CS-E4340 Cryptography: Exercise Sheet 5

—Symmetric Encryption Schemes (SE)—

Submission Deadline: October 10, 11:30 via MyCourses

You may not use theorems from the Crypto companion for this exercise sheet.

Each exercise can give up to two participation points, 2 for a mostly correct solution and 1 point for a good attempt. Overall, the exercise sheet gives at most 4 participation points.

Exercise Sheet 5 is intended to help...

(a) ...understanding the security games indistinguishability under chosen plaintext attacks (IND-CPA) and the authenticated encryption security (AE) for a symmetric encryption scheme (SE).
(b) ...familiarizing yourself further with the notion of a reduction and learning to carry out reductions yourself. In particular, you will be able to practice inlining and spotting errors in an inlining proof.

Exercise 1 then follows the tradition and shows that some schemes that look somewhat unnatural can still be secure.
Exercise 2 returns to the question of what is a good definition.
Exercise 3 considers a whole class of encryption schemes at the same time and shows that they can never be secure according to our definition.
Exercise 4 takes up the challenge from lecture 5 on how to combine MACs and IND-CPA-secure encryption schemes into an authenticated encryption scheme.
Exercise 5 considers a quite canonical encryption scheme.
Ex. 2, 3 & 4 ask for adversary constructions.
Ex. 1, 4 & 5 cover security proofs via reduction.

Exercise 1. (Candidate Encryption Schemes) Let se be an IND-CPA secure encryption scheme.

1. Do you think these encryption schemes are IND-CPA secure? Justify your intuition for each scheme.
2. Choose one of the secure schemes and prove security by giving a reduction (in pseudocode) and analyze its success probability.

$\underline{se_1.enc(k,m)}$
$c' \leftarrow\$\ se.enc(k,m)$
$c \leftarrow c'||1$
return c

$\underline{se_1.dec(k,c)}$
$c' \leftarrow c[1..|c|-1]$
$m \leftarrow se.dec(k,c')$
return m

$\underline{se_2.enc(k,m)}$
$m' \leftarrow m||0$
$c \leftarrow\$\ se.enc(k,m')$
return c

$\underline{se_2.dec(k,c)}$
$m' \leftarrow se.dec(k,c)$
$m \leftarrow m'[1..|m|-1]$
return m

$\underline{se_3.enc(k,m)}$
$c_0 \leftarrow\$\ se.enc(k,m)$
$c_1 \leftarrow\$\ se.enc(k,m)$
$c \leftarrow (c_0,c_1)$
return c

$\underline{se_3.dec(k,c)}$
parse $(c_0,c_1) \leftarrow c$
$m \leftarrow se.dec(k,c_0)$
$m' \leftarrow se.dec(k,c_1)$
if $m = m'$ **return** m
else return $\perp$

Exercise 2. (Different Definitions) The goal of AE security is to make sure an adversary is unable to "come up" with a decrypting ciphertext on its own. Consider the following approach to capture this property. Let the games $\texttt{Gae'}^0_{se}$ and $\texttt{Gae'}^1_{se}$ be defined as

$\underline{\texttt{Gae'}^0_{se}}$

Parameters

λ:	sec. parameter
se:	sym. enc. sch.

Package State

k:	key

ENC(x)

if $k = \bot$:
  $k \leftarrow\!\$\ \{0,1\}^\lambda$
$c \leftarrow\!\$\ se.enc(k, x)$
return c

DEC(c)

if $k = \bot$:
  $k \leftarrow\!\$\ \{0,1\}^\lambda$
$x \leftarrow se.dec(k, c)$
return x

$\underline{\texttt{Gae'}^1_{se}}$

Parameters

λ:	sec. parameter
se:	sym. enc. sch.

Package State

k:	key
$\mathcal{L}$:	set

ENC(x)

if $k = \bot$:
  $k \leftarrow\!\$\ \{0,1\}^\lambda$
$x' \leftarrow 0^{|x|}$
$c \leftarrow\!\$\ se.enc(k, x')$
$\mathcal{L} \leftarrow \mathcal{L} \cup \{c\}$
return c

DEC(c)

if $c \in \mathcal{L}$
  $x \leftarrow se.dec(k, c)$
  return x
else
  return $\bot$

Show that no correct encryption scheme is secure according to this definition by giving a successful attacker against the AE' security (in pseudocode) and analyze its success probability.

Exercise 3. (Deterministic Encryption Schemes[1]) Consider a different, correct encryption scheme $se' = (se'.enc, se'.dec)$ such that $se'.enc$ is *deterministic*. Show that se' is not IND-CPA-secure by giving a successful attacker against the IND-CPA security of se' (in pseudocode) and analyze its success probability.

Exercise 4. (Authenticated Encryption Schemes) Let $se = (se.enc, se.dec)$ be a symmetric encryption scheme, and let $m = (m.mac, m.ver)$ be a message authentication code. Let the operation **parse** be an operation with parses a variable $v = (v_0, v_1)$ encoding a pair into its individual component. We construct three encryption schemes $se_a^{se,m}$ with $a \in \{\text{m+e}, \text{mte}, \text{etm}\}$ where m+e (mac-and-encrypt), mte (mac-then-encrypt), and etm (encrypt-then-mac):

[1] In the literature, you might also encounter nonce-based encryption. Note that in the syntax of nonce-based encryption, the nonce takes the place of the randomness, and the syntax is then described as a deterministic function which maps the key, the message and the nonce to a ciphertext. As long as nonces don't repeat, nonce-based encryption is also a reasonable alternative way of formalizing syntax of an encryption scheme. Note that in this case, the IND-CPA definition needs to be adapted. We omit the discussion here. See `https://web.cs.ucdavis.edu/~rogaway/papers/nonce.pdf` if you are curious.

$se_{\text{m+e}}.enc(k, x)$

parse $(k_m, k_{se}) \leftarrow k$
$c' \leftarrow\!\!\$\ se.enc(k_{se}, x)$
$\tau \leftarrow m.mac(k_m, x)$
$c \leftarrow (c', \tau)$
return c

$se_{\text{mte}}.enc(k, x)$

parse $(k_m, k_{se}) \leftarrow k$
$\tau \leftarrow m.mac(k_m, x)$
$x' \leftarrow (x, \tau)$
$c \leftarrow\!\!\$\ se.enc(k_{se}, x')$
return c

$se_{\text{etm}}.enc(k, x)$

parse $(k_m, k_{se}) \leftarrow k$
$c' \leftarrow\!\!\$\ se.enc(k_{se}, x)$
$\tau \leftarrow m.mac(k_m, c')$
$c \leftarrow (c', \tau)$
return c

$se_{\text{m+e}}.dec(k, c)$

parse $(k_m, k_{se}) \leftarrow k$
parse $(c', \tau) \leftarrow c$
$x \leftarrow se.dec(k_{se}, c')$
if $1 \leftarrow m.ver(k_m, x, \tau)$
  return x
else return $\perp$

$se_{\text{mte}}.dec(k, c)$

parse $(k_m, k_{se}) \leftarrow k$
$x' \leftarrow se.dec(k_{se}, c)$
parse $(x, \tau) \leftarrow x'$
if $1 \leftarrow m.ver(k_m, x, \tau)$:
  return x
else return $\perp$

$se_{\text{etm}}.dec(k, c)$

parse $(k_m, k_{se}) \leftarrow k$
parse $(c', \tau) \leftarrow c$
$x \leftarrow se.dec(k_{se}, c')$
if $1 \leftarrow m.ver(k_m, c', \tau)$:
  return x
else return $\perp$

Say for each for the following 6 statements whether you think that it is true or false. Justify your opinion: For all `UNF-CMA` MAC schemes m and for all `IND-CPA` secure symmetric encryption schemes se, it holds that...

(1) ...the encryption scheme $se^{se,m}_{\text{m+e}}$ is `AE`-secure.
(2) ...the encryption scheme $se^{se,m}_{\text{m+e}}$ is `IND-CPA`-secure.
(3) ...the encryption scheme $se^{se,m}_{\text{mte}}$ is `AE`-secure.
(4) ...the encryption scheme $se^{se,m}_{\text{mte}}$ is `IND-CPA`-secure.
(5) ...the encryption scheme $se^{se,m}_{\text{etm}}$ is `AE`-secure.
(6) ...the encryption scheme $se^{se,m}_{\text{etm}}$ is `IND-CPA`-secure.

Exercise 5. (Constructing secure encryption from PRF) (Advanced) Let f be a secure $(\lambda, *)$-PRF. Consider the candidate encryption scheme se.

$se.enc(k, m)$

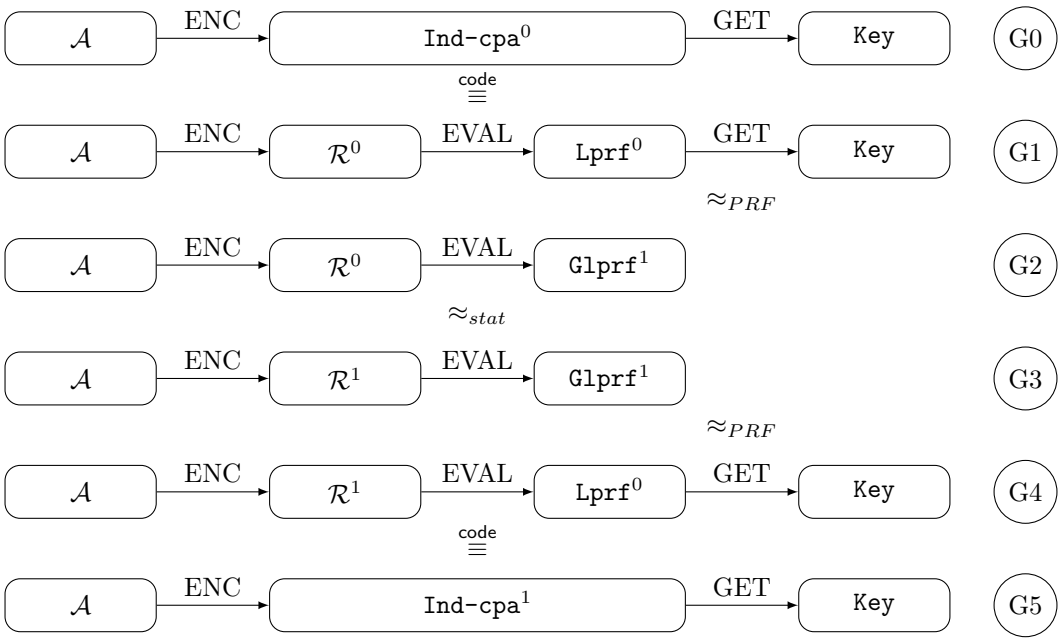
$r \leftarrow\!\!\$\ \{0,1\}^{\lambda}$
$r' \leftarrow f(k, r, |m|)$
$c \leftarrow ((r' \oplus m), r)$
return c

$se.dec(k, c)$

parse $(c', r) \leftarrow c$
$r' \leftarrow f(k, r, |c'|)$
$m \leftarrow c' \oplus r'$
return m

Show (via reduction) that se is an IND-CPA-secure encryption scheme. Provide pseudocode for your reduction and show that the probability of winning the `IND-CPA` game is negligible. See Security Definition 3.13 in the Crypto Companion (p. 30) for the definition of $\texttt{Glprf}^0$ and $\texttt{Glprf}^1_f$.

Hint: The following graphs show the intermediate steps that proof that the real game is indistinguishable from the ideal game. Feel free to skip some of the steps if you are stuck. $\approx_{\text{stat}}$ refers to a (negligible) *statistical* difference between the two games.



Definition 1 (IND-CPA). *A symmetric encryption scheme se is IND-CPA-secure if the real game* $\texttt{Gind-cpa}^0$ *and the ideal game* $\texttt{Gind-cpa}^1$ *are computationally indistinguishable, that is, for all PPT adversaries* $\mathcal{A}$*, the advantage*

$$\begin{aligned}&\mathbf{Adv}_{\mathcal{A}}^{\texttt{Gind-cpa}^0,\texttt{Gind-cpa}^1}(\lambda)\\ &:= \left|\Pr\left[1 = \mathcal{A} \circ \texttt{Gind-cpa}^0\right] - \Pr\left[1 = \mathcal{A} \circ \texttt{Gind-cpa}^1\right]\right|\end{aligned}$$

is negligible in λ*.*

$\underline{\underline{\texttt{Gind-cpa}^0_{se}}}$

Parameters

λ:	*sec. parameter*
se:	*sym. enc. sch.*

Package State

k:	*key*

ENC(x)

if $k = \bot$:
  $k \leftarrow\!\!\$\ \{0,1\}^\lambda$

$c \leftarrow\!\!\$\ se.enc(k, x)$
return c

$\underline{\underline{\texttt{Gind-cpa}^1_{se}}}$

Parameters

λ:	*sec. parameter*
se:	*sym. enc. sch.*

Package State

k:	*key*

ENC(x)

if $k = \bot$:
  $k \leftarrow\!\!\$\ \{0,1\}^\lambda$
$x' \leftarrow 0^{|x|}$
$c \leftarrow\!\!\$\ se.enc(k, x')$
return c