RbCripto Manual

About RbCripto

RbCripto is a simple program that was designed with academic purposes. It can perform file encryption and decryption using the modern concept of elliptic curves combined with Diffie-Hellman algorithm and associated with AES 256-bit symmetric key.

Requirements

The program consists only a small executable with less than 1MB and requires no installation, so it can be kept in flash drive or memory card. Its operation requires no more than 10MB of memory. To encrypt or decrypt files, requires memory and disk space according to the size of the file. For example, for a 30MB file, it requires the same 30MB of disk space and preferably 30MB of RAM. For large files, for example, 5GB, you need 5GB of disk space and the more free memory, the better. RbCripto has an algorithm to optimize memory allocation, thus avoiding the pagination (situation where the system uses disk space when there is not much free memory).

Sample statistics

RbCripto was tested in a computer based on the AMD X2 1.9GHz with 1GB of RAM and SATA2 HD. For a file with 4.2 GB (the size of a home DVD player), it needed 12 minutes to perform the encryption and 9 minutes to perform the decryption. For this work it used about 700MB RAM free. For a 300MB file, it took 1 minute and a half to encrypt and decrypt.

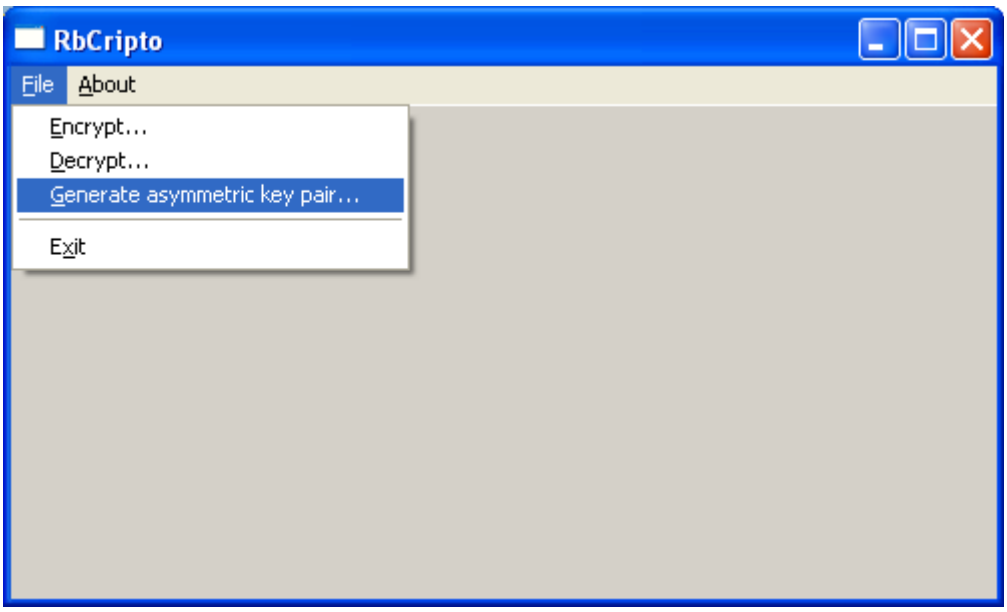
Symmetric encryption versus asymmetric encryption

The difference between the two concepts is simple: in symmetric encryption, the same key used to encrypt the file should be used to decrypt it; in asymmetric encryption, the keys are distinct and have a correlation based on complex mathematical concepts. RbCripto supports both types. To use symmetric encryption, the user must enter a password, which will be converted into cryptographic key to encrypt the file. To decrypt it, you should remember the same password to enable the reverse process. To use asymmetric encryption, the user need only generate a key pair, which will be saved in files.

Using the program

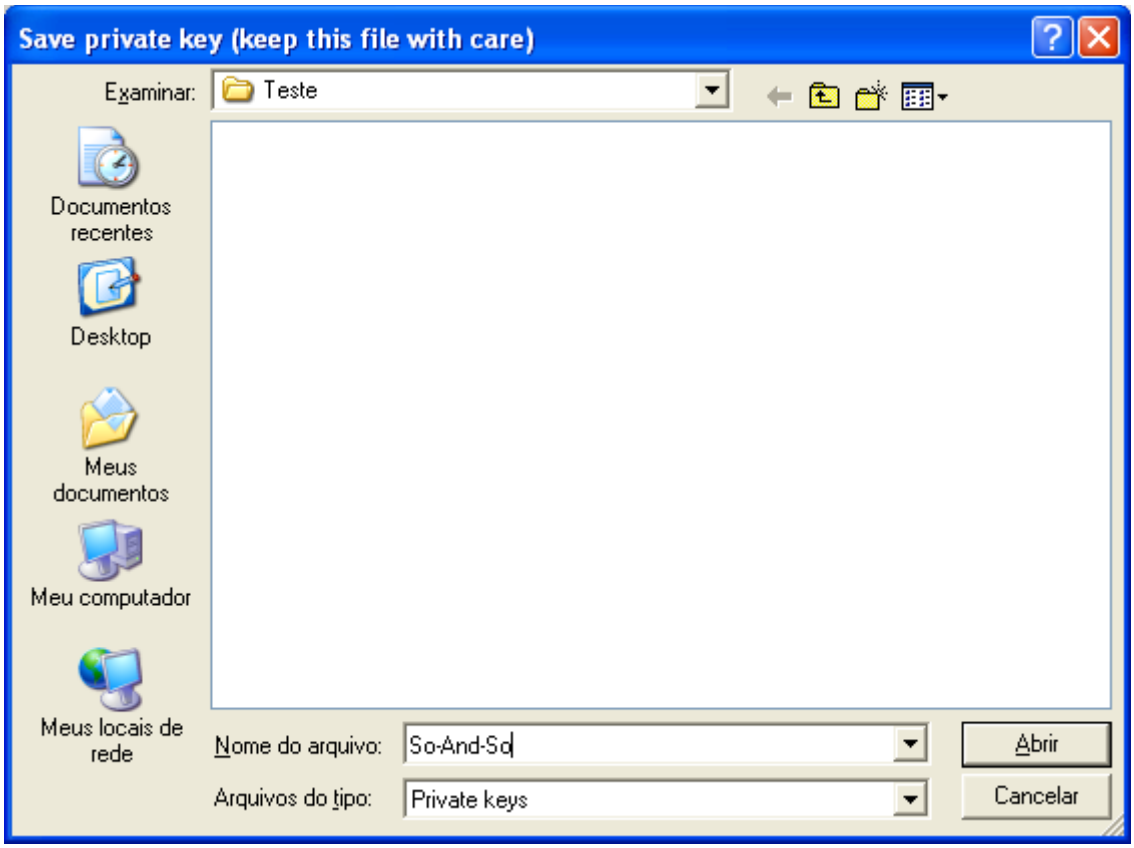
Generation of asymmetric keys

Asymmetric keys are used to make the encryption and decryption of the file without using passwords. Works as follows: the user creates an asymmetric key pair within RbCripto:

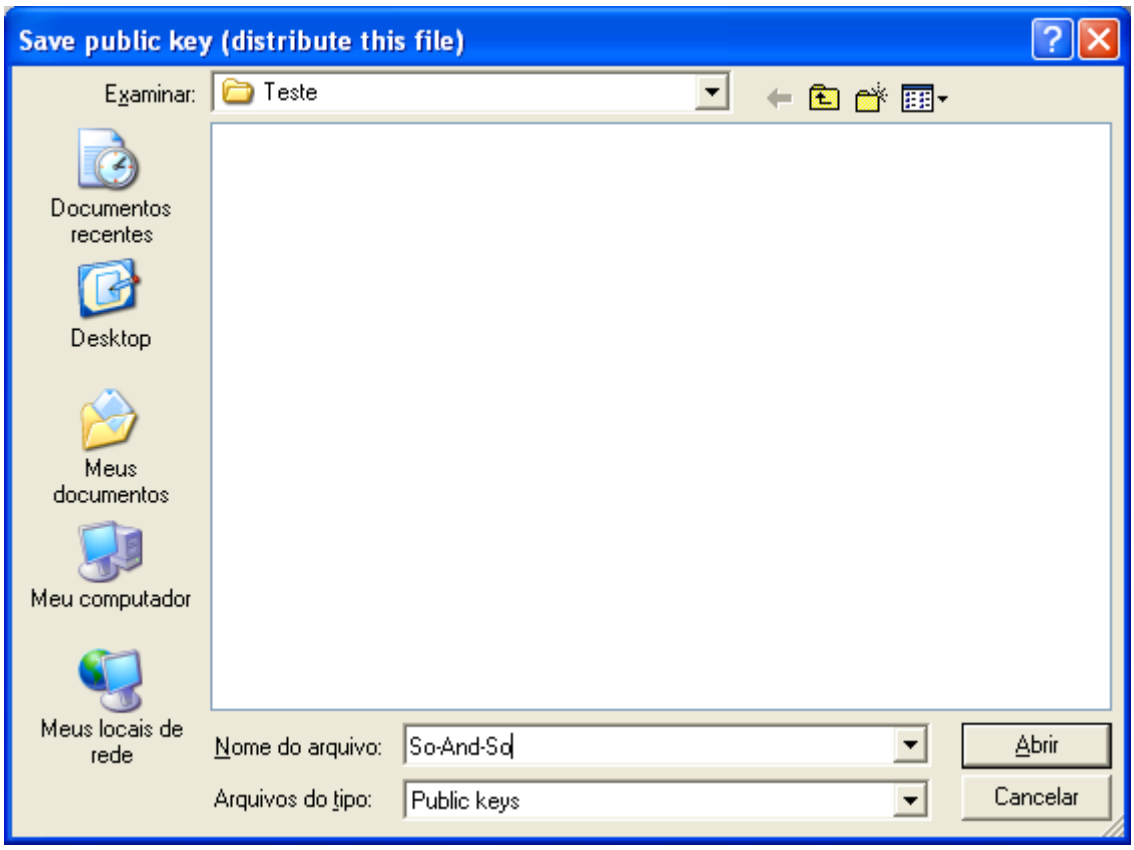


The program automatically generates keys and opens dialog boxes to save files.

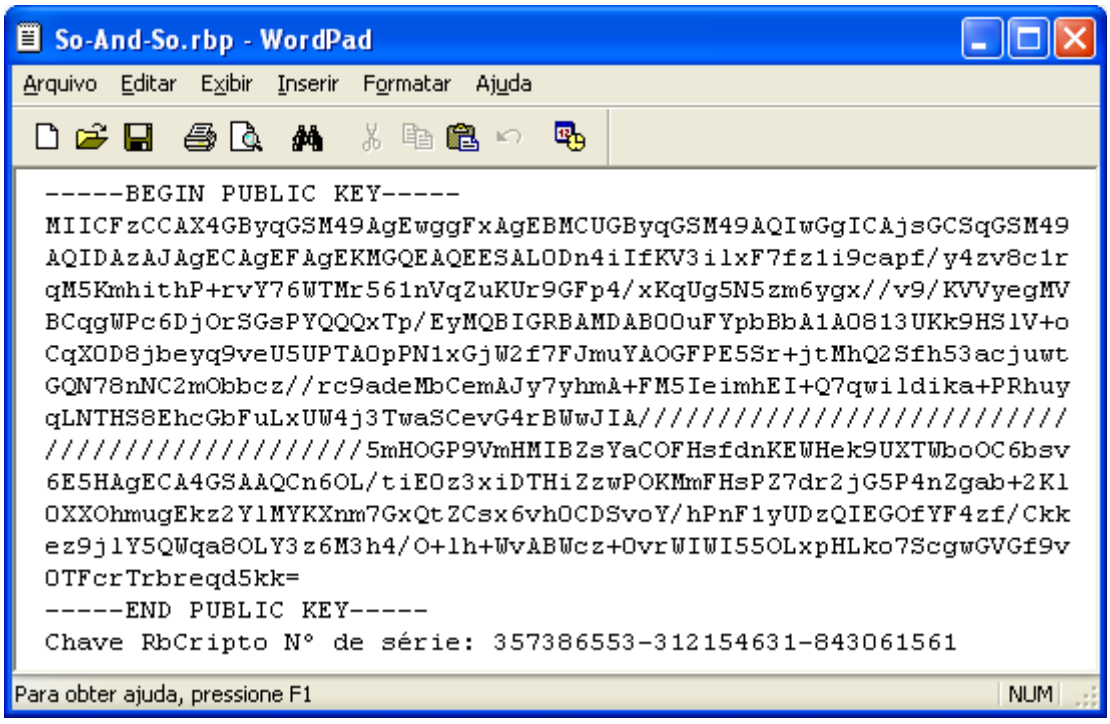
The first file to be written is the private key - this file should be stored carefully and never be published:



The second file is the public key - this file must be distributed and other people will use it to encrypt files which can only be decrypted with its private key, which was guarded with care:

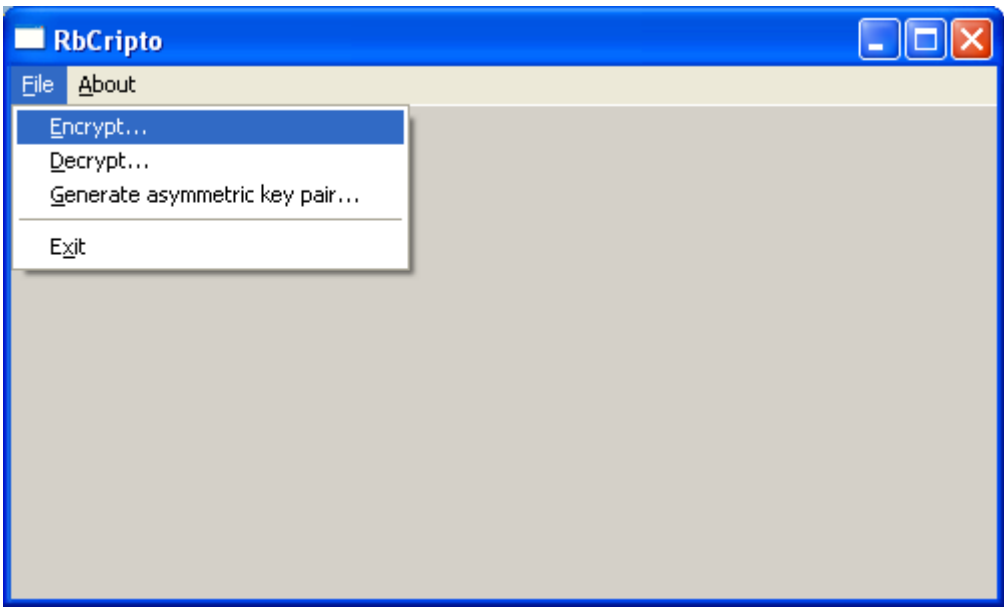


This is an example of a RbCripto key:

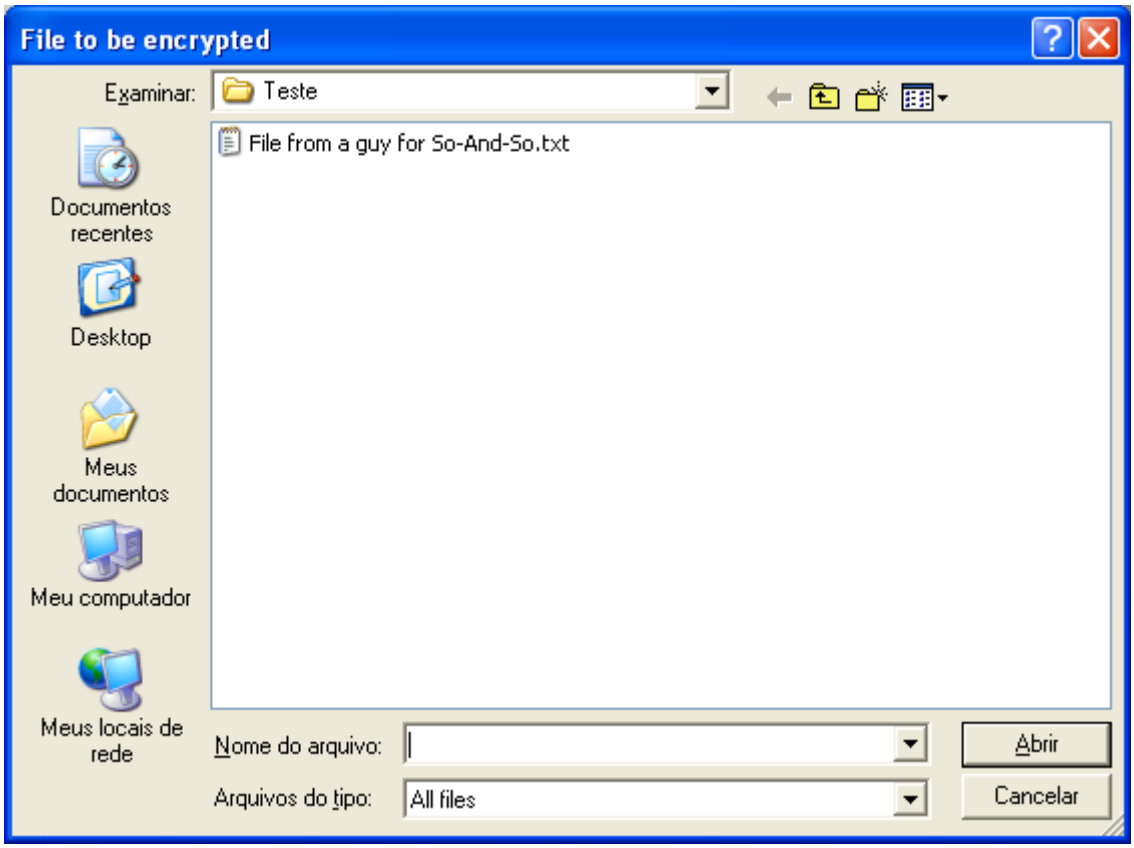


Encryption

It's very simple, click the ENCRYPT function:

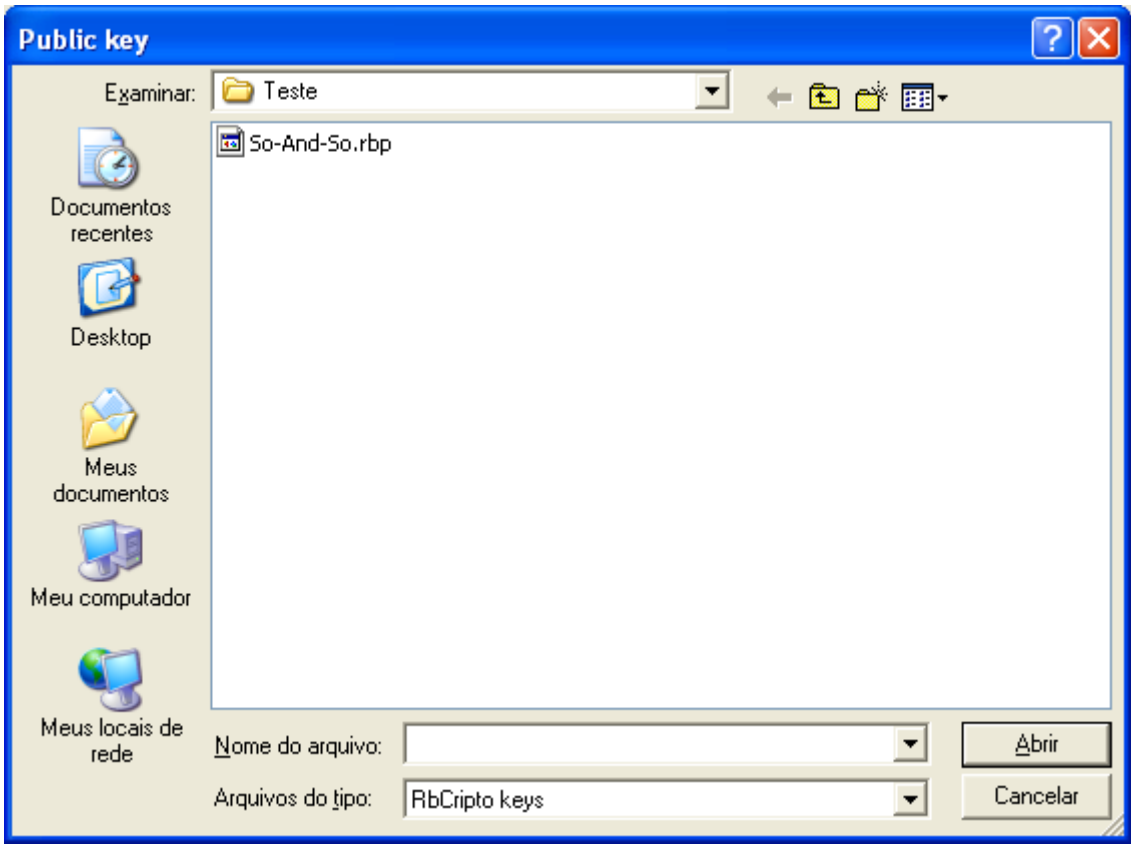


RbCripto opens the box to choose the file to be encrypted:

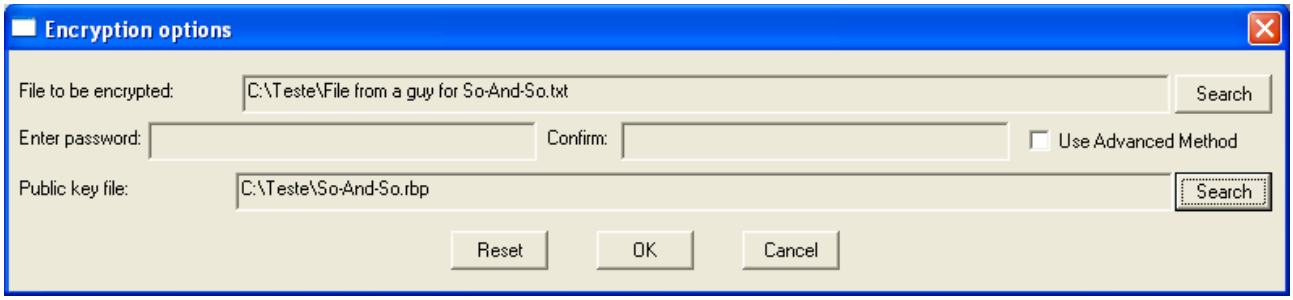


After choosing, it opens a box of options like this:

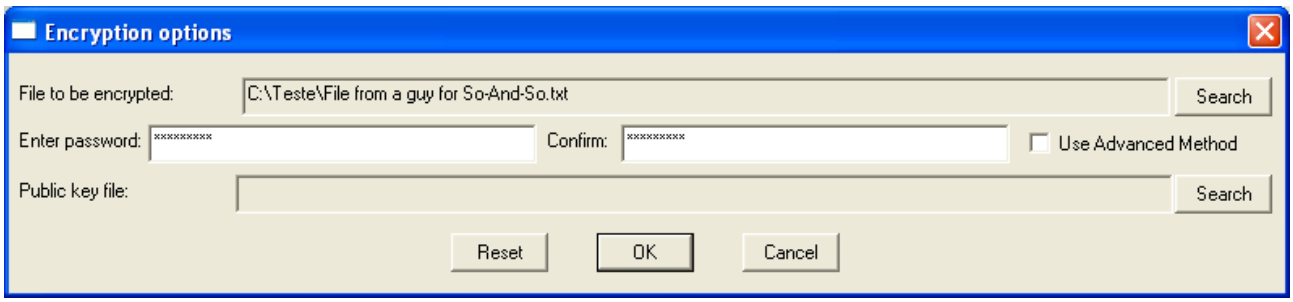
If you want to use asymmetric encryption, just use the SEARCH button next to "public key file" to locate the corresponding file:



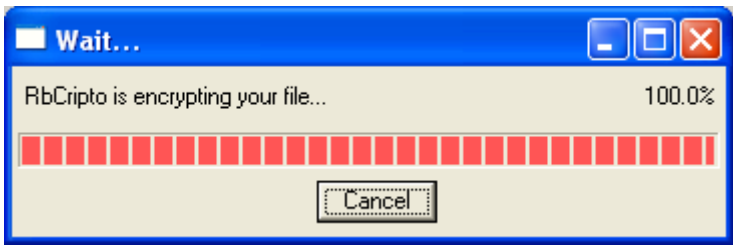
In the example below, any person (or So-and-So himself) will encrypt a file which can be decrypted only by So-and-So, because he alone has the corresponding private key:



If you want to use symmetric encryption, you must enter and confirm a password:

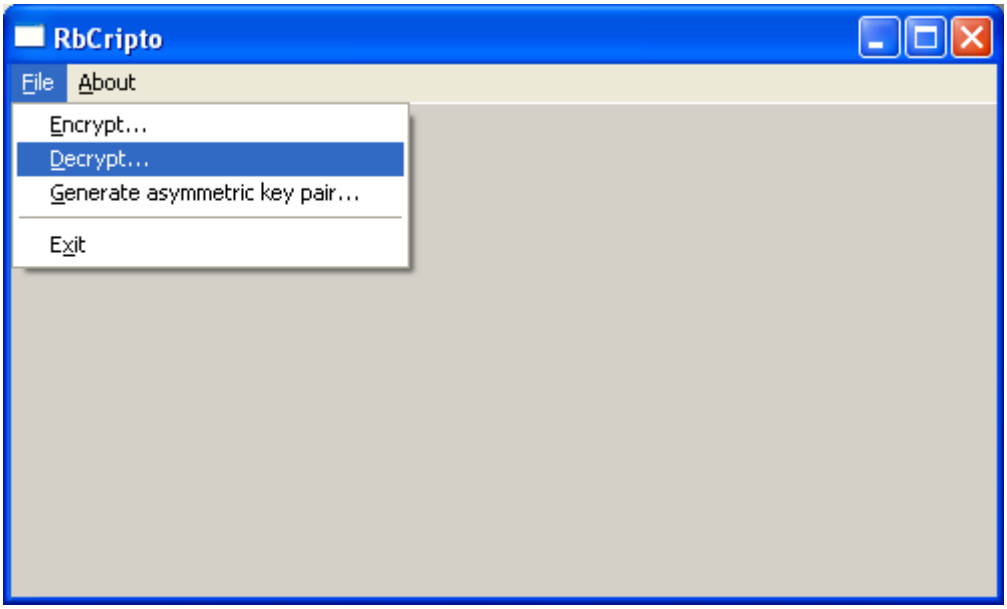


In any case, it is recommended to select "Use Advanced Method", so RbCripto will use variations on algorithm to make the file more secure without requiring more time for that. During processing, RbCritpo displays a progress bar like this:

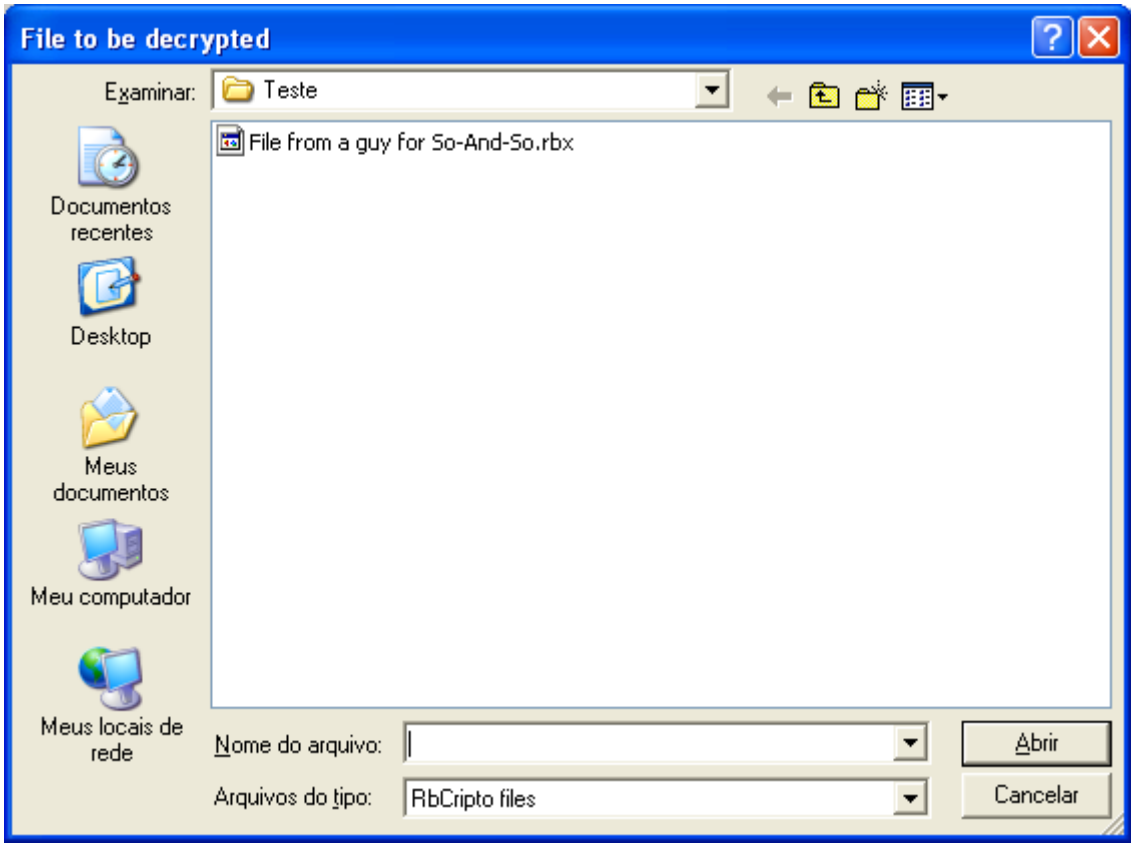


Decryption

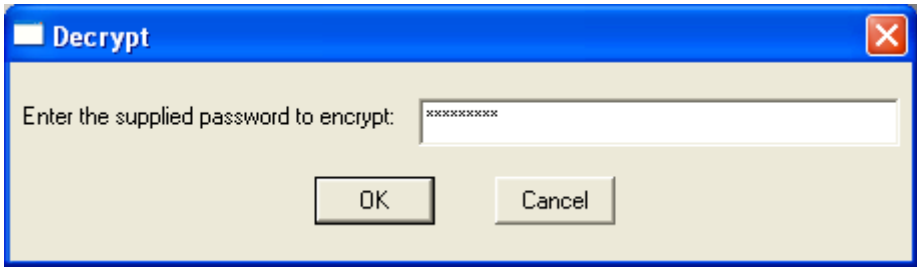
To decrypt a file, the process is even simpler. Choose the DECRYPT function:



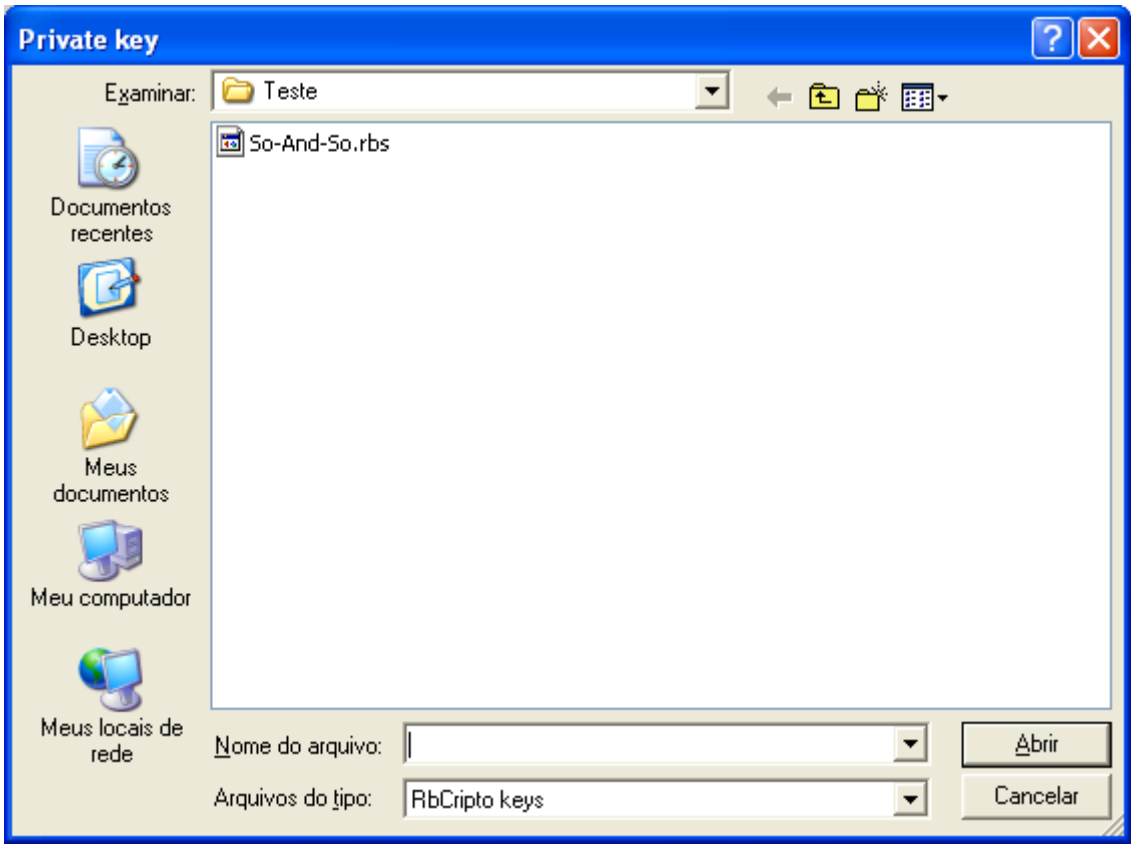
Select the file to be decrypted:



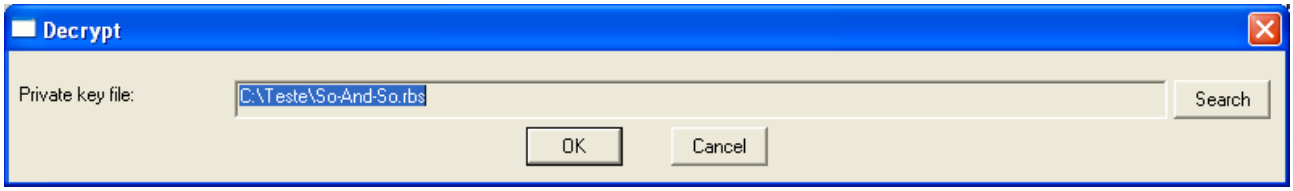
RbCripto automatically detects the options used in encryption and acts accordingly. If you have been used symmetric encryption, it asks for a password in a box like this, in which the user must enter the same password used to encrypt:



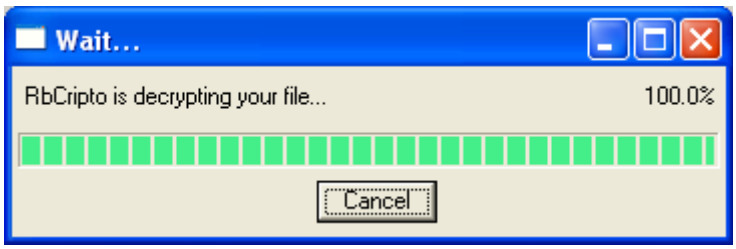
If the user has opted for asymmetric cryptography, RbCripto need the private key file corresponding to file public key used to encrypt. For example, if someone (or So-and-So himself) used the So-and-So's public key to encrypt, the program will need the So-and-So's private key (only So-and-So has) and it asks for file which has this key. When it open the box, just select the correct file:



And the file can be decrypted normally:



During processing, RbCritpo displays a progress bar like this:



Advanced – public key by sending e-mail

For asymmetric encryption, you must distribute the public key. It is clear that e-mail is the first idea to the distribution. How these days many ISPs block e-mail attachments, the public key file (RBP extension) attachment can cause problems. But there is a way around this: just open the file with Wordpad, for example, copy and paste its contents in the message body. The only restriction is that the message can not have HTML formatting (any e-mail program or webmail interface on the Internet allows this option). The person receiving the e-mail simply need to paste the text into Wordpad window and save the file with RBP extension.