**Data Security and Cryptology – Project Report**

**Project Topic:**   
An Application for secure email exchange: encryption-decryption with Cast-128 in CBC mode, secret key delivery with RSA + EL-Gamal signature.

**Components:**

ENCRYPTION :

1. Check KEY – MUST be between 40 and 128 and divided by 8
2. Text –partition the text to 64bit each block,

If the last block is less than 64bit then add ‘0’ to all the bits to be 64 bit

1. In CBC mode loop on the blocks as hex and encrypt each block using Cast128
2. Concatenation all ciphertext to one string and return it.

DECRYPTION:

1. Check KEY – MUST be between 40 and 128 and divided by 8
2. Text –partition the text to 64bit each block
3. Decrypt according to cast128 and CBC mode

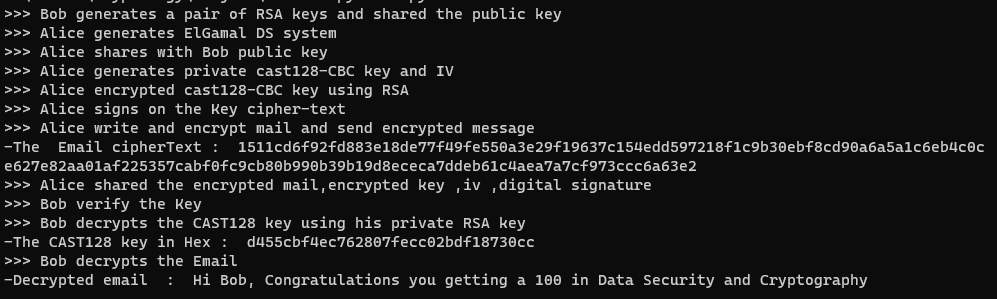
Key Exchange: For resolving key exchange in the symmetric algorithm, we use an asymmetric **RSA** algorithm by encrypting the cast128 key with the public key received from the person who wants to decrypt it.

Digital signature: In order to verify the identity of the person who send the cast128 key we anticipate that the key will be signed, in our case we use the **El-Gamal** algorithm shared by authorized authority.

**The Project Flow:**  
Alice wants to send a secure email to Bob.

1. Alice generates El-Gamal keys (private key, secret key)
2. Bob Generate RSA keys
3. Alice and Bob shared their public key with each other.
4. Alice use RSA and bob public key to encrypt the CAST128 key.
5. Alice signs on the cipher text of the cast128 key.
6. Bob receives the cipher text of the CAST128 key and verifies it.
7. Bob decrypts cipher text using RSA and his private key.
8. Alice encrypts the message using Cast128 CBC mode
9. Alice send the message to Bob.
10. Bob use Cast128 key and decrypt the message .

Example Output :



**Conclusions:**   
the algorithm implemented employs a robust combination of cryptographic techniques to ensure secure communication and data protection.

The CAST-128 encryption algorithm in CBC mode offers strong confidentiality by encrypting data blocks with a secret key.

The RSA encryption scheme, used for secret key delivery, provides secure key exchange through the encryption of a symmetric key. Additionally, the ElGamal signature scheme adds an extra layer of authentication and integrity verification, ensuring the legitimacy of the sender.

Together, these techniques provide a comprehensive approach to encryption, decryption, and secure key delivery, bolstering the overall security of the system.

**References:**

* [rfc2144.txt.pdf (rfc-editor.org)](https://www.rfc-editor.org/rfc/pdfrfc/rfc2144.txt.pdf)
* [RSA Algorithm: Theory and Implementation in Python – AskPython](https://www.askpython.com/python/examples/rsa-algorithm-in-python)
* [- awnonbhowmik/RSA-Python: The RSA algorithm coded in Python](https://github.com/awnonbhowmik/RSA-Python)
* [PowerPoint-esitys (hut.fi)](http://www.tcs.hut.fi/Studies/T-79.5502/2006SPR/presentations/ElGamal.pdf)
* [GitHub - amritesh-dasari/ElGamal-signature-scheme: Verification of the ElGamal Signature Scheme Cryptography](https://github.com/amritesh-dasari/ElGamal-signature-scheme)