

User Documentation for CryptoSMS.org Version 0.3

CryptoSMS.org

20/06/09

Contents

| | | |
|----------|--|-----------|
| 1 | About this Documentation | 3 |
| 1.1 | License of this Document | 3 |
| 2 | Short introduction to CryptoSMS | 5 |
| 2.1 | How it works in brief | 5 |
| 2.2 | Public / Private Keys | 5 |
| 2.3 | Differences to GPG/PGP | 6 |
| 2.4 | Supported languages | 7 |
| 3 | Documentation of functions | 9 |
| 3.1 | First time setup (Initialization) | 9 |
| 3.2 | Starting the program after Initialization or after timeout . . . | 10 |
| 3.3 | The main menu | 10 |
| 3.4 | Writing SMS | 11 |
| 3.5 | Receiving SMS | 13 |
| 3.6 | The Three Inboxes | 14 |
| 3.6.1 | Old Messages | 14 |
| 3.6.2 | Received Keys | 14 |
| 3.7 | Sent Messages | 16 |
| 3.8 | The Addressbook | 16 |
| 3.9 | The Info Screen | 17 |
| 3.10 | The Log Screen | 17 |
| 4 | Frequently Asked Questions | 19 |
| 4.0.1 | What if someone calls when the program is running? . | 19 |
| 4.0.2 | What if you receive an non-encrypted SMS while you are using Cryptosms? | 19 |
| 4.0.3 | What about incoming encrypted SMS while the pro- gram is not started? | 19 |
| 4.1 | Known Issues | 19 |
| 4.1.1 | Missing features | 19 |
| 4.2 | Memory Issues | 20 |
| 4.3 | Trust | 20 |

4.4 Planned Features 20

Chapter 1

About this Documentation

This documentation is version 0.3, written 20 June 2009
It refers to the software version 1.1.1 released 20 June 2009
Homepage of the project: www.cryptosms.org
Download address of the program to install: www.cryptosms.org/ota
Sourcecode is located at: codecoop.org/projects/CryptoSMS/
General contact: interest@cryptosms.org
Get our gpg key here: www.cryptosms.org/contact.html
For Developer, Release and User Support mailinglists, see www.cryptosms.org/lists.html
Feedback is very important! So please tell us about your experience, successful or not, with this software on your mobile phone: users@cryptosms.org
This documentation, as well as the installation documentation and the still to be written technical documentation, can be found at: www.cryptosms.org/docu.html
NOTE: CryptoSMS is an open-source software project, released under the GNU General Public License, Version 2. Any reference to CryptoSMS in this document means cryptosms.org.

Important Warning: you are using the fifth release of CryptoSMS, which is still a beta release. There are some very interesting features still missing and errors may occur. This software has not yet been widely tested – neither on a lot of phones, countries or networks. Please report every issue that occurs.

1.1 License of this Document

This document is licensed under Creative Commons Attribution-NonCommercial-ShareAlike 3.0. Please see <http://creativecommons.org/licenses/by-nc-sa/3.0/> for further details.

This Documentation in no way guarantees that it fully describes the software

cryptosms.org. Please report errors and missing stuff to interest@cryptosms.org.

Chapter 2

Short introduction to CryptoSMS

Cryptosms encrypts and decrypts SMS on your mobile phone. It aims to be as simple as possible, to also attract people who are not using encryption tools on their computers like PGP or GPG. We have tried to make the user interfaces as much as similar to the usual, plaintext SMS handling on mobile phones.

2.1 How it works in brief

CryptoSMS uses public / private key concepts. To encrypt a message to someone else, you need the public key of that person. If you don't have their key, CryptoSMS will not allow you to send an SMS to her. This prevents users from sending SMS in plaintext whilst thinking it was sent encrypted. Only your public key will be send in plaintext. Once you have received the recipient's key, any SMS to this person will be automatically encrypted and then sent. Receiving a SMS, that is encrypted to you, is almost the same as receiving a plaintext SMS, as you are not forced to enter a passphrase at that moment. The encrypted message will be automatically decrypted with your private key. Entering of a passphrase is needed when the program is started or after a time out.

This version of CryptoSMS supports fingerprint verification to authenticate received keys.

2.2 Public / Private Keys

Public / private key encryption (or asymmetric encryption) offers some advantages over symmetric encryption (most notably you don't need a shared

secret), where the encryption and decryption process is done with the same key and / or password. The problem with symmetric encryption is that once a third person gets hold of the encryption key, all messages can be decrypted. With public / private keys it is different. You can distribute your public key anywhere so that anyone may use it to encrypt a message to you. To decrypt the message which is encrypted using your public key, your private key is needed, which is in your possession only. The asymmetric encryption used by CryptoSMS has one disadvantage over most symmetric encryption schemes: it is not built for performance and therefore can take a while. On some phones it can take up to some minutes to encrypt a message, although this rarely the case. While CryptoSMS works on these phones, it makes the usage of CryptoSMS a bit unwieldy. However this issue will not play such a significant role in the future as mobile phones are released with greater processing power.

2.3 Differences to GPG/PGP

While the public / private key scheme is well known for users of PGP or GPG, there are differences between the encryption technologies of CryptoSMS and PGP/GPG, both from the user-point-of-view and from the mathematical point of view.

1: With CryptoSMS you don't need to type in a passphrase for message decryption, messages are automatically decrypted when CryptoSMS receives them. This makes the handling of decryption much easier on the limited interface that mobile phones have (imagine having to type in a strong 12 character passphrase into your mobile everytime you wanted to read a SMS). The passphrase is created by you as part of the configuration process when CryptoSMS is first started on your mobile. After this, the passphrase must be entered whenever the program is opened or a time out after a period of inactivity has occurred. This makes sure that only you can access the messages that are sent and received by CryptoSMS. The passphrase is also used for a second purpose, the encryption of the data in your mobile phone.

2: The algorithm used for the cryptography is completely different to what PGP/GPG uses. CryptoSMS is based on elliptic curve encryption. Using elliptic curves for encrypting data results in shorter keys than the ones used in PGP/GPG and results in better performance. Shorter keys are just as suitable for the targeted media SMS, which is limited to 160 characters. If you want to know more about elliptic curve encryption, see http://en.wikipedia.org/wiki/Elliptic_Curve_Cryptography.

CryptoSMS uses elliptic curves following the ECIES Standard and follows the implementation proposed in 'Guide to elliptic curve encryption' by Hankerson, Menezes, Vanstone (2004), Pages 189ff.

About ECIES, see: <http://en.wikipedia.org/wiki/ECIES>

CryptoSMS uses elliptic curve encryption from sending to receiving and first time reading SMS. The second time you read an encrypted SMS, CryptoSMS keeps the SMS encrypted on your phone (unless you delete it), but this time with AES256 symmetric encryption, a cipher widely used and known e.g for harddisk encryption. AES256 is very fast, so reopening of a SMS will be much faster than the first time.

Information for AES256 can be found at <http://en.wikipedia.org/wiki/Aes256>.

2.4 Supported languages

Cryptosms is designed to offer User Interface Support in a lot of languages. This means that any language that is encodeable in utf-8 may technically be supported. This release offers support for:

- English
- French
- German
- Italian
- Japanese
- Norwegian
- Polish
- Portuguese
- Russian
- Slovenian
- Spanish
- Swedish
- Turkish

Depending on the language settings in your mobile phone, CryptoSMS tries to use the selected language. If this language is not supported by the program, it uses English as the default. The languages supported by your phone depend highly on the region you are in, your telecom provider's settings and the manufacturer's settings.

(Note: if you are interested in integrating more languages or in translating this document, please contact interest@cryptosms.org)

Chapter 3

Documentation of functions

The following passages describe the usage of CryptoSMS. Please note: as mobile phones have a different handling, use different keys for e.g. *OK* buttons, CryptoSMS might look and feel different on your mobile to the description below. All pictures are taken from a Nokia 6230i screen, other phone's screens will look different. Even the key combinations how to reach some menu points will differ.

If you encounter problems in using CryptoSMS with your mobile, please contact users@cryptosms.org.

NOTE: Instead of Info Screens as shown in the screenshots, this version of CryptoSMS informs you about its actions done via a ticker in a light blue line above the main menu.

3.1 First time setup (Initialization)

The first time you start CryptoSMS, you will see the initialization screen.

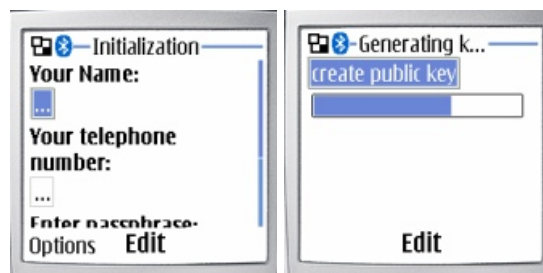


Figure 3.1: Initialization and Key Generating Screen

You are asked to enter your mobile phone number, your name (any name possible, but at least 2 characters), and a passphrase. Depending on your mobile, you will have to press the *OK* button to enter the fields or other buttons. Please try out. If the numbers or characters entered in the field

are not suitable, a message will pop up. Empty fields are not accepted. The passphrase must be at least 5 characters long, but not longer than 20 characters. We recommend at least eight characters. Any characters accessible through your phone's keypad are possible.

You have to type in the passphrase everytime you start the program and after a period of inactivity. So, find a good one that is not too hard to type but still not easy to guess. Do not use your pin code. This would be a too easy guess. The passphrase has to be typed in two times, to make sure you did not mistype it.

Note: As of this version, there is no possibility to change your passphrase or to recover it!

When finished, the generation of your public/private keypair begins. This is indicated by a (pseudo) progressbar. The progress bar looks different on different phones. Depending on the calculating speed of your mobile, this part takes between 10 seconds and 5 minutes. When it is done, you will see the main menu.

3.2 Starting the program after Initialization or after timeout

Only once you have to go through the initialization procedure: the first time you start CryptoSMS. From then on, the start of the program is different: you are asked to enter your passphrase. Passphrase entry is view protected by stars. Only the character you are actually typing is displayed in clear text. When finished, press *OK* or, depending on your mobile, via *Options* and then *OK*.

If you typed in the wrong passphrase you will see an error screen and you have three more times to enter the right passphrase. If you fail four times, the program quits.

If your program was idle for some time (no input activity) you have to reenter the passphrase as well.

3.3 The main menu

The main menu consists of eight parts which are described in the following sections:

- Write SMS
- New Messages
- Old Messages
- Sent Messages



Figure 3.2: Entering the Passphrase and the Main Menu

Note: *Outbox* as shown is now called *Sent Messages*. *Clear Memory* is not part of this release anymore.

- Addressbook
- Received Keys
- Info
- Log
- Exit application

3.4 Writing SMS

This is the direct way to write and send SMS. To do so, enter the menu entry *Write SMS* by typing the *OK* button (or via *Options*, then *OK* on some mobiles). You then see a text input box similar to the one from the standard SMS program on your mobile phone. Note: the length of an encrypted SMS is limited to 100 characters, as the encrypted sms contains some overhead. Depending on your mobile phone, you may use text writing features such as *T9* or fast selection of special characters. Once you finished writing, press *OK* or *Options* and then *OK* depending on your mobile.

The next screen allows the selection of the recipient. If you choose the highlighted entry, this will be the recipient.

Note: you can only send encrypted messages, if you have the recipient's key.

Via *Options* you can choose between four options in *Select Address*:

- Send Message
- New entry
- Edit entry
- View

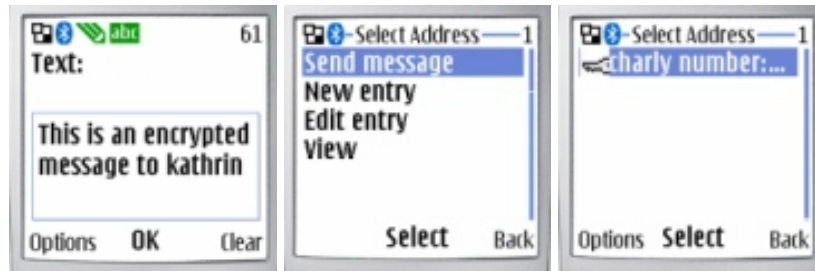


Figure 3.3: The Message Text Field and the Selection of the Message's Recipient

Note: SMS can only be send if you have the recipient's key.

New Entry lets you add an entry to your addressbook, but this makes only sense, if you have already received the person's key – otherwise you can not send this sms to that person!

Edit Entry lets you change the settings for the chosen recipient, e.g. change the name or number.

View show you the complete entry of the chosen recipient.

Once you have chosen a recipient, CryptoSMS starts the encryption of the SMS for the receipt. Depending on your mobile phone, this may take between 10 seconds and some minutes. Your mobile is now heavily calculating points on elliptic curves...

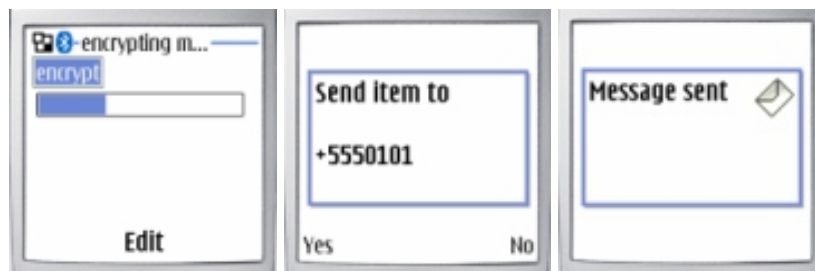


Figure 3.4: Encryption in progress, Confirmation Screen to send SMS and Message Sent Info

Depending on the trust model (see *Known Issues*), you might be asked something like *Send item to* and then the recipient or *allow to send SMS?* after encryption.

Once you confirmed the sending of your SMS, a screen confirms that the message was sent.

3.5 Receiving SMS

There are three different scenarios about receiving an encrypted SMS:

- Your phone is switched off
- Your phone is switched on but CryptoSMS is not running
- Your phone is on and CryptoSMS is running

All three scenarios are handled by CryptoSMS. When your provider wants to transmit an encrypted sms to you, it is held back as long as you start CryptoSMS. It will not be transmitted into your plaintext SMS inbox. How long the encrypted SMS is provided for you by your provider only depends on the provider's policy on SMS in general and has nothing to do with CryptoSMS.

This release of CryptoSMS does not notify you about a received SMS via sounds or vibrations. All you get is a short notice on the display of your mobile, offering to start CryptoSMS.

If you are using CryptoSMS while a new encrypted SMS comes in, you are notified via the floating text in the top of the screen. The new SMS can then be found in the *New Messages* folder via main menu. The message is here referenced with the sender's name (if it is in your addressbook, else the phone number), time and date of reception. If the SMS contains a key and not a content SMS, it will not be found in *New Messages* but in *Received Keys* via the main menu!

Via *Options* in *New Messages* you may:

- View the message
- Delete the message without having read it
- View details about the message

If you chose to view the message, the decryption starts. Depending on your mobile, this may take between 5 seconds and some minutes.

Note: only new messages take long to decrypt. Once they have been decrypted and read, they are automatically transferred into the *Old Messages* folder. The contents of his folder (as all data that is stored on your phone and belongs to CryptoSMS) is encrypted symmetrically and decrypts much faster.

When the decryption finished, the displays show the content of your received SMS. By pressing *OK* the message automatically moves into the *Old Messages* folder. Therefore, the *New Messages* folder will be empty then again



Figure 3.5: New Message Info Screen, Decryption in Progress and the decrypted Message

until the next encrypted SMS arrives.

Via *Options* you have a shortcut to reply directly without typing in the number of the receiver. Now type your reply and send it.

3.6 The Three Inboxes

As mentioned before, CryptoSMS provides three different inboxes, one called “New Messages“ , one called “Old Messages“ and one called “Received Keys“. Every new received encrypted SMS that contains text can be found in the “New Messages“ (see “Receiving Messages“). Every received key can be found in “Received keys“ and every already read text message can be found in “Old Messages“.

3.6.1 Old Messages

As every once read message is sorted into the *Old Messages* folder, you will find most of your messages that you have not deleted manually here. As these messages are stored encrypted by a symmetric encryption algorithm (aes256), which is a fast one, to view a message is much faster than viewing a message in *New Messages*. Via *Options* you have the following possibilities:

- View a message
- Delete a message
- View more information about the Message

3.6.2 Received Keys

Via the main menu, you find the entry *Received Keys*. This folder contains a list of all public keys that people have sent to you. This list is important as you can only write encrypted messages if you have the recipient’s key. The

keys in this list have to be deleted manually, otherwise they will remain in this list. In case you deleted an address entry, you can still import the key a second time.

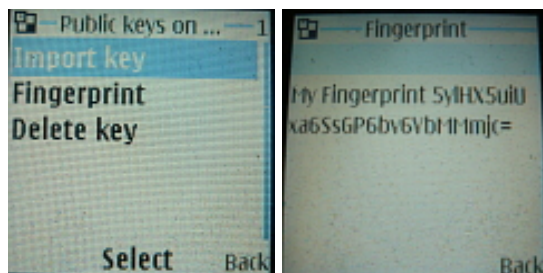


Figure 3.6: Options Screen for a received Key and the Fingerprint of a key.

The following options are available:

- Import key
- Fingerprint
- Delete key

Fingerprint shows you the Fingerprint of the received key in Base64 encoding. The Fingerprint of the received key is a means to make sure that you received the actual key and not a forged one. To ensure there has been no man in the middle sending you a fake key, you should exchange the Fingerprint of the received key with the person that has sent the key to you. Do this via calling the person or any other channel of communication that ensures you are communicating with the person. Face to face Fingerprint comparison is also a good method. If you want to know more about Fingerprints and why they are important in cryptography, see http://en.wikipedia.org/wiki/Public_key_fingerprint. CryptoSMS uses SHA1 hashes for fingerprints.

Note: before importing a key, there has to be an entry in your addressbook with the same phonenumber that is referenced by the key! If CryptoSMS does not find a matching number, import fails. In case there is an entry, but still the import fails, check whether the number in the addressbook entry is written in national or international style (e.g. 0049 or +49 at the start). With this release, CryptoSMS may have problems in recognizing international style writing of phonenumber. Change the Adressbook entry exactly to the number displayed for the received key. The number the key contains is the reference. Correct the number in your addressbook entry that it corresponds to the number that is displayed in the received keys list. Otherwise CryptoSMS will not be able to combine key and addressbook entry.

3.7 Sent Messages

Then there is the *Sent Messages* option in the main menu. Here you find all messages that you have sent via CryptoSMS, including the messages that contained only your keys that you have sent to people.

Via *Options* there are the following possibilities:

- View a message
- Delete a message
- View more information about the message

All functions work similar to the functions in *Old Messages*. As well, all messages in your Sent Messages folder are symmetric encrypted, so decryption for viewing them is fast.

3.8 The Addressbook

Via the main menu, you can enter your addressbook. This is the most important list of CryptoSMS. Opened the first time, it contains no addresses at all.

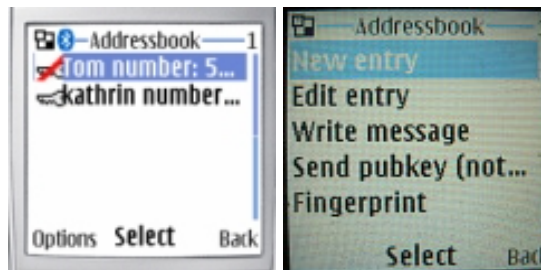


Figure 3.7: Addressbook containing first entry without and second entry with a key, the menu for an addressbook entry.

See the picture on the left. There is one entry without a key icon (Tom) and on that has the key icon (Kathrin).

You have the following options:

- New Entry
- Edit Entry
- Write Message
- Send Pubkey
- Fingerprint

- Import key
- Delete address
- View

To add a new entry to your addressbook, enter *New Entry* and type in the name and number. When finished, press *OK* or via options *OK*.

Note: An entry without a corresponding key will only allow sending your public key to this person, not a SMS!

Edit Entry allows to change existing entries in your addressbook.

Write Message lets you write an encrypted SMS to that person (if you have the corresponding key).

Send Pubkey offers to send your public key to any entry in your addressbook, whether you have the person's key already or not. Exchange of keys is the first step before you may communicate with a person securely. You will use this option quite often in the beginning.

Fingerprint shows you the fingerprint of the key that is attached to this addressbook entry.

Import Key lets CryptoSMS look up all keys in your *received keys* list. If a key matches against the highlighted addressbook entry, the key will be imported (and also remain in your *Received keys* list).

Delete Address: yes! This deletes the address!

View shows more information about this Addressbook entry.

3.9 The Info Screen

In the main menu you find an entry named *Info*. Entering this you will see the name that you have typed in for yourself during initialization, your phone number, the version of CryptoSMS that is running on your device the date this version of CryptoSMS was build and the SMSC of your mobile phone provider.

Also here you can check to see the fingerprint of your public key. The fingerprint is represented in Base64 Encoding.

3.10 The Log Screen

Following the *Info* entry in the main menu is the *Log* entry. This entry informs about processes of CryptoSMS regarding sending and receiving SMS. This log will be deleted once you quit the program, it is only a session log. It is mainly for debugging purposes or for the interested user, but is has no features besides that.



Figure 3.8: The Log of the current CryptoSMS session.

Chapter 4

Frequently Asked Questions

4.0.1 What if someone calls when the program is running?

If someone calls while CryptoSMS is running, the program will halt and pause operations automatically. After you finished your call, it will continue running. This includes all operations of CryptoSMS. Possibly you are asked to type in the passphrase again.

4.0.2 What if you receive an non-encrypted SMS while you are using Cryptosms?

Any other media that your mobile receives while CryptoSMS is running will be stored in its default place, e.g. the usual messages folder outside CryptoSMS. You may access it by quitting CryptoSMS or, depending on your mobile, through some key combinations while CryptoSMS is running.

4.0.3 What about incoming encrypted SMS while the program is not started?

Your mobile notices that it is an CryptoSMS that is registered to be used by CryptoSMS and then asks you to start CryptoSMS.

4.1 Known Issues

4.1.1 Missing features

With this release, CryptoSMS does not support the following features well known from the plaintext SMS program on your mobile or from other crypto software:

- Sending of one SMS to more than one recipient.
- Saving and loading of a text that you want to send later (or again).

- Exporting keys to e.g. your Desktop Computer for Backup purposes
- Change of your public/private key pair
- Change of your passphrase

4.2 Memory Issues

Cryptosms file size is 144 kB in this version. Some older phones which meet the java criteria for CryptoSMS only support java software below that.

4.3 Trust

CryptoSMS is not a commercial product and does not come with a commercial certificate. Your mobile will not recognize CryptoSMS as a trusted program because of this and therefore will not allow sending of SMS automatically. You may change this via Options in the folder where you installed CryptoSMS, depending on your mobile. Most likely, you will be ask to allow CryptoSMS the sending of SMS for every single SMS. This is not convenient. Maybe later versions of CryptoSMS will have solved this.

4.4 Planned Features

Besides the missing features in “Known Issues“, very interesting features would be

- Key exchange via bluetooth
- CryptoSMS exchange via bluetooth
- the same functionality CryptoSMS provides for other media formats, e.g. MMS
- device specific builds of CryptoSMS for better graphical user interfaces, sound suport...

Maybe the next release will have some of these features. If you are interested in participating in coding CryptoSMS, please contact developers@cryptosms.org!

Check out the sources of CryptoSMS at codedoop.org/projects/CryptoSMS!