



Übung 4

Ausgabe: Freitag, 9. November 2007

Besprechung: Montag, 12. November bis Freitag, 16. November

Klausuranmeldung

Bitte melden Sie sich bis zum Ende der 5. Vorlesungswoche im Webreg für die Klausur *Einführung in die Kryptographie* an. Zusätzlich ist eventuell noch eine Anmeldung in Ihrem Prüfungssekretariat notwendig. Sie können, falls es Ihre Studienordnung zulässt, sich auch noch nach der 5. Woche von der Klausur abmelden.

Aufgabe 1 (Ununterscheidbarkeit)

Wir betrachten in dieser Aufgabe die Frage, ob man sich für den Begriff der Ununterscheidbarkeit auf eine kleine Klasse von Unterscheidern beschränken kann.

Wir betrachten im folgenden nur Zufallsvariablen X, Y mit n -Bit Ausgaben $z \in \{0, 1\}^n$ (für Eingabe 1^n). Für $i = i(n) \in \mathbb{N}$ bezeichne D_i einen effizienten Algorithmus, der als Eingabe 1^n und das i -te Bit des Wertes z erhält (bzw. sofern z weniger als i Bits hat, sei dieses Bit konstant 0). Dabei darf i vom Parameter n abhängen. Es bezeichne \mathbf{D} die Menge aller dieser effizienten Algorithmen. Wir definieren dann folgenden Ununterscheidbarkeitsbegriff:

Zwei Zufallsvariablen X und Y sind *bit-ununterscheidbar*, geschrieben $X \stackrel{\text{bit}}{\approx} Y$, wenn für alle $D_i \in \mathbf{D}$ gilt:

$$|\text{Prob}[D_i(1^n, X(1^n)) = 1] - \text{Prob}[D_i(1^n, Y(1^n)) = 1]| \approx 0.$$

Offensichtlich gilt $X \approx Y \Rightarrow X \stackrel{\text{bit}}{\approx} Y$. Zeigen Sie, dass die Umkehrung im Allgemeinen nicht gilt.

Hinweis: Es genügt, zwei Zufallsvariablen X und Y zu finden, die nachweislich bit-ununterscheidbar sind, aber die für Algorithmen, die mehrere Bits der Eingabe lesen, leicht zu unterscheiden sind.

Aufgabe 2 (Pseudozufallsgeneratoren)

Sei G ein PRG mit Ausgabelänge $m = n + 1$. Welche der folgenden Methoden ergibt dann einen PRG mit Ausgabelänge $n + 2$?

$$G_1(x) = G(x)||1, \quad G_2(x) = G(G(x)), \quad G_3(x) = x||\text{lsb}_2(G(x)).$$

Die Definition von G_2 bedeutet, dass G zunächst für Eingabe x eine Ausgabe der Länge $n + 1$ erzeugt, und dann (quasi mit Parameter $n' = n + 1$) eine Ausgabe der Länge $n' + 1 = n + 2$ erzeugt. Sie können für diesen Generator mit Hilfe der Ununterscheidbarkeit von transformierten, ununterscheidbaren Zufallsvariablen und der (eingeschränkten) Transitivität der Ununterscheidbarkeit argumentieren. Mit $\text{lsb}_2(G(x))$ bezeichnen wir die beiden untersten Bits des Wertes $G(x)$.

Aufgabe 3 (Unvorhersagbarkeit)

In der Vorlesung wurde erwähnt, dass ein Pseudozufallsgenerator unvorhersagbar ist, d.h. dass kein effizienter Algorithmus die Ausgabe vorher erraten kann. Wir weisen diese Aussage hier formal nach.

Eine Zufallsvariable X heißt *unvorhersagbar*, wenn für alle effizienten Algorithmen \mathcal{A} gilt:

$$\text{Prob}[\mathcal{A}(1^n) = X(1^n)] \approx 0,$$

wobei die Wahrscheinlichkeit über die Zufallsbits von \mathcal{A} und X gebildet wird.

Überlegen Sie sich zunächst kurz, dass Zufallsvariablen mit 1-Bit-Ausgaben *nicht* unvorhersagbar sind. Zeigen Sie dann, dass jeder Pseudozufallsgenerator G —genauer die Zufallsvariable $G(U_n)$ — unvorhersagbar ist.

Hinweis: Nutzen Sie für den zweiten Teil aus, dass die uniforme Verteilung U_m für $m > n$ unvorhersagbar ist, und zeigen Sie, dass ein erfolgreicher Vorhersager für den PRG somit einen erfolgreichen Unterscheider \mathcal{D} gegen den PRG liefert.