Seitenkanalattacken gegen Kryptographie auf Elliptischen Kurven

Ausarbeitung zu dem Seminarvortrag im Seminar „Seitenkanalattacken“ im Sommersemester 2005 am Lehrstuhl für Theoretische Informationstechnik der RWTH Aachen

Peter Schwabe

17.05.2005

Zusammenfassung

Seit Victor S. Miller 1985 die Idee vorstellte, Elliptische Kurven in Kryptosystemen zu verwenden [Mil86], wurden mehrere auf diesem Ansatz basierende Verfahren standardisiert, die etwa auch von der NSA zur Verwendung empfohlen werden.
Da diese Systeme im Vergleich zu anderen Verfahren mit relativ kurzen Schlüssellängen arbeiten können, werden sie vor allem auf resourcenbeschränkter Hardware wie etwa Smartcards eingesetzt. Diese Tatsache macht es interessant, zu betrachten, wie sich bekannte Seitenkanalattacken, wie etwa das Ausnutzen von Informationen, die man durch Stromverbrauchsanalyse, Zeitmessung, Messung elektromagnetischer Abstrahlung oder durch gezieltes Induzieren von Fehlern erhält, auf solche Systeme übertragen lassen.
Diese Arbeit wird in der ersten Hälfte den mathematischen Hintergrund vorstellen um dann darauf aufbauend vor allem Seitenkanalattacken vorzustellen, die sich die Messung des Stromverbrauchs zunutze machen.

Inhaltsverzeichnis

Abbildungsverzeichnis

1 Einleitung

Auf Elliptischen Kurven basierende Kryptosysteme werden heutzutage vor allem auf resourcenbeschränkter Hardware eingesetzt. Diese Seminararbeit stellt zunächst den mathematischen Hintergrund dar, geht dann auf einige Details realer Implementierungen ein, um schließlich Seitenkanalattacken gegen eben diese Kryptosysteme vorzustellen. Hierbei wird ein Schwerpunkt auf solche Seitenkanalattacken gelegt, die sich Stromverbrauchanalyse zunutze machen, aber auch auf andere Seitenkanalattacken wird kurz eingegangen.

2 Mathematischer Hintergrund

Dieser Abschitt wird einige mathematische Grundlagen darstellen, die für das Verständnis von Kryptosystemen, die Elliptische Kurven nutzen, notwendig sind.

2.1 Die Gruppenaxiome

Eine Grundlage zur Betrachtung Elliptischer Kurven ist die Kenntnis der Definition einer Gruppe.

Definition 2.1. Sei M eine Menge, $+ : M \times M \to M$ eine Verknüpfung. Dann heißt $(M, +)$ Gruppe, wenn das folgende gilt:

(i) $a + (b + c) = (a + b) + c \quad \forall\, a, b, c \in M$ (Assoziativität)

(ii) $\exists\, e \in M : a + e = a \quad \forall\, a \in M$ (Neutrales Element)

(iii) $\forall\, a \in M\ \exists\, u \in M : a + u = e$ Man bezeichnet dieses inverse Element mit $-a$.

Definition 2.2. Eine Gruppe $(M, +)$ heißt *abelsch* oder *kommutativ*, wenn das folgende gilt:

$$a + b = b + a \quad \forall\, a, b \in M$$

2.2 Endliche Körper

Da — wie wir im folgenden sehen werden — Elliptische Kurven über Körpern definiert werden und da diese Körper in kryptographischen Anwendungen immer endlich sind, seien an dieser Stelle die hierfür wichtigen Aussagen über endliche Körper zusammengefasst, es wird hierbei davon ausgegangen, dass die Definition eines Körpers bekannt ist.
Ist $p \in \mathbb{Z}$ prim, so ist $\mathbb{Z}/p\mathbb{Z}$ ein endlicher Körper mit p Elementen. Einen Körper mit p Elementen bezeichnet man mit $\mathbb{F}_p$ oder mit $GF(p)$, das 'F' steht in beiden Bezeichnugnen für die englische Bezeichung *field*, das 'G' in letzterer Bezeichnung geht auf den Mathematiker Galois zurück, der die Theorie der endlichen Körper begründete.
Ist p prim und f ein irreduzibles Polynom vom Grad k über $\mathbb{F}_p$, dann ist $\mathbb{F}_p[X]/f \cdot\ \mathbb{F}_p[X]$ ein Körper mit p^k Elementen, hier ist die Bezeichnung F_{p^k} bzw. $GF(p^k)$ üblich. Für festes p und k sind alle Körper mit p^k Elementen isomorph zueinander. Die Charakteristik eines Körpers mit p^k Elementen ist p.

2.3 Elliptische Kurven

Definition 2.3. Sei K Körper und seien $a_1, a_2, a_3, a_4, a_6 \in K$. Dann definiert die folgende Gleichung eine Elliptische Kurve:

$$E : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$$

Bemerkung 2.4. Hat K Charakteristik > 3, dann lässt sich die Gleichung aus Definition 2.3 vereinfachen zu:

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

Hierbei sind $a, b \in K$.
Im folgenden werde ich mich ausschließlich auf diese vereinfachte — aber weniger allgemeine Form — beziehen und darauf aufmerksam machen, wenn Aussagen nicht mehr auf die in Definition 2.3 beschriebene allgemeine Form zutreffen.

Definition 2.5. Sei L ein Erweiterungskörper von K, E eine Elliptische Kurve über K. Dann heißt

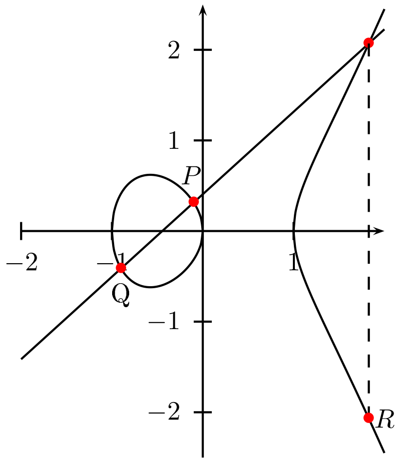
$$E(L) = \{(x, y) \in L \times L \mid y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6\} \cup \{\infty\}$$

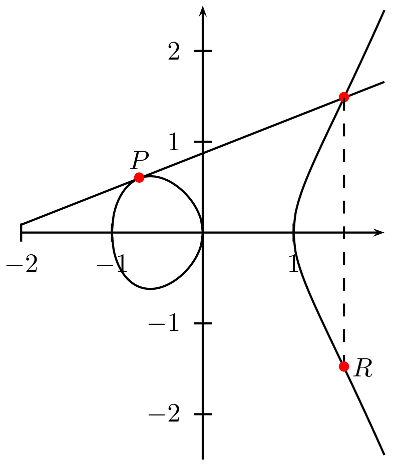
die Menge L-rationaler Punkte auf E. Das mit ∞ bezeichnete Element ist der imaginäre unendlich ferne Punkt.

Auf diese Menge L-rationaler Punkte lässt sich eine Verknüpfung $+$ definieren, die die in 2.1 und 2.2 beschriebenen Kriterien erfüllt, das neutrale Element ist hierbei ∞ — somit erhalten wir die Gruppenstruktur $(E(L), +)$. Wie diese aus historischen Gründen additiv geschriebene Verknüpfung definiert ist, beschreibt der folgende Abschnitt.

2.3.1 Das Gruppengesetz von Elliptischen Kurven

Im Falle von unendlichen Körpern, wie etwa $\mathbb{R}$ hat die Addition von Elementen aus $E(L)$ eine graphische Interpretation, die wie folgt aussieht:
Werden zwei verschiedene Punkte auf der Kurve miteinander addiert, so ist das Ergebnis der Addition der an der Abszissenachse gespiegelte dritte Schnittpunkt der Gerade durch die beiden Punkte mit der Kurve. Abbildung 2.3.1 zeigt die Addition $P + Q = R$. Existiert kein dritter Schnittpunkt mit der Kurve — dies ist genau dann der Fall, wenn sich Q und P nur in der y-Koordinate unterscheiden, so erhält man als Ergebnis ∞, daraus ergibt sich zu einem Punkt $P = (x_P, y_P)$ direkt der inverse Punkt $-P = (x_P, -y_P)$.
Die Addition eines Punktes P zu sich selbst wird als Punktverdopplung bezeichnet und hat eine ähnliche graphische Interpretation wie die Addition zweier verschiedener Punkte. Hier ist das Ergebnis der Berechnung $2 \cdot P = R$ der an der Abszissenachse gespiegelte Schnittpunkt der Tangente durch P mit der Kurve. Diese Verdopplung ist in Abbildung 2.3.1 dargestellt. In kryptographischen Anwendungen werden ausschließlich Elliptische Kurven über endlichen Körpern betrachtet, hierzu ist es nötig, die Punktaddition in einer allgemeinen, von dem Körper K unabhängigen Art zu definieren. Für Kurven über Körpern der Charakteristik > 3 geben die folgenden Definitionen Formeln für Punktaddition und -verdopplung:





Definition 2.6. Sei K Körper, $char(K) > 3$, L ein Erweiterungskörper von K, E eine Elliptische Kurve über K und seien $P = (x_p, y_p)$ und $Q = (x_Q, y_Q)$ zwei Elemente aus $E(L)$ mit $x_P \neq x_Q$ und $P, Q \neq \infty$. Dann sind die Koordinaten des Punktes $P + Q = R = (x_R, y_R)$ gegeben durch:

$$x_R = \left(\frac{y_Q - y_P}{x_Q - x_P}\right)^2 - x_P - x_Q$$

$$\text{und } y_R = \left(\frac{y_Q - y_P}{x_Q - x_P}\right)(x_P - x_R) - y_P$$

Definition 2.7. Sei K Körper, $char(K) > 3$, L ein Erweiterungskörper von K, E eine Elliptische Kurve über K und seien $P = (x_p, y_p) \in E(L)$, $P \neq \infty$ und $y_p \neq 0$. Dann sind die Koordinaten des Punktes $2 \cdot P = R = (x_R, y_R)$ gegeben durch:

$$x_R = \left(\frac{3x_P^2 + a}{2y_P}\right)^2 - 2x_P$$
$$\text{und } y_R = \left(\frac{3x_P^2 + a}{2y_P}\right)(x_P - x_R) - y_P$$

Für ein Element $P \in E(L)$, dessen y-Koordinate 0 ist, gilt $2 \cdot P = \infty$, weiterhin gilt dass $(x_P, y_P) + (x_P, -y_P) = \infty$ und $P + \infty = P \ \forall P \in E(L)$.

3 Implementation von Elliptischen Kurven

In dem vorigen Abschnitt sind zwar prinzipiell alle Grundlagen gelegt, die man benötigt, um die Gruppenstruktur $(E(L), +)$ zu implementieren, dieser Abschnitt geht aber nun darauf ein, wie dies effizient geschehen kann und legt so weitere Grundlagen für das Verständnis von Seitenkanalattacken gegen Kryptographie auf Elliptischen Kurven.
Weiterhin wird in diesem Teil erklärt, inwiefern sich die vorgestellte Gruppenstruktur in Kryptosystem nutzen lässt.

3.1 Darstellung in Projektiven Koordinaten

Betrachtet man die obigen Formeln für Punktaddition und -verdopplung unter Gesichtspunkten des Berechnungsaufwandes, so stellt man fest, dass die aufwändigste Operation in der Regel die multiplikative Invertierung ist. Es stellt sich also die Frage, ob man diese Operation vermeiden kann.
Dieser Abschnitt stellt eine Darstellung von Punkten vor, mit Hilfe derer genau diese Einsparung erreicht werden kann: Die Darstellung in projektiven Koordinaten.
Die Idee ist hierbei die folgende: Seien $c, d \in L^*$ beliebig. Für jedes $\theta \in L^*$ lässt sich der Punkt $(x, y) \in L \times L$ auch darstellen als $(\theta^c x : \theta^d y : \theta)$. Für die Darstellung von Punkten auf Elliptischen Kurven über Körpern der Charakteristik > 3 hat sich die Wahl von $c = 2$ und $d = 3$ als besonders günstig erwiesen, diese Wahl führt zu der Darstellung in Jacobischen projektiven Koordinaten:
Substituiert man in der eine Elliptische Kurve definierenden Gleichung $y^2 = x^3 + ax + b$ die Variable x durch $\frac{X}{Z^2}$ und y durch $\frac{Y}{Z^3}$ so erhält man die Gleichung

$$\frac{Y^2}{Z^6} = \frac{X^3}{Z^6} + a\frac{X}{Z^2} + b$$

Multipliziert man diese auf beiden Seiten mit Z^6 so erhält man

$$Y^2 = X^3 + aXZ^4 + bZ^6$$

Ein Punkt auf der Kurve kann nun dargestellt werden durch $(X : Y : Z)$, diese projektive Darstellung entspricht der affinen Darstellung $(\frac{X}{Z^2}, \frac{Y}{Z^3})$. Das Z kann in der projektiven Darstellung frei aus $L*$ gewählt werden, die projektiven Koordinaten $(1 : 1 : 0)$ sind eine Darstellung für den unendlich fernen Punkt. Für Punkte, die in Jacobischen projektiven Koordinaten vorliegen ergibt sich folgende Formel für die Addition:

Seien $P = (X_P : Y_P : Z_P)$ und $Q = (X_Q, Y_Q, 1)$ Elemente aus $E(L)$ mit $Z_P \neq 0$ und $P \neq \pm Q$. Die Summe $P + Q = R = (X_R : Y_R : Z_R)$ ist gegeben durch:

$$
\begin{aligned}
X_R &= (Y_Q Z_P^3 - Y_P)^2 - (X_Q Z_P^2 - X_P)^2 (X_P + X_Q Z_P^2) \\
\text{und } Y_R &= (Y_Q Z_P^3 - Y_Q)(X_P (X_Q Z_P^2 - X_P)^2) - X_R) - Y_P (X_Q Z_P^2 - X_P)^3 \\
\text{und } Z_R &= (X_Q Z_P^2 - X_P) Z_P
\end{aligned}
$$

Diese Addition lässt sich durch geschicktes Speichern einiger Zwischenergebnisse mit 8 Multiplikationen und 3 Quadrierungen in dem Körper L berechen (siehe [HMV04]), wie gewünscht ist keine Invertierung mehr notwendig.
Wie wir im nächsten Abschnitt sehen werden, ist die Einschränkung $Z_Q = 1$ in den Algorithmen, die für Kryptosysteme notwendig sind, ohne großen Aufwand erreichbar, allgemeine Formeln für die Addition von Punkten in Jacobischen projektiven Koordinaten finden sich in [CAMTO98].
Für die Verdopplung eines Punktes erhält man folgende Formel: Sei $P = (X_P : Y_P : Z_P) \in E(L), P \neq -P$, dann sind die Koordinaten von $2 \cdot P = R = (X_R : Y_R : Z_R)$:

$$
\begin{aligned}
X_R &= (3X_P^2 + aZ_1^4)^2 - 8X_P Y_P^2 \\
\text{und } Y_R &= (3X_P^2 + aZ_1^4)(4X_P Y_P^2 - X_R) - 8Y_1^4 \\
\text{und } Z_R &= 2Y_P Z_P
\end{aligned}
$$

Diese Berechnung lässt sich ebenfalls durch geschicktes Speichern einiger Zwischenergebnisse mit 4 Multiplikationen und 6 Quadrierungen in L berechnen, es ist also eine Körperoperation weniger notwendig, als bei der Addition (siehe [HMV04]). Hinzu kommt, dass sich Quadrierungen in einigen endlichen Körpern etwas schneller berechnen lassen als Multiplikationen und dass im Falle einer Addition mehr Zwischenergebnisse gespeichert werden müssen. Eine Addition ist also, wenn man Jacobische projektive Koordinaten verwendet, signifikant aufwändiger als eine Verdopplung, eine Tatsache, die man für Seitenkanalattacken ausnutzen kann.

3.2 Das DLP für Elliptische Kurven

Bevor man sich Seitenkanalattacken gegen Kryptographie auf Elliptischen Kurven widmen kann, stellt sich noch die Frage, wie solche Kryptosysteme aussehen können. Aus Kryptosysteme wie etwa dem ElGamal-System oder auch dem Diffie-Hellman Schlüsselaustauschprotokoll ist das Diskrete Logarithmus Problem, kurz DLP, über Z_n^* recht bekannt. Dieses Problem sieht wie folgt aus:
Gegeben $n \in \mathbb{N}, a \in \mathbb{Z}_n^*, a \neq 1$ und $b \in \langle a \rangle$, bestimme das kleinste x, so dass $a^x = b \mod n$.
In der Regel wählt man a als eine Primitivwurzel $\mod n$, so dass $\langle a \rangle = \mathbb{Z}_n^*$, auf jeden Fall muss man aber darauf achten, dass $|\langle a \rangle|$ „groß genug“ ist, da von dieser Größe die Schwierigkeit des Problems direkt abhängt.
Da für die Formulierung dieses Problems nicht mehr als eine endliche Gruppenstruktur benötigt wird, lässt es sich direkt auf die Gruppe $(E(L), +)$ übertragen, dies sieht wie folgt aus:
Seien K Körper, E eine Elliptische Kurve über K, L ein Erweiterungskörper von K. Seien weiterhin $P \in E(L), P \neq \infty$ und $Q \in \langle P \rangle$. Bestimme das kleinste

$k \in L$, so dass $k \cdot P = Q$.
Auch hierfür ist es entscheidend, dass $|\langle P \rangle|$ groß ist, in der Regel wählt man P als einen Erzeuger der größten zyklischen Untergruppe von $E(L)$.
Um das DLP in Elliptischen Kurven für kryptographische Systeme nutzen zu können, ist es erforderlich, die Berechnung $k \cdot P$ auch für große k effizient durchführen zu können, dies geschieht in der Regel mit der additiven Variante des „Square-and-multiply"-Algorithmus, die als „Double-and-Add"-Algorithmus bezeichnet wird: In dem auf diese Weise formulierten Algorithmus geht man davon

Algorithm 1 Double-and-Add

INPUT: $P \in E(L), k \in \mathbb{Z}, k = (k_{n-1} \dots k_0)_2$
OUTPUT: $k \cdot P$
1: $Q \leftarrow P$
2: **for** $i = n - 2$ to 0 **do**
3:     $Q \leftarrow 2 \cdot Q$
4:     **if** $k_1 = 1$ **then**
5:         $Q \leftarrow Q + P$
6:     **end if**
7: **end for**

aus, dass das Bit k_{n-1}, also das höchststellige Bit, den Wert 1 hat, mögliche führende 0-Bits sind nicht berücksichtigt. Man sieht an dieser Stelle auch, warum man in Abschnitt 3.1 bei der Addition von $Z_Q = 1$ ausgehen konnte — vor der skalaren Multiplikation wird man P in genau diese Form bringen, die dann während der Berechnung nicht mehr verändert werden muss.

4 Seitenkanalattacken

Nachdem in den vorigen Abschnitten sowohl die mathematischen als auch die algorithmischen Grundlagen für die Betrachtung von Seitenkanalattacken gegen Kryptographie auf Elliptischen Kurven gelegt wurden, wird dieser Abschnitt einige Attacken, die vor allem auf der Stromverbrauchsanalyse basieren, beschreiben. Der Gedanke bei allen diesen Attacken ist es, durch Beobachtungen, die während der Berechnung von $k \cdot P$ gemacht werden, Rückschlüsse auf k zu ziehen.

4.1 Simple Power Analysis Attacken

Wird das Vielfache eines Punktes auf einer Elliptischen Kurve mit Algorithmus 1 berechnet, so wird die Addition in Zeile 5 in dem Schritt, in dem i den Wert l hat, genau dann ausgeführt, wenn $k_l = 1$ ist. Gehen wir davon aus, dass Punkte in Jacobischen Projektiven Koordinaten dargestellt sind, dass also eine Punktaddition signifikant aufwändiger ist als eine Punktverdopplung, so sollte es die Aufzeichung des Stromverbrauchs über der Zeit während der Berechnung von $k \cdot P$ ermöglichen, den geheimen Schlüssel k zu bestimmen. Diese Vermutung wurde in [GG03] bestätigt, aus diesem Paper stammt auch Abbildung 1. Diese Abbildung zeigt den Stromverbrauch über der Zeit bei der Berechnung von $k \cdot P$, man erkennt deutlich die mit 'S' gekennzeichneten kürzeren Abschnitte, in denen eine Verdopplung stattfindet und die mit 'A' gekennzeichneten Additionschritte, der in diesem Ausschnitt verwendete Schnlüsselabschnitt ist also 0101.

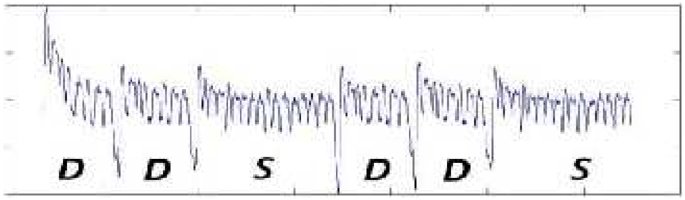


Abbildung 1: Stromverbrauch eines SC140DSP Prozessors über der Zeit bei der Berechnung von Punktverdopplungen (D) und -additionen (S)

4.1.1 Gegenmaßnahmen

Dass ein so einfach per Seitenkanalattacke angreifbarer Algorithmus in sicherheitsrelevanten Bereichen nicht eingesetzt werden kann, ist offensichtlich, es stellt sich also die Frage nach geeigneten Gegenmaßnahmen.
Eine mögliche Gegenmaßnahme besteht darin, für Punktaddition und -verdopplung identische mathematische Formeln zu verwenden. Wie dies möglich ist, wurde von Clavier und Joyce in [CJ01] 2001 gezeigt.
In realen Implementierungen kommen allerdings vorwiegend zwei andere Methoden zum Einsatz, zum einen eine abgewandelte Form des „Double-and-Add"-Algorithmus, zum anderen das Einfügen von Dummy-Operationen in die Berechnung.
Damit der direkte Zusammenhang zwischen Additionsoperationen in der Berechnung von $k \cdot P$ und dem Auftreten von 1-Bits in der Binärdarstellung von k aufgehoben wird, verwendet man überlicherweise einen Algorithmus, der die Addition $Q + P$ immer ausführt und nur abhängig von dem Wert des Bits k_i entscheidet, mit welchem Wert weiterzurechnen ist, dieser Algorithmus heißt „Double-and-Add-always"-Algorithmus:

Algorithm 2 Double-and-Add-always

INPUT: $P \in E(L), k \in \mathbb{Z}, k = (k_{n-1} \dots k_0)_2$
OUTPUT: $k \cdot P$
1: $Q_0 \leftarrow P$
2: **for** $i = n - 2$ to 0 **do**
3: $\quad Q_0 \leftarrow 2 \cdot Q_0$
4: $\quad Q_1 \leftarrow Q_0 + P$
5: $\quad Q_0 \leftarrow Q_{k_i}$
6: **end for**

Dieser Algorithmus ist zwar aufgrund der höhren Anzahl von Additionen weniger effizient, bietet aber einen Schutz vor der oben geschilderten Attacke.
In der Arbeit von Gebotys und Gebotys [GG03] wird gezeigt, dass auch das Einfügen von Dummy-Operationen diese Art von Attacken vermeiden oder zumindest stark erschweren kann, Abbildung 2 zeigt, wie etwa die Powertrace aussehen kann, wenn Dummy-Operationen so eingefügt werden, dass eine Addition

so aussieht wie zwei Verdopplungen.

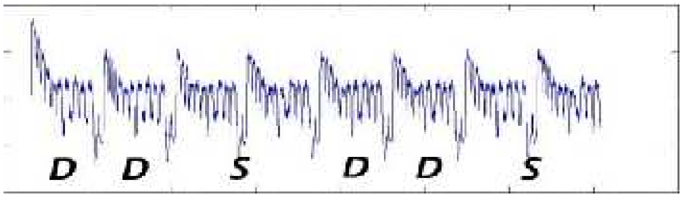


Abbildung 2: Stromverbrauch eines SC140DSP Prozessors über der Zeit bei der Berechnung von Punktverdopplungen (D) und -additionen (S), wenn Dummy-Operationen dafür sorgen, dass eine Punktaddition so aussieht wie zwei Punktverdopplungen

4.2 Differential Power Analysis Attacken

Attacken, die wie die oben vorgestellte auf der Aufzeichnung des Stromverbrauchs von nur einer Berechnung beruhen, bezeichnet man in der Regel als Simple Power Analysis Attacken, kurz SPA-Attacken. Dem gegenüber stehen Differential Power Analysis (DPA) Attacken. Die Idee ist hier die folgende: Ein Angreifer rät einen Teil des geheimen Schlüssels (in der Regel nur ein Bit) so, dass er basierend auf diesem geratenen Schlüsselteil den Anfangsteil der Berechnung durchführen kann. Werden nun viele Berechnungen mit demselben geheimen Schlüssel durchgeführt, so kann der Angreifer ein nur von dem geratenen und ggfls. bereits bekannten Teil des Schlüssels abhängiges Zwischenergebnis ebenfalls berechnen. Nun teilt man die Menge der aufgezeichneten Powertraces nach dem Wert eines Bits dieses geratenen Zwischenergebnisses in zwei Teile. Für jeden dieser Teile bildet man die Mittelung der Powertraces und berechnet anschließend die Differenz dieser beiden Mittelungen, das Ergebnis bezeichnet man als differentielle Powertrace. Wenn nun der Schlüsselteil falsch geraten war, dann ist die Sortierung der Powertraces nahezu zufällig, war er aber richtig geraten, und hat das eine Bit, nach dem die Powertraces sortiert wurden, einen Einfluss auf den Stromverbrach, so kann man in der differentiellen Powertrace eine deutliche Spitze erwarten.
Wie nun lässt sich diese Idee einer Attacke auf Kryptographie auf Elliptischen Kurven anwenden? Angenommen ein Angreifer zeichnet den Stromverbrauch von den Berechnungen $k \cdot P_1 \ldots k \cdot P_n$ auf, die mit Algorithmus 2 durchgeführt wurden. Das Bit k_{n-1} hat per Voraussetzung den Wert 1, angegriffen werden soll Bit k_{n-2}.
Der Angreifer rät nun, dass dieses Bit den Wert 0 hat. Wenn diese Annahme korrekt ist, so wird in Zeile 3 des Algorithmus 2 im zweiten Schleifendurchlauf $4 \cdot P$ berechnet. Also berechnet der Angreifer $4 \cdot P_1 \ldots 4 \cdot P_n$ und sortiert die aufgezeichneten Powertraces nach dem Wert eines Bits der Bitrepräsentation des jeweiligen Punktes $4 \cdot P_i$. Nun bildet er wie oben angegeben die differentielle Powertrace und überprüft, ob sie eine Spitze enthält, ist dies der Fall, so hat er das Bit k_{n-2} richtig geraten, anderenfalls war die Vermutung falsch.

4.2.1 Gegenmaßnahmen

Auch die Möglichkeit einer solchen — etwas aufwändigeren — Attacke muss für viele Einsatzbereiche vermieden werden und der Ansatzpunkt für algorithmische Gegenmaßnahmen besteht darin, dass ein Angreifer die interne Darstellung der Punkte $4 \cdot P_i$ kennen muss, um die Powertraces sortieren zu können. Folgt man Kerkhoffs Prinzip, so kann man davon ausgehen, dass die Implementation des Kryptosystems offenliegt, die Frage ist also, wie man dennoch dafür sorgen kann, dass dieses entscheidende Wissen dem Angreifer nicht zugänglich ist. Hierfür gibt es drei mögliche Verfahren:

Zufällige Projektive Koordinaten Die Idee dieser Gegenmaßnahme ist es, den Ausganspunkt $P = (x, y)$ zufällig in projektive Koordinaten zu überführen, die Schritte des Verfahrens sehen also wie folgt aus:

- Wähle zunächst ein $\theta \in L^*$ zufällig und bilde die Jacobische Projektive Form $(\theta^2 x : \theta^3 y : \theta)$ des Punktes P
- Berechne $(X_R : Y_R : Z_R) = kP$ mit diesen zufälligen Projektiven Koordinaten.
- Transformiere zurück in affine Koordinaten: $kP = (X_R/Z_R^2, Y_R/Z_R^3)$.

Man sieht, dass die interne Darstellung des Punktes $4 \cdot P$ von der Wahl von θ abhängt. Dies ist aber zufällig aus einer hinreichend großen Menge gewählt, die oben beschriebene Seitenkanalattacke ist also nicht mehr möglich.

Zufällige Kurven-Isomorphismen Auch bei dieser Gegenmaßnahme wählt man zufällig ein θ aus L^*, diesmal allerdings um eine isomorphe Kurve E' zu finden und die Berechnungen auf dieser Kurve durchzuführen. Gegeben eine Kurve der Form $E : y^2 = x^3 + ax + b$, also eine Kurve über einem Körper der Charakteristik > 3 läuft die Berechnung wie folgt ab:

- Wähle ein $\theta \in L^*$ zufällig und berechne: $a' = \theta^{-4}a$ und $b' = \theta^{-6}b$
- Man erhält eine isomorphe Kurve $E' : y^2 = x^3 + a'x + b'$
- Dieser Kurvenisomorphismus induziert den Gruppenisomorphismus $\varphi : E(L) \to E'(L), \begin{cases} \infty \mapsto \infty \\ (x, y) \mapsto (\theta^{-2}x, \theta^{-3}y) \end{cases}$
- Berechne also $R' = k * \varphi(P)$ in der isomorphen Gruppe $E'(L)$.
- Berechne $R = \varphi^{-1}(R')$

Klar ist, dass auch hier die interne Darstellung von Zwischenergebnissen einem Angreifer nicht bekannt ist. Entscheidend für diese Gegenmaßnahme ist allerdings, dass sie sich nicht auf Elliptische Kurven über Körpern der Charakteristik 2 einsetzen lässt. Während nämlich alle Kurvenisomorphismen von Kurven über Körpern der Charakteristik > 3 die oben angegebene Form haben [JCT01], haben Kurvenisomorphismen von Kurven über Körpern der Charakteristik 2 die Eigenschaft, dass die x-Koordinate der Punkte unverändert bleibt.

Zufällige Körper-Isomorphismen Wie in Abschnitt 2.2 erläutert, wird ein endlicher Körper mit p^k Elementen von einem irreduziblen Polynom über $\mathbb{F}_p$ von Grad k generiert. Unterschiedliche Polynome vom Grad k über $\mathbb{F}_p$ definieren unterschiedliche, aber isomorphe Körper. Idee dieser Gegenmaßnahme ist es also, alle Körperoperationen in einem zufälligen, zu L isomorphen Körper zu berechnen, genauer gesagt geht dies folgendermaßen:

- Der Körper L mit p^m Elementen wird generiert von einem irreduziblem Polynom f über $\mathbb{F}_p$
- Finde ein anderes irreduzibles Polynom f' von Grad k über $\mathbb{F}_p$, dies generiert den zu L isomorphen Körper L', der Körperisomorphismus von L nach L' sei φ.
- Berechne E' über K' mit Parametern $\varphi(a)$ und $\varphi(b)$
- Berechne $P' = (\varphi(x), \varphi(y))$ und $k \cdot P' = (x_{R'}, y_{R'})$ in $E'(L')$
- Berechne schließlich $R = (\varphi^{-1}(x_{R'}), \varphi^{-1}(y_{R'}))$

Die Frage, wie man zufällig ein anderes irreduzibles Polynom finden kann und wie man daraus den Körperisomorphismus $\varphi : L \to L'$ bestimmt, beantwortet [JCT01].

4.3 Goubins Attacke

Obwohl etwa in [BMJ02] gesagt wird, dass aufgrund der im vorigen Abschnitt vorgestellten Gegenmaßnahmen DPA-Attacken keine echte Bedrohung für Kryptographie auf Elliptischen Kurven darstellen, gibt es eine Möglichkeit, Kryptosysteme mit einer DPA-ähnlichen Attacke anzugreifen, selbst wenn diese durch eine der DPA-Gegemaßnahmen sowie durch Double-and-Add-always geschützt werden. Diese Attacke wird detailliert in [Gou03] beschrieben, in etwas verkürzter Form soll die Idee auch hier erläutert werden.
Hat eine Elliptische Kurve einen Punkt, dessen eine Koordinate Null ist, der also die Form $(x, 0)$ oder $(0, y)$ hat, so wird diese Koordinate bei keiner der drei oben vorgestellten Transformationen verändert. Die Attacke basiert genau hierauf, man geht also davon aus, dass ein solcher Punkt existiert, nennen wir ihn P_0. Gehen wir weiter davon aus, dass Algorithmus 2 für die Punktvervielfachung verwendet wird und dass sich die Attacke gegen das Bit k_i richtet. In dem Schleifendurchlauf, in dem die Laufvariable den Wert i hat, ist der für Q_0 berechnete Wert in der letzten Zeile genau

$$Q_0 = \left(\sum_{j=i+1}^{n-1} k_j 2^{j-i} \quad + k_i \right) \cdot P$$

Hat das Bit k_i den Wert 0, so wird im folgenden

$$Q_1 = \left(\sum_{j=i+1}^{n-1} k_j 2^{j-i+1} \quad + 1 \right) \cdot P$$

berechnet, hat das Bit k_i allerdings den Wert 1, so sieht der Wert von Q_1 wie folgt aus:

$$Q_1 = \left(\sum_{j=i+1}^{n-1} k_j 2^{j-i+1} \quad + 3 \right) \cdot P$$

Um den Wert des Bits k_i zu bestimmen, rät der Angreifer zunächst diesen Wert, nehmen wir an, er rät $k_i = 0$.
Nun invertiert der Angreifer $\left(\sum_{j=i+1}^{n-1} k_j 2^{j-i+1} \;\; + 1 \right)$ modulo der Gruppenordung von $E(L)$, multipliziert dieses Inverse mit P_0 und erhält P_1:

$$P_1 = \left[\left(\sum_{j=i+1}^{n-1} k_j 2^{j-i+1} \quad + 1 \right)^{-1} \mod |E(L)| \right] \cdot P_0$$

Nun führt der Angreifer sehr häufig die Berechnung $k \cdot P_1$ durch und zeichnet währendessen den Stromverbrauch auf — zu beachten ist hierbei, dass die Powertraces deutlich unterschiedlich sein dürften, da man davon ausgeht, dass eine der DPA-Gegenmaßnahmen verwendet wird. Hat der Angreifer den Wert des Bits k_i richtig geraten, so wird im Schleifendurchlauf, in dem die Laufvariable gerade den Wert $i - 1$ hat, der Punkt P_0 als Zwischenergebnis berechnet. Bestimmt man nun die Mittelung aller Powertraces, so erkennt man — aufgrund der in allen diesen Berechnungen identischen 0-Koordinate — an einer Stelle eine Spitze, hat man falsch geraten, so sollte die Mittelung aufgrund der zufälligen Transformationen keinerlei Spitzen aufweisen, man kann also überprüfen, ob man den Wert des Bits richtig geraten hat.

Bemerkungen Es bleibt noch die Frage, welche Elliptischen Kurven überhaupt Punkte der Form $(0, y)$ bzw. $(x, 0)$ haben. Eine weitere Frage ist, ob sich $\left(\sum_{j=i+1}^{n-1} k_j 2^{j-i+1} \;\; + 1 \right)$ überhaupt modulo der Gruppenordnung von $E(L)$ invertieren lässt. Goubin betrachtet, um diese Frage zu beantworten, von FIPS, ANSI, ISO/IEC und im WTLS standardisierte Kurven und stellt fest, dass die meisten Kurven Punkte besitzen, deren eine Koordinate 0 ist. Weiterhin stellt er fest, dass nahezu alle Gruppen die Ordung $q, 2q, 4q$ oder $6q$ für eine Primzahl q haben. In den ersten drei Fällen ist klar, dass man den obigen Wert modulo der Gruppenordnung invertieren kann, da dieser ungerade und also relativ prim zu $q, 2q$ und $4q$ ist. Bei $6q$ könnte natürlich der Fall vorliegen, dass der Wert durch 3 teilbar ist, in diesem Falle kann man allerdings raten, dass das Bit den Wert 1 hat, dann muss man $\left(\sum_{j=i+1}^{n-1} k_j 2^{j-i+1} \;\; + 3 \right)$ modulo der Gruppenordnung invertieren, was möglich ist, da von zwei aufeinanderfolgenden ungerade Zahlen nur eine durch 3 teilbar sein kann.
Als offensichtliche Gegenmaßnahme gegen diese Attacke bietet es sich an, Kurven zu verwenden, die keine Punkte haben, deren eine Koordinate Null ist.

4.4 Andere Seitenkanalattacken

Bisher wurden ausführlich Attacken dargestellt, die auf der Analyse des Stromverbrauchs basieren, es gibt aber auch Seitenkanalattacken gegen Kryptographie auf elliptischen Kurven, die andere Ansätze verfolgen.

Timing-Attacken, also Angriffe, die auf Zeitmessungen während der Operation mit dem geheimen Schlüssel basieren, konnten in der Vergangenheit nicht erfolgreich gegen Kryptosysteme eingesetzt werden, die auf Elliptischen Kurven basieren, hinzu kommt hier die Einschränkung, dass sie sich — ähnlich wie DPA-Attacken — nicht als Attacken gegen Signaturverfahren eignen, da bei Signaturverfahren mit einem bei jeder Signaturerzeugung neu gewählten Sessionkey gearbeitet wird und also die Aufzeichnung von Informationen bei vielen Berechnungen mit demselben Schlüssel nicht möglich ist.
Die Analyse von elektromagnetischer Abstrahlung ermöglicht häufig Attacken, selbst wenn die auf Hardwareebene Schutzmaßnahmen gegen Stromverbrauchsanalyse verwendet werden. Bei der Anwendung algorithmischer Gegenmaßnahmen kann man Systeme aber auf ähnliche Weise gegen solche Attacken schützen, wie gegen Attacken, die auf der Aufzeichnung des Stromverbrauchs basieren. Dies bestätigt etwa [DMBÖ+04], in diesem Paper wird auch Double-and-Add-always als Gegenmaßnahme gegen SEMA-Attacken vorgeschlagen.
Attacken, die sich Informationsgewinn durch gezielte Indizierung von Fehlern in die Berechnung zunutze machen, also sogenannte Fault-Analysis-Attacks lassen sich auch gegen Kryptosysteme basierend auf Elliptischen Kurven einsetzen. Hierbei wird versucht, während der Berechnung von $k \cdot P$ einen Fehler derart zu induzieren, dass das Ergebnis ein Punkt ist, der nicht mehr auf der ursprünglichen Kurve $E : y^2 = x^3 + ax + b$ sondern auf einer anderen Kurve $E' : y^2 = x^3 + ax + b'$ liegt, bei der also nur der Parameter b geändert ist, der bei der Punktaddition und -verdopplung nicht verwendet wird. Ist die Kurve E' kryptographisch unsicher, lässt sich also für diese Kurve das DLP effizient lösen, so bietet dies einen Ansatz, das Kryptosystem zu kompromittieren.
Vermeiden lässt sich diese Attacke einfach dadurch, dass am Ende der Berechnung überprüft wird, ob der resultierende Punkt wieder ein Punkt auf der Kurve ist. Lange Zeit galt diese Art der Attacke als die einzige Fault-Analysis-Attacke gegen Kryptographie auf Elliptischen Kurven, 2004 wurde allerdings in [BMOJPS04] gezeigt, dass auch das gezielte Ändern der y-Koordinate von Punkten während der Berechnung möglich ist und als Grundlage für einen Angriff auf bestimmte Implementierungen dienen kann.

5 Zusammenfassung

Diese Seminararbeit hat dargestellt, inwiefern auf Elliptischen Kurven basierende Kryptosysteme mit Seitenkanalattacken angreifbar sind, ein besonderer Schwerpunkt wurde auf Seitenkanalattacken gelegt, die auf der Stromverbrauchsanalyse basieren. Weiterhin wurden geeignete algorithmische Gegenmaßnahmen diskutiert. Es wurde offensichtlich, dass alle diese Gegenmaßnahmen nur auf Kosten von Effizienz implementierbar sind. Mit der Darstellung von Goubins Attacke wurde darauf hingewiesen, dass selbst umfangreich gegen Seitenkanalattacken geschützte Systeme unter bestimmten Umständen dennoch angegriffen werden können.
Bei der Implementation von Kryptosystemen, die Elliptische Kurven verwenden, ist es also extrem wichtig, sich über den Anwendungsbereich der konkreten Implementation klar zu sein und geeignete Maßnahmen zu ergreifen um diese vor realistischen Angriffsszenarien zu schützen.

Literatur

[BMJ02] Eric Brier and Marc Joyce. Weierstrass Elliptic Curves and Side-Channel Attacks. *Lecture Notes in Computer Science*, 2274:335–345, 2002.

[BMOJPS04] Johannes Blömer and Martin Otto Jean-Pierre Seifert. Sign Change Fault Attacks on Elliptic Curve Cryptosystems, 2004. `http://eprint.iacr.org/2004/227.pdf`.

[CAMTO98] Henri Cohen, Atsuko Miyaji, and Takatoshi Ono. Efficient Elliptic Curve Exponentiation Using Mixed Coordinates. In *Proceedings of Asiacrypt '98, LNCS 1514*, pages 51–65. Springer Verlag Heidelberg, 1998.

[CJ01] C. Clavier and M. Joyce. Universal Exponentiation Algorithm — A First Step towards Provable SPA-Resistance. In *Proceedings of CHES 2001, LNCS 2162*, pages 300–308. Springer-Verlag, 2001.

[DMBÖ+04] E. De Mulder, P. Buysschaert, S. B. Örs, P. Delmotte, B. Preneel, G. Vandenbosch, and I. Verbauwhede. Electromagnetic Analyis Attack on an FPGA Implementation of Elliptic Curve Cryptosystem, 2004. `http://www.esat.kuleuven.ac.be/~ieee_sb/documents/20042005/IEEE_spc_DeMulder_Buysschaert.pdf`.

[GG03] Joachim Von Zur Gathen and Jürgen Gerhard. *Modern Computer Algebra*. Cambridge University Press, New York, NY, USA, 2003.

[Gou03] Louis Goubin. A Refined Power-Analysis Attack on Elliptic Curve Cryptosystems. In *Proceedings of PKC 2003, LNCS 2567*, pages 199–211. Springer-Verlag Berlin Heidelberg, 2003.

[HMV04] D. Hankerson, A. Menezes, and S. A. Vanstone. *Guide to Elliptic Curve Cryptography*. Springer, New York, 2004.

[JCT01] Marc Joyce and Christophe Tymen. Protection against Differential Analysis for Elliptic Curve Cryptography — An Algebraic Approach. In *Proceedings of CHES 2001, LNCS 2162*, pages 377–390. Springer-Verlag Berlin Heidelberg, 2001.

[Mil86] Victor S. Miller. Uses of Elliptic Curves in Cryptography. In *Proceedings of CRYPTO '85, LNCS 218*, pages 417–426. Springer-Verlag, 1986.