Ruhr-Universität Bochum
Lehrstuhl für Kryptologie und IT-Sicherheit
Dr. Christopher Wolf
Alexander Meurer, Florian Giesen

RUHR UNIVERSITÄT BOCHUM RUB

Hausübungen zur Vorlesung

Kryptographie 2

SS 2010

Blatt 5 / 16. Juni 2010 / Abgabe 28. Juni, 13:00 Uhr, Kasten NA 02

In der Präsenzübung haben wir den Begriff der *Offline/Online Signaturen* motiviert und eine mögliche Konstruktion diskutiert. Nun wollen wir eine alternative Konstruktion untersuchen.

AUFGABE 1. Offline/Online Signaturen. (5 Punkte)
Sei $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Vrfy})$ ein sicherers[1] Signaturschema und sei $(\mathsf{Gen}', \mathsf{Mac}', \mathsf{Vrfy}')$ ein sicherer *Message Authentication Code* (siehe Präsenzübung oder Vorlesung Krypto I, Folien 83 und 84 für Definition und Sicherheitsspiel). Betrachten Sie folgendes Offline/Online Signaturverfahren $(\mathsf{Gen}^{\mathsf{off/on}}, \mathsf{Sign}^{\mathsf{off/on}}, \mathsf{Vrfy}^{\mathsf{off/on}})$.

- $\mathsf{Gen}^{\mathsf{off/on}}(1^n)$ gibt $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^n)$ aus.
- $\mathsf{Sign}^{\mathsf{off/on}}(m)$
  - Offline-Phase: Wähle $s \leftarrow \mathsf{Gen}'(1^n)$ und berechne $\sigma_1 \leftarrow \mathsf{Sign}_{\mathsf{sk}}(s)$
  - Online-Phase: Berechne $\sigma_2 \leftarrow \mathsf{Mac}_s(m)$

  Gib die Signatur $\sigma = (\sigma_1, \sigma_2, s)$ aus.

a) Geben Sie eine sinnvolle Verifikation $\mathsf{Vrfy}^{\mathsf{off/on}}$ an und begründen Sie die Korrektheit.

b) Ist die Konstruktion vernünftig? Beweisen Sie die Sicherheit des Verfahrens *oder* geben Sie einen effizienten Angreifer an.

[1] Sicher bedeutet hier stets *existentiell unfälschbar unter Chosen Message Angriffen (CMA)*, vgl. Folie 69.

AUFGABE 2. Random Oracle. (5 Punkte)
Sei $(\mathsf{Gen}, \mathsf{Mac}, \mathsf{Vrfy})$ ein Message Authentication Code für n-Bit Nachrichten definiert wie folgt: Sei $\mathsf{Mac}_k(m) := H(k||m)$ mit $|m| = |k| = n$ und

$$\mathsf{Vrfy}_k(m, \mathsf{tag}) = \begin{cases} 1 & \text{falls } \mathsf{tag} = H(k||m) \\ 0 & \text{sonst} \end{cases}$$

wobei $\mathsf{Gen}(1^n)$ einen zufälligen Schlüssel $k \in_R \{0,1\}^n$ ausgibt und $H : \{0,1\}^{2n} \to \{0,1\}^n$ eine Funktion ist. Zeigen Sie, dass dies ein sicherer MAC ist, wenn H als Random Oracle modelliert ist.

Will man die Existenz von Einwegsignaturen unter der Annahme der Existenz von kollisionsresistenten Hashfunktionen beweisen (vgl. Folie 87), so nutzt man aus, dass die Existenz von Hashfunktionen die Existenz von Einwegfunktionen impliziert. Wir wollen dies in der folgenden Aufgabe formal beweisen. Zunächst erinnern wir hierfür an die folgende Definition einer *Familie von Einwegfunktionen*.

Definition: Eine Familie von Funktionen Π_f besteht aus den 3 ppt Algorithmen:

- $I \leftarrow \mathsf{Gen}(1^n)$ wobei I eine Urbildmenge $\mathcal{D}_I$ und einen Wertebereich $\mathcal{R}_I$ definiert.
- $x \leftarrow \mathsf{Samp}(I)$ wobei $x \in_R \mathcal{D}$.
- $y = f(I, x)$ mit $y \in \mathcal{R}_I$ und $x \in \mathcal{D}_I$.

(vgl. Permutationsfamilie) Das Spiel $\mathsf{Invert}_{\mathcal{A},\Pi_f}(n)$ ist analog zu dem im Skript angegebenen Spiel für Permutationsfamilien und wir sagen, dass eine Familie von Funktionen eine *Einwegfunktionsfamilie* ist, falls für alle ppt. Angreifer $\mathcal{A}$ eine vernachlässigbare Funktion negl existiert, so dass

$$Pr[\mathsf{Invert}_{\mathcal{A},\Pi_f}(n) = 1] \leq \mathsf{negl}(n).$$

AUFGABE 3. Konstruktives. (5 Punkte)
Sei (Gen_H, H) eine kollisionsresistente Hashfunktion wobei wir uns der Einfachheit halber auf den Fall $H : \{0,1\}^{2n} \to \{0,1\}^n$ beschränken, d.h. streng genommen betrachten wir lediglich *Kompressionsfunktionen*. Konstruieren Sie daraus eine *Familie von Eingwegfunktionen*, d.h. geben Sie ein Tupel von Algorithmen $(\mathsf{Gen}, \mathsf{Samp}, f)$ wie oben an und weisen Sie die Einwegeigenschaft gemäß dem Invertierspiel auf Folie 61 nach.

Hinweis: Übernehmen Sie (Gen_H, H) für die Konstruktion der Familie von Einwegfunktionen und realisieren Sie Samp auf triviale Art und Weise.

AUFGABE 4. Parametermanipulation. (5 Punkte)
In der Definition des Digital Signature Standards (DSS) auf Folie 98 benutzt man eine nicht näher spezifizierte Funktion $H : \{0,1\}^* \to \mathbb{Z}_q$. Mittels dieser Funktion bildet man anschliessend beliebige Nachrichten $m \in \{0,1\}^*$ in die Gruppe $\mathbb{Z}_q$ ab, was sowohl während des Signierens als auch Verifizierens benötigt wird.
In der Praxis realisiert man H, indem man zunächst eine kollisionsresistente Hashfunktion wählt, die auf 160-Bit abbildet, bspw. SHA1, und anschliessend modulo q reduziert, d.h. $H(m) := \mathsf{SHA1}(m) \bmod q$. Gehen Sie im Folgenden von dieser Wahl für H aus.
Wir wollen nun ein Szenario konstruieren, in dem ein Angreifer die Paramter (p, q, g) derart wählt, so dass er in der Lage ist eine Signatur für eine vorher von ihm gewählte Nachricht zu beliebigen geheimen Schlüsseln zu fälschen, sofern er *eine einzige* gültige Signatur zu einer ebenfalls von ihm gewählten Nachricht erhält. Es ist bspw. denkbar, dass ein Angreifer die Rolle eines Internetanbieters einnimmt, der eine DSS-Implementierung mit bösartig gewählten Parametern (p, q, g) anbietet und beim ersten Verbinden dazu auffordert, eine vordefinierte Nachricht `Dies ist ein Test` zu signieren. Anschliessend ist der Angreifer in der Lage, eine Signatur für eine zweite von ihm vordefinierte Nachricht `Überweise 10.000$ auf mein Konto` für jeden Nutzer zu fälschen.

a) Finde Sie eine allgemeine Bedingung für zwei Nachrichten $m \neq m'$, so dass beide Nachrichten die selbe Signatur erhalten. Ist diese Bedingung alleine bereits problematisch, wenn wir eine kollisionsresistente Hashfunktion verwenden?

b) Kann ein Angreifer diese Bedingung verwenden, indem er den Parameter q gezielt in Abhängigkeit von zwei Nachrichten (m, m') wählt? Überlegen Sie hierzu, mit welcher Wahrscheinlichkeit ein zufälliges Paar (m, m') einen brauchbaren Parameter q liefert, d.h. q ist eine 160-Bit Primzahl. Hierbei dürfen Sie benutzen, dass eine zufällige 160-Bit-Zahl $n \in \mathbb{N}$ mit Wahrscheinlichkeit $\frac{1}{160 \ln 2} \approx 1/111$ eine Primzahl ist.

c) Wie kann man sich gegen solche Angriffe schützen?