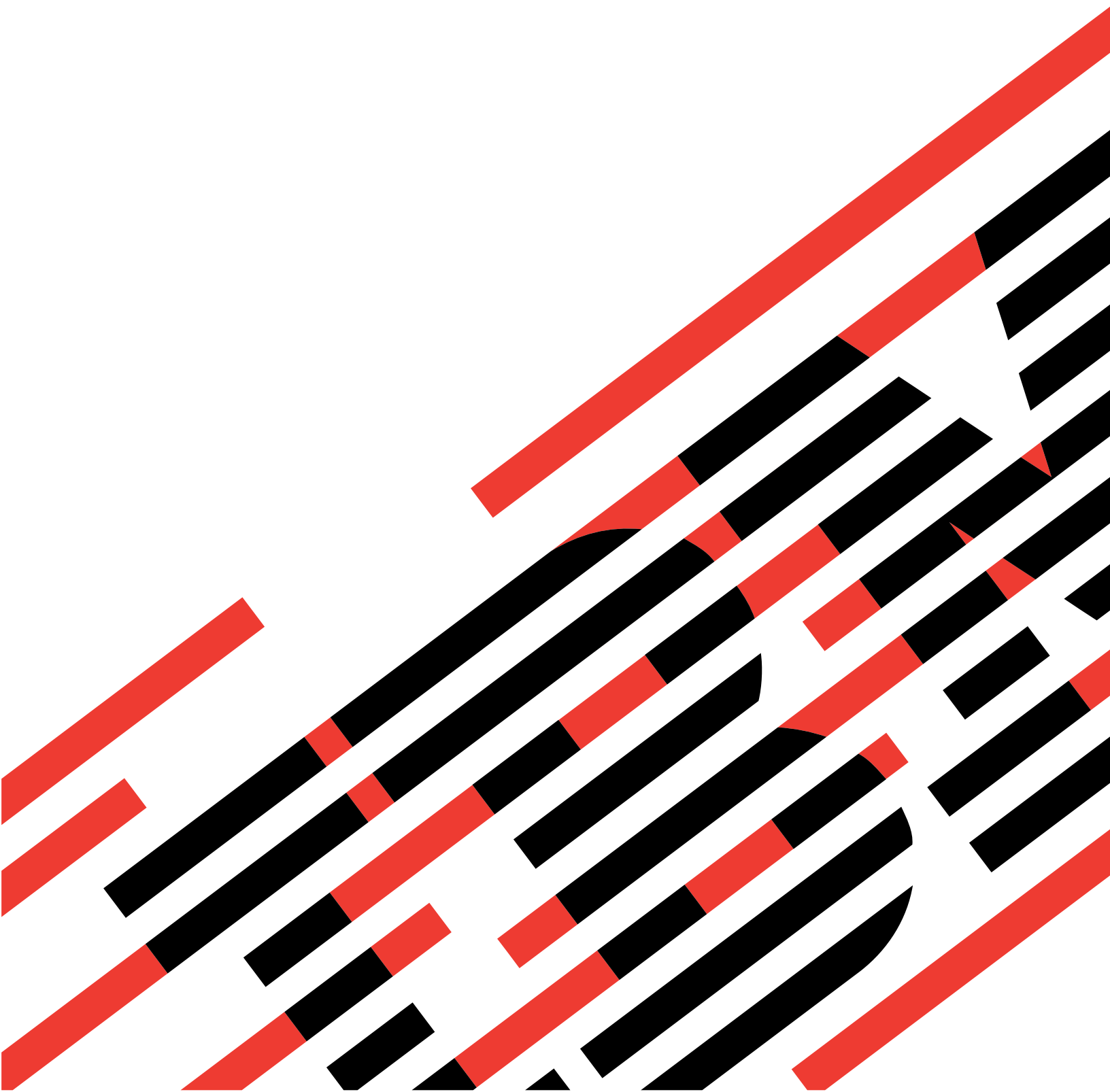


IBM

@server

iSeries

Cryptographic hardware





@server

iSeries

Cryptographic hardware

Contents

Part 1. 2058 Cryptographic Accelerator.	1	Chapter 4. 2058 Cryptographic Accelerator for iSeries	13
Chapter 1. Print this topic	3	2058 Cryptographic Accelerator features	13
Chapter 2. What's new for V5R2	5	Cryptographic hardware scenario: Enhance iSeries SSL performance	13
Chapter 3. Concepts	7	Plan for the 2058 Cryptographic Accelerator	14
		Configure the 2058 Cryptographic Accelerator.	15

Part 1. 2058 Cryptographic Accelerator

Chapter 1. Print this topic

You can view or download the PDF version of these topics:

- Cryptographic hardware (about 756 KB or 292 pages) contains all of the information regarding IBM® cryptographic hardware supported for the iSeries™ server at V5R2.
- 2058 Cryptographic Accelerator (about 79 KB or 24 pages) contains information regarding the 2058 Cryptographic Accelerator hardware supported for the iSeries server at V5R2.

Saving PDF files

To save a PDF on your workstation for viewing or printing:

1. Right-click the PDF in your browser.
2. Click **Save Target As**.
3. Navigate to the directory in which you would like to save the PDF.
4. Click **Save**.

Downloading Adobe Acrobat Reader

If you need Adobe Acrobat Reader to view or print these PDFs, you can download a copy from the Adobe Web site

(www.adobe.com/products/acrobat/readstep.html) .

Chapter 2. What's new for V5R2

If you are looking for the latest information regarding new cryptographic hardware, and added features to existing cryptographic hardware options for iSeries servers, you have come to the right place.



New cryptographic hardware: IBM 2058 e-Business Cryptographic Accelerator

The IBM 2058 e-Business Cryptographic Accelerator (Hardware Feature code 4805, and hereafter referred to as the 2058 Cryptographic Accelerator) is available, in addition to the 4758 Cryptographic Coprocessor. Designed to improve iSeries performance by rerouting the processing of private keys away from the system processors, this hardware option is an excellent choice for iSeries implementations that handle high volumes of SSL (Secure Sockets Layer) transactions. While the 2058 Cryptographic Accelerator is an excellent choice for enhancing the SSL performance of iSeries servers, and is easy to install and initialize, it does not offer the wide range of configuration options that the 4758 Coprocessor offers.

See 2058 Cryptographic Accelerator for iSeries for more information that can help you decide which cryptographic hardware option works best for your iSeries server implementation.

Additional function: 4758 Cryptographic Coprocessor

The 4758 Cryptographic Coprocessor offers customers the following new capabilities:

- Financial pin processing: Unique key per transaction (UKPT)
- Common Cryptographic Architecture (CCA) 2.4

New cryptographic hardware scenarios

To give you some ideas on how you can use cryptographic hardware with your iSeries server, we have added the following scenarios to the iSeries Information Center:

- Cryptographic hardware scenario: Enhance iSeries SSL performance

To find other information about what's new or changed this release, see the Memo to Users



How to see what's new or changed

To help you see where technical changes have been made, this information uses:

- The



image to mark where new or changed information begins.

- The  image to mark where new or changed information ends.

Chapter 3. Concepts

Cryptography

Cryptography is the art and science of keeping data secure. Basic cryptography services ensure that messages are private, that the integrity of a message is maintained, that the communicating parties are authenticated and that a party involved in a communication cannot refute having sent a message

Cryptography allows you to store information or to communicate with other parties while preventing non-involved parties from understanding the stored information or understanding the communication. Encryption transforms understandable text into an unintelligible piece of data (ciphertext). Decrypting restores the understandable text from the unintelligible data. Both processes involve a mathematical formula or algorithm and secret data (the key).

Cryptographic algorithms

There are two types of cryptographic algorithms:

1. With a secret or **symmetric key algorithm**, one key is a shared secret between two communicating parties. Encryption and decryption both use the same key. The Data Encryption Standard (DES) and Triple DES are examples of secret key algorithms.
2. With a public key or **asymmetric key algorithm**, a pair of keys is used. One of keys, the private key, is kept secret and not shared with anyone. The other key, the public key, is not secret and is shared with anyone. When data is encrypted by one of the keys, it can only be decrypted and recovered by using the other key. The two keys are mathematically related, but it is virtually impossible to derive the private key from the public key. The RSA algorithm is an example of a public key algorithm.

Both types of algorithms use keys to determine how to change the data. Different cryptographic processes use an algorithm to achieve one of several purposes. You choose the cryptographic process to use depending on the purpose, for example generating a message authentication code (MAC) to ensure data integrity. A user-written application for the 4758 Cryptographic Coprocessor calls the cryptographic process by using the corresponding security application programming interface (SAPI). Together the key and cryptographic process transform the data. A user with authorization to the SAPI has access to that cryptographic process. Therefore, the key controls access to the data. You must safeguard the keys to protect the data. If you keep the key value secret, you ensure the security of your data with each use of the algorithm with the key.

Encryption

With field level encryption, the user application explicitly requests cryptographic services. The user application completely controls key generation, selection, and distribution. The user application also controls what data to encrypt and what data to keep as plain-text. With encryption at the session layer, the system is requesting cryptographic services instead of your application. Your application may or may not be aware that encryption is happening. Link level encryption is performed at the lowest level of the protocol stack and usually by specialized hardware for that purpose. The 4758 Coprocessor supports both field level encryption and Secure

Sockets Layer (SSL) session establishment encryption, but not VPN or SNA session level encryption. The 2058 Cryptographic Accelerator only supports SSL session establishment encryption.

Data integrity

To rely on data, you need to know that it comes from an authorized source and is unchanged. This is known as data authenticity and data integrity. Your 4758 Coprocessor can ensure authenticity and integrity by creating a Message Authentication Code (MAC), a message digest, or a digital signature.

Message Authentication Code (MAC)

The MAC process is a data integrity technique in which you define critical data elements. For example, you could define the amount in a funds transfer message. The critical data elements, cryptographic algorithm, and secret MAC key generate the MAC. The MAC becomes part of the message and travels with it. The MAC process uses DES or Triple DES keys.

The receiver of the message uses the same MAC key, algorithm, and procedure as the sender to reproduce the MAC. If the receiver's MAC matches the MAC sent with the message, they can accept the MAC as unaltered.

The MAC process helps authenticate received messages, but does not prevent unauthorized reading because the transmitted data remains as plaintext. By using the MAC process and then encrypting the entire message, you can more effectively protect both data privacy and integrity.

Message digest

A message digest process can be performed on data to produce a digest value which can be thought of as a cryptographically generated checksum. If any portion of the data is modified, a different digest would be generated. You can keep copies of message digests, and compare them. Message digests that are identical indicate that no data has been modified.

Digital signature

A digital signature can also be used to verify authenticity and integrity. It is a two step process:

1. First a digest is generated from the data and then the digest is encrypted using a private RSA key. The result is a digital signature. The signature can be verified by decrypting the signature using the public key to recover the original digest.
2. Another digest is generated from the data and is compared to the original digest. If the two are identical, then the signature is validated and you can be confident that the data has not been altered.

Key types associated with the 4758 Cryptographic Coprocessor

Your 4758 Coprocessor uses various key types. Not all DES or Triple DES keys can be used for all symmetric key operations. Likewise, not all public key algorithm (PKA) keys can be used for all asymmetric key operations. This is a list of the various key types which the 4758 Coprocessor uses:

Master key

This is a clear key, which means that no other key encrypted it.

The 4758 Coprocessor uses the master key to encrypt all operational keys. The 4758 Coprocessor stores the master key in a tamper-responding module. You cannot retrieve the master key from the 4758 Coprocessor. The 4758 Coprocessor responds to tamper attempts by destroying the master key and destroying its factory certification. The 4758-023 has two master keys: one for encrypting DES keys and one for encrypting PKA keys.

Double-length key-encrypting keys

Your 4758 Coprocessor uses this type of Triple-DES key to encrypt or decrypt other DES or Triple DES keys. Key-encrypting-keys are generally used to transport keys between systems. However, they can also be used for storing keys offline for backup. If key-encrypting-keys are used to transport keys, the clear value of the key-encrypting-key itself must be shared between the two systems. Exporter key-encrypting keys are used for export operations where a key encrypted under the master key is decrypted and then encrypted under the key-encrypting key. Importer key-encrypting keys are used for import operations where a key encrypted under the key-encrypting key is decrypted and then encrypted under the master key.

Double-length PIN keys

Your 4758 Coprocessor uses this type of key to generate, verify, encrypt, and decrypt PINs used in financial operations. These are Triple DES keys.

MAC keys

Your 4758 Coprocessor uses this type of key to generate Message Authentication Codes (MAC). These can be either DES or Triple DES keys.

Cipher keys

Your 4758 Coprocessor uses this type of key to encrypt or decrypt data. These can be either DES or Triple DES keys.

Single-length compatibility keys

Your 4758 Coprocessor uses this type of key to encrypt or decrypt data and generate MACs. These are DES keys and are often used when encrypted data or MACs are exchanged with systems that do not implement the Common Cryptographic Architecture.

Private keys

Your 4758 Coprocessor uses private keys for generating digital signatures and for decrypting DES or Triple DES keys encrypted by the public key.

Public keys

Your 4758 Coprocessor uses public keys for verifying digital signatures, for encrypting DES or Triple DES keys, and for decrypting data encrypted by the private key.

Key forms

The 4758 Coprocessor works with keys in one of four different forms. The key form, along with the key type, determines how a cryptographic process uses that key. The four forms are:

Clear form

The clear value of the key is not protected by any cryptographic means. Clear keys are not usable by the 4758 Coprocessor. The

clear keys must first be imported into the secure module and encrypted under the master key and then stored outside the secure module.

Operational form

Keys encrypted under the master key are in operational form. They are directly usable for cryptographic operations by the 4758 Coprocessor. Operational keys are also called internal keys. All keys that are stored in the server key store file are operational keys. However, you do not need to store all operational keys in the key store file.

Export form

Keys encrypted under an exporter key-encrypting key as the result of an export operation are in export form. These keys are also called external keys. A key in export form can also be described as being in import form if an importer key-encrypting key with the same clear key value as the exporter key-encrypting key is present. You may store keys in export form in any manner that you choose, however, you can not store them in key store files.

Import form

Keys encrypted under an importer key-encrypting key are in import form. Only keys in import form can be used as the source for an import operation. These keys are also called external keys. A key in import form can also be described as being in export form if an exporter key-encrypting key with the same clear key value as the importer key-encrypting key is present. You may store keys in import form in any manner that you choose, however, you can not store them in key store files.

Function control vector

IBM provides a digitally signed value known as a function control vector. This value enables the cryptographic application within the 4758 Coprocessor to yield a level of cryptographic service consistent with applicable import regulations and export regulations. The function control vector is shipped with the IBM Cryptographic Access Provider (5722-ACx) product you install on your system. The path name of the file is /QIBM/ProdData/CAP/FCV.CRT. The function control vector provides your 4758 Coprocessor with the key length information necessary to create keys.

Control vectors

A control vector, different from a function control vector, is a known value associated with a key that governs the following:

- Key type
- What other keys this key can encrypt
- Whether your 4758 Coprocessor can export this key
- Other allowed uses for this key

The control vector is cryptographically linked to a key and can not be changed without changing the value of the key at the same time.

Key store file

An OS/400® database file that is used to store keys which you encrypted under the master key of the 4758 Coprocessor.

Key token


A data structure that can contain a cryptographic key, a control vector, and

other information related to the key. Key tokens are used as parameters on most of the CCA API verbs that either act on or use keys.

Chapter 4. 2058 Cryptographic Accelerator for iSeries



The 2058 Cryptographic Accelerator is available for customers to use with a V5R2 (or later) iSeries server. The 2058 Cryptographic Accelerator provides a competitive option to customers who do not require the high security of a 4758 Cryptographic Coprocessor, but do need the high cryptographic performance that hardware acceleration provides to offload a host processor. The 2058 Cryptographic Accelerator has been designed to improve the performance of those SSL applications that do not require secure key storage. It does not provide tamper-resistant storage for keys, like the 4758 Cryptographic Coprocessor. You can install up to four 2058 Cryptographic Accelerator cards in an iSeries server.

The 2058 Cryptographic Accelerator provides special hardware which is optimized for RSA encryption (modular exponentiation) with data key lengths up to 2048 bits. The 2058 Accelerator uses multiple RSA (Rivest, Shamir and Adleman algorithm) engines. Refer to the iSeries Performance Management  web site for performance information specific to your iSeries server model.

For more information about the 2058 Cryptographic Accelerator, refer to the following pages:

- 2058 Cryptographic Accelerator features
- Cryptographic hardware scenario: Enhance iSeries SSL performance
- Plan for the 2058 Cryptographic Accelerator
- Configure the 2058 Cryptographic Accelerator

2058 Cryptographic Accelerator features

Some features of the 2058 Cryptographic Accelerator include:

- Single card high performance cryptographic adapter (standard PCI card)
- Designed and optimized for RSA encryption
- Onboard hardware-based RNG (random number generator)
- Five mounted IBM UltraCypher Cryptographic Engines

See the following information regarding the 2058 Cryptographic Accelerator:

- Plan for the 2058 Cryptographic Accelerator
- Configure the 2058 Cryptographic Accelerator

Cryptographic hardware scenario: Enhance iSeries SSL performance

To give you an idea of how you can use this cryptographic hardware with your iSeries server, we have added this usage scenario.

Situation

A company's iSeries server handles thousands of secured Internet transactions per day. The company's transactions utilize the Secure Sockets layer and Transport Layer Security protocols (SSL and TLS) – a common method for securing Internet transactions. This company's system administrator, Sue, wants to free up server

resources for additional application processing, including the ability to support even more SSL transactions. Sue is looking for a solution that fits these objectives:

- A sizeable increase in the available server resources for application processing, including additional SSL transactions
- Minimal installation and configuration effort
- Minimal resource management requirements

Based on these objectives, Sue orders and installs an IBM 2058 e-Business Cryptographic Accelerator. (hereafter referred to as a 2058 Cryptographic Accelerator). The 2058 Cryptographic Accelerator is a PCI (Peripheral Component Interconnect) card, which is specially designed to accelerate the very compute intensive processing required when establishing a SSL/TLS session. On iSeries servers the 2058 Cryptographic Accelerator can be obtained by ordering hardware feature code 4805.

Details

1. The iSeries server has a 2058 Cryptographic Accelerator installed and configured.
2. The iSeries server receives a high number of SSL transaction requests from the network.
3. The 2058 Cryptographic Accelerator performs the cryptographic processing in the initiation of SSL transactions, and caches the private keys that are associated with the digital certificates for SSL transactions.

Prerequisites and assumptions

This scenario assumes that Sue has planned for the installation of the 2058 Cryptographic Accelerator, and then configured the card properly (see Plan for the 2058 Cryptographic Accelerator, and Configure the 2058 Cryptographic Accelerator). This scenario also assumes that Sue has already set up a digital certificate for SSL.

Configuration steps

Sue completes the following steps to enhance the SSL performance of her company's iSeries server:

1. Order Hardware Feature code 4805, which provides the 2058 Cryptographic Accelerator.
2. Install the 2058 Cryptographic Accelerator.
3. Create a device description for the 2058 Cryptographic Accelerator, and vary-on the device (see Configure the 2058 Cryptographic Accelerator for details).

Plan for the 2058 Cryptographic Accelerator

Your server must meet these requirements before you install and use the 2058 Cryptographic Accelerator.

Hardware requirements

The IBM e-Business Cryptographic Accelerator (orderable feature code 4805, and hereafter referred to as the 2058 Cryptographic Accelerator). The 4805 feature is a standard PCI card, and is supported on the following iSeries server models:

- 270
- 810, 820, 825, 830, 840, 870 and 890
- SB2 and SB3
- Expansion units 5074, 5075, 5078, 5079, 5088, 5094, 5095 and 5294

OS/400 and SSL requirements

The 2058 Cryptographic Accelerator requires the OS/400 V5R2M0 (Version 5 Release 2 Modification 0) software. Although the 2058 Cryptographic Accelerator is fully enabled for cryptographic operations, the Cryptographic Access Provider 128-bit (5722-AC3) licensed program product must also be installed on the iSeries server to enable the cryptographic functions in OS/400 that SSL also uses.

Configure the 2058 Cryptographic Accelerator

You must create a device description so that OS/400 SSL can direct RSA cryptographic operations to the 2058 Cryptographic Accelerator. You can create a device description by using the Create Device Crypto CL command.


Create device description

To create a device description using the CL command, follow these steps:

1. Type CRTDEVCRP at the command line.
2. Specify a name for the device as prompted.
3. Accept the default name of the PKA key store: *NONE.
4. Accept the name default of the DES key store: *NONE.
5. Specify a description as prompted. This is optional.
6. Use either the Vary Configuration (VRYCFG) or the Work with Configuration Status (WRKCFGSTS) CL commands to vary on the device once you have created the device description.

For digital certificates that are generated by software, and stored in software, OS/400 SSL automatically starts using the 2058 Cryptographic Accelerator once the device is varied-on. The private key processing associated with SSL and TLS session establishment is off-loaded to the 2058 Cryptographic Accelerator. When the device is varied-off, OS/400 SSL switches back to software based encryption for establishing SSL and TLS sessions, thereby placing the private key processing load back on the server.

Note: This is only true for certificates and private keys that were not created by the 4758 Cryptographic Coprocessor. If a certificate was generated using the 4758 Cryptographic Coprocessor, the 4758 Cryptographic Coprocessor has to be used for those SSL or TLS sessions which use that particular certificate.

The Cryptographic hardware scenario: Enhance iSeries SSL performance page offers an iSeries server usage scenario for the 2058 Cryptographic Accelerator, once it has been installed and varied on. 



Printed in U.S.A.