Classic Client 5.1 for Linux

User Guide

gemalto
security to be free

GEMALTO, B.P. 100, 13881 GEMENOS CEDEX, FRANCE.

Tel: +33 (0)4.42.36.50.00 Fax: +33 (0)4.42.36.50.90

Printed in France.

Document Reference: DOC116987A

June 30, 2008

Contents

List of Figures

Welcome to Gemalto Classic Client for Linux.

You have made a wise investment by purchasing Classic Client as a safeguard for secure network services.

This chapter presents an overview of Classic Client, the documentation provided with it, and additional resources available for working with Classic Client.

Classic Client

Classic Client is for individual users, who want to use a smart card/token to protect information and transactions made via computers, including stand-alone workstations and Citrix client-server environments.

Note: A token is in fact a smart card embedded in a device that can be plugged into the USB port of a PC. In this document, “connecting a device” can mean inserting a card in a reader or PC or plugging a token in the USB port of a PC.

With Classic Client you can use a digital certificate stored on a smart card/token to:

- Sign electronic documents.
- Open and verify signed documents.
- Send and receive secure e-mail using Mozilla e-mail software.
- Connect securely with a Web server.

Classic Client also includes features for managing certificates and smart card/token security.

This guide introduces you to Classic Client and provides easy-to-follow instructions. Read the entire guide for assistance in the installation, configuration, and use of Classic Client.

Who Should Read This Book

This guide is intended for Classic Client users who are familiar with smart cards/tokens and smart card reader technology, as well as PC hardware and software.

It is assumed that the user of Classic Client has:

- an understanding of the basic operations in a Linux OS.
- administrative privileges for the PC on which Classic Client will be installed.

Documentation

Classic Client is delivered with the following documentation:

- *Classic Client 5.1 for Linux* (this document). The file for this document is located on the Classic Client 5.1 CD and in the Classic Client installation folder.
- A *Readme* file. This contains any relevant information about the installation and the complete version history.

This document is best viewed with Adobe Acrobat Reader, version 7.0 or later. You can download Adobe Acrobat Reader from Adobe's Web site at: www.adobe.com.

Conventions

The following conventions are used in this document:

Typographical Conventions

Classic Client documentation uses the following typographical conventions to assist the reader of this document.

Convention	Example	Description
`Courier`	transaction	Code examples.
Bold	Enter **libgclib.dylib**	Actual user input or screen output.
>	Select File > Open	Indicates a menu selection. In this example you are instructed to select the **"Open**" option from the "**File"** menu.

Note: Example screen shots of the Classic Client for Linux software are provided throughout this document to illustrate the various procedures and descriptions. These screen shots were produced with Classic Client running on Debian.

Additional Resources

For further information or more detailed use of Classic Client, additional resources and documentation are available by contacting Gemalto technical support.

Contact Our Hotline

If you do not find the information you need in this manual, or if you find errors, contact the Gemalto hotline at http://support.gemalto.com/.

Please note the document reference number, your job function, and the name of your company. (You will find the document reference number at the bottom of the legal notice on the inside front cover.)

1

Installation

This chapter discusses information related to the installation of Classic Client 5.1 for Linux.

The installation requirements are outlined below.

This chapter describes:

- The hardware and software you need to use Classic Client 5.1 for Linux.
- How to install Classic Client 5.1 on your computer.

System Requirements

The following sections describe the hardware, operating systems, peripherals and software you need to use Classic Client 5.1. You must have administrator rights to the computer on which you are installing Classic Client.

Computer

The workstation must have at least 15 MB of available hard disk space and meet the normal system requirements to run the version of Linux installed.

Operating Systems

Classic Client for Linux supports the following operating systems:

- Debian Etch
- RHEL 5 (Red Hat Enterprise Linux)

Gemalto recommends that your machine has a RAM at least equal to that normally recommended for the OS. If this RAM requirement is met, Classic Client for Linux should run normally.

Applications

Classic Client 5.1 is compliant with:

Browsers

To view secure Web sites from your computer with Classic Client 5.1 for Linux, you can use any of the following Web browsers:

- Mozilla Firefox version 2.0/3.0

  You can download the latest version, free of charge, from www.mozilla.org.

- Iceweasel (Iceweasel is the Debian version of Firefox)

  You can download the latest version, free of charge, from various sites on the internet. Further information is available in http://wiki.debian.org/Iceweasel.

- Netscape Navigator 9

  You can download the latest version, free of charge, from

  http://browser.netscape.com.

E-mail Applications

Classic Client 5.1. supports Mozilla Thunderbird version 2.0.

Note: You can download the latest version of Thunderbird, free of charge, from www.mozilla.org.

Peripherals

Classic Client 5.1 for Linux requires the following peripherals:

- A CD ROM drive.
- An available USB port.

Smart Card Readers

One of the following plug-and-play smart card readers:

- **USB Shell Token V2 (old name = GemPCKey)**
- **PC Twin Reader (old name = GemPC Twin)**

Secure PIN Pad Readers

Classic Client 5.1 for Linux supports smart card readers providing a highly secure way to enhance your smart card-based application by protecting the smart card PIN code from unauthorized access. The PIN code is entered locally and safely on the reader, and is never transmitted to the PC.

The following Secure PIN PAD readers are supported:

- PC Pinpad from Gemalto

Smart Cards

Gemalto has recently renamed many of its products. “Table 1” gives the full list of smart cards supported by Classic Client and gives their previous names.

Note: The cards actually supported by your Classic Client for Linux, depends on the package you have bought. For example, one package supports cards with the Classic Applet V1 only, another supports cards with Classic Applet V2 or Classic Applet V3

Table 1 - Supported Smart Cards

Card and installed applets	Previous name
Classic TPC IXS (Classic Applet V1)	GemSafeXpresso 16K (GemSafe v1.11 applet)
Classic TPC IS (Classic Applet V1)	GemSafeXpresso 32K (GemSafe v1.11 applet)
Classic TPC IS v2 (Classic Applet V2 default - V1 on demand)	New product
Classic TPC IS CC (Classic Applet V2)	GemSafeXpresso 36K CC (GemSafe v2 applet by default or GemSafe V1.11 on demand)
Classic TPC IM (Classic Applet V2)	New product
Classic TPC IM CC (Classic Applet V2 default - V1 on demand)	GemSafeXpresso 64K CC (GemSafe v2 applet by default or GemSafe V1.11 on demand)
Classic TPC IM CC v3 (Classic Applet V3)	New product

Installing Classic Client 5.1 for Linux

Installing the Classic Client 5.1 for Linux Software

Caution: Before installing the software make sure that your system has the latest version of the PC/SC Lite and CCID drivers.

To install Classic Client 5.1:

1 Begin by doing one of the following:

- If your administrator has provided an installation CD-ROM, insert the CD-ROM into the CD-ROM reader of your PC.
- If your administrator has made the installation program available from a network device, navigate to the network location and download the installation program to your computer.

2 Open a terminal and go to the directory where your installation program is. Start the installation program by typing the following line at the command prompt according to your OS:

- In Debian, type DPKG -i libclassicclient_x.x.x-xx_i386.deb
- In RHEL 5, type RPM -i libclassicclient_x.x.x-rhel5-xx_i386.rpm

Note: You can do this without going to the installation directory first, but if you do, you must type the full path for the file name.

Classic Client installs on your PC with no need for further intervention.

Connecting the Smart Card Reader

To use Classic Client on your workstation, you must connect a smart card reader to your computer.

If the card reader is not recognized on your workstation, you may need to install the latest card reader drivers. You can download these from http://support.gemalto.com.

Configuring Gemalto Cryptographic Security Modules

Security Modules are software add-ons that provide a variety of cryptographic services, such as secure browsing, and support the use of smart cards/tokens.

In Classic Client 5.1 for Linux, the PKCS#11 security module is installed automatically as it is included with the Classic Client software.

In order to enable the Mozilla applications Firefox and Thunderbird to communicate with Classic Client, the PKCS#11 security module must be registered in the Mozilla application.

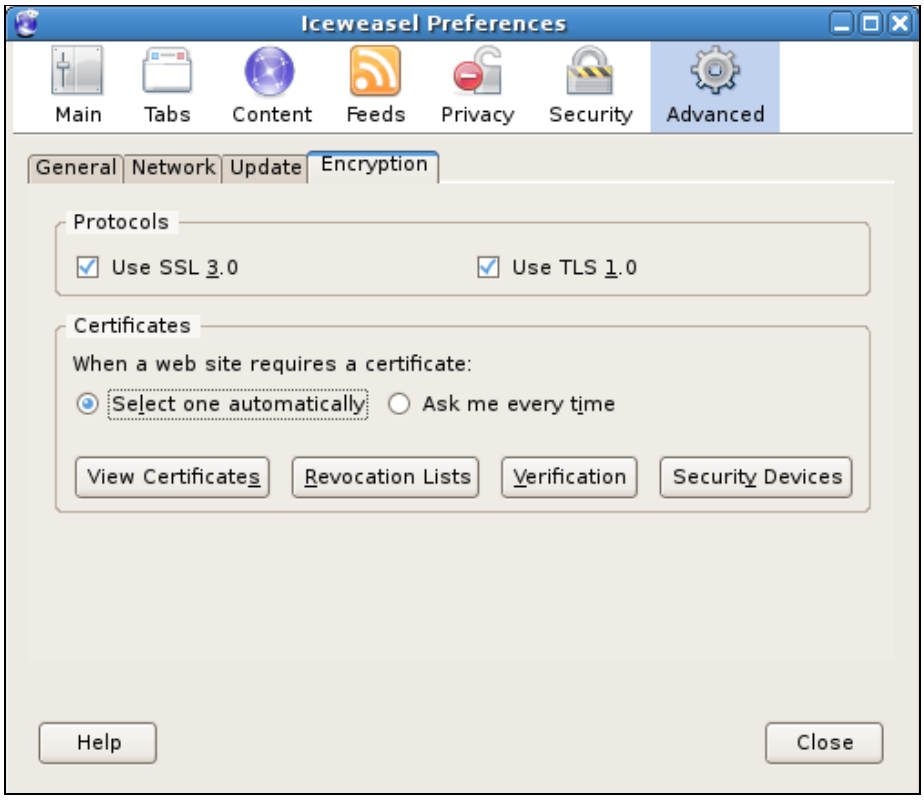
Note: The screen shots in this section were taken on a PC running the Debian OS. In Debian, the Firefox browser is called Iceweasel, and its appearance is slightly different although its functionality is the same.

To configure Firefox (or Iceweasel) to recognize the security module:

1 Open **Firefox** or **Iceweasel** and from the **Edit** menu choose **Preferences**.

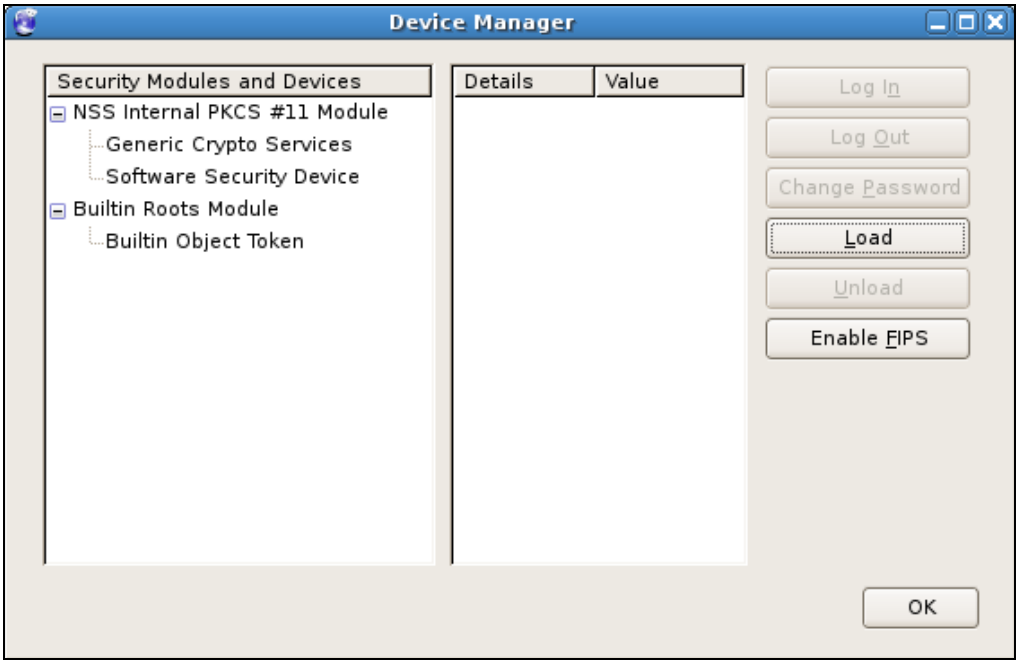
2 In the dialog box that opens, click the **Advanced** icon, then the **Encryption** tab to display the settings as shown in “Figure 1”.

Figure 1 - Encryption Tab in Advanced Dialog



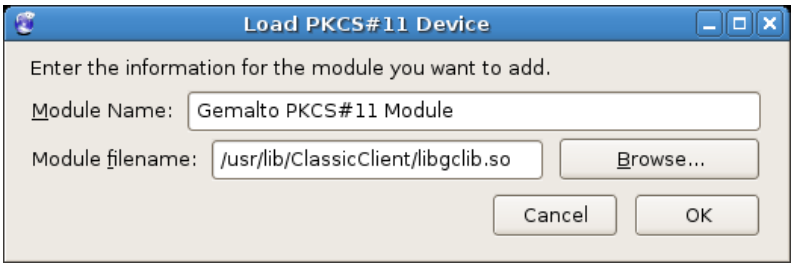
3 Click **Security Devices** to display the **Device Manager** window. This displays the modules currently available as shown in “Figure 2” on page 5.

Figure 2 - Device Manager



4 Click the **Load** button to the right in the dialog. This displays the **Load PKCS#11 Device** window, as shown in “Figure 3”.

Figure 3 - The Load PKCS#11 Device Dialog Box



5 Enter a **Module Name.**

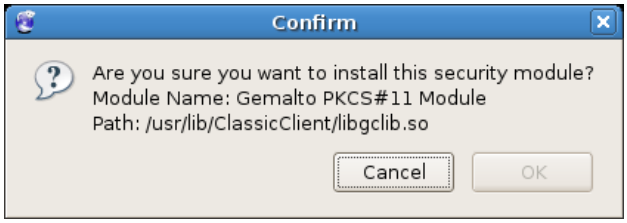
6 In **Module filename**, enter the full path and filename for the libgclib.so file:

By default, this is

/usr/lib/ClassicClient/libgclib.so

7 Click **OK**. The confirmation dialog appears as shown in the following figure:

Figure 4 - Confirm Dialog

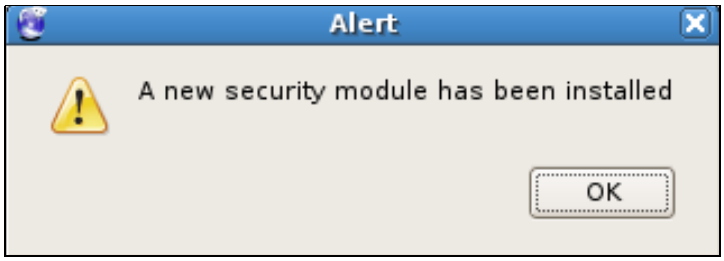


8 Click **OK**.

A brief progress dialog appears indicating that the module is being loaded.

When this is completed the following **Alert** indicates that the module has been installed.

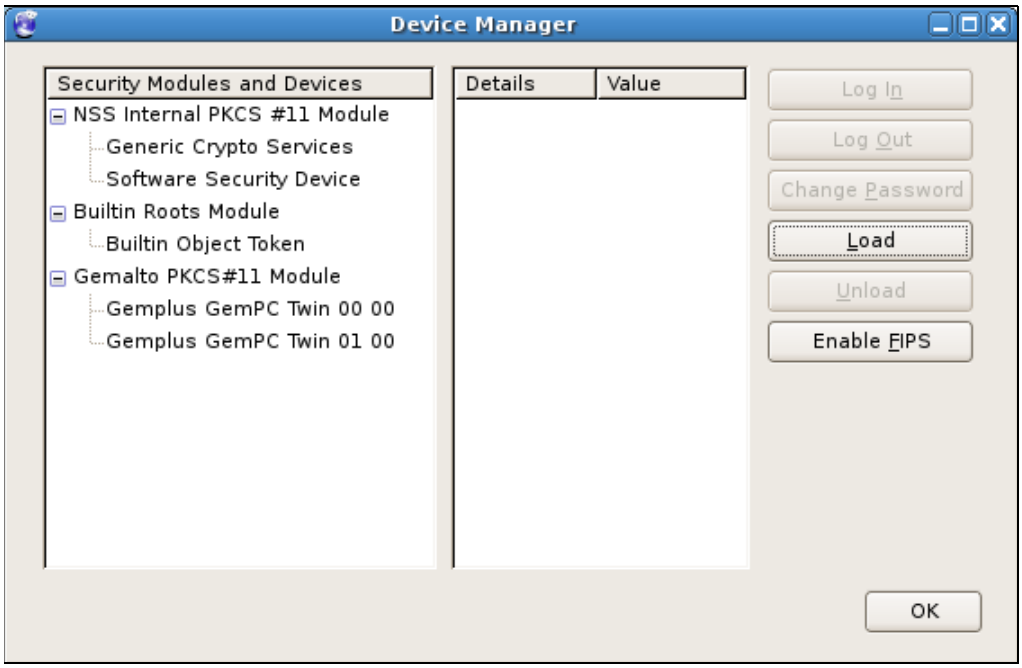
Figure 5 - Alert Dialog



9 Click **OK** to close this **Alert**.

The **Device Manager** indicates the presence of the new module as shown in “Figure 6”.

Figure 6 - Cryptographic Modules Available



To manually install a security module for Netscape Navigator 9:

The procedure is identical to that described for Firefox and Iceweasel on page 4, except that in step 1, you choose **Preferences** from the **Navigator** menu instead of from the **Firefox** or **Iceweasel** menu.

2

PIN Management

This chapter discusses the Classic Client PIN Management tool, the dedicated tool for managing PINs and the tasks it can be used to perform.

About PINs

PIN Types

Classic Client recognizes two types of PIN that may be in a smart card/token:

- Admin PIN – the PIN that is necessary to unblock the card/token (for example after too many consecutive incorrect presentations of the User PIN).
- User PIN – the standard PIN used by a user to access the card/token.

The Administrator PIN

This is the PIN used to unblock a User PIN. Normally only administrators know the value of this PIN.

The administrator PIN is an extremely important part of the security of the smart card/token. Knowledge of this PIN means you can change the value of all the user PINs on the card/token and unblock the card/token if the user PIN is blocked.

It is extremely important for smart card/token administrators to keep the value of the admin PIN secure and secret. The administrator must know the admin PIN value for all smart cards/tokens he or she has deployed. The admin PIN value of a card/token should never be shared with anyone else, and it is strongly recommended not to give this value to the card/token user, unless your security policy requests it.

Caution: Once an administration PIN has been entered incorrectly the requisite number of times, it becomes blocked and the card/token can never be used again.

The original Admin PIN value of a smart card/token is included in the packaging of the card/token. If you are an administrator you may want to change the Admin PIN value of the cards/tokens you deploy so that only you, the administrator, knows it.

The User PIN

A PIN (*Personal Identification Number*) is a private code. It can be a sequence of numeric or alphanumeric characters or a mix of the two and is used as a type of password. Your User PIN must be verified before you can perform security tasks with the card/token, such as logging on to a workstation, or creating a digital signature.

The user PIN of a smart card/token may be the original PIN value set at the time of manufacture *or* it may be a PIN value assigned by the administrator.

The user PIN should be unique to your card/token and known only to you. It is standard practice, upon reception of a smart card/token, to change the user PIN value so that only you, the user, knows it. Your administrator can even force you to change the PIN value upon first use in the software.

To perform a security operation, you must prove that you know the User PIN. Software that performs a security operation usually displays a window requesting you to enter the PIN before performing the security operation.

- When creating a digital signature, successful PIN validation proves that you are the real card/token holder and enables you to sign with the selected key.
- By using the PIN to log on a network, you prove both that your card/token is valid in the system and that you card/token holder, is physically there.

Caution: Do not allow the User PIN for your card/token to be blocked. If, for example, you forget the user PIN and enter a predetermined number of failed validation attempts (the PIN is entered incorrectly), the card/token becomes blocked and you cannot perform any further security operations with it. If you know the Admin PIN you can unblock your card/token as described in “How to Unblock a User PIN” on page 10. However most companies’ security policy does not allow this, in which case you must ask your Classic Client system administrator to unblock the card/token using the Administrator PIN. If you have the necessary rights, you may be able to unblock your card/token remotely. This operation is described in “How to Remotely Unblock a Connected Smart Card/Token” on page 11. Sometimes card/token technology or software on-board the card/token limits the absolute number of these unblocking operations. For more information, see your card/token technology documentation.

PIN Security Policies

PIN policies are established according to a company’s security policy, but they are also established in relation to the particular type of smart card/token you use and the on-board software the card/token features. For example, some cards/tokens allow a user PIN to be a minimum of 4 characters, and other cards/tokens allow a minimum of 6 characters. Please see your card/token documentation for more information.

Classic Client PIN Management Tool

The Classic Client PIN Management tool allows you to make changes to the PINs associated with a particular smart card/token.

PIN PAD Readers

You can use the Gemalto PIN PAD, “PC Pinpad” with the PIN Tool. PC Pinpad behaves like a normal reader in transparent mode.

PIN Management Tasks

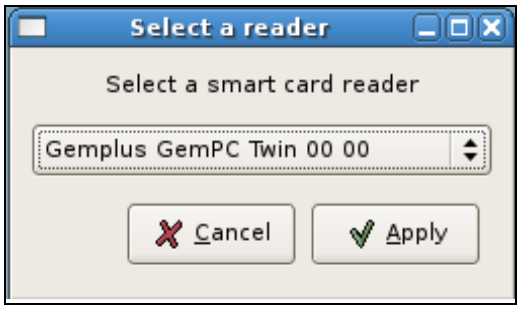
This section describes the tasks that you can perform with the PIN Management Tool.

How to Access the Classic Client PIN Management Tool

To access the PIN Tool:

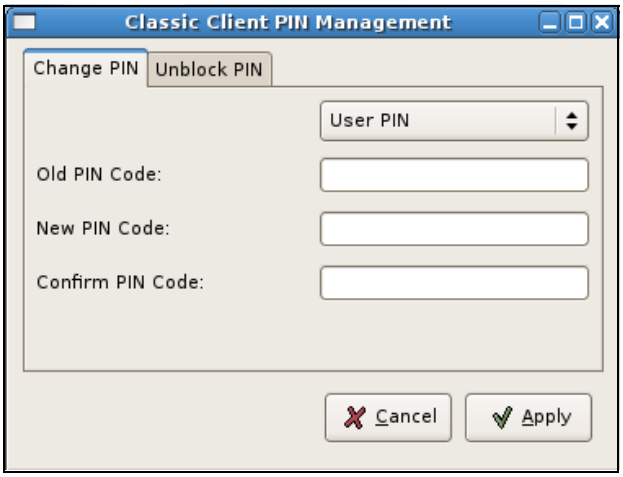
1. Make sure that your smart card/token is connected to your computer.
2. Either browse to /usr/bin/ and double-click **CCChangePinTool** or open a terminal, go to /usr/bin/ and type ./**CCChangePinTool**.
3. When the window shown in “Figure 7” appears, select a smart card reader from the list and click **Apply**.

Figure 7 - Selecting a Smart Card Reader for the PIN Management Tool



This opens the **Classic Client PIN Management** Window as shown in “Figure 8”.

Figure 8 - Classic Client PIN Management - Change PIN Function



How to Change an Administration PIN or User PIN

To change the Admin PIN, you will need to know its current value. This means that normally you will not be able to change an Admin PIN unless you are an administrator.

To change a PIN

1. Connect the smart card/token whose Admin PIN or User PIN you want to change to the PC.
2. Open the PIN Management window as described in “How to Access the Classic Client PIN Management Tool” on page 9.
3. If it is not already selected, click **Change PIN** at the top of the window (see “Figure 8” on page 9).
4. Select the PIN whose value you want to change from the list, **Admin PIN** or **User PIN**.
5. Enter the current value of the PIN in **Old PIN Code**, and the new value in **New PIN Code** and again in **Confirm PIN Code**.
6. Click the **Apply** button at the bottom of the window. A pop-up window appears to confirm a successful PIN change or to display an error message if unsuccessful.

How to Unblock a User PIN

Note: It is not possible to unblock an Admin PIN. If the Admin PIN becomes blocked, the smart card/token can no longer be used.

If you know the Admin PIN for your card/token, you can unblock your User PIN by using the Classic Client PIN Management tool.

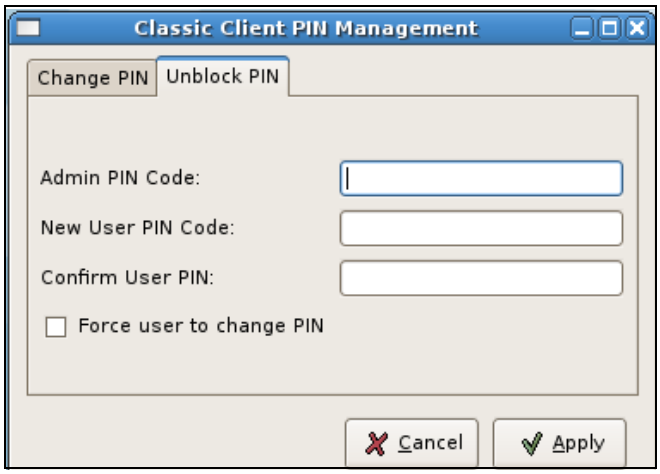
In most cases, if you are not an administrator you will not know the Admin PIN – it depends on your company’s security policy. In such cases, there are two possibilities;

- The administrator must unblock the smart card/token for you. You must return the smart card/token to the administrator so he or she can unblock it on his or her PC.
- If you have been given the necessary rights, you can unblock your PIN remotely as described in “How to Remotely Unblock a Connected Smart Card/Token” on page 11.

To unblock a PIN as an administrator:

1. Connect the blocked smart card/token to your administrator PC.
2. Open the Classic Client PIN Management window as described in “How to Access the Classic Client PIN Management Tool” on page 9.
3. If it is not already selected (in blue), click **Unblock PIN** at the top of the window as shown in “Figure 9” on page 11.

Figure 9 - Classic Client PIN Management - Unblock PIN Function



4 Enter the Admin PIN in **Admin PIN Code**, and the new value for the User PIN in **New User PIN Code** and again in **Confirm User PIN**.

5 For security reasons, Gemalto recommends that you check the box **Force user to change PIN**. This is particularly useful if the user whose PIN is being unblocked is not the administrator (as in most cases).

6 Click the **Apply** button at the bottom of the window. A pop-up window appears to confirm a successful **Unblock PIN** operation or to display an error message if unsuccessful.

How to Remotely Unblock a Connected Smart Card/Token

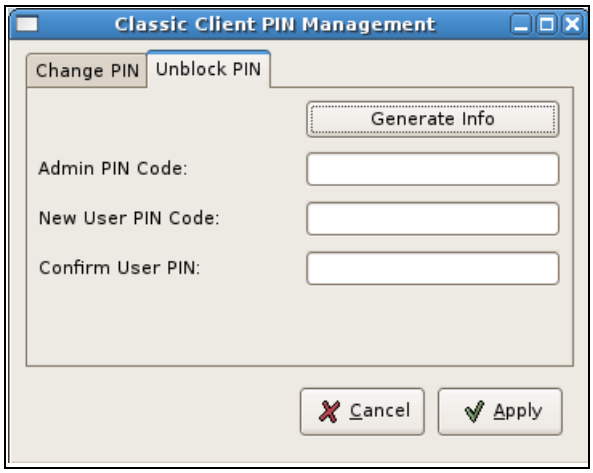
To unblock a smart card/token remotely:

1 Connect the blocked smart card/token to the PC.

2 Open the PIN Management window as described in “How to Access the Classic Client PIN Management Tool” on page 9.

3 If it is not already selected, click **Unblock PIN** as shown in “Figure 10”.

Figure 10 - Classic Client PIN Management - Remote Unblock PIN Function



4 Click **Generate Info**. If this button does not appear in the window, you do not have the rights necessary to unblock your User PIN remotely. A window like the one shown in “Figure 11” on page 12 appears.

Figure 11 - Remote Unblock PIN - Information for Help Desk

5 Call your administrator or Help Desk at the number given in the window, and tell him or her the CSN and Random Number that appear in the window. Click **Close** to close the window.

6 The administrator or Help Desk will provide you with an encrypted value of the Admin PIN. Enter this in **Admin PIN Code.**

7 Enter the new value for your User PIN in **New User PIN Code** and again in **Confirm User PIN**.

8 Click **Apply**. A pop-up window appears to confirm a successful **Unblock PIN** operation or to display an error message if unsuccessful.

3

Tasks

This chapter discusses information related to specific tasks that you will most often be required to carry out when using the Classic Client 5.1 for Linux software and where to find the information about them.

These tasks are:

- “How to Get a Certificate” on this page.
- “How to Use E-mail Securely” on page 14
- “How to View Secure Web Sites” on page 21

Tasks concerning PINs are described in “Chapter 2 - PIN Management”.

How to Get a Certificate

A digital certificate contains information about the user and the user’s public key, and is used to authenticate the user’s identity during secure transactions. The certificate identifying the user must be registered with a certificate authority and this information must be available to both parties. To use smart cards/tokens and certificates together, the user must generate a key pair on his card/token and then get a digital certificate corresponding to the public key and store it on the card/token.

You can get a digital certificate from a Certificate Authority (CA). CA’s are trusted organizations that issue and manage digital certificates, such as Verisign. For more information about CA’s refer to “What is a Certificate Authority?” on page 25.

Tips

When you request a certificate, you will be asked to enter information about yourself such as your name, e-mail address, and the type of certificate you want. The type of information required depends upon what organization is issuing the certificate, and may include the following:

- **Key length value.** The default value is 1024. Classic Client also supports 2048–bit keys, though this ability may be restricted by the card/token used.
- **Cryptographic Module** (sometimes referred to as security device). You will need this if requesting a certificate using Mozilla Firefox or Netscape Navigator.

  You must give the correct name, which is **Classic Smart Card**. If you give a different name, your certificate will be stored on your hard drive instead of your smart card/token.

How to Use E-mail Securely

The following sections explain how to send secure e-mail using Classic Client 5.1 for Linux.

About Secure E-mail

With Classic Client 5.1 for Linux, you can improve e-mail security by using the digital certificate on your smart card/token to:

- Sign your e-mail so that the recipient can verify that the message is really from you and has not been altered.
- Encrypt, or “scramble” a message so that only the intended recipient can read it. This eliminates concerns about intercepted messages and e-mail monitoring.
- Sign or encrypt your message using one e-mail program, while your intended recipient can read it with any other S/MIME-enabled e-mail program.
- Receive signed and encrypted e-mail messages.

Setting up Secure E-mail

You must do the following before you can send secure e-mail:

- **Configure the application to recognize the PKCS#11 security module**
- **Configure security settings**

  Set the security settings for digitally signing and/or encrypting the contents and attachments of outgoing messages.
- **Specify certificates to be used for signing and encryption**

  Choose the digital certificate(s) that you will use to encrypt and digitally sign your e-mails. You can use the same certificate for both operations or two different ones. These certificates are associated with your e-mail account.
- **Send yourself a digitally-signed e-mail**

  When you send a signed e-mail, you sign it with the private key. The recipient receives the corresponding public key with the mail which he or she uses to decipher your mail.

  Before you can send e-mails to anybody else, you need to send a signed message to yourself in order for Thunderbird or Icedove to store your public key.

  Then you can send your public key to other people, for example by sending them a signed message. Once they have your public key, they can use it to encrypt mails they send to you (which you decipher using your private key).

The following sections describe how to perform the above operations using the Mozilla Thunderbird (or Icedove) e-mail programs. The dialog boxes shown may differ slightly from your own software, depending on what version you are using.

Working with Mozilla Thunderbird or Icedove.

The following sections explain how to set up and send secure e-mail with Mozilla's Thunderbird or the Icedove e-mail programs. Icedove is the Debian version of Thunderbird and the two applications are identical except for their appearance.

There are three stages:

1 Configure Thunderbird to recognize the Security Module, described in the following section.

2 Configure the security settings and specify the certificates to use for signing and encryption, described on page 16.

3 Send a digitally signed e-mail to yourself in order to store your public key in Thunderbird/Icedove, described on page 19.

Configure Thunderbird/Icedove to Recognize the Security Module

You only need to do this once.

To configure Mozilla Thunderbird or Icedove

1 Make sure your smart card/token is connected.

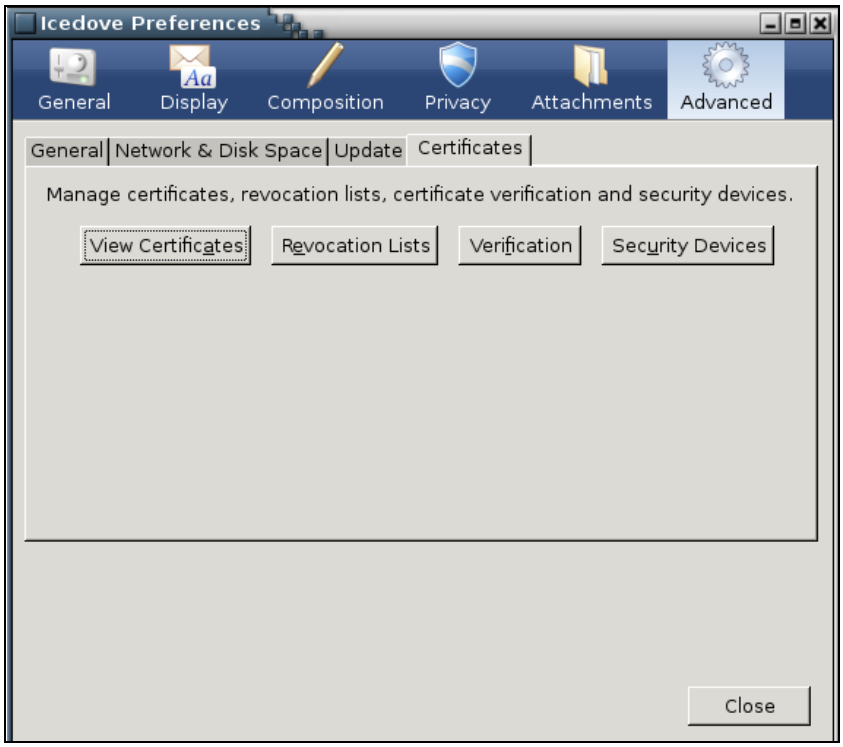
2 Start **Mozilla Thunderbird** or **Icedove**.

3 Enter your password if you are prompted for it and click on **OK**.

4 From the **Edit** menu, choose **Preferences**.

5 In the dialog box that opens, click the **Advanced** icon, then the **Certificates** tab to display the settings as shown in "Figure 12".

Figure 12 - Icedove - Certificates Tab



6 The rest of the procedure is the same as that described for Firefox. Continue from step 3 on page 4.

This new module will be used with all e-mail you send with Thunderbird or Icedove.

Configuring Settings and Specifying Certificates

You only need to do this the first time you use your card/token to sign or encrypt an e-mail.

Note: Although selecting the certificates is mandatory, this does not mean that you must sign and encrypt e-mails.

1 Make sure your smart card/token is connected.

2 Start **Mozilla Thunderbird** or **Icedove**.

3 Enter your password if you are prompted for it.

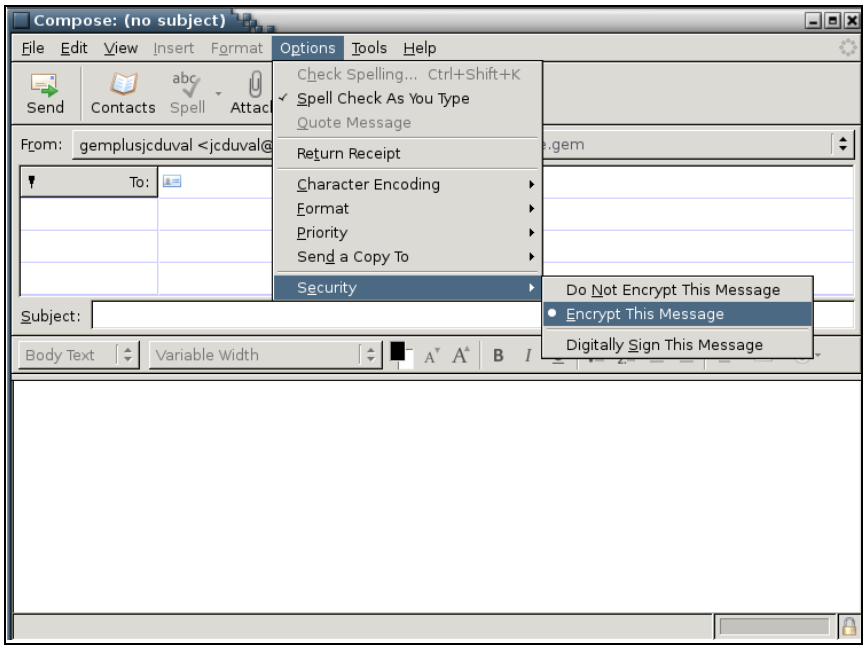
4 In **Thunderbird** or **Icedove**, click the **Write** icon 

.

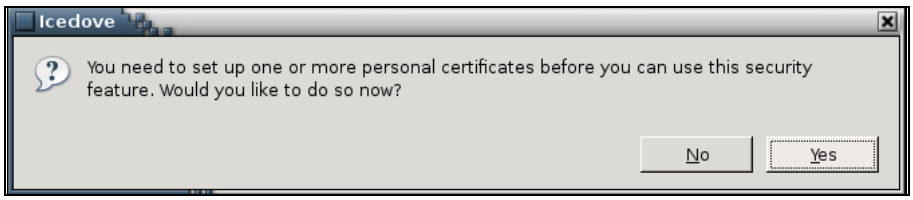
This opens the **Compose** window.

5 In the **Compose** window's **Options** menu, choose **Security** > **Encrypt This Message** as shown in "Figure 13".

Figure 13 - Icedove – Encrypt This Message

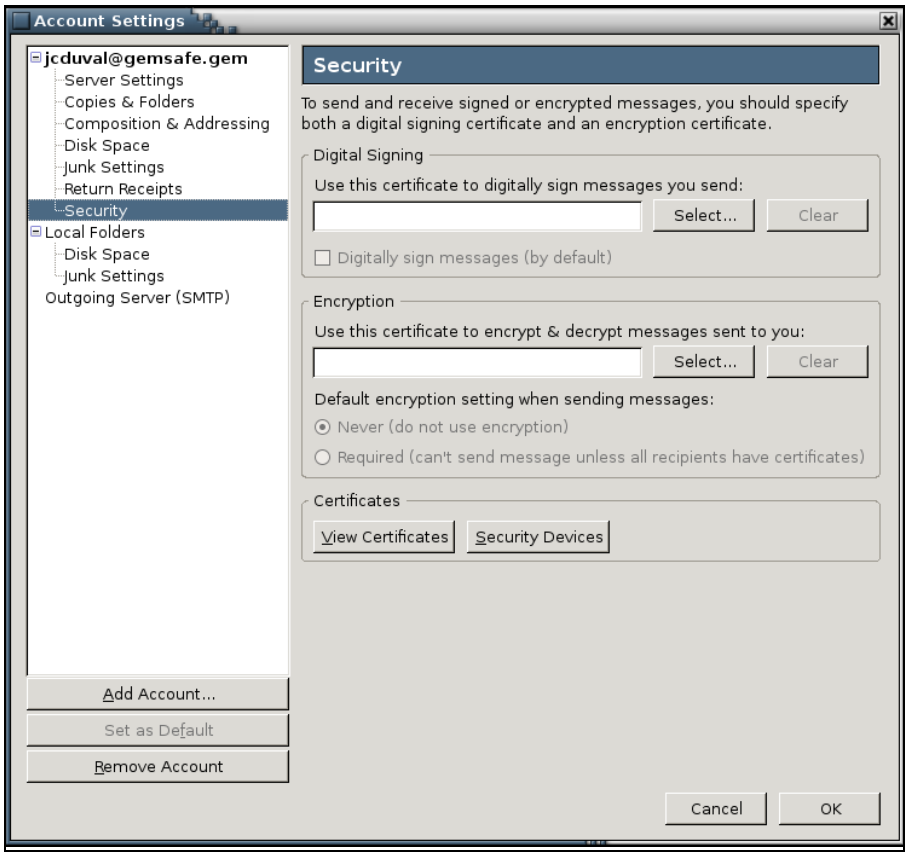


As the certificates in the card/token are not yet set up, the following message appears:



6 Click **Yes**. This opens the security account settings window for your e-mail account as shown in "Figure 14" on page 17.

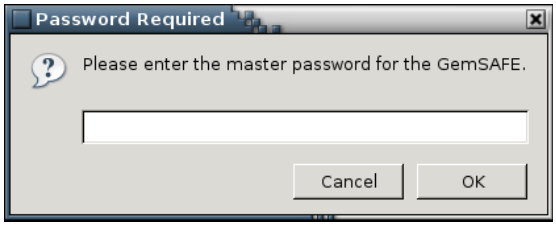
Figure 14 - Icedove – Security Account Settings



7 In **Digital Signing**, click **Select** and choose the certificate you want to use from the list that appears.

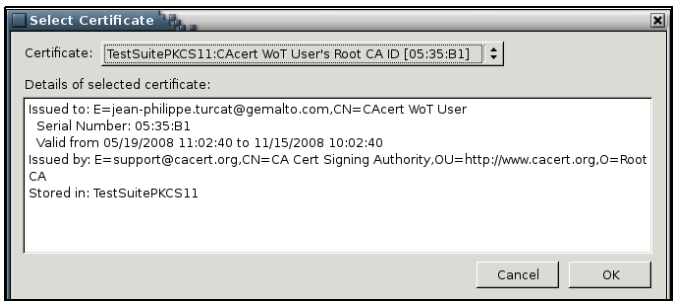
Note: You may be prompted to enter a “master password” as shown in “Figure 15”. If so, enter the PIN for the card and click **OK**.

Figure 15 - Icedove - Enter Password



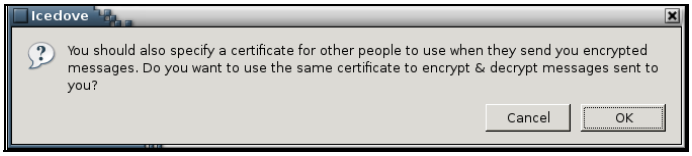
The details of the selected certificate appear, as shown in “Figure 16”.

Figure 16 - Icedove - Details of Selected Certificate



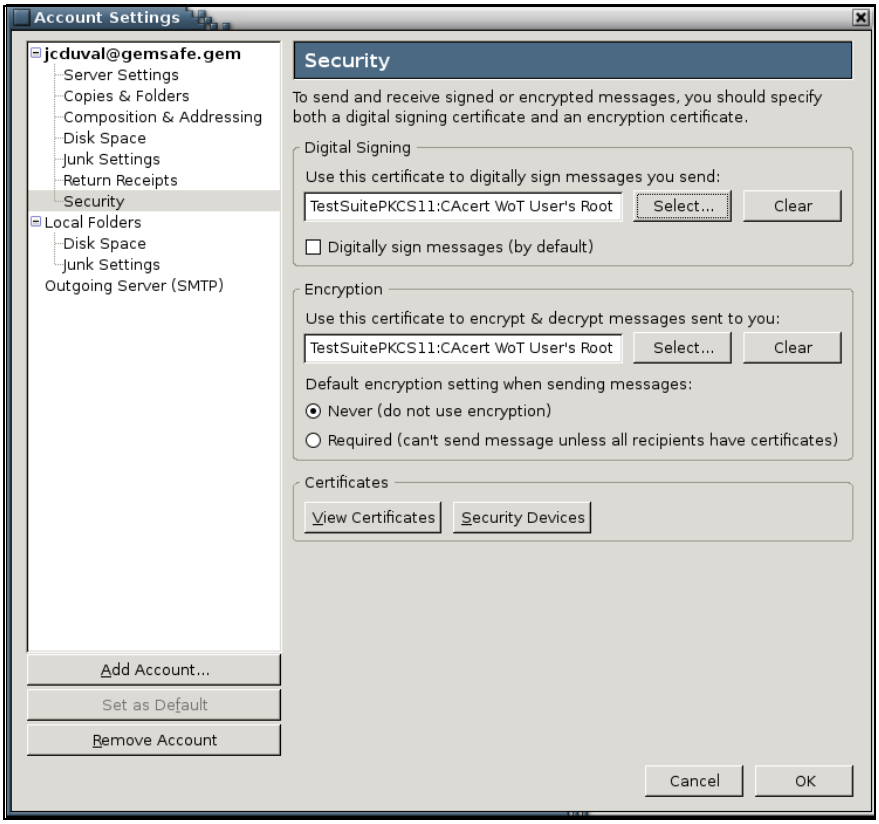
8 Click **OK**. The following message appears:

Figure 17 - Icedove – "Use Same Certificate" Message



9 If you want to use the same certificate to encrypt and decrypt messages, click **OK**. This selects the certificate for you in the **Encryption** panel as shown in "Figure 18". Otherwise click **Cancel**.

Figure 18 - Icedove – Security Account Settings (2)



10 If you want all of your e-mails to be digitally signed by default, check the box **Digitally sign messages (by default)**.

11 In **Encryption**, if you chose not to use the same certificate as the one used for digital signing, click **Select** and choose the certificate from the list that appears. A message similar to the one in "Figure 17" on page 18 appears, but this time asking if you want to use the Encryption certificate for digital signing. This is just in case you select your encryption certificate before you select your digital signature certificate.

12 In **Default encryption setting when sending messages**, choose one of the option buttons **Never** or **Required**.

13 Click **OK** to close the **Security Account Settings** window.

Note: If you want to modify the account settings at any point, open the **Account Settings** window from the **Tools** menu by choosing **Account Settings**. This can be done either from the **Compose** window or directly in Thunderbird or Icedove.

Sending Digitally Signed E-mail with Mozilla Thunderbird or Icedove

To send a signed e-mail to yourself with Mozilla Thunderbird or Icedove

1 Make sure your smart card/token is connected.

2 Start **Mozilla Thunderbird** or **Icedove**.

3 Enter your password if you are prompted for it.

4 In **Thunderbird** or **Icedove**, click the **Write** icon 

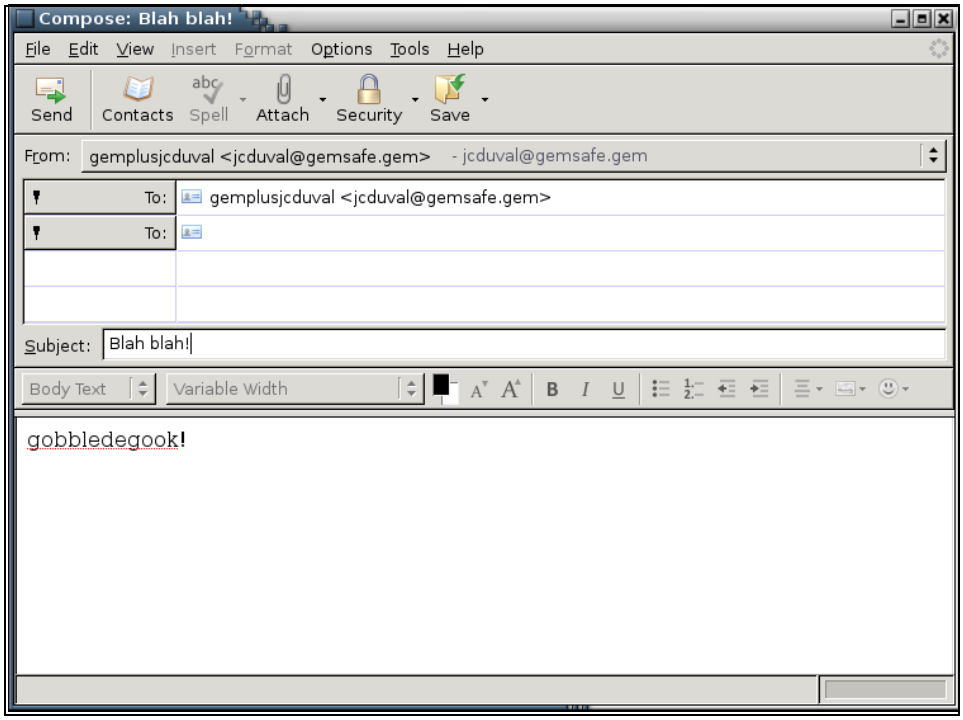
.

This opens the **Compose** window.

5 In the **Compose** window, write a short message ***addressed to yourself***.

Be sure to include a subject heading.

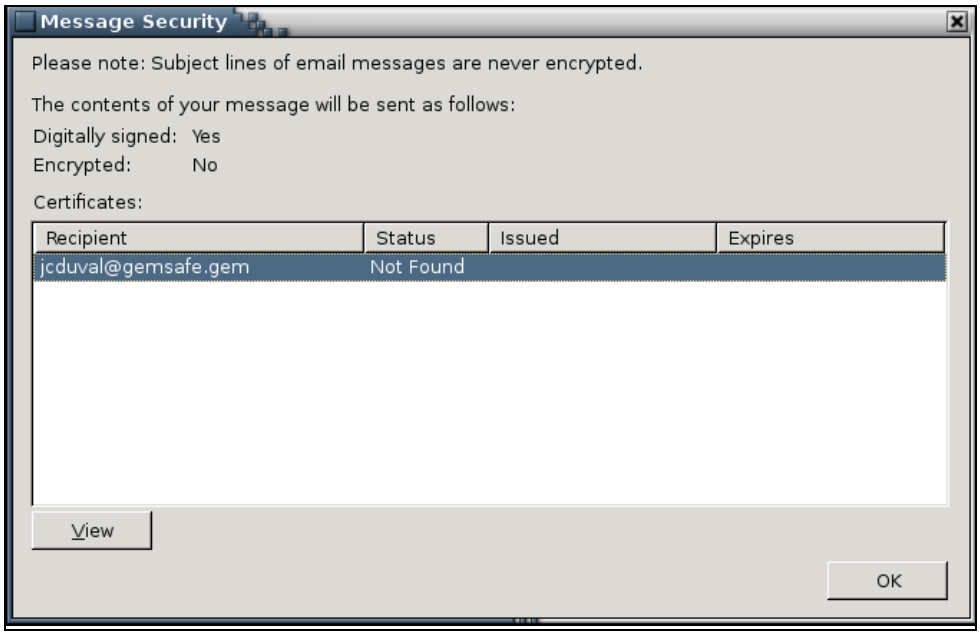
Figure 19 - Icedove New Msg Composition Window



6 From the **Options** menu in the **Compose** window choose **Security > Digitally Sign this Message** in order to sign the message.

Note: You can check the security settings for your message in the **Compose** window by choosing **View** > **Message Security Info**. This displays the **Message Security Info** window as shown in “Figure 20” on page 20.

Figure 20 - Icedove Message Security Info Window



You can display details about the certificate by clicking **View**.

7 Click **OK** to close the **Message Security** window.

8 Back in the **Compose** window, click **Send**.

If you are prompted for a master password for your security module, as shown in “Figure 15” on page 17, then enter the User PIN for your smart card/token.

9 Open the message you sent yourself from in your inbox.

Notice the  icon showing you that the message has been signed.

You have successfully sent yourself a digitally signed e-mail.

Now that Thunderbird or Icedove recognizes your public key, you can send signed messages to other people, thus sending them your public key.

Sending Encrypted E-mail with Mozilla Thunderbird or Icedove

Once you have configured your e-mail account in **Mozilla Thunderbird** or Icedove, you can retrieve a person’s public key when he or she sends a signed message to you. When you send e-mail to that person, you use his or her public key to encrypt the e-mail. This is done automatically by Thunderbird or Icedove; you just need to specify the recipient(s) of the mail. Since no one except the person who has the private key can decrypt it, the e-mail is secure.

To send an encrypted e-mail:

Follow the same steps as “To send a signed e-mail to yourself with Mozilla Thunderbird or Icedove” on page 19, except in the **Compose** window, choose **Encrypt this message** from the **Options** menu.

How to View Secure Web Sites

Communicating and conducting business on the Web is quickly becoming the most convenient, effective means of transaction. Therefore, Web sites must be secure to protect the corporation, the individual and the information exchanged.

With your Classic Client smart card/token, you can browse secure Web sites knowing that your private key and digital certificate are safely stored on your smart card/token instead of your hard drive, where they might be susceptible to unauthorized access.

Note: All secure Web site addresses must begin with https://. Browsers display a lock icon at the bottom of the browser window indicating that the site is secure. A closed lock indicates that you are operating in secure mode. You may need to configure your organization's network to allow secure browsing.

When you connect to a secure Web site, your certificate must be specified in your browser so that you can authenticate yourself to the Web server. For example, when you bank online, your bank must be sure that you are the correct person to get account information. Your certificate confirms your identity to the online bank.

The following sections explain how to check that your certificates are correctly registered in your browsers when authenticating with secure web sites using Mozilla Firefox (or the Debian equivalent Iceweasel) and Netscape.

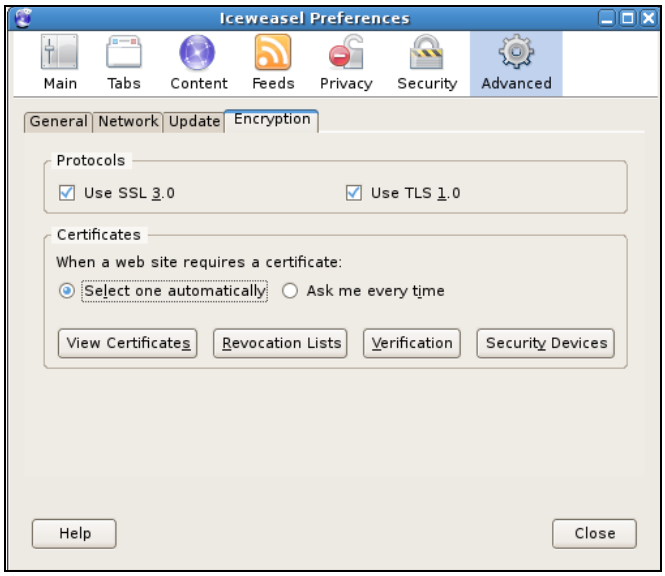
Choosing a Certificate Used to View Web Sites

To authenticate using either the Mozilla Firefox or Netscape Navigator browser, your certificate must be registered in the browser. This section describes how to check that a certificate is registered and also how to tell the browser whether it should select the certificate itself, or ask you.

To check certificates registered in Mozilla Firefox or Netscape Navigator:

1. Make sure your card/token is connected.
2. Open the browser (Mozilla Firefox or Netscape Navigator).
3. From the **Edit** menu (or **Navigator** menu if using Netscape) choose **Preferences**.
4. Click the **Advanced** icon, then the **Encryption** tab as shown in "Figure 21".

Figure 21 - Mozilla Firefox Options Dialog

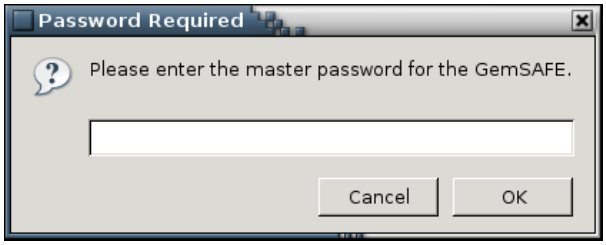


5 In **Certificates**, choose one of the options for the action to take when a web site requires a certificate:

- Select one automatically
- Ask me every time

6 To display the certificates that are on your card/token, click **View Certificates**. You will be prompted for a password as shown in “Figure 22”.

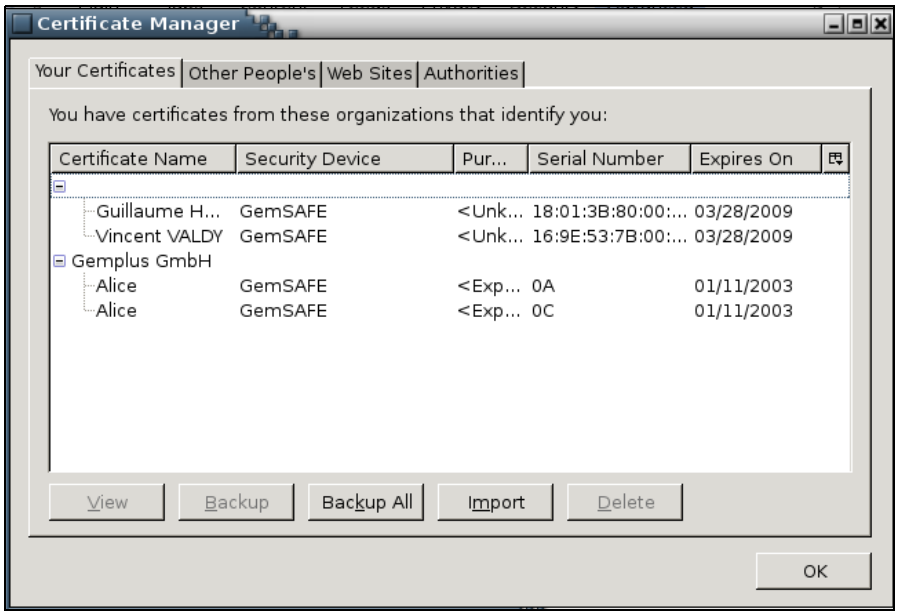
Figure 22 - Password Required



7 Enter the User PIN for your card/token.

The **Certificate Manage**r window appears.

Figure 23 - Certificate Manager Window



8 Under **Your Certificates** appears the certificates that are stored on the card/token. To display the properties of a particular certificate, select it and click **View**.

A

Security Basics

This chapter introduces you to the IT security standards integral to Classic Client.

Cryptography

Communicating and conducting business electronically is quickly becoming the most convenient, effective means of transaction. An essential condition for the continued growth toward an electronic market is security. The identities of both corporations and individuals must be authentic. The integrity and privacy of information must be guaranteed.

Encryption/decryption enables you to send and receive secure e-mail and documents to protect confidential or private information. You can use the signature function to sign your messages. By signing messages, you can prove to the recipient that you are who you claim to be.

The IT industry uses cryptography to render information secret and known only by authorized entities.

There are two types of cryptography:

- Secret Key Cryptography.
- Public Key Cryptography

Both cryptographic systems use *keys* to digitally sign or encrypt/decrypt data. A key is a value in electronic format used to perform cryptographic functions on electronic data.

The differences between secret key and public key cryptography include:

- Key management.
- Complexity of the key structure.

Key management is central to having a successful crypto system. If keys are not managed in a secure environment, the overall security of the crypto system is at risk. Keys must also be convenient to use.

The complexity of a key length is determined by the degree of mathematical properties applied to the random numbers that comprise the key.

Secret Key Cryptography

Secret key cryptography is the traditional crypto system, which remains in widespread use even today. Secret key cryptography uses a single secret key to digitally sign or encrypt/decrypt electronic data. The most widely used secret key crypto systems are DES and RC2 (also known as symmetric key cryptography).

The sender and receiver must use the same secret key for the session in which secure information is exchanged. The sender uses the secret key to encrypt the message; the receiver uses the same secret key to decrypt the message.

The primary advantage of secret key cryptography is the speed at which data can be encrypted/decrypted.

The primary weakness of secret key cryptography regards key management. Because sender and receiver must share knowledge of the secret key, there must be a transfer of the secret key at some point. Introducing a third party (such as a telephone line or courier) to deliver the secret key to the receiver presents a security risk.

Secret keys are included in the cryptographic functionality of Mozilla e-mail and browser products.

Public Key Cryptography

Public key cryptography was introduced in 1976 and is the most advanced, secure crypto system for digitally signing and encrypting/decrypting electronic data. Public key cryptography refers to a crypto system that uses key pairs. The most popular and widely-used public key crypto system uses the RSA key pair.

A key pair is a matched set of keys used to digitally sign or encrypt/decrypt electronic data. RSA key pairs, like secret keys, are strings of random numbers. However, RSA keys are not only significantly longer than secret keys, they also possess complex mathematical properties.

A single user *owns* an RSA key pair. One key is private, while the other key is public. The private key remains private and accessible only to the owner of the key pair. The public key is made available by the owner to public users. The public key is used to encrypt data. The private key is used to decrypt data.

The strengths of using an RSA key pair is that the need for sender and receiver to share knowledge of the single secret key used in secret key crypto systems is eliminated.

Classic Client takes advantage of the speed the secret key offers and the robust security and convenience of the RSA key pair. When you use Classic Client to send secure e-mail, the actual message data is encrypted using a secret key. The secret key is then encrypted using the public key of the intended recipient. Only the recipient's private key can decrypt the secret key. Only the secret key can decrypt the message data.

Classic Client offers the most advanced digital security at the greatest speed and convenience.

What is a digital certificate?

A digital certificate is an electronic document that serves as your digital passport. Your digital certificate stores your public key and other personal information about you and the certificate.

The most widely accepted standard for digital certificates is defined by *International Telecommunications Union standard ITU-T X.509*. Version three is the most current version of X.509.

The X.509v3 certificate includes the following data:

- Version.
- Serial number.
- Signature algorithm ID.
- Issuer name.
- Expiration Date.
- User name.
- User public key information.
- Issuer unique identifier.
- User unique identifier.
- Extensions.
- Signature on the above fields.

As a convenience to recipients, it is standard practice to attach your digital certificate to every secure e-mail that you send. The recipient uses your public key, included in your digital certificate, to encrypt e-mail addressed to you. If you do not attach your digital certificate to outgoing e-mails, recipients must retrieve your public key from a public directory if they want to reply to you with an encrypted e-mail.

What is a Certificate Authority?

Certificate Authorities (CAs) are trusted third parties that issue digital certificates. CAs vouch for the identity of the individual or enterprise to whom they are issuing a certificate. CAs provide a transfer of trust from CA to the individual or enterprise. When you trust the CA certificate, you can transfer that trust to all certificates published by that CA.

When you obtain your digital certificate, you provide the CA with your public key and any personal information requested by the CA. The CA verifies your personal information and the integrity of your public key. After the verification process, the CA signs your public key, stores appropriate personal information and your public key on the digital certificate, and issues your digital certificate to you.

CAs issue certificates with varying levels of identification requirements. CA policies and the level of identification of the digital certificate determine the method and requirements for proving your identity to the CA. The most simple digital certificate only requires your e-mail address and name. However, some CAs require a driver's license, notarized certificate request form, or any other personal documentation attesting to your identity. Some CAs may even go as far as requiring biometric data such as fingerprints.

The CA public key must be widely available so that users can validate the authenticity of all certificates published by this CA.

What is a digital signature?

A digital signature is a piece of information created using message data and the owner's private key. Digital signatures provide message authentication, non-repudiation of origin, and data integrity.

Digital signatures are created by mathematical, or *hash*, and private signing functions. The one-way hash function produces a message digest, a condensed version of the original message text. The message digest is encrypted using the sender's private key, turning it into a digital signature.

The digital signature can only be decrypted using the public key of the same sender. The recipient of the data decrypts the digital signature and compares the result with a message digest, recalculated from the original message text. If the two are identical, the message was not manipulated, thus is authentic.

What is S/MIME?

Secure/Multipurpose Internet Mail Extensions (S/MIME) is an open protocol standard, that provides encryption and digital signature functionality to Internet e-mail. S/MIME uses public key cryptography standards to define e-mail security services.

S/MIME enables you to encrypt and digitally sign Internet e-mail using Web messaging applications such as Mozilla Thunderbird. S/MIME also enables you to authenticate incoming messages.

S/MIME provides the following security functions:

- **Sender Authentication** to verify the sender's identity. By reading the sender's digital signature, the recipient can see who signed the message and view the certificate for additional details.
- **Message Encryption** to ensure that your messages remain private. Mozilla Thunderbird supports domestic and export-level public key and secret key encryption.
- **Data Integrity** to guard against unauthorized manipulation of messages. S/MIME uses a secure hashing function to detect message tampering.
- **Inter-operability** to work with other S/MIME-compliant software.

What is SSL?

Secure Sockets Layer (SSL), developed by Netscape Communications, is a standard security protocol that provides security and privacy on the Web. The protocol allows client/server applications to communicate securely. SSL uses both public and secret key cryptography.

The SSL protocol is application independent, which enables higher-level protocols such as Hyper Text Transfer Protocol (HTTP) to be layered on top of it transparently. Therefore, the client can negotiate encryption and authentication with the server before data is exchanged by the higher-level application.

The SSL Handshake Protocol process includes two phases:

- **Server Authentication** in which the client requests the server's certificate. In response, the server returns its digital certificate and signature to the client. The server certificate provides the server's public key. The signature proves that the server currently has the private key corresponding to the certificate.
- **Client Authentication** (optional) in which the server requests the client's certificate. In response, the client sends the digital certificate and signature to the server. If the SSL Server requests it, the client is prompted to enter a PIN to visit a secure Web site.

The SSL process is repeated for every secure session you attempt to establish unless you specify a permanent session. The SSL process will not proceed if the Web server's certificate is expired.

Note: In some instances, the SSL Handshake takes place between the Web server and the browser and does not require the client's certificate.

SSL provides the following security functions:

- **Data Encryption** to ensure data security and privacy. Both public key and secret key encryption are used to achieve maximum security. All traffic between an SSL server and SSL client is encrypted using both public key and secret key algorithms. Encryption thwarts the capture and decryption of TCP/IP sessions.
- **Mutual Authentication** to verify the identities of the server and client. Identities are digital certificates. The entity presenting the certificate must digitally sign the data to prove ownership of the certificate. The combination of the certificate and signature authenticates the entity.
- **Data Integrity** to ensure that SSL session data is not manipulated en route. SSL uses hash functions to provide the integrity service.

What is Classic Client?

Classic Client is a smart card–based solution designed to secure e–mail communications and Internet transactions. Classic Client smart cards/tokens support encryption/decryption and signature functions.

Classic Client and a smart card/token provide the following advantages:

- Your private key is never removed from your smart card/token.
- The smart card/token is hardware-based security.
- The PIN code protects key use.
- Classic Client is portable and convenient.

The encryption/decryption function enables you to send and receive secure e-mail to protect confidential or private information. You can use the signature function to sign your messages. By signing messages, you can prove to the recipient that you are who you claim to be.

Classic Client combines the privacy, integrity, and authentication functionality provided by cryptographic algorithms with the simplicity, portability, and convenience of smart cards/tokens. Your private key, digital certificate, and other personal information are securely stored on your Classic Client smart card/token to prevent fraudulent use of your electronic identity.

The latest industry standards such as SSL3 (for Web access) and S/MIME (for e–mail) enable inter–operability of security services between any browser interface and any Web server. However, the security hole in SSL3 and S/MIME is the management of your private key and digital certificate. Without Classic Client, your private key and digital certificate are stored on your hard drive, which makes them susceptible to unauthorized access and fraudulent use. Without Classic Client, your electronic identity is at risk.

Classic Client provides double-barreled security! Classic Client, you get the hardware-based security inherent in smart cards/tokens and software-based encryption security, as well as the added advantage of individual PIN codes. Hardware-based security is a principal security advantage. It is significantly more secure than software-only solutions. Without the possession of your smart card/token and knowledge of your PIN code, no one can use your identity.

Classic Client is your electronic passport to the digital world.

What is a Smart Card/Token?

A smart card is the size of a conventional credit card. But unlike the credit card, which has a magnetic stripe, the smart card has a silicon microprocessor chip to store and process electronic data and applications. The advantage of the smart card is **security**.

Gemalto manufactures various types of smart cards. Contact smart cards use a microprocessor chip to store and process data. They must be inserted into a smart card reader. Contactless smart cards use a microprocessor chip and antenna to store and process data.

Smart cards can also be embedded in tokens such as USB devices, that you can plug directly into a PC.

Smart cards/tokens provide the most sophisticated security available on the market.

What is the Classic Client Smart Card/Token?

Your Classic Client smart card/token stores your private key and digital certificate. In the past, your only option was to store your private key on your local hard drive, rendering it susceptible to theft and fraudulent use. With Classic Client, your electronic identity is secure. You must have both the smart card/token and PIN code to use the smart card/token.

The Classic Client smart card/token is tamper resistant. The structure and operating system of the smart card/token make it practically impossible to penetrate, probe, or pilfer smart card/token data.

Perhaps the most convenient aspect of the Classic Client smart card/token is portability. With Classic Client, you can carry your electronic passport with you at all times and use it on any Classic Client–equipped computer in the world.

The Classic Client smart card/token has a robust and flexible design. These features offer greater freedom and enhanced security.

On-board Key Generation

The Classic Client smart card/token offers on-board key generation. With this feature, every time you enroll a new certificate on your smart card/token, a new key pair is generated on your smart card/token. In other words, you are not limited to using the same key pair for every certificate that you enroll.

One significant advantage of onboard key generation is the ability to monitor and control the life span of your RSA key pairs and that the generated key pair is unique.

Increased Certificate Storage

You can store up to six key pairs and multiple digital certificates on your Classic Client smart card/token, depending upon the size of your certificates and space available on your smart card/token. This feature provides the convenience of using up to eight digital certificates for whatever purposes you want; for example, you can use certificates with varying degrees of encryption (from 1024–bit to 2048–bit RSA key pairs) to communicate securely with contacts in various parts of the world.

Another reason for obtaining more than one digital certificate is the level of certification that the Certificate Authority (CA) requires. You may want to obtain and use a digital certificate from a CA that requires stringent identity certification if you are using the certificate for sensitive business communications or financial transactions. However, if you want to encrypt/sign data for personal communications, you may decide that a certificate from a CA that requires minimal identity certification meets your needs.

The costs of obtaining a digital certificate from a CA are somewhat based on the degree of identity certification the CA requires.

B

End User License Agreement

IMPORTANT-READ CAREFULLY: This End-User License Agreement for Gemplus software (“EULA”) is a legal and binding agreement between you and the subsidiary or affiliate of Gemalto NV. (“Gemplus”) that distributed this version of the Software (as defined below) under this EULA (“Gemplus”). “You” are a person or legal entity wishing to use the Software. This EULA governs your use of all of the Software distributed or delivered hereunder. “Software” means all computer software, associated media, any printed materials and any accompanying “online” or electronic information provided to you hereunder. By downloading, installing, copying, breaking any seal on, or otherwise using the Software, you acknowledge that you have read this EULA and agree to be bound by its terms. If you do not agree to the terms and provisions of this EULA, do not download, install, copy, or otherwise use the Software; if you have already broken the seal or paid for this Software and do not agree to the terms and conditions of the EULA, please return the Software and any accompanying items to Gemplus within thirty (30) days of the date of purchase for a full refund of amounts paid. If these terms are considered an offer, acceptance is expressly limited to these terms.

1 Ownership. The Software is owned by Gemplus or its third party suppliers and is licensed (and not sold) to you. Gemplus's third party suppliers or distributors may assert and protect any of their rights (and with Gemplus's permission, Gemplus's rights) in connection with this EULA.

2 Grant of License. Subject to the terms of this EULA, Gemplus grants you the world-wide, non-exclusive, non-sublicensable, non-transferable (except as set forth in this EULA), license to use one copy of the Software, in object code format, solely for your internal use.

a) You must reproduce on any copy all copyright notices and any other ownership, confidentiality or proprietary legends that are on the original copy of the Software and accompanying documentation, and you may make only one copy of the Software solely for backup or archival purposes, provided that such backup copy is not installed on any computer.

b) You may not reverse-engineer, decompile, or disassemble the Software, or otherwise reduce the code of the Software to a human perceivable form, except and only to the extent that the restriction of such activity is prohibited by applicable law, and in such event you shall provide Gemplus prompt notification of such activities. You may not alter or remove any of Gemplus's trademarks affixed to or otherwise contained on or within the Software.

c) You may not market, distribute, transfer copies of the Software to others or electronically transfer the Software from one computer to another over a network except for Software installations permitted under Section 2 of this EULA. You may not rent, lease, or lend the Software. You may not modify,

adapt or translate the Software or create derivative works based on the Software.

d) All rights not expressly granted to you in this EULA are reserved by Gemplus and its suppliers. No rights are granted by implication or otherwise.

3 Termination. This EULA may be terminated by Gemplus upon notice and without further action upon the breach of any of your obligations or the license rights granted to you under this EULA. Upon termination, all use of the Software by you must cease and all rights granted to you under this EULA are terminated. Upon termination you hereby agree to return to Gemplus or to destroy all copies of the Software in your possession or control within fourteen (14) days of such termination and certify the same in an affidavit to Gemplus upon request.

This remedy is in addition to any other remedies available to Gemplus. Provisions contained in this EULA that expressly or by their sense and context are intended to survive the expiration or termination of this Agreement shall so survive the expiration or termination including without limitation Sections 1, 3, 4, and 7 through 10 hereof.

4 Proprietary Rights. All rights, title, and proprietary rights in and to the Software (including, but not limited to, any patents, trade secrets, trademarks, copyrights, images, photographs, animations, video, audio, music, text, software code and “applets” incorporated into the Software) are owned by Gemplus or its suppliers. The Software is protected by copyright laws, international treaty provisions, and other laws. An act in violation of this EULA may also be a crime punishable by fine or imprisonment under applicable law. You understand that Gemplus may update or revise the Software in its sole discretion, but has no obligation to furnish any Software updates or revisions to you. If you upgrade the Software to a higher-numbered or later version of the Software (e.g., from Program 3.x to Program 4.x) or to a comparable Gemplus software product including versions for different operating systems (“Replacement Software”), unless otherwise indicated in any end user license agreement accompanying such Replacement Software, this EULA is terminated to the extent it covers the replaced software and your rights in the Replacement Software will be governed by the end user license terms applicable to that Replacement Software. If any Replacement Software, or other Software of Gemplus, is distributed to you without a separate end user license agreement, this EULA shall govern all your rights and obligations therein.

5 Compliance with Laws. In the performance of the obligations under this Agreement, you shall at all times comply with the laws, regulations, and orders in effect and applicable to their performance hereunder. Without limiting the generality of the preceding sentence, you shall comply with applicable U.S. Foreign Corrupt Practices Act provisions (regarding, among other things, payments to government officials) and all applicable U.S. export laws and restrictions and regulations of the U.S. Department of Commerce, the Department of Treasury Office of Foreign Assets Control (“OFAC”), or other U.S. or non-U.S. agency or authority, and not export, or allow the export or re-export of any Gemplus product (or any product incorporating such Gemplus product) in violation of any such restrictions, laws or regulations. You shall obtain and bear all expenses relating to any necessary licenses and/or exemptions with respect to the export from the U.S. of all Gemplus products to any location and shall demonstrate to Gemplus compliance with all applicable laws and regulations prior to delivery thereof by Gemplus.

6 U.S. Government Restricted Rights. If a user of the Software is an agency, department, or other entity of the United States government (the “Government”), the use, duplication, reproduction, release, modification, disclosure, or transfer of such Software, or of any related documentation of any kind, including technical data, is restricted in accordance with Federal Acquisition Regulation (“FAR”) 12.212, Defense Federal Acquisition Regulation Supplement (“DFARS”) 227.7202,

subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subparagraphs (c)(1) and (2) of the Commercial Computer Software-Restricted Rights at 48 CFR 52.227-19, as applicable. The Software is commercial computer software and commercial computer software documentation. The use of this Software by the Government is further restricted in accordance with the terms of this EULA.

7 Limited Warranty, Disclaimer of Implied Warranties & Duties, Limited Warranty Remedy.

a) Disclaimer of Implied Warranties and Duties. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW AND EXCEPT AS SET FORTH IN SECTION 7.b, GEMPLUS, ITS SUPPLIERS, AND DISTRIBUTORS PROVIDE THE SOFTWARE AND ANY (IF ANY) SUPPORT SERVICES RELATED TO THE SOFTWARE (“SUPPORT SERVICES”) WITHOUT ANY EXPRESS WARRANTY OR INDEMNITY, AND THE SOFTWARE AND SUPPORT SERVICES ARE PROVIDED “AS IS” AND “WITH ALL FAULTS”. GEMPLUS HEREBY DISCLAIMS ALL IMPLIED INDEMNITIES AND WARRANTIES, INCLUDING, BUT NOT LIMITED TO, ANY (IF ANY) IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, TITLE, OR THAT THE OPERATION OF THE SOFTWARE WILL BE ACCURATE, VIRUS-FREE, OR WILL CORRESPOND TO ANY DOCUMENTATION. SOME JURISDICTIONS DO NOT ALLOW EXCLUSIONS OF AN IMPLIED WARRANTY, SO THE ABOVE EXCLUSION MAY NOT FULLY APPLY TO YOU, AND YOU MAY HAVE OTHER RIGHTS THAT VARY BETWEEN JURISDICTIONS.

b) Limited Warranty Remedy. Gemplus warrants for a period of 90 days from the date of purchase that the medium on which the Software is provided will be free from defects in material and workmanship. This limited warranty covers only such defects. This limited warranty does not cover any other defects or problems of any kind.

Gemplus's sole obligation, and your exclusive remedy, under the limited warranty set forth in this Section 7.b shall be, at the sole discretion of Gemplus, to supply you with a corrected or replacement copy of the Software or a refund of all amounts received by Gemplus from you for the subject Software provided under this EULA. Any replacement Software will be warranted as set forth above for the remainder of the original warranty period or thirty (30) days from your receipt of such replacement Software, whichever is longer. This warranty gives you specific legal rights, and you may have other rights that vary between jurisdictions.

c) Gemplus does not warrant that the Software will be resistant to all possible efforts to defeat or disable its functions, including its security mechanisms, and Gemplus shall not incur, and disclaims, any liability in this respect. Security mechanisms' resistance and strength necessarily evolve according to the applicable state of the art in security and with reference to the emergence of new technologies and methods developed in efforts to defeat or disable such mechanisms. To the maximum extent permissible by law, Gemplus shall not be held liable for any third party actions and in particular in case of any successful effort to defeat or disable security functions of the Software, or computing devices and equipment using, accessing or incorporating the Software.

d) NO ORAL OR WRITTEN INFORMATION OR ADVICE GIVEN BY GEMPLUS, ITS DEALERS, DISTRIBUTORS, AGENTS OR EMPLOYEES CREATES A WARRANTY AND YOU MAY NOT RELY ON ANY SUCH INFORMATION OR ADVICE.

8 Exclusion of Incidental, Consequential and Certain Other Damages. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT SHALL GEMPLUS OR ITS SUPPLIERS OR DISTRIBUTORS BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT, OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING, BUT NOT LIMITED TO, DAMAGES FOR LOSS OF PROFITS, REVENUES OR CONFIDENTIAL OR OTHER INFORMATION, FOR BUSINESS INTERRUPTION, OR LOSS OF PRIVACY), ARISING OUT OF OR IN ANY WAY RELATED TO THE USE OF OR INABILITY TO USE THE SOFTWARE PRODUCT OR OTHERWISE UNDER OR IN CONNECTION WITH ANY PROVISION OF THIS EULA, IN TORT (INCLUDING NEGLIGENCE), STRICT LIABILITY, BREACH OF CONTRACT OR UNDER ANY OTHER LEGAL THEORY, AND EVEN IF GEMPLUS OR ANY SUPPLIER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE LIMITATIONS MAY NOT APPLY TO YOU.

9 Limitation of Liability and Remedies. Notwithstanding any damages that you might incur for any reason whatsoever (including, without limitation, all damages referenced above and all direct or general damages), the entire liability of Gemplus and any of its suppliers under this EULA and your exclusive remedy for all of the foregoing is limited to the greater of the amount actually paid by you for the Software or U.S. $50.00. The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if such remedy fails its essential purpose. You hereby waive and forever release Gemplus from any and all claims in excess of that amount.

10 General Provisions. This EULA contains the entire agreement between the parties with respect to its subject matter, and supersedes all prior or contemporaneous agreements or understandings (oral or written). This EULA is governed by and shall be interpreted in accordance with the laws of the jurisdiction or location (“Location”) of the offices of the Gemplus entity distributing the Software to you or as indicated in the documentation accompanying the Software, without giving effect to any applicable choice of law principles. Any and all disputes, claims or legal proceedings arising hereunder shall be subject to the nonexclusive jurisdiction of the competent courts of the Location. This EULA is not governed by the United Nations Convention on Contracts for the International Sales of Goods, the application of which is expressly excluded. This EULA may not be modified except by a written addendum issued by a duly authorized representative of Gemplus. No provision of this EULA can be waived unless such waiver is in writing and signed by a duly authorized representative of Gemplus. Unless you notify Gemplus that you prefer not to be listed as a customer, which you may do by emailing us at hotline@gemplus.com, Gemplus may list you as a customer and describe in general terms the services provided by Gemplus under this EULA in proposals and other marketing materials and Gemplus may use your logos and trademarks in support thereof. If any part of this EULA is found to be unenforceable or void, the remainder that part shall be limited or eliminated to the minimum extent necessary so that the remainder of this EULA shall otherwise stay valid and enforceable.

If you have any questions about this EULA, please immediately contact Gemalto at www.gemalto.com.

==========

This product contains code from pcsc-lite http://pcsclite.alioth.debian.org/

Copyright (c) 1999-2003 David Corcoran <corcoran@linuxnet.com> 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. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

   Changes to this license can be made only by the copyright author with explicit written consent.

   THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``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 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 contains code from OpenSC

http://www.opensc-project.org/

The files from OpenSC are:

./src/libopensc/asn1.c

./src/libopensc/asn1.h

./src/libopensc/card.c

./src/libopensc/cards.h

./src/libopensc/dir.c

./src/libopensc/errors.c

./src/libopensc/errors.h

./src/libopensc/internal.h

./src/libopensc/log.c

./src/libopensc/log.h

./src/libopensc/opensc.h

./src/libopensc/padding.c

./src/libopensc/pkcs15.c

./src/libopensc/pkcs15.h

./src/libopensc/pkcs15-algo.c

./src/libopensc/pkcs15-cache.c

./src/libopensc/pkcs15-cert.c

./src/libopensc/pkcs15-data.c

./src/libopensc/pkcs15-pin.c

./src/libopensc/pkcs15-prkey.c

./src/libopensc/pkcs15-pubkey.c

./src/libopensc/pkcs15-syn.c

./src/libopensc/pkcs15-wrap.c

./src/libopensc/sc.c

./src/libopensc/sec.c

./src/libopensc/types.h

./src/libopensc/ui.h

./src/scconf/scconf.c

./src/scconf/scconf.h

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the “Lesser” General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a “work based on the library” and a “work that uses the library”. The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called “this License”).

Each licensee is addressed as “you”.

A “library” means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The “Library”, below, refers to any such software library or work which has been distributed under these terms. A “work based on the Library” means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term “modification”.)

“Source code” for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1 You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2 You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3 You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4 You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-

readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5 A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a “work that uses the Library”. Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a “work that uses the Library” with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a “work that uses the library”. The executable is therefore covered by this License.

Section 6 states terms for distribution of such executables.

When a “work that uses the Library” uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not.

Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6.

Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6 As an exception to the Sections above, you may also combine or link a “work that uses the Library” with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable “work that uses the Library”, as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the “work that uses the Library” must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7 You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8 You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9 You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10 Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You

may not impose any further restrictions on the recipients' exercise of the rights granted herein.

You are not responsible for enforcing compliance by third parties with this License.

11 If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/ donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12 If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13 The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time.

Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and “any later version”, you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14 If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15 BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.

EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16 IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

Abbreviations

CA	Certificate Authority
ID	Identification
IMAP	Internet Message Access Protocol
OS	Operating System
PIN	Personal Identification Number
PKCS	Public Key Cryptography Standard
PKCS#11	Public Key Cryptography Standard **#11. For further information about this and other PKCS standards, refer to the RSA Laboratories web sit at http://www.rsa.com/rsalabs/**
POP	Post Office Protocol
RHEL	Red Hat Enterprise Linux
RSA	Rivest, Shamir, Adleman (inventors of public key cryptography standards)
S/MIME	Secure/Multipurpose Internet Mail Extensions
SSL	Secure Sockets Layer A protocol, v.3.0.v, for securing TCP/IP sessions

Glossary

Algorithm	A mathematical formula used to perform computations that can be used for security purposes.
Certificate	A certificate provides identification for secure transactions. It consists of a public key and other data, all of which have been digitally signed by a CA. It is a condition of access to secure e-mail or to secure Web sites.
Certificate Authority	An entity with the authority and methods to certify the identity of one or more parties in an exchange (an essential function in public key crypto systems).
Cryptography	The science of transforming confidential information to make it unreadable to unauthorized parties.
Digital Signature	A data string produced using a Public Key Crypto system to prove the identity of the sender and the integrity of the message.
Encryption	A cryptographic procedure whereby a legible message is encrypted and made illegible to all but the holder of the appropriate cryptographic key.
Key	A value that is used with a cryptographic algorithm to encrypt, decrypt, or sign data. Secret key crypto systems use only one secret key. Public key crypto systems use a public key to encrypt data and a private key to decrypt data.
Key Length	The number of bits forming a key. The longer the key, the more secure the encryption. Government regulations limit the length of cryptographic keys.
Public Key Crypto system	A cryptographic system that uses two different keys (public and private) for encrypting data. The most well-known public key algorithm is RSA.
SSL	Secure Sockets Layer: A Security protocol used between servers and browsers for secure Web sessions.
SSL Handshake	The SSL handshake, which takes place each time you start a secure Web session, identifies the server. This is automatically performed by your browser.
S/MIME	A Standard offline message format for use in secure e-mail applications.
Token	In a security context, a token is a hardware object like a smart card, but it could also be a pluggable software module designed to interact with a specific hardware module, such as a smart card. Token-based authentication provides enhanced security because success depends on a physical identifier (the smart card) and a personal identification number (PIN).