RSA - Key Generation

Luc Spachmann

Friedrich-Schiller-Universität Jena

10.12.2021

Primzahltest von Miller-Rabin

- Nichtdeterministischer Primzahltest
- Sehr schnell auch für große Zahlen
- W'Keit von 'False Negative' ist Null
- W'Keit von 'False Positive' $< \frac{1}{4}$
- Durch wiederholte Anwendung kann Fehlerwahrscheinlichkeit verringert werden

Primzahltest von Miller-Rabin

Require: $n \in \mathbb{N}$

1: Bestimme ungerades m mit $n - 1 = 2^k \cdot m$
2: Wähle zufälliges $2 \leq a < n$
3: $b = a^m \mod n$
4: **if** $b \equiv 1 \mod n$ **then**
5: **return Prim**
6: **end if**
7: **for** $i = 1$ to k **do**
8: **if** $b \equiv -1 \mod n$ **then**
9: **return Prim**
10: **else**
11: $b = b^2 \mod n$
12: **end if**
13: **end for**
14: **return Zusammengesetzt**

Schlüsselgenerierung

- Generiere 2 ausreichend große Primzahlen p, q
- Beide Primzahlen sollten nicht zu nahe an einander liegen
- Erzeuge dazu zufälliges z in der gewünschten Größe
- Teste $30z + i$ für $i \in \{1, 7, 11, 13, 17, 19, 23, 29, 30 + 1, ...\}$
- Erzeuge zufälliges e Teilerfremd zu $\varphi(pq)$
- Berechne d mit $de \equiv 1 \mod \varphi(pq)$
- Schlüssel sind dann $(e, pq), (d, pq)$

Verfahren der Differenz der Quadrate

- Faktorisierungsidee für p und q mit geringem Abstand
- Sei $N = pq$
- Suche u, w mit $N = u^2 - w^2$
- Dann ist $N = (u - w)(u + w)$

Require: N

1: $u = \lceil \sqrt{N} \rceil$
2: **while** not $is\_square(u^2 - N)$ **do**
3: $\quad u = u + 1$
4: **end while**
5: $w = \sqrt{u^2 - N}$
6: **return** $(u + w, u - w)$

Aufgaben

- Erweitert das Programm letzter Woche um die Schlüsselgenerierung
- Implementiert und testet das Verfahren der Differenz der Quadrate