Elliptische Kurven in „Unicrypt & UniVote“

Christian Lutz, Unifr

Inhalt

- Wer bin ich?
- Elliptische Kurven, weshalb?
- Elliptische Kurven?!
- Herausforderungen
- Mapping
- Message-Size

Wer bin ich?

- Name: Christian Lutz
- Alter: 25
- Wohnort: Courtepin
- Ausbildung:
  - Bachelor an der Unifr
  - Zur Zeit am Master an der UniBNF(Bern, Neuenburg, Freiburg)

Bachelorarbeit

- Bachelorarbeit mit Reto als Betreuer
- Elgamal auf Restklassen – auf Elliptischen Kurven, lohnt sich ein Wechsel?

Schlüssellänge Elgamal

Date	Minimum of Strength	Symmetric Algorithms	Asymmetric	Discrete Logarithm Key	Discrete Logarithm Group	Elliptic Curve	Hash (A)	Hash (B)
2010 (Legacy)	80	2TDEA*	1024	160	1024	160	SHA-1** SHA-224 SHA-256 SHA-384 SHA-512	SHA-1 SHA-224 SHA-256 SHA-384 SHA-512
2011 - 2030	112	3TDEA	2048	224	2048	224	SHA-224 SHA-256 SHA-384 SHA-512	SHA-1 SHA-224 SHA-256 SHA-384 SHA-512
> 2030	128	AES-128	3072	256	3072	256	SHA-256 SHA-384 SHA-512	SHA-1 SHA-224 SHA-256 SHA-384 SHA-512
>> 2030	192	AES-192	7680	384	7680	384	SHA-384 SHA-512	SHA-224 SHA-256 SHA-384 SHA-512
>>> 2030	256	AES-256	15360	512	15360	512	SHA-512	SHA-256 SHA-384 SHA-512

- Quelle: Nist, keylength.com

Kleine Demo

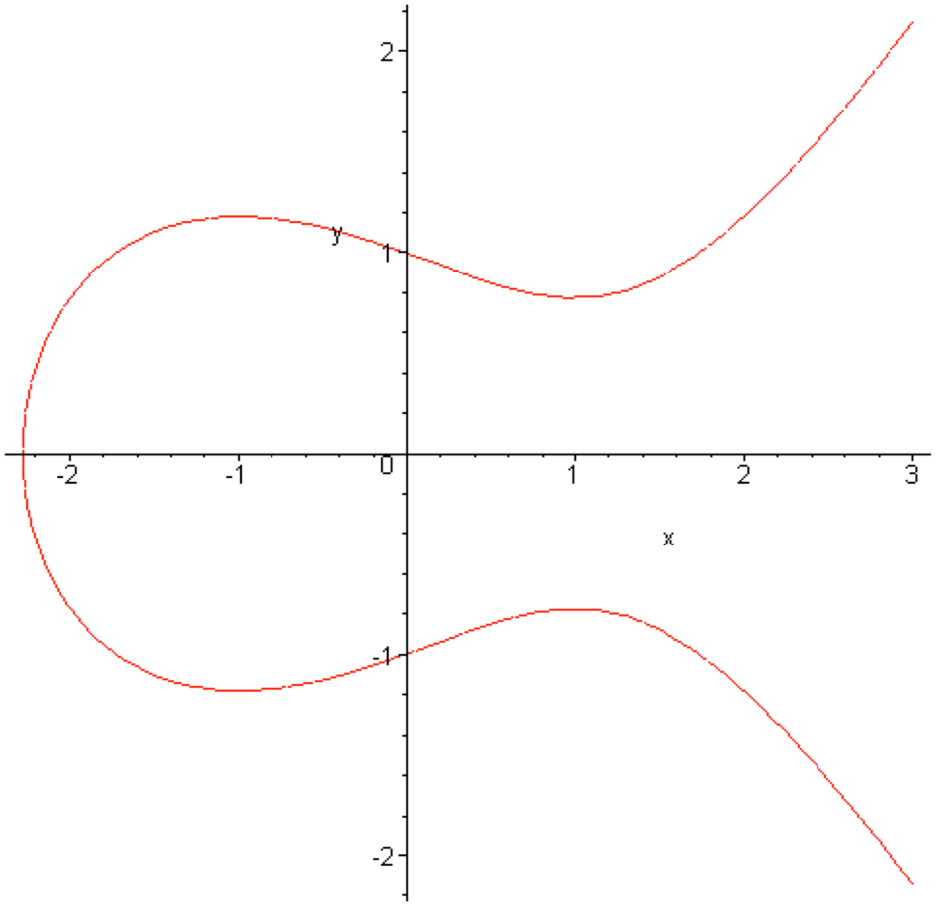
http://www.elgamal.geoforest.ch

Masterarbeit

- Implementation der Elliptischen Kurven in Unicrypt - Java
- Implementation der Elliptischen Kurven in Univote – JavaScript
- Lösen der damit verbundenen Problemen

Elliptische Kurven über R

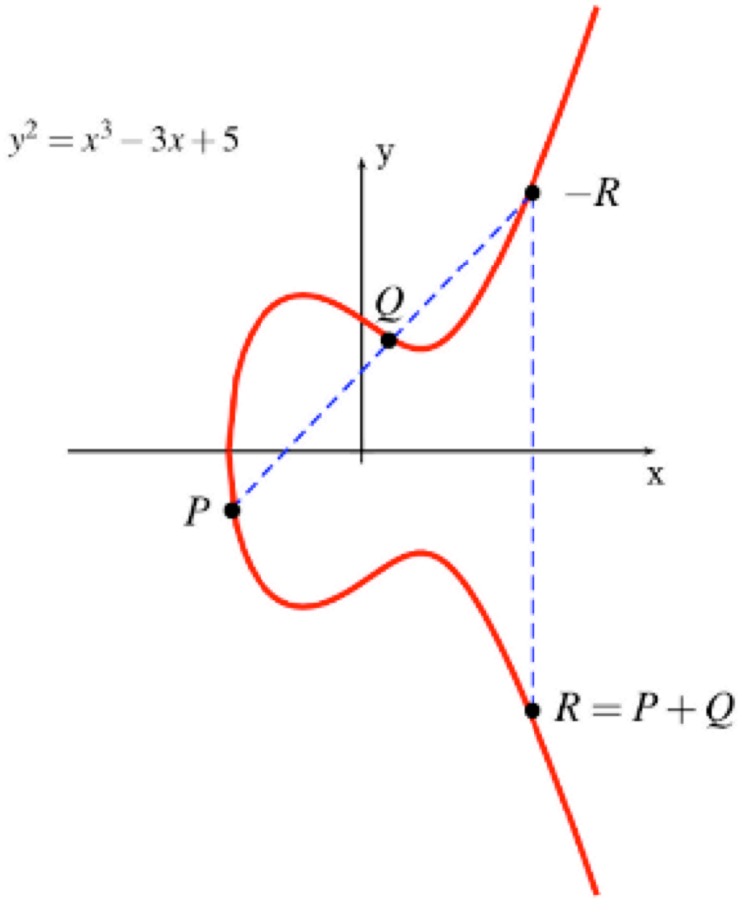
$y^2=x^3+ax+b$



Eigenschaften der EC

- Additive Gruppe
  - Addition a+b=c wobei a,b,c Punkte der EC
  - Assoziativgesetz (a + b) + c = a + (b + c)
  - Neutrales Element 0 so dass a+0=a
  - Inverses Element b so dass a+b=0
- Vergl. Elgamal über RK ->EC
  - Multiplikation -> Addition
  - Power -> Skalare Multiplikation
- Grund für kürzere Schlüssel
  - Keine Multiplikation definiert!

Addition geometrisch

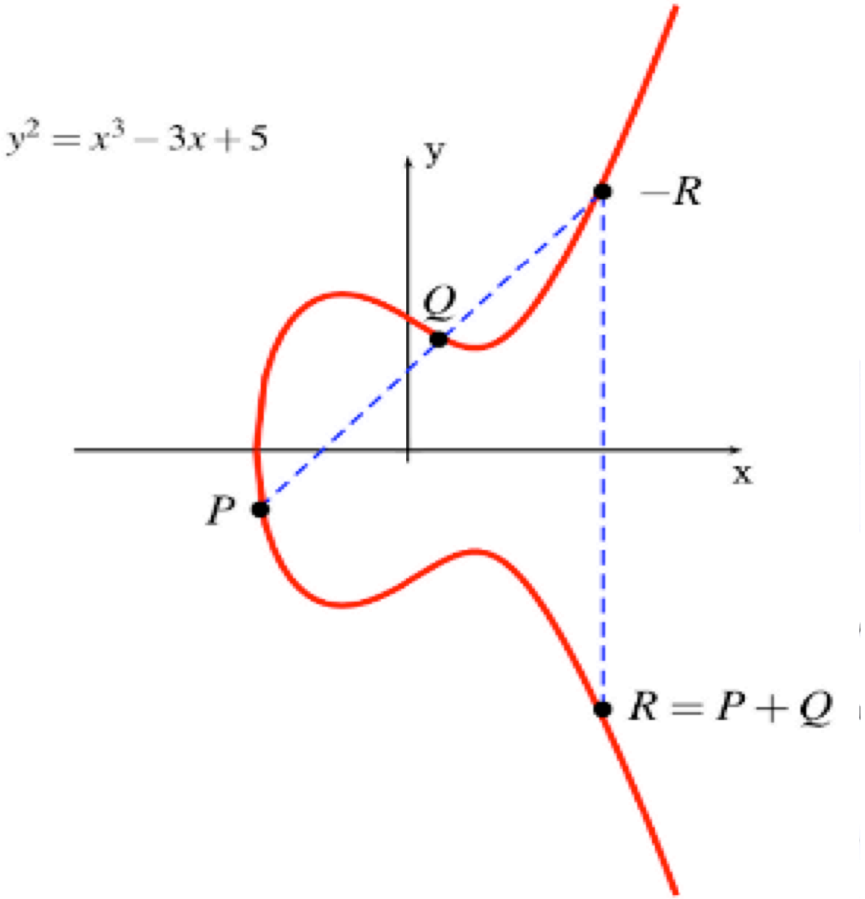


Addition algebraisch

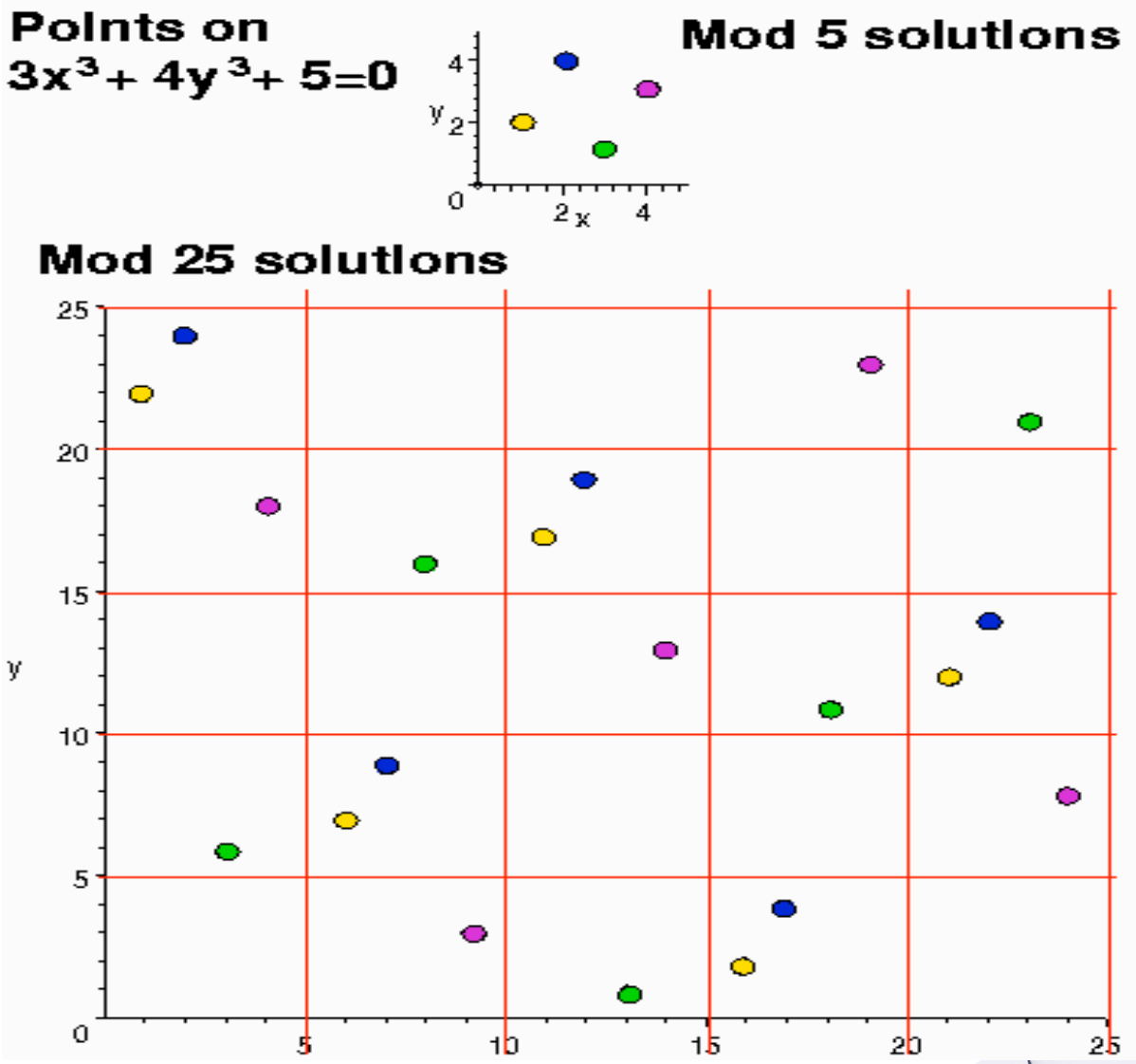
- $P+Q=R \rightarrow (x_p, y_p)+(x_q, y_q)=(x_r, y_r), P \neq Q$
  - $s=y_p-y_q \div x_p-x_q$
  - $x_r=s^2-x_p-x_q$
  - $y_r=s(x_p-x_r)-y_p$
- $P+P=R \rightarrow 2*(x_p, y_p)=(x_r, y_r)$
  - $s=(3*x_p{}^2+a) \div (2*y_p)$
  - $x_r=s^2-2*x_p$
  - $y_r=s(x_p-x_r)-y_p$

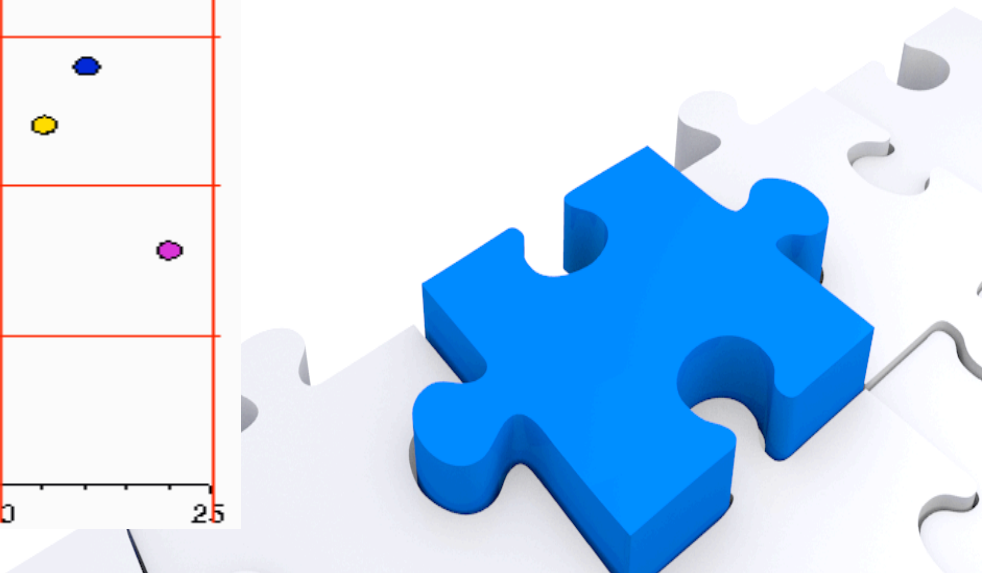
Weitere Gruppeneigenschaften

- Neutrales Element
  - Der Punkt im „unendlichen“
- Inverses
  - inv((x,y))=(x,-y)



Ellitpische Kurven über endlichen Körper





Elliptische Kurven in UniCrypt

- Über Restklassen ->ECZModPrime
- Über den Polynomen -> ECBinaryPolinomialField
- Soll sich in die vorgegebenen Strukturen einfügen lassen

Herausforderungen mit EC

- Ordnung der Gruppe bestimmen
- Generator finden
- Ordnung des Generators bestimmen
- $Mapping\, \mathbb{Z}\, mod\ p \rightarrow EC\ und\ zurück$

Ordnung der Gruppe/Generator

- Schoof–Elkies–Atkin algorithm
  - $O(\log^4 p)$ für Kurven über Fp
  - Nicht online einsetzbar in Cryptolibrary
- Bekannte Domainparameter
  - Testen ob Ordnung prim ist
  - Testen ob Ordnung*Generator = 0 ist

Bekannte Standarts

- SECG – Standards for efficient cryptographie group
- NIST
- ISO
- ECC-Brainpool
- BSI

Sicherheit der Standarts

- NSA=NIST?
- NIST Mitglied von SECG -> NSA=SECG?
- BND=BSI?
- BSI Mitglied von ECC-Brainpool -> BND=ECC-Brainpool?

Sicherheit der Standarts 2

- EC bereits gut erforscht?!
- Nur eine additive Gruppe
- Auch andere mathematische Strukturen könnten betroffen sein

Mapping

- Mapping ZModPrime – EC – ZModPrime
  - Im Durchschnitt nur jedes 2. x -> y
- Einfachster Weg:
  - i*Generator, mit i Element aus ZModPrime
  - i kann nicht zurückberechnet werden -> diskreter Logarithmus
- Kobliz-Methode
  - ASCII-Codierung -> Buchstaben einzeln verschlüsseln

Mapping 2

- Probabilistische Methode encode
  - k Bits an Nachricht m anhängen
  - m=m+1 bis gültiges x gefunden ist oder m+k erreicht ist.
  - Wahrscheinlichkeit für kein gültiges x ca. $\rightarrow \frac{1}{2^k}$

Mapping 3

- Probabilistische Methode decode
  - X-Koordinate letzte k Bits entfernen
  - m=X.shiftRight(k)
- Problem
  - Homomorphe Eigenschaften gehen verloren
- Weitere Methoden werden zur Zeit gesucht/studiert

Message-Size

- Bei perfektem Mapping
  - Message-Size<=Order der Gruppe
- Für Elgamal auf Restklassen
  - 4096Bit Primzahl -> maximal 4096 Bit Message-Size
- Für Elgamal auf EC
  - Ordnung 256Bit – 521 Bit
  - Evtl. zu kleine Message-Size

Besten Dank!

Fragen?

Quellen

- http://www.keylength.com/
- http://de.wikipedia.org/wiki/Elliptische_Kurve
- http://www.embedded.com
- http://www.math.lsu.edu