Kryptographie: Verteidigung gegen die dunklen Künste in der digitalen Welt

Prof. Dr. Rüdiger Weis

Beuth Hochschule für Technik Berlin

Tag der Mathematik 2015

Flächendeckendes Abhören

Regierungen scheitern beim Schutz der Bürger

Kryptographie und Open Source können flächendeckendes Abhören verhindern. Diplomatie bisher nicht.

Die gute Nachricht

Die gute Nachricht:

- **Wissenschaftlich starke Kryptographie ist auch für übermächtige Geheimdienste nicht brechbar.**
-

Bruce Schneier, Guardian, 6. September 2013

"Trust the math. Encryption is your friend."

Nach Snowden: Nicht stärkste Kryptographie ist schwach.

- In der Kryptographie rechnet man schon immer mit einem Angreifer, der alle Nachrichten abhören kann und Milliarden Dollars zum Brechen der Verschlüsselung zur Verfügung hat.
-
- **Nach Snowden** wissen wir genauer, an welchen Kabelstellen abgehört wird und auf den Cent genau, wie viel Geld für Kryptoangriffe vorhanden ist.[1]

[1] Nicht uninteressant, aber wissenschaftlich betrachtet nur eine Fußnote.

Anpassungen Übungen, 1. Semester

Blatt 1, Aufgabe 2 (18 Punkte) Schlüssellängen

Untersuchen Sie die Angriffsdauer von Brute-Force-Angriffen für die Schlüssellängen von 40, 56, 64, 112 und 128 bit in folgenden Szenarien:

- ...
- In wie vielen Jahren könnte mit einem Etat von **1 Mrd Euro** unter der Annahme der Weitergeltung von Moore's Law eine Schlüsselsuchmaschine gebaut werden, welche eine durchschnittliche Suchzeit von 24 Stunden benötigt?

Kryptomagie = Mathematik + Freie Software

Kryptomagie = Mathematik + Freie Software

Kryptographie ermöglicht durch **Mathematik**
auf einer kleinerfingernagelgroßen Fläche oder
mit einer handvoll Programmzeilen,
Daten sicher selbst gegen eine weltweite
Geheimdienstzusammenarbeit zu verschlüsseln.
Freie Software ermöglicht dies kosten- und hintertürenfrei.

RSA in Python

```
rsa = lambda m : m ** e % n
```

Grundlagen der PK Kryptographie

Bisher: Ein Schlüssel für Sender **und** Empfänger

('Secret-Key' oder "symmetrische" Krypographie

Nun: Ein Teil des Schlüssels ist nur dem Empfänger bekannt. Der auch dem Sender bekannte Teil kann sogar „veröffentlicht“ werden. Man spricht dann von einem **Schlüsselpaar**, bestehend aus

- dem 'Public Key' für den Sender und
- dem 'Private Key', exklusiv für den Empfänger. Ohne Private Key kann nicht entschlüsselt werden!

'Public Key' oder "asymmetrische" Kryptographie)

Szenario

Alice sendet eine Nachricht an Bob.
Beide verwenden symmetrische Kryptographie („secret-key“).

- **Problem:** Schlüsseltransport
  Lösung: Alice schickt den Schlüssel an Bob ???
- **Problem:** T Teilnehmer, T *sehr groß*
  Lösung: Speichern von $T(T-1)/2$ Schlüsseln ???

PK Kryptographie (Geschichte)

1974 „Merkle Puzzles“

1976 Diffie und Hellman

1977 Rivest, Shamir, Adleman (RSA)

Diskrete Logarithmen

- Primzahl p, Zahlen $x, g \in \{2, \ldots, p-2\}$
- Zahl $B = g^x \bmod p$
- Das Problem, zu B, g, p einen Wert x zu finden mit $B = g^x \bmod p$ bezeichnet man als

  „Diskreten Logarithmus von B zur Basis g“.

  Dieses Problem gilt als extrem schwierig.

Diffie-Hellman Schlüsselaustausch

Primzahl p und Generator g festgelegt.

Alice: wählt a als geheimen Schlüssel
öffentlicher Schlüssel: $A = g^a \bmod p$.

Bob: wählt b als geheimen Schlüssel
öffentlicher Schlüssel: $B = g^b \bmod p$.

Geheimer Sitzungsschlüssel:

$$A^b = (g^a)^b = g^{ab} = g^{ba} = (g^b)^a = B^a$$

Das „Diffie-Hellman Problem“

Gegeben A und B, berechne K.

Dieses Problem ist gilt als als extrem schwierig.

Wenn man das DL-Problem effizient lösen kann, kann man das DH-Problem auch effizient lösen. (Warum?)

Ob die Umkehrung auch gilt, ist leider unklar.

Faktorisierung großer ganzer Zahlen

Das **Faktorisieren** großer ganzer Zahlen gilt als extrem schwierige Aufgabe. Dies ist etwas überraschend, da doch die **Multiplikation** einfach ist – sogar der **Test, ob eine Zahl prim ist**, ist ja vergleichsweise einfach.

Anekdote

Frank Cole widerlegt 1903 eine fast 200 Jahre alte Vermutung von Mersenne:

> *Obwohl er die Sonntage dreier Jahre benötigte, um die Faktoren von* $2^{67} - 1$ *zu finden, konnte er innerhalb weniger Minuten, ohne weitere Worte darüber zu verlieren, ein großes Publikum davon überzeugen, daß diese Zahl keine Primzahl war, indem er einfach die Arithmetik der Berechnungen von* $2^{67} - 1$ *und* $193707721 * 761838257287$ *aufschrieb.*

Definition: Gruppe

Sei $\odot$ eine innere Verknüpfung einer nicht leeren Menge G. Es ist $(G, \odot)$ eine **Gruppe**, wenn gilt:

- Assoziativität: $(a \odot b) \odot c = a \odot (b \odot c)$
- Neutrales Element: es gibt ein neutrales Element $\underline{1} \in G$: $\underline{1} \odot a = a$ und
- Inverses Element: zu jedem a existiert ein inverses Element $\overline{a}$: $a \odot \overline{a} = \underline{1}$.

Restklassengruppen

$$
\begin{aligned}
a|b &\Leftrightarrow \text{es gibt } k \text{ mit } k * a = b \\
\mathbb{Z}_n &= \{0, 1, \ldots, n-1\} \quad (*) \\
a \equiv b \bmod n &\Leftrightarrow n|(a-b) \\
\mathbb{Z}_n^* &= \{i \in \mathbb{Z}_n \mid \mathrm{ggT}(i, n) = 1\}
\end{aligned}
$$

- $(\mathbb{Z}_n, +)$ ist eine Gruppe.
- $(\mathbb{Z}_n^*, *)$ ist eine Gruppe.

Konkret:

G	$\odot$	$\underline{1}$	$\overline{a}$
$G = \mathbb{Z}_n$	$+$	0	-1
$G = \mathbb{Z}_n^*$	$*$	1	$a^{-1} \bmod n$

Eulersche φ Funktion

Euler'sche φ-Funktion: $\varphi(n) = \left|\mathbb{Z}_n^*\right|$

Theorem

- *p ist prim* $\Leftrightarrow \varphi(p) = p - 1$.
- $p \neq q$ *sind beide prim*
  $\Rightarrow \varphi(p \cdot q) = \varphi(p) \cdot \varphi(q) = (p-1) \cdot (q-1)$

Satz vom Euler

Theorem (Satz von Euler)

Ist $ggT(a, p) = 1$, *so gilt*

$$a^{\varphi(p)} \equiv 1 \bmod p.$$

Kleiner Satz von Fermat

Theorem (Kleiner Satz von Fermat)

Es sei p eine Primzahl, so gilt

$$a^{p-1} \equiv 1 \bmod p.$$

Fermatsche Primzahltest

- Wähle eine Basis a mit $1 < a < p$ aus.
- Wenn p und a nicht teilerfremd sind, dann return(p keine Primzahl).
- Wenn
$$a^{p-1} \not\equiv 1 \mod p,$$
dann return(p keine Primzahl).
- Sonst ist das Ergebnis keine Aussage

gpg Primzahlentests

Prime-Number-Generator Subsystem Architecture

The primality test works in three steps:

- The standard sieve algorithm using the primes up to 4999 is used as a quick first check.
- A Fermat test filters out almost all non-primes.
- A 5 round Rabin-Miller test is finally used. The first round uses a witness of 2, whereas the next rounds use a random witness.

Primzahlen finden

Aufgabe: Gegeben Werte a und B mit $a < B - 1$. Finde eine zufällige Primzahl p mit $p \geq a$ und $p < B$.
(Typisch: $a = 2^n$, $B = 2a = 2^{n+1}$.)

Lösung:

Wiederhole:

- wähle eine Zufallszahl $z \in \{a, \ldots B - 1\}$,
- teste, ob z prim ist oder zusammengesetzt,

bis z eine Primzahl ist.
Gib z aus.

Primzahlen finden(2)

Frage: Wie effizient ist dieses Verfahren?

- Wie oft wird die Schleife durchlaufen?

  (Häufigkeit der Primzahlen) $(*)$
- Kann man effizient testen, ob z eine Primzahl ist?

  Es gibt sehr effiziente probabilistische Algorithmen für Primzahltests (z.B. „Miller-Rabin“).
  Vor kurzem fanden indische Informatiker sogar einen deterministischen Primzahltest in Polynomialzeit. Dies ist ein interessantes theoretisches Ergebnis („PRIMES $\in$ P“).

Häufigkeit der Primzahlen

Def.: $\pi(x)$ ist die Anzahl der Primzahlen $\leq x$.

Bsp.: $\pi(1) = 0$, $\pi(3) = 2 = \pi(4) = 2$, $\ldots$, $\pi(124) = 30$.

Dem *Primzahlsatz* zufolge gilt

$$\pi(x) \approx \frac{x}{\ln(x)}.$$

Diese Approximation ist sogar recht genau. Für $x \geq 17$ gilt z.B.

$$\frac{x}{\ln(x)} \; < \; \pi(x) \; < \; 1,25506\,\frac{x}{\ln(x)}.$$

RSA

1. Zwei zufällig gewählte *große* Primzahlen p und q.
   Seien $n = pq$ und $e \in \mathbb{Z}^*_{\varphi(n)}$.
   Sei d das multiplikative Inverse von e modulo $\varphi(n)$
2. Schlüssel ist das Tripel (e, d, n).
   Öffentlicher Schlüssel ist das Paar (e, n).
3. Verschlüsselungsfunktion E: $E_{(e,n)}(x) = x^e \bmod n$

- Entschlüsselungsfunktion D: $D_{(e,d,n)}(y) = y^d \bmod n$

Defense Against the Dark Arts

Eduard Snowden, Guardan, 11. März 2014

"Crypto works. It's not an arcane black art. It is a basic protection, the Defense Against the Dark Arts for the digital world. We must implement it, actively research it"