Spezialthema Kryptographie - RSA

Lukas Wais

14. Dezember 2023

1 Funktionsweise RSA

Das RSA-Verfahren ist ein asymmetrisches Verschlüsselungsverfahren.
Der öffentliche Schlüssel und der zugehörige private Schlüssel wird wie folgt berechnet:

1. Wähle (geheim) zwei verschiedene Primzahlen p und q.
2. Berechne $n = p \cdot q$.
3. Berechne Die Eulersche φ Funktion $\varphi(n) = (p-1) \cdot (q-1)$.
4. Wähle eine Zahl $e > 1$, die zu $\varphi(n)$ teilerfremd ist, also $\mathrm{ggT}(e, \varphi(n)) = 1$. Berechne mit dem Euklidischen Algorithmus eine ganze Zahl
   $d > 1 \,\text{mit}\, e \cdot d \equiv 1 \bmod \varphi(n)$.

Die beiden Zahlen $\boldsymbol{n}$ und $\boldsymbol{e}$ sind der öffentliche Schlüssel. Die Zahl $\boldsymbol{d}$ ist der private Schlüssel.

2 Nachricht Senden

Alice möchte an Bob eine verschlüsselte Nachricht senden.

1. Alice erhält den öffentlichen Schlüssel (n, e) von Bob.
2. Alice kann damit jede natürliche Zahl m mit $1 \leq m < n$ an Bob übertragen. Alice berechnet $m^e \bmod n$ und sendet das Ergebnis c an Bob.

3 Schlüssel Berechnen

Wir erzeugen den öffentlichen Schlüssel und den privaten Schlüssel mit
$p = 7, q = 13$ und $e = 11$.

$$n = 7 \cdot 13 = 91 \quad \varphi(n) = 6 \cdot 12 = 72$$

Wir berechnen mit dem erweiterten Euklidischen Algorithmus eine ganze Zahl

$$d > 1 \,\text{mit}\, e \cdot d \equiv 1 \bmod \varphi(n)$$

Um d zu finden müssen wir das Modular Multiplikative Inverse finden.

$$d = e^{-1} \; mod \; \varphi(n)$$

Dafür verwenden wir den erweiterten Euklidischen Algorithmus.

Somit haben wir die öffentlichen Schlüssel: $n = 91, \; e = 11$ und den privaten Schlüssel $d = 59$.

4 Verschlüsseln

Bob hat den öffentlichen Schlüssel $n = 91,\ e = 11$. Verschlüssle damit die Nachricht $m = 42$. Wir berechnen $c = m^e \bmod n$.

$$c = 42^{11} \bmod 91 = 35$$

5 Entschlüsseln

Alice möchte die Empfangene Nachricht $c = 35$ entschlüsseln. Die öffentlichen Schlüssel $n = 91,\ e = 11$ sind bekannt. Ihr privater Schlüssel ist $d = 59$. Wir berechnen $m = c^d \mod n$.

$$m = 35^{59} \bmod 91 = 42$$

6 Warum Funktioniert RSA?

Wir müssen für alle Zahlen m mit $1 \leq m < n$ die folgende Identität zeigen:

$$(m^e)^d \equiv m \mod n \tag{1}$$

Der private Schlüssel d wurde so gewählt, dass $e \cdot d \equiv 1 \bmod \varphi(n)$ gilt. Der private Schlüssel d wurde so gewählt, dass $e \cdot d \equiv 1 \bmod \varphi(n)$ gilt.

$$m \cdot m^{k \cdot \varphi(n)} \equiv m \mod n \tag{2}$$

Fall 1: m und n sind teilerfremde Zahlen.
(2) folgt aus dem Satz von Euler $\left(m^{\varphi(n)} \equiv 1 \bmod n\right)$

$$m \cdot m^{k \cdot \varphi(n)} = m \cdot \left(m^{\varphi(n)}\right)^k \equiv m \cdot 1^k = m \mod n \checkmark$$

Fall 2: m und $n = p \cdot q$ sind nicht teilerfremd. Da p und q verschiedene Primzahlen sind, können wir statt (2) auch zeigen, dass

$$m \cdot m^{k \cdot \varphi(n)} \equiv m \mod p \quad \text{und} \quad m \cdot m^{k \cdot \varphi(n)} \equiv m \mod q \quad \text{gilt.} \tag{3}$$

Die kleinste positive natürliche Zahl s, für die $\mathrm{ggT}(s, p) > 1$ und $\mathrm{ggT}(s, q) > 1$ gilt, ist $s = p \cdot q$. Da $m < p \cdot q$ und $\mathrm{ggT}(m, p \cdot q) > 1$ gilt, bleiben also nur 2 Möglichkeiten:

a) $\mathrm{ggT}(m, q) = 1$ und $p \mid m$

   oder

b) $\mathrm{ggT}(m, p) = 1$ und $q \mid m$

Wir zeigen (3) mit Hilfe von a). Mit b) funktioniert es gleich, p und q vertauschen nur ihre Rollen.

$$p \mid m \Longrightarrow \underbrace{m \cdot m^{k \cdot \varphi(n)}}_{\equiv 0} \equiv \underbrace{m}_{\equiv 0} \bmod p \checkmark$$

Wegen $\mathrm{ggT}(m, q) = 1$ können wir den Satz von Euler verwenden: $m^{\varphi(q)} \equiv 1 \bmod q$ $m \cdot m^{k \cdot \varphi(n)} \equiv m \bmod q$ gilt, weil:

$$m \cdot m^{k \cdot \varphi(n)} = m \cdot \left(m^{\varphi(q)}\right)^{k \cdot \varphi(p)} \equiv m \cdot 1^{k \cdot \varphi(p)} = m \mod q \checkmark$$

7 Anhang

7.1 Teilbarkeit

Eine ganze Zahl a teilt eine ganze Zahl b genau dann, wenn es eine ganze Zahl n gibt, so dass $a \cdot n = b$. Man sagt auch a teilt b, oder b ist ein Vielfaches von a.

7.2 Modulo

Die Funktion $\lfloor \cdot \rfloor : \mathbb{R} \to \mathbb{Z}$ ist definiert durch

$$\lfloor x \rfloor := \max\{m \in \mathbb{Z} : m \leq x\}$$

(gesprochen: Floor x).
Die Funktion $\bmod : \mathbb{R} \times (\mathbb{R} \backslash \{0\}) \to \mathbb{R}$ ist definiert durch

$$\operatorname{mod}(x, y) := x - y \left\lfloor \frac{x}{y} \right\rfloor$$

Statt $\operatorname{mod}(x, y)$ schreibt man auch $x \bmod y$; Sprechweise: x modulo y.
Beispiele:

- $\operatorname{mod}(7, 5) = 2,\ \operatorname{mod}(7, -5) = -3,\ \operatorname{mod}(-7, 5) = 3,\ \operatorname{mod}(-7, -5) = -2.$
- Eine Uhr mit einem gewöhnlichen analogen Ziffernblatt zählt die Stunden modulo 12.

7.3 Euklidischer Algorithmus

Der größte gemeinsame Teiler zweier gegebener Zahlen $a, b \in \mathbb{Z}$ lässt sich mit dem euklidischen Algorithmus berechnen:

```
static int Euclid(int m, int n)
{
    if (n == 0)
    {
        return m;
    }

    return Euclid(n, m % n);
}
```

Beispiel: $a = 19 \cdot 27 \cdot 47 \cdot 61 = 1470771, b = 19 \cdot 23 \cdot 43 \cdot 59 = 1108669$. Der euklidische Algorithmus berechnet eine Folge von Divisionsresten, $r_1 = a, r_2 = b$, und $r_k = \operatorname{mod}(r_{k-2}, r_{k-1})$ für $k \geq 2$:

$$\begin{aligned}
r_3 &= \operatorname{mod}(1470771, 1108669) &&= 362102,\\
r_4 &= \operatorname{mod}(1108669, 362102) &&= 22363,\\
r_5 &= \operatorname{mod}(362102, 22363) &&= 4294,\\
r_6 &= \operatorname{mod}(22363, 4294) &&= 893,\\
r_7 &= \operatorname{mod}(4294, 893) &&= 722,\\
r_8 &= \operatorname{mod}(893, 722) &&= 171,\\
r_9 &= \operatorname{mod}(722, 171) &&= 38,\\
r_{10} &= \operatorname{mod}(171, 38) &&= 19,\\
r_{11} &= \operatorname{mod}(38, 19) &&= 0.
\end{aligned}$$

Der erweiterte euklidische Algorithmus berechnet neben dem ggT von a, b auch s, t die folgende Gleichung erfüllt:

$$\mathrm{ggT}(a, b) = s \cdot a + t \cdot b$$

```
static int ExtendedEuclid(int a, int b, out int s, out int t)
{
    if (b == 0)
    {
        s = 1;
        t = 0;
        return a;
    }

    int gcd = ExtendedEuclid(b, a % b, out int s1, out int t1);
    s = t1;
    t = s1 - (a / b) * t1;

    return gcd;
}
```

Beispiel: $a = 19 \cdot 27 \cdot 47 \cdot 61 = 1470771,\ b = 19 \cdot 23 \cdot 43 \cdot 59 = 1108669$

g	u	v	g'	u'	v'
1470771	1	0	1108669	0	1
1108669	0	1	362102	1	-1
362102	1	-1	22363	-3	4
22363	-3	4	4294	49	-65
4294	49	-65	893	-248	329
893	-248	329	722	1041	-1381
722	1041	-1381	171	-1289	1710
171	-1289	1710	38	6197	-8221
38	6197	-8221	19	-26077	34594
19	-26077	34594	0	58351	-77409

Daraus folgt $\gcd(a, b) = 19 = -26077a + 34594b$.

7.4 Eulersche φ Funktion

Die Phi-Funktion ist definiert durch $\varphi : \mathbb{N}^+ \to \mathbb{N}^+$ und

$$\varphi(n) := |\{a \in \mathbb{N} \mid 1 \leq a \leq n \wedge \mathrm{ggT}(a, n) = 1\}|$$

Sie ordnet jeder natürlichen Zahl n die Anzahl der natürlichen Zahlen a von 1 bis n zu, die zu n teilerfremd sind, für die also $ggT(a, n) = 1$ ist.