

Using Certificates on the Web

Galen Hunter, www.VInsurance.com; Andrew Michalik, www.TekCoach.com

Introduction to RSA

RSA is a public-key cryptosystem developed by MIT professors: Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman in 1977 in an effort to help ensure internet security. As Steve Burnett of RSA Data Security, Inc. described it, a cryptosystem is simply an algorithm that can convert input data into something unrecognizable (encryption), and convert the unrecognizable data back to its original form (decryption).

To encrypt data, enter the data ("plaintext") and an encryption key to the encryption portion of the algorithm. To decrypt the "ciphertext," a proper decryption key is used at the decryption portion of the algorithm. Those keys, which contain simply a string of numbers, are called public key and private key, respectively. For example, suppose Alice intends to send e-mail to Bob. Through a public-key directory, she finds his public key. Then, she encrypts her message using the key and sends it to Bob. This public key; however, will not decrypt the ciphertext. Knowledge of Bob's public key will not help an eavesdropper. In order for Bob to decrypt his ciphertext, he must use his private key. If Bob wants to respond to Alice, he encrypts his message using **her** public key.

The challenge of public-key cryptography is developing a system in which it is impossible to determine the private key. This is accomplished through the use of a one-way function. With a one-way function, it is relatively easy to compute a result given some input values. However, it is extremely difficult, nearly impossible, to determine the original values if you start with the result. In mathematical terms, given x , computing $f(x)$ is easy, but given $f(x)$, computing x is nearly impossible. The one-way function used in RSA is multiplication of prime numbers. It is easy to multiply two big prime numbers, but for most very large primes, it is extremely time-consuming to factor them. Public-key cryptography uses this function by building a cryptosystem, which uses two large primes to build the private key and the product of those primes to build the public key.

The Model

RSA uses modular arithmetic and elementary number theory to do certain computation. Before you start, please make sure you understand how it works. And remember, our model is nothing compared to the **REAL** RSA algorithm which involves two LARGE primes and messages of unconditional length. In our model, VERY small primes are used and only one letter will be encrypted.

Instructions:

- This model is best carried out by two persons. Name them Alice and Bob.
- Alice will pick two available primes. Public and private keys will be generated by computer.
- Record the public key containing the exponent value, E and the product of the primes, N . And, record the private key, D .
- Give E and N to Bob (your partner).
- Bob will go to another page to pick a letter to encrypt. Enter E and N . The encrypted message / number will be generated.
- Bob will send or give the encrypted message to Alice.
- Alice will go to decryption page. Enter the message, D and N . The message will be decrypted to the original letter. Later, Alice can check with Bob to see if it is the right letter.

Remember, the main purpose of this model is in understanding the RSA algorithm, not necessarily for encryption purpose. A lot of simplification has been made, while the mathematics and algorithm stay the same.

KEY GENERATION

OBJECTIVE:

The purpose of this page is to generate a public key pair and a private key pair, by choosing two available primes. In real cryptography system, very large primes are used. By entering two primes, P and Q, computer will generate N, E and D. E and N are the public key pair, while D and N are the private key pair. It is very important that you **write down those numbers (N, E and D.)** You will need E and N for encryption and D and N for decryption.

Pick 2 primes (P & Q): **5 & 3**

Actual Key Generation:

$$N = P \times Q = 5 \times 3 = 15$$

$$PHI = (P-1)(Q-1) = 8$$

The public exponent E will be generated by the computer so that the greater common divisor of E and PHI is 1. In other words, E is relatively prime with PHI.

$$E = 3$$

N and E are your public keys. Your private key (D) is the inverse of E modulo PHI.

By using extended Euclidian algorithm, the private key, **D, is 3**

Don't forget to record N, E and D!

Now, you can give your pair of public keys, E and N, to Bob so that he can send you encrypted letter by using those key. Later, you can decrypt it, using D and N.

ENCRYPTION

OBJECTIVE:

The purpose is to encrypt a letter using RSA algorithm. Knowing E and N is required.

Pick a letter to cipher **J**

Enter Alice's Exponent key, E: **3**

Enter Alice's N value: **15**

Actual Encryption:

Letter, J is converted to number: 10

$$C = M^E \text{ mod } N = 10^3 \text{ mod } 15 = 10$$

Thus, the encrypted message is **10**

DECRYPTION

OBJECTIVE:

This is the final step of understanding RSA algorithm. The purpose is to decrypt the encrypted message, by knowing the private key pair, D and N.

Enter the encrypted message: **10**

Enter your N value: **15**

Enter your private key, D: **3**

The encrypted message will be decrypted using the following method:

$$M = C^D \text{ mod } N = 10^3 \text{ mod } 15 = 10$$

10 is converted to letter. The original message is **J**

Thanks to Cary Sullivan and Rummy Makmur, authors of The Cryptography Home Page at <http://osu.orst.edu>