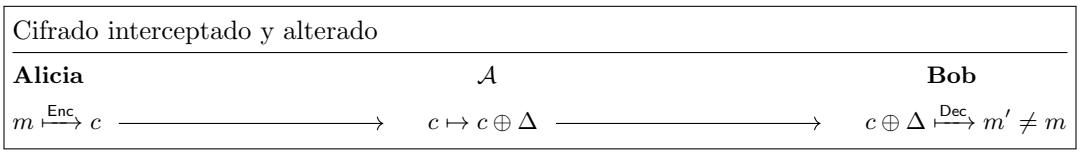
Clase 10: adversarios activos e integridad de datos

Fernando Virdia, versión: 0.0.1, junio 2024

5 Integridad de mensajes

Hasta ahora pudimos garantizar la seguridad si $\mathcal{A}$ se limita a observar cifrados en transito. Ahora consideren el siguiente escenario

Cifrado interceptado y alterado

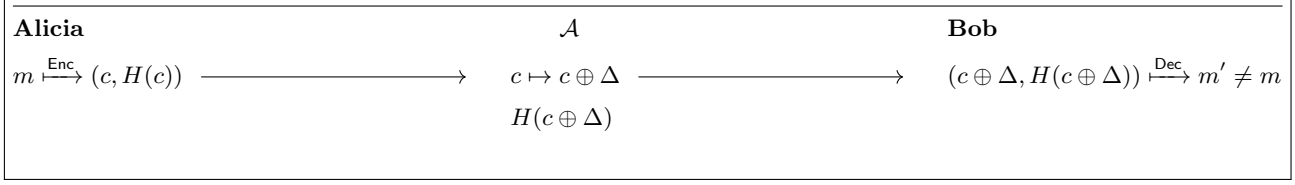


En este caso, un SKES no nos brinda garantías.

PREGUNTA: Alguien conoce algún método para chequear la integridad de datos?

Idea: Usar un checksum/hash.
Estas son funciones $H\colon \{0,1\}^* \to \{0,1\}^\ell$ que "parecen aleatorias" (mas detalles luego).

Intento de autenticado con un hash



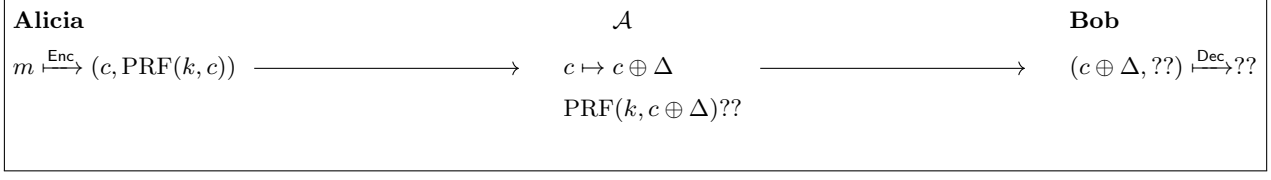
No funciona (a lo sumo, hashing ofrece "integridad sin adversarios", aunque un código de corrección de errores es una mejor solución a ese problema). Ideas?

Si proponen $c||H(m)$:

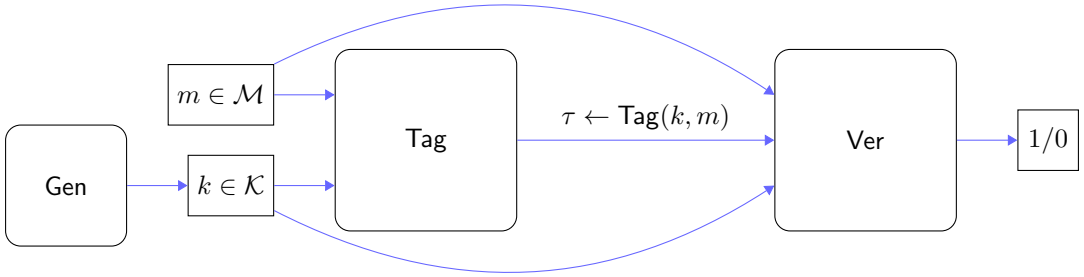
- "Con acceso a $\mathcal{O}^{\mathsf{Enc}}$, $\mathcal{A}$ puede $c' \leftarrow \mathcal{O}^{\mathsf{Enc}}(m', m')$ y transmitir $c'||H(m')$.

Idea: Y si... usamos una PRF?

Intento de autenticado con una PRF



Para saber si funciona, necesitamos una definición de seguridad. Que sintaxis queremos? Sirve para verificar la autenticidad de un "mensaje" (que puede ser un cifrado).



Definición 14 (MAC) *Sean $\mathcal{M}, \mathcal{T}$ conjuntos, $\mathcal{K}$ un conjunto finito. Decimos que $\Pi = (\mathsf{Gen}, \mathsf{Tag}, \mathsf{Ver})$ es un código de autenticación de mensajes (MAC, "message authentication code") con espacio de llave $\mathcal{K}$, de mensajes $\mathcal{M}$, y de etiquetas (tags) $\mathcal{T}$, si*

- Gen *es un algoritmo aleatorio* $\mathsf{Gen}\colon \emptyset \to \mathcal{K}$,
- Tag *es un algoritmo (posiblemente aleatorio)* $\mathsf{Tag}\colon \mathcal{K} \times \mathcal{M} \to \mathcal{T}$,
- Ver *es un algoritmo (posiblemente aleatorio)* $\mathsf{Ver}\colon \mathcal{K} \times \mathcal{M} \times \mathcal{T} \to \{0, 1\}$,

donde

$$\forall k \in \mathcal{K}, m \in \mathcal{M}, \quad \Pr[\mathsf{Ver}(k, m, \mathsf{Tag}(k, m)) = 1] = 1.$$

Comentario 30 *Para proteger comunicaciones, va a ser necesario transmitir un tag τ junto a los datos transmitidos. Por eso, idealmente queremos que τ sea lo mas corto posible. Posiblemente $|\tau|$ seria independiente de $|m|$.*

Comentario 31 *Un sistema con* Tag *determinista tiene la ventaja de que* Ver *es muy simple:*

$$\mathsf{Ver}(k, m, \tau) \coloneqq [\![\mathsf{Tag}(k, m) = \tau]\!].$$

Dada la sintaxis arriba, cualquier $\Pi = (\mathsf{Gen}, \mathsf{Tag}, \mathsf{Ver})$ es un MAC si $\mathsf{Ver}(k, m, \tau) \coloneqq 1$. Necesitamos definir una noción de seguridad para que "$1 \leftarrow \mathsf{Ver}(k, m, \tau)$" tenga un significado.

Definición 15 (MAC security) *Sea $\Pi = (\mathsf{Gen}, \mathsf{Tag}, \mathsf{Ver})$ un MAC con espacios $\mathcal{K}$, $\mathcal{M}$, $\mathcal{T}$.*

$\mathsf{Exp}^{\mathrm{MAC}}(\mathcal{A})$	$T(m)$
1 : $k \leftarrow \mathsf{Gen}()$	1 : $\tau \leftarrow \mathsf{Tag}(k, m)$
2 : $Q \leftarrow \{\}$	2 : $Q \leftarrow Q \cup \{(m, \tau)\}$
3 : $(m^*, \tau^*) \leftarrow \mathcal{A}^T()$	3 : **return** τ
4 : // $(m^*, \tau^*) \notin Q$	
5 : $b \leftarrow [\![\mathsf{Ver}(k, m^*, \tau^*) = 1]\!]$	
6 : **return** b	

Decimos que Π es (ε, t, q)-fuertemente infalsificable en un ataque de mensajes elegidos (SUF-CMA, "strongly unforgeable under chosen-message attacks"), si cualquier adversario $\mathcal{A}$ que corre en tiempo $\leq t$ y usa $\leq q$ queries a $T(\cdot)$, y que retorna $(m^*, \tau^*) \notin Q$, tiene ventaja

$$\mathsf{Adv}(\mathsf{Exp}^{\mathrm{MAC}}, \mathcal{A}) \coloneqq \Pr[\mathsf{Exp}^{\mathrm{MAC}}(\mathcal{A}) \Rightarrow 1] \leq \varepsilon.$$

Comentario 32 *Hay cosas que se pueden cuestionar en esta definición.*

- *$\mathcal{A}$ gana si genera un cualquier nuevo tag, inclusive sobre un mensaje que ha ya sido autenticado. Que problema hay en generar una nueva etiqueta para algo que sabemos ser autentico?*
- *Al $\mathcal{A}$ le permitimos generar tags. Por que también no verificarlos? Podemos notar que para Π con $\mathsf{Ver}(k, m, \tau) \coloneqq [\![\mathsf{Tag}(k, m) = \tau]\!]$, $T(\cdot)$ puede ser usado también para verificar. Usaremos esta definición y tipo de MAC para simplificar la exposición. Mas allá de la estructura de* Ver*, se puede demostrar que otorgarle oráculos de verificación al adversario no lo ayuda de manera significativa [BS23, § 6.2].*

- *La noción le permite a $\mathcal{A}$ de mandar "replays" de mensajes previamente autenticados. Estas no son falsificaciones, y de cualquier manera pueden si ser peligrosas para el usuario. Piensen en el replay de una orden de compra.*

Ahora si podemos demostrar que usar una PRF nos permite autenticar datos.

Lemma 10 (PRF $\Rightarrow$ fixed-length MAC) *Sea $F\colon \mathcal{K}\times\mathcal{D}\to\mathcal{R}$ una (ε,t,q)-PRF. Definimos un MAC $\Pi = (\mathsf{Gen},\mathsf{Tag},\mathsf{Ver})$*

$\mathsf{Gen}()$	$\mathsf{Tag}(k,m)$	$\mathsf{Ver}(k,m,\tau)$
$k \xleftarrow{\$} \mathcal{K}$	$\tau \leftarrow F(k,m)$	$\tau' \leftarrow F(k,m)$
return k	**return** τ	**return** $[\![\tau' = \tau]\!]$

Π es un MAC con espacio de claves $\mathcal{K}$, de mensajes $\mathcal{M}=\mathcal{D}$, y de tags $\mathcal{T}=\mathcal{R}$. Π ofrece (ε',t',q')-seguridad MAC con $\varepsilon' = \varepsilon + \frac{1}{|\mathcal{R}|}$, $t' \approx t$, $q' = q-1$.

Demostración. Construimos un adversario $\mathcal{B}$ que juega $\mathsf{Exp}^{\mathrm{PRF}}$.

$\mathcal{B}^{\mathcal{O}}()$	$T(m)$
$(m^*,\tau^*) \leftarrow \mathcal{A}^{T}()$	$\tau \leftarrow \mathcal{O}(m)$
if $\mathcal{O}(m^*) = \tau^*$:	**return** τ
return 0 ("PRG")	
else :	
return 1 ("random")	

Consideremos $\mathsf{Exp}^{\mathrm{PRF}}(\mathcal{B},1)$, donde $\mathcal{O}$ es una función realmente aleatoria, R.

$$\Pr[\mathsf{Exp}^{\mathrm{PRF}}(\mathcal{B},1)\Rightarrow 0] = \Pr[R(m^*)=\tau^*].$$

Por definición de seguridad MAC, $\mathcal{A}$ retorna un par (m^*,τ^*) no visto antes. Dado que R tiene una sola imagen de m^*, $\mathcal{A}$ tiene que nunca haber evaluado $R(m^*)$ para poderlo retornar. Por ende, tuvo que adivinar el valor $\tau^* \in \mathcal{R}$. Dado que R es una función aleatoria, $\Pr[R(m^*)=\tau^*] = \frac{1}{|\mathcal{R}|}$.

Ahora consideramos $\mathsf{Exp}^{\mathrm{PRF}}(\mathcal{B},0)$. Aquí $\mathcal{O} = F(k,\cdot)$, una PRF. Por construcción, el entorno de $\mathcal{A}$ dentro de $\mathcal{B}$ en $\mathsf{Exp}^{\mathrm{PRF}}(\mathcal{B},0)$ es idéntico al de $\mathcal{A}$ en $\mathsf{Exp}^{\mathrm{MAC}}$, cuando el $\mathsf{Tag}(k,\cdot) = F(k,\cdot)$. Sigue que

$$\Pr[\mathsf{Exp}^{\mathrm{PRF}}(\mathcal{B},0)\Rightarrow 0] = \Pr[F(k,m^*)=\tau^*] = \Pr[\mathsf{Ver}(k,m^*,\tau^*)=1] = \Pr[\mathsf{Exp}^{\mathrm{MAC}}(\mathcal{A})\Rightarrow 1].$$

Usando la desigualdad triangular,

$$\begin{aligned}
\mathcal{A}(\mathsf{Exp}^{\mathrm{MAC}},\mathcal{A}) &= \Pr[\mathsf{Exp}^{\mathrm{MAC}}(\mathcal{A})\Rightarrow 1] \\
&= \Pr[\mathsf{Exp}^{\mathrm{PRF}}(\mathcal{B},0)\Rightarrow 0] \\
&= \Big|\Pr[\mathsf{Exp}^{\mathrm{PRF}}(\mathcal{B},0)\Rightarrow 0] - \Pr[\mathsf{Exp}^{\mathrm{PRF}}(\mathcal{B},1)\Rightarrow 0] + \Pr[\mathsf{Exp}^{\mathrm{PRF}}(\mathcal{B},1)\Rightarrow 0]\Big| \\
&\le \Big|\Big(1 - \Pr[\mathsf{Exp}^{\mathrm{PRF}}(\mathcal{B},0)\Rightarrow 1]\Big) - \Big(1 - \Pr[\mathsf{Exp}^{\mathrm{PRF}}(\mathcal{B},1)\Rightarrow 1]\Big)\Big| + \Big|\Pr[\mathsf{Exp}^{\mathrm{PRF}}(\mathcal{B},1)\Rightarrow 0]\Big| \\
&= \mathsf{Adv}(\mathsf{Exp}^{\mathrm{PRF}},\mathcal{B}) + \Big|\Pr[\mathsf{Exp}^{\mathrm{PRF}}(\mathcal{B},1)\Rightarrow 0]\Big| \\
&= \mathsf{Adv}(\mathsf{Exp}^{\mathrm{PRF}},\mathcal{B}) + \frac{1}{|\mathcal{R}|}
\end{aligned}$$

Dado que $\mathcal{A}$ y $\mathcal{B}$ corren en tiempos parecidos, y que $\mathcal{B}$ hace una query mas a $\mathcal{O}$ que $\mathcal{A}$ a T, Π es (ε',t',q')-MAC con $\varepsilon' = \varepsilon + \frac{1}{|\mathcal{R}|}$, $t' \approx t$, $q'+1 = q$. □

Comentario 33 *Un MAC con espacios de mensajes $\{0,1\}^d$ con $d<\infty$ ("fixed-length"), puede ser limitante en la mayor parte de los casos. Hay varias maneras de extender la amplitud de los mensajes. Vamos a ver una técnica que utiliza funciones de hash.*