Gruppenbasierte Kryptographie

Benutzt zyklische Gruppen von (fast) Primzahlordnung:
$G = \langle g \rangle$ und $\#G = \ell = c\ell_0$ mit c klein und ℓ_0 prim.

$\forall b \in G : \exists x \in \mathbb{Z} : b = g^x$. Das Element x heißt diskreter Logarithmus von b zur Basis g, ist modulo ℓ eindeutig bestimmt.
Diskretes Logarithmus Problem (DLP): Gegeben g, b finde x.

Für geeignete Gruppen ist das DLP (vermutlich) schwer, also $x \mapsto g^x$ eine Einwegfunktion.

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 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.

Drei Probleme

Wir haben also drei algorithmische Probleme:

1. Diskretes Logarithmus Problem (DLP).
2. Diffie-Hellman Problem (CDH, computational DH).
3. 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.

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.

DLP, CDH und DDH

Eine Numeration ist eine Injektion $\sigma : (\mathbb{Z}/\ell\mathbb{Z})^+ \to \{0,1\}^n$.

Ein generischer Algorithmus erwartet als Eingabe eine Liste von Gruppenelemente $L = (\sigma(x_1), \ldots, \sigma(x_k))$ und hat Zugang zu einem Orakel, welches die Werte $\sigma(x_i \pm x_j)$ berechnet. Solche neu berechneten Werte werden zu L hinzugefügt.

Sei ℓ prim.

Thm: Das DDH kann durch einen generischen Algorithmus A bei zufälliger Numeration σ und $\leq m$ Orakelanfragen an σ mit von $1/2$ um maximal m^2/ℓ abweichender Wahrscheinlichkeit gelöst werden.

Laufzeit für konstante Erfolgswahrscheinlichkeit daher mindestens exponentiell in $\log(\ell)$.

DLP, CDH und DDH

Weitere Verhältnisse zwischen DLP, CDH und DDH (grob angedeutet):

- 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.

Shanks Baby Step Giant Step

Ist Anwendung von Meet-in-the-Middle Prinzip. Löst DLP deterministisch in Zeit $O(\sqrt{\ell})$ und Speicher $O(\sqrt{\ell})$.

Schreibe x mit $0 \leq x < \ell$ eindeutig als $x = jm + i$ mit $0 \leq i < m$ und $0 \leq j \leq \ell/m$. Schreibe $x \mapsto g^x$ als $(i,j) \mapsto g^{jm+i}$.

Seien g, b gegeben und x mit $b = g^x$ gesucht.
Trenne i und j: Für $x = jm + i$ gilt $b/g^i = g^{jm}$.

Daher linke Seite für alle i ausrechnen und speichern, dann alle j für Kollision durchprobieren.

Zeit-optimiert für $m \approx \sqrt{\ell}$.

Pollard rho

Analog wie beim Kollisionsfinden für Hashfunktionen. Löst DLP probabilistisch in erwarteter Zeit $O(\sqrt{\ell})$ und Speicher $O(1)$.

Idee: Wir brauchen einen Zufallsweg (random walk) in G von der Form $b^{u_i}g^{v_i}$ mit bekannten u_i, v_i, welche nur von $b^{u_{i-1}}g^{v_{i-1}}$ abhängen. Eine Kollision $b^{u_i}g^{v_i} = b^{u_j}g^{v_j}$ liefert dann $b^{u_i-u_j} = g^{v_j-v_i}$ und folglich $x = (u_i - u_j)/(v_j - v_i) \bmod \ell$, wenn $(v_j - v_i) \not\equiv 0 \bmod \ell$.

Definition von u_i, v_i ($u_0 = 0$, $v_0 = 0$):

- durch $u_i = H_1(b^{u_{i-1}}g^{v_{i-1}})$ und $v_i = H_2(b^{u_{i-1}}g^{v_{i-1}})$ für zwei unabhängige Hashfunktionen H_1 und H_2.
- oder wie folgt: Zerlege G in disjunkte Teilmengen S_1, S_2, S_3 und definiere $(u_i, v_i) = \begin{cases} (u_{i-1}, v_{i-1}+1) & \text{für } b^{u_{i-1}}g^{v_{i-1}} \in S_1 \\ (u_{i-1}+1, v_{i-1}) & \text{für } b^{u_{i-1}}g^{v_{i-1}} \in S_2 \\ (2u_{i-1}, 2v_{i-1}) & \text{für } b^{u_{i-1}}g^{v_{i-1}} \in S_3 \end{cases}$.

Pohlig-Hellman

Beruht auf dem Struktursatz für endlich erzeugte, abelsche Gruppen.

Löst DLP für G beliebig zyklisch, ℓ_0 größter Primfaktor von $\#G$, durch Shanks oder Pollard Methoden in Laufzeit $O^{\sim}(\sqrt{\ell_0})$ und Speicher $O(\sqrt{\ell_0})$ oder $O(1)$.

Idee 1 (chinesischer Restsatz):

- Sei $\#G = \prod_{i=0}^{r} \ell_i^{e_i}$, $n_i = 1 \bmod \ell_i^{e_i}$ und $n_i = 0 \bmod \ell_j^{e_j}$ für alle $j \neq i$, $\phi_i : G \to G$ mit $\phi(z) = z^{n_i}$. Wegen $\gcd\{n_0, \ldots, n_r\} = 1$ gilt $\cap_i \ker(\phi_i) = 1$.
- Die Ordnung von $G_i = \phi_i(G)$ ist $\ell_i^{e_i}$.
- DLP in jedem G_i lösen (falls lösbar) und mit chinesischem Restsatz zusammensetzen: Finde x_i mit $\phi_i(b) = \phi_i(g)^{x_i}$. Dann ist $x = \sum_i x_i n_i$ der DL.

Pohlig-Hellman

Idee 2 („Hensel Lifting“):

- Annahme $\#G = \ell^e$, ℓ prim und $g^{\ell^e} = 1$ mit e minimal.
- Dann $b = g^x$ mit $x = x_0 + x_1\ell + \ldots x_{e-1}\ell^{e-1}$ und $0 \leq x_i < \ell$.
- Löse induktiv DLPs $b^{\ell^{e-1-j}} / g^{\ell^{e-1-j}(x_0 + x_1\ell + \ldots x_{j-1}\ell^{j-1})} = g^{\ell^{e-1-j}x_j\ell^j}$ in x_j für $j = 0, \ldots, e-1$. Sind definiert in Gruppe $\langle g^{\ell^{e-1}} \rangle$ der Ordnung ℓ.

Daher betrachten wir nur zyklische Gruppen mit „möglichst“ primer Ordnung.

Die Verwendung nicht zyklischer Gruppen macht ebenfalls keinen Sinn, da wir nur in der von g erzeugten Untergruppe arbeiten.

Index Calculus

Die Methoden von Shanks, Pollard und Pohlig-Hellman funktionieren in jeder Gruppe gleichermaßen (also für Black-Box Gruppen).

Die folgende Methode benötigt mehr Information über die verwendete Gruppe.

Sei $G \subseteq \mathbb{F}_p^{\times}$ der Primordnung ℓ und $g, b \in G$ mit $b = g^x$ und x gesucht.

- Sei $S = \{p_1, \ldots, p_s\}$ eine Menge von Primzahlen.
- Bestimme zufällige Werte $b^{u_i}g^{v_i}$, lifte sie nach $[0, p-1] \cap \mathbb{Z}$ und „faktorisiere“ sie über S. Geht dies, erhalten wir $b^{u_i}g^{v_i} = \prod_{j=1}^{s} p_j^{e_{i,j}}$. Wiederhole dies mindestens $i \geq s+1$ mal.
- Durch Anwendung von $\log_g$ erhalten wir die linearen Relationen $u_i x + v_i = \sum_{j=1}^{s} e_{i,j} \log_g(p_j) \bmod \ell$. Bei genügend vielen Zeilen können wir z_i nicht alle Null mit $\sum_i z_i e_{i,j} = 0$ für alle j ausrechnen.

Index Calculus

Dann gilt $\sum_i z_i(u_i x + v_i) = 0 \bmod \ell$, folglich $x = -(\sum_i z_i v_i)/(\sum_i z_i u_i) \bmod \ell$.

Grobe Laufzeitanalyse für $p \to \infty$ ...

- $S = \{p \mid p \text{ prim und } p \leq y\}$, $\#S \approx y/\log(y)$.
- $\Pr_{p,y} = \Pr(z \text{ mit } 1 \leq z < p \text{ faktorisiert über } S) \approx u^{-u}$ für $u = \log(p)/\log(y)$ und $u \leq \log(p)^{0.9}$ (Glattheitswahrscheinlichkeit).
- Erwarteter Aufwand, $(e_{i,j})_{i,j}$ zu finden: $\approx (y/\log(y))u^u$,
- also $\approx \exp((1+o(1))\log(y) + (\log(p)/\log(y))\log(\log(p)/\log(y)))$.
- Wird minimiert für $\log(y) = (\mu + o(1))(\log(p)\log(\log(p)))^{1/2}$, nimmt Wert $L_p(1/2, \nu) = \exp((\nu + o(1)(\log(p)\log(\log(p))))^{1/2})$ an.
- Matrixschritt noch $L_p(1/2, 2)$, daher insgesamt $L_p(1/2, \lambda)$. (μ, ν, λ geeignete Konstanten).

$\Rightarrow$ Subexponentielle Laufzeit! Pollard nur $L_p(1, 1/2)$.