Step 1: Load the program on google colab.

Step 2: Upload the dataset 'network_data.csv' on the colab

Step 3: First run the code for training the Ml model.

Step 4: Now, you can run the code for dynamic encryption.

You can test the code for various situations by changing the last input line.

Some sample inputs:

encryption_system([100, 10], rf_model)

encryption_system([80, 5], rf_model)

encryption_system([260, 55], rf_model)

encryption_system([220, 22], rf_model)

Step 5: Enter the text which you want to encrypt

Output:

Normal Scenario:

```
encryption_system([100, 15], rf_model)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:493
  warnings.warn(
Enter your message for encryption: Hello
Network Status: Normal
Original message: Hello
Encrypted message: 2e70290dc6abcdde2b3931ab68b28a55
Decrypted message: Hello
```

Attack Scenario:

```
[ ]     print("Encrypted message:", ciphertext.hex())

        if(network_status == 1):
            print("Tag:", tag.hex())
            print("Salt:", salt.hex())

        print("Decrypted message:", plaintext.decode())


    #Input the network parameters
    encryption_system([300, 25], rf_model)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:493:
  warnings.warn(
Enter your message for encryption: Hello
Network Status: Under Attack
Original message: Hello
Encrypted message: d9fe2a86771a1a6ac9c02495e04a
Tag: 2706eb640c30f3682831d7ddc8f63ca4
Salt: 87cf70786b1230d93e25a90069f0cbb5
Decrypted message: Hello
```