

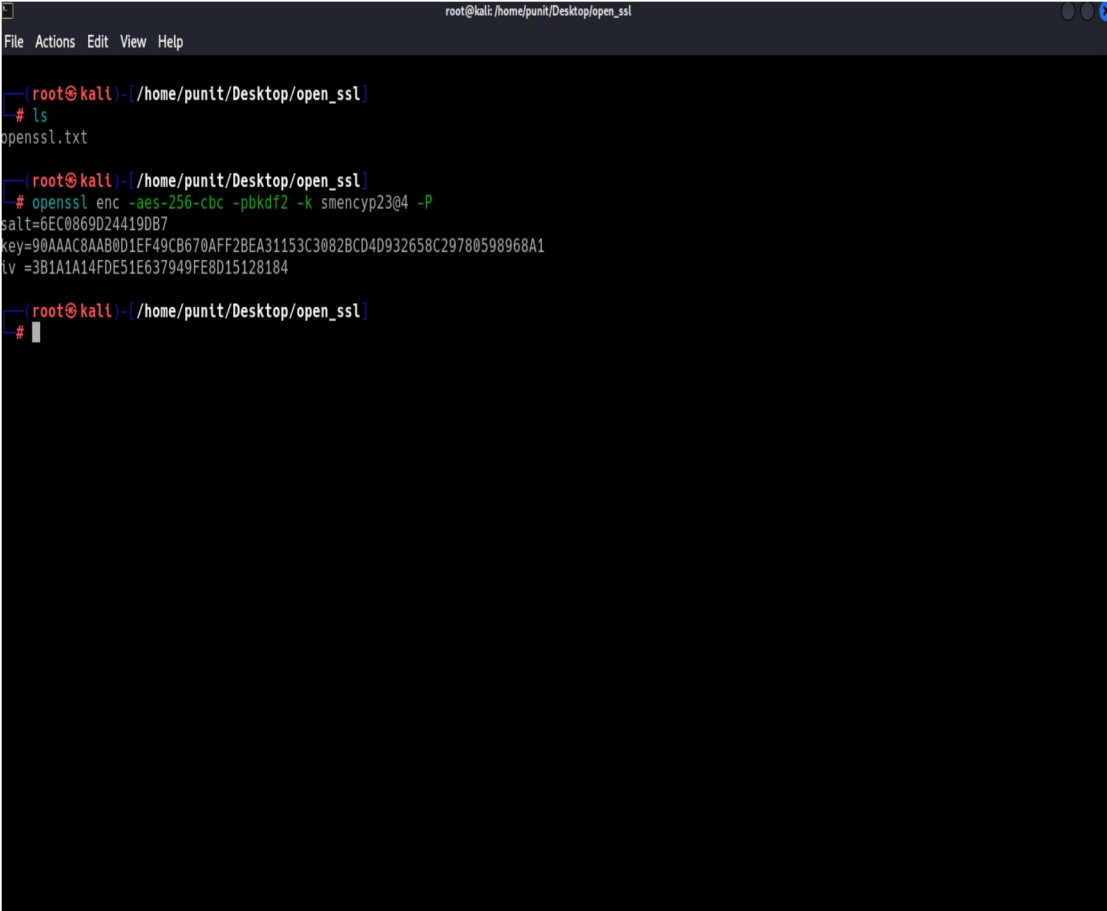
PRACTICAL 9

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`



```
root@kali: /home/punit/Desktop/open_ssl
File Actions Edit View Help

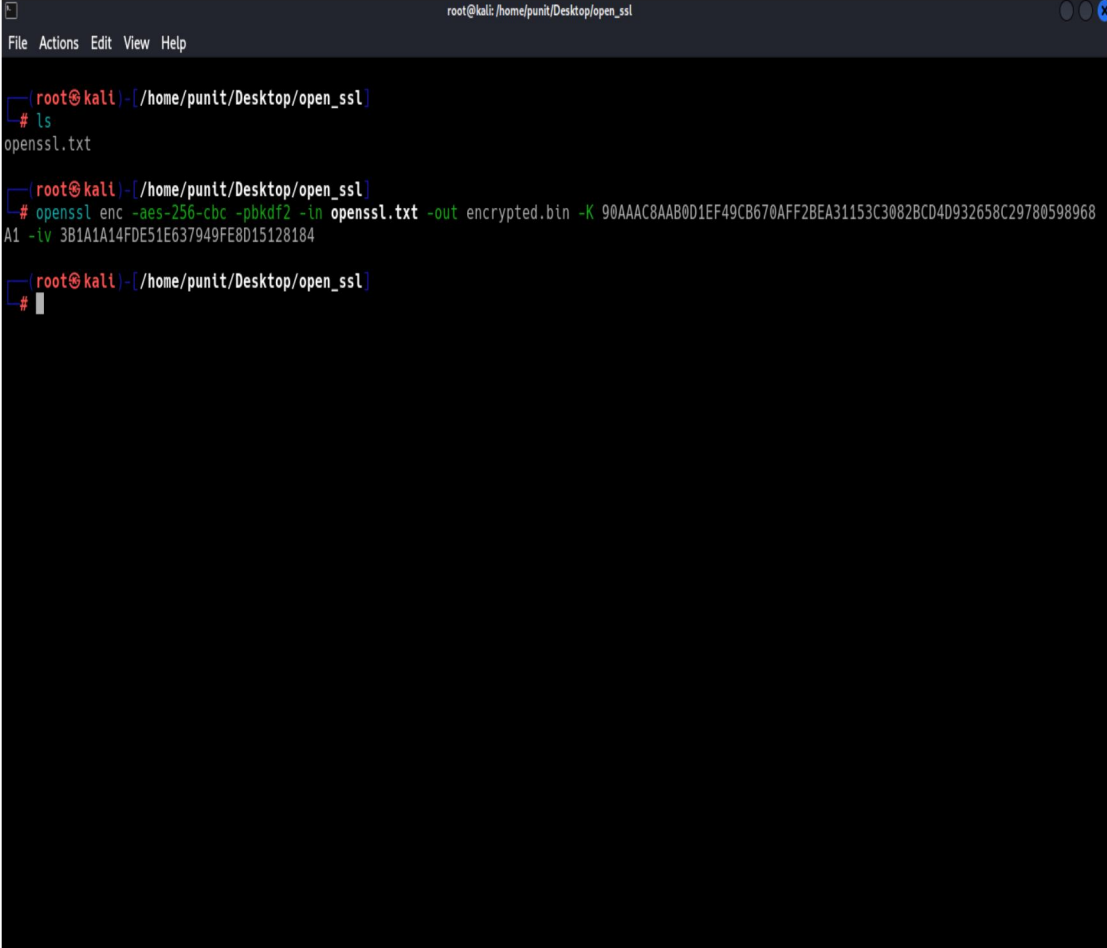
(root@kali)~/Desktop/open_ssl
# ls
openssl.txt

(root@kali)~/Desktop/open_ssl
# openssl enc -aes-256-cbc -pbkdf2 -k smencyp23@4 -P
salt=6EC0869D24419DB7
key=90AAAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968A1
iv =3B1A1A14FDE51E637949FE8D15128184

(root@kali)~/Desktop/open_ssl
#
```

2. Encrypt a File:

```
openssl enc -aes-256-cbc -pbkdf2 -in openssl.txt -out encrypted.bin -K <key> -iv <iv>  
key=90AAAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968A1  
iv =3B1A1A14FDE51E637949FE8D15128184
```

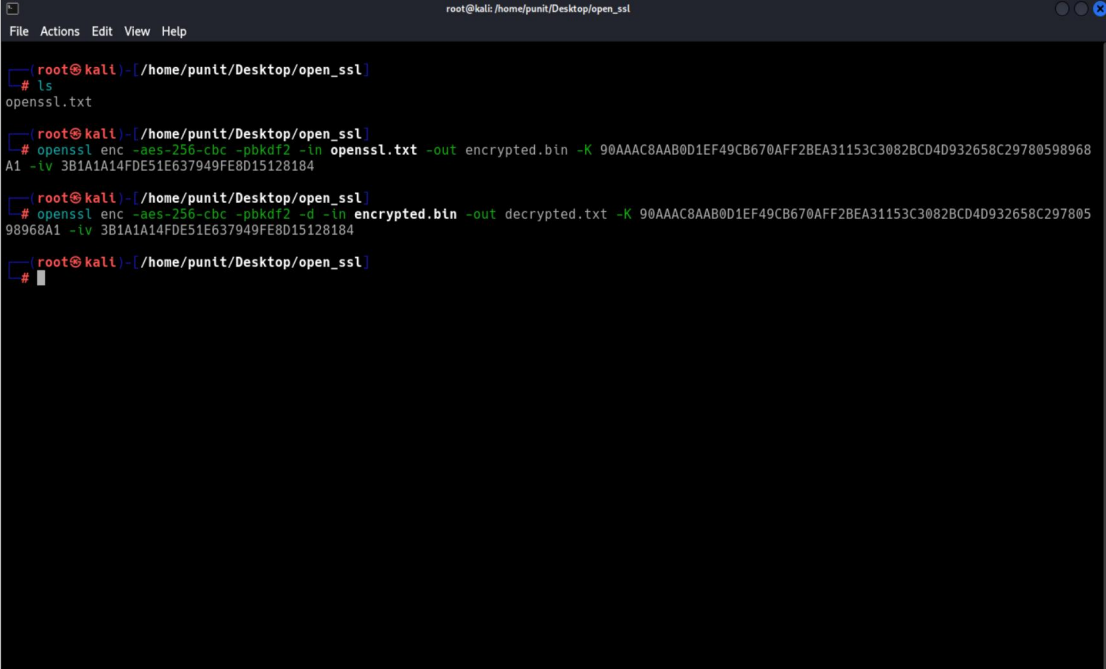


The screenshot shows a terminal window titled 'root@kali: /home/punit/Desktop/open_ssl'. The user is in a directory named 'open_ssl'. They run 'ls' and see 'openssl.txt'. Then they run the command: 'openssl enc -aes-256-cbc -pbkdf2 -in openssl.txt -out encrypted.bin -K 90AAAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968A1 -iv 3B1A1A14FDE51E637949FE8D15128184'. The command is executed successfully, and the prompt returns.

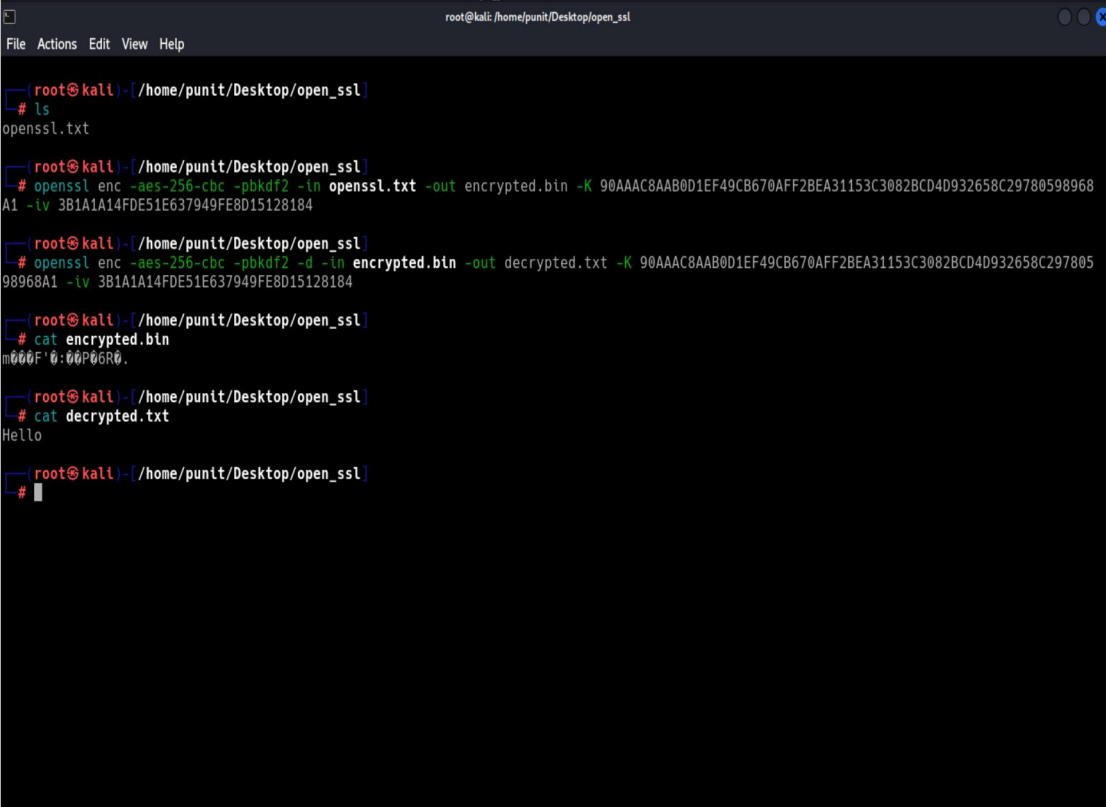
```
root@kali: /home/punit/Desktop/open_ssl  
File Actions Edit View Help  
root@kali: /home/punit/Desktop/open_ssl  
# ls  
openssl.txt  
root@kali: /home/punit/Desktop/open_ssl  
# openssl enc -aes-256-cbc -pbkdf2 -in openssl.txt -out encrypted.bin -K 90AAAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968A1 -iv 3B1A1A14FDE51E637949FE8D15128184  
root@kali: /home/punit/Desktop/open_ssl  
#
```

3. Decrypt the File:

```
openssl enc -aes-256-cbc -pbkdf2 -d -in encrypted.bin -out decrypted.txt -K  
90AAAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C297805989  
68A1 -iv 3B1A1A14FDE51E637949FE8D15128184
```

A terminal window titled 'root@kali: /home/punit/Desktop/open_ssl' showing the encryption of 'openssl.txt' into 'encrypted.bin'. The user runs 'ls' to confirm the file exists, then 'openssl enc -aes-256-cbc -pbkdf2 -in openssl.txt -out encrypted.bin -K 90AAAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968A1 -iv 3B1A1A14FDE51E637949FE8D15128184'. Finally, they run 'openssl enc -aes-256-cbc -pbkdf2 -d -in encrypted.bin -out decrypted.txt -K 90AAAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968A1 -iv 3B1A1A14FDE51E637949FE8D15128184' to create the decrypted file.

```
root@kali: /home/punit/Desktop/open_ssl  
# ls  
openssl.txt  
  
root@kali: /home/punit/Desktop/open_ssl  
# openssl enc -aes-256-cbc -pbkdf2 -in openssl.txt -out encrypted.bin -K 90AAAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968A1 -iv 3B1A1A14FDE51E637949FE8D15128184  
  
root@kali: /home/punit/Desktop/open_ssl  
# openssl enc -aes-256-cbc -pbkdf2 -d -in encrypted.bin -out decrypted.txt -K 90AAAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968A1 -iv 3B1A1A14FDE51E637949FE8D15128184  
  
root@kali: /home/punit/Desktop/open_ssl  
#
```

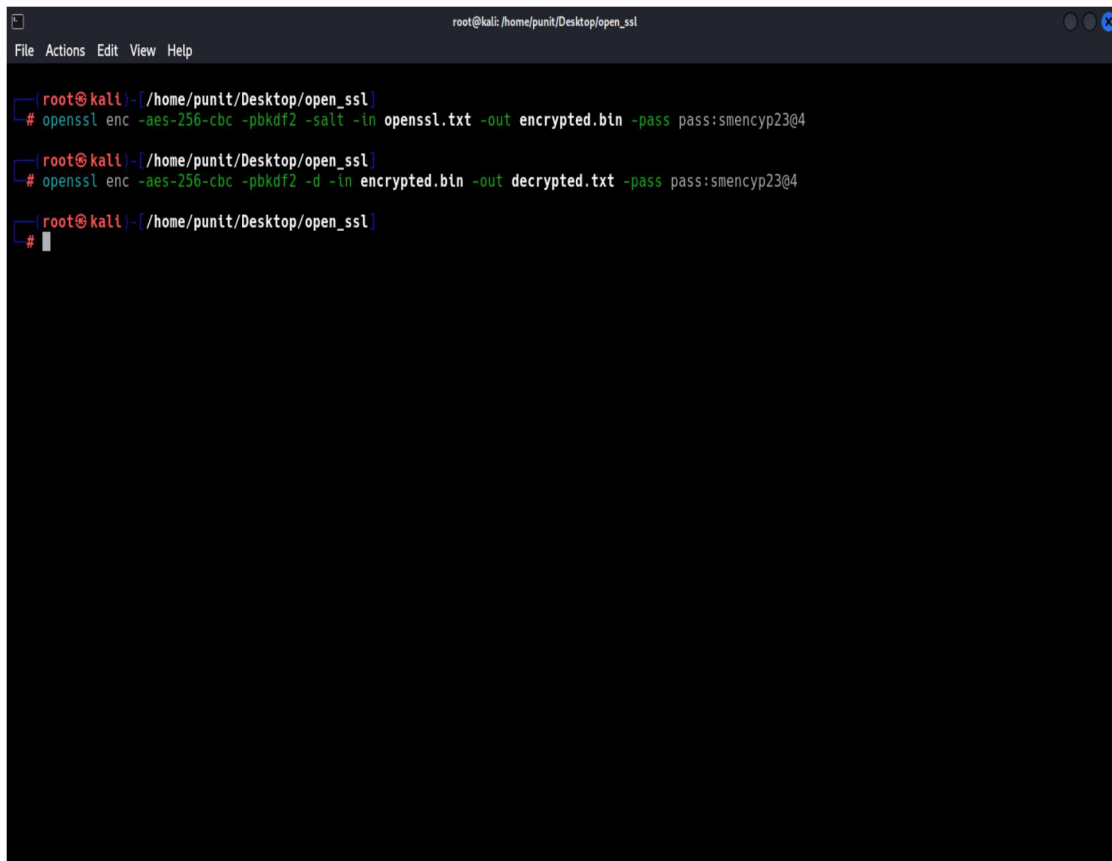
A terminal window titled 'root@kali: /home/punit/Desktop/open_ssl' showing the decryption of 'encrypted.bin' into 'decrypted.txt'. The user runs 'cat encrypted.bin' which shows the hex representation of the file, and 'cat decrypted.txt' which shows the plaintext 'Hello'.

```
root@kali: /home/punit/Desktop/open_ssl  
# ls  
openssl.txt  
  
root@kali: /home/punit/Desktop/open_ssl  
# openssl enc -aes-256-cbc -pbkdf2 -in openssl.txt -out encrypted.bin -K 90AAAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968A1 -iv 3B1A1A14FDE51E637949FE8D15128184  
  
root@kali: /home/punit/Desktop/open_ssl  
# openssl enc -aes-256-cbc -pbkdf2 -d -in encrypted.bin -out decrypted.txt -K 90AAAC8AAB0D1EF49CB670AFF2BEA31153C3082BCD4D932658C29780598968A1 -iv 3B1A1A14FDE51E637949FE8D15128184  
  
root@kali: /home/punit/Desktop/open_ssl  
# cat encrypted.bin  
m000F 0:00P06R0.  
  
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
```



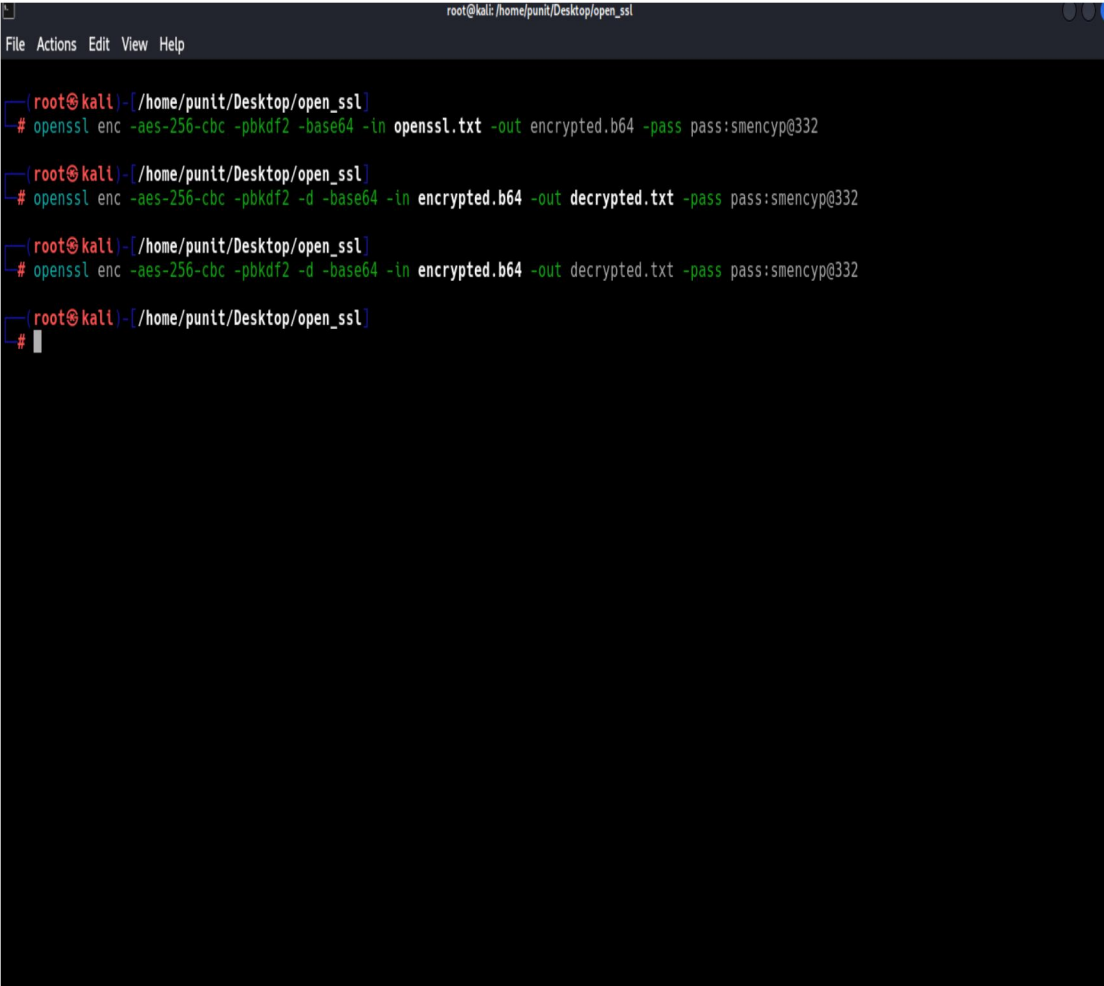
The screenshot shows a terminal window titled 'root@kali: /home/punit/Desktop/open_ssl'. The terminal displays the following commands and their outputs:

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

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
```



```
root@kali: /home/punit/Desktop/open_ssl  
File Actions Edit View Help  
root@kali:~/home/punit/Desktop/open_ssl  
# openssl enc -aes-256-cbc -pbkdf2 -base64 -in openssl.txt -out encrypted.b64 -pass pass:smencyp@332  
root@kali:~/home/punit/Desktop/open_ssl  
# openssl enc -aes-256-cbc -pbkdf2 -d -base64 -in encrypted.b64 -out decrypted.txt -pass pass:smencyp@332  
root@kali:~/home/punit/Desktop/open_ssl  
# openssl enc -aes-256-cbc -pbkdf2 -d -base64 -in encrypted.b64 -out decrypted.txt -pass pass:smencyp@332  
root@kali:~/home/punit/Desktop/open_ssl  
#
```