A COMPARITIVE STUDY OF ASYMMETRIC KEY CRYPTOGRAPHY

Dr. Prashant P. Pittalia

Associate Professor, Department of Computer Science, Gujarat, India

ABSTRACT

Cryptography is use to secure the computer networks. Cryptography converts the plain text into cipher text such a way that it is impossible for intruders to read and understand the content. Asymmetric key cryptography is based on the public key and private key concepts to support the confidentiality and authentication services in computer networks. This paper describes how confidentiality and authentication service provides by asymmetric key algorithms. In this paper the discussion and comparison of a asymmetric key algorithms like RSA, Diffie-Hellman, ElGamal and ECC.

Keyword: - *RSA, Authentication, Public key, Private Key, Confidentiality*

1. INTRODUCTION

Today in global village the internet security is the challenging aspect and cryptography is used for it. Cryptography is the mathematical techniques of information security. It supports confidentiality, privacy, and data integrity and entity authentication. Cryptography systems classified into two main categories symmetric-key cryptography that use a symmetric key used by both sender and recipient and a public-key cryptography that use two keys, a public key known to everyone in the network and a private key that is used by only the recipient of the messages. Symmetric algorithms use the same secret key for both encryption of plaintext and decryption of cipher text. The keys may be same or there may be an easy transformation to go between the two keys. The key is shared between the two parties (sender and receiver) who want to communicate. Public-key cryptography (Asymmetric key cryptography), in which two separate keys are required. One is secret (or private) key and other one is public key. These two keys are mathematically linked. The public key is used to encrypt the plain text to provide confidentiality service or to decrypt the encrypted text to verify a digital signature. The private key is used to decrypt cipher text for confidentiality or to encrypt the message to provide the authentication service. The term "asymmetric" means to convert plaintext into cipher text the key is different than the key used for convert cipher text into plain text. In symmetric key algorithms same key is used for encryption and decryption. Public-key cryptography, or asymmetric cryptography, is a cryptographic system that uses pairs of keys: public keys which may be disseminated widely, and private keys which are known only to the owner. The generation of such keys depends on cryptographic algorithms based on mathematical problems to produce one-way functions. Effective security only requires keeping the private key private; the public key can be openly distributed without compromising security [1]. Robust authentication is also possible. A sender can combine a message with a private key to create a short digital signature on the message. Anyone with the corresponding public key can combine a message, a putative digital signature on it, and the known public key to verify whether the signature was valid, i.e. made by the owner of the corresponding private key.[2][3]

2. PUBLIC KEY CRYPTOGRAPHY

The main issue with the symmetric key algorithm is that it must required to share the secret key amongst the communicating parties and it may be possible that the intruder might comes to know about that key and do the malicious task. While in asymmetric cryptography public and private key concepts is used. Public key is shared amongst all the people and private key is kept with the user itself. It is very difficult for attacker to crack the message. Here in asymmetric key algorithms the very large prime numbers are used for mathematical operations.

The most important algorithm in asymmetric key algorithm is RSA. The Diffie-Hellman algorithm is used for mostly the key exchange.

3. RSA

It is widely used in the internet world to securely transfer the data with SSL and TLS protocol. In RSA, this asymmetry is based on the practical difficulty of the factorization of the product of two large prime numbers, the "factoring problem". The acronym RSA is made of the initial letters of the surnames of Ron Rivest, Adi Shamir, and Leonard Adleman, who first publicly described the algorithm in 1977. Clifford Cocks, an English mathematician working for the British intelligence agency Government Communications Headquarters (GCHQ), had developed an equivalent system in 1973, but this was not declassified until 1997.[4] A user of RSA creates and then publishes a public key based on two large prime numbers, along with an auxiliary value. The prime numbers must be kept secret. Anyone can use the public key to encrypt a message, but with currently published methods, and if the public key is large enough, only someone with knowledge of the prime numbers can decode the message feasibly [5]. Due to the large prime numbers it is very difficult to identify the prime factors. Also in RSA the key length is 1024 or 2048 bits. It is possible that attacker may crack the key with length of 1024, so mostly organization now a days use the 2048 bits key length. For explanation here I have taken a very small prime numbers, but in actual implementation it may be in size of 1024 or 2048 bits.

1. Select the two prime numbers
   P=3 & q =11
2. Find the product of two prime numbers
   n = p * q = 33
3. Find the value of Ø(n)
   Ø (n) = (p – 1) * (q – 1) = 20
4. Now choose any prime number d
   Lets say d = 7
5. Find the value of e such that d * e mod Ø(n) = 1
   i.e. 7 * e mod 20
   We select the e =3
6. Now we have two keys with us public key (e,n) and private key (d,n)
7. To encrypt the data the formula is $C = p^{e} \bmod n$
8. To decrypt the data the formula is $P = c^{d} \bmod n$

For the explanation we consider that we have to assign each alphabet a unique number starting with 1 for A, 2 for B and so on. Now to encrypt the message like “HELLO”, we may say that the value of corresponding character is 8, 5, 12, 12 and 15 respectively. The encryption and decryption is as follow.

Encryption Process

Symbol	**Numeric**	**P^3**	**P^3 mod 33**
H	8	512	17
E	5	125	26
L	12	1728	12
L	12	1728	12
O	15	3375	9

Decryption Process

C^7	**C^7 mod 33**	**Symbol**
17	8	H
26	5	E
12	12	L
12	12	L
9	15	O

Here the encryption key is (3, 33) and decryption key (7 , 33). Both the keys are different but we get the result on the other side.
It has been widely deployed in many standards such as PGP, SMIME, IPSEC and others. The government, military, banking and financial sector, public services, and e-commerce industry have extensively used it. RSA is very suitable for all levels of secrecy. It is recommended for top secret and secret information. In short, RSA is the choice of encryption for small amounts of data, which need extremely high level of secrecy.

4. Diffie-Hellman

Diffie-Hellman key is a method mostly used for exchanges the keys between the sender and the receiver. All users agree on global parameters:

- Large prime integer or polynomial q
  a being a primitive root mod q
- Each user generates their key
  Let's say user A –
  Chooses a secret key (number): $X_A < q$
  Compute their public key: $Y_A = a^{X_A} \bmod q$
  Let's say user B –
  Chooses a secret key (number): $X_B < q$
  Compute their public key: $Y_B = a^{X_B} \bmod q$
- Each user makes public that key Y_A, Y_B
- shared session key for users A & B is K_{AB}:
- To calculate the secret key K_{AB} by user A.
  $K_{AB} = (Y_B)^{X_A} \bmod q$
- To calculate the secret key K_{AB} by user B.
  $K_{AB} = (Y_A)^{X_B} \bmod q$

K_{AB} is used as session key in private-key encryption scheme between user A and user B. if A and B subsequently communicate, they will have the same key as before, unless they choose new public-keys. The personal keys X_A and X_B are used in the calculation of K_{AB}, have not been transmitted in a public network. Because it is a large and apparently random number, a potential hacker has almost no chance of correctly guessing x, even with the help of a powerful computer to conduct millions of trials. Users A and B can communicate privately over a public medium with an encryption method of their choice using the decryption key K_{AB}. This method could not support for the authentication. It may be vulnerable with the Man in the middle attack.

5. ElGamal

Alternative to RSA algorithm people may use ElGamal Algorithm for public key algorithm. Security of the ElGamal algorithm depends on the difficulty of computing discrete logs in a large prime modulus. ElGamal has the disadvantage that the cipher text is twice as long as the plaintext. But at the same time, it has the advantage that the same plaintext gives a different cipher text each time it is encrypted.
Let's say two user A and B.

User A chooses

- A large prime pA (200 - 300 digits),
- A primitive element α A modulo pA,
- A (possibly random) integer dA with $2 \leq dA \leq pA - 2$.
- A computes $\beta A \equiv \alpha A\ dA \pmod{pA}$.
- A's public key is (pA, α A, β A) and private key is dA.

User B encrypts a short message M (M < pA) and sends it to user A like below:

- User B chooses a random integer k (which he keeps secret).
- B computes $r \equiv \alpha A\ k \pmod{pA}$ and $t \equiv \beta A\ k\ M \pmod{pA}$ and then discards k.
- User B sends encrypted message (r, t) to User A.

- When user A receives the encrypted message (r, t), he/she decrypts (using private key dA) by computing t r−dA.
- Note tr−dA ≡ βA k M (α A k)−dA (mod pA)
  ≡ (α A dA) k M (α A k)−dA (mod pA)
  ≡ M (mod pA)

Even if intruders intercepts the cipher text (r, t), he/she cannot perform the calculation above because he/she doesn't know dA. β A ≡ α A dA (mod pA), so dA ≡ L αA (β A). Intruder can find dA if he/she can compute a discrete log in the large prime modulus pA, presumably a computation that is too difficult to be practical[6].

6. ECC (Elliptic Curve Cryptography)

Elliptic Curve cryptography, is the least used scheme because it is very difficult to understand the mathematical nature of the scheme. It requires a lot of care to implement. But it is the most efficient and highly secure technique. Compared to RSA, it provides the same level of security with very shorter key lengths. The 1024 bit security of RSA is equivalent to just 256 bits security of elliptic curve. Also it's very easy to implement due to the simplicity of the algorithm. The output per unit time compared to RSA is very large because elliptic curve doesn't contain long exponents and too many mathematical functions.

7. COMPARISION OF ASYMMETRIC ALGORITHMS

Asymmetric cryptographic algorithms are used in various applications as per their strength. The table below shows that the RSA & Elliptic Curve algorithms are suitable for all the applications with good key lengths. In current Internet security most of the organization use the RSA algorithm. Diffie-Hellman algorithm is mostly used only to exchange the keys between the two users.

Algorithm	**Key Exchange (Symmetric Key Distribution)**	**Digital Signature**	**Encryption / Decryption**
RSA	Yes	Yes	Yes
Diffie-Hellman	Yes	No	No
ElGamal	No	Yes	No
Elliptic Curve	Yes	Yes	Yes

Table -1: Comparison of asymmetric algorithms

8. CONCLUSIONS

In internet world the communication between the two users must be secure with the help of a cryptographic techniques. The asymmetric key algorithms that are discussed like RSA, Diffie-Hellman, ElGamal, Elliptic Curve are very useful to provide the confidentiality and authentication service in the online communication. The explanation of each algorithm shows that there is a complex calculation to generate the public key and private key and also to calculate the shared key between the two users. It is very difficult for an attacker to crack the keys.

9. REFERENCES

[1]. William Stallings, Cryptography and Network Security: Principles and Practice. Prentice Hall. p. 165. ISBN 9780138690175.

[2] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Voanstone (October 1996). "11: Digital Signatures" Handbook of Applied Cryptography. CRC Press. ISBN 0-8493-8523-7. Retrieved 14 November 2016.

[3] Daniel J. Bernstein (1 May 2008). "Protecting communications against forgery" (PDF). Algorithmic Number Theory. MSRI Publications. 44. §5: Public-key signatures, pp. 543–545. Retrieved 14 November 2016.

[4] Smart, Nigel (February 19, 2008). "Dr Clifford Cocks CB". Bristol University. Retrieved August 14, 2011.

[5] Rivest, R.; Shamir, A.; Adleman, L. (February 1978). "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems" (PDF). Communications of the ACM. 21 (2): 120–126.

[6] http://homepages.math.uic.edu/~leon/mcs425-s08/handouts/el-gamal.pdf