Mariam Mostafa Amin Mostafa

```python
from cryptography.fernet import Fernet

# Generate a key
key = Fernet.generate_key()
cipher_suite = Fernet(key)

# Save the key to a file (important for decrypting later)
with open("secret.key", "wb") as key_file:
    key_file.write(key)

```

For encryption of data to occur, a secret key should be created first. The Fernet class from cryptography.fernet implements symmetric encryption. In other words, the same key is used for encryption and decryption.

Here, we generate a key with Fernet.generate_key(). Saving this key to a file ensures we can decrypt the data later.

Create a sample file :

```python
with open("sample_data.txt", "w") as file:
    file.write("This is some sensitive data that needs encryption.")
```

Read the file contents and encrypt:

```
with open("sample_data.txt", "rb") as file:
    file_data = file.read()

# Encrypt the data
encrypted_data = cipher_suite.encrypt(file_data)

# Save the encrypted data to a file
with open("sample_data_encrypted.txt", "wb") as encrypted_file:
    encrypted_file.write(encrypted_data)
```

The file contents are read in binary mode ("rb"). We then use cipher_suite.encrypt(file_data) to encrypt the data and save it to a new file, sample_data_encrypted.txt.

Decrypt the Data File:

Load the key:

```
with open("secret.key", "rb") as key_file:
    key = key_file.read()
cipher_suite = Fernet(key)
```

Read the encrypted file and decrypt:

```
with open("sample_data_encrypted.txt", "rb") as encrypted_file:
    encrypted_data = encrypted_file.read()

# Decrypt the data
decrypted_data = cipher_suite.decrypt(encrypted_data)

# Save the decrypted data to a file
with open("sample_data_decrypted.txt", "wb") as decrypted_file:
    decrypted_file.write(decrypted_data)
```

The key is then reloaded from the file to ensure we're using the correct one. Now, the encrypted file is decrypted by using cipher_suite.decrypt(encrypted_data), and store this decrypted data in sample_data_decrypted.txt.

Significance of Encryption in Data Security

Explanation: Encryption plays a critical role in protecting data, both in transit and at rest:

Data in Transit:

When data is sent across a network-over the internet, for example-it is considered intercepted by unauthorized parties. The encryption of data in transit will ensure that even if it gets intercepted, the data cannot be read without a decryption key.

Data at Rest:

It is all kinds of data that reside on the disk-hard drive or even cloud-based-and yet could still be accessible to some unauthorized person. Encrypting this at rest, unauthorized access to read the data is prevented through a decryption key.