

ePassNG

User's Guide

Version 1.0

Feitian Technologies Co., Ltd. ("Feitian" for short) will do their best to keep the content of this document as accurate as possible. But Feitian 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.

Revision History:

Date	Version	Description
November 2006	1.0	1st Edition

Feitian Technologies Co., Ltd.

Software Developer's Agreement

All Products of Feitian Technologies Co., Ltd. (Feitian) including, but not limited to, evaluation copies, diskettes, CD-ROMs, hardware and documentation, and all future orders, are subject to the terms of this Agreement. If you do not agree with the terms herein, please return the evaluation package to us, postage and insurance prepaid, within seven days of their receipt, and we will reimburse you the cost of the Product, less freight and reasonable handling charges.

1. Allowable Use - You may 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. You may make archival copies of the Software.

2. Prohibited Use - The Software or hardware or any other part of the Product may not be copied, reengineered, disassembled, decompiled, revised, enhanced or otherwise modified, except as specifically allowed in item 1. You may not reverse engineer the Software or any part of the product or attempt to discover the Software's source code. You may not use the magnetic or optical media included with the Product for the purposes of transferring or storing data that was not either an original part of the Product, or a Feitian provided enhancement or upgrade to the Product.

3. Warranty - Feitian warrants that the hardware and Software storage media are substantially free from significant defects of workmanship or materials for a time period of twelve (12) months from the date of delivery of the Product to you.

4. Breach of Warranty - In the event of breach of this warranty, Feitian's sole obligation is to replace or repair, at the discretion of Feitian, any Product free of charge. Any replaced Product becomes the property of Feitian.

Warranty claims must be made in writing to Feitian during the warranty period and within fourteen (14) days after the observation of the defect. All warranty claims must be accompanied by evidence of the defect that is deemed satisfactory by Feitian. Any Products that you return to Feitian, or a Feitian authorized distributor, must be sent with freight and insurance prepaid.

EXCEPT AS STATED ABOVE, THERE IS NO OTHER WARRANTY OR REPRESENTATION OF THE PRODUCT, EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE

IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

5. Limitation of Feitian's Liability - Feitian's entire liability to you or any other party for any cause whatsoever, whether in contract or in tort, including negligence, shall not exceed the price you paid for the unit of the Product that caused the damages or are the subject of, or indirectly related to the cause of action. In no event shall Feitian be liable for any damages caused by your failure to meet your obligations, nor for any loss of data, profit or savings, or any other consequential and incidental damages, even if Feitian has been advised of the possibility of damages, or for any claim by you based on any third-party claim.

6. Termination - This Agreement shall terminate if you fail to comply with the terms herein. Items 2, 3, 4 and 5 shall survive any termination of this Agreement.

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 Technoogy Equipment.

USB

This equipment is USB based.

WEEE

Dispose in separate collection.

Technical Terms and Abbreviations

Term	Description
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 achieves 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® platforms.
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_FT11	Portable cryptographic device integrates smartcard and USB port, which is released by Feitian. It inherits the advantages of smartcard device and also portable. It supports PKI applications.
ePass2000_FT11 (Driver-free ePass2000)	A non-driver USB Token introduced by Feitian. It has the same functionalities as ePass2000_FT11 but does not need to install hardware driver.
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 support. It supports PKI applications.
TSP (Token Service Provider)	Abstract hardware layer in ePassNG framework architecture. It provides common I/O interfaces for all kinds of devices. This design eliminates the inconvenience from hardware differences to some extent.

Contents

1. INTRODUCTION	1
1.1 ePASSNG FRAMEWORK ARCHITECTURE	2
1.2 ePASSNG FEATURES	3
2. EPASSNG ADMIN TOOL	5
2.1 PREREQUISITE	6
2.2 PROFILE	6
2.2.1 <i>Interface without Token Plugged in</i>	6
2.2.2 <i>Interface with Token Plugged in</i>	6
2.2.3 <i>Admin Tool Menu</i>	7
2.2.4 <i>"Operation" Menu</i>	8
2.2.5 <i>"View" Menu</i>	8
2.2.6 <i>Right-Click Menu in Slot Tree</i>	9
2.2.7 <i>Information Displayed After Plugging in Token</i>	10
2.2.8 <i>Information Displayed When No Token Plugged in</i>	10
2.3 CHECKING SLOT LIST INFORMATION	11
2.4 CHECKING TOKEN INFORMATION	11
2.5 LOGIN	11
2.6 CHANGING USER PIN	12
2.7 CHANGING TOKEN NAME	12
2.8 CHANGING SO PIN	13
2.9 UNBLOCKING TOKEN	14
2.10 INITIALIZING TOKEN	14
2.11 DATA MANAGEMENT IN UN-LOGIN STATE	15
2.12 DATA MANAGEMENT IN LOGIN STATE	16
2.13 IMPORTING CERTIFICATES	17
2.14 EXPORTING CERTIFICATES	18
2.15 SHOWING DATA INFORMATION	19
2.16 DELETING DATA	21

1. Introduction

ePassNG is a new generation platform-independent data security product framework. It mainly provides hardware support to upper-level PKI applications. The certificates, key pairs and other classified information are all stored in ePass Token. ePassNG provides standard PKCS#11 and CryptoAPI programming interfaces to support standard PKI applications. It is easy to be used by ISVs (Independent Software Vendors) to develop their own PKI applications for their end users. Moreover, because of the simple framework structure, hardware providers can integrate their hardware units into the ePassNG framework by implementing a TSP (Token Service Provider), so as to integrate their hardware units into the PKI framework easily.

This chapter contains the following topics:

- ePassNG Framework Architecture
- ePassNG Features

1.1 ePassNG Framework Architecture

ePassNG provides standard PKCS#11 and CryptoAPI programming interfaces for upper-level PKI applications. ISVs can develop their own PKI applications based on these interfaces. Moreover, interfaces provided by ePassNG can be seamlessly integrated with any standard PKI applications through simple configuration.

The ePassNG's framework architecture is described as follows.

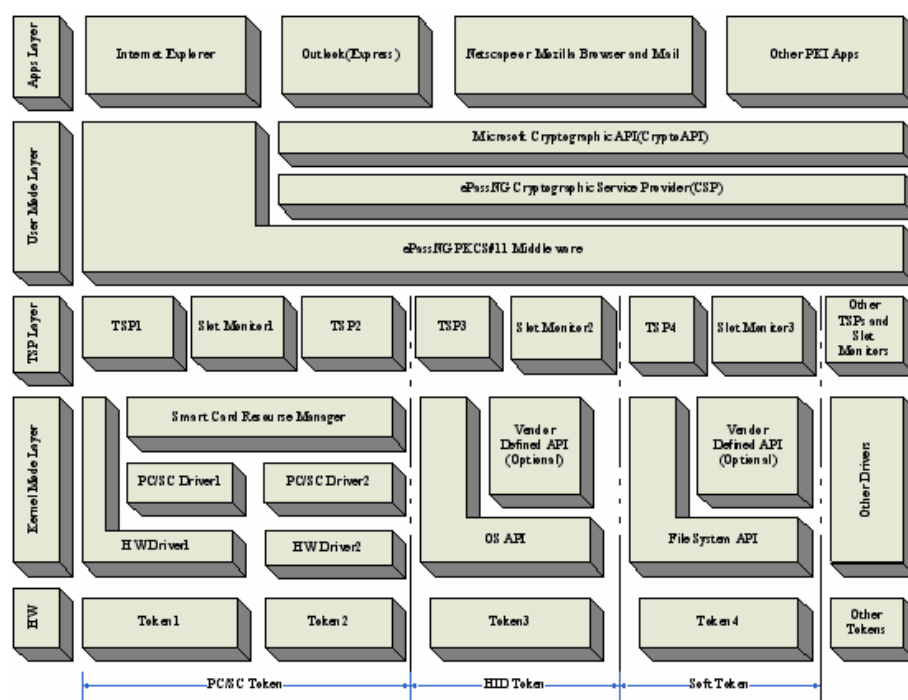


Figure 1.1

It can be seen that ePassNG architecture is comprised of 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 architecture. It contains various tokens including the hardware circuitry, firmware programs and wires. Any tokens compliant with PC/SC and HID standard can be supported by this hardware layer, such as ePass1000, ePass2000_FT11, ePass2000_FT12, ePass3000, ePass3000ND, various other smartcard readers and smartcard combined third-party USB-Keys. Their common feature is that the tokens must be able to be controlled by the operating system's Smart Card Resource Manager. Tokens can also be HID (Human Interface Device) devices, such as ePassND (non-driver USB key introduced by Feitian), USB flash memory disks and even files on the hard disk. Different kinds of tokens or multiple tokens of the same type can work together.

- **Core Driver Layer**

Core driver layer manages data communications and processes access request from the TSP layer between the client computer and hardware layer. To PC/SC tokens, this layer works like a hardware driver, PC/SC driver and operation system's Smart Card Resource Manager. For HID tokens, this layer can be treated as an operating system's build-in drivers. For file system tokens, this layer functions like operating system's file operation system.

- **Abstract Hardware Layer**

Abstract hardware layer provides standard abstract interfaces to the application interface layer. Communications between the 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 interfaces and MS CryptoAPI interface for upper layer PKI applications.

Moreover, the PC/SC application interface compliant with Microsoft® PC/SC standard is also provided. Developers can develop applications with their familiar PC/SC function set. This interface is platform-independent and can 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 can use this familiar interface set to design their own applications.

1.2 ePassNG Features

- **Platform Independent**

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

- **Interface Standard**

ePassNG provides standard PKI interface for its upper-level applications, including RSA PKCS#11 and MS CryptoAPI (this interface can only be applied under Windows platforms). All the applications using either interface can use ePassNG to store certificates and keys, processing cryptographic operations. For the extension of a hardware token, standard interface

to third-party vendors for their hardware implementations is provided.

- **Good Compatibility**

ePassNG is fully compatible with Feitian's ePass2000_FT11 hardware products. Previous certificates and key pairs are still applicable within ePassNG. Moreover, the certificates applied by ePassNG in one platform can still be used in another platform. This enables users to use unique identification crossing different platforms.

- **Support for Various Tokens**

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

- **Easy for Extension**

Third-party vendors can integrate their products into the ePassNG framework by signing the related agreement with Feitian. Using the TSP developing interface provided by Feitian, vendors just need to make minor modifications, or even nothing, for the integration.

- **Growing Up Everyday**

Feitian's ePass2000_FT11 products have obtained many domestic and international authoritative qualifications, including CheckPoint, CFCA etc., allowing ePass2000_FT11 to become an ever increasingly successful product, stable and secure. It will gain more and more qualifications step by step.

2. ePassNG Admin Tool

ePassNG's administrative tools' interface and operating method are similar under different system platforms so as to provide more conveniences to users. Furthermore, ePass1000ND, ePass2000_FT11, ePass2000_FT12, ePass3000ND, ePass3000 and ePass3000OEM share the same administrative tool.

There are two versions of the ePassNG administrative tool: administrator edition and end user edition. Administrator edition provides more functionality of "Initialize Token", "Unblock PIN" and "Change SOPIN".

Using the administrator edition of ePassNG GUI administrative tool with ePass2000_FT11 under the Windows® operating systems:

This chapter describes the following functionalities:

- 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 Prerequisite

Because the administrative tool is based on ePassNG middleware and will access the hardware token, make sure that the ePassNG products (including middleware and hardware driver) have already been installed properly before use.

2.2 Profile

2.2.1 Interface without Token Plugged in

Running administrative tool, system will display the interface as follows:

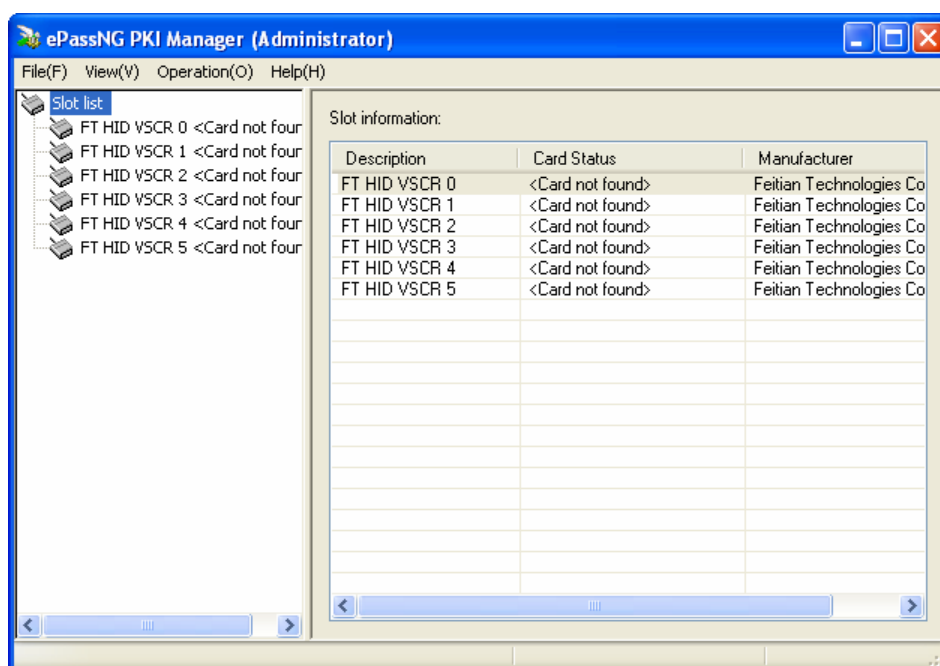


Figure 2.1

Left column lists all the supported slots. Right column gives the basic information for each of these slots.

2.2.2 Interface with Token Plugged in

Plugging an USB token named "ePass Token" into the USB port of the computer, the administrative tool will recognize the token's basic information and display the interface as follows:

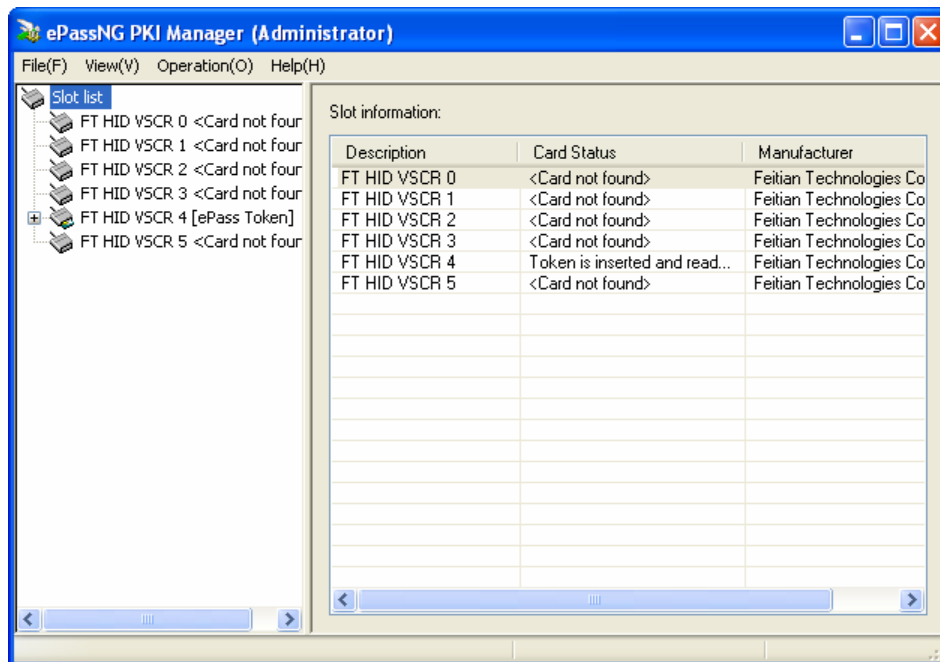


Figure2.2

2.2.3 Admin Tool Menu

The main interface may look like the following:

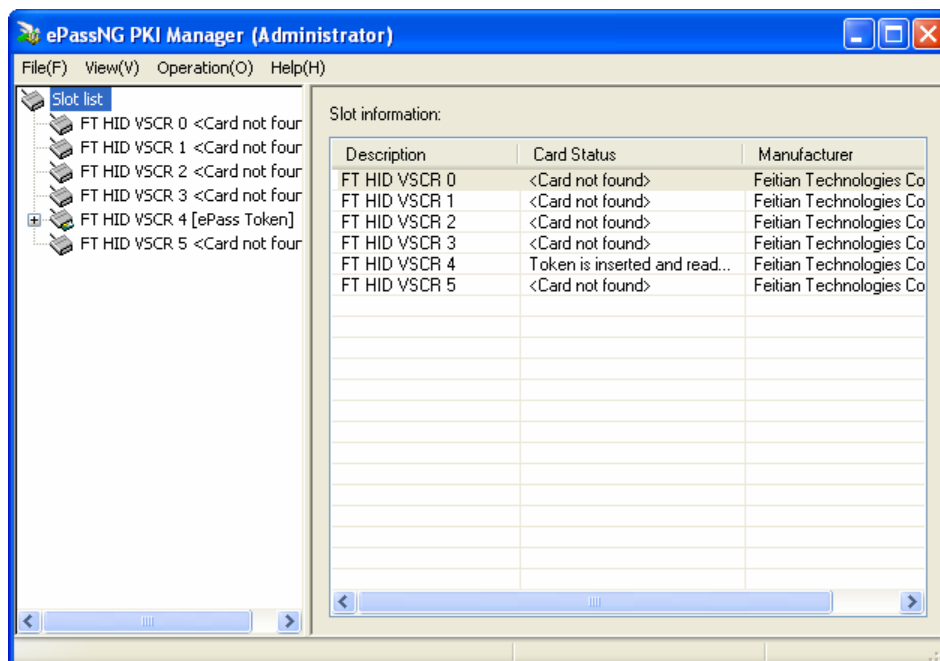


Figure 2.3

The main menu includes: File (exit the tool), View (check the slot information), Operation (operations related to the slots) and Help (version information etc.).

2.2.4 “Operation” Menu

See the following figure for the specific options.

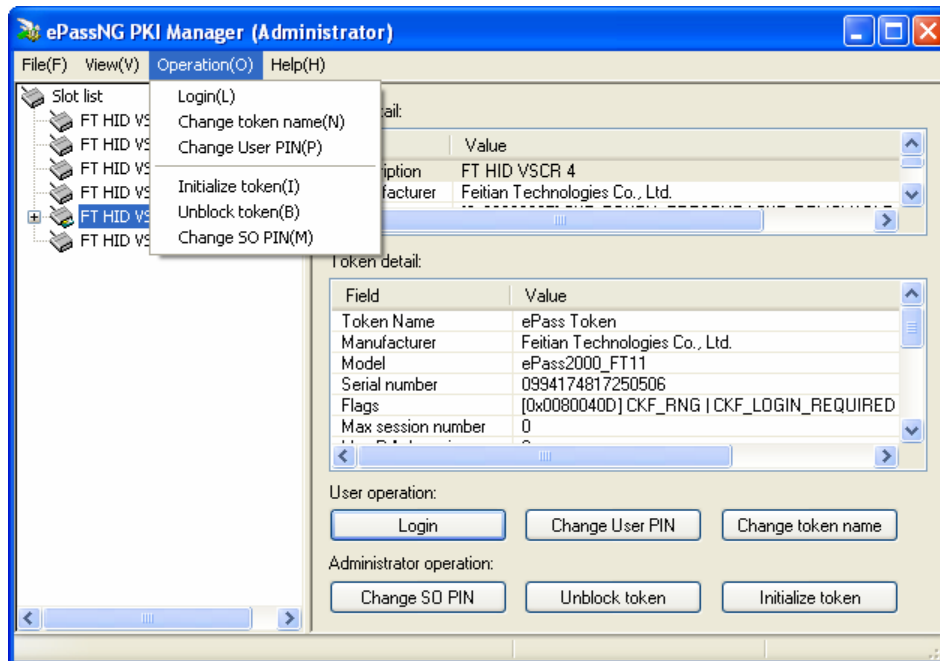


Figure 2.4

Drop down menu lists the applicable options.

2.2.5 “View” Menu

See the following figure for the specific options.

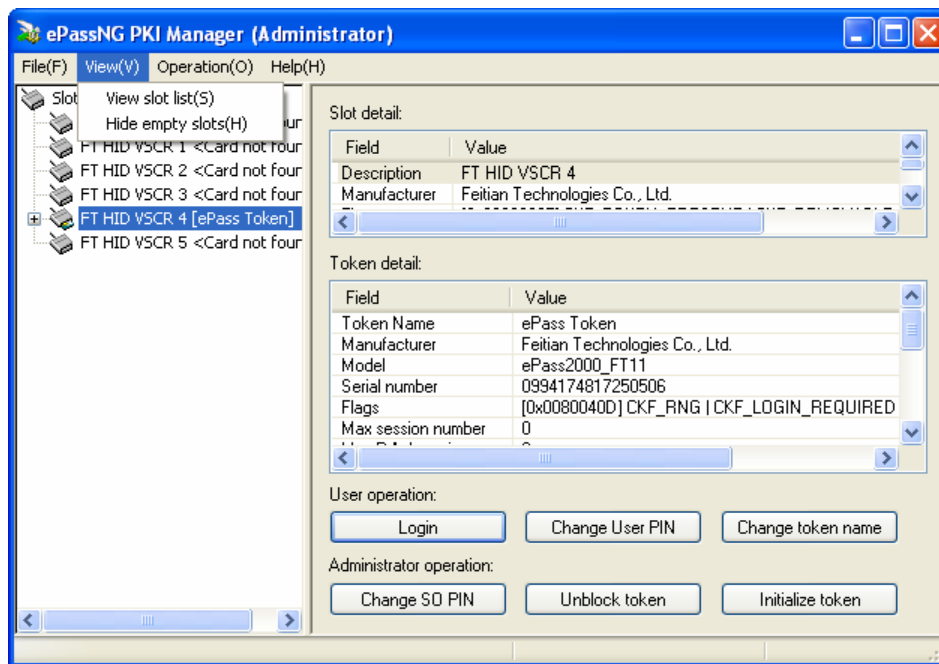


Figure 2.5

2.2.6 Right-Click Menu in Slot Tree

Right-click on any slot listed on the left-side slot tree - system will pop-up a menu as shown in Figure 2.6:

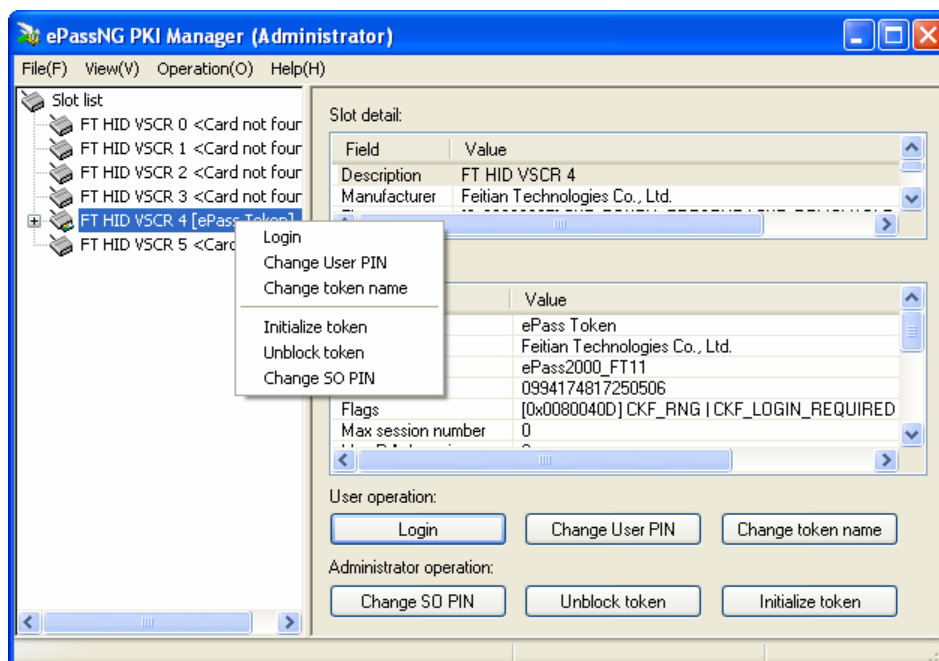


Figure 2.6

Operations include "Login", "Change User PIN", "Change token name", "Change SO PIN",

“Unblock token” and “Initialize token”.

2.2.7 Information Displayed After Plugging in Token

Clicking on any slot, its information and related possible operations will be displayed on the right side, as shown in Figure 2.7:

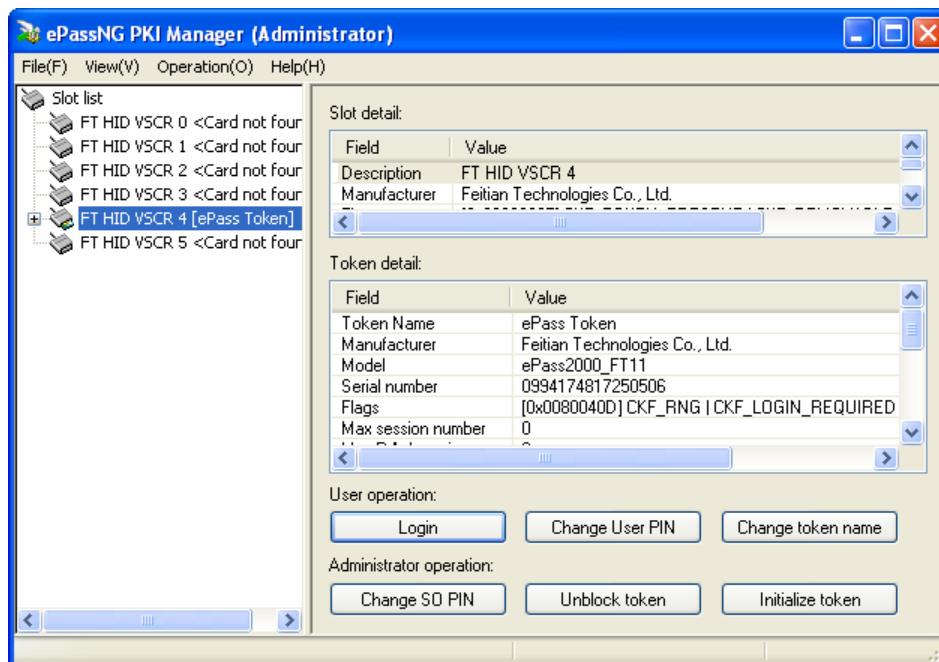


Figure 2.7

Information displayed on the right side includes the slot's status, token's 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 Plugged in

Click on any empty slot. The following panel will be displayed:

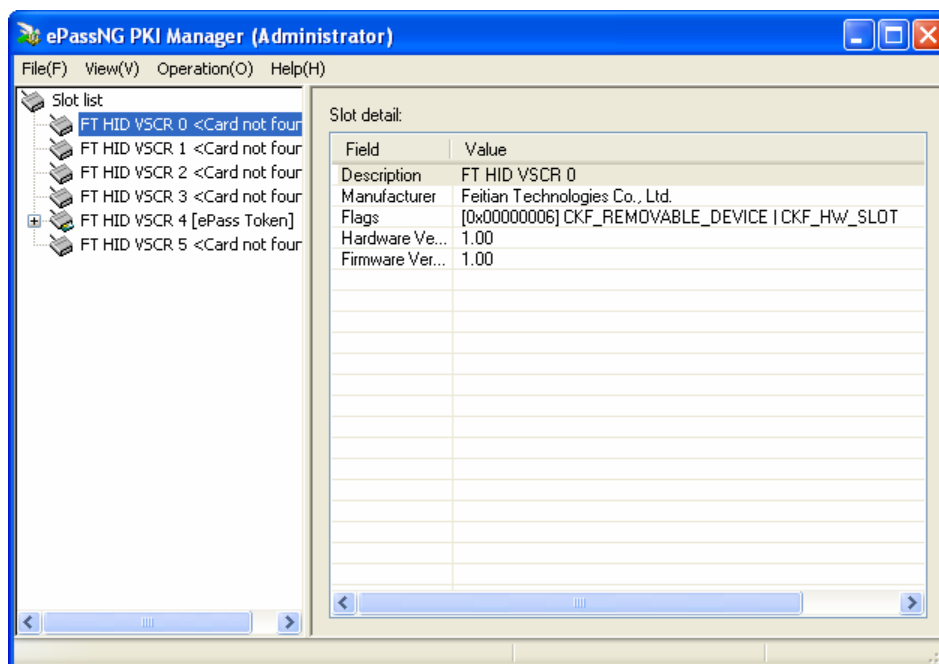


Figure 2.8

When no token is plugged in a slot, click on slot. The slot's detailed information will be displayed.

ePassNG supports multiple tokens. By choosing a token, users can process operations as demonstrated in Figure 2.4, Figure 2.6 and Figure 2.7.

2.3 Checking Slot List Information

Click left side of the slot list or select "View" > "View slot list", system will display the slot's information, as shown in Figure 2.1.

2.4 Checking Token Information

Left click on any slot. The related information will be displayed in the right side. If a token is plugged in, it will display the token's detailed information, as shown in Figure 2.7. If slot is empty, the information about the slot will be displayed, as shown in Figure 2.8.

2.5 Login

Before login, users can only view public information of the token. Private information can only

be retrieved after user process the login operation with the correct PIN number. Click on “Login” button, system will prompt a login dialog box, as shown in Figure 2.9:

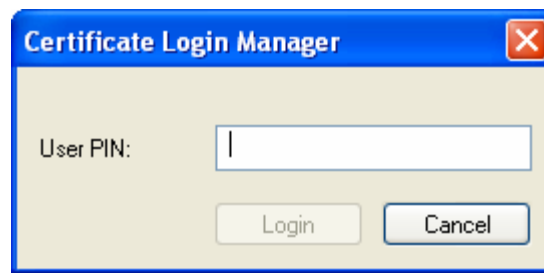


Figure 2.9

After inputting the correct PIN number, click the “OK” button to login the management tool.

2.6 Changing User PIN

User is recommended to change token's initial PIN number to another by clicking “Change User PIN” button. System will prompt a dialog box as shown in Figure 2.10:



Figure 2.10

When changing the user PIN, users must input the old PIN number, input the new PIN number and confirm the new PIN number again. Click “OK” button to process the operation.

2.7 Changing Token Name

Generally, a token is identified by its serial number. But the serial number is not indicative and is hard to remember. Token name can also be used to identify a token. Users can specify a

unique name for a token, whatever they like.

Click “Change token name” button. System will prompt the following dialog box:



Figure 2.11

Input any name for token and click the “OK” button. System will refresh and display the token’s new name.

2.8 Changing SO PIN

User can change token’s SO PIN number by click the “Change SO PIN” button. System will prompt the following dialog box:

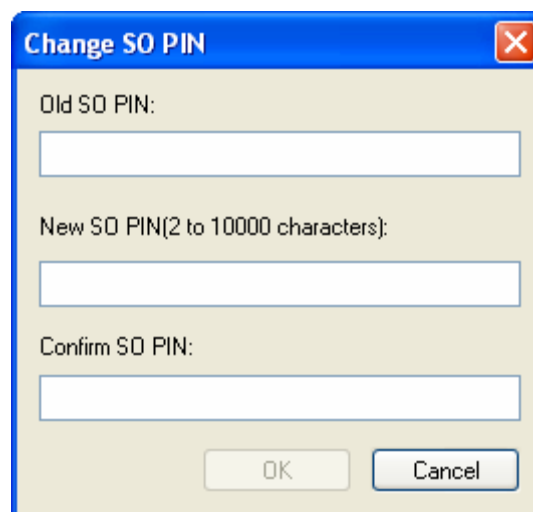
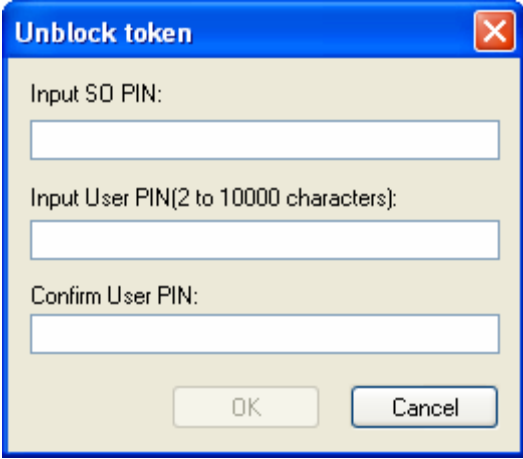


Figure 2.12

When changing the SO PIN, users must input the old SO PIN number, input the new SO PIN number and confirm the new SO PIN number again. Click the “OK” button to process the operation.

2.9 Unlocking Token

When user has failed to input the correct user PIN number over a certain number of times, user PIN will be locked. Even if the user inputs the correct PIN again, the token cannot be accessed. The Administrator must unblock the token by clicking the “Unblock token” button. System will prompt the following dialog box:

A Windows-style dialog box titled "Unblock token" with a blue title bar and a red close button. The dialog has a light beige background. It contains three text input fields: "Input SO PIN:", "Input User PIN(2 to 10000 characters):", and "Confirm User PIN:". At the bottom, there are two buttons: "OK" and "Cancel".

Unblock token

Input SO PIN:

Input User PIN(2 to 10000 characters):

Confirm User PIN:

OK Cancel

Figure 2.13

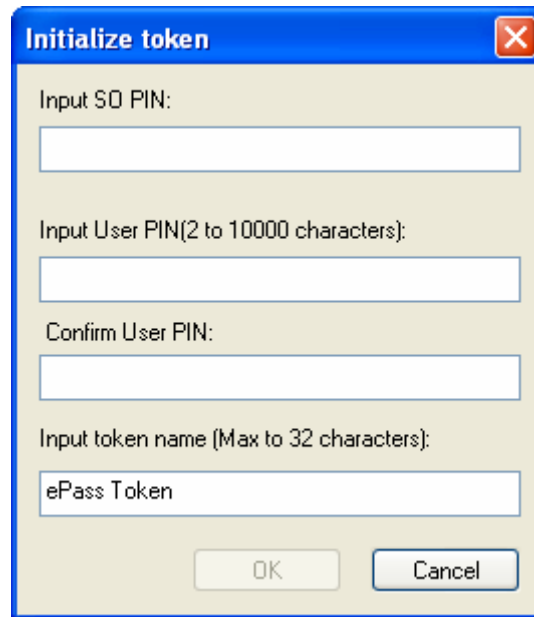
To unblock the user PIN number, SO PIN must be provided. The user PIN must be reset and confirmed. Click the “OK” button to unblock the token. After unblocking the PIN number, the user can log into the token with the new PIN number.

2.10 Initializing Token

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

WARNING: Process this operation, ALL the information in the token including PKI certificates, public and private keys and user data will be entirely removed.

Clicking “Initialize token” button, system will prompt the following dialog box:

A Windows-style dialog box titled "Initialize token" with a blue header bar and a red close button. The dialog contains four text input fields. The first is labeled "Input SO PIN:". The second is labeled "Input User PIN(2 to 10000 characters):". The third is labeled "Confirm User PIN:". The fourth is labeled "Input token name (Max to 32 characters):" and contains the text "ePass Token". At the bottom are "OK" and "Cancel" buttons.

Initialize token

Input SO PIN:

Input User PIN(2 to 10000 characters):

Confirm User PIN:

Input token name (Max to 32 characters):

ePass Token

OK Cancel

Figure 2.14

Initializing token requires the 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, the system will refresh and change the token's login state.

2.11 Data Management in Un-login State

In the left area of the administrative tool, each token has a data management function. Click it in un-login state; the system will display the token's public information and related possible operations on the right side, as shown in Figure 2.15:

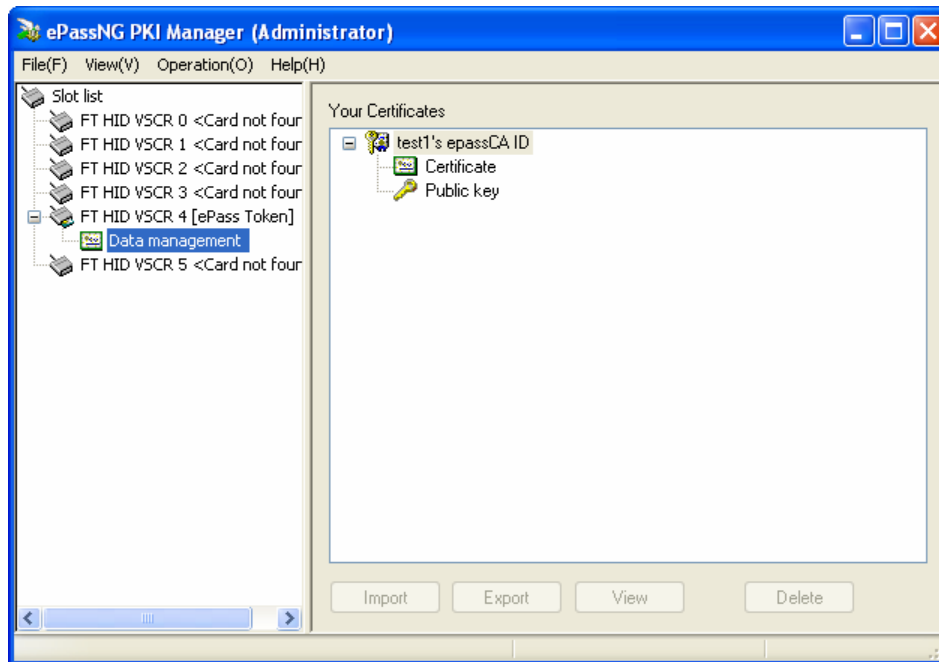


Figure 2.15

2.12 Data Management in Login State

After logging into the token, the system will display the following interface:

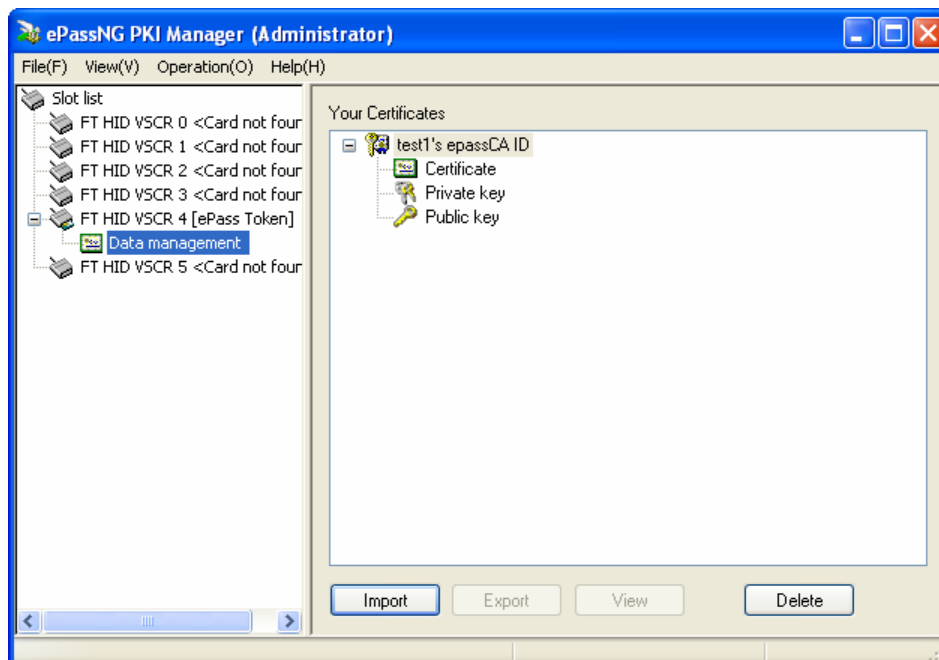


Figure 2.16

After login, users can view both token's public information and private information. The "Import" button will only be enabled after user login. The "Export" button will be enabled when a

certificate is selected. The “View” button is always enabled except for token name. The “Delete” button is always enabled.

After user login, initialized token, changed user PIN or unblocked token, The “Login” button will be disabled, demonstrating the token has changed into login state, as shown in Figure 2.17:

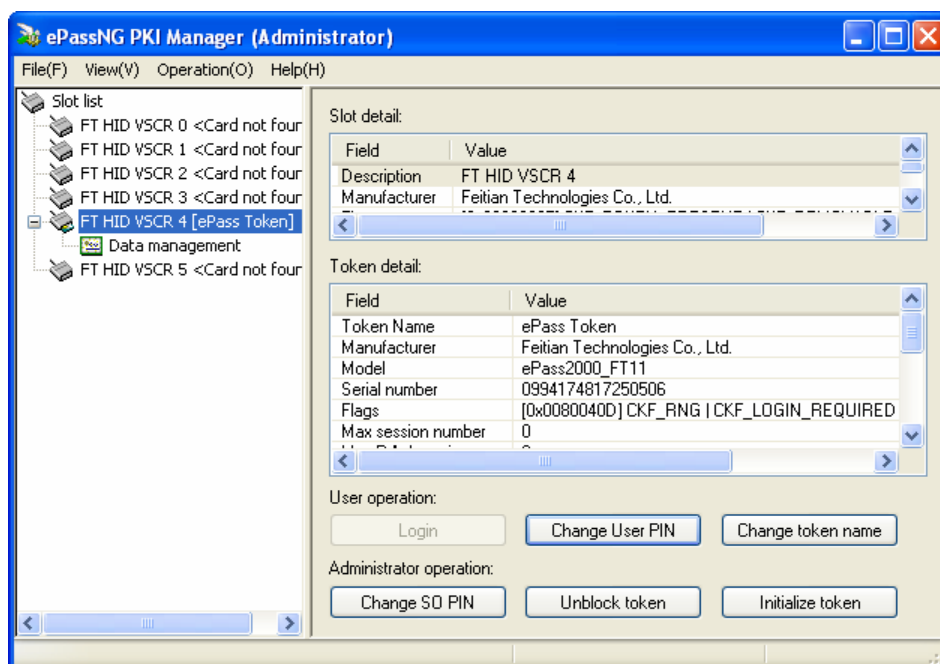


Figure 2.17

2.13 Importing Certificates

When the user wants to import .PFX, P12, P7B, or CER certificates into the token, click on the “Import” button. System will prompt the following dialog box:

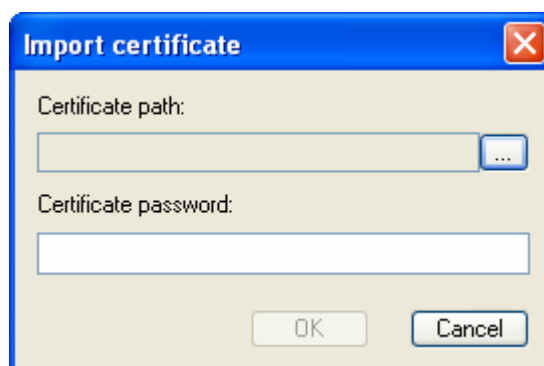


Figure 2.18

Only P12 certificates need to confirm certificate password. When importing other types of

certificates, the 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 the “OK” button, the system will import the certificate into the token and refresh automatically, as shown in Figure 2.19:

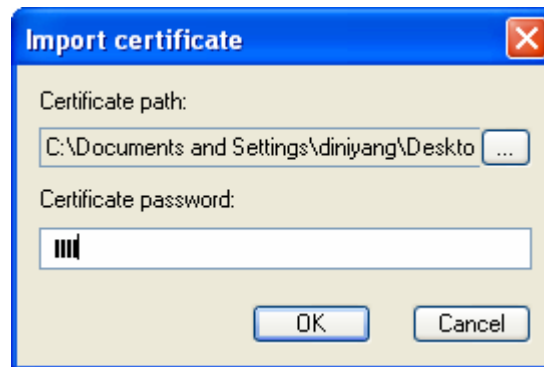


Figure 2.19

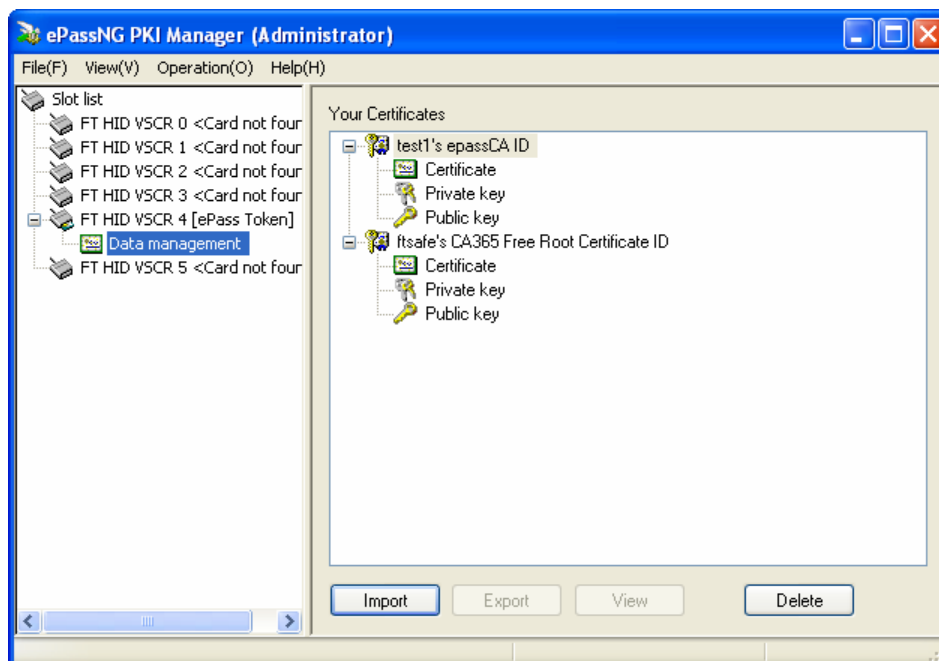


Figure 2.20

2.14 Exporting Certificates

When the user wants to export a certificate, click on the “Export” button and the system will prompt the following dialog box:

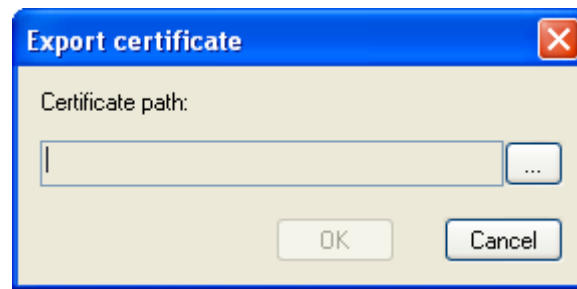


Figure 2.21

Click “...” button and choose the directory. Then click the “OK” button to export the certificate.

2.15 Showing Data Information

When the user wants to view detailed information of the certificate, public key, private key and other data, the user can select a special item and click on the “View” button. System will prompt the following dialog box:

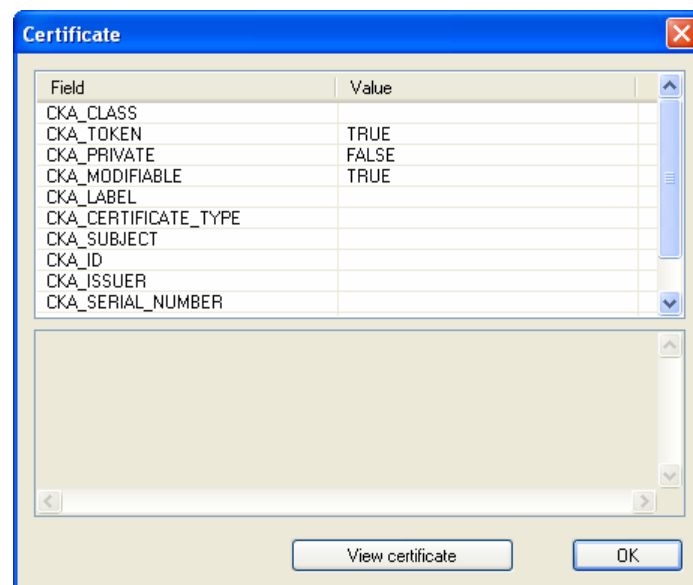


Figure 2.22

Only when viewing certificate's content, the view certificate button will appear. Click on it, and the system will display the information as shown in Figure 2.23:

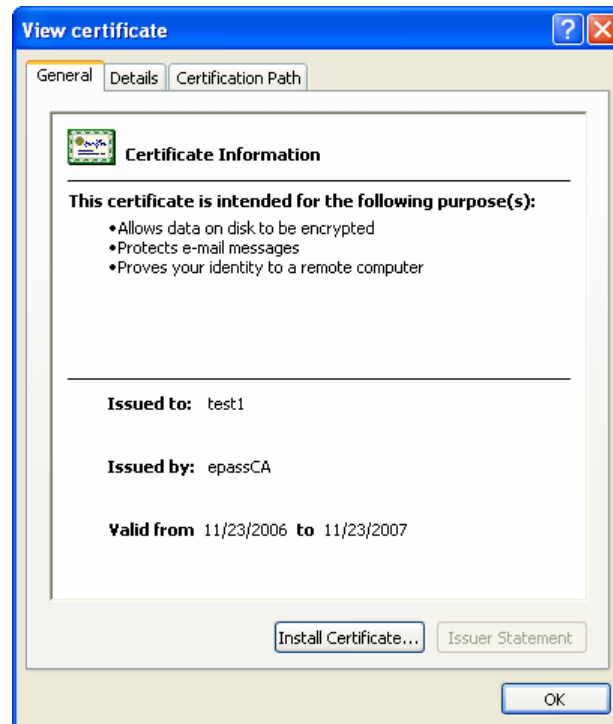


Figure 2.23

To view other data information (such as public key, private key or other data), system will prompt the following dialog box:

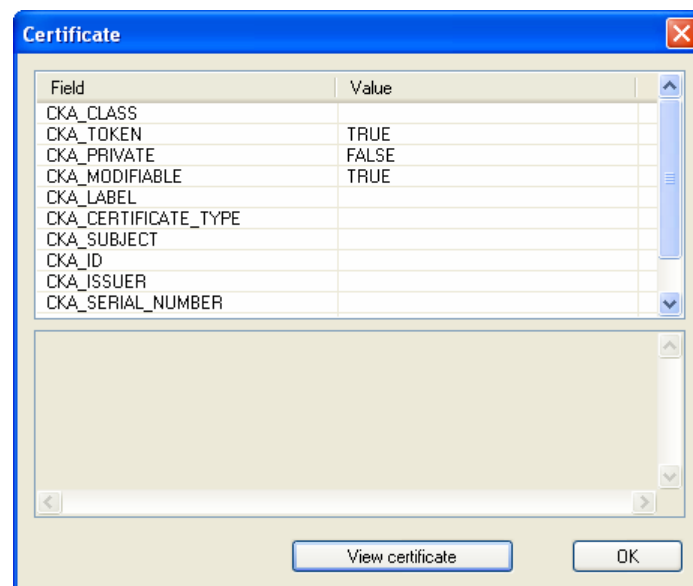


Figure 2.24

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

2.16 Deleting Data

When user wants to delete information in the token after login, select the information the user wants to delete and click on the “Delete” button. System will prompt the following dialog box:

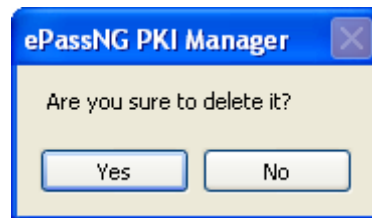


Figure 2.25

Data cannot be retrieved after the deletion.