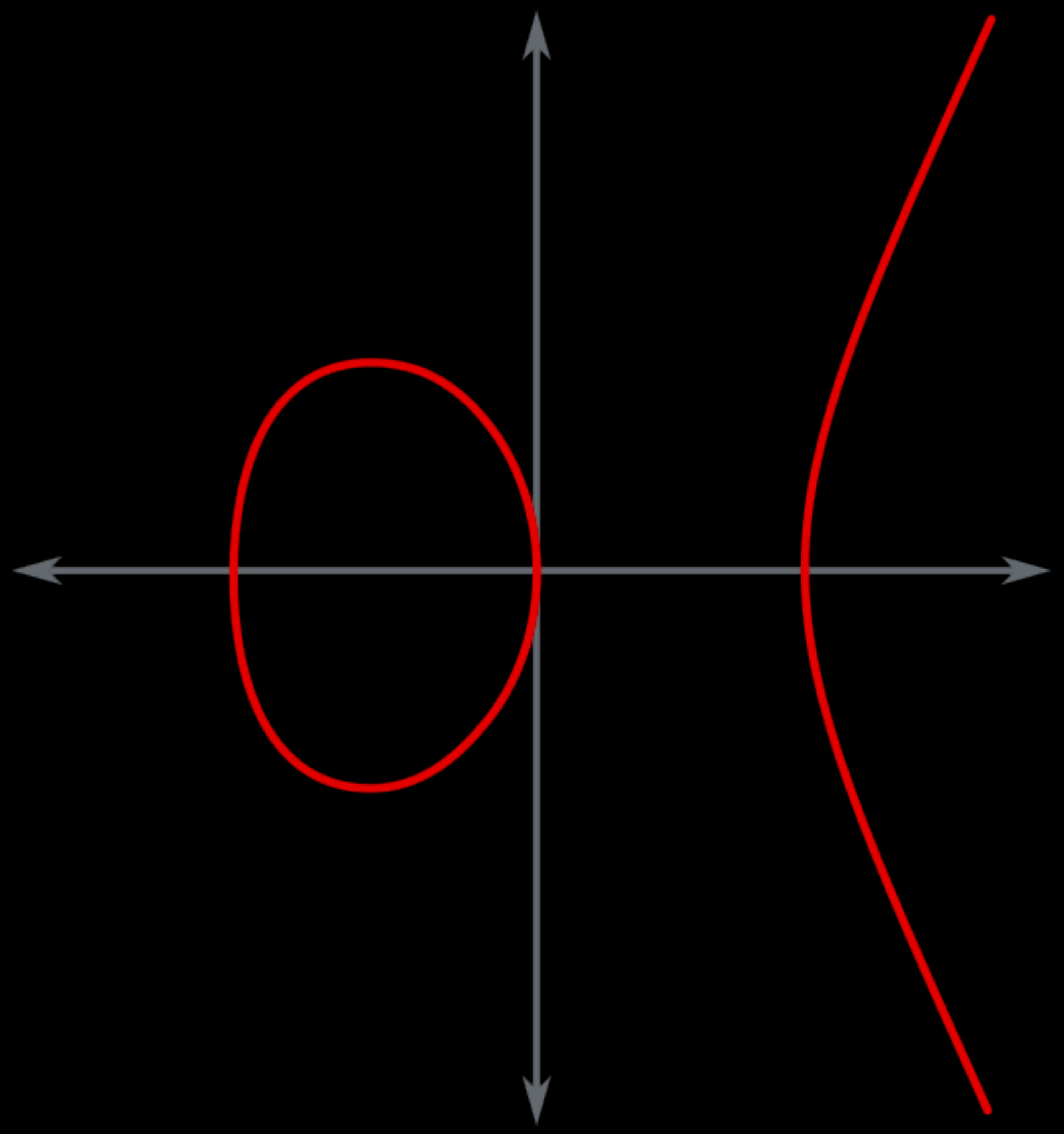
- Prior cryptosystems used finite field ($\mathbb{Z}_p$) based Discrete Log and Factoring
  - These have additional structure that have yielded subexponential time algorithms
- As a result, recommended key sizes are quite large
  - At least 2048 bit keys for RSA and Diffie-Hellman
  - Larger keys means slower operations

Why Elliptic Curves

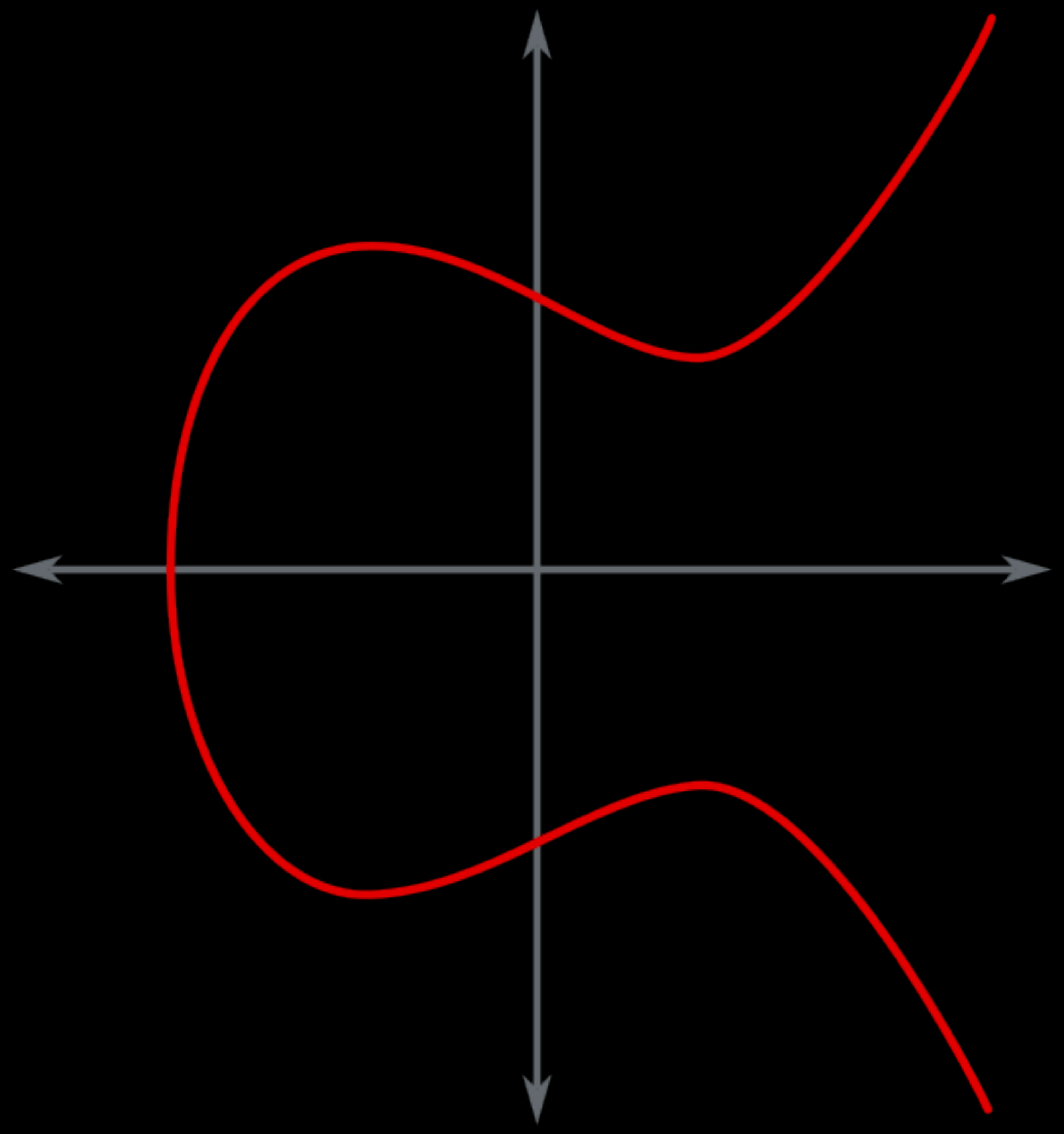
- Prior cryptosystems used finite field ($\mathbb{Z}_p$) based Discrete Log and Factoring
  - These have additional structure that have yielded subexponential time algorithms
- As a result, recommended key sizes are quite large
  - At least 2048 bit keys for RSA and Diffie-Hellman
  - Larger keys means slower operations

Elliptic Curves in general do not have subexponential time cryptanalysis so we can use much smaller keys for similar level of security

Elliptic Curves



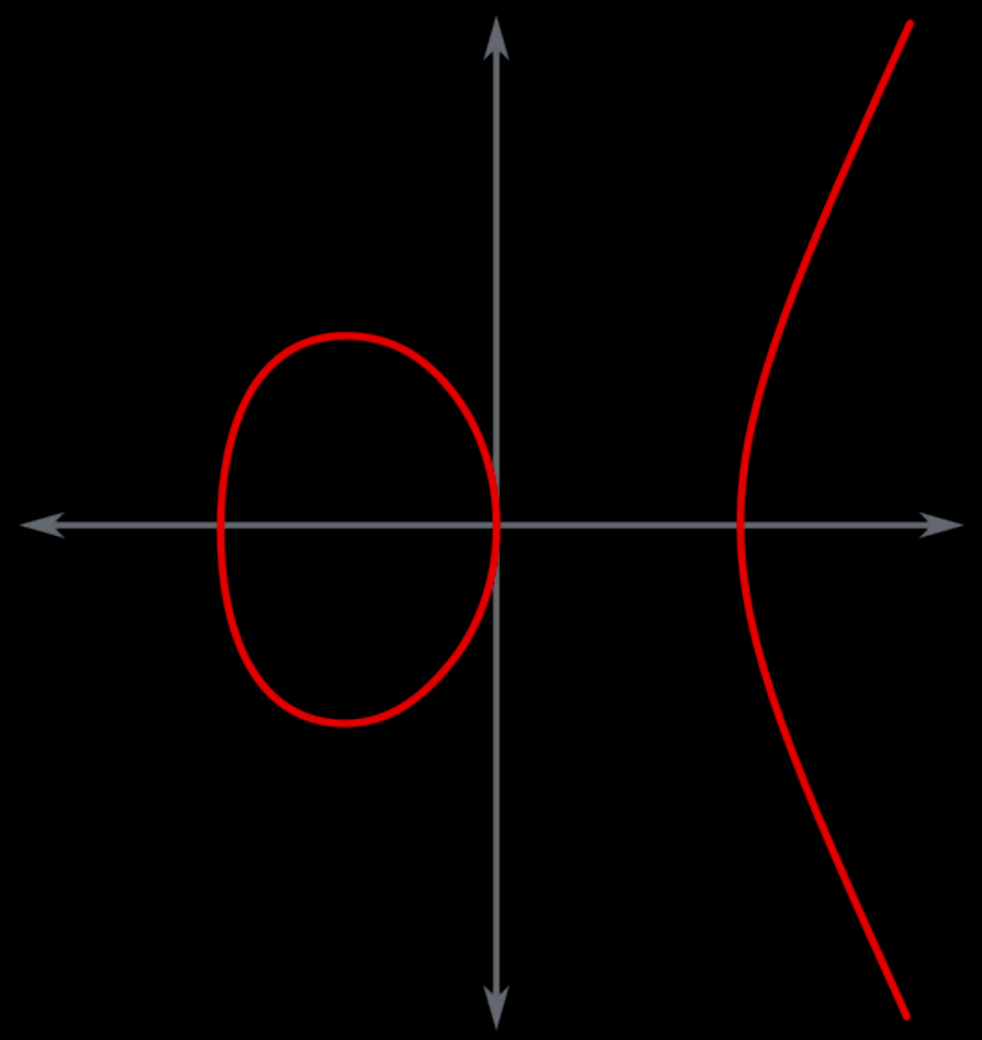
$$y^2 = x^3 - x$$



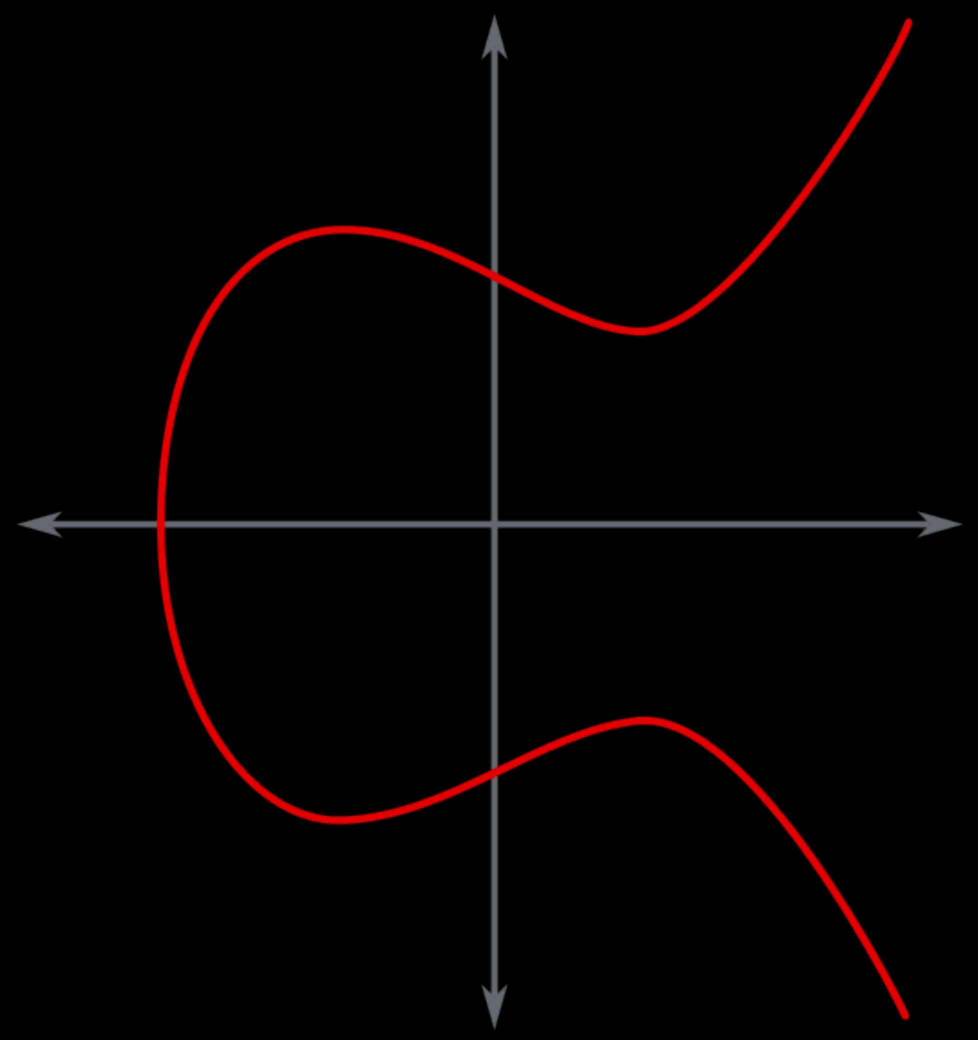
$$y^2 = x^3 - x + 1$$

What is an Elliptic Curve

- A curve defined by an equation $y^2 = x^3 + ax + b$
- These curves were plotted over $\mathbb{R}^2$



$y^2 = x^3 - x$

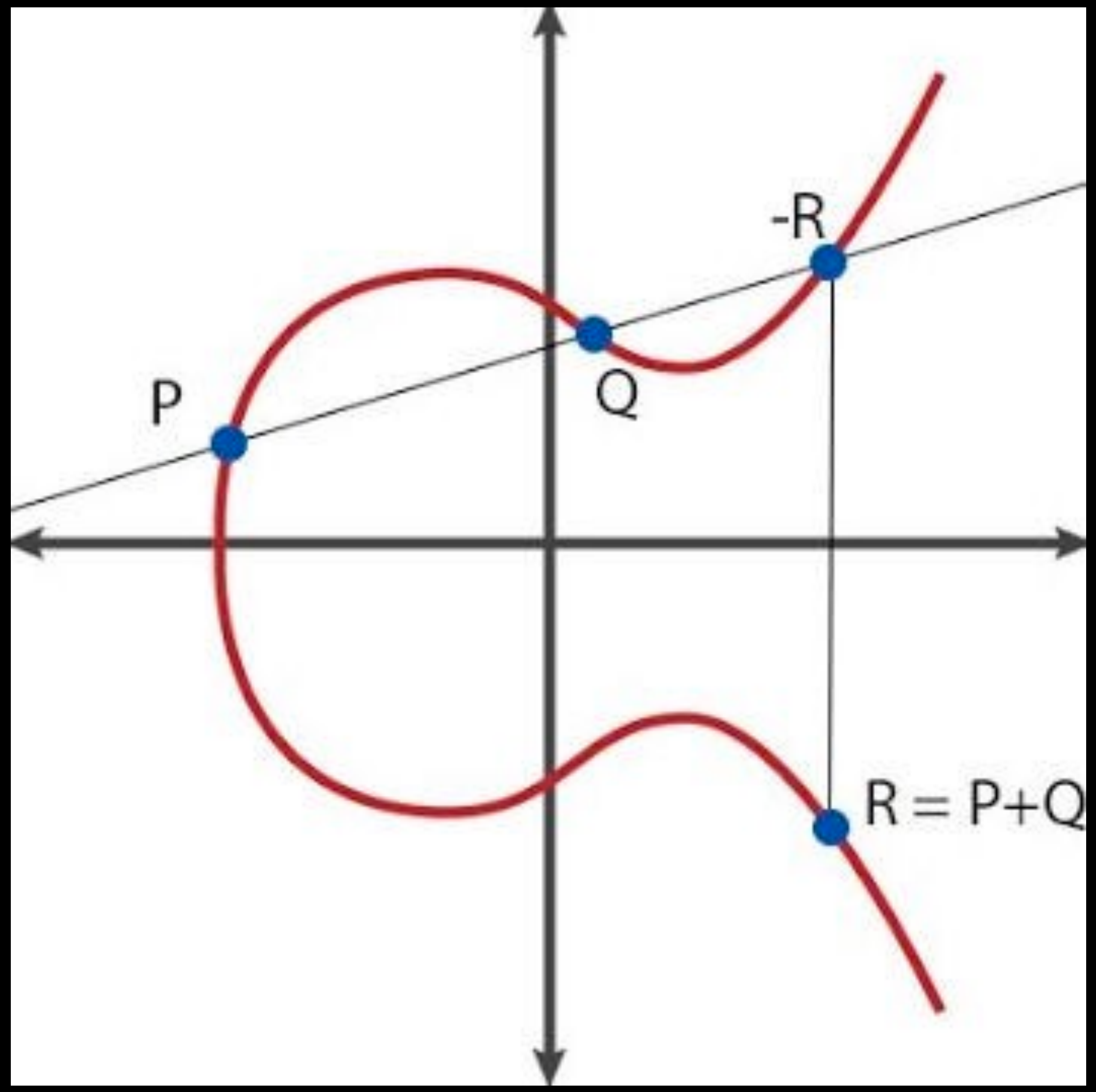


$y^2 = x^3 - x + 1$

Elliptic Curve Points

- Poincare’s method for finding rational points: Take 2 rational points P,Q and define a line that goes through them. We can solve this to find additional rational points. In this process we obtain two new rational points R,-R

- We can define this as a Group law: P+Q = R



Elliptic Curves over Finite Fields

- A Finite Field is an extension of a group that is a group over both addition and multiplication.
  - $\mathbb{Z}_p$ is a field, often denoted as $\mathbb{F}_p$
- For cryptography we define curves over $\mathbb{F}_p$
- Weierstrauss form: $y^2 = x^3 + ax + b,\ a, b \in \mathbb{F}_p,\ 4a^3 + 27b^2 \neq 0$
  - Every Elliptic Curve can be written in this form

Elliptic Curves as a Group

- We claim Elliptic Curves are a group over “addition”, P+Q=R
- Writing the group law formally in terms of (x,y) coordinates requires multiple different cases.
- The identity element is the “point at infinity” $\mathcal{O}$
- Inverses are points symmetric over the x-axis (R and -R)

Types of Elliptic Curves

- Different Elliptic Curves have advantages
- Curves in Montgomery form have faster addition algorithms
- Edwards curves have a simpler addition group law
  - For $\mathrm{P}_1 = (x_1, y_1)$, $\mathrm{P}_2 = (x_2, y_2)$:

$$\mathrm{P}_1 + \mathrm{P}_2 = \left(\frac{x_1y_2 + x_2y_1}{1 + dx_2y_1y_2}, \frac{y_1y_2 - x_1x_2}{1 - dx_2y_1y_2}\right)$$

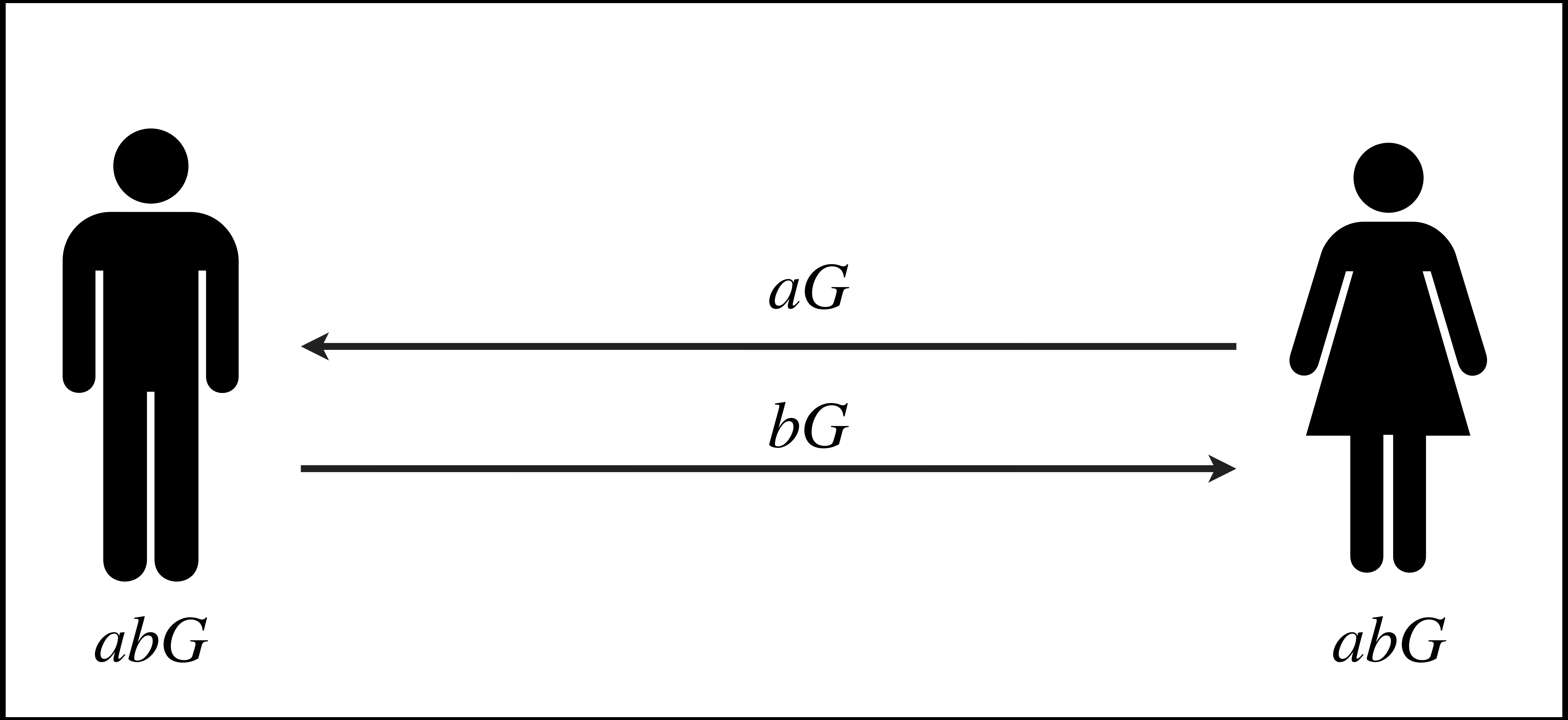
Elliptic Curve Scalar Multiplication

- Classically, scalar multiplication is repeated addition (5x = x+x+x+x+x)
- Scalar multiplication for Elliptic Curves is repeated group “addition”
- To get point aP, we apply group “addition” on P a times
- There is an efficient algorithm for this: Double and Add. (This is analogous to the square and multiply algorithm we have seen previously)

Elliptic Curves and Discrete Log

- Given a point Q, find a such that Q=aP
- Best known algorithms for EC discrete log (pollard rho, baby-step-giant-step) take time $O(\sqrt{n})$ for group of order n
- Still broken in polynomial time by a quantum computer

Elliptic Curve Diffie-Hellman



ECDSA

Key Generation

An elliptic curve over $\mathbb{F}_p$ E

G a group generator on E of order n

$$Q = dG$$

Output:

$$pk = Q$$

$$sk = d$$

Signing

Generate random k

Denote r as the x coordinate of kG

$s = k^{-1}(H(m) + dr) \ mod \ n$

Output (r, s)

Verification

$u_1 = H(m)s^{-1} \ mod \ n$

$u_2 = rs^{-1} \ mod \ n$

Check if r matches the x coordinate of $u_1G + u_2Q$

Note about Elliptic Curve Notation

- Because the group law is “addition”, many references use additive notation for EC groups (P+Q = R, aP = P+P+P+P+P)
- So far our previous examples using groups in cryptography used multiplicative notation ($a \cdot b = c$, $g^a = g \cdot g \cdot g \cdot g \cdot g$)
- Generally, in cryptography we use groups abstractly and treat them as multiplicative groups.
  - The underlying group might be additive
  - This is fine because there is a 1-1 mapping for inverses, identities, and group operations

Standardized Elliptic Curves

- Hard to develop curves resistant to known attacks so everyone uses a small set of curves

Standardized Elliptic Curves

- Hard to develop curves resistant to known attacks so everyone uses a small set of curves
- NIST P256
    - Curve over $\mathbb{F}_p$ with $p \approx 2^{256}$
    - Has prime order $log(q) \approx 256$
    - Parameters have a “suspicious” origin

Standardized Elliptic Curves

- Hard to develop curves resistant to known attacks so everyone uses a small set of curves
- NIST P256
  - Curve over $\mathbb{F}_p$ with $p \approx 2^{256}$
  - Has prime order $log(q) \approx 256$
  - Parameters have a “suspicious” origin
- Curve25519
  - Edwards Curve created by Daniel J. Bernstein
  - Simple group law that is protects against common side channels
  - Little point validation needed
  - Great when only x coordinate validation needed

Next time:

- Protocols and TLS