Das ElGamal-Verschlüsselungsverfahren

Clemens Mühlberger

22.06.2002

1 Geschichte

Im Juli 1985 von Taher ElGamal ($*$ 18. August 1955 in Kairo, seit 1984 bei Hewlett-Packard) veröffentlicht. Dieses Verschlüsselungsverfahren hängt eng mit dem Diffie-Hellman-Problem[1] zusammen (siehe 7).

2 zur Erinnerung

Mit $\mathbb{Z}/m\mathbb{Z} = \{0 + m\mathbb{Z}, 1 + m\mathbb{Z}, \ldots, (m-1) + m\mathbb{Z}\}$ bezeichnet man die Menge aller Restklassen $\bmod m$, wobei $|\mathbb{Z}/m\mathbb{Z}| = m$, und mit $(\mathbb{Z}/m\mathbb{Z})^* = \{a + m\mathbb{Z} : \gcd{}^2(a, m) = 1\}$ die prime Restklassengruppe $\bmod m$.

3 Schlüsselerzeugung

Es ist eine Primzahl p und eine Primitivwurzel $g \bmod p$ gegeben (genaueres hierzu unter 8). **Alice** wählt sich nun zufällig und gleichverteilt einen Exponent $a \in \{0, \ldots, p-2\}$ und berechnet $A = g^a \bmod p$. Damit besteht ihr geheimer Schlüsselanteil aus dem Exponenten a. Der öffentliche Schlüssel lautet (p, g, A) und steht ein für allemal fest.

4 Verschlüsselung

Damit ist der Klartextraum $P = \{0, 1, \ldots, p-1\}$ und der zu verschlüsselnde Klartext sei $m \in P$. Nun holt sich **Bob** den öffentlichen Schlüssel (p, g, A) und wählt sich eine Zufallszahl $b \in \{1, \ldots, p-2\}$. Damit berechnet **Bob** $B = g^b \bmod p$. Somit besteht sein geheimer Schlüsselanteil aus der Zufallszahl b. **Bob** bestimmt außerdem $c = mA^b \bmod p$ (zum Vergleich: beim Diffie-Hellman-Verfahren lautet der Schlüsselanteil: $A^b \bmod p = g^{ab} \bmod p$)[3]. **Bob** schickt schließlich an **Alice** den Schlüsseltext (B, c).

5 Entschlüsselung

Alice erhält von **Bob** (B, c) und sie kennt ihren geheimen Schlüsselanteil a. Sie bestimmt zunächst den Exponent $x = p - 1 - a$ mit $1 \leq x \leq p-2$ (da auch $1 \leq a \leq p-2$)[4] und berechnet damit $B^x c \bmod p = m$, da gilt

$$B^x c \overset{x \text{ eing.}}{=} B^{p-1-a} c \overset{B \text{ eing.}}{=} g^{b(p-1-a)} c \overset{c \text{ eing.}}{=} g^{b(p-1-a)} A^b m \equiv$$

[1] Das Diffie-Hellman-Problem ist die Berechnung des Diskreten Logarithmus a: $A = g^a \bmod p$ bzw. $a = \log_g A$

[2] gcd (greatest common divisor) = ggT (größter gemeinsamer Teiler)

[3] beim Diffie-Hellman-Verfahren gilt: $A = g^a$

[4] unter 4 stand zwar $a \in \{0, \ldots, p-2\}$, **aber**:
ist $a = 0$ (oder auch $a = p-1$), dann ist $A = g^0 \bmod p \equiv g^{p-1} mod p \equiv 1$, was nicht sehr sinnvoll wäre

$$\equiv \quad (g^{p-1}g^{-a})^b A^b m \equiv \underbrace{(g^{p-1})^b}_{=1}\underbrace{(g^a)^{-b}}_{=A^{-b}} A^b m \overset{\text{Fermat}}{\equiv} \underbrace{A^b A^{-b}}_{=1} m =$$

$$= \quad m \bmod p.$$

6 Effizienz

Bei der Entschlüsselung benötigt man eine modulare Exponentiation, die der Chinesische Restsatz[5] **nicht** mehr beschleunigen kann. Bei der Verschlüsselung benötigt man sogar zwei modulare Exponentiationen (nämlich $A^b \bmod p$ und $B = g^a \bmod p$), wobei aber **Bob** $A^b \bmod p$ und $B = g^a \bmod p$ vorberechnen kann, da beide unabhängig von m sind.
Wichtig, falls man A^b und B vorausberechnet, müssen diese Werte sicher gespeichert werden, etwa auf einer Chipkarte oder ähnlichem. Mit Vorausberechnung bleibt also insgesamt nur eine modulare Exponentiation und zwar die beim Entschlüsseln.
Der **Nachteil** des ElGamal-Verfahrens ist die *Nachrichtenexpansion*, d. h. der Schlüsseltext (B, c) ist doppelt so lang wie der Klartext m. **Dafür** ist das ElGamal-Verfahren aber ein randomisiertes Verschlüsselungsverfahren (siehe 9) und auch die Länge des öffentlichen Schlüssels kann verkürzt werden indem das gesamte System die selben p und g verwendet (birgt aber auch gleichzeitig ein gewisses **Risiko**, falls für Primzahl p der Diskrete Logarithmus mod$p$ leicht berechenbar ist!).

7 ElGamal und Diffie-Hellman

Falls der Diskrete Logarithmus mod p berechenbar ist, dann kann man auch ElGamal brechen, d. h. aus A erhält man a.
Aber: Die Gegenrichtung ist (noch) unbekannt: Jemand der ElGamal brechen kann, kann auch den Diskreten Logarithmus berechnen?
ElGamal ist genauso schwer wie Diffie-Hellman:

1. **Oskar** kann Diffie-Hellman brechen, d. h. bei gegebenen p, g, A kann er B und $K^{ab} \bmod p$ berechnen: **Oskar** will (B, c) entschlüsseln und kennt (p, g, A). Da er Diffie-Hellman brechen kann, kann er $K = g^{ab} \bmod p$ bestimmen und somit auch $K^{-1}c \bmod p = m$.

2. **Oskar** kann ElGamal brechen, d. h. bei gegebenen p, g, A, B, c kann er m berechnen:
   Um den Schlüssel $K = g^{ab} \bmod p$ für die Diffie-Hellman-Entschlüsselung aus p, g, A, B zu berechnen setzt **Oskar** $c = 1$ (da c nicht im Diffie-Hellman-Verfahren vorhanden ist) und er weiß:
   mit $c = g^{ab}m \bmod p$ gilt: $1 = g^{ab}m \bmod p$ und somit $K \equiv g^{ab} \equiv m^{-1} \bmod p$.

8 Parameterwahl

Um die Berechnung des Diskreten Logarithmus mit heutigen Verfahren[6] zu vermeiden sollte die verwendete Primzahl p wenigstens 512 Bits, besser 768 Bits oder sogar 1024 Bits lang sein.
Achtung: Da die effiziente Berechnung des Diskreten Logarithmus aber eventuell in Zukunft auch für solch langen Primzahlen möglich sein könnte, sollte man sein p am besten zufällig und gleichverteilt wählen!
Bob muß zudem bei jeder Verschlüsselung ein neues b wählen, denn für $b' = b$ und $m' \neq m$ gilt:

$$\left.\begin{array}{l} c = A^b m \bmod p \\ c' = A^b m \bmod p \end{array}\right\} \Rightarrow c'c^{-1} = (A^b m' \bmod p)(A^{-b}m^{-1} \bmod p) = \underbrace{A^b A^{-b}}_{=1} m'm^{-1} \bmod p \equiv m'm^{-1} \bmod p$$

und somit folgt: jeder Angreifer, der m kennt, kennt dann auch m'!
Weiter ist es nützlich, wenn $p - 1$ keine **kleinen** Primfaktoren besitzt.

[5] simultane Kongruenz $x \equiv a_1 \bmod m_1, \ldots, x \equiv a_n \bmod m_n$ hat Lösung x, die eindeutig ist zu mod$m = \prod_{i=1}^{n} m_i$
[6] Enumeration, Shanks Babystep-Giantstep-Algorithmus, Pollard-ρ-Algorithmus

9 ElGamal als randomisiertes Verschlüsselungsverfahren

Der Verschlüsselungsprozeß wird durch die zufällige Wahl von b randomisiert, d. h. wird der selbe Klartext m zweimal verschlüsselt, so wird beim ersten Mal mit dem zufälligen b der Schlüsseltext ($B = g^b \bmod p, c = A^b m \bmod p$), beim zweiten Mal mit einem anderen Exponenten b' der Schlüsseltext ($B' = g^{b'} \bmod p, c' = A^{b'} m \bmod p$) berechnet. Da b und b' zufällig und gleichverteilt $\in \{0, \ldots, p-2\}$ sind, sind auch (B, c) und (B', c') zufällig und gleichverteit $\in \{0, \ldots, p-2\}^2$
Als Voraussetzung muß gelten, dass a Primitivwurzel mod$p$ ist, d. h. $\gcd(a, p-1) = 1$. Dadurch wird die Kryptoanalyse, z. B. durch statistische Tests, erschwert.

10 Verallgemeinerung

Ein wichtiger Vorteil des ElGamal-Verfahrens ist, dass ElGamal nicht nur in primen Restklassengruppen modulo Primzahl, sondern in jeder anderen zyklischen Gruppe anwendbar ist; einzig die Schlüsselerzeugung, Verschlüsselung und Entschlüsselung müssen effizient sein und das Diffie-Hellman-Problem, insbesondere das des Diskreten Logarithmus muß schwer lösbar sein. Dies ist deshalb wichtig, da es sein kann, dass irgendwann das "Diskreter Logarithmus"-Problem in $(\mathbb{Z}/m\mathbb{Z})^*$ lösbar ist, dann wäre aber ElGamal unsicher in $(\mathbb{Z}/m\mathbb{Z})^*$
Folgende Gruppen sind für ElGamal geeignet:

1. Die Jacobische Varietät hyperelliptischer Kurven über einem endlichen Körpern
2. Eine Klassengruppe imaginär-quadratischer Ordnung
3. Die Punktgruppe einer elliptischen Kurve über einem endlichen Körper[7]

11 Allgemeines Verfahren

1. Sei $(G, \circ)$ endliche Gruppe, $g \in G$ und $H =< g >$
2. Wähle $a \in \{1, \ldots, |H| - 1\}$ zufällig, berechne $A = g^a$ (Rechnen in $(G, \circ)$!)
3. Klartextraum $P = G$, Chiffrentextraum $C = G \times G$
4. Öffentlicher Schlüssel $K = ((G, \circ), g, A)$, geheimer Schlüssel a
5. Verschlüsselung mit öffentlichem Schlüssel K:
   - Klartext m in Gruppe G darstellen
   - Wähle $b \in \{1, \ldots, |H| - 1\}$ zufällig
   - Berechne $C_1 = g^b$, $C_2 = m \circ A^b$
   - $E_k(m) =_{def} (C_1, C_2)$
6. Entschlüsselung mit privatem Schlüssel $K = a$: für Chiffrentext $C = (C_1, C_2)$ definiere $D_k(C) =_{def} C_2 \circ (C_1^a)^{-1}$ und es gilt:

$$D_k(E_k(m)) = D_k(g^b, m \circ A^b) = (m \circ A^b) \circ ((g^b)^a)^{-1} = m \circ (g^a)^b \circ ((g^b)^a)^{-1} = m$$

[7] siehe Kapitel 12.2 in J. Buchmann: *Einführung in die Kryptographie*. Springer-Verlag Berlin, 2001

12 Sonstiges

Das ElGamal-Verfahren wird z. B. in PGP 5.0 (Grundlage für Internetstandard RFC2440) mit 768 Bits bis 4096 Bits verwendet. Es besitzt keine Exportbeschränkungen und ist nicht patentiert, aber es könnte unter das Diffie-Hellman-Patent (US Patent #4,200,770; 29.04.1980) fallen – dieses Patent ist aber am 29.4.1997 ausgelaufen.
Nun noch einige Berechnungszeiten (SPARC II, Exponentenlänge 160 Bits, unterschiedliche Modullängen):

	512 bits	**768 bits**	**1024 bits**
Verschlüsselung	0,33 sec	0,80 sec	1,09 sec
Entschlüsselung	0,24 sec	0,58 sec	0,77 sec