Clase 12: confidencialidad e integridad, el cifrado autenticado

Fernando Virdia, versión: 0.0.1, junio 2024

7 Cifrado Autenticado

Resumen: hemos conseguido

- cifrados para mensajes múltiples y arbitrariamente largos
- MACs para cadenas de bits arbitrariamente largas

Que nos falta: integrar ambos mecanismos.

Vamos a protegernos de ambos escenarios limitando la posibilidad de $\mathcal{A}$ de modificar cifrados.

Definición 17 (INT-CTXT) *Sea $\perp$ un símbolo que indica rechazo. Sea $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ un SKES con $\perp \in \mathcal{M}$, tal que Dec puede retornar $\perp$ si recibe un cifrado no valido.*

$\mathsf{Exp}^{\mathrm{CTXT}}(\mathcal{A})$	$E(m)$
$k \xleftarrow{\$} \mathsf{Gen}()$	$c \leftarrow \mathsf{Enc}(k, m)$
$Q_c \leftarrow \{\}$	$Q_c \leftarrow Q_c \cup \{c\}$
$c^* \leftarrow \mathcal{A}^E()$	**return** c
$// \; c^* \notin Q_c$	
$m^* \leftarrow \mathsf{Dec}(k, c^*)$	
$b \leftarrow [\![m^* \neq \perp]\!]$	
return b	

Decimos que Π otorga (ε, t, q)-integridad de cifrado (INT-CTXT, "ciphertext integrity"), si cualquier adversario $\mathcal{A}$ que corre en tiempo $\leq t$ y hace $\leq q$ queries a $\mathcal{O}$, y que retorna $c^* \notin Q_c$, tiene ventaja

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

Comentario 37 *Claramente, PRF-CTR no satisface INT-CTXT, dado que $(c_0, c_1 \oplus \Delta)$ descifra a $m \oplus \Delta$ para cualquier Δ.*

Definición 18 (AE) *Sea Π un SKES. Decimos que Π otorga cifrados $(\varepsilon_{\mathrm{CPA}}, \varepsilon_{\mathrm{CTXT}}, t, q)$-autenticados (AE, "authenticated encryption"), si otorga seguridad $(\varepsilon_{\mathrm{CPA}}, t, q)$-IND-CPA y $(\varepsilon_{\mathrm{CTXT}}, t, q)$-INT-CTXT.*

Comentario 38 *Bajo definición de seguridad AE, $\mathcal{A}$ no puede distinguir entre cifrados de mensajes diferentes por IND-CPA, y no puede tampoco generar nuevos cifrados validos (de la nada, o modificando cifrados que obtuvo a través de un oráculo de cifrado) por INT-CTXT.*

Comentario 39 *Un SKES que otorga AE, también otorga una noción que no vamos a explorar de inmediato, IND-CCA. $\mathsf{Exp}^{\mathrm{CCA}}$ es igual a $\mathsf{Exp}^{\mathrm{CPA}}$, pero $\mathcal{A}$ tiene acceso a oráculos de cifrado $E(m_0, m_1)$ y de descifrado $D(c)$, donde el segundo puede ser utilizado solamente para descifrar cifrados no retornados por $E(\cdot, \cdot)$. En el ámbito de la criptografía simétrica,* AE $\Rightarrow$ IND-CCA *dado que $\mathcal{A}$ no puede generar cifrados validos a menos de no obtenerlos de $E(\cdot, \cdot)$.*

Definición 19 (EtM) *Sean $\Pi = (\mathsf{Gen}^{\Pi}, \mathsf{Enc}, \mathsf{Dec})$ un SKES, y $\Sigma = (\mathsf{Gen}^{\Sigma}, \mathsf{Tag}, \mathsf{Ver})$ un MAC. Definimos el SKES $\Pi' = (\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ "encrypt-then-MAC" (EtM).*

$\mathsf{Gen}'()$	$\mathsf{Enc}'(k, m)$	$\mathsf{Dec}'(k, c)$
$k_e \leftarrow \mathsf{Gen}^{\Pi}()$	$(k_e, k_m) \leftarrow k$	$(k_e, k_m) \leftarrow k$
$k_m \leftarrow \mathsf{Gen}^{\Sigma}()$	$c_e \leftarrow \mathsf{Enc}(k_e, m)$	$(c_e, \tau) \leftarrow c$
$k \leftarrow (k_e, k_m)$	$\tau \leftarrow \mathsf{Tag}(k_m, c_e)$	$b \leftarrow \mathsf{Ver}(k_m, c_e, \tau)$
return k	$c \leftarrow (c_e, \tau)$	$m \leftarrow \mathsf{Dec}(k_e, c_e)$
	return c	**if** $b = 1$:
		return m
		else return $\perp$

Lemma 13 *Sean* Π, Σ, Π' *como en Definition 19. Supongamos que* Π *otorga* $(\varepsilon_{\text{CPA}}, t, q)$*-IND-CPA y* Σ *otorga* $(\varepsilon_{\text{MAC}}, t, q)$*-MAC. Entonces* Π' *otorga* $(\varepsilon'_{\text{CPA}}, \varepsilon'_{\text{CTXT}}, t', q')$*-AE con* $\varepsilon'_{\text{CPA}} = \varepsilon_{\text{CPA}}$, $\varepsilon'_{\text{CTXT}} = \varepsilon_{\text{MAC}}$, $t' \approx t$, $q' = q$.

Demostración. Por definición de AE, necesitamos demostrar que

- Π' otorga $(\varepsilon'_{\text{CPA}}, t', q')$-IND-CPA,
- Π' otorga $(\varepsilon'_{\text{CTXT}}, t', q')$-INT-CTXT.

IND-CPA: Podria parecer "obvio", dado que un cifrado de Π' incluye un cifrado de Π. (Vamos a ver que no siempre esto es inmediato). Dado un adversario $\mathcal{A}$ contra Π' que juega $\mathsf{Exp}^{\text{CPA}}$, construimos $\mathcal{B}$ contra Π que juega $\mathsf{Exp}^{\text{CPA}}$,

$\mathcal{B}^E()$	$\mathcal{O}(m_0, m_1)$
1 : $k_m \leftarrow \mathsf{Gen}^{\Sigma}()$	1 : $c_e \leftarrow E(m_0, m_1)$
2 : $b' \leftarrow \mathcal{A}^{\mathcal{O}}()$	2 : $\tau \leftarrow \mathsf{Tag}(k_m, c_e)$
3 : **return** b'	3 : $c \leftarrow (c_e, \tau)$
	4 : **return** c

Del punto de vista de $\mathcal{A}^{\mathcal{O}}$, el ambiente dentro de $\mathcal{B}$ es idéntico al de $\mathsf{Exp}^{\text{CPA}}(\mathcal{A}, b)$. Dado que $\mathcal{B}$ retorna el output de $\mathcal{A}$,

$$\Pr[\mathsf{Exp}^{\text{CPA}}(\mathcal{B}, b) \Rightarrow 1] = \Pr[\mathsf{Exp}^{\text{CPA}}(\mathcal{A}, b) \Rightarrow 1], \quad \forall b \in \{0, 1\}.$$

De consecuencia, $\mathsf{Adv}(\mathsf{Exp}^{\text{CPA}}, \mathcal{B}) = \mathsf{Adv}(\mathsf{Exp}^{\text{CPA}}, \mathcal{A})$, $t' \approx t$ y $q' = q$, y Π' otorga $(\varepsilon_{\text{CPA}}, \approx t, q)$-IND-CPA.

INT-CTXT: Ahora supongamos que un adversario $\mathcal{A}$ contra $\mathsf{Exp}^{\text{CTXT}}$, y construyamos $\mathcal{B}$ contra $\mathsf{Exp}^{\text{MAC}}$.

$\mathcal{B}^T()$	$\mathcal{O}(m)$
1 : $k_e \leftarrow \mathsf{Gen}^{\Pi}()$	1 : $c_e \leftarrow \mathsf{Enc}(k_e, m)$
2 : $c^* \leftarrow \mathcal{A}^{\mathcal{O}}()$	2 : $\tau \leftarrow T(c_e)$
3 : $(c_e^*, \tau^*) \leftarrow c^*$	3 : $c \leftarrow (c_e, \tau)$
4 : // c_e^* es el mensaje "tagueado"	4 : **return** c
5 : **return** (c_e^*, τ^*)	

Por definición de seguridad INT-CTXT, $\mathcal{A}$ retorna $c^* \notin Q_c$. Dado que Π' produce cifrados $c = (c_e, \mathsf{Tag}(k_m, c_e))$, podemos usar $\mathcal{A}$ para generar un par (m^*, τ^*) donde $m^* \leftarrow c_e^*$. De tal manera, notamos que Q_c en $\mathsf{Exp}^{\text{CTXT}}$ corresponde a Q en $\mathsf{Exp}^{\text{MAC}}$, dado que cada cifrado generado por $\mathcal{O}(\cdot)$ corresponde a un tag generado en $T(\cdot)$. Por lo tanto,

$$c^* = (c_e^*, \tau^*) \notin Q_C \text{ en } \mathsf{Exp}^{\text{CTXT}} \Rightarrow (m^* = c_e^*, \tau^*) \notin Q \text{ en } \mathsf{Exp}^{\text{MAC}},$$

y $\mathcal{B}$ es un adversario valido bajo la definición de seguridad MAC.

también observamos que el entorno de $\mathcal{A}$ en $\mathcal{B}$ es idéntico a su entorno en $\mathsf{Exp}^{\text{CTXT}}$. Sigue que podemos calcular

$$\begin{aligned} \Pr[\mathsf{Exp}^{\text{MAC}}(\mathcal{B}) \Rightarrow 1] &= \Pr[\mathsf{Ver}(k, m^*, \tau^*) = 1] && (\text{en } \mathsf{Exp}^{\text{MAC}}) \\ &= \Pr[\mathsf{Ver}(k_m, c_e^*, \tau^*) = 1] && (\text{en } \mathsf{Exp}^{\text{CTXT}} \text{ de } \Pi'). \end{aligned}$$

Por definición de $m = \mathsf{Dec}'(k_m, c = (c_e, \tau))$, '$\mathsf{Ver}(k_m, c_e, \tau) = 0$' $\Rightarrow$ '$m = \perp$'. Por contra-positivo, es equivalente que '$m \neq \perp$' $\Rightarrow$ '$\mathsf{Ver}(k_m, c_e, \tau) = 1$'. Sigue que

$$\Pr[\mathsf{Exp}^{\text{MAC}}(\mathcal{B}) \Rightarrow 1] = \Pr[\mathsf{Ver}(k_m, c_e^*, \tau^*) = 1] \geq \Pr[m^* \neq \perp] = \Pr[\mathsf{Exp}^{\text{CTXT}}(\mathcal{A}) \Rightarrow 1].$$

Por lo tanto, tenemos $\varepsilon'_{\text{CTXT}} \leq \varepsilon_{\text{MAC}}$, y por construcción de $\mathcal{B}$, $t' \approx t$, $q' = q$, y Π' otorga $(\varepsilon_{\text{MAC}}, \approx t, q)$-INT-CTXT.

Y por lo tanto, Π' otorga $(\varepsilon_{\text{CPA}}, \varepsilon_{\text{MAC}}, \approx t, q)$-AE. □

Comentario 40 *Si definiéramos la seguridad MAC de manera que a $\mathcal{A}$ le fuera permitido generar nuevos tags τ' a partir de un mensaje ya etiquetado por $T(\cdot)$ (o sea, el MAC no sería "fuertemente" infalsificable), el resultante cifrado EtM no otorgaría AE. Esto es porque $\mathcal{A}$ podría ver un valido cifrado $c = (c_e, \tau)$ y generar un nuevo cifrado $c' = (c_e, \tau')$. Esto podría ser un problema en el caso especifico de EtM, pero la definición de IND-CTXT es general, y no tendría que solamente funcionar para EtM.*

Ejemplo 3 *Combinar esquemas AE puede no otorgar trivialmente AE. Considérese el esquema natural obtenido concatenando cifrados:* $\mathsf{Enc}'(k, m_1||m_2) := \mathsf{Enc}(k, m_1)||\mathsf{Enc}(k, m_2)$, *donde* Enc *es parte de un SKES que otorga AE, y cada componente del cifrado $c = (c_1, c_2)$ es descifrado individualmente. Este sistema no otorga AE, por que?*

Dado un cifrado (c_1, c_2) valido, el cifrado (c_2, c_1) es también valido. Esto quiere decir que $\mathcal{A}$ puede generar nuevos cifrados validos, invalidando INT-CTXT.

Comentario 41 *AE nos permite de detectar si un cifrado es autentico. Pero no impide que $\mathcal{A}$ reordene, elimine o repita cifrados en transito desde Alicia a Bob. Sin embargo, podemos intentar detectar este tipo de ataque en el nivel aplicativo, definiendo números de secuencia. Esto generalmente resulta en consideraciones dependientes de la aplicación, y que pueden encontrarse al menos parcialmente fuera del dominio de la criptografía.*

Comentario 42 *En algunas aplicaciones a nivel de red, puede ser necesario transmitir datos cifrados, junto a meta-datos (e.g., "packet headers") públicos. Por ejemplo, los meta-datos podrían ser útiles para "routear" los paquetes de datos. En esta situación, una primitiva conocida como "authenticated encryption with associated data" (AEAD) puede ser útil. Esta garantiza confidencialidad de datos, e integridad del cifrado y de los meta-datos. Los dos AEAD mas famosos son AES-GCM y Chacha20-Poly1305.*

Definición 20 (Un AEAD de ejemplo) *Sean* $\Pi = (\mathsf{Gen}^{\Pi}, \mathsf{Enc}, \mathsf{Dec})$ *un SKES,* $\Sigma = (\mathsf{Gen}^{\Sigma}, \mathsf{Tag}, \mathsf{Ver})$ *un arbitrary-length MAC. Definimos un AEAD*

$\mathsf{Gen}'()$	$\mathsf{Enc}'(k, \boxed{d}, m)$	$\mathsf{Dec}'(k, \boxed{d}, c)$
$k_e \leftarrow \mathsf{Gen}^{\Pi}()$	$(k_e, k_m) \leftarrow k$	$(k_e, k_m) \leftarrow k$
$k_m \leftarrow \mathsf{Gen}^{\Sigma}()$	$c_e \leftarrow \mathsf{Enc}(k_e, m)$	$(c_e, \tau) \leftarrow c$
$k \leftarrow (k_e, k_m)$	$\tau \leftarrow \mathsf{Tag}(k_m, \boxed{d}\|\|c_e)$	$b \leftarrow \mathsf{Ver}(k_m, \boxed{d}\|\|c_e, \tau)$
return k	**return** (c_e, τ)	$m \leftarrow \mathsf{Dec}(k_e, c_e)$
		if $b = 1$:
		return m
		else return $\perp$

Comentario 43 *No proponemos una definición de seguridad AEAD. Invitamos el lector a consultar [BS23, § 9.5].*

Con esto terminamos la componente simétrica. Hemos aprendido a cifrar y transmitir cifrados de manera segura entre dos partes Alicia y Bob que comparten una misma clave secreta k. Pero como pueden llegar Alicia y Bob a compartir tal clave? Claro, podrían compartirla de antemano de persona, de manera de estar seguras que ningún adversario $\mathcal{A}$ pueda verla, pero esto no es conveniente. Todos los días navegamos a paginas web, posiblemente por la primera ves, y hacemos pagos online con diferentes entidades. Pre-compartir claves con todas seria prácticamente imposible.

En la segunda parte del curso vamos a ver como transmitir mensajes de manera segura a entidades con las cuales no nos hemos nunca comunicado.