SIEMENS

CORPORATE TECHNOLOGY

# Kryptographie mit Elliptischen Kurven

Erwin Heß
Corporate Technology
Dept. CT IC 3

Information & Communications Security

# Was sind elliptische Kurven?

***Elliptische Kurven sind keine Ellipsen!***

- Eine elliptische Kurve E ist eine ebene Kurve, die durch eine kubische Gleichung beschrieben werden kann, typischerweise von der Form:

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

- Die Lösungen der Gleichung (x,y) mit $x,y \in K$ (K ein Körper) sind die Punkte der Kurve.

**Bemerkenswert :**

- Die Punkte einer elliptischen Kurve lassen sich „addieren“:

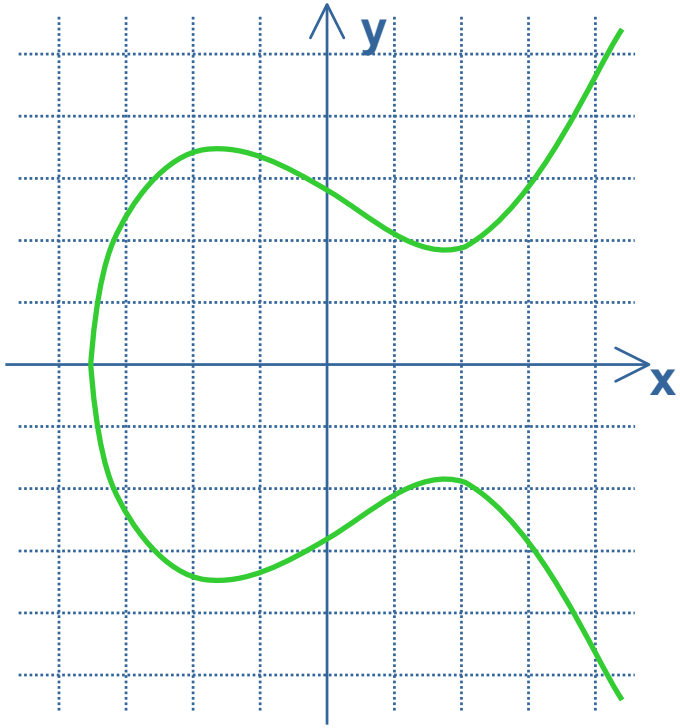
$$P_1, P_2 \rightarrow P_1 + P_2$$

- Diese Punktaddition genügt den „Gruppenaxiomen“.

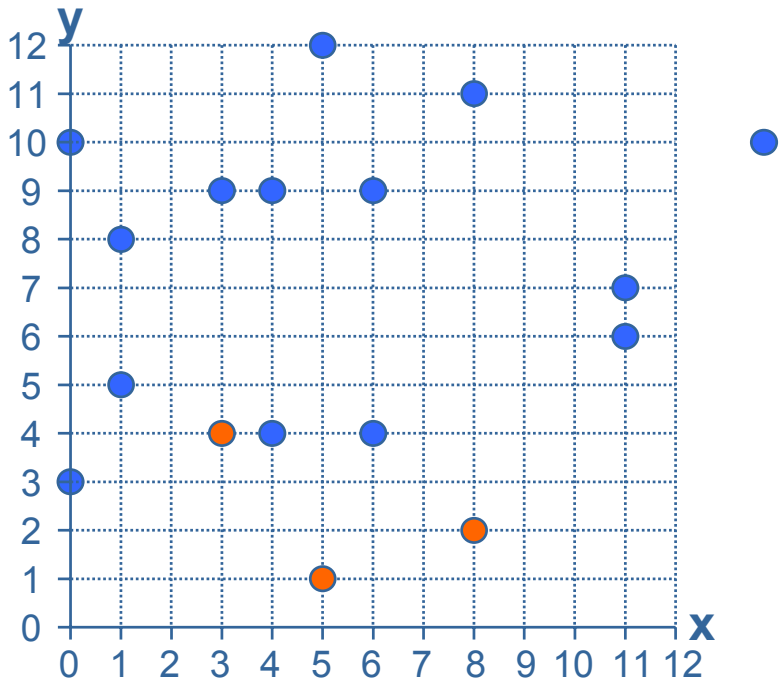
# Wie sehen elliptische Kurven aus?

Die elliptische Kurve

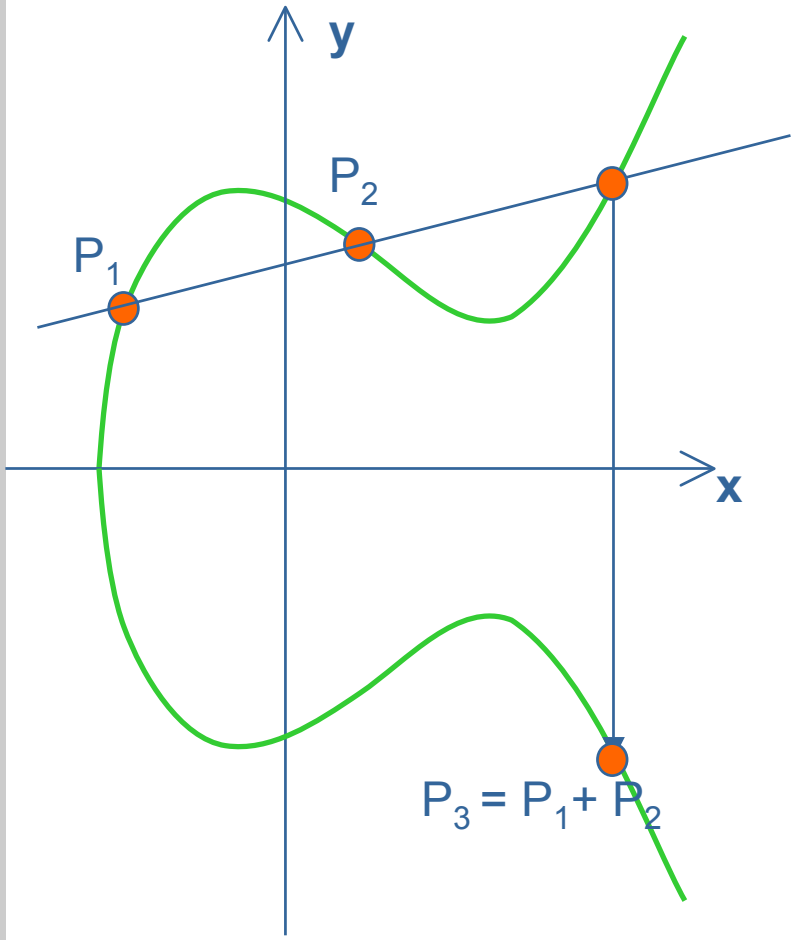
$y^2 = x^3 - 2x + 2$ über den reellen Zahlen.




Die elliptische Kurve $y^2 = x^3 + 2x + 9$ über dem endlichen Körper $F_{13}$



# Wie addiert man Kurvenpunkte? (I)




**Geometrische Konstruktion von $P_1 + P_2$**

1. Zeichne Verbindungsgerade $P_1 P_2$ . Diese schneidet die elliptische Kurve in genau <u>einem</u> weiteren Punkt.
2. Spiegelbild des Schnittpunktes an der x-Achse = $P_3$.

<u>Formeln</u>:

$x_3 = r^2 - x_1 - x_2$, $y_3 = r(x_1 - x_3) - y_1$ mit:

$r = (y_2 - y_1)/(x_2 - x_1)$, falls $P_1 \neq P_2$ , $-P_2$

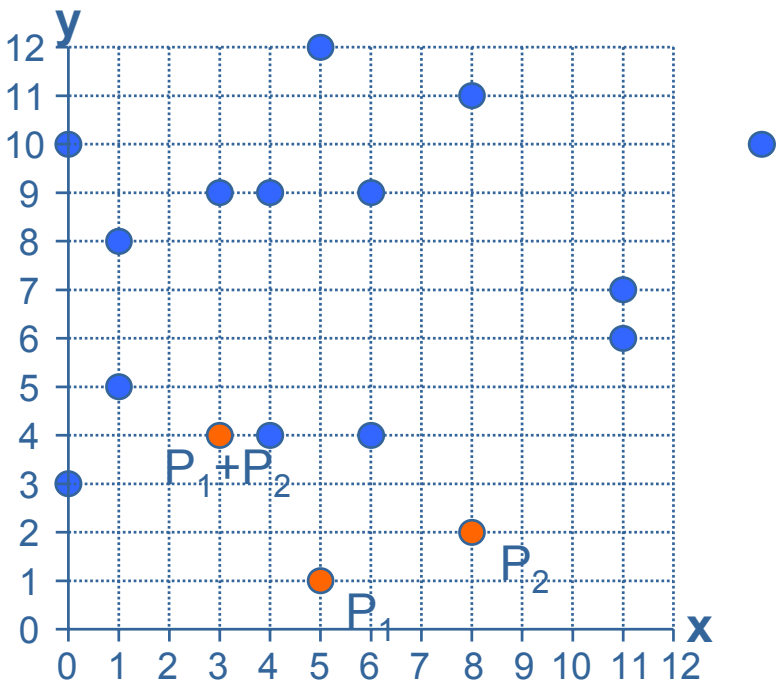
$r = (3\, x_1^2 + a)/2y_1$, falls $P_1 = P_2$

CORPORATE TECHNOLOGY

Information & Communications Security

# Wie addiert man Kurvenpunkte? (II)

**Elliptische Kurve über endlichem Körper**




**Die Formeln zur Addition von Kurvenpunkten gelten auch für elliptische Kurven über endlichen Körpern :**

$x_3 = r^2 - x_1 - x_2$,

$y_3 = r(x_1 - x_3) - y_1$ mit:

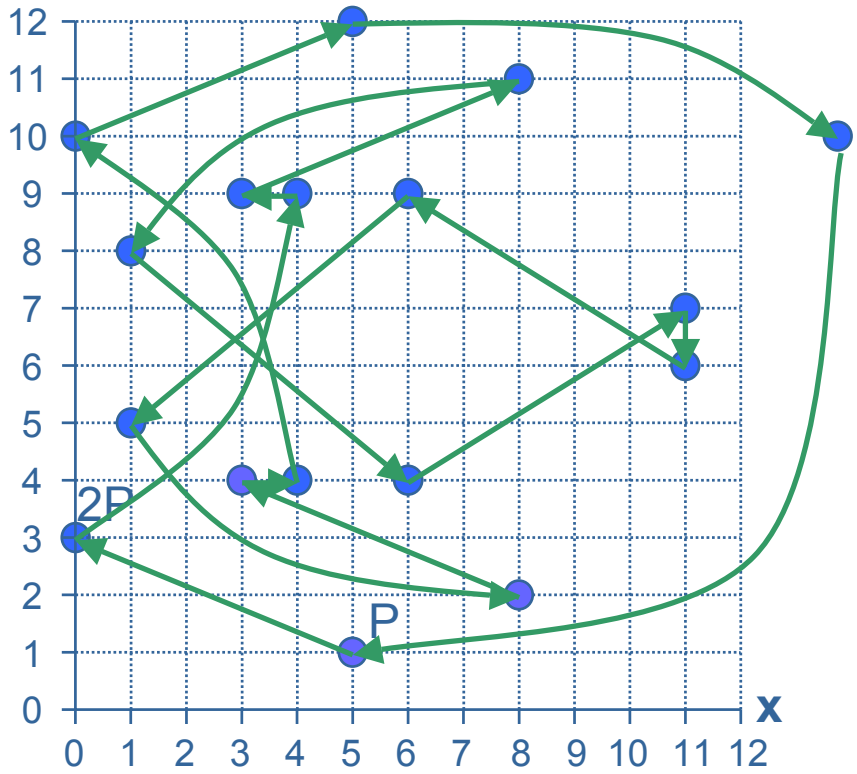
$r = (y_2 - y_1)/(x_2 - x_1)$, falls $P_1 \neq P_2$ , $-P_2$

$r = (3\,x_1^2 + a)/2y_1$, falls $P_1 = P_2$

CORPORATE TECHNOLOGY

Information & Communications Security

# Erzeugender Kurvenpunkt




**Beispiel:** $y^2 = x^3 + 2x + 9$ über $F_{13}$.

**Wähle P = (5,1).**

- **Durch sukzessive Addition von P erhält man alle Kurvenpunkte.**
- Man spricht von einer **zyklischen Gruppe mit erzeugendem Element P**.

CORPORATE TECHNOLOGY

Information & Communications Security

# Relevanz elliptischer Kurven für die Kryptographie

- Durch Iteration der Punktaddition definiert man eine Multiplikation von Kurvenpunkten mit ganzen Zahlen: $kP = P + P + \ldots + P$,
  (k-fache Summe)

- Diese Multiplikation von Kurvenpunkten mit ganzen Zahlen ist leicht und schnell auszuführen.
  Erforderlicher Rechenaufwand: $\sim 1{,}5 \log_2(k)$ Punktadditionen

- Für die Umkehraufgabe zur Multiplikation von Kurvenpunkten mit ganzen Zahlen steht im Normalfall kein Algorithmus subexponentieller Laufzeit zur Verfügung.

PUNKTMULTIPLIKATION = EINWEGFUNKTION
PUBLIC-KEY-VERFAHREN

# Wie „funktionieren“ Kryptosysteme mit elliptischen Kurven?

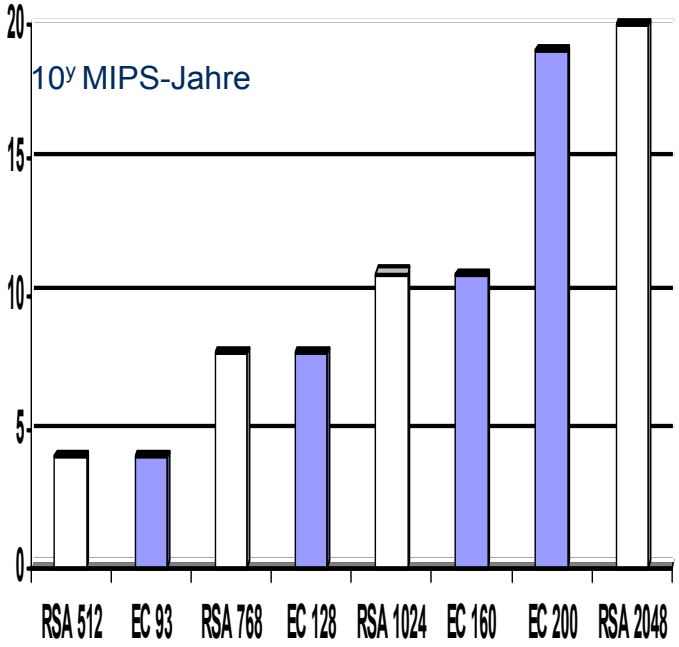
Grundidee (V. Miller, N. Koblitz 1985):

Ersetze in den Kryptoprotokollen zum Diskreten Logarithmus-Problem mod p die diskrete Exponentiation mod p durch die Multiplikation von Punkten auf einer elliptischen Kurve.

Fast alle Verfahren zum DL-Problem mod p lassen sich so auf elliptische Kurven übertragen:

| | | |
|---|---|---|
| ElGamal | → | ElGamal-Analogon für elliptische Kurven |
| | → | ECGDSA |
| Diffie - Hellman | → | Diffie-Hellman-Analogon für elliptische Kurven ( = ECDH) |
| DSA | → | DSA-Analogon für elliptische Kurven ( = ECDSA) |

# Elliptische Kurven und RSA im Vergleich




- Public-Key-Verfahren auf Basis elliptischer Kurven (EC) über endlichen Körpern bieten das beste Verhältnis von "Sicherheit pro Bit“ unter allen heutigen Public-Key-Verfahren.
- Zum Vergleich:

  RSA-512 ~ EC- 93

  RSA-768 ~ EC-128

  RSA-1024 ~ EC-160

  RSA-2048 ~ EC-200
- Parameterverlängerung bei elliptischen Kurven um nur 2 Bit kann bei EC-Verfahren das Sicherheitsniveau verdoppeln.

# Welche elliptische Kurven sind geeignet?

**Anforderungen an Kurve:**

- Kurve $E$ ist über "implementierungsfreundlichem" Körper definiert, etwa $\boldsymbol{F_p}$.
- #(E) (Punktezahl von *E)* muß von einer Primzahl $n > 2^{160}$ geteilt werden ($n \neq p$). (Pollard, Shanks)
- Der Quotient #E /n sollte möglichst klein sein.
- n darf kein Teiler von $p^2 - 1$, $p^3 - 1$, $p^4 - 1$, ... sein. (Frey-Rück, MOV)
- $\#(E) \neq p$. (Smart, Semaev)

**Gewinnung geeigneter elliptischer Kurven:**

- Wesentlich schwieriger als die Erzeugung von RSA-Zahlen.
- Erforderliche Techniken inzwischen verfügbar.

# Beispiel einer geeigneten elliptische Kurve

Die elliptische Kurve $E$ über $\boldsymbol{F_p}$ mit

$$p = 5444517870735016944133051197007201608043,$$

gegeben durch die Gleichung

$$y^2 = x^3 + 2x + 13797622129676014164845224464603762583 86$$

hat die Eigenschaften:

- $\#E = 10 \cdot n$, mit $n = 5444517870735016944280625149596878041 69$,
  n ist eine Primzahl.
- Der Punkt $G = (x, y)$ aus $\boldsymbol{F_p} \times \boldsymbol{F_p}$ mit den Koordinaten

  $x = 8$ und

  $y = 2810764757762282195634100207553387583 11$

  liegt auf der Kurve $E$ und erzeugt eine zyklische Punktegruppe der Ordnung n.
- Die kleinste Zahl der Form $p^t - 1$, die von n geteilt wird, ist

  $$p^{9074196451225028240467708582661463402 8} - 1.$$

# Das Signaturverfahren ECGDSA

| | |
|---|---|
| ***Systemparameter:*** | **Ellipt. Kurve über $F_p$, p prim,** |
| | **Kurvenpunkt G der Ordnung n, n prim** |
| ***privater Schlüssel:*** | **Zufallszahl d < n** |
| ***öffentlicher Schlüssel:*** | **Kurvenpunkt Q = t G, mit t·d = 1 mod n** |

**Hashwert H(m) < n zur Nachricht m**

***Signierer***

Wählt Zufallszahl $k < n$.

Berechnet
$k \cdot P = (P_x, P_y)$
$r = P_x \bmod n$.

Berechnet
$s = (k \cdot r - H(m)) \bmod n$.

→ **m, r, s**

***Verifizierer***

Berechnet
$u_1 = H(m) \cdot r^{-1} \bmod n$
$u_2 = s \cdot r^{-1} \bmod n$
$X = (X_x, Y_y) = u_1 G + u_2 Q$

Verifiziert
$X_x \bmod n = r$.

# Vergleich: ECGDSA - DSA - RSA

| | ECGDSA | DSA 1024 | RSA 1024 (bei e=65537) |
|---|---|---|---|
| Systemparameter | ~ 480 Bit | ~ 2200 Bit | ---------- |
| Privater Schlüssel | 160 Bit | 160 Bit | 1024 Bit |
| Öffentlicher Schlüssel | 161 Bit | 1024 Bit | 1041 Bit |
| Signaturlänge | 320 Bit | 320 Bit | 1024 Bit |
| Erforderliche Arithmetik | 160 Bit | 1024 + 160 Bit | 1024 Bit |

**Vorteile des EC-Ansatzes:**

⇒ Geringere Speicherplatzanforderungen für Schlüssel und Zertifikate

⇒ Kürzere Rechenzeit

⇒ Ablauffähig auf existierenden Krypto-Koprozessoren

# Elliptische Kurven in SELMA

Einsatz in SELMA-Meßgeräten zur Erzeugung digitaler Signaturen von Meßwerten.

Technische Anforderungen an qualifizierte Signatur nach Signaturgesetz werden erfüllt:

- Signaturalgorithmus ECGDSA
- Elliptische Kurven über $\boldsymbol{F_p}$ mit Primzahlen $p \geq 192$ Bit, $n \geq 180$ Bit
- Seitenkanalresistente Implementierung auf Security-IC, z. B. SLE66Cx640P
- Hashfunktion RIPEMD-160
- Physikalischer Zufallszahlengenerator