





TD3

Ce TD porte sur différents aspects cryptologiques.

Exercice n°1 : protocole de Diffie-Helman

Soient la suite d'échanges suivant entre Alice et Bob :

	 Alice	 Bob
Etape 1	Alice et Bob choisissent ensemble un grand nombre premier p et un entier $1 \leq a \leq p-1$. Cet échange n'a pas besoin d'être sécurisé.	
Etape 2	Alice choisit secrètement x_1 .	Bob choisit secrètement x_2 .
Etape 3	Alice calcule $y_1 = a^{x_1} \pmod{p}$	Bob calcule $y_2 = a^{x_2} \pmod{p}$
Etape 4	Alice et Bob s'échangent les valeurs y_1 et y_2 . Cet échange n'a pas besoin d'être sécurisé.	
Etape 5	Alice calcule $y_2^{x_1} = (a^{x_2})^{x_1} = a^{x_1 x_2} \pmod{p}$ et appelle ce nombre k	Bob calcule $y_1^{x_2} = (a^{x_1})^{x_2} = a^{x_1 x_2} \pmod{p}$ et appelle ce nombre k

(a) D'après vous, est-ce que Alice et Bob ont la même valeur k ? Par conséquent, à quoi peut servir le protocole Diffie-Hellman ?

(b) Testez l'algorithme avec $p = 23$ et $a = 3$.

(c) D'après vous, sur quel principe mathématique se fonde le protocole Diffie-Hellman ?

(d) D'après vous, ce protocole est-il vulnérable à une attaque de type homme du milieu ? Montrez comment une attaque du milieu pourrait survenir. Proposez une solution.

Exercice n°2 : étude de HMAC (*keyed-Hash Message Authentication Code*) et implémentation avec SHA256.

(a) Lisez l'extrait de la RFC-2104.

(b) Donnez un algorithme générique du calcul HMAC.

(c) Donnez une implémentation en langage C de HMAC-SHA256.

Fichier *hmac_sha256.h*

```
#ifndef HMAC_SHA256_H
#define HMAC_SHA256_H

#include "sha256.h"

void hmac_sha256(BYTE text[], int text_len, BYTE key[], int key_len, BYTE hash[]);

#endif
```

Fichier *hmac_sha256_example.c*

```
#include <stdio.h>
#include <string.h>

#include "hmac_sha256.h"

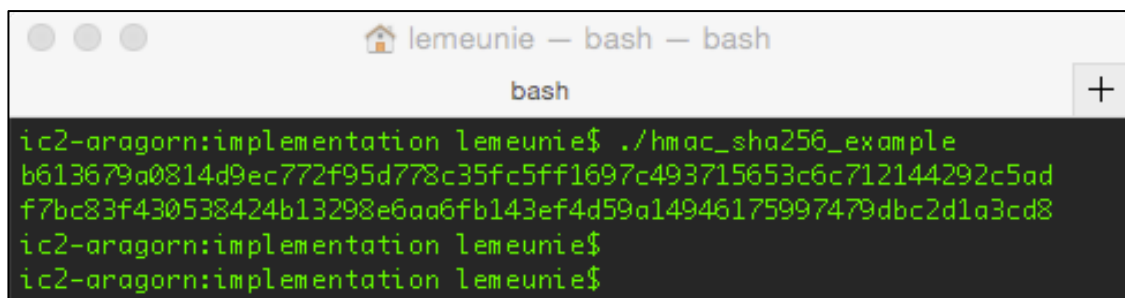
void print_hash(unsigned char hash[])
{
    int idx;
    for (idx=0; idx < 32; idx++)
        printf("%02x", hash[idx]);
    printf("\n");
}

int main()
{
    unsigned char text1[]={""},
                  text2[]={"The quick brown fox jumps over the lazy dog"},
                  hash[32];

    // HMAC one
    hmac_sha256(text1, strlen(text1), "", 0, hash);
    print_hash(hash);

    // HMAC two
    hmac_sha256(text2, strlen(text2), "key", 3, hash);
    print_hash(hash);

    return 0;
}
```



```
ic2-aragorn:implementation lemeunie$ ./hmac_sha256_example
b613679a0814d9ec772f95d778c35fc5ff1697c493715653c6c712144292c5ad
f7bc83f430538424b13298e6aa6fb143ef4d59a14946175997479dbc2d1a3cd8
ic2-aragorn:implementation lemeunie$
ic2-aragorn:implementation lemeunie$
```

Exercice n°3 : le carré de Polybe et le chiffre ADFGVX

La carré de Polybe (d'après l'excellent site web « La cryptographie expliquée » <http://www.bibmath.net/crypto/index.php>)

Polybe est un historien grec qui vécut environ de -205 avant JC jusque -125 av. JC. A 40 ans, il est emmené parmi 1000 otages par les Romains suite à la bataille de Pydna en Macédoine et à la victoire de Paul-Émile sur les Grecs. Polybe tomba en admiration devant la civilisation romaine de l'époque, et d'otage il devint même ami de la famille de Paul-Émile.

Polybe est à l'origine d'une méthode très originale pour chiffrer, et qui est même antérieure au code de César. Pour cela, il dispose les lettres dans un tableau 5*5 (nous sommes ici obligés d'identifier le i et le j) :

	1	2	3	4	5
1	A	B	C	D	E
2	F	G	H	I, J	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

On remplace alors chaque lettre par ses coordonnées dans le tableau, en écrivant d'abord la ligne, puis la colonne. Par exemple, le A est remplacé par 11, le B est remplacé par 12, le F par 21, le M par 32.... Si nous codons :

LONGTEMPS JE ME SUIS COUCHE DE BONNE HEURE

nous obtenons :

313433224415323543 2415 3215 133445132315 1415 1234333315 2315454215

Le carré de Polybe possède quelques propriétés intéressantes. En particulier, il réduit le nombre de symboles utilisés pour le codage, ce qui rend son analyse plus difficile. C'est en cela un précurseur des méthodes modernes. Remarquons que nous pouvons remplir le tableau de façon différente de ce qui est fait ici, par exemple en commençant par remplir avec un mot-clé, puis par ordre alphabétique. Ainsi, on a un vrai algorithme de chiffrement par substitution monoalphabétique, où on peut définir une clé qui permet de changer la façon dont le texte est codé.

Le chiffre ADFGVX (d'après la fiche Wikipédia)

Le chiffre ADFGVX est un système de chiffrement allemand inventé par le colonel Fritz Nebel et introduit à la fin de la Première Guerre mondiale afin de sécuriser les communications radiophoniques lors de l'offensive sur Paris. Il fut toutefois cassé par le lieutenant Georges Painvin début juin 1918, conférant un avantage crucial à l'armée française.

Son originalité réside dans l'union d'une substitution inspirée du carré de Polybe et d'une transposition. Le nom du chiffre, initialement appelé GEDEFU 18 (GEheimschrift DER FUnker 18, « chiffre des radiotélégraphistes 18 »), provient des coordonnées des lettres dans le carré. Les chiffres du carré de Polybe sont en effet remplacés par les lettres A, D, F, G, V et X, choisies en raison de leur code morse très différents les uns des autres, de façon à éviter les erreurs de transmission radio.

Le chiffre ADFGVX (d'après le site web « La cryptographie expliquée »)

Le chiffre ADFGVX est constitué d'une substitution de type carré de Polybe, suivie d'une transposition. Pour réaliser la substitution, les 26 lettres de l'alphabet et les 10 chiffres sont rangés dans un tableau 6×6, aux extrémités desquelles on a ajouté les lettres ADFGVX. Ce

rangement est désordonné et est changé chaque jour. Il constitue la première clé du chiffre ADFGVX.

	A	D	F	G	V	X
A	Q	Y	A	L	S	E
D	Z	C	R	X	H	0
F	F	O	4	M	8	7
G	3	I	T	G	U	K
V	P	D	6	2	N	V
X	1	5	J	9	W	B

Chaque lettre est codée par le couple de lettres qui correspond à sa ligne et à sa colonne. Ainsi, R est codé DF, et 3 par GA. Le message « RENFORT COMPIEGNE 16H10 » devient donc après cette première étape : DFAXV VFAFD DFGFD DFDFG VAGDA XGGVV AXXAV FDVXA DX

On choisit ensuite, pour faire la transposition, une clé qui est un mot courant, par exemple « DEMAINE ». On écrit cette clé dans un tableau, et on recopie le texte intermédiaire dans le tableau comme ci-dessous. On numérote chaque colonne suivant l'ordre alphabétique des lettres de la clé :

D	E	M	A	I	N
2	3	5	1	4	6
D	F	A	X	V	V
F	A	F	D	D	F
G	F	D	D	F	D
F	G	V	A	G	D
A	X	G	G	V	V
A	X	X	A	V	F
D	V	X	A	D	X

Le message chiffré est obtenu en lisant d'abord la colonne numérotée 1, puis la colonne numérotée 2,... On obtient donc :

XDDAG AADFG FAADF AFGXX VVDFG VVDAF DVGXX VFDDV FX

Codez le message « LE TD EST FINI » avec la même clé de substitution et avec la clé de transposition égale à « GENIAL ».