Cryptographie :

Application à la **T**ransport **L**ayer **S**ecurity (TLS)

Modalités

- Travail individuel en autonomie
- 1 jour en présentiel

Objectifs

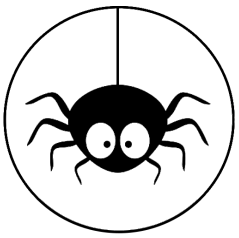
L'objectif de cette journée est de découvrir des bases de la cryptographie et d'en découvrir les fondements mathématiques. La journée sera consacrée à l'étude de la négociation TLS, nous en élaborerons ensemble une version simplifiée, ce qui mettra en lumière deux aspects fondamentaux de la cryptographie : le chiffrement symétrique et asymétrique. Nous aborderons brièvement un autre aspect fondamental : le hashing et ses applications.

Compétences

- Chiffrer/ Déchiffrer un message par la méthode de Vigenère.
- Générer une clé de session avec la méthode de Diffie-Hellman.

CryptoMaster

Tout au long de cette journée vous pourrez interagir avec CryptoMaster, votre compagnon qui essaie désespérément d'établir une connexion TLS avec vous.



CryptoMaster

Crypto Master c'est lui, il sera disponible sur Discord toute la journée et vous répondra inlassablement. Pour communiquer avec lui, rien de plus simple : des routes seront disponibles via **`$crypto`** sur Discord. Les arguments à donner vous seront indiqués à chaque activité.

Vous pouvez essayer dès à présent :

```
$crypto - sayHello
```

Une deuxième route qui pourra vous être utils :

```
$crypto - infos
```

Activité 1 - Familiarisation avec la négociation TLS (45min à 1h max)

Avant de nous lancer tête baissée dans l'étude plus précise des différentes étapes de la négociation TLS je vous encourage à aller étudier différentes ressources pour comprendre à quoi ressemble le paysage.

Nous pouvons prendre un temps à la fin de cette étude pour en discuter en groupe.

Ressources :

- Computerphile :

1. https://www.youtube.com/watch?v=0TLDTodL7Lc
2. https://www.youtube.com/watch?v=86cQJ0MMses

- Geekflare : https://geekflare.com/tls-101/
- Cloudflare : https://www.cloudflare.com/fr-fr/learning/ssl/what-happens-in-a-tls-handshake/
- TLS byte by byte : https://tls.ulfheim.net/

Activité 2 - Communication avec CryptoMaster via chiffrement **symétrique**

La cryptographie symétrique, également dite à clé secrète (par opposition à la cryptographie asymétrique), est la plus ancienne forme de chiffrement. Elle permet à la fois de chiffrer et de déchiffrer des messages à l'aide d'un même mot clé. On a des traces de son utilisation par les Égyptiens vers 2000 av. J.-C.

L'un des concepts fondamentaux de la cryptographie symétrique est la *clé*. Une clé est une donnée qui (traitée par un algorithme) permet de chiffrer et de déchiffrer un message. Quiconque découvre la clé peut déchiffrer le message sans autre information. Une fois l'algorithme découvert, tous les messages chiffrés par lui deviennent lisibles.

Extraits légèrement modifiés de
Wikipédia : Cryptographie symétrique

Nous allons maintenant essayer de communiquer avec le bot discord de manière sécurisée en utilisant deux techniques de chiffrement symétrique : le chiffre de César et le chiffre de Vigenère.

2.1 — Chiffre de César (substitution monoalphabétique)

20min — Présentiel

CONSIGNES

Lisez la ressource Wikipédia et essayez de comprendre l'idée générale (c'est plutôt simple!). Chiffrez un message quelconque en utilisant une clé de votre choix. Vous pourrez interagir avec le bot discord pour vérifier vos résultats en l'interpellant et en lui passant deux paramètres : clé et message.

VOCABULAIRE DE CRYPTOMASTER

Tout au long de cette journée, CryptoMaster n'aura qu'un vocabulaire très limité. Il connaît les lettres de l'alphabet (uniquement en minuscule), ainsi qu'une ponctuation relativement simple : “ **!** ” - “ **,** ” - “ **.** ” - “ **:** ” - “ **?** ” - “ ” `(caractère espace)`

OPTIONNEL

Coder un algorithme de chiffrement/ déchiffrement en JavaScript.

RESSOURCES

- https://fr.wikipedia.org/wiki/Chiffrement_par_d%C3%A9calage

CRYPTOMASTER

Demande à CryptoMaster de chiffrer un message

```
$crypto - encodeCesar - <message> - <clé>
```

Demande à CryptoMaster de déchiffrer un message

```
$crypto - decodeCesar - <message> - <clé>
```

2.2 — Chiffre de Vigenère (substitution polyalphabétique)

45min — Présentiel

CONSIGNES

Lisez la ressource Wikipédia et essayez de comprendre le principe de ce chiffrage.
Chiffrez un message quelconque en utilisant une clé de votre choix. Faîtes ce calcul à la main pour bien comprendre ce qu'il se passe. Aidez-vous de la "table de Vigenère" ci-après pour coder et décoder vos messages.

Vous pourrez interagir avec le bot discord pour vérifier vos résultats en l'interpellant et en lui passant deux paramètres : la clé et le message.

Lettre en clair

Lettre de la clé

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?	
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A
B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B
C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D
E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E
F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F
G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G
H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H
I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I
J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J
K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K
L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L
M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N
O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
R	S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
S	T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
T	U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
U	V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
V	W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
W	X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
X	Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Y	Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y
Z	!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
!	,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!
,	.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,
.	:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.
:	?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:
?		A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	!	,	.	:	?	

Table de Vigenère

OPTIONNEL

Coder un algorithme de chiffrement/ déchiffrement en JavaScript.

RESSOURCES

- https://fr.wikipedia.org/wiki/Chiffre_de_Vigen%C3%A8re

CRYPTOMASTER

Demande à CryptoMaster de chiffrer un message

```
$crypto - encodeVigenere - <message> - <clé>
```

Demande à CryptoMaster de déchiffrer un message

```
$crypto - decodeVigenere - <message> - <clé>
```

Activité 3 - Communication avec CryptoMaster via chiffrement **asymétrique**

Par opposition à la cryptographie symétrique, la cryptographie asymétrique est un domaine très récent (1976). En cryptographie asymétrique il existe une distinction entre données publiques et privées, en opposition à la cryptographie symétrique où la fonctionnalité est atteinte par la possession d'une donnée secrète commune entre les différents participants.

Extraits légèrement modifiés de
Wikipédia : Cryptographie symétrique

2.1 — Temps d'échange

CONSIGNES

Discussions en groupe autour des principes, limites et applications des techniques de chiffrement symétrique et asymétrique, de la notion de signature numérique et de certificat.

2.2 — Diffie-Hellman Key Exchange et communication.

CONSIGNES

- Lisez la ressource Wikipédia et regardez la vidéo de Computerphile pour comprendre l'idée générale de l'échange de clé par Diffie-Hellman (idée ayant valu un prix Turing en 2015!).
- Essayez de vous échanger une clé secrète avec un binôme en utilisant des post-its. Le 1er partenaire choisira des paramètres publics de Diffie-Hellman. Essayez de construire une clé en commun, et vérifiez que vous avez bien la bonne !
- Engagez un échange de clé avec CryptoMaster :
    - Une première route vous permettra d'envoyer les paramètres publics, ainsi que la première étape intermédiaire de calcul.
    - CryptoMaster vous renverra un nombre en échange et calculera la clé commune.
    - Vous calculerez vous aussi la clé de session.
    - Vous pourrez demander à CryptoMaster de publier la clé commune secrète pour vérifier vos résultats.

- Utilisez la clé de chiffrement commune de Diffie-Hellman pour chiffrer vos messages en utilisant le chiffrage de Vigenère.

Remarque :

Le nombre secret partagé vous donne les décalages dont vous devez vous servir pour chiffrer avec la méthode de Vigenère.

Exemple :

Si le nombre secret partagé est **132** alors :
- Message non chiffré : `aaaaaaa`
- Message chiffré : `bdcbdcb`

RESSOURCES

- https://fr.wikipedia.org/wiki/%C3%89change_de_cl%C3%A9s_Diffie-Hellman

CRYPTOMASTER

Envoi des paramètres publics Diffie-Hellman et du nombre intermédiaire :

```
$crypto - DH_clientHello - <generator> - <modulus> - <nombre_intermediaire>
```

Demande à CryptoMaster de déchiffrer un message

```
$crypto - DH_decodeMsg - <message>
```

Récupération des informations sur vous :

```
$crypto - infos
```

OPTIONNEL

Coder un algorithme de calcul de la clé secrète commune.

Félicitations !

Vous êtes maintenant capables de communiquer de manière sécurisée avec CryptoMaster.

Mais…

Etait-ce bien CryptoMaster ?

Ou …

Un usurpateur ?

Activité 4 - Signature numérique

Les méthodes de chiffrement que nous avons mises en place jusqu'à présent sont vulnérables aux attaques dites de "L'homme du milieu" (Man in the middle attack).

Prenons l'exemple de l'échange de clé par Diffie-Hellman entre Alice et Bob (noms classiques en cryptologie : https://fr.wikipedia.org/wiki/Alice_et_Bob). Mallory souhaite attaquer la communication via une attaque de type Man in the Middle :

- Mallory va alors intercepter la clé publique d'Alice et envoyer la sienne à Bob (qui croit recevoir celle d'Alice)
- Mallory fait de même avec la clé publique envoyée par Bob à destination d'Alice (Alice croit alors recevoir celle de Bob)
- Alice et Mallory se mettent alors d'accord sur une clé commune de chiffrement.
- Bob et Mallory se mettent alors d'accord sur une clé commune de chiffrement.
- Après ces échanges Mallory peut alors intercepter les messages échangés entre Alice et Bob, les déchiffrer, les lire, éventuellement les modifier, pour finalement les chiffrer et les transmettre au destinataire, qui ne se doute de rien…

Afin de rétablir la confiance et s'assurer que le serveur avec lequel nous communiquons est bien celui qu'il prétend être, nous utiliserons le concept de signature numérique.

Comment créer une signature numérique ?

Nous devons une fois encore faire appel au concept de chiffrement asymétrique. La méthode de chiffrement que nous utiliserons cette fois-ci sera le chiffrement RSA.

2.2 — Chiffrement RSA

Le chiffrement RSA repose sur l'utilisation d'une paire de clés : clé publique et clé privée. L'idée générale est que le chiffrement est possible avec n'importe laquelle des deux clés, mais le déchiffrement nécessite toujours la clé complémentaire (chiffrement avec clé privée/ déchiffrement avec clé publique, et inversement).

Ce concept nous permet d'échanger de l'information de manière sécurisée (nous aurions pu utiliser le chiffrement RSA en lieu et place de Diffie-Hellman pour l'échange d'une clé de session) mais aussi de garantir l'identité de l'auteur d'un message.

Idée (simplifiée) : Imaginons que Bob veuille prouver à Alice qu'il est bien l'auteur d'un message (ce message n'est pas nécessairement chiffré, seule l'identité de l'émetteur nous intéresse). Il suffit à Bob d'envoyer deux messages à Alice :

1. La clé publique de Bob
2. Le message non chiffré que Bob veut transmettre à Alice.
3. Le même message chiffré avec la clé privée de Bob.
4. Alice pourra alors déchiffrer le message en utilisant la clé publique de Bob.
5. Si Alice est capable de déchiffrer le message chiffré en utilisant la clé publique de Bob, cela signifie donc que ce message a été chiffré en utilisant la clé privée de Bob, qu'il est le seul à connaître.
6. C'est donc bien Bob l'auteur du message !

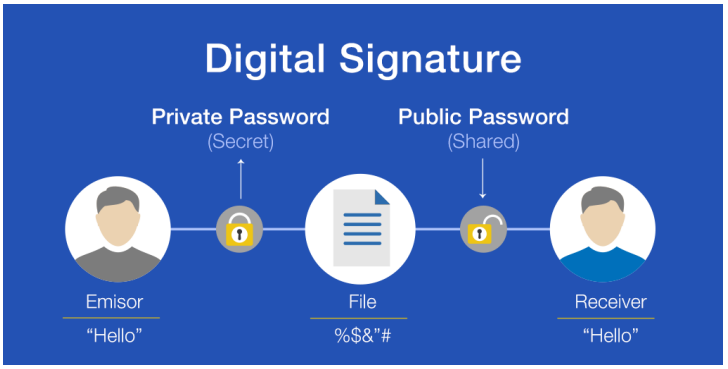


Illustration très simplifiée du mécanisme de signature électronique

DISCLAIMER

Soyez conscients que le processus de signature électronique tel que montré dans ce document a vocation pédagogique. Certaines briques sont manquantes pour pouvoir obtenir un système applicable en conditions réelles. Voici deux éléments manquants, et pourtant fondamentaux de ce système (nous pourrons en parler en groupe si le temps nous le permet !)

Fonction de hashing cryptographique :

- https://fr.wikipedia.org/wiki/Fonction_de_hachage_cryptographique

Public key infrastructure :

- https://fr.wikipedia.org/wiki/Infrastructure_%C3%A0_cl%C3%A9s_publiques

CONSIGNES

- Lisez les ressources Wikipédia et Brilliant.
- Demandez à CryptoMaster de vous prouver son identité. Il vous enverra un message, ainsi que le même message chiffré avec sa clé privée. Vous devriez être capables de d'établir (ou non) l'identité de l'émetteur.
- Essayez maintenant de vous transmettre des messages en binôme. Pour cela vous devrez chacun générer une paire de clé publique/ privée et les utiliser pour vous transmettre des messages (https://fr.wikipedia.org/wiki/Chiffrement_RSA#Engendrer_les_clefs).

CRYPTOMASTER

- $crypto - proveIdentity

RESSOURCES

- https://fr.wikipedia.org/wiki/Chiffrement_RSA
- https://brilliant.org/wiki/rsa-encryption/

Pour aller plus loin

Nous avons été gratter la surface concernant la cryptographie et ses applications dans le cadre du Web. Si vous souhaitez aller plus loin aujourd'hui, plusieurs options s'offrent à vous :

- Commencez par implémenter toutes les fonctions et méthodes de chiffrage vues jusqu'à maintenant en JavaScript.
- Vous pouvez aller vous renseigner sur la notion de **hashage cryptographique** si vous ne l'avez pas encore fait.
    - https://fr.wikipedia.org/wiki/Fonction_de_hachage_cryptographique
    - Je peux vous donner des pistes de choses à faire dans cette direction. C'est un concept très important à comprendre.
- Vous pouvez coder un bot Discord qui interagit avec CryptoMaster.
    - https://www.digitalocean.com/community/tutorials/how-to-build-a-discord-bot-with-node-js-fr
- Vous pouvez aider les autres !