A leader in digital security

www.gemalto.com

gemalto
security to be free

Classic Client 6.3 for Mac OS X

User Guide

gemalto
security to be free

www.gemalto.com

gemalto
security to be free

Contents

List of Figures

Welcome to Gemalto Classic Client for Mac OS X.

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 Mac. In this document, "connecting a device" can mean inserting a card in a reader or plugging a token in the USB port of a Mac.

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 and native Mac 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 Mac hardware and software.

It is assumed that the user of Classic Client has:

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

Documentation

Classic Client is delivered with the following documentation:

- *Classic Client for Mac OS X User Guide* (this document). The file for this document is located in /Library/Documentation/Classic_Client_MAC_User_Guide.pdf.

- A *ReleaseNotes.pdf.* This contains any relevant information about the installation and the complete version history.

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 Mac OS X software are provided throughout this document to illustrate the various procedures and descriptions. Some of these screen shots were produced with Classic Client running on older versions of Mac OS X.

Additional Resources

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

For Further Help

Further help is provided in the Gemalto Self Support portal at support.gemalto.com.

You can find information on how to contact your Gemalto representative by clicking **Contact Us** at the Gemalto web site, www.gemalto.com.

If You Find an Error

Gemalto makes every effort to prevent errors in its documentation. However, if you discover any errors or inaccuracies in this document, please inform your Gemalto representative. Please quote the document reference number found at the bottom of the legal notice on the inside front cover.

Installation

This chapter discusses information related to the installation of Classic Client for Mac OS X. It describes:

■ The peripherals you need to use Classic Client for Mac OS X.

■ How to install Classic Client on your computer.

System Requirements

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

Classic Client for Mac OS X is compiled into two different package files each package supporting different smart cards. The versions of the Mac OS, and the applications and smart card readers supported differ from one package to the other. Please refer to the Release Notes for detailed information about what each package supports.

Computer

The workstation must have at least 22 MB of available hard disk space and meet the normal system requirements to run the version of Mac OS installed.

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 Mac OS X should run normally.

Peripherals

Classic Client for Mac OS X requires the following peripherals:

■ A CD ROM drive.

■ An available USB port

■ A smart card reader (or a secure PIN Pad reader) and its corresponding driver.

■ A smart card

For information about the smart card readers, drivers and smart cards that are supported for the version of Classic Client you have bought, please refer to the accompanying Release Notes (the ReleaseNotes.pdf file) that is supplied.

Installing Classic Client for Mac OS X

Installing the Classic Client for Mac OS X Software

Caution: Before installing the software, make sure that your system has the latest version of the CCID driver.

To install Classic Client:

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 Mac. The window in "Figure 1" on page 2 appears.

■   If your administrator has made the installation program available from a network device, navigate to the network location and download the .dmg file to your Mac.

   –   Double-click the .dmg file. The window in "Figure 1" on page 2 appears.

Figure 1 - Classic Client Initial Window
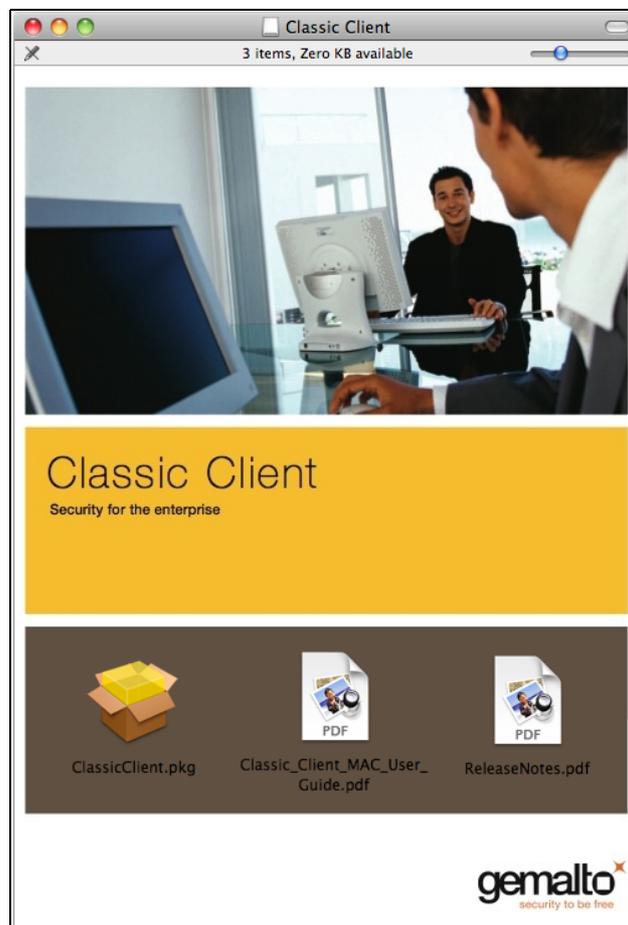
2   Gemalto recommends that you save the Release Notes to your computer's hard disk at this point by dragging the ReleaseNotes.pdf file to your Mac desktop.

3   Double-click the **ClassicClient.pkg** icon to start installing the Classic Client software.

   The installation program begins by displaying the **Introduction** window as shown in "Figure 2" on page 3.

Figure 2 - Installation — Introduction Dialog Box

4   Click **Continue.** The **License** window appears as shown in "Figure 3".

Figure 3 - Installation — License Window
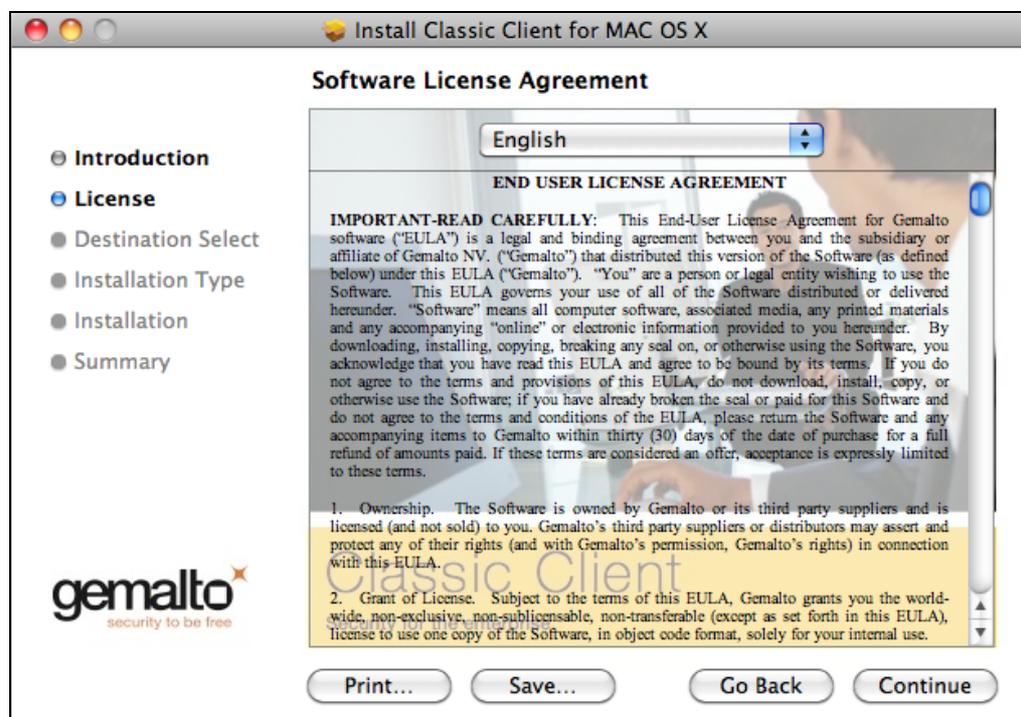
You can click **Print** to print out the file or **Save** to save it to the hard disk.

Note: The license is not installed automatically, so Gemalto recommends that you do at least one of the above.

5 When you have read the license, click **Continue**. The dialog box in "Figure 4" appears.

Figure 4 - Installation — Agree or Disagree to License Window
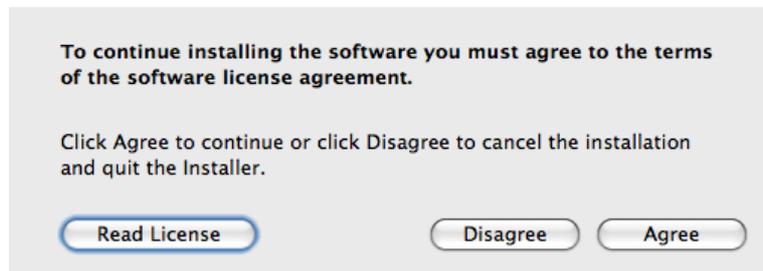
6 Click **Agree** to continue the installation. If you want to look at the license again first, click **Read License**. The **Installation Type** window displays as shown in "Figure 5".

Figure 5 - Installation Type Window
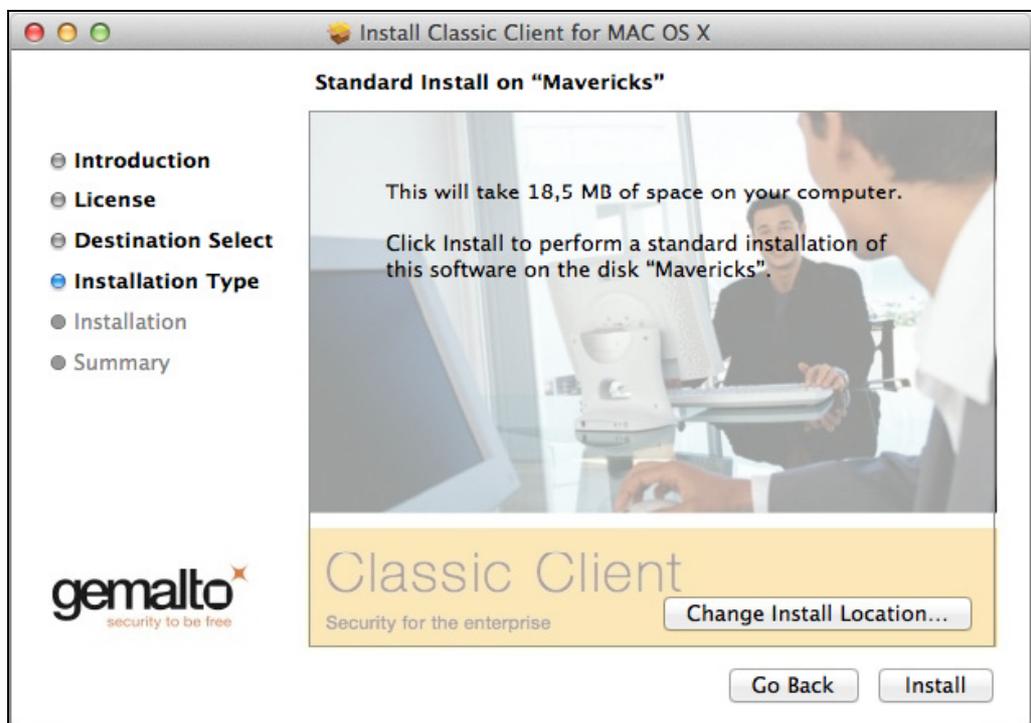
7 Click **Install** (ignore the **Change Install Location** button – you cannot change the location of the installation). The **Authenticate** dialog box shown in "Figure 6" appears.

Figure 6 - Installation — Authenticate Dialog Box

This dialog box appears because you must have the necessary rights to install the Classic Client software on the Mac.

Note: Make sure that you specify a user account that has the necessary rights to install software on your Mac.

8   Enter the **Name** and **Password**, then click **OK**.

A progress bar displays during the installation. When the installation ends, the **Summary** window appears as shown in "Figure 7".

Figure 7 - Installation — Summary Window
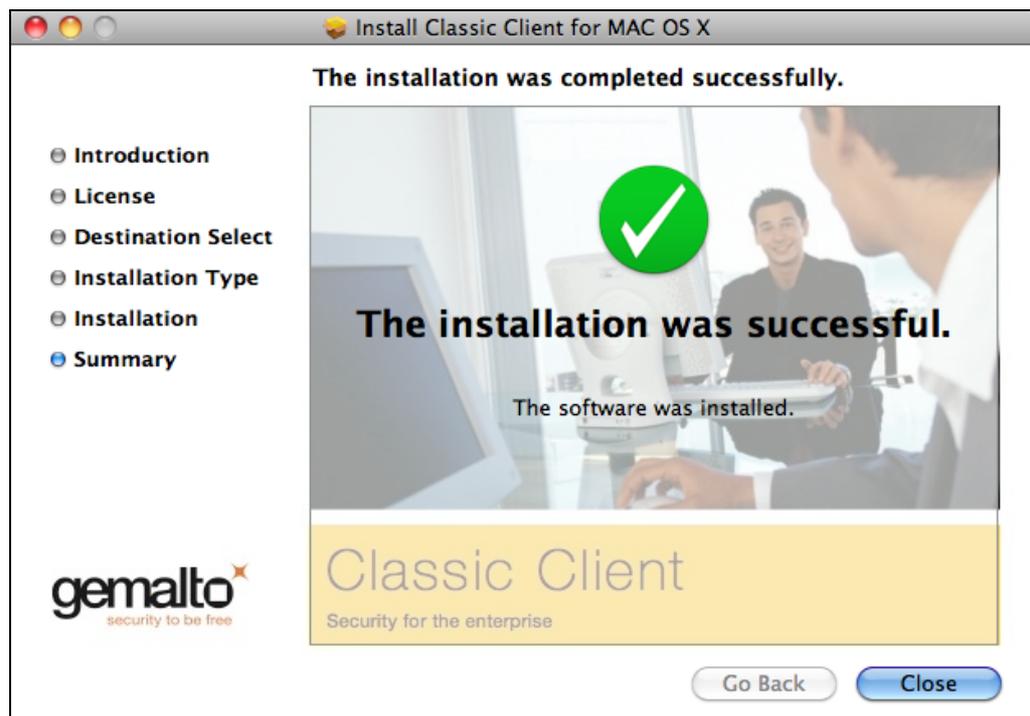
9   Click **Close** to complete the installation.

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.

Classic Client for Mac OS X includes two security modules that are installed automatically as part of the Classic Client software.

Caution:

Even though Apple has officially deprecated the Token D since OS X Lion:

http://smartcardservices.macosforge.org/post/apple-deprecates-smart-card-services-in-os-x-lion-v107/

Gemalto has decided to include a TokenD in the Classic Client Mac OS X package, in order to ease the migration of its customers.

Nevertheless, Gemalto cannot commit to provide maintenance for this TokenD security module, due to the official lack of support from Apple from now on.

Gemalto recommends its customers to use the PKCS#11 security module of Classic Client Mac OS X, for which the maintenance can be ensured by Gemalto.

- The Tokend security module enables Safari and the native Mac e-mail application to communicate with the smart card. There is no need for further configuration.

- The PKCS#11 security module enables the Mozilla applications Firefox (browser) and Thunderbird (e-mail) to communicate with the smart card. However it must first be registered in each Mozilla application.

  There are two methods of configuring the Mozilla applications to recognize PKCS#11. The first is simpler and is the recommended method, however it is valid for Firefox only. The second is valid for both Firefox and Thunderbird.

To configure Firefox to recognize the security module:

1 Open **Firefox** and from the **Firefox** 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 8".

Figure 8 - Encryption Tab in Advanced Dialog
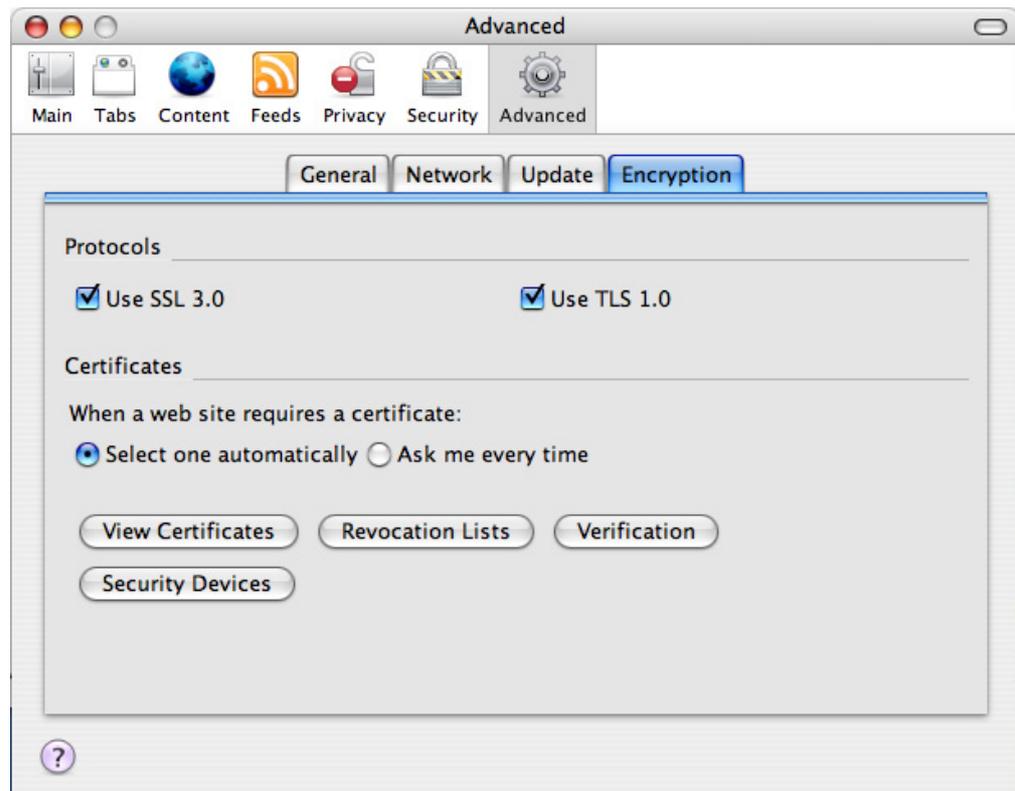
3   Click **Security Devices** to display the **Device Manager** window. This displays the modules currently available as shown in "Figure 9" on page 7.

Figure 9 - Device Manager
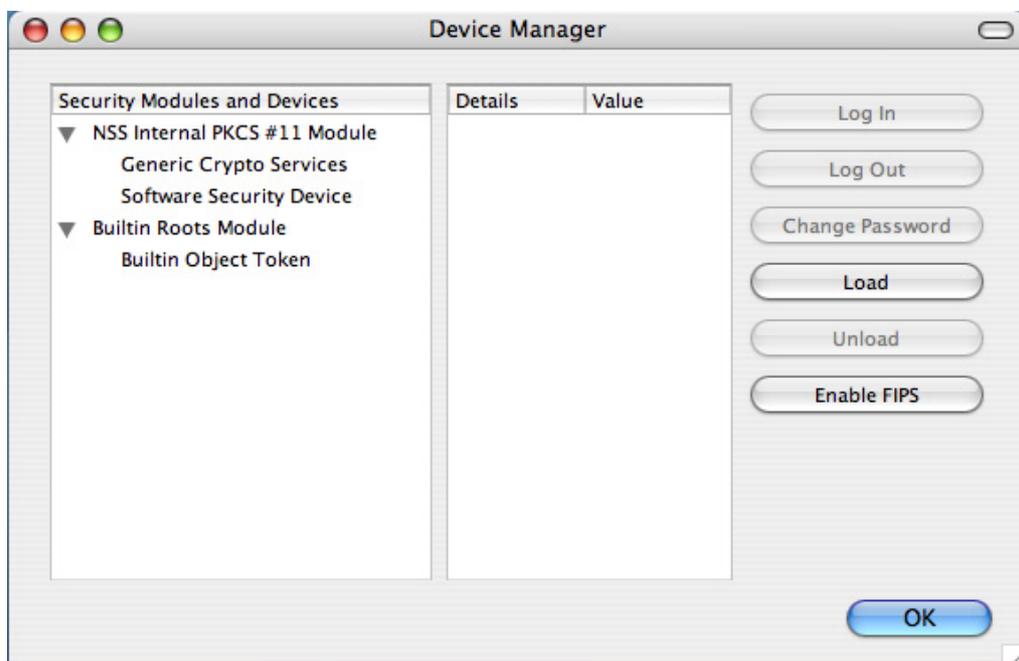
4   Click the **Load** button to the right in the dialog. This displays the **Load PKCS#11 Device** window, as shown in "Figure 10".

Figure 10 - The Load PKCS#11 Device Dialog Box
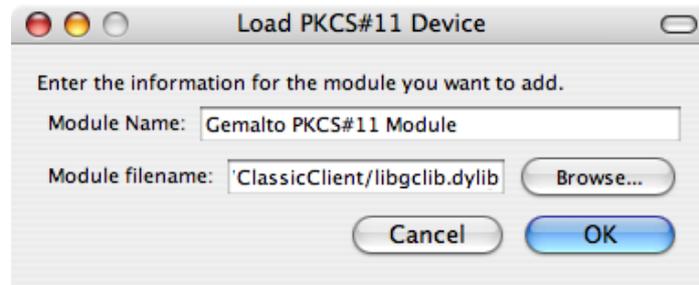
5   Enter a **Module Name.**

6   In **Module filename**, enter the full path and filename for the module. This is can be either:

   –   /usr/lib/ClassicClient/libgclib.dylib

   –   /Library/Frameworks/GemaltoClassicClient.framework/GemaltoClassicClient

Note: Do not use the **Browse** button to navigate to this file.

For certain applications you must choose the .framework file.

7   Click **OK**. The confirmation dialog appears as shown in "Figure 11" on page 8:

Figure 11 - Confirm Dialog
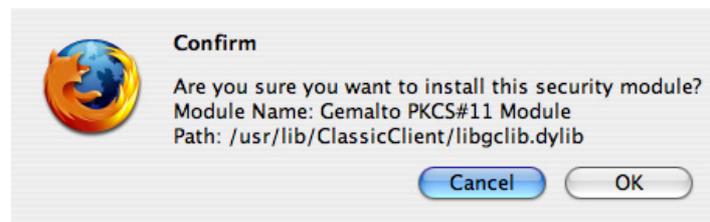
8   Click **OK**.

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 12 - Alert Dialog
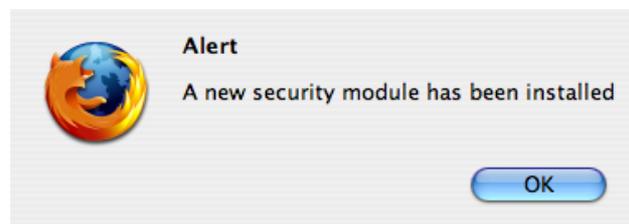
9   Click **OK** to close this **Alert**.

The **Device Manager** indicates the presence of the new module as shown in "Figure 13":

Figure 13 - Cryptographic Modules Available
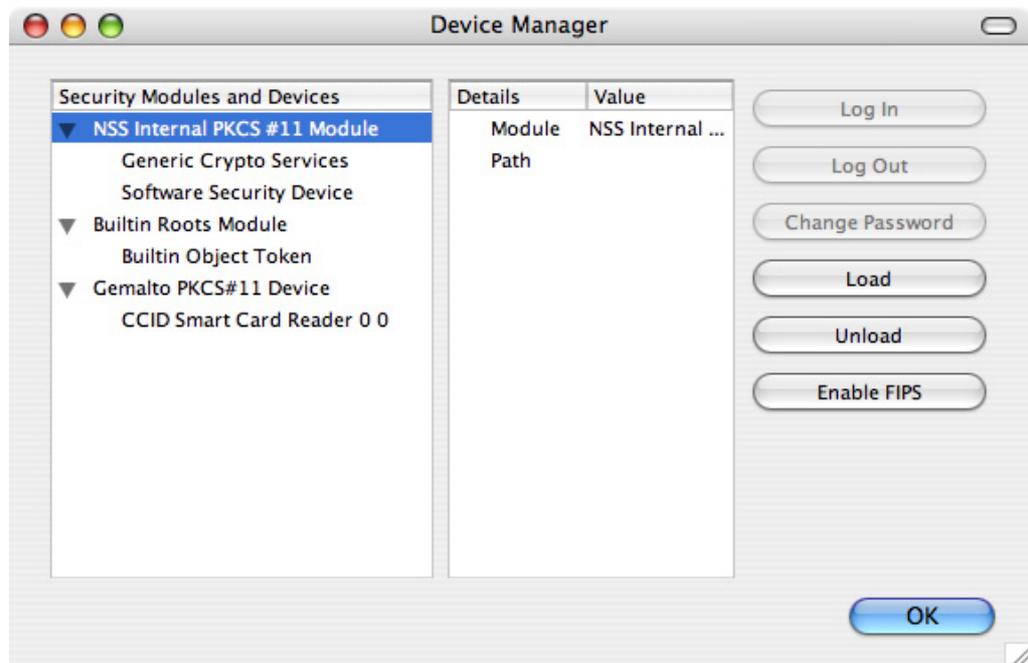
PIN Management

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 13. 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 14. 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 Management Tasks

This section describes the tasks that you can perform with the PIN Management Tool. For information on how to use it with a PIN Pad reader, please refer to "PIN Management Tasks (With a PIN Pad Reader)" on page 15.

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   From **Finder**, go to **Applications** > **Gemalto** and double-click the **Classic Client PIN Management** icon .

3   When the window shown in "Figure 14" appears, select a smart card reader from the list and click **OK**.

Figure 14 - Selecting a Smart Card Reader for the PIN Management Tool
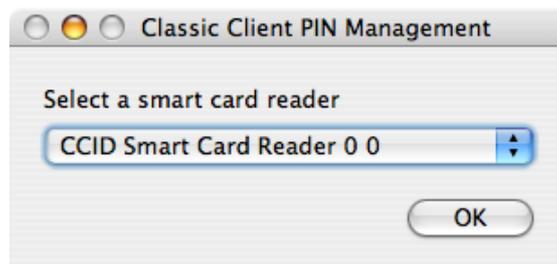
This opens the **Classic Client PIN Management** Window as shown in "Figure 15".

Figure 15 - Classic Client PIN Management - Change PIN Function
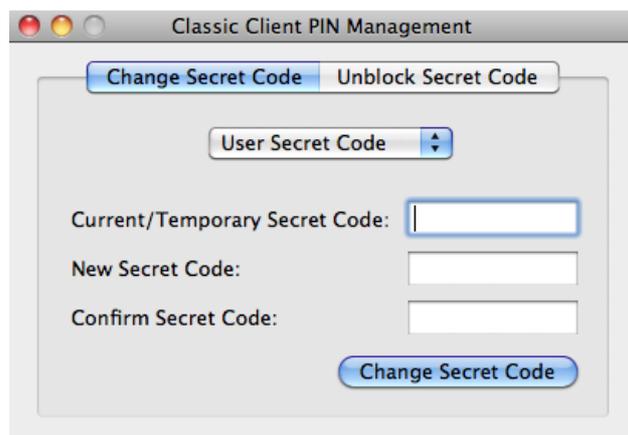
Caution: The PIN Management Tool detects only those cards and token that are connected at the time that it starts. Consequently, if you remove a card or token and reinsert it (even if in the same reader or USB port), you must restart the PIN Management Tool in order for it to be detected again. Similarly, if you connect a second card/token after the tool was started, you will need to restart it in order for it to detect the additional card/token.

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 Mac.

2   Open the PIN Management window as described in "How to Access the Classic Client PIN Management Tool" on page 12.

3   If it is not already selected, click **Change Secret Code** at the top of the window (see "Figure 15" on page 12).

4   Select the PIN whose value you want to change from the list, **Admin Secret Code** or **User Secret Code**.

5   Enter the current value of the PIN in **Current/Temporary Secret Code**, and the new value in **New Secret Code** and again in **Confirm Secret Code**.

6   Click the **Change Secret Code** 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 Mac.

■   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 14.

To unblock a PIN as an administrator:

1   Connect the blocked smart card/token to your administrator Mac.

2   Open the Classic Client PIN Management window as described in "How to Access the Classic Client PIN Management Tool" on page 12.

3   If it is not already selected, click **Unblock Secret Code** at the top of the window as shown in "Figure 16" on page 14.

Figure 16 - Classic Client PIN Management - Unblock PIN Function
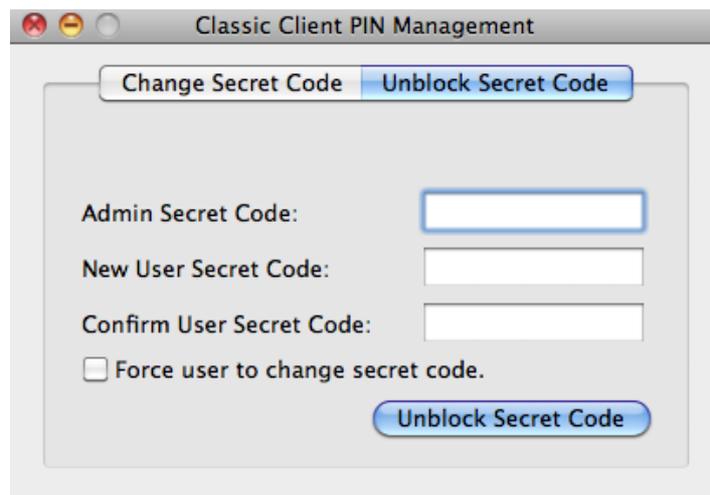
4   Enter the Admin PIN in **Admin Secret Code**, and the new value for the User PIN in **New User Secret Code** and again in **Confirm User Secret Code**.

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

6   Click the **Unblock Secret Code** button at the bottom of the window. A pop-up window appears to confirm a successful **Unblock Secret Code** 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 Mac.

2   Open the PIN Management window as described in "How to Access the Classic Client PIN Management Tool" on page 12.

3   If it is not already selected (in blue), click **Unblock Secret Code** as shown in "Figure 17".

Figure 17 - Classic Client PIN Management - Remote Unblock PIN Function
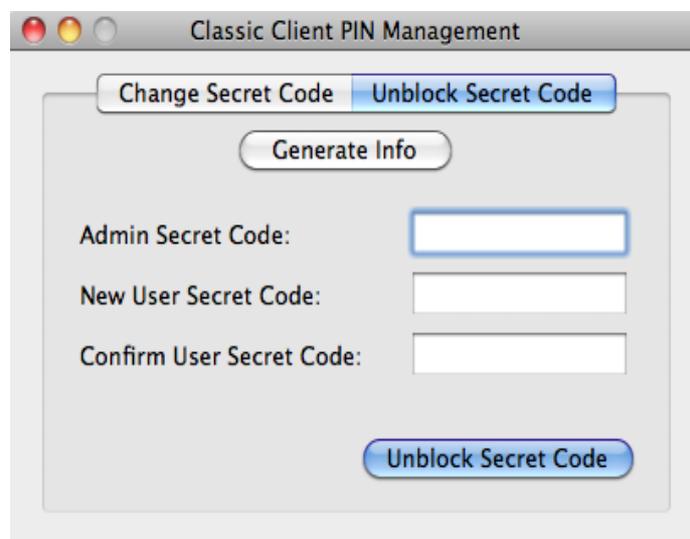
4   Click **Generate Info**. If this button does not appear in the window, you do not have the rights necessary to unblock your User PIN remotely. A window like the one shown in "Figure 18" appears.

Figure 18 - Remote Unblock Secret Code - Information for Help Desk
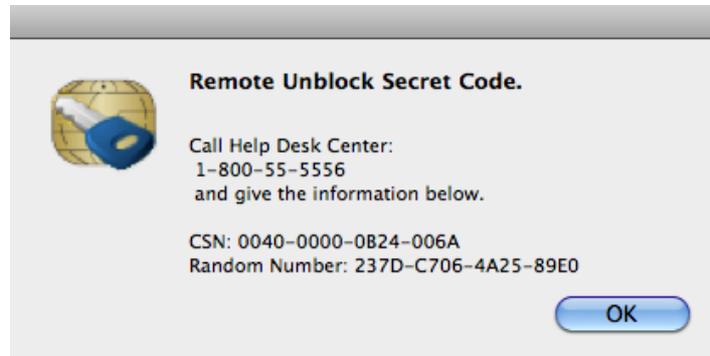
5   Call your administrator or Help Desk at the number given in the window, and tell them the CSN and Random Number that appear in the window. Click **OK** to close the window.

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

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

8   Click the **Unblock Secret Code** 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.

PIN Management Tasks (With a PIN Pad Reader)

This section describes the tasks that you can perform with the PIN Management Tool, using a PIN Pad reader.

How to Access the Classic Client PIN Management Tool (with a PIN Pad Reader)

To access the PIN Tool:

1   Make sure that your smart card/token is inserted in your PIN Pad reader and that the Pad reader is connected to your computer.

2   From **Finder**, go to **Applications** > **Gemalto** and double-click the **Classic Client PIN Management** icon .

3   When the window shown in "Figure 19" appears, select the PIN Pad reader from the list and click **OK**.

Figure 19 - Selecting a Smart Card Reader for the PIN Management Tool
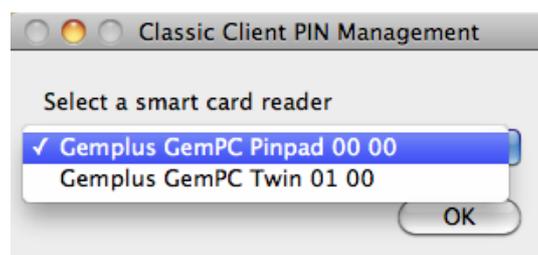
This opens the **Classic Client PIN Management** Window as shown in "Figure 20".

Figure 20 - Classic Client PIN Management - Change PIN Function

Caution: The PIN Management Tool detects only those cards and tokens that are connected at the time that it starts. Consequently, if you remove a card or token and reinsert it (even if in the same reader or USB port), you must restart the PIN Management Tool in order for it to be detected again. Similarly, if you connect a second card/token after the tool was started, you will need to restart it in order for it to detect the additional card/token.

How to Change an Administration PIN or User PIN (with a PIN Pad Reader)

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   Follow the instructions in "How to Access the Classic Client PIN Management Tool (with a PIN Pad Reader)" on page 15, until you see the window in "Figure 20".

2   If it is not already selected, click **Change Secret Code** at the top of the window.

3   Select the PIN whose value you want to change from the list, **Admin Secret Code** or **User Secret Code**.

4   Click the **Change Secret Code** button at the bottom of the window, then follow the instructions displayed on the PIN Pad reader. 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 (with a PIN Pad Reader)

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, 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 Mac.

Note: With PIN Pad readers, it is not possible to remotely unblock a PIN.

To unblock a PIN as an administrator:

1   Follow the instructions in "How to Access the Classic Client PIN Management Tool (with a PIN Pad Reader)" on page 15, until you see the window in "Figure 20".

2   Click **Unblock Secret Code** at the top of the window as shown in "Figure 21".

Figure 21 - Classic Client PIN Management - Unblock PIN Function

3   Click the **Unblock Secret Code** button at the bottom of the window, then follow the instructions displayed on the PIN Pad reader. A pop-up window appears to confirm a successful **Unblock Secret Code** operation or to display an error message if unsuccessful.

Note: The "**Force user to change secret code**" feature is not available, when using a PIN Pad reader.

Tasks

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

These tasks are:

■ "How to View Card Contents Using Keychain Access" on this page.

■ "How to Use E-mail Securely" on page 19.

■ "How to View Secure Web Sites" on page 27.

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

How to View Card Contents Using Keychain Access

You can use Keychain Access to display the certificates and public and private keys that are stored in your smart card. Any applications that use Keychains to store credentials can communicate with Classic Client smart cards.

To view the contents of your smart card:

1   Make sure your smart card/token is connected.

2   From **Finder**, go to **Applications** > **Utilities** and double-click **Keychain Access**. This opens the window shown in "Figure 22".

Figure 22 - Keychain Access Window
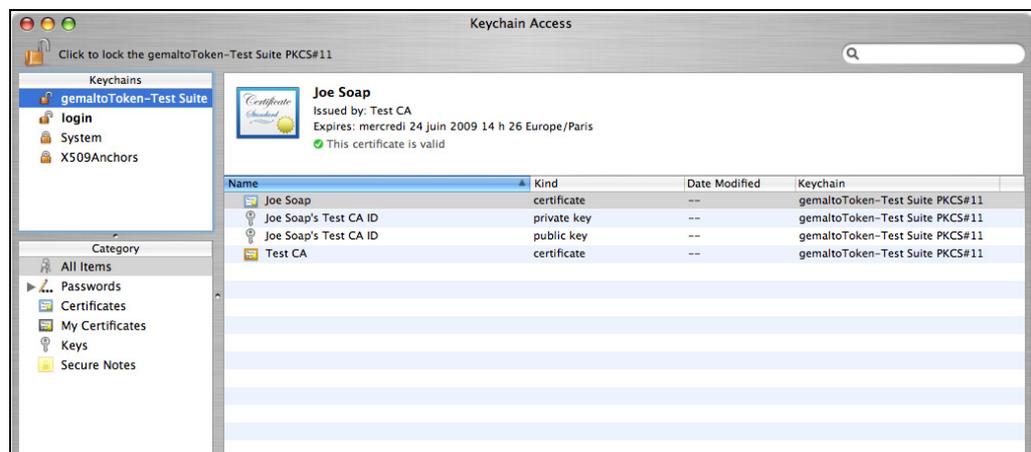
The smart card appears at the top of the **Keychains** pane as "gemaltoToken-xxx" where xxx is the name of your smart card (TestSuite in our example).

3   If not already selected, select the smart card in Keychains, as in "Figure 22". The main window on the right, displays the certificates and keys that are in the card.

4   To display the details of a certificate or key double-click it in the list. The details appear as shown in "Figure 23".

Figure 23 - Certificate Details
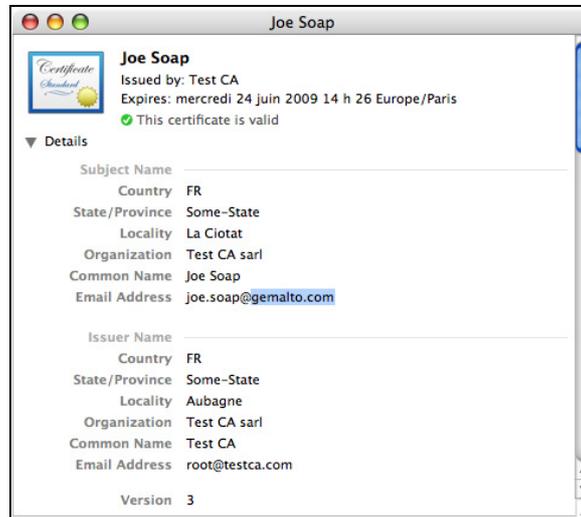
How to Use E-mail Securely

The following sections explain how to send secure e-mail using Classic Client for Mac OS X.

About Secure E-mail

With Classic Client for Mac OS X, 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

Depending on your e-mail application you will have to do some or all of the following before you can send secure e-mail:

■   **Configure the application to recognize the PKCS#11 security module**

Necessary for Thunderbird, not for Mail.

■   **Configure security settings**

Set the security settings for digitally signing and/or encrypting the contents and attachments of outgoing messages.

Necessary for Thunderbird, not for Mail.

■ **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.

Necessary for Thunderbird, not for Mail. Mail chooses the certificate itself, see "Working with Mail (Mac's native Mail System)" on page 26.

■ **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 and native Mail 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.

The following sections explain how to set up and send secure e-mail with Mozilla's Thunderbird e-mail program. 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 21

3 Send a digitally signed e-mail to yourself so that Thunderbird recognizes your public key, described on page 24.

Configure Thunderbird to recognize the Security Module

You only need to do this once in Thunderbird and the method to do this is almost identical to Firefox.

To configure Mozilla Thunderbird

1 Make sure your smart card/token is connected.

2 Start **Mozilla Thunderbird**.

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

4 For the rest of the procedure, follow the instructions in "To configure Firefox to recognize the security module:" on page 6, except that in step 2 of those instructions, choose the **Certificates** tab instead of the **Encryption** tab.

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

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**.

3 Enter your password if you are prompted for it.

4 In **Thunderbird**, click the **Write** icon as shown in "Figure 24".

Figure 24 - Thunderbird Write Icon
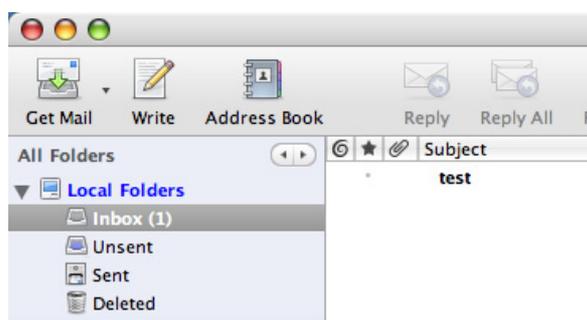
This opens the **Compose** window.

5 In the **Compose** window's **Options** menu, choose **Security** > **Encrypt this Message** as shown in "Figure 25".

Figure 25 - Thunderbird – Encrypt This Message
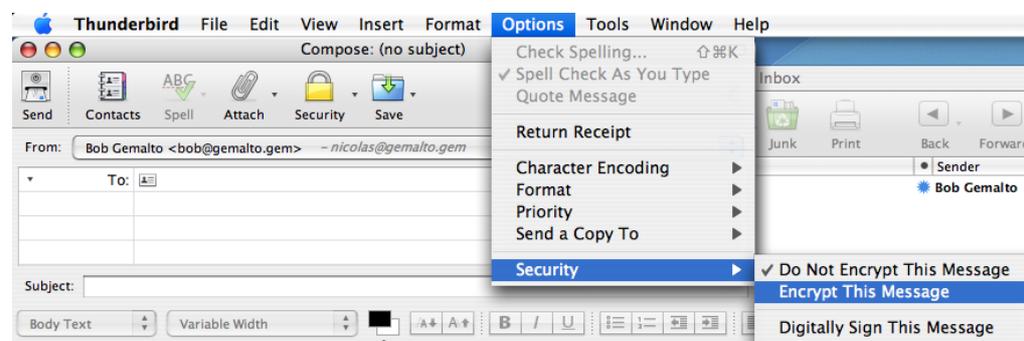
As the certificates in the card/token are not yet set up, the following message appears:
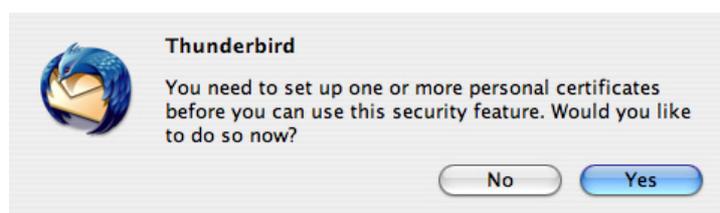
6 Click **Yes**. This opens the security account settings window for your e-mail account as shown in "Figure 26" on page 22.

Figure 26 - Thunderbird – Security Account Settings
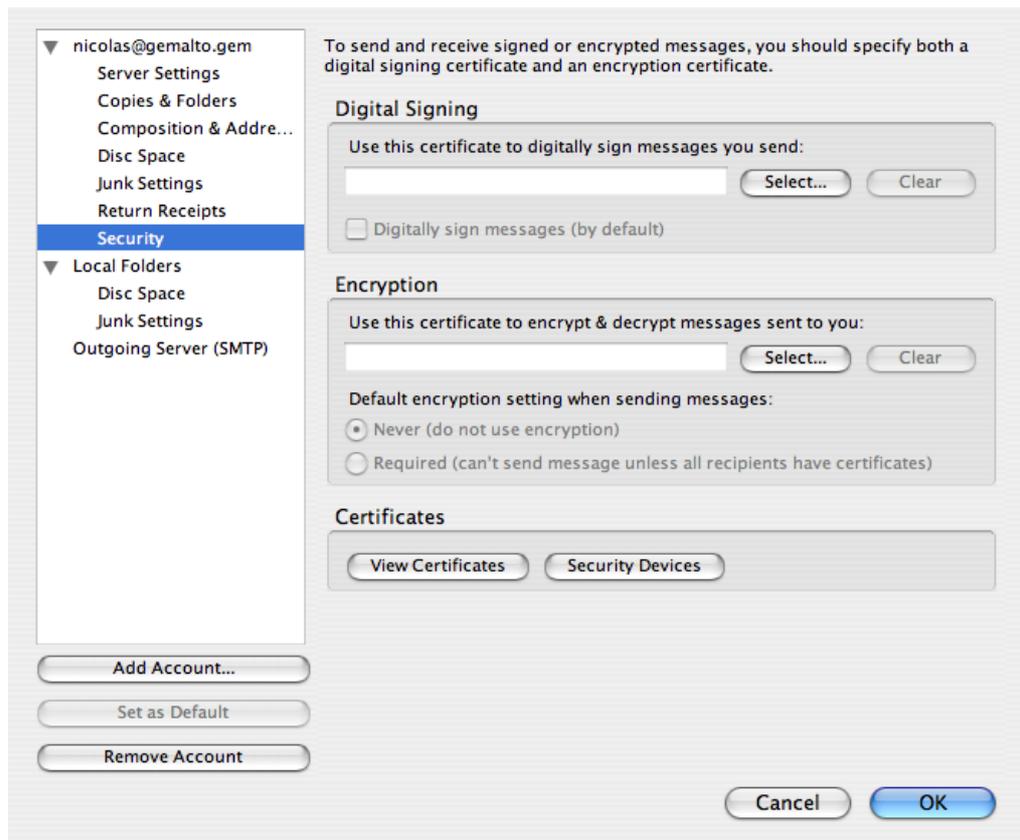
7   In **Digital Signing**, click **Select** and choose the certificate you want to use from the list that appears.

Note: You may be prompted to enter a "master password" as shown in "Figure 27". If so, enter the PIN for the card and click **OK**.

Figure 27 - Thunderbird - Enter Password
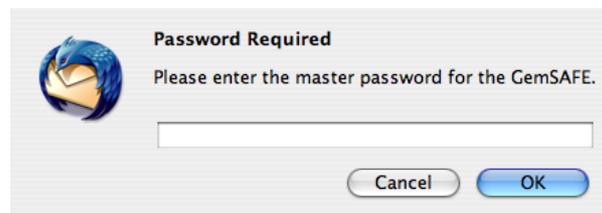
The details of the selected certificate appear, as shown in "Figure 28" on page 23.

Figure 28 - Thunderbird - Details of Selected Certificate
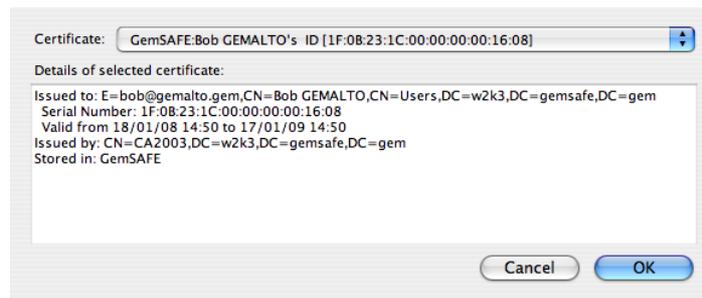
8   Click **OK**. The following message appears:

Figure 29 - Thunderbird – "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 30". Otherwise click **Cancel**.

Figure 30 - Thunderbird – Security Account Settings (2)
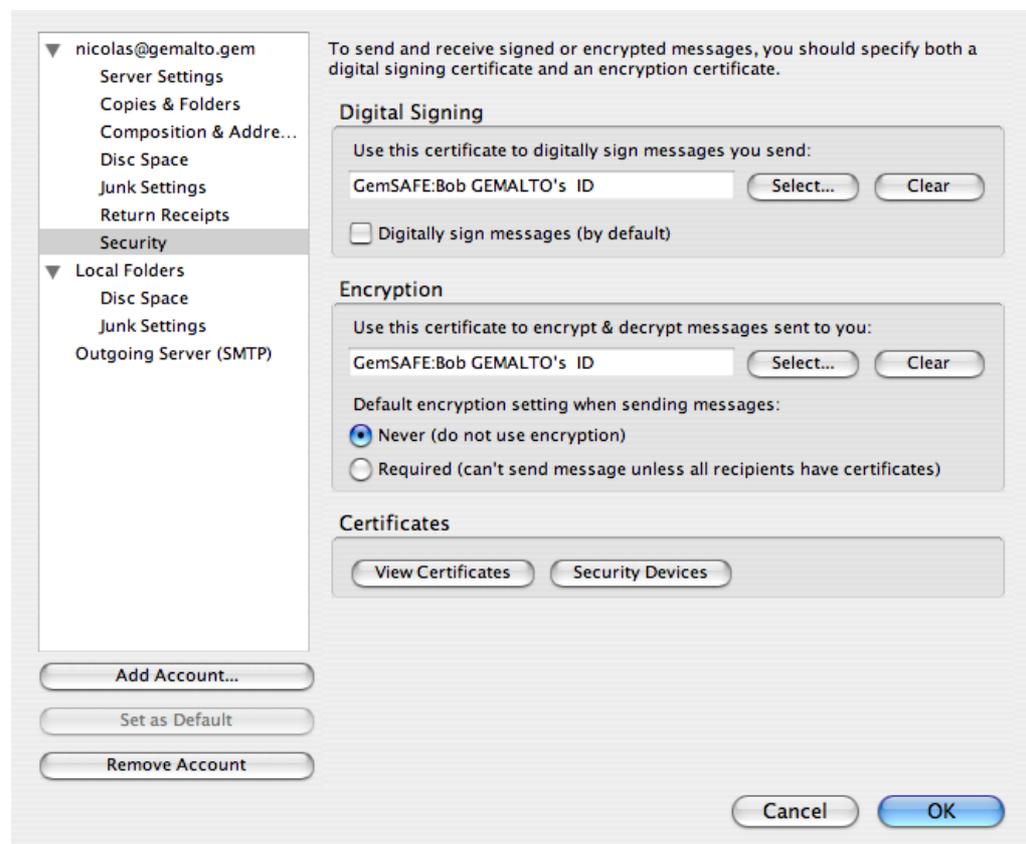
10 If you want all of your e-mails to be digitally signed by default, check the box **Digitally sign messages (by default)**.

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 29" on page 23 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.

Sending Digitally Signed E-mail with Mozilla Thunderbird

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 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).

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

1 Make sure your smart card/token is connected.

2 Start **Mozilla Thunderbird**.

3 Enter your password if you are prompted for it.

4 In **Thunderbird**, click the **Write** icon as shown in "Figure 31".

Figure 31 - Thunderbird Write Icon
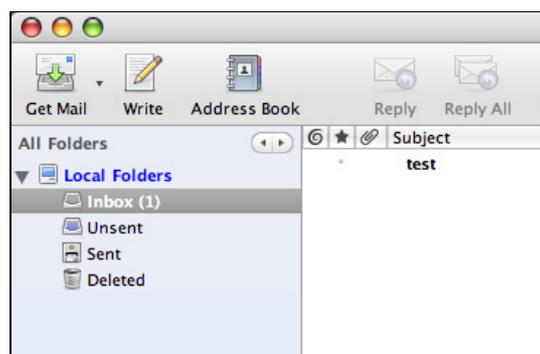
This opens the **Compose** window.

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

Be sure to include a subject heading.

Figure 32 - Thunderbird New Msg Composition Window
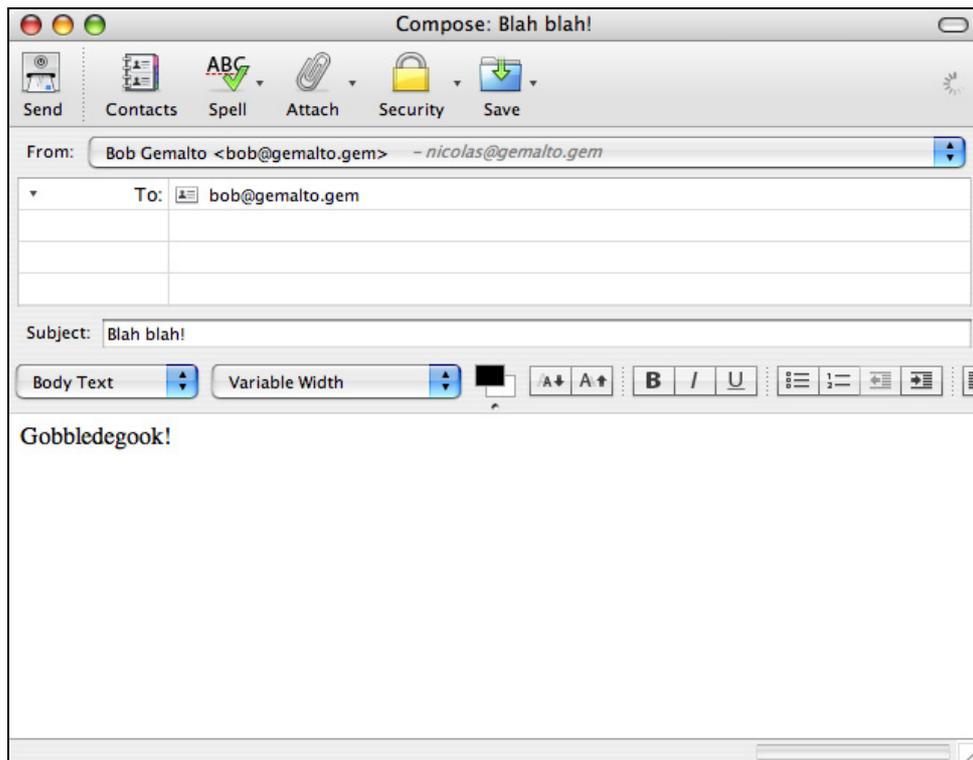
6   From the **Options** menu in the **Compose** window choose **Security > Digitally Sign this Message** in order to sign the message.

> **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 33".

Figure 33 - Thunderbird Message Security Info Window
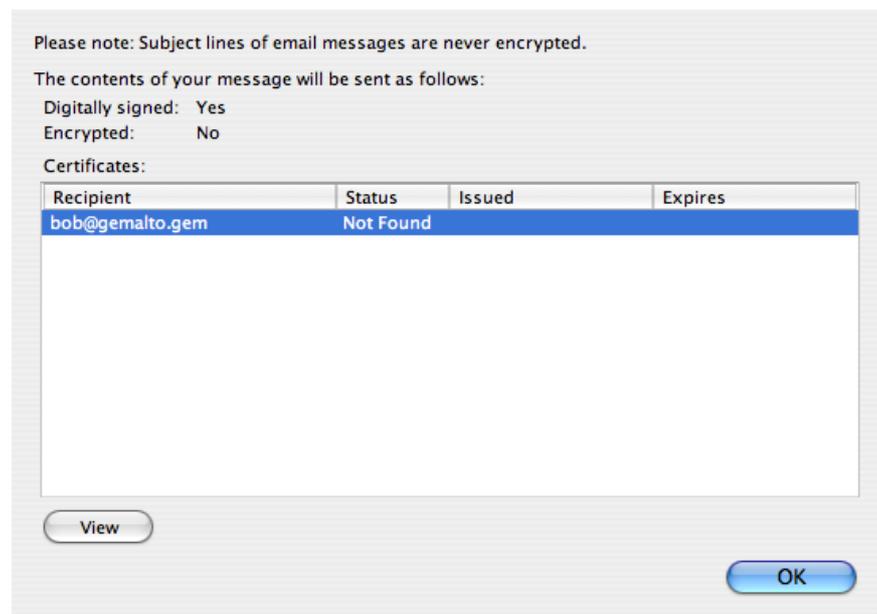
You can display details about the certificate by clicking **View**.

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 27" on page 22, 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 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

Once you have configured your e-mail account in **Mozilla Thunderbird**, 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; 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" on page 24, except in the **Compose** window, choose **Encrypt this message** from the **Options** menu.

Working with Mail (Mac's native Mail System)

The following sections explain how to set up and send secure e-mail with Mail, Mac's native e-mail application.

There is no need to install any security module or assign particular certificates for encryption and digital signatures (like you do in other e-mail applications such as Thunderbird).

When you send an encrypted and/or digitally signed e-mail in Mail, Mail searches the card for a correct certificate, that is, a certificate with an e-mail address that corresponds to the account of the user who is logged in. The card appears as a keychain in the Mac.

You need to send a digitally signed e-mail to yourself so that Mail recognizes your public key.

To send a digitally signed mail in Mail:

1   Make sure your smart card/token is connected.

2   Start **Mail** by clicking the Mail icon 📮 .

3   Enter your password if you are prompted for it.

4   In **Mail**, click the **New** icon 📝 .

This opens the **New Message** window.

5   In the **New Message** window, write a short message *addressed to yourself* as shown in "Figure 34" on page 27.

Be sure to include a subject heading.

Figure 34 - Mail New Msg Window
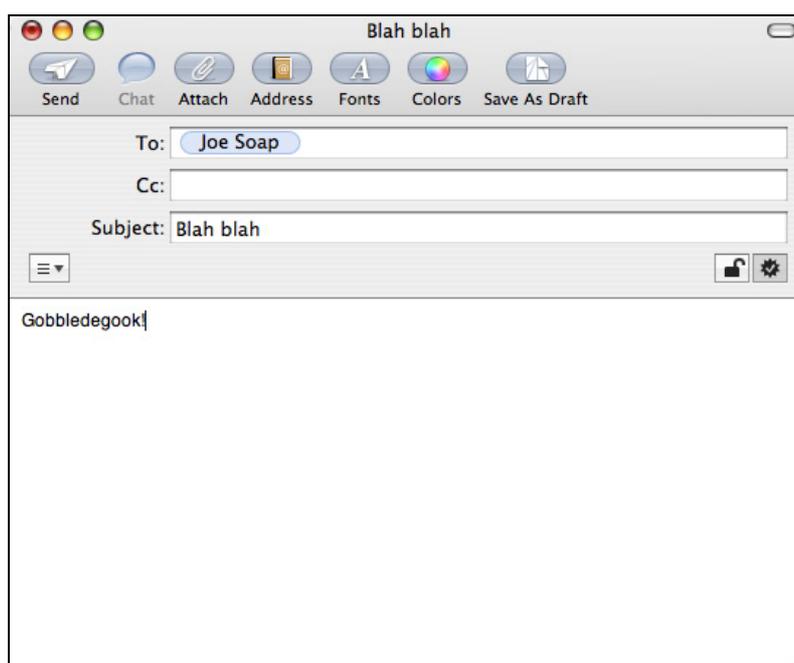
6   Sign the message by clicking the **Sign** icon ⚜ .

7   Click **Send** 📨 to send the mail.

To send an encrypted e-mail:

Follow the same steps as "To send a digitally signed mail in Mail:" on page 26, except in the **New Message** window, click the **Encrypt** icon 🔓 .

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 different browsers.

Choosing a Certificate to Authenticate Yourself to Secure Web Sites

To use your smart card to authenticate yourself to web sites on the internet, a certificate in the smart card must be recognized by your browser.

Safari

To use Safari, there are no further configurations to make. Once you connect your smart card/token, it appears as a keychain.

Safari searches the keychains in the order that they appear in the list (as illustrated in "Figure 22" on page 18). This means that the smart card is searched first, then **login** and so on.

The certificate search is as follows:

1 Safari searches the keychains in order and tries to access the web site using the first **valid** certificate that it finds. A valid certificate is one that has the same domain name as the web site you are trying to access.

2 If no valid certificate can be found in any of the keychains, then a message displays to tell you that you are not authorized to access the web site.

Mozilla Firefox

In Firefox, you do need to register a certificate. This section tells you how to do this and how to choose whether the browser should choose a certificate automatically or ask you first.

To check certificates registered in Mozilla Firefox:

1 Make sure your card/token is connected.

2 Open Mozilla Firefox.

3 From the **Firefox** menu choose **Preferences**.

4 Click the **Advanced** icon, then the **Encryption** tab as shown in "Figure 35".

Figure 35 - Mozilla Firefox Encryption Options Dialog
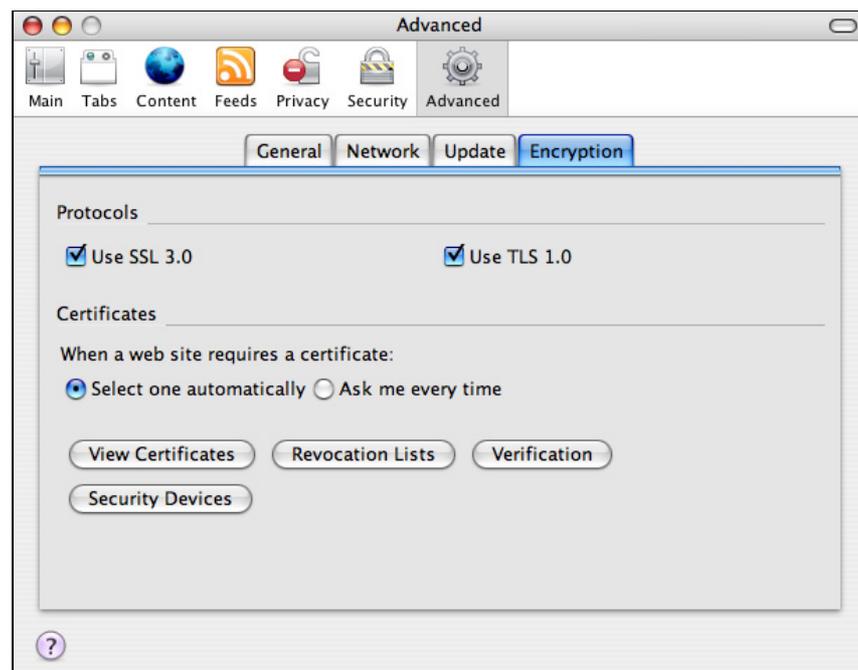
5   In **Certificates**, choose one of the options for the action to take when a web site requires a certificate:

–   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 36" on page 29.

Figure 36 - Password Required
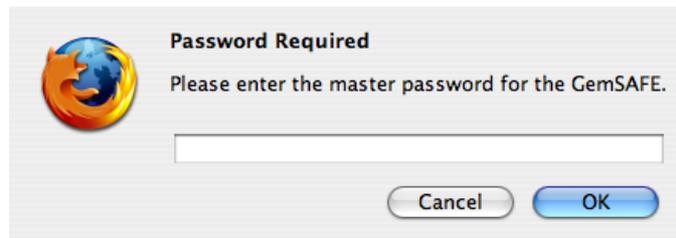
7   Enter the User PIN for your card/token.

The **Certificate Manage**r window appears.

Figure 37 - Certificate Manager Window
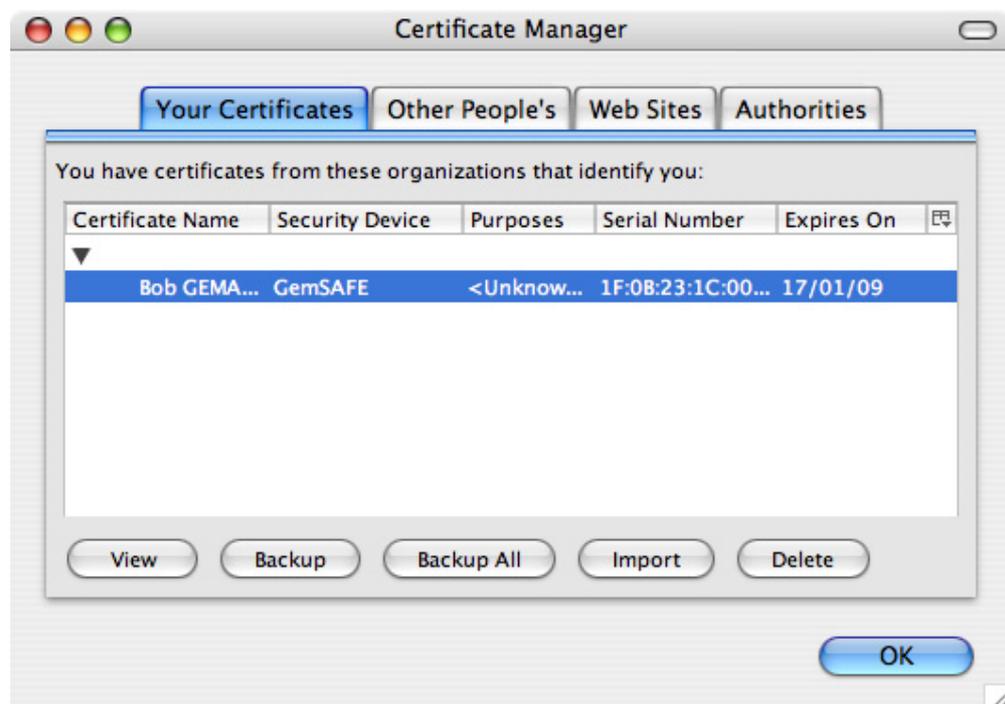
8   Under **Your Certificates** appears the certificates that are stored on the card/token. To display the properties of a particular certificate, select it and click **View**.

Contactless Cards

Contactless cards behave in the same way as contact cards. However, some contactless cards have an additional feature. This feature is available if requested from Gemalto. The contactless secure data (CSD) mechanism is designed to protect confidential data about the cardholder from being read by a third party without the cardholder's consent or knowledge.

When a reader tries to access the Classic Applet V2 or V3 in the smart card, the applet returns the **Classic Client CSD** dialog box as shown in "Figure 38" on page 30).

Figure 38 - The CSD Dialog Box
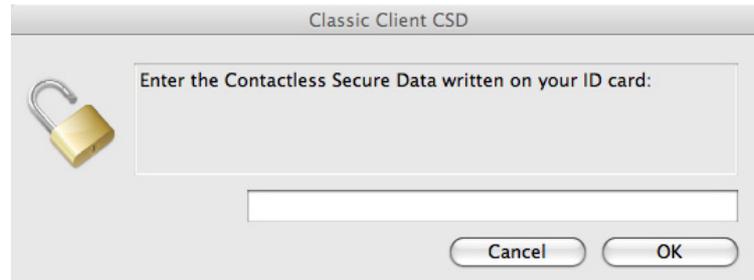
Enter the CSD and click **OK**. The reader can then access the Classic Applet V2 or V3.

The CSD is specified by the card issuer but is typically information such as the last four digits of the card serial number as printed on the card.

If you click **Cancel**, you must remove the smart card from the reader before it can be reread. Normally the **Classic Client CSD** dialog box displays once only at the beginning of each card session, that is, when the card is first read by the reader. However, under certain circumstances (such as if the card session is broken for some reason) it is possible that it may be displayed again to reprompt for the CSD.

Note: As an extra security measure against "brute force" attacks (where the reader may attempt to read the applet many times in a short time), the smart card deliberately slows down the verification of the CSD code after an incorrect CSD entry. The more incorrect CSD attempts are made, the slower the response to process the next CSD attempt.

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 and native Mac 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 Mac.

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.

Abbreviations

CA	Certificate Authority
ID	Identification
IMAP	Internet Message Access Protocol
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
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).
Tokend	The preferred means to work with smart cards on Mac OS X is by using Keychain Services. The Mac OS X implements the Tokend interface that allows smart card developers to make their cards appear to be keychains