Stichworte: ElGamal-Verschlüsselung und ElGamal-Signatur auf Untergruppe $\langle x \rangle$ einer beliebigen abelschen Gruppe $(G,+)$, Hashfunktion, Motivation: nimm für $(G,+)$ die Gruppe einer elliptischen Kurve

---

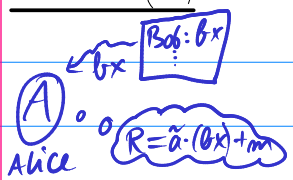
1.2.3 ElGamal-Verschlüsselung (entwickelt von T. ElGamal)

Allen Teilnehmern bekannt sei eine abelsche Gruppe $(G,+)$ und ein Gruppenelement $x \in G$ von (großer) Ordnung $n = \mathrm{ord}(x)$.
Jeder Nutzer wählt eine Zufallszahl $d \in \{1,\dots,n-1\}$ als privaten Schlüssel und erzeugt einen öffentlichen Schlüssel $d \cdot x$:

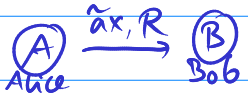
	geheim	öffentlich
Alice	a	ax
Bob	b	bx

1.) Alice möchte eine geheime Botschaft $m \in G$ an Bob schicken.
Das Verfahren geht wie folgt:

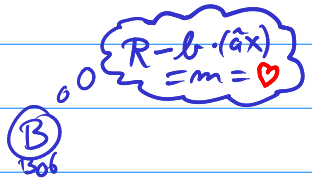
Schritt (1.) Alice wählt eine Zufallszahl $\tilde{a} \in \{1,\dots,n-1\}$ und berechnet $\tilde{a} \cdot x$.
Alice besorgt sich Bobs öffentlichen Schlüssel bx und berechnet $R = \tilde{a} \cdot (bx) + m$.



Schritt (2.) Alice schickt $\tilde{a}x$ und R an Bob.



Schritt (3.) Bob berechnet $b \cdot (\tilde{a}x) = \tilde{a} \cdot (bx)$
und die Nachricht durch $R - b \cdot (\tilde{a}x) = m$.



2.) Ein Unbefugter, der die Daten $G, x, n, bx, \tilde{a}x$ kennt und R abgehört hat, kann m genau dann berechnen, wenn er ein Diffie-Hellman-Problem lösen kann (d.h. das Element $\hat{a}b \cdot x \in G$ berechnen.)

3.) Alice könnte $\tilde{a} = a$ wählen. Für die Sicherheit dieses Verfahrens ist es aber wichtig, dass sie bei jeder ihrer Nachrichten ein neues $\tilde{a}$ wählt:
Sonst könnte ein Unbefugter, der die Übertragungen $\tilde{a}x$, $R_1 = \hat{a}(bx) + m_1$ und $\hat{a}x$, $R_2 = \hat{a}(bx) + m_2$ abhört und schon die Nachricht m_1 kennt, über $R_2 - R_1 + m_1 = (m_2 - m_1) + m_1 = m_2$ auch m_2 berechnen.

§1.3 Digitale Unterschriften

1.3.1 ElGamal- bzw. DSA-Signatur

Geg. wieder eine abelsche Gruppe $(G,+)$, $x \in G$ mit $n = \mathrm{ord}(x)$ groß.
Alice will eine Nachricht m an Bob digital unterschreiben.
Wieder hat sie einen geheimen Schlüssel $a \in \{1,\dots,n-1\}$
und einen öffentlichen Schlüssel $ax \in G$.

4.) Sei $\mathcal{M}$ die Menge aller möglichen Nachrichten (etwa beliebig lange Folgen von 0 und 1), und gg. sei eine Funktion $h: \mathcal{M} \to \{0,1,\dots,n-1\}$, deren Werte $h(m)$ für $m \in \mathcal{M}$ leicht zu berechnen sind und die die folgenden beiden Eigenschaften hat:

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

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

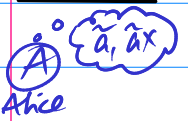
Def.: Eine solche Funktion heißt eine Hashfunktion.

5.) Bsp.: Sei p prim mit $2^{1023} < p \leq 2^{1024} - 1$ und g ein Erzeuger der multiplikativen Gruppe $\mathbb{Z}_p^*$, d.h. $\langle g \rangle = \mathbb{Z}_p^*$. Dann ist nach heutigem Wissen $h: \mathbb{Z}_p^* \to \mathbb{Z}_p^*$, $h(z) = g^z \bmod p$ eine Hashfunktion. Das ab 7.) beschriebene Verfahren kann dann mit $G = \mathbb{Z}_p^*$, $x = g$ durchgeführt werden (in der Praxis nimmt man für p eine Sophie-Germain-PZ, d.h. p prim mit $\frac{p-1}{2}$ auch prim, denn dann ist etwa jedes zweite Element ein Erzeuger und daher leicht ein Erzeuger zu finden).

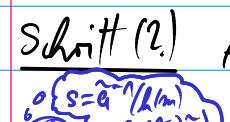
6.) Öffentlich zugänglich seien die Daten $(G,+)$, $x \in G$, $n = \mathrm{ord}(x)$, h und $ax \in G$, sowie eine Bijektion $\chi: \langle x \rangle \to \{0,1,\dots,n-1\}$, deren Werte effektiv berechenbar seien (in der Praxis reicht eine Fkt. deren Urbildmenge $\chi^{-1}(k)$ von jedem $k \in \{0,\dots,n-1\}$ klein ist).

7.) Nun das Verfahren zur Signatur, wie Alice ihre Nachricht $m \in \mathcal{M}$ unterschreiben kann:

Schritt (1.) Alice wählt eine Zufallszahl $\tilde{a} \in \{1,\dots,n-1\}$ mit $ggT(\tilde{a},n)=1$ und berechnet das Gruppenelement $\tilde{a}x \in G$.



Schritt (2.) Alice berechnet das Inverse $\tilde{a}^{-1}$ von $\tilde{a}$ in $\mathbb{Z}_n$ (euklidischer Algo!) sowie $s = \tilde{a}^{-1}(h(m) - \Psi(\tilde{a}x)\cdot a)$ in $\mathbb{Z}_n$.



Schritt (3.) Alice schickt die Nachricht m und ihre Unterschrift $\tilde{a}x, s$ an Bob.

Schritt (4.) (Verifikation) Bob berechnet $\Psi(\tilde{a}x)\cdot ax + s\tilde{a}x$ sowie den Hashwert $h(m)$.
Bob akzeptiert die Unterschrift als echt,
wenn $\Psi(\tilde{a}x)\,ax + s\tilde{a}x = h(m)\cdot x$ in G ist,
was nur stimmt, wenn $\Psi(\tilde{a}x)\,a + s\tilde{a} \equiv h(m) \bmod n$ gewählt ist,
da ja $n = \mathrm{ord}(x)$ in G gilt.

8) Bem.: Kann hier ein Unbefugter die Unterschrift von Alice fälschen?
Dazu müsste er s, kx finden mit $\Psi(kx)\,ax + s\cdot kx = h(m)\,x$
für ein beliebiges k anstelle $\tilde{a}$. Er würde kx berechnen und müsste s passend wählen, wofür ein DL-Problem in $\langle x\rangle \subseteq G$ zu lösen wäre, denn a kennt er nicht.

9) Bem.: Auch hier ist für die Sicherheit des Verfahrens nötig, dass Alice für jede Unterschrift ein neues $\tilde{a}$ wählt: erzeugt Alice zwei Unterschriften $(\tilde{a}x, s_1)$ für m_1 und $(\tilde{a}x, s_2)$ für m_2, ist $s_2 - s_1 \equiv \tilde{a}^{-1}(h(m_2)-h(m_1)) \bmod n$. Wenn $h(m_2)-h(m_1)$ invertierbar in $\mathbb{Z}_n$ ist, kann der Unbefugte $\tilde{a} \bmod n$ berechnen. Wegen $\Psi(\tilde{a}x)\,a \equiv h(m_1) - s_1\tilde{a} \bmod n$ ist dann auch a berechenbar, falls $\Psi(\tilde{a}x)$ invertierbar in $\mathbb{Z}_n$ ist.

10.) Bem.: Wozu eine Hashfunktion h?
- Könnte man leicht Urbilder unter h berechnen, ist das Unterschriftenfälschen einfach: Der Unbefugte wählt $j \in \mathbb{Z}$ beliebig und berechnet $r = jx - ax$, $s = \Psi(r)$ und bestimmt m (nicht von Alice!) mit $h(m) \equiv \Psi(r)\,j \bmod n$. Dann ist r, s eine für Bob verifizierbare Unterschrift der falschen Nachricht m, denn es gilt:

$$\Psi(r)\,ax + \underbrace{\Psi(r)}_{s}\,\underbrace{(jx-ax)}_{r} = \Psi(r)\,jx = h(m)\cdot x.$$

- Wäre h nicht kollisionsresistent und ein Auffinden von $m' \in M$ mit $h(m) = h(m')$ leicht, kann man Alice' Unterschrift unter m' fälschen, wenn man eine gültige Unterschrift $\tilde{a}x, s$ für m hat wegen

$$\gamma(\tilde{a}x)\, a_x + s\tilde{a}x = h(m) \cdot x = h(m') \cdot x.$$

11.) Bem.: Bob muss sicher sein, dass Alice öffentlicher Schlüssel a_x auch wirklich von Alice stammt und nicht von einem Unbefugten gefälscht wurde. Man löst das Problem, indem sich jeder Nutzer bei einer "Certification Authority", kurz CA, registrieren lässt. Bob würde von dieser eine "beglaubigte Kopie" von Alice öffentlichen Schlüssel erhalten; Einzelheiten vgl. Fachliteratur.

12.) Das beschriebene Verfahren heißt ElGamal-Signatur-Verfahren. Eine rechnerisch vorteilhafte Variante heißt DSA (= digital signature algorithm). Das mit der Gruppe einer elliptischen Kurve realisierte DSA-Verfahren heißt ECDSA (= elliptic curve digital signature algorithm), wir besprechen es später genauer.

13.) Motivation: Eine auf Koblitz/Miller zurückgehende Idee ist nun, dass für die ElGamal-Verfahren eine beliebige zyklische Gruppe $\langle x \rangle$ verwendbar ist, wie etwa die, die von Punkten auf elliptischen Kurven erzeugt werden. Da für (geeignete) elliptische Kurven das DL-Problem bzw. DH-Problem schwieriger als für $\mathbb{Z}_m^*$ ist, gilt diese Art von Verschlüsselungstechnik heute als besonders sicher und wird vielfältig industriell angewendet; wegen der kleineren Schlüssellänge ist diese auch rechnerisch praktischer als z.B. RSA. Wir werden die Mathematik elliptischer Kurven im folgenden §2 der Vorlesung näher kennenlernen.