Paillier-GMP

Generated by Doxygen 1.7.6.1

Fri Sep 7 2012 22:31:42

Contents

Chapter 1

Main Page

1.1 Requirements

You need a system with /dev/urandom and gmp to run this program.

1.2 Syntax for the built-in interpreter

Available commands:

- paillier keygen [public key file name] [private key file name] [bit length]
- paillier encrypt [output ciphertext file name] [input plain text file name] [public key file name]
- paillier decrypt [output plaintext file name] [input ciphertext file name] [private key file name]

1.3 Building the program

Make options:

- "make all" will build the documentation, the interpreter and the static library.
- "make release" will build the interpreter.

- "make doc" will build the documentation.
- "make debug" will build the interpreter with debug symbols.

Chapter 2

Module Index

2.1 Modules

Here is a list of all modules:

Chapter 3

Data Structure Index

3.1 Data Structures

Here are the data structures with brief descriptions:

Chapter 4

File Index

4.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 5

Module Documentation

5.1 Command interpreter for Paillier cryptosystem

Functions

- int main (int argc, char ∗argv[ ])

Variables

- const char ∗ hlp_message = " decrypt [out_file] [in_file] [private_key_file]\n"

5.1.1 Detailed Description

This file is part of Paillier-GMP.

5.1.2 Function Documentation

5.1.2.1 int main (int *argc,* char ∗ *argv[ ]*)

Main function

Run key generation, encryption or decryption.

Parameters

in	*argc*	number of arguments
in	*argv*	arguments • keygen [public_key_file] [private_key_file] [bit length] • encrypt [out_file] [in_file] [public_key_file] • decrypt [out_file] [in_file] [private_key_file]

5.1.3 Variable Documentation

5.1.3.1 const char* hlp_message = " decrypt [out_file] [in_file] [private_key_file]\n"

Help message

5.2 Paillier cryptosystem

Data Structures

- struct paillier_private_key
- struct paillier_public_key

Functions

- int paillier_ell (mpz_t result, mpz_t input, mpz_t ninv, mp_bitcnt_t bits)
- void paillier_public_init (paillier_public_key ∗pub)
- void paillier_private_init (paillier_private_key ∗priv)
- void paillier_public_clear (paillier_public_key ∗pub)
- void paillier_private_clear (paillier_private_key ∗priv)
- int paillier_public_out_str (FILE ∗fp, paillier_public_key ∗pub)
- int paillier_private_out_str (FILE ∗fp, paillier_private_key ∗priv)
- int paillier_public_in_str (paillier_public_key ∗pub, FILE ∗fp)
- int paillier_private_in_str (paillier_private_key ∗priv, FILE ∗fp)
- int paillier_keygen (paillier_public_key ∗pub, paillier_private_key ∗priv, mp_bitcnt_t bits)
- int paillier_keygen_str (FILE ∗public_key, FILE ∗private_key, int bits)
- int paillier_encrypt (mpz_t ciphertext, mpz_t plaintext, paillier_public_key ∗pub)
- int paillier_encrypt_str (FILE ∗ciphertext, FILE ∗plaintext, FILE ∗public_key)
- int paillier_decrypt (mpz_t plaintext, mpz_t ciphertext, paillier_private_key ∗priv)
- int paillier_decrypt_str (FILE ∗ciphertext, FILE ∗plaintext, FILE ∗private_key)

5.2.1 Detailed Description

This file is part of Paillier-GMP.

5.2.2 Function Documentation

5.2.2.1 int paillier_decrypt (mpz_t *plaintext*, mpz_t *ciphertext*, paillier_private_key ∗ *priv*)

Decrypt

Parameters

`out`	*plaintext*	output plaintext m
`in`	*ciphertext*	input ciphertext
`in`	*priv*	input private key

5.2.2.2 int **paillier_decrypt_str** (FILE ∗ *ciphertext*, FILE ∗ *plaintext*, FILE ∗ *private_key*)

Decrypt from stdio stream

Parameters

`out`	*plaintext*	output stream for plaintext m
`in`	*ciphertext*	input stream for ciphertext c
`in`	*private_key*	input stream for private key

5.2.2.3 int **paillier_ell** (mpz_t *result*, mpz_t *input*, mpz_t *ninv*, mp_bitcnt_t *bits*)

Function L(u)=(u-1)/n

Parameters

`out`	*result*	output result (u-1)/n
`in`	*input*	u
`in`	*ninv*	input n^{-1} mod 2^len
`in`	*bits*	input bit length len

5.2.2.4 int **paillier_encrypt** (mpz_t *ciphertext*, mpz_t *plaintext*, **paillier_public_key** ∗ *pub*)

Encrypt

Parameters

`out`	*ciphertext*	output ciphertext c=g^m∗r^n mod n^2
`in`	*plaintext*	input plaintext m
`in`	*pub*	input public key

5.2.2.5 int **paillier_encrypt_str** (FILE ∗ *ciphertext*, FILE ∗ *plaintext*, FILE ∗ *public_key*)

Encrypt from stdio stream

Parameters

`out`	*ciphertext*	output stream for ciphertext c=g^m∗r^n mod n^2
`in`	*plaintext*	input stream for plaintext m

in	*public_key*	input stream for public key

5.2.2.6 int **paillier_keygen** (**paillier_public_key** ∗ *pub,* **paillier_private_key** ∗ *priv,* mp_bitcnt_t *bits*)

Key generation

Parameters

out	*pub*	output public key
out	*priv*	output private key
in	*bits*	input bit length of public modulus

5.2.2.7 int **paillier_keygen_str** (FILE ∗ *public_key,* FILE ∗ *private_key,* int *bits*)

Key generation to stdio stream

Parameters

out	*public_key*	output stream for public key
out	*private_key*	output stream for private key
in	*bits*	input bit length of public modulus

5.2.2.8 void **paillier_private_clear** (**paillier_private_key** ∗ *priv*)

Free memory for private key

Parameters

in	*priv*	input private key

5.2.2.9 int **paillier_private_in_str** (**paillier_private_key** ∗ *priv,* FILE ∗ *fp*)

Input private key

Parameters

out	*priv*	output private key
in	*fp*	input stream

5.2.2.10 void **paillier_private_init** (**paillier_private_key** ∗ *priv*)

Memory allocation for private key

Parameters

in	*priv*	input private key

5.2.2.11 int **paillier_private_out_str** (FILE ∗ *fp*, **paillier_private_key** ∗ *priv*)

Output private key

Parameters

out	*fp*	output stream
in	*priv*	input private key

5.2.2.12 void **paillier_public_clear** (**paillier_public_key** ∗ *pub*)

Free memory for public key

Parameters

in	*pub*	input public key

5.2.2.13 int **paillier_public_in_str** (**paillier_public_key** ∗ *pub*, FILE ∗ *fp*)

Input public key

Parameters

out	*pub*	output public key
in	*fp*	input stream

5.2.2.14 void **paillier_public_init** (**paillier_public_key** ∗ *pub*)

Memory allocation for public key

Parameters

in	*pub*	input public key

5.2.2.15 int paillier_public_out_str (FILE ∗ *fp*, paillier_public_key ∗ *pub*)

Output public key

Parameters

`out`	*fp*	output stream
`in`	*pub*	input public key

5.3 Tools for Paillier-GMP

Defines

- #define BIT2BYTE(a) (a+7)/8

Functions

- int debug_msg (const char ∗str)
- int gen_random (mpz_t rnd, mp_bitcnt_t bits)
- int gen_prime (mpz_t prime, mp_bitcnt_t bits)
- int crt_exponentiation (mpz_t result, mpz_t base, mpz_t exp_p, mpz_t exp_q, mpz_t pinvq, mpz_t p, mpz_t q)

5.3.1 Detailed Description

This file is part of Paillier-GMP.

Paillier-GMP is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

Paillier-GMP is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with Paillier-GMP. If not, see <http://www.gnu.org/licenses/>.

5.3.2 Define Documentation

5.3.2.1 #define **BIT2BYTE**(*a*) (a+7)/8

Convert bit length to byte length

5.3.3 Function Documentation

5.3.3.1 int **crt_exponentiation** (mpz_t *result,* mpz_t *base,* mpz_t *exp_p,* mpz_t *exp_q,* mpz_t *pinvq,* mpz_t *p,* mpz_t *q*)

Exponentiation with Chinese Remainder Theorem

Parameters

out	*result*	output exponentiation result
in	*base*	input basis of the exponentiation
in	*exp_p*	input exponent for modulo p exponentiation

in	*exp_q*	input exponent for modulo q exponentiation
in	*pinvq*	input CRT parameter
in	*p*	input modulus p
in	*q*	input modulus q

5.3.3.2 int debug_msg (const char * *str*) [inline]

Print debug message

Parameters

in	*str*	input debug message

5.3.3.3 int gen_prime (mpz_t *prime,* mp_bitcnt_t *bits*)

Generate prime number

Parameters

out	*prime*	output prime number, randomness coming from /dev/random
in	*bits*	input bit length of prime number to generate

5.3.3.4 int gen_random (mpz_t *rnd,* mp_bitcnt_t *bits*)

Generate random number

Parameters

out	*rnd*	output random number, randomness coming from /dev/urandom
in	*bits*	input bit length of the random number to generate

Chapter 6

Data Structure Documentation

6.1 paillier_private_key Struct Reference

```
#include <paillier.h>
```

Data Fields

- mp_bitcnt_t bitlen
- mpz_t lambda
- mpz_t mu
- mpz_t p2
- mpz_t q2
- mpz_t p2invq2
- mpz_t ninv
- mpz_t n

6.1.1 Detailed Description

Private key

6.1.2 Field Documentation

6.1.2.1 mp_bitcnt_t paillier_private_key::bitlen

bit length of n

6.1.2.2 mpz_t paillier_private_key::lambda

least common multiple of p and q

6.1.2.3 mpz_t paillier_private_key::mu

Modular inverse

6.1.2.4 mpz_t paillier_private_key::n

n=p∗q

6.1.2.5 mpz_t paillier_private_key::ninv

n^{-1} mod 2^l

6.1.2.6 mpz_t paillier_private_key::p2

square of prime number p

6.1.2.7 mpz_t paillier_private_key::p2invq2

p^{-2} mod q^2

6.1.2.8 mpz_t paillier_private_key::q2

square of prime number q

The documentation for this struct was generated from the following file:

- src/paillier.h

6.2 paillier_public_key Struct Reference

```
#include <paillier.h>
```

Data Fields

- mp_bitcnt_t bitlen
- mpz_t n

6.2.1 Detailed Description

Public key

The generator is 1+n.

6.2.2 Field Documentation

6.2.2.1 mp_bitcnt_t paillier_public_key::bitlen

bit length of n

6.2.2.2 mpz_t paillier_public_key::n

modulus

The documentation for this struct was generated from the following file:

- src/paillier.h

Chapter 7

File Documentation

7.1 src/main.c File Reference

`#include <stdio.h> #include "paillier.h"`

Functions

- int main (int argc, char ∗argv[ ])

Variables

- const char ∗ hlp_message = " decrypt [out_file] [in_file] [private_key_file]\n"

7.1.1 Detailed Description

Date

Created on: Aug 25, 2012

Author

Camille Vuillaume

Copyright

Camille Vuillaume, 2012

7.2 src/paillier.c File Reference

`#include <stdlib.h> #include "paillier.h" #include "tools.h"`

Functions

- int paillier_ell (mpz_t result, mpz_t input, mpz_t ninv, mp_bitcnt_t bits)
- int paillier_keygen (paillier_public_key ∗pub, paillier_private_key ∗priv, mp_bitcnt_t bits)
- int paillier_encrypt (mpz_t ciphertext, mpz_t plaintext, paillier_public_key ∗pub)
- int paillier_decrypt (mpz_t plaintext, mpz_t ciphertext, paillier_private_key ∗priv)

7.2.1 Detailed Description

Date

Created on: Aug 25, 2012

Author

Camille Vuillaume

Copyright

Camille Vuillaume, 2012

This file is part of Paillier-GMP.

7.3 src/paillier.h File Reference

`#include <stdio.h> #include <gmp.h>`

Data Structures

- struct paillier_private_key
- struct paillier_public_key

Functions

- void paillier_public_init (paillier_public_key ∗pub)
- void paillier_private_init (paillier_private_key ∗priv)
- void paillier_public_clear (paillier_public_key ∗pub)
- void paillier_private_clear (paillier_private_key ∗priv)
- int paillier_public_out_str (FILE ∗fp, paillier_public_key ∗pub)
- int paillier_private_out_str (FILE ∗fp, paillier_private_key ∗priv)
- int paillier_public_in_str (paillier_public_key ∗pub, FILE ∗fp)
- int paillier_private_in_str (paillier_private_key ∗priv, FILE ∗fp)
- int paillier_keygen (paillier_public_key ∗pub, paillier_private_key ∗priv, mp_bitcnt_t bits)
- int paillier_keygen_str (FILE ∗public_key, FILE ∗private_key, int bits)
- int paillier_encrypt (mpz_t ciphertext, mpz_t plaintext, paillier_public_key ∗pub)
- int paillier_encrypt_str (FILE ∗ciphertext, FILE ∗plaintext, FILE ∗public_key)
- int paillier_decrypt (mpz_t plaintext, mpz_t ciphertext, paillier_private_key ∗priv)
- int paillier_decrypt_str (FILE ∗ciphertext, FILE ∗plaintext, FILE ∗private_key)

7.3.1 Detailed Description

Date

Created on: Aug 25, 2012

Author

Camille Vuillaume

Copyright

Camille Vuillaume, 2012

7.4 src/paillier_io.c File Reference

```
#include "paillier.h"
```

Functions

- int paillier_keygen_str (FILE ∗public_key, FILE ∗private_key, int bits)
- int paillier_encrypt_str (FILE ∗ciphertext, FILE ∗plaintext, FILE ∗public_key)
- int paillier_decrypt_str (FILE ∗plaintext, FILE ∗ciphertext, FILE ∗private_key)

7.4.1 Detailed Description

Date

Created on: Sep 06, 2012

Author

Camille Vuillaume

Copyright

7.5 src/paillier_manage_keys.c File Reference

```
#include "paillier.h"
```

Functions

- void paillier_public_init (paillier_public_key ∗pub)
- void paillier_private_init (paillier_private_key ∗priv)
- void paillier_public_clear (paillier_public_key ∗pub)
- void paillier_private_clear (paillier_private_key ∗priv)
- int paillier_public_out_str (FILE ∗fp, paillier_public_key ∗pub)
- int paillier_private_out_str (FILE ∗fp, paillier_private_key ∗priv)
- int paillier_public_in_str (paillier_public_key ∗pub, FILE ∗fp)
- int paillier_private_in_str (paillier_private_key ∗priv, FILE ∗fp)

7.5.1 Detailed Description

Date

Created on: Sep 06, 2012

Author

Camille Vuillaume

Copyright

Camille Vuillaume, 2012

7.6 src/tools.h File Reference

`#include <stdio.h> #include <gmp.h>`

Defines

- #define BIT2BYTE(a) (a+7)/8

Functions

- int debug_msg (const char ∗str)
- int gen_random (mpz_t rnd, mp_bitcnt_t bits)
- int gen_prime (mpz_t prime, mp_bitcnt_t bits)
- int crt_exponentiation (mpz_t result, mpz_t base, mpz_t exp_p, mpz_t exp_q, mpz_t pinvq, mpz_t p, mpz_t q)

7.6.1 Detailed Description

Date

Created on: Aug 25, 2012

Author

Camille Vuillaume

Copyright

Camille Vuillaume, 2012