# FORM AUTHENTICATION (LAB DA – 4)

**SUBMITTED BY**

**GAURAV KUMAR SINGH (19BCE2119)**

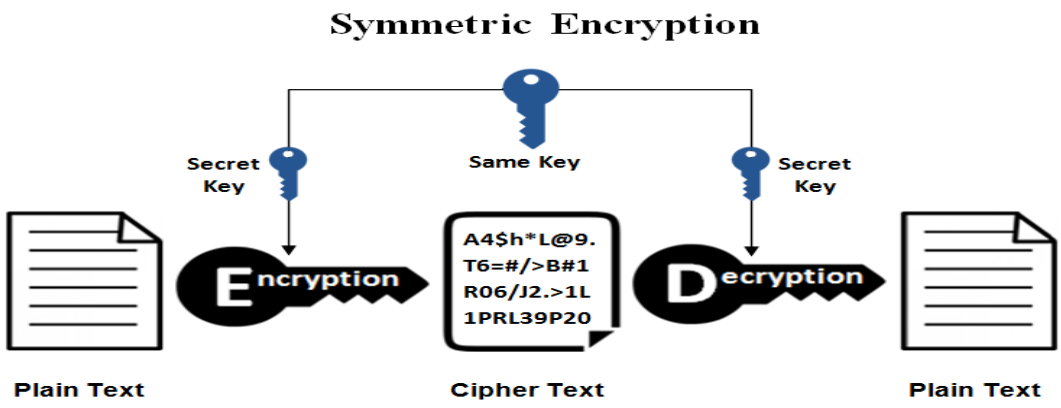
**AGARWAL CHIRAG SANJAY (19BCI0202)**

**PENKE LOHITH SASI ANVESH (19BCE2069)**

## INTRODUCTION ABOUT SYMMETRIC CRYPTOGRAPHY:

Symmetric cryptography, can also be referred as secret key cryptography, it is the used to single shared secret to encrypted information between two parties. Ciphers in this category are called symmetric because you use the same key to encrypt and to decrypt the data. In simple terms, the sender encrypts data using a password, and the recipient must know that password to access the data.

Symmetric encryption is a two-way process. With a block of plaintext and a given key, symmetric ciphers will always produce the same ciphertext. Likewise, using that same key on that block of ciphertext will always produce the original plaintext. Symmetric encryption is useful for protecting data between parties with an established shared key and frequently, is used to store confidential data. For example, ASP.NET uses 3DES to encrypt cookie data for a forms authentication ticket.




**FERNET SYMMETRIC ENCRYPTION USING CRYPTOGRAPHY MODULE IN PYTHON:**

Cryptography is the practice of securing useful information while transmitting from one computer to another or storing data on a computer. Cryptography deals with the encryption of plaintext into ciphertext and decryption of ciphertext into plaintext. Python supports a cryptography package that helps to encrypt and decrypt data. The fernet module of the cryptography package has inbuilt functions for the generation of the key, encryption of plaintext into ciphertext, and decryption of ciphertext into plaintext using the encrypt and decrypt methods respectively

Fernet guarantees that a message encrypted using it cannot be manipulated or read without the key. Fernet is an implementation of symmetric "which is also known as secret key" authenticated cryptography.

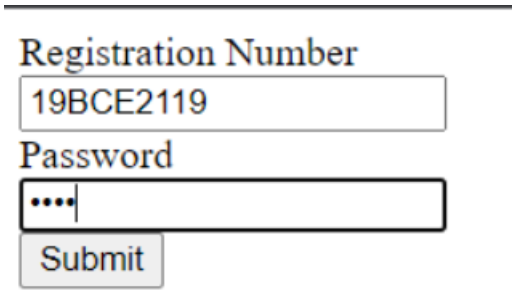
**For example**,

A “C” named organization uses Fernet key to encrypt passwords in the connection configuration and the variable configuration. It guarantees that a password encrypted using it cannot be manipulated or read without the key. Fernet is an implementation of symmetric (also known as “secret key”) authenticated cryptography.

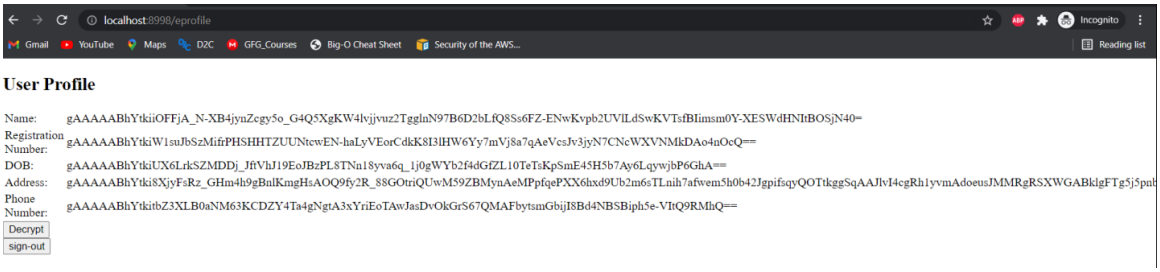
The first time “C” organization is started, the **C.cfg file** is generated with the default configuration and the unique Fernet key. The key is saved to option **fernet_key** of section. After, we need to generate a new fernet key you can use the code snippet.

Later, rotate encryption keys by Once connection credentials and variables have been encrypted using a fernet key, changing the key will cause decryption of existing credentials to fail. So, rotate the fernet key without invalidating existing encrypted values, export the new key to the **fernet_key** setting, run **C rotate-fernet-key**, and then drop the original key from **fernet_key**.

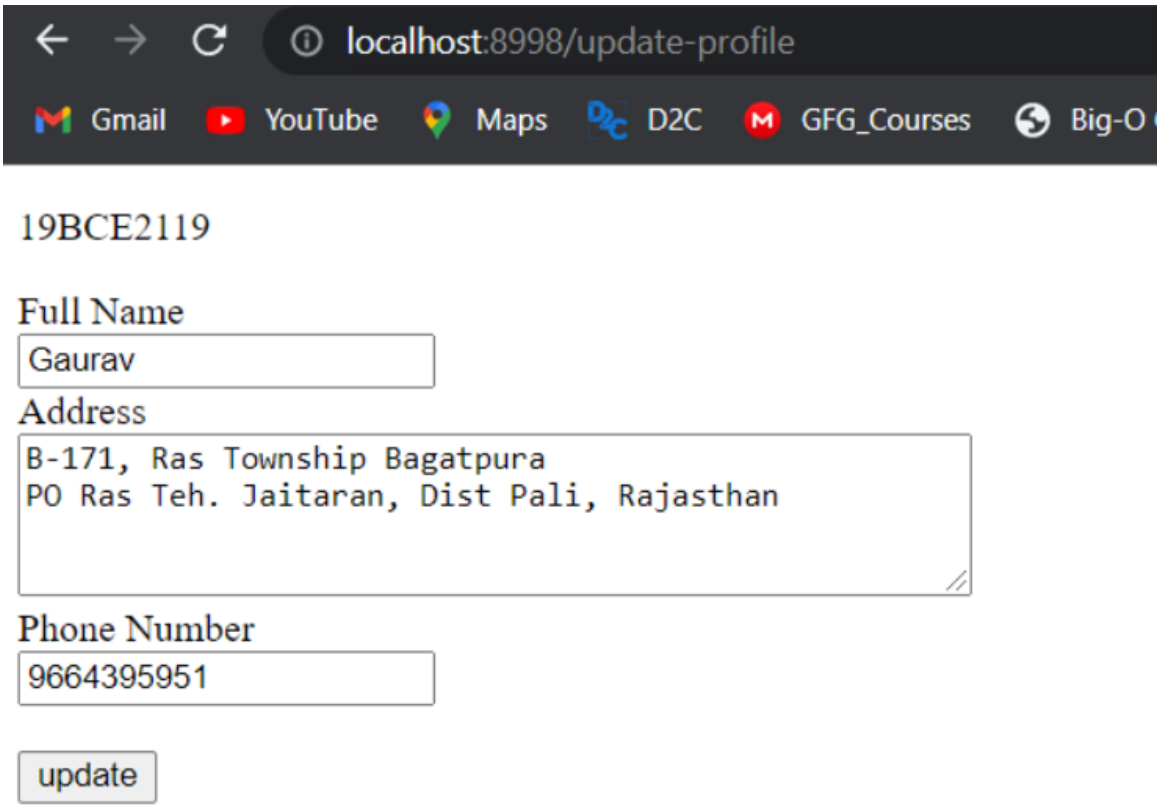
***Front End:***



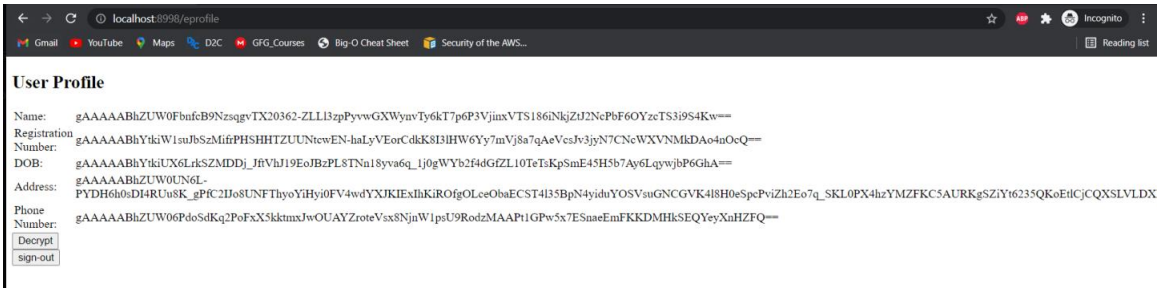
## ***Frontend On Encryption:***



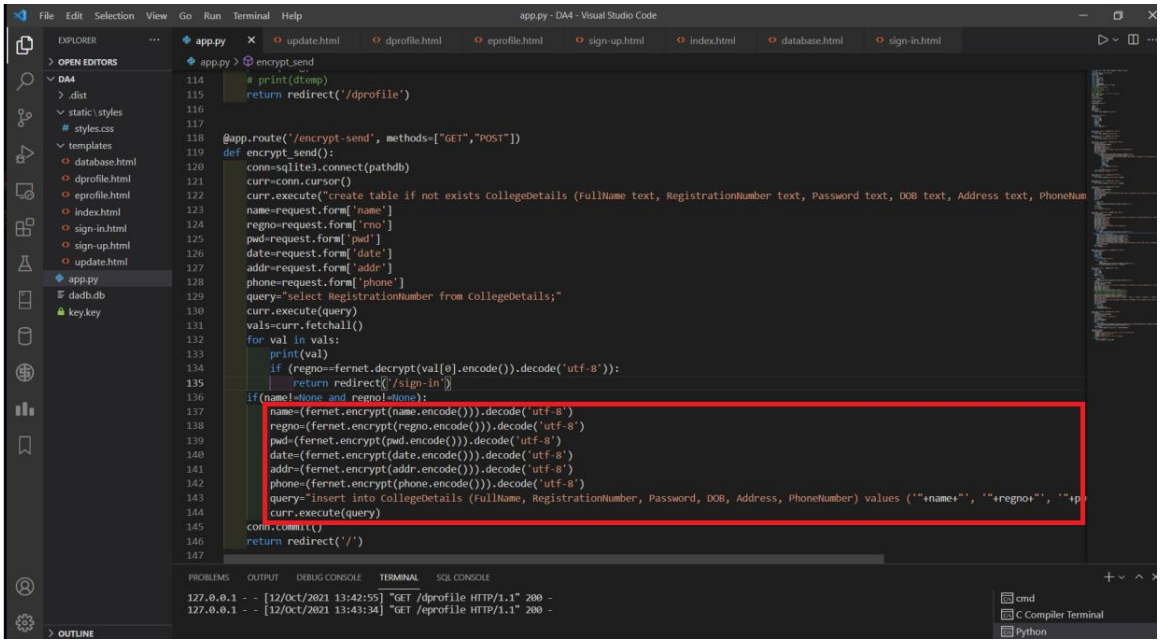

## ***Update Message's***



## ***Updated Encrypted Frontend:***

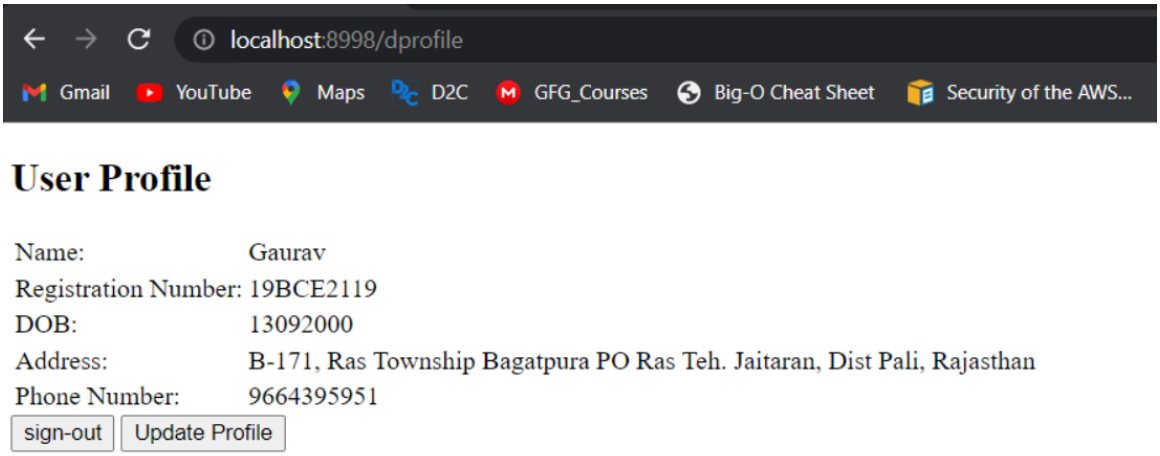



## ***Encryption Code Snapshot:***

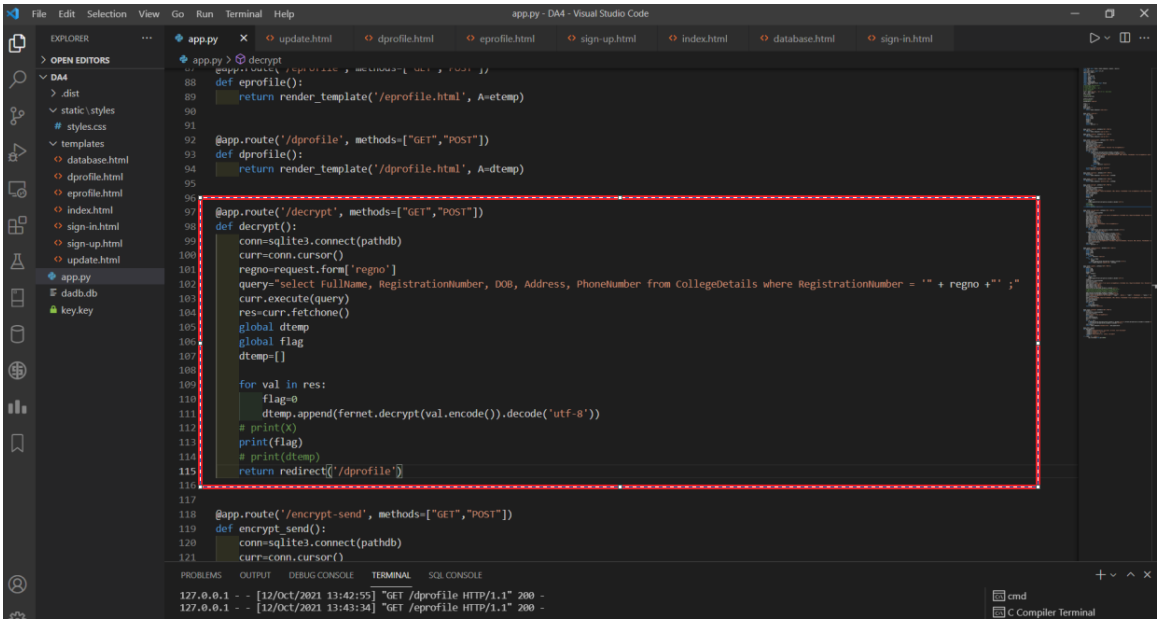


## ***Decrypted User Profile:***


## ***Backend Decryption Display of Updated Message:***



## ***Backend Decryption Code :***




**GITHUB URL :** https://github.com/Gaurav1020/Form-data-encyption-and-decryption

THANK YOU