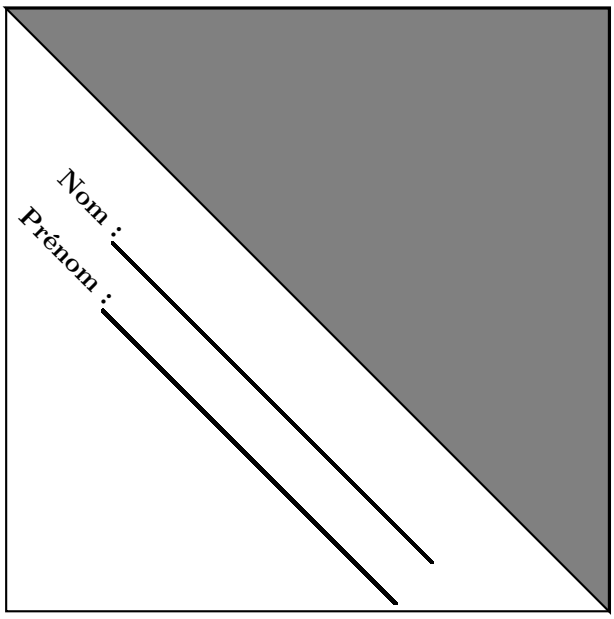
Université Côte d'Azur
SI5, CyberSec & M2 informatique et interactions
Cryptographie & Sécurité 2022–2023

Examen novembre 2022

Durée : 2h

Note :



L'examen comporte quatre parties indépendantes. Veuillez répondre sur la copie avec clarté et concision.

1 Quizz sur la sécurité [5 points]

1. Dans quel cas peut-on utiliser une fonction à sens unique sans trappe ?

2. Par quelle technique arrive-t-on à rendre RSA sémantiquement sûr ?

3. Expliquez succinctement le fonctionnement des deux principaux modèles de confiance utilisés pour les certificats.

4. Expliquez brièvement la différence entre `/dev/random` et `/dev/urandom`.

5. Expliquez pourquoi on utilise un mode de chaînage avec les chiffres à clé secrète.

2 Hachage compressif [4 points]

Nous nous intéressons à la construction de Merkle-Damgård d'une fonction de hachage issue de la fonction de compression suivante :

Soient $b(x)$ et $k(x)$ deux polynômes sur $\mathbb{F}_2[x]$ tels que : $\deg(b) \leq 3$ et $\deg(k) \leq 2$. Soit θ l'application qui à un mot binaire b (bit de poids faible à droite) associe sa représentation polynomiale. On a pour le mot b de 4 bits : $b_3b_2b_1b_0$, $\theta(b) = b_3x^3 + b_2x^2 + b_1x + b_0$. L'application réciproque θ^{-1} permet d'associer un mot binaire à un polynôme. La fonction de compression g est définie par :

$$g(k, b) = \theta^{-1}\left(x \cdot \theta(k) + \theta(b) \mod x^3 + x^2 + 1\right)$$

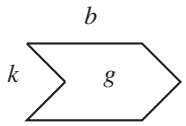


FIGURE 1 – Illustration du fonctionnement de chaînage de la fonction de compression g.

1. Donnez les paramètres de la fonction de compression g (nombre de bits d'entrée et de sortie).

2. Calculez l'empreinte du mot hexadécimal `1a` de codage binaire `0 0 0 1 1 0 1 0` avec IV= 0.

3. Utilisez le paradoxe des anniversaires pour trouver combien d'entrées il faudrait considérer pour trouver une collision avec une probabilité supérieure à 3/4. On rappelle que $\ln(2) \simeq 0.7$, $\ln(3) \simeq 1.1$, $\ln(4) \simeq 1.4$ et que $\sqrt{1+x} \simeq 1 + \frac{x}{2}$ au voisinage de 0.

4. Dans notre cas, il est possible de construire une collision. Expliquez comment et illustrez votre construction.

3 Code de Huffman [3 points]

Une source qui émet 6 symboles a donné lieu à l'arbre de Huffman de la Figure 2. Chaque symbole est porté par une feuille de l'arbre et les bits qui construisent les mots du code sont inscrits sur les arêtes.

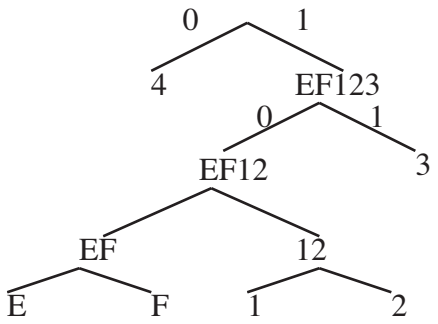


FIGURE 2 – Arbre de Huffman.

1. Décodez le mot suivant issu de la compression par Huffman (lu de la gauche vers la droite) :

0 1 1 0 1 0 1 1 0 1 0 0 1 0 1 0 0 0 0 1 1 0 1 0 1 0

On rappelle ci-dessous les valeurs hexadécimales du code ASCII sur 8 bits des lettres majuscules :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
41	42	43	44	45	44	47	48	49	$4a$	$4b$	$4c$	$4d$	$4e$	$4f$
P	Q	R	S	T	U	V	W	X	Y	Z				
50	51	52	53	54	55	56	57	58	59	$5a$	20			

Si on code en hexadécimal la version compressée (avec un bourrage de tête par des `0`), bit de poids faible à droite, on obtient la valeur hexadécimale : `1ad286a`.

2. Calculez le rapport de compression lorsque la donnée brute et la donnée compressée sont exprimées :

1. en binaire :

2. en hexadécimal :

Quelle est la donnée compressée la plus intéressante en terme de rapport de compression ?

3. Expliquez pourquoi une opération de compression ne peut pas donner de bon résultat lorsqu'elle est appliquée après le chiffrement.

4 Procédé de signcryption [8 points]

On considère un procédé qui combine un chiffrement (à clé secrète) et une signature d'El Gamal entre Alice (qui envoie le message m chiffré et signé) et Bob (qui le reçoit).
Ils partagent p un grand entier premier, g un générateur de $\mathbb{Z}_p^\star$, h une fonction de hachage cryptographique et un algorithme de chiffrement (resp. déchiffrement) à clé secrète E_k (resp. D_k).

On note y_a et y_b les valeurs échangées entre Alice et Bob par Diffie-Hellman ; a, la valeur aléatoire choisie par Alice et b celle choisie par Bob.

1. Rappelez le contenu de y_a et y_b échangés entre Alice et Bob lors de l'échange initial par Diffie-Hellman.

Alice connait à présent y_b et Bob y_a. Alice choisit alors x, une valeur aléatoire de $\mathbb{Z}_p^\star$ et calcule :

- $k = y_b^x \mod p$ et coupe k en k_1 et k_2 pour obtenir deux clés ;
- $c = E_{k_1}(m)$ où c est le chiffré de m qu'Alice veut transmettre à Bob ;
- $r = h(k_2||m)$ où $||$ est l'opération de concaténation ;
- $s = x \cdot (r + a)^{-1} \mod p - 1$

Alice transmet le triplet (c, r, s) à Bob.

2. Dites quelles sont les propriétés de sécurité (confidentialité,...) assurées par ce mécanisme en justifiant brièvement vos réponses :

Bob doit maintenant récupérer m et en vérifier la signature à partir du triplet (c, r, s) reçu. Il doit tout d'abord retrouver la valeur de k à l'origine des clés.

3. Montrez que k est retrouvée en calculant $(y_a \cdot g^r)^{s \cdot b} \mod p$.

Une fois que Bob a retrouvé la clé k, il la coupe en $k_1||k_2$.

4. Expliquez quelle est la trappe de ce mécanisme à clé publique et ce qui en assure la sécurité.

5. Dites comment Bob peut maintenant retrouver m, chiffré par Alice.

Bob connaît maintenant $k = k_1||k_2$ et m, obtenus à partir du triplet (c, r, s).

6. Dites comment Bob vérifie la signature de m :

7. Décrivez à présent une attaque active (simple) contre ce procédé de signcryption et une manière de la prévenir.

8. Quel serait l'usage idéal de ce procédé de signcryption ?