1. Public-Key-Kryptographie

Mit Kryptographie bezeichnet man das Studium mathematischer Techniken, welche die Sicherheit von Informationen betreffen. In der Vergangenheit lag die Bedeutung der Kryptographie vor allem auf dem militärischen und diplomatischen Sektor. Dabei wurden sogenannte symmetrische kryptographische Verfahren verwendet, um geheime Nachrichten zu verschlüsseln. Bevor die verschlüsselten Botschaften übermittelt werden können, einigen sich Sender und Empfänger hier auf einen gemeinsamen geheimen Schlüssel (bei einem persönlichen Treffen, durch einen Kurier ...). Natürlich besteht dabei das Risiko, daß der Schlüssel belauscht oder gestohlen wird. Mit Hilfe dieses Schlüssels kodiert der Sender die geheimen Botschaften und verschickt sie dann durch einen eventuell nicht abhörsicheren Kanal (Brief, Radio ...) an den Empfänger. Dieser benutzt den Schlüssel, um aus dem erhaltenen Kryptogramm wieder die ursprüngliche Botschaft zu machen.

Ein Problem bei diesen symmetrischen Verschlüsselungsverfahren ist der geheime Schlüsselaustausch, der vor dem Senden verschlüsselter Botschaften erfolgen muß. Wenn dazu immer persönliche Treffen oder reitende Boten nötig sind, so müssen diese Verfahren notwendig auf eine kleine, feste Gruppe von Benutzern beschränkt bleiben. In unserer modernen vernetzten Welt bietet sich der Kryptographie allerdings noch ein ganz anderes Anwendungsgebiet. Hier möchte eine große, wechselnde Gruppe von Benutzern per Internet einkaufen und dabei persönliche Daten geheimhalten, elektronische Geldgeschäfte sicher abschließen, Nachrichten digital signieren usw.

Das wirft folgende Probleme auf:

- Wie kann man über öffentliche Kanäle Schlüssel austauschen (die dann für symmetrische Verfahren verwendet werden können)?

A. Werner, *Elliptische Kurven in der Kryptographie*

- Wie verschlüsselt man Nachrichten, ohne vorher Schlüssel auszutauschen?
- Wie kann man sich durch eine "digitale Unterschrift" ausweisen?

Eine Antwort auf all diese Fragen gibt die Public-Key-Kryptographie oder asymmetrische Kryptographie. Sie geht auf Ideen von Diffie und Hellman aus den siebziger Jahren zurück. Bei Public-Key-Verfahren hat jeder Nutzer A einen öffentlichen Schlüssel, den jeder einsehen kann, und einen privaten Schlüssel, den niemand sonst kennt. Nachrichten werden hier mit Hilfe von Funktionen $x \mapsto f(x)$ verschlüsselt, die zwar leicht zu berechnen, aber nur mit Kenntnis des privaten Schlüssels zu invertieren sind. Kennt man also nur $f(x)$, so ist es praktisch unmöglich, x zu berechnen, es sei denn, man kennt den privaten Schlüssel des rechtmäßigen Empfängers. Damit wird sichergestellt, daß - obwohl jeder $f(x)$ mithören kann - nur eine Person daraus wieder x ableiten kann.

In den folgenden Abschnitten werden wir zwei wichtige Beispiele für solche Einwegfunktionen kennenlernen. Das erste ist das sogenannte RSA-Verfahren, so benannt nach seinen Entwicklern Rivest, Shamir und Adleman. Das zweite Beispiel ist entscheidend für unsere Zwecke. Hier ist f die Funktion "k-tes Vielfaches" in einer endlichen abelschen Gruppe. Die kryptographische Anwendung elliptischer Kurven, die Thema dieses Buches ist, ergibt sich, indem man als Gruppe eine elliptische Kurve über einem endlichen Körper nimmt. Im Rahmen dieses zweiten Beispiels stellen wir Methoden zum Schlüsselaustausch, zur Verschlüsselung und für digitale Unterschriften vor, die die oben angesprochenen Probleme lösen.

1.1 RSA

Wir geben nur einen kurzen Abriß des Verfahrens, für weitere Details sei auf [Bu], 7.2 verwiesen. Hier sehen der öffentliche und der private Schlüssel folgendermaßen aus: Ein Nutzer B, sagen wir Bob, wählt zwei verschiedene (große) Primzahlen p und q und berechnet $n = pq$. Zusätzlich wählt Bob eine Zahl e zwischen 1 und $\varphi(n) = (p-1)(q-1)$, die teilerfremd zu $\varphi(n)$ ist, wobei φ die Eulersche φ-Funktion bezeichnet (siehe 6.3). Er berechnet eine weitere Zahl d zwischen 1 und $\varphi(n)$, so daß

$$ed \equiv 1 \bmod \varphi(n)$$

ist. Dazu kann Bob den erweiterten Euklidischen Algorithmus (siehe 6.1) benutzen, mit dessen Hilfe sich Zahlen d und y mit $1 = de + y\varphi(n)$ bestimmen lassen. Hier ist entscheidend, daß er p und q kennt und damit $\varphi(n)$ berechnen kann.

Bobs öffentlicher Schlüssel ist das Paar (n, e), sein privater Schlüssel ist die Zahl d. Ein weiterer Nutzer A, sagen wir Alice, will Bob eine Nachricht schicken. Sie besorgt sich zunächst Bobs öffentlichen Schlüssel (n, e) und konstruiert damit die Verschlüsselungsfunktion

$$f_B : \mathbb{Z}/n\mathbb{Z} \to \mathbb{Z}/n\mathbb{Z}, \text{ definiert durch } f_B(x) = x^e,$$

Will Alice Bob also die geheime Nachricht $x \in \mathbb{Z}/n\mathbb{Z}$ zukommen lassen, so berechnet sie $f_B(x) = x^e$ und schickt diesen Wert an Bob. Bob empfängt diese chiffrierte Nachricht x^e und berechnet damit $(x^e)^d = x^{ed}$. Wir behaupten nun, daß $x^{ed} = x$ in $\mathbb{Z}/n\mathbb{Z}$ ist. Da nämlich $ed \equiv 1 \bmod \varphi(n)$ ist, gibt es eine ganze Zahl a mit $ed = 1 + a\varphi(n)$. Falls x teilerfremd zu p ist, so gilt

$$x^{ed} = x \cdot x^{a\varphi(n)} \equiv x \bmod p$$

nach dem kleinen Satz von Fermat, da $\varphi(p) = p - 1$ ein Teiler von $\varphi(n)$ ist. Falls x hingegen ein Vielfaches von p ist, so gilt $x^{ed} \equiv x \bmod p$, da beide Seiten modulo p Null sind. Genauso zeigt man $x^{ed} \equiv x \bmod q$. Daraus folgt in der Tat nach dem Chinesischen Restsatz (siehe 6.2), daß $x^{ed} \equiv x \bmod n$ ist. Somit erhält Bob also die ursprüngliche Botschaft x zurück.

Bei diesem Verfahren ist tatsächlich $f_B(x)$ und auch Bobs Entschlüsselung $x^e \mapsto (x^e)^d$ relativ leicht zu berechnen. Die Aufgabe, aus $f_B(x) = x^e$ ohne Kenntnis von d die Botschaft x zu berechnen, ist hingegen ein schwieriges mathematisches Problem. Man kann natürlich versuchen, aus der Kenntnis des öffentlichen Schlüssels (n, e) den privaten Schlüssel d zu ermitteln. Es ist klar, daß ein Angreifer, der die Primfaktoren p und q von n ermitteln kann, auch im Besitz von d ist: er kann ja einfach wie Bob den erweiterten Euklidischen Algorithmus benutzen, um d zu berechnen. Die Aufgabe, eine gegebene Zahl n in ihre Primfaktoren zu zerlegen, heißt Faktorisierungsproblem. Im Rahmen des RSA-Verfahrens ist das Faktorisierungsproblem für n genauso schwierig wie die Bestimmung des privaten Schlüssels aus dem öffentlichen Schlüssel (siehe [Bu], 7.2.4). In der Praxis muß man daher p und

q so groß wählen, daß alle bekannten Faktorisierungsverfahren noch zu langsam wären. Man kann allerdings bisher nicht beweisen, dass das RSA-Verfahren sicher ist. Weder ist nämlich bekannt, ob man nicht auch ohne Kenntnis des geheimen Schlüssels d die Nachricht x^e entschlüsseln kann, noch kann man sicher sein, daß es nicht eines Tages so schnelle Faktorisierungsverfahren gibt, daß p und q für praktische Zwecke zu groß gewählt werden müßten.

1.2 Diskreter Logarithmus

In einer Reihe von wichtigen Verfahren der Public-Key-Kryptographie ist die Verschlüsselungsfunktion f definiert mit Hilfe einer endlichen abelschen Gruppe G, die wir additiv schreiben, d.h. die Gruppenoperation ist

$$(P, Q) \mapsto P + Q$$

und das neutrale Element bezeichnen wir mit 0. Allgemein bekannt sei hier G und ein Element $P \in G$. Ferner sei n die Ordnung der von P erzeugten zyklischen Untergruppe

$$\langle P \rangle = \{kP : k \in \mathbb{Z}\}$$

von G, d.h. n ist die kleinste natürliche Zahl mit $nP = 0$. Nun ist f die Funktion

$$f : \mathbb{Z}/n\mathbb{Z} \longrightarrow \langle P \rangle$$

$$k \bmod n \longmapsto kP.$$

Hier kommen nur solche Gruppen G und Elemente $P \in G$ infrage, für die einerseits kP zu gegebenem k leicht zu berechnen ist, andererseits aber die Bestimmung von k bei bekanntem kP hinreichend schwierig ist. Letzteres heißt auch "Problem des diskreten Logarithmus" in G:

Problem des diskreten Logarithmus (DL-Problem): Bestimme zu den gegebenen Daten $G, P \in G$, $n = \mathrm{ord}(P)$ und $Q \in \langle P \rangle$ das Element $k \bmod n$ in $\mathbb{Z}/n\mathbb{Z}$ mit

$$Q = kP.$$

Die Bezeichnung "Logarithmus" erklärt sich daraus, daß wir hier eine Umkehrabbildung zu der Funktion $k \mapsto kP$ suchen. Hätten wir nämlich

die Verknüpfung in G multiplikativ geschrieben (also $(P,Q) \mapsto P \cdot Q$), so wäre dies gerade die Exponentialfunktion $k \mapsto P^k$. Wir benutzen hier allerdings immer die additive Schreibweise, weil dies bei elliptischen Kurven so üblich ist.

Gegeben sei nun eine endliche abelsche Gruppe G und ein $P \in G$, so daß das DL-Problem schwer zu lösen ist. Das bedeutet unter anderem, daß die Ordnung n von P hinreichend groß sein muß. (Sonst könnte man ja alle Elemente $P, 2P, 3P, \ldots$ durchprobieren.)

1.2.1 Diffie-Hellman-Schlüsselaustausch

Dies ist ein Verfahren, mit dem Alice und Bob durch einen öffentlichen Kanal einen geheimen Schlüssel austauschen, der dann für ein symmetrisches Verschlüsselungsverfahren verwendet werden kann. Allgemein zugänglich seien hier die Daten G, n und P. Nun passiert folgendes:

1) Alice wählt zufällig eine ganze Zahl d_A in $\{1, 2, \ldots, n-1\}$ und schickt das Gruppenelement $d_A P$ an Bob.
2) Bob wählt seinerseits zufällig eine Zahl d_B in $\{1, 2, \ldots, n-1\}$ und schickt das Gruppenelement $d_B P$ an Alice.
3) Alice berechnet mit ihrer Zahl d_A das Element $d_A(d_B P) = d_A d_B P$; Bob berechnet $d_B(d_A P) = d_B d_A P = d_A d_B P$.

Nun sind also beide im Besitz des Elementes $d_A d_B P$, ohne daß Bob Alice' Geheimzahl d_A oder daß Alice Bobs Geheimzahl d_B kennt. Wir nehmen nun einmal an, die Spionin Eva versucht in den Besitz des Geheimnisses $d_A d_B P$ zu gelangen. Sie kennt die Gruppe G und das Element P und hat die Übertragungen $d_A P$ und $d_B P$ mitgehört. Aus diesen Daten möchte sie nun das Element $d_A d_B P$ berechnen. Diese Aufgabe heißt auch Diffie-Hellman-Problem:

Diffie-Hellman-Problem: Berechne zu zwei Elementen kP und lP in $\langle P \rangle$ das Element klP in $\langle P \rangle$.

Kann Eva das Diffie-Hellman-Problem lösen, so ist sie im Besitz des geheimen Elementes $d_A d_B P$. Es ist klar, daß Eva das Diffie-Hellman Problem lösen kann, wenn sie das DL-Problem in G lösen kann. Bisher ist allerdings nicht bekannt, ob auch die Umkehrung gilt, d.h. ob eine Gruppe, in der das DL-Problem schwer zu lösen ist, auch die Eigenschaft hat, daß das Diffie-Hellman-Problem schwer zu lösen ist.

Das Lösen des Diffie-Hellman-Problems ist allerdings für Eva nicht die einzige Möglichkeit, dieses Verfahren anzugreifen. Sie könnte auch versuchen, sich erst als Alice auszugeben und so mit Bob wie oben einen Schlüssel auszutauschen und dann als Bob getarnt mit Alice einen Schlüssel auszutauschen. Gelingt dies, so muß sie nur noch die verschlüsselten Nachrichten von Alice an Bob abfangen, sie mit ihrem Alice-Schlüssel dekodieren und mit ihrem Bob-Schlüssel wieder verschlüsseln und an Bob weiterleiten. Auf diese Weise kann sie die gesamte geheime Korrespondenz abhören. Dies nennt man auch "Man-in-the-middle Attacke". Es ist also hier entscheidend, daß Alice und Bob sicher sein können, wirklich mit dem angegebenen Absender zu kommunizieren.

1.2.2 ElGamal-Verschlüsselung

Dieses Verfahren, wie auch das folgende, wurde von T. ElGamal entwickelt (siehe [EG]). Jeder Nutzer wählt hier zufällig eine ganze Zahl d in $\{1,\ldots,n-1\}$ und erzeugt damit seinen öffentlichen Schlüssel dP. Die Zahl d ist sein privater Schlüssel. Alice möchte eine geheime Botschaft an Bob schicken. Wir nehmen an, daß diese Nachricht ein Element m aus G ist, d.h. daß man auf bekannte Weise Nachrichten (oder zumindest Teilstücke davon) mit Elementen aus G identifizieren kann. Nun passiert folgendes:

1) Alice wählt zufällig eine ganze Zahl k in $\{1,\ldots,n-1\}$ und berechnet $Q = kP$. Sie besorgt sich Bobs öffentlichen Schlüssel d_BP und berechnet damit $R = k(d_BP) + m$.
2) Dann schickt sie das Paar (Q,R) an Bob.
3) Bob nimmt seinen privaten Schlüssel d_B, um $d_BQ = d_BkP$ zu berechnen. Nun kann er die Nachricht m ermitteln, indem er $R - d_BQ = kd_BP + m - d_BkP = m$ ausrechnet.

Die Spionin Eva kennt in diesem Fall natürlich G, n und P, sowie Bobs öffentlichen Schlüssel d_BP. Außerdem hat sie die Daten $Q = kP$ und $R = kd_BP + m$ mitgehört. Sie kann nun m berechnen genau dann, wenn sie kd_BP berechnen kann. Dazu muß sie ein Diffie-Hellman Problem lösen.

Für die Sicherheit des ElGamal-Verfahrens ist es wichtig, daß Alice für jede Nachricht, die sie verschicken will, ein neues k wählt. Falls

sie nämlich dasselbe k benutzt, um die Nachrichten m_1 und m_2 zu verschlüsseln, so kann Eva aus den Übertragungen $(Q, R_1 = kd_BP + m_1)$ und $(Q, R_2 = kd_BP + m_2)$ die Differenz $m_1 - m_2 = R_1 - R_2$ berechnen und so m_2 ermitteln, falls sie die Nachricht m_1 schon kennt.

1.2.3 ElGamal-Signatur

Alice will eine Nachricht m an Bob digital unterschreiben. Dazu verwendet sie den gleichen öffentlichen Schlüssel d_AP und privaten Schlüssel d_A wie in 1.2.2.

Es sei $\mathcal{M}$ die Menge aller möglichen Nachrichten (etwa beliebig lange Folgen von Nullen und Einsen). Dann benötigen wir eine allgemein bekannte Hashfunktion, d.h. eine Funktion

$$h : \mathcal{M} \longrightarrow \{0, 1, \ldots, n-1\}.$$

Diese Funktion h muß folgende Eigenschaften haben:

i) Es ist praktisch unmöglich, Urbilder unter h zu berechnen, d.h. zu gegebenem $x \in \{0, 1, \ldots, n-1\}$ ein $m \in \mathcal{M}$ zu finden mit $h(m) = x$.

ii) h ist kollisionsresistent, d.h. es ist praktisch unmöglich, zwei verschiedene Elemente m und m' in $\mathcal{M}$ zu finden mit $h(m) = h(m')$.

Außerdem brauchen wir eine allgemein bekannte, effektiv berechenbare Bijektion $\psi : \langle P \rangle \to \{0, 1, \ldots, n-1\}$.

Nun passiert folgendes:

1) Alice wählt zufällig eine zu n teilerfremde Zahl k zwischen 1 und $n-1$ und berechnet das Gruppenelement $r = kP$.

2) Dann berechnet sie das Inverse k^{-1} von k in $\mathbb{Z}/n\mathbb{Z}$ sowie das Element $s = k^{-1}(h(m) - \psi(r)d_A)$ in $\mathbb{Z}/n\mathbb{Z}$..

3) Alice schickt die Nachricht m zusammen mit ihrer Unterschrift (r, s) an Bob.

Wenn Bob prüfen möchte, daß Alice' Unterschrift echt ist, so berechnet er aus $m, (r, s)$ und Alice' öffentlichem Schlüssel d_AP das Gruppenelement $\psi(r)d_AP + sr$ sowie den Hashwert $h(m)$. Er akzeptiert Alice' Unterschrift, wenn

$$\psi(r)d_AP + sr = h(m)P$$

ist. Diese Prüfung klappt offenbar nur dann, wenn

$$\psi(r)d_A + sk \equiv h(m) \bmod n$$

ist, also s wie in 2) gewählt ist!

Angenommen, die Betrügerin Eva möchte Alice' Unterschrift fälschen. Dazu muß sie r und s finden, so daß $\psi(r)d_AP + sr = h(m)P$ ist. Wählt Eva etwa ein beliebiges k und versucht, zu $r = kP$ ein geeignetes s zu finden, so muß sie ein DL-Problem in $\langle P \rangle$ lösen.

Auch hier ist es wichtig, daß Alice für jede Unterschrift ein neues k wählt. Falls sie nämlich die Unterschriften (r_1, s_1) für m_1 und (r_2, s_2) für m_2 mit demselben k erzeugt, so ist $r_1 = r_2$ und $s_1 - s_2 = k^{-1}(h(m_1) - h(m_2)) \bmod n$. Wenn $h(m_1) - h(m_2)$ invertierbar in $\mathbb{Z}/n\mathbb{Z}$ ist, so kann Eva hieraus $k \bmod n$ bestimmen. Da $\psi(r_1)d_A \equiv h(m_1) - s_1k \bmod n$ ist, kann Eva nun den privaten Schlüssel d_A von Alice berechnen, falls $\psi(r_1)$ invertierbar in $\mathbb{Z}/n\mathbb{Z}$ ist.

Wofür braucht man die oben beschriebenen Eigenschaften der Hashfunktion h? Falls Eva Urbilder von h berechnen kann, so kann sie Alice' Unterschrift folgendermaßen fälschen: Sie wählt eine beliebige ganze Zahl j und berechnet $r = jP - d_AP$. Dann setzt sie $s = \psi(r)$ und sucht ein m mit $h(m) \equiv \psi(r)j \bmod n$. Nun ist (r, s) eine gültige Unterschrift für die Nachricht m! Falls hingegen h nicht kollisionsresistent ist, und Eva ein $m' \in \mathcal{M}$ mit $h(m) = h(m')$ findet, so kann sie Alice' Unterschrift unter m' fälschen, wenn sie im Besitz einer gültigen Unterschrift für m ist. Diese beiden Unterschriften lassen sich nämlich nicht unterscheiden.

In der Praxis ist die Annahme, daß die Abbildung ψ eine Bijektion ist, zu strikt. Es genügt, daß die Urbildmenge jedes Elementes in $\{0, \ldots, n-1\}$ hinreichend klein ist. Außerdem wird meist eine Variante des ElGamal-Verfahrens verwendet (vgl. 5.3).

1.3 Geeignete Gruppen

Um diese Verschlüsselungsverfahren, die auf dem DL-Problem basieren, anwenden zu können, benötigen wir also endliche abelsche Gruppen, in denen das DL-Problem schwer zu lösen ist. Nun gibt uns jeder

endliche Körper $\mathbb{F}_q$ mit q Elementen zwei endliche abelsche Gruppen an die Hand, nämlich die additive Gruppe $\mathbb{F}_q$ und die multiplikative Gruppe $\mathbb{F}_q^\times$.

Die Gruppe $\mathbb{F}_q$ ist für unsere Zwecke völlig ungeeignet. Haben wir nämlich hier ein $P \in \mathbb{F}_q$ und ein $Q = kP$ aus der von P erzeugten zyklischen Gruppe $\langle P \rangle = \{0, P, 2P, 3P, \ldots\}$ vorliegen, so ist entweder $P = Q = 0$ oder aber

$$k = \frac{Q}{P}$$

der diskrete Logarithmus, der sich einfach durch eine Division in dem Körper $\mathbb{F}_q$ berechnen läßt.

Die multiplikative Gruppe $\mathbb{F}_q^\times = \mathbb{F}_q \backslash \{0\}$ ist für geschickt gewähltes q schon besser geeignet. Noch besser geeignet sind allerdings die "Punktegruppen" $E(\mathbb{F}_q)$ zu elliptischen Kurven über $\mathbb{F}_q$, denn es gibt Algorithmen zur Lösung des DL-Problems in $\mathbb{F}_q^\times$, die sich (bisher?) nicht auf solche Gruppen $E(\mathbb{F}_q)$ übertragen lassen. Darauf werden wir in 5.2 näher eingehen.

Zuvor sollen elliptische Kurven definiert und untersucht werden. Es handelt sich hierbei um interessante und ausgiebig studierte Objekte der Algebraischen Geometrie, in deren Untersuchung sich Algebra, Zahlentheorie, Geometrie und komplexe Analysis treffen. Seit etwa 1985 werden ihre Anwendungsmöglichkeiten in der Public-Key-Kryptographie untersucht. Im Moment liefern sie die effizientesten bekannten Public-Key-Verfahren. Sie sind in verschiedenen neuen Verschlüsselungsstandards vorgesehen, so daß vielfältige industrielle Anwendungen zu erwarten sind.