

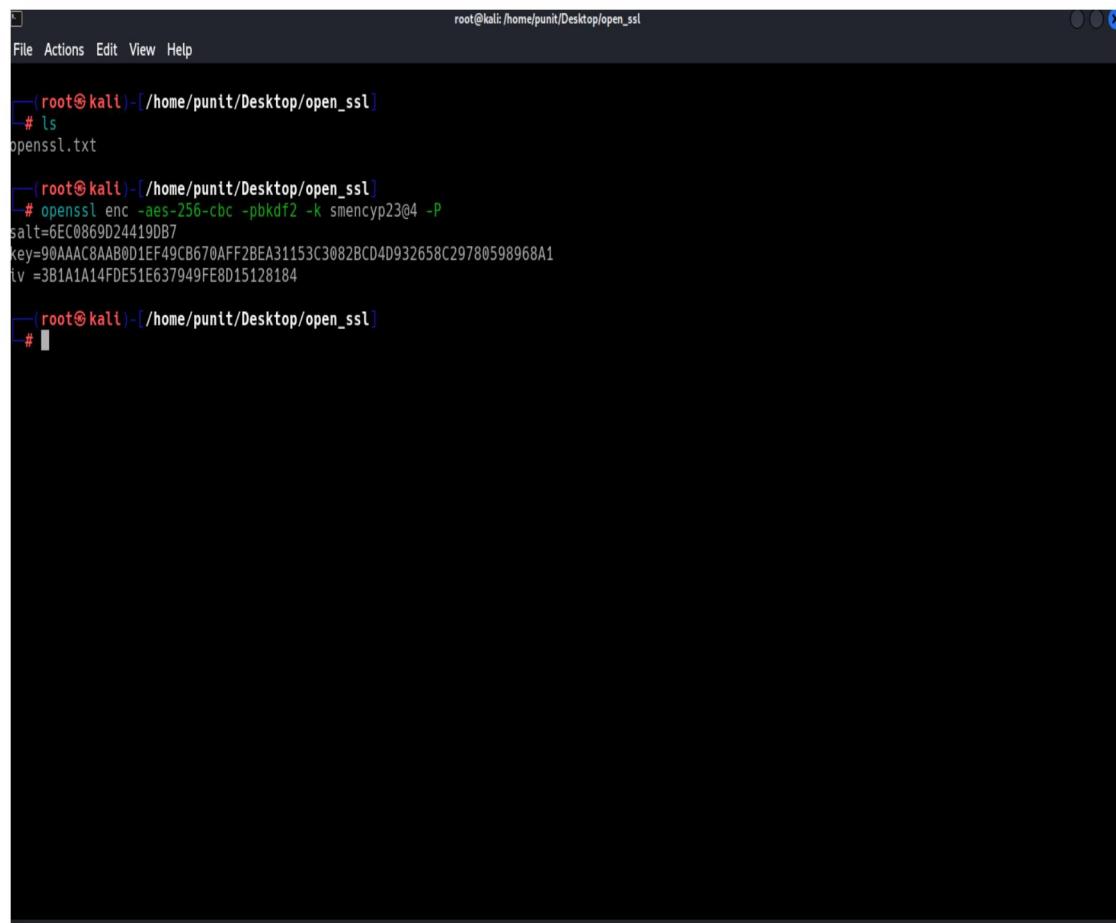
PRACTICAL 9

AIM: Implement encryption and decryption with openssl.

Steps:

1. Generate an Encryption Key and IV:

```
openssl enc -aes-256-cbc -pbkdf2 -k smencyp23@4 -P
```



A terminal window titled "root@kali: /home/punit/Desktop/open_ssl" showing the execution of an OpenSSL command. The command generates an encryption key and IV using PBKDF2 with salt and key values displayed.

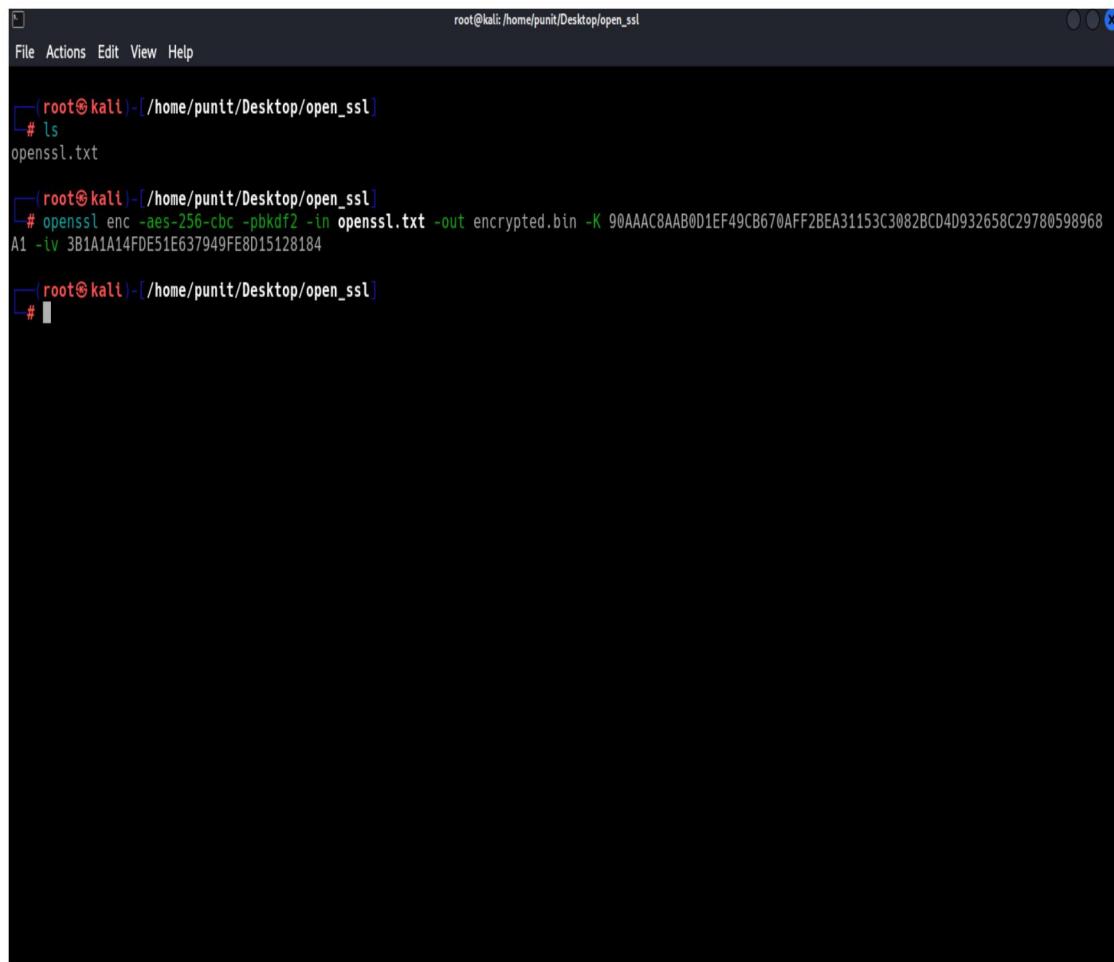
```
root@kali: /home/punit/Desktop/open_ssl
File Actions Edit View Help
[root@kali ~]# ls
openssl.txt

[root@kali ~]# openssl enc -aes-256-cbc -pbkdf2 -k smencyp23@4 -P
salt=6EC0869D24419DB7
key=90AAC8AA80D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968A1
iv =3B1A1A14FDE51E637949FE8D15128184

[root@kali ~]#
```

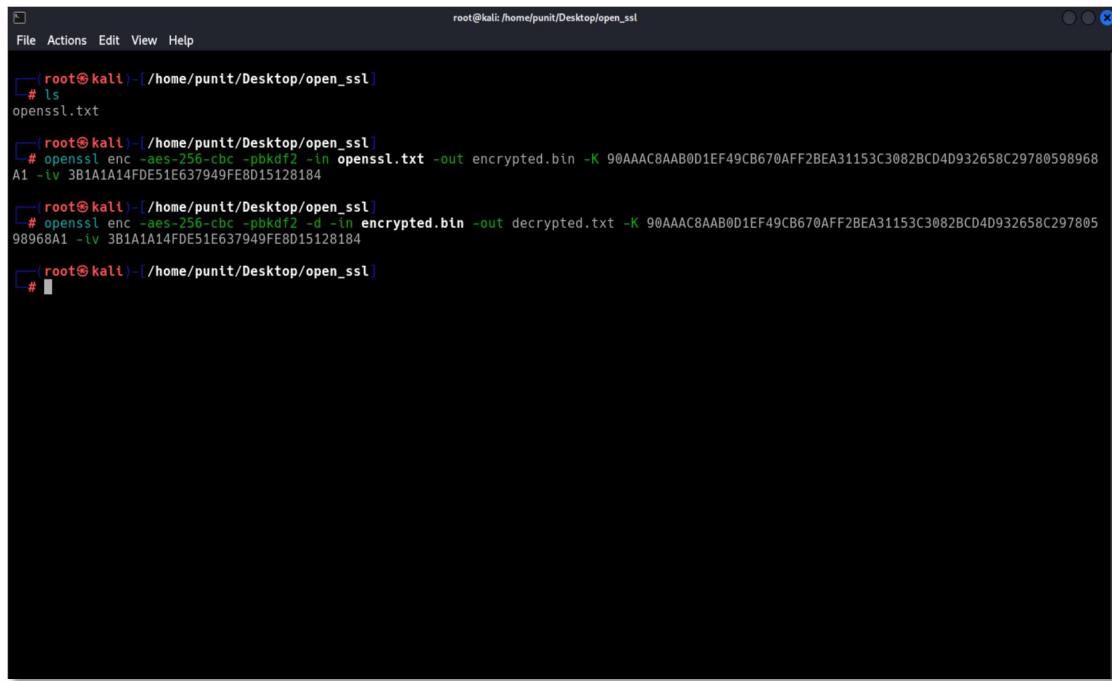
2. Encrypt a File:

```
openssl enc -aes-256-cbc -pbkdf2 -in openssl.txt -out encrypted.bin -K <key> -iv <iv>
key=90AAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968A1
iv =3B1A1A14FDE51E637949FE8D15128184
```

A screenshot of a terminal window titled "root@kali: /home/punit/Desktop/open_ssl". The window shows a command-line session where a file named "openssl.txt" is being encrypted into "encrypted.bin" using the OpenSSL command. The command includes options for AES-256-CBC encryption, PBKDF2 keying, and specific key and IV values. The terminal interface has a dark background with light-colored text, and the window title bar is visible at the top.

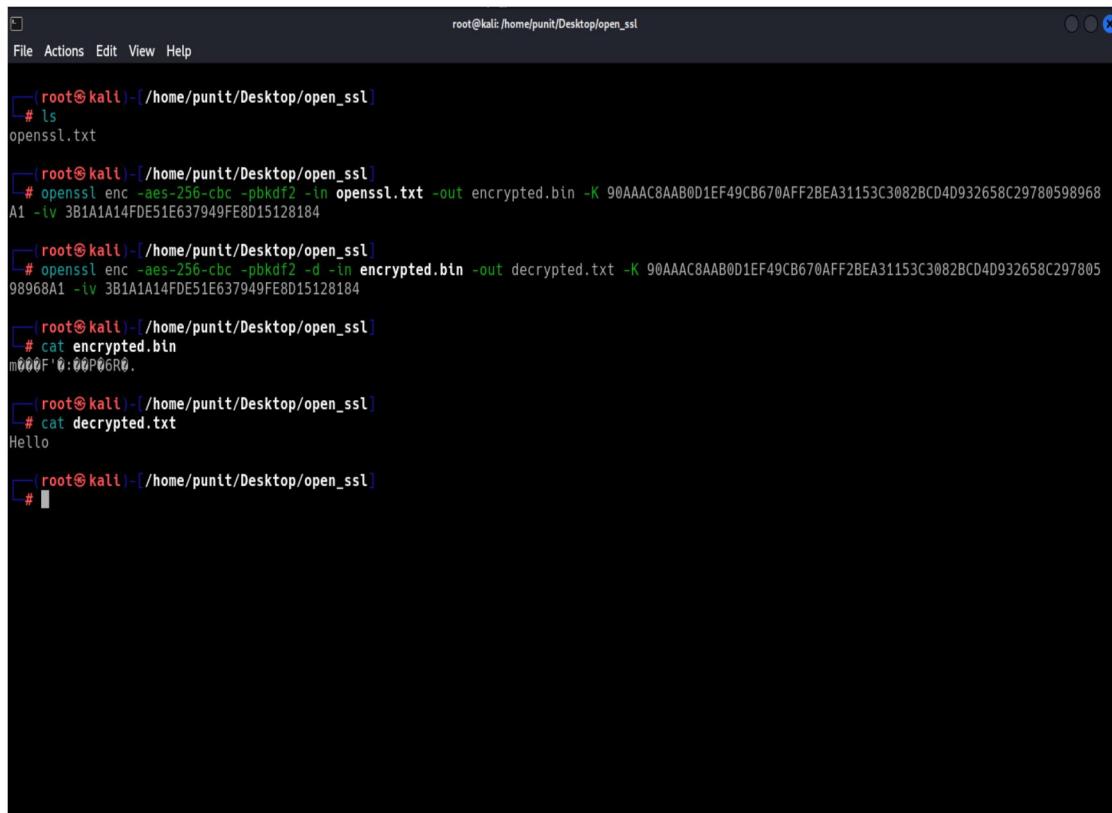
3. Decrypt the File:

```
openssl enc -aes-256-cbc -pbkdf2 -d -in encrypted.bin -out decrypted.txt -K  
90AAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C297805989  
68A1 -iv 3B1A1A14FDE51E637949FE8D15128184
```



This terminal window shows the command to decrypt the file. It starts with an 'ls' command to show the directory contents, which includes 'openssl.txt'. Then, it runs the decryption command, resulting in the creation of 'decrypted.txt'. Finally, it lists the directory again to show that 'decrypted.txt' has been successfully created.

```
File Actions Edit View Help  
root@kali:~/home/punit/Desktop/open_ssl]  
# ls  
openssl.txt  
[root@kali:~/home/punit/Desktop/open_ssl]  
# openssl enc -aes-256-cbc -pbkdf2 -d -in openssl.txt -out encrypted.bin -K 90AAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968  
A1 -iv 3B1A1A14FDE51E637949FE8D15128184  
[root@kali:~/home/punit/Desktop/open_ssl]  
# openssl enc -aes-256-cbc -pbkdf2 -d -in encrypted.bin -out decrypted.txt -K 90AAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968  
98968A1 -iv 3B1A1A14FDE51E637949FE8D15128184  
[root@kali:~/home/punit/Desktop/open_ssl]  
#
```



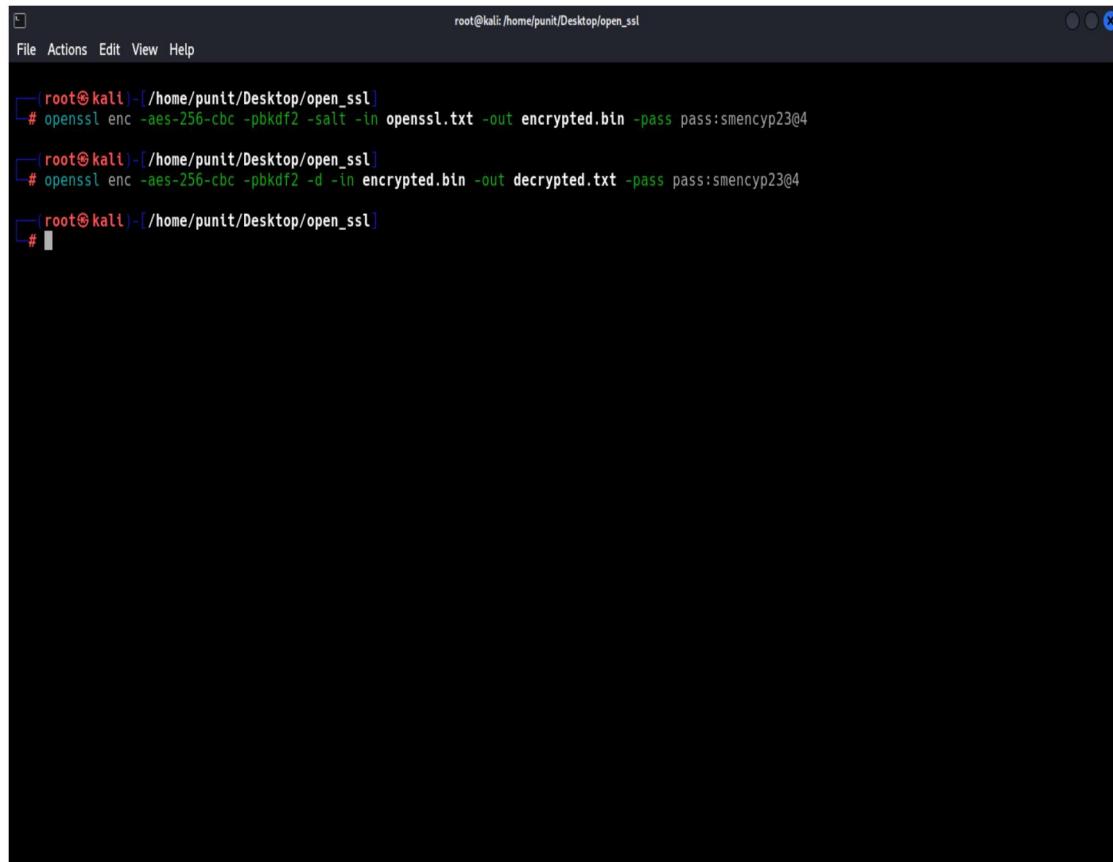
This terminal window shows the verification of the decryption process. It first lists the directory to confirm the presence of 'encrypted.bin' and 'decrypted.txt'. Then, it uses 'cat' to read the content of both files. The 'encrypted.bin' file contains binary data starting with 'm\x00\x0F\x00:\x00P\x06R\x00'. The 'decrypted.txt' file contains the word 'Hello', confirming successful decryption.

```
File Actions Edit View Help  
root@kali:~/home/punit/Desktop/open_ssl]  
# ls  
openssl.txt  
[root@kali:~/home/punit/Desktop/open_ssl]  
# openssl enc -aes-256-cbc -pbkdf2 -d -in openssl.txt -out encrypted.bin -K 90AAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968  
A1 -iv 3B1A1A14FDE51E637949FE8D15128184  
[root@kali:~/home/punit/Desktop/open_ssl]  
# openssl enc -aes-256-cbc -pbkdf2 -d -in encrypted.bin -out decrypted.txt -K 90AAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968  
98968A1 -iv 3B1A1A14FDE51E637949FE8D15128184  
[root@kali:~/home/punit/Desktop/open_ssl]  
# cat encrypted.bin  
m\x00\x0F\x00:\x00P\x06R\x00.  
[root@kali:~/home/punit/Desktop/open_ssl]  
# cat decrypted.txt  
Hello  
[root@kali:~/home/punit/Desktop/open_ssl]  
#
```

4. Encrypt Using a Password Instead of a Key:

```
openssl enc -aes-256-cbc -pbkdf2 -salt -in openssl.txt -out encrypted.bin -pass  
pass:smencyp23@4
```

```
openssl enc -aes-256-cbc -pbkdf2 -d -in encrypted.bin -out decrypted.txt -pass  
pass:smencyp23@4
```

A screenshot of a terminal window titled "root@kali: /home/punit/Desktop/open_ssl". The window has a dark theme with white text. It shows three lines of terminal history:

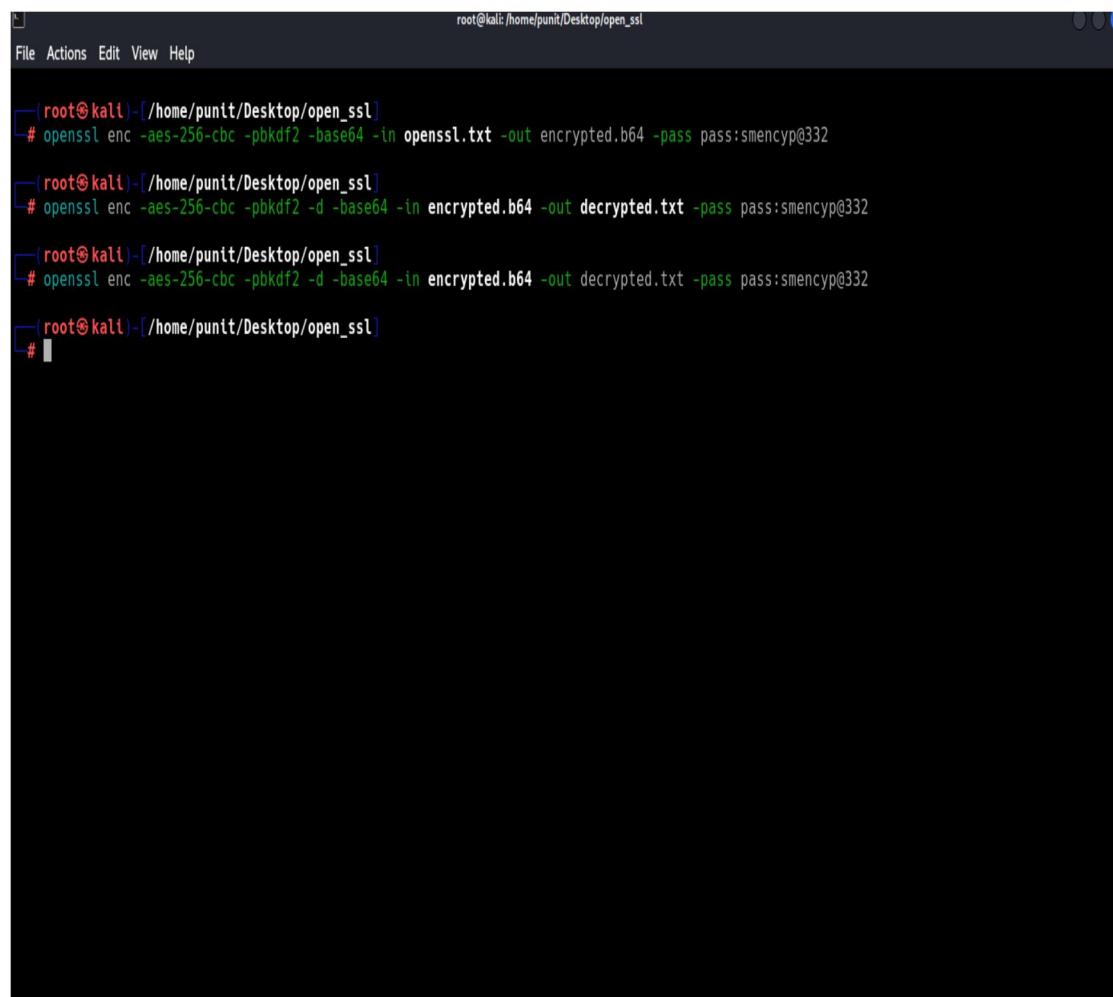
```
root@kali:~/Desktop/open_ssl  
# openssl enc -aes-256-cbc -pbkdf2 -salt -in openssl.txt -out encrypted.bin -pass pass:smencyp23@4  
root@kali:~/Desktop/open_ssl  
# openssl enc -aes-256-cbc -pbkdf2 -d -in encrypted.bin -out decrypted.txt -pass pass:smencyp23@4  
root@kali:~/Desktop/open_ssl  
#
```

The terminal prompt is "#".

5. View Encrypted File in Base64(a way to represent binary data as a string of text):

```
openssl enc -aes-256-cbc -pbkdf2 -base64 -in openssl.txt -out encrypted.b64 -pass  
pass:smencyp@332
```

```
openssl enc -aes-256-cbc -pbkdf2 -d -base64 -in encrypted.b64 -out decrypted.txt -  
pass pass:smencyp@332
```

A screenshot of a terminal window titled "root@kali: /home/punit/Desktop/open_ssl". The window shows a root shell session with the following commands and outputs:

```
root@kali:~/Desktop/open_ssl]  
# openssl enc -aes-256-cbc -pbkdf2 -base64 -in openssl.txt -out encrypted.b64 -pass pass:smencyp@332  
root@kali:~/Desktop/open_ssl]  
# openssl enc -aes-256-cbc -pbkdf2 -d -base64 -in encrypted.b64 -out decrypted.txt -pass pass:smencyp@332  
root@kali:~/Desktop/open_ssl]  
# openssl enc -aes-256-cbc -pbkdf2 -d -base64 -in encrypted.b64 -out decrypted.txt -pass pass:smencyp@332  
root@kali:~/Desktop/open_ssl]  
#
```

The terminal window has a dark background with light-colored text. The title bar and menu bar are visible at the top.