TD 1 Cryptographie

Léo Colisson Palais

Exercice 1: Librairies indistinguables

1. Calculer $\Pr[\mathcal{A}_1 \diamond \mathcal{L}_1 = \mathsf{true}]$, $\Pr[\mathcal{A}_1 \diamond \mathcal{L}_2 = \mathsf{true}]$, $\Pr[\mathcal{A}_2 \diamond \mathcal{L}_1 = \mathsf{true}]$, $\Pr[\mathcal{A}_2 \diamond \mathcal{L}_2 = \mathsf{true}]$ avec

$\mathcal{A}_1$
$r_1 \leftarrow \text{RAND}(6)$ $r_2 \leftarrow \text{RAND}(6)$ return $r_1 \stackrel{?}{=} r_2$

$\mathcal{A}_2$
$r_1 \leftarrow \text{RAND}(6)$ $r_2 \leftarrow \text{RAND}(6)$ return $r_1 \stackrel{?}{\geq} 3$

$\mathcal{L}_1$
RAND(n): $r \xleftarrow{\$} \mathbb{Z}_n$ return r

$\mathcal{L}_2$
RAND(n): return 0

2. Les librairies suivantes sont elles indistinguables ? (si oui, décrivez un distingueur et calculez sa probabilité de succès, exercice à faire sur Moodle en python):

(a)

$\mathcal{L}_A$		$\mathcal{L}_B$
SAMPLE(): **return** 42	$\stackrel{?}{\approx}$	SAMPLE(): **return** 45

(b)

$\mathcal{L}_A$		$\mathcal{L}_B$
SAMPLE(): $x \xleftarrow{\$} \mathbb{Z}_{10}$ **return** x	$\stackrel{?}{\approx}$	SAMPLE(): $x \xleftarrow{\$} \mathbb{Z}_9$ **return** x

(c)

$\mathcal{L}_A$		$\mathcal{L}_B$
SAMPLE(): $x \xleftarrow{\$} \mathbb{Z}_{10}$ **return** x	$\stackrel{?}{\approx}$	SAMPLE(): $x \xleftarrow{\$} \mathbb{Z}_{10}$ **return** $2 + x$

(d)

$\mathcal{L}_A$		$\mathcal{L}_B$
SAMPLE(): $x \xleftarrow{\$} \mathbb{Z}_{10}$ **return** x	$\stackrel{?}{\approx}$	SAMPLE(): $x \xleftarrow{\$} \mathbb{Z}_{10}$ **return** $(2 + x) \mod 10$

(e)

$\mathcal{L}_A$		$\mathcal{L}_B$
$a \coloneqq 0$ SAMPLE(): **return** 42	$\stackrel{?}{\approx}$	$a \coloneqq 1$ SAMPLE(): $b \coloneqq 8$ **return** 42

(f)

$\mathcal{L}_A$		$\mathcal{L}_B$	$\diamond$	$\mathcal{L}_C$
SAMPLE(): **return** 9	$\stackrel{?}{\approx}$	SAMPLE(): **return** SQUARE (3)	$\diamond$	SQUARE(x): **return** x^2

(g)

$\mathcal{L}_A$		$\mathcal{L}_B$
SAMPLE(): $x \xleftarrow{\$} \mathbb{Z}_{10}$ **if** $x \stackrel{?}{=} 0$ **then** $x \xleftarrow{\$} \mathbb{Z}_{10}$ **return** x	$\stackrel{?}{\approx}$	SAMPLE(): $x \xleftarrow{\$} \mathbb{Z}_{10}$ **return** x

(h) Les librairies de la définition de sécurité IND-CPA avec le chiffrement $\mathsf{Gen}(1^\lambda)$ qui retourne toujours 0, et $\mathsf{Enc}_k(m) \coloneqq \bar{m}$, où $m \in \{0,1\}^\lambda$, $\bar{m}$ inverse m bit à bit (0 devient 1 et 1 devient 0).

(i) Les librairies de la définition de sécurité IND-CPA avec le chiffrement du One-Time Pad, i.e. $\mathsf{Gen}(1^\lambda)$ qui retourne une clé aléatoire dans $\{0,1\}^\lambda$, et $\mathsf{Enc}_k(m) \coloneqq m \oplus k$.

(j) Les librairies de la définition de sécurité IND-CPA avec un chiffrement inconnu mais déterministe.

Exercice 2: Fonction négligeable et manipulation de librairie

1. Lesquelles de ces fonction sont négligeables ? (Prouvez-le, la formule $a^b = 2^{b \log a}$ peut-être utile)

$$\frac{1}{2^{\lambda/2}} \quad \frac{1}{2^{\log(\lambda^2)}} \quad \frac{1}{\lambda^{\log(\lambda)}} \quad \frac{1}{\lambda^2} \quad \frac{1}{2^{\log \lambda^2}} \quad \frac{1}{\lambda^{1/\lambda}} \quad \frac{1}{\sqrt{\lambda}} \quad \frac{1}{2^{\sqrt{\lambda}}}$$

2. Montrer que si f et g sont négligeable, alors $f+g$ et fg le sont également.

3. Montrer que si $f = \mathsf{poly}(\lambda)$ et $g = \mathsf{negl}(\lambda)$, alors $fg = \mathsf{negl}(\lambda)$.

Exercice 3: Un simple partage de secret

Nous considérons les librairies suivantes :

$\mathcal{L}_{\text{ot-real}}$
QUERY$(m \in \{0,1\}^\lambda)$:
$r \xleftarrow{\$} \{0,1\}^\lambda$
$y \coloneqq r \oplus m$
return y

$\mathcal{L}_{\text{ot-rand}}$
QUERY$(m \in \{0,1\}^\lambda)$:
$r \xleftarrow{\$} \{0,1\}^\lambda$
return r

$\mathcal{L}_{\text{left}}$
QUERY$(m \in \{0,1\}^\lambda)$:
$r \xleftarrow{\$} \{0,1\}^\lambda$
$y \coloneqq r \oplus m$
return (r, y)

$\mathcal{L}_{\text{right}}$
QUERY$(m \in \{0,1\}^\lambda)$:
$r \xleftarrow{\$} \{0,1\}^\lambda$
$y \coloneqq r \oplus m$
return (y, r)

1. Montrer que $\mathcal{L}_{\text{ot-real}} \equiv \mathcal{L}_{\text{ot-rand}}$ Indice: utiliser la méthode "calcul des probabilités".

2. Utilise cette propriété pour montrer que le OTP est sécurisé en usage unique, i.e. que les deux librairies suivantes sont indistinguables :

$\mathcal{L}^{\Sigma}_{\text{ots-L}}$
EAVESDROP$(m_L, m_R \in \mathcal{M})$:
$k \leftarrow \mathsf{Gen}(1^\lambda)$
return $\mathsf{Enc}_k(m_L)$

$\equiv$

$\mathcal{L}^{\Sigma}_{\text{ots-R}}$
EAVESDROP$(m_L, m_R \in \mathcal{M})$:
$k \leftarrow \mathsf{Gen}(1^\lambda)$
return $\mathsf{Enc}_k(m_R)$

3. Montrez que $\mathcal{L}_{\text{left}} \equiv \mathcal{L}_{\text{right}}$. Pouvez-vous utiliser directement le fait que $\mathcal{L}_{\text{ot-real}} \equiv \mathcal{L}_{\text{ot-rand}}$? Si oui, prouvez-le, sinon, montrez où la preuve naïve échoue.

4. Un schéma de partage secret à seuil t sur n (TSSS) consiste en deux algorithmes

   - $\mathsf{Share}(m \in \mathcal{M})$ qui produit une séquence $s = (s_1, \ldots, s_n)$ de parts,
   - $\mathsf{Reconstruct}(\{s_1, \ldots, s_k\})$ qui produit un message $m \in \mathcal{M}$ si $k \geq t$ et $\perp$ sinon.

   tels que :

   - Correction : pour tout $m \in \mathcal{M}$ et tout $U \subseteq \{1, \ldots, n\}$ avec $|U| \geq t$, et pour tout $s \leftarrow \mathsf{Share}(m)$, on a $\mathsf{Reconstruct}(\{s_i \mid i \in U\}) = m$,
   - Sécurité : on a

$\mathcal{L}_{\text{tsss-L}}$
SHARE(m_L, m_R, U):
if $|U| \geq t$, return err
$s \leftarrow$ SHARE(m_L)
return $\{s_i \mid i \in U\}$

$\equiv$

$\mathcal{L}_{\text{tsss-R}}$
SHARE(m_L, m_R, U):
if $|U| \geq t$, return err
$s \leftarrow$ SHARE(m_R)
return $\{s_i \mid i \in U\}$

(1)

(a) Expliquez pourquoi on appelle cela un "schéma de partage secret".

(b) La construction suivante est-elle sécurisée ? Si oui, prouvez-le, sinon, trouvez un attaquant explicite.

$\mathcal{M} = \{0,1\}^{500}$, $t = 5$, $n = 5$

Share(m):
split m into $m = s_1 \| \cdots \| s_5$, where each $|s_i| = 100$
return $(s_1, \ldots, s_5)$

Reconstruct($s_1, \ldots, s_5$):
return $s_1 \| \cdots \| s_5$

(c) On considère un schéma simple de partage secret 2-sur-2, où Share est défini comme la QUERY dans $\mathcal{L}_{\text{left}}$. Décrivez la procédure Reconstruct.

(d) Prouvez que ce schéma est sécurisé.

Indice : faites une distinction de cas selon la taille/valeur de U

(e) Pouvez-vous généraliser cette construction pour obtenir un schéma de partage secret 2-sur-k pour un $k \in \mathbb{N}^*$ quelconque et en prouver la sécurité?

Exercice 4: Security of OTP

1. Quelqu'un se rend compte que l'OTP révèle le message quand la clé est $0 \ldots 0$, et propose de choisir la clé dans $\{0,1\}^\lambda \setminus \{0^\lambda\}$ au lieu de $\{0,1\}^\lambda$. Est-ce plus (ou moins) sécurisé ? Si oui, prouvez-le, sinon trouvez un attaquant cassant la one-time security du schéma (c'est-à-dire un adversaire qui distingue $\mathcal{L}^{\Sigma}_{\text{ots-L}}$ de $\mathcal{L}^{\Sigma}_{\text{ots-R}}$).

2. Pour obtenir une sécurité supplémentaire, Alice décide de chiffrer deux fois le message avec OTP. Quels sont les impacts réels en termes de sécurité *(i)* si Alice utilise la même clé k pour les deux chiffrages *(ii)* si Alice utilise des clés différentes ?

3. Qu'y a-t-il de particulier avec la fonction XOR de l'OTP ? Est-ce que cela fonctionnerait correctement et/ou serait sécurisé avec, par exemple, un AND au lieu d'un XOR ? Est-ce que ça marcherait si on interprète les chaînes comme des entiers modulo 2^λ et remplace le XOR par une addition modulaire ? (prouvez formellement vos affirmations)

4. Montrez que le schéma de chiffrement suivant n'a pas de secret one-time, en construisant un programme qui distingue les deux bibliothèques pertinentes de la définition de one-time security.

$\mathcal{K} = \{1, \ldots, 9\}$, $\mathcal{M} = \{1, \ldots, 9\}$, $\mathcal{C} = \mathbb{Z}_{10}$

Gen:
$k \leftarrow \{1, \ldots, 9\}$
return k

Enc(k, m):
return $k \times m \% 10$

5. Toi (Eve), tu as intercepté deux textes chiffrés :

$$c_1 = 11111001011110011100110000010111110000110$$
$$c_2 = 11111010011001111101110100001001100010000$$

Tu sais que les deux sont des chiffrés OTP, encryptés avec la même clé. Tu sais que soit *(i)* c_1 est le chiffré de `alpha` et c_2 est le chiffré de `bravo`, soit *(ii)* c_1 est le chiffré de `delta` et c_2 est le chiffré de `gamma` (tous convertis en binaire à partir de l'ASCII de manière standard, c'est-à-dire $\mathtt{a} = 97, \mathtt{b} = 98 \ldots$). Quelle de ces deux possibilités est correcte, et pourquoi ? Peux-tu retrouver la clé ?