Ruhr-Universität Bochum
Lehrstuhl für Kryptologie und IT-Sicherheit
Prof. Dr. Alexander May
Elena Kirshanova, Ilya Ozerov

Hausübungen zur Vorlesung

Kryptanalyse

SS 2014

Blatt 1 / 14. April 2014

Abgabe: 24. Oktober 2014, 14.00 Uhr, Kasten NA/02

AUFGABE 1 (5 Punkte)**:**

Sei (N, e) ein öffentlicher RSA-Schlüssel und (N, d) der zugehörige geheime Schlüssel. Zeigen Sie, dass die RSA Entschlüsselung auch für Nachrichten $m \in \mathbb{Z}_N \setminus \mathbb{Z}_N^*$ korrekt ist.

AUFGABE 2 (5 Punkte)**:**

(a) Zeigen Sie, dass $\phi(ab) = \phi(a)\phi(b)$, falls $\text{ggT}(a, b) = 1$.

(b) Zeigen Sie, dass $\phi(n) = n \cdot \prod_{p|n \text{ mit } p \text{ prim}}(1 - \frac{1}{p})$ für alle $n \in \mathbb{N}$.

Hinweis: Aufgabe 2 aus Präsenzübung 1 darf verwendet werden.

AUFGABE 3 (5 Punkte)**:**

(a) Zeigen Sie, dass es ein $n \in \mathbb{N}$ gibt mit $\phi(n) \leq \frac{n}{4}$.

(b) Sei $n_{\min} := \min\{n \in \mathbb{N} \mid \phi(n) \leq \frac{n}{4}\}$. Zeigen Sie mit Hilfe von *sage*, dass $\phi(n) > \frac{n}{4}$ für alle $n < n_{\min}$. Geben Sie dazu $\frac{\phi(n)}{n}$ für alle $n \leq n_{\min}$ in einem Plot aus. Die ϕ-Funktion können Sie in *sage* mit `euler_phi` berechnen lassen. Mit

$$\mathsf{L} = [\mathsf{euler\_phi}(\mathsf{n}) \text{ for } \mathsf{n} \text{ in } \mathsf{range}(1, 101)]$$

kann z.B. eine Liste L der ersten 100 Werte der ϕ-Funktion erstellt werden. Diese kann mit `list_plot(L)` geplottet werden. Geben Sie den Quellcode Ihres Programms mit ab.

(c) Zeigen Sie, dass es ein $n \in \mathbb{N}$ gibt mit $\phi(n) \leq \frac{n}{8}$. Geben Sie das minimale n mit dieser Eigenschaft an und zeigen Sie, dass es minimal ist.

AUFGABE 4 (5 Punkte)**:**

Alice hat eine Einladung zu ihrer Geburtstagsparty an 17 Freunde verschickt. Diese besitzen die paarweise teilerfremden RSA-Moduln $N_1, N_2, \ldots, N_{17}$ und verwenden alle den öffentlichen Exponenten $e = 17$. Die Einladung wurde symmetrisch mit einem 40 Bit CDMF[1] Schlüssel k verschlüsselt, welcher in einer (für alle Gäste identischen) Nachricht m asymmetrisch mit RSA verschlüsselt wurde. Die Nachricht m (Padding + Schlüssel) ist ein gültiger Klartext für alle Moduln, d.h. $m < \min\{N_1, N_2, \ldots, N_{17}\}$. Eve ist nicht zur Party eingeladen, konnte aber die Chiffrate $c_1, c_2, \ldots, c_{17}$ mitschneiden.

(a) Implementieren Sie einen Broadcast-Angriff (Präsenzübung 1, Aufgabe 4) in *sage*. Die öffentlichen Parameter sind als Listen N und C in der Datei `crta.txt` gegeben. Wie lautet der Schlüssel k (die letzten 40 Bit von m)? Geben Sie den Quellcode Ihres Programms mit ab.

(b) Die zweite Nachricht geht zusätzlich an einen neuen Gast, ein anderer hat dafür abgesagt, so dass die Nachricht weiterhin an 17 Partygäste versendet wird. Die öffentlichen Daten finden Sie in der Datei `crtb.txt`. Weshalb funktioniert der Broadcast-Angriff nun nicht mehr? Modifizieren Sie den Angriff und bestimmen Sie m. Geben Sie den Quellcode Ihres Programms mit ab.

Hinweis: Sie können sich bspw. den Modulus N_i mit $\mathsf{N[i-1]}$ ausgeben lassen. Sie können in *sage* mit der Funktion crt den verallgemeinerten Chinesischen Restsatz und mit gcd den größten gemeinsamen Teiler berechnen lassen. Zur Berechnung von $\sqrt[n]{a}$ können Sie sich mit $\mathsf{R.\langle x\rangle = PolynomialRing(ZZ)}$ zunächst einen Polynomring über den ganzen Zahlen definieren und dann mit $\mathsf{(x\hat{}n - a).roots()}$ die ganzzahlige Nullstelle der Funktion $f(x) = x^n - a$ berechnen. Zur Bestimmung der letzten 40 Bit von m können Sie mit hex die Nachricht hexadezimal darstellen. Die modulare Exponentiation $a^b \bmod c$ kann mit $\mathsf{pow(a, b, c)}$ berechnet werden, zur Bestimmung von inversen Elementen mit $b = -1$.

[1] http://en.wikipedia.org/wiki/CDMF