# Seminararbeit zum Thema Differenzielle Fehleranalyse

(engl. Orginaltitel: Differential Fault Analysis of Secret Key Cryptosystems)

zum Seminar: Kryptografia, bezpieczenstwo komputerowe

von Patrick Hentschke
Matrikelnummer 129292

## Einleitung

Im September 1996 gaben Boneh, Demillo and Lipton (Forscher des Bellcore-Instituts) einen neuen Typus eines kryptoanalytischen Angriffs, unter der Ausnutzung von Berechnungsfehlern in Verschlüsselungssystemen, bekannt. Dieser basiert auf algebraische Eigenschaften der modularen Arithmetik und ist daher nur auf Public Key Systeme anwendbar. Private Key Systeme, deren typischer Vertreter und Standard von symmetrischen Verschlüsselungsverfahren zum Beispiel der DES (Data Encryption Standard) darstellt, entfallen somit.
Einen dazu verwandten kryptoanalytischen Angriff, genannt *Differential Fault Analysis* oder auch kurz *DFA* (zu deutsch: Differentielle Fehleranalyse), beschreiben dagegen Biham und Shamir (zwei bekannte Forscher auf dem Gebiet der Kryptographie). Mit dessen Hilfe es möglich ist Kryptosysteme aufzubrechen, die auf verschiedene Fehlermodelle (z.B. der Speicherasymmetrie) und verschiedene kryptoanalytische Techniken beruhen. Insbesondere zeigen Biham und Shamir, daß unter der Ausnutzung der gleichen Hardware-Fehlermodelle der Bellcore-Forscher, der volle DES-Schlüssel durch die Analyse von 50 bis 200 Chiffretexten (generiert von unbekannten, aber verwandten Klartexten) gewonnen werden kann. In einigen speziellen Fällen genügen schon in etwa fünf Chiffretexte, um den Schlüssel zu erhalten. Die Mächtigkeit von DFA wird sogar in der Tatsache bekräftigt, daß beim Ersetzen von DES durch Triple-DES (dessen Schlüssellänge 168 Bit beträgt und damit praktisch nicht brechbar ist) über den gleichen Angriff mit der im Wesentlichen gleichen Anzahl von gegebenen Chiffretexten, entschlüsselt werden kann. Außerdem ist DFA in der Lage viele weitere symmetrische Blockchiffre, wie IDEA (International Data Encryption Algorythm), Feal (Fast Data Enchipherment Algorythm) und RC5 aufzubrechen. Beachtet werden sollte allerdings, daß kleine Differenzen in den Fehlermodellen sich entscheidend auf die Fähigkeiten und die Komplexität der Angriffe auswirken können.

## Der DES-Angriff

Heutige Computer sind relativ zuverlässig. Daher sind einige Gruppen daran interessiert vorsätzlich Fehler bei Computerberechnungen herbeizuführen. Smartkarten (benutzt bei Identifikationssystemen, der Zugangskontrolle oder dem elektronischen Geld) können infolge ihrer Einfachheit und Fähigkeit der Kontrolle der Umgebung durch den Eigentümer bewußt auf verschiedene Weise zum Versagen gebracht werden.
Bei der Nutzung des Fehlermodells (ähnlich dem, den Boneh, Demillo und Lipton nutzten) nimmt die Smartkarte zufällige transiente Fehler in ihren Registern an, die mit geringer Wahrscheinlichkeit in jedem Bit auftreten, so daß während jeder Verschlüsselung/ Entschlüsselung eine kleine Anzahl von Fehlern auftritt und den Wert der Bits von eins in null umkehrt oder anders herum. Genauer gesagt treten ein oder einige Fehler zu zufälligen Zeiten während der Berechnungen und bei zufälliger Wahl der Register ein. In seinem Angriff nutzt nun der Angreifer zweimal die Smartkarte zum verschlüsseln von (möglicherweise unbekanntem) Klartext, vergleicht die beiden Ergebnisse auf Unterschiede und erlangt bei zwei sich aufgrund eines Fehlers unterscheidenen Chiffretexten als Ergebnis einen korrekt chiffrierten und einen fehlerhaft chiffrierten Text. In dem ersten Schritt des Angriffs wird die Runde (der 16 Runden) identifiziert, in dem sich der Fehler ereignet hat. Die Identifizierung ist einfach und effektiv, falls sich der Fehler in der rechten Hälfte der Runde 16 ereignet hat, denn nur ein Bit hat sich in der rechten Hälfte des Chiffretext umgekehrt und unterscheidet (vor der finalen Permutation) die beiden Chiffretexte voneinander. Die linke Hälfte des Chiffretextes kann sich nur in den ausgehenden Bits von der S-Box unterscheiden, zu welchen ein Einzelbit eintritt. In solch einem Falle können die sechs Schlüsselbits einer jeden solchen S box in der letzten Runde geschätzt werden und die Werte verworfen werden, welche nicht mit dem erwarteten Unterschied dieser S-Boxen übereinstimmen. Falls sich die Fehler in der 15.Runde ereignen, können die Informationen über die Schlüsselbits in mehr als zwei S-Boxen in der letzten Runde erworben werden. Der Einzelbitfehler der Runde 15 wird abgeschätzt und überprüft, ob der erwartete Ausgangsunterschied verursacht wurde und ob auch der Unterschied der rechten Hälfte den erwarteten Unterschied am Ausgang der F-Funktion in der letzten Runde verursacht hat (d.h. für den Unterschied der linken Hälfte des Chiffretextes XOR “exlusiv Oder“ zum Fehler). Die Analyse der Fehler in der 14.Runde erfolgt über einen ähnlichen Weg über Zählmethoden, wobei für jede S-Box separat gezählt wird.
Letztendlich können mit einem geeignetem Analyseprogramm bei einem möglichen Angriff über einen Personalcomputer zufällige Einzelfehler in allen Runden und somit der ganze Unterschlüssel über 50 bis 200 gegebene Chiffretexte errechnet werden. Ist der Unterschlüssel einmal bekannt, kann der Fakt genutzt werden, daß der Unterschlüssel 48 Schlüsselbits des 56 Bit langen Schlüssels des DES enthält und die restlichen 8 Bit über $2^8 = 256$ Wege erraten werden können. Alternativ kann der letzte Unterschlüssel, die letzte Runde abziehend und die schon identifizierten Fehler entfernend, über die Analyse der vorhergehenden Runden und den gleichen Daten ermittelt werden. Diese letzte Näherung macht es möglich Triple-DES anzugreifen. Sollte der Angreifer in der Lage sein, Fehler in gewählten Positionen oder in einer gewählten Zeit während der Verschlüsselung herbeizuführen, so kann er beim Vorkommen der Fehler in den letzten zwei, drei oder vier der 16 Runden das Ergebnis um einen großen Faktor auf etwa 10 benötigte Chiffretexte verbessern.
In einigen Implementationen ist der DES-Schlüssel über zwei 28 Bit-Shiftregister realisiert und wechselt in jeder Runde bis er am Ende der Verschlüsselung in den orginalen Zustand versetzt wird. Die Gesamtanzahl der Verschiebungen bemißt sich dabei auf 28 während der 16 Runden, falls die Fehler die Änderungen dieser Register betreffen und die folgenden Verschlüsselungen den Schlüssel zu einem verwandten Schlüssel verändern, so kann DFA zusammen mit anderen kryptoanalytischen Angriffen kombiniert werden.

## Das differenzielle Fehlermodell

Die Voraussetzung hinter dem Fehlermodell liegt in der Aufbewahrung des kryptographischen Schlüssels in einem asymmetrischen Speicher, in welchem beim einem Fehlereintritt die Wahrscheinlichkeit größer ist, daß Bits von dem Wert eins zu null als von null zu eins übergehen. Die meisten nicht-flüchtigen Speichertypen (wie z.B. das EEPROM) weisen das Merkmal der Asymmetrie auf. Bei einem kryptoanalytischen Angriff wird zu einem unveränderlichem Klartext m der aktuell im nicht-flüchtigen Speicher gespeicherte unbekannte Schlüssel k wiederholend zur Verschlüsselung genutzt, um einen Strom von Chiffretexten $\{c_o, ..., c_r\}$ bei einer Spannungsversorgungstrennung nach jeder Verschlüsselung zu erhalten. Jede Änderung der Chiffretexte ist darauf zurückzuführen, daß jeweils ein weiteres Schlüsselbit von dem Wert eins zu null übergeht, somit einen Wechsel beim aktuellen Schlüssel $k_i$ zu einer neuen Variante $k_{i+1}$ bewirkt und bis $k_f$ fortgeführt wird, womit $c_f$ das Ergebnis der Verschlüsselung von m darstellt, da in etwa n/2 Einser-Bits in dem orginalen unbekannten Schlüssel vorhanden waren. Nun erfolgt in einer zweiten Phase des Angriffs die Wiedererlangung der Ein-Bit-Positionen über das wiederzurückfolgen des Weges über den bekannten Schlüssel $k_f$ in *O(n)* Stufen zum Orginalschlüssel $k_0$, *O(n)* Schlüssel versuchend in jeder Stufe. Der Angriff ist somit erfolgreich, wenn das Fehlermodell zufriedenstellend ist und in seiner Gesamtkomplexität $O(n^2)$ Verschlüsselungen enthält.

## DFA-immune Kryptosysteme

In der Tat gibt es auch Kryptosysteme, bei denen DFA an seine Grenzen stößt. Spezifische Implementationen besitzen schlüsselprüfende Mechanismen, welche die Nutzung von zufälligen Bitmustern garantieren und daher Schutz gegenüber Übertragungsfehlern und nicht korrekt gewählten Schlüsseln geben.

## Angriffe auf weitere Systeme

Erfolgreich konnten auch Identifikationsverfahren, wie das von Fiat-Shamir, welches auf dem modularen Quadratwurzelproblem beruht, sowie das von Schnorr auf dem Diskreten Logarithmusproblem beruhende, angegriffen werden. Dabei handelt es sich um Identifikationsprotokolle bei denen es, am Beispiel von Fiat-Shamir erläutert, um folgendes Prinzip handelt:

Es wird ein Szenario beschrieben, in welchem ein Kommunikationspartner (Alice) ihre Identität ihrem Gegenüber (Bob) nachzuweisen hat. Zur Schlüsselerzeugung wählt Alice zwei große Primzahlen und bildet das Produkt aus diesen: n = p*q. Natürlich sind p und q geheim zu halten. Alice wählt eine beliebige Zahl $s \in Z_n^*$ und berechnet *v* als

$$v := s^2 \bmod n$$

Die Zahl *s* ist das Geheimnis, welches Alice für sich behalten muß; *v* dagegen wird veröffentlicht, da mit Hilfe von *v* eine Person, hier Alice, ihre Identität nachweisen kann.

Die Feststellung von Alice durch Bob, ist nun durch das folgende Protokoll festgelegt:

1. Alice wählt eine Zufallszahl $r \in Z_n^*$ und bestimmt

$$z := r^2 \bmod n$$

Die Zahl $z$ schickt Alice an Bob.

2. Bob wählt nun ein Zufallsbit $b$, also $b = 0$ oder $b = 1$ und schickt dieses Alice.

3. Alice berechnet

$$y := r\, s^b \bmod n$$

Das $y$ schickt sie wieder an Bob.

4. Bob überprüft nun, ob die folgende Identität erfüllt ist

$$y^2 \bmod n = z\, v^b \bmod n$$

Wenn nun Gleichheit gegeben ist, dann kann sich Bob ziemlich sicher sein, daß es sich bei seinem Kommunikationspartner um Alice handelt.

## Abschließende Betrachtung

DFA ist als wesentlicher Forschungsgegenstand für technikweisende und implementationsabhängige Sicherheitsempfehlungen zu sehen. Die Demonstration dessen, daß selbst sicher geltende Verschlüsselungsysteme nicht gefeit sind vor Angriffen, sollte zu bedenken geben, daß es notwendig ist noch weitere Forschungsarbeit in Richtung der Computersicherheit zu betreiben.

## Quellenangabe


- E. Biham and Shamir, Differential Fault Analysis, LNCS 1294, Advances in Cryptology, Proceedings of Crypto'97, Springer-Verlag, pp. 513-525, 1997
- P. Paillier, Evaluating Differential Fault Analysis of Unknown Cryptosystems, Public-Key Cryptography, vol. 1560 of Lecture Notes in Computer Science, pp. 235-244, Springer-Verlag, 1999
- http://bofriis.dk/security/OntheImportanceofEliminatingErrors.pdf.