Ruhr-Universität Bochum
Lehrstuhl für Kryptologie und IT-Sicherheit
Prof. Dr. Alexander May
Ilya Ozerov

Präsenzübungen zur Vorlesung

Kryptographie I

WS 2012/13

Blatt 6 / 07./09. Januar 2013

AUFGABE 1:

Sei $\Pi' = (\mathsf{Gen}', \mathsf{Mac}', \mathsf{Vrfy}')$ ein sicherer MAC für Nachrichten der festen Länge n. Betrachten Sie den folgenden MAC $\Pi = (\mathsf{Gen}, \mathsf{Mac}, \mathsf{Vrfy})$ für Nachrichten *variabler* Länge:

$\mathsf{Gen}(1^n)$: Gib $k \leftarrow \mathsf{Gen}'(1^n)$ zurück.

$\mathsf{Mac}_k(m)$: Für $k \in \{0,1\}^n$ und $m = m_1, \ldots, m_\ell \in (\{0,1\}^{n/3})^\ell$ wähle $r \in_R \{0,1\}^{n/3}$ und

$$t_i \leftarrow \mathsf{Mac}'_k(r, i, m_i) \text{ für } i = 1, \ldots, \ell,$$

wobei i in $\{0,1\}^{n/3}$-Strings kodiert wird. Gib $t := (r, t_1, \ldots, t_\ell)$ zurück.

Im Vergleich zur Konstruktion Π_{MAC2} aus der Vorlesung (Folie 110) ist die (variable) Länge ℓ nicht Teil der Eingabe.

(a) Geben Sie eine korrekte Vrfy-Funktion an.

(b) Zeigen Sie, dass Π nicht sicher ist.

AUFGABE 2:

Betrachten Sie folgende Modifikationen des CBC-MAC aus der Vorlesung (siehe Folie 117).

(a) Beweisen Sie, dass der CBC-MAC aus der Vorlesung nicht sicher ist, wenn wir Nachrichten mit *variabler* Länge zulassen.

(b) Anstelle eines fixen $t_0 := 0^n$ wählen wir nun zufälliges $t_0 \in_R \{0,1\}^n$ und geben dieses am Ende zusätzlich mit aus, d.h. der Tag hat die Form $t := (t_0, t)$. Zeigen Sie, dass diese Konstruktion unsicher ist.

AUFGABE 3:

Sei (Gen, H) eine kollisionsresistente Hashfunktion. Betrachten Sie nun die Hashfunktion $(\mathsf{Gen}, \hat{H})$ mit $\hat{H}_s(x) := (s, H_s(x))$.

Ist die neue Hashfunktion kollisionsresistent? Falls ja, geben Sie eine Reduktion an, die aus einer Kollision in $\hat{H}$ eine Kollision in H bestimmt. Falls nein, geben Sie einen Angreifer an, der eine Kollision in $\hat{H}$ berechnet.

AUFGABE 4:

Überlegen Sie sich für jede der folgenden Modifikationen der Merkle-Damgard-Transformation, ob sie kollisionsresistent sind. Falls ja, geben Sie einen Beweis an, falls nein, einen Angriff.

Modifizieren Sie die Konstruktion so, dass ...

(a) ... die Eingabelänge nicht gehasht wird, d.h. statt $h_s(z_B, L)$ wird z_B zurückgegeben.

(b) ... statt $h_s(z_B, L)$ der String (z_B, L) zurückgegeben wird.

(c) ... statt ein z_0 zu verwenden die Berechnung mit $z_1 := x_1$ beginnt und dann wie gewohnt fortgesetzt wird.