

# Table of Contents

<b>Part I Introduction</b>	<b>3</b>
1 What is SecExMail ? .....	3
2 Quick Start .....	6
3 Acknowledgements .....	7
<b>Part II Configuration</b>	<b>11</b>
1 Starting SecExMail .....	11
2 Mail server tab .....	12
3 Encryption tab .....	13
4 Desktop tab .....	14
5 My Keys tab .....	15
6 Friends tab .....	20
7 In-Tray Tab .....	24
8 Out-Tray Tab .....	25
9 Watch tab .....	25
10 Distributing SecExKeys .....	26
11 Send Mail Authentication .....	27
12 Passphrase cache .....	27
13 Outgoing Mail Filter .....	28
14 Incoming Mail Filter .....	30
15 Configuration Wizard .....	31
16 Automatic Key Exchange .....	34
<b>Part III Keys</b>	<b>34</b>
1 Create your personal SecExMail keys .....	34
2 Personal Details Screen .....	35
3 Key Size Screen .....	36
4 Passphrase Screen .....	36
5 Entropy Screen .....	37
6 Progress Screen .....	38
<b>Part IV Email Clients</b>	<b>38</b>
1 Basic Email Client Configuration .....	38
2 Netscape Mail .....	39
3 Modifying Netscape Mail .....	44
4 KMail .....	48

5	Ximian Evolution .....	50
---	------------------------	----

## **Part V Technical 52**

1	RSA Public Key Encryption .....	52
2	ISAAC Random Number Generator .....	53
3	The SecExMail Cipher .....	53
4	SecExMail Message Format .....	55
5	SecExMail Key Transparency .....	57
6	SecExMail Keys .....	57
7	SecExMail Key File Format .....	58
8	Entropy Collection .....	59
9	One-Time Pads .....	63
10	IP / DNS Spoofing .....	63
11	Known Plain Text Attack .....	64

## **Part VI FAQ 65**

1	What email clients work with SecExMail ? .....	65
2	Does SecExMail work with IMAP? .....	65
3	How secure are SecexMail keys ? .....	65
4	Is SecExMail legal in my country ? .....	65
5	Does SecExMail support signatures ? .....	66
6	Does SecExMail work with PGP ? .....	66
7	Is the source code available for SecExMail ? .....	66
8	Why can I not mix clear text and cipher recipients ? .....	67

## **Part VII Just Linux 67**

1	WINE Configuration .....	67
2	KDE Autostart Menu .....	74
3	Font & Display Issues .....	76

## **Part VIII About 77**

1	About SecExMail .....	77
2	About Bytefusion Ltd. ....	77
3	Requirements .....	78

# 1 Introduction

## 1.1 What is SecExMail ?



### **SecExMail - Secure Email Made Easy**

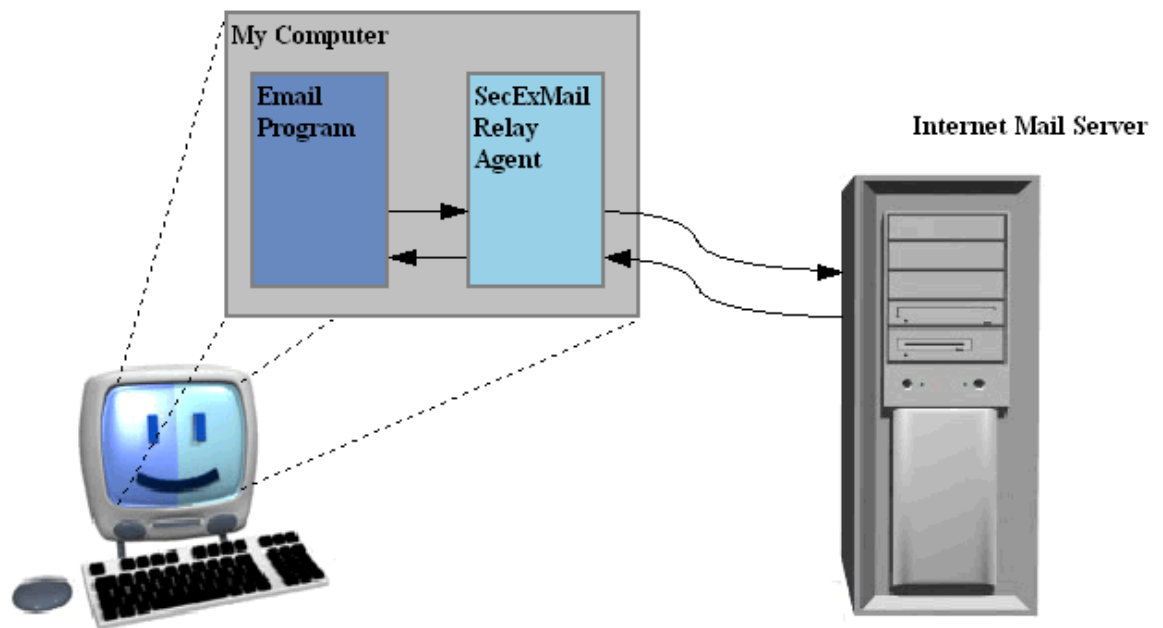
Maintaining your privacy on the net is not easy in today's world. Until now, secure email could only be achieved by encryption solutions that were not user friendly. SecExMail changes all that, bringing you secure encrypted email that is so easy to use, you will forget it is there!

### **Advantages at a Glance**

- Auto key management - no more sending public keys and having to import your friend's keys
- Easy to use - no need to change how you work in order to start sending secure email
- Automatic - SecExMail encrypts messages if you have the recipients key, otherwise the message is sent in plain text
- HTML Filtering - Everybody gets spam and some of it contains malicious code or "call home" features so the spammer knows you opened the mail and will send more spam. SecExMail can filter out the HTML tags so you see plain, harmless text.
- Interoperates with users of Crypto Anywhere, Secure Information Courier and SecExMail on any platform.

SecExMail implements open standard encryption algorithms to create a seamless security framework to protect the privacy of your email on the public internet. SecExMail operates as a relay agent between your email client to your mail server encrypting and decrypting email streams in real-time and therefore requires no additional plugin software for your favorite email client program.

### **How SecExMail works :**



### **General Features :**

- Easy to configure

SecExMail requires no plugins or other email client specific software. Simply configure SecExMail to communicate with your email server and set your favorite email client to talk to SecExMail. That's it.

- Seamless integration

Probably the greatest obstacle to wide-spread use of secure email is that most encryption systems don't integrate seamlessly with popular email clients. In some cases the use of plugins means that encrypted messages held in mail folders are not searchable via the standard email client interface. In other cases, encrypted messages held in mail folders become irretrievable once the encryption plugin is unloaded. In most cases security is an after-thought and the normal work flow is disrupted to accommodate security. Because SecExMail operates unobtrusively in the background, encrypting and decrypting email streams to and from your email client in real-time, you continue to work with your email client as usual.

- Fail Safety

Many plugin based encryption systems require the user to treat secure mail differently from ordinary mail. Some require the user to remember that mail to a specific recipient should always be encrypted and take special action to invoke the encryption. If the user forgets, the message is sent in plain text and confidential information may be compromised. Equally, if a plugin is accidentally unloaded or crashes, sensitive information may be compromised because messages designated secure are inadvertently spilled onto the internet in clear text. SecExMail is engineered from the ground up to provide fail safety. Because SecExMail acts as a relay agent or mail proxy and requires the email client to be configured to communicate with its mail server via this proxy, a failure in SecExMail simply means that no mail is sent until the error condition is alleviated. Once the public key for a particular recipient is entered into SecExMail, all mail to that recipient will be sent encrypted by

default and without further user intervention.

- Proactive Security

SecExMail does not stop at simply encrypting your email messages. It also provides for message stealth at the protocol level. The information contained in the header of most emails provides a wealth of information to the cryptanalyst. For example, the header contains a subject line which tells the cryptanalyst which messages are worth examining. Furthermore the header contains information about the type of message being sent, the so called "MIME type". The MIME type indicates to the cryptanalyst if the message contains only text or perhaps a photograph and if so in what format the photograph is stored ( JPG, GIF, etc ). The latter can be exploited in a [known plain text attack](#). For this reason, SecExMail not only encrypts the message subject but also obscures MIME type information. This means a hacker can neither deduce whether the message is worth examining nor what file attachments, if any, are being sent.

- Easy Key Handling

Exchanging keys with friends and business associates is child's play. SecExMail has a built in "one touch" SMTP client and automatic key update feature to facilitate easy distribution of SecExMail keys.

- Protect Account Information

Most conventional email communication involves the exchange of clear text passwords. This means that anyone with the right wire tapping equipment, or in fact any skilled system administrator working for your telecommunications company, can collect your password information and subsequently read all your email without your knowledge. SecExMail can protect your user and password information by encapsulating all email traffic in a Secure Socket Layer ( SSL ) or Transport Layer Security (TLS) tunnel. See [Mail Server](#) configuration for details. (Offshore and Corporate edition only )

- Trojan Horse Protection

SecExMail protects against attempts to trick you into revealing your password information to third parties. See IP/DNS spoofing for technical details. (Offshore and Corporate edition only )

- Key Transparency

SecExMail is engineered with a focus on transparency to give you the assurance that no backdoor keys or key recovery is embedded in encrypted messages. See [Key Transparency](#).

### **Technical Features :**

- Public Key Encryption

SecExMail uses standard [RSA based public key encryption](#). Supported key sizes are 2048, 4096 and 8192 bits ( up to 10240 bits for offshore edition and corporate edition ). Two messages are never encrypted with the same session key. Instead the public key associated with the recipient of a message is used to encrypt a random session key which is used to encrypt the message. Generation of strong session keys is based on a sophisticated [entropy collection system](#).

- Message Encryption

Individual messages are double encrypted via 64 bit ISAAC and 256 bit Twofish encryption .  
[See SecExMail Cipher](#).

- Coexistence with other encryption standards

SecExMail encrypts the mail stream and therefore does not interfere with existing methods of encryption. As such, it is possible to encrypt with PGP or GPG first, and then send the resulting cipher text through SecExMail for further encryption. On the remote end, the recipients SecExMail restores the PGP cipher text which can then be decrypted by the user's email client or associated PGP decryption module.

## 1.2 Quick Start



**This is the "manual" quick guide to getting you up and running with SecExMail !**

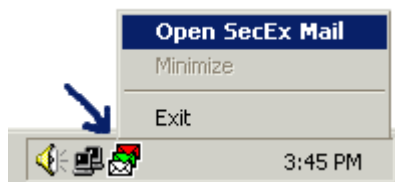
For information on wizard based setup, see [Configuration Wizard](#).

- **Step 1**

Start the SecEx Key Generator and create your own SecExMail keys. [Click here to learn how.](#)

- **Step 2**

Open SecExMail by right-clicking the SecExMail icon in bottom right corner of your screen as shown below and [configure your mail server](#) . [Click here to learn how.](#)



If you cannot see the SecExMail icon in bottom right corner of your screen, you will need to [start SecExMail manually](#) first.

- **Step 3**

Configure your email client program to use SecExMail as a relay agent. [Click here to learn how.](#)

- **Step 4**

Send your SecExMail key to your friends. [Click here to learn how.](#)

- **Step 5**

Tell your friends about SecExMail and let SecExMail defend your privacy !

## 1.3 Acknowledgements

- **ISAAC Random Number Generator**

At the time of writing, the ISAAC home page can be found at <http://burtleburtle.net/bob/rand/isaacafa.html>.

ISAAC has been placed into the public domain by its author, Bob Jenkins in 1996.

-----  
My random number generator, ISAAC.  
(c) Bob Jenkins, March 1996, Public Domain  
You may use this code in any way you wish, and it is free. No warrantee.  
-----

- **RSA Public Key Encryption**

The RSA algorithm was patented until September 2000 when RSA® Security Inc. released the algorithm into the public domain. *"BEDFORD, Mass., September 6, 2000 -- RSA® Security Inc. (NASDAQ: RSAS) today announced it has released the RSA public key encryption algorithm into the public domain, allowing anyone to create products that incorporate their own implementation of the algorithm."* At the time of writing a copy of this statement can be found at <http://www.rsasecurity.com/news/pr/000906-1.html>

- **Twofish Block Cipher**

The Twofish block cipher by Counterpane Labs was developed and analyzed by Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall and Niels Ferguson. Twofish was one of the five Advanced Encryption Standard finalists. At the time of writing the Twofish homepage can be found at <http://www.counterpane.com/twofish.html>. The cipher has been made available to the general public by the following statement on <http://www.counterpane.com/about-twofish.html> :

" Twofish is unpatented, and the source code is uncopyrighted and license-free; it is free for all uses. Everyone is welcome to download Twofish and use it in their application. There are no rules about use, although I would appreciate being notified of any commercial applications using the algorithm so that I can list them on this website. "

- **ZLIB Compression Library**

ZLIB is a lossless data-compression library written by Jean-loup Gailly and Mark Adler. ZLIB is made available as free, unpatented software to the general public at <http://www.gzip.org/zlib/>. The license

conditions are set forth at [http://www.gzip.org/zlib/zlib\\_license.html](http://www.gzip.org/zlib/zlib_license.html) and reproduced below :

```
" Copyright (C) 1995-2002 Jean-loup Gailly and Mark Adler
```

```
This software is provided 'as-is', without any express or implied  
warranty. In no event will the authors be held liable for any damages  
arising from the use of this software.
```

```
Permission is granted to anyone to use this software for any purpose,  
including commercial applications, and to alter it and redistribute it  
freely, subject to the following restrictions:
```

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

```
Jean-loup Gailly jloup@gzip.org  
Mark Adler madler@alumni.caltech.edu "
```

- **RIPEMD-160**

The RIPE message digest was written by Antoon Bosselaers for Katholieke Universiteit Leuven, Department of Electrical Engineering ESAT/COSIC, Belgium. License conditions ask us to quote the following :

```
" RIPEMD-160 software written by Antoon Bosselaers,  
available at http://www.esat.kuleuven.ac.be/~cosicart/ps/AB-9601/ "
```

- **Viking Art - SecExMail Logo**

Katja Bengtsson of Stockholm, Sweden ( [katja@offshoremailroom.com](mailto:katja@offshoremailroom.com) )

- **Embedded Mozilla Browser Component**

SecExMail contains Mozilla web browser software from the *The Mozilla Organization*. Mozilla is licensed under the Mozilla Public License (MPL) which can be found at <http://www.mozilla.org/MPL/MPL-1.1.html>. SecExMail uses unmodified Mozilla executable code; the source code for this executable code is freely available at <http://www.mozilla.org>. Software distributed under the MPL is distributed on an "AS IS" basis, without warranty of any kind, either express or implied. See the [MPL License](#) for the specific governing rights and limitations.

- **OpenSSL Project**

SecExMail contains cryptographic software from the OpenSSL project at [www.openssl.org](http://www.openssl.org) which is licensed under a "BSD-style" open source licenses. These licenses asks us to state the following :

```
"This product includes software developed by the OpenSSL Project for use  
in the OpenSSL Toolkit. (http://www.openssl.org/) "
```

"This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com)."

SecExMail is an independent, derived work and no endorsement of SecExMail by the OpenSSL project is implied. The full text of the OpenSSL license and the original SSLeay License is reproduced below.

#### OpenSSL License

=====  
Copyright (c) 1998-2001 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment:  
"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)"
4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact [openssl-core@openssl.org](mailto:openssl-core@openssl.org).
5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment:  
"This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=====  
This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Original SSLeay License

Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)  
All rights reserved.

This package is an SSL implementation written  
by Eric Young (eay@cryptsoft.com).  
The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as  
the following conditions are adhered to. The following conditions  
apply to all code found in this distribution, be it the RC4, RSA,  
lhash, DES, etc., code; not just the SSL code. The SSL documentation  
included with this distribution is covered by the same copyright terms  
except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in  
the code are not to be removed.

If this package is used in a product, Eric Young should be given  
attribution

as the author of the parts of the library used.

This can be in the form of a textual message at program startup or  
in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without  
modification, are permitted provided that the following conditions  
are met:

1. Redistributions of source code must retain the copyright  
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright  
notice, this list of conditions and the following disclaimer in the  
documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software  
must display the following acknowledgement:  
"This product includes cryptographic software written by  
Eric Young (eay@cryptsoft.com)"  
The word 'cryptographic' can be left out if the routines from the  
library  
being used are not cryptographic related :-).  
4. If you include any Windows specific code (or a derivative thereof)  
from  
the apps directory (application code) you must include an  
acknowledgement:  
"This product includes software written by Tim Hudson  
(tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND  
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE  
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR  
PURPOSE

ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE  
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR  
CONSEQUENTIAL

DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS  
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)  
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,  
STRICT

LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY  
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF  
SUCH DAMAGE.

The licence and distribution terms for any publicly available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

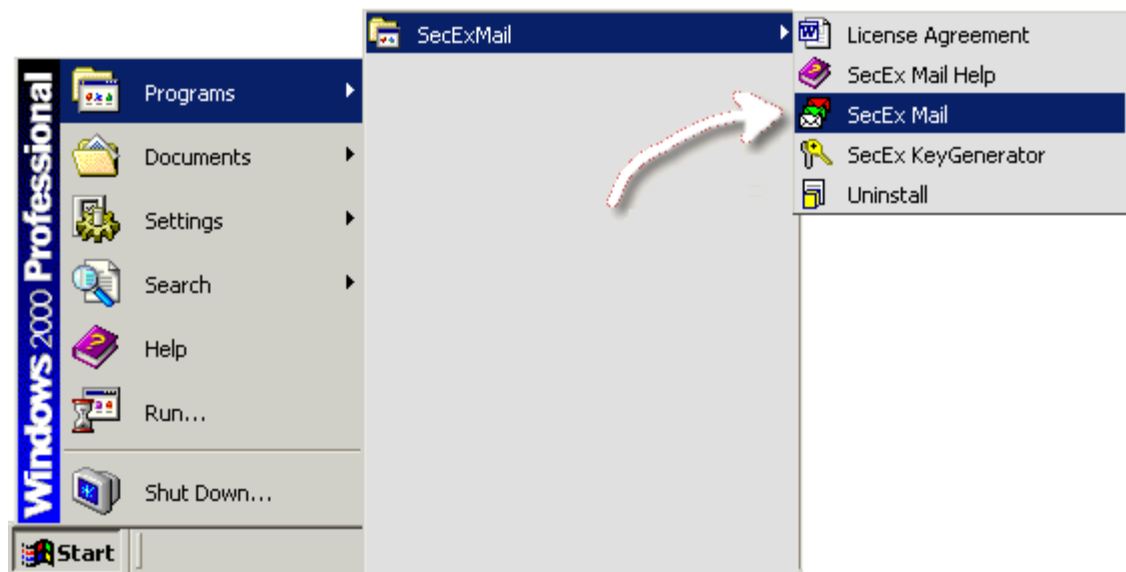
- **SecExMail Cipher**

Chris Kohlhepp and Mark Robertson, Bytefusion Ltd.

## 2 Configuration

### 2.1 Starting SecExMail

To start SecExMail, click "**Start**", "**Programs**", "**SecExMail**" and "**SecEx Mail**" as shown below. This will start the SecExMail service and place the icon into the system tray.

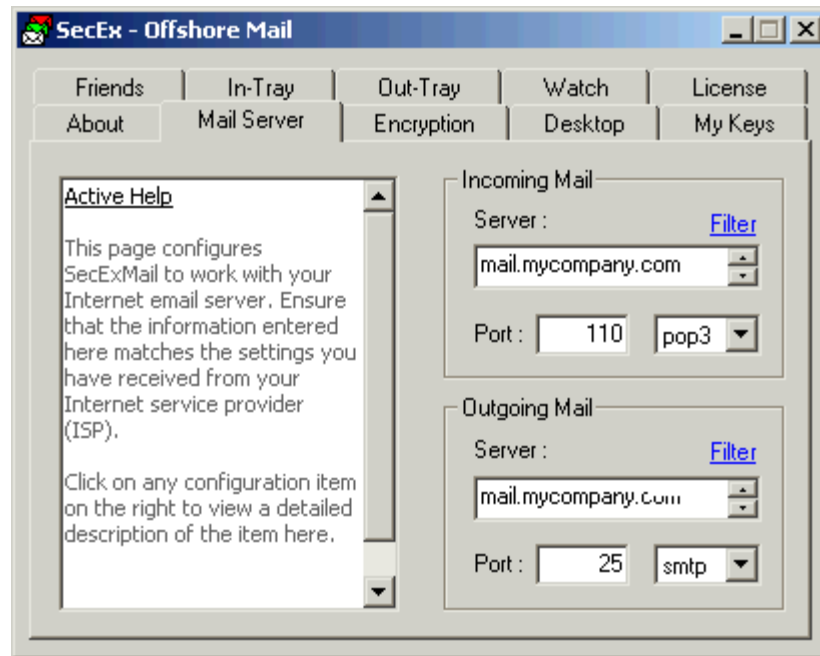


Right-click on the system tray icon with your mouse and select "**Open SecEx Mail**" as show below to open the configuration screen.



## 2.2 Mail server tab

SecExMail operates as a [relay agent](#) between your email client software and your internet service provider's mail server. Therefore you need to tell your email client to check and send mail via SecExMail and tell SecExMail about your mail server. To do so, select the **Mail Server** tab and enter the details provided for SMTP server and POP3 server by your internet service provider ( ISP ) as shown below.



- **Incoming Mail Server**

This is the Internet email server which stores your incoming mail. You may specify either a DNS host name or IP address here. The server must be POP3 compliant. Modifying this setting will have immediate effect.

- **Incoming Mail Port**

If you are behind a firewall and your incoming mail server operates on a non-standard port, please update the POP3/POP3S port setting here accordingly. Modifying this setting will have immediate effect.

- **Incoming Mail Protocol**

If your service provider supports secure POP3 (POP3S) via Secure Socket Layer (SSL) or Transport Layer Security (TLS), set the port type here to "pop3s". Otherwise set the port type to "pop3". The standard port for POP3 is 110. The standard port for POP3S is 995. POP3S not available in Home Edition. Modifying this setting will have immediate effect.

- **Outgoing Mail Server**

This is the Internet email server you use to send outgoing mail. You may specify either a DNS host name or IP address here. The server must be SMTP compliant. Modifying this setting will have immediate effect.

- **Outgoing Mail Port**

If you are behind a firewall and your outgoing mail server operates on a non-standard port, please update the SMTP/SMTPS port setting here accordingly. Modifying this setting will have immediate effect.

- **Outgoing Mail Protocol**

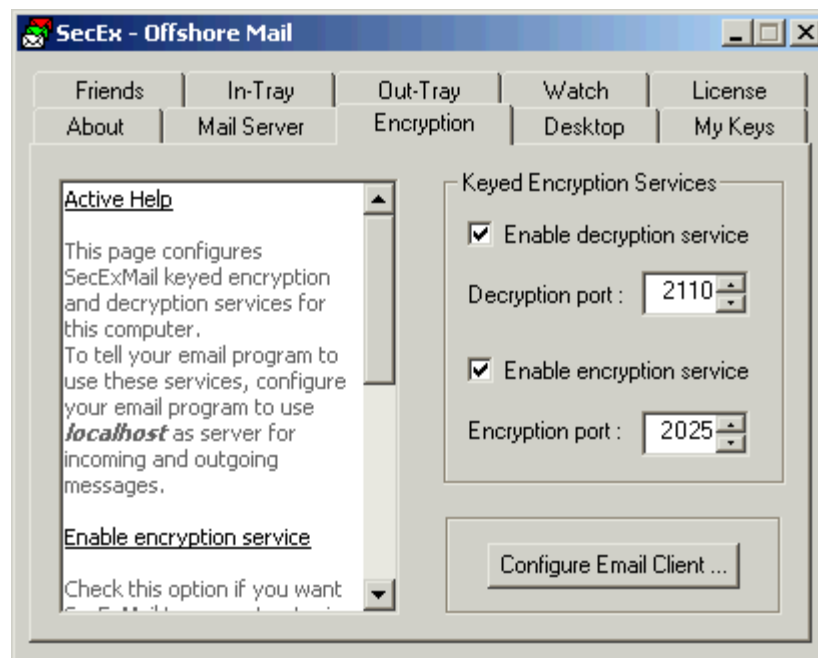
If your service provider supports secure SMTP (SMTPS) via Secure Socket Layer (SSL) or Transport Layer Security (TLS), set the port type here to "smtps". Otherwise set the port type to "smtp". The standard port for SMTP is 25. The standard port for SMTPS is 465. Note that only \*fully\* encrypted SMTP is supported here. This feature does not support RFC 2487 demand encryption of SMTP over port 25. SMTPS not available in Home Edition. Modifying this setting will have immediate effect.

- **Filter**

Click [Filter](#) to configure the [SMTP filter](#).

## 2.3 Encryption tab

The Encryption tab allows you to configure SecExMail keyed encryption services for your computer. The default values set by the installation program will serve most people. However, if you operate a firewall or other proxy service on your computer, you might have to adjust these settings. All users will need to [configure their email client](#) to work with SecExMail.



- **Enable encryption service**

Check this option if you want SecExMail to encrypt outgoing messages when sending mail to recipients listed on "Friends" page. Modifying this setting will take effect the next time you restart SecExMail.

- **Decryption Port**

This specifies the POP3 port your email client will connect to on your computer (localhost) when retrieving new messages from the internet. Modifying this setting will take effect the next time you restart SecExMail.

- **Enable decryption service**

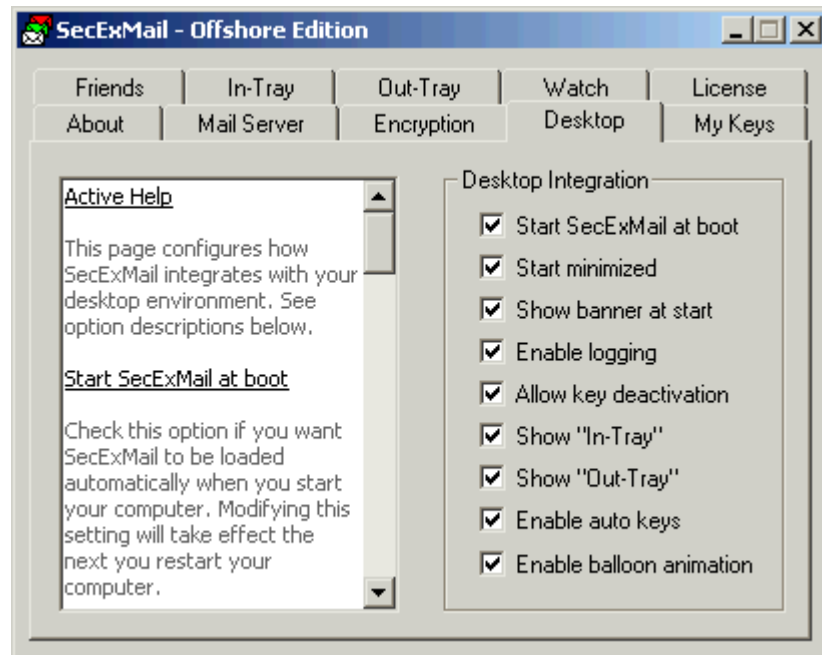
Check this option if you want SecExMail to scan incoming mail for SecExMail encrypted messages and decrypt them. Modifying this setting will take effect the next time you restart SecExMail.

- **Encryption Port**

This specifies the SMTP port your email client will connect to on your computer (localhost) when sending messages. Modifying this setting will take effect the next time you restart SecExMail.

## 2.4 Desktop tab

This page configures how SecExMail integrates with your desktop environment. See option descriptions below.



- **Start SecExMail at boot**

Check this option if you want SecExMail to be loaded automatically when you start your computer. Modifying this setting will take effect the next you restart your computer.

- **Start minimized**

Set this option if you want SecExMail to be minimized into the system tray when started. Modifying this setting will take effect the next time you restart SecExMail.

- **Show banner at start**

Enables SecExMail splash screen at program start. Modifying this setting will take effect the next time you restart SecExMail.

- **Enable logging**

Set this option if you want SecExMail to write its "Watch" tab log to disk. Modifying this setting will take effect the next time you restart SecExMail.

- **Allow key deactivation**

Check this option if you want to be able to disable keys without deleting them. Modifying this setting will have immediate effect.

- **Show "In-Tray"**

Set this option if you want to be able to view the "raw content" of inbound messages. Modifying this setting will have immediate effect.

- **Show "Out-Tray"**

Check this option if you want to be able to view the "raw content" of outgoing messages. This is useful to view the encrypted versions of messages you have sent. Modifying this setting will have immediate effect.

- **Enable Auto Keys**

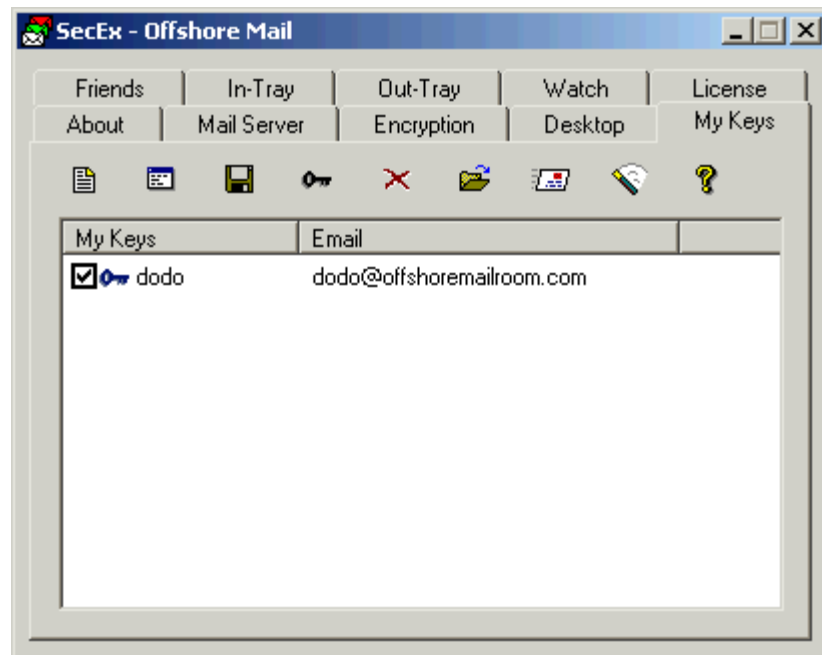
Check this option if you want SecExMail to exchange keys with other users automatically. If this option is enabled, SecExMail will append your public key to outgoing e-mails as a tag line and accept such public key updates from other people. Once secure communication with another person is established, your public key is no longer appended to emails sent to that person.

- **Enable balloon animation**

Check this option if you want SecExMail to notify you via small pop-up screens when encrypting and decrypting mail.

## 2.5 My Keys tab

The **My Keys** tab lists your own SecExMail keys. From here you can create new keys, change the passphrase on existing keys, back up and restore keys to and from disk, email your key to others and manage the passphrase cache. Technically, keys shown here are private keys while keys listed on the [Friends tab](#) are public keys. See [SecExMail Keys](#) for details.



#### ☒ Key Activation State :

Use this check box to enable / disable encryption to the person associated with the respective key. If unchecked, messages sent to the key owner via SecExMail will be transmitted in the clear. If checked, messages sent to the key owner via SecExMail will be encrypted. Since the **My Keys** tab shows your own keys, this feature will mainly affect messages you carbon copy or "CC" to yourself.

#### New Key Button

Use this button to generate a new key for yourself. This will invoke the SecEx Key Generator as shown below. Technically the SecEx Key Generator will generate a two component key for you, comprised of one public key and one private key. The private key will appear here on the **My Keys** tab and will be used to decrypt incoming messages. The public key component will be sent to your friends so as to enable your friends to encrypt messages to you. It is also used to encrypt messages you carbon copy or "CC" to yourself. See [SecExMail Keys](#) for details.



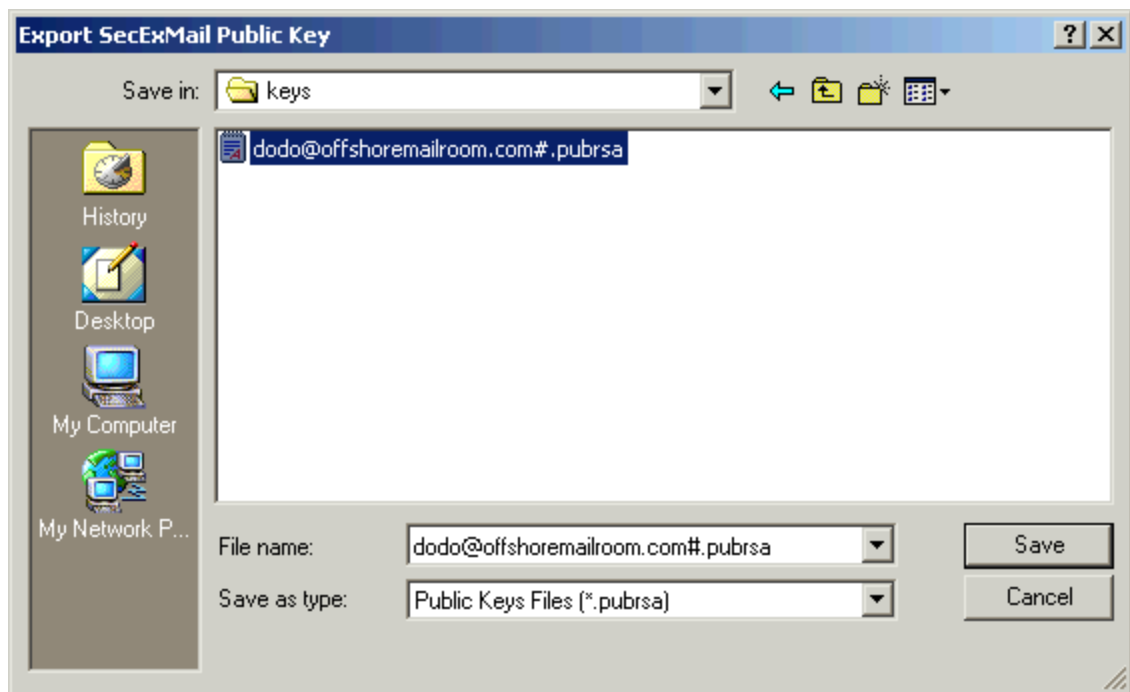
 **Key Properties Button :**

Use this button to display the properties of a selected key. The key properties include details of the key owner, the email address associated with the key, the key type and size as well as the key fingerprint. The properties for a typical key are shown below.

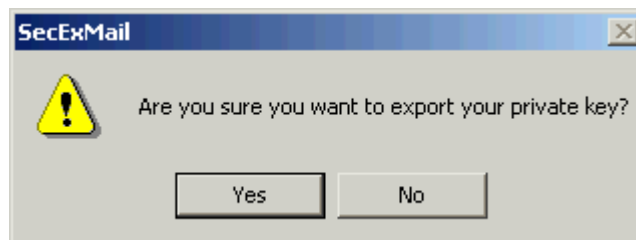
**Export Key Button :**

Use this button to save your keys to floppy or hard disk for backup purposes or to exchange them with others.

Because your own keys are comprised of a public and a private key component, exporting your own key involves a two stage process. During the first stage the public key component is exported - a typical dialog to export SecEx public keys is shown below.



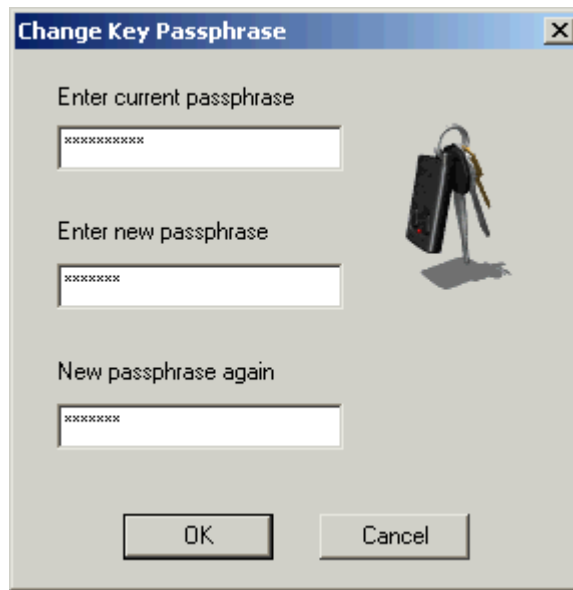
You will then be prompted to decide if you wish to export the private key component also. See image below.



This is the second stage. If you are backing up your key to floppy or similar such medium, you will probably want to export your private key component also since you will not be able to fully restore your key later without the private key component. If you are sending your key to a friend to enable that person to send encrypted email to you, you will only need to export the public key component, **NOT** the private key component. Private keys are stored in 3DES encoded, chained block cipher format and protected with a passphrase. See [SecExMail Key File Format](#).

#### Change Key Passphrase Button :

Use this button to change the passphrase on the selected key. See image below.



 **Delete Key Button :**

Use this button to remove a selected key from the list.

 **Import Key Button :**

Use this button to import new keys or restore backups from floppy or hard disk.

 **Email Key to Friend Button :**

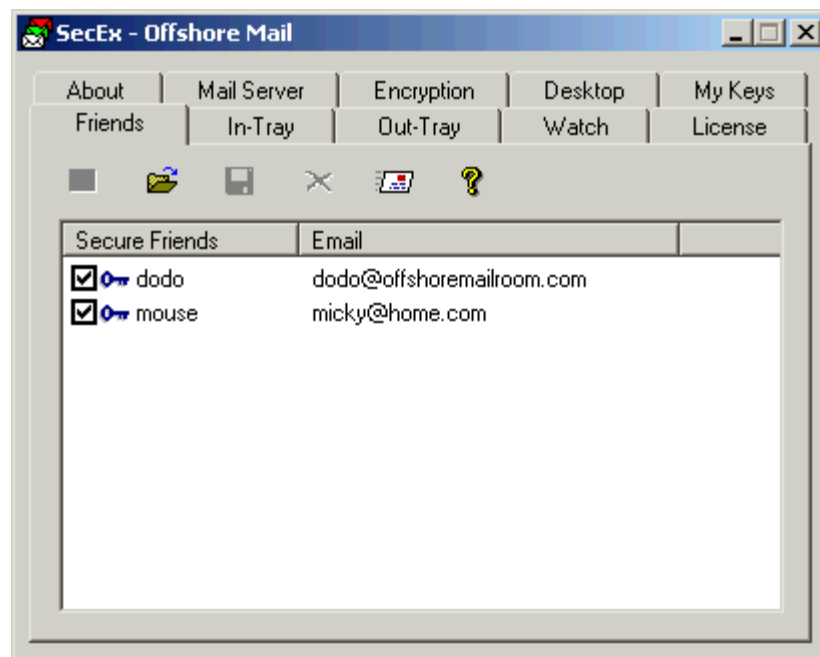
Use this button to email the public key component of your SecExMail key to others. This will enable the receiver to send encrypted mail to you. See [Distributing SecExMail Keys](#).

 **Clear Passphrase Cache Button :**

Use this button to clear a passphrase for the associated key from the registry. See [passphrase cache](#) for details.

## 2.6 Friends tab

The Friends tab lists people on your secure contact list. Email sent via SecExMail to people on the Friends list will be encrypted automatically and without the need for further interaction by you, the user. On this screen you can add and remove friends from your secure contact list, display key properties including fingerprints, and email the keys of friends to other people. Technically, a secure friend is someone for whom only a public key is held on file, i.e. you can encrypt messages to this person, but you cannot decrypt messages sent to this person. See [SecExMail Keys](#) for details.



☒ **Key Activation State :**

Use this check box to enable / disable encryption to the person associated with the respective key. If unchecked, messages sent to the key owner via SecExMail will be transmitted in the clear. If checked, messages sent to the key owner via SecExMail will be encrypted.

 **Key Properties Button :**

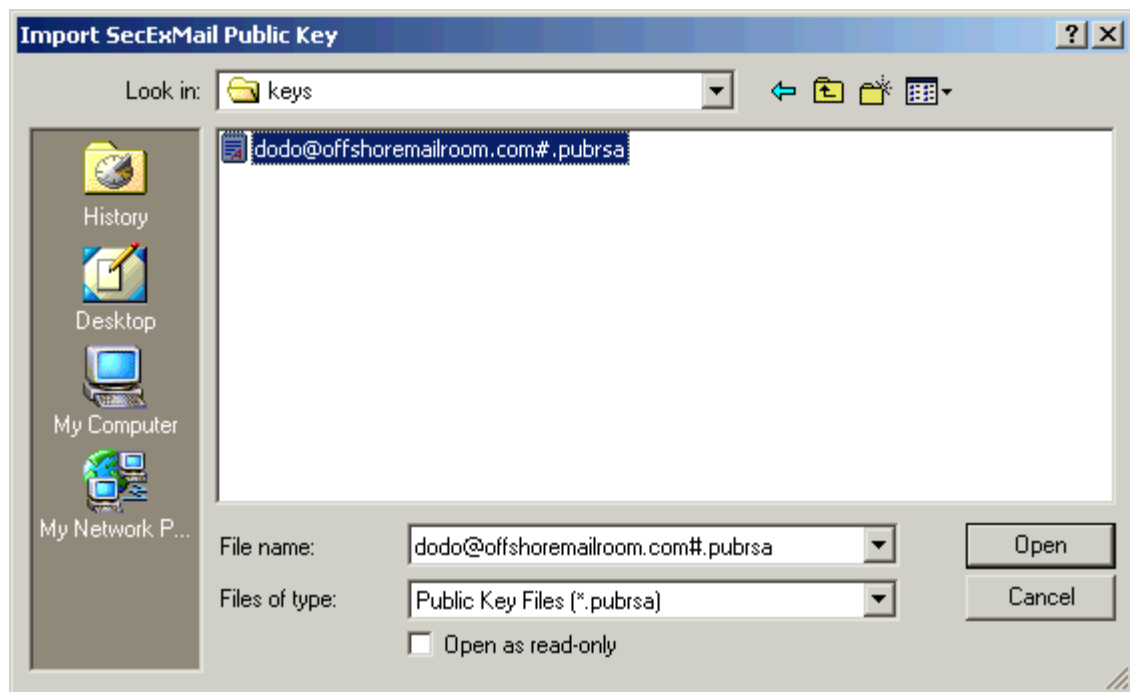
Use this button to display the properties of a selected key. The key properties include details of the key owner, the email address associated with the key, the key type and size as well as the key fingerprint. The properties for a typical key are shown below.



#### Import Key Button :

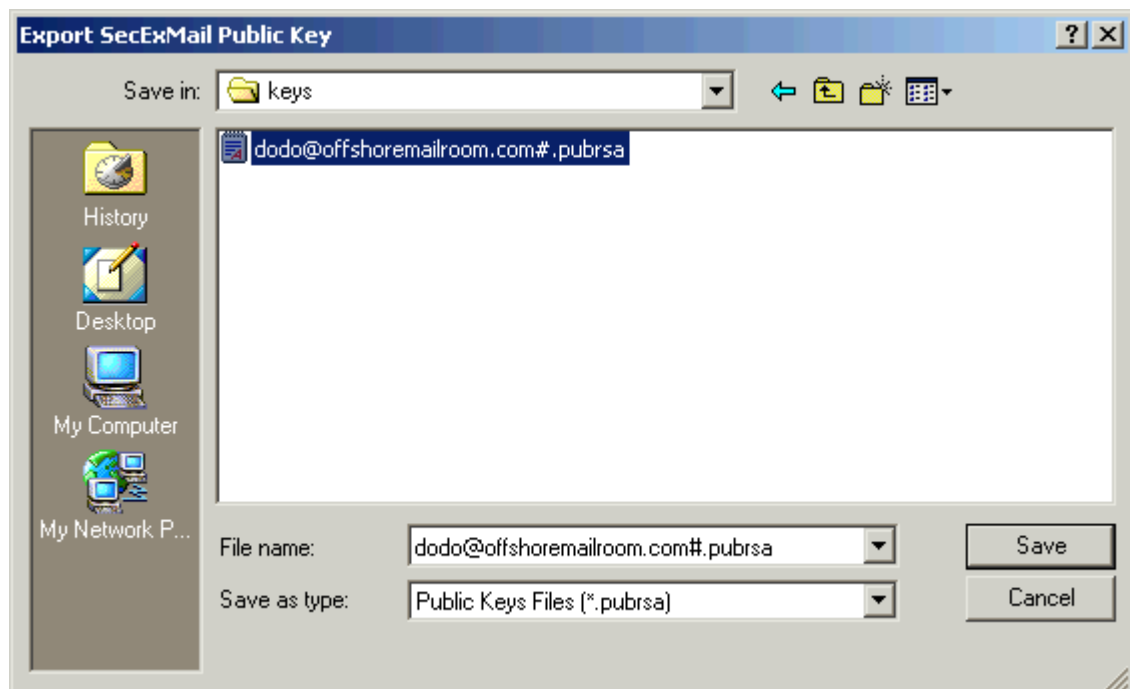
Use this button to import new keys from floppy or hard disk. Friend keys stored on disk must end "#.pubrsa".

A typical dialog to import SecEx public keys is shown below.

**Export Key Button :**

Use this button to save friend keys to floppy or hard disk for backup purposes or to exchange them with others.

A typical dialog to export SecEx public keys is shown below.

**Delete Key Button :**

Use this button to remove a friend from the secure contact list.

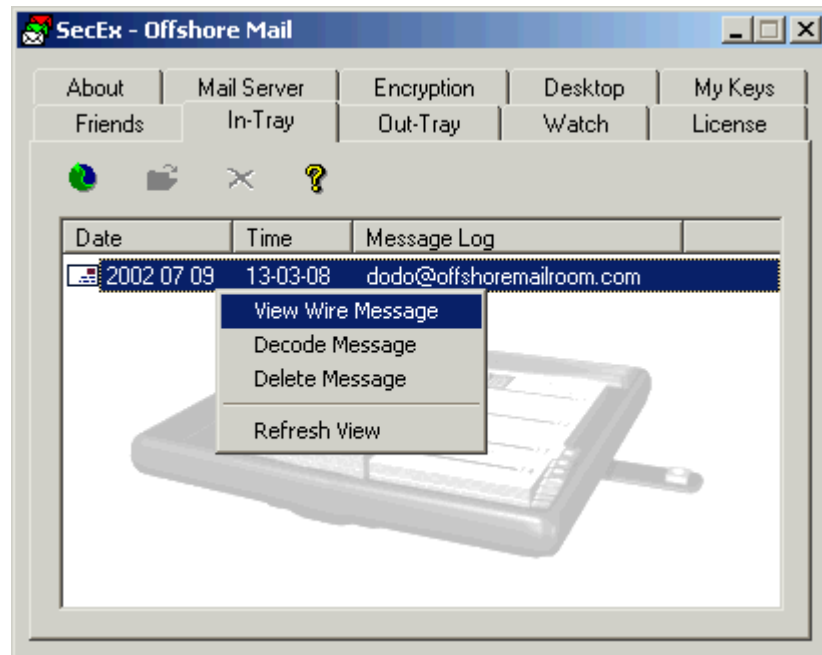


#### **Email Key to Friend Button :**

Use this button to email a friend's key to others. See [Distributing SecExMail Keys](#).

## 2.7 In-Tray Tab

The **In-Tray** tab shows incoming messages that SecExMail has processed. It is similar to your email client's inbox. Here you can view messages as they "came in off the wire" and decrypt messages which were encrypted to keys for which you hold the private key component.



#### **Refresh View Button :**

Use this button to refresh the list on the **In-Tray** tab. This list is updated when it is first displayed and thereafter only when you click the refresh view button.



#### **Open Message Button :**

Use this button to open a selected message. This feature will allow you to view "raw wire" message in notepad.

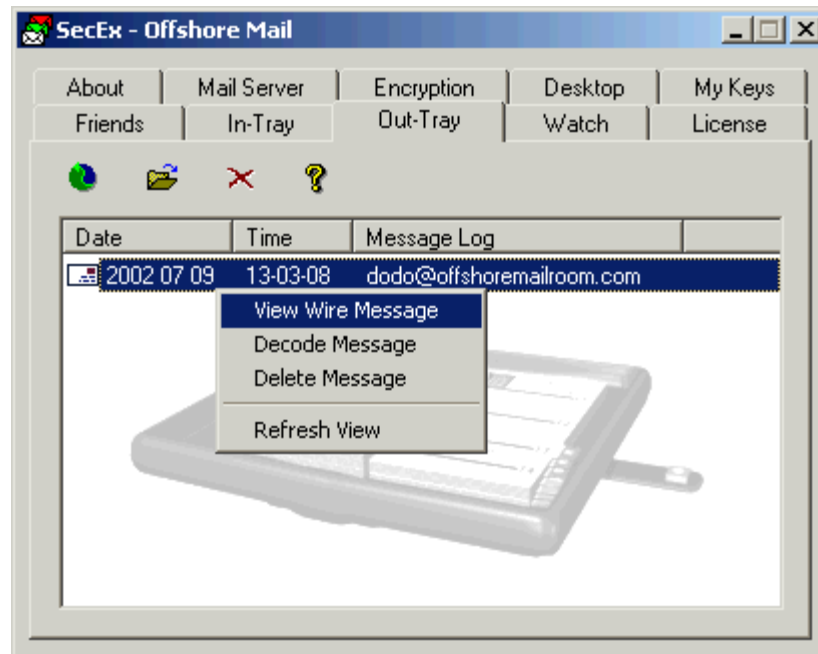


#### **Delete Message Button :**

Use this button to delete selected message(s).

## 2.8 Out-Tray Tab

The **Out-Tray** tab shows outgoing and sent messages that SecExMail has processed. It is similar to your email client's sent items folder. Here you can view messages exactly as they have been send and decrypt messages which were encrypted to keys for which you hold the private key component. Note that you will not be able to decrypt messages which have been encrypted to your friends unless you have carbon copied ( "CC" ) them to yourself.



### Refresh View Button :

Use this button to refresh the list on the **Out-Tray** tab. This list is updated when it is first displayed and thereafter only when you click the refresh view button.

### Open Message Button :

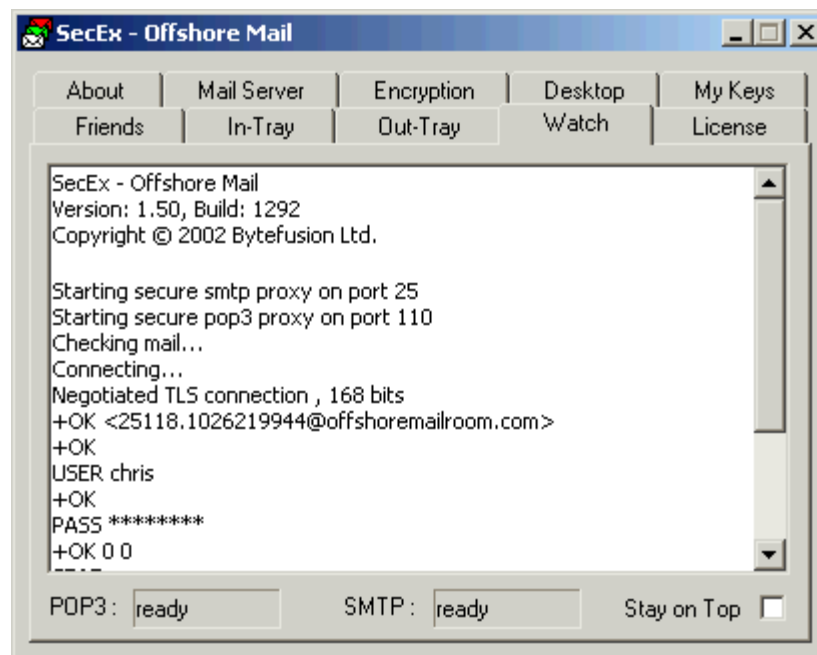
Use this button to open a selected message. This feature will allow you to view "raw wire" message in notepad.

### Delete Message Button :

Use this button to delete selected message(s).

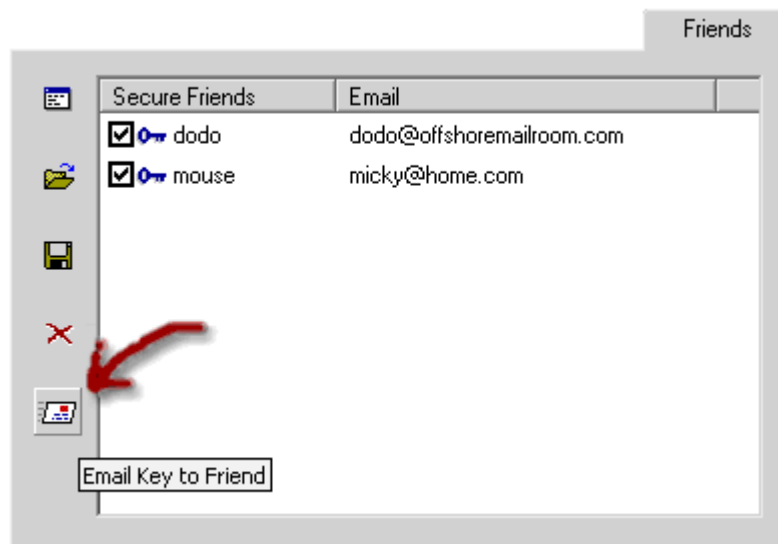
## 2.9 Watch tab

The **Watch tab** shows SecExMail log information in real-time as you send and receive email. A full record of all log entries can be found in the SecExMail application directory under "**secexmail.log**".

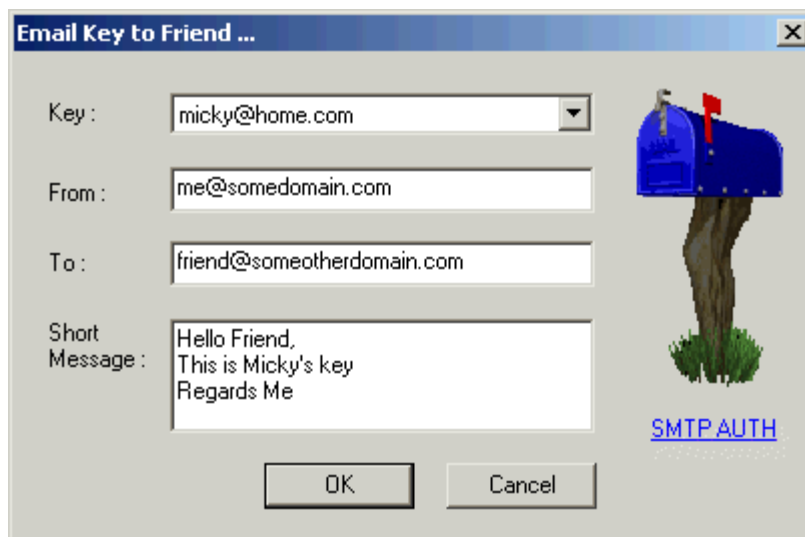


## 2.10 Distributing SecExKeys

Sending friends your public key "manually" is easy. See also [Automatic Key Exchange](#). Simply click the "Email Key to Friend" button on the **Friends** tab. See image.



Finally, select the key you wish to send from the list of keys in the "**Key**" drop-down box, enter your own email address in the "**From**" field, enter the recipient's email address in the "**To**" field, and optionally a short personal message in the "**Short Message**" field. Click "**OK**". The display will switch to the **Watch** tab for the sending of the key via your email server.



If your send mail server requires you to authenticate in order send mail, see [SMTP AUTH](#).

## 2.11 Send Mail Authentication

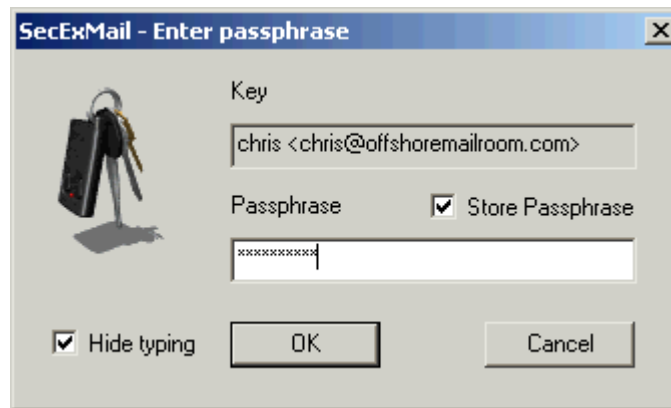
If your send mail server requires you to authenticate in order send mail click the **SMTP AUTH** link on the [Email Key to Friend](#) dialog. This will invoke the dialog shown below. Enter your user name and password as issued by your internet service provider and click Ok.



The SecExMail [Email Key to Friend](#) facility uses the Extended Simple Mail Transfer Protocol (ESMTP) AUTH LOGIN command to support user authentication.

## 2.12 Passphrase cache

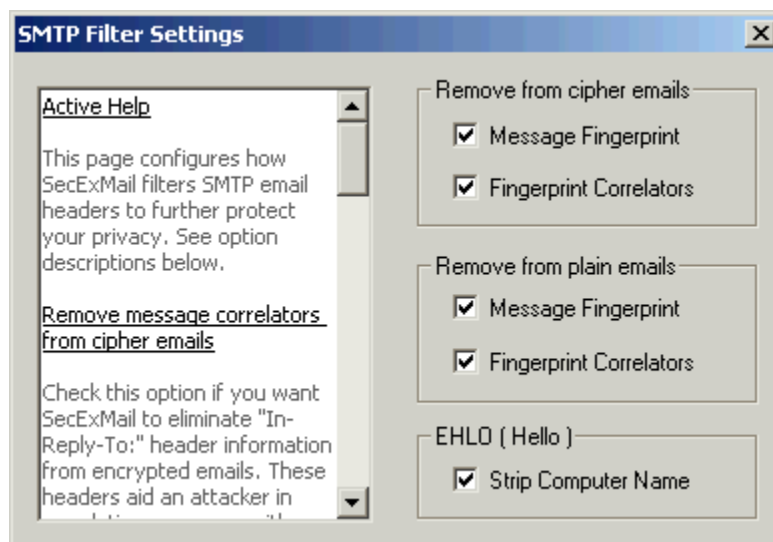
When SecExMail starts you will be prompted to enter the passphrase for any private keys loaded from the registry or disk. This passphrase is required to decode private keys which are stored in 3DES encrypted format.



Optionally, you can instruct SecExMail to cache the passphrase for your key in the registry by checking the "Store Passphrase" option. This will prompt SecExMail to write the passphrase for your key to your computer's registry in Blowfish encrypted format. Nonetheless, this option is only recommended if you can safeguard access to your computer and its registry. In particular you should disable the remote registry service (Windows NT / 2000 / XP). You can clear the passphrase cache for your key on the [My Keys](#) tab.

## 2.13 Outgoing Mail Filter

This page configures how SecExMail filters SMTP email headers to further protect your privacy.



Every email message bears a unique fingerprint or message identifier which may be used to track the message across a network or indeed across the global internet. Further, each email reply contains a fingerprint correlator that allows the reply to be uniquely associated with the original message. SecExMail allows you to remove both message fingerprints as well as fingerprint correlators. Note that this will also disable message threading functionality in email client software.

- **Remove message fingerprint from cipher emails**

Check this option if you want SecExMail to eliminate "Message-ID:" header information from encrypted emails. Modifying this setting will have immediate effect.

Shown below is an excerpt from a typical SMTP header containing a Message-ID fingerprint :

```
Return-Path: <yourname@offshoremailroom.com>
Delivered-To: john_doe@offshoremailroom.com
Received: from (HELO yourmachine_name) (yourname@203.134.3.41)
  by localhost with SMTP; 23 Jul 2002 09:52:57 -0000
From: "Your Name" <yourname@offshoremailroom.com>
To: <john_doe@offshoremailroom.com>
Subject: Testing
Date: Tue, 23 Jul 2001 10:55:35 +0100
Message-ID: <000221c27717$gax278f1$0800a8c0@yourname>
MIME-Version: 1.0
```

The same header would appear as follows with the **Remove message fingerprint** option checked :

```
Return-Path: <yourname@offshoremailroom.com>
Delivered-To: john_doe@offshoremailroom.com
Received: from (HELO yourmachine_name) (yourname@203.134.3.41)
  by localhost with SMTP; 23 Jul 2002 09:52:57 -0000
From: "Your Name" <yourname@offshoremailroom.com>
To: <john_doe@offshoremailroom.com>
Subject: Testing
Date: Tue, 23 Jul 2001 10:55:35 +0100
MIME-Version: 1.0
```

- **Remove fingerprint correlators from cipher emails**

Check this option if you want SecExMail to eliminate "In-Reply-To:" header information from encrypted emails. Modifying this setting will have immediate effect.

Shown below is an excerpt from a typical SMTP header containing a fingerprint correlator :

```
Return-Path: <yourname@offshoremailroom.com>
Delivered-To: john_doe@offshoremailroom.com
Received: from (HELO yourmachine_name) (yourname@203.134.3.41)
  by localhost with SMTP; 23 Jul 2002 09:52:57 -0000
From: "Your Name" <yourname@offshoremailroom.com>
To: <john_doe@offshoremailroom.com>
Subject: Testing
Date: Tue, 23 Jul 2001 10:55:35 +0100
Message-ID: <000221c27717$gax278f1$0800a8c0@yourname>
In-Reply-To: <000801c23789$3r040dc0$1441a8c0@nimitz>
MIME-Version: 1.0
```

The same header would appear as follows with the **Remove fingerprint correlators** option checked :

```
Return-Path: <yourname@offshoremailroom.com>
Delivered-To: john_doe@offshoremailroom.com
Received: from (HELO yourmachine_name) (yourname@203.134.3.41)
  by localhost with SMTP; 23 Jul 2002 09:52:57 -0000
```

```
From: "Your Name" <yourname@offshoremailroom.com>
To: <john_doe@offshoremailroom.com>
Subject: Testing
Date: Tue, 23 Jul 2001 10:55:35 +0100
Message-ID: <000221c27717$gax278f1$0800a8c0@yourname>
MIME-Version: 1.0
```

- **Remove message fingerprint from plain emails**

Check this option if you want SecExMail to eliminate "Message-ID:" header information from clear text emails. Modifying this setting will have immediate effect.

- **Remove fingerprint correlators from plain emails**

Check this option if you want SecExMail to eliminate "In-Reply-To:" header information from clear text emails. Modifying this setting will have immediate effect.

- **Erase Computer Name from EHLO command**

Email clients initiate the SMTP handshake with the EHLO command or "Hello" command. Usually the EHLO command will be used to identify your computer to the mail server. If you operate on a private network and behind a firewall the EHLO command will disclose your computer's true identity even if the firewall is used to mask your private IP address. This can be useful information to a hacker in locating sensitive information on a corporate network should firewall security ever be compromised. Check "Erase Computer Name" if you want SecExMail to erase your computer's identity from the EHLO command. Modifying this setting will have immediate effect.

Shown below is an excerpt from a typical SMTP header containing EHLO computer identity information :

```
Return-Path: <yourname@offshoremailroom.com>
Delivered-To: john_doe@offshoremailroom.com
Received: from (HELO yourmachine_name) (yourname@203.134.3.41)
    by localhost with SMTP; 23 Jul 2002 09:52:57 -0000
Date: Tue, 23 Jul 2001 10:55:35 +0100
```

The same header would appear as follows with the **Erase Computer Name** option checked :

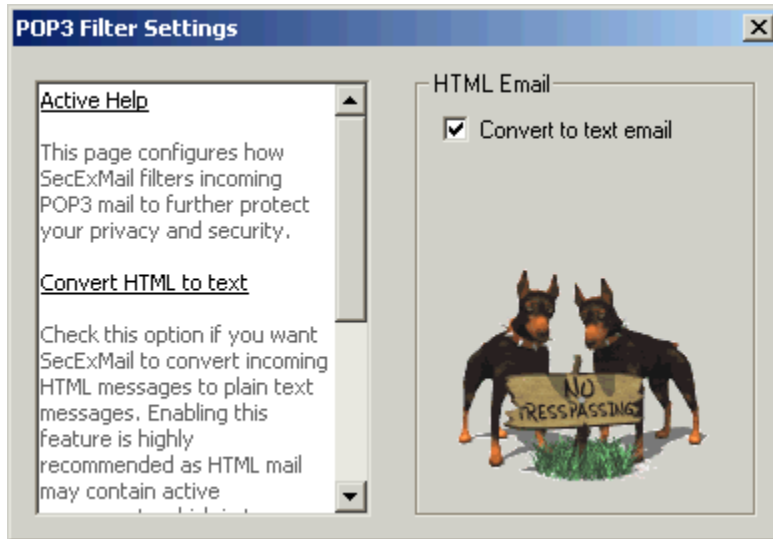
```
Return-Path: <yourname@offshoremailroom.com>
Delivered-To: john_doe@offshoremailroom.com
Received: from (HELO ) (yourname@203.134.3.41)
    by localhost with SMTP; 23 Jul 2002 09:52:57 -0000
Date: Tue, 23 Jul 2001 10:55:35 +0100
```

The exact message layout depends on your mail server and email client software. The SMTP protocol uses the HELO command. EHLO is used by the Extended Simple Mail Protocol ( ESMTP ).

## 2.14 Incoming Mail Filter

SecExMail supports a number of privacy filter options. Incoming HTML messages may be converted from their native HTML format to plain text. The HTML filter is not enabled by default.

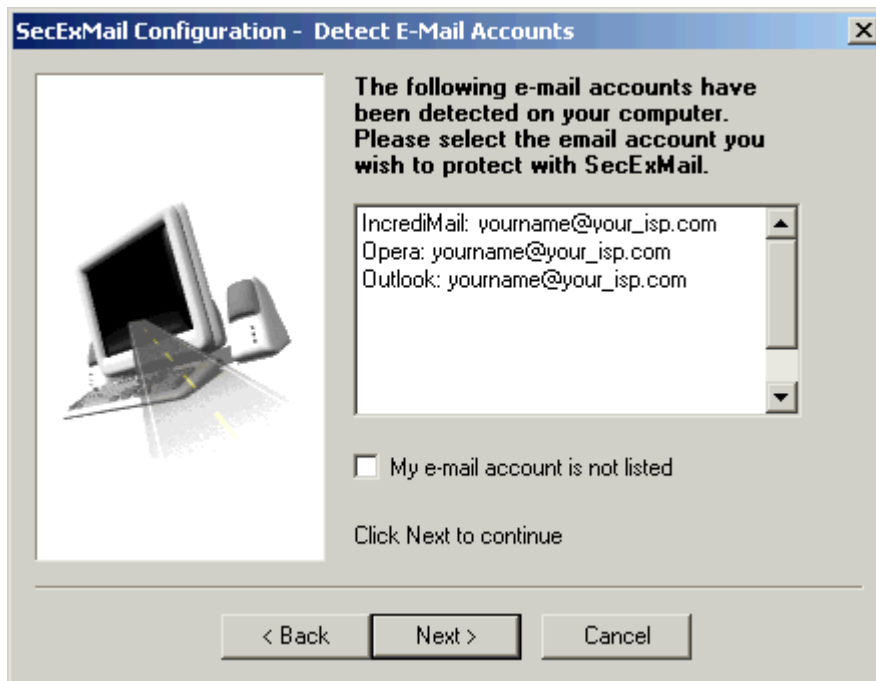
Check this option if you want SecExMail to convert incoming HTML messages to plain text messages. Enabling this filter is highly recommended as HTML mail may contain active components which in turn may be used to distribute viruses or stage malicious attacks on your computer. HTML email is especially dangerous, because malicious exploits may execute as soon as the email is being viewed and without your knowledge. Modifying this setting will have immediate effect.



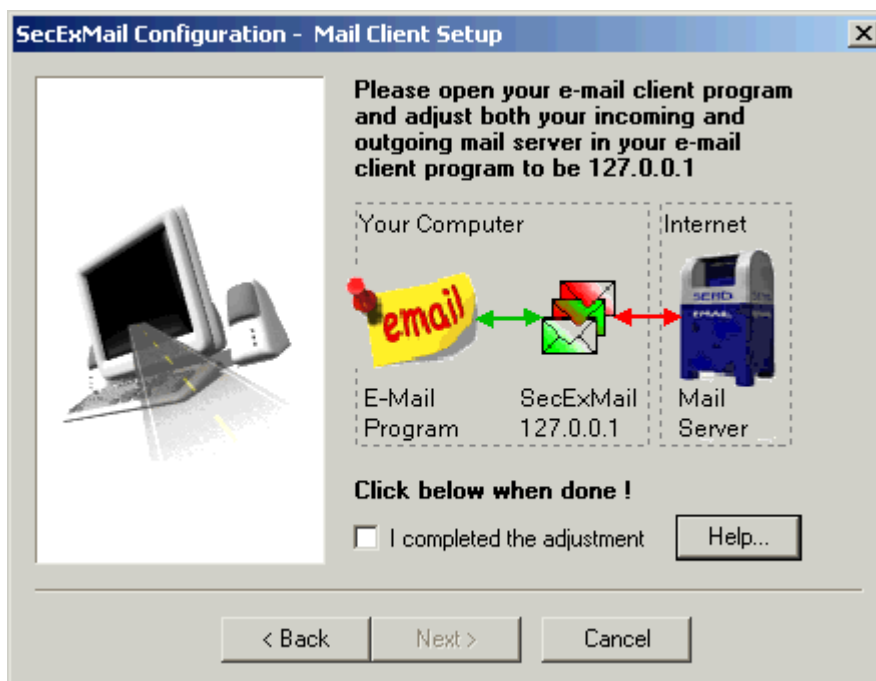
Enabling this filter will not impact your ability to receive HTML attachments.

## 2.15 Configuration Wizard

When you start SecExMail for the first time you will have the option to configure SecExMail using the configuration wizard. The configuration wizard will attempt to detect existing e-mail accounts and import the settings into SecExMail as well as adjust your e-mail client program to work with SecExMail. At present, automatic configuration is supported for the following e-mail client programs : Microsoft Outlook, IncrediMail, and Opera Mail.



All SMTP/POP3 compliant e-mail clients are SecExMail compatible. Some e-mail clients may require manual configuration.



SecExMail allows automatic discovery of encryption keys, so you don't have to manually exchange encryption keys with other SecExMail users. Keys are processed automatically as tag lines in incoming and outgoing mail.

After secure communication between yourself and another user is established, key tag lines are omitted. Automatic key exchange is enabled by default.



SecExMail supports a number of privacy filter options. Incoming HTML messages may be converted from their native HTML format to plain text. This feature protects your computer against malicious HTML containing exploits based on active message content which may be embedded in HTML code. The HTML filter is not enabled by default. Enabling this filter will not impact your ability to receive HTML attachments.



Microsoft Outlook, IncrediMail, and Opera Mail are registered trademarks or trademarks of their respective owners.

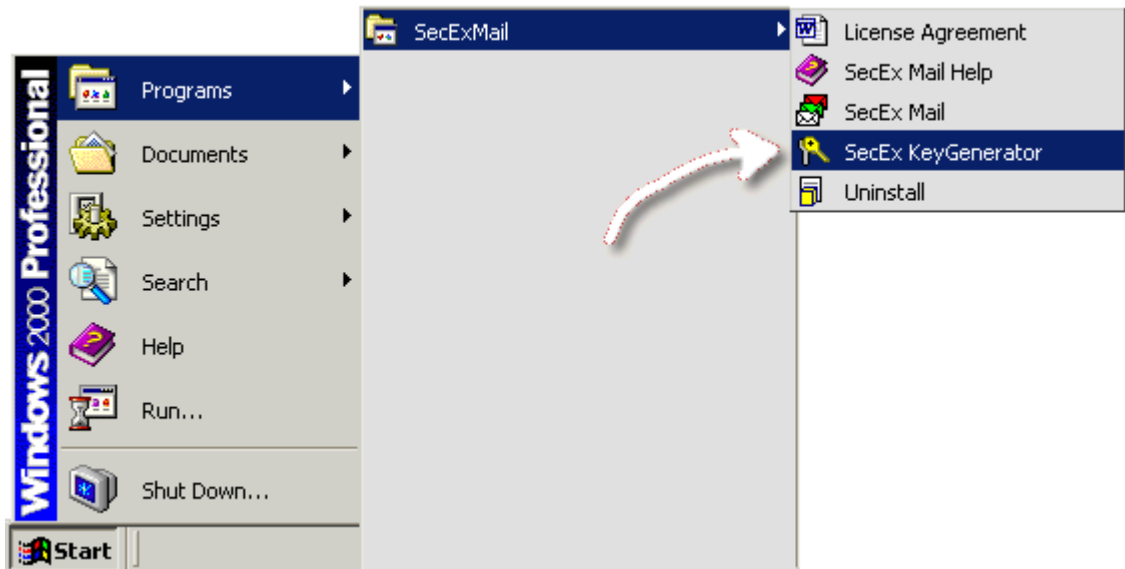
## 2.16 Automatic Key Exchange

If the "Enable auto keys" option is checked on the [Desktop Tab](#), SecExMail will discover the encryption keys of other SecExMail users automatically - without any user intervention. SecExMail will simply append your public key to outgoing e-mails as a tag line and accept such public key updates from other people. Once secure communication with another person is established, your public key is no longer appended to emails sent to that person.

## 3 Keys

### 3.1 Create your personal SecExMail keys

The SecEx Key Generator creates [encryption keys](#) for you which will enable your friends to send you encrypted mail and enable you to decrypt mail sent to you by your friends. To invoke the SecEx Key Generator, click "**Start**", "**Programs**", "**SecExMail**" and "**SecEx KeyGenerator**" as shown below.



This will start the SecEx Key Generator which will guide you through the process of creating your own SecExMail keys.



Click **Next** to proceed to the [Personal Details](#) screen.

## 3.2 Personal Details Screen

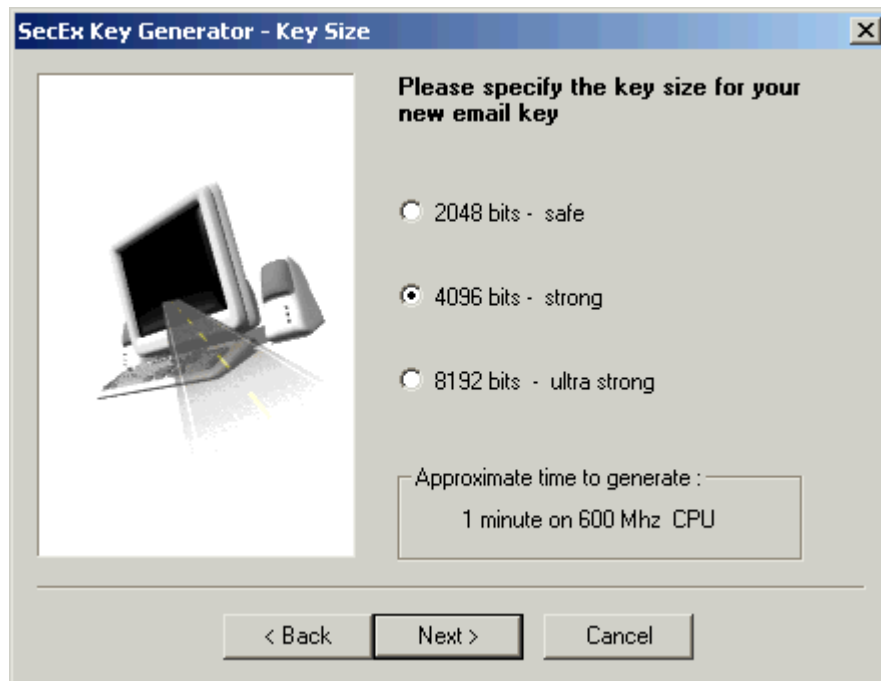
The **Personal Details** screen collects information about you and your email address. This information will later appear on the [My Keys tab](#) in SecExMail. SecEx Key Generator will not disclose your information to anyone and you control whom you share your key information with.

The screenshot shows a Windows-style dialog box titled "SecEx Key Generator - Personal Details". On the left is the same computer graphic as the previous screen. On the right, the text reads: "Please provide the name and email address that will be associated with this keypair." Below this are two input fields. The first is labeled "Your name" and contains the text "dodo bird". The second is labeled "Email address" and contains the text "dodo@offshoremailroom.com". At the bottom are three buttons: "< Back", "Next >", and "Cancel".

Click **Next** to go to the [Key Size](#) screen.

### 3.3 Key Size Screen

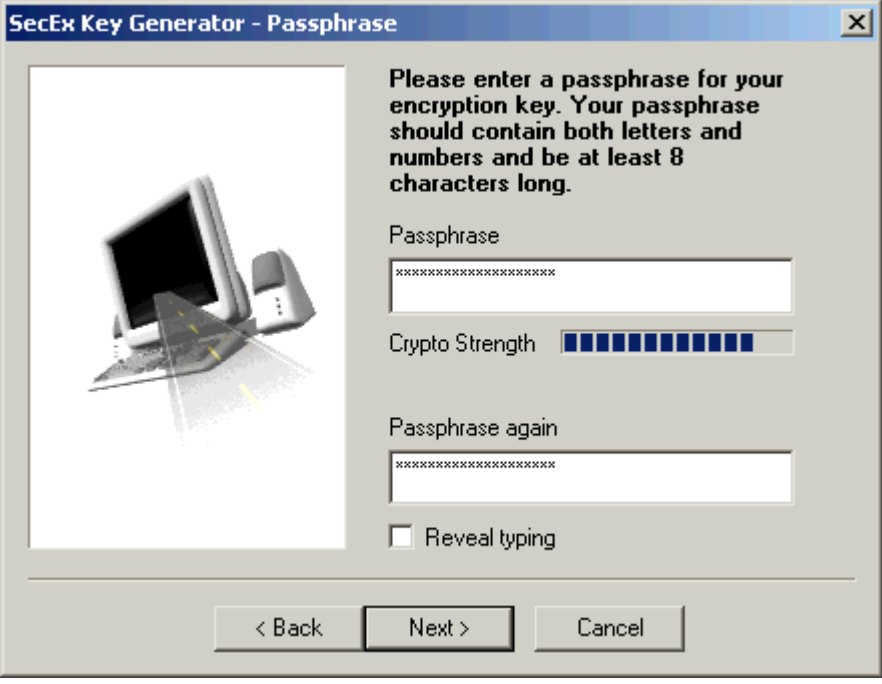
Deciding on the key size of your new email key is a matter of personal judgement. It is commonly held that RSA keys of 1024 bits will withstand conventional cryptanalytic attacks while key sizes of 512 bits or less are to be regarded as insecure. In general, the cryptographic community is divided over the recommended key size for [asymmetric keys](#) and what is referred to as the "huge key debate". If 1024 bit keys are secure, why use larger keys ? The counter argument is that the only disadvantage to using larger keys is the longer time required to process them. CPU cycles are cheap and getting cheaper every year. This aids the cryptographer and cryptanalyst alike. So why not use the largest key size contemporary computers can handle ? The decision is yours ...



Click **Next** to proceed to the [Passphrase](#) screen.

### 3.4 Passphrase Screen

Your new key will be stored in your computer's registry. To protect confidential key information from unauthorized access, it will be [encrypted and protected](#) with a passphrase that only you know. Please chose a long phrase containing both letters and numbers and avoid using the names of girlfriend, wife, boyfriend or husband. Do not use your date of birth.

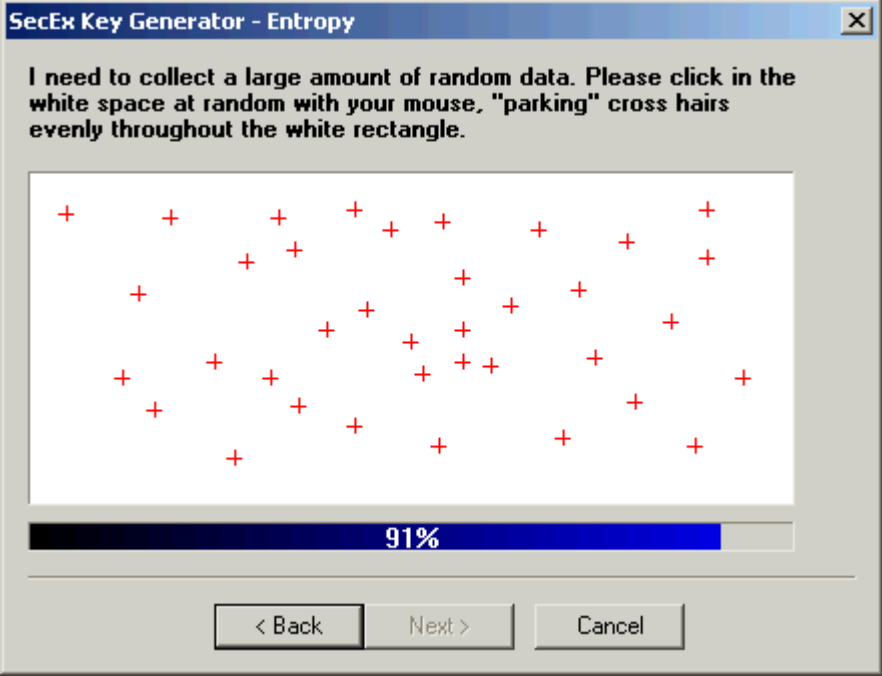


The screenshot shows a window titled "SecEx Key Generator - Passphrase". On the left is an image of a computer monitor and keyboard. The main text area contains the instruction: "Please enter a passphrase for your encryption key. Your passphrase should contain both letters and numbers and be at least 8 characters long." Below this are two text input fields, both containing "xxxxxxxxxxxx". Between the fields is a "Crypto Strength" indicator consisting of ten blue bars. Below the second field is a checkbox labeled "Reveal typing" which is currently unchecked. At the bottom are three buttons: "< Back", "Next >", and "Cancel".

Click **Next** to proceed to the [Entropy](#) screen.

### 3.5 Entropy Screen

Your new key will be generated from prime numbers produced by a random number generator. In order for your key to be unpredictable we need to collect a large amount of random data from the only source in the system which is unique : **you**. This data will be used to seed the random number generator. To ensure maximum security, the SecEx Key Generator does not avail itself of previously calculated prime numbers or so called "canned primes." All prime numbers are generated "on the spot".

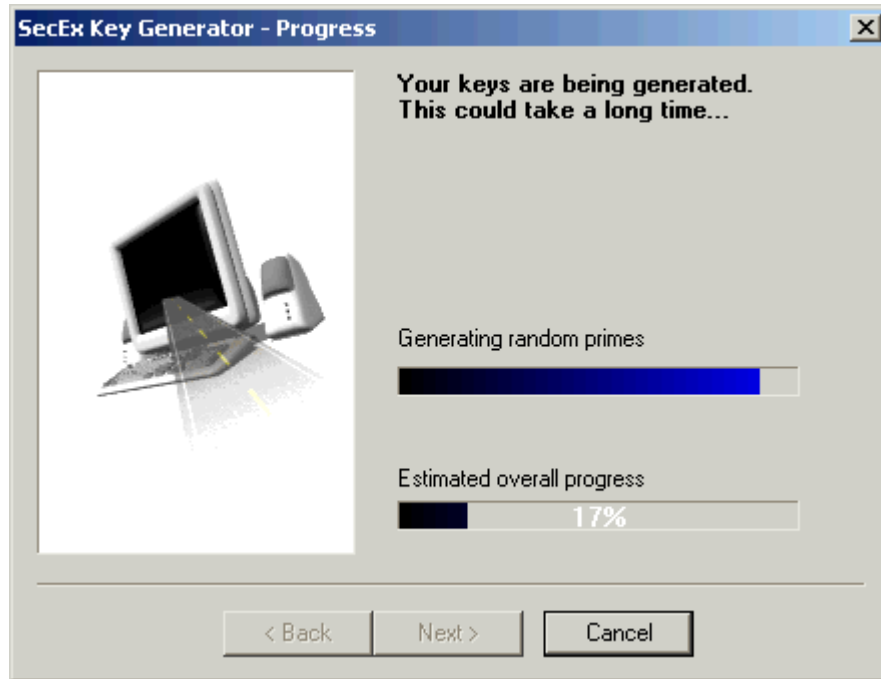


The screenshot shows a window titled "SecEx Key Generator - Entropy". The main text area contains the instruction: "I need to collect a large amount of random data. Please click in the white space at random with your mouse, 'parking' cross hairs evenly throughout the white rectangle." Below the text is a large white rectangle containing numerous red plus signs (+) scattered randomly. At the bottom of the rectangle is a progress bar that is mostly filled with blue, with the text "91%" displayed in the center. Below the progress bar are three buttons: "< Back", "Next >", and "Cancel".

Click **Next** to proceed to the [Progress](#) screen.

## 3.6 Progress Screen

The progress screen shows the estimated time to completion. The estimated time to finish may be readjusted during key generation and you will be advised when key generation is complete. At that time, click **Finish** to save your keys and restart SecExMail if necessary.



## 4 Email Clients

### 4.1 Basic Email Client Configuration

Configuring your email client to send and receive email via SecExMail is easy. Simply set your email client to receive mail at "**localhost**" or IP address "**127.0.0.1**" on port **2110** and send email via "**localhost**" or IP address "**127.0.0.1**" on port **2025**. The port settings may vary and should match the decryption and encryption port settings on the [Encryption tab](#). Detailed guides for popular email clients are provided in this section.



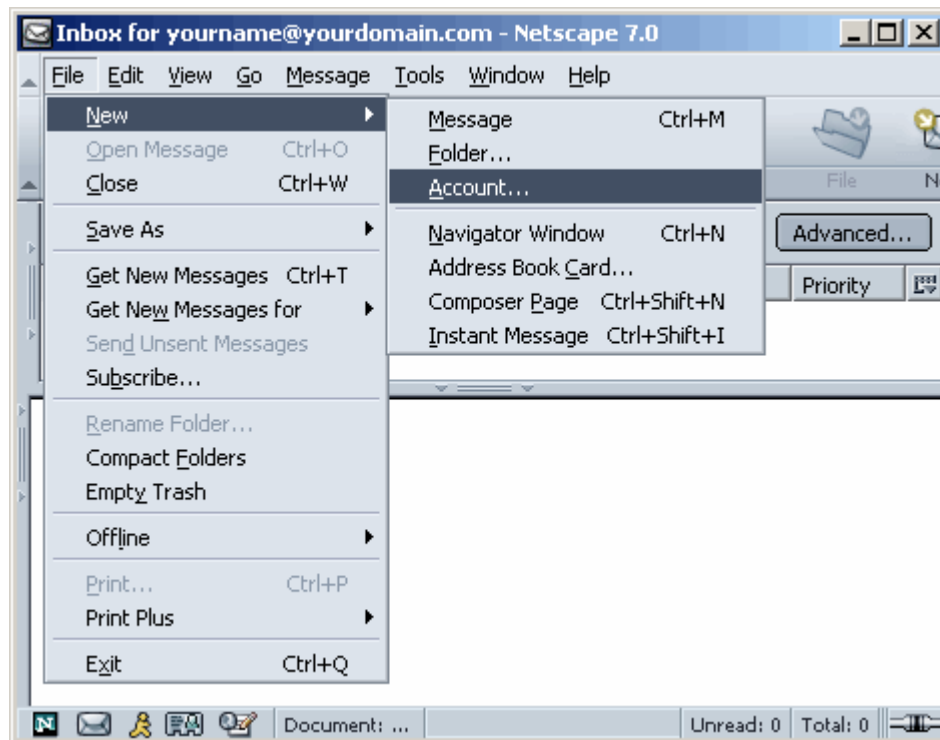
## 4.2 Netscape Mail

To configure Netscape Mail to work with SecExMail, you will need the password given to you by your internet service provider (ISP) or your system administrator. SecExMail operates as a go-between or relay agent between Netscape Mail and your ISP's mail server. It encrypts and decrypts messages to and from people on your [Friends](#) list so Netscape Mail must be configured to send and receive mail via SecExMail. Follow the steps detailed below to configure a new email account in Netscape Mail for use with SecExMail. These instructions apply to Netscape 7.0

If you are modifying an existing Netscape Email account for use with SecExMail, please refer to [modifying Netscape email accounts](#).

- **Step 1**

Open Netscape Mail and click on **File > New > Account** with your mouse.



- **Step 2**

A pop-up menu will appear. Select **Email account** and click on **Next**. This will invoke the Netscape Mail account wizard.



- **Step 3**

Enter your name as you wish it to appear on your emails and enter your e-mail address, click **Next** to continue.



**Account Wizard**

**Identity**

Each account can have its own identity, which is the information that identifies you to others when they receive your messages.

Enter the name you would like to appear in the "From" field of your outgoing messages (for example, "John Smith").

Your Name:

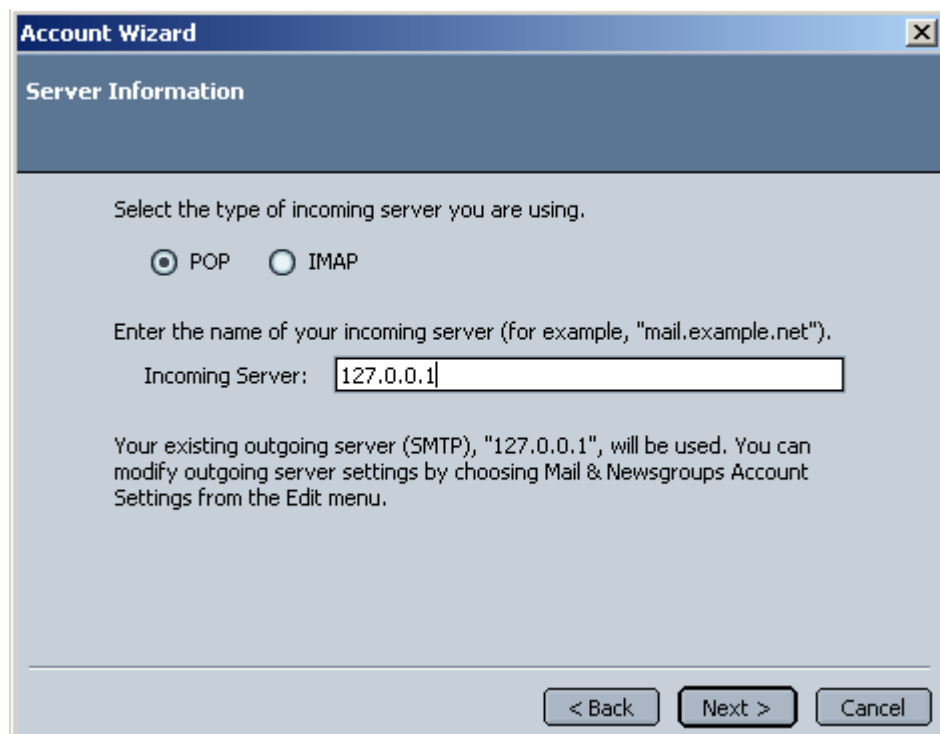
Enter your email address. This is the address others will use to send email to you (for example, "user@example.net").

Email Address:

< Back   Next >   Cancel

- **Step 4**

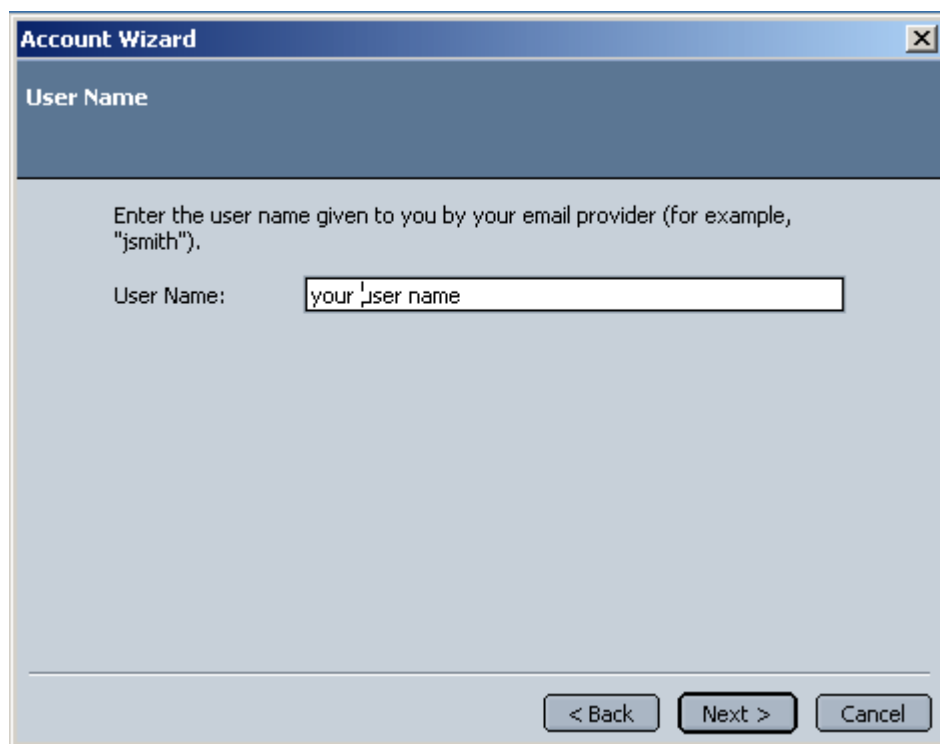
Select **POP** as the incoming mail server type. Enter the IP address 127.0.0.1 as the incoming mail server. This IP address is the loopback address for all computers running the Internet Protocol and is located on your computer. SecExMail is listening on this IP address and will process all incoming e-mail messages. Do not enter your internet service provider's (ISP) mail server details here - see "[Configuring your mail server](#)". You will need to specify port 2110 for POP3 and port 2025 for SMTP under advanced settings. The Windows edition of SecExMail uses proxy ports 110 and 25, but Linux prohibits non root processes from occupying reserved ports. See [modifying Netscape email accounts](#). Click **Next** to continue.



The screenshot shows a window titled "Account Wizard" with a close button in the top right corner. Below the title bar is a header section labeled "Server Information". The main area contains the text "Select the type of incoming server you are using." followed by two radio buttons: "POP" (which is selected) and "IMAP". Below this is the text "Enter the name of your incoming server (for example, 'mail.example.net')." followed by a text input field labeled "Incoming Server:" containing the text "127.0.0.1". A paragraph of text follows: "Your existing outgoing server (SMTP), '127.0.0.1', will be used. You can modify outgoing server settings by choosing Mail & Newsgroups Account Settings from the Edit menu." At the bottom right are three buttons: "< Back", "Next >", and "Cancel".

- **Step 5**

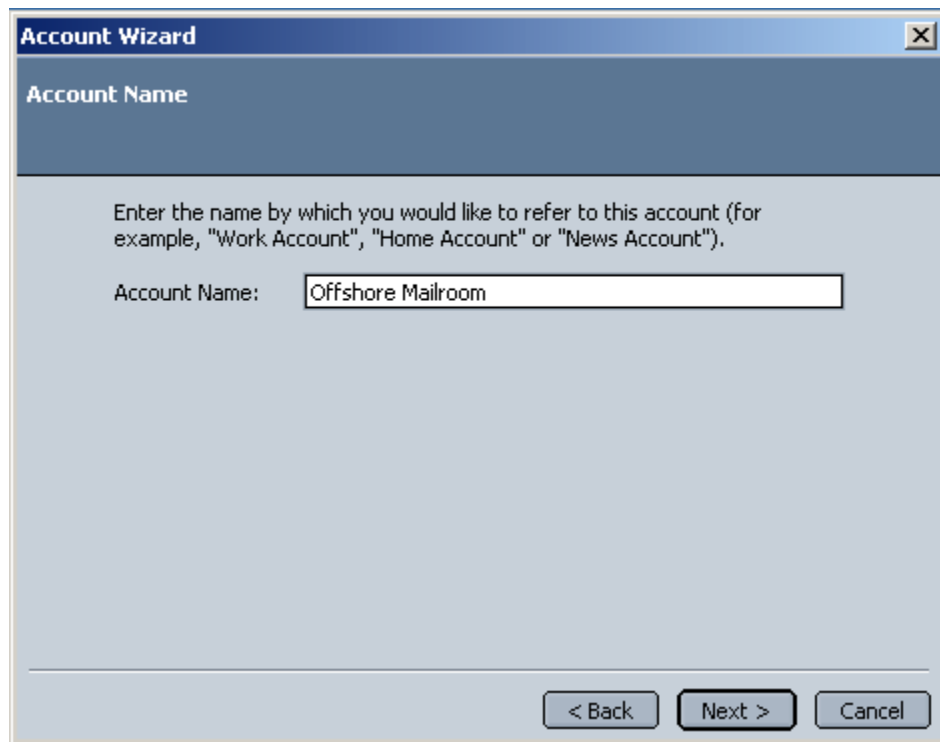
Enter your user ID as given to you by your ISP or system administrator. Click **Next** to continue



The screenshot shows a window titled "Account Wizard" with a close button in the top right corner. Below the title bar is a header section labeled "User Name". The main area contains the text "Enter the user name given to you by your email provider (for example, 'jsmith')." followed by a text input field labeled "User Name:" containing the text "your user name". At the bottom right are three buttons: "< Back", "Next >", and "Cancel".

- **Step 6**

Personalise your account with a name of your choice and click **Next** to finish



**Account Wizard**

**Account Name**

Enter the name by which you would like to refer to this account (for example, "Work Account", "Home Account" or "News Account").

Account Name:

< Back   Next >   Cancel

- **Step 7**

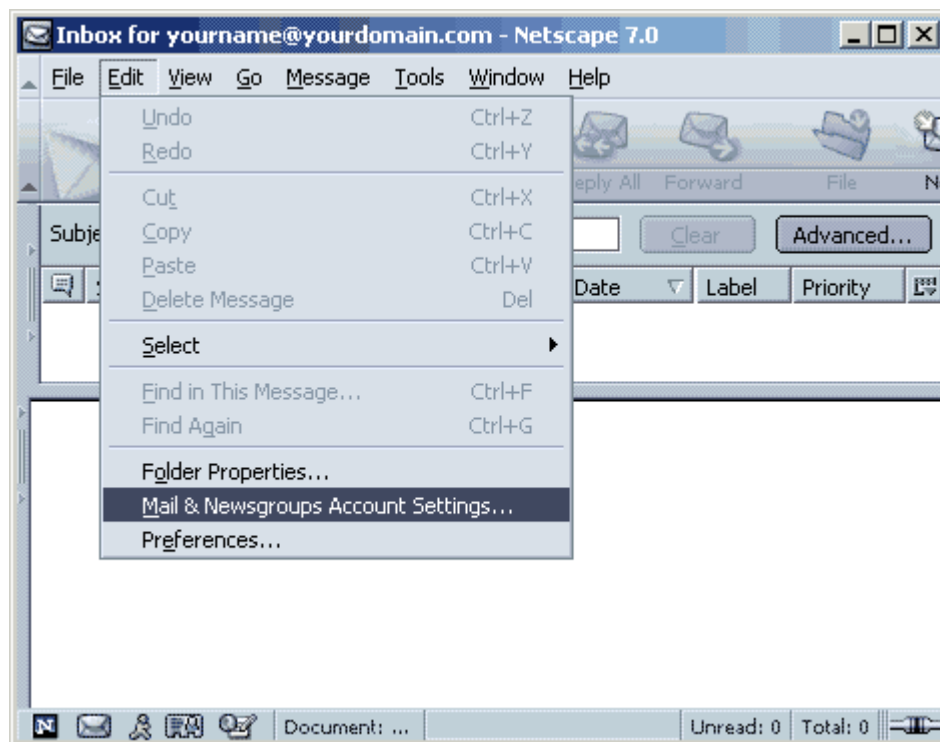
Finally, enter your internet service provider's (ISP) mail server details on the **Mail Server** tab - see "[Configuring your mail server](#)".

## 4.3 Modifying Netscape Mail

To modify an existing Netscape mail account for use with SecExMail, please follow the steps detailed below.

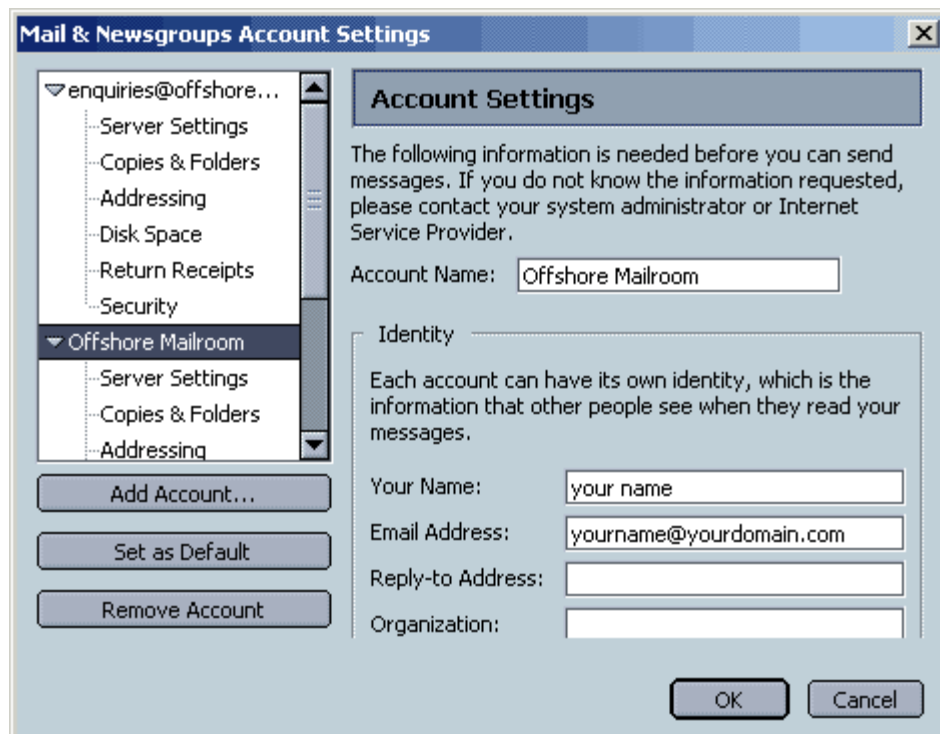
- **Step 1**

Open Netscape Mail and click on ***Edit > Mail & Newsgroups Account Settings***.



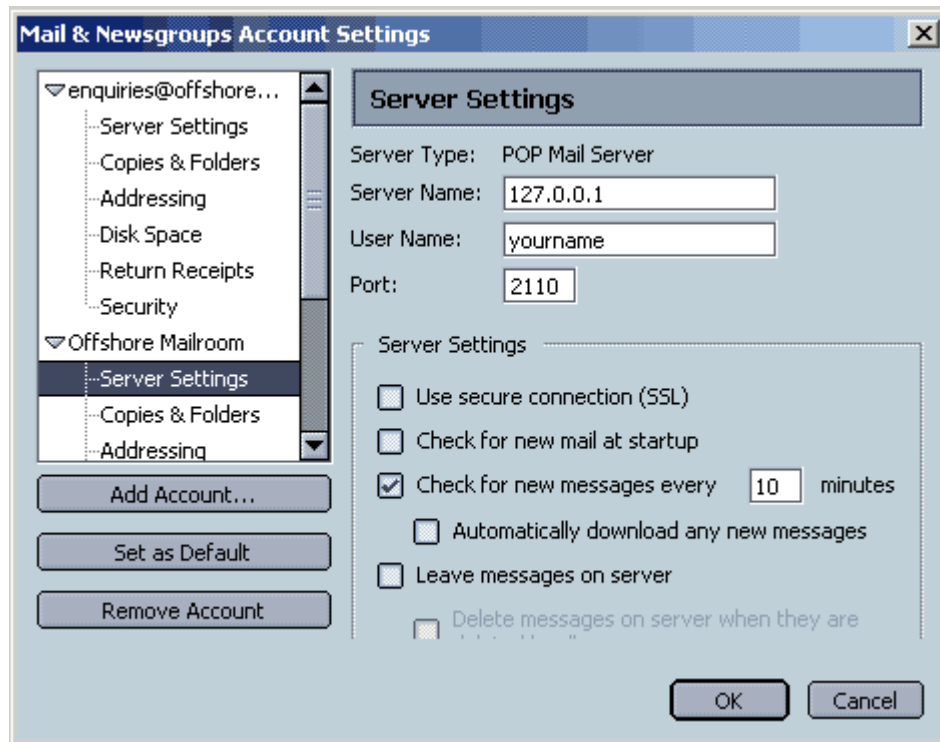
- **Step 2**

A pop-up menu will appear. Select the account you wish to modify.



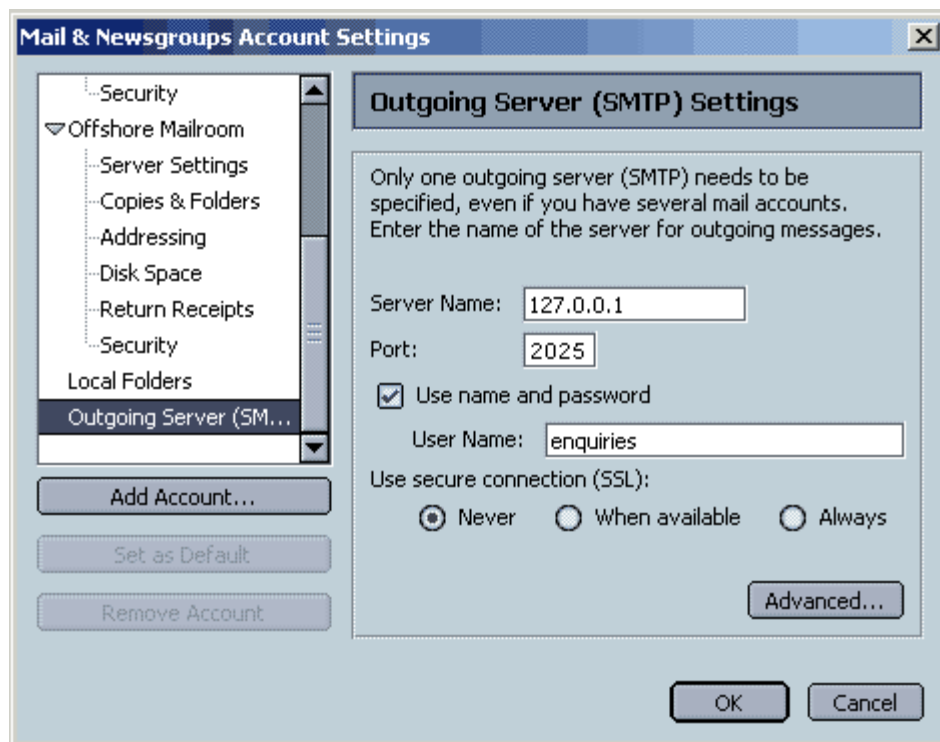
- **Step 3**

Select **Server Settings** and in the server name box enter 127.0.0.1. You will need to specify port 2110 for POP3 and port 2025 for SMTP under advanced settings. The Windows edition of SecExMail uses proxy ports 110 and 25, but Linux prohibits non root processes from occupying reserved ports.



- **Step 4**

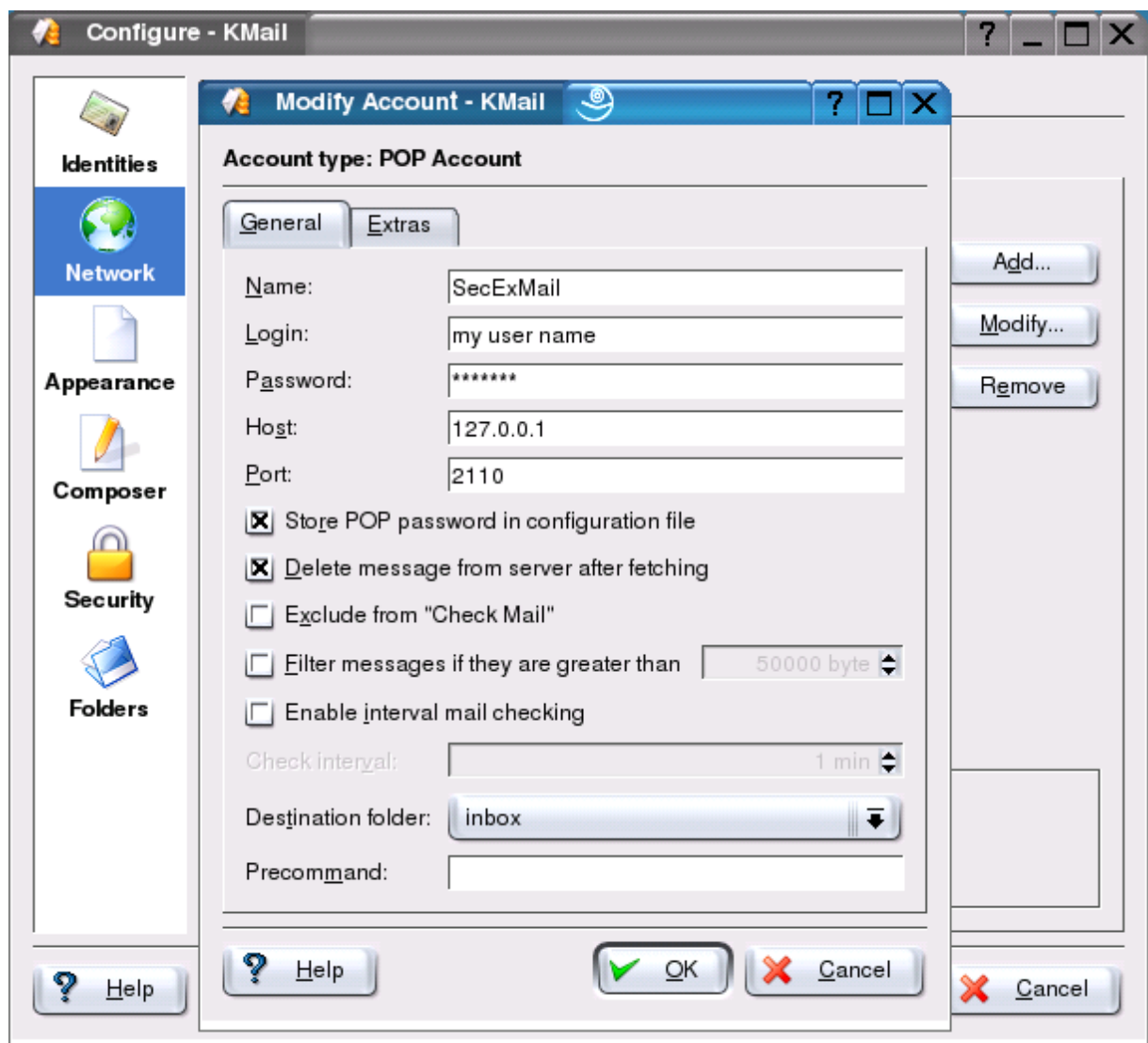
Select **Outgoing Server (SMTP)**. In the Server Name box enter 127.0.0.1. You will need to specify port 2110 for POP3 and port 2025 for SMTP under advanced settings. The Windows edition of SecExMail uses proxy ports 110 and 25, but Linux prohibits non root processes from occupying reserved ports. Click **OK** to finish.

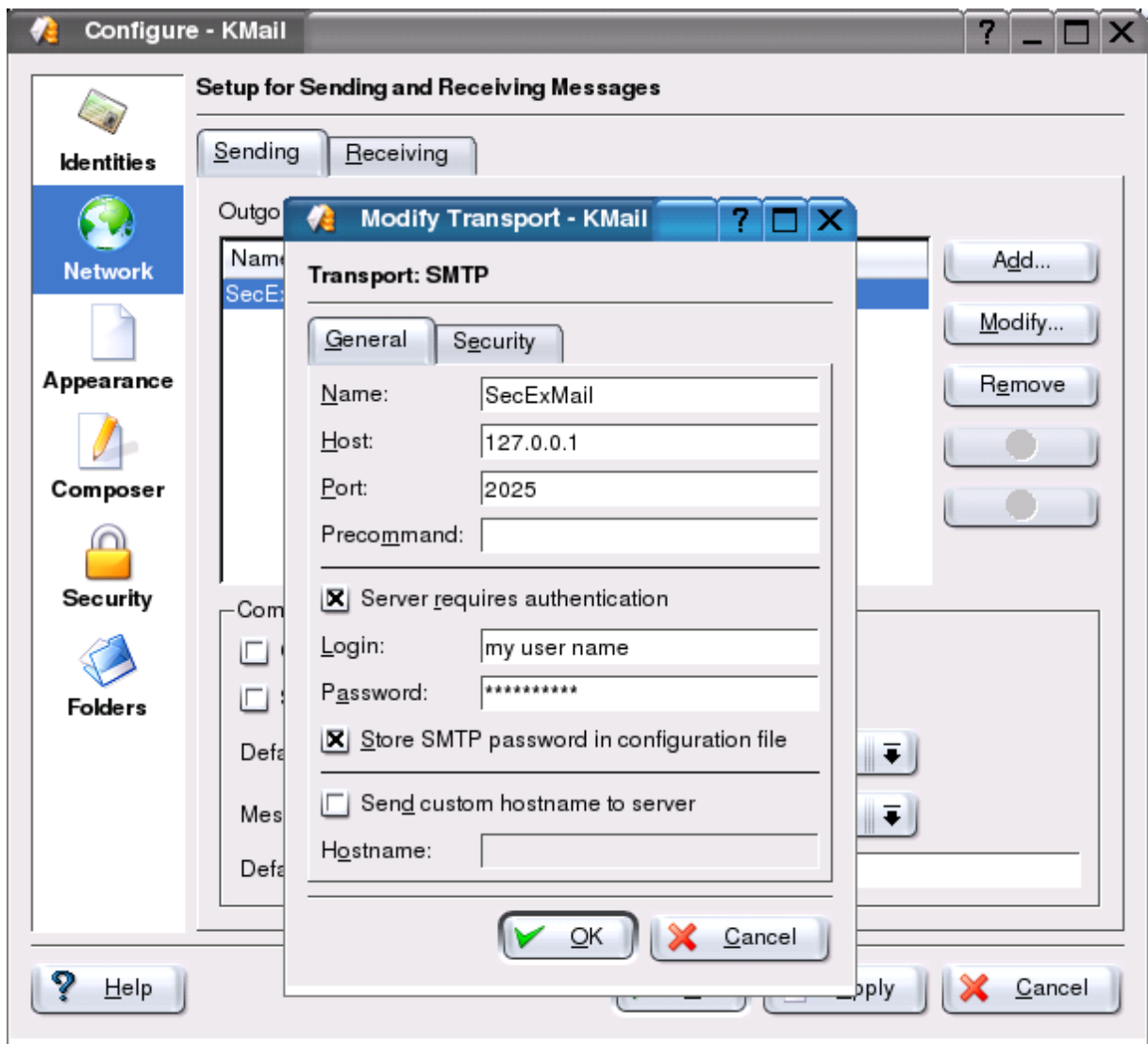


## 4.4 KMail

To configure KMail to work with SecExMail, you will need the password given to you by your internet service provider (ISP) or your system administrator. SecExMail operates as a go-between or relay agent between KMail and your ISP's mail server. It encrypts and decrypts messages to and from people on your [Friends](#) list, so KMail must be configured to send and receive mail via SecExMail.

Firstly, enter your mail server information on the SecExMail [Mail Server tab](#). Then configure KMail to use 127.0.0.1, port 2110, for incoming mail and 127.0.0.1, port 2025, for outgoing mail as shown below.

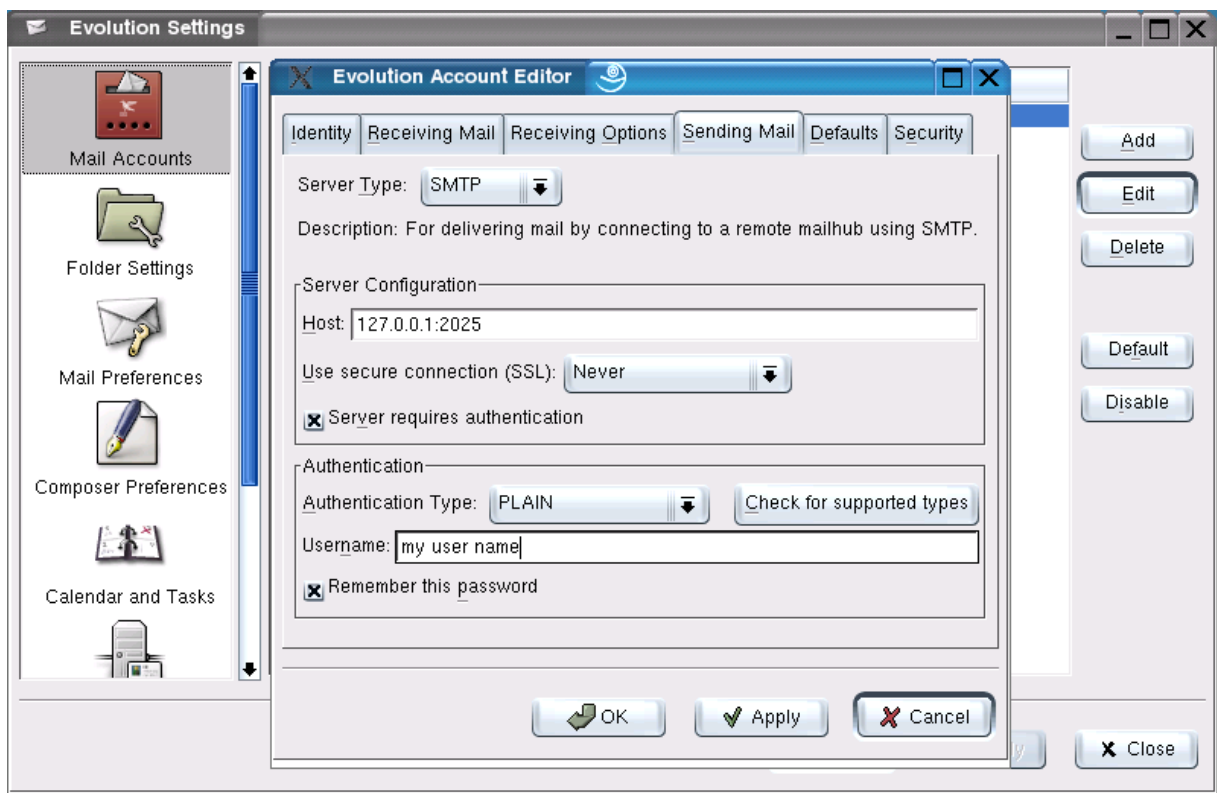
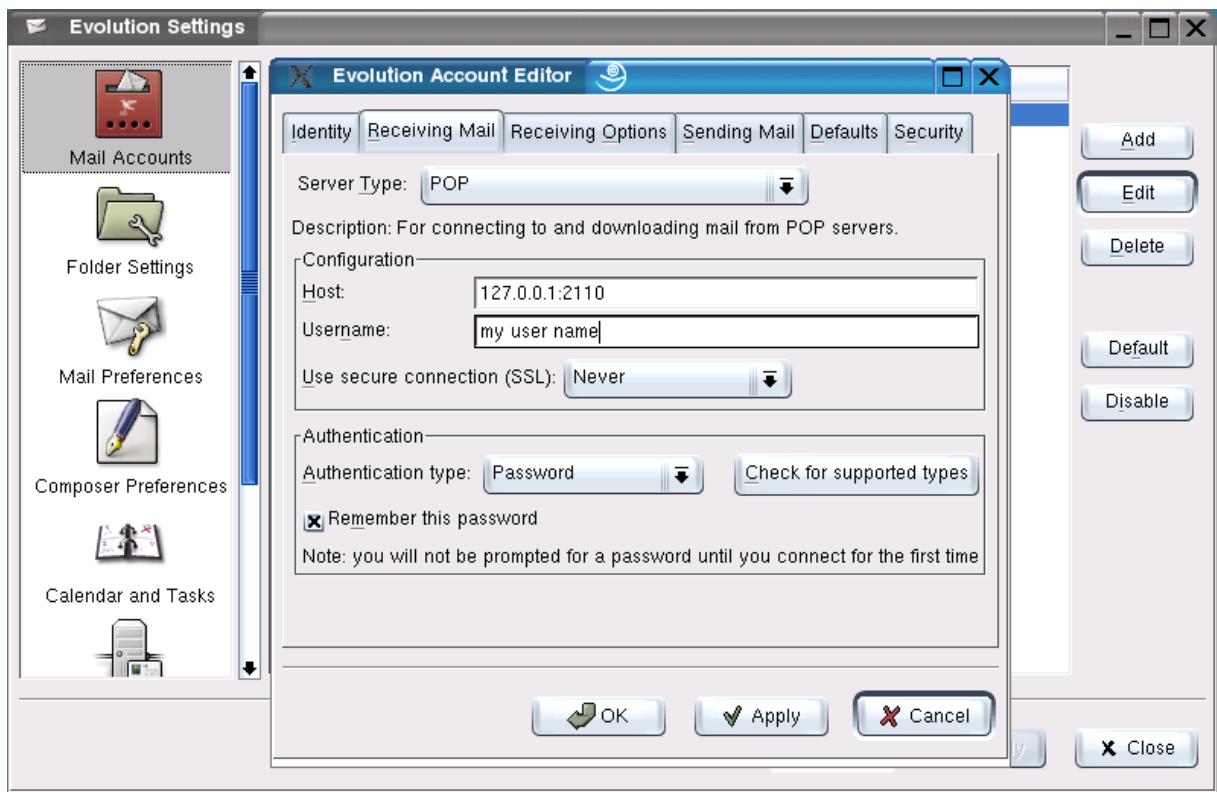




## 4.5 Ximian Evolution

To configure Ximian Evolution to work with SecExMail, you will need the password given to you by your internet service provider (ISP) or your system administrator. SecExMail operates as a go-between or relay agent between Ximian Evolution and your ISP's mail server. It encrypts and decrypts messages to and from people on your [Friends](#) list, so Ximian Evolution must be configured to send and receive mail via SecExMail.

Firstly, enter your mail server information on the SecExMail [Mail Server tab](#). Then configure Ximian Evolution to use 127.0.0.1, port 2110, for incoming mail and 127.0.0.1, port 2025, for outgoing mail as shown below.

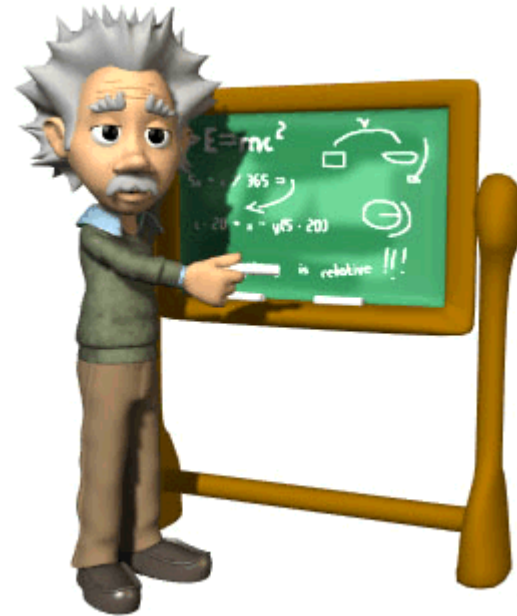


## 5 Technical

### 5.1 RSA Public Key Encryption

" $c = me \bmod n$ " is the algorithm that turns the world of e-commerce. Introduced in 1978 by Rivest, Shamir and Adleman after whom the cipher is named, RSA is the worlds foremost public key encryption system. Contrary to the design of classic encryption algorithms where the same key is used to lock and unlock the information, public key encryption relies on "two key" algorithms. The sender encrypts the message with the recipients public key who, upon receipt of the message, is able to decipher the same with the private key counterpart. This development was revolutionary in the field of cryptography because parties wishing to establish secure communications no longer had to meet in "secret" to exchange confidential keying information.

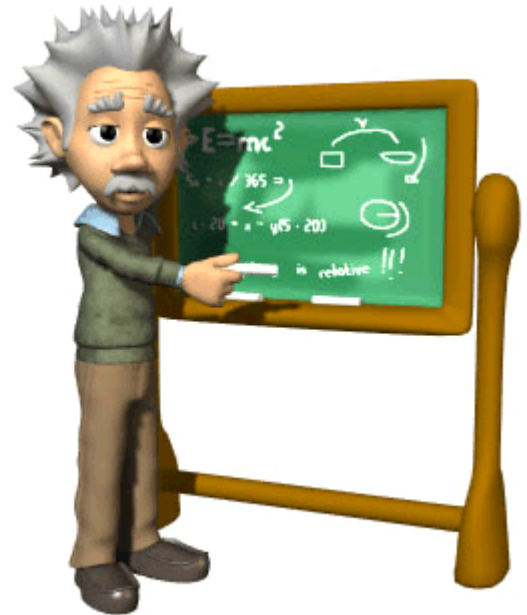
The [SecExMail public key](#) infrastructure uses industry standard RSA encryption as developed by the OpenSSL project. See [Acknowledgements](#).



## 5.2 ISAAC Random Number Generator

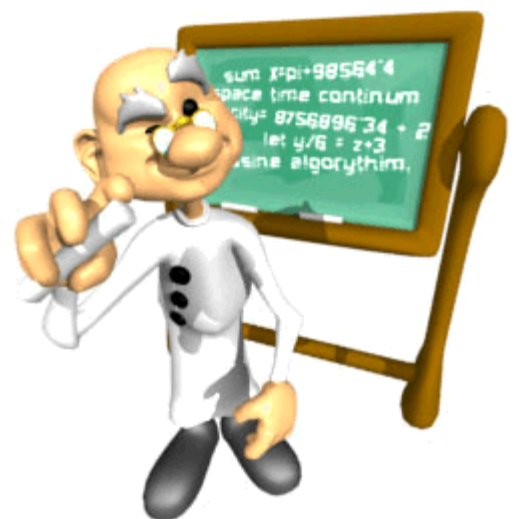
ISAAC (Indirection, Shift, Accumulate, Add, and Count) is a cryptographically secure pseudo random number generator. With an average cycle length of 2 to the 8295th power its output is uniformly distributed and unpredictable. ISAAC has been developed by Bob Jenkins and placed into the public domain in 1996. See [Acknowledgements](#) for legal information on ISAAC.

ISAAC is at the heart of SecExMail's entropy collection system and comprises the stream cipher subsystem of the SecExMail cipher.

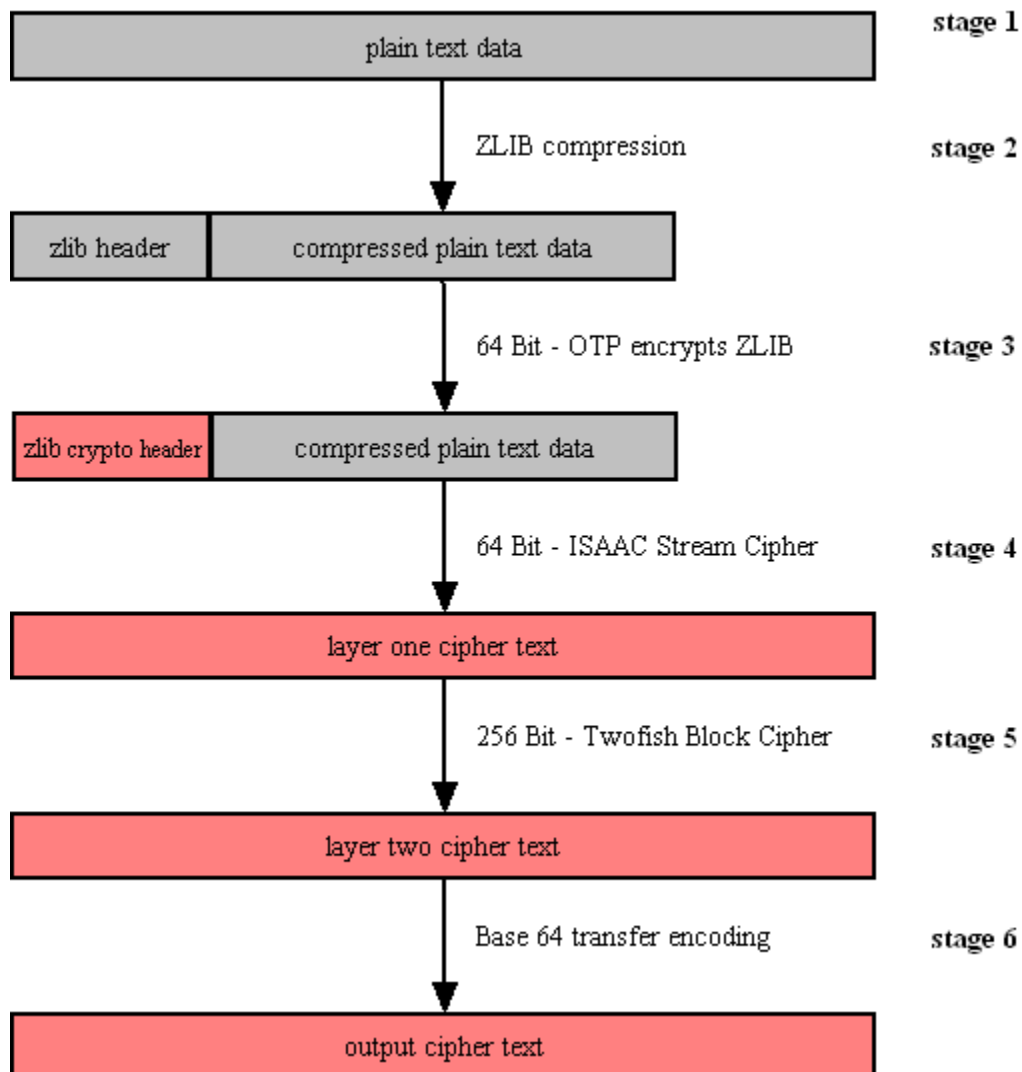


## 5.3 The SecExMail Cipher

The SecExMail cipher is a composite cipher specifically designed to operate on real-time email streams. It uses cryptographic primitives which are available to the general public and have been subject to extensive peer review. The SecExMail cipher incorporates RSA public key encryption. Message encryption is performed via the Twofish block cipher and the ISAAC stream cipher. The SecExMail cipher is warranted to be free from spy-ware, key escrow or key recovery features of any kind. The email encryption process is described in detail below. See diagram.



### SecExMail Composite Cipher



- **Stage 1**

Email data is received in variable length data blocks. SecExMail parses SMTP header info, mail and data bodies.

- **Stage 2**

Because email messages frequently contain known plain text, such as salutation and or tag lines, which gives rise to [known plain text attacks](#) on the encrypted message and in order to minimize overall message expansion, the plain text is first compressed using the ZLIB compression algorithm. The net effect of deflating large amounts of data, containing both tidbits of known plain text such as greeting or tag lines as well as unknown message text into a compressed data stream is that any known plain text is effectively obscured.

- **Stage 3**

The ZLIB stream has a fixed header format which in itself might be exploited as known plain text by a savvy cryptanalyst. For this reason, the first 64 bits of the stream are enciphered by way of a [One Time Pad](#), using standard XOR masking. This approach acknowledges that email messages will contain portions of known plain text and proactively manages this problem.

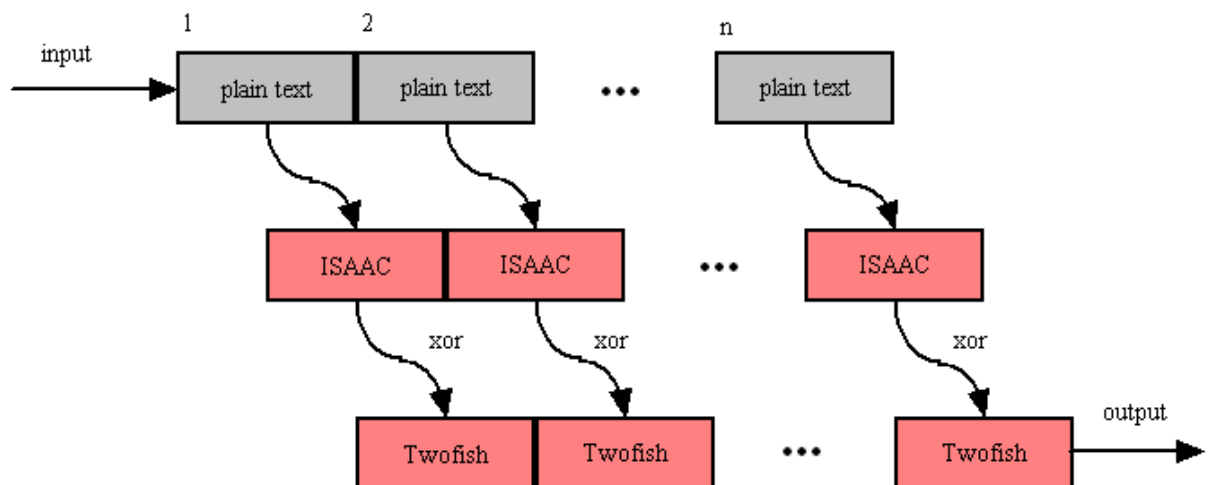
- **Stage 4**

At this point the compressed data is encoded using the 64 bit ISAAC stream cipher creating the layer one cipher text.

- **Stage 5**

The next step in the encryption process is to encrypt the layer one cipher text using the 256 bit Twofish block cipher. Twofish is used in chained block mode, but instead of XOR'ing the previous block's cipher text into the plain text of the current block, the output from the ISAAC layer is "chained in". This chaining process is illustrated below.

### ISAAC Twofish Block Chaining



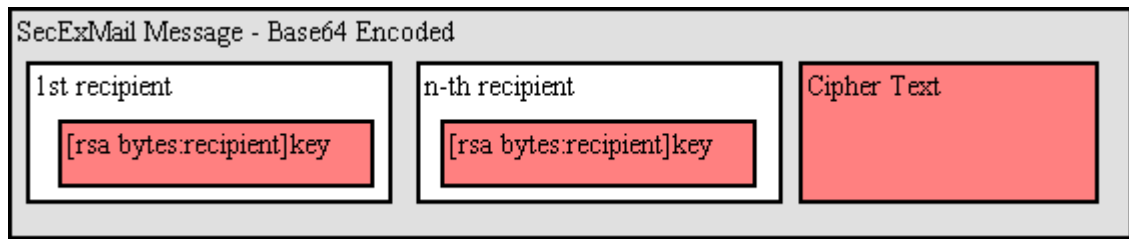
- **Stage 6**

The final step is to assemble the output in base64 transfer encoded format for transmission via mail transfer agents (MTA).

## 5.4 SecExMail Message Format

SecExMail messages are transferred in base64 encoded format. Messages may be encrypted to multiple recipients. The internal message layout is defined as follows :

**[<rsa bytes>:<recipient>]key[<rsa bytes>:<recipient>]key...cipher text**



- **RSA Bytes**

This is the size of the recipient's RSA key in bytes. Therefore a 2048 bit RSA key would be listed as having a size of 256 bytes. RSA This parameter is defined for RSA key sizes of 2048, 4096, and 8192 bits.

- **Recipient**

This is the email address of the recipient to whom the message is encoded.

- **Key**

This is the SecExMail session key material, encrypted with the [RSA public key](#) of the recipient. The SecExMail session key is used to encrypt the message body of the email message and is comprised of a 64 bit [One Time Pad](#) key, a 64 bit [ISAAC stream cipher](#) key, and a 256 bit Twofish key.

- **Cipher Text**

This is the message body encrypted with the [SecExMail Cipher](#).

A typical SecExMail enciphered message is depicted below :

```
--Begin SecEx 1.1--
WzI1NjpaHJpc0BvZmZzaG9yZWlhaWxyb29tLmNvbV0dJyyJnwwCm0LI0659zpBY/asERA3FRG9
9
OYRhm5f+rwohYORt8Wp3rmwI2Nguhk38KvH5pg8ZRTXXWiEHYMaKQPPXpbnaJepJFZeXTcNMTi/
d
p0Rc5HCTui5okW/00Gv8Sp328Ldh3DlGQcGW7oYt9qxG/cJ/PaVxxxEfDM3I4cnsCyLjfx+I0JY
6
h+emWt4U/N6u+K0tPL4ua2OfGhGoBXo+6KK042bXGpk/Pj6WEOQMCKyR+VrsOx6ZcTgpqS3WCcU
c
2/JDy9zHqlkPLohXcT4G2Hiwp/1JhviaQtoKA2NYYimuY5ZjNUGPMsIaN0h6AKS3/qZsHhK1Ltc
A
WpLnuoFbQleekuJngBCC1RIIILI4lfFgMkxoUkZrtXg6E217Q6GMMhHMANJ4EU3D2c1BgauDYAQ
G
Rpz0p8efm/WAZoXai6KVElMEiK7tv98s8wu9LpUxN44QYj2eNRVI+721GPfkBoKvr6eK5/TU4cH
N
Dg9VxCGj4n8KDvfYsPRpBSNzLL+Ta4iz7toQ/MGdPCQa
--End SecEx Mail--
```

## 5.5 SecExMail Key Transparency

SecExMail is engineered with a focus on transparency to give you the assurance that no backdoor keys or key recovery is embedded in encrypted messages. This means you as the recipient or sender of an encrypted SecExMail message can verify what keys have been used in the encryption of that message.

The non-technical approach to this is to right-click an encrypted message listed on the [in-tray tab](#) or [out-tray tab](#) with your mouse and select "**decode**" from the pop-up menu. Click the [watch tab](#) to review the analysis of the offline decryptor. The watch tab will report both whom the message was addressed to and whom the message was encrypted to. See sample output below :

```
Offline decryptor starting file analysis
This email was sent to:
dodo@offshoreemailroom.com -> have key
This message was encrypted to:
dodo@offshoreemailroom.com ...session key ok
Decrypted in 0.24 seconds
```

If the message was encrypted to multiple recipients, the log output will indicate this. Note that you will only be able to decrypt messages for which you hold private keys. This means that if you send email to people on your [friends](#) list, only your friends will be able to decrypt these messages. See [SecExMail Keys](#).

The more technical approach to verifying keys used in encryption of a particular messages is to use base64 decoding software and examine the raw data. See [SecExMail Message Format](#) for details.

## 5.6 SecExMail Keys

SecExMail employs public key encryption. Messages are encrypted to one or more recipients using their **public keys**. Only the intended recipient can, upon receipt of the message, recover the plain text using his/her **private key**. Public key encryption differs from classical encryption because the recipient of a message does not use the same key for decryption as the sender used for encryption.

In cryptography the fictional characters "Alice" and "Bob" are often used for illustration purposes. Consider the following scenario : Alice lives in New York and Bob lives in Los Angeles. Alice wants Bob to be able to send her confidential mail. She goes to her local hardware store and purchases a dozen or so combination padlocks, sets the unlocking code on each padlock, confuses the dials again, and sends the open padlocks to Bob in Los Angeles.



Bob is now in possession of Alice's padlocks, but not the unlocking codes. When Bob wants to send Alice a confidential letter, he places the letter inside a steel box and locks it with one of Alice's padlocks. Once the padlock is snapped shut, even he himself cannot re-open the box since he is not in possession of the combination which will release the lock. Only Alice will be able to open the box and therefore read the letter once she has received Bob's parcel in the mail.

Public key encryption works much in the same manner. The **public key** may be thought of as an open, electronic padlock. You can send this electronic padlock to all your friends. Your friends may then use that padlock to secure their emails to you in an electronic box. This electronic box is the encrypted

email. Upon receipt of the encrypted email, you dial the secret combination which is your **private key** and retrieve the original message.

SecExMail does all this for you.

## 5.7 SecExMail Key File Format

The SecExMail keys are stored in conventional text files ending in ".pubrsa" and ".privrsa" for public keys and private keys respectively. Files are divided into an administrative segment and a data segment. The administrative segment contains information required by SecExMail for key management.

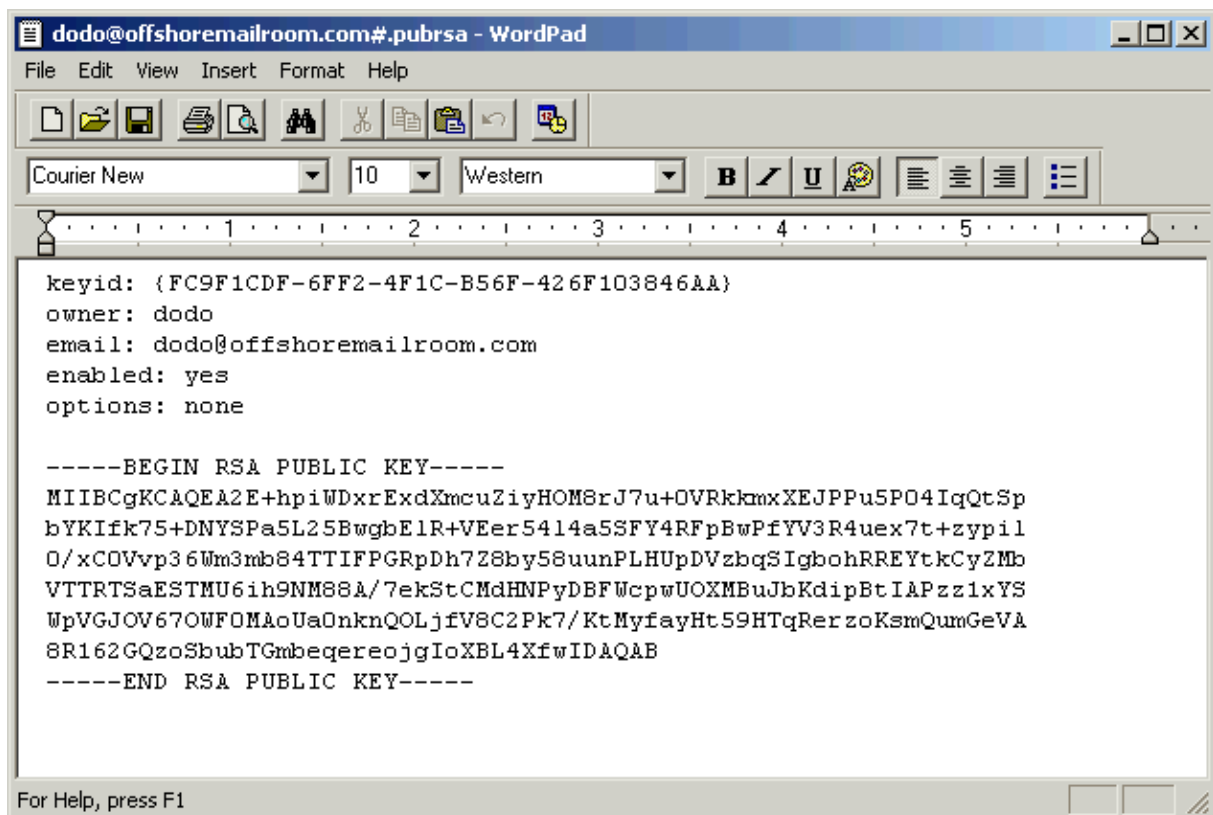
### Administrative Segment

keyid	Globally unique key identifier; used by SecExMail to associate private and public key components.
owner	owner of the SecExMail key
email	Email address of key owner
enabled	reserved for future use
options	vendor options field - reserved for future use

New lines in the administrative section are denoted by carriage return line feed pairs (ASCII characters 13 + 10).

### Data Segment

The data section is comprised of a single RSA key in base 64 encoded format. New lines in the data section are denoted by a single linefeed ( ASCII character 10 ). Private RSA keys are stored in 3DES encoded, chained block cipher format and protected with a passphrase.



SecExMail keys held in the registry are stored in a format analogous to keys stored on file - with each parameter represented as a registry value. See image below.

Name	Type	Data
(Default)	REG_SZ	(value not set)
email	REG_SZ	dodo@offshoremailroom.com
enabled	REG_SZ	yes
keyid	REG_SZ	{FC9F1CDF-6FF2-4F1C-B56F-426F103846AA}
options	REG_SZ	none
owner	REG_SZ	dodo
rsa	REG_SZ	-----BEGIN RSA PUBLIC KEY-----MIIBCgKCAQEA2E

## 5.8 Entropy Collection



Individual email messages are encrypted via session keys using the Twofish block cipher and ISAAC stream cipher. Each session key is then encrypted with the SecExMail public key for the recipient of the message. Upon receipt of the message, the session key is decrypted via the recipient's private key. Once the session key for the message has been retrieved, the message itself can be decrypted. In order for the message to be secure this session key must be both random and unknowable.

Consider the following scenario where a home owner protects his garden shed with a combination pad lock. Assume further that the brand of pad lock the home owner purchased has four dials, each bearing the digits "1" through "9", and that the dials have a slight tendency to snag on the number "7". If the tendency is slight enough so as to be hardly noticeable many a buyer will, without being aware of this, chose a combination involving one or more sevens. Ordinarily a thief would be compelled to try 10,000 settings in an exhaustive search for the correct combination. On average, therefore, a thief will succeed after 5000 tries. The educated thief, however, knows that all locks of this brand have a tendency to snag on the number "7". If the thief establishes that the first two dials are so affected, then only the second pair of dials is truly unknown. With some luck, the thief only needs to examine 100 combinations, 00..99 on the second pair of dials, in order to open the lock. Our number lock, although it provides for a "key space" of 10,000 combinations has a statistical bias - some combinations are more likely than others. In order for the combination lock to be useful, its combination must be entirely unknowable.

Much the same applies to encryption keys - size does matter. But for a large encryption key to be strong, it must be unknowable to a potential attacker. This requires the input of good random numbers during key generation. If the inputs to the key generation are not random, an attacker will be able to exploit the statistical bias. Why cut the lock, when you can simply dial the correct number ? Good randomness, unfortunately, is difficult to produce for modern computers. Computers are calculating machines which perform predefined operations according to predefined scripts, called programs. Nothing about a computer is random. Computing is 100% deterministic albeit complex and sometimes opaque to the human observer. To compensate for this shortcoming, random number generators accept what is referred to as a seed. The seed initializes the internal state of the random number generator and thus sets a starting point. Thereafter a complex mathematical sequence is applied to produce statistically pattern free output. If the starting point, or seed, to the mathematical sequence is unknowable, the random number generator can be said to be "truly random". This unknowable starting point is referred to as "entropy". The entropy of a system is the measure of its unpredictability.

Because computers are inherently deterministic, the best source of unpredictability is the human user. Many encryption systems make the mistake to digest state information about the computer, such as screen shots or process lists, gathered in short term observations into entropy data. Many encryption tools are confined to gathering entropy in this manner by the nature of their design. An encryption plugin for an email client, for example, is only invoked for a very short period of time when it is asked to encrypt an email message. It is then free to digest short term state information about the computer into entropy. The problem with this approach is that the next invocation will possibly produce similar state information. Thus if little has changed in the computer's state since the last invocation, the entropy collected at each invocation will exhibit a high degree of correlation. Some designs safeguard against this by writing a seed file to disk which transfers state information from one invocation to the next. However, the amount of entropy gathered by a program which only exists in computer memory for but a brief time is inherently limited.

For this reason, and to mine the entropy which may be found in the interaction between the human user and the computer, SecExMail operates an entropy collection subsystem which runs continuously during the operation of the computer, even when no email is being sent or received. The entropy collection extracts unknowable user data and re-seeds small subsections of the random number generator's state array as entropy data becomes available. A perfect source of randomness are keyboard timings and mouse clicks, mouse movements and mouse timings. The entropy collection subsystem **NEVER** records your keystrokes or any other user information, but exploits the timing information contained in system events generated by the user. Since some users tend to be slower typists than others, and yet other users make predominant use of the mouse, SecExMail uses two strategies to distill

"unknowability" from the data it collects.

### 1) Modulus Calculation

The modulus operation is a mathematical calculation which computes the remainder of integer division. For example  $7 \text{ MOD } 3$  equals 1. Perhaps you recall this kind of arithmetic from "Kindergarten Math". "How often does 3 go into 7?" Answer 2, remainder 1. This kind of arithmetic is useful in removing bias from nearly random data. For example one might conduct a survey recording the height of shoppers in a mall. When asking "random" adults how tall they are, the answers are likely to fall into a certain range : 5 foot 9, 6 foot 1, 5 foot 11, etc. While the exact answers will be unknowable to someone who did not accompany the surveyor, the collected data will not be entirely unknowable as the majority of answers will fall into a certain range. However, consider what happens when we compute the "modulus 3" of the above values.

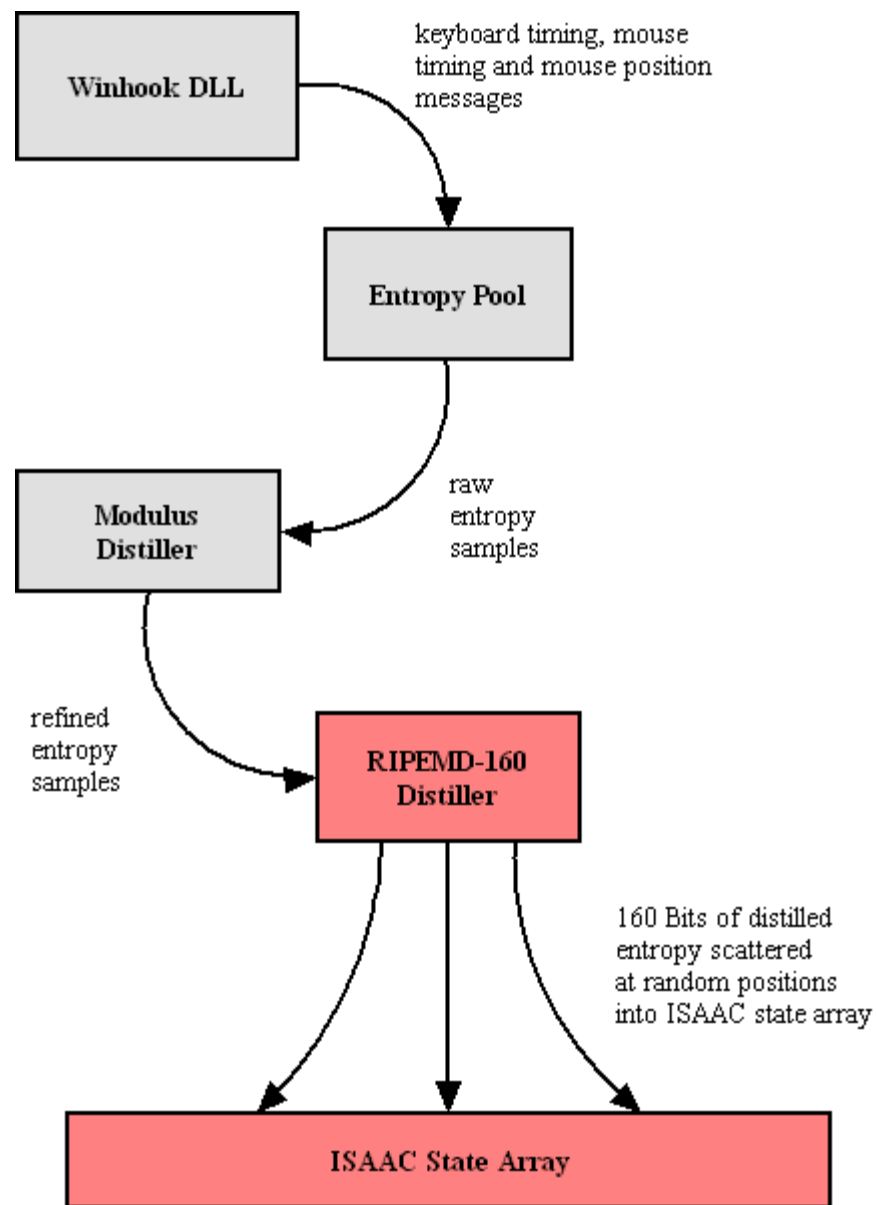
Data	Inches	Modulus 3
5 ' 9"	69 "	0 ( $23 * 3 = 69$ , remainder 0 )
6 ' 1 "	73 "	1 ( $24 * 3 = 72$ , remainder 1 )
5 ' 11"	71 "	2 ( $23 * 3 = 69$ , remainder 2 )

Regardless of the ethnicity of the population surveyed, someone who did not accompany the surveyor will be unable to predict the sequence of zeroes, ones and twos which will emerge. The average height of the examined population is biased, but whether a population member's height is an integral multiple of 3 inches is entirely unknowable.

### 2) Secure Hash Functions

The second strategy employed by the entropy collection subsystem to distill randomness from biased data is to apply a secure hash function. Secure, one way hash functions are used for digital signatures and cryptographic checksums. According to Ron Rivest, one of the designers of [RSA encryption](#), hash functions are designed such that "It is computationally infeasible to find two messages that hashed to the same value. No attack is more efficient than brute force." As such, hash functions tend to preserve the smallest differences in a sample and have the added property of preserving the entropy found in the sample. It is important to note that SecExMail does not use secure hash functions to "stretch" the randomness in small data sets, but to distill the entropy in data pools containing hundreds or thousands of data items to 160 bits of entropy. SecExMail employs the RIPEMD-160 message digest.

Entropy is gathered in entropy pools, distilled as described above and finally scattered randomly into the state array of the ISAAC random number generator. The diagram shown below depicts the flow of data in the entropy collection subsystem.

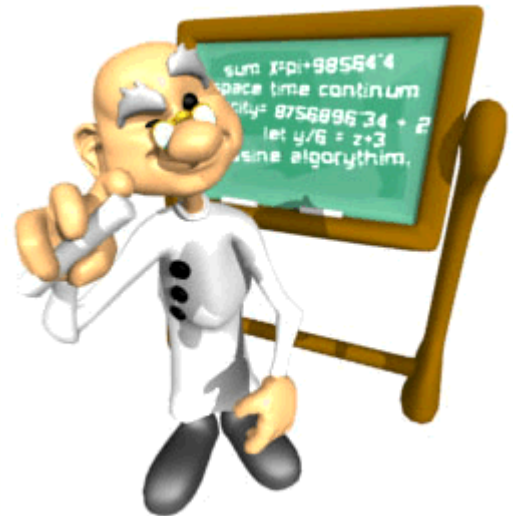


To prevent attacks on the random number generator based on the monitoring of key strokes and mouse events by third parties, the SecExMail entropy collection does not employ event timings which represent times at which the operating system recorded the events, but instead uses internal timestamps which represent the times at which the events were recorded by SecExMail's own message queue. This message queue is subject to further timing distortions by other events and the general multitasking behavior of the application.

## 5.9 One-Time Pads

A one-time pad is a block of random data used to encrypt a block of equal length plain text data. Encryption is usually by way of XOR'ing the one-time pad with the message text. This process may be thought of as a 100% noise source used to mask the message. The one-time pad is secure if it is comprised of random data and is never reused. Because of this, one-time pads have limited application in modern ciphers, but are commonly acknowledged as the holy grail of cryptography.

SecExMail uses one-time pads to encrypt the ZLIB compression header in [SecExMail messages](#).

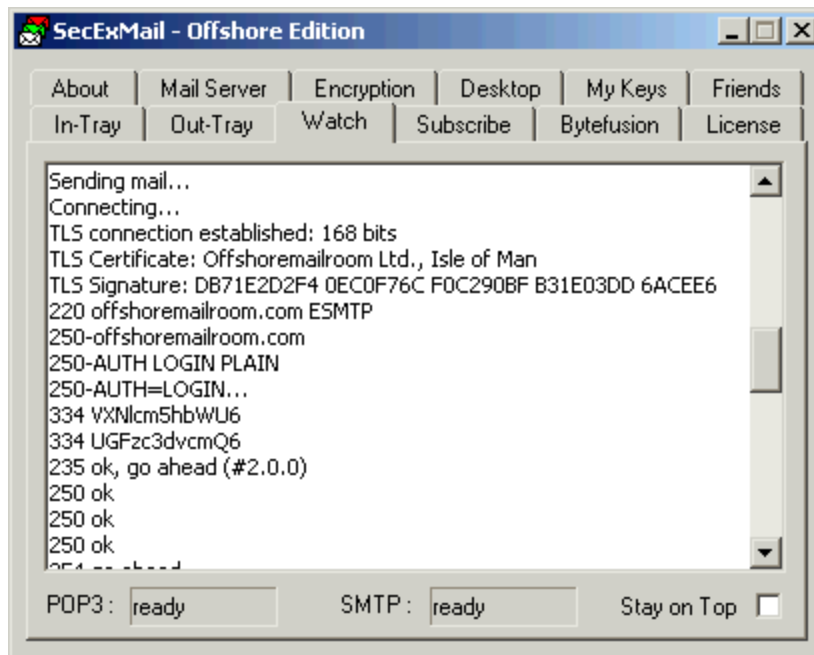


## 5.10 IP / DNS Spoofing

IP spoofing is the creation of IP packets using someone else's IP address. DNS spoofing is the substitution of a different IP address for a DNS name. DNS spoofing is commonly achieved by corrupting the DNS database of the DNS server your computer connects to in order to match human readable computer names to physical IP addresses. In both instances, the computer you are connecting to is not the server you expect.

This can be used, for example, to trick you into giving your server user name and password to the computer acting as the impostor. Alternatively, the impostor might simply act as a conduit whilst talking to the real server on your behalf. This is called a "Man-in-the-middle attack" and is commonly used to intercept network traffic without the knowledge of the original participants.

SecExMail protects against IP and DNS spoofing via SSL / TLS certificates. At the start of each connection attempt, the server certificate is verified to establish the server's true identity and the digital signature of the certificate is recorded. (Offshore and Corporate edition only)



## 5.11 Known Plain Text Attack

A known plain text attack is the attempt by a cryptanalyst to break a cipher based on knowledge about the plain text of a message prior to its encryption. Simply put, if the cryptanalyst knows the method of encryption, any encryption, part or all of the plain text input to the cipher, and is able to observe the encrypted message text, he / she will likely be able to infer the key used to encrypt the message. This in turn can compromise the security of future messages sent with that key. In greatly simplified terms :

**Plain Text + Key = Cipher Text**  
**Cipher Text - Plain Text = Key**

Consider the following scenario : Alice sends Bob an email and attaches her favorite holiday snapshot. The email is encrypted. Assume further that she sends the same holiday snapshot to her mother in plain text. Steve, who wishes to spy on Alice and Bob, was able to intercept her email to Mom and now has a copy of "myholiday.jpg". If the picture consisted of 200 Kilobytes of data (about 200,000 letters) and Alice included only a short personal message to Bob with the picture ( say 50 letters ), then Steve already knows 99% of the message contents prior to encryption and now has greatly improved chances of breaking Alice's key if he comes into possession of the corresponding cipher text.



SecExMail includes comprehensive protection against known plain text attacks. See [SecExMail Cipher](#) and "Proactive Security" under [What is SecExMail](#).

## 6 FAQ

### 6.1 What email clients work with SecExMail ?

The following email clients have been tested and are known to work with SecExMail :

- Outlook 98/2000/Express
- Outlook XP/2002
- Eudora
- Pegasus
- The BAT
- Calypso
- IncrediMail
- [Netscape Mail](#)

However, the only formal requirement is that your email client be compatible with the SMTP and POP3 mail protocols for sending and receiving mail respectively. You must [configure your email client](#) to talk to your computers loop-back address ( **127.0.0.1** or **localhost** ) and [configure SecExMail](#) to talk to your ISP's email server.

If your email client is POP3 and SMTP compatible and is unable to send and receive mail through SecExMail, we want to hear from you. Please contact us at [support@bytefusion.com](mailto:support@bytefusion.com) with details of your email client including its version number as well as the details of your operating system.

### 6.2 Does SecExMail work with IMAP?

SecExMail does not work with IMAP mailboxes. [SecExMail requires SMTP and POP3.](#)

### 6.3 How secure are SecexMail keys ?

SecExMail uses industry standard [RSA keys](#). RSA is the worlds foremost public key encryption system. The security of your personal SecExMail key will vary depending on the key size. Please consult the documentation on the [SecEx Key Generator](#) and [key size](#) for details.

The session keys size is 256, 64 and 64 for the Twofish block cipher, the ISAAC stream cipher and the One-Time Pad respectively. See [the SecExMail composite cipher](#) for details.

### 6.4 Is SecExMail legal in my country ?

SecExMail is distributed from the Isle of Man, a self governing offshore island in the Irish Sea. The Isle of Man has no laws restricting encryption technologies and is committed to pro e-commerce legislation. See [E-Island Fact Sheet](#) published on the Isle of Man government web site [www.ecommerce.gov.im](http://www.ecommerce.gov.im). However, you are advised to check that the laws of your country allow the import and use of mass encryption software before downloading and/or using SecExMail. Please see the terms of the license agreement for details.

Bytefusion Ltd. endeavors to comply with the highest standards of local and international law. Persons who reside in any country which is listed or becomes listed as embargoed by the United Nations with regard to export of encryption software are prohibited from using SecExMail.

SecExMail is compliant with the [Wassenaar Arrangement](#) on encryption technologies; it is exempt from the Wassenaar dual-use restrictions as per **Category 5, Section 2, "Information Security"**. SecExMail complies with **Note 3** on **cryptography** :

- SecExMail is generally available to the public by being sold via electronic and mail order transactions.
- The cryptographic functionality in SecExMail cannot be easily changed by the user.
- SecExMail is designed for installation by the user without further substantial support by the supplier.
- Details of the encryption algorithm are available in this document. Printed versions available upon request.

SecExMail is also compliant with the European Union Council Regulation No 2432/2001 which implements the Wassenaar Arrangement as detailed above.

SecExMail is warranted to be free from spy-ware, key escrow or key recovery features of any kind.

## **6.5 Does SecExMail support signatures ?**

Digital signatures are not supported at this time. The next version of SecExMail will support this feature.

## **6.6 Does SecExMail work with PGP ?**

SecExMail does not decrypt PGP messages and does not encrypt messages which can be read by PGP users.

However, SecExMail encrypts the mail stream and therefore does not interfere with existing methods of encryption. As such, it is possible to encrypt with PGP first, and then send the resulting cipher text through SecExMail. This effectively results in "doubled-up" super cipher encryption.

## **6.7 Is the source code available for SecExMail ?**

The source code for the core cryptographic components of SecExMail is freely available on the internet.

- Twofish block cipher

At the time of writing the Twofish homepage can be found at <http://www.counterpane.com/twofish.html>.

- ISAAC

At the time of writing, the ISAAC home page can be found at <http://burtleburtle.net/bob/rand/isaacafa.html>.

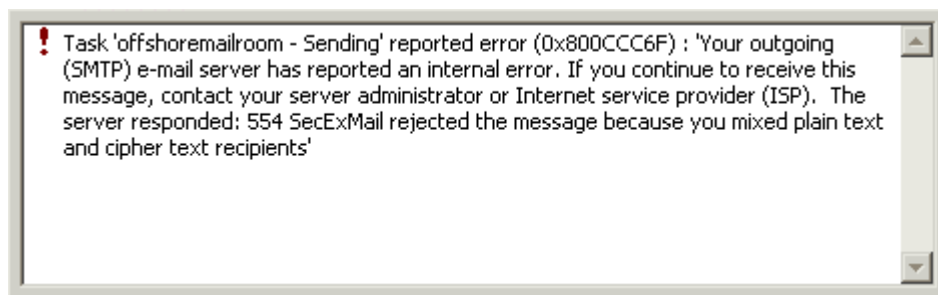
- OpenSSL

SecExMail contains cryptographic software from the OpenSSL project at [www.openssl.org](http://www.openssl.org)

## 6.8 Why can I not mix clear text and cipher recipients ?

I tried to send email to someone on my secure friends list and CC'ed ( *carbon copied* ) the message to someone not on my friends list and Outlook reported the following error :

**"SecExMail rejected the message because you mixed plain text and cipher text recipients."**



Consider the following scenario. You compose a message and address it to two friends, Bob and Alice. Bob has a SecExMail key - Alice does not. However, your email client sends one message only. Your mail server is responsible for distributing your single message to all recipients, not your email client. This is a design feature of SMTP, the protocol which governs distribution of email messages across the public internet. Because the message has been encrypted however, Alice cannot read it and asks you to resend the same message again, in plain text. You oblige her.

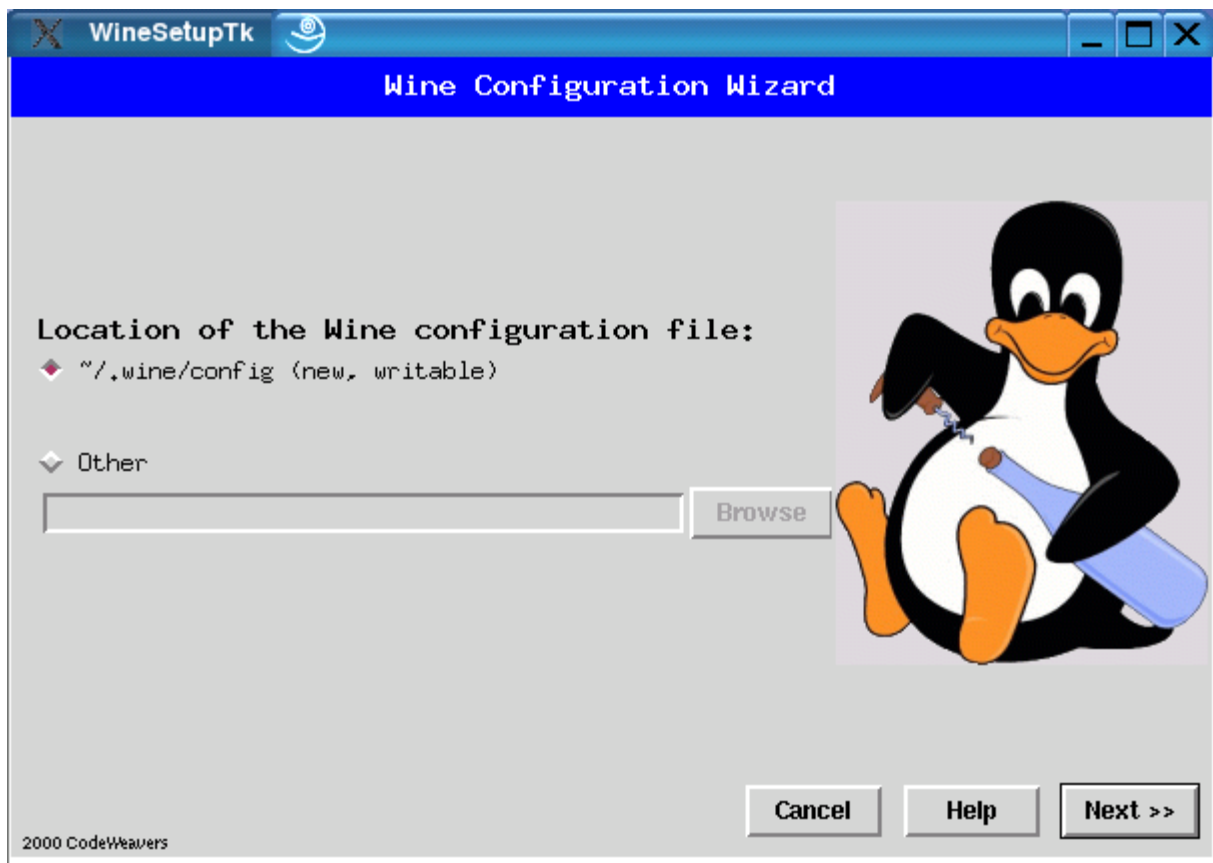
If someone were to observe you sending the same message twice to the same recipient, once in cipher text and once in plain text, they could reasonably mount what is called a [known plain text attack](#) on the session key or your SecExMail key because they know both the input ( *plain text* ) and the output ( *cipher text* ) and might therefore deduce your key. For this reason, SecExMail has disallowed the action.

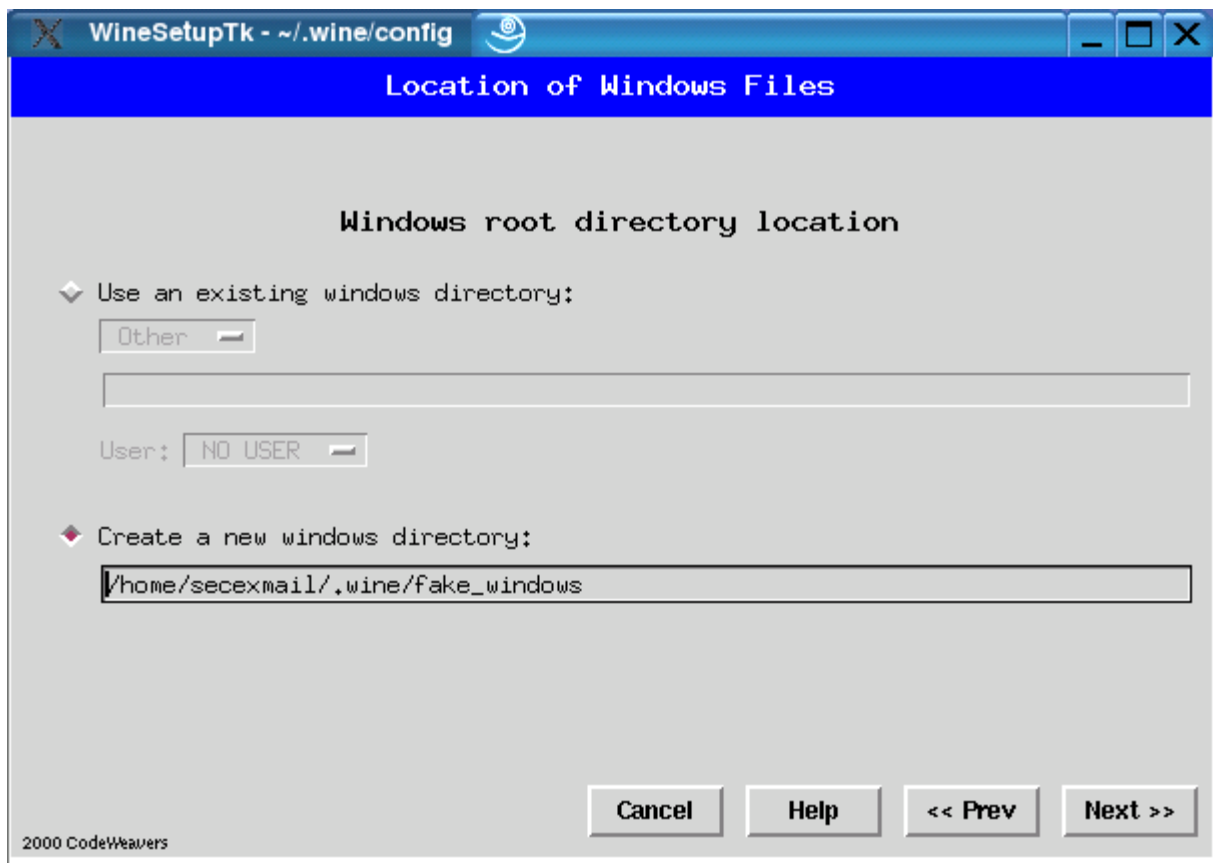
Simply compose two messages, one to the person on your friends list and another, separate message to the person who is not on your friends list.

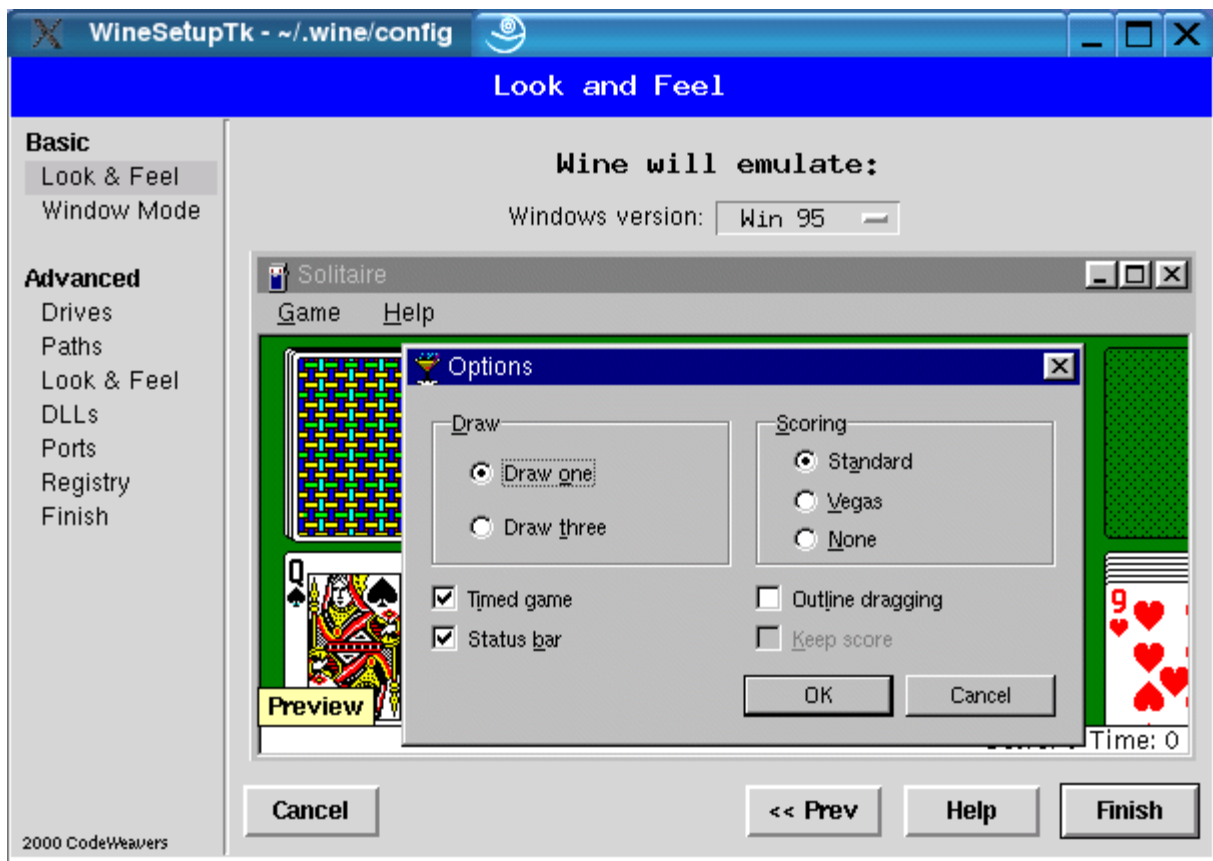
## 7 Just Linux

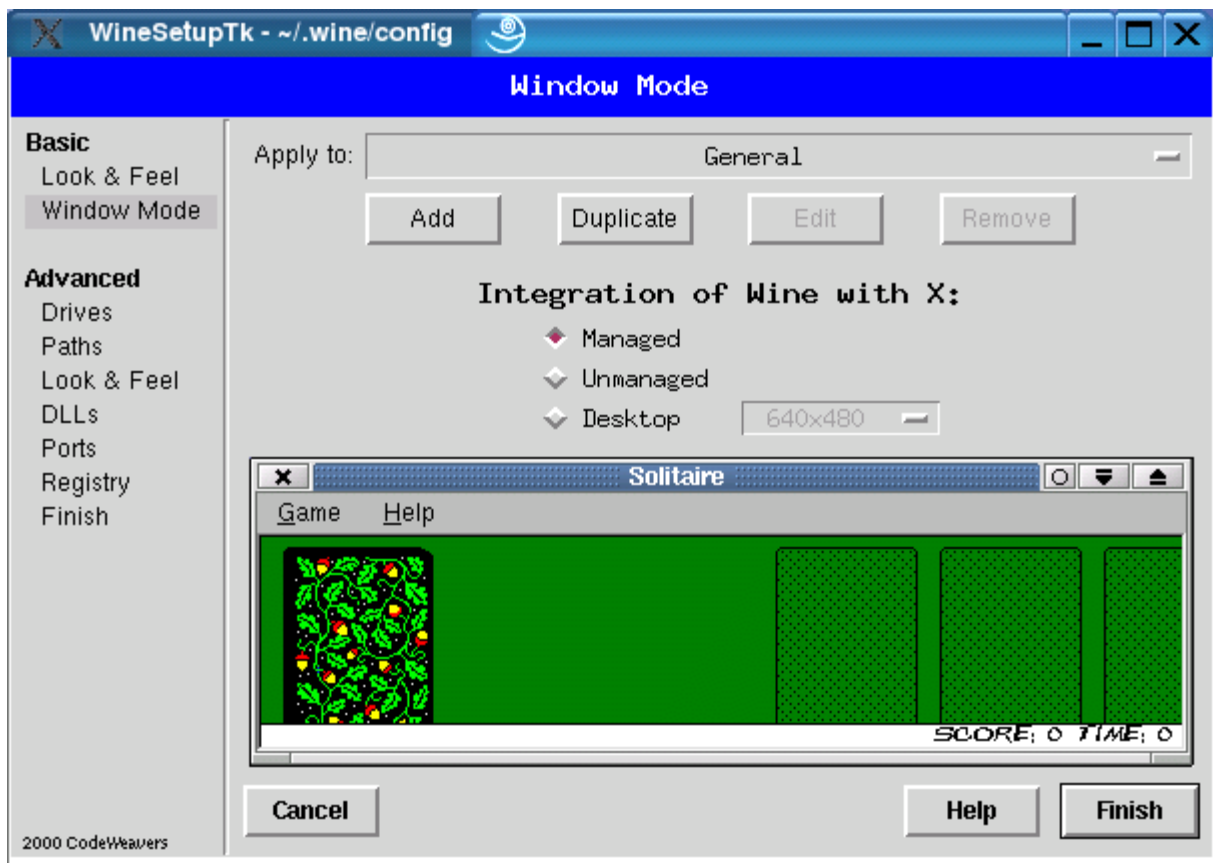
### 7.1 WINE Configuration

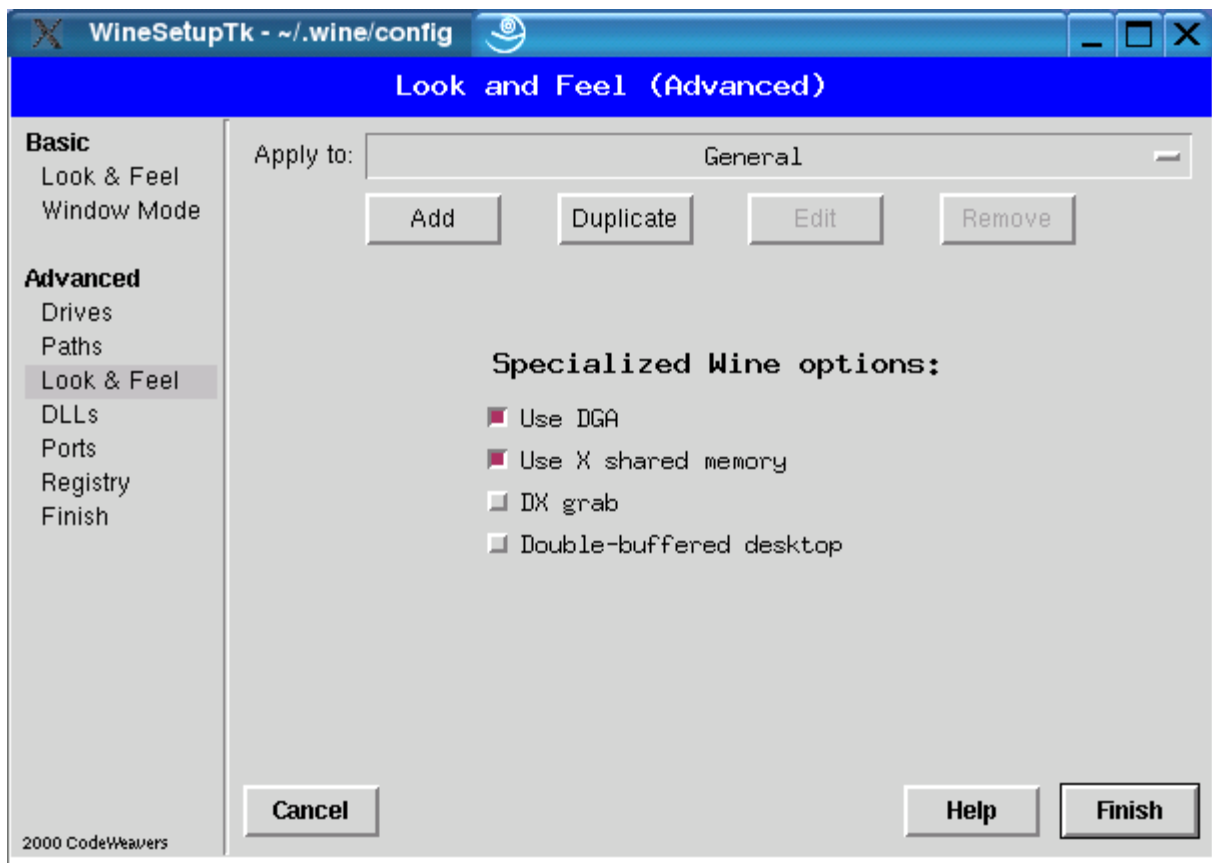
SecExMail for Linux has been tested to work with WINE, version 20030217, under SuSE Linux 8.2. All SecExMail functionality is available using the stock WINE configuration in Windows 95 mode as created during SuSE 8.2 setup. If you experience difficulties or wish to configure WINE for SecExMail using a different Linux distribution, kindly refer to the WINE configuration screenshots shown below.

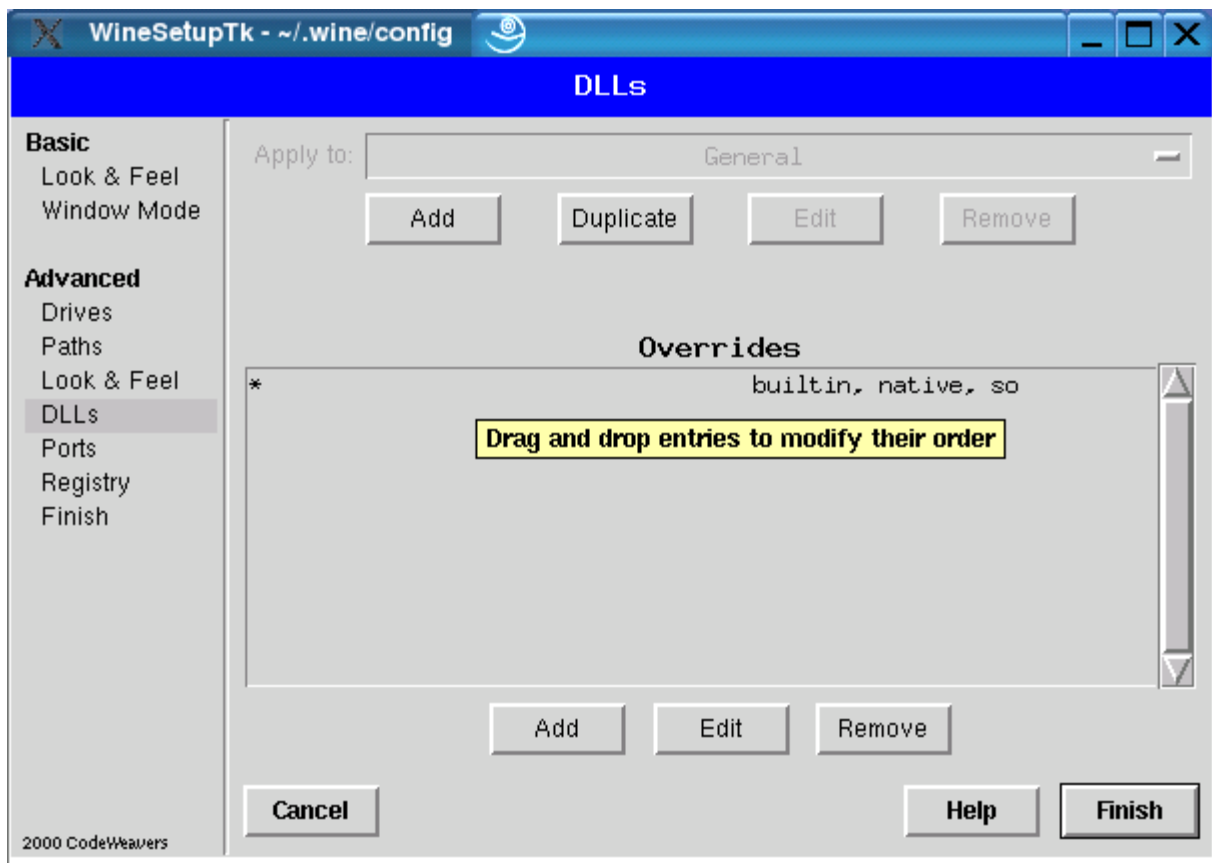


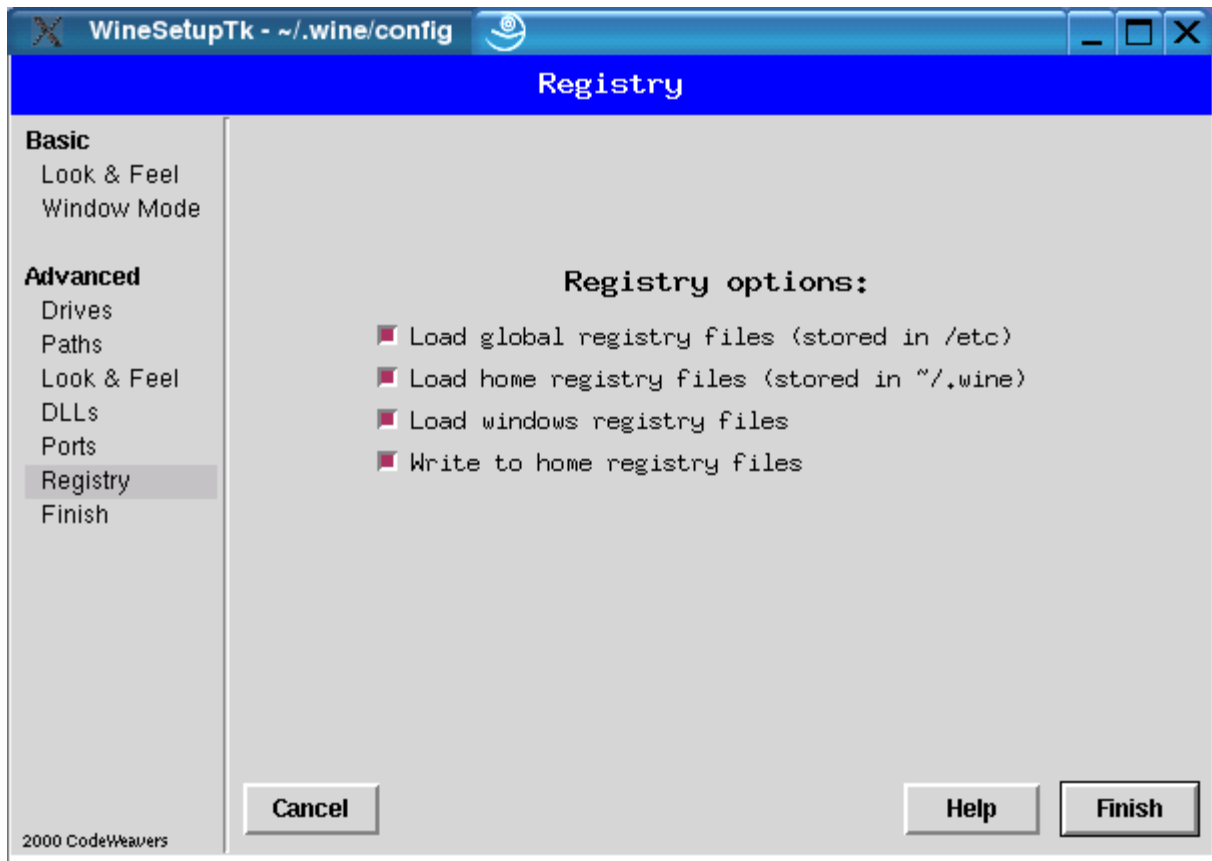








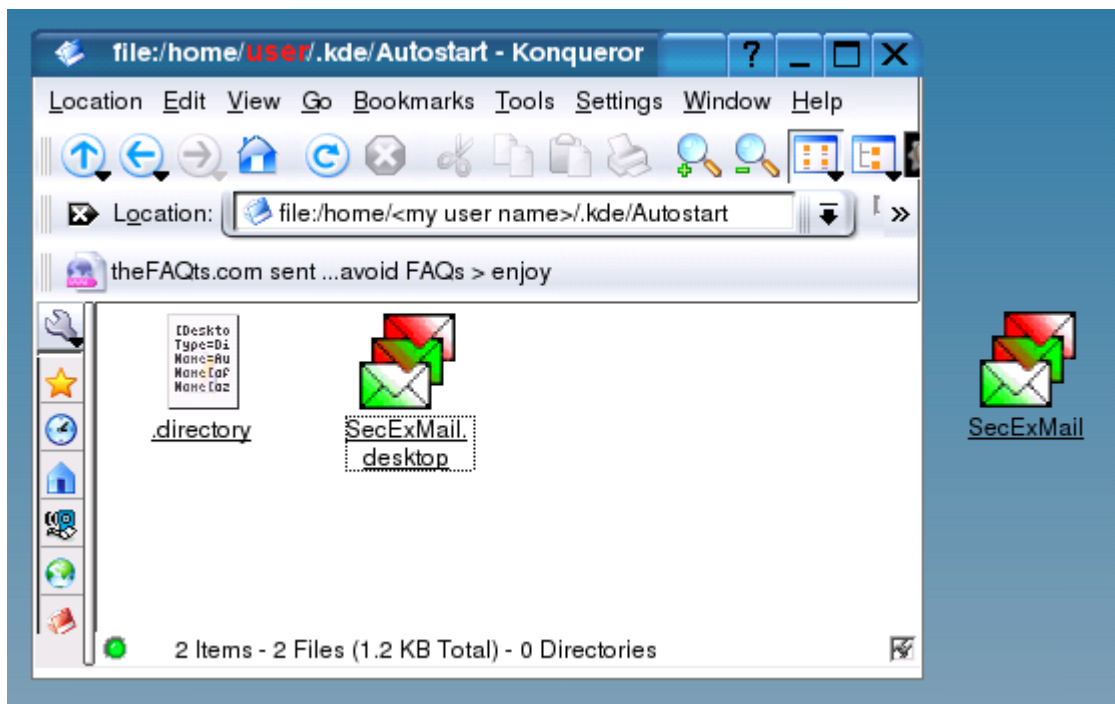




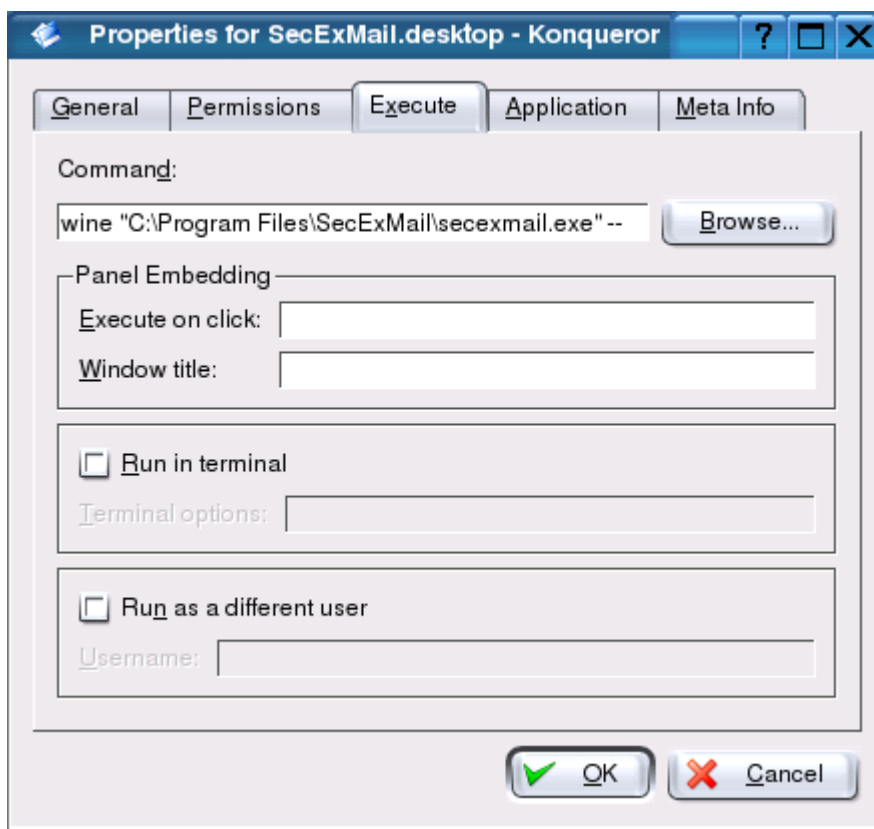
Microsoft Windows is a trademark of Microsoft Corporation.

## 7.2 KDE Autostart Menu

If you want SecExMail to start automatically immediately after you log in, you need to create a shortcut in the Autostart folder of your user home directory. Simply copy the desktop shortcut created by the setup into this folder as shown below.

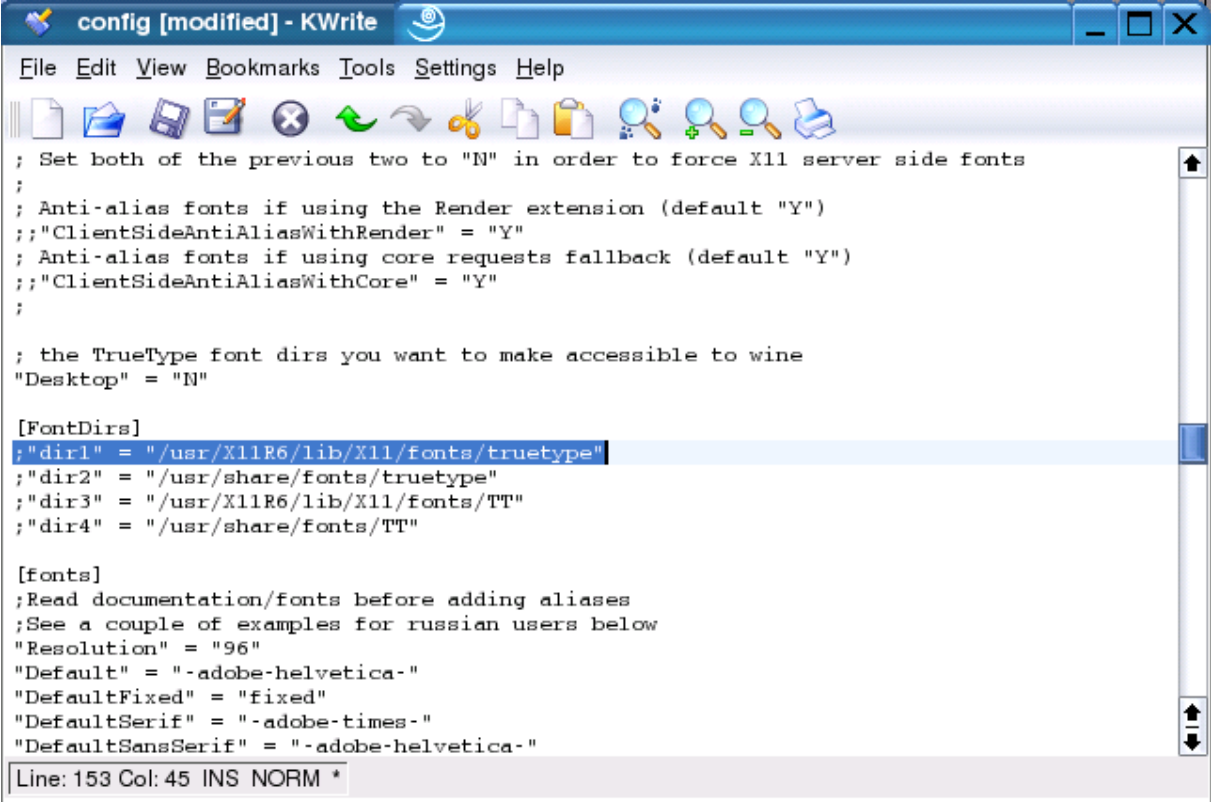


Optionally, you may create the shortcut manually as indicated below.



## 7.3 Font & Display Issues

If you configured WINE manually for user accounts created after the initial setup of SuSE Linux, some dialogs in SecExMail may not display properly depending on the fonts installed on your system. WINE setup attempts to detect and where available use X11 true type fonts available on your system. In some cases the use of X11 true type fonts may lead to display problems. You can comment out true type fonts by prefixing the relevant line in your WINE configuration file with a semicolon as shown below. Your WINE configuration file should be located in your home directory under ".wine/config".



```
; Set both of the previous two to "N" in order to force X11 server side fonts
;
; Anti-alias fonts if using the Render extension (default "Y")
;;"ClientSideAntiAliasWithRender" = "Y"
; Anti-alias fonts if using core requests fallback (default "Y")
;;"ClientSideAntiAliasWithCore" = "Y"
;

; the TrueType font dirs you want to make accessible to wine
"Desktop" = "N"

[FontDirs]
;"dir1" = "/usr/X11R6/lib/X11/fonts/truetype"
;"dir2" = "/usr/share/fonts/truetype"
;"dir3" = "/usr/X11R6/lib/X11/fonts/TT"
;"dir4" = "/usr/share/fonts/TT"

[fonts]
;Read documentation/fonts before adding aliases
;See a couple of examples for russian users below
"Resolution" = "96"
"Default" = "-adobe-helvetica-"
"DefaultFixed" = "fixed"
"DefaultSerif" = "-adobe-times-"
"DefaultSansSerif" = "-adobe-helvetica-"

Line: 153 Col: 45 INS NORM *
```

## 8 About

### 8.1 About SecExMail



**SecExMail**  
**Version 1.5**  
**Copyright © 2002-2004, Bytefusion Ltd.**  
**All Rights Reserved**

### 8.2 About Bytefusion Ltd.



**Bytefusion Ltd.**  
**22 Duke Street**  
**Douglas, IOM**  
**IM1 2AY**  
**British Isles**

**Inquiries: [sales@bytefusion.com](mailto:sales@bytefusion.com)**

## 8.3 Requirements

- SuSE Linux 8.2 or later
- WINE 20030217
- KDE 3.1 or later
- SMTP and POP3 compliant email client, such as Netscape Mail, Evolution, KMail, etc.
- Access to internet mail server ( SMTP & POP3 )
- Pentium class IBM compatible computer

