RSA Public-Key Encryption and Signature Lab

SEED 2.0

Contents

[Task 1 2](#_Toc146983367)

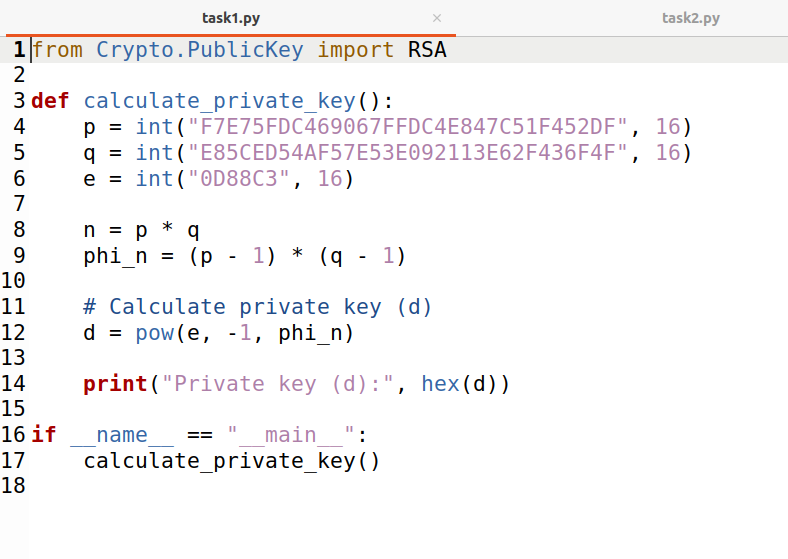
[Task 2 3](#_Toc146983368)

[Task 3 4](#_Toc146983369)

[Task 4 5](#_Toc146983370)

# Task 1

This code calculates the private key (d) based on the provided hexadecimal values for p, q, and e. It follows the RSA key generation algorithm to calculate the private exponent d using the given prime numbers p and q, and the public exponent e. The private key (d) is printed in hexadecimal form.

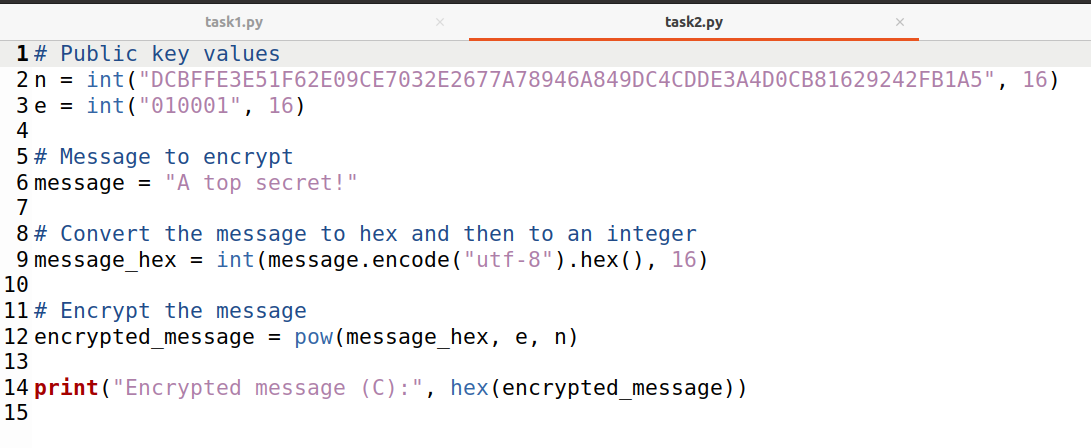


Following is the calculated private key.



# Task 2

This code demonstrates how to encrypt a message using the provided public key (n, e) in an RSA encryption scheme. It converts the message to hex, then to an integer, and encrypts it using the RSA encryption formula. The encrypted message (ciphertext) is printed in hexadecimal form.



Following is the encrypted message.



# Task 3

This code demonstrates how to decrypt a message using the provided public/private key (n, d) in an RSA encryption scheme. It uses the RSA decryption formula to decrypt the ciphertext and obtain the original message (plaintext). The decrypted message is then printed.

A screenshot of a computer

Description automatically generated

Following is the decrypted message of the provided encrypted message.



# Task 4

This code demonstrates how to generate signatures for messages using the provided public/private key (n, d) in an RSA encryption scheme. It calculates the signature for both the original and a modified message by applying the RSA signing operation. The resulting signatures for both messages are printed in hexadecimal form.

A screenshot of a computer

Description automatically generated

It is clearly observable that only changing the value by only one character the signature has been changed. This tells that the signature is unique to the message provided.

