ASP Team 103 – Aufgabe 504: RSA

Technische Universität München

TUM School of Science

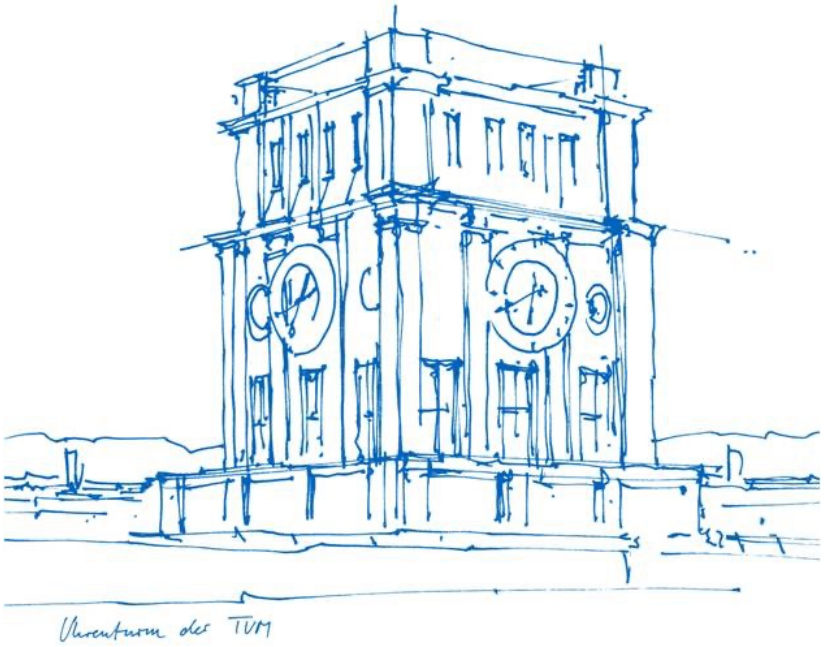
Lehrstuhl für Rechnerarchitektur und parallele Systeme

Teammitglieder:

Guo Linfeng

Özakay Baris

Julian Julian



Inhaltsverzeichnis

RSA und die Funktionsweise

- Im Jahr 1977 veröffentlicht
- Nach seinen Erfindern Rivest, Shamir und Adleman ernannt
- Öffentlichen und- privaten Schlüssel
    - (e, N) und (d, N)
    - Bestimmten mathematischen Eigenschaften müssen erfüllen
- Wird bei E-Mails, Webservern oder Banken verwendet
- Das Verfahren besteht aus vier Schritten:
    - Schlüsselgenerierung
    - Schlüsselverteilung
    - Verschlüsseln
    - Entschlüsseln

Generierung von N

- Zwei Primzahlen p und q => p * q = N
- Lineare Kongruenzgenerator seed * mul + inc = rand
- Kryptographisch gesehen nicht sicher
- Absichtlich p und q mit 32bit begrenzt
- Naive Primzahlenchecker
- Faktorisieren bei “composite“ Zahlen

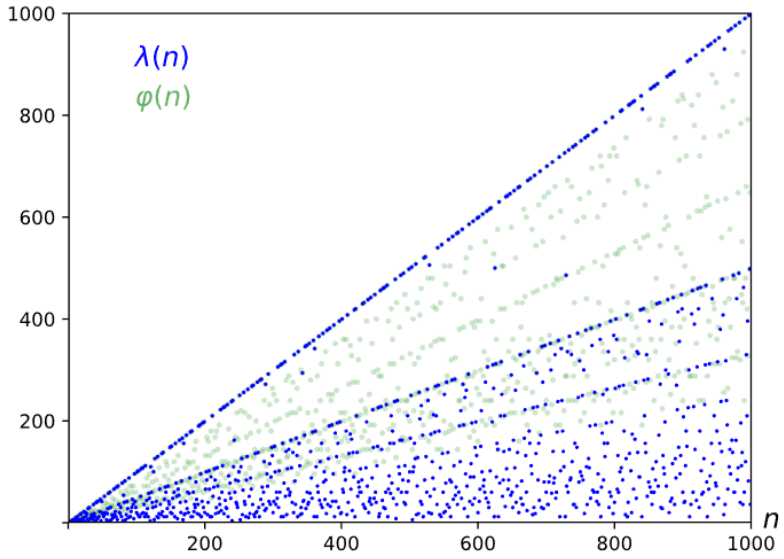
Der Verschlüsselungsexponent e

- Eigenschaft: Teilerfremd zu $\varphi(N)$ und $1 < e < \varphi(N)$
- Performanz und Effizienz
- Weniger Multiplikation
- Weniger Modulo
- Kleineres Hamming Gewicht und Bit Länge
- Carmichael

Camichael Funktion

- λ Funktion
- Berechnet den kleinsten positiven Quotienten der φ- Funktion
- Kleinere obere Schranke

$$\lambda(N) = \frac{p-1 \cdot q-1}{gcd(p-1, q-1)}$$



Der Entschlüsselungsexponent d

- Der größte gemeinsame Teiler GCD, Euklischer Algorithmus
- Erweiterter euklidischer Algorithmus

$$ax + by = gcd(a, b)$$

$$x_{new} = y, y_{new} = x - (floor(r/s)) \cdot y$$

$$ed + \varphi(N)y = gcd(e, \varphi(N)) = 1$$

$$ed = 1 mod(\varphi(N))$$

Performanzanalyse

- 4 Hauptvarianten und extra 2 für Benchmarking
- C-Implementierung schneller als Assembly ☹
- Alles 1 Mio. mal durchlaufen gelassen

Variante	Sekunden
V4	0.0420×10^{-2}
V5	0.0388×10^{-2}

Variante	Sekunden
V1	0.0097×10^{-2}
V2	0.0090×10^{-2}

Schwierigkeiten bei der Implementierung

Zusammenfassung

Danke fürs Zuhören !