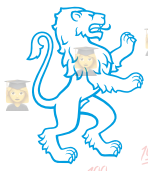

Kantonsschule Im Lee

Informatik

Kryptologie

Skript

Judith Beestermöller

Inhaltsverzeichnis

Kapitel 1

Asymmetrische Kryptosysteme

1.1 Einführung

Bei den Kryptosystemen, die wir im ersten Kapitel gesehen haben wurde jeweils derselbe Schlüssel zum Verschlüsseln wie zum Entschlüsseln verwendet. Diese Kryptosystem werden als **symmetrische** Kryptosysteme bezeichnet. Ein Problem bei symmetrischen Kryptosystemen ist, dass man häufig von der verschlüsselten Nachricht auf den Schlüssel schliessen kann. Dies erlaubt einem Angreifer das Kryptosystem zu knacken. Dies haben wir bereits bei den Kryptosystemen von Caesar und Vigenère gesehen.

Bei einem **asymmetrischen Kryptosysteme**, oft auch **Public-Key-Kryptosystem** genannt, nutzt man ein Paar mathematisch verbundener Schlüssel. Einen öffentlichen Schlüssel (**Public-Key**), welche für jeden zugänglich ist, und eine privaten Schlüssel (**Private Key**), welcher geheim gehalten werden muss.

Man kann sich ein asymmetrisches Kryptosystem ein wenig so vorstellen wie einen Briefkasten an einem Haus. Es ist kein Geheimnis, wo sich der Briefkasten befindet. Jedoch genügt dieses Wissen einem Angreifer nicht, um Briefe, die in den Briefkasten geworfen werden, zu lesen. Nur der Besitzer des Briefkastens kann die Briefe dem Briefkasten entnehmen.

Die Grundidee von einem asymmetrischen Kryptosystem lautet wie folgt:

Alice und Bob generieren jeweils ein Schlüsselpaar, einen öffentlichen Schlüssel (🔒), der von allen Personen gesehen und verwendet werden kann, und einen privaten Schlüssel (🔑), der nur im Besitz von Alice bzw. Bob ist. Wenn Alice eine Nachricht an Bob senden will, verwendet sie Bobs *öffentlichen* Schlüssel, um die Nachricht zu verschlüsseln. Bob verwendet seinen *privaten* Schlüssel zur Entschlüsselung, also den Schlüssel, auf den *nur* er Zugriff hat (siehe 1.1).Wichtig ist, dass die Nachricht die Alice an Bob schickt, mit dem öffentlichen Schlüssel nicht entschlüsselt werden kann.

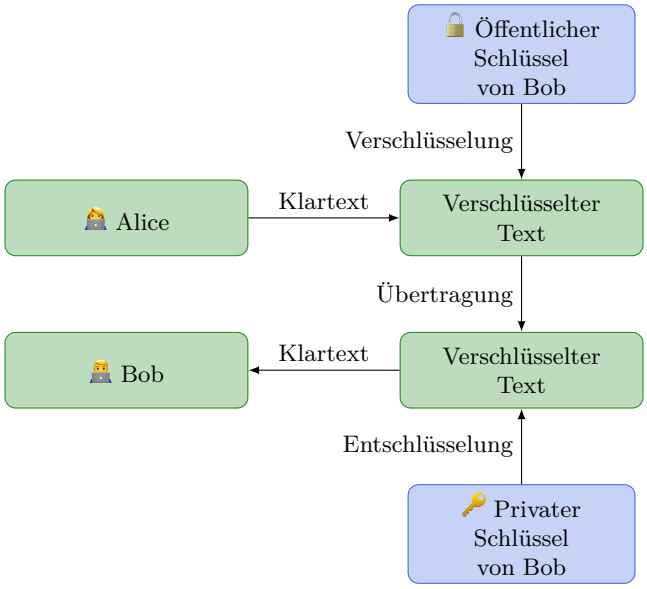

Abbildung 1.1: Prinzip der Public-Key-Verschlüsselung

Das Diffie-Hellman-Merkel Protokoll, welches wir uns zuletzt angesehen haben, ist eine Public-Key-Kryptosystem.

Aufgabe 1.1

Was ist der Public-Key und was der Private Key bei Diffie-Hellman-Merkel?

Lösungsvorschlag zu Aufgabe 1.1

Der Public Key ist für Alice und Bob gleich und besteht aus (g, p). Alices Private Key ist a und Bobs Private Key ist b.

Ein asymmetrisches Kryptosystem kann auch verwendet werden, um die **Authentizität** einer Nachricht zu beweisen. D.h. Alice kann mit einem asymmetrischen Kryptosystem beweisen, dass die Nachricht tatsächlich von ihr und niemanden anderes stammt. Hierfür verschlüsselt Alice die Nachricht mit ihrem Private Key. Jede Person kann überprüfen, dass die Nachricht von Alice stammt, indem er oder sie Alices öffentlich zugänglichen Public Key verwendet, um die Nachricht wieder zu entschlüsseln. Da niemand ausser Alice den Private Key kennt, muss die Nachricht von Alice stammen. Dieses Verfahren wird auch als **Digitale Signatur** bezeichnet.

Damit Bob am Ende wieder die entschlüsselte Nachricht erhält, muss der Private Key den Public Key rückgängig machen. In der Mathematik nennt man eine solche Funktion **Umkehrfunktion** oder **Inversfunktion**.

Definition 1.1 (Umkehrfunktion)**:**
Sei $f : A \to B$ eine Funktion. Die Funktion $g : B \to A$ ist die Umkehrfunktion von F wenn

$$g(f(a)) = ahspace0.5cm\forall a \in A.$$

Häufig schreibt man die Inversefunktion einer Funktion f auch als f^{-1}.

Ein asymmetrisches Kryptosystem kann theoretisch mit jeder invertierbaren Funktion ausgeführt

werden.

Aufgabe 1.2

Wählen Sie und Ihr Nachbar gemeinsam eine beliebige invertierbare Funktion f aus und bestimmen Sie ihre Inverse f^{-1}. Verschlüsseln Sie nun jeder eine Nachricht, und senden Sie diese an Ihren Nachbarn. Entschlüsseln Sie danach die Nachricht, die Sie erhalten haben.

In Aufgabe 1.2 haben wir gezeigt, dass eine Menge Funktionen in Frage kommen, mit denen Alice und Bob ihre Nachrichten ver-/ bzw. entschlüsseln können. Da aber Eve ebenfalls den öffentlichen Schlüssel kennt, muss die Funktion, die dem asymmetrischem Kryptosystem zu Grunde liegt, so gewählt werden, dass Eve das Inverse nicht berechenen kann.

In der Mathematik nennen wir solche Funktionen **Einwegfunktionen**. Eine Einwegfunktion ist eine mathematische Funktion, die komplexitätstheoretisch „leicht“ berechenbar, aber „schwer“ umzukehren ist. Eine mathematische Einwegfunktion $F : X \to Y$ muss folgende Eigenschaften aufweisen:

- Man kann die Funktion $f(x) = y$ “einfach” berechnen; d.h. der Computer kann dies effizient.
- Die Umkehrfunktion $f^{-1}(y) = x$ ist “schwierig” zu berechnen, und kann auch von einem Computer nicht effizient berechnet werden.

Jedes asymmetrische Kryptoverfahren verwendet eine Einwegfunktion für die Verschlüsselung. In **??** haben wir bereits den diskreten Logarithmus als Einwegfunktion kennengelernt. Im nächsten Abschnitt lernen wir ein weiters asymmetrischen Kryptosystem kennen, welches auf dem **Faktorisierungsproblem** aufbaut. Das Faktorisierungsproblem ist die Aufgabe, eine gegebene Zahl in ihre Primfaktoren zu zerlegen. Derzeit ist keinen schnellen Algorithmus, welcher grosse Zahlen effizient in seine Primfaktoren zerlegen kann.

Wichtig zu wissen ist, dass es für alle derzeit bekannten Einwegfunktionen derzeit keinen Beweis für die Schwierigkeit ihrer Invertierung gibt. Das heisst, derzeit ist kein effizenter Algorithmus bekannt, der das Inverse der Einwegfunktion effizient bestimmt. Wir können aber auch nicht beweisen, dass dies wirklich der Fall ist. Sollte aber ein solcher Algorithmus gefunden werden, so ist jedes asymmetrische Kryptosystem, welches darauf aufbaut, hinfällig!

1.2 RSA

Eines der auch heute am häufigsten benutzten asymmetrischen Kryptosysteme ist RSA, benannt nach seinen Erfindern Ron Rivest, Adi Shamir und Leonard Adleman.

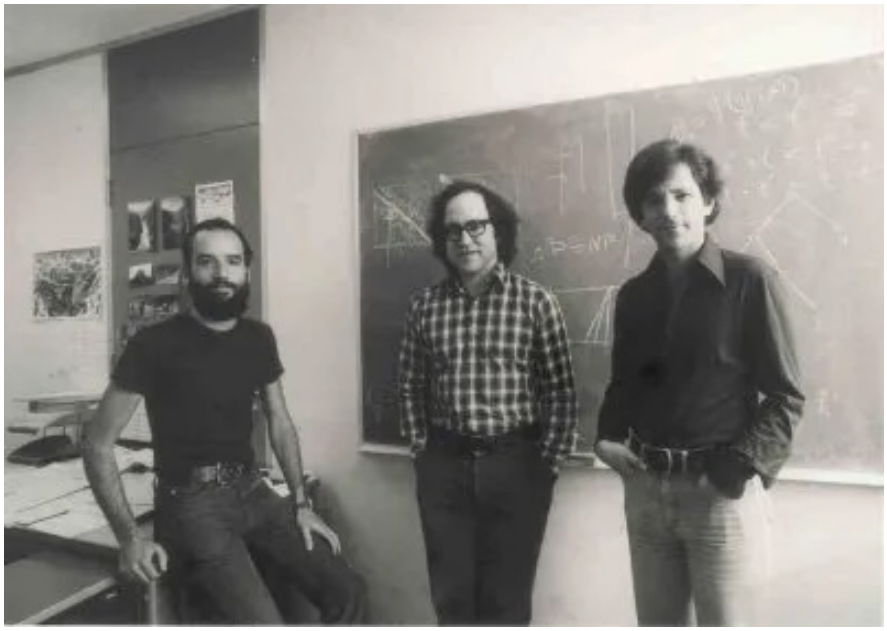

Abbildung 1.2: Ron Rivest, Adi Shamir und Leonard Adleman, die Erfinder des RSA-Verfahrens

Das RSA-Verfahren besteht aus folgenden Schritten:

Vorgehen 1.1: 1. **Schlüsselerzeugung:**
- Wählen Sie zwei (möglichst grosse) Primzahlen p und q.
- Berechnen Sie das RSA-Modul $n = p \cdot q$.
- Berechnen Sie die Eulersche Funktion $\varphi(n) = (p-1)(q-1)$.
- Wählen Sie den *Verschlüsselungsexponenten* e, so dass dieser teilerfremd zu $\varphi(n)$ und kleiner als $\varphi(n)$ ist. Dies bedeutet, dass der GGT von e und $\varphi(n)$ 1 ist ($\text{ggT}(e, \varphi(n)) = 1$).
- Berechnen Sie den *Entschlüsselungexponenten* d, so dass $(e \cdot d) \mod \varphi(n) = 1$ (privater Schlüssel).
- Die Zahlen p, q und $\varphi(n)$ werden nun nicht mehr benötigt und können gelöscht werden.

2. **Verschlüsselung:**
   - Der Absender verschlüsselt eine **Klartext-Nachricht m** (als Zahl) mit dem **öffentlichen Schlüssel** $(\mathbf{e}, \mathbf{n})$:

$$c = \mathbf{m}^e \mod n$$

   - Das Ergebnis c ist der Kryptotext.

3. **Entschlüsselung:**
   - Der Empfänger entschlüsselt die **verschlüsselte Nachricht c** mit dem **privaten Schlüssel** $(\mathbf{d}, \mathbf{n})$:

$$\mathbf{m} = c^d \mod n$$

   - Das Ergebnis **m** ist die ursprüngliche Nachricht.

Beispiel 1.1:
Um den RSA Algorithmus besser zu verstehen, rechnen wir ein einfaches Beispiel mit kleinen

Zahlen durch:

1. Wir wählen zwei (für diese Übung kleine) Primzahlen aus, $p = 3$ und $q = 11$. Daraus folgt:

$$\begin{aligned} n &= p \cdot q \\ &= 3 \cdot 11 \\ &= 33 \end{aligned}$$

$$\begin{aligned} \varphi(n) &= (p-1)(q-1) \\ &= 2 \cdot 10 \\ &= 20 \end{aligned}$$

2. Wir wählen $e = 3$, da $\text{ggT}(3, 20) = 1$.
3. Wir berechnen d, so dass $(e \cdot d) \mod \varphi(n) = 1$ ergibt:

$$(d \cdot 3) \mod 20 = 1 \quad \rightarrow \quad d = 7.$$

4. Der öffentliche Schlüssel ist $(e, n) = (3, 33)$, der private Schlüssel $(d, n) = (7, 33)$.

Verschlüsselung: Die Nachricht sei das Wort "Code", welches wir darstellen durch die Position der Buchstaben im Alphabet $m = 3, 15, 4, 5$. Berechnen Sie für jeden Buchstaben (hier nur für "C", also 3, gezeigt):

$$\begin{aligned} c &= m^e \mod n \\ &= 3^3 \mod 33 \\ &= 27 \mod 33 \\ &= 27. \end{aligned}$$

Der erste Buchstabe des Kryptotext wird also verschlüsselt als $c = 27$.

Entschlüsselung:

$$\begin{aligned} m &= c^d \mod n \\ &= 27^7 \mod 33 \\ &= 3 \end{aligned}$$

Daraus ergibt sich $m = 3$, also die ursprüngliche Nachricht.

Aufgabe 1.3

Alice möchte eine Nachricht an Bob mit RSA verschlüsseln. Bob wählt die Hilfs-Primzahlen $p = 5$ und $q = 7$. Generieren Sie den öffentlichen Schlüssel (e, n) und den privaten Schlüssel (d, n) von Bob, indem Sie folgende Schritte ausführen:

1. Berechnen Sie n und $\varphi(n)$.
2. Sie wählen e und d aus.

Verschlüsseln Sie nun die Nachricht $m = 9$ mit dem öffentlichen Schlüssel (e, n).

Entschlüsseln Sie danach die verschlüsselte Nachricht c wieder mit dem privaten Schlüssel (d, n).

✔ Lösungsvorschlag zu Aufgabe 1.3

1. $n = p \cdot q = 5 \cdot 7 = 35$, $\varphi(n) = (p-1)(q-1) = 4 \cdot 6 = 24$.
2. Für e können wir beispielsweise 5 wählen, da 5 teilerfremd mit 24 ist.
3. $(5 \cdot d) \mod 24 = 1$. Dies finden wir beispielsweise mit $d = 5$.
4. Verschlüsselung:

$$\begin{aligned} c &= m^e \mod n \\ &= 9^5 \mod 35 \\ &= 4 \end{aligned}$$

.

5. Entschlüsselung:

$$\begin{aligned} m &= c^d \mod n \\ &= 4^5 \mod 35 \\ &= 9 \end{aligned}$$

Nehmen wir uns einen Moment Zeit, um zu verstehen, warum es so schwierig ist, das RSA Protokoll zu brechen: Wir wissen, dass $d \cdot e \mod \varphi(n) = 1$. Das bedeutet, dass wenn Eve $\varphi(n)$ herausfindet, dann kann sie einfach d bestimmen.

Die Funktion $\varphi : \mathbb{N}_+ \to \mathbb{N}_+$ ist definiert als $\varphi(x) :=\mid \{a \in \mathbb{N} \mid 1 \leq a \leq x \wedge ggt(a,x) = 1\} \mid$. Das heisst, sie zählt die Anzahl natürlicher Zahlen, die teilerfremd zu x sind. Für eine Primzahl p ist es einfach zu sehen, dass $\varphi(p) = p - 1$. Man kann auch zeigen, dass wenn a und b teilerfremd sind, dann gilt für $c = a \cdot b$ das $\varphi(c) = (a-1)(b-1)$.

Im RSA Protokoll sind p und q Primzahlen und somit teilerfremd. Daher gilt $\varphi(n) = (p-1)(q-1)$. Die Zahl $n = p \cdot q$ ist Teil des öffentlichen Schlüssels und somit bekannt. Die Schwierigkeit des Faktorisierungsproblems besteht dadrin, dass es extrem schwierig ist, p und q herauszufinden, auch wenn n bekannt ist. Auch können wir $\varphi(n)$ nur schwierig bestimmen, da wir dafür alle Teiler von n kennen müssten.

Aufgabe 1.4

Versuchen Sie RSA zu brechnen. Der öffentliche Schlüssel $(e, n) = (7, 33)$. Die verschlüsselte Nachricht, die Sie erhalten, ist 29. Was war die ursprüngliche Nachricht?

✔ Lösungsvorschlag zu Aufgabe 1.4

Wenn $n = 33$, dann müssen $p = 3$ und $q = 11$ (oder umgekehrt) und $\varphi(n) = 20$. Um d zu bestimmen lösen wir $7d \mod 20 = 1 \Rightarrow d = 3$ $m = 29^3 \mod 33 = 2$.

1.3 Lernziele Asymmetrische Kryptosysteme und RSA

- ☐ Ich weiss, was symmetrische und asymmetrische Kryptosysteme sind, und kenne mindestens zwei Beispiele für diese zwei Arten von Kryptosystemen.

- ☐ Ich weiss, wie man mit einem asymmetrischem Kryptosystem die Echtheit einer Nachricht beweisen kann.
- ☐ Ich weiss, was eine Einwegfuntkion ist, und ich kann erklären, warum asymmetrische Kryptosysteme Einwegfunktionen verwenden müssen. Ich kenne mindestens zwei Beispeiele für Einwegfunktionen.
- ☐ Ich weiss, wie das RSA Protokoll Funktioniert, und kann für gegebene p und q ein Schlüsselpaar erstellen.