

# Projet Sécurité 2ème Partie

Nour Charfeddine  
Nader Ben Ammar  
Mohamed Arbi Ghanmi





# Plan

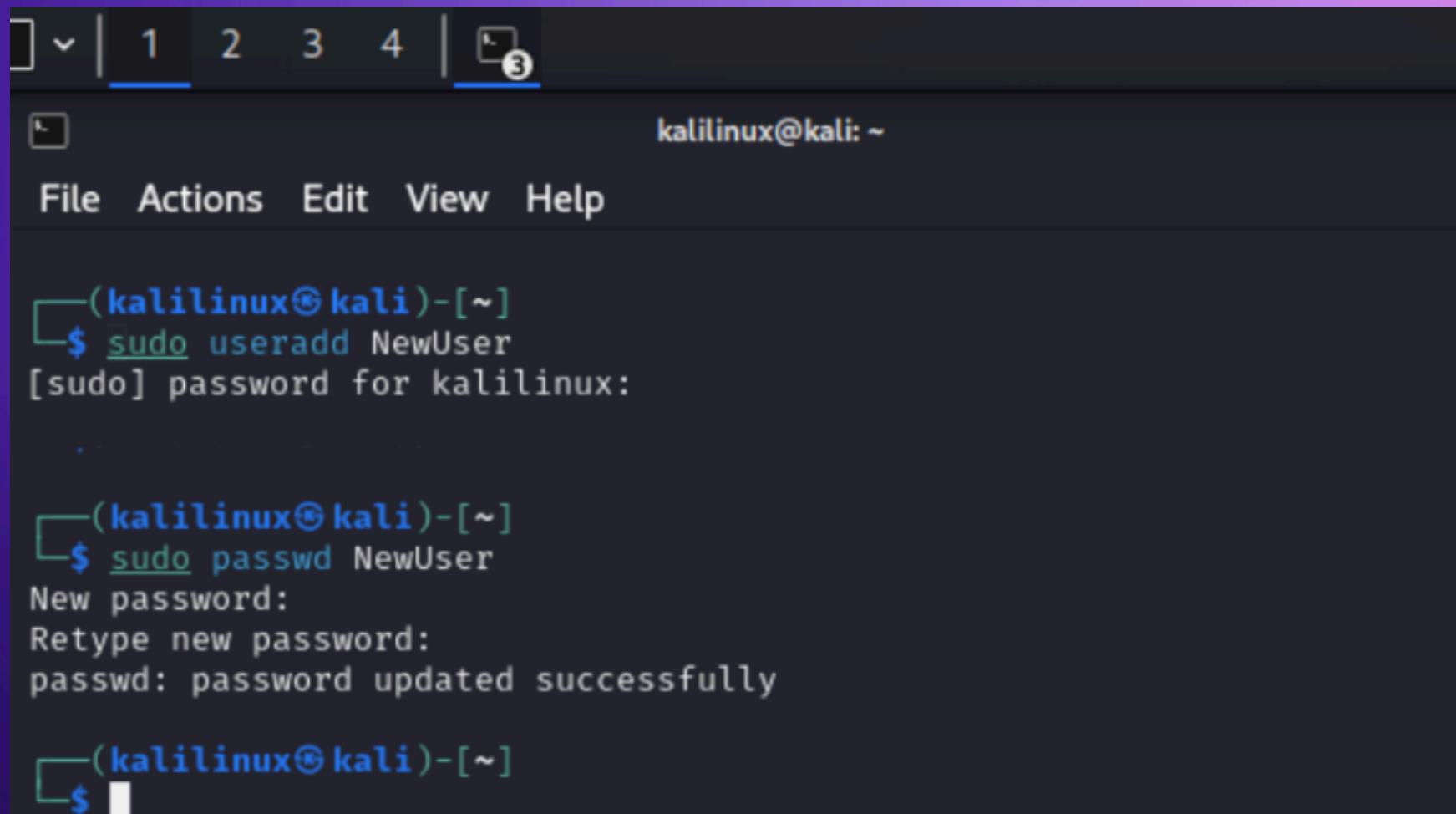
- 01/ Partie C: Cracking passwords
- 02/ Partie D: cryptographie+

# Partie C: Cracking passwords



## Partie C: Cracking passwords

# 1. Ajouter un nouvel utilisateur sur la plateforme utilisée



The screenshot shows a terminal window with a dark background and light-colored text. At the top, there's a navigation bar with tabs labeled 1, 2, 3, 4, and a file icon. The title bar says "kalilinux@kali: ~". Below the title bar is a menu bar with File, Actions, Edit, View, and Help. The main area of the terminal shows the following command sequence:

```
(kalilinux㉿kali)-[~]
$ sudo useradd NewUser
[sudo] password for kalilinux:
.

(kalilinux㉿kali)-[~]
$ sudo passwd NewUser
New password:
Retype new password:
passwd: password updated successfully

(kalilinux㉿kali)-[~]
$
```

On peut générer un nouvel utilisateur en ligne de commande en lançant la commande `useradd`, suivie de l'utilisation de `passwd` pour définir le mot de passe associé à cet utilisateur.

## Partie C: Cracking\_passwords



### 2. Identifier le fichier de mot de passe sur cette plateforme

Pour visualiser le contenu du fichier /etc/shadow, il est nécessaire d'être un utilisateur root ou d'avoir les autorisations d'accès adéquates. Pour cela, on peut recourir à la commande sudo pour afficher le contenu du fichier /etc/shadow.

```
File Actions Edit View Help

(kalilinux㉿kali)-[~]
$ sudo grep NewUser /etc/shadow
[sudo] password for kalilinux:
Sorry, try again.
[sudo] password for kalilinux:
NewUser:$y$j9T$0YYScY4JrRej7VenK2wuQ/$FkTch318HlWjk8Q1tWTPJICn.IU8XeotrvNeCn
keJ7:19839:0:99999:7 :::

(kalilinux㉿kali)-[~]
$
```

## Partie C: Cracking passwords

### 3. Utiliser un outil de crack pour divulguer le mot de passe de l'utilisateur ajouté.

Pour tenter de déchiffrer le mot de passe crypté, on commence par installer l'outil John the Ripper.

Ensuite, on peut utiliser la commande john pour essayer de casser le mot de passe.

```
(kalilinux㉿kali)-[~]
└─$ sudo apt-get install john
[sudo] password for kalilinux:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
john is already the newest version (1.9.0-Jumbo-1+git20211102-0kali7).
john set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 434 not upgraded.

(kalilinux㉿kali)-[~]
└─$ sudo unshadow /etc/passwd /etc/shadow > ~/combined.txt
Created directory: /root/.john

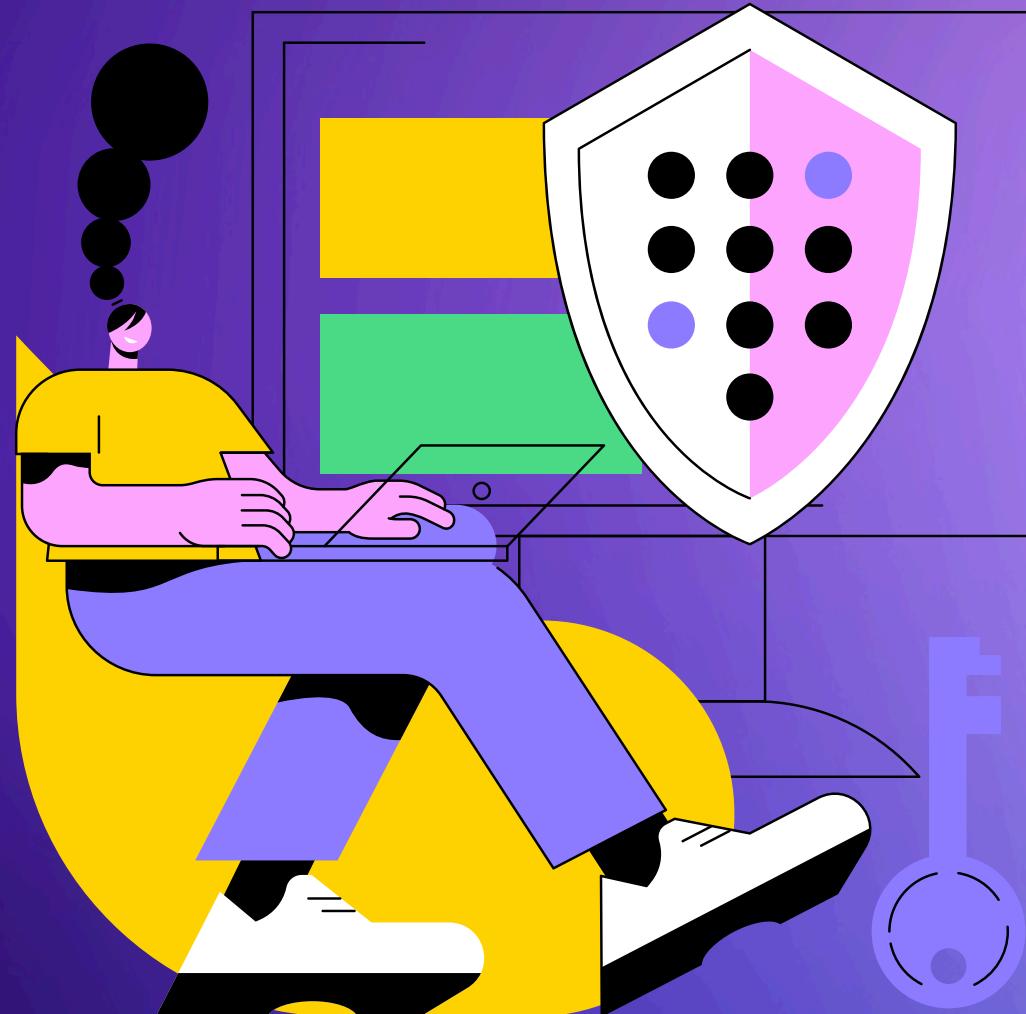
(kalilinux㉿kali)-[~]
└─$ john ~/combined.txt
Created directory: /home/kalilinux/.john
Using default input encoding: UTF-8
Loaded 1 password hash (HMAC-SHA224 [password is key, SHA224 128/128 SSE2 4x])
)
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 5 candidates buffered for the current salt, minimum 8 needed for performance.
```

```
(kalilinux㉿kali)-[~]
└─$ sudo unshadow /etc/passwd /etc/shadow > ~/combined.txt
[sudo] password for kalilinux:

(kalilinux㉿kali)-[~]
└─$ john ~/combined.txt
Using default input encoding: UTF-8
Loaded 1 password hash (HMAC-SHA224 [password is key, SHA224 128/128 SSE2 4x])
)
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Crash recovery file is locked: /home/kalilinux/.john/john.rec
```

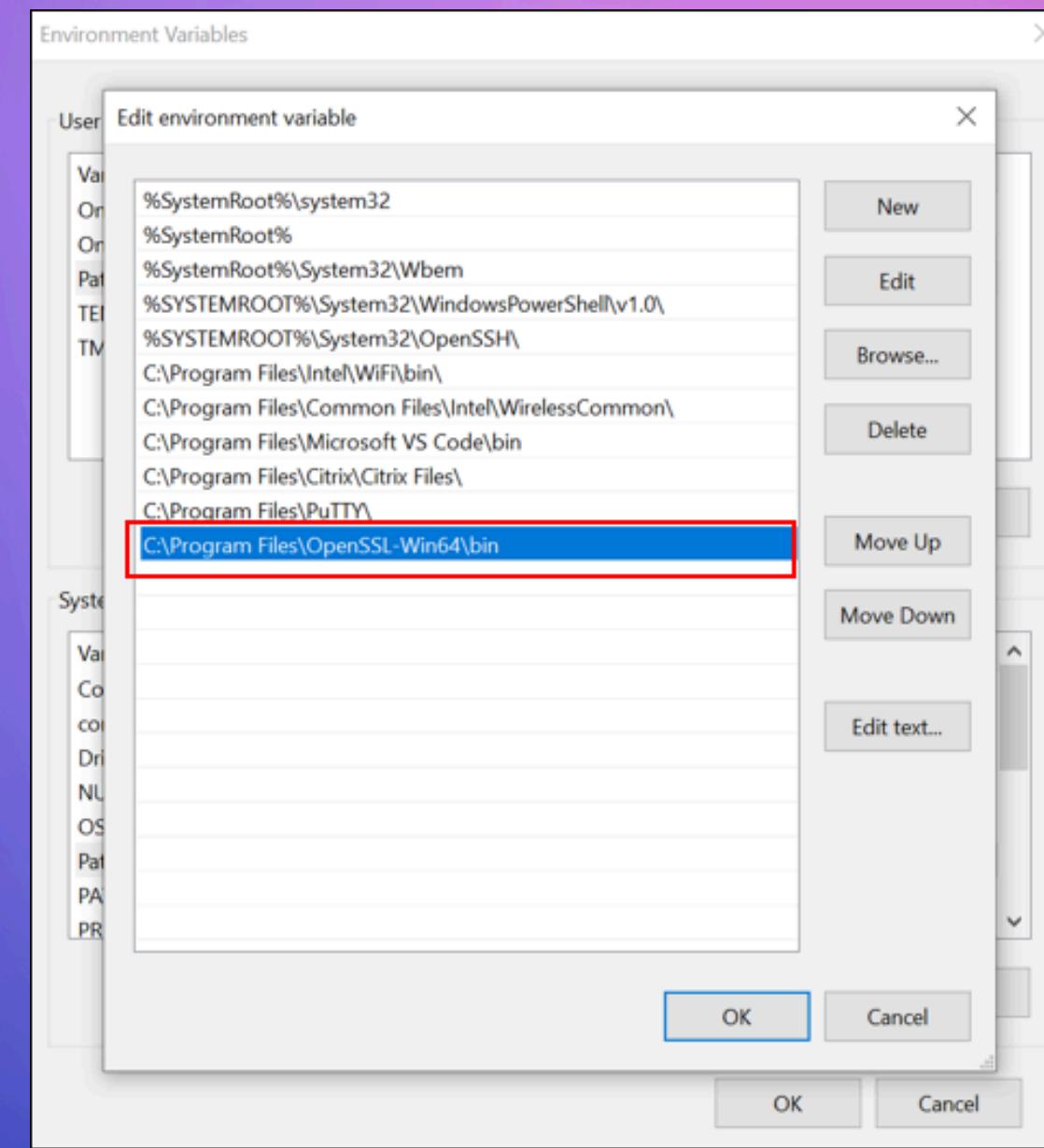
---

# Partie D: cryptographie+

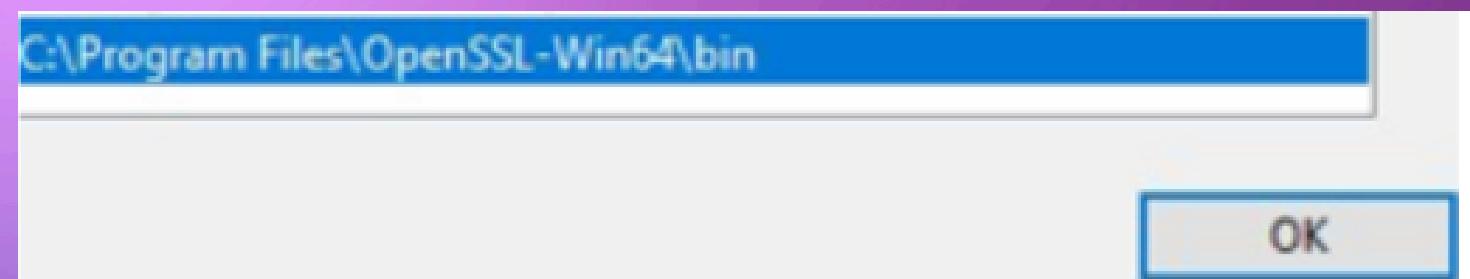


## Partie D: cryptographie+

- On commence par télécharger OpenSS.
- La configuration sera réalisée en premier, suivie de l'exécution via l'invite de commande cmd.
- OpenSSL a été inclus dans les variables d'environnement afin de le rendre accessible pour une exécution depuis n'importe où



- OpenSSL a été inclus dans les variables d'environnement afin de le rendre accessible pour une exécution depuis n'importe où.





# 1/ Cryptage Symétrique :

- Cette commande détaille notre intention d'utiliser un cryptage symétrique avec les spécifications suivantes :
- '-in "D:\Downloads\textb.txt" ' : Précise le chemin du fichier d'entrée à crypter.
- '-out b-des.txt' : Indique le chemin et le nom du fichier de sortie qui recevra le résultat du cryptage.
- '-enc' : Instruit OpenSSL à utiliser les fonctionnalités de cryptage.
- '-des-cbc' : Sélectionne l'algorithme de chiffrement symétrique DES en mode CBC (Cipher Block Chaining).
- '-a' : Sollicite OpenSSL pour effectuer un encodage Base64 du fichier de sortie, le transformant en un format texte.
- '-salt' : Utilise un sel aléatoire pour renforcer la sécurité du chiffrement.

Cette commande détaille notre intention d'utiliser un cryptage symétrique avec les spécifications suivantes :

**'-in "D:\Downloads\textb.txt'** : Précise le chemin du fichier d'entrée à crypter.

**'-out b-des.txt'** : Indique le chemin et le nom du fichier de sortie qui recevra le résultat du cryptage.

**'-enc'** : Instruit OpenSSL à utiliser les fonctionnalités de cryptage.

**'-des-cbc'** : Sélectionne l'algorithme de chiffrement symétrique DES en mode CBC (Cipher Block Chaining).

**'-a'** : Sollicite OpenSSL pour effectuer un encodage Base64 du fichier de sortie, le transformant en un format texte.

**'-salt'** : Utilise un sel aléatoire pour renforcer la sécurité du chiffrement.

Fichier	Modifier	Affichage
		U2FsdGVkX19fvIR5Cir2DS006nY6mnFRwKrdKe3towMAHGAB1hF306yA+5Q4vt3



# 2/ Cryptage asymétrique :

- "2048" : La longueur de la clé en bits. Dans cet exemple, une clé de 2048 bits est générée. D'autres valeurs telles que 1024, 3072 ou 4096 peuvent être choisies en fonction des exigences de sécurité.
- "-out key.pem" : Spécifie que la clé privée sera sauvegardée dans un fichier nommé key.pem. Le nom du fichier de sortie peut être personnalisé selon tes préférences.
- "genrsa" : Cette commande est utilisée pour générer une clé RSA.

```
C:\> Invite de commandes

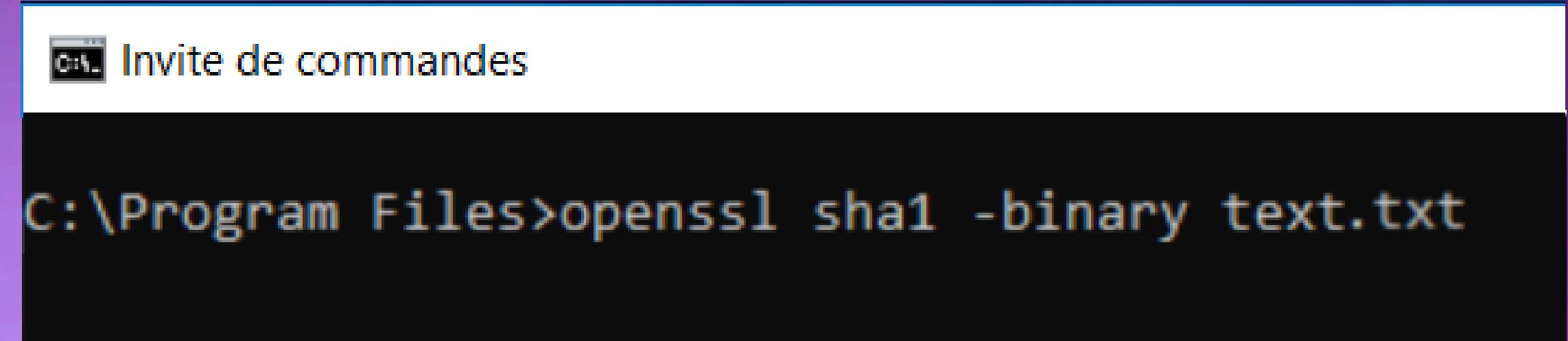
C:\Program Files>openssl genrsa -out key.pem 2048
```

```
-----BEGIN ENCRYPTED PRIVATE KEY-----
MIIFHDBOBgkqhkiG9w0BBQ0wQTApBgkqhkiG9w0BBQwwHAQI95wKs/ZyO0YCAggA
MAwGCCqGSIB3DQIJBQAwFAYIKoZIhvcNAwcECOKAOaDtdWZxBIIEyGWhmop9+haE
/dLy+ciWVv38DuHJVe49xVLHzoIyfe6Jkc2jidXSQ71tc6to2hgFXPG3y0ZPrUti
gAc5NifmLbMBlOX2d4y9Ztzq1EN1lvcV8mg9WzAGGlG1m5anJHKegO8ajw43mLk
K3S47aRnG9o3oj0mr00EL6/NnUvjQLWSDBL3cgAx54MbVZr/lu6exiy/3MupC+US
of7fQrVEV7n0QZ10B3qaP4MZBX6X36/Hzm0dPu+Si97924albm3j9wz8ohPUpn1G
/cObexZtiI4UcyCJvk2wPMH4bVhwI7HU5IQaowjW9egLNGBbSTIoLzba1H1kfWa5
0JZyGvcs4nc13B6y2Pau6jVRzKqGt18KdrXtinX3P+2E4bycIRir81jhgg6toBoa
FLXPKoXkNk0+DG1HtEq8eZc3dk+k2U1K8d4QAzSD54g0mb88QztEaKvOwVH39CkC
U5iGVLRgxnG+oAh0JGK04sxfmUIjMH0XjaCM9MbHgr/844aXbdgUzc/GuDC37Jom
NB4u3IApULGCM0dkrt29j0sQ2hcaLn4m1AlxBjvPS1IYq+j1nwnJEI1s8BpNLrCH
SZGVVV+rXrream+kgakkpuiuI5MXRM8h4mzxFxWkF93qijkJ1UFZ0E1NmHPGEeG6/
yyPw9Xxd3JCTQzSz60iN42Q7KAyxASKPx02NNoaZkJvSMu4ruhHxj2GRUKFzKj
00NUaShVzQpsVoZ1GET0wOC6vNDflkFYUjkvg3vxFgWd6EgEpSasQaEeWdlbDxm
/Ij4T3ZtQ10S1cT1BceconJCRuc37eV0gCfdwbQeMExp004bYFYh/VKui7ITLUew
n0c4VHo0WxkCCX2/MH4hwWaYpraFYfIwfMykah38BVOTzntucTEkSoB/q+9us1kR
IgWwypN/N7Q0qHXfZK/k0e7usFNDzdSg/iK144EMZ7WnPjM3jVzKTpqB+nHS6uo
tokBspmwaVB68MH/+goybzWnHSrntm1yOjGKmxjapNwtnUT5w0a1Vy4KFKMk6R9Q
h2VWPfkVZ/nB7Gmx7bwj2LNNVURdREwLanM1vhrmKVugTuv3uUsUz/lMP+NxBgw
rCo37JD4hvUNbjQItHvTKJATXbCZ1+GmvU57ciyFM1Jx9rKf3FQv1+JnT5WqrJMr
iIc6kgkZeb2gW49XdV9Nfk2Wmt+jHnesRKGg2Fx+kKRzFMKcRqJ+b731V9zNg9Be
6IvwZvyGdgLR/eqid5FgCV0eo+0xGwDCNQOK1Q9gwcEb5CeQYgtTBDL8o77BoNh1
dvn9/N4yVqZA4UWQQQ4ivkSqcKIhaUXpA9Tu26oPDYkI1MZD1AH5EMTOm62563SB
QdUEM1LF0mW1Vp+6t0WEATmtBnAmh5HYd78tnX741iNdWIiuLzYcrWi3Wo4q1+aK
gnqq2EXRR7fb/V76MWSHnnJspBUqK/Oq58Pjq1xska+kq81CRTAZphHT5/mQrZYm
q5fHKcsog8iUdR74TvrjpXXaancVirigPVWk1J/y406KzEiH/VgDAPs0Gvvny/4Q
9pg17XbpvvGUSSyduc3Nzh7ar+rVuw4vlBotB2Uw7PMc6cHpx9EcSRVdwNp6dVwo
zH3FpnhtpCpDSoxjZoVKzg==
-----END ENCRYPTED PRIVATE KEY-----
```

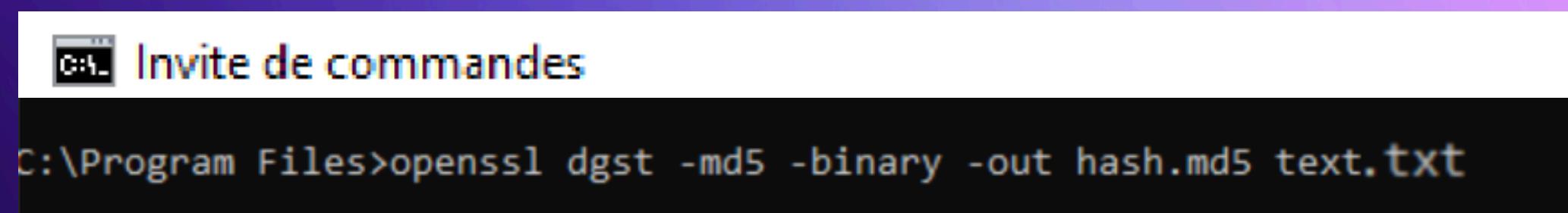
## Partie C: Cracking\_passwords

### 3/ Hachage :

- "**openssl sha1**" : Commande pour générer un hash SHA-1.
- "**-binary**" : Indique à OpenSSL de produire une sortie binaire plutôt qu'une sortie hexadécimale, pratique pour les données binaires ou les fichiers.
- "**textb.txt**" : Nom du fichier pour lequel le hash SHA-1 sera généré. Assurez-vous que le fichier est dans le même répertoire d'exécution.
- "**-out hash.sha1**" : Spécifie que le hash SHA-1 sera sauvegardé dans un fichier nommé hash.sha1. Vous pouvez remplacer hash.sha1 par le nom de fichier de votre choix.



```
C:\Program Files>openssl sha1 -binary text.txt
```



```
C:\Program Files>openssl dgst -md5 -binary -out hash.md5 text.txt
```

- "**textb.txt**" : Nom du fichier pour lequel le hash MD5 sera généré.
- "**-md5**" : Paramètre indiquant à OpenSSL d'utiliser l'algorithme MD5 pour le hash.
- "**-binary**" : Paramètre indiquant à OpenSSL de produire une sortie binaire plutôt qu'une sortie hexadécimale, ce qui est souvent utile pour les données binaires ou les fichiers.
- "**-out hash.md5**" : Spécifie que le hash MD5 sera sauvegardé dans un fichier nommé hash.md5. Vous pouvez remplacer hash.md5 par le nom de fichier de votre choix.
- "**openssl dgst**" : Commande pour générer un hash à partir de données.

Merci