IBM SecureWay Cryptographic Products

IBM 4758 PCI Cryptographic Coprocessor

General Information Manual

GC31-8608-00

Note!

Before using this information and the products it supports, be sure to read the general information under “Notices” on page v.

First Edition (June, 1997)

This edition, GC31-8608-00, applies to the PCI Cryptographic Coprocessor product as announced in June, 1997.

Changes are made periodically to the information herein; before using this publication in connection with the operation of IBM systems, consult your IBM representative to be sure that you have the latest edition and any Technical Newsletters.

IBM does not stock publications at the address given below; requests for IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for reader’s comments is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM Corporation, Department VM9A, MG81, 8501 IBM Drive, Charlotte, NC 28262-8563, U.S.A. IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Contents

Notices . . . v
Copying and Distributing Softcopy Files . . . v
Trademarks . . . vi

About This Publication . . . vii

Related Publications . . . ix
IBM 4758 PCI Cryptographic Coprocessor Publications . . . ix
Cryptography Publications . . . ix
Other IBM Cryptographic Product Publications . . . ix

Summary of Changes . . . xi

Chapter 1. Introduction . . . 1-1
The PCI Cryptographic Coprocessor . . . 1-1
Current Product Features . . . 1-2
The PCI Cryptographic Coprocessor and Other IBM SecureWay Products . . . 1-3
Choosing a Cryptographic Solution . . . 1-3

Chapter 2. Cryptographic Support Overview . . . 2-1
DES Symmetric Key Algorithm . . . 2-2
DES and Data Confidentiality . . . 2-2
DES and Data Integrity . . . 2-2
RSA Public Key Cryptography . . . 2-3
Basic RSA Cryptography . . . 2-3
RSA Key Distribution . . . 2-4
Digital Signatures . . . 2-5
Hashing . . . 2-5
Certification Authorities . . . 2-6
The Complete Digital Signature Process . . . 2-7
Cryptographic Key Management . . . 2-8
Key Generation and Distribution . . . 2-8
Role-Based Access Control . . . 2-9
Additional PCI Cryptographic Coprocessor Capabilities . . . 2-10
Credit Card Transaction Security . . . 2-10
Private Key Protection for Certification Authorities . . . 2-10
Data Confidentiality within Systems Network Architecture . . . 2-11
Governmental Regulations on Cryptography . . . 2-11

Chapter 3. PCI Cryptographic Coprocessor Features . . . 3-1
IBM 4758 PCI Cryptographic Coprocessor . . . 3-1
Overview . . . 3-1
Description . . . 3-2
Requirements and Specifications . . . 3-3
CCA Cryptographic Coprocessor Support Program Feature . . . 3-3
Overview . . . 3-3
Description . . . 3-4
Requirements and Specifications . . . 3-5
Ordering Information . . . 3-5
Before You Order . . . 3-5

Ordering the Products . . . 3-6

List of Abbreviations . . . X-1

Glossary . . . X-3

Index . . . X-9

Figures

2-1. Protecting Data with Encryption . . . 2-1
2-2. DES Algorithm . . . 2-2
2-3. RSA Algorithm . . . 2-3
2-4. Using RSA for DES Key Exchange . . . 2-4
2-5. Complete Digital Signature Process . . . 2-8
3-1. The IBM 4758 PCI Cryptographic Coprocessor . . . 3-1

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights or other legally protectable rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, programs, or services, except those expressly designated by IBM, are the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY, 10594, USA.

Copying and Distributing Softcopy Files

For online versions of this book, we authorize you to:

- Copy, modify, and print the documentation contained on the media, for use within your enterprise, provided you reproduce the copyright notice, all warning statements, and other required statements on each copy or partial copy.
- Transfer the original unaltered copy of the documentation when you transfer the related IBM product (which may be either machines you own, or programs, if the program's license terms permit a transfer). You must, at the same time, destroy all other copies of the documentation.

You are responsible for payment of any taxes, including personal property taxes, resulting from this authorization.

THERE ARE NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Some jurisdictions do not allow the exclusion of implied warranties, so the above exclusion may not apply to you.

Your failure to comply with the terms above terminates this authorization. Upon termination, you must destroy your machine readable documentation.

Trademarks

The following terms, denoted by an asterisk (*) in this publication, are trademarks of the IBM Corporation in the United States or other countries or both:

3090
ACF/VTAM
AIX
AIX/6000
Application System/400
AS/400
CICS
Enterprise System/3090
Enterprise System/9000
Enterprise System/9370
ES/3090
ES/9000
ES/9370
IBM
IBM net.commerce
IBM World Registry
Micro Channel
Operating System/2
OS/2
Operating System/400
OS/400
Personal Security
Personal System/2
PS/2
PS/ValuePoint
POWERserver
POWERstation
RACF
RS/6000
SecureWay
System/360
System/370
System/390
S/390 G3 Enterprise Server
S/390 Multiprise
Systems Application Architecture

The following terms, denoted by a double asterisk (**) in this publication, are the trademarks of other companies:

Acrobat Reader	Adobe Systems, Inc.
Java	Sun Microsystems
MASTERCARD	MasterCard International, Incorporated
Pentium	Intel Corporation
NCR	National Cash Register Corporation
RSA	RSA Data Security, Inc.
UNIX	UNIX Systems Laboratories, Incorporated
VISA	VISA International Service Association

About This Publication

This *IBM* 4758 PCI Cryptographic Coprocessor General Information Manual* describes the IBM 4758 PCI Cryptographic Coprocessor. This manual's audience includes customer executives, system analysts, application programmers, and others who:

- Evaluate the benefits and the advantages of the product.
- Select and order a suitable configuration of the product.
- Create applications that employ the product.

Chapter 1, “Introduction” introduces the PCI Cryptographic Coprocessor.

Chapter 2, “Cryptographic Support Overview” introduces the cryptographic services used in commercial practice and describes how the PCI Cryptographic Coprocessor performs these services.

Chapter 3, “PCI Cryptographic Coprocessor Features” describes the current PCI Cryptographic Coprocessor features and includes ordering information.

A list of abbreviations, a glossary, and an index complete the manual.

Related Publications

The list below reflects source information regarding the PCI Cryptographic Coprocessor, commercial cryptography, and products that use the IBM Common Cryptographic Architecture (CCA) Application Programming Interface (API).

IBM 4758 PCI Cryptographic Coprocessor Publications

Check the product Web site at http://www.ibm.com/security/cryptocards for the availability of these publications. From the Web site, you can download, view, and print publications available in the Adobe Acrobat** PDF format.

- *IBM 4758 PCI Cryptographic Coprocessor Installation Manual*, SC31-8623
- *IBM 4758 CCA Basic Services Reference and Guide*, SC31-8609
- *IBM 4758 CCA Services Installation and I/O Guide*, SC31-8610

Cryptography Publications

The following publications describe cryptographic standards, research, and practices relevant to the PCI Cryptographic Coprocessor:

- *Applied Cryptography: Protocols, Algorithms, and Source Code in C, Second Edition*, Bruce Schneier, John Wiley & Sons, Inc. ISBN 0-471-12845-7 or ISBN 0-471-11709-9
- *IBM Systems Journal* Volume 30 Number 2, 1991, G321-0103
- *IBM Systems Journal* Volume 32 Number 3, 1993, G321-5521
- *IBM Journal of Research and Development Volume 38 Number 2, 1994*, G322-0191
- *USA Federal Information Processing Standard (FIPS):*
  - *Data Encryption Standard,* 46-1-1988
  - *Secure Hash Algorithm,* 180-1, May 31, 1994
  - *Cryptographic Module Security,* 140-1
- *ISO 9796 Digital Signal Standard*
- *Internet Engineering Taskforce RFC 1321*, April 1992, MD5
- *Secure Electronic Transaction Protocol Version 1.0*, May 31, 1997

Other IBM Cryptographic Product Publications

The following publications describe products that utilize the IBM Common Cryptographic Architecture (CCA) Application Programming Interface (API).

- *IBM Transaction Security System General Information Manual*, GA34-2137
- *IBM Transaction Security System Basic CCA Cryptographic Services*, SA34-2362
- *IBM Transaction Security System I/O Programming Guide*, SA34-2363

- *IBM Transaction Security System Finance Industry CCA Cryptographic Programming*, SA34-2364
- *IBM Transaction Security System Workstation Cryptographic Support Installation and I/O Guide*, GC31-4509
- *IBM 4755 Cryptographic Adapter Installation Instructions*, GC31-4503
- *IBM Transaction Security System Physical Planning Manual*, GC31-4505
- *IBM Common Cryptographic Architecture Services/400 Installation and Operators Guide, Version 2*, SC41-0102
- *IBM Common Cryptographic Architecture Services/400 Installation and Operators Guide, Version 3*, SC41-0102
- *IBM ICSF/MVS General Information*, GC23-0093
- *IBM ICSF/MVS Application Programmer's Guide*, SC23-0098

Summary of Changes

This first revision of the *IBM 4758 PCI Cryptographic Coprocessor General Information Manual*, GC31-8608, contains product information that is current with the IBM 4758 PCI Cryptographic Coprocessor announcements made in June 1997.

Chapter 1. Introduction

This chapter:

- Introduces the PCI Cryptographic Coprocessor.
- Explains the relationship between the PCI Cryptographic Coprocessor and the SecureWay product line.
- Lists points to consider when choosing a data security solution.

For general information on cryptography, see Chapter 2, “Cryptographic Support Overview.”

The PCI Cryptographic Coprocessor

The IBM 4758 PCI Cryptographic Coprocessor—a member of the SecureWay family of products—ensures the privacy and integrity of business, financial, and personal transactions by furnishing proven cryptographic services for protecting stored and trasmitted data, data authentication, and e-Commerce.

Do you need to:

- Transfer a confidential message that can be read only by your intended receiver?
- Ensure that a message has not been altered?
- Know that you can hold a party liable for the content of a message?
- Exchange cryptographic keys with other users?
- Make secure credit card transactions over the Internet?
- Protect an RSA** private key on both primary and backup equipment?
- Ensure that your digital signature cannot be copied from transmissions you have signed?

These and other data-protection needs are addressed by the PCI Cryptographic Coprocessor. The PCI Cryptographic Coprocessor furnishes a comprehensive set of cryptographic services using the IBM Common Cryptographic Architecture (CCA) Application Programming Interface (API). Dedicated hardware executes cryptographic computation for fast and secure throughput.

The CCA enables a consistent approach to cryptography on major IBM computing platforms, and supports application software written in a variety of programming languages. Application software can call on CCA services to perform a wide range of cryptographic functions, including DES and RSA encryption.

The DES and RSA algorithms are the most widely used in commercial practice. Given the strength of the algorithms, the vulnerability of many cryptographic systems lies with its key management system. When a key is compromised, the data encrypted with that key is vulnerable. To address this issue, the PCI Cryptographic Coprocessor is designed to extend comprehensive protection to the cryptographic keys protecting your data.

To prevent attacks on the key management system, the PCI Cryptographic Coprocessor and CCA have been expertly designed for flexible and secure key storage and distribution. For example:

- Cryptographic keys are triple-encrypted using a master key stored within the dedicated hardware.
- The CCA services protect end-to-end data communications and furnish various key management techniques.
- A role-based access-control system allows you to define user levels (such as “general user” or “key management officer”) for consistent use throughout your organization.
- A hardware-based random-number generator ensures unpredictable keys.

In addition, IBM and other vendors can prepare custom software to meet your specifications. The resulting code is securely loaded onto the cryptographic hardware, where your keys and other data can then be safely processed.

Current Product Features

The following PCI Cryptographic Coprocessor product features are available now:

IBM 4758 PCI Cryptographic Coprocessor

The PCI Cryptographic Coprocessor is a state-of-the-art, programmable PCI-bus card offering high-performance DES and RSA cryptographic processing. The cryptographic processes are performed within a secure enclosure on the card that is designed to meet the stringent requirements of the FIPS PUB 140-1 level 4 standard. All software operating within the coprocessor's secure environment is first authenticated using digital signature techniques.

With the appropriate software, the PCI Cryptographic Coprocessor supports:

- DES data privacy and MAC processing
- RSA key generation and digital signature processing
- RSA- and DES-based key exchange
- SET acquirer-and-merchant processes
- SHA-1 and MD5 hashing algorithms
- Custom security-subroutine processing

The PCI Cryptographic Coprocessor provides pipe-lined DES processing and transports data into and out of the card using two DMA channels. RSA and other public-key algorithms employ an integral 1024-bit arithmetic processor and are supported by microcode for RSA-2048-bit key-length processing. A hardware random-number generator, clock-calendar, serial port, advanced tamper-detecting and tamper-responding circuitry, Intel-compatible 80486 processor, and various memory technologies together provide the basis for high-integrity, high-performance security and cryptographic processing.

For more detailed information about cryptographic functions, see Chapter 2, “Cryptographic Support Overview.” For more information about the IBM 4758 PCI Cryptographic Coprocessor, see “IBM 4758 PCI Cryptographic Coprocessor” on page 3-1.

CCA Cryptographic Coprocessor Support Program feature

The CCA Cryptographic Coprocessor Support Program is an optional feature of the PCI Cryptographic Coprocessor. It includes the following software: device drivers, utilities, and the IBM Common Cryptographic Architecture (CCA) interface. This feature is compatible with IBM RS/6000* machines running the AIX* operating system, or personal computers running the OS/2* or Windows NT** operating systems.

Several variations of the Coprocessor Support Program are available to meet the cryptographic import-and-export regulations imposed by national governments. For more information about the CCA Cryptographic Coprocessor Support Program, see “CCA Cryptographic Coprocessor Support Program Feature” on page 3-3.

The PCI Cryptographic Coprocessor and Other IBM SecureWay Products

The IBM 4758 PCI Cryptographic Coprocessor is part of the IBM SecureWay family of products available across the IBM product line. It is distinguished by the following features:

- Industry-leading, high-speed cryptographic processing
- Integration with the IBM Common Cryptographic Architecture API: application programs can use the cryptographic and key management services without external utilities
- Role-based access control for cryptographic functions, as described in “Role-Based Access Control” on page 2-9

The Coprocessor Support Program supports a subset of the IBM CCA services available with Transaction Security System (TSS) products, the IBM S/390* integrated cryptographic feature, and the IBM AS/400* integrated cryptographic feature. In addition, SET processing and Windows NT compatibility are included.

IBM anticipates extending, in future software releases, the CCA services offered with the PCI Cryptographic Coprocessor.

Choosing a Cryptographic Solution

Once you have determined that cryptographic techniques are part of your security application requirements, how do you decide what products to use? Consider these points:

- Is there an off-the-shelf application that provides adequate security? Can it be extended to take advantage of the cryptographic services available through the CCA Application Programming Interface (API)?

  For session-level SNA encryption, IBM communication and transaction middleware supports CCA and the PCI Cryptographic Coprocessor. See “Data Confidentiality within Systems Network Architecture” on page 2-11.
- When custom applications are used, are the cryptographic processes employed exclusively within the application software? How well integrated are the encryption, decryption, and key management functions?

The history of cryptographic practice is rife with failed approaches, often the result of very subtle mistakes. Adherence to standards and the use of products designed to a carefully-constructed architecture will help you to avoid the pitfalls of the past.

Often, systems are defeated, not by breaking the cryptographic algorithm but by attacks on the key management system. The CCA organizes cryptographic functions into a set of services that protects your secret and private keys.

The IBM PCI Cryptographic Coprocessor implementation of CCA—including key management and proven DES and RSA algorithms—provides controls and services necessary for a secure solution.

The IBM net.commerce* product and other electronic commerce middleware products can use the PCI Cryptographic Coprocessor products to enhance data security.

- What application programming language is best for you?

  The CCA API has been designed according to Systems Application Architecture (SAA) concepts. Calls to the programming interface can be made by most high-level application programming languages, and the API specifically supports the C programming language. In addition, the same interface is available on each of the supported computing platforms. Consistent, multi-platform solutions are generally more effective and less costly than independent, platform-specific solutions.

- Why use hardware when software solutions can be less expensive?

  Software may be appropriate in some cases, but consider these points:

  – Is the software environment trustworthy?

    - Are your private RSA keys stored on the hard disk, where they are vulnerable to attack?
    - Does the open, “personal” environment that makes PCs and workstations so useful allow your users to misuse their authority or bypass controls?

    A secure hardware implementation makes it extremely difficult to bypass controls established by security management.

  – Can you demonstrate to your auditors that you are practicing a realistic approach to security given the risk of loss as compared to the cost of hardware-based protection?

  – Can software running on a general-purpose processor provide adequate throughput?

- The PCI Cryptographic Coprocessor is a complete system with secure hardware *and* supporting software; it allows your application programs to offload sensitive operations.

- The PCI Cryptographic Coprocessor is internally programmable, and securely loads and validates internal software. IBM and other vendors can design and implement new functions at your request, or we can combine functions into a single application call to improve the performance of complex processes.

IBM security products come from a long tradition of cryptographic research and development. The DES algorithm is a direct descendant of Lucifer (an early encryption algorithm developed by IBM). Recently, IBM has been instrumental in outlining—in open dialog—ways to protect credit transactions in Internet-based electronic commerce. IBM played a key role in bringing Mastercard**, Visa**, and other organizations together to establish the Secure Electronic Transaction** (SET) standard.

Now, with the SecureWay family of products, IBM is accelerating its investment in cryptographic technology. Examples include:

- The PCI Cryptographic Coprocessor.
- The System/390 Integrated Cryptographic Coprocessor designed for high-performance cryptography on large server machines.
- The development of a *key recovery* technique you can use—under controlled procedures—to recover a cryptographic key. This technique can address law enforcement and national security requirements.

IBM and the SecureWay Team want to work with you to address your security needs.

Chapter 2. Cryptographic Support Overview

This chapter describes cryptographic processes commonly used in commercial practice, and explains how they are executed by the PCI Cryptographic Coprocessor and the CCA software feature.

Historically, cryptographic techniques are used to disguise information as it is moved from place to place so that an adversary can not learn the true meaning of the information. In modern life, information and communications are digitized, recorded in data banks, and transmitted through computer networks; it is increasingly difficult to know how or where an adversary might intercept, modify, or replay information.

To make digitized information “secret,” you can process the data through an algorithm that *enciphers* the data into *ciphertext*. Later, you—or an authorized recipient—can retrieve the original information by *deciphering* the data back to the original *plaintext*.

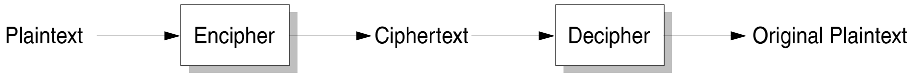

Figure 2-1. Protecting Data with Encryption

In addition, you might need to ensure that the data you receive from others has not been altered, and to confirm the identity of the parties with whom you share data. Valuable information should be protected and validated wherever it is used.

Modern cryptographic techniques address the following data-protection concerns:

Data Confidentiality This is the classic use of cryptography: transforming information so that the meaning of the data is not apparent to an adversary.

Data Integrity Append a code to data so that the recipient can verify that the data is unmodified.

Non-repudiation Include a *digital signature* with data so that the originator can not later falsely deny originating the data.

Many ciphering schemes devised throughout history—including digital ciphers—later proved weak. Creating strong cryptographic systems, and knowing that the systems are strong, is an arcane pursuit. For valuable information you will want a strong system. And, if you are going to interchange your information with other organizations or applications, your security will need to be compatible with them. For these reasons, the best approach to commercial cryptography is to use standardized algorithms that have been widely and openly discussed, reviewed, and tested.

DES Symmetric Key Algorithm

In commercial practice, the algorithm most widely used for implementing strong data confidentiality is the Data Encryption Algorithm (DEA). This algorithm was designed by IBM more than twenty years ago in collaboration with the United States National Security Agency. The algorithm has withstood years of scrutiny by cryptanalysts, and although it is now “old,” it is still considered among the strongest of block ciphers. The DEA is now generally called “DES” (Data Encryption Standard) and is standardized by the US Government, ANSI, ISO, and other organizations.

DES and Data Confidentiality

DES is an effective tool for protecting the confidentiality of your data. To protect data, the DES algorithm uses a 56-bit data string known as a *key*. The key determines the relationship between the input and the output of the DES algorithm. Upon receipt of the encrypted data, the receiver recovers the original plaintext by decrypting the ciphertext using the same key. Only a matching key can successfully decrypt the data.

DES is a *secret-key*, or *symmetric*, algorithm because it employs the same key in both the enciphering and deciphering processes.

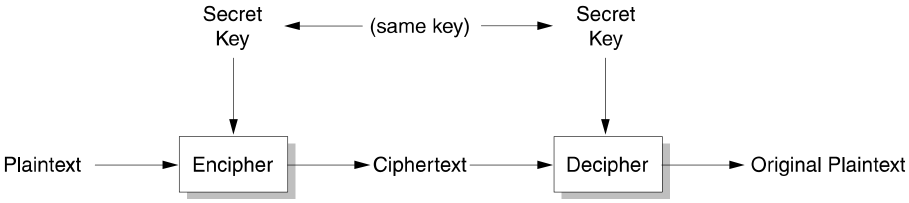

Figure 2-2. DES Algorithm

The only known way to attack DES is to:

1. Obtain a block of ciphertext data and the corresponding plaintext.
2. Use all possible combinations of encryption keys to decrypt the ciphertext data until the result matches the plaintext.

The DES algorithm uses a 56-bit encryption key, so there are 2^{56} keys possible. Given this extremely large number of keys, attempts to break this cipher are believed to require years of repetitive trials. Since the DES algorithm is public knowledge, the secrecy of ciphertext is related to your ability to keep the key a secret, and to be sure that you have used a sufficiently-random quantity as a key.

To protect data confidentiality, the PCI Cryptographic Coprocessor encrypts data using DES according to CBC and ANSI X9.23 “last block” padding rules.

DES and Data Integrity

DES can ensure data integrity by creating a *message authentication code* (MAC). A MAC is computed on a string of data using the DES algorithm and rules defined in ANSI X9.9 or ISO 8730. The data originator uses a secret key to append a 32- or 64-bit MAC value to a data string of any length. A recipient with the same secret key can then compute the MAC value. If the received and locally-computed MACs are the same, the data can be accepted as unchanged.

The IBM Common Cryptographic Architecture (CCA)—using *control vector* technology—can separate the ability to verify a MAC from the ability to generate one, so the sender of a MAC can deny the receiver the ability to generate a valid MAC from the same key and data used in transmission. The PCI Cryptographic Coprocessor provides specific MAC generation-and-verification services, and the services and support required to distribute and protect MAC keys.

RSA Public Key Cryptography

Although DES cryptography is strong, you can not use DES to encrypt a confidential message to another party unless you have previously established a shared secret key. This problem has been addressed by a branch of cryptography known as *asymmetric*, or *public-key*, cryptography. Public-key cryptography is based on the *RSA* algorithm, which was first published in 1978 by Rivest, Shamir, and Adleman. It is now widely used in commercial cryptographic applications.

In combination with other cryptographic technologies, it is used:

- To encrypt and decrypt DES symmetric keys.
- As part of digital signature and other cryptographic processes.

Basic RSA Cryptography

The RSA algorithm uses two different keys, commonly called the *public key* and the *private key*, to cipher data. With RSA encryption, you can freely distribute or publish your public keys. As shown in Figure 2-3, others can encipher messages using your public key without first establishing and distributing a secret key, since the message cannot be decrypted without the private key kept in your possession.

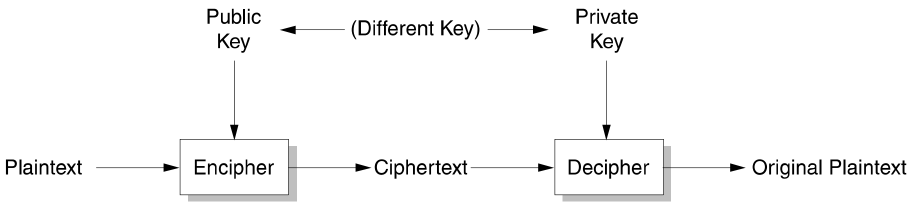

Figure 2-3. RSA Algorithm

Whereas symmetric cryptography enables two parties to exchange information in a secure manner, both with equal capability and obligation, asymmetric (public-key) cryptography provides different capabilities and imposes different responsibilities on the sender and receiver. Public-key cryptography has a many-to-one and one-to-many capability requiring:

- A single entity to be held responsible for secure storage and controlled usage of a private key
- Users to verify the authenticity and status of the public keys they receive

Mathematically the RSA algorithm relies on the difficulty of factoring a large number that is the product of prime numbers, and uses keys constructed of several large integer values. RSA keys are typically 512- to 2048-bits in length. The keys are mathematically related, but so that—even with full knowledge of the public key, plaintext, and ciphertext—an adversary can not derive the associated private key. The PCI Cryptographic Coprocessor generates RSA keys with lengths of 512- to 2048-bits.

RSA Key Distribution

Public key algorithms like RSA are computationally intensive, and the bit length of an RSA-ciphered data item is limited to the RSA key length. In practice, messages are not usually encrypted with a public key [1]. Instead, RSA keys are used to securely transmit symmetric DES keys. For example:

1. The sender enciphers a message using a randomly-generated DES key.
2. The sender enciphers the DES key using the recipient's public key.
3. The sender transmits the DES-encrypted message and the RSA-encrypted DES key to the receiver.
4. The recipient uses his or her private key to recover the DES key.
5. The recipient deciphers the message using the DES key.

This key distribution method, as illustrated in Figure 2-4, offers both the speed of DES encryption and the convenience of public-key cryptography.

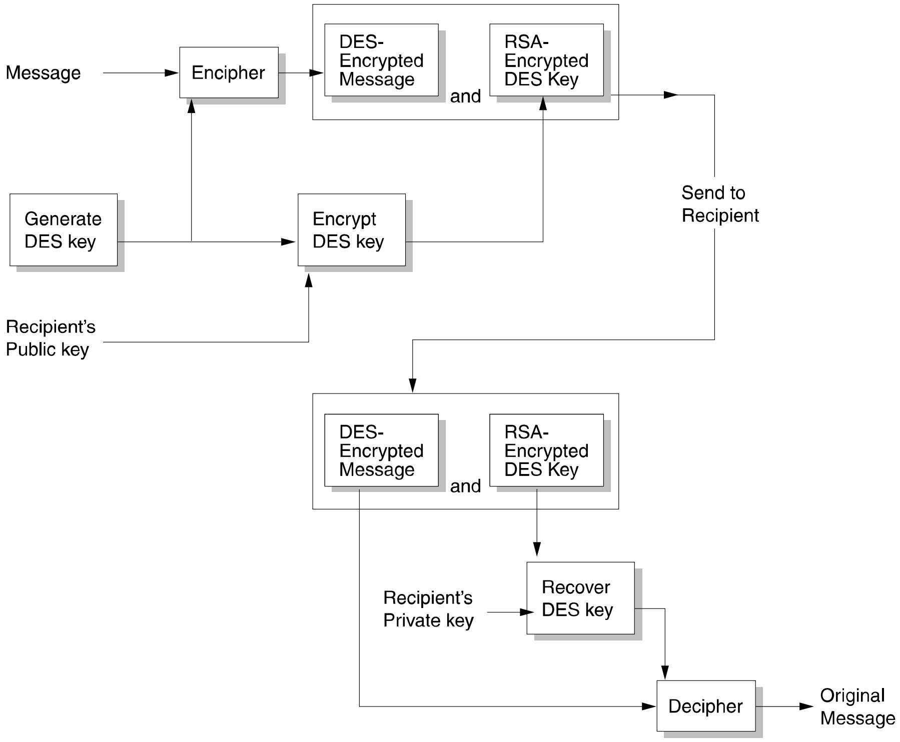

Figure 2-4. Using RSA for DES Key Exchange

[1] United States export regulations restrict distributing a product that performs general data encryption using public cryptography; this is a reason, secondary to performance, for not ciphering general data using the RSA algorithm.

The PCI Cryptographic Coprocessor allows programmers to implement secure message and key exchange. The encipher and decipher services support the DES CBC mode of data encryption, and support several methods for accommodating data not a multiple of eight bytes, including the ANSI X9.23 method. The RSA encipherment of the DES data key follows the approach documented in the PKCS #1[2] standard commonly used in Internet protocols.

Digital Signatures

Digital signatures are generated using RSA encryption and cryptographic *hashing* algorithms. With a digital signature, you can:

- Know that data has not changed since leaving the control of the sender.
- Go to court and prove that the originator of disputed data *must* have originated that data.

A *digital signature* is computed on a message (or any data object) using a private key. Since the private key is known to one entity only, no one else can generate the same digital signature on the message. This signature can be verified by any recipient possessing the originator's public key.

Digital signatures are a powerful tool for electronic commerce and other networked application systems. However, certain practical problems must be addressed:

1. Since public-key algorithms are computationally intensive, ciphering a large data object is not feasible.
2. The verifier of a digital signature must be certain that the public key is valid and belongs to the claimed sender.

The first difficulty is addressed by *hashing* data objects, and the second difficulty is addressed through the use of *certification authorities*.

Hashing

Hashing—also known as *message digesting*—is a one-way process that converts a data object to a fixed-length (128- or 160-bit) *hash value*. The hash value serves as a representation of the data, but cannot be used to recreate it. Possession of a hash value serves as proof that you generated the hash using the original data object.

In practical use, the hash value is formatted—usually with signature-method identifying codes—and the result is expanded to a bit length appropriate to the public-key algorithm and the key length currently in use.

[2] PKCS standards are documented by RSA Data Security, Inc.

A useful hashing process has the following characteristics:

- Very sensitive to any change in hashed data; hashing an altered version of the original object results in a different hash value
- Unique output: infeasible to devise another message that yields the same hash value
- Reasonably efficient, as some data objects are quite large

Finding a hashing algorithm that meets all of these criteria has been difficult. Based on research information, the PCI Cryptographic Coprocessor supports the following hashing algorithms, although others can be used:

- SHA-1 (FIPS 180-1, May 31, 1994)
- MD5 (RFC 1321, dated April 1992)

Unlike information ciphered using the DES algorithm, information processed by the RSA algorithm must have certain properties needed to block cryptographic attack, so the format of information to be ciphered with RSA has to be considered carefully. The IBM 4758 PCI Cryptographic Coprocessor supports CCA services to format the hash according to both the ISO 9796-1 and the PKCS #1 standards.

Certification Authorities

In order to use public key cryptography, you must know that:

- A public key is the correct value for the entity of interest.
- The public key is or was considered valid.

A common way to address these concerns is to distribute public keys in the form of public-key *certificates*.
A public-key certificate is a message that contains the following:

- The public key
- An identifier for the public-key owner
- The digital signature of a *certification authority (CA)*
- An expiration date
- Optional information about the key's owner, the CA, or the key

The certificate is usually hundreds of bytes long and formatted into a complicated structure, typically according to the ISO X.509 version 3 standard.

The CA ensures that the *credentials* presented by the public-key owner are appropriate, and that the public key is unique among the set of public keys certified by the authority. On this basis, the CA will create, or sign, the certificate and distribute the certificate to the public-key owner or to a directory service.

To trust a certificate, you must trust the public key you have for the CA issuing the certificate. There must be a way for users to obtain the CA's public-key certificate from a trusted source. To address this issue, the public key can be hard-coded into an application, or the certification authority certificate may be given to the user as he or she joins a consortium of users.

Given that a user trusts a CA's public key, that key can be used to validate the certificates of other certification authorities. If you do not trust the public key from the certification authority that signed the certificate, you can obtain a chain of certificates (and their corresponding public keys) to verify the previous public key until you have a certificate signed by a public key that you trust.

The Complete Digital Signature Process

After you choose a hashing algorithm and obtain a certified public key, you can establish:

Data Integrity The verified message is identical to the originally-signed message

Non-repudiation The originator must have signed the message because only the originator possesses the associated private key needed to compute the digital signature

As shown in Figure 2-5 on page 2-8, to digitally sign data (such as a file) you:

1. Hash the data file: for common hashing algorithms, the resulting hash is 128- or 160-bits long.
2. Format the hash according to the PKCS #1 or ISO 9796 standard.
3. Encrypt the hash value using your private key: since the signature is a function of the data in the signed message, the recipient cannot copy your signature.
4. Transmit the original file and the signed hash value to the recipient.
5. The recipient hashes his or her copy of the original file using the algorithm and format used in Steps 1 and 2.
6. The recipient uses your public key to decrypt the signed hash value.
7. If the decrypted hash value matches the hash value generated in Step 5, the recipient has proof that you signed the file. (The file cannot be modified, because changes to the file will alter its hash value.)

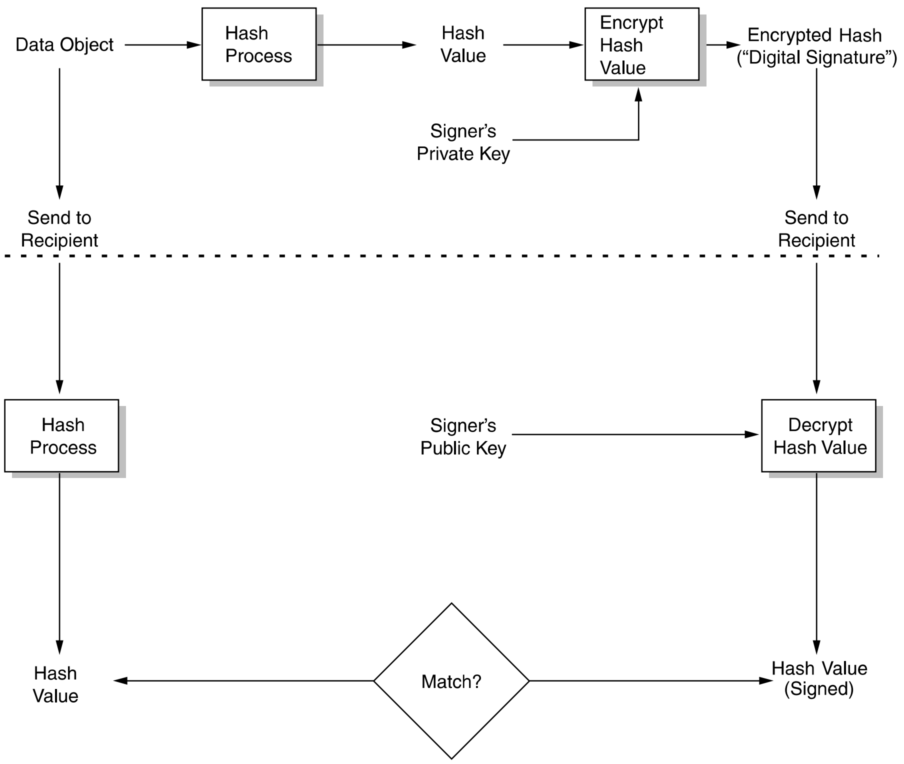

Figure 2-5. Complete Digital Signature Process

Cryptographic Key Management

Since enciphering and deciphering usually occur at different times and places, keeping the key secret is a major problem. *Key management* is the term given to the set of disciplines that addresses techniques for obtaining keys, keeping keys secret or private, and knowing that the key is valid.

Effective key management is difficult and is a major source of weakness in cryptographic systems. The IBM Common Cryptographic Architecture (CCA) includes major improvements in distributed-system key management, and is the basis for cryptographic support in the SecureWay product line. A hallmark of CCA design is the co-existence of flexibility and strict control.

This section discusses the following key management features:

- Key generation and distribution
- Role-based access control

Key Generation and Distribution

A cryptographic node must accomplish both data protection and key management. The CCA Cryptographic Coprocessor Support Program features the following key management characteristics:

- *Control vectors* and a command architecture that ensures the correct and limited use of keys in a distributed system
- Master-key protection for an unlimited number of local keys
- Dynamic master-key change capability
- Secure transport of keys between cryptographic nodes

- Ability to “lock” a key to a node
- Distribution of keys using IBM CCA and RSA techniques
- Split-knowledge, dual-control techniques for key backup and recovery
- Implementation of key-generation centers and key-translation centers
- Backup of RSA private keys

Cryptography relies on the availability of unpredictable keys, so one measure of a cryptographic implementation is its random-number generator. Most programming languages generate pseudo-random numbers, but the quality of such support varies. The PCI Cryptographic Coprocessor uses a hardware-based noise source to ensure the unpredictability of the random numbers it generates.

CCA products store a single *master key* in a register within the secure hardware. Other keys are then triple-encrypted under the master key and can be securely held outside of the hardware until they are needed. The master-key approach permits the use of an unlimited number of working keys. In order to support continuous operation, working keys are securely re-encrypted by utility programs or application programs when a master key is changed.

A data set managed by the CCA software supports the storage of keys under a *key label*; this approach is often taken for long-life keys. Keys can also be stored by application programs and passed to the cryptographic system as required; this approach is often taken for short-life keys. Regardless of a key's life cycle, you can use either approach to store it.

The CCA *control vector technology* allows you to flag a key as “not exportable.” This allows you to selectively ensure that keys can not be removed from the system even though they temporarily reside outside of the hardware.

Role-Based Access Control

Using the CCA Cryptographic Coprocessor Support Program’s role-based *access-control tables*, you can define a set of roles to correspond with the security responsibilities assigned to your users. The access privileges for each user ID on your system are determined by the role to which each is assigned. For example, you could create the following roles for a basic system:

- **Access Control Administrator**: Can create new user IDs and modify the access rights of current users.

- **Key Management Officer**: Can change the cryptographic keys for a PCI Cryptographic Coprocessor.

- **General User**: Can use cryptographic services to protect his or her work, but has no administrative capabilities.

In general, there would be few individuals assigned to the roles of Key Management Officer or Administrator; the larger population would be assigned the role of General User. This system allows you to design a standard set of user roles, rather than having to define access rights for each user.

The access-control system manages the use of sensitive commands by restricting the types of commands available to each role, and by defining specific conditions under which these commands are enabled. For example, you can restrict access to the system during certain times of the day, and restrict access on weekends and holidays.

The CCA Cryptographic Coprocessor Support Program feature includes a Node Management Utility for building your access-control system, and provides the same capabilities to your application software through the CCA API.

Additional PCI Cryptographic Coprocessor Capabilities

This section describes the following capabilities and applications of the PCI Cryptographic Coprocessor:

- Credit card transaction security
- Private key protection for certification authorities
- Data Confidentiality within Systems Network Architecture (SNA)

Credit Card Transaction Security

The PCI Cryptographic Coprocessor supports the Secure Electronic Transaction (SET) specification for protecting transactions on open networks like the Internet. Compliance with the SET specification allows widespread digital certification for merchants and consumers using credit cards. Merchants receive a public-key certificate from the credit card company, and credit card holders receive a certificate from the banks issuing their cards.

Using these certified public keys, SET credit card transactions are encrypted and digitally signed—protecting cardholders, merchants, and financial institutions from fraud and misrepresentation.

Private Key Protection for Certification Authorities

Underlying the use of public-key cryptography is the need to know, with certainty, that you have the correct, current public key for the entity with whom you intend to communicate (see “Certification Authorities” on page 2-6). The usual technique is to obtain a certificate containing the other entity's public key, issued by a trusted certification authority (CA); the CA signs the certificate using its private key.

In general, certification authorities sign a large number of certificates and the loss or compromise of the private key would require replacement of every public key certified by that CA. Therefore, the protection of a CA's private key is of paramount importance.

The PCI Cryptographic Coprocessor is ideally suited for use in a certification authority application since it furnishes:

- **Excellent protection for a private key**: When the PCI Cryptographic Coprocessor's advanced physical-security design detects physical tampering, it renders the active copy of the private key useless.
- **Authorization control based on passwords**: Use of the CA private key can be specifically restricted.
- **Backup and recovery for the private key**: You can distribute a CA key to secure backup hardware at the same or other locations through the use of CCA DES key management practices—providing backup security without compromising the security of the key. In addition, you can install a duplicate key on another coprocessor to provide increased throughput.

In general, an entity does not present itself to the CA to obtain a certificate. To satisfy this operational problem, the CA establishes a network of trusted *local registration authorities* (LRAs). After being satisfied that the entity should have a certificate, an LRA will transmit a certificate request in the form of a *credentials* message to the CA. These messages are generally not confidential, but require data-integrity protection. IBM Net Registry and World Registry products and services can use the PCI Cryptographic Coprocessor to protect a private key.

Data Confidentiality within Systems Network Architecture

The SNA communications architecture defines how data can be enciphered between logical units in a network. This architecture is implemented by the following products:

- IBM Communications Manager/2
- IBM Communications Server/2
- IBM ACF/VTAM
- IBM LAN Distributed Platform
- IBM LAN Distributed Platform/2

Each of these products calls cryptographic facilities in its environment to control cryptographic-key management and DES-based data enciphering and deciphering. The CS/2, CM/2, and LANDP/2 products can use the PCI Cryptographic Coprocessor for the services required.

The ACF/VTAM product can use any of the following products:

- The IBM 4753 Network Security Processor and its MVS support program
- The IBM System/390 Integrated Cryptographic Coprocessor
- The Integrated Cryptographic Feature on high-end Enterprise System/9000* processors
- The Programmed Cryptographic Facility (PCF) product

Governmental Regulations on Cryptography

In the past, the use of strong cryptography was largely confined to the military, governmental security agencies, and the financial services infrastructure. Recently, implementations of strong cryptographic techniques have become widely available. In the interest of national security and law enforcement, almost all governments exercise some form of control over the distribution of these implementations, whether based on software or hardware.

In recognition of the fact that our “wired world” depends on legitimate data protection, governments are re-evaluating their position. It is reasonable to expect changes in the regulations pertaining to cryptographic product manufacture and distribution.

With respect to the PCI Cryptographic Coprocessor: In addition to the prohibition against delivering any cryptographic product to customers in a few selected countries, the following items are subject to governmental regulation:

- The strength of the process used to encrypt *general data*
- The key length used in public-key cryptography when distributing keys that can be used to protect general data

For the most part, governments are concerned with the concealment of information, but not with technologies used to confirm identity or validate data. There is, therefore, little restriction on cryptographic implementations related to digital signatures and other data-integrity controls. Likewise, there is little restriction on using *triple DES* to encipher keys.

The PCI Cryptographic Coprocessor generally addresses three export-control-defined customer sets:

1. USA and Canadian customers, and USA companies and subsidiaries outside of the USA and Canada:

   Usually, no restrictions.

2. Financial institutions outside of the USA and Canada:

   In general, RSA key bit-lengths are limited to 512 for enciphering symmetric keys, and 1024 for SET key exchange. Key lengths used for digital signatures are not restricted.

3. Other customers outside of the USA and Canada:

   In addition to restrictions above, services that can obscure information in data may not use DES keys with an effective strength in excess of 40 bits[3].

To address the need for a good approach to data encryption while building on the existing infrastructure and knowledge of the strength of DES, IBM defined and implemented the *Commercial Data Masking Facility* (CDMF) algorithm. This algorithm operates in the same way as DES CBC, except that the cryptographic key is weakened within the data ciphering and deciphering services to an effective strength of 40 bits. Key-management services are not affected.

Limitations on key distribution capabilities and on the type of data encryption support provided are specified at the time products are ordered from IBM; IBM applies for appropriate export and import licenses.

[3] Effective January 1997, new rules by the US government have changed this situation. Regulations by other governments are also likely to change. See the IBM SecureWay web page at http://www.ibm.com/security for the latest information.

Chapter 3. PCI Cryptographic Coprocessor Features

The PCI Cryptographic Coprocessor offers a broad range of cryptographic services based on the DES and RSA algorithms. This chapter includes descriptions of, operating environment requirements, and ordering information for:

- The PCI Cryptographic Coprocessor hardware.
- The CCA Cryptographic Coprocessor Support Program feature.

For performance specifications—such as throughput rates—visit the product Web site at http://www.ibm.com/security/cryptocards.

IBM 4758 PCI Cryptographic Coprocessor

When PCs and IBM RS/6000 computers require DES and RSA cryptographic processing, the IBM 4758 PCI Cryptographic Coprocessor—a two-thirds length adapter card—provides specialized, high-performance hardware within a secure enclosure. With the optional CCA Cryptographic Coprocessor Support Program feature, the coprocessor functions in the AIX, OS/2, and Windows NT environments.

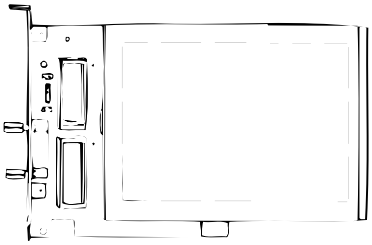

Figure 3-1. The IBM 4758 PCI Cryptographic Coprocessor

Overview

The IBM 4758 PCI Cryptographic Coprocessor offers the following features:

- **Extensive DES and RSA Cryptographic Functions** to support the cryptographic data security requirements of the financial industry, the Internet, and other electronic environments. Cryptographic capabilities are accessed through the IBM Common Cryptographic Architecture (CCA). The CCA control-vector-based key separation techniques provide logical security to augment the physical security of the hardware.
- **High-security Design** conforms to FIPS 140-1 level 4 requirements for resistance to physical penetration, temperature, radiation, and voltage attack.
- **Hardware for Security and Performance** when your requirements demand strong security. In distributed processing systems and an era of virus infection, secured hardware keeps your cryptographic keys secure and ensures that only authorized individuals can access them. Dedicated processing yields unmatched performance.

- **Cryptographic Data Protection Capabilities** for data encryption, digital signatures, hashing, message authentication, and support for the Secure Electronic Transaction (SET) standard. The card meets the cryptographic requirements of industry standards such as PKCS #1.0, PKCS #1.1, PKCS #1.2[1], SHA-1, ISO 9796, and ANSI X9.9 and X9.23.
- **International Export Options** to meet normal and exceptional export conditions associated with the delivery of strong cryptography equipment. The Commercial Data Masking Facility (CDMF) algorithm addresses restrictive export situations.
- **Custom Cryptographic Applications** developed to your specifications by IBM and securely loaded on the PCI Cryptographic Coprocessor can address evolving cryptographic processes without compromising the integrity of the cryptographic infrastructure.

The following section describes some of these features in greater detail.

Description

The PCI Cryptographic Coprocessor operates in an industry-standard PCI bus slot. Information is pipelined through two DMA ports internally connected to FIFO registers, and the internal DES processor is connected to the FIFO buffers. An internal 1024-bit-wide large-integer processor performs 1024-bit RSA private key operations. The Coprocessor Support Program feature uses this processor to support 2048-bit keys.

Also within the secure computing module are a time-of-day clock, a hardware-based random-number generator, and an RS-232 serial port. The subsystem uses an Intel-compatible 80486 processor with 4MB of Random Access Memory (RAM) and 2MB of flash memory. Separate battery-powered RAM stores critical security parameters that are erased when tampering is detected.

The PCI Cryptographic Coprocessor is supplied with a bootstrap load routine that authenticates software before it is stored for internal use. When shipped from the factory, its memory holds a certified device key that will digitally sign test messages affirming that the card is genuine and that no tampering has occurred.

Cryptographic processing and the storage of primary cryptographic keys occurs within a secure module mounted on the PCI Cryptographic Coprocessor.
This module contains:

- Mechanisms that meet the rigorous FIPS PUB 140-1 level 4 standard for detecting physical penetration, voltage attacks, temperature attacks, and radiation attacks. If activated, these mechanisms zero the keys required to access security-relevant data stored in the module.
- A general-purpose processor and memory.
- Dedicated hardware to implement the DES and RSA algorithms.
- An electronically-protected external memory bus.
- RS-232 electronics.
- Protected programming storage for additional cryptographic functions and subroutines developed by IBM or other vendors.

[1] PKCS standards and the MD5 hashing algorithm (see RFC 1321) are developments of RSA Data Security, Incorporated.

Requirements and Specifications

Hardware Requirements:

A personal computer, workstation, server, communication device or RS/6000 with an available PCI (Version 2.1) bus slot.

Physical Dimensions:

The PCI Cryptographic Coprocessor is a two-thirds length PCI card,
4.2 in. by 6.9 in. (106mm by 175mm).

Power Requirements:

Power consumption in active mode: 6.3 watts
Power consumption in energy-saving mode: 6.0 watts
Power dissipation: 6.3 watts (typical)
Voltage: +5.0 VDC, plus-or-minus 5 percent

Operating Environment:

Temperature: 10 to 40 degrees C (50 to 104 degrees F)
Relative Humidity: 8 to 80%

Shipping and Storage Environment:

Temperature: -15 to 60 degrees C (5 to 140 degrees F)

The PCI Cryptographic Coprocessor destroys its factory certification if allowed to cool below -15 degrees C (5 degrees F).

CCA Cryptographic Coprocessor Support Program Feature

This section describes the CCA Cryptographic Coprocessor Support Program feature.

Overview

The CCA Cryptographic Coprocessor Support Program includes device drivers, utilities, and the IBM Common Cryptographic Architecture (CCA) interface software to use the IBM 4758 PCI Cryptographic Coprocessor on RS/6000 machines running the AIX operating system, or on personal computers running the OS/2 Warp or Windows NT operating systems. A utility loads digitally-signed application software that implements CCA services within the card's tamper-resistant module.

The CCA Application Programming Interface (API) enables your applications to access the services furnished by the CCA Cryptographic Coprocessor Support Program software feature.

The CCA offers the following features:

- **Sophisticated Key Management Techniques** that feature control-vector-based key separation to assure uniform and controlled capabilities in distributed systems, as well as support of industry-standard RSA key distribution techniques.

- **Standards-Based Cryptographic Services** such as DES encryption and message authentication, digital signatures and hashing, and random-number generation.
- **Application Development in a Common Manner** for AIX, OS/2, and NT systems. Calls to the CCA application program interface (API) can be made from most high-level programming languages, with explicit support for the C programming language. The API is the same in each operating system environment, and is consistent with the CCA API offered with Transaction Security System products, the S/390 integrated cryptographic feature, and the AS/400 cryptographic feature.
- **Access Control Tables** that define what commands can be performed and under what conditions. You can ensure that split-knowledge, dual-control procedures are followed to securely activate the cryptographic processor and limit the use of sensitive commands to selected user-roles.

  You can set the value of these tables using utilities provided with the CCA Cryptographic Coprocessor Support Program feature, or by calling CCA services from your application programs.
- **Role-Based Access Control Tables** can predefine the level of access given to a category of users. For example, you create access-control tables for roles like “key management officer” and “general user,” and then you define the operating privileges of your users by assigning their user accounts to the appropriate role. The roles simplify and standardize the administration of your cryptographic system.

Description

The CCA Cryptographic Coprocessor Support Program exploits the full capabilities of the PCI Cryptographic Coprocessor hardware to furnish services including:

- Secure Electronic Transaction (SET) support
- DES and CDMF encryption and decryption with support for CBC and ANSI X9.23 last-block processing
- ANSI X9.9-1 MAC generation and verification
- DES key management functions that permit secure generation of keys under the coprocessor's master key
- Secure import and export of DES keys using DES key-encrypting keys
- Secure import and export of DES data keys using RSA with PKCS #1.2 formatting
- Hashing using the SHA-1 and MD5 algorithms
- RSA signature generation and verification, with signatures formatted according to ISO 9796, PKCS #1.0, and PKCS #1.1 rules
- Secure generation of RSA key pairs with modulus lengths as long as 2048 bits

The supported CCA services generally conform to, and are a subset of, services offered with other IBM CCA products. Support for the SET standard is a new CCA feature. For a complete description of the services offered by CCA, refer to the *IBM 4758 CCA Basic Services Reference and Guide*, SC31-8609. For information regarding the availability of this publication, and the latest performance specifications for the product, visit the product Web site at http://www.ibm.com/security/cryptocards.

[2] PKCS standards and the MD5 hashing algorithm (see RFC 1321) are developments of RSA Data Security, Incorporated.

Export-Controlled Cryptographic Function satisfies governmental export and import control requirements; the support software uses a *function-control vector* to specify the cryptographic capabilities of the card. Certain functions are not available in geographic locations where their use is restricted by law. For more information, see "Governmental Regulations on Cryptography" on page 2-11.

Requirements and Specifications

Software Requirements:

For a PCI Cryptographic Coprocessor installed in a PC:
IBM OS/2 Warp, Version 3 or later, or
Windows NT, Version 4.0 or later.

For a card installed in an RS/6000 machine:
AIX Version 4.1, or later.

For all installations:
Java** 1.1.1 converter.

Ordering Information

This section explains how to order the PCI Cryptographic Coprocessor, and how to select the appropriate CCA Cryptographic Coprocessor Support Program feature code.

Before You Order

1. Choose the PCI Cryptographic Coprocessor hardware:

Description	Machine	Model	Feature
PCI Cryptographic Coprocessor	4758	001	
Replacement Battery Kit			1008

The optional battery kit contains three batteries: two to replace the original batteries, and a third to be used as a temporary power source during the replacement process.

2. Choose the CCA Cryptographic Coprocessor Support Program feature code for the operating system on which you will use the software:

Support Program	Feature
for OS/2	4372
for AIX	4374
for NT	4376

Note: The PCI Cryptographic Coprocessor does not function without support software.

3. As permitted by import or export regulations, choose a *function-control vector*. The vector specifies the level of cryptographic support offered by the software, which varies according to government regulations. An IBM export regulation coordinator can help you determine the limitations that apply to you, and help you apply for an exception to the standard practice.

Function-Control Vector	**Feature**
CCA for DES, 1024-bit RSA symmetric-key management, and SET	5200
CCA for DES, 512-bit RSA symmetric-key management, and SET	5201
CCA for CDMF, 512-bit RSA symmetric-key management, and SET	5202

Ordering the Products

1. Visit the product Web site at http://www.ibm.com/security/cryptocards for the latest information about the product offerings.

2. Contact your local IBM representative or your IBM Business Partner. OEM customers in the United States should call 1-800-IBMS-OEM (1-800-426-7636).

   Your IBM representative will provide you with a customer number and an order number; these numbers are later used to decrypt and validate the CCA Cryptographic Coprocessor Support Program.

3. Download the CCA Cryptographic Coprocessor Support Program from the product Web site at http://www.ibm.com/security/cryptocards. Follow instructions on the Web page. Selected portions of the software are encrypted, so you cannot use the software yet.

4. Obtain a verification number from an IBM World-Wide Key Center. Your verification number is based on your customer number and the validated order number you received in Step 2. As you request, the Key Center delivers the verification number by phone, fax, or e-mail.

   When the product becomes available, the phone numbers for the Key Centers will be listed on the product Web site.

5. Download the software from the product Web site. The resulting file is a program you can execute.

6. Execute the downloaded program.

7. As prompted by the program, enter your IBM Customer Number and Verification Number. The program decrypts, decompresses, and installs the software you ordered.

8. On certain systems, such as AIX, run the system installation program. For details, refer to the *IBM 4758 CCA Services Installation and I/O Guide*, SC31-8610.

9. Using the instructions in the *IBM 4758 CCA Services Installation and I/O Guide*, load the function-control vector and coprocessor portions of the software into the PCI Cryptographic Coprocessor.

List of Abbreviations

ANSI	American National Standards Institute
AIX	Advanced Interactive eXecutive operating system
API	Application Programming Interface
ASCII	American National Standard Code for Information Interchange
AS/400	Application System/400
C	Celsius
CA	Certification Authority
CBC	Cipher-Block Chaining
CCA	Common Cryptographic Architecture
CDMF	Commercial Data Masking Facility
CICS	Customer Information Control System
CKDS	Cryptographic-Key Data Set
cm	centimeter
CV	Control Vector
DEA	Data Encryption Algorithm
DES	Data Encryption Standard
DMA	Direct Memory Access
DOS	Disk Operating System
ECB	Electronic Code Book
EEPROM	Electrically Erasable, Programmable Read-Only Memory
F	Fahrenheit
FCC	Federal Communications Commission
FIPS	Federal Information Processing Standard
IBM	International Business Machines
ICRF	Integrated Cryptographic Facility
ICSF	Integrated Cryptographic Service Facility
ICSF/MVS	Integrated Cryptographic Service Facility/Multiple Virtual Storage
in	inch
I/O	Input/Output
IPL	Initial Program Load
ISO	International Standards Organization
LU	Logical Unit
MB	MegaByte
MAC	Message Authentication Code
MBps	MegaBytes per second
MD5	Message Digest 5 hashing algorithm
MDC	Modification Detection Code
mm	millimeter
MVS	Multiple Virtual Storage
NT	New Technology
OEM	Original Equipment Manufacturer
OS/2	Operating System/2
OS/400	Operating System/400
PC	Personal Computer
PCF	Programmed Cryptographic Facility
PCI	Peripheral Component Interconnect
PIN	Personal Identification Number
PKA	Public Key Algorithm
PKCS	Public-Key Cryptography Standard
POS	Point Of Sale
PROM	Programmable Read-Only Memory
RAM	Random Access Memory
ROM	Read-Only Memory
RSA	Rivest, Shamir, and Adleman
SAA	Systems Application Architecture
SHA	Secure Hashing Algorithm
SNA	Systems Network Architecture
V	volt
VDC	Volts Direct Current

Glossary

This glossary includes some terms and definitions from the *IBM Dictionary of Computing*, New York: McGraw Hill, 1994. This glossary also includes some terms and definitions from:

- The *American National Standard Dictionary for Information Systems*, ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies may be purchased from the American National Standards Institute, 11 West 42 Street, New York, New York 10036. Definitions are identified by the symbol (A) after the definition.
- The *Information Technology Vocabulary*, developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1). Definitions of published parts of this vocabulary are identified by the symbol (I) after the definition; definitions taken from draft international standards, committee drafts, and working papers being developed by ISO/IEC JTC1/SC1 are identified by the symbol (T) after the definition, indicating that final agreement has not yet been reached among the participating National Bodies of SC1.

A

access. A specific type of interaction between a subject and an object that results in the flow of information from one to the other.

access control. Ensuring that the resources of a computer system can be accessed only by authorized users in authorized ways.

access method. A technique for moving data between main storage and input/output devices.

Advanced Communications Function for the Virtual Telecommunications Access Method (ACF/VTAM) program. An IBM-licensed program that controls communication and the flow of data in a SNA network.

Advanced Interactive eXecutive (AIX) operating system. IBM's implementation of the UNIX** operating system.

American National Standard Code for Information Interchange (ASCII). The standard code, using a coded character set consisting of 7-bit characters (8 bits including parity check), that is used for information interchange among data processing systems, data communication systems, and associated equipment. The ASCII set consists of control characters and graphic characters. (A)

American National Standards Institute (ANSI). An organization consisting of producers, consumers, and general interest groups that establishes the procedures by which accredited organizations create and maintain voluntary industry standards in the United States. (A)

Application System/400 (AS/400) system. One of a family of general purpose midrange systems with a single operating system, Operating System/400, that provides application portability across all models.

authentication. (1) A process used to verify the integrity of transmitted data, especially a message. (T) (2) In computer security, a process used to verify the user of an information system or protected resource.

authorization. (1) The right granted to a user to communicate with or make use of a computer system. (T) (2) The process of granting a user either complete or restricted access to an object, resource, or function.

authorize. To permit or give authority to a user to communicate with or make use of an object, resource, or function.

Authorized Program Facility (APF). A facility that permits identification of programs authorized to use restricted functions.

B

bus. In a processor, a physical facility along which data is transferred.

C

card. (1) An electronic circuit board that is plugged into a slot in a system unit. (2) A plug-in circuit assembly.

CDMF algorithm. An alternate algorithm for data confidentiality applications, based on the DES algorithm and possessing 40-bit key strength.

ciphertext. (1) Text that results from the encipherment of plaintext. (2) See also *plaintext*.

Cipher Block Chaining (CBC). A mode of operation that cryptographically connects one block of ciphertext to the next plaintext block.

cleartext. (1) Text that has not been altered by a cryptographic process. (2) Synonym for *plaintext*. (3) See also *ciphertext*.

Common Cryptographic Architecture (CCA) API. The programming interface that is described in the *IBM 4758 CCA Basic Services Reference and Guide*.

control vector. In the CCA, a 16-byte string that is exclusive-ORd with a master key or a key-encrypting key to create another key that is used to encipher and decipher data or data keys. A control vector determines the type of key and the restrictions on its use.

coprocessor. (1) A supplementary processor that performs operations in conjunction with another processor. (2) A microprocessor on an expansion card that extends the address range of the processor in the host system or adds specialized instructions to handle a particular category of operations; for example, an I/O coprocessor, math coprocessor, or networking coprocessor.

cryptographic coprocessor (4758). An expansion card that provides a comprehensive set of cryptographic functions for a workstation.

Cryptographic Key Data Set (CKDS). A data set that contains the encrypting keys used by an installation.

cryptography. The transformation of data to conceal its meaning.

D

data-encrypting key. (1) A key used to encipher, decipher, or authenticate data. (2) Contrast with *key-encrypting key*.

Data Encryption Algorithm (DEA). A 64-bit block cipher that uses a 64-bit key, of which 56 bits are used to control the cryptographic process and 8 bits are used for parity checking.

Data Encryption Standard (DES). The National Institute of Standards and Technology (NIST) Data Encryption Standard, adopted by the U.S. government as Federal Information Processing Standard (FIPS) Publication 46 which allows only hardware implementations of the data encryption algorithm.

decipher. (1) To convert enciphered data into clear data. (2) Synonym for *decrypt*. (3) Contrast with *encipher*.

decrypt. (1) Synonym for *decipher*. Contrast with *encrypt*.

driver. A program that contains the code needed to attach and use a device.

E

Electronic CodeBook (ECB) Operation. A mode of operation used with block-cipher cryptographic algorithms in which plaintext or ciphertext is placed in the input to the algorithm and the result is contained in the output of the algorithm.

encipher. (1) To scramble data or to convert data to a secret code that masks the meaning of the data. (2) Synonym for *encrypt*. (3) Contrast with *decipher*.

enciphered data. (1) Data whose meaning is concealed from unauthorized users or observers. (2) See also *ciphertext.*

encrypt. (1) To convert cleartext into ciphertext. (2) Synonym for *encipher*. (T) (3) Contrast with *decrypt*.

Erasable Programmable Read-Only Memory (EPROM). A PROM that can be erased by a special process and then reused. (T)

EXPORTER key. (1) In the CCA, a type of DES key-encrypting key that can encipher a key at a sending node. (2) Contrast with *IMPORTER key*.

F

feature. A part of an IBM product that can be ordered separately.

Federal Communications Commission (FCC). A board of commissioners, appointed by the President under the Communications Act of 1934, having the power to regulate all interstate and foreign communications by wire and radio originating in the United States.

Federal Information Processing Standard (FIPS). A standard that is published by the US National Institute of Science and Technology.

H

hertz (Hz). A unit of frequency equal to one cycle per second. **Note:** In the United States, line frequency is 60 Hz or a change in voltage polarity 120 times per second; in Europe, line frequency is 50 Hz or a change in voltage polarity 100 times per second.

I

IMPORTER key. (1) In the CCA, a type of DES key-encrypting key that can decipher a key at a receiving node. (2) Contrast with *EXPORTER key*.

Initial Program Load (IPL). (1) The initialization procedure that causes an operating system to commence operation. (2) The process by which a configuration image is loaded into storage at the beginning of a work day or after a system malfunction. (3) The process of loading system programs and preparing a system to run jobs.

Integrated Cryptographic Service Facility (ICSF). An IBM-licensed program that supports the cryptographic hardware feature for the high-end System/390 processor in an MVS environment.

interface. (1) A shared boundary between two functional units, defined by functional characteristics, signal characteristics, or other characteristics, as appropriate. The concept includes the specification of the connection of two devices having different functions. (T) (2) Hardware, software, or both, that links systems, programs, and devices.

International Organization for Standardization (ISO). An organization of national standards bodies established to promote the development of standards to facilitate the international exchange of goods and services, and to develop cooperation in intellectual, scientific, technological, and economic activity.

J

jumper. A wire that joins two unconnected circuits on a printed circuit board.

K

key. In computer security, a sequence of symbols used with an algorithm to encrypt or decrypt data.

Key-Encrypting Key (KEK). (1) A key used for the encryption and decryption of other keys. (2) Contrast with *data-encrypting key*.

key storage. In CCA products, a data file that contains cryptographic keys.

key token. In CCA products, data structure that can contain a cryptographic key, a control vector, and other information related to the key.

L

LAN/DP. An IBM-licensed program product, LAN Distributed Processing.

M

master key. In computer security, the top-level key in a hierarchy of key-encrypting keys.

MegaByte (MB). 1 048 576 bytes.

Message Authentication Code (MAC). In computer security, (1) a number or value derived by processing data with an authentication algorithm, (2) the cryptographic result of block-cipher operations on text or data using a Cipher Block Chaining (CBC) mode of operation, (3) a digital signature code.

Multiple Virtual Storage/Extended Architecture (MVS/XA) product. Multiple Virtual Storage/Extended Architecture product, consisting of MVS/System Product Version 2 and the MVS/XA Data Facility Product, operating on a System/370 processor in the System/370 extended architecture mode. The MVS/XA product allows virtual-storage addressing up to two gigabytes.

multi-user environment. A computer system that supports terminals and keyboards for more than one user at the same time.

N

National Institute of Science and Technology (NIST). Current name for the US National Bureau of Standards.

O

Operating System/2 (OS/2). An operating system for personal computers.

Operating System/400 (OS/400). An operating system for the IBM Application System/400 computers.

P

password. In computer security, a string of characters known to the computer system and a user; the user must specify it to gain full or limited access to a system and to the data stored therein.

plaintext. (1) Data that has not been altered by a cryptographic process. (2) Synonym for *cleartext.* (3) See also *ciphertext.*

Power-On Self Test (POST). A series of diagnostic tests that are run automatically by a device when the power is turned on.

private key. (1) In computer security, a key that is known only to the owner and used with a public-key algorithm to decrypt data. The data is encrypted using the related public key. (2) Contrast with *public key*. (3) See also *public-key algorithm*.

procedure call. In programming languages, a language construct for invoking execution of a procedure. (I) A procedure call usually includes an entry name and possible parameters.

profile. Data that describes the significant characteristics of a user, a group of users, or one-or-more computer resources.

Programmed Cryptographic Facility (PCF). An IBM licensed program (program number 5740-XY5) that provides facilities for enciphering and deciphering data and for creating, maintaining, and managing cryptographic keys.

PRPQ. Programming Request for Price Quotation.

public key. (1) In computer security, a key that is widely known and used with a public-key algorithm to encrypt data. The encrypted data can be decrypted only with the related private key. (2) Contrast with *private key*. (3) See also *Public-Key Algorithm*.

Public-Key Algorithm (PKA). (1) In computer security, an asymmetric cryptographic process that uses a public key to encrypt data and a related private key to decrypt data. (2) Contrast with *Data Encryption Algorithm* and *Data Encryption Standard algorithm*. (3) See also *Rivest-Shamir-Adleman algorithm*.

R

Random Access Memory (RAM). (1) A storage device into which data are entered and from which data are retrieved in a non-sequential manner. (2) See also *direct access storage*.

Read-Only Memory (ROM). Memory in which stored data cannot be modified by the user except under special conditions.

Reduced Instruction-Set Computer (RISC). A computer that uses a small, simplified set of frequently-used instructions for rapid processing.

Rivest-Shamir-Adleman (RSA) algorithm. A process for public-key cryptography that was developed by R. Rivest, A. Shamir, and L. Adleman.

RS-232. A specification that defines the interface between data terminal equipment and data circuit-terminating equipment, using serial binary data interchange.

RS-232C. A standard that defines the specific physical, electronic, and functional characteristics of an interface line that uses a 25-pin connector to connect a workstation to a communication device.

RSA algorithm. Rivest-Shamir-Adleman encryption algorithm.

S

security. The protection of data, system operations, and devices from accidental or intentional ruin, damage, or exposure.

Session-Level Encryption (SLE). A Systems Network Architecture (SNA) protocol that provides a method for establishing a session with a unique key for that session. This protocol establishes a cryptographic key and rules for deciphering and enciphering information in a session.

system administrator. The person at a computer installation who designs, controls, and manages the use of the computer system.

Systems Network Architecture (SNA). The description of the logical structure, formats, protocols, and operational sequences for transmitting information units through, and controlling the configuration and operation of, networks. **Note:** The layered structure of SNA allows the ultimate origins and destinations of information, that is, the end users, to be independent of and unaffected by the specific SNA network services and facilities used for information exchange.

T

throughput. (1) A measure of the amount of work performed by a computer system over a given period of time; for example, number of jobs per day. (A) (I) (2) A measure of the amount of information transmitted over a network in a given period of time; for example, a network's data-transfer-rate is usually measured in bits per second.

token. A string of characters that is treated as a single entity.

U

utility program. A computer program in general support of computer processes. (T)

V

verb. A function that has an entry-point name and a fixed-length parameter list. The procedure call for a verb uses the standard syntax of a programming language.

W

workstation. A terminal or microcomputer, usually one that is connected to a mainframe or to a network, at which a user can perform applications.

Numerics

4753. IBM 4753 Network Security Processor.

4754. IBM 4754 Security Interface Unit.

4755. IBM 4755 Cryptographic Adapter.

4758. IBM 4758 PCI Cryptographic Coprocessor.

Index

Numerics

4758
  See IBM 4758 PCI Cryptographic Coprocessor

A

access-control tables 2-9, 3-4
Adleman
  See RSA-algorithm development
AIX operating system
  CCA Cryptographic Coprocessor Support Program 3-3
algorithms
  CDMF 2-12, 3-5
  DES 2-2
  MD5 2-6
  RSA 2-3
  SHA-1 2-6
American National Standards Institute (ANSI)
  X.509 standard 2-6
  X9.23 standard 2-2, 3-2, 3-4
  X9.9 standard 2-2, 3-2, 3-4
ANSI
  See American National Standards Institute
applications
  data confidentiality in a SNA network 2-11
  protecting a CA private key 2-10
asymmetric key 2-3

C

CA
  See certification authority
CBC
  See Cipher-Block Chaining
CCA
  See Common Cryptographic Architecture
CDMF
  See Commercial Data Masking Facility
certificate, public-key 2-6
certification authority (CA) 2-6, 2-10
  certificate 2-6
  credentials 2-6
Cipher-Block Chaining 2-2, 3-4
ciphertext 2-1, 2-3
Commercial Data Masking Facility (CDMF) 2-12, 3-5
Common Cryptographic Architecture (CCA)
  access-control tables 2-9, 3-4
  key management 2-8
  support program 3-3
control vector 2-3, 2-8
credentials 2-6
credit card security 2-10
cryptography
  CDMF algorithm 2-12, 3-5
  data confidentiality 2-1
  data integrity 2-1
  DES algorithm 2-2
  hashing algorithms 2-6
  non-repudiation 2-1
  RSA algorithm 2-3

D

data
  confidentiality 1-2, 2-1
    ANSI X9.23 standard 2-2
    CDMF algorithm 2-12, 3-5
    product application 2-11
  hashing 2-5
  integrity 2-1, 2-2
    digital signature 2-5
    ISO 8730 standard 2-2
    MAC 2-2
    X9.9 ANSI standard 2-2
  non-repudiation 2-1, 2-5
Data Encryption Algorithm (DEA) 2-2
Data Encryption Standard (DES)
  algorithm 2-2
  ANSI X9.23 standard 3-4
DEA
  See Data Encryption Algorithm
DES
  See Data Encryption Standard
digital certificates 2-6
digital signature 1-2, 2-5
  hash 2-5
  ISO 9796 standard 2-6
  message digesting 2-5
  process 2-7

E

export/import controls 2-11, 3-5

F

FIPS 140-1 level 4 3-2
FIPS 180-1 2-6

G

Governmental regulations, on cryptographic products 2-11, 3-5

H

hashing algorithms
  MD5 2-6
  SHA-1 2-6

I

IBM 4758 PCI Cryptographic Coprocessor
  dimensions 3-3
  environment
    operating 3-3
    shipping 3-3
    storage 3-3
  international options 3-2
  ordering 3-5
  product information 3-1
  requirements
    hardware 3-3
    power 3-3
    software 3-5
import/export controls 2-11
International Organization for Standardization (ISO)
  8730 standard 2-2
  9796 standard 2-6, 3-4
ISO
  See International Organization for Standardization

K

key
  asymmetric 2-3
  distribution 2-8
  distribution using RSA 2-4
  generation 2-8
  label 2-9
  management 2-8
  master 2-9
  private 2-3
  public 2-3
  recovery 1-5
  secret 2-2
  symmetric 2-2

L

LAN Distributed Platform (LANDP, LANDP/2)
  data confidentiality in a SNA network 2-11
LANDP and LANDP/2
  See LAN Distributed Platform (LANDP, LANDP/2)
large-integer processor 3-2
Local Registration Authorities (LRA) 2-10
LRA
  See Local Registration Authorities

M

MAC
  See Message Authentication Code
master key 2-9
MD5
  See hashing algorithms
message authentication 2-2
Message Authentication Code (MAC) 1-2, 2-2, 3-4
message digest
  MD5 algorithm 2-6
  SHA-1 algorithm 2-6

N

non-repudiation, data 2-1, 2-5

O

Operating System/2 (OS/2)
  CCA Cryptographic Coprocessor Support Program 3-3
OS/2
  See Operating System/2

P

PCF
  See Programmed Cryptographic Facility
PKCS
  See Public-Key Cryptography Standard
plaintext 2-1, 2-3
private key 2-3, 2-10
product applications
  data confidentiality in a SNA network 2-11
  protecting a CA private key 2-10
Programmed Cryptographic Facility (PCF) 2-11
pseudo-random numbers 2-9
public key 2-3
public-key certificate 2-6
Public-Key Cryptography Standard (PKCS)
  PKCS #1 2-5, 2-6, 3-2
  PKCS #1.0 3-4
  PKCS #1.1 3-4
  PKCS #1.2 3-4

R

random-number generator 2-9, 3-2
  See also pseudo-random numbers
registration authorities 2-10
Rivest
  See RSA-algorithm development
role-based access control 2-9, 3-4
roles 2-9
RSA public-key cryptography, algorithm
  digital signature 2-5

RSA public-key cryptography, algorithm *(continued)*
  key distribution using RSA 2-4
  private key 2-3
  public key 2-3
RSA-algorithm development 2-3

S

secret key 2-2
Secure Electronic Transaction 2-10
Secure Hash Algorithm (SHA)
  FIPS 180-1 2-6
SecureWay products 1-3
SET
  See Secure Electronic Transaction
SHA
  See Secure Hash Algorithm
SHA-1
  See hashing algorithms
Shamir
  See RSA-algorithm development
SNA data confidentiality 2-11
symmetric key 2-2

T

tamper-detection 3-2

W

Web site, product 3-1, 3-6
Windows NT operating system
  CCA Cryptographic Coprocessor Support Program 3-3

X

X.509 ANSI standard 2-6
X9.23 ANSI standard 2-2, 3-2, 3-4
X9.9 ANSI standard 2-2, 3-2, 3-4

Communicating Your Comments to IBM

IBM SecureWay Cryptographic Products
IBM 4758 PCI Cryptographic Coprocessor

General Information Manual

Publication No. GC31-8608-00

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

If you are mailing a readers' comment form (RCF) from a country other than the United States, you can give the RCF to the local IBM branch office or IBM representative for postage-paid mailing.

- If you prefer to send comments by mail, use the RCF at the back of this book.
- If you prefer to send comments by FAX, use this number:

  United States and Canada: **1-800-227-5088**
- If you prefer to send comments electronically, use this network ID:
  - IBM Mail Exchange: **USIB2HPD at IBMMAIL**
  - IBMLink: **CIBMORCF at RALVM13**
  - Internet: **USIB2HPD@VNET.IBM.COM**

Make sure to include the following in your note:

- Title and publication number of this book
- Page number or topic to which your comment applies.

Help us help you!

IBM SecureWay Cryptographic Products
IBM 4758 PCI Cryptographic Coprocessor

General Information Manual

Publication No. GC31-8608-00

We hope you find this publication useful, readable and technically accurate, but only you can tell us! Your comments and suggestions will help us improve our technical publications. Please take a few minutes to let us know what you think by completing this form.

Overall, how satisfied are you with the information in this book?	Satisfied	Dissatisfied
	□	□

How satisfied are you that the information in this book is:	Satisfied	Dissatisfied
Accurate	□	□
Complete	□	□
Easy to find	□	□
Easy to understand	□	□
Well organized	□	□
Applicable to your task	□	□

Specific Comments or Problems:

____

____

____

Please tell us how we can improve this book:

____

____

____

Thank you for your response. When you send information to IBM, you grant IBM the right to use or distribute the information without incurring any obligation to you. You of course retain the right to use the information in any way you choose.

____
Name

____
Company or Organization

____
Phone No.

____
Address

____

____

Help us help you!
GC31-8608-00

Cut or Fold
Along Line

Fold and Tape **Please do not staple** Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

Information Development
Department CGMD
International Business Machines Corporation
PO BOX 12195
RESEARCH TRIANGLE PARK NC 27709-9990

Fold and Tape **Please do not staple** Fold and Tape

Cut or Fold
Along Line

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

GC31-8608-00