Gregor Leander
7.07.2008

Fakultät für Mathematik
Ruhr-Universität Bochum

Kryptographie II – Lösungsvorschläge zum Übungsblatt 6

Aufgabe 1 *Diskrete Logarithmen*

1. Ein Algorithmus nutzt zum Berechnen des Diskreten Logarithmus in einer (zyklischen) Gruppe G primer Ordnung nur die Gruppenoperation und keine speziellen Eigenschaften der Gruppe.

   Wie viele Operationen muss er mindestens Ausführen, um den Diskreten Logarithmus eines zufälligen Elements in G mit einer Wahrscheinlichkeit größer als $1/2$ zu lösen? Begründe Deine Antwort.

2. Was bedeutet diese Aussage für die Schwierigkeit des DL Problems über elliptischen Kurven?

3. Ein Algorithmus A berechnet beliebige Diskrete Logarithmen in einer gegebenen, festen Gruppe G mit etwa 2^{160} Elementen in ungefähr 2^{53} Gruppenoperationen. Was bedeutet diese Aussage für die Gruppe G und den Algorithmus A?
   Zeige mindestens zwei Szenarien, in denen diese Situation geschehen kann.

Lösung

1. Da der Algorithmus in einer generischen Gruppe G mit primer Ordnung $\#G = p$ arbeitet, kommen weder der SPH-Algorithmus (setzt kleine Untergruppen voraus) noch der Index-Calculus-Algorithmus (ist nicht generisch) in Frage. Für eine Aufwandsabschätzung bleibt also nur das Orakelmodell von Shoup-Nechaev, dass den Aufwand für $\mathcal{P}_{Erfolg} \geq \frac{1}{2}$ auf
   $$\frac{1}{\sqrt{2}}\sqrt{\#G}$$
   schätzt.

2. Da der SPH Algorithmus für elliptische Kurven (noch) nicht angepasst werden konnte, bleiben für das DL-Problem in elliptischen Kurven zwei entscheidende Algorithmen übrig: Der BSGS-Algorithmus und der SPH-Algorithmus. Mit Hilfe des SPH-Algorithmus reduziert man die Komplexität für eine gegebene Gruppe auf die Komplexität ihrer grössten Untergruppe primer Ordnung. Deren Komplexität ist

wiederrum durch den BSGS-Algorithmus (bzw. unseren Orakelalgorithmus) gegeben.

Wenn man annimmt, dass für ein sicheres Verfahren ein Angriff mindestens 2^{80} Gruppenoperationen benötigen muss und die p_i der grösste Primfaktor der Gruppenordnung ist, dann erhält man:

$$2^{80} \approx \frac{1}{\sqrt{2}}\sqrt{p_i} \;\Rightarrow\; 2\left(2^{80}\right)^2 = 2^{161} \approx p_i$$

Damit reicht für das DL-Problem in elliptischen Kurven eine deutliche kürzere Schlüssellänge als für das selbe Problem in $(\mathbb{Z}_m^*, \cdot)$

3. Der Orakel-Algorithmus benötigt ca. $m \approx \frac{1}{\sqrt{2}}\sqrt{|G|}$ Gruppenoperationen. Mit $|G| = 2^{160}$ ergibt sich: $m \approx \frac{1}{\sqrt{2}}2^{80} \gg 2^{53}$ Da der Algorithmus somit deutlich effizienter als der generische Orakelalgorithmus arbeitet, muss er zusätzliche Eigenschaften dieser Gruppe ausnutzen. Es könnte sich also beispielsweise um den Index Calculus Algorithmus in $(\mathbb{Z}_m^*, \cdot)$ oder um den SPH Algorithmus in einer Gruppe, deren Ordnung in kleine Primfaktoren zerfällt handeln.

□

Aufgabe 2 *Zufalls Wahl*
Wir haben bei den Generischen Gruppen keine Operation für die zufällige Wahl eines Elementes vorgesehen. Wie kann ein Algorithmus mit Zugriff auf das Gruppen Oracle eine solche Operation umsetzen?

Lösung
Sei G eine Gruppe primer Ordnung und g ein Erzeuger. Das Orakel für G bietet keine Zufallswahlen an. Das macht aber nichts, denn man kann einfach eine zufällige Zahl $n \in \{0, \ldots, |G| - 1\}$ wählen und dann mit dem Hilfe des Orakels und dem Square-and-Multiply Algorithmus g^t ausrechnen. Dies ist ein zufälliges Element in G. □

Aufgabe 3 *Nullstellen*
Sei p prim und $n, d \in \mathbb{N}$ mit $d \leq n-1$. Nach einem Lemma von Schwarz hat ein Polynom

$$P \in \mathbb{Z}_p[X_1, \ldots, X_n]$$

vom Grad d höchstens dp^{n-1} Nullstellen $(x_1, \ldots, x_n)$ in $\mathbb{Z}_p^n$.

1. Zeigen Sie, dass diese Grenze scharf ist. Finden Sie hierfür für jedes p, n und d eine Polynom mit genau dp^{n-1} Nullstellen.

2. Zeigen Sie, durch Angabe eines Gegenbeispiels, dass der Satz falsch ist, wenn p nicht prim ist.

Lösung
Hier ist ein Fehler auf dem Aufgabenblatt. Die Einschränkung lautet richtig $d \leq p - 1$.

1. Das Polynom
$$P = \prod_{i=0}^{d-1}(X_1 - i)$$
hat als Polynom in $\mathbb{Z}_p[X_1]$ sicherlich d Nullstellen. Als Polynom in $Z_p[X_1, \ldots, X_n]$ hat es damit dp^{n-1} Nullstellen, da der Wert von P gar nicht von X_2 bis X_n abhängt.

2. Hier gibt es viele Beispiele. Ein ganz einfaches: Sei $n = pq$ mit p, q prim. Betrachte das Polynom $P = pX$ in $\mathbb{Z}_n[X]$. Diese Polynom hat p Nullstellen, nämlich alle vielfachen von q. Es sollte aber laut dem Satz nur eine Nullstelle haben.

□

Aufgabe 4 *Generische Gruppen*

1. Warum kann der Satz über Diskrete Logarithmen für generische Gruppen nicht primer Ordnung nicht stimmen?

2. An welcher Stelle im Beweis über die Schwierigkeit von Diskreten Logarithmen in generischen Gruppen geht entscheiden ein, dass die Gruppenordnung prim ist?

Lösung

1. Der Satz kann nicht stimmen, da der Silver-Pohlig-Hellman Algorithmus sonst nicht funktionieren dürfte. Der SPH Algorithmus nutzt nur die Gruppenoperation, ist also auch generisch in unserem Sinne.

2. Im Beweis wird genutzt, dass Polynome vom Grad eins höchstens eine Nullstelle haben. Dies folgt aus dem Lemma von Schwarz. Wie wir in der vorherigen Aufgabe gesehen haben gilt das Lemma aber nur für Primzahlen.

□
