# Grundlagen der Elliptic Curve Cryptography

*Mathias Schmalisch* *Dirk Timmermann*

Institut für Angewandte Mikroelektronik und Datentechnik, Universität Rostock
Richard-Wagner-Str. 31, 18119 Rostock
Tel.: +49 (381) 498 35 36, Email: mathias.schmalisch@uni-rostock.de

**Abstract.** Im Jahre 1986 wurden von Neal Koblitz [KOB87] und Victor Miller [MIL86] erstmals Möglichkeiten vorgestellt, wie elliptische Kurven in der Kryptographie eingesetzt werden können. Seit dieser Zeit hat sich die sogenannte Elliptic Curve Cryptography (ECC) einen bedeutenden Platz in der Public Key Kryptographie erobert. Dieses Verfahren hat einige entscheidende Vorteile gegenüber den klassischen Verfahren.

## Einführung

Wir leben in einer Zeit, die auch als Informationszeitalter bezeichnet wird. Denn die Information ist inzwischen ein wertvolles Gut. Das heißt auch, daß dieses Gut vor den Zugriff Unbefugter gesichert werden muß. Dies ist insbesondere von Bedeutung, wenn Informationen ausgetauscht werden sollen. Sei es nun durch ein Telefongespräch, über das Internet oder ähnliches, der Austausch geschieht normalerweise über öffentliche Netze. Einen wirksamen Schutz bietet in diesem Zusammenhang die Kryptographie. Dabei werden die Information auf der einen Seite verschlüsselt, dann Übertragen und auf der anderen Seite wieder entschlüsselt. Somit kann sicher gestellt werden, das die Informationen nicht Unbefugten, wie z.B. Industriespionen, Behörden u.a., in die Hände fallen.

In der modernen Kryptographie werden zwei Arten von Verfahren unterschieden. Das sind die symmetrischen und asymmetrischen Verfahren, wobei jede Art ihre Vor- und Nachteile hat. Bei den symmetrischen Verfahren, auch Private Key Verfahren genannt, gibt es nur einen Schlüssel. Dieser wird sowohl zu Ver- als auch zum Entschlüsseln benötigt. Daher ist es wichtig, eine sichere Möglichkeit zu finden, um den Schlüssel an beide Kommunikationsteilnehmer zu verteilen. Im Gegensatz dazu kommen bei den asymmetrischen Verfahren, auch Public Key Verfahren genannt, zwei Schlüssel zum Einsatz. Ein Schlüssel ist der öffentliche und der andere der private Schlüssel. Der öffentliche Schlüssel kann an alle Nutzer verteilt werden, ohne die Sicherheit zu gefährden. Wenn nun jemand Informationen austauschen möchte, verschlüsselt er diese mit dem öffentlichen Schlüssel des Empfängers. Diese verschlüsselte Information kann dann nur mit dem privaten Schlüssel des Empfängers wieder entschlüsselt werden. Dadurch ist der Schlüsselaustausch bei den asymmetrischen Verfahren wesentlich unproblematischer als bei den symmetrischen Verfahren. Allerdings haben die asymmetrischen Verfahren auch einen Nachteil, denn sie sind etwa 100 bis 1000 mal langsamer als vergleichbare symmetrische Verfahren.

Dieser Geschwindigkeitsunterschied hängt unter anderem mit der benötigten Schlüssellänge zusammen. An dieser Stelle kommt der Vorteil der ECC zum tragen, denn sie kommt mit einem wesentlich kleinerem Schlüssel bei gleicher Sicherheit aus. In einer aktuellen Untersuchung [LEN00] wurden verschiedene Verfahren hinsichtlich ihrer Sicherheit miteinander verglichen. Mit Hilfe dieser Untersuchung wurde die Abbildung 1 erstellt, in dieser Abbildung wird das RSA Verfahren [RIV78] dem ECC Verfahren gegenübergestellt. Dabei wird die Sicherheit über das Jahr angezeigt, ab dem das Verfahren mit der entsprechenden Schlüssellänge wahrscheinlich gebrochen werden kann. In der Abbildung erscheint das ECC Verfahren zwei mal, "ECC" und "ECC+". Dabei wird beim "ECC" davon ausgegangen, daß durch Kryptoanalyse keine Möglichkeiten gefunden werden, um das Verfahren schneller zu brechen. Da das ECC Verfahren aber noch relativ neu ist, sollte davon ausgegangen werde, daß die Kryptoanalyse noch einige Möglichkeiten aufdeckt, mit der das ECC Verfahren schneller gebrochen werden kann. In diesem Fall gilt dann der "ECC+" Graph. Weiterhin ist in Abbildung 1 sehr gut zu sehen, daß die Schlüssellänge beim RSA Verfahren wesentlich stärker steigt als beim ECC Verfahren.

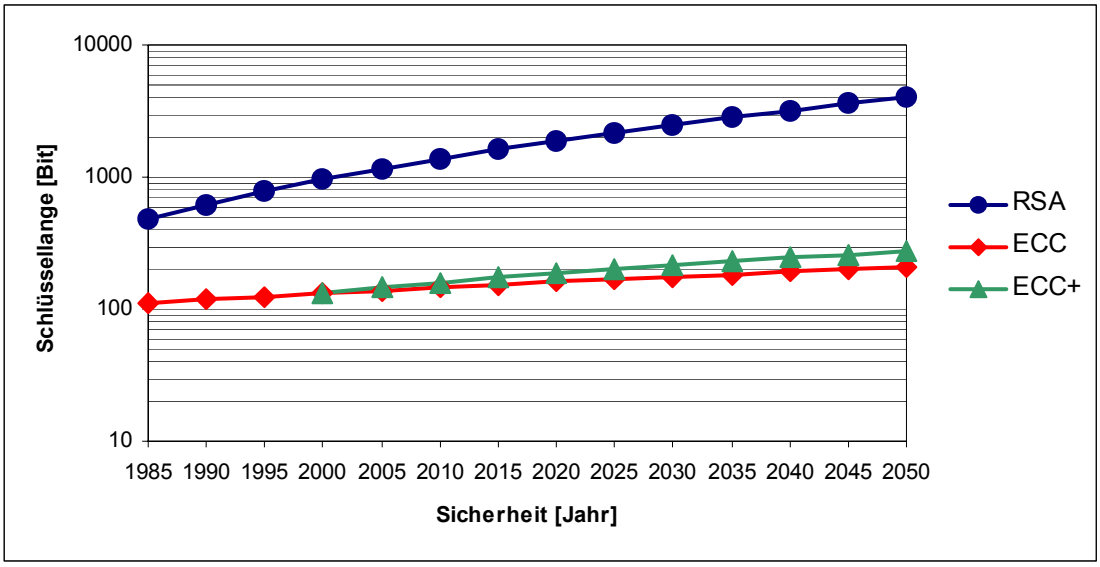


Abbildung 1: Vergleich Schlüssellänge / Sicherheit

Bei der Wahl der Schlüssellänge für die Verschlüsselung von Informationen kommt es darauf an, wie lange diese Information geheimgehalten werden soll. Wenn man die Information nur für eine kurze Zeit von Bedeutung ist, wird auch nur eine kürzerer Schlüssel benötigt. Als Beispiel, die Information soll nur bis zum Jahr 2002 sicher sein, so ist beim RSA Verfahren eine Schlüssellänge von 1028 bzw. beim ECC Verfahren von 139 Bit erforderlich. Ist die Information aber über einen langen Zeitraum zu schützen, z.B. bis ins Jahr 2050, dann sollte beim RSA Verfahren eine Schlüssellänge von 4047 und beim ECC Verfahren von 272 gewählt werden. Wie an diesen Beispielen zu sehen ist, hat sich die Schlüssellänge beim RSA Verfahren über fast 50 Jahre etwa vervierfacht, während sie sich beim ECC Verfahren etwa verdoppelt hat.

Diese Zahlen sind natürlich etwas großzügig gewählt, so ist es bis heute gerade ein einziges mal und mit großem Aufwand gelungen das RSA Verfahren mit einer Schlüssellänge von 512 Bit zu brechen. Mit einer Schlüssellänge von 1024 Bit, wie sie heute üblich ist, sollte das Verfahren auch noch für die nächsten Jahre sicher sein.

## 1. Elliptische Kurven

Elliptische Kurven werden auch als kubische Kurven bezeichnet. Dabei handelt es sich um eine Menge von Punkten (x, y) in der Ebene, deren Koordinaten eine bestimmte Gleichung erfüllen. Die allgemeine Weierstraß-Normalform für eine kubische Kurven lautet:

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6 \tag{1}$$

Diese Form läßt sich unter einigen Voraussetzungen wesentlich vereinfachen. So wird aus der elliptischen Kurve in Gleichung (1) über eine Körper $GF(p^n)$, mit $p \neq 2$ und 3 folgende Kurve E, siehe dazu [MEN93]:

$$E : y^2 = x^3 + ax + b \tag{2}$$

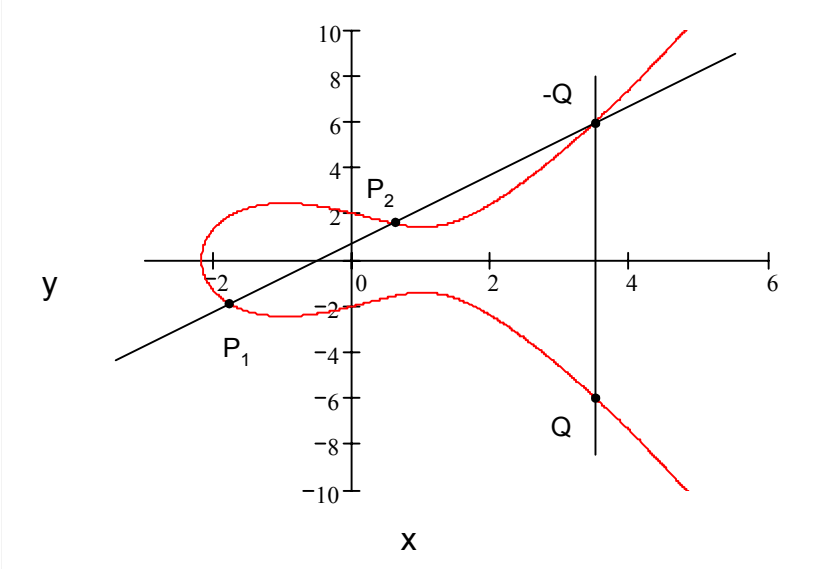


Abbildung 2: Elliptische Kurve $y^2 = x^3 - 3x + 3$

Ein Beispiel einer solche Kurve ist in Abbildung 2 dargestellt, dabei müssen die Zahlen a und b ganze Zahlen sein. Um eine solche Kurve für die Kryptographie einsetzen zu können, muß sie auch einige Bedingungen erfüllen. So darf die Kurve nicht singulär sein. Die Singularität einer Kurve kann mit Hilfe der Diskriminante Δ bestimmt werden.

$$\Delta = -16\left(4a^3 + 27b^2\right) \tag{3}$$

Damit eine Kurve der Gleichung (2) nicht singulär ist, muß die Diskriminante $\Delta \neq 0$ sein.

## 2. Addition zweier Punkte

Mit den Punkten auf einer solchen Kurve E lassen sich mathematische Operationen ausführen. Die einfachste Operation ist die Addition zweier Punkte. Dabei werden die beiden Punkte $P_1 = (x_1, y_1)$ und $P_2 = (x_2, y_2)$ zum Punkt $Q = (x_3, y_3)$ addiert, siehe Abbildung 2. Um den dritten Punkt zu erhalten wird eine Gerade L durch die Punkte $P_1$ und $P_2$ gelegt, die E in einem dritten Punkt schneidet, dieser Punkt ist der inverse Punkt zu Q. Die Gerade L läßt sich durch folgende Gleichung darstellen:

$$L: \; y = \lambda x + \nu \tag{4}$$

Um die Steigung λ der Geraden zu bestimmen, sind zwei Fälle zu betrachten:

1. $P1 \neq P2$, mit $x1 \neq x2$. In diesem Fall ist λ die Steigung der Sekante durch die Punkte $P_1$ und $P_2$. Daher kann hier das Verfahren für die Sekantensteigung angewendet werden.

$$\lambda = \frac{y_1 - y_2}{x_1 - x_2} \tag{5}$$

2. $P_1 = P_2$, mit $y_1 \neq 0$. Dann ist λ die Steigung der Tangente durch den Punkt $P_1$. Die Steigung der Tangente entspricht der ersten Ableitung von E im Punkt $P_1$, so erhält man für λ:

$$\lambda = \frac{3x_1^{\,2} + a}{2y_1} \tag{6}$$

Wie hier gezeigt wurde, hängt die Steigung λ davon ab, ob zwei unterschiedliche Punkte miteinander addiert werden, oder ob eine Punktverdopplung stattfindet, indem ein Punkt mit sich selbst addiert wird.

Aus der Gleichung (2) der Kurve E läßt sich das Polynom $F(x, y) = x^3 + ax + b - y^2$ bilden. Wenn jetzt für y die Geradengleichung (4) eingesetzt wird, erhält man ein Polynom der Form $F(x, \lambda x + \nu)$, das vom Grad 3 in x ist. Da die Punkte $P_1$, $P_2$ und $-Q$ auf der Geraden L und der elliptischen Kurve E liegen (siehe Abbildung 2), sind $x_1$, $x_2$ und $x_3$ Nullstellen des Polynoms $F(x, \lambda x + \nu)$. Durch Zerlegung des Polynoms in seine Linearfaktoren und anschließenden Koeffizientenvergleich der $x^2$-Glieder erhält man die Koordinaten des Punktes Q:

$$\begin{aligned} x_3 &= \lambda^2 - x_1 - x_2 \\ y_3 &= \lambda(x_1 - x_3) - y_1 \end{aligned} \tag{7}$$

Damit ist die Addition zweier Punkte abgeschlossen.

## 3. Endliche Körper

Elliptische Kurven lassen sich nicht nur über unendliche Körper (R, +, *) der reellen Zahlen darstellen, sondern auch über endliche Körper. Bei einem endlichen Körper sind die Anzahl der Elemente beschränkt. Die Anzahl der Elemente bezeichnet man als Ordnung des Körpers. Zu jeder Primzahlpotenz $p^n$, wobei p prim und n ein natürliche Zahl ist, gibt es einen endlichen Körper der Ordnung $p^n$. Zum Beispiel hat der Körper bei n = 1 genau p Elemente. Der endliche Körper der Ordnung $p^n$ wird auch als Galois-Feld $GF(p^n)$ bezeichnet, was gleichbedeutend mit $F_{p^n}$ ist. In der Literatur finden beide Schreibweisen Verwendung, siehe Tabelle 1. Kryptosysteme lassen sich am besten mit den Spezialfällen n = 1 oder p = 2 realisieren.

| | $GF(p^n)$ | $F_{p^n}$ |
|---|---|---|
| n = 1 | GF(p) | $F_p$ |
| p = 2 | $GF(2^n)$ | $F_{2^n}$ |

Tabelle 1: Schreibweisen in der Literatur

Ein endlicher Körper GF(p) ist zum Beispiel der Körper der ganzen Zahlen modulo einer Primzahl p, wie es als Beispiel in Abbildung 3 zu sehen ist. Die Addition zweier Punkte auf dieser Kurve verläuft nach dem selben Schema, wie es im Abschnitt 2 beschrieben wurde. Nur das die Ergebnisse modulo p reduziert werden. Für die Berechnung der Steigung λ wird eine Division benötigt, dies wird im GF(p) durch die Multiplikation mit dem modular Inversen gelöst. Für die Berechnung des modular Inversen wird der erweiterte Euclidische Algorithmus verwendet, wie er in Algorithmus 1 dargestellt ist.

Algorithmus 1: Erweiterter Euclidischer Algorithmus
INPUT: Modulo M, Zahl B
OUTPUT: $B^{-1}$ mod M
S ← M; V ← 0
R ← B; U ← 1
Repeat
    Q ← ⌊S / R⌋
    tmp ← S – Q * R; S ← R; R ← tmp
    tmp ← V – Q * U; V ← U; U ← tmp
Until ( R = 0 )
Return V

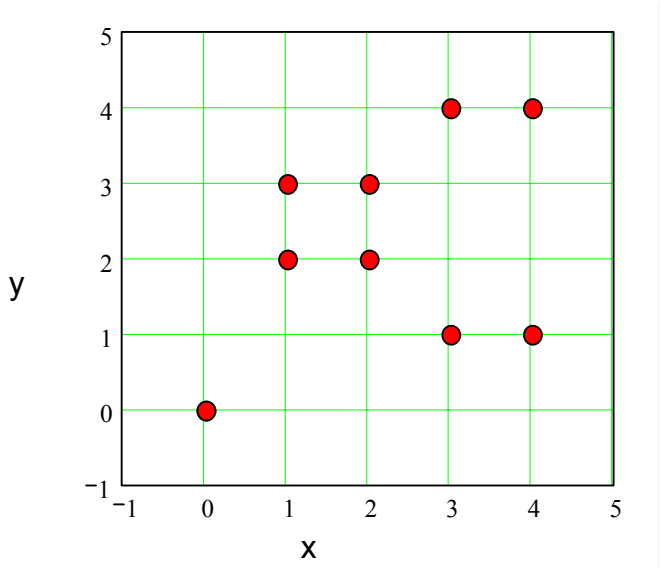


Abbildung 3: Elliptische Kurve $y^2 = x^3 + 3x$ im GF(5)

## 4. Anwendung in der Kryptographie

Im folgenden ein Beispiel für die Anwendung von endlichen elliptischen Kurven in der Kryptographie anhand eines Schlüsselaustausches nach Diffie-Hellmann:

Alice und Bob möchten sich ein Nachricht schicken. Dazu einigen sie sich auf eine elliptische Kurve E und eine Punkt F auf dieser Kurve. Anschließend wählt Alice eine geheime Zahl a und berechnet den Punkt PA = a * F, siehe Algorithmus 2, und veröffentlicht diesen Punkt. Bob wählt ebenfalls eine geheime Zahl b und berechnet den Punkt PB = b * F. Alice multipliziert ihre geheime Zahl a mit dem öffentlichen Punkt von Bob: a * PB. Das Ergebnis ist ein geheimer Schlüssel, der für einen symmetrischen Algorithmus (z.B. DES, AES) verwendet werden kann. Bob kann diese Zahl berechnen, indem er b * PA berechnet, denn:

$$b * PA = b * (a * F) = b * (F * a) = (b * F) * a = a * (b * F) = a * PB \tag{8}$$

Um einen Punkt F mit einer Zahl a zu multiplizieren, kann der nachfolgende Algorithmus angewendet werden:

Algorithmus 2: Brechnung von X = a * P
INPUT: Punkt P, Zahl a
OUTPUT: Punkt X = a * P
If ( $a_0$ = '1' ) Then
    X ← P
Else
    X ← 0
EndIf
For i = 1 To $\log_2(a)$
    P ← 2P  Punktverdopplung
    If ( $a_i$ = '1' ) Then
        X ← X + P  Punktaddition
EndFor
Return X

Wie in diesem Algorithmus zu sehen ist, wird die Multiplikation von Punkt F mit der Zahl a auf Punktadditionen und Punktverdopplungen abgebildet. Damit ein Angreifer die Nachricht von Alice und Bob wieder entschlüsseln kann, muß er aus dem Punkt F und PA die geheime Zahl a berechnen. Die Sicherheit dieses Verfahrens beruht darauf, den diskreten Logarithmus in einer endlichen Gruppe zu berechnen (Discrete Logarithm Problem, DLP). Die einfachste Möglichkeit, um a aus den Punkten F und PA zu berechnen, währe den Punkt F so oft mit sich selbst zu addieren, bis als Ergebnis der Punkt PA rauskommt. Die Anzahl Additionen ergeben dann die Zahl a. Allerdings dauert diese Berechnung sehr viel länger als die Berechnung von PA aus a und F mit dem Algorithmus 2, wenn für den Körper GF(p) der Elliptischen Kurve E und der geheimen Zahl a große Zahlen gewählt werden.

Das in diesen Beispiel vorgestellte Verfahren hat einen großen Vorteil gegenüber dem RSA Verfahren. Denn bei jeder nachfolgenden Kommunikation zwischen Alice und Bob kann eine neue Kurve E, Punkt F und geheime Zahlen a und b gewählt werden, z.B. durch einen Zufallszahlengenerator. Falls es dabei nun einem Angreifer gelingen sollte einen Schlüssel zu berechnen, kann er gerade mal eine einzige Nachricht entschlüsseln. Während er beim RSA Verfahren alle mit den berechneten Schlüssel verschlüsselten Nachrichten entschlüsseln kann. Das hängt damit zusammen, daß die Berechnung der Schlüssel beim RSA Verfahren sehr aufwendig ist, und so nicht bei jeder Kommunikation neu durchgeführt wird. Damit ist die Sicherheit beim ECC Verfahren wesentlich größer. Da ein Angreifer für jede einzelne Nachricht einen immensen Aufwand betreiben muß, um diese entschlüsseln zu können. Wogegen beim RSA Verfahren dieser Aufwand nur ein einziges mal betrieben werden muß.

## 5. Vorteile bei der Hardwarerealisierung

Für die Realisierung von ECC in Hardware eignen sich besonders Kurven über einen Körper von $GF(2^n)$. Da in diesem Fall die Charakteristik gleich 2 ist (p = 2), kann die Gleichung (1) auf folgenden elliptische Kurve vereinfacht werden:

$$\mathrm{E} : y^2 + xy = x^3 + ax^2 + b \tag{9}$$

Der Vorteil für eine Hardwarerealisierung liegt hier in der Arithmetik im Körper von $GF(2^n)$. Bei der Addition zweier Zahlen gibt es keinen Übertrag, wie es bei natürlichen Zahlen der Fall ist. Das heißt, daß zwei Zahlen a und b durch bitweises XOR-Verknüpfen miteinander addiert werden. Dies läßt sich in Hardware besonders schnell realisieren. Dadurch sind positive und negative Zahlen identisch (a = -a), womit aus einer Subtraktion eine Addition wird. Da die Multiplikation auf der Addition beruht, ist diese durch einfaches Schieben und XOR-Verknüpfen zu implementieren. Für die Berechnung des modulare Inversen kann weiterhin der erweiterte Euclidische Algorithmus angewandt werden, allerdings werden die Subtraktion zu Additionen. Eine geeignete Hardwarerealisierung des Euclidischen Algorithmus ist in [BRU93] zu finden.

## 6. Zusammenfassung

In diesem Beitrag wurden die Grundlagen für Verfahren nach der Elliptic Curve Cryptography aufgezeigt. Anhand der Mathematik ist deutlich zu erkennen, das dieses Verfahren wesentlich komplizierter ist als andere asymmetrische Verfahren. Dafür bietet es aber einige Vorteile gegenüber den klassischen asymmetrischen Verfahren, wie z.B. dem RSA Verfahren. Innerhalb der kurzen Zeit, seit dem das ECC Verfahren existiert hat es sich eine bedeutenden Platz in der Kryptographie erobert. In Zukunft wird das ECC Verfahren weiterhin an Bedeutung gewinnen, und daher ist es notwendig die Entwicklung auf diesem Gebiet voranzutreiben.

## Literatur

[BRU93] Brunner, H.; Curiger, A.; Hofstetter, M.:
On computing multiplicative inverses in $GF(2^m)$
IEEE Transactions on Computer, Vol. 42, No. 8, pp. 1010-1015, 1993

[KOB87] Koblitz, N.:
Elliptic curve cryptosystems
Mathematics of Computation, vol. 48, pp. 203--209, 1987

[LEN00] Lenstra, A. K. ; Verheul, E. R.:
Selecting cryptographic key sizes
PKC 2000, Springer-Verlag, Lecture Notes in Computer Science, 2000

[MEN93] Menezes, A.J.:
Elliptic Curve Public Key Cryptosystems
Klumer Academic Publishers, 1993

[MIL86] Miller, V.:
Uses of elliptic curves in cryptography
CRYPTO'85, Springer-Verlag, Lecture Notes in Computer Science, 1986

[RIV78] Rivest, R.L.; Shamir, A.; Adleman, L.:
A Method of obtaining digital signature and public key cryptosystems
Communications of the ACM, Vol.21, pp.120-146, 1978