2. Übungsblatt
Kryptographie und Mediensicherheit (SoSe 2016)

Bauhaus-Universität Weimar, Professur für Mediensicherheit
Prof. Dr. Stefan Lucks, Jakob Wenzel
URL: `http://www.uni-weimar.de/de/medien/professuren/mediensicherheit/teaching`

Abgabe: Bis zum 09.05.2016, 13:30 Uhr zu Beginn der Übung oder per E-Mail an `jakob.wenzel(at)uni-weimar.de`.
Lösungen sind bevorzugt in LaTeX zu verfassen. Ein Template finden Sie auf der Übungsseite der Veranstaltung.

Definition 1 (PZBG-Vorteil)**.** *Ein PZBG-Angreifer $\mathcal{A}$ interagiert mit einem Orakel, dass einen geheimen Schlüssel $K \in \{0,1\}^k$ hat und eine faire Münze B wirft. Das Orakel gibt einen n-Bit-Wert $X \in \{0,1\}^n$ an $\mathcal{A}$. Bei $B = 1$ gibt das Orakel $X \in \{0,1\}^n$ als Ergebnis eines PZBGs $F : \{0,1\}^k \to \{0,1\}^n$ aus; bei $B = 0$ hingegen wählt es X als Ergebnis von n Würfen mit einer fairen Münze. Das Ziel von $\mathcal{A}$ ist es, B korrekt zu erraten. Wir bezeichnen mit $\mathcal{A} \Rightarrow 1$ das Ereignis, dass $\mathcal{A}$ rät dass $B = 1$ sei. Der PZBG-Vorteil von $\mathcal{A}$ auf F ist*

$$\boldsymbol{Adv}_F^{PZBG}(\mathcal{A}) = \left|\Pr\left[\mathcal{A} \Rightarrow 1 | B = 1\right] - \Pr\left[\mathcal{A} \Rightarrow 1 | B = 0\right]\right|.$$

Aufgabe 1 – PZBG-Sicherheit (4 Punkte)
Sei $F : \{0,1\}^k \to \{0,1\}^n$ ein sicherer PZBG und $\wedge$ die bitweise logische AND-Verknüpfung. Seien K_1 und K_2 zwei unabhängige geheime Schlüssel und der PZBG $F' : \{0,1\}^k \times \{0,1\}^k \to \{0,1\}^n$ definiert als $F'(K_1, K_2) = F(K_1) \wedge F(K_2)$.

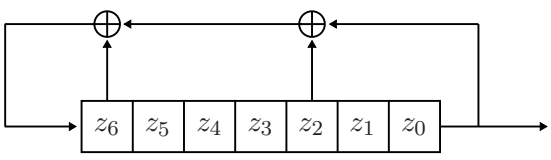
a) Sei $n = 1$ Bit. Geben Sie einen effizienten PZBG-Angreifer $\mathcal{A}$ mit signifikantem Vorteil auf F' an. Was ist der Vorteil von $\mathcal{A}$?

b) Sei $n = 3$ Bit. Geben Sie einen effizienten PZBG-Angreifer $\mathcal{A}$ mit höherem Vorteil auf F' an als in a) und berechnen Sie seinen Vorteil.

Aufgabe 2 – Linear Feedback Shift Register (LSFR) (4 Punkte)
In der Vorlesung werden LSFRs nicht behandelt. Dennoch spielen Sie eine große Rolle für das Verständnis von Stromchiffren. Das Bildungsgesetz eines n-Bit-LSFRs für einen Zustand $z = (z_0, \ldots, z_{n-1})$ und einer Feedback-Belegung $a = (a_0, \ldots, a_{n-1})$ wird allgemein beschrieben durch:

$$z_n = f_a(z) = \bigoplus_{i=0}^{n-1} a_i \cdot z_i.$$

Im folgenden sei ein Beispiel-LFSR zur Veranschaulichung gegeben:

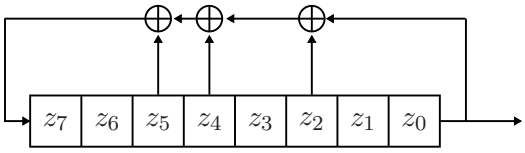


Für einen Zustand $(z_0, \ldots, z_6) = (0, 1, 1, 0, 1, 0, 0)$ und eine Feedback-Belegung von $(a_0, \ldots, a_6) = (1, 0, 1, 0, 0, 0, 1)$ gilt:

$$z_7 = \bigoplus_{i=0}^{6} a_i \cdot z_i = 1 \cdot 0 \oplus 0 \cdot 1 \oplus 1 \cdot 1 \oplus 0 \cdot 0 \oplus 0 \cdot 1 \oplus 0 \cdot 0 \oplus 1 \cdot 0 = 1.$$

Daraus ergibt sich der neue Zustand $(z_1, \ldots, z_7) = (1, 1, 0, 1, 0, 0, 1)$, und die Ausgabe ist 0, da $z_0 = 0$.
(*Lassen Sie sich von der (spiegelverkehrten) graphischen Darstellung nicht verwirren.*)

Für diese Aufgabe betrachten wir den folgenden 8-Bit-LFSR als PZBG, welcher genutzt werden soll, um einen Schlüsselstrom zu berechnen.



Dabei stellt der Anfangszustand des LFSR den geheimen Schlüssel $K = (z_0, \ldots, z_7)$ dar, der genutzt wird, um den Schlüsselstrom zu generieren. Gegeben sind der 8-Bit-Klartext M und der dazugehörige 8-Bit-Chiffretext C.

$$\text{Klartext } M = [1, 0, 0, 1, 0, 1, 1, 0]$$
$$\text{Chiffretext } C = [0, 1, 0, 0, 1, 1, 0, 1]$$

Für den Schlüsselstrom werden die Ausgabe-Bits $z_8, \ldots, z_{15}$ genutzt, d.h. die Verschlüsselung erfolgt durch:

$$C_0 = M_0 \oplus z_8, C_1 = M_1 \oplus z_9, \ldots, C_7 = M_7 \oplus z_{15}.$$

Rekonstruieren Sie den Startzustand $\mathbf{z_0, \ldots, z_7}$.

Aufgabe 3 – PRF-Sicherheit (4 + Bonus(3) Punkte)
Sei $F : \{0,1\}^k \times \{0,1\}^* \to \{0,1\}^n$ eine sichere Pseudozufallsfunktion (PRF). Das heißt, es existiert kein effizienter Algorithmus, der die Ausgaben von F mit signifikantem Vorteil von aus n Würfen einer fairen Münze erhaltene Zufallsbits unterscheiden kann. Geben Sie für **vier** (für den Bonus: **sieben**) der folgenden Funktionen F' jeweils an ob auch sie sichere PRFs sind *oder* beschreiben Sie jeweils kurz einen effizienten Angriff mit max. 2 Anfragen:

a) $F'_K(X) = F_K(X) \oplus c$, wobei c eine öffentlich bekannte Konstante ist.

b) $F'_K(X) = F_K(X)[0..n-2]$ (das letzte Bit wird abgeschnitten).

c) $F'_{K_1,K_2}(X) = F_{K_1}(X) \oplus F_{K_2}(X)$.

d) $F'_K(X) = F_K(X) \oplus \overline{F_K(X)}$, wobei $\overline{X}$ das bitweise Inverse von X bezeichnet.

e) $F'_K(X) = F_K(X) \,\|\, c$ (der Ausgabe wird ein konstanter m-Bit Werte c angehangen).

f) $F'_K(X) = F_K(X) \oplus F_K(\overline{X})$.

g) $F'_K(X) = \text{reverse}(F_K(X))$ (die Reihenfolge der Bits wird umgedreht).

Aufgabe 4 – Perfekte Kryptographie (6 Punkte)
Seien die Ziffern $0, \ldots, 11$ und das Leerzeichen codiert als Elemente aus der Menge $Z_{13} = \{0, \ldots, 12\}$. Für ein Tupel $(a, b) \in \mathcal{K} \subset \mathbb{Z}_{13} \times \mathbb{Z}_{13}$ und einer Nachricht $x \in Z_{13}$ sei die Funktion $f : (\mathbb{Z}_{13} \times \mathbb{Z}_{13}) \times \mathbb{Z}_{13} \to \mathbb{Z}_{13}$ wie folgt definiert:

$$f_{a,b}(x) := a \cdot x - b \pmod{13}.$$

Für welche der angegebenen Verschlüsselungsfunktionen $E_{a,b}$ zusammen mit der jeweiligen Schlüsselmenge $\mathcal{K}$ und der Klartextmenge $\mathcal{P}$ kann man eine Chiffretextmenge $\mathcal{C}$ und eine Entschlüsselungsfunktion $D : \mathcal{K} \times \mathcal{C} \to \mathcal{P}$ angeben, so dass $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ eine Chiffre ist? Welche der Chiffren ist perfekt? Es wird angenommen, dass jeder Schlüssel (Klartext) mit der gleichen Wahrscheinlichkeit auftritt.

a) $E_{a,b}(x) := f_{a,b}(x)$, $\mathcal{K} = \mathbb{Z}_{13} \times \mathbb{Z}_{13}, \mathcal{P} = \mathbb{Z}_{13}$

b) $E_{a,b}(x) := f_{a,b}(x)$, $\mathcal{K} = \mathbb{Z}^*_{13} \times \mathbb{Z}_{13}, \mathcal{P} = \mathbb{Z}_{13}$, wobei $\mathbb{Z}^*_{13} = \{1, \ldots 12\}$ ist.

c) $E_{a,b}(x) := f_{a,b}(x)$, $\mathcal{K} = \mathbb{Z}^*_{13} \times \mathbb{Z}^*_{13}, \mathcal{P} = \mathbb{Z}_{13}$

d) $E_{a,b}((x, x')) := (f_{a,b}(x), f_{a,b}(x'))$, $\mathcal{K} = \mathbb{Z}^*_{13} \times \mathbb{Z}_{13}, \mathcal{P} = \mathbb{Z}_{13} \times \mathbb{Z}_{13}$

Aufgabe 5 – LFSR – Programmieraufgabe (4 Punkte)
Schreiben Sie ein Programm in Python, welches ein beliebiges LFSR realisieren kann. Ihr Programm soll die folgenden Kommandozeilenparameter übergeben bekommen:

- Feedback-Belegung: Bestimmt durch die Koeffizienten a_i
- Startzustand: Bestimmt durch die Werte z_i
- Länge des Schlüsselstroms

Ihr Programm soll anschließend den generierten Schlüsselstrom ausgeben.
(Hinweis: Sie können Ihr Programm mit Aufgabe 2 testen.)

Ihr Program soll fehlerhafte Eingaben (falscher Typ, falsche Anzahl, ...) sinnvoll behandeln und dem Anwender mit einer `usage`-Funktion auf die Nutzung des Programmes hinweisen.

Schicken Sie Ihre Lösung als Anhang einer E-Mail

`lfsr_<MatrNr>.py`

an `jakob.wenzel(at)uni-weimar.de` mit dem Betreff `[Krypto SS16] Beleg 2` bis zum 09. Mai 2016, 13:30 Uhr. Es reicht dabei eine Matrikelnummer aus der Gruppe anzugeben. Die vollständigen Namen und Matrikelnummern sollen als Kommentar im Python-Skript stehen.