THALES

ProtectToolkit 5.9.1
PTK-M

USER GUIDE

Document Information

Last Updated	2024-04-18 12:30:10-04:00

that independent testing and verification by the person using the product is particularly encouraged, especially in any application in which defective, incorrect or insecure functioning could result in damage to persons or property, denial of service, or loss of privacy.

All intellectual property is protected by copyright. All trademarks and product names used or referred to are the copyright of their respective owners. No part of this document may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, chemical, photocopy, recording or otherwise without the prior written permission of Thales Group.

CONTENTS

PREFACE: About the ProtectToolkit-M User Guide

This document provides instructions on configuration, administration, and troubleshooting for the ProtectToolkit-M suite of applications. It contains the following chapters:

This preface also includes the following information about this document:

For information regarding the document status and revision history, see "Document Information" on page 2.

Document Conventions

This document uses standard conventions for describing the user interface and for alerting you to important information.

Notes

Notes are used to alert you to important or helpful information. They use the following format:

> **NOTE**   Take note. Contains important or helpful information.

Cautions

Cautions are used to alert you to important information that may help prevent unexpected results or data loss. They use the following format:

> **CAUTION!**   Exercise caution. Contains important information that may help prevent unexpected results or data loss.

Warnings

Warnings are used to alert you to the potential for catastrophic data loss or personal injury. They use the following format:

> ****WARNING****   **Be extremely careful and obey all safety and security measures. In this situation you might do something that could result in catastrophic data loss or personal injury.**

Command Syntax and Typeface Conventions

Format	Convention
bold	The bold attribute is used to indicate the following: > Command-line commands and options (Type **dir /p**.) > Button names (Click **Save As**.) > Check box and radio button names (Select the **Print Duplex** check box.) > Dialog box titles (On the **Protect Document** dialog box, click **Yes**.) > Field names (**User Name:** Enter the name of the user.) > Menu names (On the **File** menu, click **Save**.) (Click **Menu > Go To > Folders**.) > User input (In the **Date** box, type **April 1**.)
italics	In type, the italic attribute is used for emphasis or cross-references to other documents in this documentation set.
<variable>	In command descriptions, angle brackets represent variables. You must substitute a value for command line arguments that are enclosed in angle brackets.
[**optional**] [<optional>]	Represent optional **keywords** or <variables> in a command line description. Optionally enter the keyword or <variable> that is enclosed in square brackets, if it is necessary or desirable to complete the task.
{**a|b|c**} {<a>|<b>|<c>}	Represent required alternate **keywords** or <variables> in a command line description. You must choose one command line argument enclosed within the braces. Choices are separated by vertical (OR) bars.
[**a|b|c**] [<a>|<b>|<c>]	Represent optional alternate keywords or variables in a command line description. Choose one command line argument enclosed within the braces, if desired. Choices are separated by vertical (OR) bars.

Support Contacts

If you encounter a problem while installing, registering, or operating this product, please refer to the documentation before contacting support. If you cannot resolve the issue, contact your supplier or Thales Customer Support.

Thales Customer Support operates 24 hours a day, 7 days a week. Your level of access to this service is governed by the support plan arrangements made between Thales and your organization. Please consult this support plan for further information about your entitlements, including the hours when telephone support is available to you.

Customer Support Portal

The Customer Support Portal, at https://supportportal.thalesgroup.com, is where you can find solutions for most common problems. The Customer Support Portal is a comprehensive, fully searchable database of support resources, including software and firmware downloads, release notes listing known problems and workarounds, a knowledge base, FAQs, product documentation, technical notes, and more. You can also use the portal to create and manage support cases.

> **NOTE** You require an account to access the Customer Support Portal. To create a new account, go to the portal and click on the **REGISTER** link.

Telephone

The support portal also lists telephone numbers for voice contact (Contact Us).

CHAPTER 1:   Overview

This section introduces ProtectToolkit-M and shows how SafeNet components and terminology apply in the Microsoft Cryptographic API environment.

ProtectToolkit-M Applications

With ProtectToolkit-M installed, applications that call the Microsoft Cryptographic API (MSCAPI) can make use of the secure key storage and high-speed cryptographic processing offered by SafeNet hardware security modules (HSMs).

The Microsoft Cryptographic API (MSCAPI) provides security services for a range of applications, such as web-based SSL processes.

Microsoft Certification Authority (MSCA) and Internet Information Services (IIS) (a Microsoft web server) use the MSCAPI and therefore may be integrated with ProtectToolkit-M. An MSCA may store CA keys on an HSM, while IIS may use HSM key storage when establishing secure socket layer (SSL) communication.

The MSCAPI Model and ProtectToolkit-M

Cryptographic Service Providers

ProtectToolkit-M is implemented as a Microsoft Cryptographic Service Provider (CSP).

A CSP is a plug-in cryptographic module that integrates with Microsoft Windows and provides the underlying key storage and security operations for the Microsoft Cryptographic API (MSCAPI). The architecture of the MSCAPI supports the development of non-Microsoft CSPs such as ProtectToolkit-M.

ProtectToolkit-M includes both "RSA Full" and "RSA SChannel" cryptographic service providers. These can be used instead of the corresponding Microsoft CSPs to provide hardware-based key storage and RSA encryption.

MSCAPI Implementation Using ProtectToolkit-M

shows how SafeNet HSMs can be utilized as part of a MSCAPI system, using ProtectToolkit-M as a CSP.

Figure 1: ProtectToolkit-M model
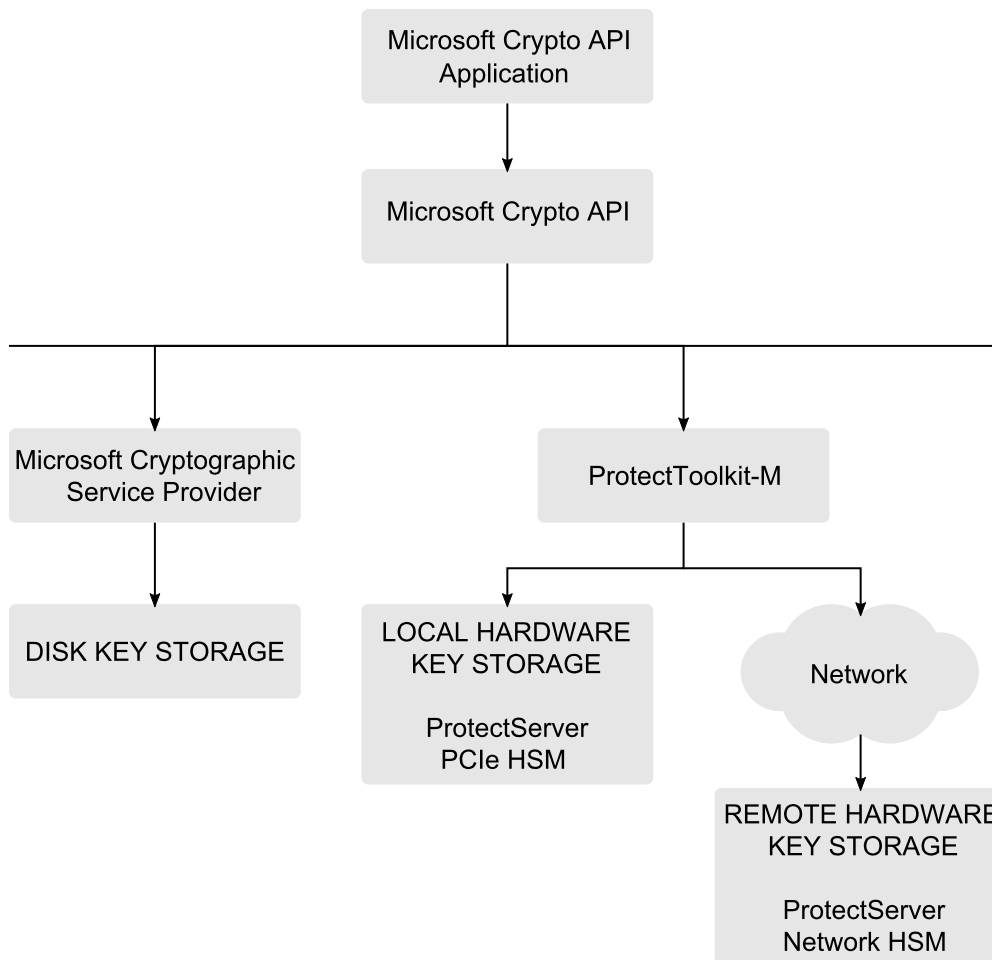
MSCAPI Keyset Model

Within MSCAPI (and hence ProtectToolkit-M), key pairs are held within a key container, which is stored within a keyset.

```
HSM Secure Memory
    │
    ├── Keyset Space
    │
    └── Keyset User (1)
            │
            └── Key Container
                    │
                    ├── Signature Key Pair
                    └── Exchange Key Pair
```
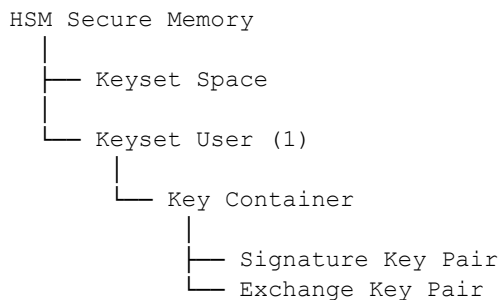
Each user requiring processing support from the ProtectToolkit-M system will need a user keyset containing a key container. Key containers may contain up to 2 key pairs: a signature key pair and an exchange key pair.

Apart from this, there are two keysets required by the ProtectToolkit-M system for its internal processes. These are the SYSTEM keyset and the MACHINE keyset, which are visible to all system users. ProtectToolkit-M cannot operate without either of these and will automatically create either set if they are not present or deleted. Shared

keys (accessible by more than one user), such as those generated automatically when Microsoft CA is installed, will also be stored in one of these keysets when using a ProtectToolkit-M CSP. Generally these shared keys are stored in the MACHINE keyset.

The physical storage location for each keyset is CSP-dependent. By default, Microsoft CSPs store keys to disk, in user profiles. When using the "Safenet RSA Full" or "Safenet RSA SChannel" CSPs, all keys are secured by ProtectToolkit-M within SafeNet hardware security modules (HSMs).

Further Documentation

The following reference material should be considered in addition to this user manual:

> *ProtectServer HSM and ProtectToolkit Installation Guide*

> Microsoft documentation on cryptographic service providers. See their web site.

CHAPTER 2:   Setup and Configuration

After installing ProtectToolkit-M it is necessary for the device administrator to:

> initialize the HSM

> set the security mode

> allocate keyset space

> create user keysets (This is optional as users may also create their own keysets)

> setup work load distribution (WLD) if required

After the device administrator has performed the above steps then users will typically need to undertake the following tasks:

> create keysets

> add containers to keysets

> generate key pairs in containers

To perform these tasks follow the procedures described in this section:

> "User Roles" below

> "Initial Configuration: Mandatory Steps" on the next page

> "Allocating Keyset Space" on page 18

> "Configuration Options" on page 19

> "SafeNet KSP for CNG Registration Utilities" on page 21

> "Configuring IIS7 (Win2008) with CNG" on page 25

User Roles

Prior to performing any configuration, it is important to understand the different ProtectToolkit-M roles available and to determine which type of role an individual will assume.

There are two defined roles available. These are:

> A ProtectToolkit-M device administrator; and

> A ProtectToolkit-M user

These roles are described below.

Device Administrator

The device administrator is responsible for tasks that involve management of the associated HSM and those applicable to ProtectToolkit-M administration. Those assigned to this role are also responsible for performing backup and restore operations for MACHINE and SYSTEM keysets and allocation of space for user keysets.

User

A ProtectToolkit-M user is responsible for the creation and management of their own keyset stored within a HSM. This includes responsibility for backup and restore of their own keyset, the key container and associated key pairs.

Initial Configuration: Mandatory Steps

Initializing the HSM

ProtectToolkit-M uses a hardware encryption HSM to store sensitive key information. The HSM needs to be initialized prior to use, or following a tamper event. A tamper event occurs, for example, if the HSM detects that someone is trying to get inside the cover. It will then automatically erase its secure memory.

HSM management tasks can only be performed by a device administrator. During HSM initialization, the device administrator password is set, and the HSM clock is synchronized with the host.

To initialize the HSM

1. Open the Administration Utility by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**. A prompt displays to set the new device administrator password. User PINs are case-sensitive, and must be 4-32 characters in length.

2. Enter the password in both the Admin Password and Confirmation fields and, if required, check "Keep Password For Session". For added security, leave the "Keep Password For Session" box unchecked. Password entry will then be required to complete every task. For convenience, check this box so that password entry will not be required again until after the utility is closed and reopened.

3. Click **OK**. The MACHINE and SYSTEM keysets are now created. The Administration Utility dialog box displays showing MACHINE and SYSTEM keyset icons under Active Adapters as shown below.
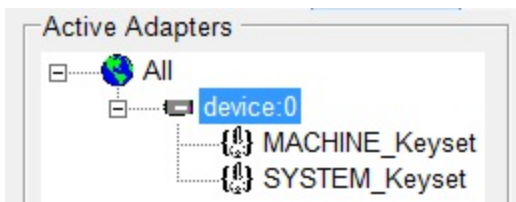
Setting Security Modes and Security Flags

The security mode for the HSM is chosen by specifying the value of a number of security flags. These flags affect both the services available to the various users of the system as well as specific security features of the HSM. These flags may be specified individually to set a custom security mode, however it is recommended that a standard security mode be used. When a standard security mode is selected the flags are assigned values automatically to meet the requirements for that mode.

> **NOTE** The security mode should be set prior to commencing normal operation. It is recommended that the impact of any proposed security mode change be carefully assessed to be applicable prior to implementation.

To set a security mode

1. If it is not already open, launch the Administration Utility from the **Start** menu by selecting **Start > Programs > Safenet > ProtectToolkit M > gmadmin**.

2. From the **Adapters** menu, select **Set Security Flags**. The **Set Security Flags - Device** dialog box displays.

3. Either:

   a. change flag values to those required (see "Security Mode Flag Descriptions" on the next page) in the **Security Mode Flags** group box

   b. click a security mode button in the **Security Modes** group box to set a standard security mode. See "Security Mode Descriptions" below.

4. Click **OK** and enter the administration password if prompted. A confirmation message is displayed.

5. Click **OK** to return to the **Administration Utility** dialog box.

Security Mode Descriptions

This section describes the security modes that can be selected from the Security Modes group box in the **Set Security Flags - All Devices** dialog box.

Set All and Clear All Modes

> Click **Set All** to set all available security flags.

> Click **Clear All** to remove all security flags.

FIPS 140 Mode

FIPS 140 Mode refers to the security flag settings required to comply with the Federal Information Processing Standards (FIPS) 140 standard.

It is important to note that the product can function outside the scope of this accreditation. Therefore, ensure that the correct configuration is set if this level of FIPS secure operation is required.

The security mode flags set in FIPS 140 mode are shown in the table below.

Restricted Mode

The **Restricted Mode** security setting is a compromise between performance and security. If Restricted Mode is selected, then the HSM will require all users to identify themselves before cryptographic services are available. This mode also inhibits any clear PINs or sensitive key material from passing through the HSM's PCIe bus interface but each individual request to the HSM does not need to be signed.

The security mode flags set in Restricted Mode are shown in the table below.

Security Mode Preconfigured Flag Settings

When the **FIPS** or **Restricted security mode** buttons are clicked in the **Set Security - All Devices** dialog box, the status of the flags is changed as shown in the table below (default values). Those settings marked with an asterisk (*) are mandatory in order to implement the requirements for the mode concerned. Additional flags, marked with a plus (+), can be changed if required. See "Security Mode Flag Descriptions" below.

Flag	FIPS 140 Mode	Restricted Mode
Tamper Before Upgrade.	Set*	Cleared+
No Public Cryptography	Set*	Set*
Entrust Compliant	Cleared*	Cleared*
No Clear PINs	Set*	Set*
Authentication Protection	Set*	Cleared*
Lock Security Mode	Set*	Set*
Increased Security Mode	Cleared+	Cleared+
Only Allow FIPS Approved Algorithms.	Set*	Cleared*
Full Secure Messaging Encryption	Cleared+	Cleared+
Full Secure Messaging Signing	Cleared+	Cleared*

Security Mode Flag Descriptions

Tamper Before Upgrade

When this flag is set, the HSM will automatically perform a soft tamper (erase all internal secure memory) as part of a firmware upgrade, FM download, or FM disable operation.

No Public Cryptography

When this flag is set, no user can perform a cryptographic operation without having authenticated themselves.

When this flag is set, each token in the system will have the PKCS #11 CKF_LOGIN_REQUIRED flag set, to indicate that applications must authenticate before operations. This security flag does not affect the Admin token, which always requires authentication for use.

> **NOTE** This setting does not impede the ability to perform RSA or other public key processing. It ensures that crypto services cannot be performed by unauthenticated users.

Entrust Compliant

When this flag is set, Entrust Compliant Mode is operational, ensuring compatibility with the Entrust range of applications. These applications require a specific security profile to operate correctly.

No Clear PINs

When this flag is set, no user PINs or other sensitive information may be passed across the host interface in an unencrypted form. This enables secure messaging encryption between applications and the HSM. It will also disable certain functions that would otherwise result in the clear transmission of sensitive data. This flag will also not allow any keys to be created with the attribute CKA_SENSITIVE=FALSE.

Authentication Protection

This flag, when set, enforces secure messaging authentication between applications and the HSM. Each request to the HSM must be digitally signed and will be verified by the HSM. The key used for this signing process is derived from a key shared by the HSM and host application as well as the user PIN.

Applications will operate in a more secure manner with this flag set, but HSM performance will suffer due to the additional operations required to sign each request and response message.

Lock Security Mode

The Lock Security Mode flag, when set, disables further modification of the security mode flag settings. Once set, this flag (or any other security mode flag) cannot be modified. A new security mode can only be implemented after a tamper operation is performed.

Increased Security Level

The Increased Security Level flag, when set, disables the mechanism CKM_EXTRACT_KEY_FROM_KEY and also does not allow the CKA_MODIFIABLE attribute to be changed from False to True.

Only Allow FIPS Approved Algorithms

The Only Allow FIPS Approved Algorithms flag, when set, disables the following non-FIPS-approved algorithms: MD2, MD5, RIPE, CAST, IDEA, RC2, RC4 and RC5.

Full Secure Messaging Encryption

The Full Secure Messaging Encryption flag is similar to the No Clear PINs flag, except that every message is encrypted in both directions between the application and the HSM. The key used for the message encryption is generated using the PKCS#3 Diffie-Hellman Key Agreement Standard.

This flag only performs two-way encryption when using the ProtectToolkit-M client library in the client/server mode over TCP/IP.

Applications will operate in a more secure manner with this flag set, but HSM performance will suffer due to the additional operations required to sign each request and response message.

Full Secure Messaging Signing

The Full Secure Messaging Encryption flag is similar to the Authentication Protection flag, except that every request in both directions between the application and the HSM is digitally signed and verified. The key used for the message encryption is generated using the PKCS#3 Diffie-Hellman Key Agreement Standard.

This flag only performs two-way encryption when using the ProtectToolkit-M client library in the client/server mode over TCP/IP. Applications will operate in a more secure manner with this flag set, but HSM performance will suffer due to the additional operations required to sign each request and response message.

Allocating Keyset Space

In order for applications to use ProtectToolkit-M's key storage facilities, keyset space must be allocated on the HSM. Enough space should be allocated to accommodate the number of users requiring key storage.

Allocation of keyset space is the responsibility of the device administrator and is performed using the ProtectToolkit-M administration utility.

> **NOTE** It is important to determine how many key sets may be needed over time and to allocate sufficient space so that additional key sets can be created without the need for a server shutdown.

To allocate keyset space

1. Launch the Administration Utility from the Start menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gtadmin**.

2. Select the device on which to create keyset space from the **Active Adapters** list.

3. Open the **Adapter** menu and choose **Allocate Space**.

   The Administration Utility prompts for the device administrator password.

   Following correct password entry, the new keyset space is displayed under the device as shown below.
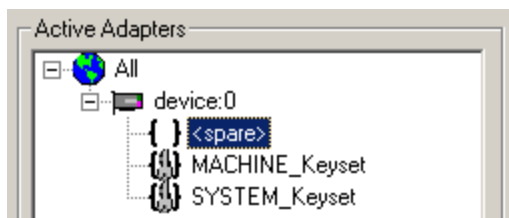
Creating User Keysets

A user keyset is required for each individual that will use the ProtectToolkit-M system. The keysets are stored on the HSM in available keyset spaces. This means that in order to create a user keyset, a free keyset space must be available (see above).

Creating user keysets is the responsibility of the device administrator and is performed using the ProtectToolkit-M administration utility.

> **NOTE** Ideally, all necessary keysets should be created prior to the system becoming operational. If this is not feasible, it is important to estimate how many key sets may be needed over time and to allocate sufficient space so that additional key sets can be created without the need for a server shutdown.

To create a user keyset

1. Launch the administration utility from the Start menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**.

2. Select the spare keyset space on which to create the keyset from the **Active Adapters** list.

3. Open the **Keyset** menu and choose **Create Keyset**.

    The administration utility prompts for the **Keyset Name** and the **Keyset Password**.

4. Enter the required information into the fields provided and click **OK** to create the new keyset. Note that the name of the keyset should match with the user login name. The new keyset displays under the device as shown below.
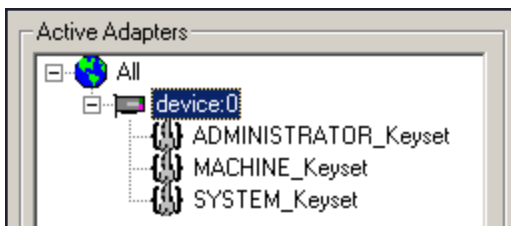
Commencing Normal Operation

Following the above steps, ProtectToolkit-M is ready for use. Additional configuration may be required in certain circumstances, as covered in "Configuration Options" below.

The device administrator or user might need to perform various operational tasks during runtime usage. These tasks are covered in the following chapters.

Configuration Options

Registry Configuration

Entries made in the Windows registry during the installation of ProtectToolkit-M are documented in "Registry Configuration" on page 79. These may be amended by expert users if required. Generally, the default values will not need to be changed. The exceptions are the Debug Level and User Keyset Password entries used to control error log file creation and silent user keyset login respectively. See the sections below for further information.

Error Log File Creation

The Debug Level registry key controls error log file creation. By default, the value of this key is set so that no error log file is produced. Should it be necessary to create an error log file, see "debugLevel" on page 80 for more options.

Silent User Keyset Login

While access to the Machine and System keysets is open, access to a User keyset requires authentication.

Typically, User keyset access authentication is achieved by prompting the user for a password when access is requested. This is not convenient/permissible in all situations, so silent user keyset login is also available.

To activate silent User keyset login

Add the following value to the Windows registry:

```
HKEY_CURRENT_USER/Software/Safenet/ProtectToolkit M/
UserKeysetPassword=<password>
```
where <password> is the clear text password for the User keyset.

Since this value is located in the **Current Users** registry hive (which is only accessible/visible when a user authenticates themselves to the Windows operating system) there is no security risk, even though the password is stored in the clear.

Work Load Distribution (WLD)

If required, more than one hardware security module (HSM) can be used in a Work Load Distribution (WLD) configuration.

WLD allows work to be balanced across a system by transferring units of work among HSM processing modules during execution, reducing the demand on any particular processing module. This produces an increase in the system's overall throughput of processing tasks. Using multiple HSMs in this way also provides redundancy. If an HSM goes down, the work will automatically be shared amongst the remaining operational HSMs.

For further information, including implementation and maintenance instructions, refer to "Work Load Distribution" on page 78.

SafeNet KSP for CNG Registration Utilities

CNG (Cryptography Next Generation) is Microsoft's cryptographic application programming interface (API), replacing the older Windows cryptoAPI (CAPI). CNG adds new algorithms along with additional flexibility and functionality. Thales provides SafeNet CSP for applications running in older Windows crypto environments (running CAPI), and SafeNet KSP for newer Windows clients (running CNG). Consult Microsoft documentation to determine which one is appropriate for your client operating system.

KSP must be installed on any computer that is intended to act via CNG as a client of the HSM, running crypto operations in hardware. You need KSP to integrate SafeNet cryptoki with CNG and to use the newer functions and algorithms in Microsoft IIS.

After you register your HSM tokens with SafeNet KSP, your KSP code should work the same whether a SafeNet HSM (crypto provider) or the default provider is selected.

The SafeNet KSP utilities are installed in **C:\Program Files\SafeNet\Protect Toolkit 5\KSP**. The installation includes the following utilities:

> **"kspcmd" below**

- "Configuring the KSP Using the Command Line" on the next page

> **"KspConfig" on page 23**

- "Configuring the KSP Using the GUI" on page 23

> **"ksputil" on page 24** — Used to display and manage partition keys that are visible to the KSP

kspcmd

You can use this utility (**C:\Program Files\SafeNet\Protect Toolkit 5\KSP\kspcmd.exe**) to register the KSP library and partitions via the Windows command line.

> **NOTE**   To register the library and partitions using a GUI, use **"KspConfig" on page 23**. It is unnecessary to use both utilities.

Syntax
kspcmd.exe

> **library** <path**cryptoki.dll**>
> **password /s** <token_label> [**/u** <username>] [**/d** <domain>]
> **usagelimit**
> **viewslots**

Argument	Shortcut	Description
library <path**cryptoki.dll**>	l	Register the library and associated provider names with KSP.

Argument	Shortcut	Description
password	**p**	Register the designated token and its user PIN to the KSP. You can specify the following options: **/s** <token_label> [Mandatory] The label of the token being registered to the KSP. **/u** <username> [Optional] The username to register for this partition. If this is not specified, all users on the client will be able to access this partition via KSP. **/d** <domain> [Optional] The domain to register for this token.
usagelimit	**u**	Set the maximum usage limit for RSA keys using KSP. Enter **0** to register unlimited uses.
viewslots	**v**	Display the registered slots by user/domain.

Configuring the KSP Using the Command Line

You can use the **"kspcmd" on the previous page** command-line tool to configure the KSP for use with your tokens. The user must complete this procedure using Administrator privileges on the client.

You can register the following user/domain combinations with the KSP:

> **Administrator** user with the domain specific to the client. Default Windows domains are in the format **WIN-XXXXXXXXXXX**.

> **SYSTEM** user with the **NT-AUTHORITY** domain

The configuration tool registers a token PIN to a specific user, so that only that user can unlock the partition.

To configure the KSP using the command line

1. In a command line, navigate to the SafeNet KSP install directory and register the **cryptoki.dll** library to the KSP.

   "kspcmd" on the previous page library /s <path**cryptoki.dll**> [**/u** <username>] [**/d** <domain>]

2. Register the designated token and its user PIN to the KSP.

   "kspcmd" on the previous page password /s <token_label> [**/u** <username>] [**/d** <domain>]

   You are prompted to enter the user PIN for the token.

3. [Optional] Display the registered slots to ensure that registration is complete.

   "kspcmd" on the previous page viewslots

4. [Optional] Set the maximum usage limit for RSA keys using KSP.

   "kspcmd" on the previous page usagelimit

   You are prompted to enter a usage limit. Enter **0** to register unlimited uses.

KspConfig

You can use this tool (**C:\Program Files\SafeNet\Protect Toolkit 5\KSP\KspConfig.exe**) to register the KSP library and partitions using a GUI.

> **NOTE** To run **KspConfig.exe**, you must first install the Visual Studio 2015 redistributable package from Microsoft.
>
> To register the library and partitions using the command line, use **"kspcmd" on page 21**. It is unnecessary to use both utilities.

Configuring the KSP Using the GUI

You can use the **"KspConfig" above** utility to configure the KSP for use with your tokens. The user must complete this procedure using Administrator privileges on the client.

You can register the following user/domain combinations with the KSP:

> **Administrator** user with the domain specific to the client. Default Windows domains are in the format **WIN-XXXXXXXXXXX**.

> **SYSTEM** user with the **NT-AUTHORITY** domain

The configuration tool registers a token PIN to a specific user, so that only that user can unlock the partition.

To configure the KSP using the GUI

1. In Windows Explorer, navigate to the SafeNet KSP install directory and launch **"KspConfig" above** as the **Administrator** user.

2. In the left panel, double-click **Register or View Security Library**. Enter the filepath to **cryptoki.dll** or click Browse to locate it.

   C:\Program Files\SafeNet\Protect Toolkit 5\Protect Toolkit C SDK\bin\hsm\cryptoki.dll
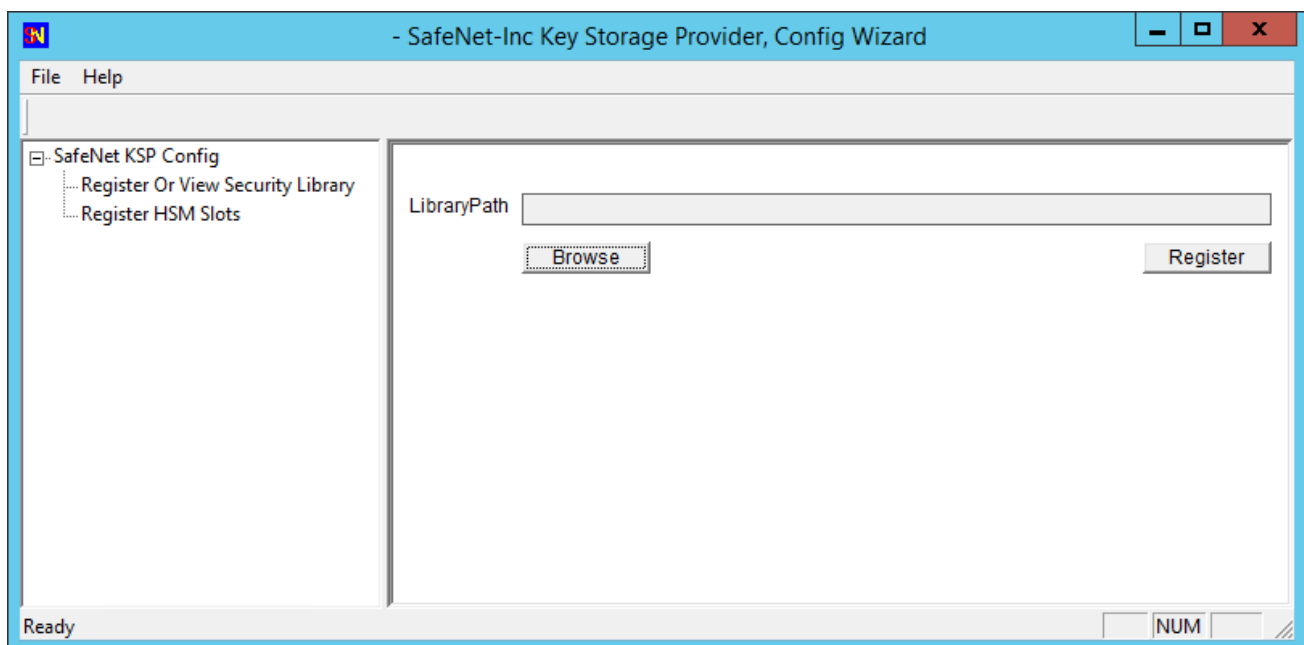
   Click **Register** to complete the registration.

3. In the left panel, double-click **Register HSM Slots**. Select the **Administrator** user, client domain, and an available token to register. Enter the user PIN and click **Register Slot**.
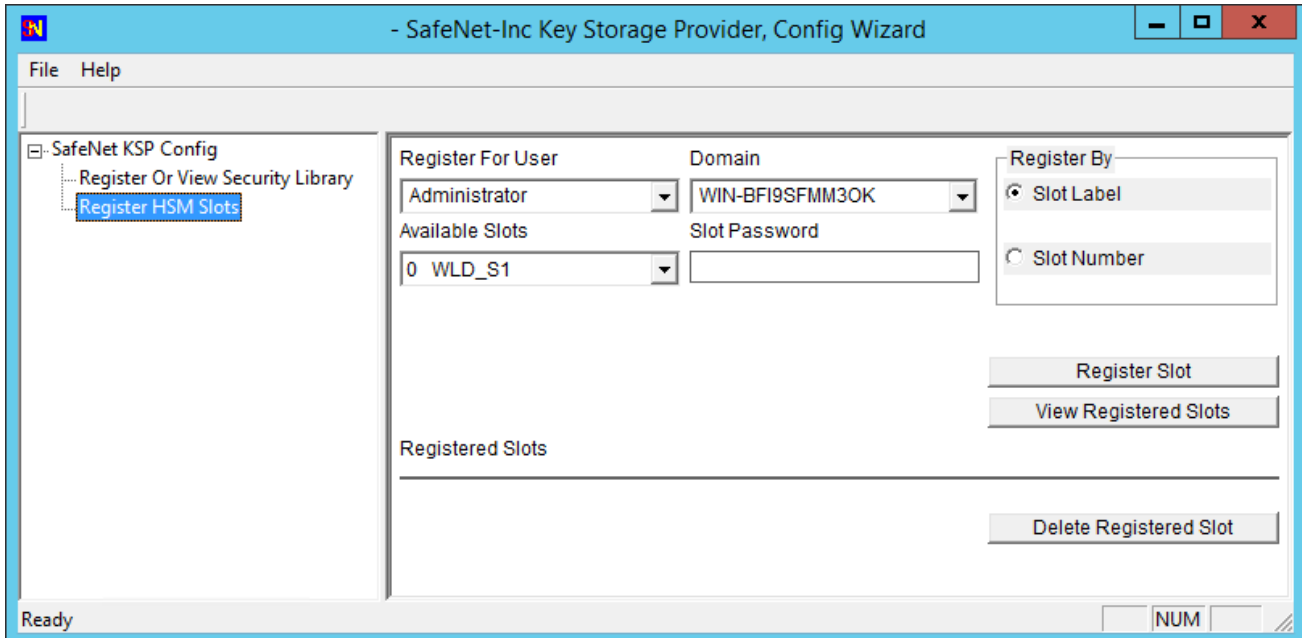
4. Select the **SYSTEM** user and **NT-AUTHORITY** domain and register for the token.

5. Repeat steps 3-4 for any other available tokens you want to register with the KSP.

    You can now begin using your applications to perform crypto operations on the registered tokens.

ksputil

KSP binds machine keys to the hostname of the crypto server that created the keys. You can use the **"ksputil" above** utility to display and manage keys that are visible to the KSP.

Syntax
ksputil

   clusterkeys /s <tokennum> **/n** <keyname> **/t** <target>
   listkeys /s <tokennum> [**/user**]

Argument	Shortcut	Description
clusterkeys	**c**	Bind a specified keypair to a different server domain. Note that this does not change the bindings of existing keys; it creates a copy of the original keypair that is bound to the new domain.
		Available options:
		/s <tokennum> [Mandatory] The number of the token where the key(s) are located.
		/n <keyname> [Mandatory] The name of the key(s) to bind to the new domain.
		/d <domain> [Mandatory] The domain to which keys will be bound.
listkeys	**l**	Display a list of KSP-visible keys.
		Available options:
		/s <tokennum> [Mandatory] The number of the token where the key(s) are located.
		/user [Optional] List keys bound to the currently logged-in user/hostname.

Configuring IIS7 (Win2008) with CNG

This section describes how to configure Microsoft Internet Information Services 7 (IIS7) on Windows Server 2008 for use with CNG.

To configure IIS7 on Windows Server 2008 for use with CNG

1. Install and configure your HSM.

2. Install and configure KSP:

   a. Register your **cryptoki.dll** file
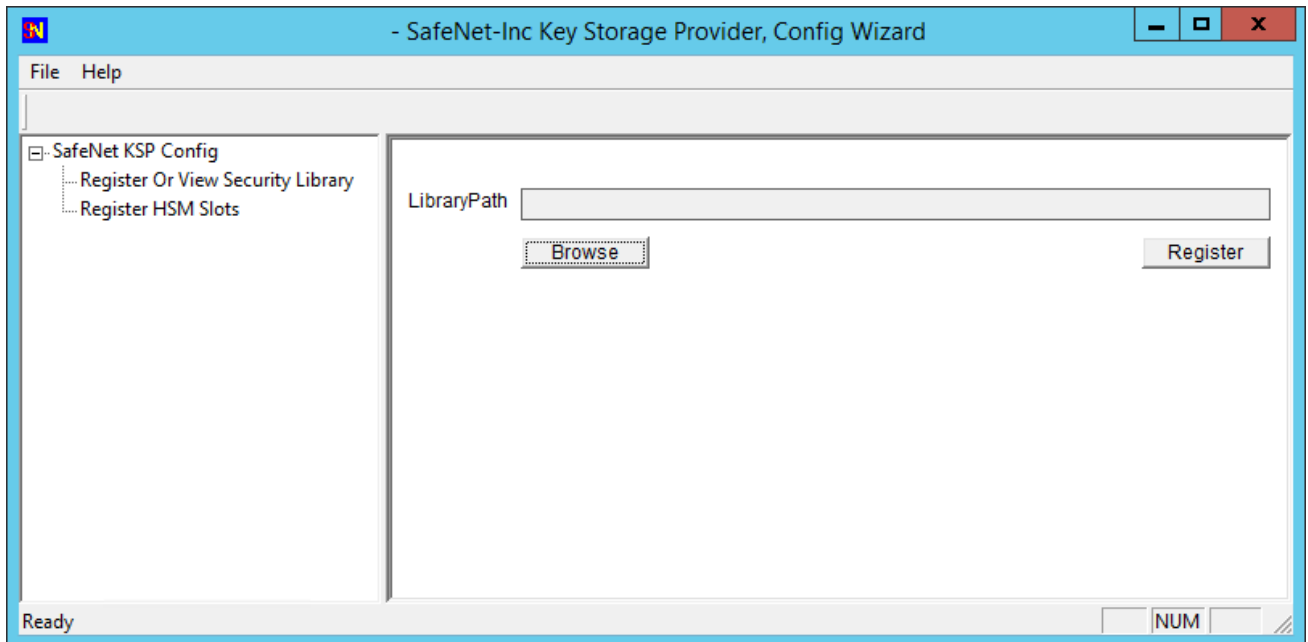
b. Register your slot for Administrator/(Server name or Domain name) and again for System/NT Authority.
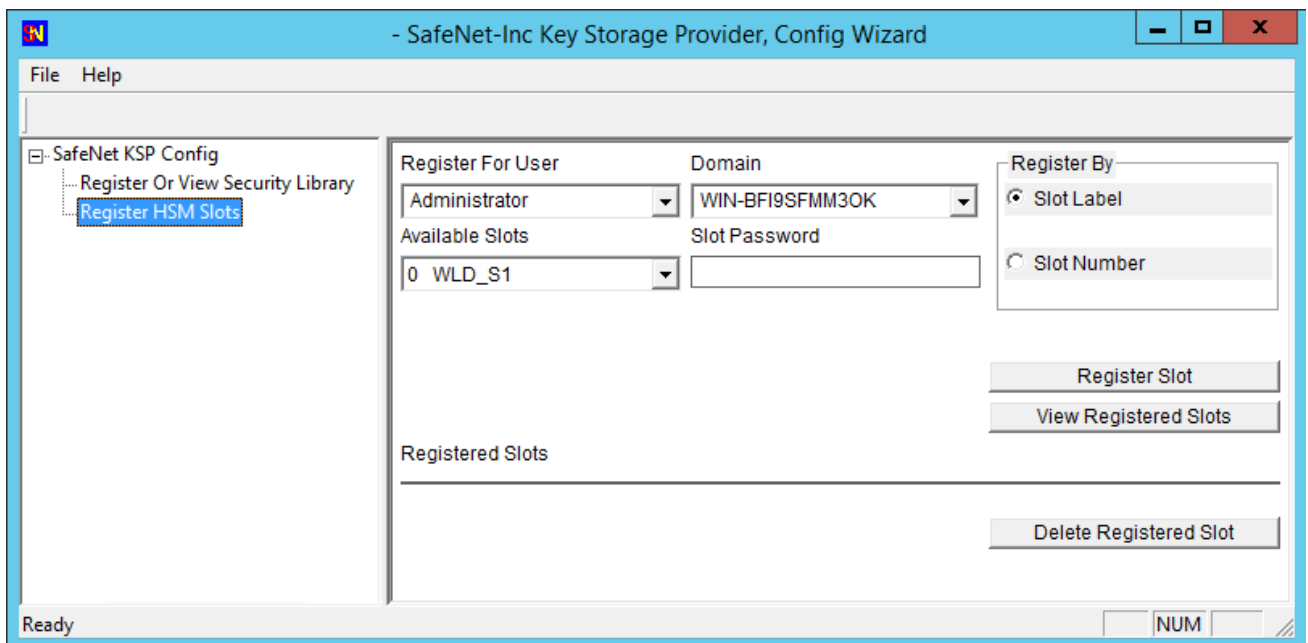
3. Create a policy file to generate a cert request. Normally, you can do this directly through the GUI, but the KSP is not yet recognized through the GUI. The policy file (call it **policy.inf**) should look like this:

```
[New Request]
      KeyUsageProperty = "NCRYPT_ALLOW_DECRYPT_FLAG"
      Providertype = 1
      RequesterName = OTT1-HANNIBAL\Administrator
      RequestType = PKCS10
      ProviderName = "SafeNet Key Storage Provider"
      Subject = "CN=OTT1-HANNIBAL,OU=Eng,O=SafeNet-Inc,
L=Ottawa,S=Ontario,C=CA"
      KeyContainer = "OTT1-HANNIBAL"
      MachineKeySet = true
```

```
HashAlgorithm = sha1
KeyAlgorithm = RSA
KeyLength = 2048
```

4. Using the above file, create your cert request:

   ```
   C:\>certreq -new policy.inf cert.req
   ```

5. Submit your cert request to a CA and obtain a signed cert, and the root cert of the CA. Move these certificates to your IIS server.

6. Install the root certificate:

   a. Open the root cert file and select "Install Certificate."

   b. At the Welcome screen, click **Next**.

   c. You'll need to specify the Certificate Store to be used. Select the "Place all certificates in the following store" radio button, and click the "Browse…" button.

   d. In the Select Certificate Store window that opens, put a check in the "Show physical stores" box, locate and expand Trusted Root Certification Authorities and select "Local Computer" then click **OK**.

7. Open Server Manager and select "Add Roles" to install Web Server (IIS). Configure to your needs, though the default options will do for the purposes of this document.

8. When the installation is complete, expand the Roles tree from the left-hand pane, then expand Web Server (IIS) and select "Internet Information Services (IIS) Manager," then select the object name (most likely your server's name) from the Connections pane, as shown below:
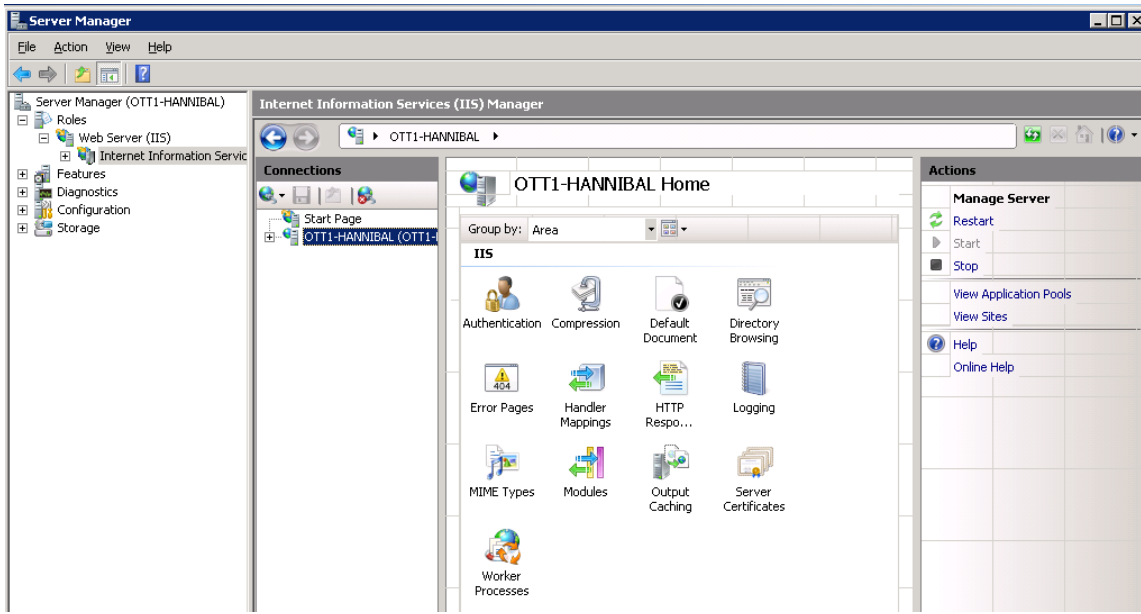
9. Under the Home pane, open Server Certificates, then select "Complete Certificate Request…" from the Actions pane.

10. Complete the form that opens; select the path to your certificate and choose a friendly name for said certificate and click **OK**:
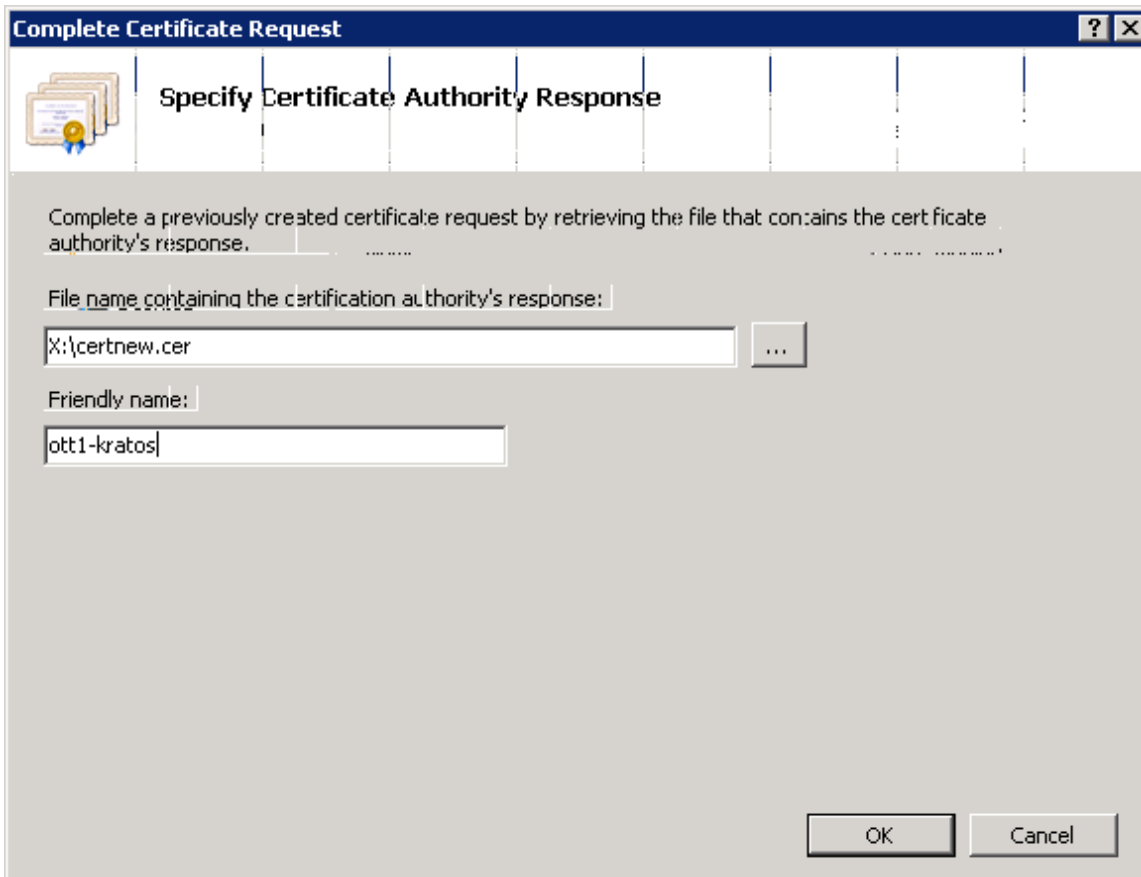
11. The certificate list will then be populated by the certificate you specified:
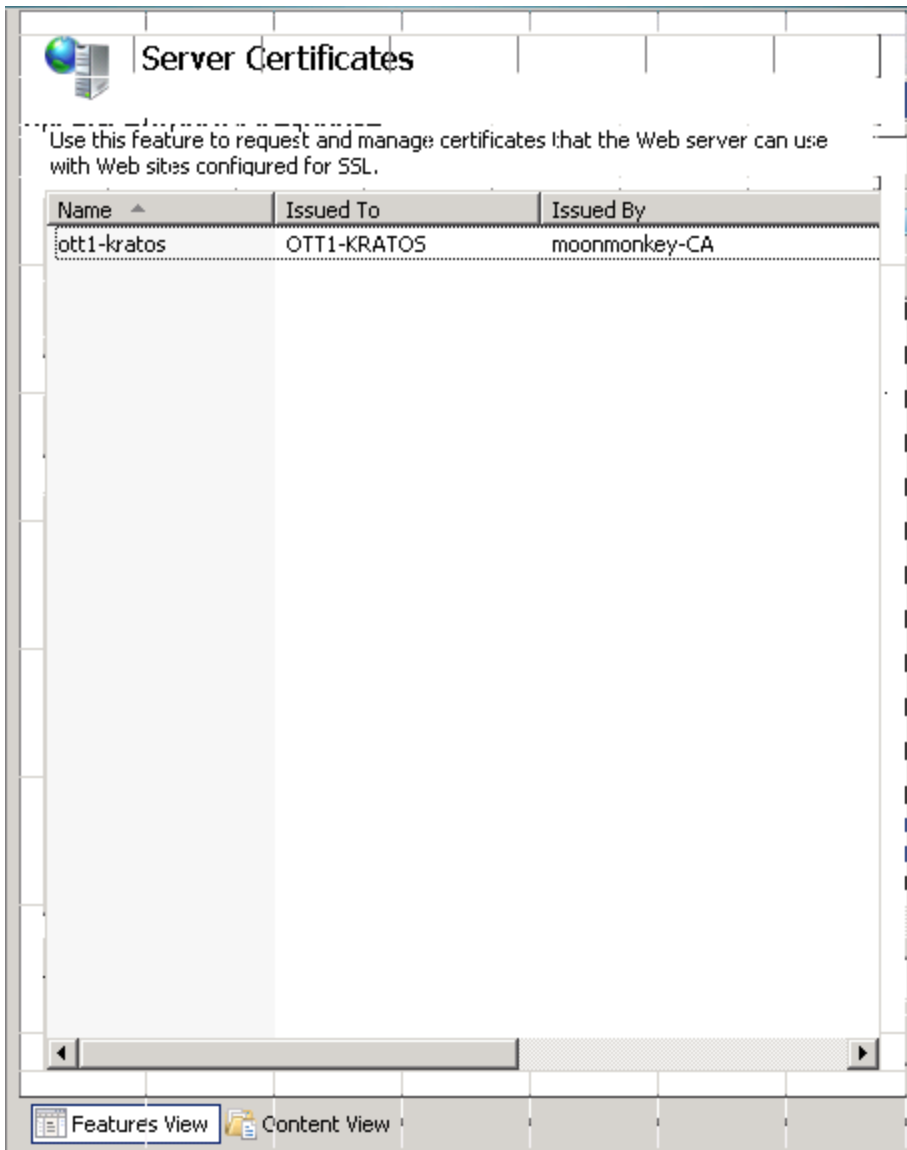
12. Under the Connections pane, expand the server hostname tree (in the example below, OTT1-HANNIBAL), then expand the Sites tree, and select "Default Web Site":
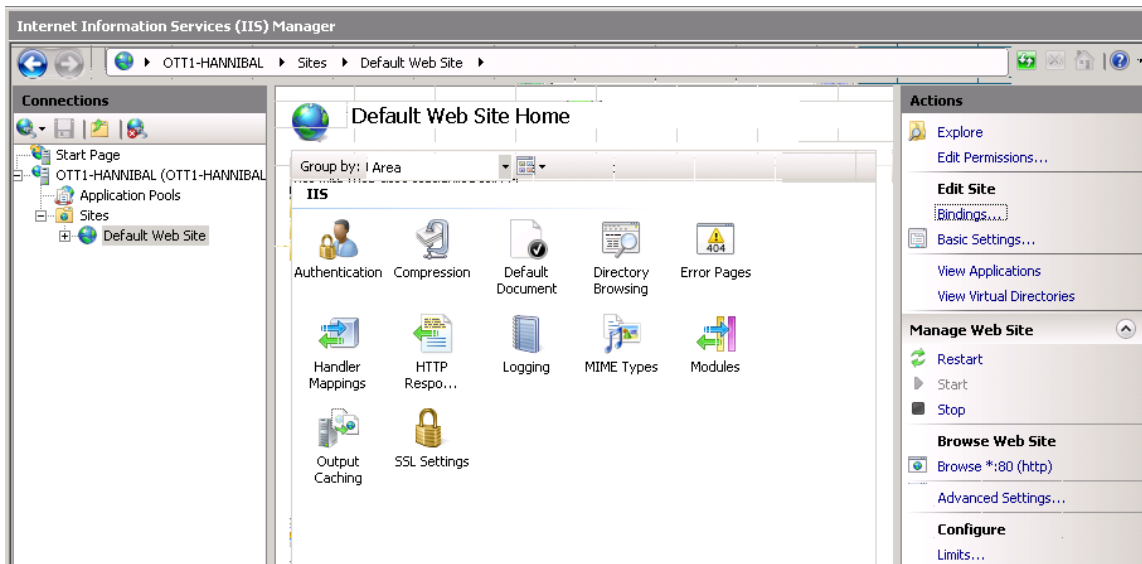
13. Select "Bindings" from the Actions pane on the right-hand side. This opens the Site Bindings box.
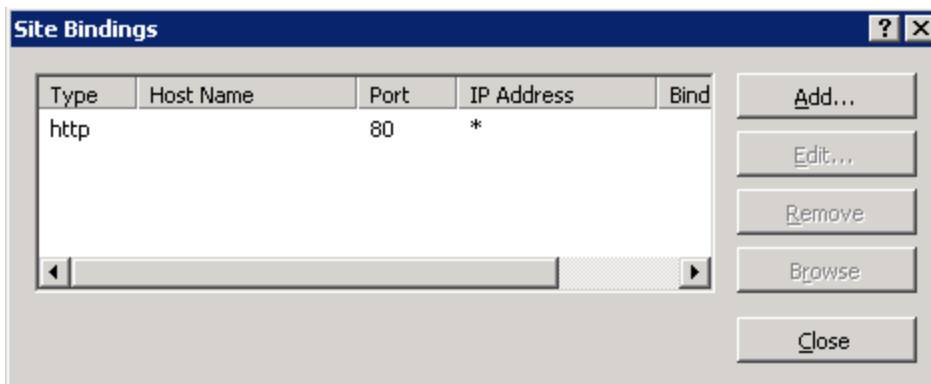
14. Click **Add**, and make the following selections:

Type	https
IP Address	Can be left as "All unassigned"
Port	Can be left as 443
SSL certificate	Select the friendly name you assigned earlier to your certificate when your completed the cert request.
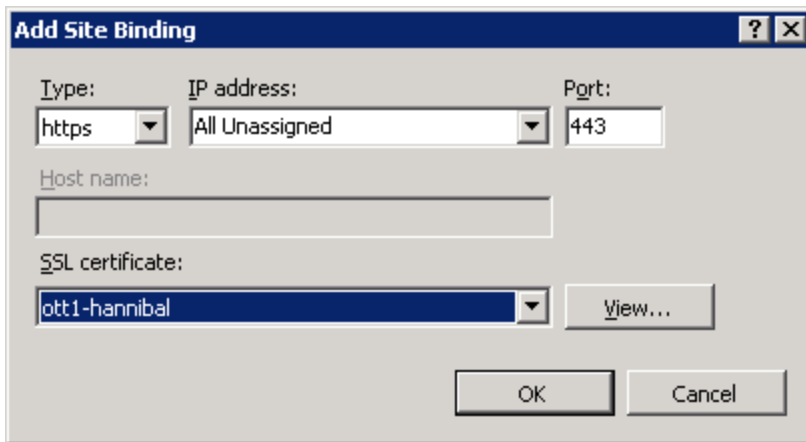
Click **OK** to continue.

15. Under the actions pane, you will now have a link labeled "Browse *:443 (https)" (this may appear slightly different, depending on the IP Address options you set in the previous step).
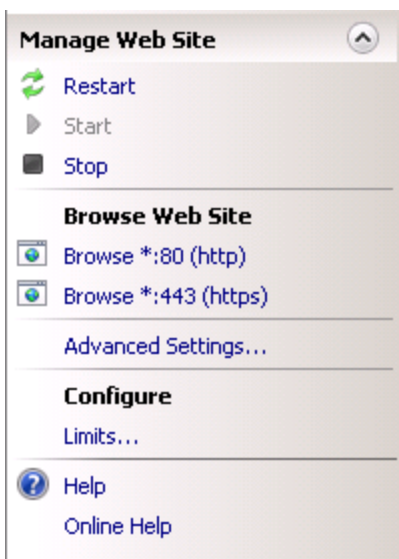
16. Select this link and it will show you your default webpage over a secure connection. Configure your website as needed.

CHAPTER 3:   Administrative Tasks

This section describes the operational procedures a device administrator may perform during normal ProtectToolkit-M operation. It contains:

> "Changing the Device Administrator Password" below

> "Allocating Keyset Space" on the next page

> "De-allocating Keyset Space" on the next page

> "Creating User Keysets" on page 34

> "Deleting a Keyset" on page 34

> "Setting the Adapter Transport Mode" on page 35

> "Correcting Clock Drift " on page 36

> "Viewing and Purging the HSM Event Log" on page 36

> "Checking and Upgrading HSM Firmware" on page 37

> "Tampering the HSM" on page 38

> "Backing up a Keyset" on page 38

> "Restoring a Keyset" on page 39

> "Enabling Private Key Clear Export" on page 39

Changing the Device Administrator Password

The device administrator can perform a password change at any time and on any token.

Changing the device administrator password may only be performed by the device administrator, using the ProtectToolkit-M administration utility.

To change the device administrator password

1. Launch the administration utility from the **Start** menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**.

2. Select the device on which to change the device administrator password from the **Active Adapters** list.

3. Open the **Adapter** menu and choose **Change Device Admin Password**.

4. The administration utility will now prompt for the current and new device administrator password.

> **NOTE**   Any existing backups of the MACHINE and SYSTEM keysets will no longer be useful following a device administrator password change, because the backup key is generated from the password. New backups must be created after changing the password.

Allocating Keyset Space

When additional user keysets are required, the system will need to be configured for additional keyset space. The number of allocated keyset spaces determines how many separate ProtectToolkit-M users, and keysets, the system can have.

Allocation of keyset space is the responsibility of the device administrator and is performed using the ProtectToolkit-M administration utility.

To allocate keyset space

1. If it is not already open, launch the administration utility from the **Start** menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**.

2. Select the device on which to create keyset space from the **Active Adapters** list.

3. Open the **Adapter** menu and select **Allocate Space**.

4. The administration utility will now prompt for the device administrator password.

5. Following correct password entry, the new keyset space will be displayed under the device.

> **NOTE**   Additional keyset space cannot be allocated if an application is currently using ProtectToolkit-M. For more information about this issue, see Session Exists Error in the "ProtectToolkit-M Troubleshooting" section of the *ProtectServer HSM and ProtectToolkit Troubleshooting Guide*.

To check if an application has any open sessions to ProtectToolkit-M, check the value shown next to Application Count in the System section of the administration utility. This will need to be "1", and the Total Session Count must be "0" in order for keyset de-allocation to succeed.

> **NOTE**   If the value of Application Count is shown as "UNAVAILABLE", your HSM firmware doesn't support live application counting. In such a case, it is advisable to upgrade the HSM firmware to the latest version. Please refer to "Checking and Upgrading HSM Firmware" on page 37.

De-allocating Keyset Space

If there are keyset spaces which are not likely to be used, it is good practice to de-allocate spare spaces from the HSM in order to prevent memory exhaustion or invalid use.

De-allocation of keyset space is the responsibility of the device administrator and is performed using the ProtectToolkit-M administration utility.

To de-allocate keyset space

1. Launch the administration utility from the Start menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**.

2. Select any spare space from the **Active Adapters** list.

3. Open the **Keyset** menu and choose **Deallocate Space**.

4. The administration utility will now prompt for the device administrator password.

5. Following correct password entry, the spare keyset space will be removed from the list of displayed keysets.

> **NOTE** Keyset space cannot be de-allocated if an application is currently using ProtectToolkit-M. For more information about this issue, see Session Exists Error in the "ProtectToolkit-M Troubleshooting" section of the *ProtectServer HSM and ProtectToolkit Troubleshooting Guide*.

To check if an application has any open sessions to ProtectToolkit-M, check the value shown next to Application Count in the System section of the administration utility. This will need to be "1", and the Total Session Count must be "0" in order for keyset de-allocation to succeed.

> **NOTE** If the value of Application Count is shown as "UNAVAILABLE", your HSM firmware doesn't support live application counting. In such a case, it is advisable to upgrade the HSM firmware to the latest version. Please refer to "Checking and Upgrading HSM Firmware" on page 37.

Creating User Keysets

In order to create a new keyset for a specific user, you will first have to make sure that there is spare keyset space available on the HSM. This can be done by opening the ProtectToolkit-M administration utility.

Should no spare space be available, you will have to allocate additional keyset space on the HSM. For details, please refer to the appropriate section above.

> **NOTE** Users can create keysets for themselves once space is available.

To create a user keyset

1. Launch the administration utility from the Start menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**.

2. Select the spare keyset space on which to create the keyset from the Active Adapters list.

3. Open the Keyset menu and choose "Create Keyset."

4. The administration utility will now prompt for the Keyset Name and the Keyset Password. Enter the required information into the fields provided and press OK to create the new keyset.

> **NOTE** The name of the keyset should match with the user login name.

The new keyset is displayed under the device.

Deleting a Keyset

Deleting user keysets is the responsibility of the device administrator and is performed using the ProtectToolkit-M administration utility.

To delete a user keyset

1. Launch the administration utility from the **Start** menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**.

2. Select the desired keyset from the **Active Adapters** list.

3. Open the **Keyset** menu and choose **Delete Keyset**.

4. The administration utility will now prompt for the device administrator Password. Prior to deletion, the administration utility will prompt for confirmation that deletion is the requested operation.

5. The keyset is removed from the displayed keysets under the selected device.

> **NOTE** A keyset cannot be deleted if an application is currently using ProtectToolkit-M. For more information about this issue, see Session Exists Error in the "ProtectToolkit-M Troubleshooting" section of the *ProtectServer HSM and ProtectToolkit Troubleshooting Guide*.

To check if an application has any open sessions to ProtectToolkit-M, check the value shown next to Application Count in the System section of the administration utility. This will need to be "1", and the Total Session Count must be "0" in order for the keyset deletion to succeed.

> **NOTE** If the value of Application Count is shown as "UNAVAILABLE", your HSM firmware doesn't support live application counting. In such a case, it is advisable to upgrade the HSM firmware to the latest version. Please refer to "Checking and Upgrading HSM Firmware" on page 37.

Setting the Adapter Transport Mode

The adapter transport mode allows an adapter HSM to be removed from the host system PCI bus without causing a tamper condition. A tamper will remove all sensitive material from the adapter including the adapter configuration, all keys, and certificates.

Setting the adapter transport mode is the responsibility of the device administrator and is performed using the ProtectToolkit-M administration utility.

To set the adapter transport mode

1. Launch the administration utility from the **Start** menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**.

2. Select the desired adapter from the **Active Adapters** list.

3. Open the **Adapter** menu and choose **Set Transport Mode**.

4. The device administrator is now prompted to choose one of three possible transport modes:

   • **None**: To be applied when adapter is installed and configured. This mode will tamper the adapter if removed from the PCIe Bus.

   • **Single**: Adapter will not be tampered after removal from the PCIe bus. Adapter will automatically change to "None" Transport Mode the next time the adapter is reset or power is removed and restored.

   • **Continuous**: Adapter will not be tampered by being removed from the PCIe bus.

5. The administration utility will now prompt for the device administrator password.

> **NOTE** The transport mode does not disable the tamper response mechanism entirely. Any attempt to physically attack the adapter will still result in a tamper response.

Correcting Clock Drift

Due to host system and HSM timing differences, such as clock drifts, it may become necessary, at certain stages, to adjust the internal time on the HSM.

Note that the HSM clock value cannot be specified directly. It is only possible to synchronize the HSM clock with the host system clock.

Synchronizing the HSM clock is the responsibility of the device administrator and is performed using the ProtectToolkit-M administration utility.

To adjust the HSM clock

1. Launch the administration utility from the Start menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**.

2. Select the desired HSM from the **Active Adapters** list.

3. Open the **Adapter** menu and choose **Sync Clock**.

4. The administration utility will now prompt for the device administrator password. Correct entry of the password will result in clock synchronization.

Viewing and Purging the HSM Event Log

ProtectServer HSMs maintain event logs in order to provide a means of tracking serious hardware or consistent operational faults. It is the device administrator's task to view and purge HSM event log data.

When the HSM event log is full, the HSM will no longer store new event records and will need to be purged.

> **NOTE** The HSM event log cannot be purged until it is full.

To view the HSM event log

1. Launch the administration utility from the Start menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**.

2. Select the desired HSM from the **Active Adapters** list.

3. Open the **Adapter** menu and select **View Event Log**.

4. The administration utility will now prompt for the device administrator password. Correct entry of the password will result in the event log being displayed.

5. The event log is shown as a series of pages. If there are more than one page of event log entries, the operator can navigate through the pages via the **first, prev, next, last** buttons.

To purge the event log

1. Launch the administration utility from the **Start** menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**.

2. Select the desired HSM from the **Active Adapters** list.

3. Open the **Adapter** menu and choose **Purge Event Log**.

4. The administration utility will now prompt for the device administrator password. Correct entry of the password will result in the event log being purged.

> **NOTE**   The event log can also be purged via the **View Event Log** dialog by clicking the **Purge** button.

Checking and Upgrading HSM Firmware

The ProtectToolkit-M firmware that operates on the HSM can be upgraded to newer versions. This facility will only allow the HSM to be upgraded to firmware versions that have been digitally signed by SafeNet.

The firmware update package, formerly supplied identically on both the ProtectToolkit-C and ProtectToolkit-M DVDs, is now available only from the Thales eService Support Portal, along with a specific Update instruction document. The instructions in this section are generic in nature, intended only to show the scope of the operation. The authoritative, detailed instructions are always in the Update document that accompanies the update package. The latest versions of the client software and HSM firmware can be found on the Thales Technical Support Customer Portal. See "Support Contacts" on page 9 for more information.

Prior to performing a firmware upgrade, the firmware upgrade file should be checked to confirm that it is a valid SafeNet upgrade file.

Depending on the security policy in place, the HSM may perform a soft-tamper before the upgrade process is executed. This tamper will erase all key and configuration data on the HSM. Prior to performing a firmware upgrade, ensure that you have performed the following:

>   All important user data and keys have been backed up

>   The current HSM configuration has been noted

>   All applications using the HSM have been closed - this may require some services to be stopped (e.g. Certificate Services, IIS)

Upgrading the HSM firmware is the responsibility of the device administrator and is performed using the ProtectToolkit-M administration utility.

To check the firmware upgrade file

1. If it is not already open, launch the administration utility from the Start menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**.

2. Select the desired HSM from the **Active Adapters** list.

3. Open the **Adapter** menu and choose **Check Firmware File**.

4. The administration utility will now prompt for the location of the firmware upgrade file.

5. The utility will show if the file is validated, or corrupt.

To upgrade the HSM firmware

1. If it is not already open, launch the administration utility from the **Start** menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**.

2. Select the desired HSM from the **Active Adapters** list.

3. Open the **Adapter** menu and choose **Upgrade Firmware**.

4. The administration utility will now prompt for the location of the firmware upgrade file.

5. If the file is validated, the utility will prompt for the device administrator password. Correct password entry will proceed with the firmware upgrade.

> **NOTE**   During the firmware upgrade, **gmadmin** will appear to stop functioning because the process can take up to 40 seconds to complete. If the utility does not respond after a number of minutes, shut down your system and reboot. If problems persist, see "Support Contacts" on page 9 to contact Thales technical support.

Tampering the HSM

The tampering of the HSM may be necessary at the end of its lifecycle or any other security-sensitive event that requires all stored data to be immediately destroyed.

A tamper formats the secure memory of the HSM and thereby erases all configuration and key data.

Due to the highly destructive nature of this action, tampering the HSM is the responsibility of the device administrator and is performed using the ProtectToolkit-M administration utility. Note that this action also requires that all sessions have been closed and that no user is accessing the HSM.

To tamper the HSM

1. Launch the administration utility from the **Start** menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**.

2. Select the desired HSM from the **Active Adapters** list.

3. Open the **Adapter** menu and choose **Tamper**.

4. The administration utility will now prompt for the device administrator password. Correct entry of the password will show a final confirmation dialog to ensure that this is the desired course of action.

5. Press **Yes** to tamper the HSM, or **No** to Cancel.

> **NOTE**   The above action cannot tamper the HSM while other applications are active. The administration utility will indicate if the tamper operation was successful. A white cross on a red background shown next to the selected HSM indicates that the device is tampered.

Backing up a Keyset

Individual, HSM stored keysets can be backed up to a secure disk file or one or more smart cards. Backed up keysets can then be restored in the event of a tamper to the HSM or if the keysets are otherwise lost.

> **NOTE**   Users are responsible for backing up their own keysets and the ProtectToolkit-M device administrator is responsible for backing up the MACHINE and SYSTEM keysets.

A triple-DES BackupKey is used to encrypt each keyset prior to storage on a smart card. A different BackupKey is automatically created for each keyset when the keysets are created but these keys are not visible under normal ProtectToolkit-M operation. A BackupKey for a keyset is derived from a combination of the password used to secure that particular keyset and the keyset name. In the case of the MACHINE and SYSTEM keysets, the device administrator's password and the keyset name are used to derive the key. Thus to restore a keyset that was previously backed up, the same password and keyset name must be used.

> **NOTE**   You cannot backup ProtectToolkit-M FIPS mode keysets. If your organization requires you to use FIPS mode tokens we recommend securing your keyset using an NofM schema. See the Key Splitting Scheme Selection in the "Operational Tasks" section of the *ProtectToolkit-C Administration Guide* for more information about NofM.

Keyset backup is accomplished with the command line tool **ctkmu** or the GUI tool **kmu**. Both the **ctkmu** utility and the **kmu** utility are included in the ProtectToolkit-C package. See CTKMU in the "Command Line Utilities Reference" section of the *ProtectToolkit-C Administration Guide* and the "Key Management Utility (KMU) in the *ProtectToolkit-J Reference Guide* for more information about these utilities.

See Key Backup Procedure in the "Operational Tasks" section of the *ProtectToolkit-C Administration Guide* for example procedures and more information.

Restoring a Keyset

Precautions

> To restore a key that was previously backed up, the same password and keyset name must be used.

> Extreme care should be taken to ensure that keys which are being restored DO NOT already exist on the ProtectToolkit-M system. A restore operation DOES NOT replace existing keys, but will restore a second instance of the same key pair. If you have accidentally created multiple instances of the same key pair, ProtectToolkit-M will mark the affected keyset as being invalid.

Keyset restore is accomplished with the command line tool **ctkmu** or the GUI tool **kmu**. Both the **ctkmu** utility and the **kmu** utility are included in the ProtectToolkit-C package. See CTKMU in the "Command Line Utilities Reference" section of the *ProtectToolkit-C Administration Guide* and Key Management Utility (KMU) Reference in the *ProtectToolkit-J Reference Guide* for more information about these utilities.

See Key Backup Procedure in the "Operational Tasks" section of the *ProtectToolkit-C Administration Guide* for example procedures and more information.

Enabling Private Key Clear Export

In order to support the key archival process, it must be possible for the host machine to obtain the value of the private key in the clear.

Due to the inherent security risks, the Allow Clear Export of Private Keys flag controls whether this value can be obtained. This is a "secure configuration item".

A secure configuration item is one which is open for reading, but requires authentication for writing. Such configuration items are stored on the HSM and protected by the password of the device administrator.

If Allow Clear Export of Private Keys flag is set to `True`, then it is possible to obtain the value of a private key in the clear using the Microsoft Crypto API (MSCAPI) (causing the key archival process to succeed).

If Allow Clear Export of Private Keys is set to `False`, then any requests to obtain the value of a private key in the clear are denied (causing the key archival process to fail).

The value of the Allow Clear Export of Private Keys flag can be changed using the ProtectToolkit-M Administration Utility.

To set or clear the Allow Clear Export of Private Keys flag

1. Launch the administration utility from the Start menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**.

2. Select the desired HSM from the **Active Adapters** list.

3. Open the **Adapter** menu and choose **Set Secure Configuration**. The **Set Secure Configuration** dialog box displays.

4. Set or clear the Allow Clear Export of Private Keys flag as required, then click **OK** to action the change.

CHAPTER 4:   User Tasks

This section describes the operational procedures a User may perform during normal ProtectToolkit-M operation. It contains sections on the following operations:

Creating Keysets

To create a new keyset, first ensure that there is enough keyset space available on the HSM. This can be confirmed by opening the ProtectToolkit-M Administration Utility.

If there is not enough space available, an administrator will have to allocate additional keyset space on the HSM. For details please refer to the previous chapter.

To create a keyset

1. Launch the administration utility from the **Start** menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**.

2. Select the spare keyset space on which to create the keyset from the **Active Adapters** list.

3. Open the **Keyset** menu and choose **Create Keyset**.

4. The administration utility now prompts for the Keyset Name and the Keyset Password. Enter the required information into the fields provided and press **OK** to create the new keyset.

    > **NOTE** The name of the keyset should match with the user login name.

5. The new keyset is displayed under the device.

Changing a Keyset Password

A keyset password may need to be changed periodically. A keyset password is changed by the keyset owner, using the ProtectToolkit-M keyset management utility.

To change the keyset password

1. Launch the keyset management utility from the **Start** menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmksm**.

2. From the displayed list, select the desired keyset.

3. Open the **Keyset** menu and choose **Change Password**.

4. The user is prompted for the current and new keyset password. Enter the required information into the fields provided and press **OK** to change the password.

> **NOTE** Any existing keyset backups will no longer be useful following a keyset password change, because the backup key is generated from the password. New backups should be created after changing the password.

Adding a Key Container

Key containers are created within a user's keyset, so that the keyset can hold key pairs. The keyset owner can add a key container using the ProtectToolkit-M keyset management utility.

To add a key container

1. Launch the keyset management utility from the **Start** menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmksm**.

2. From the displayed list, select the desired keyset.

3. Open the **Keyset** menu and choose **Add Container**.

4. The user is prompted for the keyset password and key container name. Enter the required information into the fields provided and press **OK** to create the key container.

Removing a Key Container

Key containers which are no longer required or hold obsolete key pairs may be removed from a keyset.

Removing a key container is performed by the keyset owner, using the ProtectToolkit-M keyset management utility.

To remove a key container

1. If it is not already open, launch the keyset management utility from the **Start** menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmksm**.

2. Select the keyset container which you wish to remove.

3. Open the **Keyset** menu and choose **Remove**.

4. The user is prompted for the keyset password and confirmation that the container removal is the required action. Press **OK** to remove the key container.

Generating a Key Pair

Key pairs are used by Crypto API to encrypt or sign data. There are two types of key pairs, and they must be created inside a key container. Please refer to "Adding a Key Container" on the previous page.

The keyset owner can generate a key pair using the ProtectToolkit-M keyset utility.

To generate a key pair

1. Launch the keyset management utility from the **Start** menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmksm**.

2. Select the keyset container in which to generate a key pair.

3. Open the **Container** menu and choose **Generate Key Pair**.

4. The user is prompted to enter the keyset password. Correct password entry will display the **generate key pair** dialog.

5. The **generate key pair** dialog will prompt for the key usage and key size.

6. Choose **Exchange** or **Sign** depending on the required key pair usage.

7. Select a Key Size from the drop-down list.

8. Check the **Exportable** checkbox if you want to be able to back up this key pair.

9. Press **OK** to generate the key pair.

Key Usage

Key pairs generated using the keyset management utility have one of two usage attributes. These are:

> *Exchange*: This type of key pair is used to encrypt session keys for the user during normal ProtectToolkit-M operation.

> *Sign*: This type of key pair is used to create digital signatures for the user during normal ProtectToolkit-M operation.

Each user will generally require both types of keys within their particular keyset.

Key Size

Key size is an important consideration when using encryption as a security measure. When discussing key size, the value is given as a bit length, referring to how many digits are represented in the key value. As a general guideline, longer bit lengths produce longer keys and more secure encryption. However, larger key sizes slow the encryption process, due to the larger calculations involved.

> **NOTE**   If the FIPS Mode security policy is enabled, the cryptographic operations of RSA, DSA, DH, and EC algorithms are restricted to key sizes within a specified range. For more information about the size limitations of keys that are created or imported in FIPS Mode, see FIPS Mode Operational Restrictions in the "Security Policies and User Roles" section of the *ProtectToolkit-C Administration Guide*.

Deleting a Key Pair

The keyset owner can delete a key pair using the ProtectToolkit-M keyset management utility.

To delete a key pair

1. launch the keyset management utility from the **Start** menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmksm**.

2. Select the key pair you wish to delete.

3. Open the **KeyPair** menu and choose **Delete**.

4. The user is prompted to enter the keyset password. Correct password entry deletes the selected key pair.

Displaying Key Pair Properties

Key pair properties can be displayed by any user of the ProtectToolkit-M keyset management utility.

To display the properties of a key pair

1. Launch the keyset management utility from the **Start** menu by selecting **Start > Programs > SafeNet > ProtectToolkit M > gmksm**.

2. Select the key pair for which to display its properties.

3. Open the **KeyPair** menu and choose **Properties**.

   Information shown includes the following:

   • *Keyset*: Displays the name of the keyset on which the selected key pair resides.

   • *Container*: Displays the name of the key container in which the selected key pair resides.

   • *Usage*: Shows the key usage attribute of the selected key pair. This value will either be "EXCHANGE" or "SIGN".

   • *Size*: Shows the key size for the selected key pair.

   • *Private Key Held*: This indicates if the private key for the selected key pair is present as part of the key pair. Since it is possible to import a public key only, this value will either be "TRUE" or "FALSE".

   • *Exportable*: Indicates whether the selected key pair can be backed up.

Backing up and Restoring Keysets

Users are responsible for backing up their own keysets. The procedures involved in backing up and restoring key pairs or keysets are detailed in "Administrative Tasks" on page 32

> "Backing up a Keyset" on page 38

> "Restoring a Keyset" on page 39

Keyset backup or restore operations should not be attempted without thorough knowledge of the procedure and the possible consequences of incorrect actions. It is strongly advised that the device administrator is consulted prior to performing a keyset backup or restore operation.

CHAPTER 5:   Administration and User Utilities

This section outlines the following utilities:

> "Administration Utility" below

> "Keyset Management Utility" on page 52

> "CREATECERT Utility" on page 55

Administration Utility

The administration utility (**gmadmin**) is designed exclusively for use by the ProtectToolkit-M device administrator, and allows the following operations:

> Initialization of HSMs

> Synchronization of HSM clock with the system clock

> Setting of the adapter transport mode

> Setting security flags

> Changing of the ProtectToolkit-M device administrator password

> Upgrade of the HSM firmware

> Allocation of keyset space

> De-allocation of keyset space

> Creation of keysets

> Deletion of keysets

> Viewing the HSM event log

> Purging the HSM event log

> Tampering the HSM

This section is intended as a reference for **gmadmin** and outlines the following topics:

> "Starting and Exiting the Administration Utility" on the next page

> "User Interface" on the next page

> "All Adapters Menu" on page 49

> "Adapter Menu" on page 50

> "Keyset Menu" on page 51

When performing administrative tasks, the reader is strongly advised to refer to "Administrative Tasks" on page 32 for details regarding each task.

Figure 2: Administration Utility User Interface
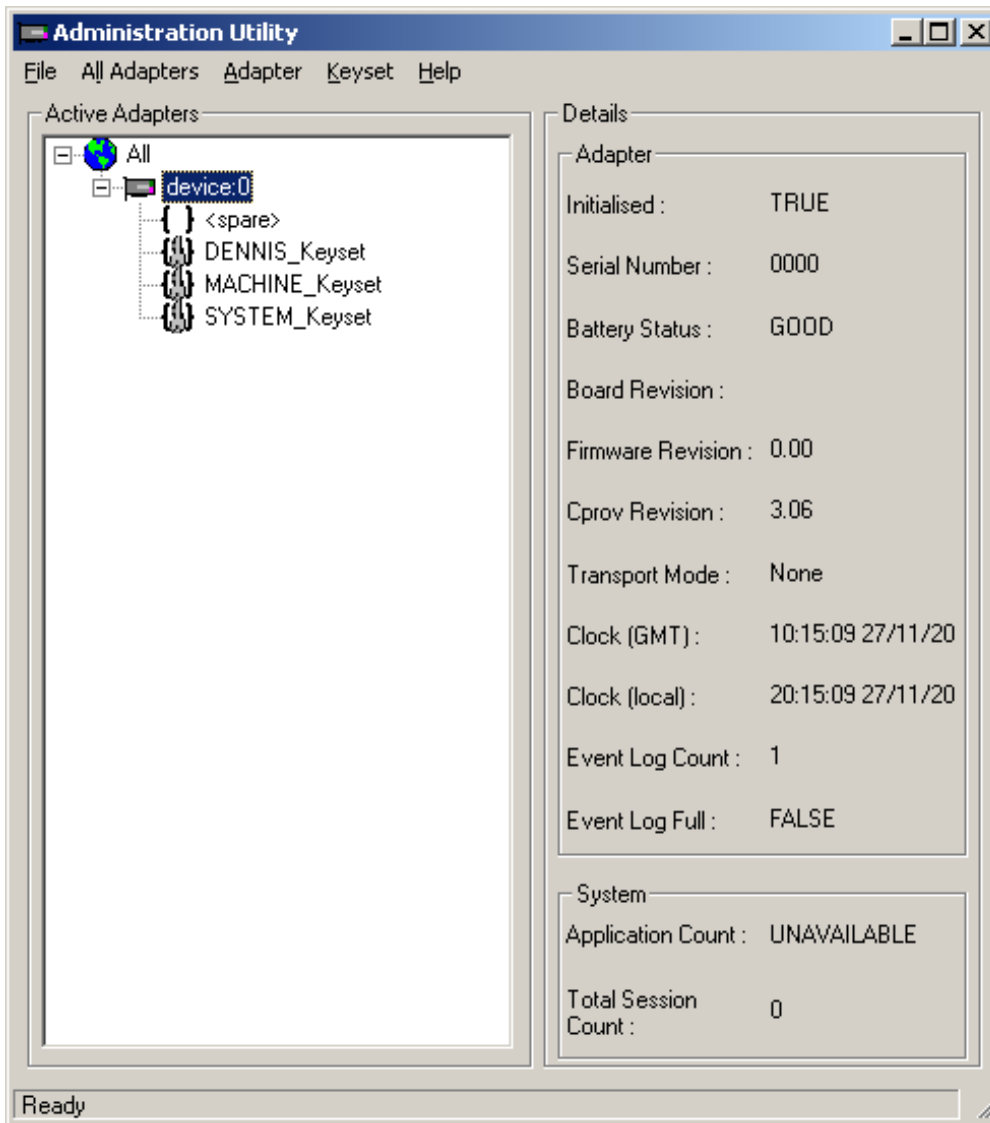
Starting and Exiting the Administration Utility

To start the administration utility

Select **Start > Programs > SafeNet > ProtectToolkit M > gmadmin**. After an initial splash screen, the main user interface is shown.

To exit the utility

Select **Exit** from the **File** menu.

User Interface

The administration utility is presented as a Graphical User Interface (GUI), which is divided into three main areas. These are:

> The **Menu Bar** - shown along the top of the utility. All available utility commands can be activated via these menus.

> The **Active Adapters** display pane - shows all hardware HSMs found on the host system and their associated keysets. These are represented as a hierarchical tree view, with HSMs being the highest member and keysets or keyset spaces shown beneath each HSM.

> The **Details** pane - broken up into two sub-groups and displays the following information.

Adapter Details	
Initialized	Shows whether the currently selected HSM has been initialized. Values are either TRUE or FALSE.
Serial Number	The serial number of the selected HSM.
Battery Status	Indicates the charge of the onboard battery of the selected HSM. This may either be GOOD or LOW. If this indicates LOW, the selected HSM may not be able to retain stored key information in the event of a system power failure. The HSM should be returned to your nearest Thales service centre for battery/HSM replacement. See "Support Contacts" on page 9.
Board Revision	Shows the hardware revision of the currently selected HSM.
Firmware Revision	Shows the firmware revision of the currently selected HSM.
Cprov Revision	Shows the revision of ProtectToolkit-C found on the currently selected HSM. This is a software component which forms part of the HSM firmware. This value may need to be quoted when contacting Thales support.
Transport Mode	Shows the transport mode which is set for the currently selected HSM. This value will be either "None", "Single", or "Continuous". Refer below for details.
Clock (GMT)	Shows date and time (GMT) on the currently selected HSM.
Clock (local)	Shows the local date and time on the currently selected HSM.
Event Log Count	Gives a total for the number of event log entries on the currently selected HSM.
Event Log Full	Shows if the event log is full and needs purging. This value may be either "FALSE", indicating that there is available space in the log, or "TRUE", indicating that the log is full.

System Details	
Application Count	Shows the number of applications which are currently accessing the ProtectToolkit-M system. This value may show as "UNAVAILABLE" which denotes that the firmware on the selected HSM does not support application counting.

System Details	
Total Session Count	Shows the number of open sessions to the ProtectToolkit-M system.

Password Entry Dialog Boxes

Most actions performed within the administration utility will require entry of the device administrator password. The device administrator password is case-sensitive and may consist of any alphanumeric characters, between 4 and 32 characters in length.

Figure 3: Device administrator password entry dialog box

Keep Password Feature

The utility can remember the device administrator password for the duration of the session. This eliminates the need to repeatedly enter the password for multiple operations.

To enable this feature, check the box next to **Keep Password For Session** when entering the device administrator password.

> **CAUTION!** When this feature is enabled, take care not to leave the administration utility unattended. To ensure that unauthorized people do not obtain management access to HSMs, close the administration utility once you have finished with your task.

Keyboard Shortcuts

All available menu items may be activated via keyboard shortcuts. The menu bar can be selected by pressing the [**Alt**] key. Commands may then be selected by pressing the first unique letter of the required command. For example: [**Alt**] followed by [**A**] will open the Adapter menu.

There are also a number of key combination shortcuts which will immediately activate a command:

> `CTRL+I` = Initialize HSM

> `CTRL+A` = Allocate Keyset Space

> `CTRL+V` = View Event Log

> `CTRL+P` = Change Admin Password

> `CTRL+U` = Upgrade Firmware

> `CTRL+T` = Tamper HSM

> **CTRL+D** = De-allocate Space

> **CTRL+K** = Create Keyset

Context Menus

Right-clicking on an item in the **Active Adapters** display pane will bring up a context menu showing available commands specific to that item.

For details about these commands, please refer to the section appropriate to the menu in question.

All Adapters Menu

The **All Adapters** menu is only available if there is more than one HSM installed on the system. This menu allows the device administrator to affect all installed HSMs with a single command. The following actions can be performed via this menu:

Initialize Adapters

This option initializes all uninitialized HSMs found on the system. Only initialized HSMs can store key information.

Synchronize Clocks

This option synchronizes all HSMs found on the system with the value of the host system clock.

Set Transport Modes

This option sets the adapter transport mode for all adapters found on the system. The adapter transport mode allows an HSM to be removed from the host system's PCIe bus without causing a tamper condition. A tamper will remove all sensitive material from the adapter, including the adapter configuration, keys, and certificates.

The device administrator is prompted to choose one of three possible transport modes:

> **None** - To be applied when adapter is installed and configured. This mode will tamper the adapter if removed from the PCIe bus.

> **Single** - Adapter will not be tampered after its next removal from the PCIe bus. Adapter will automatically change transport mode to **None** the next time the adapter is reset or power is removed and restored.

> **Continuous** - Adapter will not be tampered by being removed from the PCIe bus.

Set Security Flags

This option allows the setting of a security mode using security flags. These flags affect both the services available to the various users of the system, as well as specific security features of the HSM. The flags may be specified individually to set a custom security mode, but a standard security mode is recommended. When a standard security mode is selected, the flags are assigned values automatically to meet the requirements for that mode. For further information see "Initial Configuration: Mandatory Steps" on page 14, "Security Mode Descriptions" on page 15, and "Security Mode Flag Descriptions" on page 16.

Set Secure Configuration

This option allows secure configuration items to be set.

A secure configuration item is one that is open for reading but requires authentication for writing. Such configuration items are stored on the HSM, protected by the password of the device administrator.

A single item is currently supported - Allow Clear Export of Private Keys. See "Enabling Private Key Clear Export" on page 39 .

Set Admin Passwords

This option changes the current device administrator password for all HSMs on the host system.

Upgrade Firmware

This option performs a firmware upgrade for all HSMs on the host system. The device administrator is prompted to enter the path to the firmware update file.

Tamper All Adapters

This option causes a tamper of all HSMs found on the system. A tamper formats the secure memory of the HSM and thereby erases all configuration and key data.

Adapter Menu

The **Adapters** menu is used to perform the following administrative actions on a selected HSM:

Initialize

This option initializes the selected HSM. Only initialized HSMs can store key information.

Allocate Space

This option allocates one keyset space on the selected HSM. Keyset space is required to create user keysets.

View Event Log

This option opens the event log viewer. The dialog shows event log entries in chronological order, with the most current event showing last. The **first**, **prev**, **next** and **last** buttons can be used to navigate through the event details, should there be more than one page of entries.

If the event log is full, it can be purged by clicking the **Purge** button.

> **NOTE**   The **Purge** button is disabled until the event log is full.

Purge Event Log

This option purges the event log.

> **NOTE**   This menu option is disabled until the event log is full.

Synchronize Clock

This menu option synchronizes the clock of the selected HSM with the host system clock.

Set Transport Mode

This menu option is used to set the adapter transport mode for the selected adapter. The adapter transport mode allows an HSM to be removed from the host system's PCIe bus without causing a tamper condition. A tamper will remove all sensitive material from the adapter, including the adapter configuration, keys, and certificates.

The device administrator is prompted to choose one of three possible transport modes:

> **None** - To be applied when adapter is installed and configured. This mode will tamper the adapter if removed from the PCIe bus.

> **Single** - Adapter will not be tampered after its next removal from the PCIe bus. Adapter will automatically change transport mode to **None** the next time the adapter is reset or power is removed and restored.

> **Continuous** - Adapter will not be tampered by being removed from the PCIe bus.

Change Admin Password

This option changes the device administrator password for the currently selected HSM.

Upgrade Firmware

This option performs a firmware upgrade for the selected HSM. The device administrator is prompted to enter the path to the firmware update file.

Check Firmware Upgrade File

This option is used to check the validity of a firmware upgrade file. The device administrator is prompted to enter the path to the firmware update file.

Tamper

This option causes a tamper of the selected HSM. A tamper formats the secure memory of the HSM, erasing all configuration and key data.

Set Security Flags

This option allows the setting of a security mode using security flags. These flags affect both the services available to the various users of the system, as well as specific security features of the HSM. The flags may be specified individually to set a custom security mode, but a standard security mode is recommended. When a standard security mode is selected, the flags are assigned values automatically to meet the requirements for that mode. For further information see "Initial Configuration: Mandatory Steps" on page 14, "Security Mode Descriptions" on page 15, and "Security Mode Flag Descriptions" on page 16.

Set Secure configuration

This menu option allows the setting of security configuration items.

A secure configuration item is one that is open for reading but requires authentication for writing. Such configuration items are stored on the HSM, protected by the password of the device administrator.

A single item is currently supported - Allow Clear Export of Private Keys. See "Enabling Private Key Clear Export" on page 39.

Keyset Menu

The **Keyset** menu is used to perform the following administrative actions on a selected keyset:

Delete

This option will delete the currently selected keyset.

Create Keyset

This option creates a keyset within the currently selected keyset space.

Deallocate

This option removes the selected spare keyset space.

Keyset Management Utility

The keyset management utility (**gmksm**) is designed for the ProtectToolkit-M user, and allows the following operations:

> Create keysets

> Generate key pairs

> Delete key pairs

> Show key pair properties

> Add key containers

> Remove key containers

> Change passwords

This section is intended as a reference for **gmksm** and outlines the following topics:

> "Starting and Exiting the Keyset Management Utility" on the next page

> "User Interface" on the next page

> "Container Menu" on page 54

> "Key Pair Menu" on page 55

When performing administrative tasks, the reader is advised to refer to "Administrative Tasks" on page 32 for details regarding each task.

Figure 4: Keyset Management Utility User Interface
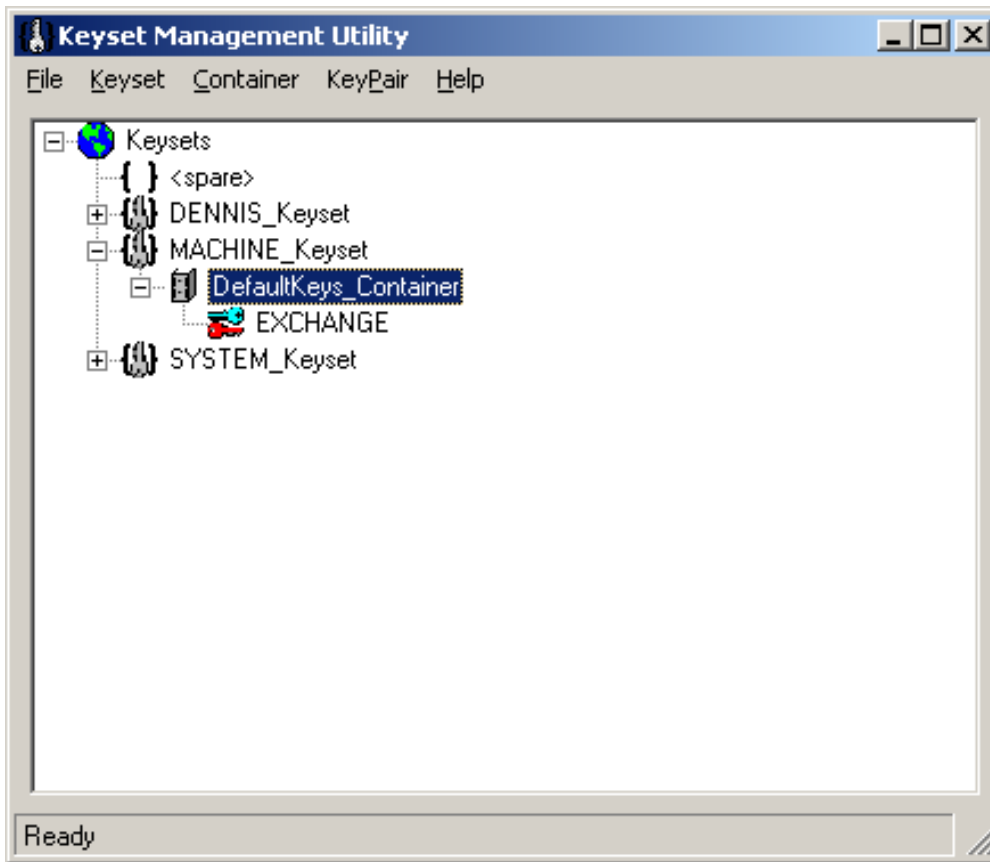
Starting and Exiting the Keyset Management Utility

To start the keyset management utility, select **Start > Programs > SafeNet > ProtectToolkit M > gmksm**. After an initial splash screen, the main user interface is shown (see"Keyset Management Utility User Interface" above).

To exit the utility, select **Exit** from the **File** menu.

User Interface

The keyset management utility is presented as a Graphic User Interface (GUI), divided into two main areas. These are:

> The **Menu Bar** - which is shown along the top of the utility. All available utility commands can be activated via these menus.

> The main display pane - shows all keysets, spare keyset spaces, key containers, and key pairs. These are represented as a hierarchical tree view, with keysets being the highest member. Key containers and key pairs are shown beneath each keyset.

Password Entry Dialogs

Most actions performed within the keyset management utility will require entry of the keyset user password. The keyset password is case-sensitive and may consist of any alphanumeric characters, between 4 and 32 characters in length.

Figure 5: Keyset password entry dialog

Keep Password Feature

The utility can remember the keyset password for the duration of the session. This eliminates the need to repeatedly enter the password for multiple operations.

To enable this feature, check the box next to **Keep Password For Session** when entering the keyset password.

> **CAUTION!** When this feature is enabled, take care not to leave the utility unattended. To ensure that unauthorized people do not obtain access to a user keyset, close the keyset management utility once you have finished with your task.

Keyboard Shortcuts

All available menu items may be activated via keyboard shortcuts. The menu bar can be selected by pressing the [**Alt**] key. Commands may then be selected by pressing the first unique letter of the required command. For example: [**Alt**] followed by [**K**] will open the Keyset menu.

There are also a number of key combination shortcuts which will immediately activate a command:

> CTRL+A = Add Container

> CTRL+R = Remove Container

> CTRL+K = Create Keyset

> CTRL+P = Change Password

> CTRL+G = Generate Key Pair

> CTRL+D = Delete Key Pair

Context Menus

Right-clicking on an item in the main display pane will bring up a context menu showing available commands specific to that item.

For details about these commands, please refer to the section appropriate to the menu in question.

Container Menu

The **Container** menu is used to perform the following user actions on a keyset container:

Remove

This option removes the selected key container.

> **CAUTION!** This action destroys any key pairs contained within the selected container.

Generate Key Pair

This option generates a key pair within the selected container. The user is prompted for:

> **Key Usage**: Select "Sign" or "Exchange".

> **Key Size**: Valid lengths are 512, 768, 1024, 2048 or 4096 bits.

> **Exportable**: Checking this box allows the generated private key to be backed up to a file on the host machine.

Key Pair Menu

The **Key Pair** menu is used to perform the following user actions to a selected key pair:

Delete

This option deletes the selected key pair

Properties

This option displays the properties of the selected key pair. The following attributes are shown:

> Keyset: Displays the name of the keyset where the selected key pair resides.

> Container: Displays the name of the key container where the selected key pair resides.

> Usage: Displays the key usage attribute of the selected key pair. This value will be "EXCHANGE" or "SIGN".

> Size: Displays the size of the selected key pair.

> Private Key Held: If "TRUE", the private key for the selected key pair is present as part of the key pair. It is only possible to import a public key .

> Exportable: Indicates whether the selected key pair can be backed up.

CREATECERT Utility

The **createcert** utility is used as a quick and simple method of creating a self-signed certificate for the ProtectToolkit-M machine.

> **NOTE** The user must be logged on as administrator to use this utility.

Synopsis

```
createcert <X509 Name>
```

Parameters

X509 Name The X.509 Certificate name. For example, **CN=<machinename>.**

CHAPTER 6:   Integration with Microsoft CA

This section contains the following instructive subsections for integration with Microsoft CA:

> "Setting Up a CA with ProtectToolkit-M" below
> "Certificate Template Support for SafeNet CSPs" on the next page
> "CA Replication (Key Backup and Recovery)" on page 58
> "Private Key Archiving and Recovery" on page 60

Setting Up a CA with ProtectToolkit-M

This section explains how to configure ProtectToolkit-M to be used with the Microsoft CA.

ProtectToolkit-M, in conjunction with Microsoft CA, provides secure storage of keys related to signing certificates.

Before you begin, ensure that:

> you have read and understood ProtectToolkit Software Installation in the *ProtectServer HSM and ProtectToolkit Installation Guide* and "Setup and Configuration" on page 13.
> Microsoft CA has NOT been installed prior to the ProtectToolkit-M installation.
> the current logged-on user has Windows administrator privileges.
> a keyset exists for the logged-on user.

An example of how to setup the CA with ProtectToolkit-M on Microsoft Windows 2008 R2 follows.

> **NOTE** This example assumes a standalone configuration for a root CA. Actual values should be chosen as required, to suit each particular installation.

To set up the CA with ProtectToolkit-M:

1. From the Windows Control Panel, select **Administrative Tools** and select **Server Manager** from the list of tools.

2. Click **Add Roles**.

3. Check the box for "Active Directory Certificate Services", click **Next**, and then **Next** again.

4. Check the box for "Certification Authority" and click **Next**.

5. Select **Standalone** and click **Next**.

6. Select **Root CA** and click **Next**.

7. Select the appropriate option (new or existing private key) and click **Next**.

8. Select the SafeNet CSP from the list, configure your cryptiographic options as required, and click **Next**.

9. Configure your CA name as required and click **Next**.

10. Set the validity period for the certificate generated for the CA as required and click **Next**.

11. Specify the locations for the certificate database and certificate database log and click **Next**.

12. Review the CA configuration. If any parameters are incorrect, use the links in the left pane to return to the appropriate page to make changes. When the configuration is correct, click **Install** to install the CA.

Following the successful completion of the above steps, ProtectToolkit-M is now selected as the CSP for Microsoft CA operations. For further details regarding the Microsoft CA, please refer to your Microsoft documentation.

Certificate Template Support for SafeNet CSPs

The current list of certificate templates in the CA do not make use of the SafeNet CSP. New templates must be created in the Certificate Templates store and then issued from the CA templates store. For example, a web server certificate template only supports the Microsoft DH and RSA providers.

In order to create new templates that support the SafeNet CSP, perform the following procedure. The procedure is basically the same for any certificate that you need to issue using the SafeNet CSPs.

To create a new template that supports the SafeNet CSP

Note that the use of the User template in this procedure is for example only. Substitute this for any other template to meet your particular requirements.

1. Start a new MMC session and add both the **Certification Authority** and **Certificate Template** snap-ins.

2. Expand the **Certificate Templates** object and locate the **User** template.

3. Right click on the **User** template and select **Duplicate template**. This will display the new template properties.

4. Enter a Template display name. Note that you cannot give it the same name as the template that already exists.

5. Go to the **Request Handling** tab and click on the **CSP** button. Either select **Requests can use any CSP available on the subject's computer** or make sure that the SafeNet RSA providers are checked.

6. Check the **Issuance Requirements** and **Security** tabs to ensure that the appropriate permissions are correct. Click **OK** to complete.

7. Now go to the CA object and select **Certificate Templates**.

8. Right click **New** and select **Certificate Template to Issue**.

9. Locate the new template that was created in steps 1-6 and click **OK**. Close the MMC console session.

10. To test that the SafeNet provider is now available, open a new MMC console and choose the **Certificates** snap-in. Select **My User Account** when prompted. The Administrator's personal certificate store is now available.

11. Right click on the personal object and select **All Tasks, Request new certificate**. The Certificate Request Wizard displays.

12. Click **Next** to reveal the certificate types available, select the new certificate and check the **Advanced** check box. Click **Next**.

13. On the CSP page that now displays, note that the SafeNet providers are now listed. Choose the RSA full provider and any other appropriate settings such as **Key is Exportable**, etc. Complete the process by clicking **OK**.

   The certificate is generated and visible in the personal store.

CA Replication (Key Backup and Recovery)

Typically, to replicate a CA installation, keys may be backed up to smart cards and then restored from the smart cards to establish the new CA installation. One smart card per keyset is required.

A triple-DES BackupKey is used to encrypt each keyset prior to storage on a smart card. A different BackupKey is automatically created for each keyset when the keysets are created but these keys are not visible under normal ProtectToolkit-M operation. A BackupKey for a keyset is derived from a combination of the password used to secure that particular keyset and the keyset name. In the case of the MACHINE and SYSTEM keysets, the device administrator's password and the keyset name are used to derive the key. Thus, to restore a keyset that was previously backed up, the same password and keyset name must be used.

Replicating a CA installation by restoring from a smart card requires access to the ProtectToolkit-C. The command line tool **ctkmu** and the GUI tool **kmu** can be used to backup or restore keys from a smart card. Both the **ctkmu** utility and the **kmu** utility are included in the ProtectToolkit-C package. See CTKMU in the "Command Line Utilities Reference" section of the *ProtectToolkit-C Administration Guide* and Key Management Utility (KMU) Reference in the *ProtectToolkit-J Reference Guide* for more information about these utilities.

> **NOTE** When logging in to a smart card, the card is locked after 7 consecutive incorrect PIN attempts. You must re-initialize the card to set a new PIN.

To Back Up Keys for a CA Installation to Smart Cards

1. Obtain a listing of all keysets by executing **ctkmu l** from a command prompt. A list of all keysets and associated slots displays.

   > **NOTE** Decide which keysets to back up. At a minimum, the MACHINE_Keyset must be backed up, as this is where the CA keys are stored.

2. Record the slot number for each keyset that you wish to back up.

3. To back up the MACHINE_Keyset to smart card, type the following in a command prompt, where *n* is the slot number of the MACHINE_keyset and *b* is the slot number representing the smart card reader. Both *n* and *b* can be found in the listing obtained at step 1.

   ```
   ctkmu x -sn -wBackupKey -c
   ```

4. When prompted for a user password, enter the default "**password**".

5. Insert a new smart card and repeat steps 3 and 4 for the SYSTEM_Keyset if required.

6. Repeat steps 3 and 4 for each of the other required keysets.

Replicating a CA Using Keys Restored from Backup Smart Cards

The following procedure takes the following key points into account:

> On the machine where the replica is to be created, ProtectToolkit-M must be installed before the Microsoft CA.

> To allow installation of a CA that utilizes the SafeNet CSP for HSM storage of keysets, both the MACHINE_ Keyset (where the CA stores keys) and a user keyset for the current user must be available. At CA installation time if either or both of these keysets are missing, the SafeNet CSP will not display in the list of CSPs available for selection.

> All keyset names and associated passwords created when establishing the replica must match the originals that are to be restored from the backup smart cards.

To Replicate a CA Using Keys Restored from Backup Smart Cards

1. Start the ProtectToolkit-M administration utility. This can be done via the Windows **Start** menu. Select **Start > Programs > SafeNet > Protect Toolkit M > gmadmin**.

   A MACHINE_Keyset and a SYSTEM_Keyset will be created. Later on, the MACHINE_Keyset created here will be replaced with the version that was backed up to smart card, containing the CA keys.

   The device administrator password will be requested, or must be set if this is the first time the HSM has been accessed.

   The Administration Utility default view displays.

2. Under **Active Adapters**, expand **All** to reveal the device and the Machine and System key sets just created on that device.

3. Highlight the device entry and select **Adapter** on the menu bar. Now select **Allocate Space** to create a keyset space.

4. Under **Active Adapters**, select the spare keyset space.

5. Select **Keyset** on the menu bar and then choose **Create Keyset**. The Administration Utility will now prompt for a keyset name to use and the password for the currently logged-on user. The default name should be accepted.

6. If additional user keysets containing keys are to be restored from smart card, create an empty replica keyset on the HSM for each keyset to be restored with the same name and user password as the original. To do this, repeat steps 4 to 6 for each keyset, using the appropriate keyset name and user password each time.

7. Obtain a listing by name of all the keysets that now exist on the HSM and their corresponding slot numbers by executing **ctkmu l** from a command prompt.

8. Import a keyset from smart card to the HSM. To do this, insert the smart card containing the keyset. ssh into the device and execute the following command from a command prompt:

   ```
   ctkmu i -sn -wBackupKey -cm
   ```

   where *n* is the slot number of the keyset on the HSM discovered in step 8, and *m* is the smart card reader slot number. This will also be shown in the listing obtained at step 8.

9. When prompted for a user password, enter the value for the keyset being restored. In the case of the machine and system keysets, the default is "**password**".

10. Insert a new smart card and repeat step 4-10 for each additional keyset until all have been restored.

11. Install the Microsoft CA.

12. Select the SafeNet CSP from the drop-down box during installation. If the SafeNet option is not present, this means that the keyset for the currently logged in user does not exist. Ensure the user is the same as the user who did the backing up of the CA initially.

13. After selection of the SafeNet CSP, click the **Use existing keys** box and select the key that corresponds to the CA key pair.

Private Key Archiving and Recovery

When requesting a certificate using the Windows CA, users have the option to have their private key archived by the CA. In a catastrophic system failure that results in the user losing their entire system, this feature allows recovery of the user's private key.

Support for this archival and recovery process is included in ProtectToolkit-M. The following examples demonstrate the use of this capability.

Private Key Archiving Example

Here are the tasks required to archive a private key using a Microsoft certification authority (CA).

> Create a key recovery agent account

> Acquire the key recovery agent certificate

> Configure the certification authority to allow key recovery

> Create a new certificate template that allows key archiving

> Acquire a user certificate that has an archived key

Prerequisites

Before doing these tasks:

> You must have a Windows Server domain controller.

> The Windows Server domain controller must also be configured as an enterprise root or subordinate CA.

> A user keyset for the user must exist. Refer to "Creating User Keysets" on page 19 for further information.

> The Allow Clear Export of Private Keys flag must be set. See the section "Enabling Private Key Clear Export" on page 39 for the procedure.

Task 1—Creating a Key Recovery Agent Account

Configure and add the Key Recovery Agent certificate template as a template that can be issued by the enterprise CA.

 To verify who can enroll the Key Recovery Agent template

1. Log on as administrator.

2. Click **Start**, **Run**, type **certtmpl.msc**, and press **Enter**.

   This opens the Certificate Templates snap-in in the Microsoft Management Console.

3. In the console tree, click "**Certificate Templates**."

4. In the details pane, right-click **Key Recovery Agent** and click **Properties**.Select the **Security** tab.

By default, the security groups that can enroll the Key Recovery Agent certificate template are Domain Administrators and Enterprise Administrators.

5. To allow other users or groups to enroll the Key Recovery Agent certificate template, click **Add** to add the user or group and grant them Read and Enroll permissions.

To change the default issuance behavior of the Key Recovery Agent template

1. In Key Recovery Agent Properties, click the **Issuance Requirements** tab.

2. Clear the **CA certificate manager approval** check box and click **OK.**

3. Close the Microsoft Management Console.

To change the request handling to allow the Safenet CSPs

1. In the Key Recovery Agent Properties, click the Request Handling tab.

2. Check the **Allow Private Key to be Exported** check box.

3. Click on the **CSP** button and click on the radio button to allow requests to use any CSP available on the subject's computer.

To configure the Certification Authority (CA) to issue Key Recovery Agent certificates

1. On the **Administrative Tools** menu, click **Certification Authority**.

    This opens the **Certification Authority** snap-in in the Microsoft Management Console.

2. In the console tree, double-click the CA, and then click **Certificate Templates**.

3. Right-click **Certificate Templates**, then click **New CertificateTemplate to Issue**.

4. In **Enable Certificate Template**, click **Key Recovery Agent**, and then click **OK**.

Task 2—Acquiring the Key Recovery Agent Certificate
In this series of steps, you will acquire a Key Recovery Agent Certificate for the purpose of recovering private keys. Begin by creating an MMC console with the Certificates snap-in loaded.

To ensure that you are logged on as the administrator

1. On the taskbar, click the **Start** button, and then click **Run**.

2. In Run, type **mmc**, and then click **OK**.

3. On the **File** menu, click **Add/Remove Snap-in**.

4. In **Add/Remove** Snap-in, click **Add**.

5. In **Add Stand-alone** Snap-in, click **Certificates**, and then click **Add**.

6. In **Certificates**, click **My User account** and then click **Finish**.

7. Click **Close**, and then click **OK**.

To acquire a Key Recovery Agent certificate

1. In the console tree of the newly-created MMC console, double-click **Certificates - Current User**.

2. In the console tree, right-click **Personal**, click **All Tasks**, **Request New Certificate**.

3. In the Certificate Request Wizard, click **Next**.

4. In Certificate Types, select **Key Recovery Agent** and the **Advanced** checkbox, and then click **Next**.

5. On the CSP page that now displays, choose the SafeNet provider for HSM key storage and any other appropriate settings such as **Key is Exportable**, etc. Then click **Next** and **Next** again.

6. On the Certificate Friendly Name and Description page, in the **Friendly Name** field, type **Key Recovery**, and then click **Next**.

7. In Completing the Certificate Request Wizard, click **Finish**.

8. In the console tree, double-click **Personal** and then click the **Certificates** folder.

9. Ensure that a certificate with the friendly name of **Key Recovery** exists.

10. Close the console without saving changes.

Task 3—Configuring the CA to allow key recovery
In this series of steps, configure the enterprise CA to use the Recovery Agent certificate acquired in Task 2. The CA must load the public key for the Key Recovery Agent to be used for encrypting the recovery data.

To configure the Recovery Agent to be the Administrator's Key Recovery Agent certificate

1. Ensure that you are logged on as the administrator.

2. In **Administrative Tools**, open **Certification Authority**.

   This opens the Certification Authority snap-in in the Microsoft Management Console (MMC).

3. In the console tree, click **the CA**.

4. Right-click the **CA**, and then click **Properties**.

5. In the **CA Properties**, on the **Recovery Agents** tab, click **Archive the key** and then click **Add**.

6. In **Key Recovery Agent Selection**, click the certificate that is displayed, and then click **OK**. The key recovery agent certificate is shown with a status of Not loaded.

7. Click **OK**, and when prompted to restart the CA, click **Yes**.

To open the Certificates console, focused on the local computer

1. On the taskbar, click the **Start** button, and then click **Run**.

2. In Run, type **mmc**, and then click **OK**.

3. On the File menu, click **Add/Remove Snap-in**.

4. In **Add/Remove Snap-in**, click **Add**.

5. In **Add Standalone Snap-in**, click **Certificates**, and then click **Add**.

6. In **Certificates Snap-in**, click **Computer account** and then click **Next**.

7. In **Select Computer**, click **Local Computer**, and then click **Finish**.

8. Click **Close**, and then click **OK**.

To verify the installation of the Key Recovery Agent (KRA) certificate

1. In the console tree, double-click **Certificates (Local Computer)**, double-click **KRA**, and then click **Certificates**.

2. In the details pane, double-click the certificate.

3. Verify that the intended use of the certificate is Key Recovery Agent and the certificate is issued to Administrator. This procedure ensures that the Key Recovery Agent has been successfully configured.

4. Click **OK** and then close the console without saving changes.

Task 4 — Creating a new certificate template that allows key archiving

In this series of steps, you define a new template that allows Key Archival and HSM key storage by using the Certificate Templates console. This will allow hardware key storage within a HSM at the client computer and key recovery in the domain in the event that the private key is lost or corrupted at the client computer.

To open the Certificate Templates console

1. Log on as the administrator.

2. On the taskbar, click the **Start** button, and then click **Run**.

3. In Run, type **mmc**, and then click **OK**.

4. On the **File** menu, click **Add/Remove Snap-in**.

5. In **Add/Remove Snap-in**, click **Add**.

6. In **Add Standalone Snap-in**, click **Certificate Templates**, and then click **Add**.

7. Click **Close**, and then **OK**.

A duplicate of the Users certificate template is now created and named Archive User. This is a shortcut to creating a template with permissions that allows both Domain Administrator and Domain User certificate enrollments. The template is then modified so that certificate enrollments made using this template will enable both key archival and the ability to use Safenet as a CSP.

To create a modified Archive User certificate template

1. In the console tree, click **Certificate Templates**.

2. In the details pane, right-click the **User** template, and click **Duplicate Template**.

3. In the **Properties of New Template** dialog box, in the **General** tab, in the **Template** display name box, type **Archive User**.

4. In the **Request Handling** tab, enable the **Archive subject's encryption private key** option (see the screen shot below). This option makes it possible for a Key Recovery Agent to recover the private key from the certificate store.
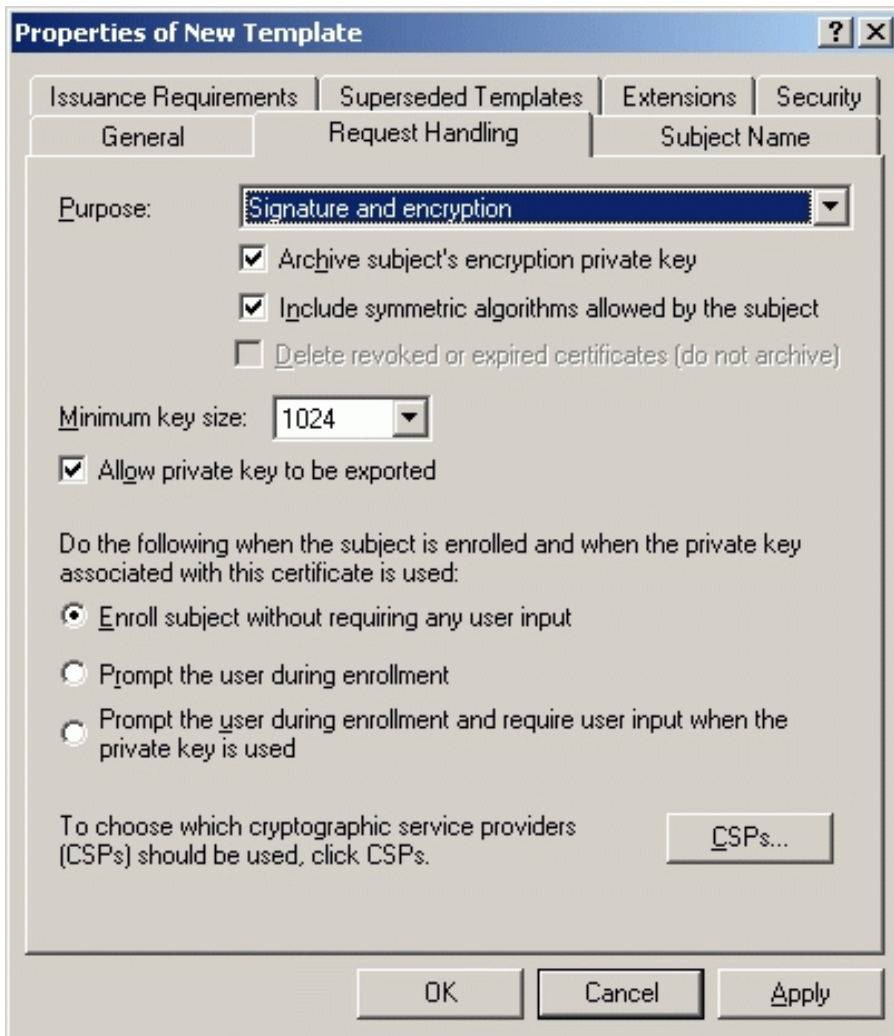
5. Click the CSPs button to enable HSM key storage using one or more SafeNet CSPs.

   The dialog box allows selection of particular CSPs or all CSPs may be enabled by selecting the appropriate radio button.

   Typically, only the **Safenet RSA Full Cryptographic Provider** is required. The **SChannel Provider** is only needed where SSL processing will be carried out.

6. After finalizing selections, click **OK** and **OK** again to apply changes and close the dialog boxes.

7. Close the console without saving changes.

Task 5—Acquiring a User certificate that has an archived key

In this series of tasks, you will configure the certification authority (CA) to issue Archive User certificates. Using a newly created account, you will act as a user to acquire an Archive User certificate from the CA and record the certificate's serial number for later use.

To configure CA to issue the new Archive User certificate template

1. Ensure that you are logged on as the administrator.

2. From **Administrative Tools**, open **Certification Authority**.

3. In the console tree, double-click the CA name, and then click **Certificates Templates**.

4. Right-click **Certificate Templates**, click **New**, and then click **Certificate Template to Issue**.

5. In **Enable Certificate Templates**, click **Archive User** and then click **OK**.

6. The **Archive User** certificate template now appears in the details pane.

7. Close **Certification Authority**.

To create a new user account

1. In **Administrative Tools**, open **Active Directory Users and Computers**.

2. Double-click the domain.

3. Select **Users** then click the **Create a new user in the current container** button.

4. Complete the following fields to create a user account:

    a. **First name**

    b. **Last name**

    c. **User logon name** (e.g. JSmith@xxxx.com)

    d. **Password**

5. Click **Next**, and then click **Finish**.

6. Double-click the new user account, select the **General** tab and enter the email address. e.g. JSmith@xxxx.com. This is required if the option to include the email name is set in the template used to create the user (Subject Name tab).

7. For the purpose of demonstration here, add the user to the Server Operators group so they are able to log on locally to the domain controller. This would not normally be required.

    a. Select the **Member of** tab.

    b. Click **Add**, in **Select Groups**, type **Server Operators**, click **Check Names**, and then click **OK**.

    c. Click **OK** to close Properties.

8. Close Active Directory Users and Computers.

9. Close all open windows and log off the computer.

To open the Certificates console

1. Log on as the user.

2. On the taskbar, click the **Start** button, and then click **Run**.

3. In Run, type **mmc**, and then click **OK**.

4. From the **File** menu, click **Add/Remove Snap-in**.

5. In **Add/Remove Snap-in**, click **Add**.

6. In **Add Stand-alone Snap-in**, click **Certificates**, click **Add**, and then click **Close**.

7. Click **OK** to close the **Add/Remove Snap-in** dialog box.

To use the Certificates MMC to acquire an Archive User certificate

1. In the newly-created MMC console, in the console tree, double-click **Certificates (Current User)**.

2. In the console tree, right-click **Personal**, click **All Tasks**, and then click **Request New Certificate**.

3. In the Certificate Request Wizard, click **Next**.

4. Under **Certificate types**, select **Archive User** and check the **Advanced** checkbox. Then click **Next**.

5. On the CSP page that is now visible, choose the SafeNet provider for HSM key storage and any other appropriate settings such as **Key is Exportable**, etc. Then click **Next** and **Next** again.

6. In **Friendly name**, type **Archive User**, and then click **Next**.

7. On Completing the Certificate Request Wizard, click **Finish**.

    If the dialog box shown below displays, the most likely cause of the problem is that the Allow Clear Export of Private Keys flag has not been set. See "Enabling Private Key Clear Export" on page 39 for details.
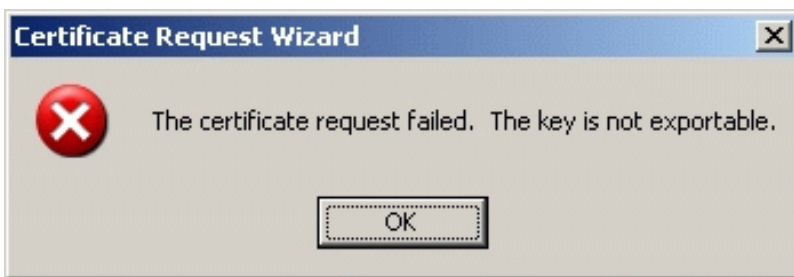
8. Double-click **Personal**, and then click **Certificates**.

9. In the details pane, double-click the certificate with the friendly name of **Archive User**.

10. In **Certificate**, click the **Details** tab.

    Note that the certificate template used to generate this certificate was Archive User, then click **OK**.

11. Close the new console without saving changes.

12. Close all windows and log off of the computer.

Private Key Recovery Example

Here are the tasks required to recover a lost private key previously archived using a Microsoft certification authority (CA).

> Perform key recovery

> Import the recovered private key

Task 1—Performing a Key Recovery

In this series of tasks, perform a key recovery by using **Certutil.exe**. For more information on Certutil, see your Microsoft documentation.

First, ensure that the private key is recoverable by viewing the Archived Key column in the Certification Authority console and obtain the certificate serial number required for recovery.

To obtain the certificate serial number of the confirmed recoverable private key

1. Log on as the administrator.

2. From **Administrative Tools**, open **Certification Authority**.

3. In the console tree, double-click the CA, and then click **Issued Certificates**.

4. From the **View** menu, click **Add/Remove Columns**.

5. In **Add/Remove Columns**, in **Available Column**, select **Archived Key**, and then click **Add**. Archived Key should now appear in Displayed Columns.

6. Click **OK** and then, in the details pane, scroll to the right and confirm that the last issued certificate to the user has a **Yes** value in the Archived Key column.

   > **NOTE** A certificate template must have been modified so that the Archive bit and Mark Private Key as Exportable attributes were enabled. The private key is only recoverable if there is data in the Archived Key column.

7. Double-click the Archive User certificate.

8. Click the **Details** tab

9. Write down the serial number of the certificate. (Do not include spacing between digit pairs.) This is required for recovery.

   The serial number is a 20 character, hexadecimal string. The serial number of the private key is the same as the serial number of the certificate.

   For the purposes of this walkthrough, the serial number is referred to as *serialnumber*.

10. Click **OK.**

11. Close **Certification Authority**.

To recover the private key into a BLOB output file using certutil.exe

1. From a command prompt, type **cd ** and then press **Enter**.

2. Ensure that you are in the **c:** directory.

3. At the command prompt, type:

   ```
   Certutil -getkey serialnumber outputblob
   ```

4. At the command prompt, type `dir outputblob`

   > **NOTE** If the file outputblob does not exist, you probably typed the serial number incorrectly for the certificate. The outputblob file is a PKCS#7 file containing the KRA certificates and the user certificate and chain. The inner content is an encrypted PKCS#7 containing the private key (encrypted to the KRA certificates).

To recover the original private/public key pair using certutil.exe

1. From a command prompt, type:

   ```
   Certutil -recoverkey outputblob <username>.pfx
   ```

2. When prompted, enter the following information:

    a. Enter new password: **password**

    b. Confirm new password: **password**

    c. Type **exit**, and then press **Enter**.

3. Close all windows and log off as the current user.

Task 2—Importing the recovered private key
Restoration of the recovered private key to the users certificate store by importing the **<username>.pfx** file.

To log on as the user and start the Certificates mmc

1. Log on as the user.

2. On the taskbar, click the **Start** button, and then click **Run**.

3. In Run, type **mmc**, and then click **OK**.

4. On the **File** menu, click **Add/Remove Snap-in**.

5. In **Add/Remove Snap-in**, click **Add**.

6. In **Add Standalone Snap-in**, click **Certificates**, click **Add**, and then click **Close** and **OK**.

To delete all certificates issued by the CA to simulate a re-installed computer

1. Right-click **Certificates - Current User**, and then click **Find Certificates**.

2. In **Find Certificates**, in **Contains**, type the CA and then click **Find Now**.

3. On the **Edit** menu, click **Select All**.

4. On the **File** menu, click **Delete**.

5. In **Certificates**, click **Yes**.

6. In **Root Certificate Store**, click **Yes**.

7. Close **Find Certificates**.

To import the certificate at c:\ <username>.pfx and let the certificates be placed automatically

1. In the console tree, right-click **Personal** and then click **All Tasks** and then click **Import**.

2. In the Certificate Import Wizard, click **Next**.

3. On **Files to Import**, in the **File** name box, type **c:\ <username>.pfx**, and then click **Next**.

4. In **Password**, type **password** and then click **Next**.

5. On **Certificate Store**, click **Automatically select the certificate store based on the type of certificate** and then click **Next**.

6. On Completing the Certificate Import Wizard, click **Finish**.

7. If the **Root Certificate Store** dialog box appears, click **Yes**.

8. In Certificate Wizard Import, click **OK**.

Two certificates were imported. The Archive User certificate for the user is located in the Personal certificates store and the CA certificate is located in the Trusted Root Certification Authorities store.

To verify the serial number of the imported certificate

1. In the console tree, double-click **Personal** and then click **Certificates**.

2. Double-click the certificate.

3. In Certificate, click the **Details** tab. Verify that the serial number matches the original.

4. Close all open windows and log off.

CHAPTER 7:   Integration With IIS

One of the uses for the Microsoft Cryptographic API is for Secure Socket Layer (SSL) processing. This section explains the steps necessary to configure and use ProtectToolkit-M in conjunction with IIS.

Prior to performing any of the following, please ensure that ProtectToolkit-M is correctly installed and configured. For details, please refer to ProtectToolkit Software Installation in the *ProtectServer HSM and ProtectToolkit Installation Guide* and "Setup and Configuration" on page 13.

This section contains the following instructive subsections for integration with IIS:

> "Creating a Certificate" below

> "Installing a Certificate for use with IIS" on page 72

Creating a Certificate

In order for ProtectToolkit-M and the HSM to be used for SSL processing, a certificate needs to be set up that specifies the details of the ProtectToolkit-M machine.

There are multiple methods of creating a certificate for the machine:

> "Using IIS" below

> "Creating a Certificate Using the Microsoft CA server" on the next page

> "Using the createcert utility" on page 72. Note that self-signed certificates created by the utility are only of use for testing purposes.

Using IIS

When using IIS to install a certificate on the host machine, the following has to be performed:

> Creating a certificate request

> Sending the certificate request to be signed by a CA

> Installing the signed certificate into IIS

To create a certificate request using IIS

1. Start the **Internet Services Manager** from the **Administrative Tools** menu.

2. Highlight the **Default Web Site** entry, and right-click to open a context menu. Select **Properties**. The default web site properties dialog opens. Select the **Directory Security** tab.

3. Click on the **Server Certificate** button. This will start the IIS Certificate Wizard.

4. Choose **Create a new certificate** from the available options and press **Next** to continue.

5. Choose **Prepare the request now, but send it later** from the available options and press **Next** to continue.

6. Select SafeNet as the security provider. On the **Name and Security Settings** page that now displays, check the **Select cryptographic service provider (CSP) for this certificate** checkbox. Click **Next** to continue.

7. Continue to follow the on-screen prompts until the certificate request is completed.

    The IIS Certificate Wizard creates the certificate request as a file. You should now forward this file onto your CA in order to have it signed. The CA returns a new file, which is the signed certificate.

    Refer to "Installing a Certificate for use with IIS" on the next page for details on how to install the signed certificate.

Creating a Certificate Using the Microsoft CA server

The Microsoft CA server provides a standard internet browser interface for the creation of certificates.

> **NOTE** Before starting the following procedure, ensure that the current logged on user has Windows administrator privileges and has a valid keyset.

To create a certificate using MS CA server

1. Start the MS CA services interface by opening your web browser and specifying the Microsoft CA server URL. For example:

    `http://hostname/certsrv`

    The opening dialog for CA services appears.

2. Select the **Request a certificate** option and press **Next** to continue. You are prompted to select the request type.

3. Choose, **Advanced request** and press **Next** to continue. You will be presented with the Advanced Certificate Requests screen.

4. Select **Submit a certificate request to this CA using a form**, and press **Next** to continue. You will be presented with a form to input the certificate details.

5. Enter the details for the certificate into the fields provided:

    a. **Certificate Name**: enter the host machine's name. This can be found by executing the standard Windows command **hostname** from a command prompt.

    b. **Intended Purpose**: choose **Server Authentication Certificate**.

    c. **Key Options**: choose **SafeNet RSA SChannel Cryptographic Provider** as the CSP

    d. **Key Usage**: choose **Exchange**

    e. **Key Size**: enter as required, eg. "1024"

    f. select **Create new key set**

    g. if you want to be able to back up the keys associated with the certificate at a later date, choose **Mark keys as exportable**

    h. choose **Use local machine store**

    i. **Additional Options**: choose **Hash Algorithm**, e.g. "MD5"

> **NOTE** If the current logged-on user's keyset does not exist when the Safenet CSP is selected, the **Hash Algorithm** list box at the bottom of the screen will be empty. Should this be the case, abort this operation and create a keyset for the currently logged-on user before attempting this task again.

6. Press the **Submit** button when you have confirmed your inputs. If the Microsoft CA was configured to "Auto Issue" certificates, you are presented with the Certificate Issued dialog.

   Click **Install this certificate** to complete the certificate request and installation.

   If CA Services is not configured to auto-issue certificates, the dialog will state that your certificate request is pending. You will have to check on the status of the certificate using the CA services at a later time. When the certificate is ready, you are presented with the **Certificate Issued** dialog.

Using the createcert utility

The **createcert** utility is provided as a simple means to create a self-signed certificate for the ProtectToolkit-M host machine.

> **NOTE** These certificates are intended for development and testing purposes only. Also ensure that the current logged on user has Windows administrator privileges and has a valid keyset.

You will need to know the machine name for the ProtectToolkit-M system. Run the standard Windows command **hostname** from a command prompt.

To create a self-signed certificate using createcert utility

From a command prompt, execute the utility **createcert**, specifying the machine name. For example, if the machine name is "betaone", the command would be as follows:

```
C:\>createcert CN=betaone
```
Successful execution of the above will result in an RSA key pair being created, as well as a certificate which is saved in the file "selfsigned.cer". This certificate is then automatically installed ready for IIS use.
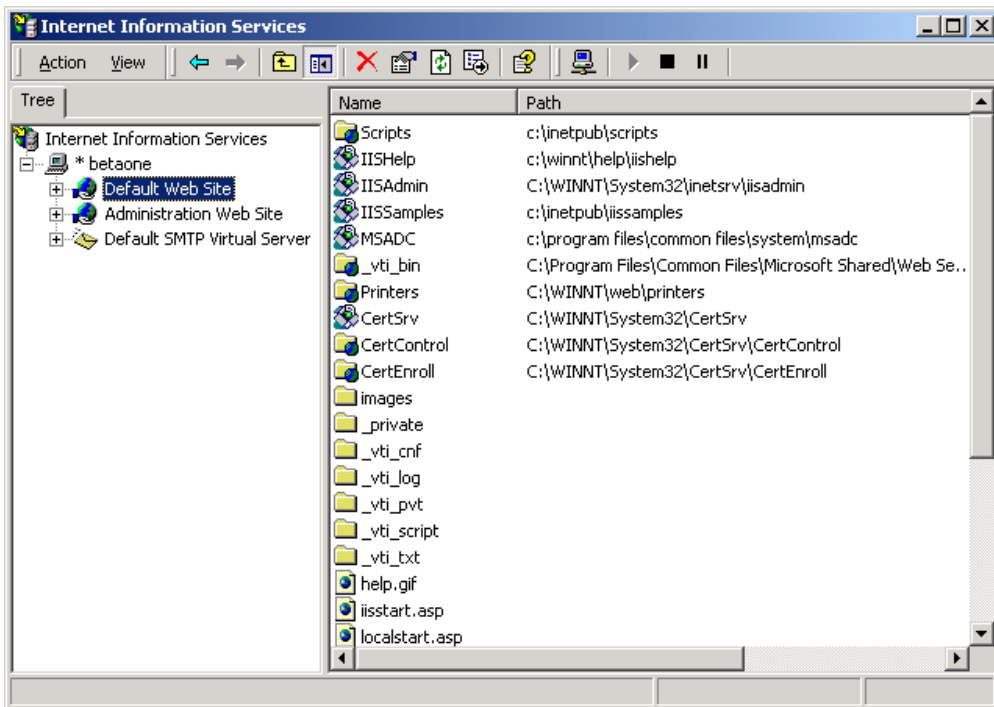
```
File calles SelfSigned.cer has been saved.
Certificate created successfully and installed
```

Installing a Certificate for use with IIS

In order to make use of the certificate in IIS, it will need to be assigned to a website.

To install the certificate with IIS

1. Start the Internet Services Manager from the Windows **Start/Programs/Administrative Tools/** menu.

2. Highlight the "Default Web Site" entry, and right-click to open a context menu. Select Properties. The default web site properties dialog opens. Select the Directory Security Tab.
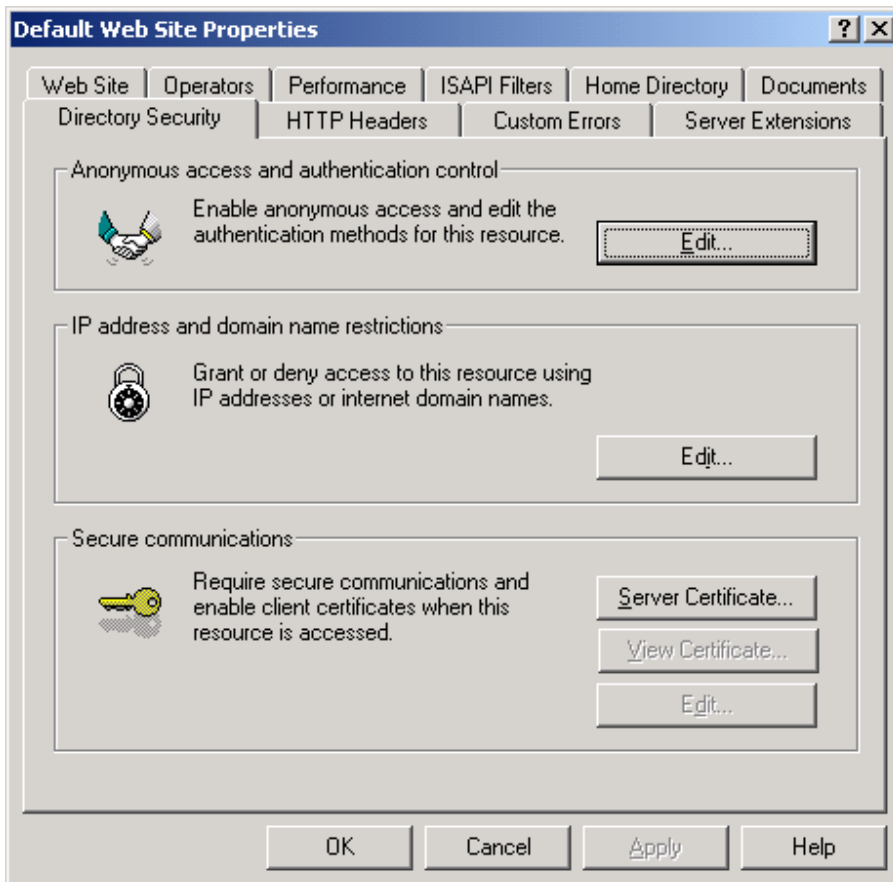
3. Click on the Server Certificate button. This will start the IIS Certificate Wizard.

4. Depending on how the certificate was generated, the IIS Certificate Wizard will prompt for the following:

If there is an outstanding certificate request

The IIS Certificate Wizard will inform the user that there is a pending certificate request.

1. When prompted, choose "Process the pending request and install the certificate".

2. Continue to follow the on-screen prompts until the certificate is installed.

If there is no outstanding certificate request

The IIS Certificate Wizard will prompt the user to assign a certificate using one of three possible methods.

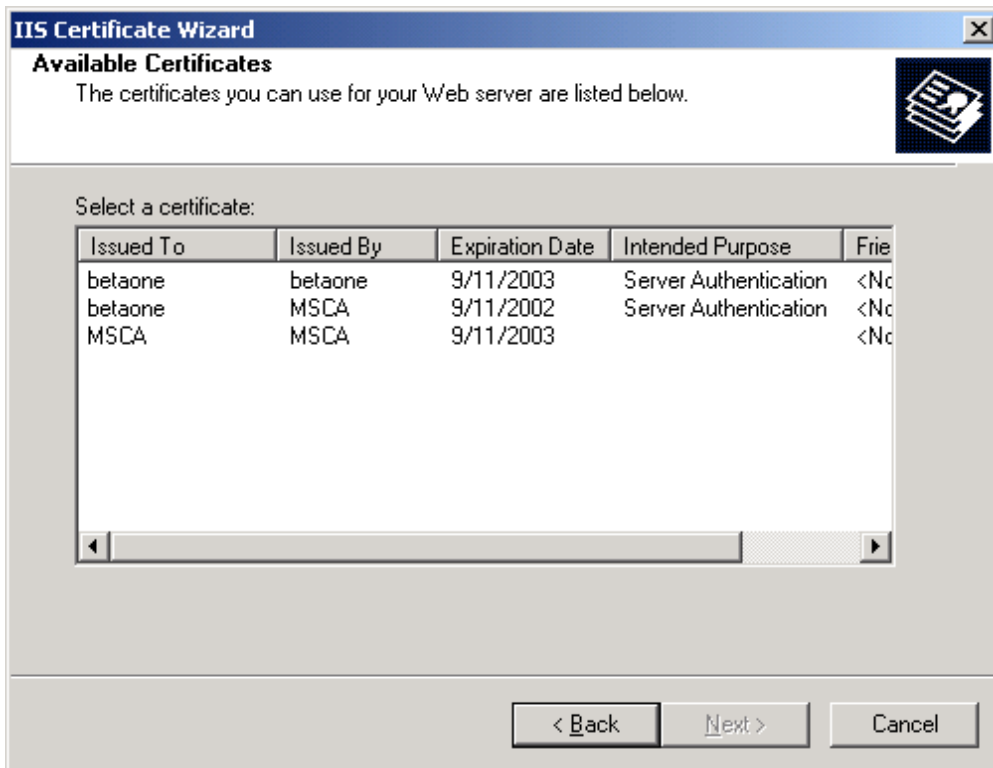**1.** Choose "Assign an existing certificate" from the available options and press **Next** to continue.
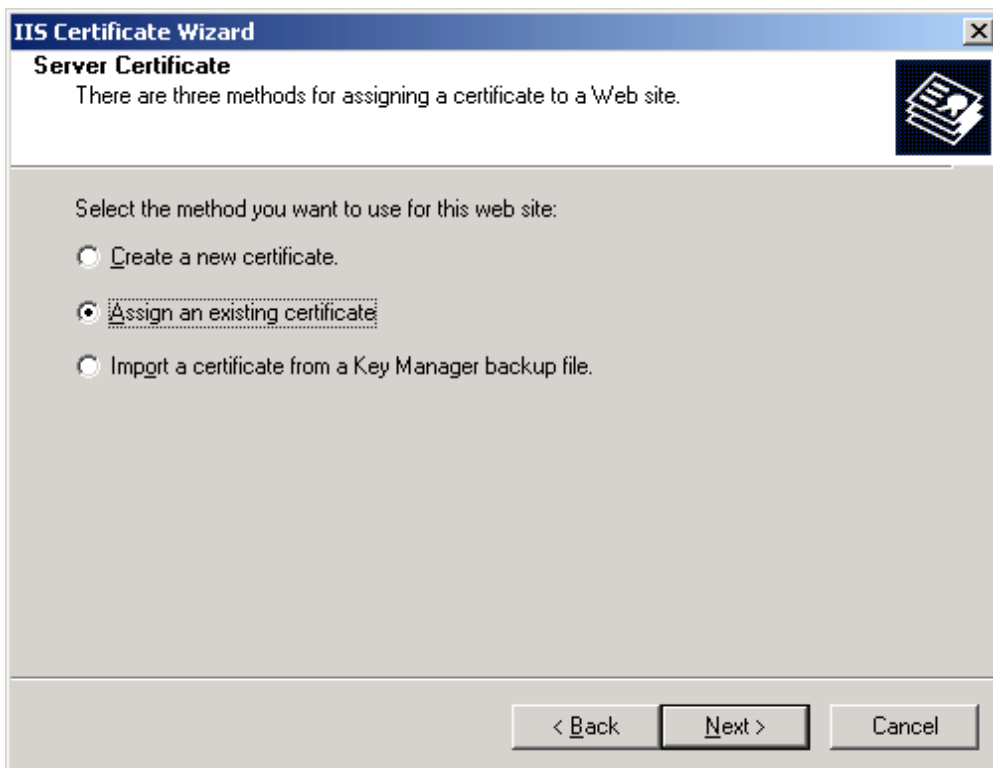
The example below shows that three certificates are currently installed.

The first listing is a self-signed certificate, created using the **createcert** utility. Note that the "Issued To" and "Issued By" fields are the same. If you decide to install this type of certificate, you will receive a "Security Alert" when trying to access the web site using Microsoft Internet Explorer. When using self-signed certificates, you will not be able to acquire a secure connection using Netscape 6.0.

The second listing in the example is a certificate generated using Microsoft CA. Note that in this case the "Issued To" and "Issued By" fields reflect that this is not a self-signed certificate.

The last listing in the example dialog is the Microsoft CA certificate used to sign certificate requests. This certificate appears in the list because the CA was set up on the same machine as the one being configured.

2. Select the certificate you wish to install and press **Next** to proceed with the certificate installation.

This concludes the certificate installation for IIS, and SSL connections can now be made to the default web site.

CHAPTER 8: PKCS #11 Attributes

Objects, as described by PKCS #11, consist of a number of attributes that define both the *object* and its *access policy*. In general, the ProtectToolkit-C system will define the object's attributes. Access policy should be provided by the user based on their particular requirements. The following attribute descriptions are intended to assist with these decisions.

Attribute	Description
CKA_LABEL	This attribute specifies a textual label for an object. This label is used to assist in differentiating the various objects stored on a token. **NOTE** Although ProtectToolkit-C does not require this attribute to be unique, various other tools may.
CKA_CLASS	This attribute is assigned by the system when an object is created. There are a number of classes in common use: > CKO_PUBLIC_KEY > CKO_PRIVATE_KEY > CKO_SECRET_KEY > CKO_CERTIFICATE > CKO_CERTIFICATE_REQUEST > CKO_DATA
CKA_KEY_TYPE	This attribute specifies the key type associated with the object. There are many key types supported by ProtectToolkit-C. For example: > CKK_AES, CKK_DES, CKK_DES2, CKK_DES3, CKK_RSA, CKK_DSA, CKK_BIP32 > CKA_ENCRYPT > CKA_DECRYPT > CKA_SIGN > CKA_VERIFY > CKA_WRAP > CKA_UNWRAP The previous attributes describe the cryptographic operations the key may be used for. Careful consideration should be given when assigning these attributes, to avoid key misuse.

Attribute	Description
CKA_IMPORT	This attribute is similar to the standard CKA_UNWRAP attribute. It is used to determine if a given key can be used to unwrap encrypted key material. The important difference between these attributes and their standard counterparts is that if CKA_IMPORT is set to True and CKA_UNWRAP attribute is set to False, then the only unwrap mechanism that can be used is CKM_WRAPKEY_DES3_CBC. With this combination, the error code CKR_MECHANISM_INVALID will be returned for all other mechanisms.
CKA_EXPORT	This attribute is similar to the CKA_WRAP attribute, in that it specifies that the key may be used to encrypt a second key, so that it may be extracted from the HSM in an encrypted form. Unlike the CKA_WRAP attribute, however, only the *Security Officer* may specify this attribute.
CKA_SENSITIVE	This attribute specifies that the key object cannot be extracted from the token in the clear. Generally this attribute should be specified to ensure the key material is not exposed. When the *No Clear PINs* flag is set only sensitive keys may be created on the HSM.
CKA_ EXTRACTABLE/ CKA_EXPORTABLE	These attributes are used to specify that the key may be extracted from the token in an encrypted (for example, wrapped) form. These attributes determine how the key may be backed up. For more information about setting these attributes to back up keys, see Secure Key Backup and Restoration in the "Operational Tasks" section of the *ProtectToolkit-C Administration Guide*.

CHAPTER 9:   Work Load Distribution

This chapter describes some of the benefits and limitations of using Work Load Distribution (WLD) in ProtectToolkit-M.

For more information about this model, see Work Load Distribution Model and High Availability in the "Cryptoki Configuration" section of the *ProtectToolkit-C Administration Guide*.

Benefits of WLD

WLD allows work to be balanced across a system by transferring units of work among HSM processing modules during execution. The demand placed on any particular processing module is thereby reduced. This results in an increase in the overall throughput of processing tasks for the system as a whole.

Utilization of multiple HSMs under WLD also provides redundancy in that if a HSM goes down, with the exception of the master HSM, the work will be shared amongst the remaining operational HSMs automatically. If the master HSM goes down this will most likely cause system failure.

WLD Limitations

Read-Only
Using ProtectToolkit-M as a CSP under WLD is severely limited. WLD does not support write/create operations. Therefore, the CSP cannot be used to create certificates when in WLD mode, as this involves creating a key pair. The CSP can, however, be used to sign certificate requests that have been generated by a client, provided the client also generated their own key pair.

Admin Token Cannot Be Distributed - Single Point of Failure
The HSM's admin token contains relevant configuration information. WLD does not allow replication of admin tokens. ProtectToolkit-M has a 'secure configuration', a collection of configuration items stored on a data object on the admin token, readable but not modifiable by anyone other than the administrator. The relevant secure configuration item here is 'clear export'. This specifies whether or not keys may be exported in the clear. Since admin tokens cannot be replicated, it is necessary to expose the admin token of one of the HSMs in the array, called the master HSM. If the master HSM fails, the admin token will no longer be available, most likely causing system failure.

CHAPTER 10:   Registry Configuration

The registry entries documented here are those created by default when ProtectToolkit-M is installed unless otherwise noted.

This section contains the following entries:

Disclaimer

The ProtectToolkit-M registry configuration, as documented in this appendix, should only be modified by personnel who are competent at making changes to the Windows registry using the **regedit** utility. Changing the registry incorrectly can leave a system in an unrecoverable state and Thales cannot be held responsible should this occur. If you are unfamiliar with editing the registry, it is strongly advised that you refer to your Windows documentation or seek help from a qualified systems administrator before attempting any changes.

ptkcRuntime

Key Location

HKEY_LOCAL_MACHINE\SOFTWARE\SAFENET\ProtectToolkit M\ptkcRuntime

Type

REG_SZ

Values

The ProtectToolkit-M product relies on the ProtectToolkit-C product. This is a string value which is used to record the version of the ProtectToolkit-C runtime installed in the ProtectToolkit-M installation directory.

CryptokiPath

Key Location

HKEY_LOCAL_MACHINE\SOFTWARE\SAFENET\ProtectToolkit M\CryptokiPath

Type

REG_SZ

Value

This string value is the path to where ProtectToolkit-M is installed. This path is used to locate the required **cryptoki.dll** file.

debugLevel

Key Location

HKEY_LOCAL_MACHINE\SOFTWARE\SAFENET\ProtectToolkit M\debugLevel

Type

REG_DWORD

Values

Valid values are 0 through 5. If turned on, debug output is saved in the **ptkm.log** file in the root directory of the current drive. Each level includes the output of all preceding levels.

> **0**: no debug output

> **1**: log the invocation of CSP related methods and if an error occurs, their return code

> **2**: Internal errors are converted to NTE_SYS_FAIL. This debug level logs the internal error at the point of conversion.

> **3**: CSP function input and output parameters - but NOT user related data (such as encrypted or clear text, or data to hash).

> **4**: Other information not covered by the preceding levels - but NOT user related data.

> **5**: Lists the Cryptoki function calls, by name only.

Note that the debug level is read when a process loads the ProtectToolkit-M library file. This means that in order to change the debug level, you must first stop any ProtectToolkit-M applications.

The log file **ptkm.log** is cleared during initialization of the ProtectToolkit-M library.

Safenet RSA Full Cryptographic Provider

Key Location

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\Defaults\Provider\Safenet RSA Full Cryptographic Provider

Description

This is the registry key (and contained values) which defines one of the CSPs installed by the ProtectToolkit-M product.

Safenet RSA SChannel Cryptographic Provider

Key Location

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\Defaults\Provider\Safenet RSA SChannel Cryptographic Provider

Description

This is the registry key (and contained values) which defines one of the CSPs installed by the ProtectToolkit-M product.

Default RSA SChannel Cryptographic Provider Type

Key

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\Defaults\Provider Types\Type 012

Description

This is the registry key which defines the default "RSA SChannel" provider. This provider is used by applications which request RSA SChannel services, but do not specify which provider (such as IIS).

Value

Name of the default provider, after installing ProtectToolkit-M. This should be "Safenet RSA SChannel Cryptographic Provider".

Default RSA Full Cryptographic Provider Type

Key Location

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\Defaults\Provider Types\Type 001

Description

This is the registry key which defines the default "RSA Full" provider. During the logon process, this provider is used to validate the entered password.

> **NOTE** The logon process requires a particular key pair to exist. This key pair does not exist in the "SafeNet RSA Full" provider. Therefore, the default should NOT be set to "SafeNet RSA Full Cryptographic Provider"

Value

Name of the default provider. This should NOT be "SafeNet RSA Full Cryptographic Provider"

Silent User Keyset Login Password

Key Location

HKEY_CURRENT_USER\Software\SafeNet\ProtectToolkit M

Description

This entry is NOT created by default. Create it manually if silent User keyset login is required. See "Silent User Keyset Login" on page 20 for further information.

Value

Enter the password as clear text for key entry "UserKeysetPassword".

Glossary

A

Adapter

The printed circuit board responsible for cryptographic processing in a HSM

AES

Advanced Encryption Standard

API

Application Programming Interface

ASO

Administration Security Officer

Asymmetric Cipher

An encryption algorithm that uses different keys for encryption and decryption. These ciphers are usually also known as public-key ciphers as one of the keys is generally public and the other is private. RSA and ElGamal are two asymmetric algorithms

B

Block Cipher

A cipher that processes input in a fixed block size greater than 8 bits. A common block size is 64 bits

Bus

One of the sets of conductors (wires, PCB tracks or connections) in an IC

C

CA

Certification Authority

CAST

Encryption algorithm developed by Carlisle Adams and Stafford Tavares

Certificate

A binding of an identity (individual, group, etc.) to a public key which is generally signed by another identity. A certificate chain is a list of certificates that indicates a chain of trust, i.e. the second certificate has signed the first, the

third has signed the second and so on

CMOS

Complementary Metal-Oxide Semiconductor. A common data storage component

Cprov

ProtectToolkit C - SafeNet's PKCS #11 Cryptoki Provider

Cryptoki

Cryptographic Token Interface Standard. (aka PKCS#11)

CSA

Cryptographic Services Adapter

CSPs

Microsoft Cryptographic Service Providers

D

Decryption

The process of recovering the plaintext from the ciphertext

DES

Cryptographic algorithm named as the Data Encryption Standard

Digital Signature

A mechanism that allows a recipient or third party to verify the originator of a document and to ensure that the document has not be altered in transit

DLL

Dynamically Linked Library. A library which is linked to application programs when they are loaded or run rather than as the final phase of compilation

DSA

Digital Signature Algorithm

E

Encryption

The process of converting the plaintext data into the ciphertext so that the content of the data is no longer obvious. Some algorithms perform this function in such a way that there is no known mechanism, other than decryption with the appropriate key, to recover the plaintext. With other algorithms there are known flaws which reduce the difficulty in recovering the plaintext

F

FIPS

Federal Information Protection Standards

FM

Functionality Module. A segment of custom program code operating inside the CSA800 HSM to provide additional or changed functionality of the hardware

FMSW

Functionality Module Dispatch Switcher

H

HA

High Availability

HIFACE

Host Interface. It is used to communicate with the host system

HSM

Hardware Security Module

I

IDEA

International Data Encryption Algorithm

IIS

Microsoft Internet Information Services

IP

Internet Protocol

J

JCA

Java Cryptography Architecture

JCE

Java Cryptography Extension

K

Keyset

A keyset is the definition given to an allocated memory space on the HSM. It contains the key information for a specific user

KWRAP

Key Wrapping Key

M

MAC

Message authentication code. A mechanism that allows a recipient of a message to determine if a message has been tampered with. Broadly there are two types of MAC algorithms, one is based on symmetric encryption algorithms and the second is based on Message Digest algorithms. This second class of MAC algorithms are known as HMAC algorithms. A DES based MAC is defined in FIPS PUB 113, see http://www.itl.n-ist.gov/div897/pubs/fip113.htm. For information on HMAC algorithms see RFC-2104 at http://www.i-etf.org/rfc/rfc2104.txt

Message Digest

A condensed representation of a data stream. A message digest will convert an arbitrary data stream into a fixed size output. This output will always be the same for the same input stream however the input cannot be reconstructed from the digest

MSCAPI

Microsoft Cryptographic API

MSDN

Microsoft Developer Network

P

Padding

A mechanism for extending the input data so that it is of the required size for a block cipher. The PKCS documents contain details on the most common padding mechanisms of PKCS#1 and PKCS#5

PCI

Peripheral Component Interconnect

PEM

Privacy Enhanced Mail

PIN

Personal Identification Number

PKCS

Public Key Cryptographic Standard. A set of standards developed by RSA Laboratories for Public Key Cryptographic processing

PKCS #11

Cryptographic Token Interface Standard developed by RSA Laboratories

PKI

Public Key Infrastructure

ProtectServer

SafeNet HSM

ProtectToolkit C

SafeNet's implementation of PKCS#11. Protecttoolkit C represents a suite of products including various PKCS#11 runtimes including software only, hardware adapter, and host security module based variants. A Remote client and server are also available

ProtectToolkit J

SafeNet's implementation of JCE. Runs on top of ProtectToolkit C

R

RC2/RC4

Ciphers designed by RSA Data Security, Inc.

RFC

Request for Comments, proposed specifications for various protocols and algorithms archived by the Internet Engineering Task Force (IETF), see http://www.ietf.org

RNG

Random Number Generator

RSA

Cryptographic algorithm by Ron Rivest, Adi Shamir and Leonard Adelman

RTC

Real-Time Clock

S

SDK

Software Development Kits Other documentation may refer to the SafeNet Cprov and Protect Toolkit J SDKs. These SDKs have been renamed ProtectToolkit C and ProtectToolkit J respectively. ☉ The names Cprov and ProtectToolkit C refer to the same device in the context of this or previous manuals. ☉ The names Protect Toolkit J and ProtectToolkit J refer to the same device in the context of this or previous manuals.

Slot

PKCS#11 slot which is capable of holding a token

SlotPKCS#11

Slot which is capable of holding a token

SO

Security Officer

Symmetric Cipher

An encryption algorithm that uses the same key for encryption and decryption. DES, RC4 and IDEA are all symmetric algorithms

T

TC

Trusted Channel

TCP/IP

Transmission Control Protocol / Internet Protocol

Token

PKCS#11 token that provides cryptographic services and access controlled secure key storage

TokenPKCS#11

Token that provides cryptographic services and access controlled secure key storage

U

URI

Universal Resource Identifier

V

VA

Validation Authority

X

X.509

Digital Certificate Standard

X.509 Certificate

Section 3.3.3 of X.509v3 defines a certificate as: "user certificate; public key certificate; certificate: The public keys of a user, together with some other information, rendered unforgeable by encipherment with the private key of the certification authority which issued it"