La Cryptographie avec OpenSSL

C'est quoi la cryptographie ?

Imagine que tu as **un journal intime** et tu veux que personne ne puisse le lire. La cryptographie, c'est comme **mettre un cadenas secret** sur tes messages !

Les types de cadenas :

1. **Chiffrement symétrique** = Une seule clé (comme une clé de maison)
   - Tu utilises la MÊME clé pour fermer et ouvrir
2. **Chiffrement asymétrique** = Deux clés (comme une boîte aux lettres)
   - Une clé PUBLIQUE (tout le monde peut l'utiliser pour t'envoyer des messages)
   - Une clé PRIVÉE (seulement TOI peux ouvrir et lire)

Préparation

 **OpenSSL :** Ta boîte à outils magique !

Imagine que tu as une boîte à **outils de super-héros** avec plein de gadgets pour protéger tes secrets :

OpenSSL, c'est comme cette **boîte magique** qui contient :

🔐 Des **cadenas de toutes les tailles** pour fermer tes messages secrets

🔑 Des **machines à fabriquer des clés** (des vraies clés magiques !)

✍️ Un **stylo spécial** pour signer tes documents (comme une **signature magique** qu'on ne peut pas copier)

🪪 Une **machine à faire des cartes d'identité** pour prouver qui tu es sur Internet

En gros : OpenSSL est **un programme gratuit** installé sur ton ordinateur qui te permet de faire de la magie avec tes fichiers pour les protéger !

Avant de commencer, On vérifie que OpenSSL est installé :

```
hanen@hanen-VMware-Virtual-Platform:~$ openssl version
OpenSSL 3.0.13 30 Jan 2024 (Library: OpenSSL 3.0.13 30 Jan 2024)
```

Ensuite on crée un dossier pour notre travail :

```
hanen@hanen-VMware-Virtual-Platform:~$ mkdir tp_crypto
hanen@hanen-VMware-Virtual-Platform:~$ cd tp_crypto
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$
```

PARTIE 1 : Chiffrement Symétrique (La clé unique)

Exercice 1.1 : AES - Le cadenas moderne

Étape 1 : Créer un fichier secret

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ echo "Ceci est mon message super secret !"
> fichier.txt
```

Étape 2 : Chiffrer avec AES

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ openssl enc -aes-256-cbc -salt -pbkdf2 -in
fichier.txt -out fichier_chiffre.enc
enter AES-256-CBC encryption password:
Verifying - enter AES-256-CBC encryption password:
```

Étape 3 : Regarder le fichier chiffré

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ cat fichier_chiffre.enc
Salted__⋄A⋄_⋄⋄[8⋄⋄⋄ly⋄⋄}b]E⋄⋄7⋄P⋄⋄8⋄B_⋄⋄i~⋄⋄iM<⋄⋄⋄⋄⋄⋄⋄hhahanhanhahhhhhah
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$
```

→ **Il est illisible !**

Étape 4 : Déchiffrer pour retrouver mon message

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ openssl enc -d -aes-256-cbc -pbkdf2 -in fichier_chiffre.enc -out
fichier_dechiffre.txt
enter AES-256-CBC decryption password:
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$
```

Étape 5 : Vérifie que ça marche

```
enter AES-256-CBC decryption password:
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ cat fichier_dechiffre.txt
Ceci est mon message super secret !
```

Exercice 1.2 : DES - L'ancien cadenas

Étape 1 : Créer un autre fichier

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ echo "Mon deuxième secret" > secret.txt
```

Étape 2 : Chiffrer avec DES

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ openssl enc -des3 -salt -pbkdf2 -in secret.txt -out secret.des3
enter DES-EDE3-CBC encryption password:
Verifying - enter DES-EDE3-CBC encryption password:
```

Étape 3 : Déchiffrer

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ openssl enc -d -des3 -pbkdf2 -in secret.des3 -out secret_dechiffr
e.txt
enter DES-EDE3-CBC decryption password:
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$
```

Étape 4 : Vérifier

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ cat secret_dechiffre.txt
Mon deuxième secret
```

📝 Question de réflexion

Question 1 : Lequel est le plus utilisé, AES ou DES ?

- **Réponse** : **AES** est plus utilisé aujourd'hui car il est PLUS SÉCURISÉ et plus rapide

Question 2 : Pourquoi on ajoute -salt ?

- **Réponse** : Le "sel" (salt) c'est comme ajouter des épices différentes à chaque fois. Même si tu chiffres le même message avec le même mot de passe, le résultat sera DIFFÉRENT à chaque fois. Ça protège mieux !

PARTIE 2 : Chiffrement Asymétrique (Les deux clés)

🔑 Exercice 2.1 : Créer mes clés RSA

Étape 1 : Générer ma clé PRIVÉE (à garder secrète !)

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ openssl genpkey -algorithm RSA -out private_key.pem -pkeyopt rsa_
keygen_bits:2048
.....+...+.....+.+......+.....................+.......+...+.........+.....+....+.....+.++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++++++*.+...+.+.....+.+..+..............+.+..+...+..............+..............
......+...+.............++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*...+....+...+..+.....+
.............+.....+......+.+.....+............+.........+...+..............+...++++++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++
..........+...+...+..+......+.+.....+.+.........+.....++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++*..++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*........+......+.........+.........+....
....+...+......+....+.....+......+.+.........+.+...+.........+.......+...+...+..+...+.......+.............+........
.+.+......+.....+............+.......+......+.....+.........+.+.....+.......+...+............+......+..+....+.....
......+...+....+.....+.......+...+.....+..............+..+...+............+....+.....+.+........+....+..+...+...
.............+......+.....+.........+......+...+......+.....+...+...+....+...+...+..+...+.........+......+...+...
....+..............+............+...+..+....+...+..+...+.+.............+..........+..........+......+......+
.......+........+.............+......+.........+..+.......+...+.........+.........+.....+....+..+......+.......+..
+....++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

Étape 2 : Créer ma clé PUBLIQUE (à partager)

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ openssl rsa -pubout -in private_key.pem -out public_key.pem
writing RSA key
```

Étape 3 : Regarder mes clés

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ cat private_key.pem
-----BEGIN PRIVATE KEY-----
MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQC+JUuTkPphz+OC
MtaYsl+aLHyxYZaqap4g/nykXFZMG4XXvGydBmn6macxqT82bCtO8GU9eRg7vupF
ioWY8LtSKR0MJXb34KKyub/AcTvPYcu4iPkUBQBb5K95EWYNHxI2zPN3pommvheI
nzbYZBeGtlFcyV7KvUiVczN2eWVRnPRKvWnFjYH1I7NO02C78rfrUdoLX23gTILS
wPSDrq98QlKol35oUxw6nMxaUfAFJfCEDx2VoQy+pDxPuLpw+smuCSMnK2daAh3D
BKLAHblQgQH3XxsGthHTR2Jh3fdoZPEeQrOMCbthAWhZou8Mt14Kn9HwIRD6mgVR
hy9uyIB3AgMBAAECggEAMHWz1beAwHXpjzCitYbcS6Aj5VcwPV1v0IvbO05XJg1r
```

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ cat public_key.pem
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAviVLk5D6Yc/jgjLWmLJf
mix8sWGWqmqeIP58pFxWTBuF17xsnQZp+pmnMak/NmwrTvBlPXkYO77qRYqFmPC7
UikdDCV29+Cisrm/wHE7z2HLuIj5FAUAW+SveRFmDR8SNszzd6aJpr4XiJ822GQX
hrZRXMleyr1IlXMzdnllUZz0Sr1pxY2B9SOzTtNgu/K361HaC19t4EyC0sD0g66v
fEJSqJd+aFMcOpzMWlHwBSXwhA8dlaEMvqQ8T7i6cPrJrgkjJytnWgIdwwSiwB25
UIEB918bBrYR00diYd33aGTxHkKzjAm7YQFoWaLvDLdeCp/R8CEQ+poFUYcvbsiA
dwIDAQAB
-----END PUBLIC KEY-----
```

🔓 Exercice 2.2 : Chiffrer un message

Étape 1 : Créer un message

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ echo "Message top secret pour RSA" > message.txt
```

Étape 2 : Chiffrer avec la clé PUBLIQUE

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ openssl pkeyutl -encrypt -pubin -inkey public_key.pem -in message
.txt -out message_encrypted.bin
```

Étape 3 : Essaye de lire le message

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ cat message_encrypted.bin
e;^◆◆◆S9y?◆◆◆◆◆◆◆X◆* ◆Ç):[@W0c◆Yb◆Þx◆
◆X◆v^p◆◆;◆◆gu↨f◆R◆◆Ŷ◆^◆h◆◆◆◆◆◆X◆◆◆x◆ñ◆iu
                                    U◆]◆◆K◆i◆W◆◆◆K◆◆◆◆◆◆◆◆G"]&◆◆+◆x◆◆p◆◆_2'◆I◆d◆eP;i◆=◆◆a◆2
                                                                                    ◆◆◆>◆◆◆◆>P◆v"◆◆◆◆H
◆E◆◆◆◆◆◆my◆OI2C◆◆Pu!◆◆~
                        ,w◆
```

⇨ **Impossible**

🔓 Exercice 2.3 : Déchiffrer le message

Étape 1 : Déchiffrer avec la clé PRIVÉE

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ openssl pkeyutl -decrypt -inkey private_key.pem -in message_encry
pted.bin -out message_decrypted.txted.txt
```

Étape 2 : Lire le message déchiffré

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ cat message_decrypted.txt
Message top secret pour RSA
```

✍️ Exercice 2.4 : Signature numérique

Étape 1 : Créer un document à signer

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ echo "Contrat important à signer" > document.txt
```

Étape 2 : Signer le document

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ openssl dgst -sha256 -sign private_key.pem -out signature.bin doc
ument.txt
```

Étape 3 : Vérifier la signature

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ openssl dgst -sha256 -verify public_key.pem -signature signature.
bin document.txt
Verified OK
```

📜 Exercice 2.5 : Créer un certificat

Étape 1 : Générer une clé pour le certificat

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ openssl genrsa -out cert_key.pem 2048
```

Étape 2 : Créer une demande de certificat

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ openssl req -new -key cert_key.pem -out cert_req.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:TN
State or Province Name (full name) [Some-State]:Monastir
Locality Name (eg, city) []: jemmel
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Esprim
Organizational Unit Name (eg, section) []:Arab
Common Name (e.g. server FQDN or YOUR name) []:hanen
Email Address []:Hanene.BENMANAA@esprim.tn
```

Étape 3 : Créer le certificat autosigné

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ openssl x509 -req -days 365 -in cert_req.csr -signkey cert_key.pe
m -out mycert.pem
Certificate request self-signature ok
subject=C = TN, ST = Monastir, L = " jemmel", O = Esprim, OU = Arab, CN = hanen, emailAddress = Hanene.BENMANAA@es
prim.tn
```

Étape 4 : Voir mon certificat

```
hanen@hanen-VMware-Virtual-Platform:~/tp_crypto$ openssl x509 -in mycert.pem -text -noout
Certificate:
    Data:
        Version: 1 (0x0)
        Serial Number:
            3b:54:3f:33:63:2d:57:03:8d:cb:30:27:9d:85:79:c6:cc:3a:99:84
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C = TN, ST = Monastir, L = " jemmel", O = Esprim, OU = Arab, CN = hanen, emailAddress = Hanene.BEN
MANAA@esprim.tn
        Validity
```