WILFRID LAURIER UNIVERSITY
LAURIER
Inspiring Lives.

CP 460 - Applied Cryptography

Elliptic Curve Cryptography

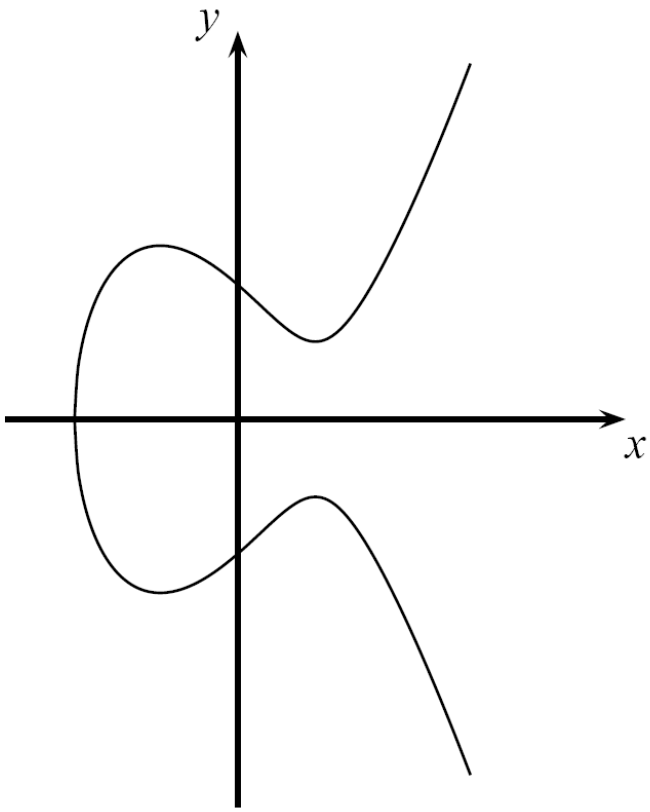
Department of Physics and Computer Science
Faculty of Science, Waterloo

Abbas Yazdinejad, Ph.D.

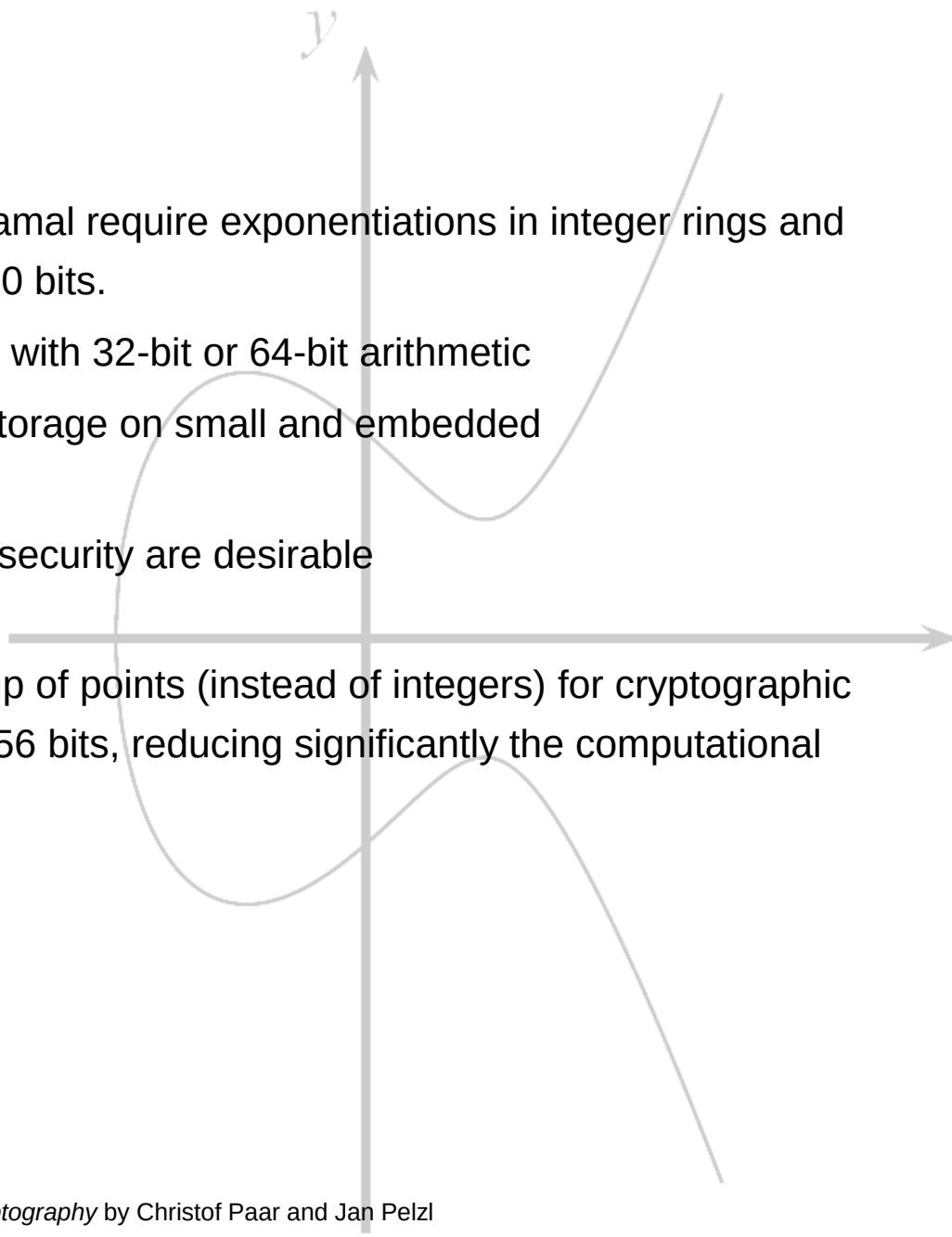
Fall 2024

Content of this Chapter

- Introduction
- Computations on Elliptic Curves
- The Elliptic Curve Diffie-Hellman Protocol
- Security Aspects
- Implementation in Software and Hardware



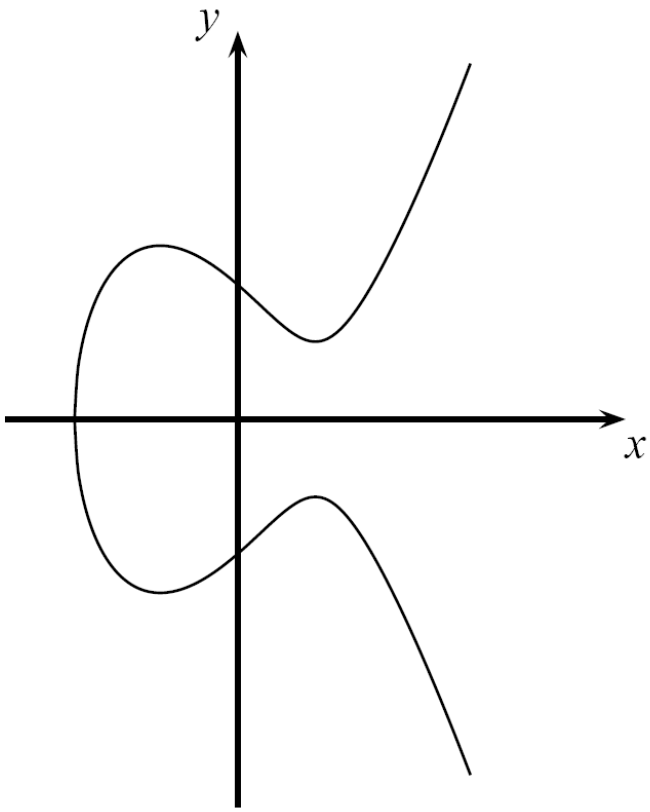
Motivation



- **Problem:**
  Asymmetric schemes like RSA and Elgamal require exponentiations in integer rings and fields with parameters of more than 1000 bits.
  - High computational effort on CPUs with 32-bit or 64-bit arithmetic
  - Large parameter sizes critical for storage on small and embedded
- **Motivation:**
  Smaller field sizes providing equivalent security are desirable
- **Solution:**
  Elliptic Curve Cryptography uses a group of points (instead of integers) for cryptographic schemes with coefficient sizes of 160-256 bits, reducing significantly the computational effort.

Content of this Chapter

- Introduction
- **Computations on Elliptic Curves**
- The Elliptic Curve Diffie-Hellman Protocol
- Security Aspects
- Implementation in Software and Hardware



Computations on Elliptic Curves

- Elliptic curves are polynomials that define points based on the (simplified) Weierstraß equation:

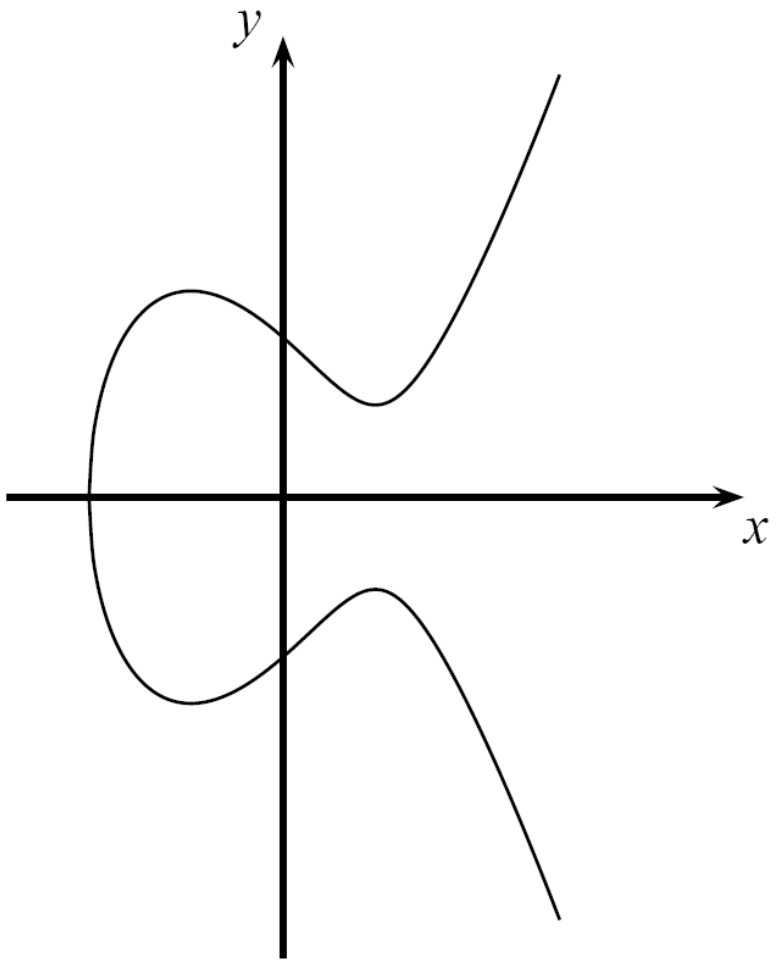
$$y^2 = x^3 + ax + b$$

for parameters a,b that specify the exact shape of the curve

- On the real numbers and with parameters a, b $\in$ R, an elliptic curve looks like this →

- Elliptic curves can not just be defined over the real numbers *R* but over many other types of finite fields.

Elliptic curves are fundamental in ECC. The Weierstrass equation defines these curves mathematically.



Example: $y^2 = x^3 - 3x + 3$ over *R*

Computations on Elliptic Curves (ctd.)

- In cryptography, we are interested in elliptic curves module a prime *p*:

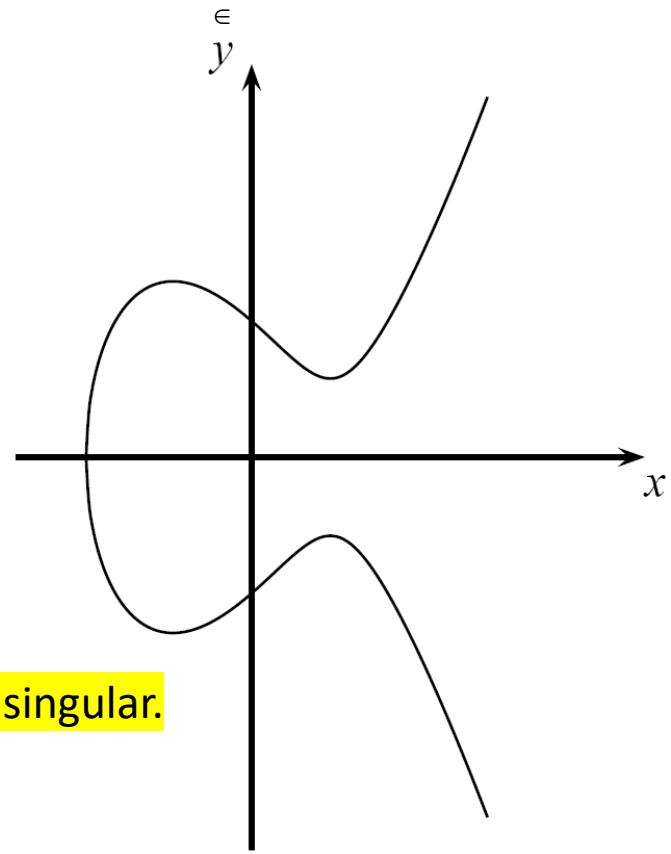
Definition: Elliptic Curves over prime fields

The elliptic curve over Z_p, p>3 is the set of all pairs (x,y) $\in Z_p$ which fulfill

$$y^2 = x^3 + ax + b \bmod p$$

together with an imaginary point of infinity θ, where a,b $\in Z_p$ and the condition

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



The definition of elliptic curve requires that the curve is nonsingular.

- Note that Z_p = *{0,1,…, p -1}* is a set of integers with modulo p arithmetic

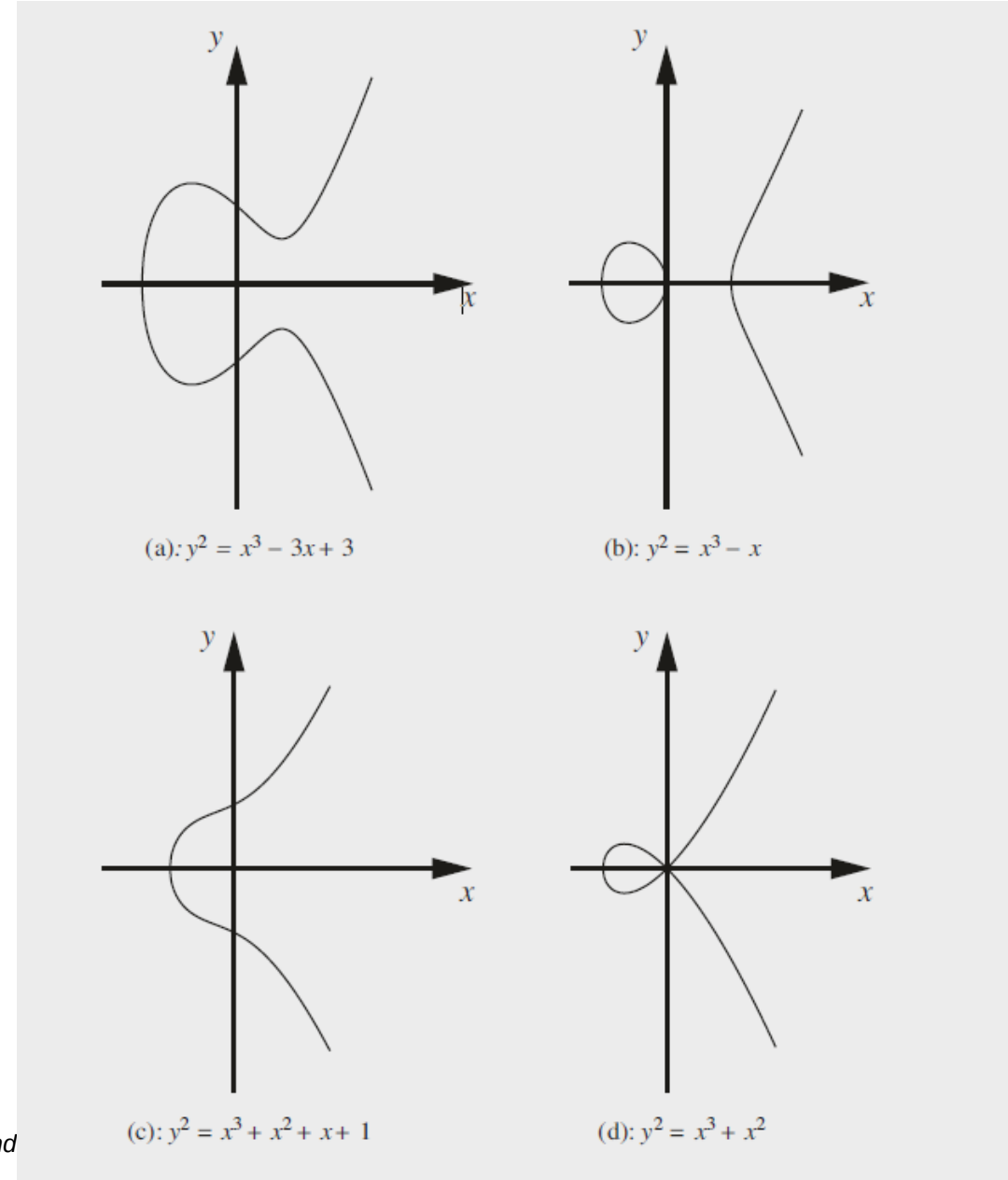
Different elliptic curves over the real numbers are shown

We notice several things from these elliptic curve plots.

First, elliptic curves are symmetric with respect to the x-axis.

Second, there are between one and three intersections with the x-axis

Group Operations on Elliptic Curves



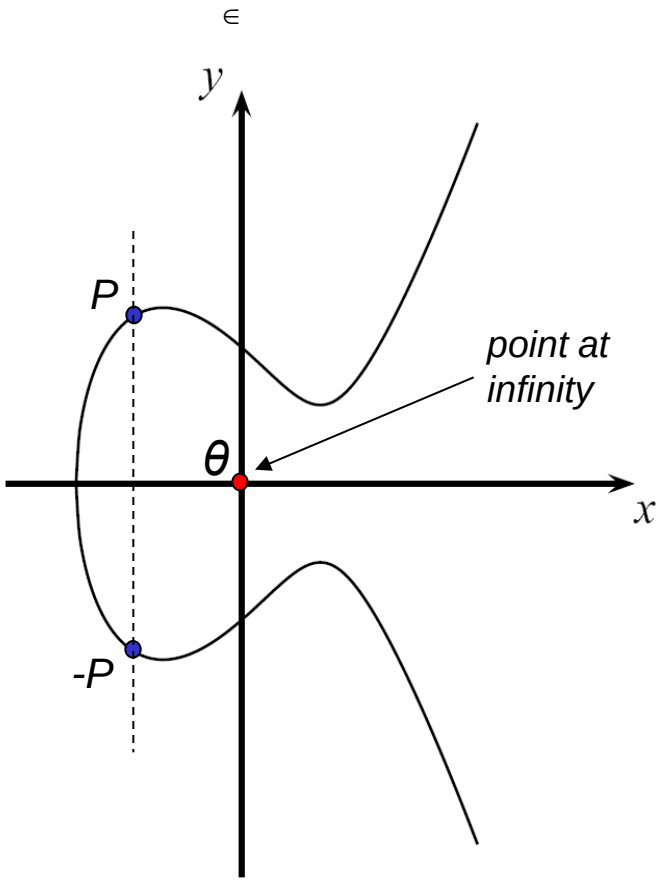
Computations on Elliptic Curves (ctd.)

- Some special considerations are required to convert elliptic curves into a group of points
  - *In any group, a special element is required to allow for the identity operation, i.e., given $P \in E$: $P + \theta = P = \theta + P$*
  - *This identity point (which is not on the curve) is additionally added to the group definition*
  - *This (infinite) identity point is denoted by θ*

- Elliptic Curve are symmetric along the *x*-axis
  - Up to two solutions *y and -y* exist for each quadratic residue *x* of the elliptic curve
  - For each point *P =(x,y),* the inverse or negative point is defined as *-P =(x,-y)*

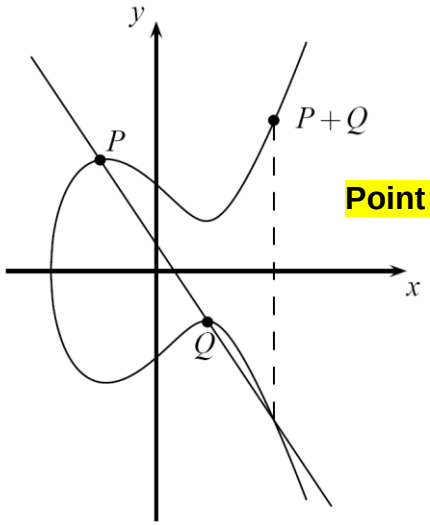
Point Addition

Point Doubling

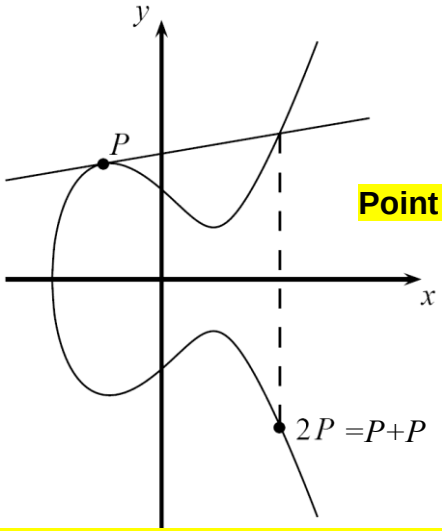


Computations on Elliptic Curves (ctd.)

- Generating a *group of points* on elliptic curves based on point addition operation $P+Q = R$, *i.e.*, $(x_P,y_P)+(x_Q,y_Q) = (x_R,y_R)$
- Geometric Interpretation of point addition operation
  - *Draw straight line through P and Q; if P=Q use tangent line instead*
  - *Mirror third intersection point of drawn line with the elliptic curve along the x-axis*
- Elliptic Curve Point Addition and Doubling Formulas



Point Addition



Point Doubling

$x_3 = s^2 - x_1 - x_2 \bmod p$ and $y_3 = s(x_1 - x_3) - y_1 \bmod p$

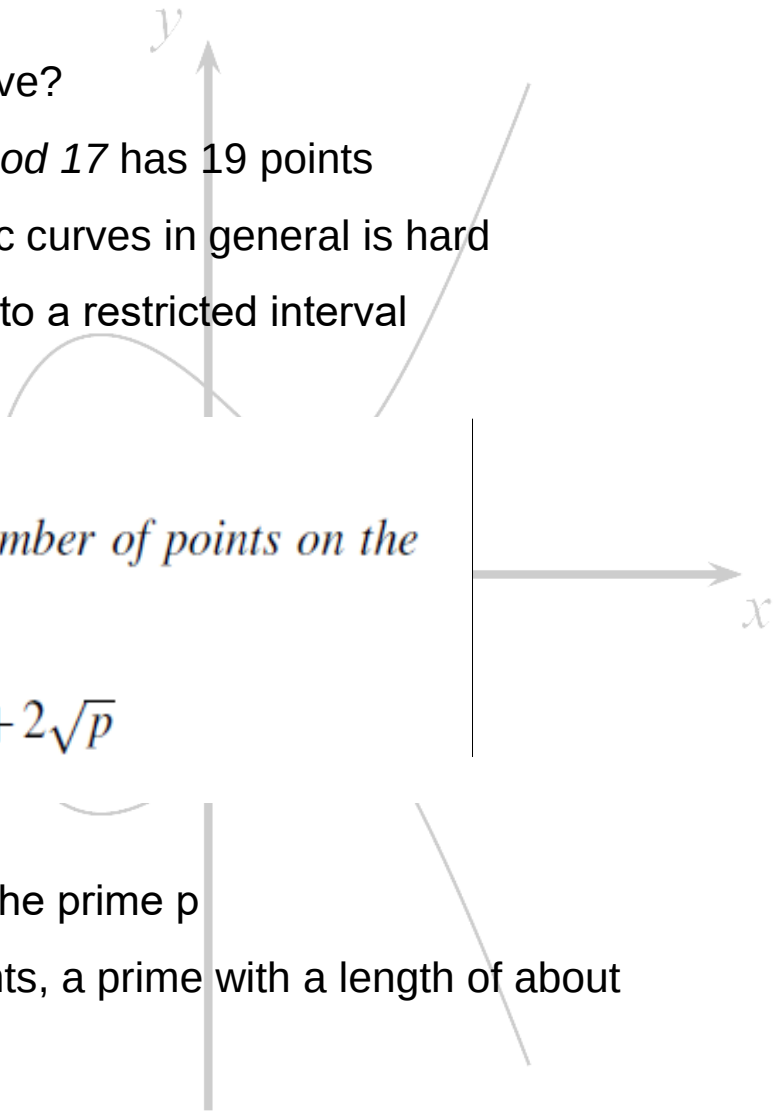
where

$$s = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1} \bmod p & \text{; if P} \neq \text{Q (point addition)} \\ \dfrac{3x_1^2 + a}{2y_1} \bmod p & \text{; if P = Q (point doubling)} \end{cases}$$

Note that the parameter s is the slope of the line through P and Q in the case of point addition and the slope of the tangent through P in the case of point doubling.

Number of Points on an Elliptic Curve

- How many points can be on an arbitrary elliptic curve?
  - Consider previous example: *E:* $y^2 = x^3+2x+2 \bmod 17$ has 19 points
  - However, determining the point count on elliptic curves in general is hard
- But Hasse's theorem bounds the number of points to a restricted interval



Theorem 9.2.2 Hasse's theorem

Given an elliptic curve E modulo p, the number of points on the curve is denoted by #E and is bounded by:

$$p+1-2\sqrt{p} \leq \#E \leq p+1+2\sqrt{p}$$

- **Interpretation:** The number of points is „close to" the prime p
- **Example:** To generate a curve with about 2^{160} points, a prime with a length of about 160 bits is required

Elliptic Curve Discrete Logarithm Problem

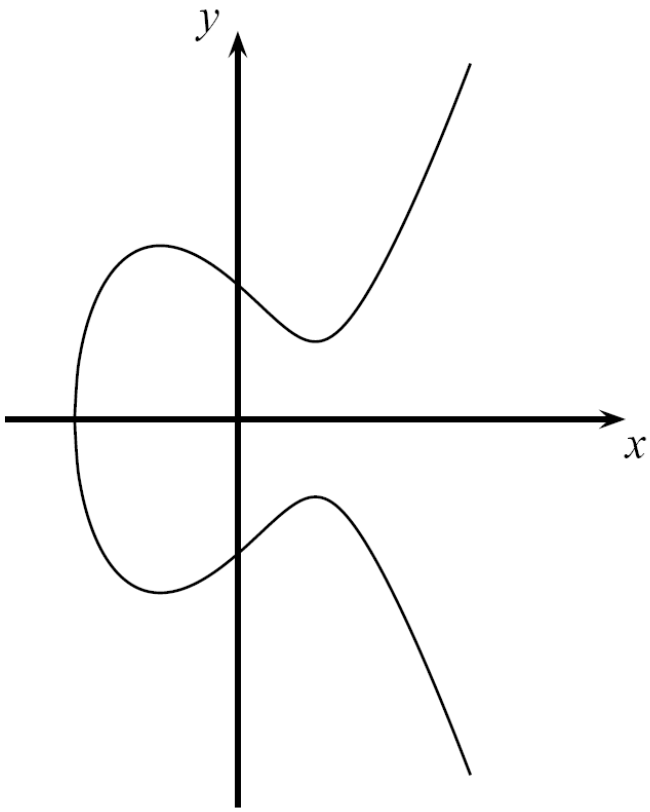
- Cryptosystems rely on the hardness of the Elliptic Curve Discrete Logarithm Problem (ECDLP)

> ***Definition: Elliptic Curve Discrete Logarithm Problem (ECDLP)***
>
> *Given a primitive element P and another element T on an elliptic curve E. The ECDL problem is finding the integer d, where 1 ≤ d ≤ #E such that*
>
> $$\underbrace{P + P + \ldots + P}_{d \text{ times}} = dP = T.$$

- Cryptosystems are based on the idea that *d* is large and kept secret and attackers cannot compute it easily
- If *d* is known, an efficient method to compute the point multiplication *dP* is required to create a reasonable cryptosystem
  - Known Square-and-Multiply Method can be adapted to Elliptic Curves
  - The method for efficient point multiplication on elliptic curves: Double-and-Add Algorithm

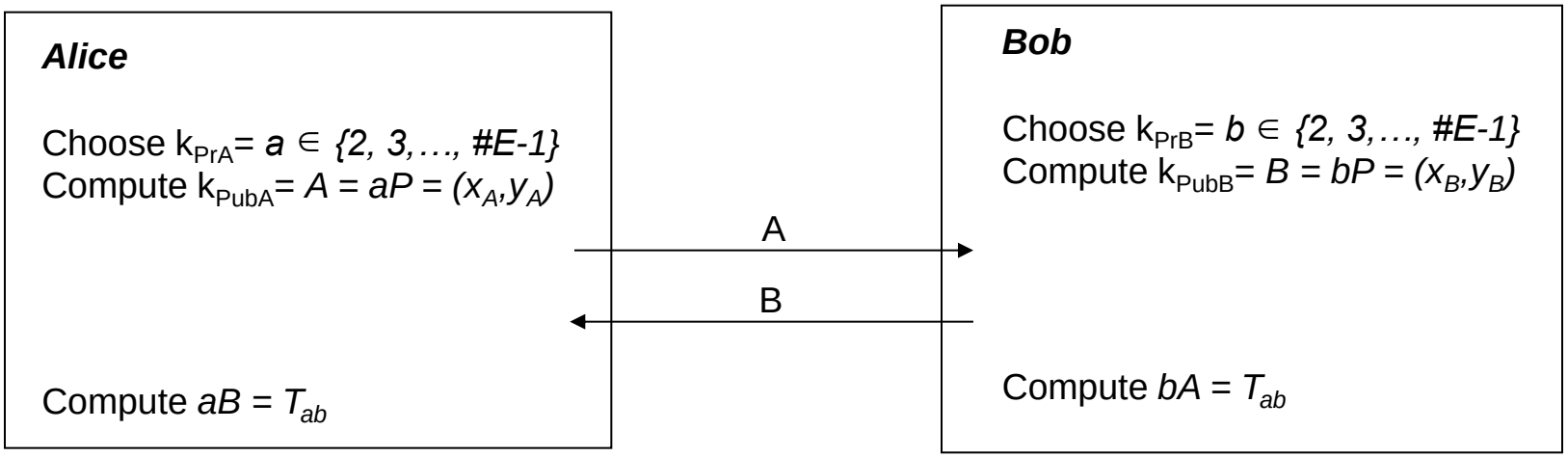
Content of this Chapter

- Introduction
- Computations on Elliptic Curves
- **The Elliptic Curve Diffie-Hellman Protocol**
- Security Aspects
- Implementation in Software and Hardware



The Elliptic Curve Diffie-Hellman Key Exchange (ECDH)

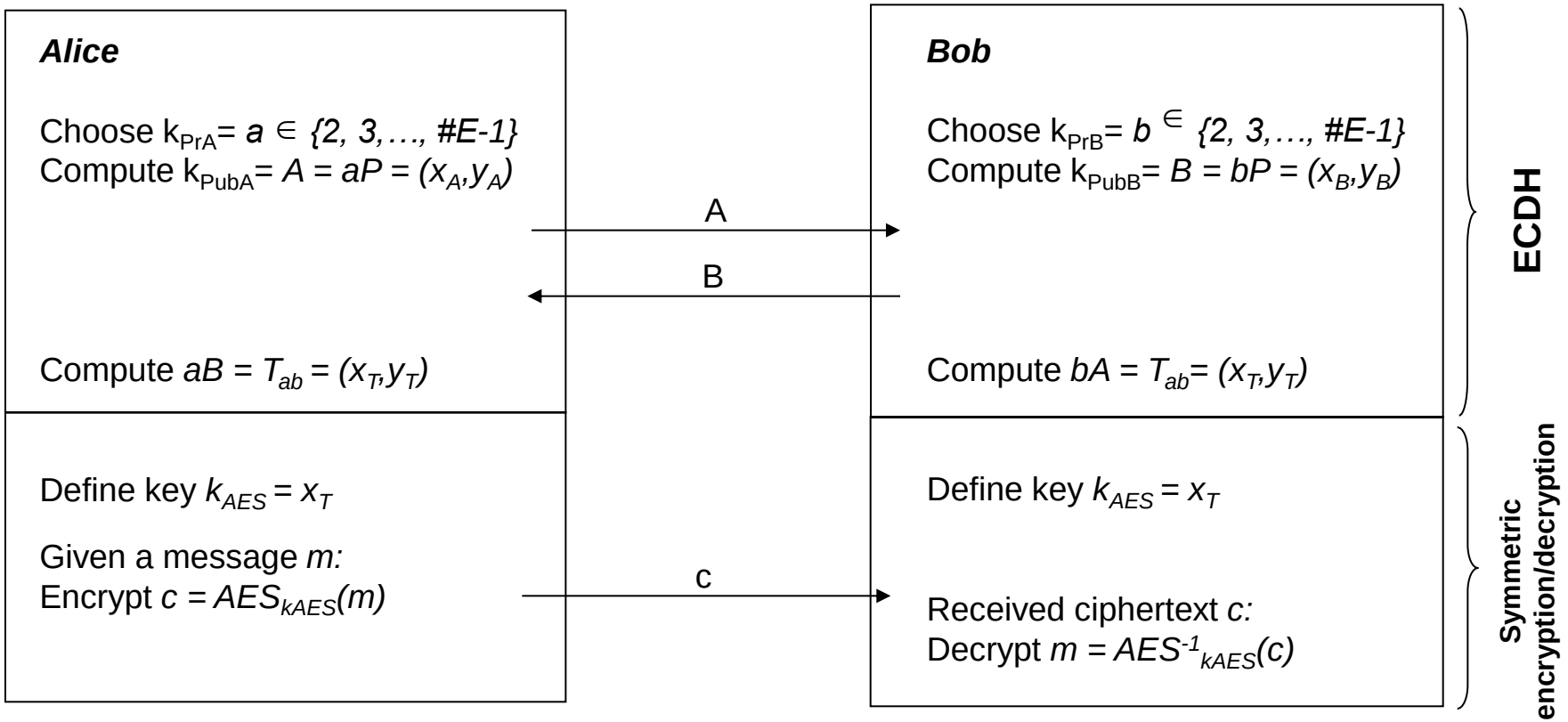
- Given a prime p, a suitable elliptic curve E and a point $P=(x_P,y_P)$
- The Elliptic Curve Diffie-Hellman Key Exchange is defined by the following protocol:



- Joint secret between Alice and Bob: $T_{AB} = (x_{AB}, y_{AB})$
- Proof for correctness:
  - Alice computes aB=a(bP)=abP
  - Bob computes bA=b(aP)=abP since group is associative
- One of the coordinates of the point T_{AB} (usually the x-coordinate) can be used as session key (often after applying a hash function)

The Elliptic Curve Diffie-Hellman Key Exchange (ECDH) (ctd.)

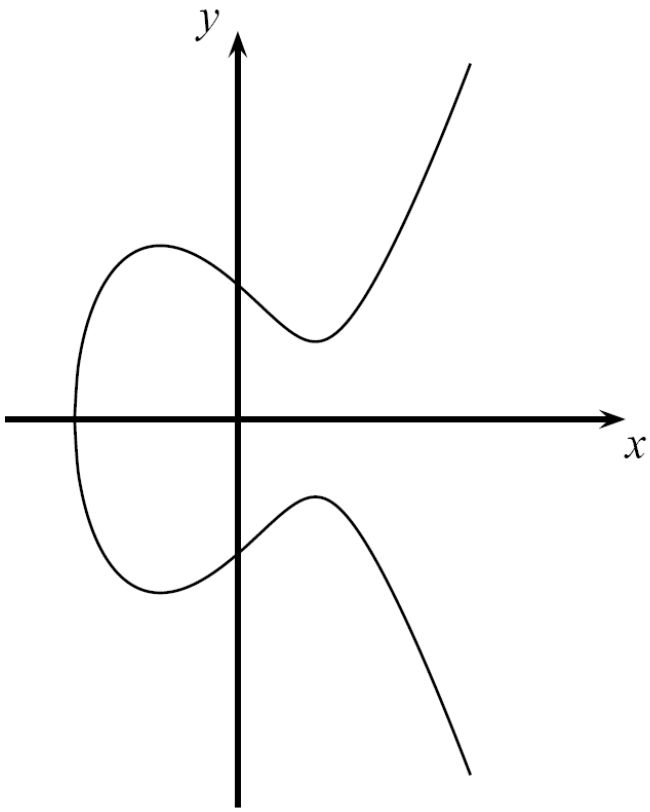
- The ECDH is often used to derive session keys for (symmetric) encryption
- One of the coordinates of the point T_{AB} (usually the x-coordinate) is taken as session key



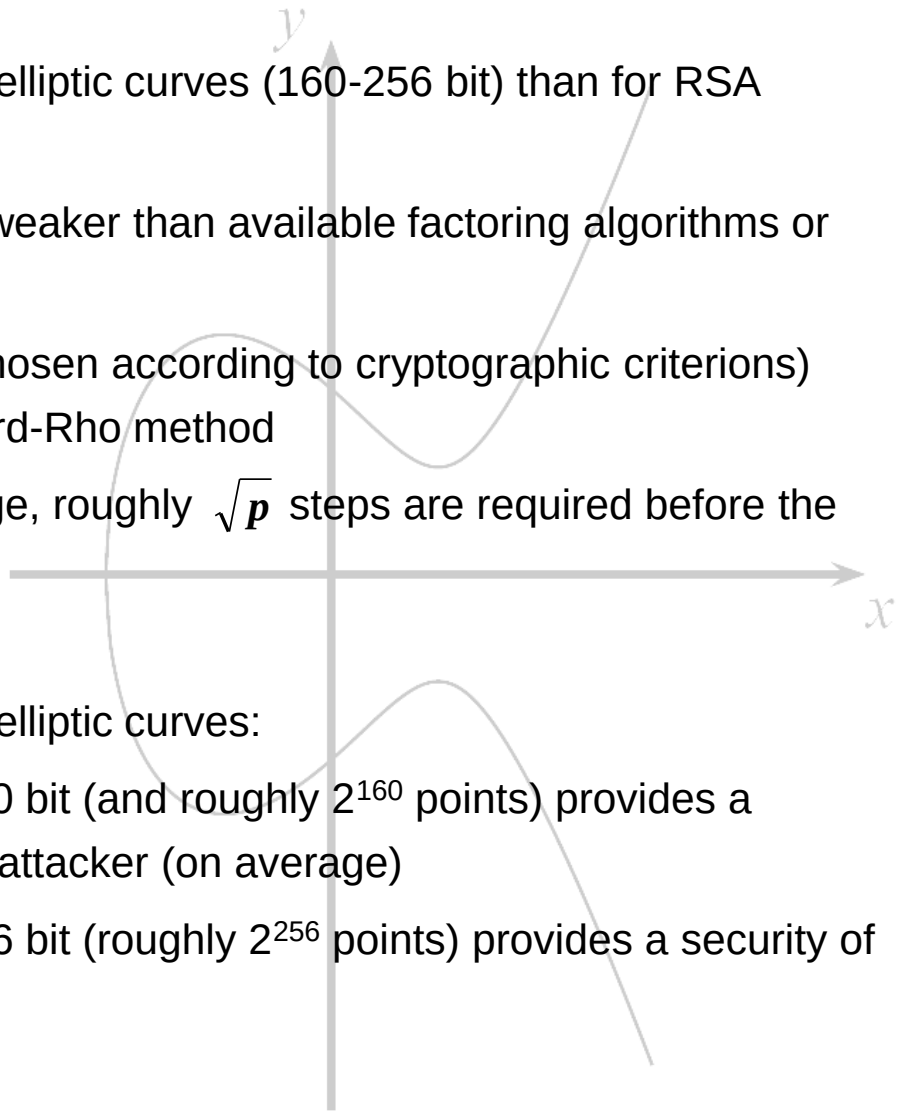
- In some cases, a hash function (see next chapters) is used to derive the session key

Content of this Chapter

- Introduction
- Computations on Elliptic Curves
- The Elliptic Curve Diffie-Hellman Protocol
- **Security Aspects**
- Implementation in Software and Hardware



Security Aspects

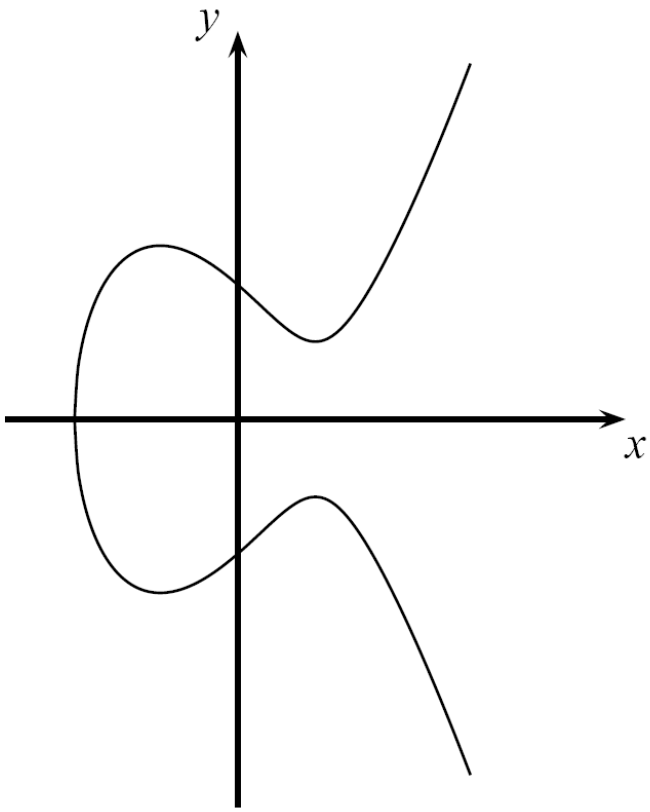


- Why are parameters signficantly smaller for elliptic curves (160-256 bit) than for RSA (1024-3076 bit)?
  - Attacks on groups of elliptic curves are weaker than available factoring algorithms or integer DL attacks
  - Best known attacks on elliptic curves (chosen according to cryptographic criterions) are the Baby-Step Giant-Step and Pollard-Rho method
  - Complexity of these methods: on average, roughly $\sqrt{p}$ steps are required before the ECDLP can be successfully solved

- Implications to practical parameter sizes for elliptic curves:
  - An elliptic curve using a prime p with 160 bit (and roughly 2^{160} points) provides a security of 2^{80} steps that required by an attacker (on average)
  - An elliptic curve using a prime p with 256 bit (roughly 2^{256} points) provides a security of 2^{128} steps on average

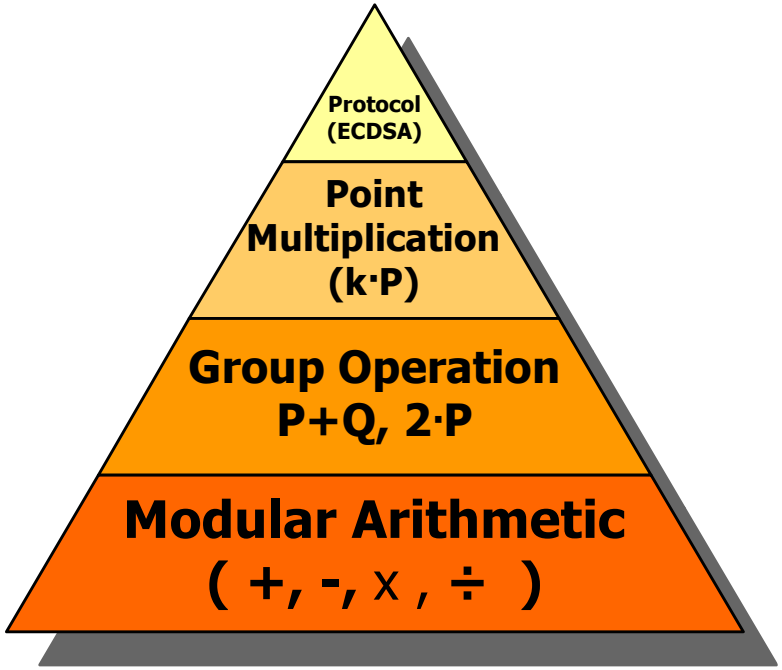
Content of this Chapter

- Introduction
- Computations on Elliptic Curves
- The Elliptic Curve Diffie-Hellman Protocol
- Security Aspects
- **Implementation in Software and Hardware**



Implementations in Hardware and Software

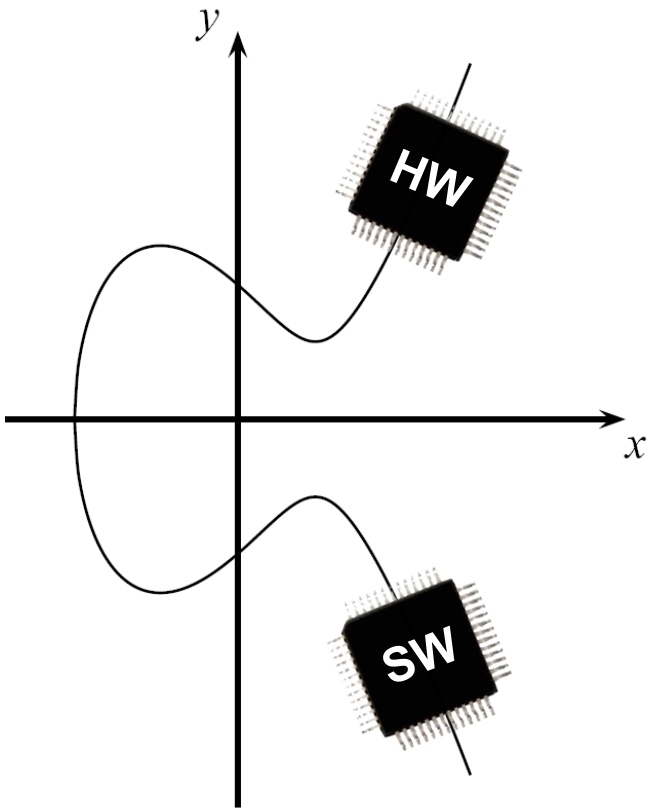
- Elliptic curve computations usually regarded as consisting of four layers:
  - Basic modular arithmetic operations are computationally most expensive
  - Group operation implements point doubling and point addition
  - Point multiplication can be implemented using the Double-and-Add method
  - Upper layer protocols like ECDH and ECDSA



- Most efforts should go in optimizations of the modular arithmetic operations, such as
  - Modular addition and subtraction
  - Modular multiplication
  - Modular inversion

Implementations in Hardware and Software

- Software implementations
  - Optimized 256-bit ECC implementation on 3GHz 64-bit CPU requires about *2 ms* per point multiplication
  - Less powerful microprocessors (e.g, on SmartCards or cell phones) even take significantly longer (*>10 ms*)
- Hardware implementations
  - High-performance implementations with 256-bit special primes can compute a point multiplication in a few hundred microseconds on reconfigurable hardware
  - Dedicated chips for ECC can compute a point multiplication even in a few ten microseconds



Below is the difference between ECC and RSA:

Parameters	ECC	RSA
Working algorithm	ECC is a cryptography technique that works just on a mathematical model of elliptic curves.	RSA cryptography algorithm is primarily based on the prime factorization approach.
Bandwidth savings	ECC gives significant bandwidth savings over RSA.	RSA provides much lesser bandwidth saving than ECC.
Encryption process	The encryption process takes less time in ECC.	The encryption process takes more time in RSA.
Decryption process	The decryption process takes more time.	Decryption is faster than ECC.
Security	ECC is much safer than RSA and is currently in the process of adapting.	RSA is heading toward the end of its tenure.

Benefits of Elliptic Curve Cryptography

- **Fast key generation:** ECC cryptography's key creation is as simple as securely producing a random integer in a specific range, making it highly quick. Any integer in the range represents a valid ECC secret key. The public keys in the ECC are EC points, which are pairs of integer coordinates x, and y that lie on a curve.
- **Smaller key size:** Cipher text, signatures, and Elliptic-curve cryptography (ECC) is a public-key encryption technique based on the algebraic structure of elliptic curves with finite fields. Compared to non-EC encryption (based on ordinary Galois fields), ECC allows for fewer keys to guarantee equal security.
- **Low latency:** Signatures can be computed in two stages, allowing latency much lower. By computing signatures in two stages, ECC achieves lower latency than the inverse throughout. ECC has robust protocols for authorized key exchange, and the technology has widespread adoption.
- **Less computation power:** Since the ECC key is shorter the computation power is also less computational power, ECC offers high security with faster, shorter keys compared to RSA and take more energy to factor than it does to calculate an elliptic curve objective function.
- **High security:** A 256-bit ECC public key ensures comparable security to a 3072-bit RSA public key. With ECC, you may obtain the same level of security with smaller keys. ECC provides strong security in a world where mobile phones must do more and more encryption with fewer computational resources.

Limitations of Elliptic Curve Cryptography

- **Large encryption size:** ECC increases the size of the encrypted message significantly more than RSA encryption. The default key length for ECC private keys is 256 bits, but many different ECC key sizes are conceivable depending on the curve.
- **A more complex:** The ECC algorithm is more complete and more difficult to implement than RSA. Algorithms cost have been computed from the computation of the elliptic curve operation and finite field operations that determine the running time of the scalar multiplication integer sub-decomposition (ISD) method.
- **Complex security:** Complicated and tricky to implement securely, mainly the standard curves. If the key size used is large enough, ECC is regarded to be highly secure. For internal communications, the US government needs ECC with a key size of either 256 or 384 bits, depending on the sensitivity level of the material being communicated.
- **Binary curves:** Processing of binary curves is costly. Elliptic curve cryptography (ECC) employs elliptic curves over finite fields Fp (where p is prime and p > 3) or F2m (where the field size p = 2 m_). This means that the field is a p x p square matrix, and the points on the curve can only have integer locations within the field.

Lessons Learned

- Elliptic Curve Cryptography (ECC) is based on the discrete logarithm problem. It requires, for instance, arithmetic modulo a prime.
- ECC can be used for key exchange, for digital signatures and for encryption.
- ECC provides the same level of security as RSA or discrete logarithm systems over Z_p with considerably shorter operands (approximately 160–256 bit vs. 1024–3072 bit), which results in shorter ciphertexts and signatures.
- In many cases ECC has performance advantages over other public-key algorithms.
- ECC is slowly gaining popularity in applications, compared to other public-key schemes, i.e., many new applications, especially on embedded platforms, make use of elliptic curve cryptography.


Thank
You