## 2.9 Die Signaturfalle

Ein Problem, das nicht die Sicherheit der Chiffre RSA, sondern die Rahmenbedingungen ihrer Anwendung betrifft, ist die Signaturfalle: Da RSA in umgekehrter Reihenfolge zur digitalen Signatur verwendet wird, muss man stets darauf achten, dass man nicht im Glauben, einen Text digital zu signieren, in Wirklichkeit einen vorgelegten Geheimtext entschlüsselt. Würden wirklich normlerweise Klartexte signiert, würde das sofort ins Auge springen. Es gibt aber (mindestens) drei Gründe, warum das nicht so ist:

1. Bei der digitalen Signatur wird aus Performanzgründen so gut wie immer ein (kryptographischer) Hash-Wert der Nachricht signiert. Dieser ist von einer Zufalls-Bitkette nicht zu unterscheiden.

2. Bei der starken Authentisierung wird statt einer Passworteingabe als Identitätsnachweis eine zufällige Bitkette digital signiert. Selbst wenn das Ergebnis ein entschlüsselter Klartext wäre – man sieht es in der Regel gar nicht, es wird direkt dem Kommunikationspartner (z. B. Zielrechner) übersandt.

3. Außerdem kann ein beliebiger Text durch „Camouflage“ vorverschlüsselt zum Signieren bzw. Entschlüsseln vorgelegt werden. Selbst wenn man sich das Ergebnis dieses Vorgangs ansieht, erkannt man nicht, dass man in Wirklichkeit entschlüsselt hat – siehe unten. Diese Eigenschaft des RSA-Verfahrens ist sogar sehr nützlich: Sie ist Grundlage der blinden Signatur und damit der Erzeugung von digitalen Pseudonymen und anonymen Berechtigungsnachweisen. Siehe
   `http://www.uni-mainz.de/~pommeren/DSVorlesung/KryptoProt/`

Es handelt sich hierbei übrigens um einen *Angriff mit gewähltem Geheimtext*. Dieses Problem wird in der Praxis dadurch umgangen, dass man für die drei – evtl. vier – Funktionen

- Chiffrierung,
- digitale Signatur,
- starke Authentisierung,
- evtl. blinde Signatur

jeweils separate Schlüsselpaare erzeugt und verwendet.

Nun also zu der „Camouflage“, die zur Verschleierung des Angriffs mit gewähltem Geheimtext dient. Der Ablauf ist:

1. Der Angreifer M hat einen Geheimtext $x = E_{\mathrm{A}}(a)$ aufgefangen. Er verschlüsselt ihn mit einer nur ihm bekannten Funktion $C$ zu $y = C(x)$.

2. Er schiebt $y$ dem Opfer A zur digitalen Signatur unter; dieses erzeugt $z = D_{\mathrm{A}}(y)$.

3. Der Angreifer entfernt die „Camouflage“ durch eine geeignete Rücktransformation $C'$. Hat er ein Paar $(C, C')$ zur Verfügung, so dass $C' \circ D_{\mathrm{A}} \circ C = D_{\mathrm{A}}$, so ist $a = D_{\mathrm{A}}(x) = C'(z)$.

Eine Besonderheit des RSA-Verfahrens ist, dass ein solches Paar $(C, C')$ von Transformationen existiert; ist $E_{\mathrm{A}}(a) = a^e \bmod n$, so ist $C$ die Verschiebechiffre auf $\mathbb{M}_n = (\mathbb{Z}/n\mathbb{Z})^\times$ mit $u^e$ und $C'$ die Multiplikation mit $u^{-1} \bmod n$, wobei $u \in \mathbb{M}_n$ zufällig gewählt wird. Der Ablauf des Angriffs ist also:

1. M wählt $u$ und bildet $y = C(x) = u^e x \bmod n$.

2. A erzeugt $z = y^d \bmod n$.

3. M berechnet $C'(z) = zu^{-1} = y^d u^{-1} = u^{ed} x^d u^{-1} = x^d = a$ in $\mathbb{Z}/n\mathbb{Z}$.