Gruppenbasierte Kryptographie

Das RSA und Rabin Verfahren basieren auf dem Restklassenring $\mathbb{Z}/n\mathbb{Z}$ und der Einheitengruppe darin.

Wir betrachten nun Kryptosysteme, welche auf beliebigen abelschen Gruppen basieren. Wir nehmen im folgenden an, daß G eine zyklische Gruppe der Ordnung ℓ ist ($\ell = c\ell_0$ mit c klein und ℓ_0 prim).

Sei G erzeugt von g. Für jedes Element $b \in G$ gibt es ein eindeutig bestimmtes $x \in \mathbb{Z}$ mit $0 \leq x \leq \ell - 1$ und $b = g^x$. Wir nennen x den diskreten Logarithmus von b zur Basis g.

Die Berechnung diskreter Logarithmen in geeigneten Gruppen ist (vermutlich) ein schwieriges Problem (ähnlich wie das Faktorisieren von n). Die Abbildung $x \mapsto g^x$ ist dann eine Einwegfunktion.

Beispiel: $(\mathbb{Z}/p\mathbb{Z}, +)$ leicht, $(\mathbb{Z}/p\mathbb{Z})^\times$ (normalerweise) schwer.

Gruppenbasierte Kryptographie

Mit zyklischen Gruppen kann man machen:

- ElGamal Verschlüsselung (ca. 1985)
- ElGamal, Schnorr, etc. Unterschriften (ca. 1985-1991)
- Diffie-Hellman Schlüsselaustausch (ca. 1976).
- vieles weitere ...

ElGamal Verschlüsselung

Schlüsselerzeugung:

- Wähle $x \in \mathbb{Z}$ mit $0 \leq x < \ell - 1$ zufällig.
- Berechne $y = g^x$.
- Der geheime Schlüssel ist x, der öffentliche Schlüssel ist y.

Verschlüsselung von $m \in G$:

- Wähle $r \in \mathbb{Z}$ zufällig.
- Berechne $u = g^r$ und $v = my^r$.
- Der Chiffretext ist (u, v).

Entschlüsselung von $(u, v) \in G \times G$:

- Berechne $m = vu^{-x}$.
- Der Klartext ist m.

ElGamal Verschlüsselung

Die Abbildung $x \mapsto g^x$ ist keine Einwegfunktion mit Falltür. Dies ist im Verfahren auch nicht erforderlich.

Für G kann man allgemeiner eine Untergruppe der multiplikativen Gruppe von endlichen Körpern $\mathbb{F}_q^\times$ verwenden.

Das ElGamal Verfahren ist randomisiert. Chiffretexte zu zufälligen Nachrichten sind wie zufällige Elemente aus $G \times G$.

Man maskiert eine Nachricht m mit einem zufälligen Wert y^r. Durch die Angabe von g^r versetzt man den Empfänger in die Lage, $y^r = (g^r)^x$ auszurechnen und so m wiederzuerhalten.

ElGamal Sicherheit

Das Diffie-Hellman Problem (CDH) ist, für g, g^a, g^b den Wert g^{ab} auszurechnen.

Das Diffie-Hellman Entscheidungsproblem (DDH) ist, für g, g^a, g^b, h zu entscheiden, ob $h = g^{ab}$ oder nicht.

Thm: Das ElGamal Verfahren ist OW-CPA sicher, wenn das Diffie-Hellman Problem schwierig ist.

Thm: Das ElGamal Verfahren ist IND-CPA sicher, wenn das Diffie-Hellman Entscheidungsproblem schwierig ist.

ElGamal Sicherheit

Das Diffie-Hellman Problem ist, zu g, g^a, g^b den Wert g^{ab} auszurechnen.

Thm: Das ElGamal Verfahren ist OW-CPA sicher, wenn das Diffie-Hellman Problem schwierig ist.

Bew: Zu g, g^a, g^b wählen wir zufällig $s, r \in \mathbb{Z}$ modulo ℓ und $z \in G$ und wenden einen Angreifer auf g, den öffentlichen Schlüssel $y = g^{as}$ und den Ciffretext (g^{br}, z) an. Wir erhalten $m = zg^{-bras}$, daraus $z/m = g^{absr}$ und schließlich $g^{ab} = (g^{absr})^{1/(sr)}$. □

Bemerkung: Ist $\gcd\{r, \ell\} = 1$, gibt es $\lambda, \mu \in \mathbb{Z}$ mit $1 = \lambda r + \mu\ell$. Damit ist g^λ die eindeutig bestimmte r-te Wurzel von g.

ElGamal Sicherheit

Das Diffie-Hellman Entscheidungsproblem ist, zu g, g^a, g^b, h zu entscheiden, ob $h = g^{ab}$ oder nicht.

Thm: Das ElGamal Verfahren ist IND-CPA sicher, wenn das Diffie-Hellman Entscheidungsproblem schwierig ist.

Bew: Sei A ein polynomieller Angreifer gegen IND. A liefert also nach Eingabe zweier Klartexte m_1, m_2 und eines Chiffretext c von m_1 oder m_2 in einer Zeit polynomiell in $\log_2(\#G)$ einen Klartext m_i zurück, welcher mit Wahrscheinlichkeit $> 2/3$ der zu c gehörige Klartext ist. Wir nehmen zuerst zusätzlich an, daß A einen Fehler ausgibt, wenn c weder zu m_1 noch zu m_2 gehört.

ElGamal Sicherheit

Bew (ctd.):

Dann gehen wir wie folgt vor: Zu g, g^a, g^b, h wählen wir zwei zufällige $m_1, m_2 \in G$ und wenden A bezüglich des Basiswerts g und des öffentlichen Schlüssels g^a auf m_1, m_2 und den „Chiffretext“ $c = (g^b, m_1 h)$ an. Gilt $h = g^{ab}$, so ist der Chiffretext eine Verschlüsselung von m_1, ansonsten nicht. Die Wahrscheinlichkeit, daß c ein Chiffretext zu m_2 ist, ist vernachlässigbar. Gibt A also m_1 aus, so geben wir „$h = g^{ab}$“ aus. Gibt A einen Fehler aus, so geben wir „$h \neq g^{ab}$“ aus. Dies liefert einen polynomiellen Algorithmus, welcher das DDH mit Wahrscheinlichkeit $> 2/3$ korrekt löst.

Probleme entstehen, wenn das Verhalten von A undefiniert ist, falls c weder zu m_1 noch zu m_2 gehört. Da die Diskussion hier etwas technisch wird, lassen wir sie aus. □

Drei Probleme

Wir haben also drei Probleme:

- das diskrete Logarithmus Problem (DLP).
- das Diffie-Hellman Problem (CDH, computational DH).
- das Diffie-Hellman Entscheidungsproblem (DDH, decision DH).

Wir können das DDH lösen, wenn wir das CDH lösen können.
Wir können das CDH lösen, wenn wir das DLP lösen können.

Weitere Verhältnisse (grob angedeutet):

- Für allgemeine Gruppen (Black-box Gruppen) ist DDH schwer.
- Es gibt keinen Algorithmus, der das CDH in allgemeinen Gruppen lösen kann, auch unter der Annahme, daß das DDH leicht ist.
- Es gibt Gruppen, in denen das DDH leicht, aber das CDH (vermutlich) schwer ist.
- Es gibt Gruppen, für die das CDH äquivalent zum DLP ist.

ElGamal Sicherheit

Jetzt gibt es wieder (ähnlich wie bei RSA) folgende Fragestellungen:

- Ist ElGamal Verschlüssung IND-CCA2 sicher bzw. was muß man dafür tun (unter der Annahme, daß DDH schwer ist)?
- Für welche speziellen Gruppen ist (oder erscheint) DDH sicher?
- Sind spezielle Bits sicher oder unsicher ...

Wir gehen im folgenden zunächst auf Punkt eins und drei ein, dann fangen wir mit Punkt zwei an.

ElGamal Sicherheit

ElGamal wie bisher vorgestellt wird auch als „Plain ElGamal“ oder „Textbook ElGamal“ bezeichnet.

Plain ElGamal ist nicht Plaintext Aware:

- Jedes Element $(z,h) \in G \times G$ ist die Verschlüsselung der Nachricht $m = hz^{-a}$ unter dem öffentlichen Schüssel g^a.

Plain ElGamal ist nicht NM-CPA sicher:

- Ist (z,h) die Verschlüsselung von m, so ist (z,sh) die Verschlüsselung von sm.

Plain ElGamal ist nicht OW-CCA2 sicher:

- Ein Angreifer erhält die Verschlüsselung (z,h) von m.
- Er erfragt die Entschlüsselung von (z,sh) für ein zufälliges $s \in G$ und erhält m'. Er berechnet $m = m'/s$.

Fujisaki-Okamoto Transformation

Ähnlich wie bei RSA und OAEP kann man durch eine zusätzliche Konstruktion ein IND-CCA2 sicheres Kryptosystem bekommen.

Die Konstruktion heißt Fujisaki-Okamoto Transformation (2000).

Seien $\mathcal{E}$ und $\mathcal{D}$ Ver- und Entschlüsselungsfunktionen.

- Die Verschlüsselung von Nachrichten m unter dem öffentlichen Schlüssel y sei $c = \mathcal{E}_y(m,r)$, wobei r den in der Verschlüsselung benutzten zufälligen Wert bezeichnet.
- Die Entschlüsselung sei $m = \mathcal{D}_a(c)$, wobei a den geheimen Schlüssel bezeichnet.

Sei H eine Hashfunktion (mit passendem Bildbereich) im Zufallsorakelmodell.

Fujisaki-Okamoto Transformation

Fujisaki-Okamoto transformierte Verschlüsselung $\bar{\mathcal{E}}_y(m,r)$:

- $c = \bar{\mathcal{E}}_y(m,r) = \mathcal{E}_y((m||r), H(m||r))$.

Fujisaki-Okamoto transformierte Entschlüsselung $\bar{\mathcal{D}}_a(c)$:

- Berechne $(m||r) = \mathcal{D}_a(c)$.
- Wenn $\mathcal{D}_a(c)$ Fehler ergibt, dann Ausgabe Fehler. Wenn $c \neq \mathcal{E}_y((m||r), H(m||r))$, dann Ausgabe Fehler.
- Ansonsten Ausgabe m.

Thm (RO): Wenn $(\mathcal{E}, \mathcal{D})$ IND-CPA sicher ist, dann ist $(\bar{\mathcal{E}}, \bar{\mathcal{D}})$ IND-CCA2 sicher.

ElGamal Sicherheit

Bereits betrachtet:

1. Plain ElGamal nicht NM-CPA oder OW-CCA2 sicher. Auch nicht PA. Daher Fujisaki-Okamoto Transformation verwenden. Macht aus IND-CPA sicherem Verschlüsselungsverfahren ein IND-CCA2 sicheres Verfahren (im Zufallsorakelmodell).
2. Vorsicht, daß wirklich IND-CPA sicher (Jacobi Symbol in $\mathbb{F}_p$ ...).

Nun:

3. Das DLP, CDH und DDH in allgemeinen Gruppen betrachten.
4. Konkrete Gruppen betrachten.

Bit Sicherheit

Thm: Ist $\gcd\{d, \ell\} = 1$ und kann man aus g, g^a den Wert $a \bmod d$ berechnen, so kann man auch ganz a berechnen.

Bew: Der Wert $g^a g^{-(a \bmod d)}$ hat einen durch d teilbaren Exponenten. Die gcd-Bedingung sagt, daß wir eindeutig d-te Wurzeln ziehen können. Wir können daher $g^{a \operatorname{div} d}$ berechnen. Induktiv erhalten wir schließlich $g^{a \operatorname{div} d} = 1$ und haben a zur Basis d mit nicht negativen Koeffizienten dargestellt. □

Für $d = 2$ ergibt sich die Äquivalenz der Sicherheit des niedrigsten Bits von a mit der Sicherheit von ganz a.

Beispiel: Ist $G = \mathbb{F}_p^\times$, so ist $\ell = p - 1$ gerade. Durch Exponieren mit $(p-1)/2$ bilden wir jedes $z \in G$ auf $\{-1, 1\}$ ab (Jacobisymbol). Für den Wert g^a gilt $a \bmod 2 = 0$ genau dann, wenn $a^{(p-1)/2} = 1$. Wir können jedoch keine eindeutigen Quadratwurzeln ziehen ...

Bit Sicherheit

Beispiel (ctd.): Sei G' die Untergruppe von G der Ordnung $(p-1)/2$, erzeugt von g^2. Hierin ist das 0-te Bit wieder sicher, wenn $(p-1)/2$ ungerade ist und das DLP sicher ist. Das 0-te Bit in G' bezüglich g^2 entspricht dem 1-ten Bit in G bezüglich g, welches daher ebenfalls sicher ist.

Bit Sicherheit in spezieller Gruppe G:

Thm: Sei $G = \mathbb{F}_p^\times$, $n = \log_2(p)$ und $\varepsilon > 0$. Können wir aus g, g^a, g^b die $\varepsilon\sqrt{n}$ höchsten Bits von g^{ab} effizient (in Abhängigkeit von n) berechnen, so können wir ganz g^{ab} effizient berechnen. Die Laufzeit ist exponentiell in $1/\varepsilon$.