

Re: Need simple lib for asymmetric encryption

Source: <http://www.derkeiler.com/Newsgroups/sci.crypt/2006-08/msg00624.html>

- *From:* Unruh <unruh-spam@xxxxxxxxxxxxxxxx>
 - *Date:* 9 Aug 2006 15:11:17 GMT
-

"amzoti" <amzoti@xxxxxxxx> writes:

Oliver Eichler wrote:

Hi,

I am not a crypto specialist, thus please forgive me if I sound a little bit clueless.

I would like to encrypt some data with key1 and decrypt it again with key2. If I got it right from all the stuff I have read so far you refer to key1 as 'public key' and to key2 as 'private key'. Normally the public key is generated from the private key. Thus, who got the private key can always generate the public key. Is this mandatory?

Public key crypto is abysmally slow. Noone ever actually encrypts data with a public key. They encrypt a random key for symmetric key crypto (ie the same key encrypts and decrypts loosely speaking) and use that much faster symmetric crypto to actually encrypt the data.

Since one HAS to generate both the encryption and decryption key and has to make sure that the decryption key actually decrypts and that it is not derivable from any public data, it would seem that the only way is to derive the public key from the private, or at least both from some other private data.

What algorithm would I need to satisfy my needs? And is there a simple to use, light weight C library?

RSA, DSA, Elliptic curve crypto.

Re: Need simple lib for asymmetric encryption

I have looked into cryptlib, beecrypt and others. But they all seem to be an overkill to my problem with a quite hard to understand API. Isn't there something like :

I think that maybe you need to learn more before trying to implement crypto. The algorithm is the least of your worries. Key control is a far worse worry.

```
generateKeypair(key1,key2)
encrypt(key1,...)
decrypt(key2,...)
```

And that's it.

Thanks for help

oliver

Did you look at libtomcrypt?

Maybe it is a better choice if things like PGP or Crypto++ are not.