Clase 20: firmas "full-domain hash"

Fernando Virdia, versión: 0.0.1, junio 2024

Vamos ahora a definir un DSS que utiliza TDPs y funciones de hash para firmar $m \in \{0,1\}^*$.

Definición 39 (TDP-FDH, o "hash-then-invert") *Sea* $\Pi = (\mathsf{Gen}, \mathsf{Eval}, \mathsf{Invert})$ *una TDP, y sea* $\{H_f\}_f$ *una familia de funciones de hash* $H_f \coloneqq \{0,1\}^* \to \mathcal{D}_f$*, donde* $f \colon \mathcal{D}_f \to \mathcal{D}_f$*. Definimos el DSS* $\Pi' = (\mathsf{Gen}', \mathsf{Sign}', \mathsf{Ver}')$ *de hash de dominio completo (FDH, "full-domain hash" o "invert-then-sign"),*

$\mathsf{Gen}'()$	$\mathsf{Sign}'(\mathsf{sk} = (t, f), m)$	$\mathsf{Ver}'(\mathsf{pk} = f, \sigma)$
$(t, f) \leftarrow \mathsf{Gen}()$	$h \leftarrow H_f(m)$	$h \leftarrow H_f(m)$
$\mathsf{sk} \leftarrow (t, f)$	$\sigma \leftarrow \mathsf{Invert}(t, f, h)$	$h' \leftarrow \mathsf{Eval}(f, \sigma)$
$\mathsf{pk} \leftarrow f$	**return** σ	**return** $[\![h' = h]\!]$
return $(\mathsf{sk}, \mathsf{pk})$		

Comentario 63 *Mas allá del tipo de primitiva usado, la mayoría de los DSS efectivamente firman hashes de un mensaje, no el mensaje directamente ("hash-then-sign"). "Deshacer" la estructura algebraica de* m *puede ser útil para obtener seguridad. Por ejemplo, esto es necesario si se usa la famosa TDP de Rivest, Shamir y Adleman (RSA).*

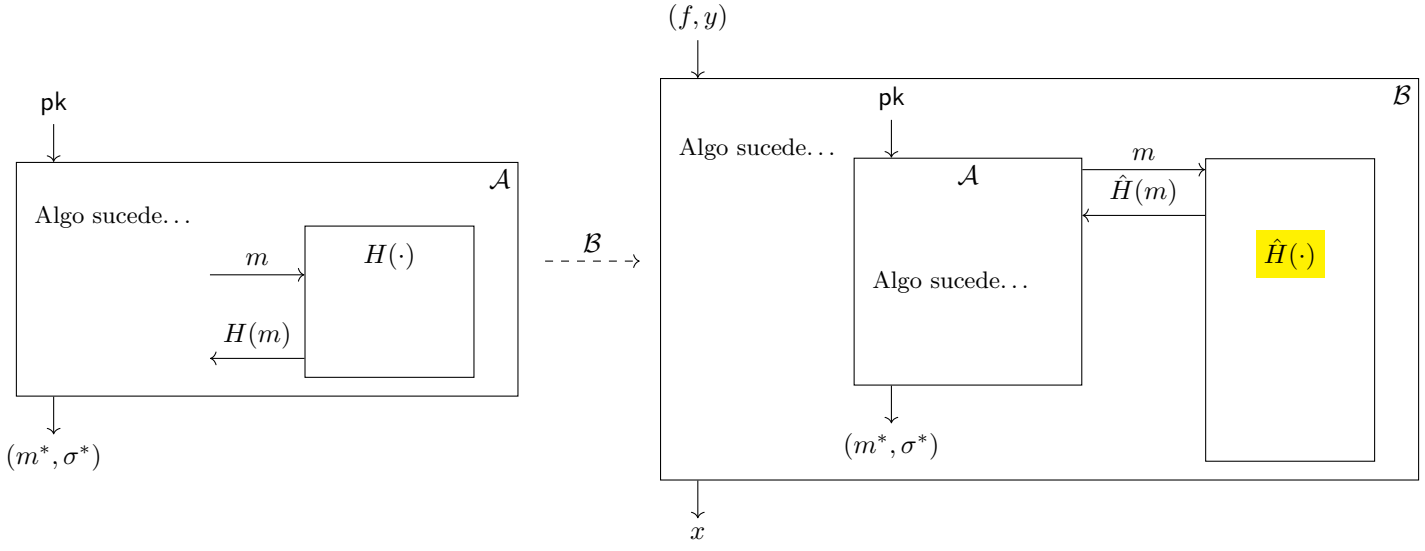
Comentario 64 *Usamos* $m \in \{0,1\}^*$*, pero uno podría definir TDP-FDH sobre cualquier dominio* $\mathcal{R}$ *tal que* $H \colon \mathcal{R} \to \mathcal{D}_f$ *es una función de hash.*

Comentario 65 *Como habrán notado, el esquema es muy parecido a usar una PRF para realizar un MAC, remplazando la PRF (o PRP!) con la TDP.*

Finalmente, demostramos que una TDP-FDH otorga SUF-CMA.

Teorema 8 (OW TDP + ROM ⇒ SUF-CMA) *Sea* Π *una TDP, y sea* Π' *el resultante DSS obtenido como TDP-FDH. Supongamos que la familia* $\{H_f\}_f$ *es compuestas de funciones totalmente aleatorias. Sea* $q_S \geq 1$ *el numero de queries a* $S(\cdot)$ *que le permitimos a un adversario* $\mathcal{A}$ *contra* Π'*. Si* Π *es* (ε, t)*-OW entonces* Π' *es* (ε', t', q')*-SUF-CMA con* $t' \approx t$*,* $q' = q_S$*, y* $\varepsilon \approx e \cdot q_S \cdot \varepsilon$*, donde* $e = 2,71828\ldots$ *es la constante de Napier.*

Demostración. Sea $\mathcal{A}$ un adversario contra TDP-FDH. *Para simplificar la demostración, vamos a suponer que* $\mathcal{A}$ *retorna una posible firma falsificada* σ^* *sobre un mensaje* m^* *solo luego de calcular* $H(m)$*. Si no fuera de ser así, podría falsificar una firma correctamente solamente con probabilidad* $\frac{1}{|\mathcal{D}_f|}$. Sea $p \in (0,1)$ una constante, el cual valor determinaremos mas tarde. Vamos a definir $\mathcal{B}$, un adversario que juega $\mathsf{Exp}^{\mathrm{OW}}$ contra la TDP. Dado que suponemos la función H ser completamente aleatoria, vamos a suponer que cuando $\mathcal{A}$ requiere $H(m)$, calcula este valor a través de un oráculo, que nosotros vamos a remplazar.



Esto puede parecer raro, pero puede ser pensado de la siguiente manera. Supongamos que $\mathcal{A}$ es un programa en C, y precisa calcular un hash $H(m)$. Posiblemente, $\mathcal{A}$ obtenga H a través de una librería de enlace dinámico (una DLL) contra la cual viene compilado, dado que es improbable que la estructura de la función de hash tenga un impacto sobre la dificultad de falsificar una firma (es común suponer que los hashes son "preimage resistant").

Obtenido $\mathcal{A}$, el adversario $\mathcal{B}$ decide de re-compilarla contra una diferente DLL, que implementa otra función aleatoria $\hat{H}(\cdot)$. La intuición es que esto no tendría que reducir la probabilidad de $\mathcal{A}$ de retornar una falsificación (m^*, σ^*) respecto al DSS TDP-FDH con hash $\hat{H}$, dado que $\mathcal{A}$ es un algoritmo aleatorio, y H es una función aleatoria, por lo que la distribución de la variable aleatoria $\mathcal{A}^{\hat{H}}(f, y)$ es la misma que la de $\mathcal{A}^{H}(f, y)$.

Decimos que este tipo de demostración se encuentra en el *modelo del oráculo aleatorio* (ROM, "random oracle model"). Mucho ha sido debatido sobre el ROM. [KM15] cubre los aspectos mas importantes del debate. A menudo este modelo nos permite demostrar la seguridad de esquemas mas eficientes [BR93], que resisten la "prueba del tiempo" una ves desplegados. Por ejemplo, la demostración de que FO es IND-CCA, también se encuentra en el ROM.

$\mathcal{B}(f, y)$

$p \leftarrow$ // un valor particular $\in (0, 1)$
$Q \leftarrow \{\}$
$L \leftarrow [\,]$
$\mathsf{pk} \leftarrow f$
$(m^*, \sigma^*) \leftarrow \mathcal{A}^{S(\cdot),\, \hat{H}(\cdot)}(\mathsf{pk})$
// $(m^*, \sigma^*) \notin Q$
$x \leftarrow \sigma^*$
return x

$\hat{H}(m)$

if $m \in L$:
  $(\ldots, h) \leftarrow L[m]$
  return h
// si no, $m \notin L$:
$r \xleftarrow{\$} (0, 1)$
if $r < p$: // con pr. p
  $\sigma \xleftarrow{\$} \mathcal{D}_f$
  $h \leftarrow \mathsf{Eval}(f, \sigma)$
  $L[m] \leftarrow (\sigma, h)$
else : // con pr. $1 - p$
  $h \leftarrow y$
  $L[m] \leftarrow (\bot, h)$
return h

$S(m)$

$\hat{H}(m)$
$(\sigma, h) \leftarrow L[m]$
if $\sigma = \bot$:
  `ERROR`
$Q \leftarrow Q \cup \{(\sigma, m)\}$
return σ

Observamos el punto de vista de $\mathcal{A}$ "dentro" de $\mathcal{B}$. Al calcular $\hat{H}(m)$, dos cosas pueden pasar:

- Si $r < p$, generamos $\sigma \sim U(\mathcal{D}_f)$, y retornamos $h := f(\sigma)$. Dado que f es una permutación, $h = f(\sigma)$ también es uniformemente aleatorio, $h \sim U(\mathcal{D}_f)$.

$$\left(\forall y \in \mathcal{D}_f,\ \Pr_{\sigma \sim U(D)}[f(\sigma) = y] = \Pr_{\sigma \sim U(D)}[\sigma = f^{-1}(y)] = \frac{1}{|\mathcal{D}_f|}\right)$$

  Por lo tanto, $\hat{H}(m) \sim U(\mathcal{D}_f)$.

- Si $r \geq p$, retornamos y, que por suposición fue generado $\sim U(\mathcal{D}_f)$ en $\mathsf{Exp}^{\mathrm{OW}}$.

En ambos casos, $\hat{H}(m) \sim U(\mathcal{D}_f)$, como $\mathcal{A}$ se espera. En el ROM, esta técnica se llama "programación del oráculo aleatorio".

Al calcular $S(m)$, primero forzamos el calculo de $\hat{H}(m)$. Luego,

- Si $h = \hat{H}(m)$ fue generado en el ramo "$r < p$" de $\hat{H}$, entonces retornamos $\sigma = f^{-1}(h)$ a $\mathcal{A}$, de manera que $\sigma \sim U(\mathcal{D}_f)$, y $\mathsf{Ver}(\mathsf{pk}, m, \sigma) = [\![f(\sigma) = \hat{H}(m)]\!] = 1$, como $\mathcal{A}$ se espera. Esto es posible porque generamos primero σ y luego $h = f(\sigma)$, por lo que no requerimos invertir f para simular la firma.

- Si h fue generado en el ramo $r \geq p$, entonces $\sigma = \bot$, y no podemos *simular* el algoritmo de firma. Esto le causa a $\mathcal{B}$ de terminar su ejecución retornando un `ERROR` y fallando en el ataque. Denotamos este evento, E, y el e vento que no ocurra un error, $\overline{E}$.

Calculamos:

$$\begin{aligned}\Pr[\mathsf{Exp}^{\mathrm{OW}}(\mathcal{B}) \Rightarrow 1] &\geq \Pr[\mathsf{Exp}^{\mathrm{OW}}(\mathcal{B}) \Rightarrow 1 \wedge \overline{E} \wedge \hat{H}(m^*) = y] \\ &= \Pr[\mathsf{Exp}^{\mathrm{OW}}(\mathcal{B}) \Rightarrow 1 \mid \overline{E} \wedge \hat{H}(m^*) = y] \cdot \Pr[\overline{E} \wedge \hat{H}(m^*) = y].\end{aligned}$$

Notamos dos hechos:

1. Como $\mathcal{B}$ retorna el output de $\mathcal{A}$,

   dado $\overline{E} \wedge \hat{H}(m^*) = y$, "$\mathcal{A}$ falsifica la única firma posible de m^*, dentro de $\mathcal{B}$" $\implies$ "$\mathsf{Exp}^{\mathrm{OW}}(\mathcal{B}) \Rightarrow 1$"

2. "$\overline{E} \wedge \hat{H}(m^*) = y$" $\implies$ "$\mathcal{A}$ no puede distinguir $\mathcal{B}$ de $\mathsf{Exp}^{\mathrm{SUF}}$" (*ie*, la distribución de $\mathcal{A}$ dentro de $\mathsf{Exp}^{\mathrm{SUF}}$ = la distribución de $\mathcal{A}$ dentro de $\mathcal{B}$ dado que "$\overline{E} \wedge \hat{H}(m^*) = y$").

Sigue que

$$\begin{aligned}\Pr[\mathsf{Exp}^{\mathrm{OW}}(\mathcal{B}) \Rightarrow 1 \mid \overline{E} \wedge \hat{H}(m^*) = y] &\geq \Pr[\mathcal{A} \text{ falsifica la firma de } m^* \text{ dentro de } \mathcal{B} \mid \overline{E} \wedge \hat{H}(m^*) = y] && \text{(por hecho 1)}\\ &= \Pr[\mathsf{Exp}^{\mathrm{SUF}}(\mathcal{A}) \Rightarrow 1]. && \text{(por hecho 2)}\end{aligned}$$

Finalmente, observamos que los eventos $\overline{E}$ y $\hat{H}(m^*) = y$ son independientes, dados como $\mathcal{B}$ muestrea r dentro de $\hat{H}$, y dado que por suposición $\mathcal{A}$ nunca pide $S(m^*)$. Por lo tanto,

$$\Pr[\overline{E} \wedge \hat{H}(m^*) = y] = \Pr[\overline{E}] \cdot \Pr[\hat{H}(m^*) = y] \geq p^{q_S} \cdot (1-p).$$

Finalmente, coleccionamos los resultados arriba y

$$\begin{aligned}\Pr[\mathsf{Exp}^{\mathrm{OW}}(\mathcal{B}) \Rightarrow 1] &\geq \Pr[\mathsf{Exp}^{\mathrm{OW}}(\mathcal{B}) \Rightarrow 1 \mid \overline{E} \wedge \hat{H}(m^*) = y] \cdot \Pr[\overline{E} \wedge \hat{H}(m^*) = y]\\ &\geq \Pr[\mathsf{Exp}^{\mathrm{SUF}}(\mathcal{A}) \Rightarrow 1] \cdot p^{q_S} \cdot (1-p)\\ \iff \mathsf{Adv}(\mathsf{Exp}^{\mathrm{SUF}}, \mathcal{A}) &\leq \frac{\mathsf{Adv}(\mathsf{Exp}^{\mathrm{OW}}, \mathcal{B})}{p^{q_S} \cdot (1-p)}.\end{aligned}$$

Como notamos al comienzo de la demostración, podemos elegir cualquier $p \in (0,1)$. Para obtener la mejor cota superior de $\mathsf{Adv}(\mathsf{Exp}^{\mathrm{SUF}}, \mathcal{A})$ (o sea, la cota superior mas baja posible), queremos elegir el valor $p = p_0$ que maximiza el denominador, $g(p) = p^{q_S} \cdot (1-p)$, con $p \in (0,1)$.

1. Calculamos la derivada:

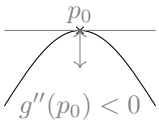
$$\begin{aligned}g'(p) &= \frac{d}{dp}\left[p^{q_S}\right](1-p) + p^{q_S}\frac{d}{dp}\left[1-p\right]\\ &= q_S \cdot p^{q_S-1}(1-p) + p^{q_S} \cdot (-1)\\ &= p^{q_S-1}(q_S - q_S \cdot p - p)\end{aligned}$$

$$g'(p) = 0 \Leftrightarrow q_S - q_S \cdot p - p = q_S - p(q_S+1) = 0 \Leftrightarrow p = \frac{q_S+1-1}{q_S+1} = 1 - \frac{1}{q_S+1} =: p_0.$$

2. Verificamos que es un máximo:

$$\begin{aligned}g''(p) &= \frac{d}{dp}[p^{q_S-1}](q_S - q_S \cdot p - p) + p^{q_S-1}\frac{d}{dp}[q_S - q_S \cdot p - p]\\ &= (q_S-1) \cdot p^{q_S-2}(q_S - q_S \cdot p - p) + p^{q_S-1}(-q_S - 1)\\ &= p^{q_S-2}\left((q_S-1) \cdot (q_S - q_S \cdot p - p) - p \cdot (q_S+1)\right)\\ &= p^{q_S-2}\left(q_S \cdot (q_S - q_S \cdot p - p) - (q_S - q_S \cdot p - p) - p \cdot (q_S+1)\right))\\ &= p^{q_S-2}\left(q_S^2 - q_S^2 \cdot p - q_S \cdot p - q_S + q_S \cdot p + p - p \cdot q_S - p)\right)\\ &= p^{q_S-2}\left(q_S^2 - q_S^2 \cdot p - q_S \cdot p - q_S + \cancel{q_S \cdot p} + \cancel{p} - \cancel{p \cdot q_S} - \cancel{p})\right)\\ &= p^{q_S-2} \cdot q_S \cdot (q_S - q_S \cdot p - p - 1)\\ \implies g''(p_0) &= p_0^{q_S-2} \cdot q_S \cdot (\cancel{q_S - q_S \cdot p - p} - 1) = -q_S \cdot p_0^{q_S-2} < 0.\end{aligned}$$

Dado que $g''(p_0)$ es negativo, $p_0 = 1 - \frac{1}{q_S+1}$ es un máximo de g (el único, dado $g'(p)$).



Ahora calculamos el denominador en p_0,

$$g(p_0) = \left(1 - \frac{1}{q_S+1}\right)^{q_S} \cdot \left(1 - \left(1 - \frac{1}{q_S+1}\right)\right)$$

$$= \frac{\left(1 - \frac{1}{q_S+1}\right)^{q_S+1}}{\left(1 - \frac{1}{q_S+1}\right)} \cdot \frac{1}{q_S+1} = \frac{\left(1 - \frac{1}{q_S+1}\right)^{q_S+1}}{\left(\frac{q_S+1}{q_S+1} - \frac{1}{q_S+1}\right)} \cdot \frac{1}{q_S+1}$$
$$= \left(1 - \frac{1}{q_S+1}\right)^{q_S+1} \cdot \frac{1}{q_S}$$

Dado que $\lim_{n\to\infty} \left(1 - \frac{1}{n}\right)^n = 1/e$ y que $q_S \gg 1$,

$$g(p_0) \approx \frac{1}{e} \cdot \frac{1}{q_S}.$$

Por lo tanto, si dentro de $\mathcal{B}$ asignamos $p \leftarrow p_0 = 1 - \frac{1}{q_S+1}$,

$$\mathsf{Adv}(\mathsf{Exp}^{\mathrm{SUF}}, \mathcal{A}) \leq \frac{\mathsf{Adv}(\mathsf{Exp}^{\mathrm{OW}}, \mathcal{B})}{g(p_0)} \lesssim e \cdot q_S \cdot \mathsf{Adv}(\mathsf{Exp}^{\mathrm{OW}}, \mathcal{B}),$$

y $t' \approx t$, $q' = q_S$ y $\varepsilon' \approx e \cdot q_S \cdot \varepsilon$. □

Comentario 66 *Notamos que las propiedades de H como función de hash no juegan un rol directo en la demostración. Esto es porque ignoramos ser H un hash, y la consideramos una oráculo de función aleatoria. De manera sutil, decidimos* suponer *que los ataques no van a ser causados por un adversario $\mathcal{A}$ que invierta o ataque la estructura interna de H, y como esta posiblemente interactúe con la estructura de la TDP.*

Comentario 67 (Error en la demostración) *Como sutilmente observado por un alumne, esta demostración tiene un error: el caso "$r \geq p$", donde $\hat{H}(m) = y$, hace que con probabilidad $1-p$ la salida de $\hat{H}$ sea y. Del punto de vista de $\mathcal{A}$, esto tendría que ocurrir solamente si el conjunto $\mathcal{D}_f$ de salida de H tuviera tamaño $|\mathcal{D}_f| = \frac{1}{1-p}$, lo que no ocurre en general.*

La demostración se puede arreglar de manera relativamente simple, pero requiere que $(\mathcal{D}_f, \times)$ sea un grupo bajo alguna operación binaria $\times$ donde la operación de inversión sea implementable de manera eficiente, y que la TDP otorgue una propiedad mas: dada $f : \mathcal{D}_f \to \mathcal{D}_f$, esta tiene que ser no solo una permutación sobre $\mathcal{D}_f$, si no también un isomorfismo de grupos, o sea $\forall x, y \in \mathcal{D}_f, f(x) \times f(y) = f(x \times y)$.[7] Como consecuencia, f^{-1} también es un isomorfismo.[8] Al momento de calcular por i-esima vez el ramo $r > p$ dentro de $\hat{H}$, generamos un valor $s_i \sim U(\mathcal{D}_f)$, que guardamos en la tabla hash L como dato accesorio, y retornamos $y \times f(s_i)$ en lugar de simplemente y. Al ser cada s_i uniforme e independiente, y f una permutación, cada $y \times f(s_i)$ es un elemento uniformemente aleatorio e independiente en $\mathcal{D}_f$,[9] y $\mathcal{A}$ no observa ninguna diferencia entre $\hat{H}$ respecto a un oráculo H efectivamente aleatorio. Una vez que $\mathcal{A}$ retorna una firma falsificada $\sigma^* = f^{-1}(\hat{H}(m^*)) = f^{-1}(y \times f(s_i)) = f^{-1}(y) \times f^{-1}(f(s_i)) = f^{-1}(y) \times s_i$, el adversario $\mathcal{B}$ puede recuperar $f^{-1}(y) = \sigma^* \times {s_i}^{-1}$, y retornarlo (en lugar de simplemente retornar $x \leftarrow \sigma^*$).

Aunque el requisito de que f sea un isomorfismo de grupos puede parecer algo exótico, en la practica las dos principales TDPs (de RSA y de Rabin) lo satisfacen. En particular, en el caso de RSA, $\mathcal{D}_f = \mathbb{Z}_N^*$ con $N = p \cdot q$, y $f(x) = x^e \bmod N$ es un isomorfismo dado que es invertible (por como e es elegido), y que $f(x) \times f(y) = x^e \cdot y^e \bmod N = (x \cdot y)^e \bmod N = f(x \cdot y)$.

[7] En este caso, el isomorfismo es de $(\mathcal{D}_f, \times)$ hacia si mismo, por lo que f es un "automorfismo".

[8] Dado $a \times b = f(f^{-1}(a)) \times f(f^{-1}(b)) = f(f^{-1}(a) \times f^{-1}(b))$, aplicamos f^{-1} obteniendo $f^{-1}(a \times b) = f^{-1}(a) \times f^{-1}(b)$.

[9] $\forall h \in \mathcal{D}_f,\ \Pr[y \times f(s_i) = h] = \Pr[f(s_i) = y^{-1} \times h] = \Pr[s_i = f^{-1}(y^{-1} \times h)] = 1/|\mathcal{D}_f|$.