# IMAGE ENCRYPTION USING RSA ALGORITHM

By – Sarthak Jain(2K19/ME/216)

Abhinav Mittolia(2K19/EN/003)

# WHAT IS RSA ENCRYPTION?

RSA is an algorithm used in the modern computer environment to encrypt and decrypt data in transform.

The RSA algorithm is an asymmetric cryptographic algorithm. Asymmetric cryptosystem means two different keys are used in the encryption and decryption.

In the two keys, one key is used for encryption and the second key is used for decryption.

This RSA algorithm is also called ‘public key cryptography’ because one of the secret keys can be given to the server(or sender) to retrieve encrypted data which means ‘public’. The other key must be kept private.

# STEPS INVOLVED IN RSA TECHNIQUE

Key Generation
(Public and private)

Encryption

Decryption

# KEY GENERATION

The key generation is the first step of RSA algorithm. The RSA involves a public key and a private key. On those keys the public key can be know to everyone and it is used for encrypting the message. Messages encrypted with the public key can be decrypted using the private key. The keys for the RSA algorithm is generated by the following steps :

1) First choose the two distinct prime numbers p and q.

2) For security purposes, the integer p and q should be chosen, and it should be of similar bit-length. Prime integers can be efficiently found by primality testing.

3) Then compute the n value, $n = p * q$.

4) n is used as the modulo operation for both encryption and decryption. Its length, usually expressed in bits, is the key length.

5) Compute $\varphi(n) = \varphi(p) * \varphi(q) = (p - 1)(q - 1) = n - (p + q - 1)$, where $\varphi$ is Euler's totient function. This value is kept private.

6) Choose an integer e such that $1 < e < \varphi(n)$ and $\gcd(e, \varphi(n)) = 1$; i.e., e and $\varphi(n)$ are co-prime. e is released as the public key. e has a short bit-length and its small Hamming weight results in more efficient encryption. However, much smaller values of e have been shown to be less secure in some settings.

7) Determine d as $d \equiv e^{(-1)} \pmod{\varphi(n)}$; i.e., d is the modular multiplicative inverse of e (modulo $\varphi(n)$). This is stated as - Solve the d given $d \cdot e \equiv 1 \pmod{\varphi(n)}$. This is computed using extended Euclidean algorithm. Another method which can be used is the 'hit and trial' method, which is a bit less efficient to the former. Both methods have been tried in this project.

8) d value is keep as the private key.

The public key consists of the modulus n and the public key e. The private key have the modulus n and the private key d, and it keep in secret. p, q, and $\varphi(n)$ values are keep in secret, because they can be used to calculate d.

# ENCRYPTION

This section explains encryption and decryption in RSA by an example:

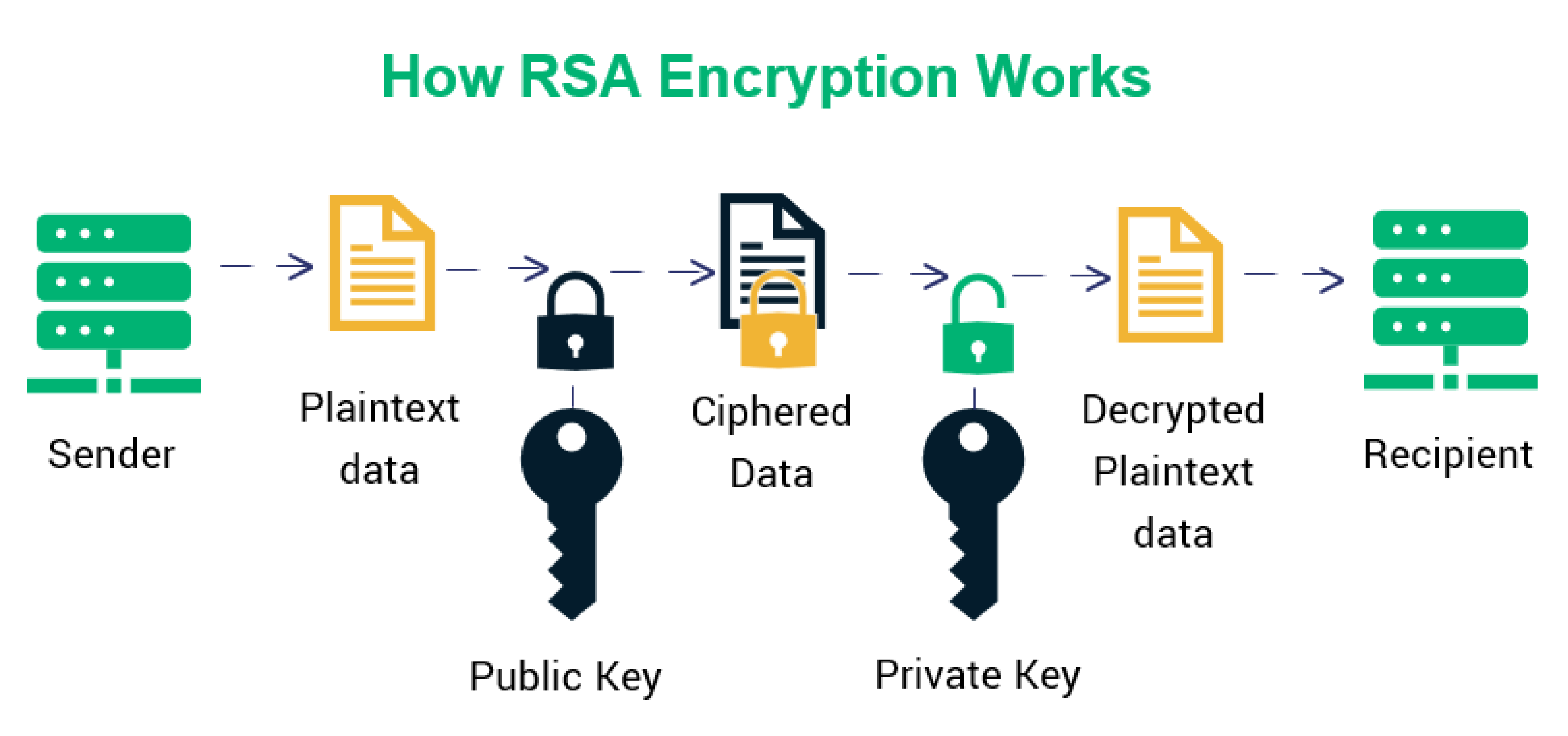
- Alice transmits her public key (n, e) to Bob and keeps the private key d secret. Bob then wishes to send a message M to Alice.
- So, he first turns M into an integer m, such that $0 < m < n$. Then it compute the cipher text c.
- This can be done efficiently, even if the numbers are 500 - bit numbers, it is using the Modular exponentiation. Bob then transmits c to Alice.

Encryption c = (msg ^ e) mod n

# DECRYPTION

- Alice can recover m from c by using her private key exponent d via computing. Given m, she can recover the original message M by reversing the padding scheme.

Decryption $m = (c \wedge d) \bmod n$



# SCHEMATIC PROCESS FLOW IN RSA

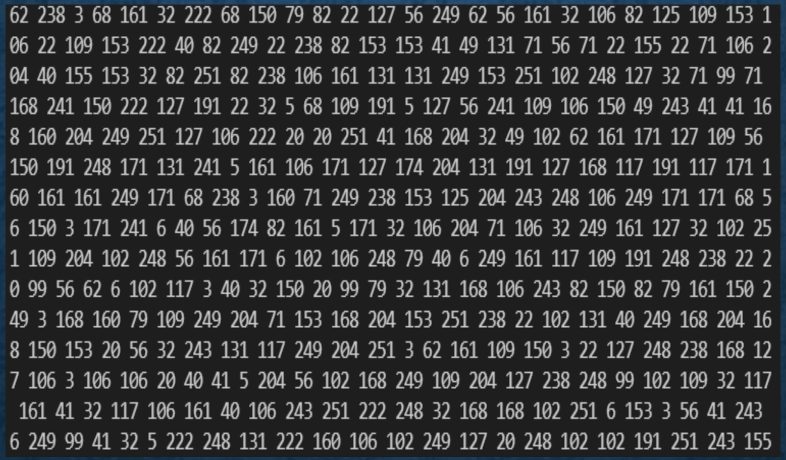
# ENCRYPTING AN IMAGE USING RSA

Now, coming to our project demo, which encrypts an image using the RSA algorithm.

1) The language used in our application is Python.
2) Python File handling can be easily used to open, read and write to an image file.
3) The opened image file can be converted into an array of bytes, i.e. [R1, G1, B1, R2, G2, B2……and so on] starting from RGB data of the top-left pixel of the image. This can be done using the bytearray() method. It stores data in the range [0, 255].
4) The values stored in every index of the obtained bytearray goes through the algorithm and gets converted into a cipher text c, which corrupts the image, making it inaccessible.
5) The image can only be restored to its original state(plain text) by decrypting using the client's private key.

# RESULT

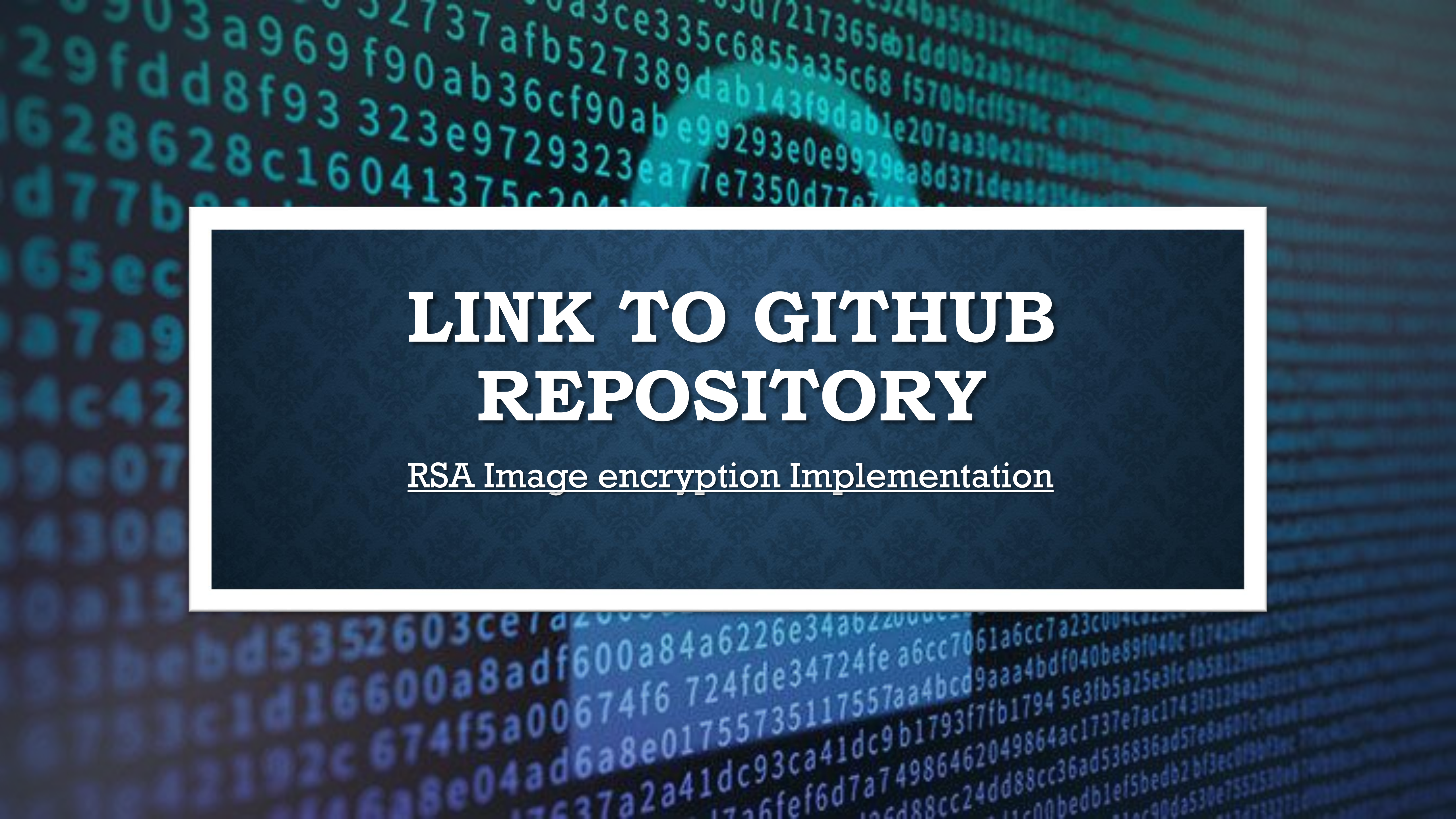
Image before encryption



```
62 238 3 68 161 32 222 68 150 79 82 22 127 56 249 62 56 161 32 106 82 125 109 153 1
06 22 109 153 222 40 82 249 22 238 82 153 153 41 49 131 71 56 71 22 155 22 71 106 2
04 40 155 153 32 82 251 82 238 106 161 131 131 249 153 251 102 248 127 32 71 99 71
168 241 150 222 127 191 22 32 5 68 109 191 5 127 56 241 109 106 150 49 243 41 41 16
8 160 204 249 251 127 106 222 20 20 251 41 168 204 32 49 102 62 161 171 127 109 56
150 191 248 171 131 241 5 161 106 171 127 174 204 131 191 127 168 117 191 117 171 1
60 161 161 249 171 68 238 3 160 71 249 238 153 125 204 243 248 106 249 171 171 68 5
6 150 3 171 241 6 40 56 174 82 161 5 171 32 106 204 71 106 32 249 161 127 32 102 25
1 109 204 102 248 56 161 171 6 102 106 248 79 40 6 249 161 117 109 191 248 238 22 2
0 99 56 62 6 102 117 3 40 32 150 20 99 79 32 131 168 106 243 82 150 82 79 161 150 2
49 3 168 160 79 109 249 204 71 153 168 204 153 251 238 22 102 131 40 249 168 204 16
8 150 153 20 56 32 243 131 117 249 204 251 3 62 161 109 150 3 22 127 248 238 168 12
7 106 3 106 106 20 40 41 5 204 56 102 168 249 109 204 127 238 248 99 102 109 32 117
 161 41 32 117 106 161 40 106 243 251 222 248 32 168 168 102 251 6 153 3 56 41 243
6 249 99 41 32 5 222 248 131 222 160 106 102 249 127 20 248 102 102 191 251 243 155
```

Encrypted cipher text using RSA technique

Decrypted Image



# LINK TO GITHUB REPOSITORY

RSA Image encryption Implementation

# MERITS OF USING IMAGE ENCRYPTION

- Peace of Mind
- Identity Theft Protection
- Safe Decommissioning of Computer
- Unauthorized Access Protection
- Compliance with Data Protection Acts

# DEMERITS OF USING ENCRYPTION

There are very limited demerits of data encryption. They are listed below.

- Cryptography is a very complex technology.
- One big disadvantage of encryption related with keys is that the security of data becomes the security of the encryption key. The data is lost effectively if one lose the keys.
- Encrypting data and creating the keys necessary to encrypt and decrypt the data is computationally expensive.

# CONCLUSION

01

In the digital world, the security of images has become more important as open network communications have increased rapidly.

02

This image encryption algorithm is efficient and highly secure with high level data encryption and less computation.

03

Hence, it is concluded that this technique is a great method for image encryption and gives security when the client’s network is public.

# THANK YOU!