Analysis of rsa algorithm communications essay

ASSIGN BUSTER

To protect and hide data from malicious attacker and irrelevant public is the fundamental necessity of a security system. So for this reason for hiding data many cryptographic primitives like symmetric and asymmetric cryptography, digital signatures, hash functions etc. The symmetric cryptography consists of same key for encrypting and also for decrypting the data. Where as asymmetric cryptography takes advantage of a pair of keys to encrypt and decrypt the message. These keys are public key and a private key. The key which is distributed to other and which is publicly known is known as a public key and the key which is kept secret is known as private key. These two keys are needed simultaneously both for encrypting and decrypting the data. Public key will encrypt the data where as private key is used to decrypt the data. Asymmetric cryptographic should satisfy following properties. They are:

- Key generation process must be computationally efficient.
- By using the public key of the receiver the sender must be able to process the cipher text for any given message.
- By using the private key the decryption of cipher text into plain text should be done by the receiver.
- It will be impossible to compute like encrypt or decrypt the data without either of the key.

RSA was designed by Ronald Rivest, Adi Shamir, and Len Adleman. It is an asymmetric cryptographic technology. As in asymmetric cryptographic encryption the public key is known by everyone where as the private key is kept undisclosed. For decryption of data which is encrypted with the public key, private key must only be used. Integers between 0 to n-1 where n is the

modulus are taken as cipher and plain text. This n is generally 1024 bits. But the suggested length of n is 2048 bits instead of 1024 bits because it is no longer secure.

Algorithm of Key generation:

The following steps describe how a set of keys are generated.

- Two different prime numbers are selected which are not equal. Say p and q. this numbers are of same bit length.
- Determine modulus n where n= p. q
- Process or calculate Ï†(pq) =(pâˆ’1)(qâˆ’1). Here Ï† is totient.
- Select an integer which is public exponent e, such that 1
- Calculate d. This can be calculated by using modular arithmetic. This should satisfy de= 1. Now this ed-1 should be evenly divided by (p-1) (q-1) .

Here (n, e) is the public key which is used for encryption and (n, d) is a private key which is used for decryption.

Encryption: The following steps describe the how encryption is done in RSA algorithm. It is illustrated with an example where in two imaginary characters are described Alice and Bob. As we know that public key is (n, e) this is transmitted by Alice to Bob by keeping her private key secret. A message say M is wished by Bob to send to Alice. Before sending the message M it is converted into an integer 0

- Get the public key which is (n, e)
- Plain text integer is represented by m.
- Calculate cipher text as shown c= me

- Cipher text c is send to the receiver.

Decryption: Now when Alice receives the message sent by Bob, she regains the original message m from cipher text c by utilizing her private key exponent d. this can be done by cd= m (mod n). Now she can recover M once she regains m by using Padding scheme. This is shown as cd = (me)d = med (mod n). Since , med = m1+kq(n) = m(mq(n))k = m (mod n) . By this we get the original message back. This can be shown in following steps.

- Private key (n, d) is used by receiver to calculate m= cd mod n.
- The plaintext m is extracted.

Computational issues of RSA:

Selection of the two prime numbers p & q: In the very first step p is selected from a set of random number. After this it is ensured that p is odd by setting its highest and lowest bit. Finally p is made prime by applying a Miller Rabin algorithm.

Choosing the value of e: By choosing a prime number for e, the mathematical equation can be satisfied. That is gcd(e, p-1) = q. Among these three numbers which are 3, 17 and 65537 e is chosen for fast modular exponentiation.

Calculating the value d: It is determined by Extended Euclidean Algorithm which is equivalent to d = e-1 (mod q(n)).

Modular exponentiation algorithm: This step of RSA is calculated by following mathematical equation: AB mod n = (

Security of RSA:

RSA cryptosystem's security system is not so perfect. Many attacks are present like Brute Force attack, Timing Attack, chosen Ciphertext attack and Mathematical attack are some prominent attack.

Brute Force Attack: In this attack the attacker finds all possible way of combinations to break the private key. If the length of the key is long then it will be difficult for Brute force attackers to break the key as the possible combinations will exponentially increases rather then linearly. RSA uses a short secret key to avoid the long computations for encrypting and decrypting the data. If the key is long the process will become little slow because of these computations. Since RSA uses a short secret key Bute Force attack can easily break the key and hence make the system insecure.

Mathematical Attacks: Since RSA algorithm is mathematical, the most prominent attack against RSA is Mathematical Attack. In the following way an attacker can attack the mathematical properties of RSA algorithm.

- * By finding out the values of p and q which are prime factors of modulus n, the Ï†(n)= (p-1)(q-1) can be found out. By finding out this it will be easy to find d = e-1(mod Ï† (n)).
- d = e-1(mod Ï† (n)). Can be directly calculated by determining the value of totient Ï†(n) without figuring the values of p and q.
- d can be figured out directly without first calculating the Ï†(n).

This attack can be circumvented by using long length of key. By doing this it would be difficult to find out prime factors. That is the reason why it was recommended to use size of modulus as 2048 bits.

Timing Attack: one of the side channel attack is timing attack in which attackers calculate the time variation for implementation. Attackers can easily determine d by calculating the time variations that take place for computation of Cd (mod n) for a given cipher text C. Many countermeasures are developed against such timing attacks. Following explains the way which this attack can be counteracted:

- If the time for all computations is made constant this attack can be counteracted but the problem in doing this is it can degrade the computational efficiency.
- By artificially showing noise to the attacker which can be produced by including a random delay to the exponentiation algorithm. This noise is virtual but appears real to the attacker.
- If we multiply a random number to the cipher text it will prevent the attacker from bit by bit scrutiny.

Chosen Ciphertext Attack: RSA is susceptible to chosen cipher text attack due to mathematical property me1me2 = (m1m2)e (mod n) product of two plain text which is resultant of product of two cipher text. For example c = me (mod n) which is cipher text is decrypted in following steps:

- Calculate x = (c x 2e) mod n.
- Receive y = xd (mod n) by submitting x as a chosen cipher text.
- Multiplicative property is then applied which is: x = (c mod n) x (2c mod n) = (mc mod n) x (2c mod n) = (2m)c mod n.

By this attacker can calculate m by using y = (2m). By padding the plain text at the implementation level this restraint can be easily solved. Several

versions of RSA cryptography standard are been implemented. PKCS Public Key Cryptography standards are latest version. The previous version was proven to be porn to Adaptive Chosen Ciphertext attack (CCA2). This adaptive chosen cipher text can be prevented by latest version which is Optimal Asymmetric Encryption Padding (OAEP). Bellare and Rogway introduced this OAEP. To process the plain text before encryption the OAEP uses a pair of casual oracles G and H which is Feistel network. Following two goals are satisfied by OAEP.

OAEP PADDING PROCEDURE

Due to addition of random numbers the probabilistic scheme are being replaced instead of the deterministic encryption scheme. If the attacker is unable to invert the trapdoor one way permutation then the partial decryption of the cipher text is prevented.