Exam Cryptography Engineering 2024

Léo Colisson Palais

Exercice 1: Combining encryptions for better security

Alice and Bob want to securely exchange a message $m \in \mathcal{M} := \{0,1\}^*$. They have access to two encryption schemes $(\mathsf{Gen}_0 \colon \mathbb{N} \to \mathcal{K}_0, \mathsf{Enc}_0 \colon \mathcal{K}_0 \times \mathcal{M} \to \mathcal{C}_0, \mathsf{Dec}_0 \colon \mathcal{K}_0 \times \mathcal{C}_0 \to \mathcal{M})$ and $(\mathsf{Gen}_1 \colon \mathbb{N} \to \mathcal{K}_1, \mathsf{Enc}_1 \colon \mathcal{K}_1 \times \mathcal{M} \to \mathcal{C}_1, \mathsf{Dec}_1 \colon \mathcal{K}_1 \times \mathcal{C}_1 \to \mathcal{M})$, but they only know that at least one of them is secure, without knowing which one is actually secure. In the following exercise, we will study how to perform this securely.

1. (0.5 pts) Here are 3 equivalences between libraries[1]: which one corresponds to the security definition of indistinguishability under (variable-length plaintext) chosen plaintext attack (IND-CPA)? In the following, we will name the corresponding libraries as, respectively, $\mathcal{L}^{\mathsf{Gen},\mathsf{Enc}}_{\text{cpa-L}}$ and $\mathcal{L}^{\mathsf{Gen},\mathsf{Enc}}_{\text{cpa-R}}$.

③	EAVESDROP($m_L, m_R \in \mathcal{M}$):		③	EAVESDROP($m_L, m_R \in \mathcal{M}$):	
④	if $\lvert m_L\rvert \neq \lvert m_R\rvert$ return `err`		④	if $\lvert m_L\rvert \neq \lvert m_R\rvert$ return `err`	
⓪	$k \leftarrow \mathsf{Gen}(1^\lambda)$	$\approx$	⓪	$k \leftarrow \mathsf{Gen}(1^\lambda)$	(1)
5L	$c := \mathsf{Enc}(k, m_L)$		5R	$c := \mathsf{Enc}(k, m_R)$	
⑦	return c		⑦	return c	

①	$k \leftarrow \mathsf{Gen}(1^\lambda)$		①	$k \leftarrow \mathsf{Gen}(1^\lambda)$	
③	EAVESDROP($m_L, m_R \in \mathcal{M}$):		③	EAVESDROP($m_L, m_R \in \mathcal{M}$):	
④	if $\lvert m_L\rvert \neq \lvert m_R\rvert$ return `err`	$\approx$	④	if $\lvert m_L\rvert \neq \lvert m_R\rvert$ return `err`	(2)
5L	$c := \mathsf{Enc}(k, m_L)$		5R	$c := \mathsf{Enc}(k, m_R)$	
⑦	return c		⑦	return c	

①	$k \leftarrow \mathsf{Gen}(1^\lambda)$		①	$k \leftarrow \mathsf{Gen}(1^\lambda)$	
②	$\mathcal{S} := \emptyset$		②	$\mathcal{S} := \emptyset$	
③	EAVESDROP($m_L, m_R \in \mathcal{M}$):		③	EAVESDROP($m_L, m_R \in \mathcal{M}$):	
④	if $\lvert m_L\rvert \neq \lvert m_R\rvert$ return `err`		④	if $\lvert m_L\rvert \neq \lvert m_R\rvert$ return `err`	
5L	$c := \mathsf{Enc}(k, m_L)$	$\approx$	5R	$c := \mathsf{Enc}(k, m_R)$	(3)
⑥	$\mathcal{S} := \mathcal{S} \cup \{c\}$		⑥	$\mathcal{S} := \mathcal{S} \cup \{c\}$	
⑦	return c		⑦	return c	
⑧	DECRYPT($c \in \mathcal{C}$):		⑧	DECRYPT($c \in \mathcal{C}$):	
⑨	if $c \in \mathcal{S}$ return `err`		⑨	if $c \in \mathcal{S}$ return `err`	
⑩	return $\mathsf{Dec}(k, c)$		⑩	return $\mathsf{Dec}(k, c)$	

2. (0.5 pts) For each of these three security definitions, specify if the One-Time Pad (OTP) encryption scheme is secure according to this definition (we temporarily assume for simplicity that the message space $\mathcal{M}$ is equal to the key space $\mathcal{K}$ and $\mathcal{M} = \mathcal{K} = \{0,1\}^n$). No formal proof is expected here, but justify your answer with one or two sentences.

3. (1.25 pts) To securely encrypt a message using Enc_0 and Enc_1 without knowing which one is actually secure, Alice proposes to encrypt m as follows:

$$\mathsf{Enc}(k := (k_0, k_1), m) := \mathsf{Enc}_1(k_1, \mathsf{Enc}_0(k_0, m)) \tag{4}$$

where keys (k_0, k_1) are generated by a procedure $(k_0, k_1) \leftarrow \mathsf{Gen}(1^\lambda)$ by sampling $k_0 \leftarrow \mathsf{Gen}_0(1^\lambda)$ and $k_1 \leftarrow \mathsf{Gen}_1(1^\lambda)$.

Assuming that Enc_1 is IND-CPA secure and that $\lvert\mathsf{Enc}_1(k, m)\rvert = l\lvert m\rvert$ for some integer $l \geq 1$, formally show that there exists Enc_0 such that Enc is *not* IND-CPA secure (more precisely, exhibit

[1]The gray numbers like ① are just used to label lines so that you can quickly refer to them without rewriting them fully.

a function Enc_0 and an adversary $\mathcal{A}$ following the Joy of Cryptography notation seen in the course, and compute its advantage according to the IND-CPA security definition).

Hint: you can choose Enc_0 arbitrarily, in particular it may not preserve the length of its input.

4. In order to avoid the above attack, Alice has the idea to use a so-called secret-sharing operation to split the message m into two "shares" m_0 and m_1 such that $m = m_0 \oplus m_1$, and encrypt m_0 with Enc_0 and m_1 with Enc_1. More precisely, we consider the procedure $(k_0, k_1) \leftarrow \mathsf{Gen}(1^\lambda)$ defined above and the encryption as:

$$\begin{array}{|ll|}\hline \multicolumn{2}{|c|}{\mathsf{Enc}(k := (k_0, k_1), m)} \\ \hline (11) & m_0 \xleftarrow{\$} \{0,1\}^{|m|} \\ (12) & m_1 := m_0 \oplus m \\ (13) & \textbf{return } (\mathsf{Enc}_0(k_0, m_0), \mathsf{Enc}_1(k_1, m_1)) \\ \hline \end{array} \tag{5}$$

(a) (0.75 pts) Describe the decryption algorithm Dec and prove its correctness, i.e. that:

$$\Pr\left[\,\mathsf{Dec}((k_0, k_1), \mathsf{Enc}((k_0, k_1), m)) = m\,\right] = 1 \tag{6}$$

(b) (1.25 pts) Assuming that Enc_1 is IND-CPA secure, show that Enc is IND-CPA secure (justify all equations and details all steps).

NB: to save typing, you can name your intermediate libraries, number lines like (20) (just use numbers greater than 18 to avoid naming clash) and reuse this number instead of rewriting the whole line.

(c) (0.75 pts) For any $m \in \{0,1\}^n$, prove that the following distribution is a uniform distribution over $S := \{(m_0', m_1') \in \{0,1\}^n \times \{0,1\}^n \mid m_0' \oplus m_1' = m\}$, i.e. for any $(m_0', m_1') \in \{0,1\}^n \times \{0,1\}^n$ such that $m = m_0' \oplus m_1'$:

$$\Pr_{\substack{m_0 \xleftarrow{\$} \{0,1\}^n \\ m_1 := m_0 \oplus m}}\left[\,(m_0, m_1) = (m_0', m_1')\,\right] = \frac{1}{2^n} \tag{7}$$

Similarly, prove that

$$\Pr_{\substack{m_0 \xleftarrow{\$} \{0,1\}^n \\ m_1 := m_0 \oplus m}}\left[\,(m_1, m_0) = (m_0', m_1')\,\right] = \frac{1}{2^n} \tag{8}$$

and conclude that

$$\begin{array}{|ll|}\hline \multicolumn{2}{|c|}{\text{share-1}} \\ \hline (14) & \underline{\textsc{sample}(m):} \\ (15) & \quad m_0 \xleftarrow{\$} \{0,1\}^n \\ (16) & \quad m_1 := m_0 \oplus m \\ (17.1) & \quad \textbf{return } (m_0, m_1) \\ \hline \end{array} \equiv \begin{array}{|ll|}\hline \multicolumn{2}{|c|}{\text{share-2}} \\ \hline (14) & \underline{\textsc{sample}(m):} \\ (15) & \quad m_0 \xleftarrow{\$} \{0,1\}^n \\ (16) & \quad m_1 := m_0 \oplus m \\ (17.2) & \quad \textbf{return } (m_1, m_0) \\ \hline \end{array} \tag{9}$$

(d) (1 pts) Assuming that Enc_0 is IND-CPA secure, show that Enc is IND-CPA secure.

NB: to save typing, you can apply the same advice as in the above proof involving Enc_1, and you can also skip the externalize-replace-inline operations by only writing the starting and ending libraries, and **quickly describing all skipped steps**.