Le projet de Cryptographie est à rendre au plus tard pour mercredi 27 mars 2024.

Pour ceux qui ont déjà assisté à mes TD, je vos conseille de le démarrer dès maintenant. En effet, tous les éléments pour le réussir ont déjà été vus à ce jour, par ailleurs cela vous permettra de garder du temps pour passer vos partiels plus sereinement.

L'équipe du projet

Vous pouvez vous mettre jusqu'à quatre pour un projet.

Le document.

Le document à rendre se compose de deux fichiers :

Le document par lui-même, qui ne doit pas dépasser trois pages, rédigé de préférence dans la police Arial 11. La clarté et la concision de la rédaction seront prises en compte.

Ce document peut contenir du code en annexe, ce code devra être commenté clairement.

Ce document avec ses annexes éventuelles constitue le premier fichier, fourni au format pdf.

Les premières lignes du document doivent contenir de manière claire les noms et prénoms des étudiants du groupe, ainsi qu'un titre et le nom du groupe de TD auquel appartiennent les étudiants.

Vous nommerez le document en vous inspirant de vos noms et prénoms (par exemple de nom prénom de l'un d'entre vous) sans dépasser 10 caractères pour le nom du fichier (en dehors de l'extension .pdf).

Le deuxième fichier doit avoir le même nom que le document, à l'extension près qui sera en .enc.

Par exemple, si vous nommez le premier fichier CapHaddock.pdf (en supposant que cela corresponde au nom de l'un d'entre vous), vous nommez le deuxième fichier CapHaddock.enc.

Ce deuxième fichier correspond à une sortie du logiciel openssl pour un chffrage AES (cf. plus bas).

Envoi du document.

Vous envoyez votre document à mon mail à l'ESILV. Ce mail doit contenir en attachement les deux documents respectivement aux formats .pdf et .enc.

Les exercices

Le document répond aux trois exercices dont l'énoncé suit.

Exercice 1

Première partie

Il s'agit de chercher un texte dont le hash par la fonction sha256 vue en TD se termine par le plus de zéros possibles, une fois écrit en hexadécimal.

La chaîne de caractère doit contenir vos noms et prénoms (ainsi deux équipes ne peuvent pas trouver la même chaîne), peut contenir d'autres caractères que des lettres et des chiffres, mais doit impérativement pouvoir être collée en entrée du logiciel Python sans ambiguïté. Vous ne mettez aucun guillement, « quotes » ou apostrophes dans votre chaîne.

Vous la présentez clairement sur une ligne entre « simple quotes ». Par exemple, si vous trouvez FrancoisDuppont425_# !, vous présentez, en faisant attention aux caractères blancs entre le texte et les guillemets/quotes, que je vous conseille d'éviter :

```
'FrancoisDuppont425_#!'
```

Deuxième partie.

Vous faites une statistique. Fixez-vous par exemple n=5, et mesurez le temps moyen T_n que vous mettez pour obtenir n zéros d'affilés en fin de chaîne (dans une écriture en hexadécimal) ainsi que le temps moyen mesuré T_{n+1} pour obtenir (n+1) zéro en fin de chaîne. Vous calculez le rapport T_{n+1}/T_n et donnez le résultat.

Exercice 2

Vous prenez un texte d'environ une demi page au format .txt, sur un sujet qui vous plaît (musique, sport, art, littérature, théâtre, cinéma, …) en évitant les sujets d'actualités et en interdisant médisance et toute discrimination de quelque nature que ce soit.

Ce texte ne doit en aucun cas m'être transmis.

A l'aide d'openssl, vous le chiffrez avec un AES 256, en mode CTR, avec le mécanisme de dérivation de clé PBKDF2.

Pour cela, vous aurez besoin d'un mot de passe et d'une valeur initiale de compteur.

Dans le cadre de cet exercice, le « mot de passe » est en fait un nombre de 6 à 9 chiffres, ne comportant aucun zéro, écrit dans le système décimal. La valeur initiale du compteur (« IV ») doit aussi être un nombre de 6 à 9 chiffres, ne comportant aucun zéro, écrit dans le système décimal.

Une fois votre mot de passe et votre IV choisis, vous constituez un nombre entier N en les accolant et en mettant quatre 0 entre eux, en commençant pas le mot de passe.

Par exemple, si vous choisissez comme mot de passe 35426517 et comme IV le nombre 756431, votre entier N vaut 354265170000756431.

Votre mot de passe, votre IV ainsi que l'entier N, ne doivent, sous aucun prétexte, m'être communiqués.

Vous allez chiffrer votre texte avec l'AES256 en mode CTR, PBKDF2, votre mot de passe et votre IV et sauvegarder le fichier en lui donnant le même nom que le rendu de votre projet mais avec l'extension .enc.

Vous allez chiffrer N et obtenir deux nombres B, C qui apparaîtront clairement dans votre rapport. Vous allez employer un chiffrement ElGamal, en employant les nombres suivants ;

- Le nombre premier p : $p = 794685132467985461324582 3$
- Le « générateur » d'un groupe d'ordre élevé : $g = 5$
- Ma clé publique A : $A = 757950179598812239342298 6.$

Bien entendu, vous décrivez ce que vous avez fait.

Il faudra que je puisse déchiffrer votre texte en recevant votre message et en employant vos nombres B et C.

Exercice 3.

Première partie.

Vous présentez dans cette partie une démonstration de la signature ElGamal. La qualité de la rédaction et de la présentation seront prises en compte. Pour ceux qui n'ont pas pu assister à mes TD depuis le début, vous pouvez choisir de me présenter une démonstration de la signature RSA. Les démonstrations doivent employer des nombres différents de ceux présentés en cours. Les exemples doivent être différents d'un groupe à l'autre. De plus, vous devez vous contenter des utilitaires (librairies / packages) permettant de faire des calculs numériques et ne pas employer des utilitaires où tout est fait à votre place !

Deuxième partie.

Un exemple d'emploi du RSA, de l'échange de clé Diffie-Hellman, du chiffrement ElGamal, ou de la signature ElGamal au sein de Python ou openssl dans la vie réelle. Vous ne choisissez qu'un seul exemple, en évitant que deux équipes fassent le même choix. Vous vous mettez à la place d'une société qui décrit un logiciel et son utilisation à un client ou à un ingénieur chez son client. Essayez de ne pas dépasser une demi-page pour cette présentation.