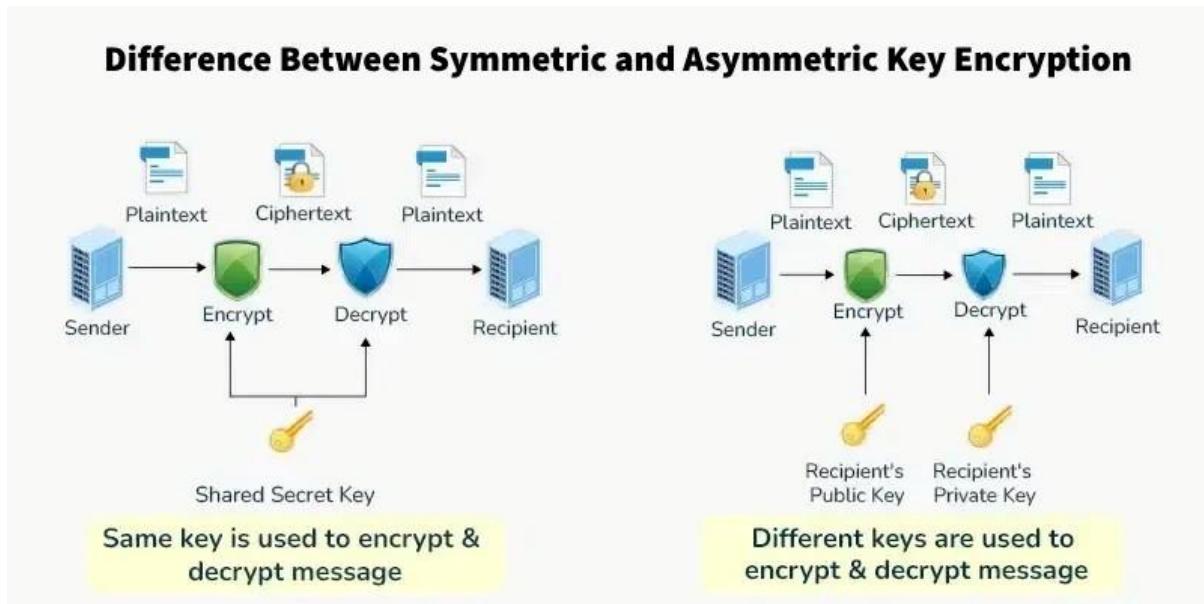


# TASK 6

## ENCRYPTION:

Encryption is the most common basic concept of cryptography, to protecting the integrity of the information from wrong hands. There are two types of techniques in encryption they are,

- ❖ Symmetric encryption
- ❖ Asymmetric encryption



## SYMMETRIC ENCRYPTION:

In this type of encryption same key will be used for the both encryption and decryption the information. It also requires a secure method to transfer the key between them, it fast but less secure.

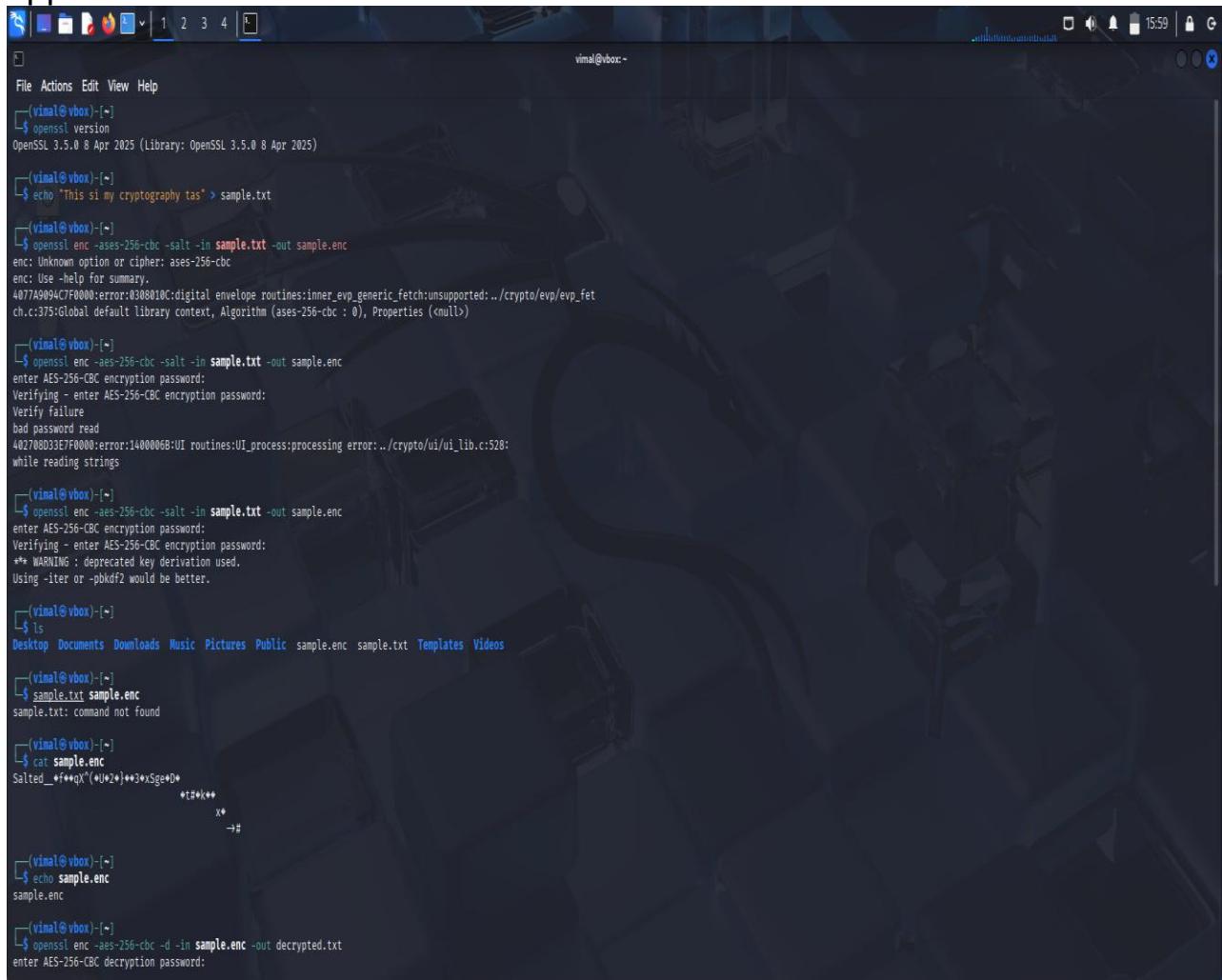
- It uses one key for both encryption and decryption.
- Faster and more efficient for large amounts of data.
- Requires a secure method to share the key between sender and receiver.
- Common algorithms include AES, DES, Blowfish.
- It is used in file encryption, VPNs, and secure data storage.

## ASYMMETRIC ENCRYPTION:

Asymmetric encryption is the most common cryptographic encryption, here one key is used to encrypted the message and it is pendent and it will be share among the users called public key. Another key is used decrypt the message and it is called private key.

- It uses two keys a public key for encryption and a private key for decryption.
- More secure but slower than symmetric encryption.
- No need to share the private key, reducing the risk of exposure.
- Common algorithms include RSA, ECC, Diffie-Hellman.
- It is used in digital signatures, SSL/TLS, and secure email communication.

I have done this encryption and decryption process with openssl application:



```

File Actions Edit View Help
(vimal@vbox) [~]
$ openssl version
OpenSSL 3.5.0 8 Apr 2025 (Library: OpenSSL 3.5.0 8 Apr 2025)

(vimal@vbox) [~]
$ echo "This is my cryptography test" > sample.txt

(vimal@vbox) [~]
$ openssl enc -aes-256-cbc -salt -in sample.txt -out sample.enc
enc: Unknown option or cipher: aes-256-cbc
enc: Use -help for summary.
4077A9094C7F0000:error:0308010C:digital envelope routines:inner_evp_generic_fetch:unsupported:../crypto/evp/evp_fet
ch.c:375:Global default library context, Algorithm (aes-256-cbc : 0), Properties (<null>)

(vimal@vbox) [~]
$ openssl enc -aes-256-cbc -salt -in sample.txt -out sample.enc
enter AES-256-CBC encryption password:
Verifying - enter AES-256-CBC encryption password:
Verify failure
bad password read
4027080D33E7F0000:error:1400006B:UI routines:UI_process:processing error:../crypto/ui/ui_lib.c:528:
while reading strings

(vimal@vbox) [~]
$ openssl enc -aes-256-cbc -salt -in sample.txt -out sample.enc
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.

(vimal@vbox) [~]
$ ls
Desktop Documents Downloads Music Pictures Public sample.enc sample.txt Templates Videos

(vimal@vbox) [~]
$ sample.txt sample.enc
sample.txt: command not found

(vimal@vbox) [~]
$ cat sample.enc
Salted__4f44qX'(u)02a)*+3*xSge*D*
          *tB*k**
          X#
          →#

(vimal@vbox) [~]
$ echo sample.enc
sample.enc

(vimal@vbox) [~]
$ openssl enc -aes-256-cbc -d -in sample.enc -out decrypted.txt
enter AES-256-CBC decryption password:

```

I done the AES encryption decryption with the text “**this is cryptography text**” using the commands:

`openssl enc -aes-256-cbc -salt -in sample.txt -out sample.enc`

- |                              |                                       |
|------------------------------|---------------------------------------|
| ▪ <code>openssl enc</code>   | -It kick off the application          |
| ▪ <code>aes-256</code>       | - Advance Encryption system 256 bits  |
| ▪ <code>cbc</code>           | -cipher block chain mode              |
| ▪ <code>salt</code>          | -random text to manipulate the cipher |
| ▪ <code>in sample.txt</code> | -file                                 |

## CRYPER TEXT : cat sample.enc – to display the encrypted file

```
(vimal@vbox)~]$ sample.txt sample.enc
sample.txt: command not found

(vimal@vbox)~]$ cat sample.enc
Salted__*f*qX^(*U2*}**3*xSge*D*
*t#*k**+
x*
→#

(vimal@vbox)~]$ echo sample.enc
sample.enc

(vimal@vbox)~]$ openssl enc -aes-256-cbc -d -in sample.enc -out decrypted.txt
enter AES-256-CBC decryption password:
```

## DECRYPTION THE TEXT - openssl enc -aes-256-cbc -d -in sample.enc -out decrypted.txt

- d -in :decrypt
- out decrypted.txt: output file

```
File Actions Edit View Help
(vimal@vbox)~]$ openssl version
OpenSSL 3.5.0 8 Apr 2025 (Library: OpenSSL 3.5.0 8 Apr 2025)

(vimal@vbox)~]$ echo "This si my cryptography tas" > sample.txt

(vimal@vbox)~]$ openssl enc -ases-256-cbc -salt -in sample.txt -out sample.enc
enc: Unknown option or cipher: ases-256-cbc
enc: Use -help for summary.
4077A9094C7F0000:error:0308010C:digital envelope routines:inner_evp_generic_fetch:unsupported:../crypto/evp/evp_fet
ch.c:375:Global default library context, Algorithm (ases-256-cbc : 0), Properties (<null>)

(vimal@vbox)~]$ openssl enc -aes-256-cbc -salt -in sample.txt -out sample.enc
enter AES-256-CBC encryption password:
Verifying - enter AES-256-CBC encryption password:
Verify failure
bad password read
402708D33E7F0000:error:1400006B:UI routines:UI_process:processing error:../crypto/ui/ui_lib.c:528:
while reading strings

(vimal@vbox)~]$ openssl enc -aes-256-cbc -salt -in sample.txt -out sample.enc
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.
```

## Final result after the decryption:

```
(vimal@vbox)~]$ cat decrypted.txt
This si my cryptography tas
```