Kryptographie und Komplexität

Einheit 5.2

ElGamal Systeme

1. Verschlüsselungsverfahren
2. Korrektheit und Komplexität
3. Sicherheitsaspekte

Das ElGamal Verschlüsselungsverfahren

- **Public-Key Verfahren von Taher ElGamal (1985)**
  - Sicherheit basiert auf Schwierigkeit des DL Problems

    Berechnung diskreter Logarithmen ist nicht in akzeptabler Zeit möglich
  - Eng verwandt mit Diffie-Hellman Schlüsselaustauschprotokoll
- **Verwendet Potenzierung von Gruppenelementen**
  - Nachricht ist Exponent der Potenzierung (anstelle der Basis)
  - Neben Restklassengruppen $(\mathbb{Z}_n, \cdot)$ sind auch andere Gruppen geeignet
- **Protokoll enthält Randomisierung**
  - Empfänger legt Diffie-Hellman Teilschlüssel offen
  - Absender verwendet Zufallszahlen zur Verschlüsselung
  - Nachricht enthält Diffie-Hellman Teilschlüssel des Absenders

Effiziente Ver-/Entschlüsselung und semantische Sicherheit

Das ElGamal Verfahren im Detail

- **Schlüsselerzeugung**
  - Wähle eine große Primzahl p und ein Gruppenelement g der Ordnung p
  - Wähle ein zufälliges $a \in \{0, .., p-2\}$ und berechne $A = g^a$
  - Lege p, g, A offen, halte a geheim
- **Verschlüsselung**
  - Gesamtschlüssel ist $K := (p, g, a, A)$, wobei p, g, A öffentlich
  - Text wird in Blöcke der Länge $\log_2 p/8$ zerlegt (ein Byte pro Buchstabe)
    Jeder Textblock wird als Binärdarstellung einer Zahl x interpretiert
  - Absender wählt zufälliges $b \in \{0, .., p-2\}$ und berechnet $B := g^b$
  - Absender verschlüsselt x zu $y := x \cdot A^b$
  - Erzeugter Schlüsseltext ist $\boldsymbol{e_K(x, b)} = (B, y)$
- **Entschlüsselung**
  - Empfänger berechnet $\boldsymbol{d_K(B, y)} = y \cdot (B^a)^{-1}$

Das ElGamal Verfahren am Beispiel

- **Einfaches Beispiel in Restklassengruppen**
  - Alice wählt $p = 23$, $g = 7 \in \mathbb{Z}_p$, $a = 6$ und berechnet $A = 7^6 \bmod 23 = 4$
  - Bob wählt $b = 3$ und berechnet $B = 7^3 \bmod 23 = 21$ und $A^b \bmod 23 = 18$
  - Verschlüsselung der Nachricht $x = 7$ ergibt Schlüsseltext $(B, y) = (21, 11)$
  - Alice erhält (B, y) und berechnet $(B^a)^{-1} = B^{p-1-a} \bmod 23 = 9$
  - Entschlüsselung von y liefert $11{\cdot}9 \bmod 23 = 7$
- **Berechnungen sind einfacher als bei RSA**
  - Bob kann $B := g^b$ und A^b im Voraus berechnen
  - Alice kann Invertierung von B^a in $\mathbb{Z}_p$ als $B^{p-1-a} \bmod p$ berechnen
- **Gute statistische Streuung der Schlüsseltexte**
  - Verschlüsselung von $4\ 5\ 6\ 7\ 8\ 9$ mit $b = 3$ ergibt y-Werte $3\ 21\ 16\ 11\ 6\ 1$

ElGamal Verfahren mit 32-Bit Zahlen

- **Schlüsselerzeugung**
  - Alice wählt $p = 3013183829$, $g = 2719263871 \in \mathbb{Z}_p$, $a = 1000000000$
  - Alice berechnet $A = g^a \bmod p = 2006813696$ und veröffentlicht p, g, A
- **Verschlüsselung**
  - Bob wählt $b = 50000000000$ und berechnet
    $B = g^b \bmod p = 1948493095$ und $A^b \bmod p = 2537054755$
  - Klartext ist Test, codiert als Zahl $x = 1415934836$
    Schlüsseltext ist $(B, y) = (B, x{\cdot}A^b \bmod p) = (1948493095, 811008022)$
- **Entschlüsselung**
  - Alice berechnet $(B^a)^{-1} = B^{p-1-a} \bmod p = 475183925$
  - Multiplikation mit y liefert $811008022{\cdot}(B^a)^{-1} \bmod p = 1415934836$
  - Konversion in 4-Buchstaben-Block liefert ursprünglichen Klartext
- **Realistische Blocklänge ist 512 oder 1024 Bit**
  - Schnelle Primzahltests/Potenzierungsalgorithmen wie bei RSA nötig

Korrektheit und Komplexität von ElGamal

- **Korrektheit: Ver-/Entschlüsselung sind invers**
  - Es ist $d_K(B,y) = y\cdot(B^a)^{-1}$ wobei $e_K(x,b) = (B,y) = (g^b, x\cdot A^b)$
  - Für beliebige b ist $d_K(e_K(x,b)) = x\cdot A^b\cdot(B^a)^{-1} = x\cdot g^{a\cdot b}\cdot(g^{b\cdot a})^{-1} = x$
- **Aufwand für Auswahl des Schlüssels (einmalig)**
  - Erzeugung der Primzahlen p (z.B. mit Miller-Rabin) $\mathcal{O}(\|p\|^3)$
  - Wahl von g, a und Berechnung von $A = g^a$ in $\mathbb{Z}_p$ $\mathcal{O}(\|p\|^3)$
- **Aufwand für Ver- und Entschlüsselung**
  - Kein Aufwand für Umwandlung zwischen Text und Zahlen
  - Potenzierung von $8|w|/\|p\|$ Blöcken in $\mathbb{Z}_p$ $\mathcal{O}(|w|\cdot\|p\|^2)$
  - Schlüsseltext doppelt so lang wie Klartext (Nachrichtenexpansion)
  - Beschleunigung mit chinesischem Restsatz nicht möglich
- **Trotzdem schneller als RSA**
  - Bei festem b ist Vorberechnung von $B := g^b$, A^b und $(B^a)^{-1}$ möglich
    Ver-/Entschlüsselung benötigt nur eine Multiplikation $\mathcal{O}(|w|\cdot\|p\|)$

Aufwand in anderen Gruppen als $\mathbb{Z}_p$ abhängig von Komplexität der Gruppenoperation ·

Sicherheit von ElGamal

- **Gleich schwer wie Diffie-Hellman Problem**
  - Kann ein Angreifer das DH-Problem lösen, so kann er aus (B, y) und $K = (p, g, A)$ auch $B^a = g^{a \cdot b}$ und damit $x = y \cdot (B^a)^{-1}$ bestimmen
  - Kann ein Angreifer das ElGamal System brechen, so kann er aus $A = g^a$, $B = g^b$ und festem $y = 1$ die Zahl $x = 1 \cdot (B^a)^{-1} = (g^{a \cdot b})^{-1}$ bestimmen. Invertierung von x löst das Diffie-Hellman Problem
- **Sicherheit des geheimen Schlüssels**
  - Bestimmung von a benötigt Lösung des DL Problems
    Äquivalenz zum Problem des diskreten Logarithmus nicht bewiesen
  - Berechnung diskreter Logarithmen ist i.w. exponentiell in $\|p\|$ $\mapsto$ §5.3
- **Semantische Sicherheit durch Randomisierung**
  - Zufällige Wahl von b macht Verschlüsselung nichtdeterministisch
  - Gleiche Klartexte werden verschieden verschlüsselt
    Resultierende Schlüsseltexte (B, y) sind zufällig und gleichverteilt

ElGamal ist sicherer oder effizienter als RSA