# RSA algorithm

RSA (Rivest-Shamir-Adelman, 1978) algorithm is an asymmetric encryption algorithm. To design an encryption/decryption key pair, two large prime numbers, p and q, , are selected, and an integer, d, is chosen that is relatively prime to (p-1)(q-1) (d and (p-1)(q-1) have no common factors other than 1). Finally, an integer e is computed such that



Euler totient function

fi(N)=fi(p)\*fi(q), N=p\*q

a^(fi(N)) mod N=1

K1\*K2=1 mod fi(N) =>K1\*K2=k\*fi(N)+1

M^( K1\*K2)=M^( k\*fi(N)+1) = M^( k\*fi(N))\*M^(1) = M^( k\*fi(N))\*M mod N= (M^( fi(N)))^k\*M mod N=(1)^k\*M mod N=M mod N =M

public key for encryption => only owner of the private key can decrypt=> confidentiality

private key for encryption => any person knowing the public key can decrypt=> no confidentiality, but can be used for authentication of the sender = digital signature

One key is (e,N), and the other is (d,N), where N=p\*q, and is referred to as the modulus.

For example, we might select p=7, and q=13. Then N=91, and fi(N)=(p-1)(q-1)=72. We can choose d=5 (which is relatively prime to 72) and e=29, because e\*d=145 and



Gcd(d, fi(N))=1=> relatively prime and inverse exists

Gcd=Euclidean Algorithm(a,b)

A=a; B=b;

1: If B==0 then return gcd=A;

R= A mod B;

A=B;

B=R;

Goto 1;

Gcd(72,5)

A=72, B=5

R=A mod B = 72 mod 5

Q=floor(A/B)=floor(72/5)=floor(14.4)= 14

R=A-Q\*B=72-14\*5=2

A=B=5; B=R=2

R= 5 mod 2=1

A=2; B=1;

R=2 mod 1=0

A=1; B=0=>gcd=A=1

M=72, b=5

A=(1,0,72), B=(0,1,5)

Q=floor(72/5)=14

T=A-Q\*B=(1-14\*0, 0-14\*1, 72-14\*5)= (1, -14, 2)

A=(0,1,5), B=(1, -14, 2)

Q=floor(5/2)=2

T=A-Q\*B=(0-2\*1, 1-2\*(-14), 5-2\*2)=(-2, 29, 1)

A=(1,-14,2), B=(-2,29,1)

Inverse is B2=29

29\*5=145 mod 72 =1 valid inverse

Extended Euclidean Algorithm(m,b) to find b^(-1) mod m

A=(A1,A2.A3)=(1,0,m); B=(B1,B2,B3)=(0,1,b);

1: If B3==0 then return gcd=A3, inverse does not exist;

If B3==1 then return inverse is B2 it is b^(-1) mod m

Q=floor(A3/B3)

T= A-Q\*B=(A1-Q\*B1, A2-Q\*B2, A3-Q\*B3);

A=B;

B=T; !! order matters

Goto 1;

Then, one key is K1=(29,91) and the other is K2=(5,91). The message to be encrypted is broken into blocks such that each block, M, can be treated as an integer between 0 and (N-1). To encrypt M into the ciphertext block, B, we perform



To decrypt B, we perform



The protocol works correctly because



More details about RSA algorithm can be found in the textbook by William Stallings, Cryptography and Network Security.

Returning to the example, assume M=2.

Then, to encrypt M, we compute



Thus, B=32. To decrypt B, we compute



which is the plaintext message M.

Obtaining of p and q is extremely difficult, hence, only knowing a secret key K2, receiver can correctly decrypt a message.

2^29??

29=16+8+4+1=2^4+2^3+2^2+2^0=11101

2^29=2^(16+8+4+1)=2^16\*2^8\*2^4\*2=16\*74\*16\*2=256\*74\*2 mod 91 =74\*74\*2 mod 91 = 16\*2=32

2^16, 2^8, 2^4

2^2=4 mod 91=4

2^4=2^(2\*2)=(2^2)^2=4^2=16 mod 91 =16

2^8=(2^4)^2=16^2=256 mod 91 =(182+74) mod 91 =74

2^16=(2^8)^2=74\*74 mod 91= 5476 mod 91 =16

Decryption

32^5=32^(4+1)=32^4\*32 mod 91=74\*32 mod 91=(74\*2)\*16 mod 91 = 148\*16 mod 91=57\*16 mod 91 = 114\*8 mod 91 = 23\*8 mod 91 = 92\*2 mod 91 = 1\*2 mod 91 = 2

32^2=1024 mod 91 = 23

32^4=(32^2)^2=23^2=529 mod 91 = 74