MODULE-2: Symmetric and Asymmetric key Cryptography

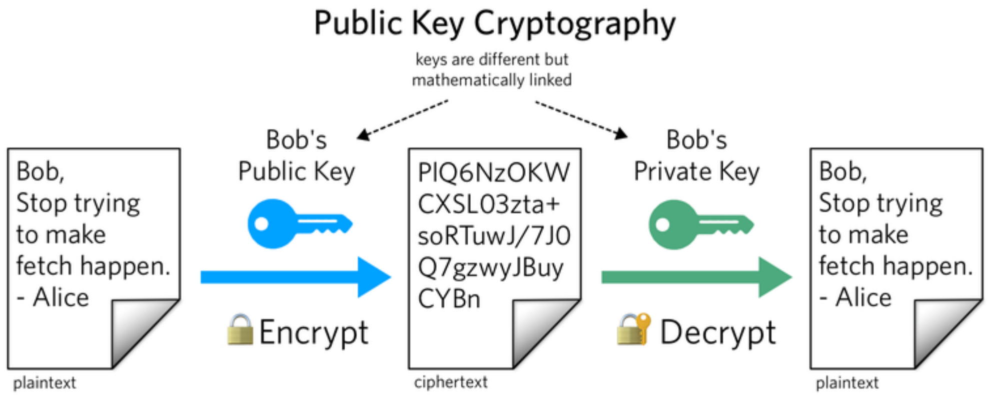
VIT | Vidyalankar Institute of Technology

Accredited A+ by NAAC

Prepared by Prof. Amit K. Nerurkar

Module 2 Public Key Cryptography

Principles of public key cryptosystems-


Public Key Cryptography
keys are different but mathematically linked
Bob's Public Key
Encrypt
Bob's Private Key
Decrypt
Bob, Stop trying to make fetch happen. - Alice
plaintext
PIQ6NzOKW CXSL03zta+ soRTuwJ/7J0 Q7gzwyJBuy CYBn
ciphertext
Bob, Stop trying to make fetch happen. - Alice
plaintext

Any public key cryptographic algorithm has six elements as follow:

Plain Text
This is a readable message which is given as input to the algorithm. In a public key algorithm, the plain text is encrypted in blocks.

Encryption Algorithm
The encryption algorithm is implemented on the plain text which performs several transformations on plain text.

Public and Private keys
These are the set of keys among which if one is used for encryption the other would be used for decryption. The transformation of plain text by encryption algorithm depends on the key chosen from the set to encrypt the plain text.

Cipher Text
This is the output of encryption algorithm. The generated cipher text totally depends on the key selected from the set of the public and private key. Both of these keys, one at a time with plain text would produce different cipher texts.

Decryption Algorithm
This would accept the output of the encryption algorithm i.e. the cipher text and will apply the related key to produce the original plain text.

The RSA algorithm

RSA algorithm is a public key encryption technique and is considered as the most secure way of encryption. It was invented by Rivest, Shamir and Adleman in year 1978 and hence name RSA algorithm.

The RSA algorithm is an asymmetric cryptography algorithm; this means that it uses a public key and a private key (i.e two different, mathematically linked keys). As their names suggest, a public key is shared publicly, while a private key is secret and must not be shared with anyone.

1. Generating the keys

1. Select two large prime numbers, x and y.

The prime numbers need to be large so that they will be difficult for someone to figure out.

2. Calculate

$n = x * y$

3. Calculate the totient function

$\phi(n)=(x-1)(y-1)$.

3. Select an integer e (Encryption Key), such that

e is co-prime to $\phi(n)$ and
$1<e<\phi(n)$.

4. The pair of numbers (e,n) makes up the public key.
5. Calculate d (Decryption Key) such that e.d =1 mod ϕ(n).
6. The pair (d,n) makes up the private key.

2. Encryption

Given a plaintext P, represented as a number, the ciphertext C is calculated as:

C = P^{e} mod n.

3. Decryption

Using the private key (n,d)(n,d), the plaintext can be found using:

P = C^{d} mod n

ElGamal Algorithm

ElGamal encryption is an public-key cryptosystem. It uses asymmetric key encryption for communicating between two parties and encrypting the message. This cryptosystem is based on the difficulty of finding discrete logarithm in a cyclic group that is even if we know ga and gk, it is extremely difficult to compute gak.

Suppose Alice wants to communicate to Bob.

Bob generates public and private key :

1. Bob chooses a very large number q and a cyclic group Fq.
2. From the cyclic group Fq, he choose any element g and an element a such that gcd(a, q) = 1.
3. Then he computes h = ga.
4. Bob publishes F, h = ga, q and g as his public key and retains a as private key.

Alice encrypts data using Bob's public key :

1. Alice selects an element k from cyclic group F such that gcd(k, q) = 1.
2. Then she computes p = gk and s = hk = gak.
3. She multiples s with M.
4. Then she sends (p, M*s) = (gk, M*s).

Bob decrypts the message :

1. Bob calculates s′ = pa = gak.
2. He divides M*s by s′ to obtain M as s = s′.

ElGamal Algorithm

Refer OneNote notes

Symmetric Key vs Asymmetric Key

Characteristic	Symmetric-Key Cryptography	Asymmetric-Key Cryptography
Key used for encryption/decryption.	Same key is used for encryption and decryption.	One key used for encryption and another, different key is used for decryption.
Speed of encryption/decryption	Very fast	Slower
size of resulting encrypted text.	Usually same as or less than the original clear text size.	More than the original clear text size.
Key agreement/exchange	A big problem	No problem at all.
Number of keys required as compared to the number of participants in the message exchange.	Equals about the square of the number of participants, so scalability is an issue.	Same as the number of participants, so scales up quite well.
Usage	Mainly for encryption and decryption (confidentiality), cannot be used for digital signatures (integrity and non-repudiation checks)	Can be used for encryption and decryption (confidentiality) as well as for digital signatures (integrity and non-repudiation checks)

References

1. https://www.twilio.com/blog/what-is-public-key-cryptography
2. https://binaryterms.com/public-key-cryptography.html
3. https://www.tutorialspoint.com/cryptography_with_python/cryptography_with_python_understanding_rsa_algorithm.htm
4. https://www.educative.io/edpresso/what-is-the-rsa-algorithm
5. https://www.geeksforgeeks.org/elgamal-encryption-algorithm/