Asymmetrische Kryptographie
u23 2015

Simon, Florob

Chaos Computer Club Cologne e.V.
https://koeln.ccc.de

Cologne
2015-10-05

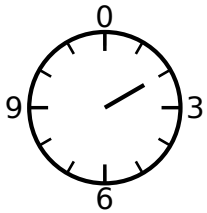
Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Uhrenarithmetik

- Rechnen mit Uhrzeiten (12 Std. Uhr)
- Fängt nach 11 Uhr von vorne an
- 9 Uhr + 4 Std. = 1 Uhr
- 11 Uhr + 9 Std. = 8 Uhr



Modulare Arithmetik

$$a \mod b = a - \left\lfloor \frac{a}{b} \right\rfloor \cdot b$$

Anschaulich: Rest bei der Ganzzahldivision

$$a = b \cdot c + m \quad \text{mit } m < b$$
$$a/b = c + m/b$$
$$a \mod b = m$$

Modulare Arithmetik: Beispiele

- $5 \mod 3 = 2$
- $42 \mod 12 = 6$
- $-2 \mod 4 = 2$

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Kongruenz

$$a \equiv b \pmod{m}$$
$$\Leftrightarrow a = b + k \cdot m \qquad \text{für ein } k \in \mathbb{Z}$$

- Zwei Zahlen mit gleichem Rest bzgl. eines Divisors m heißen *kongruent*
- $19 \equiv 5 \equiv -2 \pmod{7}$
- $0 \equiv 2 \equiv 4 \pmod{2}$

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Halbgruppe

Menge von Elementen (G) mit einem Operator ($\cdot$), sodass gilt:

- Abgeschlossen: $\forall a, b \in G : \ a \cdot b \in G$
- Assoziativität: $\forall a, b, c \in G : \ (a \cdot b) \cdot c = a \cdot (b \cdot c)$

Beispiel Halbgruppe

$$\mathbb{N} \setminus \{0\} = \{1, 2, 3, 4, \ldots\} \text{ unter Addition}$$

- Abgeschlossen: Die Summe zweier natürlicher Zahlen ist eine natürliche Zahl $42 + 23 = 65$
- Assoziativ: z. B. $(2 + 5) + 3 = 7 + 3 = 10 = 2 + 8 = 2 + (5 + 3)$

Gruppe

Menge von Elementen (G) mit einem Operator ($\cdot$), sodass gilt:

- Abgeschlossen: $\forall a, b \in G : \ a \cdot b \in G$
- Assoziativität: $\forall a, b, c \in G : \ (a \cdot b) \cdot c = a \cdot (b \cdot c)$
- Neutrales Element: $e \in G$, sodass $\forall a \in G : \ a \cdot e = e \cdot a = a$
- Inverses Element: $\forall a \in G \ \exists a^{-1} \in G$, sodass $a \cdot a^{-1} = a^{-1} \cdot a = e$

Beispiel Gruppe

$$M_2^* := \{A \in M(2, \mathbb{R}) | \det(A) \neq 0\} \text{ unter Multiplikation}$$

- Abgeschlossen: $\forall A, B \in M_2^* : \ AB \in M_2^*$
- Assoziativität: $\forall A, B, C \in M_2^* : \ (AB) \cdot C = A \cdot (BC)$
- Neutrales Element: $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$
- Inverses Element:

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \qquad A^{-1} = \underbrace{\frac{1}{ad - cb}}_{\frac{1}{\det(A)}} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$$

Abelsche Gruppe

Menge von Elementen (G) mit einem Operator ($\cdot$), sodass gilt:

- Abgeschlossen: $\forall a, b \in G : \ a \cdot b \in G$
- Assoziativität: $\forall a, b, c \in G : \ (a \cdot b) \cdot c = a \cdot (b \cdot c)$
- Neutrales Element: $e \in G$, sodass $\forall a \in G : \ a \cdot e = e \cdot a = a$
- Inverses Element: $\forall a \in G \ \exists a^{-1} \in G$, sodass $a \cdot a^{-1} = a^{-1} \cdot a = e$
- Kommutativ: $\forall a, b \in G : \ a \cdot b = b \cdot a$

Beispiel Abelsche Gruppe

$$\mathbb{Z}^n = \{\ldots, -2, -1, 0, 1, 2, \ldots\}^n \text{ unter Addition}$$

- Abgeschlossen: $\forall \vec{a}, \vec{b} \in \mathbb{Z}^n : \ \vec{a} + \vec{b} \in \mathbb{Z}^n$
- Assoziativität: $\forall \vec{a}, \vec{b}, \vec{c} \in \mathbb{Z}^n : \ (\vec{a} + \vec{b}) + \vec{c} = \vec{a} + (\vec{b} + \vec{c})$
- Neutrales Element: $\vec{0} \in \mathbb{Z}^n$, sodass $\forall \vec{a} \in \mathbb{Z}^n : \ \vec{a} + \vec{0} = \vec{0} + \vec{a} = \vec{a}$
- Inverses Element: $\forall \vec{a} \in \mathbb{Z}^n \ \exists \overrightarrow{-a} \in \mathbb{Z}^n$, sodass $\vec{a} + \overrightarrow{-a} = \overrightarrow{-a} + \vec{a} = \vec{0}$
- Kommutativ: $\forall a, b \in \mathbb{Z}^n : \ a + b = b + a$

Zusammenfassung

- **Halbgruppe** abgeschlossen + assoziativ
- **Gruppe** Halbgruppe + neutrales Element + inverses Element
- **abelsche Gruppe** Gruppe + kommutativ

Ring

Menge von Elementen (R) mit zwei Operatoren ($+$, $\cdot$), sodass gilt:

- $(R, +)$ ist eine abelsche Gruppe
- $(R, \cdot)$ ist eine Halbgruppe
- Distributiv: $\forall a, b, c \in R : \ a \cdot (b + c) = a \cdot b + a \cdot c$

Beispiel Ring

$$\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$$

- $(\mathbb{Z}, +)$ ist eine abelsche Gruppe, neutrales Element 0, inverses Element $-a$
- $(\mathbb{Z}, \cdot)$ ist eine Halbgruppe
- Distributiv: $\forall a, b, c \in R : \ a \cdot (b + c) = a \cdot b + a \cdot c$

Körper

Menge von Elementen (K) mit zwei Operatoren ($+$, $\cdot$), sodass gilt:

- $(K, +)$ ist eine abelsche Gruppe
- $(K \setminus \{0\}, \cdot)$ ist eine abelsche Gruppe
- Distributiv: $\forall a, b, c \in K : \; a \cdot (b + c) = a \cdot b + a \cdot c$

Beispiel Körper

$$\mathbb{Q} = \left\{ \frac{a}{b} | a, b \in \mathbb{Z} \right\}$$

- $(\mathbb{Q}, +)$ ist eine abelsche Gruppe, neutrales Element 0, inverses Element $-a$
- $(\mathbb{Q} \setminus \{0\}, \cdot)$ ist eine abelsche Gruppe, neutrales Element 1, inverses Element $\frac{1}{a}$
- Distributiv: $\forall a, b, c \in K : \ a \cdot (b + c) = a \cdot b + a \cdot c$

Restklassenring

$$\mathbb{Z}_n = \mathbb{Z}/n\mathbb{Z} = \{[x]_n | x \in \{1, \ldots, n\}\}$$

- Restklasse: $[a]_n = \{x | x \equiv a \pmod{n}\} = [a+n]_n$
- Addition: $[a]_n + [b]_n := [a+b]_n$
- Multiplikation: $[a]_n \cdot [b]_n := [a \cdot b]_n$
- z. B. für $n = 5$: $[3]_5 + [8]_5 = [3+8]_5 = [11]_5 = [1]_5$
- einfacher zu Schreiben als Kongruenz: $3 + 8 \equiv 11 \equiv 1 \pmod{5}$

Multiplikative Teilgruppe

$$\mathbb{Z}_n^* = \{x | \, \text{ggT}(x', n) = 1, \; x' \in x, \; x \in \mathbb{Z}_n\}$$

- Idee: Inverse bzgl. Multiplikation modulo n
- z. B. $7 \cdot 3 \equiv 21 \equiv 1 \pmod{10}$, also $7^{-1} \equiv 3 \pmod{10}$
- existiert nicht immer: $5^{-1} \equiv ? \pmod{10}$
- x invertierter modulo $n \Leftrightarrow \text{ggT}(x, n) = 1$

Restklassenkörper

$$\mathbb{Z}_p = \mathbb{Z}/p\mathbb{Z} \quad \text{für eine Primzahl } p$$

- $\mathrm{ggT}(x, p)$ ist stets 1, jedes Element besitzt ein Multiplikatives Inverses

$\Rightarrow$ $\mathbb{Z}_p$ ist nicht nur Ring, sondern auch Körper

Inverse Berechnen 1: Erweiterter euklidischer Algorithmus

- Berechnet r, s in $\mathrm{ggT}(a, b) = ra + sb$
- Sage: `xgcd(a, b) = (ggT(a,b),r,s)`
- Inverses von x modulo n:
  - $\mathrm{ggT}(x, n) = 1 = rx + sn$

  $\Rightarrow r \equiv x^{-1} \pmod{n}$

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Eulersche Phi-Funktion

$$\varphi(n) := |\mathbb{Z}_n^*|$$

- Anzahl der Zahlen $[1, n]$ mit einem multiplikativen Inversen modulo n
- $\varphi(m \cdot n) = \varphi(m) \cdot \varphi(n)$
- $\varphi(p) = p - 1$ für eine Primzahl p
- Für eine Primfaktorzerlegung von $n = \prod_i p_i^{x_i}$ gilt:

$$\varphi(n) = \prod_i (p_i - 1)$$

Kleiner fermatscher Satz

$$a^p \equiv a \pmod{p}$$
$$a^{p-1} \equiv 1 \pmod{p} \qquad a \not\equiv 0 \pmod{p}$$

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Satz von Euler

$$a^{\varphi(n)} \equiv 1 \pmod{n} \qquad \text{ggT}(a, n) = 1$$

- Erweiterung des kleinen feramtschen Satzes
- Konsequenz: $5^{723} \equiv 5^{723 \mod \varphi(7)} \equiv 5^{723 \mod 6} \equiv 5^3 \pmod{7}$

Inverse Berechnen 2

- $x^{p-1} \equiv x^{p-2} \cdot x \equiv 1 \pmod{p}$

$\Rightarrow$ $x^{-1} \equiv x^{p-2} \pmod{p}$

- z. B. $3^5 \equiv 243 \equiv 5 \pmod{7}$
- equivalent $x^{-1} \equiv x^{\varphi(n)-1} \pmod{n}$

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Primfaktorzerlegung

Finde für gegebenes n Primzahlen p_i, sodass $n = \prod_i p_i^{x_i}$

- Kein bekannter effizienter Algorithmus

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Diskrete Wurzeln

Finde für gegebenes (y, e, n) eine Zahl x, sodass $x^e \equiv y \pmod{n}$

- Anschaulich: die x-te Wurzel aus y modulo n
- trivial falls n prim (siehe RSA)
- für n zusammengesetzt, kein besserer Algorithmus als faktorisieren bekannt

Diskreter Logarithmus

Finde für gegebenes (y, x, n) eine Zahl e, sodass $y \equiv x^e \pmod{n}$

- Kein bekannter effizienter Algorithmus
- unendlich viele kongruente Lösungen modulo $\varphi(n)$

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Literatur

- Skripts zu Vorlesungen an der Uni Bonn
- Krypto School, Springer,
  ISBN 978-3662484234

Prof. Joachim von zur Gathen

Problem

Alice und **B**ob wollen verschlüsselt kommunizieren.

Frage: Wie kommen sie an den Schlüssel?

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Problem

Alice und **B**ob wollen verschlüsselt kommunizieren.

Frage: Wie kommen sie an den Schlüssel?

(Eine) Antwort: Schlüsselaustausch nach Diffie-Hellman

- **A** und **B** berechnen denselben Wert
- Schlüsselaustausch über unsicheres Medium
- Mitschnitt verrät den Schlüssel nicht

Zutaten

- (endliche) Gruppe $(G, \cdot)$
- *Generator* $g \in G$
- Schreibweise: $G = \langle g \rangle$

Beispiel

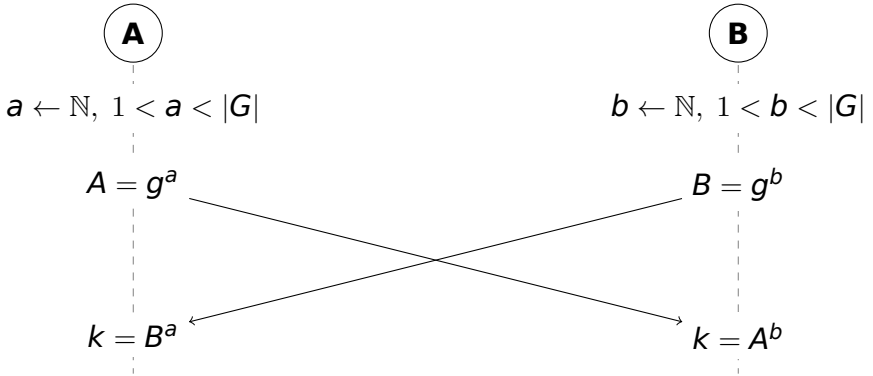
$G = \mathbb{Z}_{17}^*$
$g = 3 \in G$

$G = \{g^x \mid 0 < x < 17\}$

```
G = Zmod(17)
g = G(3)

map(lambda x:g^x, range(17))
```

Verfahren



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

Sicherheit

Übertragen wird nur g^a, g^b.

Frage: Gibt es $f(g^a, g^b) = g^{ab}$?

Sicherheit

Übertragen wird nur g^a, g^b.

Frage: Gibt es $f(g^a, g^b) = g^{ab}$?

$A = g^a$
einfach, $O(log_2 a)$

$a = \log_g A$
Schwer, *Diskreter Logarithmus*

Antwort: Klar gibts f, es geht nur nicht effizient :)

Aufgabe

- Primzahl p aussuchen
- Generator g für $\mathbb{Z}_p^*$ finden
- Halbschlüssel g^a berechnen und austauschen
- Schlüssel k berechnen und freuen
- Optional: Diskreten Logarithmus berechnen und Fields-Medaille bekommen

- `Zmod`
- `gcd` $\gcd(g, p-1) = 1$
- `random_prime(n)`, `G.random_element()`

Angriffe

(Wo)Man-in-the-middle Alice kann $A - B$ nicht von $A - M - B$ unterscheiden.
Abhilfe z.B. Signaturaustausch über (g^a, g^b)

ungeschickter Zufall Wenn a oder b geraten werden kann...

unglückliche Primzahl $\log_g$ kann leichter gefunden werden, wenn $p - 1$ nur kleine Teiler hat.

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

RSA

- 1977 veröffentlicht von Ron **R**ivest, Adi **S**hamir und Leonard **A**dleman
- eines der ersten praktikablen asymmetrischen Kryptosysteme

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Textbook-RSA

Öffentliche Parameter:

- $n = pq$ mit p, q prim
- $e \in \mathbb{Z}^*_{\varphi(n)}$

Private Parameter:

- $d = e^{-1} \mod \varphi(n)$

Textbook-RSA

Verschlüsselung:

$$y = x^e \mod n$$

Entschlüsselung:

$$y^d \mod n = x$$

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Textbook-RSA

$$
\begin{aligned}
& y^d \\
\equiv & (x^e)^d \\
\equiv & x^{ed} \\
\equiv & x^{1+k\varphi(n)} \\
\equiv & x^1 \cdot \underbrace{x^{k\varphi(n)}}_{\equiv 1} \\
\equiv & x \pmod{n}
\end{aligned}
$$

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Das RSA Problem

Gegeben (n, e, y) finde x

Mögliche Lösungen

1. diskrete Wurzel ermitteln
2. $\varphi(n)$ effizient berechnen
3. n faktorisieren
4. d ermitteln

- Probleme 2-4 sind beweisbar gleich schwer
- Problem 1 könnte leichter sein, aber keine effiziente Lösung bekannt

Angriff auf Textbook-RSA

- Chosen-Ciphertext:
  - Berechne $y' = y \cdot m^e \mod n$
  - Erhalte $x' = x \cdot m \mod n$
  - Berechne $x = x' \cdot m^{-1} \mod n$
- zu kleines x:
  - $x^e < n$
  - x lässt sich berechnen als $\sqrt[e]{x}$

Padding

- Anhängen von zusätzlichen Bytes um eine bestimmte Größe zu erhalten
- mit 0en
- mit der Länge des Klartextes

Aufgaben

Geheimtext: $y = 196679683910$
Öffentlicher Schlüssel: $e = ?$ $n = 258805871659$
Privater Schlüssel: $d = 179172562165$ $p = 677827$ $q = 381817$

1. Implementiere Textbook-RSA
2. Berechne den Klartext zum gegebenen Geheimtext
3. Berechne das zum gegebenen Schlüssel passende e
4. Zwei öffentliche Schlüssel haben die Werte
   $n_1 = 78081401837$ $n2 = 133885222687$

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

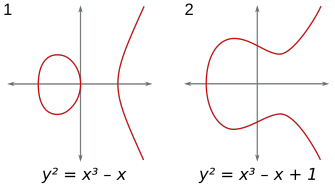
Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

Anfang

Kurve einer Funktion f: $\{(x,y) \mid y = f(x)\}$

Weierstraß-Gleichung:

$$E\colon y^2 + a_1 \cdot xy + a_3 \cdot y = x^3 + a_2 \cdot x^2 + a_4 \cdot x + a_6$$

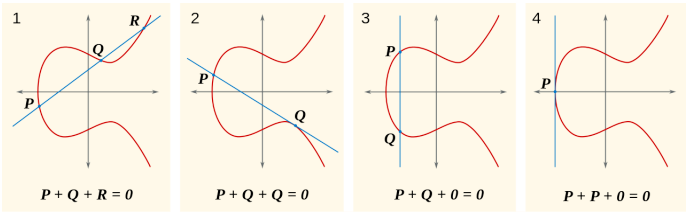


Mit Punkten rechnen

Eine Linie verbindet immer drei Punkte P, Q, R

$$P + Q + R = 0 \Leftrightarrow P + Q = -R$$

Tangente, senkrechte Linien? Spezialpunkt $\mathcal{O}$ („unendlich“)



CC-BY-SA-3.0, SuperManu, Wikimedia

Mit Punkten rechnen

Punkte auf $E \cup \{\mathcal{O}\}$ sind eine Gruppe.

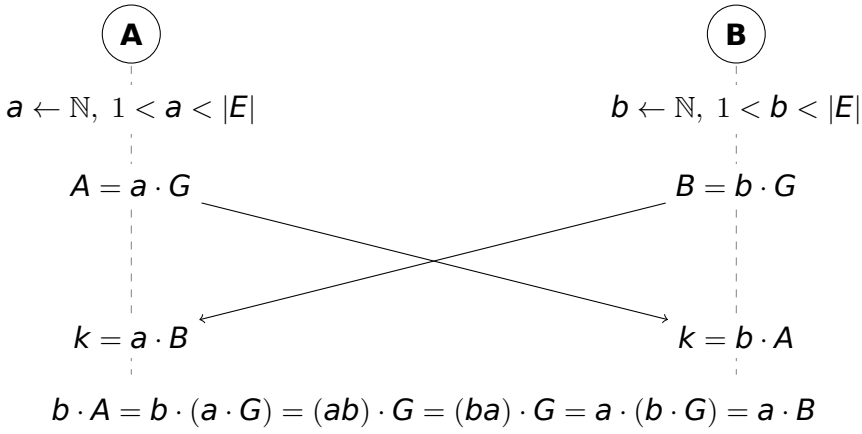
- P $P + Q \in E \cup \{\mathcal{O}\}$
- A $(P + Q) + R = P + (Q + R)$
- N $\mathcal{O}$
- I $\forall P \exists -P \colon P + (-P) = \mathcal{O}$
- C $P + Q = Q + P$

Operationen auf $E \cup \{\mathcal{O}\}$

- Addition $P + Q$
- Negation $-P$
- Punktexponentiation $n \cdot P = \underbrace{P + P + \cdots + P}_{n \text{ Mal}}$

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

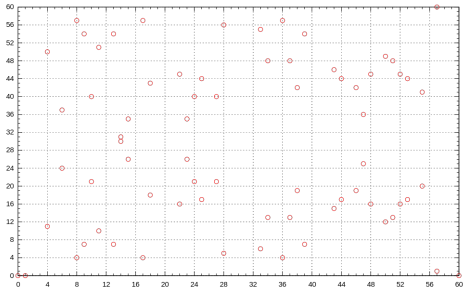
Verfahren


A
B
$a \leftarrow \mathbb{N},\ 1 < a < |E|$
$b \leftarrow \mathbb{N},\ 1 < b < |E|$
$A = a \cdot G$
$B = b \cdot G$
$k = a \cdot B$
$k = b \cdot A$
$b \cdot A = b \cdot (a \cdot G) = (ab) \cdot G = (ba) \cdot G = a \cdot (b \cdot G) = a \cdot B$

Elliptische Kurven über endliche Körper

Was für x und y erfüllen die Weierstraß-Gleichung?
Aus welcher Welt kommen a_1, a_2, a_3, a_4, a_6 ?

Antwort: Jeder *Ring* ist erlaubt.
Beliebt sind *Primkörper* $\mathbb{F}_p$



Curve25519

- Koordinaten sind 32 Byte lang, fast jeder Wert erlaubt
- Primkörper $\mathbb{F}_p$ für $p = 2^{255} - 19$
- $E : y^2 = x^3 + 486662 \cdot x^2 + x$
- $g \in E, X_g = 9$

Byte	Bit	Wert
0	0, 1, 2	0
31	7	0
31	6	1

Curve25519 als Funktion

```
def Curve25519(n, p):
    # E ist die Kurve25519
    P = E.lift_x(p)
    Q = n*p
    return Q[0] # X-Koordinate
```

Siomon <schneid@cs.uni-bonn.de>, Florob <florob@babelmonkeys.de>

EllipticCurves in Sage

Primkörper `F = FiniteField(2**255 - 19)`

Elliptische Kurve `E = EllipticCurve(F, [0, 486662, 0, 1, 0])`

Punkt auf E `g = E.lift_x(9)`

Addition `p+q`

Negation `-p`, `p-q`

Exponentiation `3*p`

Aufgabe

- Primkörper F_p für $p = 2^{255} - 19$ erzeugen
- Elliptische Kurve für $y^2 == x^3 + 486662 \cdot x^2 + x$ erzeugen
- Generator G mit $X_G = 9$ finden
- Halbschlüssel $a \cdot G$ berechnen und austauschen
- Schlüssel k berechnen und freuen

- `FiniteField`, `EllipticCurve`
- `EllipticCurve.lift_x`