zu Aufgabe 7.1 [PRF als MAC]

Vorbemerkungen

Bisher hatten wir für das Konzept der *Pseudo-Random Function* (PRF) nur Abbildungen betrachtet, bei denen die Definitionsmenge mit der Zielmenge übereinstimmt. Diese Einschränkung ist eigentlich nicht nötig, allgemeiner können wir auch Abbildungen $\{0,1\}^* \to \{0,1\}^\ell$ betrachten.

Sei also $\mathcal{K}$ eine Menge von Schlüsseln, und sei für jedes $K \in \mathcal{K}$ eine Abbildung

$$f_K : \{0,1\}^* \to \{0,1\}^\ell$$

definiert. Das PRF-Angriffsspiel für einen Angreifer $\boldsymbol{B}$ sieht aus wie gewohnt: Der Angreifer hat Zugriff auf ein Orakel, das entweder Werte von f_K für ein zufälliges K bietet (dafür schreiben wir $\boldsymbol{B}^{f_K(\cdot)}$) oder Werte einer gleichverteilt zufälligen Abbildung $f : \{0,1\}^* \to \{0,1\}^\ell$ (wir schreiben $\boldsymbol{B}^{f(\cdot)}$). Letzteres heißt, die $f(x)$ für alle $x \in \{0,1\}^*$ sind unabhängig gleichverteilt in $\{0,1\}^\ell$; für diese Verteilung schreiben wir $f \in_\$ \mathrm{Func}(\{0,1\}^*, \{0,1\}^\ell)$. Der Angreifer gibt schließlich ein Bit aus. Sein Erfolg wird gemessen als PRF-Vorteil

$$\mathrm{Adv}^{\mathsf{PRF}}_{f_K,\boldsymbol{B}} = \Pr_{K \in_\$ \mathcal{K}}\left[\boldsymbol{B}^{f_K(\cdot)} \Rightarrow 1\right] - \Pr_{f \in_\$ \mathrm{Func}(\{0,1\}^*,\{0,1\}^\ell)}\left[\boldsymbol{B}^{f(\cdot)} \Rightarrow 1\right].$$

Von *PRF-Sicherheit* der Familie f_K sprechen wir (wie gewohnt) dann, wenn dieser Vorteil für jeden denkbaren Angreifer verschwindend gering bleibt.

Einen *Message Authentication Code* (MAC) haben wir definiert als $(\mathcal{K}_M, MAC, VF)$ mit

- einem *Schlüsselerzeugungsalgorithmus* $\mathcal{K}$ zur Generierung eines K;
- einem *MAC-Erzeugungsalgorithmus,* der zu jedem $m \in \{0,1\}^*$ eine Ausgabe $MAC_K(m) \in \{0,1\}^\ell$ erzeugt;
- und einem *MAC-Überprüfungsalgorithmus,* der zu $m \in \{0,1\}^*$ und $t \in \{0,1\}^\ell$ eine Ausgabe $VF_K(m,t) \in \{\mathsf{true}, \mathsf{false}\}$ erzeugt.

Ein Message Authentication Code soll *korrekt* sein: Zu von $\mathcal{K}$ erzeugtem K und $m \in \{0,1\}^*$ muss stets gelten $VF_K\big(m, MAC_K(m)\big) = \mathsf{true}$. Außerdem wünschen wir Sicherheit, definiert über ein *MAC-Angriffsspiel* mit Angreifer $\boldsymbol{A}$:

- Der Angreifer hat Orakelzugriff auf $MAC_K(\cdot)$ und auf $VF_K(\cdot,\cdot)$ zu einem zufälligem K;
- dabei sind Anfragen $VF_K(m, \mathit{tag})$ nicht zulässig, falls *tag* vorher die Antwort auf eine Anfrage $MAC_K(m)$ war;
- der Angreifer gewinnt das Spiel, wenn $VF_K(\cdot,\cdot)$ jemals auf eine zulässige Anfrage die Antwort true gibt.

Der MAC-Vorteil $\mathrm{Adv}^{\mathsf{MAC}}_{(\mathcal{K},MAC,VF),\boldsymbol{A}}$ ist die Wahrscheinlichkeit, dass $\boldsymbol{A}$ in diesem Angriffsspiel gewinnt. Der Message Authentication Code ist *sicher,* wenn dieser Vorteil für jeden denkbaren Angreifer verschwindend gering ist.

Aufgabe

Nun kann man versuchen, Pseudo-Random Functions (die durch effiziente Algorithmen realisiert werden können) als MAC einzusetzen:

- $\mathcal{K}$ als Schlüsselerzeugungsalgorithmus wählt ein gleichverteilt zufälliges Element der Menge $\mathcal{K}$;
- $MAC_K(m)$ gibt den Wert $f_K(m)$ aus;
- $VF_K(m, tag)$ berechnet zunächst $t = MAC_K(m)$ und gibt dann true aus, falls $tag = t$; false sonst.

Zeigen Sie dafür:

a. Mit dieser Definition ist $(\mathcal{K}, MAC, VF)$ als MAC korrekt.

b. $(\mathcal{K}, MAC, VF)$ ist als MAC sicher, falls die Familie der f_K PRF-sicher ist.

Hinweise zu Aufgabe 7.1 b

Nehmen Sie einen MAC-Angreifer $\boldsymbol{A}$ auf $(\mathcal{K}, MAC, VF)$ an, der höchstens q Anfragen an das Orakel für $VF_K(\cdot,\cdot)$ stellt. Beschreiben Sie auf Basis von $\boldsymbol{A}$ einen PRF-Angreifer $\boldsymbol{B}$, der $\boldsymbol{A}$ verwendet und mit Hilfe seines eigenen Orakels ($f_K(\cdot)$ oder $f(\cdot)$) das MAC-Angriffsspiel für $\boldsymbol{A}$ nachbildet und der schließlich den Wert 1 ausgibt, falls $\boldsymbol{A}$ im nachgebildeten Angriffsspiel gewinnt; 0 sonst:

- Was muss $\boldsymbol{B}$ tun, wenn $\boldsymbol{A}$ eine Anfrage m an sein erwartetes Orakel für $MAC_K(\cdot)$ stellt?
- Was muss $\boldsymbol{B}$ tun, wenn $\boldsymbol{A}$ eine Anfrage (m, tag) an sein erwartetes Orakel für $VF_K(\cdot,\cdot)$ stellt?

Folgern Sie nun

$$\mathrm{Adv}^{\mathsf{MAC}}_{(\mathcal{K},MAC,VF),\boldsymbol{A}} \;\leq\; \mathrm{Adv}^{\mathsf{PRF}}_{f_K,\boldsymbol{B}} + \frac{q}{2^\ell},$$

indem Sie einerseits

$$\Pr_{K\in_\$\mathcal{K}}\left(\boldsymbol{B}^{f_K(\cdot)} \Rightarrow 1\right)$$

betrachten und andererseits

$$\Pr_{f\in_\$\mathrm{Func}(\{0,1\}^*,\{0,1\}^\ell)}\left(\boldsymbol{B}^{f(\cdot)} \Rightarrow 1\right).$$

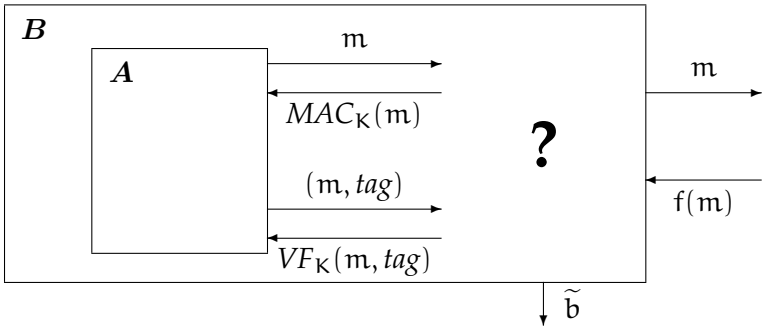
Die Ungleichung bietet die gewünschte Sicherheitsaussage.

a. Korrektheit des MAC heißt: Für $K \xleftarrow{\$} \mathcal{K}$ und $m \in \{0,1\}^*$ gilt stets

$$VF_K\left(m, MAC_K(m)\right) = \mathsf{true}.$$

Das ist hier offensichtlich erfüllt, denn $VF_K(m, tag)$ berechnet gerade $MAC_K(m)$ und gibt true aus, falls $MAC_K(m) = tag$.

b. Wir wollen den MAC-Angreifer $\boldsymbol{A}$ in einem PRF-Angreifer $\boldsymbol{B}$ verwenden:



$\boldsymbol{A}$ erwartet zwei Orakel. Das eine soll auf Anfragen m stets $MAC_K(m)$ liefern und darf (im Rahmen der Laufzeit) beliebig oft mit beliebigen Anfragen verwendet werden. Das andere soll zu zulässigen Anfragen (m, tag) den Wahrheitswert $VF_K(m, tag)$ liefern und wird höchstens q-mal befragt.

Wir können ohne Einschränkung annehmen, dass $\boldsymbol{A}$ hier nur zulässige Anfragen stellt: $\boldsymbol{A}$ muss also solche Anfragen $VF_K(m, tag)$ vermeiden, bei denen *tag* die Antwort auf eine vorherige Anfrage $MAC_K(m)$ für das gleiche m war. Falls irgendein $\boldsymbol{A}$ das nicht beachtet, können wir es leicht entsprechend umbauen, indem die Fragen und Antworten an das MAC_K-Orakel mitprotokolliert werden. So lassen sich unzulässige Anfragen $VF_K(m, tag)$ erkennen und abfangen: Sie sind stets mit true zu beantworten (denn es gilt tatsächlich $tag = MAC_K(m)$ laut früherer Auskunft des MAC-Orakels), ohne dafür das Verifizierungs-Orakel zu verwenden.

$\boldsymbol{B}$ läuft in einer von zwei Umgebungen: Entweder mit Orakelzugriff auf $f_K(\cdot)$ für ein festes $K \in_\$ \mathcal{K}$ oder mit Orakelzugriff auf $f(\cdot)$ für eine feste zufällige Abbildung $f\colon \{0,1\}^* \to \{0,1\}^\ell$ (was wir als $f \in_\$ \mathrm{Func}(\{0,1\}^*, \{0,1\}^\ell)$ schreiben). Allgemein nennen wir das Orakel $f(\cdot)$.

Als Teil von $\boldsymbol{B}$ soll der gegebene Angreifer $\boldsymbol{A}$ ablaufen. Deshalb müssen wir geeignete Antworten erzeugen, wenn $\boldsymbol{A}$ seine Orakel befragen möchte. Um das von $\boldsymbol{A}$ erwartet Orakel jedenfalls im Fall von $f_K(\cdot)$ zu realisieren, können wir wie folgt vorgehen:

- Bei einer Anfrage m an das MAC-Orakel erfragt $\boldsymbol{B}$ vom eigenen Orakel $f(m)$ und gibt diese Antwort an $\boldsymbol{A}$ weiter.
- Bei einer Anfrage (m, tag) an das Verifizierungs-Orakel erfragt $\boldsymbol{B}$ vom eigenen Orakel $f(m)$ und gibt $\boldsymbol{A}$ die Antwort true, falls $tag = f(m)$ ist; false sonst.

Jetzt lässt sich leicht nachprüfen, dass $\boldsymbol{A}$ tatsächlich exakt im MAC-Angriffsspiel abläuft, falls $\boldsymbol{B}$ es hierbei mit einem Orakel für $f_K(\cdot)$ zu tun hat (mit am Anfang des Spiels festgelegtem $K \xleftarrow{\$} \mathcal{K}$). Lassen wir den Angreifer $\boldsymbol{B}$ am Ende des Spiels also genau dann den Wert 1 ausgeben, wenn $\boldsymbol{A}$ jemals vom Verifizierungsorakel auf eine zulässige Anfrage hin die Antwort true bekommen hat: Dann haben wir

$$\Pr_{K \in_\$ \mathcal{K}}\left[\boldsymbol{B}^{f_K(\cdot)} \Rightarrow 1\right] \;=\; \mathrm{Adv}^{\mathsf{MAC}}_{(\mathcal{K}, MAC, VF), \boldsymbol{A}}$$

erreicht; denn der MAC-Vorteil von $\boldsymbol{A}$ ist nach Definition die Wahrscheinlichkeit, dass $\boldsymbol{A}$ in einem Ablauf des MAC-Angriffsspiels irgendwann auf eine zulässige Anfrage die Antwort true bekommt.

Falls $\boldsymbol{B}$ allerdings statt dessen ein Orakel für $f \in_\$ \mathrm{Func}(\{0,1\}^*, \{0,1\}^\ell)$ hat, ergibt sich für $\boldsymbol{A}$ eine ungewöhnliche Umgebung, die mit dem konkreten MAC gar nichts zu tun hat. Für jedes $m \in \{0,1\}^*$ ist $f(m)$ ein gleichverteilt zufälliges Element von $\{0,1\}^\ell$, und entsprechend bestimmen sich die Antworten, die $\boldsymbol{A}$ von den (vermeintlichen) MAC- und Verifizierungsorakeln bekommt. Wenn $\boldsymbol{A}$ hier mit einer zulässigen Anfrage (m, tag) ein true vom Verifizierungsorakel erhält, dann hat $\boldsymbol{A}$ mit *tag* einen Wert $f(m)$ gefunden, ohne diesen vorher über das MAC-Orakel erfragt zu haben. Wegen der Gleichverteilung der $f(m)$ kann das $\boldsymbol{A}$ bei maximal q Versuchen höchstens mit Wahrscheinlichkeit $q/2^\ell$ gelingen. Also haben wir

$$\Pr_{f \in_\$ \mathrm{Func}(\{0,1\}^*, \{0,1\}^\ell)}\bigl(\boldsymbol{B}^{f(\cdot)} \Rightarrow 1\bigr) \;\le\; \frac{q}{2^\ell},$$

denn $\boldsymbol{B}$ gibt am Ende des Spiels genau dann 1 aus, wenn $\boldsymbol{A}$ im Verlauf des Spiels ein solcher Rateerfolg gelungen ist.

Nun haben wir als Zwischenergebnisse

$$\Pr_{K \in_\$ \mathcal{K}}\bigl(\boldsymbol{B}^{f_K(\cdot)} \Rightarrow 1\bigr) \;=\; \mathrm{Adv}^{\mathsf{MAC}}_{(\mathcal{K}, \mathit{MAC}, \mathit{VF}), \boldsymbol{A}}$$

und

$$\Pr_{f \in_\$ \mathrm{Func}(\{0,1\}^*, \{0,1\}^\ell)}\bigl(\boldsymbol{B}^{f(\cdot)} \Rightarrow 1\bigr) \;\le\; \frac{q}{2^\ell}$$

beisammen für das oben beschriebene $\boldsymbol{B}$. Nach Definition ist

$$\mathrm{Adv}^{\mathsf{PRF}}_{f_K, \boldsymbol{B}} \;=\; \Pr_{K \in_\$ \mathcal{K}}\bigl(\boldsymbol{B}^{f_K(\cdot)} \Rightarrow 1\bigr) - \Pr_{f \in_\$ \mathrm{Func}(\{0,1\}^*, \{0,1\}^\ell)}\bigl(\boldsymbol{B}^{f(\cdot)} \Rightarrow 1\bigr),$$

und so ergibt sich

$$\mathrm{Adv}^{\mathsf{PRF}}_{f_K, \boldsymbol{B}} \;\ge\; \mathrm{Adv}^{\mathsf{MAC}}_{(\mathcal{K}, \mathit{MAC}, \mathit{VF}), \boldsymbol{A}} - \frac{q}{2^\ell}$$

und daraus durch eine leichte Umformung sofort

$$\mathrm{Adv}^{\mathsf{MAC}}_{(\mathcal{K}, \mathit{MAC}, \mathit{VF}), \boldsymbol{A}} \;\le\; \mathrm{Adv}^{\mathsf{PRF}}_{f_K, \boldsymbol{B}} + \frac{q}{2^\ell};$$

q. e. d.!

Weil wir den PRF-Angreifer $\boldsymbol{B}$ aus einem beliebigen MAC-Angreifer $\boldsymbol{A}$ konstruiert haben (und $\boldsymbol{B}$ dabei im wesentlichen die gleiche Laufzeit hat wie $\boldsymbol{A}$), sagt diese Ungleichung uns: Wenn die Familie f_K PRF-sicher ist, so kann sie wie in der Aufgabe beschrieben auch als sicherer MAC verwendet werden – es sei denn, ℓ ist so klein, dass $q/2^\ell$ nicht (für jedes q, das ein denkbarer Angreifer im Rahmen seiner Laufzeit erreichen könnte) verschwindend gering bleibt.

(Man kann sich überlegen, dass die Länge ℓ ganz allgemein für *jeden* sicheren MAC hinreichend groß sein muss: Sonst sind MAC-Fälschungen durch bloßes Raten zu befürchten. Eine sichere PRF kann es dagegen auch für sehr kleines ℓ geben; erst für den Einsatz als MAC wird das ein Problem.)