

ePassNG User Manual

Ver. 1.0

EnterSafe will do their best to keep the content of this document as accurate as possible. But EnterSafe will not take the responsibilities for any direct or indirect loss that may be caused by this document. The content of this document will be amended along with the updating of the product without notification.

Editing History:

Date	Version	Edition
Jan 29th, 2008	1.0	2nd Edition

Software Developer's Agreement

IMPORTANT - READ CAREFULLY: This Software Developer's Agreement (SDA) is a legal agreement between you (either an individual or a single entity) and EnterSafe Corporation for the software that accompanies this SDA, which includes computer software and may include associated media, printed materials, "online" or electronic documentation, and Internet-based services ("Software"). YOU AGREE TO BE BOUND BY THE TERMS OF THIS SDA BY INSTALLING, COPYING, OR OTHERWISE USING THE SOFTWARE. IF YOU DO NOT AGREE, DO NOT INSTALL, COPY, OR USE THE SOFTWARE; YOU MAY RETURN IT TO YOUR PLACE OF PURCHASE FOR A FULL REFUND, IF APPLICABLE..

1. GRANT OF LICENSE. EnterSafe grants you the rights described in this SDA provided that you comply with all terms and conditions of this SDA.

1.1 EnterSafe grants you a limited, nonexclusive license to use the Software, and to make and use copies of the Software, for the purposes of designing, developing and testing your software applications.

1.2 EnterSafe grants you to merge and link the Software with other programs for the sole purpose of protecting those programs in accordance with the usage described in the Developer's Guide.

1.3 You may make archival copies of the Software. If EnterSafe makes a request via public announcement or press release to stop using the copies of the Software, you will comply immediately with this request.

2. LIMITATIONS ON REVERSE ENGINEERING, DECOMPILE, AND DISASSEMBLY. You may revise, reverse engineer, decompile, disassemble, enhanced or otherwise modified the Software, except only to the extent that such activity is expressly permitted by applicable law notwithstanding this limitation.

3. NO RENTAL OR COMMERCIAL HOSTING. You may not rent, lease, lend or provide commercial hosting services with the Software.

4. LIMITATION OF LIABILITY AND REMEDIES. Notwithstanding any damages that you might incur for any reason whatsoever (including, without limitation, all damages referenced herein and all direct or general damages in contract or anything else), the entire liability of EnterSafe and any of its suppliers under any provision of this SDA and your exclusive remedy hereunder shall be limited to the greater of the actual damages you incur in reasonable reliance on the Software up to the amount actually paid by you for the Software.

5. DISCLAIMER OF WARRANTIES. To the maximum extent permitted by applicable law, EnterSafe and its suppliers provide the Software and support services (if any) AS IS AND WITH ALL FAULTS, and hereby disclaim all other warranties and conditions, whether express, implied or statutory, including, but not limited to, any (if any) implied warranties, duties or conditions of merchantability, of fitness for a particular purpose, of reliability or availability, of accuracy or completeness of responses, of results, of workmanlike effort, of lack of viruses, and of lack of negligence, all with regard to the Software and the provision of or failure to provide support or other services, information, software, and related content through the Software or otherwise arising out of the use of the Software. ALSO, THERE IS NO WARRANTY OR CONDITION OF TITLE, QUIET ENJOYMENT, QUIET POSSESSION, CORRESPONDENCE TO DESCRIPTION, OR NON-INFRINGEMENT WITH REGARD TO THE SOFTWARE.

6. RESERVATION OF RIGHTS AND OWNERSHIP. EnterSafe reserves all rights not expressly granted to you in this SDA. The Software is protected by copyright and other intellectual property laws and treaties. EnterSafe own the title, copyright, and other intellectual property rights in the Software.

7. TERMINATION. This SDA is effective until terminated. Upon any violation of any of the provisions of this SDA, rights to use the Software shall automatically terminate and the Software must be returned to EnterSafe or all copies of the Software destroyed. You may also terminate this SDA at any time by destroying all copies of the Software in your possession or control. The provisions of paragraphs 2, 3, 4, 5 and 6 will survive any termination of this SDA.

CE Attestation of Conformity



The equipment complies with the principal protection requirement of the EMC Directive (Directive 89/336/EEC relating to electromagnetic compatibility) based on a voluntary test.

This attestation applies only to the particular sample of the product and its technical documentation provided for testing and certification. The detailed test results and all standards used as well as the operation mode are listed in

Test report No. 70407310011

Test standards: EN 55022/1998 EN 55024/1998

After preparation of the necessary technical documentation as well as the conformity declaration the CE marking as shown below can be affixed on the equipment as stipulated in Article 10.1 of the Directive. Other relevant Directives have to be observed.

FCC certificate of approval



This Device is conformance with Part 15 of the FCC Rules and Regulations for Information Technology Equipment.

USB



This equipment is USB based.

WEEE



Dispose in separate collection.

Technical Terms and Abbreviations

Terms	Explanation
PKCS#11Interface	Software programming interface which is presented by RSA (www.rsasecurity.com). It maps cryptographic devices into a type of universal logical model, i.e. Cryptographic Token, for the usage of system's upper applications. This design could achieve the device independent and resource sharing.
CryptoAPI interface (CAPI for short)	Cryptographic operation interface presented by Microsoft. It provides device independent or software implemented cryptographic algorithms' encapsulation, which is easy to use for developers to design their own PKI applications, including data encryption, certificate verification and digital signature, under Windows® platform.
Token	General name of all cryptographic devices, such as smartcards, devices having passwords and certificates storage functionalities etc.
USB Token	Cryptographic devices with USB port. Portable and easy to use.
ePass2000_FT12	Portable cryptographic device integrates smartcard and USB port, which is released by Feitian. It inherits the advantages of smartcard device and also portable. Supporting PKI applications.
ePass2000_FT11 (ePass2k Without Driver)	A non-driver USB Token released by Feitian. It has the same functionalities like ePass2000_FT12.
ePassNG (ePass Next Generation)	A new generation middleware framework product released by Feitian, supporting Feitian's ePass series products. Easy to be extended with new hardware supporting. Supports PKI applications.
TSP (Token Service Provider)	Abstract hardware layer in ePassNG framework. It provides common I/O interfaces for all kinds of devices. This design could provide a certain extent against hardware's difference.

Catalog

Chapter 1 ePassNG Introduction	1
1.1 ePassNG framework structure	1
1.2 ePassNG features	2
Chapter 2 EnterSafe PKI Manager	4
2.1 Precondition.....	4
2.2 Profile	4
2.2.1 Interface without Token plugged in.....	4
2.2.2 Interface with Token plugged in.....	5
2.2.3 Menu of EnterSafe PKI Manager	5
2.2.4 “Operation” Menu	6
2.2.5 “View” Menu.....	6
2.2.6 Right-Click Menu in Slot Tree	7
2.2.7 Information Displayed After Plug in Token	7
2.2.8 Information Displayed When No Token is Plugged in	8
2.3 Check Slot List Information	8
2.4 Check Token Information	9
2.5 Login	9
2.6 Change User PIN.....	9
2.7 Change Token Name.....	10
2.8 Change SO PIN	10
2.9 Unblock Token	10
2.10 Initialize Token	11
2.11 Data Management in Un-login State.....	12
2.12 Data Management in Login State	12
2.13 Import Certificate	13
2.14 Export Certificate	14
2.15 View Data Information	15
2.16 Delete Data	16

Chapter 1 ePassNG Introduction

ePassNG is a new generation platform independent data security products framework.

It mainly provides hardware supports to upper layer of PKI applications. The certificates, key pairs and other classified information are all stored in ePassToken. ePassNG provides standard PKCS#11 and CryptoAPI programming interfaces to support standard PKI applications. It is easy to be redeveloped by ISVs (Independent Software Vendors) for their end users. Moreover, because of the simple framework structure, the hardware providers could integrate their hardware into ePassNG framework by implemented a TSP (Token Service Provider). So it is easy to integrate their hardware into PKI framework..

This chapter contains the following topics:

- ePassNG framework structure
- ePassNG features

1.1 ePassNG framework structure

ePassNG provides standard PKCS#11 and CryptAPI programming interfaces to upper PKI applications. ISVs could develop their own applications based on these interfaces. Moreover, interfaces provided by ePassNG could be seamlessly integrated with any standard PKI applications by only making a little configuration..

The ePassNG's framework structure is demonstrated in figure 1-1.

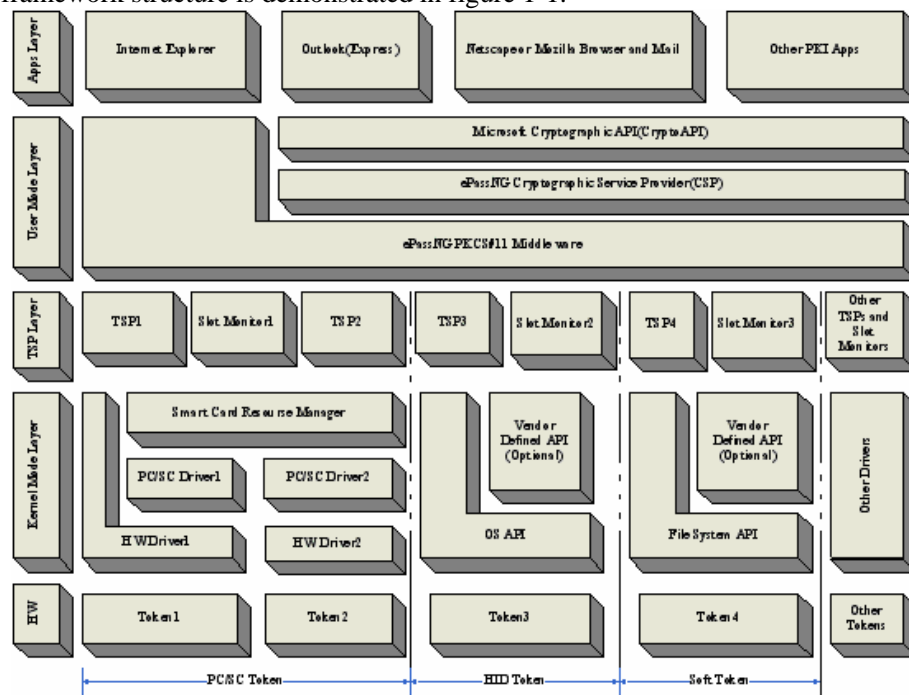


Figure 1-1

From figure1-1, it could be seen that ePassNG framework contains five layers. They are hardware layer, core driver layer, abstract hardware layer, application interface layer and application layer.

Hardware Layer

This layer is the infrastructure of the entire structure. It contains various tokens including their hardware

circuitries, firmware programs and wires. Any tokens compliant with PC/SC standard can be supported by this hardware layer, such as ePass1000, ePass2000, ePass3000, ePass3000ND, various other smartcard reader, smartcard combined third-party USB-Keys. Their common feature is the token must be able to be controlled by operation system's Smart Card Resource Manager. Tokens could also be HID (Human Interface Device) devices, such as ePassND (non-deriver USB-KEY release by Feitian), USB flash disk and even files in hard disk. Different kinds of tokens or multiple tokens of the same type could work together.

Core Driver Layer

Core driver layer manage data communications between client computer and hardware layer, process access request from TSP layer. To PC/SC tokens, this layer works like hardware driver, PC/SC driver and operation system's Smart Card Resource Manager. For HID token, this layer could be treated as operation system's build-in drivers. For file system token, this layer functions like operation system's file operation system.

Abstract Hardware Layer

Abstract hardware layer provides abstract interfaces to application interface layer. Communications between computer and different devices (including token) use the same interfaces provided by this layer. This design effectively hides the difference among hardware. The software implementation of this layer is called TSP (Token Service Provider).

Application Interface Layer

Application interface layer provides the standard implementations of PKCS#11 interface and MS CryptoAPI interface for upper layer of PKI applications.

Moreover, PC/SC application interface compliant with Microsoft® PC/SC standard is also provided. Developers could develop applications with their familiar PC/SC function set. This interface is platform independent and could be applied to any platforms compliant with ePassNG.

Application Layer

Application layer includes various ePassNG applications and other applications. Because ePassNG provides different types of standard programming interfaces, it is compatible with most existing applications and moreover, developers could use their familiar interface set to design their own applications.

1.2 ePassNG features

1. Platform Independent

Currently, ePassNG supports Windows, Redhat Linux, Mandrake Linux, Mac OS X, Knoppix Linux platforms. Its core library uses the same codes (other than some software use different codes for different platform) so as to be a real platform independent product. More platforms will be supported in future.

2. Interface Standard

ePassNG provides standard PKI interface to its upper layer applications, including RSA PKCS#11 and MS CryptoAPI (this interface could only be applied under Windows platforms). All the applications using either interface could use ePassNG to store certificates and key pairs, processing cryptographic operations. For the extension of hardware token, we provides standard interface to third-party vendors for their hardware implementations.

3. Good Compatibility

ePassNG is fully compatible with Feitian's ePass3000 and ePass3000ND hardware products. Previous certificates and key pairs are still applicable with ePassNG. Moreover, the certificates applied by ePassNG in one platform could still be used in another platform. This enables user use unique identification crossing different platforms.

4. Supporting Various Tokens

ePassNG's open framework design makes it be able to support different kinds of tokens, supporting them working at the same time. User could choose any token according to their usage. If TSP is implemented, ePassNG could even support various virtual tokens such as U-disk, disk files, floppy disk and CD-ROM etc.

5. Easy for Extension

Third-party vendors could integrate their products into ePassNG framework by signing the related agreement with EnterSafe. Using TSP developing interface provided by EnterSafe, vendors just need to make a little modifications, or even no modification, for the integration.

6. Growing Up Everyday

Feitian's ePass3000 and ePass3000ND products have obtained many domestic and international authoritative qualifications, including CheckPoint, CFCA etc. Referring to those successful products' advantages, ePassNG becomes more stable and secure. It will gain more and more qualifications step by step.

Chapter 2 EnterSafe PKI Manager

The interface and operating method of EnterSafe PKI Manager are similar under different system platforms so as to provide more convenience to users. Furthermore, ePass1000ND, ePass2000_FT11, ePass2000_FT12, ePass3000ND, ePass3000 and ePass3000OEM share the same Manager.

There are two versions of EnterSafe PKI Manager: administrator edition and end user edition. Administrator edition provides more functions, such as “Initialize Token”, “Unblock Token” and “Change SO PIN”.

To let you know how to use the administrator edition of EnterSafe PKI Manager with ePass2000_FT12 product under Linux system, this chapter will explain the following functions(This document describes with Ubuntu 5.10):

- Initialize Token (Only applicable for administrator edition)
- Unblock Token (Only applicable for administrator edition)
- Change SO PIN(Only applicable for administrator edition)
- Login (Verify user PIN)
- View Token and Slot Information
- Change User PIN
- Change Token Name
- Manage Token Data

2.1 Precondition

Because EnterSafe PKI Manager is based on ePassNG middleware and will access hardware token, before using EnterSafe PKI Manager, please confirm the ePassNG products (including middleware) have been installed properly.

2.2 Profile

2.2.1 Interface without Token plugged in

Run EnterSafe PKI Manager, system will display the interface like figure 2-1:

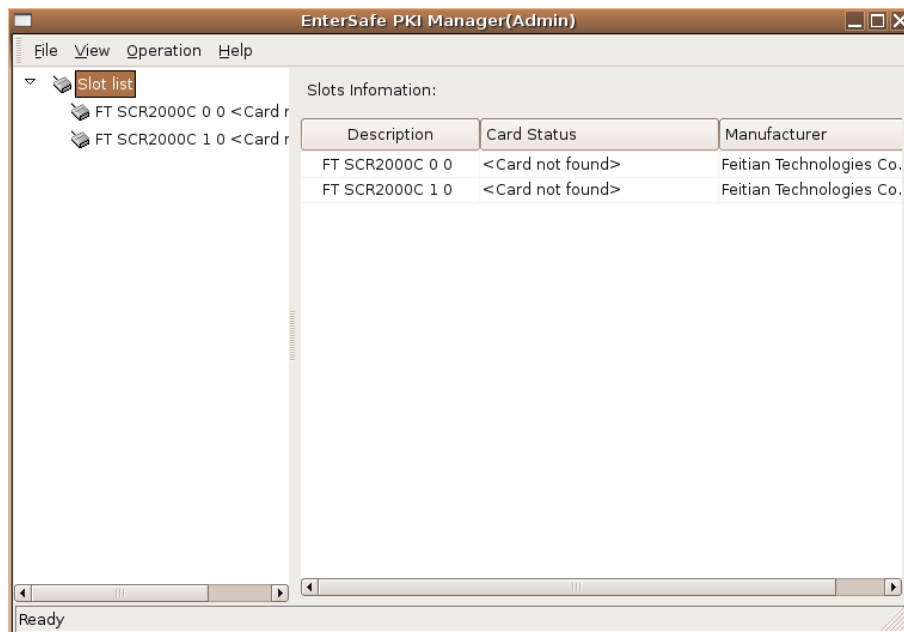


Figure 2-1 Interface without Token plugged in

Left column lists all the supported slots. Right column briefs the basic information of these slots.

2.2.2 Interface with Token plugged in

Plugging an USB token named “ePass Token” in USB port of the computer, the EnterSafe PKI Manager will recognize the basic information of the token and display the interface like figure2-2:

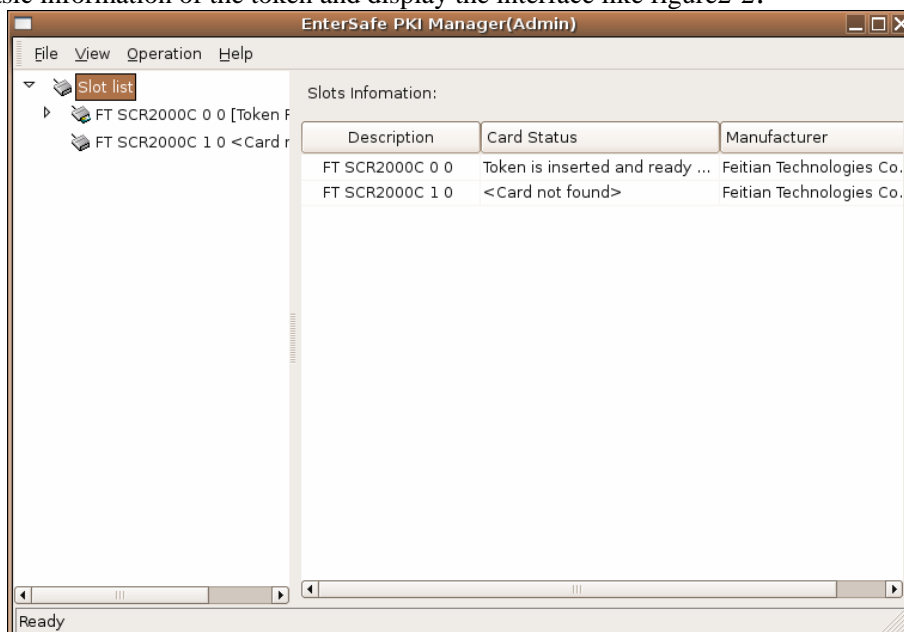


Figure2-2 Interface with Token plugged in

2.2.3 Menu of EnterSafe PKI Manager

Like figure 2-3:

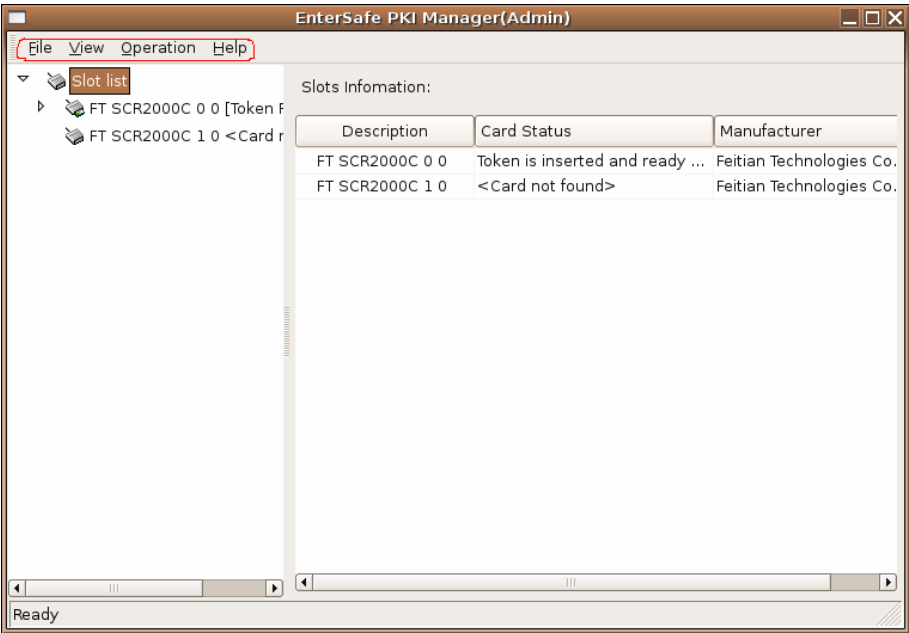


Figure2-3 menu of the Manager

The main menu includes: File (exit the Manager and select a language), view (Check information of the slot), Operation (Operations about the slot) and Help (Version Information).

2.2.4 “Operation” Menu

Detailed options refers to figure 2-4:

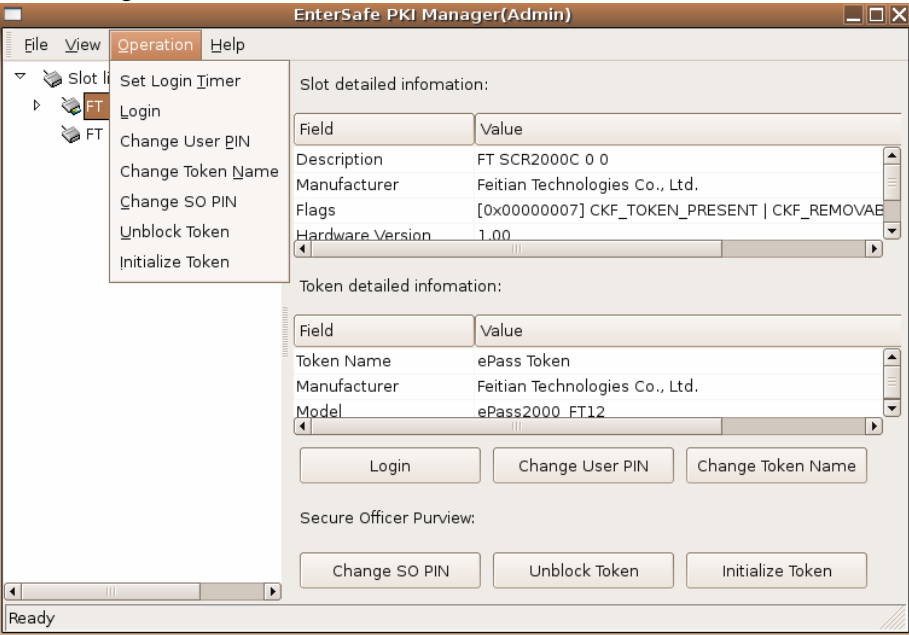


Figure 2-4 Options in “Operation” Menu

Drop down menu lists the applicable options.

2.2.5 “View” Menu

Detailed options refers to figure 2-5:

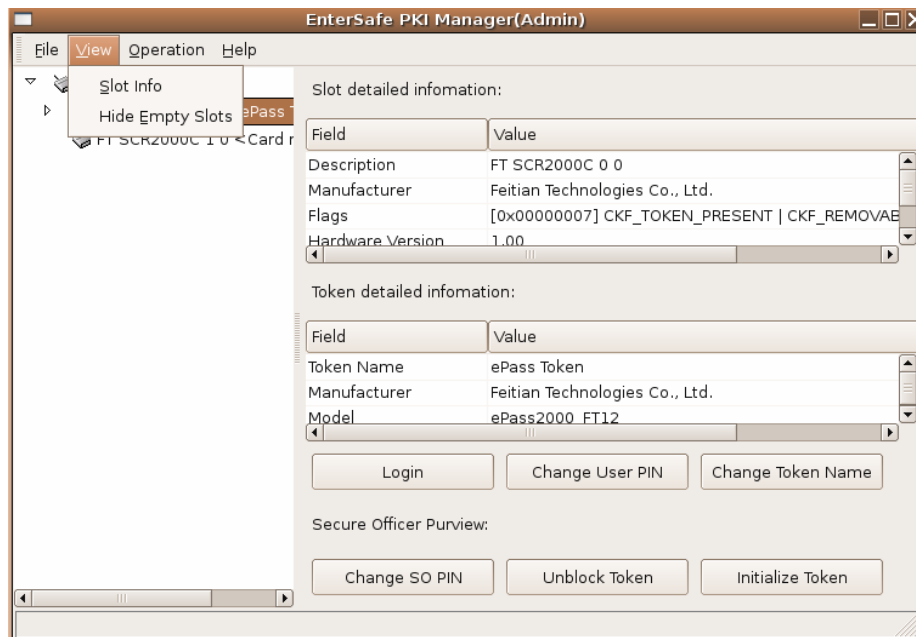


Figure 2-5 Options in “View” Menu

2.2.6 Right-Click Menu in Slot Tree

Right-click on any slot listed in right-side slot tree, system will prompt the menu shown in figure 2-6:

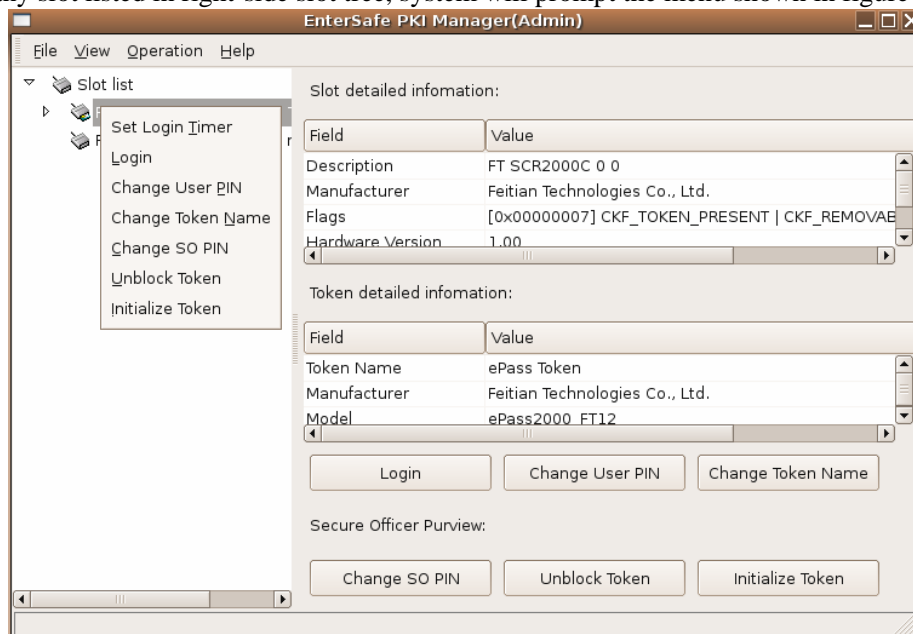


Figure 2-6 Right-Click Menu in Slot Tree

Operations includes “Set Login Timer”, “Login”, “Change User PIN”, “Change Token Name”, “Change SO PIN”, “Unblock Token” and “Initialize Token”.

2.2.7 Information Displayed After Plug in Token

Click on any slot, its information and related possible operations will be displayed in right hand side, like figure 2-7:

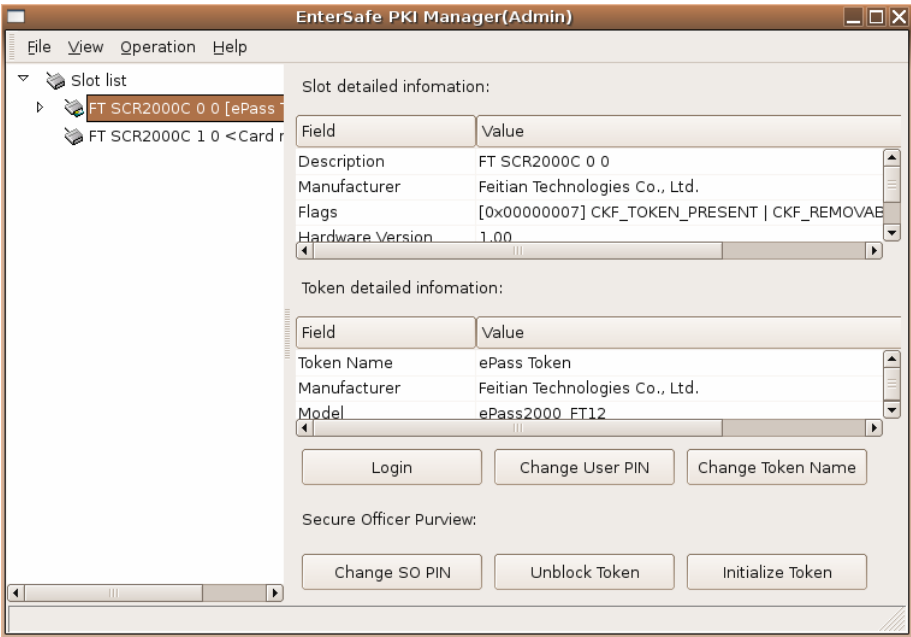


Figure 2-7 Information Displayed After Plug in Token

Information displayed in right includes the slot status, token detailed information and all of the possible operation buttons. Buttons which are currently not applicable will be disabled.

2.2.8 Information Displayed When No Token is Plugged in

Click any empty slot, the information displayed looks like figure 2-8:

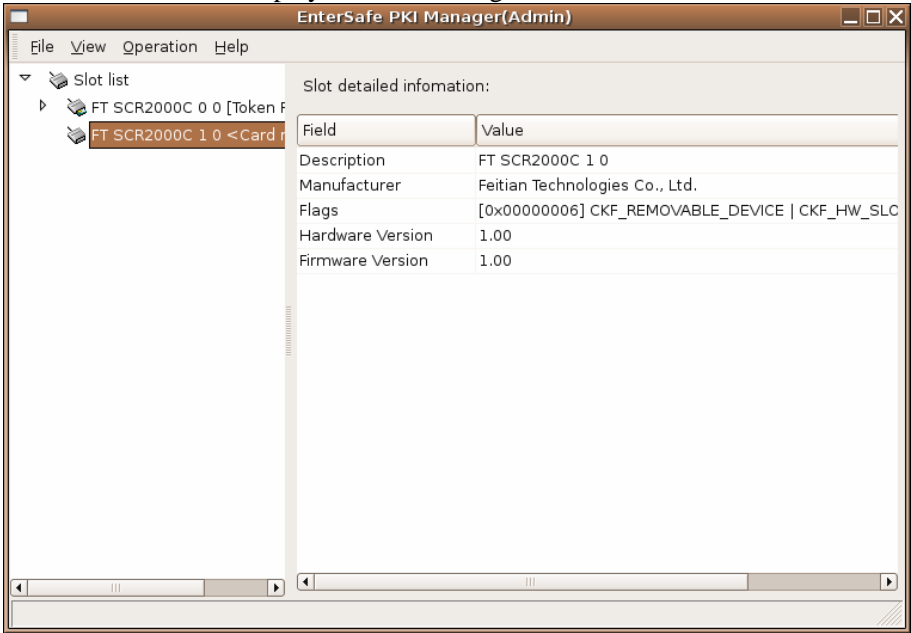


Figure 2-8 Information Displayed When No Token is Plugged in

ePassNG supports multiple tokens. Choose any token, user could process operations demonstrated in figure 2-4, figure 2-6 and figure 2-7.

2.3 Check Slot List Information

Click left hand side of the slot list or select “view→Slot info”, system will display the slot’s information like

figure 2-1.

2.4 Check Token Information

Click left hand side of the slot list. The related information will be displayed in the right hand side. If a token is plugged in and it will display the token's detailed information like figure 2-7. If slot is empty, the information about the slot will be displayed like figure 2-8.

2.5 Login

Before login, user could only view public information of the token. Private information could only be retrieved after user login with the correct PIN number. Click on "Login", system will prompt the Token login dialog box like figure 2-9:

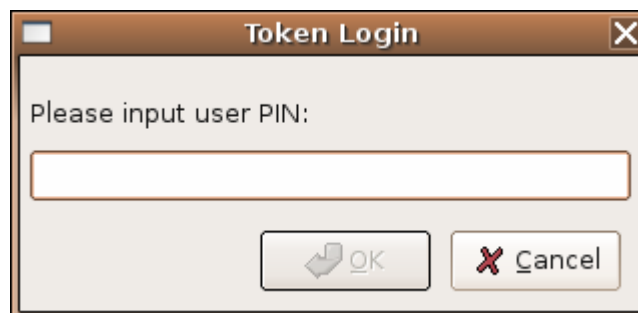


Figure2-9 Login Dialog

After inputted correct user PIN, click "OK" to login the PKI Manager.

2.6 Change User PIN

The initial user PIN of the token is "1234", and you are recommended to change user PIN. By clicking "Change User PIN". System will prompt the dialog box like figure 2-10:

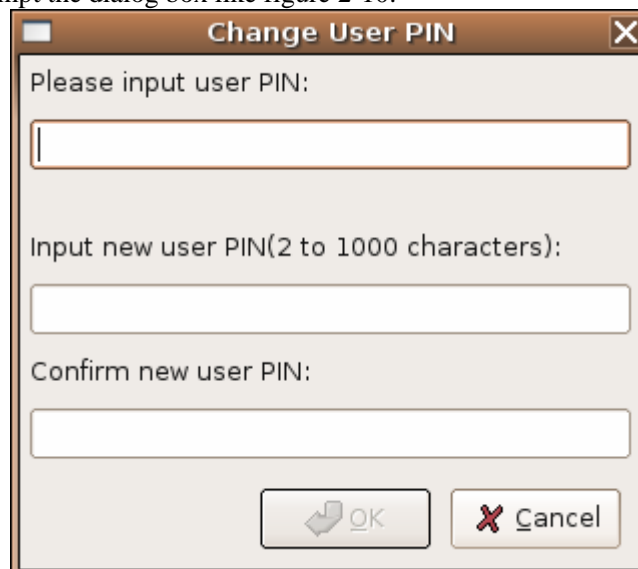


Figure2-10 Change User PIN Dialog

When changing the user PIN, user must input the old user PIN, input the new user PIN and confirm the user PIN. Then click "OK" to process the operation.

2.7 Change Token Name

Generally, token is distinguished by their serial number. But the serial number is not significative and hard to be remembered. Token name could also be used to identify a token. It could be specified by user's will.

Click "Change Token Name", system will prompt the dialog like figure 2-11:

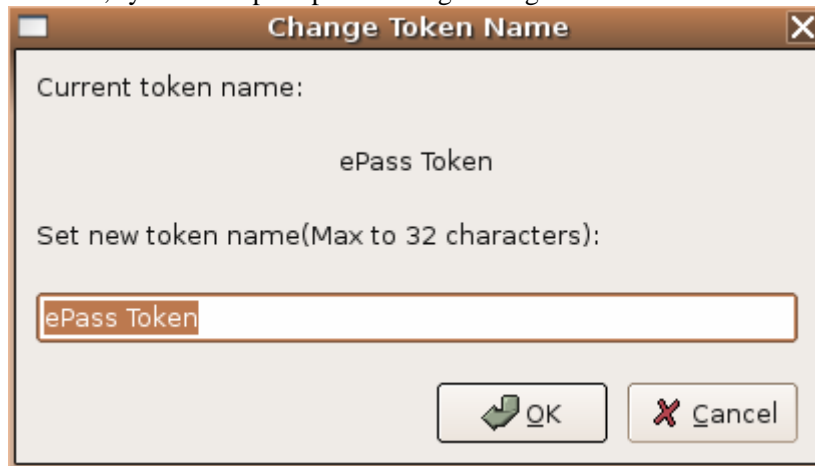


Figure 2-11 Change Token Name Dialog

Input any name for the token and click "OK", system will refresh and display the token's new name.

2.8 Change SO PIN

The initial SO PIN of the token is "rockey". User could change it by click "Change SO PIN". System will prompt the dialog box like figure 2-12:

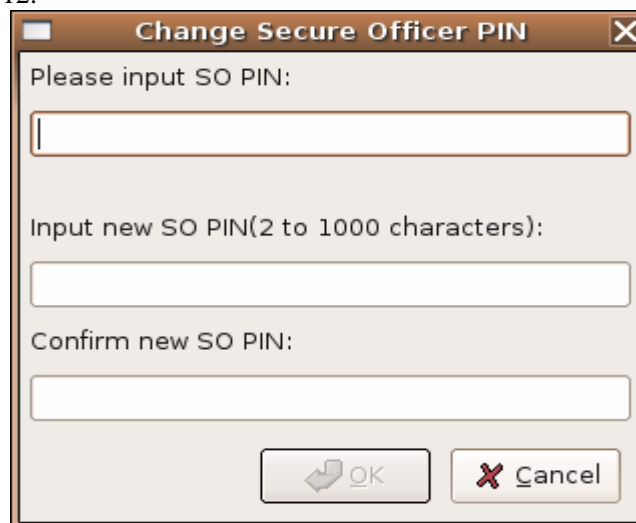


Figure 2-12 Change SO PIN Dialog

When changing the SO PIN, user must input the old SO PIN, input the new SO PIN and confirm the new SO PIN. Then click "OK" to process the operation.

2.9 Unblock Token

When user failed to input the correct user PIN over a certain number, user PIN will be blocked. Even user input the correct user PIN again, token could not be accessed. Administrator must unblock the token by click "Unblock Token". System will prompt the dialog box like Figure 2-13:

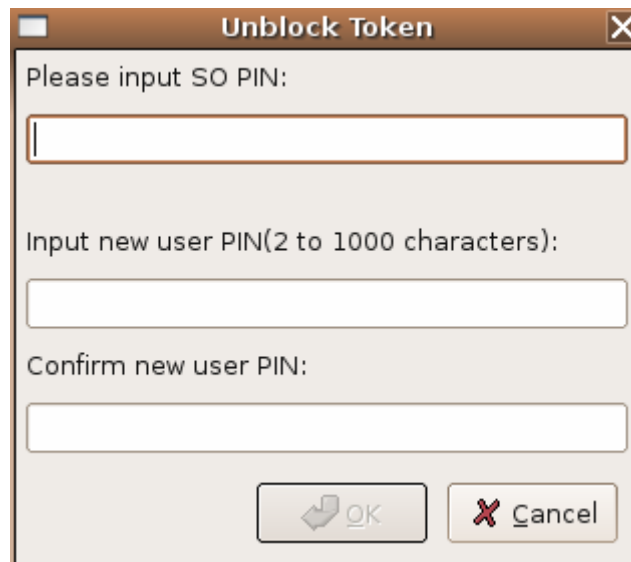
A dialog box titled "Unblock Token" with a close button (X) in the top right corner. It contains three input fields: "Please input SO PIN:" (empty), "Input new user PIN(2 to 1000 characters):" (empty), and "Confirm new user PIN:" (empty). At the bottom, there are two buttons: "OK" with a green arrow icon and "Cancel" with a red X icon.

Figure 2-13 Unblock Token Dialog

To unblock the Token, SO PIN must be provided. The user PIN must be reset. After inputting and confirming the new user PIN, click “OK” to unblock the token. After unblocked the token, user could login with the new user PIN.

2.10 Initialize Token

This operation will clear all the information in the token and reset the token hardware into PKI operation hardware token.

WARNING: ALL the information in the token including PKI certificates, public and private keys and user data will be totally removed after you initialize the token.

Click “Initialize Token”, system will prompt the dialog like figure 2-14:

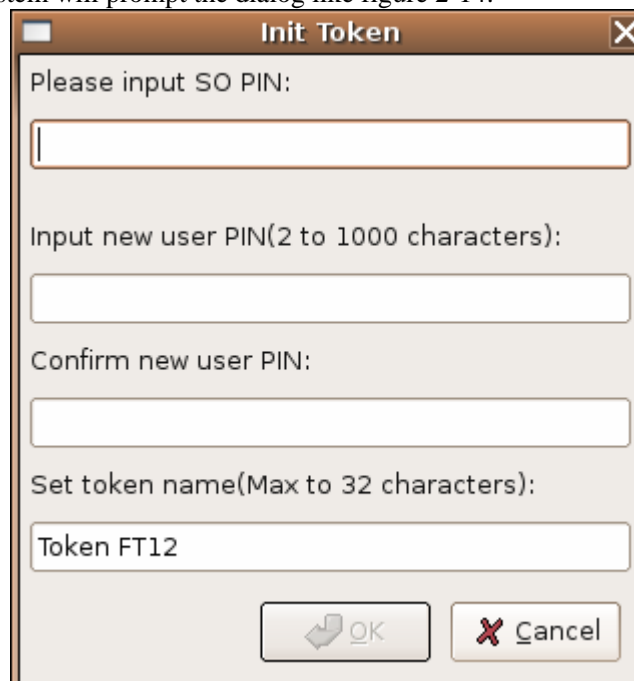
A dialog box titled "Init Token" with a close button (X) in the top right corner. It contains four input fields: "Please input SO PIN:" (empty), "Input new user PIN(2 to 1000 characters):" (empty), "Confirm new user PIN:" (empty), and "Set token name(Max to 32 characters):" (containing the text "Token FT12"). At the bottom, there are two buttons: "OK" with a green arrow icon and "Cancel" with a red X icon.

Figure 2-14 Initialize Token Dialog

Initializing token needs administrator input SO PIN. user PIN and the token name must be reset. All the data in the token will be erased. After successful initialization, system will refresh and change to token’s login state.

2.11 Data Management in Un-login State

In the left area of the PKI Manager, each token has a data management function. Click it in un-login state, system will display token's public information and related possible operations in right hand side like figure 2-15:

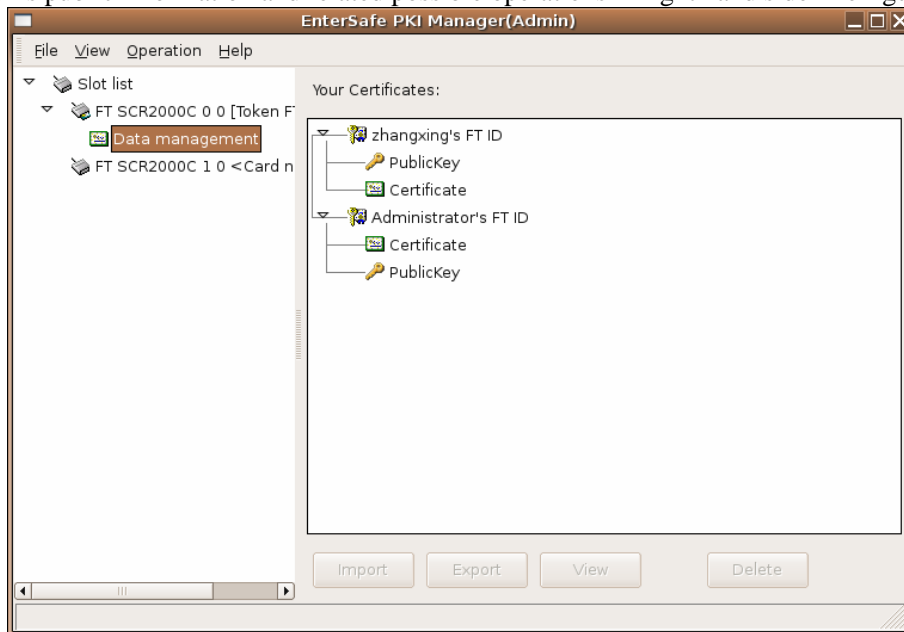


Figure 2-15 Data Management in Un-login State

2.12 Data Management in Login State

After user login the token, system interface looks like figure 2-16:

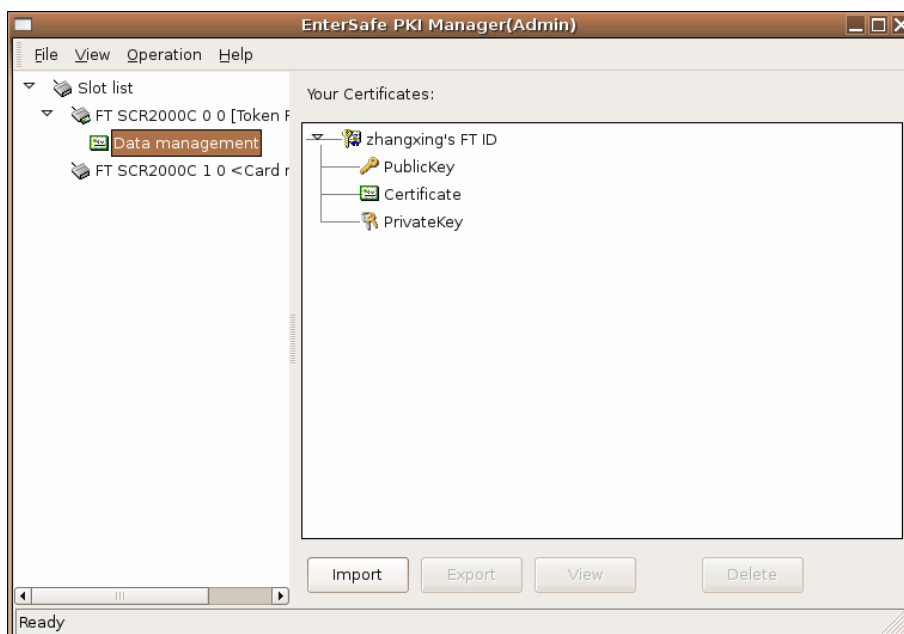


Figure 2-16 Data Management in Login State

After login, user could view both token's public information and private information. "Import" button will only be enabled after user login. "Export" button will be enabled when a certificate is selected. "View" button is always enabled except for contain name. "Delete" button is always enabled when login.

After user login, “Login” button will be disabled, demonstrating the token has changed into login state, like figure 2-17:

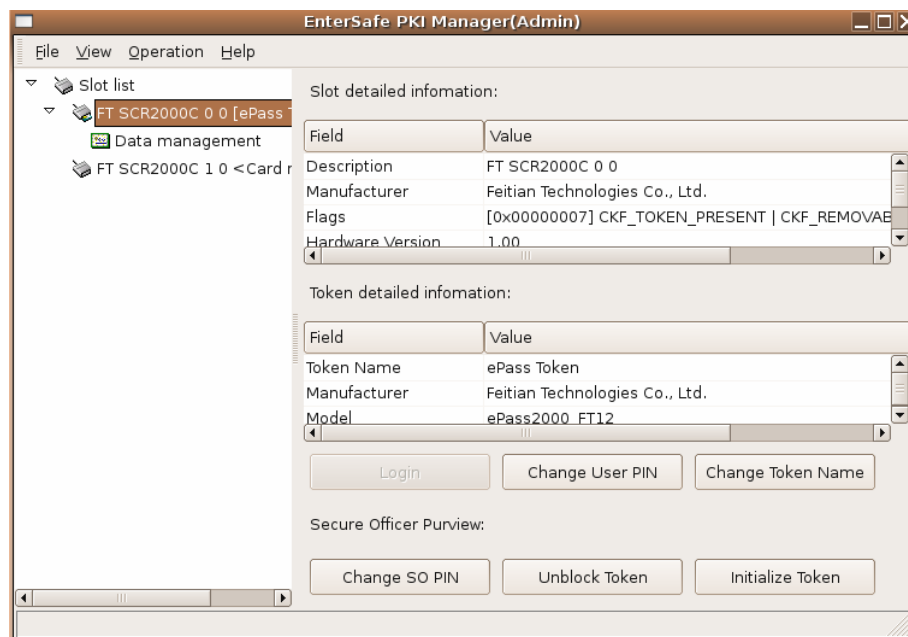


Figure 2-17

2.13 Import Certificate

When user wants to import P12, P7B, CER, PFX, CRT certificate into the token, click “Import”. System will prompt the dialog box like figure 2-18:

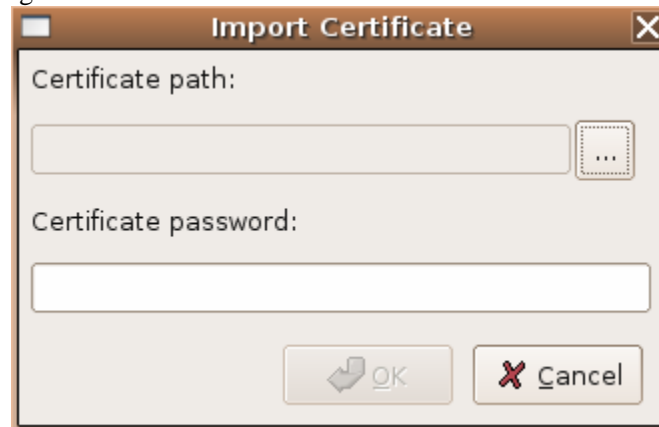


Figure 2-18 Import Certificate Dialog box 1

Only P12 and PFX certificate needs to confirm certificate password. Importing other types of certificate, system will disable the password gap. Click “...” button and choose the certificate that needs to be imported, confirm the correct certificate access password and click “OK”, system will import the certificate into the token and refresh automatically, like figure 2-19 and figure 2-20:

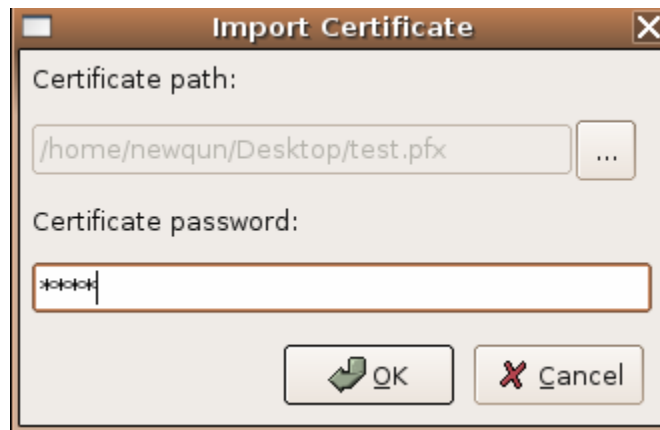


Figure 2-19 Import Certificate Dialog 2

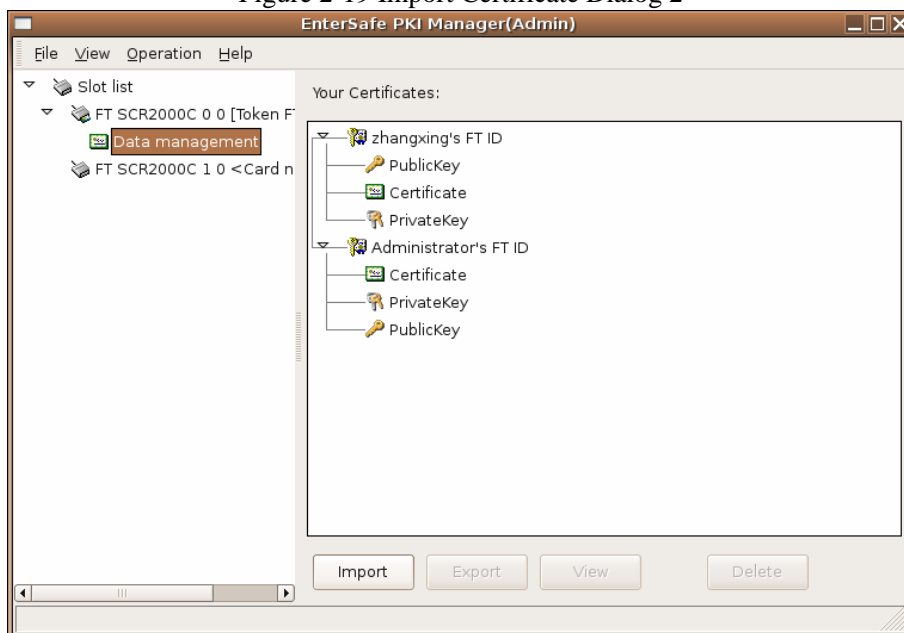


Figure 2-20 Data Management Interface after Certificate Imported

2.14 Export Certificate

When user wants to export a certificate, click on “Export” and the system will prompt the dialog like figure 2-21:

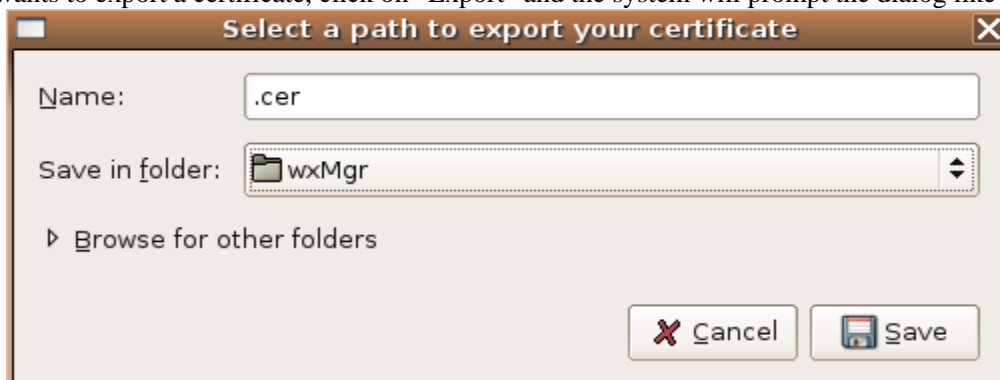


Figure 2-21 Certificate Export Dialog

Click “Browse for other folders” button and choose the directory. Then click “OK” to export the certificate.

2.15 View Data Information

When user wants to view detailed information of certificate, public key, private key and other data, user could select a special item and click “View”. System will prompt the dialog box like figure 2-22:

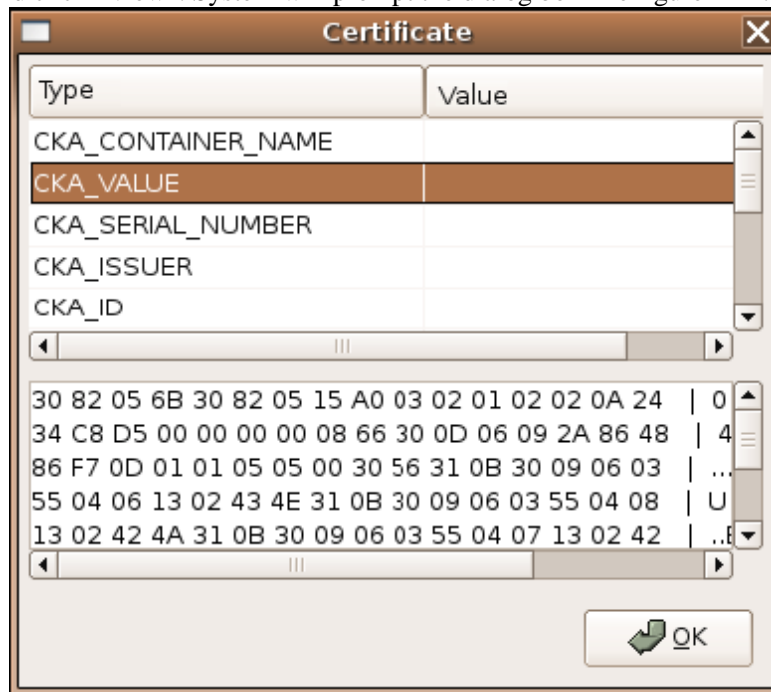


Figure 2-22 View Certificate Information Dialog

To view other data information (such as public key, private key or other data), system will prompt the dialog box like figure 2-24:

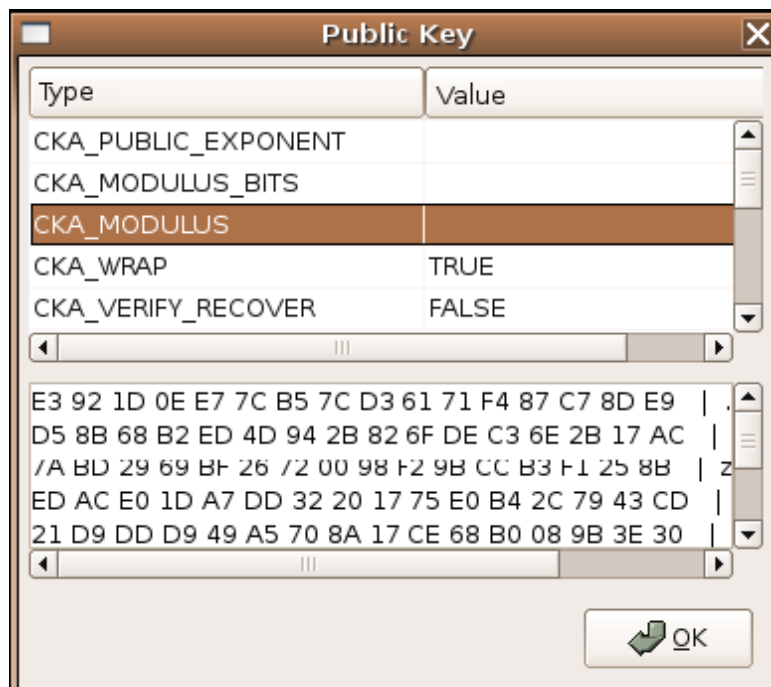


Figure 2-24 View Public Key Information Dialog

Click on any attribute button, its detailed information will be displayed in the bottom.

2.16 Delete Data

When user wants to delete data in the token, after login, select the information to delete and click on “Delete”. System will prompt the dialog box like figure 2-25:



Figure 2-25 Delete Data Dialog

Data can not be retrieved after the delete operation.