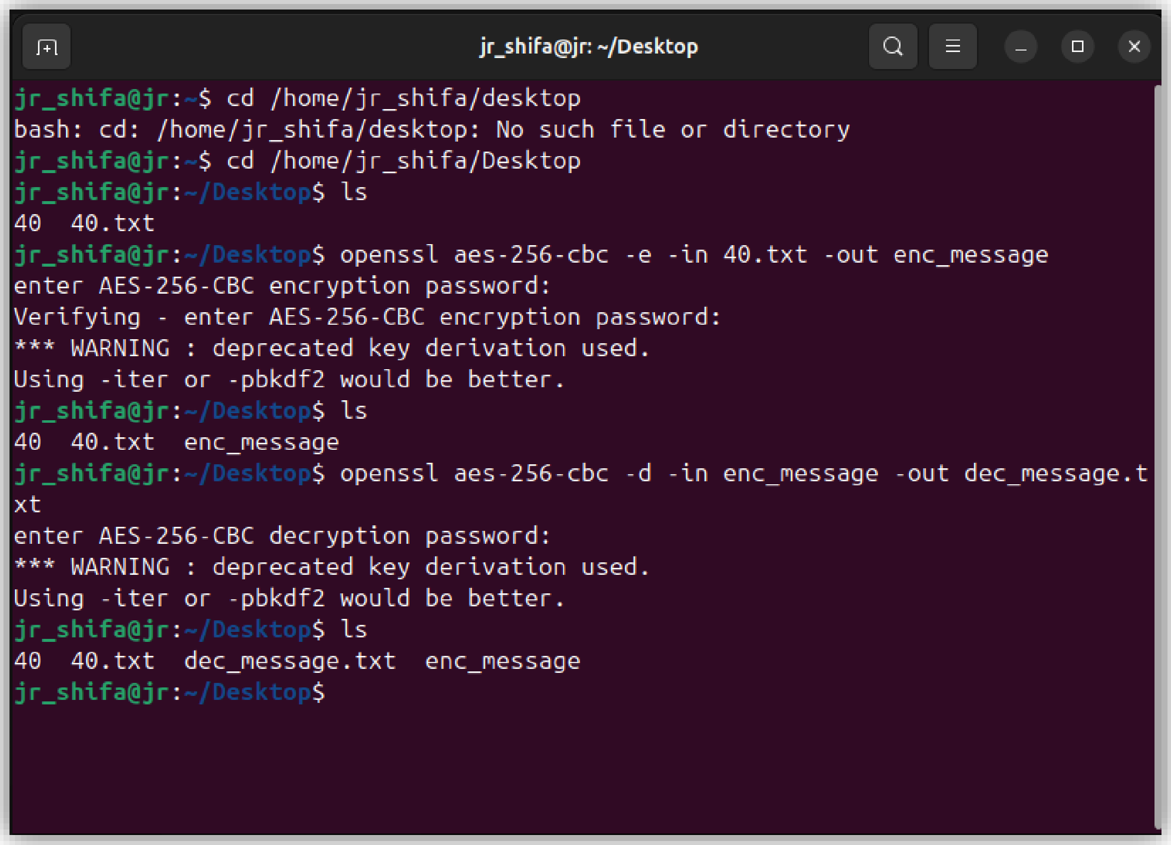
Session 3 Assignment (Practical Exercise)

Data Encryption Demonstration:



```
jr_shifa@jr:~$ cd /home/jr_shifa/desktop
bash: cd: /home/jr_shifa/desktop: No such file or directory
jr_shifa@jr:~$ cd /home/jr_shifa/Desktop
jr_shifa@jr:~/Desktop$ ls
40  40.txt
jr_shifa@jr:~/Desktop$ openssl aes-256-cbc -e -in 40.txt -out enc_message
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.
jr_shifa@jr:~/Desktop$ ls
40  40.txt  enc_message
jr_shifa@jr:~/Desktop$ openssl aes-256-cbc -d -in enc_message -out dec_message.t
xt
enter AES-256-CBC decryption password:
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
jr_shifa@jr:~/Desktop$ ls
40  40.txt  dec_message.txt  enc_message
jr_shifa@jr:~/Desktop$
```

1.Document the steps to encrypt and decrypt the file, including the commands or code used:

File List Command :

`/home/jr_shifa/Desktop` listed files and found `40.txt`.

Encryption Command:

used the `openssl` command to encrypt the file `40.txt: by commend`

```
openssl aes-256-cbc -e -in 40. txt -out enc_message
```

This command uses AES-256-CBC to encrypt the file `40.txt`, saving the encrypted output as `enc_message`.

OpenSSL ask to provide and verify a password for encryption.

OpenSSL ask to provide and verify a password for encryption.

WARNING

```
WARNING: deprecated key derivation used. Using -iter or -pbkdf2
would be better.
```

This warning means that the key derivation method used by default in OpenSSL is older and less secure than using the `-iter` or `-pbkdf2` options, which make it more difficult for attackers to guess the password.

Decryption Command:

decrypt the file `enc_message` by commend:

```
openssl aes-256-cbc -d -in enc_message -out dec_message.txt
```

However, you encountered some typos:

The command mistakenly used `aes-2S6-cbc` instead of `aes-256-cbc`.

Listed files to see if file encrypted or no:

enc_message (encrypted file)

dec_message (decrypted file)

2. Explain the significance of encryption in data security and how it protects data in transit and at rest.

The significance of encryption in protecting data in transit and at rest is critical for ensuring data privacy, confidentiality, and integrity.

Confidentiality: Encryption keeps data private by ensuring that only authorized parties can access it.

Integrity: Encryption helps maintain data accuracy by preventing unauthorized changes. Altered data will fail decryption, signaling potential tampering.

Here’s a closer look at how encryption enhances security in these two states:

1. Protecting Data in Transit

Definition: Data in transit refers to data actively moving through a network, such as when it’s sent over the internet, between devices, or across cloud environments.

Risks: Data in transit is especially vulnerable to interception by attackers, who may use techniques like “man-in-the-middle” attacks to eavesdrop or capture sensitive information.

Encryption’s Role: Encrypting data in transit ensures that even if someone intercepts the data, they cannot read or understand it without the decryption key. Protocols like SSL/TLS (used in HTTPS) encrypt data between a client and a server, protecting sensitive information like passwords, personal information, and financial data.

Example: When you enter credit card information on an e-commerce site, encryption ensures that this information is secure as it travels from your device to the merchant’s server.

2. Protecting Data at Rest

Definition: Data at rest refers to data that is stored on a device or in a system, such as on a hard drive, in a database, or in cloud storage.

Risks: Data at rest is a target for attacks, especially if attackers gain unauthorized access to the storage device or system. Without encryption, they could read or extract this data directly.

Encryption’s Role: Encrypting data at rest ensures that even if an attacker accesses the storage system, they cannot read or use the data without the correct decryption key. It safeguards stored information, preventing data breaches and unauthorized access.

Example: Many organizations encrypt entire databases, so even if someone physically steals a device or accesses the system illegally, they cannot read sensitive data like customer records or financial documents.