RSA

Salem Almotiry; 382124012

Rivest-Shamir-Adleman

Introduction

The RSA algorithm (Rivest-Shamir-Adleman) is the basis of a cryptosystem -- a suite of cryptographic algorithms that are used for specific security services or purposes -- which enables public key encryption and is widely used to secure sensitive data, particularly when it is being sent over an insecure network such as the internet.

Key

RSA, Diffie-Hellman and other asymmetric algorithms use much larger keys than their symmetric counterparts. Common key sizes include 1024-bits and 2048-bits, and the keys need to be this **large** because factoring, while still a difficult operation, is much easier to perform than the exhaustive key search approach used with symmetric algorithms. The relative slowness of public key encryption systems is also due in part to these larger key sizes. Since most computers can only handle 32-bits of precision, different “tricks” are required to emulate the 1024-bit and 2048-bit integers. However, the additional processing time is somewhat justified, since for security purposes 2048-bit keys are considered to be secure “forever”—barring any exponential breakthroughs in mathematical factoring algorithms, of course.

RSA functionality and capabilities

1- Encryption and Decryption Methods

Encryption/Decryption strength and prosses is directly tied to key size. Doubling key length can deliver an exponential increase in strength, although it does impair performance. RSA does not work good for large file can be very slow.

2- Signatures/ verification

A message can be “signed” using a privately held decryption key. Anyone can verify this signature using the corresponding publicly revealed encryption key. Signatures cannot be forged, and a signer cannot later deny the validity of his signature. This has obvious applications in “electronic mail” and “electronic funds transfer” systems. Unlike the capability in encryption and decryption, sign and verify are much effectively