Fault Attacks on RSA-CRT

- Boneh/deMillo/Lipton (Bellcore) [1997]
- Shamir [1999]
- Aumüller/Bier/Fischer/Hofreiter/Seifert (Infineon) [2002]

RSA – Seitenkanalanalyse

- RSA
- RSA-CRT

  [CRT = Chinese Remainder Theorem]
- Bellcore Attacke
- Lenstra Attacke
- Shamir Gegenmaßnahme
- Infineon Gegenmaßnahme

RSA – klassisch

- $N = p \times q$ p, q prim

- Signierung durch:
- $S = m^d \bmod N$ d ist „privat“
- $Phi(N) = (p - 1)(q - 1)$
- $e \times d = 1 \pmod{(p - 1)(q - 1)}$ e ist „öffentlich“

- Verifizierung durch:
- $S^e \bmod N$

RSA – CRT (Garner)

→ schnellere Verschlüsselung (4-mal schneller)

- Restklasse N sehr groß
- Restklassen p und q wesentlich kleiner

- CRT ermöglicht Rechnen mit kleineren Resteklassen (p und q)
- $d_p = d \bmod (p - 1)$
- $d_q = d \bmod (q - 1)$
- $Q_p = q^{-1} \bmod p$
- $S_p = m^{dp} \bmod p$
- $S_q = m^{dq} \bmod q$
- $S = S_q + ((S_p - S_q) \times Q_p \bmod p) \times q$

Fehlermodell

- Permanente Fehler
- Transiente Fehler:
  - Kurzfristige Fehler
  - z.B. plötzlich wechselndes Bit

Bellcore Attacke

- $S_p = m^{dp} \mod p$
- $S_q = m^{dq} \mod q$
- $S = S_q + ((S_p - S_q) \times Q_p \mod p) \times q$
- $S = aS_p + bS_q \mod N$
- a und b über *erweiterten euklidischen Algorithmus* berechenbar
- CRT: $0 = a \mod q = b \mod p$; $1 = a \mod p = b \mod q$

- o.B.d.A Fehler in $m^{dp} \mod p \rightarrow S_p' \neq S_p$
- $S' = S_q + ((S_p' - S_q) \times Q_p \mod p) \times q$
- $S - S' \neq 0$; $S - S' \neq 0 \mod p \times q$
- $S - S' = 0 \mod q$
- $ggT(S - S', N) = q$
- Voraussetzung: $S - S' \neq 0 \mod p$
- Benötigte Mittel: S' und S

Lenstra Attacke

- Ähnlich wie Bellcore Attacke
- Benötigte Mittel: S' und m

- $S^e = m$
- $ggT(m - S'^e, N) = q$

Folgerung für RSA-CRT

- RSA- CRT extrem anfällig für Fehlerattacken
- 1 Fehler reicht aus, um komplettes System zu brechen
- → reiche Literatur an DFA – countermeasures RSA-CRT
- Einfache Wiederholung der Rechnung reicht nicht aus

Gegenmaßnahme von Shamir

- normale Gegenmaßnahmen erfordern doppelte Laufzeit
  - zweifache Berechnung der Verschlüsselung und anschließende Kontrolle
- neue Methode von Shamir (Patent 1997):
  - r prim
  - $p' = p \times r$
  - $q' = q \times r$
  - $d_p' = d \bmod (p - 1)(r - 1)$
  - $d_q' = d \bmod (q - 1)(r - 1)$
  - $S_p' = (m \bmod p')^{dp'} \bmod p'$
  - $S_q' = (m \bmod q')^{dq'} \bmod q'$
  - **if** $(S_p' - S_q') \bmod r \neq 0$ **then Error**
  - **else** $S_p = S_p' \bmod p$; $S_q = S_q' \bmod q$
  - $S = S_q + ((S_p - S_q) \times Q_p \bmod p) \times q$
  - → Randomisierung Modul

Verbesserung im Infineon-Paper

Randomisierung Modul und Exponent

input: $m, p, q, d_p, d_q, q^{-1} \bmod p$

let t be a short prime number, e.g., 32 bits

$p' := p * t$
$d'_p := d_p + \text{random}_1 * (p-1)$
$S'_p := m^{d'_p} \bmod p'$
if $\neg(p' \bmod p \equiv 0 \ \wedge \ d'_p \bmod (p-1) \equiv d_p)$ **then return**(error)

$q' := q * t$
$d'_q := d_q + \text{random}_2 * (q-1)$
$S'_q := m^{d'_q} \bmod q'$
if $\neg(q' \bmod q \equiv 0 \ \wedge \ d'_q \bmod (q-1) \equiv d_q)$ **then return**(error)

Verbesserung im Infineon-Paper

$S_p := S'_p \bmod p$
$S_q := S'_q \bmod q$
$S := S_q + ((S_p - S_q) * q^{-1} \bmod p) * q$
if $\neg((S - S'_p \bmod p \equiv 0) \ \wedge \ (S - S'_q \bmod q \equiv 0))$ **then return**(error)

$S_{pt} := S'_p \bmod t$
$d_{pt} := d'_p \bmod (t-1)$
$S_{qt} := S'_q \bmod t$
$d_{qt} := d'_q \bmod (t-1)$
if $(S_{pt}^{d_{qt}} \equiv S_{qt}^{d_{pt}} \bmod t)$ **then**
  return(S)
else
  return(error)

output: $m^d \bmod (p * q)$