EJERCICIOS CRIPTOGRAFÍA

Prerrequisitos

- Busca información sobre el paquete hash-identifier en Kali y comprueba si lo tienes instalado. En caso que no lo esté, instálalo. Crea un fichero de texto llamado "ejercicio_crypto.txt" en la ruta $HOME/Crypto con este texto: b2pvIGltcG9ydGFudGUgcXVlIGNvZGlmaWNhciBubyBlcyBsbyBtaXNtbyBxdWUgY2lmcmFy

```
┌──(root㉿kalilinux)-[/home]
└─# mkdir $HOME/Crypto
```

```
┌──(root㉿kalilinux)-[/home/Crypto]
└─# touch ejercicio_crypto.txt
```

```
┌──(root㉿kalilinux)-[/home/Crypto]
└─# cat ejercicio_crypto.txt
b2pvIGltcG9ydGFudGUgcXVlIGNvZGlmaWNhciBubyBlcyBsbyBtaXNtbyBxdWUgY2lmcmFy
```

Ejercicios - Funciones hash

1.Crea un hash MD5 del fichero ejercicio_crypto.txt.

```
┌──(root㉿kalilinux)-[/home/Crypto]
└─# md5sum ejercicio_crypto.txt
3244441dda3489d1fa5cdb1cf72342dd  ejercicio_crypto.txt
```

2.Comprueba con hash-identifier el resultado con el hash obtenido. ¿Acierta en la predicción del tipo de hash?
Acierta la predicción del tipo de hash MD5

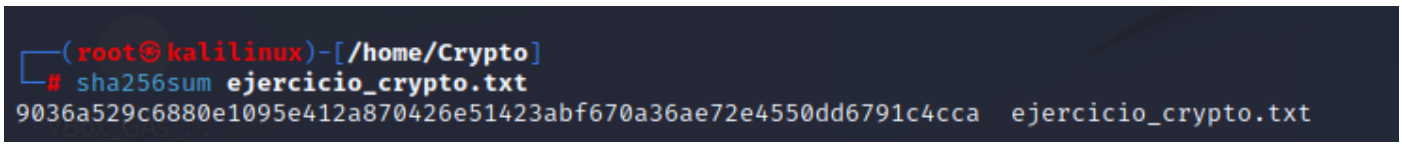
```
┌──(root㉿kalilinux)-[/home/Crypto]
└─# hash-identifier 3244441dda3489d1fa5cdb1cf72342dd
   #########################################################################
   #     __  __                     __           ______    _____           #
   #    /\ \/\ \                   /\ \         /\__  _\  /\  _ `\         #
   #    \ \ \_\ \     __      ____ \ \ \___     \/_/\ \/  \ \ \/\ \        #
   #     \ \  _  \  /'__`\   / ,__\ \ \  _ `\      \ \ \   \ \ \ \ \       #
   #      \ \ \ \ \/\ \_\ \_/\__, `\ \ \ \ \ \      \_\ \__ \ \ \_\ \      #
   #       \ \_\ \_\ \___ \_\/\____/  \ \_\ \_\     /\_____\ \ \____/      #
   #        \/_/\/_/\/__/\/_/\/___/    \/_/\/_/     \/_____/  \/___/  v1.2 #
   #                                                             By Zion3R #
   #                                                    www.Blackploit.com #
   #                                                   Root@Blackploit.com #
   #########################################################################
--------------------------------------------------

Possible Hashs:
[+] MD5
[+] Domain Cached Credentials - MD4(MD4(($pass)).(strtolower($username)))
```

3.Crea un hash SHA-1 del fichero ejercicio_crypto.txt.

```
┌──(root㉿kalilinux)-[/home/Crypto]
└─# sha1sum ejercicio_crypto.txt
ddcf5b509a5a1677fc214ae2bb1795983a55fd4d  ejercicio_crypto.txt
```

4.Comprueba con hash-identifier el resultado con el hash obtenido. ¿Acierta en la predicción del tipo de hash?
Se comprueba la predicción.

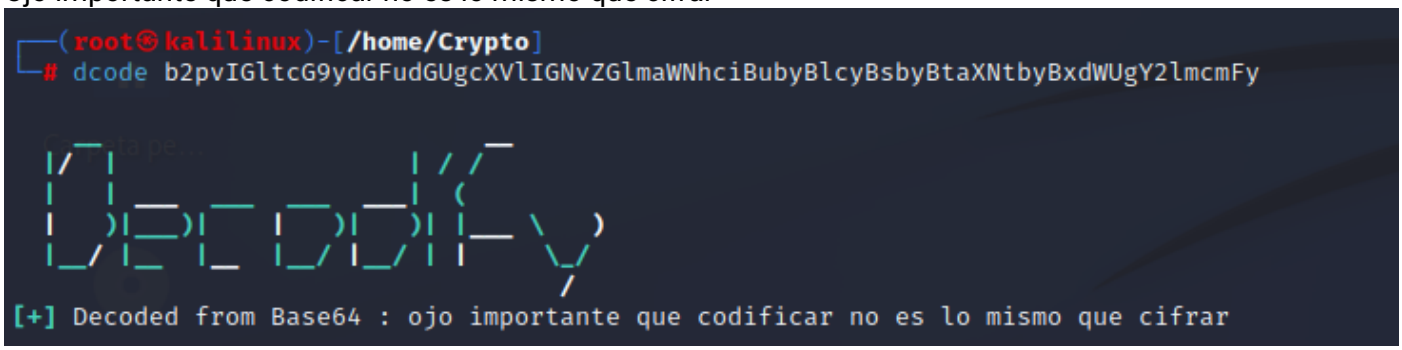
5.Crea un hash SHA-256 del fichero ejercicio_crypto.txt.



6.Comprueba con hash-identifier el resultado con el hash obtenido. ¿Acierta en la predicción del tipo de hash?
Acierta la predicción

Ejercicios – Codificación

7.Comprueba la cadena de texto que contiene el fichero. ¿Qué formato es? Utiliza la herramienta Decodify para identificarla y una vez identificada intenta decodificarla utilizando openssl.
Base 64
Ojo importante que codificar no es lo mismo que cifrar



```
┌──(root㉿kalilinux)-[/home/Crypto]
└─# openssl base64 -d -in ejercicio_crypto.txt -out ejercicio_crypto1.txt

┌──(root㉿kalilinux)-[/home/Crypto]
└─# ls
ejercicio_crypto1.txt  ejercicio_crypto.enc  ejercicio_crypto.txt
```

```
┌──(root㉿kalilinux)-[/home/Crypto]
└─# cat ejercicio_crypto1.txt
ojo importante que codificar no es lo mismo que cifrar
```

8.Codifica el texto "No metemos gente en criptas" en el mismo formato usando openssl.

```
┌──(root㉿kalilinux)-[/home/Crypto]
└─# nano ejercicio_crypto2.txt

┌──(root㉿kalilinux)-[/home/Crypto]
└─# openssl base64 -in ejercicio_crypto2.txt -out ejercicio_crypto3.txt

┌──(root㉿kalilinux)-[/home/Crypto]
└─# ls
ejercicio_crypto1.txt  ejercicio_crypto2.txt  ejercicio_crypto3.txt  ejercicio_crypto.enc  ejercicio_crypto.txt

┌──(root㉿kalilinux)-[/home/Crypto]
└─# cat ejercicio_crypto3.txt
Tm8gbWV0ZW1vcyBnZW50ZSBlbiBjcmlwdGFzCg==

┌──(root㉿kalilinux)-[/home/Crypto]
└─# cat ejercicio_crypto2.txt
No metemos gente en criptas
```

Ejercicios - Cifrado simétrico

9.Cifra el texto "AES es un tipo de cifrado simetrico" con AES-256 Cipher (aes-256-cbc) y con password "AES".

```
┌──(root㉿kalilinux)-[/home/Crypto]
└─# echo 'AES es un tipo de cifrado simetrico' | openssl enc -aes-256-cbc -a

enter AES-256-CBC encryption password:
Verifying - enter AES-256-CBC encryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
U2FsdGVkX19PLsX9Jo199ZUScwfCqU1RHM4jpv6mB4FwIi/x0RqSDrmcok9FPhay
xaQrBh4yzS5vxalna6vLcQ==
```

10.Descifra el resultado del ejercicio anterior con su password para recuperar el contenido.

```
┌──(root㉿kalilinux)-[/home/Crypto]
└─#  echo 'U2FsdGVkX19PLsX9Jo199ZUScwfCqU1RHM4jpv6mB4FwIi/x0RqSDrmcok9FPhayxaQrBh4yzS5vxalna6vLcQ==' | openssl enc -aes-256-cbc -d -a
enter AES-256-CBC decryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
AES es un tipo de cifrado simetrico
```

11.Descifra la cadena de texto indicada con la clave oculta AES 256 siguiente:
Clave oculta: cXVlIHNlcmEgZXN0byBkZSAweCA3MCA2MSA3MyA3MyA3NyAzMCA3MiA2NA==
Cadena de texto: U2FsdGVkX1+bYl9elFTkoc6qzP/zV0QXirGvitorwZiljKtv1FN6PwCtkIKVmyBP

VER LA CORRECCION, INTENTE DE MIL MANERAS ☹

(Opcional) Ejercicios - Cifrado asimétrico

12.Genera una clave privada RSA 2048 y guardala en un fichero privada.pem usando openssl.

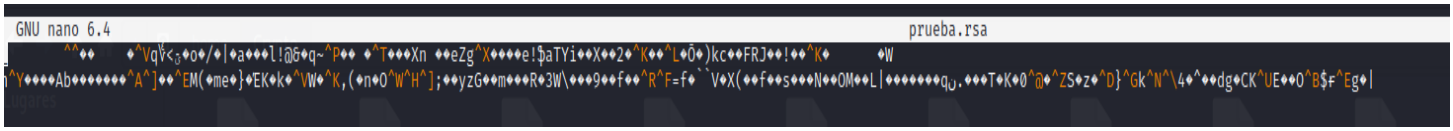
```
┌──(root㉿kalilinux)-[/home/Crypto]
└─# openssl genrsa -aes256 -out llave_privada 2048
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:

┌──(root㉿kalilinux)-[/home/Crypto]
└─# chmod 744 llave_privada

┌──(root㉿kalilinux)-[/home/Crypto]
└─# cat llave_privada
-----BEGIN ENCRYPTED PRIVATE KEY-----
MIIFLTBXBgkqhkiG9w0BBQ0wSjApBgkqhkiG9w0BBQwwHAQIo0HtYoqffroCAggA
MAwGCCqGSIb3DQIJBQAwHQYJYIZIAWUDBAEqBBAjXkqadaSMRPLk89uT002QBIIE
0KeR5dbT49SD5XPPQiBa+QNsNZcNq1y5f1fvLlYP6Fo34×2qTFi2EZqQo2oc2u2k
SLz0P9zxJF5+gW3yDaD5Kf6sdulVR83ynDTLKUCWn6X3NN/TpAX30UoFMmH7NVDn
gIzREk2Wsbl5odzF927hnm537aubM5LKyA+vk+FpV8lKoaqQA0UZP8MQPFbLfoxO
Hcdi6wCILwfNLWVJ3PDrKzdbvSabWneNJOTv4pSQfefjzIT9lOFOs9nsdrgImZaP
90VnJwozN5b9xQYlgBO5f/1Jr00kcdXNM6zOXnv7koGUKdYbztfpHzJE7BphfVM2
Lf87ctyH+DPVDhBiIMcqZgriOlDf2UlWJFk3vPc3yv4qOi5dbIHXy6ffMkPwdF3S
GCF3tF5nRgzASPTlV8xRLK4Uwg/cn9xGdKbHXZmlisyJFvAWtoAThuZu809IQVvC
eQnl9rHvq49yC5DMXB/Gk2aHqVLdIP2tZOTgPGSGHAPWjEqUsUnMo9k4kJbo8rSR
E/rK4R3HVbjdeJ546kfQGSFSzoziTQd/OIE4uGUHsruCM3cZTpkVluGEYfc9eKMo
4YWVI8tbFZ9wMdHKveDQ7p2/LTjYCBMWoOpmjlY1z++cEpqq6u5KGm+SSth73SJJ
FgUskq186SVN1vWCNKE8G9dgOet1ab5rtHY3JjaF8LKD9MYrFtGO5+3Pk82V+5ME
+IQzfEweZIreTIFpA241oL9EKH6SUmqz48t3CZ8p1cpybtg+OG7aExghc+yHCmFz
JuYbq0Ca4MR0NuB6W7G96c12DxQ+u1Gtk4En+YziOeEz2RMgkBS8xnkvG7hEn7Dp
gKbK/qQ8ZJ4fv/a+V+QbROD6NlBf63MDph2s037mJOQJIV78ebFU3vka40Q0fASq
LILnJEaikMO41s5Khti2qv84H2cH5ouGdobocFqYV5QAQbsaw8aLZlqNzZO5NMG3
W/emdAaSyY3E/mqSNge3683jkNk7LBimJM0yvlQ/rJ8Sky/afyfKZ4/DTwpVv4nX
daz9qwaCL4Rimp33EAX8aso264wzRhxgHCAsYpl+XksxozVprIGlQDxkI4W22Rry
DOp2PyznFWd3P9fSeCEqX0bBY0p//ULeywAsAr7OJnb54NnMKhhXLwm9AiuSN0gj
Favc4FIah0YRXzIGSs9dxAgrHEj3HaUeOZdOGIIXebJuGQ4enn6DhbIXdOmwrw2v
YdGGzoCsMycKqWZpi/wV2oe9I8sdxzUaAmi3ootg0wY811LybYiSay6f70VNWdik
KlTjkvpBz/CO2tS+42pHn5ERI2j36qnJYvI+mlKH32ayIzlW3FTAvyAG8F90aCYU
IIIdNx0DaNd77NG18SwqXe+kHRq6/rkw+C/+xkmqpOYJM0PX+mcLx6xIOYakHtyc
E5MR+BrieC1Le6V0QzX4nPjdf8N5oe06Qkn9pahjR9KSx9L7EN1ULyryS41xepMF
RTmdZ1uH8FFwl2N3GvkptynLUsedIGDMBuaQ5grkoj0Z32h16wSl194gkVUTJVTD
zNubT2fXJpb9fXo955Jco/b3FKTtL4Jr0wQArppAl5dayCx5vQcWmt+hNmYv3X5z
3NbTL5sozKT5B4xjUsTk6Dcei0zsRqwAYPLKd+wqEvM5
-----END ENCRYPTED PRIVATE KEY-----
```

13.Genera una clave pública (publica.pem) desde la clave privada anterior usando openssl.

```
┌──(root㉿kalilinux)-[/home/Crypto]
└─# openssl rsa -pubout -out llave_publica -in llave_privada
Enter pass phrase for llave_privada:
writing RSA key

┌──(root㉿kalilinux)-[/home/Crypto]
└─# chmod 744 llave_publica

┌──(root㉿kalilinux)-[/home/Crypto]
└─# cat llave_publica
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAy9CLSV8mPBWZHyDIn8hV
tqQbsGUlNWg7/JEAP3CctSwvgiPVgsoFV2vmt23Jfp4w0ZfbFg8YnOl6Z1muMsoR
3bhUoVMaKIj1kWpUABFc+H4URpTgLdql91L3Z/Tb1OADTptUE4XJBt7bhp/rn2Oj
VT2xcifVYsym3GgpSqLSblt6FDYWOi5lT3yITaGBMBD8vkofj4J/NWXk2MlPwi8u
HHoCPTheOF51C8cZ6KGMbW0P2TnHI9C1GAZ0kx8qeZGriiSBaZgxZt6GpOpTSS1Q
XsZtkcOnXXSA57t6rj8Y3f4P+58JjTeMCjZfVMNdogq1E26tqGxvlRu36+/sor0X
6QIDAQAB
-----END PUBLIC KEY-----
```

14.Cifra el archivo "ejercicio_crypto.txt" usando la clave PRIVADA RSA 2048 y guardalo como prueba.rsa
En este punto por un tema de pruebas, cifre con la clave publica para probar que puedo decifrar con la clave privada en el ejercicio 15.

```
┌──(root㉿kalilinux)-[/home/Crypto]
└─# cat ejercicio_crypto.txt
b2pvIGltcG9ydGFudGUgcXVlIGNvZGlmaWNhciBubyBlcyBsbyBtaXNtbyBxdWUgY2lmcmFy

┌──(root㉿kalilinux)-[/home/Crypto]
└─# openssl rsautl -encrypt -pubin -inkey llave_publica -in ejercicio_crypto.txt -out prueba.rsa
The command rsautl was deprecated in version 3.0. Use 'pkeyutl' instead.

┌──(root㉿kalilinux)-[/home/Crypto]
└─# openssl pkeyutl -encrypt -pubin -inkey llave_publica -in ejercicio_crypto.txt -out prueba.rsa

┌──(root㉿kalilinux)-[/home/Crypto]
└─# nano ejercicio_crypto.txt

┌──(root㉿kalilinux)-[/home/Crypto]
└─# nano prueba.rsa
```



15.Descifra el archivo prueba.rsa usando la clave PRIVADA RSA 2048 y comprueba el contenido.
COMPROBADO 😊

```
┌──(root㉿kalilinux)-[/home/Crypto]
└─# openssl pkeyutl -inkey llave_privada -in prueba.rsa -out prueba2.txt
Enter pass phrase for llave_privada:
Error: The input data looks too long to be a hash

┌──(root㉿kalilinux)-[/home/Crypto]
└─# openssl pkeyutl -decrypt -inkey llave_privada -in prueba.rsa -out prueba2.txt
Enter pass phrase for llave_privada:

┌──(root㉿kalilinux)-[/home/Crypto]
└─# nano prueba2.txt
```