SafeNet Authentication Client (Mac)

Version 8.2 SP1 Revision A

Administrator's Guide

All attempts have been made to make the information in this document complete and accurate.

SafeNet, Inc. is not responsible for any direct or indirect damages or loss of business resulting from inaccuracies or omissions. The specifications contained in this document are subject to change without notice.

SafeNet and SafeNet Authentication Manager are either registered with the U.S. Patent and Trademark Office or are trademarks of SafeNet, Inc., and its subsidiaries and affiliates, in the United States and other countries. All other trademarks referenced in this Manual are trademarks of their respective owners.

SafeNet Hardware and/or Software products described in this document may be protected by one or more U.S. Patents, foreign patents, or pending patent applications.

Please contact SafeNet Support for details of FCC Compliance, CE Compliance, and UL Notification.

Date of publication: July 2013

Last update: Thursday, December 27, 2012 2:45 pm

Support

We work closely with our reseller partners to offer the best worldwide technical support services. Your reseller is the first line of support when you have questions about products and services. However, if you require additional assistance you can contact us directly at:

Telephone

You can call our help-desk 24 hours a day, seven days a week:
USA: 1-800-545-6608
International: +1-410-931-7520

Email

You can send a question to the technical support team at the following email address:
support@safenet-inc.com

Website

You can submit a question through the SafeNet Support portal:
http://c3.safenet-inc.com/secure.asp

Additional Documentation

The following SafeNet publications are available:

- SafeNet Authentication Client (Mac) 8.2 SP1 User's Guide
- SafeNet Authentication Client (Mac) 8.2 SP1 ReadMe

Table of Contents

1 Introduction

SafeNet Authentication Client enables Token operations and the implementation of Token based PKI solutions.

In this chapter:

- Overview
- New Features

Overview

Public Key Infrastructure (PKI) is a framework for creating a secure method for exchanging information based on public key cryptography, providing for trusted third-party vetting of, and vouching for, user identities. It is an arrangement that consists of a system of digital certificates, Certificate Authorities, and other registration authorities that verify and authenticate the validity of each party involved in an internet transaction.

SafeNet's Authentication Client enables integration with various security applications. It enables token security applications and third party applications to communicate with the token. These include PKI solutions using PKCS#11 or proprietary token application.

SafeNet Authentication Client enables the implementation of strong two-factor authentication using standard certificates as well as encryption and digital signing of data. Generic integration with PKCS#11 security interfaces enables out-of-the-box interoperability with a variety of security applications offering secure web access, PC and data security, secure email, and more. PKI keys and certificates can be created, stored, and used securely from within token hardware or software devices.

SafeNet Authentication Client can be deployed and updated using any standard software distribution system.

The SafeNet Authentication Client Tools application is installed by the SafeNet Authentication Client, providing easy-to-use configuration tools for users and administrators.

New Features

The following features were introduced in SafeNet Authentication Client 8.2 (Mac):

- Support for OS x 10.8 (Mountain Lion)
- Support for Common Criteria (CC) certified devices and CC digital signatures.
- Support for following SHA2 algorithms: SHA256, SHA384, SHA512
- Support for onboard hashing: SHA1, SHA256
- Licensing Activation function
- Certificate expiry alert function
- Support for additional tokens:
  - SafeNet eToken Pro CC
  - SafeNet eToken 5100/5105
  - SafeNet eToken 5200/5205
  - SafeNet eToken 7300
  - SafeNet iKey 2032
  - SafeNet iKey 4000

2 System Requirements

This chapter describes the requirements for SafeNet Authentication Client (Mac) 8.2 SP1.

In this chapter:

- Supported Operating Systems
- Supported SHA 2 Algorithms
- Supported Algorithms for Onboard Hashing
- Supported Tokens
- Required Hardware
- PCSC-Lite

Supported Operating Systems

- Mac OS X 10.8 preview build (Mountain Lion) - Intel 64-bit
- Mac OS X 10.7.3 and 10.7.4 (Lion) - Intel 64-bit
- Mac OS X 10.6.7 and 10.6.8 (Snow Leopard) - Intel 32-bit and Intel 64-bit

Supported SHA 2 Algorithms

- SHA256
- SHA384
- SHA512

Supported Algorithms for Onboard Hashing

- SHA1
- SHA256

Supported Applications

Browsers

- Firefox
- Safari

Email Clients

- Thunderbird
- Mail.app (Mac OS X built-in email client)

Other

- Adobe Reader

Supported Tokens

- SafeNet eToken PRO (Mask 8)
- SafeNeteToken PRO CC
- SafeNet eToken PRO Anywhere
- SafeNet eToken Virtual/Rescue
- SafeNet eToken NG-OTP
- SafeNet eToken NG-Flash 4.5
- SafeNet eToken NG-Flash 5.3
- SafeNet eToken NG-Flash 5.3 Anywhere
- SafeNet eToken 5100/5105
- SafeNet eToken 5200/5205
- SafeNet eToken 7300
- SafeNet iKey 2032
- SafeNet iKey 4000

Required Hardware

- USB port (for physical Token devices)
- Recommended display resolution (for SafeNet Authentication Client Tools) 1024 x 768 pixels and higher.

PCSC-Lite

SafeNet Authentication Client (Mac) 8.2 SP1 uses the default PCSC-Lite that is installed with Mac OS X. SafeNet Authentication Client 8.2 SP1 installs a plug-in and driver for PCSC-Lite, during the normal installation process.

PCSC-Lite is managed by the Mac OS X Security Manager. When a device is inserted, the service runs automatically.

The SafeNet Authentication Client (Mac) 8.2 SP1 installation runs PCSC after reboot even if a token device is not inserted. This is required to support SafeNet eToken Virtual on a flash device.

3 Installation

This chapter describes the installation options for SafeNet Authentication Client (Mac) 8.2 SP1.

In this chapter:

- Installing with the Installer
- Installing from the Terminal
- Uninstalling
- Installing the Firefox Security Module
- Installing the Thunderbird Security Module
- Configuring Acrobat Security Settings

Installing with the Installer

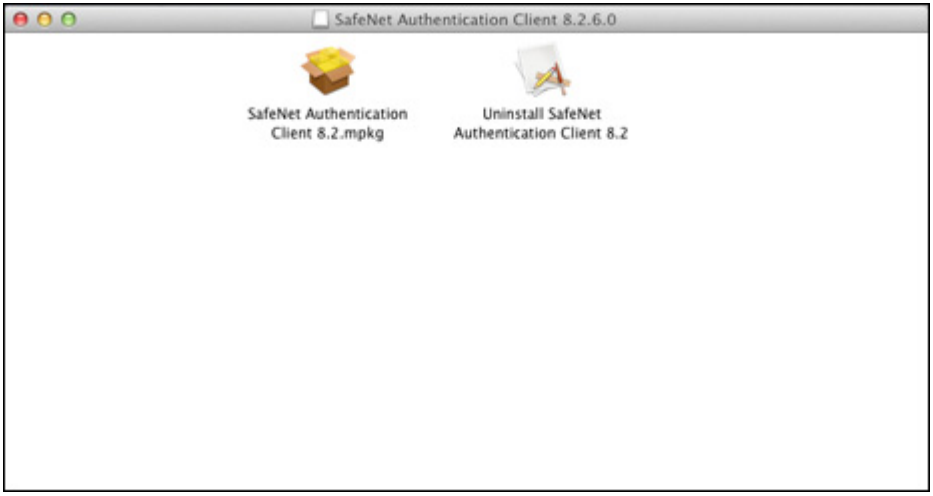
The installation packaging for SafeNet Authentication Client 8.2 SP1 (Mac) is PackageMaker.

The installation package is `SafeNetAuthenticationClient.8.2.x.0.dmg`.

To install with the installer:

1 Double click the `SafeNetAuthenticationClient.8.2.x.0.dmg` file.

A new disk image file is created in the Finder window, including an mpkg installation file and an uninstall application.



2 To start the installation, double click **SafeNet Authentication Client 8.2 SP1.mpkg**.

The *Welcome to the SafeNet Authentication Client Installer* window opens.

3 Click **Continue**.

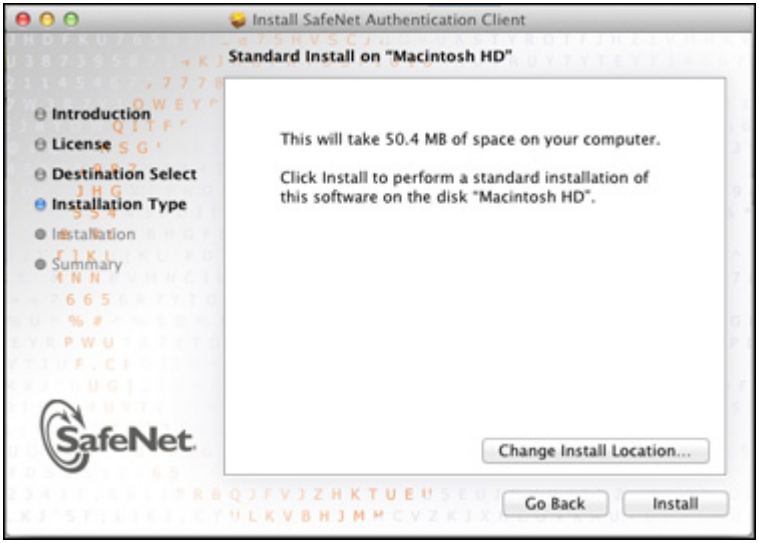
The *Software License Agreement* window opens.

4 Click **Continue**.

The agreement window opens.

5 Click **Agree** to accept the software license agreement.

The *Standard Install on Macintosh HD* window opens.



6 Click **Install**.

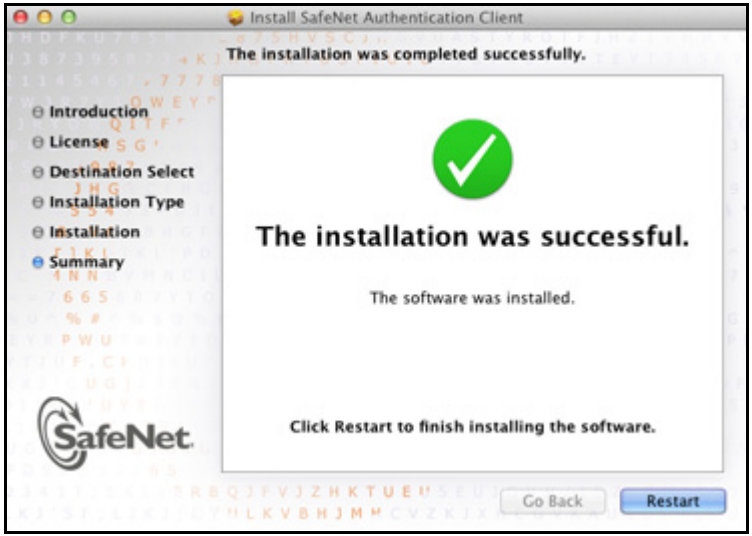
The *Authenticate* window opens.

7 Enter *Name* and *Password* and click **OK**.

NOTE You require Administrator permissions to install SafeNet Authentication Client.

SafeNet Authentication Client installs.

The *Installation completed successfully* screen opens.



8 Click **Restart**.

Mac OS X restarts.

9 Log in again to Mac OS X.

Installing from the Terminal

To install from the terminal:

1. Extract the *SafeNet Authentication Client 8.2.mpkg* file from the dmg file.
2. At the location in the terminal in which you extracted the file run `sudo installer -pkg ./SafeNet\Authentication\Client\ 8.2.mpkg/ -target /`
3. Enter your root password when prompted.

   SafeNet Authentication Client (Mac) 8.2 SP1 is installed.
4. Following installation, restart Mac OS X.

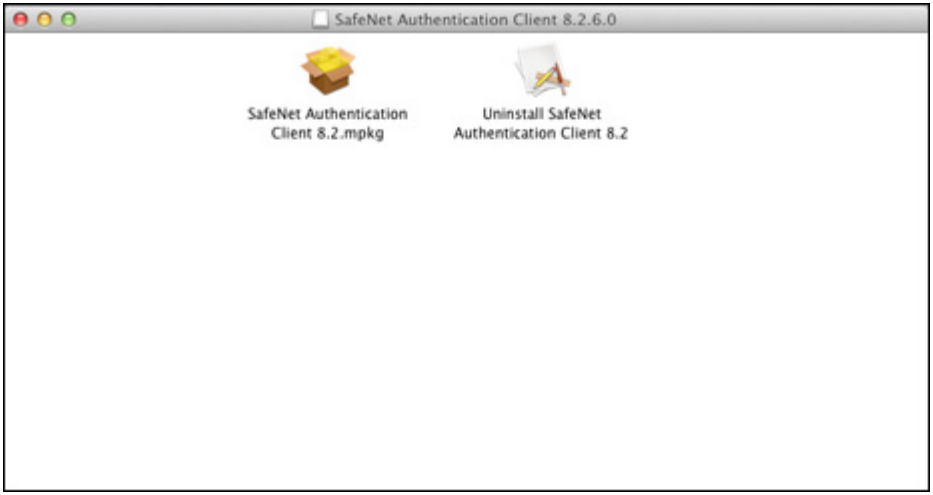
Uninstalling

NOTE Before uninstalling SafeNet Authentication Client (Mac) 8.2 SP1, make sure that SafeNet Authentication Client Tools is closed.

To uninstall SafeNet Authentication Client (Mac) 8.2 SP1

1 Double click *SafeNetAuthenticationClient.8.2.x.0.dmg* file.

A new disk image file is created in the Finder window, including an mpkg installation file and an uninstall application.



2 Click **Uninstall SafeNet Authentication Client (Mac) 8.2 SP1**.

The *Welcome to the SafeNet Authentication Client Uninstaller* window opens.

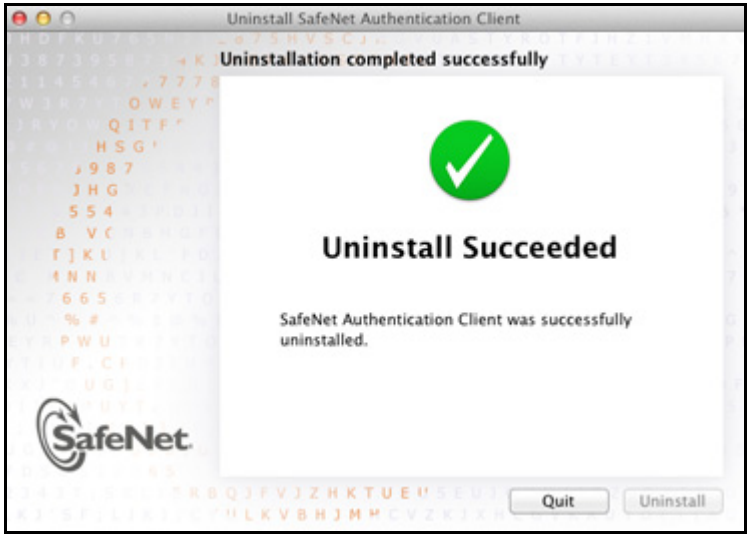
3 Click **Uninstall**.

The *Authenticate* window opens.

4 Enter **Name and Password** and click **OK**.

NOTE You require Administrator permissions to uninstall SafeNet Authentication Client (Mac) 8.2 SP1.

SafeNet Authentication Client uninstalls and the *Uninstallation completed successfully* window opens.


Uninstall SafeNet Authentication Client
Uninstallation completed successfully
Uninstall Succeeded
SafeNet Authentication Client was successfully uninstalled.
SafeNet
Quit
Uninstall

5 Click **Quit**.

Installing the Firefox Security Module

When SafeNet Authentication Client (Mac) is installed, it does not install the security module in Firefox. This must be done manually.

To install the security module in Firefox

1. Select **Settings > Advanced**.
2. On the *Encryption* tab click **Security Devices**.

   The *Device Manager* window opens.
3. Click **Load**.

   The *Load PKCS#11 Device* window opens.
4. In the *Module Filename* field enter the following string:
   /usr/local/lib/libeTPkcs11.dylib

   The *Confirm* window opens.
5. Click **OK**.

   The new security module is installed.

Installing the Thunderbird Security Module

When SafeNet Authentication Client (Mac) is installed, it does not install the security module in Thunderbird. This must be done manually.

To install the security module in Thunderbird

1. Select **Thunderbird > Preferences > Advanced**.
2. On the *Security* tab click **Security Devices**.

   The *Device Manager* window opens.
3. Click **Load**.

   The *Load PKCS#11 Device* window opens.
4. In the *Module Filename* field enter the following string:
   /usr/local/lib/libeTPkcs11.dylib

   The *Confirm* window opens.
5. Click **OK**.

   The new security module is installed.

Configuring Acrobat Security Settings

Adobe Acrobat can be configured to protect PDF documents using a .CER certificate.

NOTE The following instructions refer to Adobe Acrobat X. Different versions may use a different procedure. See Adobe documentation for more details.

To set Adobe Acrobat security settings:

1. Select the **Tools** tab.
2. Select **Protection > More Protection > Security Settings.**

   The *Security Settings* window opens.
3. Select **PKCS#11 and Tokens**.
4. If a PKCS#11 Module is not attached, click **Attach Module,** browse to the required PKCS#11 module and click **Open**.
5. Close the Security Settings window and select **Sign & Certify >More Sign & Certify > Manage Trusted Identities**.

   The *Manage Trusted Identities* window opens.
6. Click **Add Contacts**.

   The *Choose Contact to Import* window opens.

7 Click **Browse**.

The *Locate Certificate File* wind opens.

8 Browse to the required certificate (.cer) and click **Open**.

You are returned to the *Choose Contact to Import* window. The user associated with the certificate is displayed in the *Contact* box.

9 Select the contact.

The certificate is displayed in the *Certificates* box.

10 Select the certificate and click **Trust**.

The *Import Trust Settings* window opens.

11 Select the required trust settings and click **OK**.

You are returned to the *Choose Contacts to Import* window.

The *Import Completed* window confirms the import.

12 Click **OK** to close the *Import Completed* window.

NOTE To verify the security settings:

1. Select **Tools > Sign & Certify > More Sign & Certify > Manage Trusted Identities**.
   The *Manage Trusted Identities* window opens.
2. Select the contact and click **Details**.
   The *Edit Contact* window opens.
3. Select the contact and click **Show Certificate...**
   The *Certificate Viewer* Window opens.
4. Select the **Trust** tab.
   - Trusted settings for the certificate are marked with a green check-mark.
   - Non-trusted settings are marked with a red cross.

4 Revision AConfigurable Settings

This chapter provides administrator guidelines for setting configuration keys.

In this chapter:

Configuration Files

SafeNet Authentication Client installs two configuration files:

- `/etc/eToken.conf`
  Requires administrator permissions `(-rw-rw-r--)`
- `/etc/eToken.common.conf`
  Does not require administrator permissions `(-rw-rw-rw-)`

Owner: root\admin

NOTE eToken.common.conf contains settings for SafeNet eToken Virtual use only.

Configuration Files Hierarchy

To enable hierarchical priorities, up to three different versions of the `eToken.conf` configuration file can be created. For each key, the setting found in the file with highest priority determines the application's behavior.

This design simulates the SafeNet Authentication Client (Windows) registry logic.

Windows Registry	Mac Installer	File Name	Priority	File Permissions
LM/Policies	Not provided	/etc/eToken.policy.conf	1(High)	Root
CU	Automatically created by GUI	~/.eToken.conf (located in user's home directory)	2	User
LM	Provided	/etc/eToken.conf	3	Root
LM	Provided	/etc/eToken.common.conf for eToken Virtual connections		All

NOTE /etc/eToken.policy.conf can be created manually by the system administrator.

Automatic Save of Configuration Files

When SafeNet Authentication Client is uninstalled, the configuration files are saved to:

```
/etc/eToken.conf.saved
/etc/eToken.common.conf.saved
```

The saved files can then be used to copy the settings to a new installation.

eToken.conf Configuration Keys

All keys that are not related to SafeNet eToken Virtual are located in `/etc/eToken.conf`.

All SafeNet eToken Virtual keys are located in `/etc/eToken.common.conf`.

General

Key Name	Description	Value	Default
PcscSlots	Number of PC/SC slots	1-16	3
SoftwareSlots	Number of software slots	1-10	2
ClientlessHID	eToken NG Flash 5.3 Anywhere	VID_0529&PID_3004	Not available

NOTE In Mac OS X, the number of slots is determined by the `PcscSlots` and `SoftwareSlots` configuration keys described here. The *Reader Settings* window in SafeNet Authentication Client (Mac) Tools displays the number of slots that have been configured, but does not allow the user to change the settings.

AccessControl

Key Name	Description	Value	Default
OpenAdvancedView	*Advanced* button in Tools application	0 = disabled 1= enabled	1
TrayIconClearEToken	Define whether Delete token content option is displayed in Tray Icon.	0 =Option is not available in tray Icon 1 = Option is available in tray Icon	0

CertStore

Key Name	Description	Value	Default
PropagateCACertificates	Export all CA certificates on the token to the Trusted CA location	0 = disabled 1= enabled	1

InitApp

Key Name	Description	Values	Default
FIPS	FIPS Support 0 = disabled 1= enabled	0 = disabled 1= enabled	0
ShowInTray	The Quick Functions menu is displayed on the desktop	0 = not displayed 1 = displayed 2= displayed when token is inserted (does not disappear when token is disconnected)	1

Log

Key Name	Description	Value	Default
Enabled	Defines whether LOGS will be created or not.	1 = Logs will be created 0 = Logs will not be created	0

PQ

Key Name	Description	Value	Default
pqModifiable	Password quality can be changed after initialization	0 = cannot be changed 1 = can be changed	1
pqHistorySize	Number of recent passwords that cannot be repeated	>=0	10
pqMaxAge	Total number of days a password is valid 0 = no expiration	>=0	0
pqMinAge	Total number of days required before a password change 0 = none	>=0	0
pqMinLen	Minimum password length	>=4	6
pqMixChars	Mixed characters required 0 = disabled 1= enabled	0/1	1

Key Name (Cont.)	Description (Cont.)	Value (Cont.)	Default (Cont.)
pqWarnPeriod	Total number of days before expiration to display warning 0 = no warning	>=0	0

UI

Key Name	Description	Value	Default
LanguageId	UI Language (supports English only)	EN	EN
linguist	Path to Linguist application		
ExpiryAlertPeriodStart	Defines the number of days before a certificate's expiration date during which a warning message is displayed	>=0 (0 = No warning)	30
FutureAlertMessage	Defines the warning message to display in a balloon during a certificate's 'Certificate Expiration Warning Period' The message can include the following keywords: 1. $EXPIRY_DATE – the certificate's expiration date 2. $EXPIRE_IN_DAYS – the number of days until expiration	Message or empty	A certificate on your token expires in $EXPIRE_IN_DAYS days.'

Key Name (Cont.)	Description (Cont.)	Value (Cont.)	Default (Cont.)
PastAlertMessage	Defines the warning message to display in a balloon if a certificate's expiration date has passed	Message or empty	'Update your token now.'
IgnoreExpiredCertificates	Determines if expired certificates are ignored, and no warning message is displayed for expired certificates	♦ Selected - Expired certificates are ignored ♦ Not selected - A warning message is displayed if the token contains expired certificates	Not selected
AlertTitle	Defines the title to display in certificate expiration warning messages	Message or empty	'SafeNet Authentication Client'
ActionDetailedMessage	If 'Show detailed message' is selected in the 'Warning Message Click Action' setting, defines the detailed message to display	Message or empty	None
ActionWebSiteURL	If 'Open website' is selected in the 'Warning Message Click Action' setting, defines the URL to display	Message or empty	None

Key Name (Cont.)	Description (Cont.)	Value (Cont.)	Default (Cont.)
UpdateAlertMinInterval	Defines the minimum interval, in days, between certificate expiration date verifications	>0	14 days
AlertMessageClickAction	Defines what happens when the user clicks the message balloon	0 = No action 1 = Show detailed message 2 = Open website	0
ShowInTray	Determines if the Tools tray icon is displayed when SafeNet Authentication Client is launched	♦ Never show ♦ Always show	Always show
ShowBalloonEvents	Determines if a notification balloon is displayed when a token is connected or disconnected	Selected = Displayed Not selected = Not displayed	Selected
CertificateExpiryAlert	Determines if a warning message is displayed when certificates on the token are about to expire	0 = Not selected - A message is not displayed 1 = Selected - A message is displayed	0

Init

Key Name	Description	Value	Default
RSASecondaryAuthenicationMode	Can be configured in SafeNet Authentication Client Tools.		
PrivateDataCaching	Can be configured in SafeNet Authentication Client Tools.		
RSA-2048	Can be configured in SafeNet Authentication Client Tools.		
HMAC-SHA1	Can be configured in SafeNet Authentication Client Tools.		

eToken.common.conf Configuration Keys

`eToken.common.conf` contains SafeNet eToken Virtual keys.

Key Name	Description	Value	Default
FileName(slot0)	File name with full path		

5 Apple Keychain

Apple Keychain is Apple Computer's password management system in Mac OS X. Keychain Access is a Mac OS X application that allows the user to access the Apple Keychain and configure its contents.

SafeNet Authentication Client (Mac) provides a plug-in to support integration with Mac OS X Keychain Access. The plug-in is installed during SafeNet Authentication Client (Mac) installation.

In this chapter:

Features Supported by Keychain Access

The SafeNet Authentication Client (Mac) Keychain Access integration supports the following features:

- Upload of certificates from the token to Keychain Access.
- Encryption and Decryption - by uploading certificates from a token to Keychain, they become available for applications, such as Mail, that can use the certificates to encrypt and decrypt mail messages.

Keychain Access Limitations

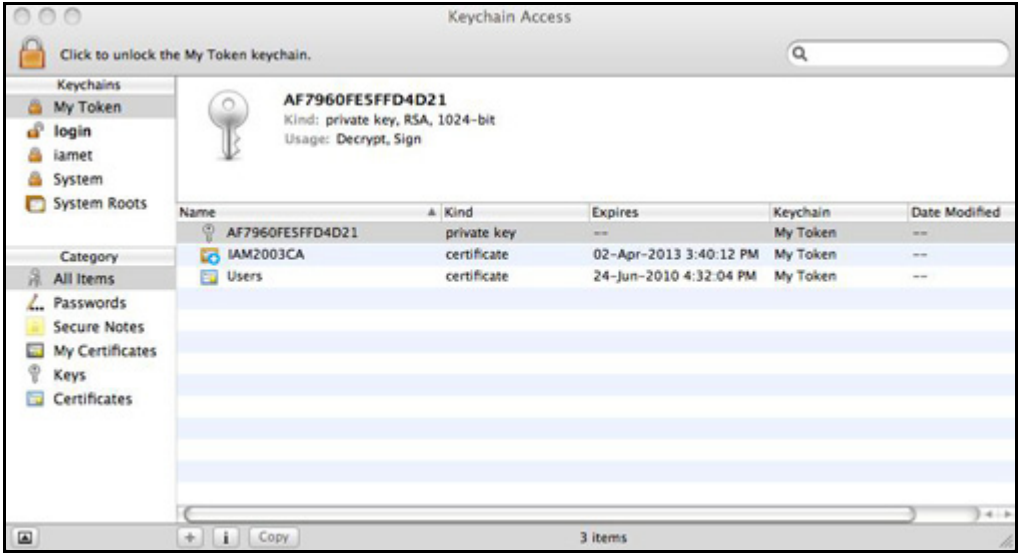
The following limitations apply when working with Keychain Access and SafeNet tokens.

- Keychain cannot be used to create new certificates. It can only upload certificates already located on the token.
- Change token password is not supported (however, it can be changed using SafeNet Authentication Client).
- Smartcards are not supported.
- It is not possible to import a certificate from a file to a token (however, certificates can be imported using SafeNet Authentication Client (Mac) Tools).
- The Keychain does not support RSA key generation from a token.

Displaying Token in Keychain Access

When you launch Keychain Access, you see a list of all the items in your Keychain, including information about each item's name, kind, creation date, and modification date.

When you insert a token, the device is displayed in the *Keychains* list.



To display token contents:

- In the *Keychains* list on the left of the window, select token, then select an item from the *Category* list.

  The details are displayed in the right section of the screen.

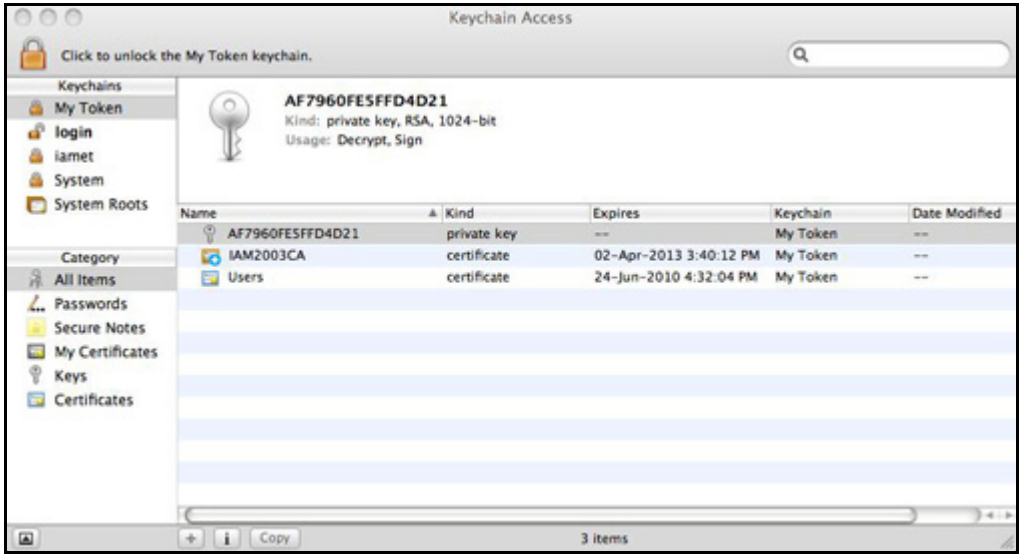
> **TIP** For details about performing additional functions with Keychain Access, refer to Mac OS X documentation.

Configuring Mac Keychain to Work with SSL and Secure Mail (S/MIME)

Mac Keychain must be configured to enable Safari to work with an SSL Connection and to enable encryption and decryption of emails.

To enable Mac Keychain to work with SSL and Secure Mail (S/MIME):

1 Open the *Keychain Access* window.



2 Double click on the root CA.

The window with the certificate details opens

3 Click on **Trust** to expand the section.

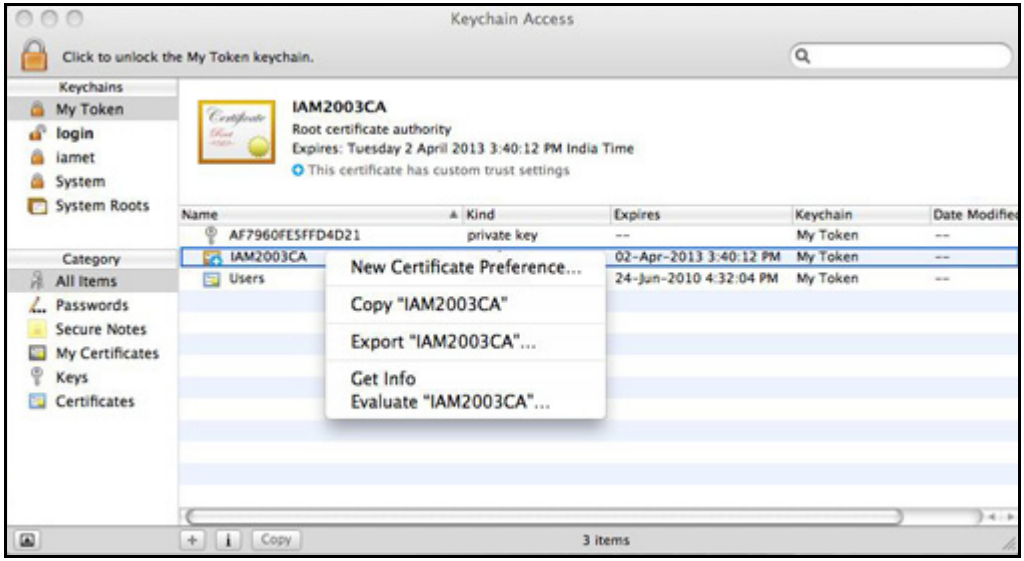
4 Set *Secure Socket Layer (SSL)* and/or *Secure Mail (S/MIME)* to **Always Trust**

5 Close the window.

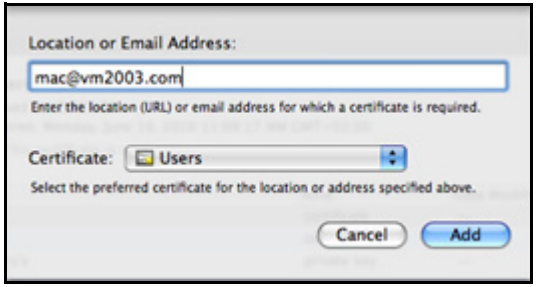
You are returned to the Keychain Access window.

The root CA certificate is now trusted for SSL and S/MIME operations.

6 Right click on the Users Certificate and select **New Certificate Preferences**.



The *Location or Email Address* window opens.



7 In the Certificate field, select the required certificate.

8 Do one of the following and click **Add**:

- For S/MIME, enter the email address of your mail account
- For SSL, enter the URL of your secured site.

The item is added to the *login* Keychain.

NOTE You must configure SSL for each required secured website.

If you configured Secure email (S/MIME), you will now be prompted to enter the token password when signing and sending an email or when decrypting an encrypted email.

If you configured SSL for your secured sites, when logging on with Safari you will be prompted for the token password.