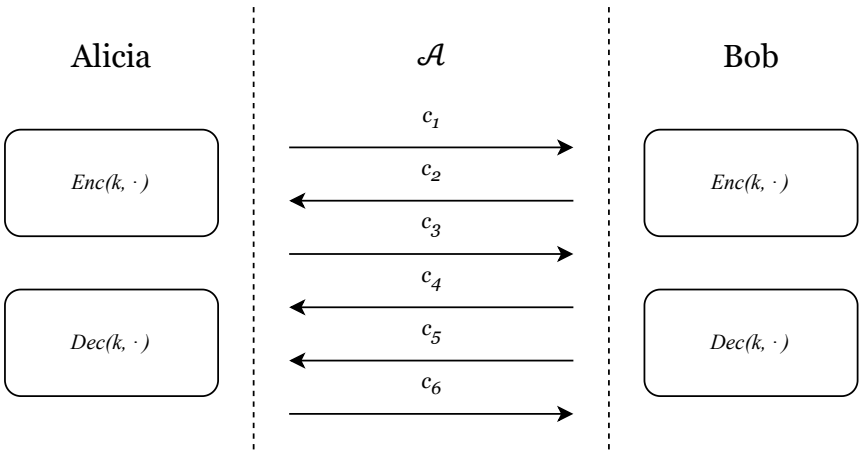
Clase 8: confidencialidad para mensajes múltiples

Fernando Virdia, versión: 0.0.1, junio 2024

4 Cifrar más de un mensaje

Como anteriormente, queremos poder usar un SKES, pero generar mas de un mensaje.



PREGUNTA: Ideas sobre como definiré seguridad "multi-mensajes"?

Podríamos permitirle a $\mathcal{A}$ de ver $c_1, \ldots, c_n$. Vamos a hacer algo mas: le vamos a permitir pedir el cifrado de mensajes arbitrarios! O sea: sin darle la clave k, le vamos a otorgar un oráculo que calcula $\mathsf{Enc}(k, \cdot)$.

PREGUNTA: Esto podría parecer insólito: como podría $\mathcal{A}$ acceder a $\mathsf{Enc}(k, \cdot)$? Ideas?

- Los mensajes podrían ser previsibles, dándole a $\mathcal{A}$ m y $\mathsf{Enc}(k, m)$.
- $\mathcal{A}$ podría tener control del input directamente, por ejemplo si es un comerciante que introduce en el POS el valor a cobrar, y quiere usar $(m, \mathsf{Enc}(k, m))$ para aprender la clave k.
- $\mathcal{A}$ podría saber que el mensaje cifrado es consecuente a alguna acción propia.

Nos inspiramos a la definición de secreto perfecto, donde el adversario trata de distinguir $\mathsf{Enc}(k, m_0)$ de $\mathsf{Enc}(k, m_1)$. En este caso imponemos $|m_0| = |m_1|$ dado que toleraremos $\mathcal{M} = \{0,1\}^*$.

Definición 12 (IND-CPA security) *Sea* $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *un SKES.*

$\mathsf{Exp}^{\mathrm{CPA}}(\mathcal{A}, b)$	$\mathcal{O}(m_0, m_1)$
$k \leftarrow \mathsf{Gen}()$	**if** $\lvert m_0\rvert \neq \lvert m_1\rvert$
$b' \leftarrow \mathcal{A}^{\mathcal{O}}()$	**return** $\perp$
return b'	$c \leftarrow \mathsf{Enc}(k, m_b)$
	return c

Decimos que Π *otorga* (ε, t, q)*-indistinguibilidad bajo ataques de mensaje elegido (IND-CPA, "indistinguishibility under chosen-plaintext attacks"), si para todo adversario* $\mathcal{A}$ *que corre en tiempo* $\leq t$ *y hace* $\leq q$ *queries a* $\mathcal{O}$*, su ventaja es*

$$\mathsf{Adv}(\mathsf{Exp}^{\mathrm{CPA}}, \mathcal{A}) := \left|\Pr[\mathsf{Exp}^{\mathrm{CPA}}(\mathcal{A}, 0) \Rightarrow 1] - \Pr[\mathsf{Exp}^{\mathrm{CPA}}(\mathcal{A}, 1) \Rightarrow 1]\right| \leq \varepsilon.$$

Comentario 26 *Existen definiciones "intermedias" entre (ε, t)-secrecy y (ε, t, q)-IND-CPA. Como referencia, vean [KL20] o [BS23].*

Ejemplo 2

- *El OTP no otorga IND-CPA, dado que $\mathcal{A}$ puede pedir $\mathcal{O}(0^n, 0^n) \Rightarrow k$, y obtenida k, puede pedir $\mathcal{O}(0^n, 1^n) \Rightarrow b^n \oplus k$, y así recuperar b.*
- *Ningún SKES donde $\mathsf{Enc}(k, \cdot)$ es determinista puede otorgar IND-CPA! En cuanto el siguiente ataque aplica:*
  1. $c_1 \leftarrow \mathcal{O}(0^n, 0^n)$
  2. $c_2 \leftarrow \mathcal{O}(0^n, 1^n)$
  3. *Si $c_1 = c_2$, entonces $b = 0$. Si $c_1 \neq c_2$, entonces $b = 1$.*

Definición 13 (PRF-CTR) *Sea $F \colon \mathcal{K} \times \mathcal{D} \to \mathcal{R}$ una (ε, t, q)-PRF con $\mathcal{D} = \{0,1\}^d$ y $\mathcal{R} = \{0,1\}^r$. Definimos un SKES $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ con $\mathcal{M} = \{0,1\}^{\ell \cdot r}$ y $\mathcal{C} = \{0,1\}^{\ell \cdot r + d}$, para algún $\ell \leq 2^d/2$.*

$\mathsf{Gen}()$	$\mathsf{Enc}(k, m)$	$\mathsf{Dec}(k, c = (c_0, c_1, \ldots, c_\ell))$
$k \xleftarrow{\$} \mathcal{K}$	$m_1 \| \ldots \| m_\ell \leftarrow m$, donde $\lvert m_i \rvert = r$	**for** $i = 1, \ldots, \ell$:
return k	$c_0 \xleftarrow{\$} [0, 2^d)$, equiv. $\{0,1\}^d$	$\quad y_i \leftarrow F(k, c_0 + i - 1 \bmod 2^d)$
	for $i = 1, \ldots, \ell$:	$\quad m_i \leftarrow y_i \oplus c_i$
	$\quad y_i \leftarrow F(k, c_0 + i - 1 \bmod 2^d)$	$m \leftarrow m_1 \| \ldots \| m_\ell$
	$\quad c_i \leftarrow y_i \oplus m_i$	**return** m
	$c \leftarrow (c_0, c_1, \ldots, c_\ell)$	
	return c	

Si usamos una PRP/cifrado de bloques en lugar de F, Π es el modo de cifrado "counter mode" (CTR).

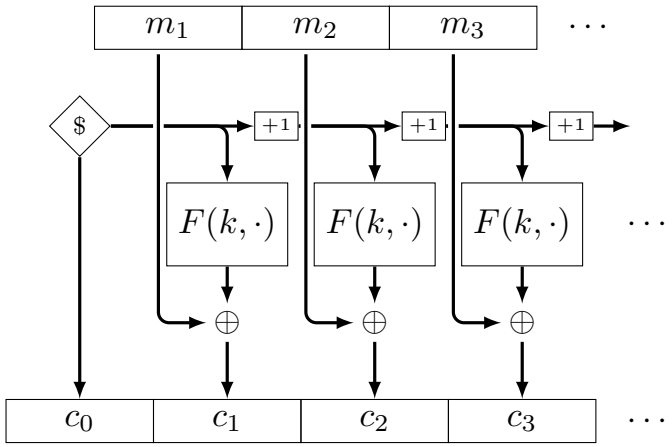


Figure 6: Diagrama de Enc en PRF-CTR. Imagen adaptada de [Ros21, Construction 8.3], usada con permiso del autor.

Comentario 27 *El PRF-CTR puede ser paralelizado de manera simple, dado que los y_i son pueden ser calculados de manera independiente. Por lo tanto tiene muy buena eficiencia al cifrar mensajes muy largos.*

Comentario 28 *Para descifrar, no es necesario "invertir" F. Esto sugiere que si m es largo entre $(\ell - 1) \cdot r$ y $\ell \cdot r$, se pueden "cortar" los últimos bits de y_n en Enc, resultando en un cifrado mas corto.*