

Kryptographie und Komplexität



Einheit 5.4



Endliche Körper und Elliptische Kurven

1. Endliche Körper
2. Elliptische Kurven über $\mathbb{R}$
3. Elliptische Kurven über $\mathbb{Z}_p$

SCHWÄCHEN VON ELGAMAL VERFAHREN ÜBER $\mathbb{Z}_p$

Nicht wesentlich besser als RSA

SCHWÄCHEN VON ELGAMAL VERFAHREN ÜBER $\mathbb{Z}_p$

Nicht wesentlich besser als RSA

- **Schlüssel müssen sehr groß werden**

- Faktorisierungsalgorithmen sind auf diskrete Logarithmen übertragbar
- Schlüssel bis 1024 Bit sind heutzutage angreifbar
- Wachsende Blockgröße macht **Verschlüsselung ineffizient**
 - Zeit für Verschlüsselung langer Nachrichten wächst linear
 - Implementierung auf SmartCards nur mit Coprozessor möglich

SCHWÄCHEN VON ELGAMAL VERFAHREN ÜBER $\mathbb{Z}_p$

Nicht wesentlich besser als RSA

- **Schlüssel müssen sehr groß werden**
 - Faktorisierungsalgorithmen sind auf diskrete Logarithmen übertragbar
 - Schlüssel bis 1024 Bit sind heutzutage angreifbar
 - Wachsende Blockgröße macht **Verschlüsselung ineffizient**
 - Zeit für Verschlüsselung langer Nachrichten wächst linear
 - Implementierung auf SmartCards nur mit Coprozessor möglich
- **Die besten Angriffe basieren auf Zahlen**
 - Einfache Algorithmen wie Shanks, Pollard ρ und Pohlig-Hellman sind gleich gut für alle Gruppen geeignet
 - Die Index-Calculus Methode und das Zahlkörpersieb benötigen schnelle arithmetische Operationen, um effizient sein zu können

SCHWÄCHEN VON ELGAMAL VERFAHREN ÜBER $\mathbb{Z}_p$

Nicht wesentlich besser als RSA

- **Schlüssel müssen sehr groß werden**
 - Faktorisierungsalgorithmen sind auf diskrete Logarithmen übertragbar
 - Schlüssel bis 1024 Bit sind heutzutage angreifbar
 - Wachsende Blockgröße macht **Verschlüsselung ineffizient**
 - Zeit für Verschlüsselung langer Nachrichten wächst linear
 - Implementierung auf SmartCards nur mit Coprozessor möglich
- **Die besten Angriffe basieren auf Zahlen**
 - Einfache Algorithmen wie Shanks, Pollard ρ und Pohlig-Hellman sind gleich gut für alle Gruppen geeignet
 - Die Index-Calculus Methode und das Zahlkörpersieb benötigen schnelle arithmetische Operationen, um effizient sein zu können
- **Verwende andere Gruppen als Basis für ElGamal**
 - $GF(p^n)$: Endliche Körper mit p^n Elementen
 - $E(p, a, b)$: Punkte einer elliptischen Kurve über $GF(p)$

Körper der Polynome mit Koeffizienten aus $\mathbb{Z}_p$

Körper der Polynome mit Koeffizienten aus $\mathbb{Z}_p$

- Für jede Primzahl p ist $(\mathbb{Z}_p, +_p, \cdot_p)$ ein Körper
 - $(\mathbb{Z}_p, +_p)$ ist abelsche Gruppe der Ordnung p
 - $(\mathbb{Z}_p^*, \cdot_p)$ ist abelsche Gruppe der Ordnung $p-1$
 - Das Distributivgesetz gilt für $+_p$ und $\cdot_p$

Körper der Polynome mit Koeffizienten aus $\mathbb{Z}_p$

- Für jede Primzahl p ist $(\mathbb{Z}_p, +_p, \cdot_p)$ ein Körper
 - $(\mathbb{Z}_p, +_p)$ ist abelsche Gruppe der Ordnung p
 - $(\mathbb{Z}_p^*, \cdot_p)$ ist abelsche Gruppe der Ordnung $p-1$
 - Das Distributivgesetz gilt für $+_p$ und $\cdot_p$
- **$K[x]$: Polynome über Körper K in Variable x**
 - Ausdrücke der Form $f(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$
mit Koeffizienten a_i und Belegungen der Variablen x aus K
 - n ist der **Grad von f** ($n = \text{deg } f$)
 - **Monome** sind Polynome, für die $a_i = 0$ für alle $i \neq n$ gilt
 - Eine **Nullstelle** von f ist ein Element $r \in K$ mit $f(r) = 0$
z.B. hat $f(x) = x^2 + 1$ über $\mathbb{Q}$ keine Nullstelle, über $\mathbb{Z}_2$ die Nullstelle 1

● Addition und Multiplikation in $K[x]$

– Sei $f(x) = \sum_{i=0}^m a_i x^i$ und $g(x) = \sum_{i=0}^n b_i x^i$ (o.B.d.A. $n \geq m$)

• $(f+g)(x) = \sum_{i=0}^n (a_i + b_i) x^i$ $\mathcal{O}(n)$ Additionen

• $(f \cdot g)(x) = \sum_{i=0}^{n+m} c_i x^i$ mit $c_k = \sum_{i=0}^k a_i b_{k-i}$ $\mathcal{O}(n \cdot m)$ Add./Mult.

– z.B. ist $(x^2 + 2x + 1) \cdot (x^3 - 2x^2 + 2) = x^5 - 3x^3 + 4x + 2$

● Addition und Multiplikation in $K[x]$

– Sei $f(x) = \sum_{i=0}^m a_i x^i$ und $g(x) = \sum_{i=0}^n b_i x^i$ (o.B.d.A. $n \geq m$)

• $(f+g)(x) = \sum_{i=0}^n (a_i + b_i) x^i$ $\mathcal{O}(n)$ Additionen

• $(f \cdot g)(x) = \sum_{i=0}^{n+m} c_i x^i$ mit $c_k = \sum_{i=0}^k a_i b_{k-i}$ $\mathcal{O}(n \cdot m)$ Add./Mult.

– z.B. ist $(x^2 + 2x + 1) \cdot (x^3 - 2x^2 + 2) = x^5 - 3x^3 + 4x + 2$

● $(K[x], +, \cdot)$ ist ein Ring mit Division

– $(K[x], +)$ ist eine abelsche Gruppe

– $(K[x], \cdot)$ ist nullteilerfreie abelsche Halbgruppe mit Einselement 1

– Das Distributivgesetz gilt für $+$ und $\cdot$

● Addition und Multiplikation in $K[x]$

- Sei $f(x) = \sum_{i=0}^m a_i x^i$ und $g(x) = \sum_{i=0}^n b_i x^i$ (o.B.d.A. $n \geq m$)
 - $(f+g)(x) = \sum_{i=0}^n (a_i + b_i) x^i$ $\mathcal{O}(n)$ Additionen
 - $(f \cdot g)(x) = \sum_{i=0}^{n+m} c_i x^i$ mit $c_k = \sum_{i=0}^k a_i b_{k-i}$ $\mathcal{O}(n \cdot m)$ Add./Mult.
- z.B. ist $(x^2 + 2x + 1) \cdot (x^3 - 2x^2 + 2) = x^5 - 3x^3 + 4x + 2$

● $(K[x], +, \cdot)$ ist ein Ring mit Division

- $(K[x], +)$ ist eine abelsche Gruppe
- $(K[x], \cdot)$ ist nullteilerfreie abelsche Halbgruppe mit Einselement 1
- Das Distributivgesetz gilt für $+$ und $\cdot$
- Division: Für $f, g \in K[x]$ gibt es eindeutige Polynome $q, r \in K[x]$
mit $f = q \cdot g + r$ und $r=0$ oder $\deg r < \deg g$
- Beweis: Konstruktion durch schriftliche Division $\mathcal{O}(m \cdot (n-m))$ Ops.
- Bezeichnung: $q = \lfloor f/g \rfloor$, $r = f \bmod g$
- z.B. $\lfloor x^3 - 2x^2 + 2 / x^2 + 2 \rfloor = x - 2$, $x^3 - 2x^2 + 2 \bmod x^2 + 2 = -2x + 6$

MATHEMATIK: POLYNOMKÖRPER

- **Konstruktion verallgemeinert $\mathbb{Z}_p$**
 - Restklassen modulo eines Ringelements $f \in K[x]$ (anstelle von $p \in \mathbb{Z}$)
 - Ringelement f muß **irreduzibel** sein (anstelle von “ p Primzahl”)

● Konstruktion verallgemeinert $\mathbb{Z}_p$

- Restklassen modulo eines Ringelements $f \in K[x]$ (anstelle von $p \in \mathbb{Z}$)
- Ringelement f muß **irreduzibel** sein (anstelle von “ p Primzahl”)
d.h. f darf nicht durch ein g mit $\deg f > \deg g \geq 1$ teilbar sein
- z.B. ist $f_1(x) = x^3+1$ reduzibel in $\mathbb{Z}_2[x]$, da $f_1(x) = (x+1)(x^2+x+1)$
- $h(x) = x^2+x+1$ ist irreduzibel in $\mathbb{Z}_2[x]$, da $h \equiv 1 \in \mathbb{Z}_2$, aber für jeden echten Teiler $g(x) = x+a$ der Wert a Nullstelle von h in $\mathbb{Z}_2$ wäre

MATHEMATIK: POLYNOMKÖRPER

● Konstruktion verallgemeinert $\mathbb{Z}_p$

- Restklassen modulo eines Ringelements $f \in K[x]$ (anstelle von $p \in \mathbb{Z}$)
- Ringelement f muß **irreduzibel** sein (anstelle von “ p Primzahl”)
d.h. f darf nicht durch ein g mit $\deg f > \deg g \geq 1$ teilbar sein
- z.B. ist $f_1(x) = x^3+1$ reduzibel in $\mathbb{Z}_2[x]$, da $f_1(x) = (x+1)(x^2+x+1)$
- $h(x) = x^2+x+1$ ist irreduzibel in $\mathbb{Z}_2[x]$, da $h \equiv 1 \in \mathbb{Z}_2$, aber für jeden echten Teiler $g(x) = x+a$ der Wert a Nullstelle von h in $\mathbb{Z}_2$ wäre

● $(K[x]/f, +, \cdot)$ ist Körper mit $|K|^{\deg f}$ Elementen

- $g \equiv h \bmod f$ falls $f \mid g-h$
- $[g]_f = g + f \cdot K[x] := \{h \mid h \equiv g \bmod f\}$ ist die Restklasse von g modulo f
- $K[x]/f$ ist die Menge aller Restklassen modulo f

● Konstruktion verallgemeinert $\mathbb{Z}_p$

- Restklassen modulo eines Ringelements $f \in K[x]$ (anstelle von $p \in \mathbb{Z}$)
- Ringelement f muß **irreduzibel** sein (anstelle von “ p Primzahl”)
d.h. f darf nicht durch ein g mit $\deg f > \deg g \geq 1$ teilbar sein
- z.B. ist $f_1(x) = x^3+1$ reduzibel in $\mathbb{Z}_2[x]$, da $f_1(x) = (x+1)(x^2+x+1)$
- $h(x) = x^2+x+1$ ist irreduzibel in $\mathbb{Z}_2[x]$, da $h \equiv 1 \in \mathbb{Z}_2$, aber für jeden echten Teiler $g(x) = x+a$ der Wert a Nullstelle von h in $\mathbb{Z}_2$ wäre

● $(K[x]/f, +, \cdot)$ ist Körper mit $|K|^{\deg f}$ Elementen

- $g \equiv h \bmod f$ falls $f \mid g-h$
- $[g]_f = g + f \cdot K[x] := \{h \mid h \equiv g \bmod f\}$ ist die Restklasse von g modulo f
- $K[x]/f$ ist die Menge aller Restklassen modulo f
- $K[x]/f$ ist Körper, da zu jedem $0 \neq g \in K[x]/f$ ein Inverses mit dem erweiterten euklidischen Algorithmus konstruiert werden kann (Folie 15, §2.1)
- $|K[x]/f| = |K|^{\deg f}$, weil jedes $[g]_f$ ein h mit $\deg h < \deg f$ enthält und die Restklassen dieser Polynome paarweise verschieden sind

MATHEMATIK: POLYNOMKÖRPER DER GRÖSSE p^n

- **Eindeutige Konstruktion möglich**

- Wähle Körper K mit p Elementen, wobei p Primzahl (z.B. $(\mathbb{Z}_p, +_p, \cdot_p)$)
- Bestimme ein irreduzibles Polynom $f \in K[x]$ mit Grad n
- $GF(p^n) = K[x]/f$ ist bis auf Isomorphie eindeutig bestimmt
Insbesondere ist $GF(p)$ isomorph zu $(\mathbb{Z}_p, +_p, \cdot_p)$

MATHEMATIK: POLYNOMKÖRPER DER GRÖSSE p^n

● Eindeutige Konstruktion möglich

- Wähle Körper K mit p Elementen, wobei p Primzahl (z.B. $(\mathbb{Z}_p, +_p, \cdot_p)$)
- Bestimme ein irreduzibles Polynom $f \in K[x]$ mit Grad n
- $GF(p^n) = K[x]/f$ ist bis auf Isomorphie eindeutig bestimmt
Insbesondere ist $GF(p)$ isomorph zu $(\mathbb{Z}_p, +_p, \cdot_p)$
- Satz: Für jeden endlichen Körper K gibt es eine Primzahl p sodaß K isomorph zu $GF(p^n)$ für ein n ist

MATHEMATIK: POLYNOMKÖRPER DER GRÖSSE p^n

● Eindeutige Konstruktion möglich

- Wähle Körper K mit p Elementen, wobei p Primzahl (z.B. $(\mathbb{Z}_p, +_p, \cdot_p)$)
- Bestimme ein irreduzibles Polynom $f \in K[x]$ mit Grad n
- $GF(p^n) = K[x]/f$ ist bis auf Isomorphie eindeutig bestimmt
Insbesondere ist $GF(p)$ isomorph zu $(\mathbb{Z}_p, +_p, \cdot_p)$
- Satz: Für jeden endlichen Körper K gibt es eine Primzahl p sodaß
 K isomorph zu $GF(p^n)$ für ein n ist

● Endliche Körper sind zyklisch

- Satz: Ist $(K, +, \cdot)$ Körper und $q = |K|$ so ist $(K^*, \cdot)$ eine zyklische Gruppe der Ordnung $q-1$ mit $\varphi(q-1)$ Erzeugern
(Buchmann, Theorem 3.21.1)

MATHEMATIK: POLYNOMKÖRPER DER GRÖSSE p^n

● Eindeutige Konstruktion möglich

- Wähle Körper K mit p Elementen, wobei p Primzahl (z.B. $(\mathbb{Z}_p, +_p, \cdot_p)$)
- Bestimme ein irreduzibles Polynom $f \in K[x]$ mit Grad n
- $GF(p^n) = K[x]/f$ ist bis auf Isomorphie eindeutig bestimmt
Insbesondere ist $GF(p)$ isomorph zu $(\mathbb{Z}_p, +_p, \cdot_p)$
- Satz: Für jeden endlichen Körper K gibt es eine Primzahl p sodaß
 K isomorph zu $GF(p^n)$ für ein n ist

● Endliche Körper sind zyklisch

- Satz: Ist $(K, +, \cdot)$ Körper und $q = |K|$ so ist $(K^*, \cdot)$ eine zyklische Gruppe der Ordnung $q-1$ mit $\varphi(q-1)$ Erzeugern
(Buchmann, Theorem 3.21.1)
 - $(GF(p^n), \cdot)$ ist eine zyklische Gruppe der Ordnung p^n-1
 - Für ungerade Primzahlen ist p^n-1 gerade, aber für $p=2$ kann 2^n-1 eine Primzahl sein
- **Konstruktion zyklischer Gruppen mit Primzahlordnung**

BEISPIEL: KONSTRUKTION VON $GF(2^3)$

- **Es gibt 8 Polynome $f(x)=x^3+a_2x^2+a_1x+a_0$**
 - Polynome mit $a_0=0$ sind teilbar durch $g(x) = x$
 - $f_1(x) = x^3+1$ ist reduzibel, da $f_1(x) = (x+1)(x^2+x+1)$
 $f_2(x) = x^3+x^2+1$ und $f_3(x) = x^3+x+1$ sind irreduzibel ($f_2 \equiv f_3 \equiv 1$)
 $f_4(x) = x^3+x^2+x+1$ ist reduzibel, da $f_4(x) = (x+1)(x^2+1)$
 - f_2 und f_3 sind geeignet als Basis für die Konstruktion

BEISPIEL: KONSTRUKTION VON $GF(2^3)$

- **Es gibt 8 Polynome $f(x)=x^3+a_2x^2+a_1x+a_0$**
 - Polynome mit $a_0=0$ sind teilbar durch $g(x) = x$
 - $f_1(x) = x^3+1$ ist reduzibel, da $f_1(x) = (x+1)(x^2+x+1)$
 $f_2(x) = x^3+x^2+1$ und $f_3(x) = x^3+x+1$ sind irreduzibel ($f_2 \equiv f_3 \equiv 1$)
 $f_4(x) = x^3+x^2+x+1$ ist reduzibel, da $f_4(x) = (x+1)(x^2+1)$
 - f_2 und f_3 sind geeignet als Basis für die Konstruktion
- **Operationen auf Koeffizienten reichen aus**
 - $GF(2^3)$ enthält nur Polynome $g(x) = a_2x^2+a_1x+a_0$ mit $a_i \in \mathbb{Z}_2$
 - Additions- und Multiplikation sind als Bitblockoperationen darstellbar
z.B. $(x^2+1) \cdot (x^2+x+1) \bmod f_3 = x^4+x^3+x+1 \bmod f_3 = x^2+x$
entspricht $101 \cdot 111 \bmod 1011 = 11011 \bmod 1011 = 110$

·	001	010	011	100	101	110	111
001	001	010	011	100	101	110	111
010	010	100	110	011	001	111	101
011	011	110	101	111	100	001	010
100	100	011	111	110	010	101	001
101	101	001	100	010	111	011	110
110	110	111	001	101	011	010	100
111	111	101	010	001	110	100	011

als Zahl

·	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7
2	2	4	6	3	1	7	5
3	3	6	5	7	4	1	2
4	4	3	7	6	2	5	1
5	5	1	4	2	7	3	6
6	6	7	1	5	3	2	4
7	7	5	2	1	6	4	3

BEISPIEL: KONSTRUKTION VON $GF(2^3)$

- **Es gibt 8 Polynome $f(x)=x^3+a_2x^2+a_1x+a_0$**
 - Polynome mit $a_0=0$ sind teilbar durch $g(x) = x$
 - $f_1(x) = x^3+1$ ist reduzibel, da $f_1(x) = (x+1)(x^2+x+1)$
 $f_2(x) = x^3+x^2+1$ und $f_3(x) = x^3+x+1$ sind irreduzibel ($f_2 \equiv f_3 \equiv 1$)
 $f_4(x) = x^3+x^2+x+1$ ist reduzibel, da $f_4(x) = (x+1)(x^2+1)$
 - f_2 und f_3 sind geeignet als Basis für die Konstruktion

- **Operationen auf Koeffizienten reichen aus**

- $GF(2^3)$ enthält nur Polynome $g(x) = a_2x^2+a_1x+a_0$ mit $a_i \in \mathbb{Z}_2$
- Additions- und Multiplikation sind als Bitblockoperationen darstellbar
z.B. $(x^2+1) \cdot (x^2+x+1) \bmod f_3 = x^4+x^3+x+1 \bmod f_3 = x^2+x$
entspricht $101 \cdot 111 \bmod 1011 = 11011 \bmod 1011 = 110$

·	001	010	011	100	101	110	111
001	001	010	011	100	101	110	111
010	010	100	110	011	001	111	101
011	011	110	101	111	100	001	010
100	100	011	111	110	010	101	001
101	101	001	100	010	111	011	110
110	110	111	001	101	011	010	100
111	111	101	010	001	110	100	011

als Zahl

·	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7
2	2	4	6	3	1	7	5
3	3	6	5	7	4	1	2
4	4	3	7	6	2	5	1
5	5	1	4	2	7	3	6
6	6	7	1	5	3	2	4
7	7	5	2	1	6	4	3

- Liefert Körper mit Basis $\mathbb{Z}_8^*$ und Nichtstandard-Multiplikation

EIGNUNG ENDLICHER KÖRPER FÜR ELGAMAL VERFAHREN

- $GF(2^n)$ scheint am besten geeignet
 - Addition und Multiplikation sind effizient implementierbar
 - Ver-/Entschlüsselung nahezu genauso schnell wie in $\mathbb{Z}_p$

EIGNUNG ENDLICHER KÖRPER FÜR ELGAMAL VERFAHREN

- **$GF(2^n)$ scheint am besten geeignet**
 - Addition und Multiplikation sind effizient implementierbar
 - Ver-/Entschlüsselung nahezu genauso schnell wie in $\mathbb{Z}_p$
- **Sicherheit von $GF(p^n)$ wenig größer als bei $\mathbb{Z}_p$**
 - Wenn $2^n - 1$ Primzahl ist oder große Primfaktoren enthält, bleibt der Pohlig-Hellman Algorithmus erfolglos

EIGNUNG ENDLICHER KÖRPER FÜR ELGAMAL VERFAHREN

- **$GF(2^n)$ scheint am besten geeignet**
 - Addition und Multiplikation sind effizient implementierbar
 - Ver-/Entschlüsselung nahezu genauso schnell wie in $\mathbb{Z}_p$
- **Sicherheit von $GF(p^n)$ wenig größer als bei $\mathbb{Z}_p$**
 - Wenn $2^n - 1$ Primzahl ist oder große Primfaktoren enthält, bleibt der Pohlig-Hellman Algorithmus erfolglos
 - Index-Calculus Methode nach wie vor anwendbar

$$\mathcal{O}(2^{(1+o(1)) \cdot n^{1/2} \cdot \log n^{1/2}})$$

EIGNUNG ENDLICHER KÖRPER FÜR ELGAMAL VERFAHREN

- $GF(2^n)$ scheint am besten geeignet

- Addition und Multiplikation sind effizient implementierbar
- Ver-/Entschlüsselung nahezu genauso schnell wie in $\mathbb{Z}_p$

- Sicherheit von $GF(p^n)$ wenig größer als bei $\mathbb{Z}_p$

- Wenn $2^n - 1$ Primzahl ist oder große Primfaktoren enthält, bleibt der Pohlig-Hellman Algorithmus erfolglos
- Index-Calculus Methode nach wie vor anwendbar

$$\mathcal{O}(2^{(1+o(1)) \cdot n^{1/2} \cdot \log n^{1/2}})$$

- Für festes n , wachsendes p ist das Zahlkörpersieb anwendbar

$$\mathcal{O}(2^{1.92 \cdot n^{1/3} \cdot \log n^{2/3}})$$

- Für festes p , wachsendes n ist Zahlkörpersieb erweiterbar zu

Funktionenkörpersieb

$$\mathcal{O}(2^{1.92 \cdot n^{1/3} \cdot \log n^{2/3}})$$

- Bei simultaner Erhöhung verwende Index-Calculus Methode

EIGNUNG ENDLICHER KÖRPER FÜR ELGAMAL VERFAHREN

- **$GF(2^n)$ scheint am besten geeignet**

- Addition und Multiplikation sind effizient implementierbar
- Ver-/Entschlüsselung nahezu genauso schnell wie in $\mathbb{Z}_p$

- **Sicherheit von $GF(p^n)$ wenig größer als bei $\mathbb{Z}_p$**

- Wenn $2^n - 1$ Primzahl ist oder große Primfaktoren enthält, bleibt der Pohlig-Hellman Algorithmus erfolglos
- Index-Calculus Methode nach wie vor anwendbar

$$\mathcal{O}(2^{(1+o(1)) \cdot n^{1/2} \cdot \log n^{1/2}})$$

- Für festes n , wachsendes p ist das Zahlkörpersieb anwendbar

$$\mathcal{O}(2^{1.92 \cdot n^{1/3} \cdot \log n^{2/3}})$$

- Für festes p , wachsendes n ist Zahlkörpersieb erweiterbar zu

Funktionenkörpersieb

$$\mathcal{O}(2^{1.92 \cdot n^{1/3} \cdot \log n^{2/3}})$$

- Bei simultaner Erhöhung verwende Index-Calculus Methode

Einfache endliche Körper reichen nicht aus

- **Begriff der mathematischen Funktionentheorie**
 - Nutzen für Kryptographie entdeckt von N. Koblitz und V. Miller

- **Begriff der mathematischen Funktionentheorie**

- Nutzen für Kryptographie entdeckt von N. Koblitz und V. Miller

- **Grundlage für effiziente ElGamal Verfahren**

- Zyklische Gruppe mit relativ effizienten, aber hochgradig unstetigen Additions- und Multiplikationsoperationen
- Problem diskreter Logarithmen viel schwerer zu lösen als für $GF(p^n)$
160 Bit Schlüssel sind genauso sicher wie 1024 Bit Schlüssel über $\mathbb{Z}_p$
- Auch sehr effizient für Signaturverfahren

- **Begriff der mathematischen Funktionentheorie**

- Nutzen für Kryptographie entdeckt von N. Koblitz und V. Miller

- **Grundlage für effiziente ElGamal Verfahren**

- Zyklische Gruppe mit relativ effizienten, aber hochgradig unstetigen Additions- und Multiplikationsoperationen
- Problem diskreter Logarithmen viel schwerer zu lösen als für $GF(p^n)$
160 Bit Schlüssel sind genauso sicher wie 1024 Bit Schlüssel über $\mathbb{Z}_p$
- Auch sehr effizient für Signaturverfahren

- **“Komplizierte” Gruppenstruktur**

- Elemente sind Punkte einer elliptischen Kurve über einem Körper K
Motivation entstammt elliptischen Kurven über reellen Zahlen
- Kryptographische Verfahren verwenden Kurven über endlichen Körpern

- **Umkehrfunktion elliptischer Integrale**
 - Konzept für die Berechnung der Bogenlänge von Ellipsen

ELLIPTISCHE KURVEN ÜBER $\mathbb{R}$

- **Umkehrfunktion elliptischer Integrale**
 - Konzept für die Berechnung der Bogenlänge von Ellipsen
- **Algebraisch: Gleichung dritten Grades über $\mathbb{R}$**
 - Menge $E(a, b)$ der Punkte $(x, y) \in \mathbb{R} \times \mathbb{R}$ für die $y^2 = x^3 + a \cdot x + b$ gilt, zusammen mit einem speziellen Punkt des Unendlichen $\mathcal{O}$

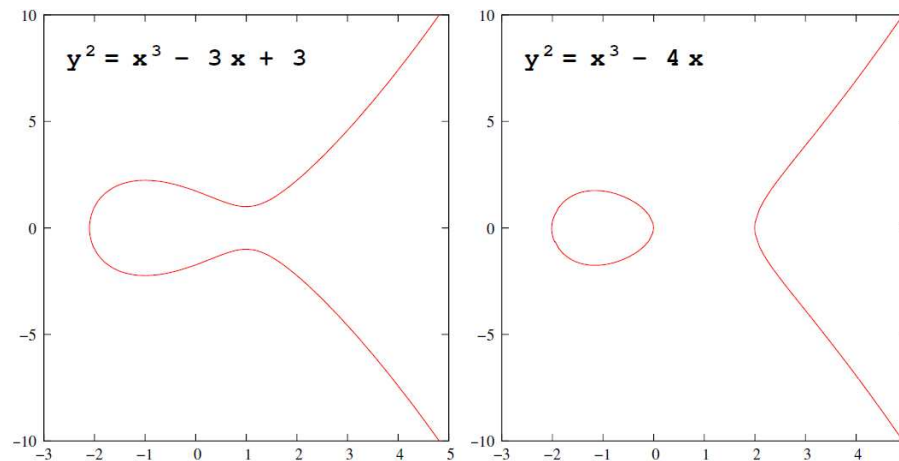
ELLIPTISCHE KURVEN ÜBER $\mathbb{R}$

- Umkehrfunktion elliptischer Integrale

- Konzept für die Berechnung der Bogenlänge von Ellipsen

- Algebraisch: Gleichung dritten Grades über $\mathbb{R}$

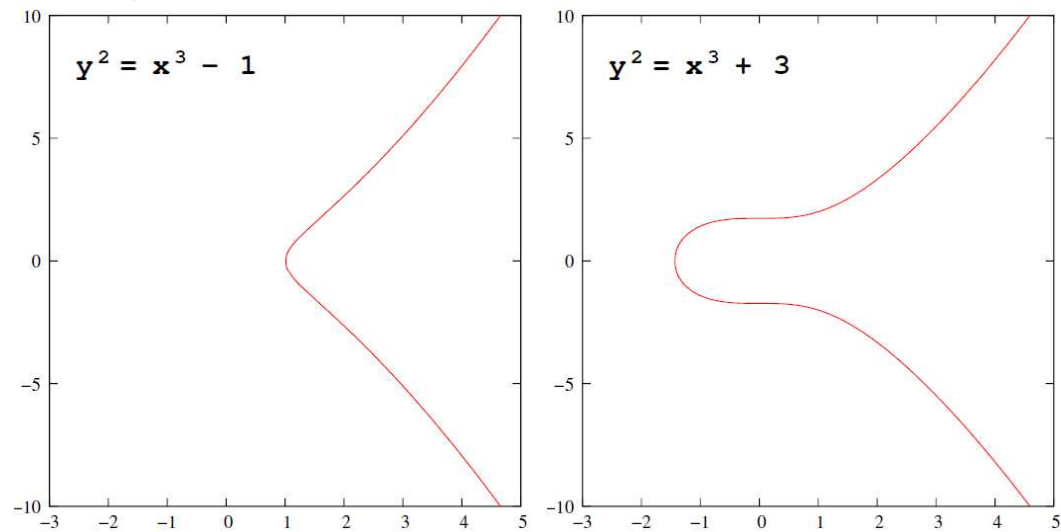
- Menge $E(a, b)$ der Punkte $(x, y) \in \mathbb{R} \times \mathbb{R}$ für die $y^2 = x^3 + a \cdot x + b$ gilt, zusammen mit einem speziellen Punkt des Unendlichen $\mathcal{O}$
- Nichtsinguläre elliptische Kurve, wenn $4a^3 + 27b^2 \neq 0$ ist



- Singuläre elliptische Kurven haben weniger als 3 (komplexe) Nullstellen

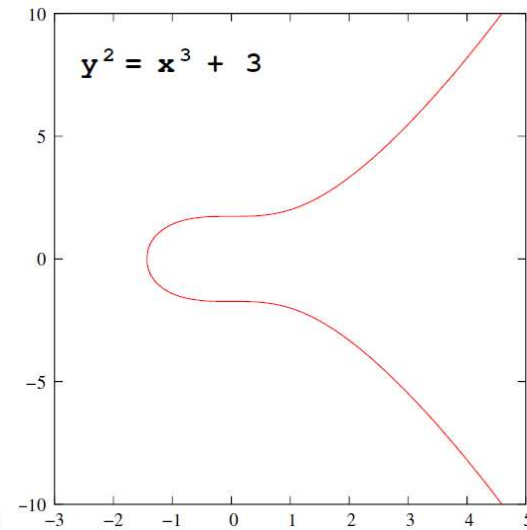
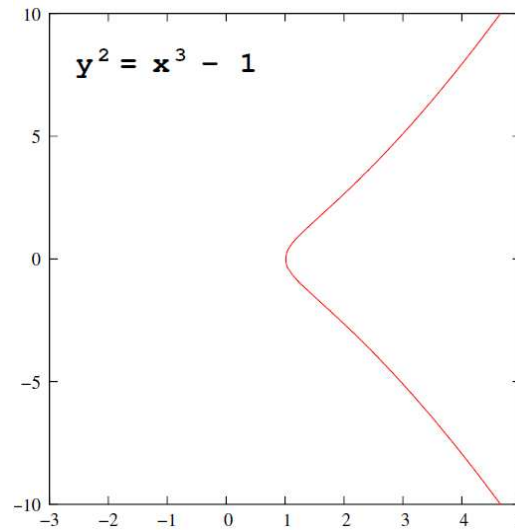
BEISPIELE ELLIPTISCHER KURVEN

- Nichtsinguläre Kurven

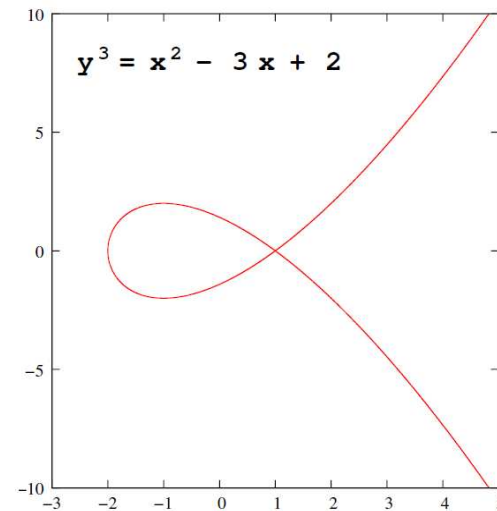
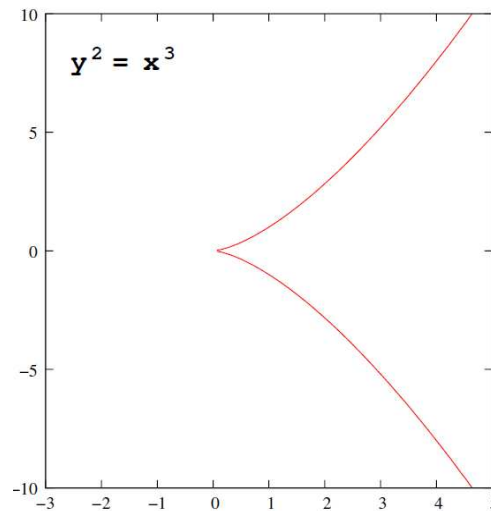


BEISPIELE ELLIPTISCHER KURVEN

● Nichtsinguläre Kurven



● Singuläre Kurven



DIE GRUPPE DER ELLIPTISCHEN KURVEN

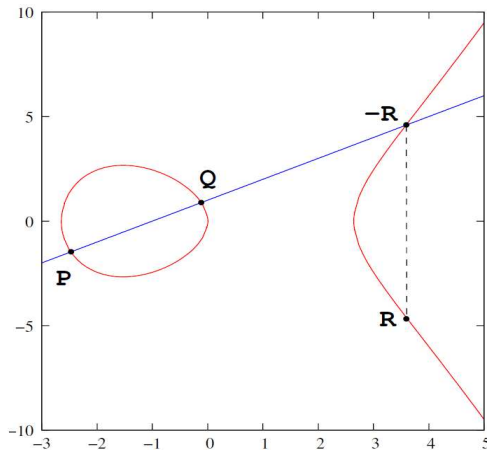
- **Verknüpfung ist Addition von Punkten**
 - Definiere Addition, skalare Multiplikation und Inverse von Punkten
 - Punkt des Unendlichen $\mathcal{O}$ ist neutrales Element
 - Beweise Gruppeneigenschaften

DIE GRUPPE DER ELLIPTISCHEN KURVEN

● Verknüpfung ist Addition von Punkten

- Definiere Addition, skalare Multiplikation und Inverse von Punkten
- Punkt des Unendlichen $\mathcal{O}$ ist neutrales Element
- Beweise Gruppeneigenschaften

● Addition von Punkten, geometrisch



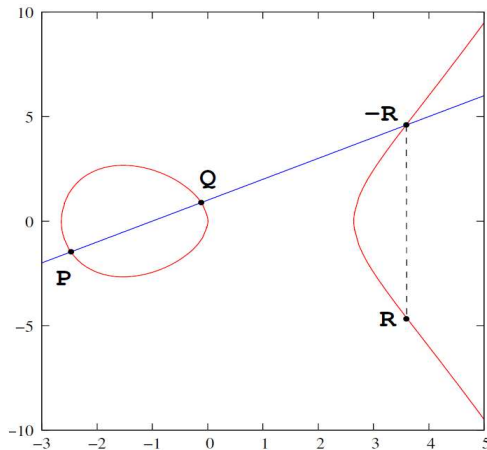
- Ziehe Gerade durch P und Q
- Bestimme Schnittpunkt $-R$ mit Kurve
- Invertierung der y -Koordinate liefert Ergebnis $R = P+Q$

DIE GRUPPE DER ELLIPTISCHEN KURVEN

● Verknüpfung ist Addition von Punkten

- Definiere Addition, skalare Multiplikation und Inverse von Punkten
- Punkt des Unendlichen $\mathcal{O}$ ist neutrales Element
- Beweise Gruppeneigenschaften

● Addition von Punkten, geometrisch



- Ziehe Gerade durch P und Q
- Bestimme Schnittpunkt $-R$ mit Kurve
- Invertierung der y -Koordinate liefert Ergebnis $R = P + Q$

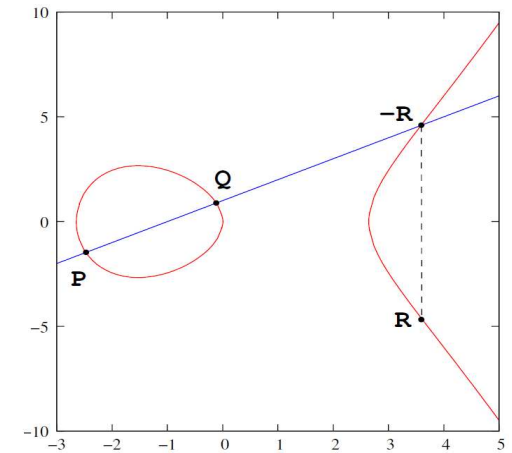
● Addition, algebraisch

- Beschreibe Konstruktion der Komponenten (x_R, y_R) von R
- Berücksichtige Sonderfälle $x_P = x_Q$ für $P \neq Q$ und $P = Q$

ADDITION AUF ELLIPTISCHEN KURVEN, ALGEBRAISCH

- Standardsituation $x_P \neq x_Q$

- Bestimme Gerade $g(x) = m \cdot x + k$ durch P und Q
Steigung $m = \frac{y_Q - y_P}{x_Q - x_P}$, Nullwert $k = y_P - m \cdot x_P$



ADDITION AUF ELLIPTISCHEN KURVEN, ALGEBRAISCH

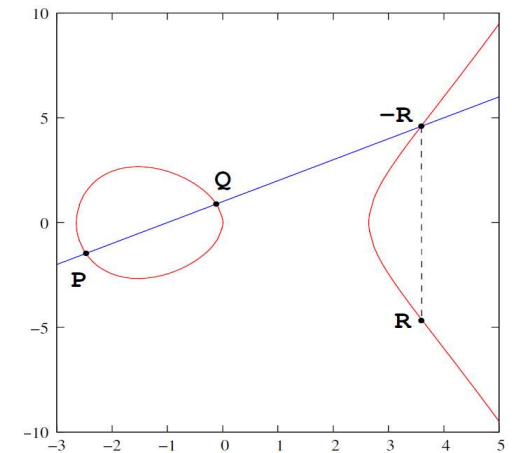
● Standardsituation $x_P \neq x_Q$

- Bestimme Gerade $g(x) = m \cdot x + k$ durch P und Q
Steigung $m = \frac{y_Q - y_P}{x_Q - x_P}$, Nullwert $k = y_P - m \cdot x_P$
- Bestimme Schnittpunkt $-R$ mit Kurve

Löse $(g(x))^2 = x^3 + a \cdot x + b$

also $x^3 - m^2 \cdot x^2 + (a - 2m \cdot k) \cdot x + b - k^2 = 0$

Ergibt nach Einsetzen $x_R = m^2 - x_P - x_Q$ (und $x_R = x_P$ bzw. $x_R = x_Q$)



ADDITION AUF ELLIPTISCHEN KURVEN, ALGEBRAISCH

● Standardsituation $x_P \neq x_Q$

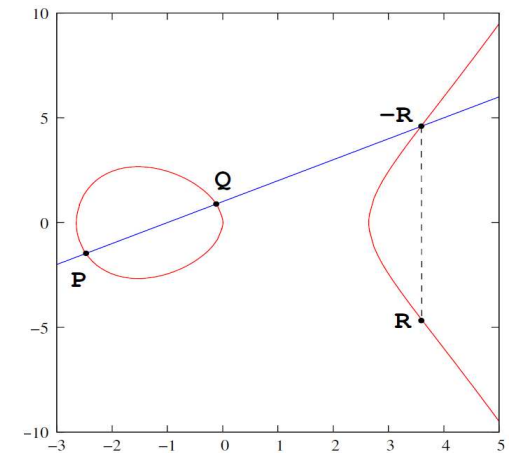
- Bestimme Gerade $g(x) = m \cdot x + k$ durch P und Q
Steigung $m = \frac{y_Q - y_P}{x_Q - x_P}$, Nullwert $k = y_P - m \cdot x_P$
- Bestimme Schnittpunkt $-R$ mit Kurve

Löse $(g(x))^2 = x^3 + a \cdot x + b$

also $x^3 - m^2 \cdot x^2 + (a - 2m \cdot k) \cdot x + b - k^2 = 0$

Ergibt nach Einsetzen $x_R = m^2 - x_P - x_Q$ (und $x_R = x_P$ bzw. $x_R = x_Q$)

- Koordinaten von R sind $(x_R, m \cdot (x_P - x_R) - y_P)$



ADDITION AUF ELLIPTISCHEN KURVEN, ALGEBRAISCH

● Standardsituation $x_P \neq x_Q$

- Bestimme Gerade $g(x) = m \cdot x + k$ durch P und Q
Steigung $m = \frac{y_Q - y_P}{x_Q - x_P}$, Nullwert $k = y_P - m \cdot x_P$

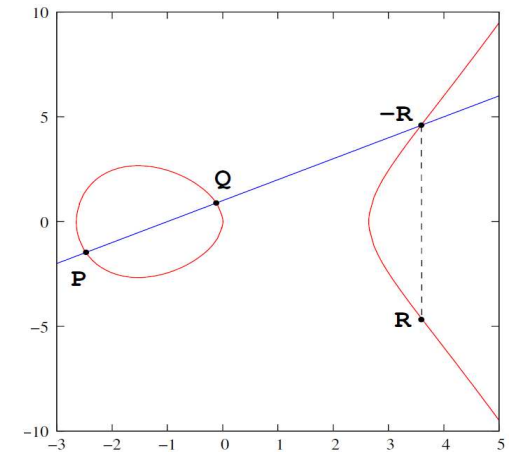
- Bestimme Schnittpunkt $-R$ mit Kurve

Löse $(g(x))^2 = x^3 + a \cdot x + b$

also $x^3 - m^2 \cdot x^2 + (a - 2m \cdot k) \cdot x + b - k^2 = 0$

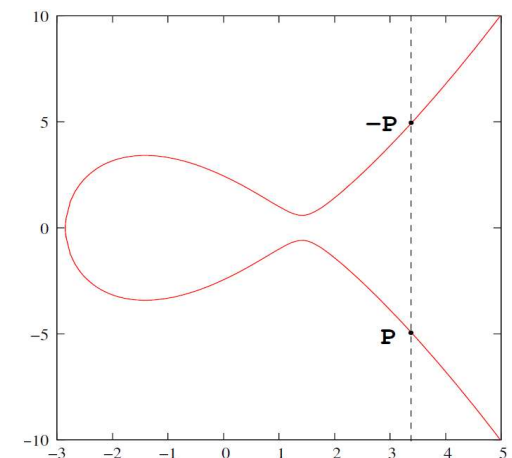
Ergibt nach Einsetzen $x_R = m^2 - x_P - x_Q$ (und $x_R = x_P$ bzw. $x_R = x_Q$)

- Koordinaten von R sind $(x_R, m \cdot (x_P - x_R) - y_P)$



● Sonderfall $x_P = x_Q$ und $y_P \neq y_Q$

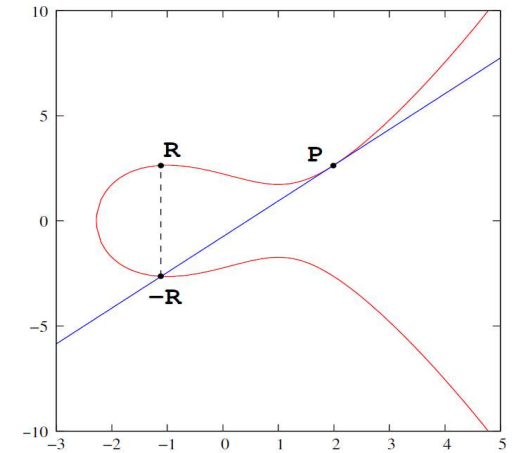
- Nach Definition elliptischer Kurven
muß $y_P = -y_Q$ sein
- Steigung der Geraden ist unendlich
- Definiere $P + Q = \mathcal{O}$



ADDITION AUF ELLIPTISCHEN KURVEN (II)

● Sonderfall $P=Q$ (Verdoppelung)

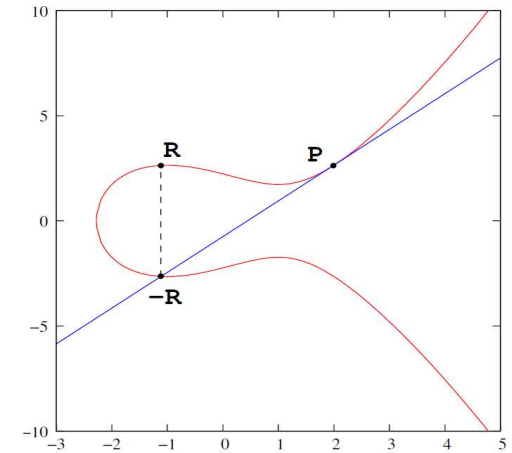
- Gerade zwischen P und Q ist Tangente in P
Löse Tangentengleichung $\frac{d}{dx}y^2 = \frac{d}{dx}x^3 + a \cdot x + b$
bzw. $2y \cdot \frac{dy}{dx} = 3x^2 + a$ für den Punkt P
- Steigung der Geraden in P ist $m = \frac{dy}{dx} == \frac{3x_P^2 + a}{2y_P}$
- Koordinaten (x_R, y_R) von R sind wie zuvor
 $x_R = m^2 - 2x_P$ und $y_R = m \cdot (x_P - x_R) - y_P$



ADDITION AUF ELLIPTISCHEN KURVEN (II)

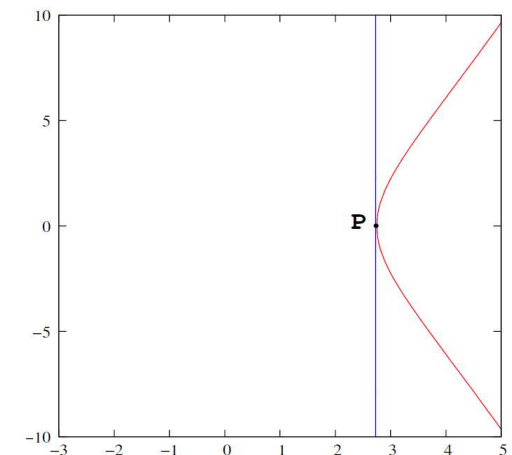
● Sonderfall $P=Q$ (Verdoppelung)

- Gerade zwischen P und Q ist Tangente in P
Löse Tangentengleichung $\frac{d}{dx}y^2 = \frac{d}{dx}x^3 + a \cdot x + b$
bzw. $2y \cdot \frac{dy}{dx} = 3x^2 + a$ für den Punkt P
- Steigung der Geraden in P ist $m = \frac{dy}{dx} == \frac{3x_P^2 + a}{2y_P}$
- Koordinaten (x_R, y_R) von R sind wie zuvor
 $x_R = m^2 - 2x_P$ und $y_R = m \cdot (x_P - x_R) - y_P$



● Verdoppelung an Randpunkten

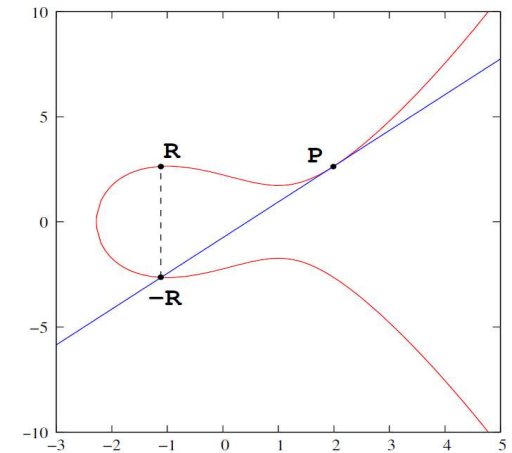
- y -Koordinate von P hat den Wert 0
- Tangente in P hat unendliche Steigung
- Definiere $P+P = \mathcal{O}$



ADDITION AUF ELLIPTISCHEN KURVEN (II)

● Sonderfall $P=Q$ (Verdoppelung)

- Gerade zwischen P und Q ist Tangente in P
Löse Tangentengleichung $\frac{d}{dx}y^2 = \frac{d}{dx}x^3 + a \cdot x + b$
bzw. $2y \cdot \frac{dy}{dx} = 3x^2 + a$ für den Punkt P
- Steigung der Geraden in P ist $m = \frac{dy}{dx} == \frac{3x_P^2 + a}{2y_P}$
- Koordinaten (x_R, y_R) von R sind wie zuvor
 $x_R = m^2 - 2x_P$ und $y_R = m \cdot (x_P - x_R) - y_P$

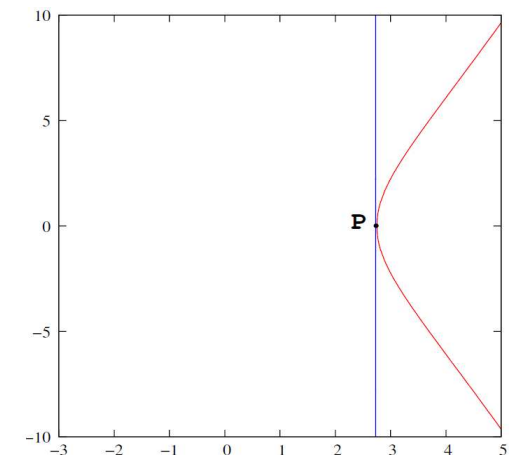


● Verdoppelung an Randpunkten

- y -Koordinate von P hat den Wert 0
- Tangente in P hat unendliche Steigung
- Definiere $P+P = \mathcal{O}$

● Skalare Multiplikation $n \cdot P$

- Iterierte Addition von P mit sich selbst



ELLIPTISCHE KURVEN BILDEN EINE ABELSCHES GRUPPE

- **Abgeschlossenheit von $E(a, b)$ unter Addition**
 - Per Konstruktion ist $P + Q \in E(a, b)$ für alle $P, Q \in E(a, b)$

ELLIPTISCHE KURVEN BILDEN EINE ABELSCHES GRUPPE

- **Abgeschlossenheit von $E(a, b)$ unter Addition**
 - Per Konstruktion ist $P + Q \in E(a, b)$ für alle $P, Q \in E(a, b)$
- **Assoziativität der Addition**
 - Ergibt sich durch Einsetzen der Gleichungen für Addition (mühsam)

ELLIPTISCHE KURVEN BILDEN EINE ABELSCHES GRUPPE

- **Abgeschlossenheit von $E(a, b)$ unter Addition**
 - Per Konstruktion ist $P + Q \in E(a, b)$ für alle $P, Q \in E(a, b)$
- **Assoziativität der Addition**
 - Ergibt sich durch Einsetzen der Gleichungen für Addition (mühsam)
- **Kommutativität der Addition**
 - Reihenfolge von P und Q ist irrelevant in geometrischer Konstruktion

ELLIPTISCHE KURVEN BILDEN EINE ABELSCHES GRUPPE

- **Abgeschlossenheit von $E(a, b)$ unter Addition**
 - Per Konstruktion ist $P + Q \in E(a, b)$ für alle $P, Q \in E(a, b)$
- **Assoziativität der Addition**
 - Ergibt sich durch Einsetzen der Gleichungen für Addition (mühsam)
- **Kommutativität der Addition**
 - Reihenfolge von P und Q ist irrelevant in geometrischer Konstruktion
- **$\mathcal{O}$ ist neutrales Element der Addition**
 - Gerade zwischen P und $\mathcal{O}$ liefert $-P$ als eindeutigen Schnittpunkt
(Zeigt warum $P+Q$ nicht einfach der Schnittpunkt der Geraden mit $E(a, b)$ sein darf)

ELLIPTISCHE KURVEN BILDEN EINE ABELSCHES GRUPPE

- **Abgeschlossenheit von $E(a, b)$ unter Addition**
 - Per Konstruktion ist $P + Q \in E(a, b)$ für alle $P, Q \in E(a, b)$
- **Assoziativität der Addition**
 - Ergibt sich durch Einsetzen der Gleichungen für Addition (mühsam)
- **Kommutativität der Addition**
 - Reihenfolge von P und Q ist irrelevant in geometrischer Konstruktion
- **$\mathcal{O}$ ist neutrales Element der Addition**
 - Gerade zwischen P und $\mathcal{O}$ liefert $-P$ als eindeutigen Schnittpunkt
(Zeigt warum $P+Q$ nicht einfach der Schnittpunkt der Geraden mit $E(a, b)$ sein darf)
- **Existenz inverser Elemente**
 - Per Konstruktion ist $-P = (x_P, -y_P)$ invers zu P

ELLIPTISCHE KURVEN ÜBER $\mathbb{Z}_p$

- **Kryptographie benötigt endliche Klartexträume**
 - Reale Computer und Smartcards sind endlich
 - Berechnungen müssen **schnell und exakt** durchgeführt werden können

ELLIPTISCHE KURVEN ÜBER $\mathbb{Z}_p$

- **Kryptographie benötigt endliche Klartexträume**
 - Reale Computer und Smartcards sind endlich
 - Berechnungen müssen **schnell und exakt** durchgeführt werden können
- **Reelle Zahlen sind ungeeignet**
 - $\mathbb{R}$ ist ein unendlicher (überabzählbar groß) Körper
 - Berechnungen auf “reellen” Zahlen im Computer sind unpräzise
 - Elliptische Kurven müssen **über endlichen Körpern** definiert werden

ELLIPTISCHE KURVEN ÜBER $\mathbb{Z}_p$

- **Kryptographie benötigt endliche Klartexträume**
 - Reale Computer und Smartcards sind endlich
 - Berechnungen müssen **schnell und exakt** durchgeführt werden können
- **Reelle Zahlen sind ungeeignet**
 - $\mathbb{R}$ ist ein unendlicher (überabzählbar großer) Körper
 - Berechnungen auf “reellen” Zahlen im Computer sind unpräzise
 - Elliptische Kurven müssen **über endlichen Körpern** definiert werden
- **Übertrage Definitionen auf $\mathbb{Z}_p$ und $GF(p^n)$**
 - **$E(p; a, b)$** ist die Menge der Punkte $(x, y) \in \mathbb{Z}_p \times \mathbb{Z}_p$ für die
$$y^2 \equiv x^3 + a \cdot x + b \pmod{p}$$
gilt, zusammen mit einem speziellen **Punkt des Unendlichen $\mathcal{O}$** , wobei $a, b \in \mathbb{Z}_p$ mit **$4a^3 + 27b^2 \pmod{p} \neq 0$**

BEISPIELE ELLIPTISCHER KURVEN ÜBER $\mathbb{Z}_p$

- **Elliptische Kurve** $y^2 \equiv x^3 + 7 \cdot x + 8 \pmod{23}$

29 Punkte erfüllen die Gleichung

(0, 10); (0, 13); (1, 4); (1, 19); (4, 10);

(4, 13); (6, 6); (6, 17); (7, 3); (7, 20);

(8, 1); (8, 22); (9, 8); (9, 15); (11, 6);

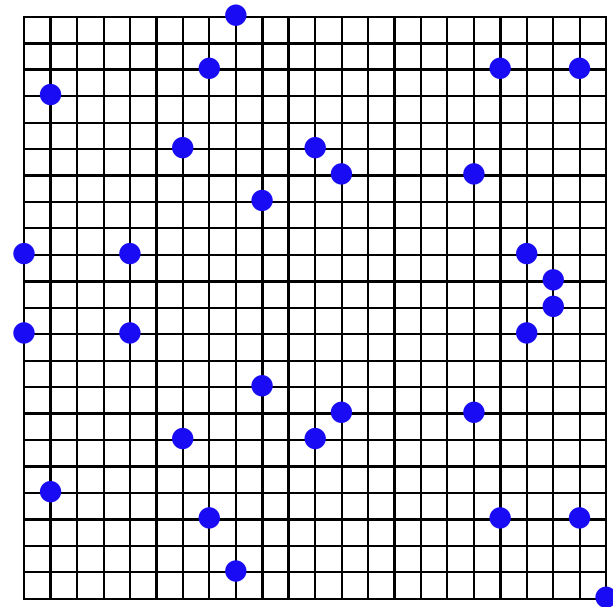
(11, 17); (12, 7); (12, 16); (17, 7); (17, 16);

(18, 3); (18, 20); (19, 10); (19, 13); (20, 11);

(20, 12); (21, 3); (21, 20); (22, 0)

Wie bei $\mathbb{R}$ maximal zwei Punkte

(x, y) und $(x, -y)$ je x -Wert



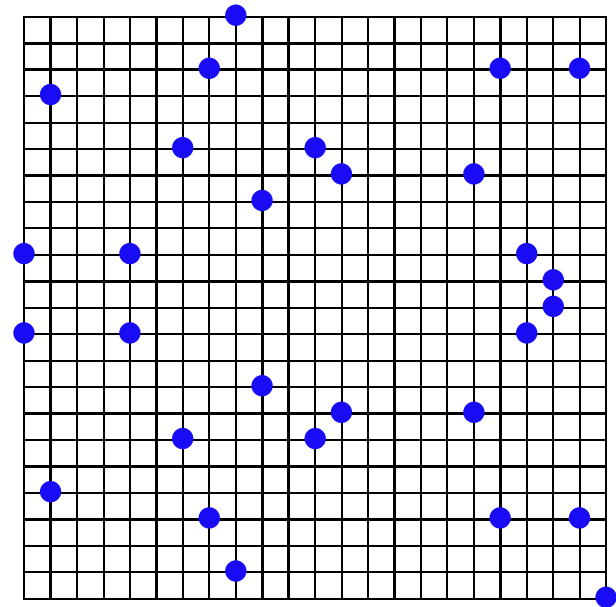
BEISPIELE ELLIPTISCHER KURVEN ÜBER $\mathbb{Z}_p$

- **Elliptische Kurve** $y^2 \equiv x^3 + 7 \cdot x + 8 \pmod{23}$

29 Punkte erfüllen die Gleichung

(0, 10); (0, 13); (1, 4); (1, 19); (4, 10);
(4, 13); (6, 6); (6, 17); (7, 3); (7, 20);
(8, 1); (8, 22); (9, 8); (9, 15); (11, 6);
(11, 17); (12, 7); (12, 16); (17, 7); (17, 16);
(18, 3); (18, 20); (19, 10); (19, 13); (20, 11);
(20, 12); (21, 3); (21, 20); (22, 0)

Wie bei $\mathbb{R}$ maximal zwei Punkte
 (x, y) und $(x, -y)$ je x -Wert

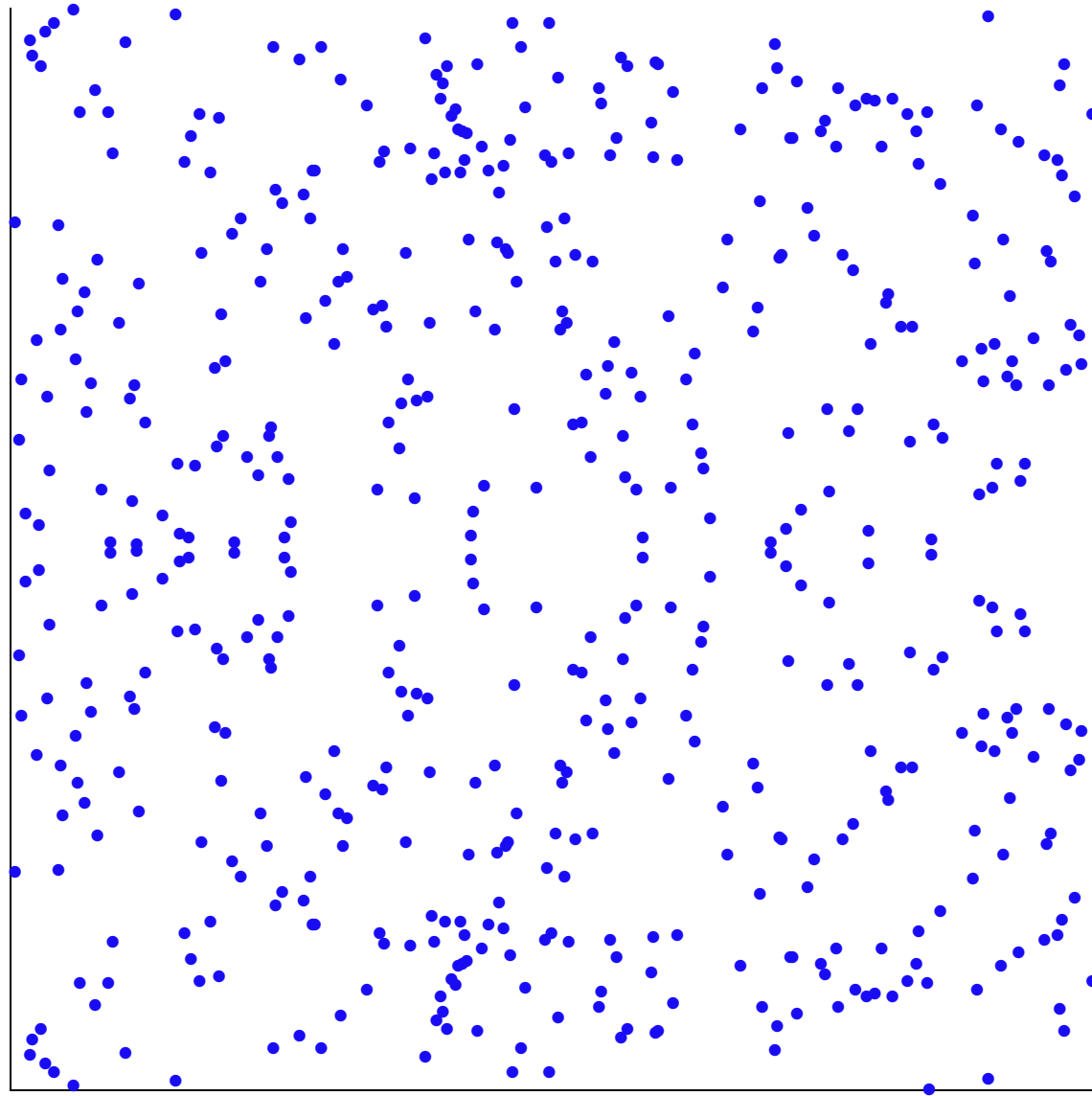


- **Elliptische Kurve** $y^2 \equiv x^3 + 2 \cdot x + 8 \pmod{23}$

– Nur 15 Punkte erfüllen die Gleichung

(0, 10); (0, 13); (3, 8); (3, 15); (6, 11); (6, 12); (10, 4); (10, 19);
(11, 2); (11, 21); (12, 9); (12, 14); (13, 0); (15, 3); (15, 20)

ELLIPTISCHE KURVE $y^2 \equiv x^3 + 2 \cdot x + 8 \pmod{499}$ (535 Punkte)



DIE GRUPPE ELLIPTISCHER KURVEN ÜBER $\mathbb{Z}_p$

- Ähnliche Konstruktion wie bei $\mathbb{R}$
 - Übertrage Definitionen von Addition, Skalarmultiplikation, Inverse
 - Konstruiere Algorithmen für schnelle Ausführung
 - Beweise Gruppeneigenschaften von $(E(p; a, b), +)$

DIE GRUPPE ELLIPTISCHER KURVEN ÜBER $\mathbb{Z}_p$

- Ähnliche Konstruktion wie bei $\mathbb{R}$

- Übertrage Definitionen von Addition, Skalarmultiplikation, Inverse
- Konstruiere Algorithmen für schnelle Ausführung
- Beweise Gruppeneigenschaften von $(E(p; a, b), +)$

- Addition von $P = (x_P, y_P)$ und $Q = (x_Q, y_Q)$

- Falls $x_P = x_Q$ und $y_P = -y_Q$, dann ist $P + Q = \mathcal{O}$

Genauso gilt $P + P = \mathcal{O}$ (Verdopplung) für $P = (x_P, 0)$

- Ansonsten gilt für die Koordinaten (x_R, y_R) von $R = P + Q$

$$x_R = m^2 - x_P - x_Q \quad \text{und} \quad y_R = m \cdot (x_P - x_R) - y_P$$

$$\text{wobei } m = \begin{cases} (y_Q - y_P)(x_Q - x_P)^{-1} & \text{falls } P \neq Q \\ (3x_P^2 + a)(2y_P)^{-1} & \text{sonst} \end{cases}$$

Rechenzeit: $\mathcal{O}(\|p\|^2)$

DIE GRUPPE ELLIPTISCHER KURVEN ÜBER $\mathbb{Z}_p$

- **Ähnliche Konstruktion wie bei $\mathbb{R}$**

- Übertrage Definitionen von Addition, Skalarmultiplikation, Inverse
- Konstruiere Algorithmen für schnelle Ausführung
- Beweise Gruppeneigenschaften von $(E(p; a, b), +)$

- **Addition von $P = (x_P, y_P)$ und $Q = (x_Q, y_Q)$**

- Falls $x_P = x_Q$ und $y_P = -y_Q$, dann ist $P + Q = \mathcal{O}$

Genauso gilt $P + P = \mathcal{O}$ (Verdopplung) für $P = (x_P, 0)$

- Ansonsten gilt für die Koordinaten (x_R, y_R) von $R = P + Q$

$$x_R = m^2 - x_P - x_Q \quad \text{und} \quad y_R = m \cdot (x_P - x_R) - y_P$$

$$\text{wobei } m = \begin{cases} (y_Q - y_P)(x_Q - x_P)^{-1} & \text{falls } P \neq Q \\ (3x_P^2 + a)(2y_P)^{-1} & \text{sonst} \end{cases}$$

Rechenzeit: $\mathcal{O}(\|p\|^2)$

- **Gruppeneigenschaften gelten wie zuvor**

- Gleichungen nahezu identisch zu denen der elliptischen Kurven über $\mathbb{R}$

ADDITION UND ITERATION AUF $E(23; 2, 8)$

● Additionstabelle

	(0, 10)	(0, 13)	(3, 8)	(3, 15)	(6, 11)	(6, 12)	(10, 4)	(10, 19)	(11, 2)	(11, 21)	(12, 9)	(12, 14)	(13, 0)	(15, 3)	(15, 20)
(0, 10)	(3, 15)	$\mathcal{O}$	(0, 13)	(10, 4)	(10, 19)	(12, 9)	(6, 12)	(3, 8)	(15, 3)	(13, 0)	(15, 20)	(6, 11)	(11, 2)	(12, 14)	(11, 21)
(0, 13)	$\mathcal{O}$	(3, 8)	(10, 19)	(0, 10)	(12, 14)	(10, 4)	(3, 15)	(6, 11)	(13, 0)	(15, 20)	(6, 12)	(15, 3)	(11, 21)	(11, 2)	(12, 9)
(3, 8)	(0, 13)	(10, 19)	(6, 11)	$\mathcal{O}$	(15, 3)	(3, 15)	(0, 10)	(12, 14)	(11, 21)	(12, 9)	(10, 4)	(11, 2)	(15, 20)	(13, 0)	(6, 12)
(3, 15)	(10, 4)	(0, 10)	$\mathcal{O}$	(6, 12)	(3, 8)	(15, 20)	(12, 9)	(0, 13)	(12, 14)	(11, 2)	(11, 21)	(10, 19)	(15, 3)	(6, 11)	(13, 0)
(6, 11)	(10, 19)	(12, 14)	(15, 3)	(3, 8)	(13, 0)	$\mathcal{O}$	(0, 13)	(11, 2)	(12, 9)	(10, 4)	(0, 10)	(11, 21)	(6, 12)	(15, 20)	(3, 15)
(6, 12)	(12, 9)	(10, 4)	(3, 15)	(15, 20)	$\mathcal{O}$	(13, 0)	(11, 21)	(0, 10)	(10, 19)	(12, 14)	(11, 2)	(0, 13)	(6, 11)	(3, 8)	(15, 3)
(10, 4)	(6, 12)	(3, 15)	(0, 10)	(12, 9)	(0, 13)	(11, 21)	(15, 20)	$\mathcal{O}$	(6, 11)	(15, 3)	(13, 0)	(3, 8)	(12, 14)	(10, 19)	(11, 2)
(10, 19)	(3, 8)	(6, 11)	(12, 14)	(0, 13)	(11, 2)	(0, 10)	$\mathcal{O}$	(15, 3)	(15, 20)	(6, 12)	(3, 15)	(13, 0)	(12, 9)	(11, 21)	(10, 4)
(11, 2)	(15, 3)	(13, 0)	(11, 21)	(12, 14)	(12, 9)	(10, 19)	(6, 11)	(15, 20)	(3, 15)	$\mathcal{O}$	(3, 8)	(6, 12)	(0, 10)	(10, 4)	(0, 13)
(11, 21)	(13, 0)	(15, 20)	(12, 9)	(11, 2)	(10, 4)	(12, 14)	(15, 3)	(6, 12)	$\mathcal{O}$	(3, 8)	(6, 11)	(3, 15)	(0, 13)	(0, 10)	(10, 19)
(12, 9)	(15, 20)	(6, 12)	(10, 4)	(11, 21)	(0, 10)	(11, 2)	(13, 0)	(3, 15)	(3, 8)	(6, 11)	(15, 3)	$\mathcal{O}$	(10, 19)	(0, 13)	(12, 14)
(12, 14)	(6, 11)	(15, 3)	(11, 2)	(10, 19)	(11, 21)	(0, 13)	(3, 8)	(13, 0)	(6, 12)	(3, 15)	$\mathcal{O}$	(15, 20)	(10, 4)	(12, 9)	(0, 10)
(13, 0)	(11, 2)	(11, 21)	(15, 20)	(15, 3)	(6, 12)	(6, 11)	(12, 14)	(12, 9)	(0, 10)	(0, 13)	(10, 19)	(10, 4)	$\mathcal{O}$	(3, 15)	(3, 8)
(15, 3)	(12, 14)	(11, 2)	(13, 0)	(6, 11)	(15, 20)	(3, 8)	(10, 19)	(11, 21)	(10, 4)	(0, 10)	(0, 13)	(12, 9)	(3, 15)	(6, 12)	$\mathcal{O}$
(15, 20)	(11, 21)	(12, 9)	(6, 12)	(13, 0)	(3, 15)	(15, 3)	(11, 2)	(10, 4)	(0, 13)	(10, 19)	(12, 14)	(0, 10)	(3, 8)	$\mathcal{O}$	(6, 11)

ADDITION UND ITERATION AUF $E(23; 2, 8)$

● Additionstabelle

	(0, 10)	(0, 13)	(3, 8)	(3, 15)	(6, 11)	(6, 12)	(10, 4)	(10, 19)	(11, 2)	(11, 21)	(12, 9)	(12, 14)	(13, 0)	(15, 3)	(15, 20)
(0, 10)	(3, 15)	$\mathcal{O}$	(0, 13)	(10, 4)	(10, 19)	(12, 9)	(6, 12)	(3, 8)	(15, 3)	(13, 0)	(15, 20)	(6, 11)	(11, 2)	(12, 14)	(11, 21)
(0, 13)	$\mathcal{O}$	(3, 8)	(10, 19)	(0, 10)	(12, 14)	(10, 4)	(3, 15)	(6, 11)	(13, 0)	(15, 20)	(6, 12)	(15, 3)	(11, 21)	(11, 2)	(12, 9)
(3, 8)	(0, 13)	(10, 19)	(6, 11)	$\mathcal{O}$	(15, 3)	(3, 15)	(0, 10)	(12, 14)	(11, 21)	(12, 9)	(10, 4)	(11, 2)	(15, 20)	(13, 0)	(6, 12)
(3, 15)	(10, 4)	(0, 10)	$\mathcal{O}$	(6, 12)	(3, 8)	(15, 20)	(12, 9)	(0, 13)	(12, 14)	(11, 2)	(11, 21)	(10, 19)	(15, 3)	(6, 11)	(13, 0)
(6, 11)	(10, 19)	(12, 14)	(15, 3)	(3, 8)	(13, 0)	$\mathcal{O}$	(0, 13)	(11, 2)	(12, 9)	(10, 4)	(0, 10)	(11, 21)	(6, 12)	(15, 20)	(3, 15)
(6, 12)	(12, 9)	(10, 4)	(3, 15)	(15, 20)	$\mathcal{O}$	(13, 0)	(11, 21)	(0, 10)	(10, 19)	(12, 14)	(11, 2)	(0, 13)	(6, 11)	(3, 8)	(15, 3)
(10, 4)	(6, 12)	(3, 15)	(0, 10)	(12, 9)	(0, 13)	(11, 21)	(15, 20)	$\mathcal{O}$	(6, 11)	(15, 3)	(13, 0)	(3, 8)	(12, 14)	(10, 19)	(11, 2)
(10, 19)	(3, 8)	(6, 11)	(12, 14)	(0, 13)	(11, 2)	(0, 10)	$\mathcal{O}$	(15, 3)	(15, 20)	(6, 12)	(3, 15)	(13, 0)	(12, 9)	(11, 21)	(10, 4)
(11, 2)	(15, 3)	(13, 0)	(11, 21)	(12, 14)	(12, 9)	(10, 19)	(6, 11)	(15, 20)	(3, 15)	$\mathcal{O}$	(3, 8)	(6, 12)	(0, 10)	(10, 4)	(0, 13)
(11, 21)	(13, 0)	(15, 20)	(12, 9)	(11, 2)	(10, 4)	(12, 14)	(15, 3)	(6, 12)	$\mathcal{O}$	(3, 8)	(6, 11)	(3, 15)	(0, 13)	(0, 10)	(10, 19)
(12, 9)	(15, 20)	(6, 12)	(10, 4)	(11, 21)	(0, 10)	(11, 2)	(13, 0)	(3, 15)	(3, 8)	(6, 11)	(15, 3)	$\mathcal{O}$	(10, 19)	(0, 13)	(12, 14)
(12, 14)	(6, 11)	(15, 3)	(11, 2)	(10, 19)	(11, 21)	(0, 13)	(3, 8)	(13, 0)	(6, 12)	(3, 15)	$\mathcal{O}$	(15, 20)	(10, 4)	(12, 9)	(0, 10)
(13, 0)	(11, 2)	(11, 21)	(15, 20)	(15, 3)	(6, 12)	(6, 11)	(12, 14)	(12, 9)	(0, 10)	(0, 13)	(10, 19)	(10, 4)	$\mathcal{O}$	(3, 15)	(3, 8)
(15, 3)	(12, 14)	(11, 2)	(13, 0)	(6, 11)	(15, 20)	(3, 8)	(10, 19)	(11, 21)	(10, 4)	(0, 10)	(0, 13)	(12, 9)	(3, 15)	(6, 12)	$\mathcal{O}$
(15, 20)	(11, 21)	(12, 9)	(6, 12)	(13, 0)	(3, 15)	(15, 3)	(11, 2)	(10, 4)	(0, 13)	(10, 19)	(12, 14)	(0, 10)	(3, 8)	$\mathcal{O}$	(6, 11)

● Iterierte Addition von $P = (0, 10)$

- $P = (0, 10)$, $2P = (3, 15)$, $3P = (10, 4)$, $4P = (6, 12)$, $5P = (12, 9)$, $6P = (15, 20)$,
 $7P = (11, 21)$, $8P = (13, 0)$, $9P = (11, 2)$, $10P = (15, 3)$, $11P = (12, 14)$,
 $12P = (6, 11)$, $13P = (10, 19)$, $14P = (3, 8)$, $15P = (0, 13)$, $16P = \mathcal{O}$
- $E(23; 2, 8)$ ist zyklische Gruppe und P ist ein erzeugendes Element

STRUKTUR ELLIPTISCHER KURVEN ÜBER $\mathbb{Z}_p$

- **Wieviele Punkte hat eine elliptische Kurve?**

- Gruppenordnung ist Zahl der Lösungen von $y^2 = x^3 + a \cdot x + b$ in $\mathbb{Z}_p \times \mathbb{Z}_p$
- Satz von Hasse: **Für die Ordnung n der Gruppe $E(p; a, b)$ gilt**
$$p+1-2\sqrt{p} \leq n \leq p+1+2\sqrt{p}$$
 - Für $p=23$ liegt die Gruppenordnung von $E(p; a, b)$ zwischen 14 und 33
 - Für $p=499$ liegt die Gruppenordnung zwischen 456 und 544
- Genaue Gruppenordnung berechnet **Algorithms von Schoof** in $\mathcal{O}(\|p\|^8)$

STRUKTUR ELLIPTISCHER KURVEN ÜBER $\mathbb{Z}_p$

- **Wieviele Punkte hat eine elliptische Kurve?**

- Gruppenordnung ist Zahl der Lösungen von $y^2 = x^3 + a \cdot x + b$ in $\mathbb{Z}_p \times \mathbb{Z}_p$
- Satz von Hasse: **Für die Ordnung n der Gruppe $E(p; a, b)$ gilt**
$$p+1-2\sqrt{p} \leq n \leq p+1+2\sqrt{p}$$
 - Für $p=23$ liegt die Gruppenordnung von $E(p; a, b)$ zwischen 14 und 33
 - Für $p=499$ liegt die Gruppenordnung zwischen 456 und 544
- Genaue Gruppenordnung berechnet **Algorithms von Schoof** in $\mathcal{O}(\|p\|^8)$
- $E(p; a, b)$ ist **zyklisch**, wenn Ordnung Produkt verschiedener Primzahlen

STRUKTUR ELLIPTISCHER KURVEN ÜBER $\mathbb{Z}_p$

● Wieviele Punkte hat eine elliptische Kurve?

- Gruppenordnung ist Zahl der Lösungen von $y^2 = x^3 + a \cdot x + b$ in $\mathbb{Z}_p \times \mathbb{Z}_p$
- Satz von Hasse: **Für die Ordnung n der Gruppe $E(p; a, b)$ gilt**
$$p+1-2\sqrt{p} \leq n \leq p+1+2\sqrt{p}$$
 - Für $p=23$ liegt die Gruppenordnung von $E(p; a, b)$ zwischen 14 und 33
 - Für $p=499$ liegt die Gruppenordnung zwischen 456 und 544
- Genaue Gruppenordnung berechnet **Algorithms von Schoof** in $\mathcal{O}(\|p\|^8)$
- $E(p; a, b)$ ist **zyklisch**, wenn Ordnung Produkt verschiedener Primzahlen

● Welche Gruppenstruktur ist zu erwarten?

- Satz: **Für Primzahlen $p > 3$ gibt es $k, m \in \mathbb{N}$ mit $k|m$ und**
$$k|(p-1), \text{ so daß } (E(p; a, b), +) \text{ isomorph zu } \mathbb{Z}_m \times \mathbb{Z}_k \text{ ist}$$
- Die Zahlen k und m können aus p, a, b berechnet werden

STRUKTUR ELLIPTISCHER KURVEN ÜBER $\mathbb{Z}_p$

● Wieviele Punkte hat eine elliptische Kurve?

- Gruppenordnung ist Zahl der Lösungen von $y^2 = x^3 + a \cdot x + b$ in $\mathbb{Z}_p \times \mathbb{Z}_p$
- Satz von Hasse: **Für die Ordnung n der Gruppe $E(p; a, b)$ gilt**
$$p+1-2\sqrt{p} \leq n \leq p+1+2\sqrt{p}$$
 - Für $p=23$ liegt die Gruppenordnung von $E(p; a, b)$ zwischen 14 und 33
 - Für $p=499$ liegt die Gruppenordnung zwischen 456 und 544
- Genaue Gruppenordnung berechnet **Algorithms von Schoof** in $\mathcal{O}(\|p\|^8)$
- $E(p; a, b)$ ist **zyklisch**, wenn Ordnung Produkt verschiedener Primzahlen

● Welche Gruppenstruktur ist zu erwarten?

- Satz: **Für Primzahlen $p > 3$ gibt es $k, m \in \mathbb{N}$ mit $k|m$ und**
$$k|(p-1), \text{ so daß } (E(p; a, b), +) \text{ isomorph zu } \mathbb{Z}_m \times \mathbb{Z}_k \text{ ist}$$
- Die Zahlen k und m können aus p, a, b berechnet werden
- $E(p; a, b)$ ist **zyklisch**, wenn $k=1$ ist
- Ansonsten gibt es eine **zyklische Untergruppe** der Ordnung m

ELGAMAL SYSTEME MIT ELLIPTISCHEN KURVEN

Potenzierung wird skalare Multiplikation

Potenzierung wird skalare Multiplikation

- **Schlüsselerzeugung**

- Wähle eine zyklische elliptische Kurve $E = E(p; a, b)$ der Ordnung n und einen erzeugenden Punkt P

Potenzierung wird skalare Multiplikation

● Schlüsselerzeugung

- Wähle eine zyklische elliptische Kurve $E = E(p; a, b)$ der Ordnung n und einen erzeugenden Punkt P
- Wähle ein zufälliges $a \in \{0, \dots, n-1\}$ und berechne $A = a \cdot P$
- Lege E, P, A offen, halte a geheim

Potenzierung wird skalare Multiplikation

● Schlüsselerzeugung

- Wähle eine zyklische elliptische Kurve $E = E(p; a, b)$ der Ordnung n und einen erzeugenden Punkt P
- Wähle ein zufälliges $a \in \{0, \dots, n-1\}$ und berechne $A = a \cdot P$
- Lege E, P, A offen, halte a geheim

● Verschlüsselung

- Gesamtschlüssel ist $K := (E, P, a, A)$, wobei E, P, A öffentlich
- Textblöcke der Länge $\log_2 n/8$ werden auf Punkte von E abgebildet

Potenzierung wird skalare Multiplikation

● Schlüsselerzeugung

- Wähle eine zyklische elliptische Kurve $E = E(p; a, b)$ der Ordnung n und einen erzeugenden Punkt P
- Wähle ein zufälliges $a \in \{0, \dots, n-1\}$ und berechne $A = a \cdot P$
- Lege E, P, A offen, halte a geheim

● Verschlüsselung

- Gesamtschlüssel ist $K := (E, P, a, A)$, wobei E, P, A öffentlich
- Textblöcke der Länge $\log_2 n/8$ werden auf Punkte von E abgebildet
- Absender wählt zufälliges $b \in \{0, \dots, n-1\}$ und berechnet $B := b \cdot P$
- Absender verschlüsselt Punkt X zu $Y := X + b \cdot A$
- Erzeugter Schlüsseltext ist ein Punktepaar $e_K(X, b) = (B, Y)$

Potenzierung wird skalare Multiplikation

● Schlüsselerzeugung

- Wähle eine zyklische elliptische Kurve $E = E(p; a, b)$ der Ordnung n und einen erzeugenden Punkt P
- Wähle ein zufälliges $a \in \{0, \dots, n-1\}$ und berechne $A = a \cdot P$
- Lege E, P, A offen, halte a geheim

● Verschlüsselung

- Gesamtschlüssel ist $K := (E, P, a, A)$, wobei E, P, A öffentlich
- Textblöcke der Länge $\log_2 n/8$ werden auf Punkte von E abgebildet
- Absender wählt zufälliges $b \in \{0, \dots, n-1\}$ und berechnet $B := b \cdot P$
- Absender verschlüsselt Punkt X zu $Y := X + b \cdot A$
- Erzeugter Schlüsseltext ist ein Punktepaar $e_K(X, b) = (B, Y)$

● Entschlüsselung

- Empfänger berechnet $d_K(B, Y) = Y - (a \cdot B)$

DAS ELGAMAL VERFAHREN AM BEISPIEL

● Schlüsselerzeugung

- Alice wählt $E = E(23; 2, 8)$ (Ordnung 16)
und $P = (0, 10)$ als erzeugenden Punkt
- Alice wählt $a = 6$ und berechnet $A = 6 \cdot (0, 10) = (15, 20)$
- Alice veröffentlicht den Schlüssel $K = (E(23; 2, 8), (0, 10), (15, 20))$

DAS ELGAMAL VERFAHREN AM BEISPIEL

● Schlüsselerzeugung

- Alice wählt $E = E(23; 2, 8)$ (Ordnung 16)
und $P = (0, 10)$ als erzeugenden Punkt
- Alice wählt $a = 6$ und berechnet $A = 6 \cdot (0, 10) = (15, 20)$
- Alice veröffentlicht den Schlüssel $K = (E(23; 2, 8), (0, 10), (15, 20))$

● Verschlüsselung

- Bob wählt $b = 3$ und berechnet $B = 3 \cdot (0, 10) = (10, 4)$
und $b \cdot A = 3 \cdot (15, 20) = (3, 15)$
- Verschlüsselung von $X = (10, 19)$ ergibt $Y = (10, 19) + (3, 15) = (0, 13)$
- Bob versendet als Schlüsseltext $e_K(X, b) = (B, Y) = ((10, 4), (0, 13))$

DAS ELGAMAL VERFAHREN AM BEISPIEL

● Schlüsselerzeugung

- Alice wählt $E = E(23; 2, 8)$ (Ordnung 16)
und $P = (0, 10)$ als erzeugenden Punkt
- Alice wählt $a = 6$ und berechnet $A = 6 \cdot (0, 10) = (15, 20)$
- Alice veröffentlicht den Schlüssel $K = (E(23; 2, 8), (0, 10), (15, 20))$

● Verschlüsselung

- Bob wählt $b = 3$ und berechnet $B = 3 \cdot (0, 10) = (10, 4)$
und $b \cdot A = 3 \cdot (15, 20) = (3, 15)$
- Verschlüsselung von $X = (10, 19)$ ergibt $Y = (10, 19) + (3, 15) = (0, 13)$
- Bob versendet als Schlüsseltext $e_K(X, b) = (B, Y) = ((10, 4), (0, 13))$

● Entschlüsselung

- Alice berechnet $-(a \cdot B) = -6 \cdot (10, 4) = -(3, 15) = (3, 8)$
- Entschlüsselung von Y liefert $Y - (a \cdot B) = (0, 13) + (3, 8) = (10, 19)$

SCHNELLE SKALARMULTIPLIKATION AUF $E(p; a, b)$

- **EC-Verschlüsselung verwendet $\approx 2^{160}$ Punkte**
 - Signifikant weniger als RSA oder ElGamal Systeme über $\mathbb{Z}_q / GF(q^n)$
 - Dennoch wesentlich zu viele für einfache iterative Berechnung von $k \cdot P$
 - Rechenzeit muß in der Größenordnung von $\|p\|$ liegen

SCHNELLE SKALARMULTIPLIKATION AUF $E(p; a, b)$

- **EC-Verschlüsselung verwendet $\approx 2^{160}$ Punkte**
 - Signifikant weniger als RSA oder ElGamal Systeme über $\mathbb{Z}_q / GF(q^n)$
 - Dennoch wesentlich zu viele für einfache iterative Berechnung von $k \cdot P$
 - Rechenzeit muß in der Größenordnung von $\|p\|$ liegen
- **Variante der schnellen Potenzierung für $(\mathbb{Z}_n, \cdot_n)$**
 - Direkte Übertragung würde Quadrieren durch Verdopplung $P \mapsto 2 \cdot P$ und Multiplikation durch Addition von Punkten ersetzen
 - Optimierung nutzt, daß Invertierung auf $E(p; a, b)$ konstante Zeit benötigt (anstelle von $\mathcal{O}(\|n\|^2)$ für $(\mathbb{Z}_n, \cdot_n)$)

SCHNELLE SKALARMULTIPLIKATION AUF $E(p; a, b)$

- **EC-Verschlüsselung verwendet $\approx 2^{160}$ Punkte**

- Signifikant weniger als RSA oder ElGamal Systeme über $\mathbb{Z}_q / GF(q^n)$
- Dennoch wesentlich zu viele für einfache iterative Berechnung von $k \cdot P$
- Rechenzeit muß in der Größenordnung von $\|p\|$ liegen

- **Variante der schnellen Potenzierung für $(\mathbb{Z}_n, \cdot_n)$**

- Direkte Übertragung würde Quadrieren durch Verdopplung $P \mapsto 2 \cdot P$ und Multiplikation durch Addition von Punkten ersetzen
- Optimierung nutzt, daß Invertierung auf $E(p; a, b)$ konstante Zeit benötigt (anstelle von $\mathcal{O}(\|n\|^2)$ für $(\mathbb{Z}_n, \cdot_n)$)

- **Verwende Binärdarstellung mit Vorzeichen**

- Eine Zahl $k \in \mathbb{Z}_n$ wird dargestellt als $k = \sum_{i=0}^l a_i 2^i$ mit $a_i \in \{0, 1, -1\}$
- Bei **NAF-Darstellung** (non-adjacent form) ist von zwei aufeinanderfolgenden Koeffizienten a_i maximal einer 1 oder -1
- z.B.: Binärdarstellung von 311 ist $256+32+16+4+2+1 = 100110111$
NAF-Darstellung ist $256+64-8-1 = 10100-100-1$

OPTIMIERTE SCHNELLE SKALARMULTIPLIKATION

- **Iterierte Verdoppelung, Addition, Subtraktion**
 - Zur Berechnung von $k \cdot P$ erzeuge NAF Darstellung $\sum_{i=0}^l a_i 2^i$ von k
 - Verdoppele für jedes a_i , addiere/subtrahiere zusätzlich für $a_i=1/-1$

OPTIMIERTE SCHNELLE SKALARMULTIPLIKATION

- **Iterierte Verdoppelung, Addition, Subtraktion**

- Zur Berechnung von $k \cdot P$ erzeuge NAF Darstellung $\sum_{i=0}^l a_i 2^i$ von k
- Verdoppele für jedes a_i , addiere/subtrahiere zusätzlich für $a_i=1/-1$

- **Funktionale Implementierung**

```
let rec ecc_mult point a_list
= if a_list = [] then  $\mathcal{O}$ 
  else let a_i::rest = a_list
        and qpoint = ecc_mult (ecc_add point point) rest
        in
        if a_i=1 then ecc_add qpoint point
        else if a_i=0 then qpoint
        else ecc_sub qpoint point
```

OPTIMIERTE SCHNELLE SKALARMULTIPLIKATION

- **Iterierte Verdoppelung, Addition, Subtraktion**

- Zur Berechnung von $k \cdot P$ erzeuge NAF Darstellung $\sum_{i=0}^l a_i 2^i$ von k
- Verdoppele für jedes a_i , addiere/subtrahiere zusätzlich für $a_i=1/-1$

- **Funktionale Implementierung**

```
let rec ecc_mult point a_list
= if a_list = [] then  $\mathcal{O}$ 
  else let a_i::rest = a_list
        and qpoint = ecc_mult (ecc_add point point) rest
        in
        if a_i=1 then ecc_add qpoint point
        else if a_i=0 then qpoint
        else ecc_sub qpoint point
```

- **Laufzeit**

- Eine Verdoppelung pro c_i , eine Addition/Subtraktion für $c_i \neq 0$
- Insgesamt maximal $2 \log_2 k$ Additionen auf $E(p; a, b)$
- Additionen sind (bis auf konstanten Faktor) genauso schnell wie in $\mathbb{Z}_p$

OPTIMIERTE SCHNELLE SKALARMULTIPLIKATION

- **Iterierte Verdoppelung, Addition, Subtraktion**

- Zur Berechnung von $k \cdot P$ erzeuge NAF Darstellung $\sum_{i=0}^l a_i 2^i$ von k
- Verdoppele für jedes a_i , addiere/subtrahiere zusätzlich für $a_i=1/-1$

- **Funktionale Implementierung**

```
let rec ecc_mult point a_list
= if a_list = [] then  $\mathcal{O}$ 
  else let a_i::rest = a_list
        and qpoint = ecc_mult (ecc_add point point) rest
        in
        if a_i=1 then ecc_add qpoint point
        else if a_i=0 then qpoint
        else ecc_sub qpoint point
```

- **Laufzeit**

$\mathcal{O}(\|p\|^3)$

- Eine Verdoppelung pro c_i , eine Addition/Subtraktion für $c_i \neq 0$
- Insgesamt maximal $2 \log_2 k$ Additionen auf $E(p; a, b)$
- Additionen sind (bis auf konstanten Faktor) genauso schnell wie in $\mathbb{Z}_p$

OPTIMIERTE SCHNELLE SKALARMULTIPLIKATION

- **Iterierte Verdoppelung, Addition, Subtraktion**

- Zur Berechnung von $k \cdot P$ erzeuge NAF Darstellung $\sum_{i=0}^l a_i 2^i$ von k
- Verdoppele für jedes a_i , addiere/subtrahiere zusätzlich für $a_i=1/-1$

- **Funktionale Implementierung**

```
let rec ecc_mult point a_list
= if a_list = [] then  $\mathcal{O}$ 
  else let a_i::rest = a_list
        and qpoint = ecc_mult (ecc_add point point) rest
        in
        if a_i=1 then ecc_add qpoint point
        else if a_i=0 then qpoint
        else ecc_sub qpoint point
```

- **Laufzeit**

$\mathcal{O}(\|p\|^3)$

- Eine Verdoppelung pro c_i , eine Addition/Subtraktion für $c_i \neq 0$
- Insgesamt maximal $2 \log_2 k$ Additionen auf $E(p; a, b)$
- Additionen sind (bis auf konstanten Faktor) genauso schnell wie in $\mathbb{Z}_p$

- **Effekt der Optimierung**

- Im statistischen Mittel sind 2/3 aller Koeffizienten Null (statt 1/2)
- Nur $\frac{4}{3}l$ Additionen/Subtraktionen nötig statt $\frac{3}{2}l$ Additionen

OPTIMIERTE SCHNELLE SKALARMULTIPLIKATION

● Iterierte Verdoppelung, Addition, Subtraktion

- Zur Berechnung von $k \cdot P$ erzeuge NAF Darstellung $\sum_{i=0}^l a_i 2^i$ von k
- Verdoppele für jedes a_i , addiere/subtrahiere zusätzlich für $a_i=1/-1$

● Funktionale Implementierung

```
let rec ecc_mult point a_list
= if a_list = [] then  $\mathcal{O}$ 
  else let a_i::rest = a_list
        and qpoint = ecc_mult (ecc_add point point) rest
        in
        if a_i=1 then ecc_add qpoint point
        else if a_i=0 then qpoint
        else ecc_sub qpoint point
```

● Laufzeit

$\mathcal{O}(\|p\|^3)$

- Eine Verdoppelung pro c_i , eine Addition/Subtraktion für $c_i \neq 0$
- Insgesamt maximal $2 \log_2 k$ Additionen auf $E(p; a, b)$
- Additionen sind (bis auf konstanten Faktor) genauso schnell wie in $\mathbb{Z}_p$

● Effekt der Optimierung

11% schneller

- Im statistischen Mittel sind 2/3 aller Koeffizienten Null (statt 1/2)
- Nur $\frac{4}{3}l$ Additionen/Subtraktionen nötig statt $\frac{3}{2}l$ Additionen

- **EC-Verschlüsselung vergrößert Nachrichten**
 - Punkte von $E(p; a, b)$ werden durch zwei Zahlen in $\mathbb{Z}_p$ dargestellt
 - ElGamal Verschlüsselung verdoppelt Klartext durch DH-Teilschlüssel
 - Schlüsseltext ist etwa vier mal so lang wie ursprünglicher Klartext

- **EC-Verschlüsselung vergrößert Nachrichten**
 - Punkte von $E(p; a, b)$ werden durch zwei Zahlen in $\mathbb{Z}_p$ dargestellt
 - ElGamal Verschlüsselung verdoppelt Klartext durch DH-Teilschlüssel
 - Schlüsseltext ist etwa vier mal so lang wie ursprünglicher Klartext
- **$E(p; a, b)$ hat maximal $2p$ Punkte**
 - Ist $P = (x_P, y_P) \in E(p; a, b)$ dann ist $-P = (x_P, p-y_P)$
der einzige Punkt mit derselben x Koordinate
 - Da p Primzahl sein muß, ist y_P gerade g.d.w. $p-y_P$ ungerade ist

- **EC-Verschlüsselung vergrößert Nachrichten**

- Punkte von $E(p; a, b)$ werden durch zwei Zahlen in $\mathbb{Z}_p$ dargestellt
- ElGamal Verschlüsselung verdoppelt Klartext durch DH-Teilschlüssel
- Schlüsseltext ist etwa vier mal so lang wie ursprünglicher Klartext

- **$E(p; a, b)$ hat maximal $2p$ Punkte**

- Ist $P = (x_P, y_P) \in E(p; a, b)$ dann ist $-P = (x_P, p-y_P)$
der einzige Punkt mit derselben x Koordinate
- Da p Primzahl sein muß, ist y_P gerade g.d.w. $p-y_P$ ungerade ist

- **Komprimiere Darstellung von Punkten**

- Repräsentation von $P = (x_P, y_P)$ ist $(x_P, y_P \bmod 2)$
- Nur ein Bit mehr als Darstellung von Texten durch Elemente von $\mathbb{Z}_p$
- Kompression kann effizient invertiert und in die ElGamal Verschlüsselung integriert werden

● Dekompression komprimierter Punkte

- Bei Eingabe der Punktkompression (x, y') berechne $z = x^3 + a \cdot x + b$
- Falls z ein quadratischer Rest modulo p ist berechne $y = \sqrt{z} \bmod p$
- Wenn $y' \equiv y \bmod 2$ dann gebe (x, y) aus und ansonsten $(x, p-y)$

● Dekompression komprimierter Punkte

- Bei Eingabe der Punktkompression (x, y') berechne $z = x^3 + a \cdot x + b$
 - Falls z ein quadratischer Rest modulo p ist berechne $y = \sqrt{z} \bmod p$
 - Wenn $y' \equiv y \bmod 2$ dann gebe (x, y) aus und ansonsten $(x, p-y)$
-
- z ist quadratischer Rest modulo p g.d.w. $z^{(p-1)/2} \equiv 1 \bmod p$
 - Wenn $p \equiv 3 \bmod 4$, dann ist $\sqrt{z} = z^{(p+1)/4}$ (vgl. Rabin Verfahren §4.4)

INTEGRIERTE EC-VERSCHLÜSSELUNG

● Dekompression komprimierter Punkte

- Bei Eingabe der Punktkompression (x, y') berechne $z = x^3 + a \cdot x + b$
 - Falls z ein quadratischer Rest modulo p ist berechne $y = \sqrt{z} \bmod p$
 - Wenn $y' \equiv y \bmod 2$ dann gebe (x, y) aus und ansonsten $(x, p-y)$
-
- z ist quadratischer Rest modulo p g.d.w. $z^{(p-1)/2} \equiv 1 \bmod p$
 - Wenn $p \equiv 3 \bmod 4$, dann ist $\sqrt{z} = z^{(p+1)/4} \quad (\text{vgl. Rabin Verfahren §4.4})$

● Vereinfachtes ECIES Verfahren

Komprimierter ECDH Teilschlüssel & verkürzte Nachrichtenchiffre

INTEGRIERTE EC-VERSCHLÜSSELUNG

● Dekompression komprimierter Punkte

- Bei Eingabe der Punktkompression (x, y') berechne $z = x^3 + a \cdot x + b$
 - Falls z ein quadratischer Rest modulo p ist berechne $y = \sqrt{z} \bmod p$
 - Wenn $y' \equiv y \bmod 2$ dann gebe (x, y) aus und ansonsten $(x, p-y)$
-
- z ist quadratischer Rest modulo p g.d.w. $z^{(p-1)/2} \equiv 1 \bmod p$
 - Wenn $p \equiv 3 \bmod 4$, dann ist $\sqrt{z} = z^{(p+1)/4} \quad (\text{vgl. Rabin Verfahren §4.4})$

● Vereinfachtes ECIES Verfahren

Komprimierter ECDH Teilschlüssel & verkürzte Nachrichtenchiffre

- Gegeben zyklische elliptische Kurve $E = E(p; a, b)$ der Ordnung n ein erzeugender Punkt P , ein zufälliges $a \in \{0, \dots, n-1\}$ und $A = a \cdot P$
- Schlüssel ist $K := (E, P, a, A, n)$, wobei nur E, P, A, n öffentlich

● Dekompression komprimierter Punkte

- Bei Eingabe der Punktkompression (x, y') berechne $z = x^3 + a \cdot x + b$
 - Falls z ein quadratischer Rest modulo p ist berechne $y = \sqrt{z} \bmod p$
 - Wenn $y' \equiv y \bmod 2$ dann gebe (x, y) aus und ansonsten $(x, p-y)$
-
- z ist quadratischer Rest modulo p g.d.w. $z^{(p-1)/2} \equiv 1 \bmod p$
 - Wenn $p \equiv 3 \bmod 4$, dann ist $\sqrt{z} = z^{(p+1)/4} \quad (\text{vgl. Rabin Verfahren §4.4})$

● Vereinfachtes ECIES Verfahren

Komprimierter ECDH Teilschlüssel & verkürzte Nachrichtenchiffre

- Gegeben zyklische elliptische Kurve $E = E(p; a, b)$ der Ordnung n ein erzeugender Punkt P , ein zufälliges $a \in \{0, \dots, n-1\}$ und $A = a \cdot P$
- Schlüssel ist $K := (E, P, a, A, n)$, wobei nur E, P, A, n öffentlich
- **Verschlüsselung**: für ein zufälliges $b \in \{0, \dots, n-1\}$ und ein $x \in \mathbb{Z}_p$ sei
$$e_K(X, b) = (\text{Compress}(b \cdot P), x \cdot x_q \bmod p), \text{ wobei } b \cdot A = (x_q, y_q)$$
- **Entschlüsselung**: für einen Schlüsseltext (B, y) sei
$$d_K(B, y) = y \cdot (x_q)^{-1} \bmod p, \text{ wobei } a \cdot \text{DeCompress}(B) = (x_q, y_q)$$

ECIES VERFAHREN AM BEISPIEL

● Schlüsselerzeugung

- Alice wählt $E = E(23; 2, 8)$ (Ordnung 16)
und $P = (0, 10)$ als erzeugenden Punkt
- Alice wählt $a = 6$ und berechnet $A = 6 \cdot (0, 10) = (15, 20)$
- Alice veröffentlicht $K = (E(23; 2, 8), (0, 10), (15, 20), 16)$

ECIES VERFAHREN AM BEISPIEL

● Schlüsselerzeugung

- Alice wählt $E = E(23; 2, 8)$ (Ordnung 16)
und $P = (0, 10)$ als erzeugenden Punkt
- Alice wählt $a = 6$ und berechnet $A = 6 \cdot (0, 10) = (15, 20)$
- Alice veröffentlicht $K = (E(23; 2, 8), (0, 10), (15, 20), 16)$

● Verschlüsselung

- Bob wählt $b = 3$ und berechnet $B = \text{Compress}(3 \cdot (0, 10)) = (10, 0)$
und $b \cdot A = 3 \cdot (15, 20) = (3, 15)$
- Verschlüsselung von $x = 22$ ergibt $y = 22 \cdot 3 \bmod 23 = 20$
- Bob versendet als Schlüsseltext $e_K(X, b) = (B, y) = ((10, 0), 20)$

ECIES VERFAHREN AM BEISPIEL

● Schlüsselerzeugung

- Alice wählt $E = E(23; 2, 8)$ (Ordnung 16)
und $P = (0, 10)$ als erzeugenden Punkt
- Alice wählt $a = 6$ und berechnet $A = 6 \cdot (0, 10) = (15, 20)$
- Alice veröffentlicht $K = (E(23; 2, 8), (0, 10), (15, 20), 16)$

● Verschlüsselung

- Bob wählt $b = 3$ und berechnet $B = \text{Compress}(3 \cdot (0, 10)) = (10, 0)$
und $b \cdot A = 3 \cdot (15, 20) = (3, 15)$
- Verschlüsselung von $x = 22$ ergibt $y = 22 \cdot 3 \bmod 23 = 20$
- Bob versendet als Schlüsseltext $e_K(X, b) = (B, y) = ((10, 0), 20)$

● Entschlüsselung

- Alice berechnet $a \cdot \text{DeCompress}(B) = 6 \cdot (10, 4) = (3, 15)$
- Entschlüsselung von y liefert $20 \cdot 3^{-1} \bmod 23 = 20 \cdot 8 \bmod 23 = 22$

KOMPLEXITÄT VON ECIES

- **Korrektheit: Ver-/Entschlüsselung sind invers**

- Es ist $d_K(B, y) = y \cdot (x_q)^{-1} \bmod p$ wobei $x_q = (a \cdot \text{DeCompress}(B))_1$
und $e_K(X, b) = (\text{Compress}(b \cdot P), x \cdot x_q \bmod p)$ wobei $x_q = (b \cdot A)_1$
- Für beliebige b ist $d_K(e_K(x, b)) = y \cdot ((a \cdot \text{DeCompress}(B))_1)^{-1} \bmod p$
 $= (x \cdot (b \cdot A)_1 \bmod p) \cdot (((a \cdot \text{DeCompress}(\text{Compress}(b \cdot P)))_1)^{-1} \bmod p)$
 $= (x \cdot (b \cdot a \cdot P)_1 \bmod p) \cdot (((a \cdot b \cdot P)_1)^{-1} \bmod p) = x$

KOMPLEXITÄT VON ECIES

- **Korrektheit: Ver-/Entschlüsselung sind invers**

- Es ist $d_K(B, y) = y \cdot (x_q)^{-1} \bmod p$ wobei $x_q = (a \cdot \text{DeCompress}(B))_1$
und $e_K(X, b) = (\text{Compress}(b \cdot P), x \cdot x_q \bmod p)$ wobei $x_q = (b \cdot A)_1$
- Für beliebige b ist $d_K(e_K(x, b)) = y \cdot ((a \cdot \text{DeCompress}(B))_1)^{-1} \bmod p$
 $= (x \cdot (b \cdot A)_1 \bmod p) \cdot (((a \cdot \text{DeCompress}(\text{Compress}(b \cdot P)))_1)^{-1} \bmod p)$
 $= (x \cdot (b \cdot a \cdot P)_1 \bmod p) \cdot (((a \cdot b \cdot P)_1)^{-1} \bmod p) = x$

- **Aufwand für Ver- und Entschlüsselung**

- Kein Aufwand für Umwandlung zwischen Text und Zahlen (!)
- Skalarmultiplikation in $E(p; a, b)$ und Multiplikation in $\mathbb{Z}_p$
für $8|w|/\|n\|$ Blöcke $\mathcal{O}(|w| \cdot \|p\|^2)$
- Schneller als ElGamal über $\mathbb{Z}_p$ wegen Verwendung kleinerer Blöcke
- Nachrichtenexpansion um Faktor 2

SICHERHEIT VON EC-ELGAMAL SYSTEMEN

- **Standard DL Algorithmen sind universell**
 - Ordnung der Gruppe $E(p; a, b)$ liegt in Größenordnung von p hat aber keinen trivialen Zusammenhang zu p , a und b
 - Shanks und Pollard ρ sind anwendbar, aber zu ineffizient für große p
 - Pohlig-Hellman kann erfolgreich sein, wenn Ordnung von $E(p; a, b)$ nur kleine Primfaktoren hat

SICHERHEIT VON EC-ELGAMAL SYSTEMEN

- **Standard DL Algorithmen sind universell**
 - Ordnung der Gruppe $E(p; a, b)$ liegt in Größenordnung von p hat aber keinen trivialen Zusammenhang zu p , a und b
 - Shanks und Pollard ρ sind anwendbar, aber zu ineffizient für große p
 - Pohlig-Hellman kann erfolgreich sein, wenn Ordnung von $E(p; a, b)$ nur kleine Primfaktoren hat
- **Index-Calculus Methode nicht anwendbar**
 - EC-Addition führt zu “chaotischen” Sprüngen in $\mathbb{Z}_p \times \mathbb{Z}_p$
 - Keine einfache Verwendung von “Faktorbasen” möglich
 - Zahlkörpersiebe und ähnliche arithmetisch basierte Methoden können (bisher) nicht übertragen werden

SICHERHEIT VON EC-ELGAMAL SYSTEMEN

- **Standard DL Algorithmen sind universell**
 - Ordnung der Gruppe $E(p; a, b)$ liegt in Größenordnung von p hat aber keinen trivialen Zusammenhang zu p , a und b
 - Shanks und Pollard ρ sind anwendbar, aber zu ineffizient für große p
 - Pohlig-Hellman kann erfolgreich sein, wenn Ordnung von $E(p; a, b)$ nur kleine Primfaktoren hat
- **Index-Calculus Methode nicht anwendbar**
 - EC-Addition führt zu “chaotischen” Sprüngen in $\mathbb{Z}_p \times \mathbb{Z}_p$
 - Keine einfache Verwendung von “Faktorbasen” möglich
 - Zahlkörpersiebe und ähnliche arithmetisch basierte Methoden können (bisher) nicht übertragen werden
- **Spezielle Methoden greifen nur Sonderfälle an**
 - MOV / Frey-Rück-Methode übertragen ECDLP auf DL Problem für $\mathbb{Z}_{p^k}$ wenn Gruppenordnung n Teiler von $p^k - 1$
 - Zahlentheoretische Methode von Satoh, Araki & Smart überträgt ECDLP auf DL Problem für $\mathbb{Z}_{p^k}$ wenn $n=p$
 - Algebraische Methode von Semaev & Rück

- **163 Bit Schlüssel reichen aus**

- Beste Attacke auf ECC DL hat Laufzeit in $\mathcal{O}(2^{\|n\|/2})$
- ECC Schlüssel nur doppelt so groß wie bei symmetrischen Verfahren
- 160 Bit ECC Schlüssel genauso sicher wie 1024 Bit RSA Schlüssel
- 224 Bit ECC Schlüssel genauso sicher wie 2048 Bit RSA Schlüssel
- 256 Bit ECC Schlüssel genauso sicher wie 3072 Bit RSA Schlüssel
- Faktor zwischen ECC und RSA wächst mit Schlüssellänge

KRYPTOGRAPHISCH GEEIGNETE ELLIPTISCHE KURVEN

● 163 Bit Schlüssel reichen aus

- Beste Attacke auf ECC DL hat Laufzeit in $\mathcal{O}(2^{\|n\|/2})$
- ECC Schlüssel nur doppelt so groß wie bei symmetrischen Verfahren
- 160 Bit ECC Schlüssel genauso sicher wie 1024 Bit RSA Schlüssel
- 224 Bit ECC Schlüssel genauso sicher wie 2048 Bit RSA Schlüssel
- 256 Bit ECC Schlüssel genauso sicher wie 3072 Bit RSA Schlüssel
- Faktor zwischen ECC und RSA wächst mit Schlüssellänge

● Empfehlungen für Auswahl der Kurven

- Gruppenordnung von $E(p; a, b)$ muß Primfaktor größer als 2^{160} haben
(Pohlig-Hellman)
- Kurve darf nicht supersingulär oder anomal sein
- supersingulär: Gruppenordnung Teiler von $p^k - 1$ (Mezenes, Okamoto, VanStone)
- anomal: Gruppenordnung ist p (Sato, Araki & Smart)
- Geheimer Schlüssel aus $\{0, \dots, n-1\}$ muß groß sein (Brute-Force Suche)

ELGAMAL VERFAHREN IM RÜCKBLICK

- **Effizientes Public-Key Kryptosystem**

- Potenzierung von Elementen einer (beliebigen) zyklischen Gruppe
- Sicherheit der Schlüssel basiert auf Problem des diskreten Logarithmus
- Erhöhte semantische Sicherheit durch Randomisierung
- Nachrichtenexpansion um Faktor 2

ELGAMAL VERFAHREN IM RÜCKBLICK

- **Effizientes Public-Key Kryptosystem**

- Potenzierung von Elementen einer (beliebigen) zyklischen Gruppe
- Sicherheit der Schlüssel basiert auf Problem des diskreten Logarithmus
- Erhöhte semantische Sicherheit durch Randomisierung
- Nachrichtenexpansion um Faktor 2

- **Effizienter als RSA**

- Bei ‘nichtarithmetischen’ Gruppen mit schwerem DL Problem hat die beste Attacke eine Laufzeit von $\mathcal{O}(2^{\lceil n \rceil / 2})$
- Schlüssellängen von 160 Bit gelten als sicher
- Implementierbar auf Smartcards mit geringer Prozessorleistung
- Elliptische Kurven gelten derzeit die beste Grundlage