Sun™ Crypto Accelerator 4000 Board Version 2.0 Installation and User's Guide

Sun Microsystems, Inc.
www.sun.com

Part No. 817-6972-10
March 2005, Revision A

Submit comments about this document at: `http://www.sun.com/hwdocs/feedback`

Contents

1. Product Overview 1

Product Features 1

What's New in Version 2.0 2

Key Protocols and Interfaces 2

Key Features 3

Supported Applications 3

Supported Cryptographic Protocols 3

Diagnostic Support 4

Cryptographic Algorithm Acceleration 4

Supported Cryptographic Algorithms 4

IPsec Acceleration 5

SSL Acceleration 6

Hardware Overview 6

Sun Crypto Accelerator 4000 MMF Adapter 6

LED Displays 8

Sun Crypto Accelerator 4000 UTP Adapter 8

LED Displays 10

Dynamic Reconfiguration and High Availability 10

Load Sharing 11

Hardware and Software Requirements 11

Required Patches 11

2. Installing the Sun Crypto Accelerator 4000 Board 13

Handling the Board 13

Installing the Board 14

▼ To Install the Hardware 14

Installing the Sun Crypto Accelerator 4000 Software With the install Script 16

Version 1.1 and 2.0 Software Contained on the CD-ROM 16

▼ To Install the Software With the install Script 17

Directories and Files 20

Removing the Sun Crypto Accelerator 4000 Software With the remove Script 22

▼ To Remove the Software With the `remove` Script on the CD-ROM 22

▼ To Remove the Software With the `/var/tmp/crypto_acc.remove` Script 22

Installing the Software Without the install Script 23

Version 1.1 and 2.0 Software Contained on the CD-ROM 23

▼ To Install the Software Without the install Script 24

Removing the Software Without the remove Script 26

▼ To Remove the Software Without the remove Script 27

3. Configuring Driver Parameters 29

Ethernet Device Driver (`vca`) Parameters 29

Network Driver Parameter Values and Definitions 30

Advertised Link Parameters 31

Flow Control Parameters 33

Gigabit Forced Mode Parameter 34

Interpacket Gap Parameters 34

Interrupt Parameters 35

Random Early Drop Parameters 36

PCI Bus Interface Parameters 37
Setting vca Driver Parameters 38
Setting Parameters Using the ndd Utility 38
▼ To Specify Device Instances for the ndd Utility 38
Noninteractive and Interactive Modes 39
Setting Autonegotiation or Forced Mode 42
▼ To Disable Autonegotiation Mode 42
Setting Parameters Using the vca.conf File 43
▼ To Set Driver Parameters Using a vca.conf File 43
Setting Parameters for All Sun Crypto Accelerator 4000 vca Devices With the vca.conf File 44
▼ To Set Parameters for All vca Devices With the vca.conf File 45
Example vca.conf File 45
Cryptographic and Ethernet Driver Operating Statistics 45
Cryptographic Driver Statistics 46
Ethernet Driver Statistics 46
Reporting the Link Partner Capabilities 51
▼ To Check Link Partner Settings 54
IPsec In-Line Acceleration Statistics 55
Network Configuration 56
Configuring the Network Host Files 56
IPsec Hardware Acceleration Configuration 58
Enabling Out-of-Band IPsec Acceleration 58
Enabling In-Line IPsec Acceleration 59
▼ To Enable In-Line IPsec Hardware Acceleration 59
Jumbo Frames Configuration 59
Cryptographic Configuration 60
Enabling AES Encryption/Decryption 60

4. Administering the Sun Crypto Accelerator 4000 Board 61

Using the `vcaadm` Utility 61

Modes of Operation 63

Single-Command Mode 63

File Mode 64

Interactive Mode 64

Logging In and Out With `vcaadm` 64

Logging In to a Board With `vcaadm` 65

Logging Out of a Board With `vcaadm` 67

Entering Commands With `vcaadm` 68

Getting Help for Commands 69

Quitting the `vcaadm` Utility in Interactive Mode 70

Initializing the Board With `vcaadm` 70

▼ To Initialize the Board With a New Keystore 71

Initializing the Board to Use an Existing Keystore 72

▼ To Initialize the Board to Use an Existing Keystore 73

Managing Keystores With `vcaadm` 74

Naming Requirements 75

Password Requirements 75

Populating a Keystore With Security Officers 76

Populating a Keystore With Users 76

Listing Users and Security Officers 78

Changing Passwords 78

Enabling or Disabling Users 78

Deleting Users 79

Deleting Security Officers 80

Backing Up the Master Key 80

Locking the Keystore to Prevent Backups 81

Multi-Admin Authentication 81

Managing Multi-Admin Mode With `vcaadm` 82

Managing Boards With `vcaadm` 88

Setting the Auto-Logout Time 88

Displaying Board Status 89

Loading New Firmware 89

Resetting the Board 90

Rekeying the Board 90

Performing a Software Zeroize on the Board 91

Using the `vcaadm diagnostics` Command 92

Managing the vcad Service 92

`vcad` Configuration File 94

Using the `vcadiag` Utility 95

Assigning Different MAC Addresses to Multiple Boards Installed in the Same Server 98

▼ To Assign Different MAC Addresses From a Terminal Window 98

▼ To Assign Different MAC Addresses From the OpenBoot PROM Level 99

5. Building PKCS#11 Applications for Use With the Sun Crypto Accelerator 4000 Board 101

Board Administration 102

Slot Description 102

Keystore Slot 103

Sun Metaslot 103

Configuring Sun Metaslot to Use the Sun Crypto Accelerator 4000 Keystore 104

Configuring Secure Failover for Sun Metaslot 104

Hardware Slot 106

PKCS#11 and FIPS Mode 107

Developing Applications to Use PKCS#11 107

Sun Crypto Accelerator 4000 PKCS#11 Implementation Specifics 108

6. Installing and Configuring Sun ONE Server Software 113

Administering Security for Sun ONE Web Servers 113

Concepts and Terminology 114

Slots and Tokens 116

Before Configuring Sun ONE Web Servers 116

Populating a Keystore 117

▼ To Populate a Keystore 118

Overview of Enabling Sun ONE Web Servers 119

Installing and Configuring Sun ONE Web Server 6.1 119

▼ To Install Sun ONE Web Server 6.1 120

Configuring Sun ONE Web Server 6.1 120

▼ To Create a Trust Database 121

▼ To Register the Board With the Web Server 122

▼ To Generate a Server Certificate 123

▼ To Install the Server Certificate 126

▼ To Enable the Web Server for SSL 127

Configuring Sun ONE Web Servers to Start Up Without User Interaction on Reboot 129

▼ To Create an Encrypted Key for Automatic Startup of Sun ONE Web Servers on Reboot 129

7. Installing and Configuring Apache Web Server Software 131

Creating a Private Key and Certificate 131

▼ To Create a Private Key and Certificate 131

Enabling Apache Web Servers 133

▼ To Enable the Apache Web Server 134

8. Diagnostics and Troubleshooting 135

Diagnostic Software 135

Performing SunVTS Diagnostics 135

Performing vcaadm Diagnostics 136

Performing vcadiag Diagnostics 136

Using `kstat` to Determine Cryptographic Activity 137

Using the OpenBoot PROM FCode Self-Test 138

▼ Performing the Ethernet FCode Self-Test Diagnostic 138

Sun's Predictive Self-Healing 141

Troubleshooting the Sun Crypto Accelerator 4000 Board 141

`show-devs` 142

`.properties` 143

`watch-net` 144

A. Specifications 145

Sun Crypto Accelerator 4000 MMF Adapter 145

Connectors 145

Physical Dimensions 147

Performance Specifications 147

Power Requirements 147

Interface Specifications 148

Environmental Specifications 148

Sun Crypto Accelerator 4000 UTP Adapter 148

Connectors 148

Physical Dimensions 150

Performance Specifications 150

Power Requirements 150

Interface Specifications 151

Environmental Specifications 151

B. Software Licenses 153

Third Party License Terms 157

C. Manual Pages 161

D. Zeroizing the Hardware 163

Zeroizing the Sun Crypto Accelerator 4000 Hardware to the Factory State 163

- To Zeroize the Sun Crypto Accelerator 4000 Board With a Hardware Jumper 164

E. Mechanisms and Restriction 167

Index 169

Tables

TABLE 1-1 IPsec Cryptographic Algorithms 4

TABLE 1-2 SSL Cryptographic Algorithms 5

TABLE 1-3 Accelerated IPsec Algorithms 5

TABLE 1-4 Supported SSL Algorithms 6

TABLE 1-5 Front Panel Display LEDs for the MMF Adapter 8

TABLE 1-6 Front Panel Display LEDs for the UTP Adapter 10

TABLE 1-7 Hardware and Software Requirements 11

TABLE 2-1 Files in the `/cdrom/cdrom0` Directory 18

TABLE 2-2 Files in the `/cdrom/cdrom0`/Sun_Crypto_Acc_4000_2_0 Directory 18

TABLE 2-3 Sun Crypto Accelerator 4000 Directories and Files 20

TABLE 2-4 Files in the `/cdrom/cdrom0` Directory 25

TABLE 3-1 `vca` Driver Parameter, Status, and Descriptions 30

TABLE 3-2 Operational Mode Parameters 32

TABLE 3-3 Read-Write Flow Control Keyword Descriptions 33

TABLE 3-4 Gigabit Forced Mode Parameter 34

TABLE 3-5 Parameters Defining `enable-ipg0` and `ipg0` 34

TABLE 3-6 Read-Write Interpacket Gap Parameter Values and Descriptions 35

TABLE 3-7 RX Blanking Register for Alias Read 35

TABLE 3-8 RX Random Early Detecting 8-Bit Vectors 36

TABLE 3-9 PCI Bus Interface Parameters 37

TABLE 3-10 Device Path Name 44
TABLE 3-11 Cryptographic Driver Statistics 46
TABLE 3-12 Ethernet Driver Statistics 46
TABLE 3-13 TX and RX MAC Counters 47
TABLE 3-14 Current Ethernet Link Properties 50
TABLE 3-15 Read-Only `vca` Device Capabilities 50
TABLE 3-16 Read-Only Link Partner Capabilities 51
TABLE 3-17 Driver-Specific Parameters 52
TABLE 3-18 Cryptographic Driver Statistics for In-Line IPsec Acceleration 55
TABLE 4-1 `vcaadm` Options 62
TABLE 4-2 `vcaadm` Prompt Variable Definitions 67
TABLE 4-3 `connect` Command Optional Parameters 68
TABLE 4-4 Security Officer Name, User Name, and Keystore Name Requirements 75
TABLE 4-5 Password Requirement Settings 75
TABLE 4-6 Key Types 91
TABLE 4-7 `vcad` Command Options 93
TABLE 4-8 Command-Line Directives for the `vcad` Command 94
TABLE 4-9 `vcadiag` Options 96
TABLE 5-1 PKCS#11 Attributes and Default Values 109
TABLE 6-1 Passwords Required for Sun ONE Web Servers 117
TABLE 6-2 Requestor Information Fields 125
TABLE 6-3 Fields for the Certificate to Install 127
TABLE A-1 SC Connector Link Characteristics (IEEE P802.3z) 146
TABLE A-2 Physical Dimensions 147
TABLE A-3 Performance Specifications 147
TABLE A-4 Power Requirements 147
TABLE A-5 Interface Specifications 148
TABLE A-6 Environmental Specifications 148
TABLE A-7 Cat-5 Connector Link Characteristics 149
TABLE A-8 Physical Dimensions 150

TABLE A-9 Performance Specifications 150
TABLE A-10 Power Requirements 150
TABLE A-11 Interface Specifications 151
TABLE A-12 Environmental Specifications 151
TABLE C-1 Sun Crypto Accelerator 4000 Online Manual Pages 161
TABLE E-1 Supported PKCS#11 Mechanisms 168

Declaration of Conformity (Fiber MMF)

Compliance Model Number: Venus-FI
Product Family Name: Sun Crypto Accelerator 4000 - Fiber (X4012A)

EMC

USA - FCC Class B

This equipment complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

1) This equipment may not cause harmful interference.

2) This equipment must accept any interference that may cause undesired operation.

European Union

This equipment complies with the following requirements of the EMC Directive 89/336/EEC:

As Telecommunication Network Equipment (TNE) in both Telecom Centers and Other Than Telecom Centers per (as applicable):

EN300-386 V.1.3.1 (09-2001) Required Limits:

EN55022/CISPR22	Class B
EN61000-3-2	Pass
EN61000-3-3	Pass
EN61000-4-2	6 kV (Direct), 8 kV (Air)
EN61000-4-3	3 V/m 80-1000MHz, 10 V/m 800-960 MHz and 1400-2000 MHz
EN61000-4-4	1 kV AC and DC Power Lines, 0.5 kV Signal Lines,
EN61000-4-5	2 kV AC Line-Gnd, 1 kV AC Line-Line and Outdoor Signal Lines, 0.5 kV Indoor Signal Lines > 10m.
EN61000-4-6	3 V
EN61000-4-11	Pass

As information Technology Equipment (ITE) Class B per (as applicable):

EN55022:1998/CISPR22:1997	Class B

EN55024:1998 Required Limits:

EN61000-4-2	4 kV (Direct), 8 kV (Air)
EN61000-4-3	3 V/m
EN61000-4-4	1 kV AC Power Lines, 0.5 kV Signal and DC Power Lines
EN61000-4-5	1 kV AC Line-Line and Outdoor Signal Lines, 2 kV AC Line-Gnd, 0.5 kV DC Power Lines
EN61000-4-6	3 V
EN61000-4-8	1 A/m
EN61000-4-11	Pass
EN61000-3-2:1995 + A1, A2, A14	Pass
EN61000-3-3:1995	Pass

Safety

This equipment complies with the following requirements of the Low Voltage Directive 73/23/EEC:

EC Type Examination Certificates:

EN 60950:2000, 3rd Edition
IEC 60950:2000, 3rd Edition
Evaluated to all CB Countries
UL 60950, 3rd Edition, CSA C22.2 No. 60950-00

Supplementary Information

This product was tested and complies with all the requirements for the CE Mark.

/S/

Dennis P. Symanski
Manager, Compliance Engineering
Sun Microsystems, Inc.
4150 Network Circle, MPK15-102
Santa Clara, CA 95054, USA
Tel: 650-786-3255
Fax: 650-786-3723

/S/

Pamela J Dullaghan
Quality Program Manager
Sun Microsystems Scotland, Limited
Springfield, Linlithgow
West Lothian, EH49 7LR
Scotland, United Kingdom
Tel: +44 1 506 672 395
Fax: +44 1 506 672 855

Declaration of Conformity (Copper UTP)

Compliance Model Number: Venus-CU
Product Family Name: Sun Crypto Accelerator 4000 - Copper (X4011A)

EMC

USA - FCC Class B

This equipment complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

1) This equipment may not cause harmful interference.

2) This equipment must accept any interference that may cause undesired operation.

European Union

This equipment complies with the following requirements of the EMC Directive 89/336/EEC:

As Telecommunication Network Equipment (TNE) in both Telecom Centers and Other Than Telecom Centers per (as applicable):

EN300-386 V.1.3.1 (09-2001) Required Limits:

EN55022/CISPR22	Class B
EN61000-3-2	Pass
EN61000-3-3	Pass

EN61000-4-2	6 kV (Direct), 8 kV (Air)
EN61000-4-3	3 V/m 80-1000MHz, 10 V/m 800-960 MHz and 1400-2000 MHz
EN61000-4-4	1 kV AC and DC Power Lines, 0.5 kV Signal Lines,
EN61000-4-5	2 kV AC Line-Gnd, 1 kV AC Line-Line and Outdoor Signal Lines, 0.5 kV Indoor Signal Lines > 10m.
EN61000-4-6	3 V
EN61000-4-11	Pass

As information Technology Equipment (ITE) Class B per (as applicable):

EN55022:1998/CISPR22:1997	Class B
EN55024:1998 Required Limits:	
EN61000-4-2	4 kV (Direct), 8 kV (Air)
EN61000-4-3	3 V/m
EN61000-4-4	1 kV AC Power Lines, 0.5 kV Signal and DC Power Lines
EN61000-4-5	1 kV AC Line-Line and Outdoor Signal Lines, 2 kV AC Line-Gnd, 0.5 kV DC Power Lines
EN61000-4-6	3 V
EN61000-4-8	1 A/m
EN61000-4-11	Pass
EN61000-3-2:1995 + A1, A2, A14	Pass
EN61000-3-3:1995	Pass

Safety

This equipment complies with the following requirements of the Low Voltage Directive 73/23/EEC:

EC Type Examination Certificates:

EN 60950:2000, 3rd Edition

IEC 60950:2000, 3rd Edition

Evaluated to all CB Countries

UL 60950, 3rd Edition, CSA C22.2 No. 60950-00

Supplementary Information

This product was tested and complies with all the requirements for the CE Mark.

/S/

Dennis P. Symanski
Manager, Compliance Engineering
Sun Microsystems, Inc.
4150 Network Circle, MPK15-102
Santa Clara, CA 95054, USA
Tel: 650-786-3255
Fax: 650-786-3723

/S/

Pamela J Dullaghan
Quality Program Manager
Sun Microsystems Scotland, Limited
Springfield, Linlithgow
West Lothian, EH49 7LR
Scotland, United Kingdom
Tel: +44 1 506 672 395
Fax: +44 1 506 672 855

Regulatory Compliance Statements

Your Sun product is marked to indicate its compliance class:

- Federal Communications Commission (FCC) — USA
- Industry Canada Equipment Standard for Digital Equipment (ICES-003) — Canada
- Voluntary Control Council for Interference (VCCI) — Japan
- Bureau of Standards Metrology and Inspection (BSMI) — Taiwan

Please read the appropriate section that corresponds to the marking on your Sun product before attempting to install the product.

FCC Class A Notice

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This device may not cause harmful interference.
2. This device must accept any interference received, including interference that may cause undesired operation.

Note: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy, and if it is not installed and used in accordance with the instruction manual, it may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user will be required to correct the interference at his own expense.

Shielded Cables: Connections between the workstation and peripherals must be made using shielded cables to comply with FCC radio frequency emission limits. Networking connections can be made using unshielded twisted-pair (UTP) cables.

Modifications: Any modifications made to this device that are not approved by Sun Microsystems, Inc. may void the authority granted to the user by the FCC to operate this equipment.

FCC Class B Notice

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This device may not cause harmful interference.
2. This device must accept any interference received, including interference that may cause undesired operation.

Note: This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/television technician for help.

Shielded Cables: Connections between the workstation and peripherals must be made using shielded cables in order to maintain compliance with FCC radio frequency emission limits. Networking connections can be made using unshielded twisted pair (UTP) cables.

Modifications: Any modifications made to this device that are not approved by Sun Microsystems, Inc. may void the authority granted to the user by the FCC to operate this equipment.

ICES-003 Class A Notice - Avis NMB-003, Classe A

This Class A digital apparatus complies with Canadian ICES-003.

Cet appareil numérique de la classe A est conforme à la norme NMB-003 du Canada.

ICES-003 Class B Notice - Avis NMB-003, Classe B

This Class B digital apparatus complies with Canadian ICES-003.

Cet appareil numérique de la classe B est conforme à la norme NMB-003 du Canada.

VCCI 基準について

クラス A VCCI 基準について

クラス A VCCI の表示があるワークステーションおよびオプション製品は、クラス A 情報技術装置です。これらの製品には、下記の項目が該当します。

この装置は、情報処理装置等電波障害自主規制協議会 (VCCI) の基準に基づくクラス A 情報技術装置です。この装置を家庭環境で使用すると電波妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ずるよう要求されることがあります。

クラス B VCCI 基準について

クラス B VCCI の表示  があるワークステーションおよびオプション製品は、クラス B 情報技術装置です。これらの製品には、下記の項目が該当します。

この装置は、情報処理装置等電波障害自主規制協議会 (VCCI) の基準に基づくクラス B 情報技術装置です。この装置は、家庭環境で使用することを目的としていますが、この装置がラジオやテレビジョン受信機に近接して使用されると、受信障害を引き起こすことがあります。取扱説明書に従って正しい取り扱いをしてください。

BSMI Class A Notice

The following statement is applicable to products shipped to Taiwan and marked as Class A on the product compliance label.

警告使用者：
這是甲類的資訊產品，在居住的環境中使用時，可能會造成射頻 干擾，在這種情況下，使用者 會被要求採取某些適當的對策。

Preface

The *Sun Crypto Accelerator 4000 Board Version 2.0 Installation and User's Guide* lists the features, protocols, and interfaces of the Sun Crypto Accelerator 4000 board and describes how to install, configure, and manage the board in your system.

This user's guide assumes that you are a network administrator with experience configuring one or more of the following: Solaris Operating System, Sun platforms with PCI I/O cards, Sun ONE and Apache Web Servers, IPsec, SunVTS™ software, and certification authority acquisitions.

How This Book Is Organized

This book is organized as follows:

- Chapter 1 lists the product features, protocols, and interfaces of the Sun Crypto Accelerator 4000 board, and describes the hardware and software requirements.
- Chapter 2 describes how to install and remove the Sun Crypto Accelerator 4000 hardware and software.
- Chapter 3 defines the Sun Crypto Accelerator 4000 tunable driver parameters, and describes how to configure them with the `ndd` utility and the `vca.conf` file. This chapter also describes how to enable autonegotiation or forced mode for link parameters at the OpenBoot™ PROM interface and how to configure the network `hosts` file.
- Chapter 4 describes how to configure the Sun Crypto Accelerator 4000 board and manage keystores with the `vcaadm` and `vcadiag` utilities.
- Chapter 5 describes how different configurations of the board work with the PKCS#11 interface.
- Chapter 6 explains how to configure the Sun Crypto Accelerator 4000 board for use with Sun ONE Web Servers.

- Chapter 7 explains how to configure the Sun Crypto Accelerator 4000 board for use with Apache Web Servers.
- Chapter 8 describes how to test the Sun Crypto Accelerator 4000 board with the `vcaadm` and `vcadiag` utilities and the on board FCode self-test. This chapter also provides troubleshooting techniques with OpenBoot PROM commands.
- Appendix A lists the specifications for the Sun Crypto Accelerator 4000 board.
- Appendix B provides software notices and licenses from other software organizations that govern the use of third-party software used with the Sun Crypto Accelerator 4000 board.
- Appendix C provides a description of the Sun Crypto Accelerator 4000 commands and lists the online manual pages for each command.
- Appendix D describes how to zeroize the Sun Crypto Accelerator 4000 board to the factory state which is the Failsafe mode for the board.
- Appendix E lists the PKCS#11 mechanisms supported by the Sun Crypto Accelerator 4000 board.

Using UNIX Commands

This document does not contain information on basic UNIX® commands and procedures such as shutting down the system, booting the system, and configuring devices.

See one or more of the following for this information:

- *Solaris Hardware Platform Guide*
- Online documentation for the Solaris Operating System, available at: `http://docs.sun.com`
- Other software documentation that you received with your system

Shell Prompts

Shell	Prompt
C shell	*machine-name*%
C shell superuser	*machine-name*#
Bourne shell and Korn shell	$
Bourne shell and Korn shell superuser	#

Typographic Conventions

Typeface	Meaning	Examples
`AaBbCc123`	The names of commands, files, and directories; on-screen computer output	Edit your `.login` file. Use `ls -a` to list all files. `% You have mail.`
`AaBbCc123`	What you type, when contrasted with on-screen computer output	`% su` `Password:`
AaBbCc123	Book titles, new words or terms, words to be emphasized	Read Chapter 6 in the *User's Guide*. These are called *class* options. You *must* be superuser to do this.
	Command-line variable; replace with a real name or value	To delete a file, type `rm` *filename*.

Accessing Sun Documentation Online

You can view, print, or purchase a broad selection of Sun documentation, including localized versions, at:

`http://www.sun.com/documentation`

Contacting Sun Technical Support

If you have technical questions about this product that are not answered in this document, go to:

`http://www.sun.com/service/contacting`

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. You can submit your comments by going to:

`http://www.sun.com/hwdocs/feedback`

Please include the title and part number of your document with your feedback:

Sun Crypto Accelerator 4000 Board Version 2.0 Installation and User's Guide, part number 817-6972-10

CHAPTER 1

Product Overview

This chapter provides an overview of the Sun Crypto Accelerator 4000 board, and contains the following sections:

- "Product Features" on page 1
- "Hardware Overview" on page 6
- "Hardware and Software Requirements" on page 11

Product Features

The Sun Crypto Accelerator 4000 board is a Gigabit Ethernet-based network interface card that supports cryptographic hardware acceleration for IPsec and SSL (both symmetric and asymmetric) on Sun servers. In addition to operating as a standard Gigabit Ethernet network interface card for unencrypted network traffic, the board contains cryptographic hardware to support a higher throughput for encrypted IPsec traffic than the standard software solution.

Once installed, the board is initialized and configured with the `vcaadm` utility which manages the keystore and user information and determines the level of security in which the board operates. Once a keystore and security officer account are configured, Java and PKCS11 applications such as Sun ONE server software, and OpenSSL applications such as Apache can be configured to use the board for cryptographic acceleration.

Note – Solaris 10 or future compatible versions of the Solaris Operating System are required to use version 2.0 of the board.

Caution – Do not use the Sun Crypto Accelerator 4000 board as a boot device.

What's New in Version 2.0

The Sun Crypto Accelerator 4000 version 2.0 adds the following features to what was provided in previous releases:

- Support for new Solaris 10 Operating System features
  - The board acts as a cryptographic service provider for the Solaris Cryptographic Framework. User applications can now access the board and other cryptographic services through the common Solaris Cryptographic Framework interfaces. Refer to the Solaris Cryptographic Framework documentation for detailed information.
  - Support for Sun's Predictive Self-Healing, which provides improved fault detection and diagnosis.
  - Support for the new Service Management Facility (SMF), which is an improved mechanism for controlling system startup and the relationship between services.
- Multi-Admin keystore security, supporting the requirement of multiple security officers to authenticate keystore backup and restore operations. This feature provides an optional added level of keystore security.
- Jumbo frames (9 Kbyte) support over the board's network interface. The larger frame size can be used to improve system performance in networks which support jumbo frames.
- Support for the AES algorithm.

Note – AES support is implemented in the firmware and provides less performance than available on the host system. AES is only intended to be used by applications that require the added security of keeping keys in the Sun Crypto Accelerator 4000 hardware. By default, AES support is not enabled.

Key Protocols and Interfaces

The Sun Crypto Accelerator 4000 board is interoperable with existing Ethernet equipment assuming standard Ethernet minimum and maximum frame size (64 to 1518 bytes), frame format, and compliance with the following standards and protocols:

- Full-size PCI 33/66 Mhz, 32/64-bit
- IEEE 802.3 CSMA/CD (Ethernet)
- IEEE 802.2 Logical Link Control
- SNMP (limited MIB)
- Full- and half-duplex Gigabit Ethernet interface (IEEE 802.z)
- Universal dual voltage signaling (3.3V and 5V)

Key Features

- Gigabit Ethernet with either copper or fiber interface
- Accelerates IPsec and SSL cryptographic functions
- Session establishment rate – up to 8000 operations per second
- Bulk encryption rate – up to 800 Mbps
- Provides up to 2048-bit RSA encryption
- Delivers up to 10 times faster 3DES bulk data encryption
- Provides tamper-proof, centralized security key and certificate administration for Sun ONE Web Server for increased security and simplified key management
- Designed for FIPS 140-2 Level 3 certification
- Low CPU utilization – frees up server system resource and bandwidth
- Secure private key storage and management
- Dynamic reconfiguration (DR) and redundancy/failover support on Sun's midframe and high-end servers
- Load balancing for RX packets among multiple CPUs
- Full flow control support (IEEE 802.3x)

The Sun Crypto Accelerator 4000 boards are designed to comply with the security requirements for cryptographic modules as documented in the Federal Information Processing Standard (FIPS) 140-2, Level 3.

Supported Applications

- Solaris Cryptographic Framework
- Sun ONE Server software
- Apache Web Server

Supported Cryptographic Protocols

The board supports the following protocols:

- IPsec for IPv4 and IPv6, including IKE
- SSLv2, SSLv3, TLSv1

The board accelerates the following IPsec functions:

- ESP (DES, 3DES) encryption
- ESP (SHA1, MD5) authentication *
- AH (SHA1, MD5) authentication *

* When configured for in-line IPsec acceleration (See "In-Line IPsec Hardware Acceleration" on page 6)

The board accelerates the following SSL functions:

- Secure establishment of a set of cryptographic parameters and secret keys between a client and a server
- Secure key storage on the board – keys are encrypted if they leave the board

Diagnostic Support

- User-executable self-test using OpenBoot PROM
- SunVTS diagnostic tests
- Security officer initiated diagnostics (`vcadiag` and `vcaadm`)

Cryptographic Algorithm Acceleration

Together with the Solaris Cryptographic Framework, the board accelerates cryptographic algorithms in both hardware and software. The reason for this complexity is that the cost of accelerating cryptographic algorithms is not uniform across all algorithms. Some cryptographic algorithms were designed specifically to be implemented in hardware, others were designed to be implemented in software. For hardware acceleration, there is the additional cost of moving data from the user application to the hardware acceleration device, and moving the results back to the user application. Note that a few cryptographic algorithms can be performed by highly tuned software as quickly as they can be performed in dedicated hardware.

Supported Cryptographic Algorithms

The Sun Crypto Accelerator 4000 driver (`vca`) examines each cryptographic request and determines the best location for the acceleration (host processor or Sun Crypto Accelerator 4000), to achieve maximum throughput. Load distribution is based on the cryptographic algorithm, the current job load, and the data size.

The board accelerates the following IPsec algorithms.

TABLE 1-1 IPsec Cryptographic Algorithms

Type	Algorithm
Symmetric	DES, 3DES
Hash*	MD5, SHA1

* When configured for in-line IPsec hardware acceleration.

The board accelerates the following SSL algorithms.

TABLE 1-2 SSL Cryptographic Algorithms

Type	Algorithm
Symmetric	DES, 3DES, AES
Asymmetric	Diffie-Hellman (Apache only) and RSA (up to 2048 bit key), DSA

IPsec Acceleration

The board supports two forms of IPsec acceleration: out-of-band and in-line. Both configurations offload high-overhead cryptographic operations from the SPARC® processor to the board. See "IPsec Hardware Acceleration Configuration" on page 58.

TABLE 1-3 Accelerated IPsec Algorithms

Algorithm	Out-of-Band	In-Line
AES	X	
DES	X	X
3DES	X	X
MD5	X	X
SHA1	X	X

Out-of-Band IPsec Hardware Acceleration

When the board is configured for out-of-band IPsec acceleration, supported encryption and decryption operations are accelerated in hardware. All IPsec specific packet processing is performed by the host Solaris IPsec software. See "Enabling Out-of-Band IPsec Acceleration" on page 58.

Note – No IPsec configuration or tuning is required to use the board for out-of-band IPsec acceleration in Solaris 10. You simply install the Sun Crypto Accelerator 4000 packages and reboot.

In-Line IPsec Hardware Acceleration

When configured for in-line IPsec acceleration, supported encryption, decryption, and authentication operations are accelerated in hardware. Portions of the IPsec specific packet processing are performed directly by the board. See "Enabling In-Line IPsec Acceleration" on page 59 for instructions on how to configure the board for in-line IPsec acceleration.

SSL Acceleration

TABLE 1-4 shows which SSL accelerated algorithms may be off-loaded to hardware and which software algorithms are provided for Sun ONE and Apache Web Servers.

TABLE 1-4 Supported SSL Algorithms

Algorithm	Sun ONE Web Servers	Apache Web Servers
AES	X	
RSA	X	X
DSA	X	X
DES	X	X
3DES	X	X

Hardware Overview

The Sun Crypto Accelerator 4000 hardware is a full-size (4.2 inches x 12.283 inches) cryptographic accelerator PCI Gigabit Ethernet adapter that enhances the performance of IPsec and SSL on Sun servers.

Sun Crypto Accelerator 4000 MMF Adapter

The Sun Crypto Accelerator 4000 MMF adapter is a single-port Gigabit Ethernet fiber optics PCI bus card. It operates in 1000 Mbps Ethernet networks only.

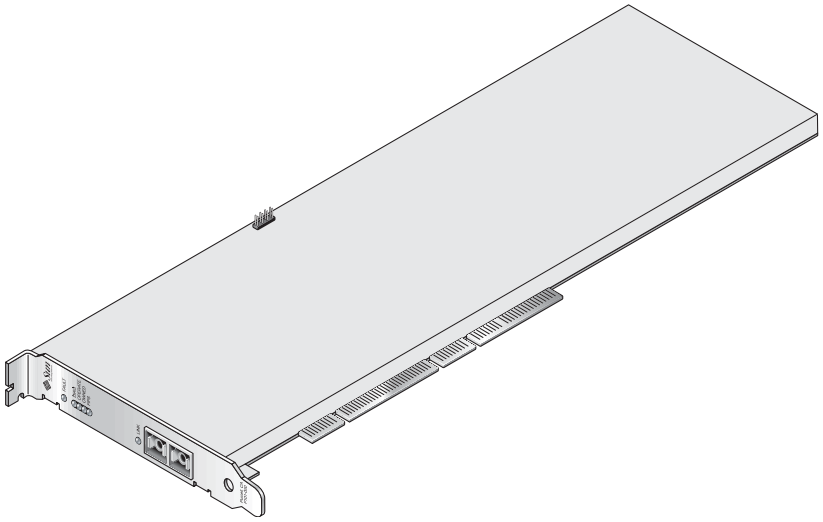

FIGURE 1-1 Sun Crypto Accelerator 4000 MMF Adapter

LED Displays

TABLE 1-5 Front Panel Display LEDs for the MMF Adapter

Label	Meaning if Lit	Color
FAULT	On when the board is in the HALTED (fatal error) state or low-level hardware initialization failed. Flashing if an error occurred during the boot process.	Red
DIAG	On in POST, DIAGNOSTICS, and FAILSAFE (firmware not upgraded) state. Flashing when running DIAGNOSTICS.	Green
OPERATE	On in POST, DIAGNOSTICS, and DISABLED (driver not attached) state. Flashing in IDLE, OPERATIONAL, and FAILSAFE states.	Green
INIT	On if the security officer has initialized the board with `vcaadm`. See "Initializing the Board With vcaadm" on page 70. Flashing if the ZEROIZE jumper is present.	Green
FIPS	On when operating in FIPS 140-2 level 3 certified mode. Off when in non-FIPS mode.	Green
LINK	On when the link is up.	Green

Sun Crypto Accelerator 4000 UTP Adapter

The Sun Crypto Accelerator 4000 UTP adapter is a single-port Gigabit Ethernet copper-based PCI bus card. It can be configured to operate in 10, 100, or 1000 Mbps Ethernet networks.

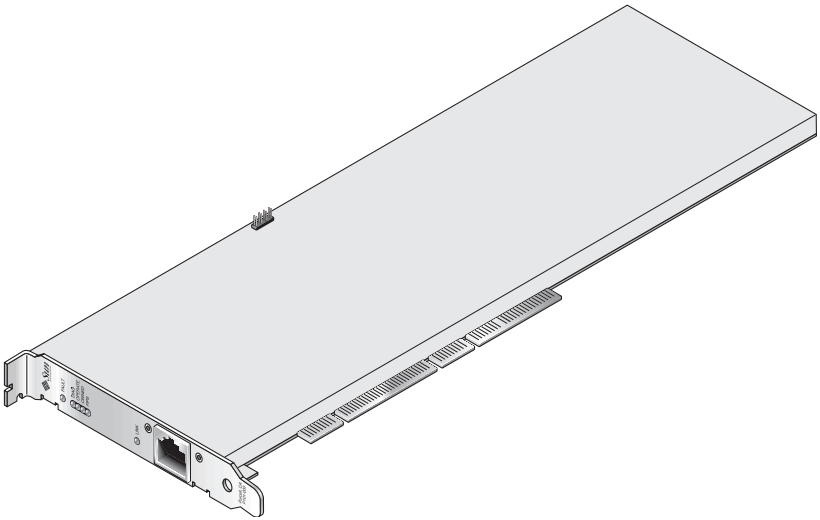

FIGURE 1-2 Sun Crypto Accelerator 4000 UTP Adapter

LED Displays

TABLE 1-6 Front Panel Display LEDs for the UTP Adapter

Label	Meaning if Lit	Color
FAULT	On when the board is in the HALTED (fatal error) state or low level hardware initialization failed. Flashing if an error occurred during the boot process.	Red
DIAG	On in POST, DIAGNOSTICS, and FAILSAFE (firmware not upgraded) state. Flashing when running DIAGNOSTICS.	Green
OPERATE	On in POST, DIAGNOSTICS, and DISABLED (driver not attached) state. Flashing in IDLE, OPERATIONAL, and FAILSAFE states.	Green
INIT	On if the security officer has initialized the board with `vcaadm`. See "Initializing the Board With vcaadm" on page 70. Flashing if the ZEROIZE jumper is present.	Green
FIPS	On when operating in FIPS 140-2 level 3 certified mode. Off when in non-FIPS mode.	Green
1000	On when using Gigabit Ethernet.	Green
ACTIVITY (no label)	On when the link is transmitting or receiving.	Amber
LINK (no label)	On when the link is up.	Green

Dynamic Reconfiguration and High Availability

The Sun Crypto Accelerator 4000 hardware and associated software provides the capability to work effectively on Sun platforms supporting Dynamic Reconfiguration (DR) and hot-plugging. During a DR or hot-plug operation, the Sun Crypto Accelerator 4000 software layer automatically detects the addition or removal of a board, and adjusts the scheduling algorithms to accommodate the change in hardware resources.

For High Availability (HA) configurations, multiple Sun Crypto Accelerator 4000 boards can be installed within a system or domain to insure that hardware acceleration is continuously available. In the unlikely event of a Sun Crypto Accelerator 4000 hardware failure, the software layer detects the failure and removes the failed board from the list of available hardware cryptographic accelerators. Sun

Crypto Accelerator 4000 software adjusts the scheduling algorithms to accommodate the reduction in hardware resources. Subsequent cryptographic requests are scheduled to the remaining boards.

Note that the Sun Crypto Accelerator 4000 hardware provides a source for high-quality entropy for the generation of long-term keys. If all the Sun Crypto Accelerator 4000 boards within a domain or system are removed, long-term keys are generated with lower-quality entropy.

Load Sharing

The Sun Crypto Accelerator 4000 software allows for the distribution of load across as many boards as are installed within the Solaris domain or system. Incoming cryptographic requests are distributed across the boards based on fixed-length work queues. Cryptographic requests are directed to the first board, and subsequent requests stay directed to the first board until it is running at full capacity. Once the first board is running at full capacity, further requests are queued to the next board available that can accept the request of this type. The queueing mechanism is designed to optimize throughput by facilitating request coalescing at the board.

Hardware and Software Requirements

TABLE 1-7 provides a summary of the hardware and software requirements for the Sun Crypto Accelerator 4000 adapter.

TABLE 1-7 Hardware and Software Requirements

Hardware and Software	Requirements
Hardware	Sun Fire™ V120, V210, V240, V250, 280R, V440, V490, V880, V880z, V890, 4800, 4900, 6800, 6900, 12K, 15K, 20K; Netra™ 20 (lw4), 120, 240; Sun Blade™ 150, 1500, 2000, 2500
Operating System	Solaris 10 and future compatible releases

Required Patches

Refer to the *Sun Crypto Accelerator 4000 Board Version 2.0 Release Notes* for detailed required patch information.

CHAPTER 2

Installing the Sun Crypto Accelerator 4000 Board

This chapter describes how to install the Sun Crypto Accelerator 4000 hardware and also how to install and remove the software with automated scripts. This chapter includes the following sections:

- "Handling the Board" on page 13
- "Installing the Board" on page 14
- "Installing the Sun Crypto Accelerator 4000 Software With the install Script" on page 16
- "Directories and Files" on page 20
- "Removing the Sun Crypto Accelerator 4000 Software With the remove Script" on page 22
- "Installing the Software Without the install Script" on page 23
- "Removing the Software Without the remove Script" on page 26

Once you have installed the hardware and software of the board, you need to initialize the board with configuration and keystore information. See "Initializing the Board With vcaadm" on page 70 for information on how to initialize the board.

Handling the Board

Each board is packed in a special antistatic bag to protect it during shipping and storage. To avoid damaging the static-sensitive components on the board, reduce any static electricity on your body before touching the board by using one of the following methods:

- Touch the metal frame of the computer.
- Attach an antistatic wrist strap to your wrist and to a grounded metal surface.

Caution – To avoid damaging the sensitive components on the board, wear an antistatic wrist strap when handling the board, hold the board by its edges only, and always place the board on an antistatic surface (such as the plastic bag it came in).

Installing the Board

Installing the Sun Crypto Accelerator 4000 board involves inserting the board into the system and loading the software tools. The hardware installation instructions include only general steps for installing the board. Refer to the documentation that came with your system for specific installation instructions.

▼ To Install the Hardware

1. **As superuser, follow the instructions that came with your system to shut down and power off the computer, disconnect the power cord, and remove the computer cover.**
2. **Locate an unused PCI slot (preferably a 64-bit, 66 MHz slot).**
3. **Attach an antistatic wrist strap to your wrist, and attach the other end to a grounded metal surface.**
4. **Using a Phillips-head screwdriver, remove the screw from the PCI slot cover.**

   Save the screw to hold the bracket in Step 5.
5. **Holding the Sun Crypto Accelerator 4000 board by its edges only, take it out of the plastic bag and insert it into the PCI slot.**
6. **Secure the screw on the rear bracket.**
7. **Replace the computer cover, reconnect the power cord, and power on the system.**

8. **Verify that the board is properly installed by entering the `show-devs` command at the OpenBoot PROM `ok` prompt:**

```
ok show-devs
.
/chosen
/packages
/upa@8,480000/SUNW,ffb@0,0
/pci@8,600000/network@1
/pci@8,600000/SUNW,qlc@4
/pci@8,600000/SUNW,qlc@4/fp@0,0
.
```

In the preceding example, the `/pci@8,600000/network@1` identifies the device path to the Sun Crypto Accelerator 4000 board. There is one such line for each board in the system.

To determine whether the Sun Crypto Accelerator 4000 device properties are listed correctly: From the `ok` prompt, navigate to the device path and type `.properties` to display the list of properties.

```
ok cd /pci@8,600000/network@1
ok .properties
assigned-addresses       82000810 00000000 00102000 00000000 00002000
                          81000814 00000000 00000400 00000000 00000100
                          82000818 00000000 00200000 00000000 00200000
                          82000830 00000000 00400000 00000000 00100000
d-fru-len                00 00 00 00
d-fru-off                00 00 e8 00
d-fru-dev                eeprom
s-fru-len                00 00 08 00
s-fru-off                00 00 e0 00
s-fru-dev                eeprom
compatible               70 63 69 38 30 38 36 2c 62 35 35 35 2e 31 30 38
reg                      00000800 00000000 00000000 00000000 00000000
                          02000810 00000000 00000000 00000000 00002000
                          02000814 00000000 00000000 00000000 00000100
                          02000818 00000000 00000000 00000000 00200000
                          02000830 00000000 00000000 00000000 00100000
address-bits             00 00 00 30
max-frame-size           00 00 40 00
network-interface-type   ethernet
device-type              network
name                     network
local-mac-address        00 03 ba 0e 99 ca
version                  Sun PCI Crypto Accelerator 4000 1000Base-T FCode
FCode 2.11.13 03/03/04
```

```
phy-type                mif
board-model             501-6039
model                   SUNW,pci-vca
fcode-rom-offset        00000000
66mhz-capable
fast-back-to-back
devsel-speed            00000001
class-code              00100000
interrupts              00000001
max-latency             00000040
cache-line-size         00000010
max-latency             00000040
min-grant               00000040
subsystem-vendor-id     0000108e
subsystem-id            00003de8
revision-id             00000002
device-id               0000b555
vendor-id               00008086
```

Installing the Sun Crypto Accelerator 4000 Software With the `install` Script

The Sun Crypto Accelerator 4000 software is included on the Sun Crypto Accelerator 4000 CD. You may need to download patches from the SunSolve web site. Refer to the *Sun Crypto Accelerator 4000 Board Version 2.0 Release Notes* for the required patches.

There are two methods to install the software, manually or with the `install` script. This section describes how to install the software with the `install` script. To install the software manually, refer to "Installing the Software Without the install Script" on page 23.

Version 1.1 and 2.0 Software Contained on the CD-ROM

The Sun Crypto Accelerator 4000 Version 2.0 CD-ROM contains both Versions 1.1 and 2.0 of the software.

Caution – Version 1.1 is for Solaris 8 and 9. Version 2.0 is supported on Solaris 10 only.

The install script path to each version is as follows:

For Version 1.1:

`/cdrom/cdrom0/Sun_Crypto_Acc_4000_1_1`

For Version 2.0:

`/cdrom/cdrom0/Sun_Crypto_Acc_4000_2_0`

The respective installation scripts are located in these directories.

▼ To Install the Software With the `install` Script

1. **The Sun Crypto Accelerator 4000 Version 1.x software should not be installed on Solaris 10. If Version 1.x exists on your Solaris 10 system, use the following command to remove all Version 1.x packages:**

```
# pkgrm SUNWkcl2o SUNWvcav SUNWvcar SUNWkcl2a SUNWkcl2u SUNWkcl2r
SUNWvcamn SUNWkcl2m SUNWkcl2i SUNWvcaa SUNWvcafw SUNWvcau
```

2. **Insert the Sun Crypto Accelerator 4000 CD into a CD-ROM drive that is connected to your system.**
   - If your system is running Sun Enterprise Volume Manager™, the system should automatically mount the CD-ROM to the `/cdrom/cdrom0` directory.
   - If your system is not running Sun Enterprise Volume Manager, mount the CD-ROM as follows:

```
# mount -F hsfs -o ro /dev/dsk/c0t6d0s2 /cdrom
```

You see the following files and directories in the `/cdrom/cdrom0` directory.

TABLE 2-1 Files in the `/cdrom/cdrom0` Directory

File or Directory	Contents
`README`	Release information
`Sun_Crypto_Acc_4000_1_1`	Contains the Sun Crypto Accelerator 4000 Version 1.1 software for Solaris 8 and 9
`Sun_Crypto_Acc_4000_2_0`	Contains the Sun Crypto Accelerator 4000 Version 2.0 software for Solaris 10 only

Refer to the *Sun Crypto Accelerator 4000 Board Version 1.1 Installation and User's Guide* (817-3693-10) for instructions on how to install the Version 1.1 software.

You see the following files and directories in the `/cdrom/cdrom0/Sun_Crypto_Acc_4000_2_0` directory.

TABLE 2-2 Files in the `/cdrom/cdrom0/Sun_Crypto_Acc_4000_2_0` Directory

File or Directory	Contents	
`README`		
`Copyright`	U.S. copyright file	
`FR_Copyright`	French copyright file	
`install`	Script that installs the Sun Crypto Accelerator 4000 software	
`remove`	Script that removes the Sun Crypto Accelerator 4000 software	
`Docs`	*Sun Crypto Accelerator 4000 Board Version 2.0 Installation and User's Guide* *Sun Crypto Accelerator 4000 Board Version 2.0 Release Notes*	
`Packages`	Contains the Sun Crypto Accelerator 4000 software packages:	
	`SUNWvcaa`	VCA administration
	`SUNWvcact`	VCA activation file
	`SUNWvcafw`	VCA firmware
	`SUNWvcamn`	VCA manual pages
	`SUNWvcaf`	VCA supplemental files
	`SUNWvcar`	VCA drivers
	`SUNWvcau`	VCA daemon

3. **Install the required software by typing:**

```
# cd /cdrom/cdrom0/Sun_Crypto_Acc_4000_2_0
# ./install
```

The `install` script analyzes the system to determine which required patches need to be installed, installs those patches, installs the main software—for example:

Note – The copyright and license information was omitted from the following example. Refer to Appendix B for copyright and software licenses.

```
# ./install
This program installs the software for the Sun Crypto Accelerator
4000, Version 2.0.


This script is about to take the following actions:
- Install Sun Crypto Accelerator 4000 support for Solaris 10

To cancel installation of this software, press 'q' followed by a Return.
         **OR**
Press Return key to begin installation:

*** Installing Sun Crypto Accelerator 4000 software for Solaris 10...
Installing required packages:
     SUNWvcaf SUNWvcact SUNWvcar SUNWvcau SUNWvcaa SUNWvcamn SUNWvcafw


Importing VCA keystore management daemon to SMF
Starting VCA keystore/management daemon

Installation of <SUNWvcaf> was successful.

Installation of <SUNWvcact> was successful.

Installation of <SUNWvcar> was successful.

Installation of <SUNWvcau> was successful.

Installation of <SUNWvcaa> was successful.

Installation of <SUNWvcamn> was successful.

Installation of <SUNWvcafw> was successful.
*** Installation complete.

To remove this software, use the 'remove' script on this CDROM, or
the following script:

        /var/tmp/crypto_acc.remove
```

```
A log of this installation can be found at:
        /var/tmp/crypto_acc.install.2005.01.31.0916
```

Directories and Files

TABLE 2-3 shows the directories created by the default installation of the Sun Crypto Accelerator 4000 software.

TABLE 2-3 Sun Crypto Accelerator 4000 Directories and Files

Directory	Contents
`/etc/opt/SUNWconn/vca/keydata`	Keystore data (encrypted)
`/opt/SUNWconn/cryptov2/bin`	Utilities
`/opt/SUNWconn/cryptov2/firmware`	Firmware
`/opt/SUNWconn/cryptov2/lib`	Support libraries
`/opt/SUNWconn/cryptov2/man`	Man pages
`/opt/SUNWconn/cryptov2/method`	SMF support files
`/opt/SUNWconn/cryptov2/sbin`	Administrative commands

FIGURE 2-1 shows the hierarchy of these directories and files.

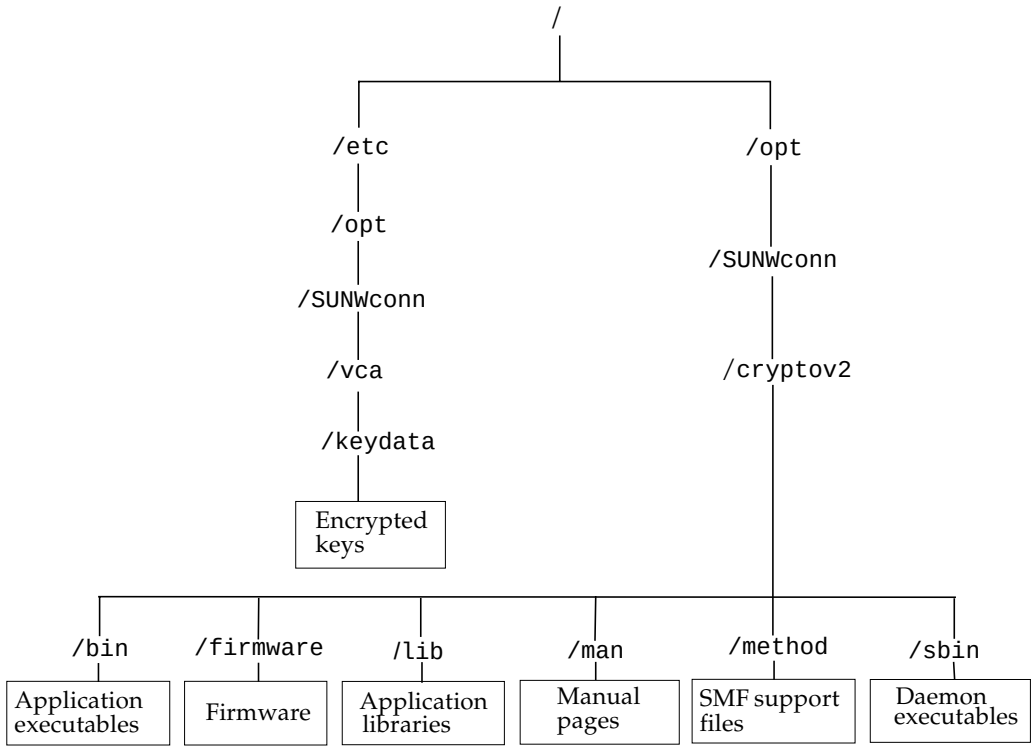

FIGURE 2-1 Sun Crypto Accelerator 4000 Directories and Files

Note – Once you install the Sun Crypto Accelerator 4000 hardware and software, you need to initialize the board with configuration and keystore information. See "Initializing the Board With vcaadm" on page 70 for information on how to initialize the board.

Removing the Sun Crypto Accelerator 4000 Software With the `remove` Script

There are three methods to remove the software: the `remove` script on the CD-ROM, the `/var/tmp/crypto_acc.remove` script on the server, or the `pkgrm` command. This section describes how to remove the software with the two removal scripts. For instructions on removing the software with the `pkgrm` command refer to "Removing the Software Without the remove Script" on page 26.

Use the `remove` script for software removal if you used the `install` script to install the software. Use the `/var/tmp/crypto_acc.remove` script if you installed the software manually. See "Removing the Software Without the remove Script" on page 26.

▼ To Remove the Software With the `remove` Script on the CD-ROM

- **Type the following with the Sun Crypto Accelerator 4000 CD-ROM inserted:**

```
# cd /cdrom/cdrom0/Sun_Crypto_Acc_4000_2_0/
# ./remove
```

▼ To Remove the Software With the `/var/tmp/crypto_acc.remove` Script

A log of this installation can be found at:

```
/var/tmp/crypto_acc.install.date
```

- **Type the following:**

```
# /var/tmp/crypto_acc.remove
```

Installing the Software Without the `install` Script

This section describes how to install the Sun Crypto Accelerator 4000 software manually without using the installation script (`/cdrom/cdrom0//Sun_Crypto_Acc_4000_2_0/install`) provided on the product CD.

The Sun Crypto Accelerator 4000 software is included on the product CD. You might need to download patches from the SunSolve web site (`http://sunsolve.sun.com`). See "Required Patches" on page 11 for more information.

Version 1.1 and 2.0 Software Contained on the CD-ROM

The Sun Crypto Accelerator 4000 Version 2.0 CD-ROM contains both Versions 1.1 and 2.0 of the software.

Caution – Version 1.1 is for Solaris 8 and 9. Version 2.0 is supported on Solaris 10 only.

The install script path to each version is as follows:

For Version 1.1:

```
/cdrom/cdrom0/Sun_Crypto_Acc_4000_1_1
```

For Version 2.0:

```
/cdrom/cdrom0/Sun_Crypto_Acc_4000_2_0
```

The respective installation scripts are located in these directories.

▼ To Install the Software Without the `install` Script

1. **The Sun Crypto Accelerator 4000 Version 1.x software should not be installed on Solaris 10. If Version 1.x exists on your Solaris 10 system, use the following command to remove all Version 1.x packages:**

```
# pkgrm SUNWkcl2o SUNWvcav SUNWvcar SUNWkcl2a SUNWkcl2u SUNWkcl2r
SUNWvcamn SUNWkcl2m SUNWkcl2i SUNWvcaa SUNWvcafw SUNWvcau
```

2. **Insert the Sun Crypto Accelerator 4000 CD into a CD-ROM drive that is connected to your system.**
   - If your system is running Sun Enterprise Volume Manager, the system should automatically mount the CD-ROM to the `/cdrom/cdrom0` directory.
   - If your system is not running Sun Enterprise Volume Manager, mount the CD-ROM as follows:

```
# mount -F hsfs -o ro /dev/dsk/c0t6d0s2 /cdrom
```

You see the following files and directories in the `/cdrom/cdrom0` directory.

TABLE 2-4 Files in the `/cdrom/cdrom0` Directory

File or Directory	Contents	
`Copyright`	U.S. copyright file	
`FR_Copyright`	French copyright file	
`install`	Script that installs the Sun Crypto Accelerator 4000 software	
`remove`	Script that removes the Sun Crypto Accelerator 4000 software	
`Docs`	*Sun Crypto Accelerator 4000 Board Version 2.0 Installation and User's Guide* *Sun Crypto Accelerator 4000 Board Release Notes*	
`Packages`	Contains the Sun Crypto Accelerator 4000 software packages:	
	`SUNWvcaa`	VCA administration
	`SUNWvcact`	VCA activation file
	`SUNWvcafw`	VCA firmware
	`SUNWvcamn`	VCA manual pages
	`SUNWvcaf`	VCA supplemental files
	`SUNWvcar`	VCA drivers
	`SUNWvcau`	VCA daemon

The required packages must be installed in a specific order and must be installed before installing any optional packages. Once the required packages are installed, you can install and remove the optional packages in any order.

3. **Install the required software packages by typing:**

```
# cd /cdrom/cdrom0/Sun_Crypto_Acc_4000_2_0/Packages
# pkgadd -d . SUNWvcact SUNWvcaf SUNWvcar SUNWvcau SUNWvcaa SUNWvcafw SUNWvcamn
```

4. **(Optional) To verify that the software is installed properly, run the `pkginfo` command.**

```
# pkginfo SUNWvcact SUNWvcaf SUNWvcar SUNWvcau SUNWvcaa SUNWvcafw SUNWvcamn
system      SUNWvcaa  VCA Crypto Accelerator/Gigabit Ethernet Admin
system      SUNWvcact VCA Crypto Accelerator/Gigabit Activation File
system      SUNWvcaf  VCA Crypto Accelerator/Gigabit Supplemental (usr)
system      SUNWvcafw VCA Crypto Accelerator/Gigabit Ethernet firmware
system      SUNWvcamn VCA Crypto Accelerator/Gigabit Ethernet Manual Pages
system      SUNWvcar  VCA Crypto Accelerator/Gigabit Ethernet Drivers
system      SUNWvcau  VCA Crypto Accelerator/Gigabit Ethernet Daemon
```

5. **(Optional) To ensure that the driver is attached, you can run the `prtdiag` command.**

```
# prtdiag -v
```

Refer to the `prtdiag`(1m) online manual pages.

6. **(Optional) Run the `modinfo` command to see that modules are loaded.**

```
# modinfo | grep Crypto
62   1317f62  20b1f 198   1  vca (VCA Crypto/Ethernet v1.232)
197  136d5d6   19b0 199   1  vcactl (VCA Crypto Control v1.39)
```

See "Directories and Files" on page 20 for a description and hierarchical diagram of the directories and files in the default installation.

Removing the Software Without the `remove` Script

Note – Remove the Sun Crypto Accelerator 4000 software manually only if you did not use the `install` script to install the software. If you installed the software with the install script, to remove the software, see "Removing the Sun Crypto Accelerator 4000 Software With the remove Script" on page 22.

If you have created keystores (see "Managing Keystores With vcaadm" on page 74), you must delete the keystore information that the Sun Crypto Accelerator 4000 board is configured with before removing the software. The `zeroize` command removes all key material, but does not delete the keystore files that are stored in the filesystem of the physical host in which the board is installed. See the "Performing a Software Zeroize on the Board" on page 91 for details on the `zeroize` command. To delete the keystore files stored in the system, become superuser and remove the keystore files. If you have not yet created any keystores, you can skip this procedure.

Caution – Do not delete a keystore that is currently in use or that is shared by other users and keystores. To free references to keystores, you might have to shut down the web server, administration server, or both

Caution – Before removing the Sun Crypto Accelerator 4000 software disable any web servers you have enabled for use with the Sun Crypto Accelerator 4000 board. Failure to do so leaves those web servers nonfunctional.

▼ To Remove the Software Without the `remove` Script

- **As superuser, use the `pkgrm` command to remove only the software packages you installed.**

Caution – Installed packages must be removed in the order shown. Failure to remove them in this order could result in dependency warnings and leave kernel modules loaded.

If you installed all the packages, you would remove them as follows:

```
# pkgrm SUNWvcamn SUNWvcafw SUNWvcaa SUNWvcau SUNWvcar SUNWvcaf
SUNWvcact
```

CHAPTER 3

Configuring Driver Parameters

This chapter describes how to configure the `vca` device driver parameters used by both the Sun Crypto Accelerator 4000 UTP and MMF Ethernet adapters. This chapter contains the following sections:

- "Ethernet Device Driver (vca) Parameters" on page 29
- "Setting vca Driver Parameters" on page 38
- "Cryptographic and Ethernet Driver Operating Statistics" on page 45
- "Network Configuration" on page 56
- "IPsec Hardware Acceleration Configuration" on page 58
- "Jumbo Frames Configuration" on page 59
- "Cryptographic Configuration" on page 60

Ethernet Device Driver (`vca`) Parameters

The `vca` device driver controls the Sun Crypto Accelerator 4000 UTP and MMF Ethernet devices. The `vca` driver is attached to the UNIX `pci` name property `pci108e,3de8` for the Sun Crypto Accelerator 4000 (`108e` is the vendor ID and `3de8` is the PCI device ID).

You can manually configure the `vca` device driver parameters to customize each Sun Crypto Accelerator 4000 device in your system. This section provides an overview of the capabilities of the Sun Crypto Accelerator 4000 Ethernet device used in the board, lists the available `vca` device driver parameters, and describes how to configure these parameters.

The Sun Crypto Accelerator 4000 Ethernet UTP and MMF PCI adapters are capable of the operating speeds of 10, 100, and 1000 in half or full duplex mode. By default, the `vca` device operates in autonegotiation mode with the remote end of the link (link partner) to select a common mode of operation for the `speed`, `duplex`, and

link-clock parameters. The link-clock parameter is applicable only if the board is operating at 1000 Mbps. The vca device can also be configured to operate in forced mode for each of these parameters.

Caution – To establish a proper link, both link partners must operate in either autonegotiation or forced mode for each of the speed, duplex, and link-clock (1000 Mbps only) related parameters. If both link partners are not operating in the same mode for each of these parameters, network errors will occur.

Network Driver Parameter Values and Definitions

TABLE 3-1 describes the parameters and settings for the vca device driver.

TABLE 3-1 vca Driver Parameter, Status, and Descriptions

Parameter	Status	Description
instance	Read and write	Device instance
adv-autoneg-cap	Read and write	Operational mode parameter
adv-1000fdx-cap	Read and write	Operational mode parameter (MMF adapter only)
adv-1000hdx-cap	Read and write	Operational mode parameter
adv-100fdx-cap	Read and write	Operational mode parameter (UTP adapter only)
adv-100hdx-cap	Read and write	Operational mode parameter (UTP adapter only)
adv-10fdx-cap	Read and write	Operational mode parameter (UTP adapter only)
adv-10hdx-cap	Read and write	Operational mode parameter (UTP adapter only)
adv-asmpause-cap	Read and write	Flow control parameter
adv-pause-cap	Read and write	Flow control parameter
pause-on-threshold	Read and write	Flow control parameter
pause-off-threshold	Read and write	Flow control parameter
link-master	Read and write	1 Gbps speed forced mode parameter
enable-ipg0	Read and write	Enable additional delay before transmitting a packet
ipg0	Read and write	Additional delay before transmitting a packet
ipg1	Read and write	Interpacket Gap parameter
ipg2	Read and write	Interpacket Gap parameter
rx-intr-pkts	Read and write	Receive interrupt blanking values

TABLE 3-1 vca Driver Parameter, Status, and Descriptions *(Continued)*

Parameter	Status	Description
rx-intr-time	Read and write	Receive interrupt blanking values
red-dv4to6k	Read and write	Random early detection and packet drop vectors
red-dv6to8k	Read and write	Random early detection and packet drop vectors
red-dv8to10k	Read and write	Random early detection and packet drop vectors
red-dv10to12k	Read and write	Random early detection and packet drop vectors
tx-dma-weight	Read and write	PCI Interface parameter
rx-dma-weight	Read and write	PCI Interface parameter
infinit-burst	Read and write	PCI Interface parameter
disable-64bit	Read and write	PCI Interface parameter
accept-jumbo	Read and write	Enable jumbo frames (9Kbyte)

Advertised Link Parameters

The following parameters determine the transmit and receive `speed` and `duplex` link parameters to be advertised by the `vca` driver to its link partner. TABLE 3-2 describes the operational mode parameters and their default values.

Note – If a parameter's initial setting is 0, it cannot be changed. If you try to change an initial setting of 0, it reverts back to 0. By default, these parameters are set to the capabilities of the `vca` device.

The Sun Crypto Accelerator 4000 UTP adapter advertised link parameters are different from those of the Sun Crypto Accelerator 4000 MMF adapter as shown in TABLE 3-2.

TABLE 3-2 Operational Mode Parameters

Parameter	Description	UTP Adapter	MMF Adapter
`adv-autoneg-cap`	Local interface capability advertised by the hardware 0 = Forced mode 1 = Autonegotiation (default)	X	X
`adv-1000fdx-cap`	Local interface capability advertised by the hardware 0 = Not 1000 Mbps full-duplex capable 1 = 1000 Mbps full-duplex capable (default)		X
`adv-1000hdx-cap`	Local interface capability advertised by the hardware 0 = Not 1000 Mbps half-duplex capable 1 = 1000 Mbps half-duplex capable (default)	X	X
`adv-100fdx-cap`	Local interface capability advertised by the hardware 0 = Not 100 Mbps full-duplex capable 1 = 100 Mbps full-duplex capable (default)	X	
`adv-100hdx-cap`	Local interface capability advertised by the hardware 0 = Not 100 Mbps half-duplex capable 1 = 100 Mbps half-duplex capable (default)	X	
`adv-10fdx-cap`	Local interface capability advertised by the hardware 0 = Not 10 Mbps full-duplex capable 1 = 10 Mbps full-duplex capable (default)	X	
`adv-10hdx-cap`	Local interface capability advertised by the hardware 0 = Not 10 Mbps half-duplex capable 1 = 10 Mbps half-duplex capable (default)	X	

If all of the parameters in TABLE 3-2 are set to 1, autonegotiation uses the highest speed possible. If all of these parameters are set to 0, you receive the following error message:

```
NOTICE: Last setting will leave vca0 with no link capabilities.
WARNING: vca0: Restoring previous setting.
```

Note – In this example, `vca0` is the Sun Crypto Accelerator 4000 device name where the string, `vca`, is used for every Sun Crypto Accelerator 4000 board. This string is always immediately followed by the device instance number of the board. Thus, the device instance number of the `vca0` board is `0`.

Flow Control Parameters

The vca device is capable of sourcing (transmitting) and terminating (receiving) pause frames conforming to the IEEE 802.3x Frame Based Link Level Flow Control Protocol. In response to received flow control frames, the vca device is capable of reducing its transmit rate. Alternately, the vca device is capable of sourcing flow control frames, requesting the link partner to reduce its transmit rate if the link partner supports this feature. By default, the driver advertises both transmit and receive pause capability during autonegotiation.

TABLE 3-3 provides flow control keywords and describes their function.

TABLE 3-3 Read-Write Flow Control Keyword Descriptions

Keyword	Description		
adv-asmpause-cap	Both the MMF and UTP adapters support asymmetric pause; therefore, the vca device can pause only in one direction. 0=Off (default) 1=On		
adv-pause-cap	This parameter has two meanings depending on the value of adv-asmpause-cap. (Default=0)		
	Parameter Value + adv-asmpause-cap=	**Parameter Value = adv-pause-cap=**	**Description**
	1	1 or 0	adv-pause-cap determines which direction pauses operate on.
	1	1	Pauses are received but are not transmitted.
	1	0	Pauses are transmitted but are not received.
	0	1	Pauses are sent and received.
	0	1 or 0	adv-pause-cap determines whether the pause capability is on or off.
pause-on-threshold	Defines the number of 64-byte blocks in the receive (RX) FIFO which causes the board to generate an XON-PAUSE frame.		
pause-off-threshold	Defines the number of 64-byte blocks in the RX FIFO which causes the board to generate an XOFF-PAUSE frame.		

Gigabit Forced Mode Parameter

For Gigabit links, this parameter determines the `link-master`. Generally, switches are enabled as a link master; in which case, this parameter can remain unchanged. If this is not the case, then the `link-master` parameter can be used to enable the `vca` device as a link master.

TABLE 3-4 Gigabit Forced Mode Parameter

Parameter	Description
`link-master`	When set to 1 this parameter enables master operation, assuming the link partner is a slave. When set to 0 this parameter enables slave operation, assuming the link partner is a master (default).

Interpacket Gap Parameters

The `vca` device supports the `enable-ipg0` programmable mode.

Before transmitting a packet with `enable-ipg0` enabled (default), the `vca` device adds an additional time delay. This delay, set by the `ipg0` parameter, is in addition to the delay set by the `ipg1` and `ipg2` parameters. The additional `ipg0` delay reduces collisions.

If `enable-ipg0` is disabled, the value of `ipg0` is ignored and no additional delay is set. Only the delays set by `ipg1` and `ipg2` are used. Disable `enable-ipg0` if other systems keep sending a large number of continuous packets. Systems that have `enable-ipg0` enabled might not have enough time on the network. You can add the additional delay by setting the `ipg0` parameter from 0 to 255, which is the media byte-time delay. TABLE 3-5 defines the `enable-ipg0` and `ipg0` parameters.

TABLE 3-5 Parameters Defining `enable-ipg0` and `ipg0`

Parameter	Values	Description
`enable-ipg0`	0 1	`enable-ipg0` enable `enable-ipg0` disable (Default=1)
`ipg0`	0 to 255	The additional time delay (or gap) before transmitting a packet (after receiving the packet) (Default=8)

The `vca` device supports the programmable interpacket gap (IPG) parameters `ipg1` and `ipg2`. The total IPG is the sum of `ipg1` and `ipg2`. The total IPG is 0.096 microseconds for the link speed of 1000 Mbps.

TABLE 3-6 lists the default values and allowable values for the IPG parameters.

TABLE 3-6 Read-Write Interpacket Gap Parameter Values and Descriptions

Parameter	Values (Byte-time)	Description
`ipg1`	0 to 255	Interpacket gap 1 (Default=8)
`ipg2`	0 to 255	Interpacket gap 2 (Default=4)

By default, the driver sets `ipg1` to 8-byte time and `ipg2` to 4-byte time, which are the standard values. (Byte time is the time it takes to transmit one byte on the link, with a link speed of 1000 Mbps.)

If your network has systems that use longer IPG (the sum of `ipg1` and `ipg2`), and if those machines seem to be slow in accessing the network, increase the values of `ipg1` and `ipg2` to match the longer IPGs of other machines.

Interrupt Parameters

TABLE 3-7 describes the receive interrupt blanking values.

TABLE 3-7 RX Blanking Register for Alias Read

Field Name	Values	Description
`rx-intr-pkts`	0 to 511	Interrupts after this number of packets have arrived since the last packet was serviced. A value of zero indicates no packet blanking (Default=3).
`rx-intr-time`	0 to 524287	Interrupts after 4.5 microseconds (Usecs) have elapsed since the last packet was serviced. A value of zero indicates no time blanking (Default=3).

Random Early Drop Parameters

These parameters provide the ability to drop packets based on the fullness of the receive FIFO. By default, this feature is disabled. When FIFO occupancy reaches a specific range, packets are dropped according to the preset probability. The probability should increase when the FIFO level increases. Control packets are never dropped and are not counted in the statistics.

TABLE 3-8 RX Random Early Detecting 8-Bit Vectors

Field Name	Values	Description
`red-dv4to6k`	0 to 255	Random early detection and packet drop vectors for a FIFO threshold greater than 4096 bytes and less than 6,144 bytes. Probability of drop can be programmed on a 12.5 percent granularity. For example, if bit 0 is set, the first packet out of every eight is dropped in this region (Default=0).
`red-dv6to8k`	0 to 255	Random early detection and packet drop vectors for a FIFO threshold greater than 6,144 bytes and less than 8,192 bytes. Probability of drop can be programmed on a 12.5 percent granularity. For example, if bit 8 is set, the first packet out of every eight is dropped in this region (Default=0).
`red-dv8to10k`	0 to 255	Random early detection and packet drop vectors for a FIFO threshold greater than 8,192 bytes and less than 10,240 bytes. Probability of drop can be programmed on a 12.5 percent granularity. For example, if bit 16 is set, the first packet out of every eight is dropped in this region (Default=0).
`red-dv10to12k`	0 to 255	Random early detection and packet drop vectors for a FIFO threshold greater than 10,240 bytes and less than 12,288 bytes. Probability of drop can be programmed on a 12.5 percent granularity. For example, if bit 24 is set, the first packet out of every eight is dropped in this region (Default=0).

PCI Bus Interface Parameters

These parameters enable you to modify PCI interface features to gain better PCI interperformance for a given application.

TABLE 3-9 PCI Bus Interface Parameters

Parameter	Description
`tx-dma-weight`	Determines the multiplication factor for accrediting the transmit (TX) side during a heavy round robin arbitration; the values are 0 to 3 (Default=0). Zero means no extra weight. The other values use an exponent of two for heavy traffic. For example, if `tx-dma-weight` = 0 and `rx-dma-weight` = 3, then as long as RX traffic is continuously arriving, the priority of RX traffic will be 8 times greater than the priority of TX traffic to access the PCI.
`rx-dma-weight`	Determines the multiplication factor for granting credit to the RX side during a weighted round robin arbitration. The values are 0 to 3 (Default=0).
`infinite-burst`	If enabled, this parameter allows the infinite burst capability to be used if the system supports infinite burst. The adapter does not free the bus until complete packets are transferred across the bus. The values are 0 or 1 (Default=0).
`disable-64bit`	Switches off 64-bit capability of the adapter. Note: for UltraSPARC® III based platforms, this parameter might be set to 1 by default. For UltraSPARC II based platforms, the default is 0. The values are 0 or 1 (Default=0, which enables 64-bit capability).

Setting vca Driver Parameters

You can set the vca device driver parameters in two ways:
- Using the ndd utility
- Using the vca.conf file

If you use the ndd utility, the parameters are valid only until you reboot the system. This method is good for testing parameter settings.

To set parameters so they remain in effect after you reboot the system, create a /kernel/drv/vca.conf file and add parameter values to this file when you need to set a particular parameter for a device in the system. See "To Set Driver Parameters Using a vca.conf File" on page 43 for details.

Setting Parameters Using the ndd Utility

Use the ndd utility to configure parameters that are valid until you reboot the system.

The following sections describe how you can use the vca driver and the ndd utility to modify (with the -set option) or display (without the -set option) the parameters for each vca device.

▼ To Specify Device Instances for the ndd Utility

Before you use the ndd utility to get or set a parameter for a vca device, you must specify the device instance for the utility.

1. **Check the /etc/path_to_inst file to identify the instance number associated with a particular device.**

```
# grep vca /etc/path_to_inst
"/pci@8,600000/network@1" 0 "vca"
"/pci@8,700000/network@1" 1 "vca"
```

Refer to the online manual pages for path_to_inst(4).

In this example, the three Sun Crypto Accelerator 4000 Ethernet instances are from the installed adapters. The instance numbers are 0 and 1.

2. **Use the instance number to select the device.**

```
# ndd -set /dev/vcaN
```

Note – In the examples in this user's guide, *N* represents the instance number of the device.

The device remains selected until you change the selection.

Noninteractive and Interactive Modes

You can use the `ndd` utility in two modes:

- Noninteractive
- Interactive

In noninteractive mode, you invoke the utility to execute a specific command. Once the command is executed, you exit the utility. In interactive mode, you can use the utility to get or set more than one parameter value. Refer to the `ndd`(1M) online manual page for more information.

Using the `ndd` *Utility in Noninteractive Mode*

This section describes how to modify and display parameter values.

- **To modify a parameter value, use the `-set` option.**

  If you invoke the `ndd` utility with the `-set` option, the utility passes *value*, which must be specified to the named `/dev/vca` driver instance, and assigns it to the parameter:

```
# ndd -set /dev/vcaN parameter value
```

When you change any `adv` parameter, a message similar to the following appears:

```
- link up 1000 Mbps half duplex
```

- **To display the value of a parameter, specify the parameter name and omit the value.**

  When you omit the `-set` option, a query operation is assumed and the utility queries the named driver instance, retrieves the value associated with the specified parameter, and prints it:

```
# ndd /dev/vcaN parameter
```

Note – In this example, *N* is the instance number of the `vca` device. This number should reflect the instance number of the board for which you are running the `kstat` command.

Using the `ndd` *Utility in Interactive Mode*

- **To modify a parameter value in interactive mode, specify `ndd /dev/vca`*N*, as shown below.**

  The `ndd` utility then prompts you for the name of the parameter:

```
# ndd /dev/vcaN
name to get/set? (Enter the parameter name or ? to view all
parameters)
```

Note – In this example, *N* is the instance number of the `vca` device. This number should reflect the instance number of the board for which you are running the `kstat` command.

After typing the parameter name, the `ndd` utility prompts you for the parameter value (see TABLE 3-1 through TABLE 3-9).

- **To list all the parameters supported by the `vca` driver, type `ndd /dev/vca`*N*.**
  (See TABLE 3-1 through TABLE 3-9 for parameter descriptions.)

```
# ndd /dev/vcaN \?
?                               (read only)
instance                        (read and write)
adv-autoneg-cap                 (read and write)
adv-1000fdx-cap                 (read and write)
adv-1000hdx-cap                 (read and write)
adv-100T4-cap                   (read and write)
adv-100fdx-cap                  (read and write)
adv-100hdx-cap                  (read and write)
adv-10fdx-cap                   (read and write)
adv-10hdx-cap                   (read and write)
adv-asmpause-cap                (read and write)
adv-pause-cap                   (read and write)
link-master                     (read and write)
use-int-xcvr                    (read and write)
enable-ipg0                     (read and write)
ipg0                            (read and write)
ipg1                            (read and write)
ipg2                            (read and write)
pause-on-threshold              (read and write)
pause-off-threshold             (read and write)
rx-enter-pkts                   (read and write)
rx-intr-time                    (read and write)
red-dv4to6k                     (read and write)
red-dv6to8k                     (read and write)
red-dv8to10k                    (read and write)
red-dv10to12k                   (read and write)
tx-dma-weight                   (read and write)
rx-dma-weight                   (read and write)
infinite-burst                  (read and write)
disable-64bit                   (read and write)
accept-jumbo                    (read and write)
hp-prog-number                  (read and write)
link_status                     (read only)
link_mode                       (read only)
link_speed                      (read only)
```

Note – In this example, *N* is the instance number of the `vca` device. This number should reflect the instance number of the board for which you are running the `kstat` command.

Setting Autonegotiation or Forced Mode

The following link parameters can be set to operate in either autonegotiation or forced mode:

- `speed`
- `duplex`
- `link-clock`

By default, autonegotiation mode is enabled for these link parameters. When either of these parameters are in autonegotiation mode, the `vca` device communicates with the link partner to negotiate a compatible value and flow control capability. When a value other than `auto` is set for either of these parameters, no negotiation occurs and the link parameter is configured in forced mode. In forced mode, the value for the `speed` parameter must match between link partners.

▼ To Disable Autonegotiation Mode

If your network equipment does not support autonegotiation, or if you want to force your network `speed`, `duplex`, or `link-clock` parameters, you can disable the autonegotiation mode on the `vca` device.

1. **Set the following driver parameters to the values that are described in the documentation delivered with your link partner device (for example, a switch):**
   - `adv-1000fdx-cap`
   - `adv-1000hdx-cap`
   - `adv-100fdx-cap`
   - `adv-100hdx-cap`
   - `adv-10fdx-cap`
   - `adv-10hdx-cap`
   - `adv-asmpause-cap`
   - `adv-pause-cap`

   See TABLE 3-2 for the descriptions and possible values of these parameters.

2. **Set the `adv-autoneg-cap` parameter to `0`.**

```
# ndd -set /dev/vcaN adv-autoneg-cap 0
```

When you change any `ndd` link parameter, a message similar to the following appears:

```
link up 1000 Mbps half duplex
```

Setting Parameters Using the vca.conf File

You can also specify the driver parameter properties by adding entries to the vca.conf file in the /kernel/drv directory. The parameter names are the same names listed in "Network Driver Parameter Values and Definitions" on page 30.

Caution – Do not remove any of the default entries in the /kernel/drv/vca.conf file.

The online manual pages for prtconf(1) and driver.conf(4) include additional details. The next procedure shows an example of setting parameters in a vca.conf file.

Variables defined in this section apply to known devices in the system. To set a variable for a Sun Crypto Accelerator 4000 board with the vca.conf file, you must know the following three pieces of information for the device: device name, device parent, and device unit address.

▼ To Set Driver Parameters Using a vca.conf File

1. **Obtain the hardware path names for the vca devices in the device tree.**

   a. **Check the /etc/driver_aliases file to identify the name associated with a particular device.**

```
# grep vca /etc/driver_aliases
vca "pci108e,3de8"
```

      In this example, the device name associated with the Sun Crypto Accelerator 4000 software driver (vca) is "pci108e,3de8".

   b. **Locate the device parent name and device unit address in the /etc/path_to_inst file.**

      Refer to the online manual pages for path_to_inst(4).

```
# grep vca /etc/path_to_inst
"/pci@8,600000/network@1" 0 "vca"
"/pci@8,700000/network@1" 1 "vca"
```

In this example, there are three columns of output: Device path name, instance number, and software driver name.

The device path name in the first line of this example is "/pci@8,600000/network@1". Device path names are made up of three parts: Device parent name, device node name, and device unit address. See TABLE 3-10.

TABLE 3-10 Device Path Name

Entire Device Path Name	Parent Name Portion	Node Name Portion	Unit Address Portion
"/pci@8,600000/network@1"	/pci@8,600000	network	1
"/pci@8,700000/network@1"	/pci@8,700000	network	1

To identify a PCI device unambiguously in the `vca.conf` file, use the entire device path name (parent name, node name, and the unit address) for the device. Refer to the `pci`(4) online manual page for more information about the PCI device specification.

2. **Set the parameters for the `vca` devices in the `/kernel/drv/vca.conf` file.**

   In the following entry, the `adv-autoneg-cap` parameter is disabled for a particular Sun Crypto Accelerator 4000 Ethernet device.

```
name="pci108e,3de8" parent="/pci@8,700000" unit-address="1" adv-autoneg-cap=0;
```

3. **Save the `vca.conf` file.**
4. **Save and close all files and programs, and exit the windowing system.**
5. **Shut down and reboot the system.**

Setting Parameters for All Sun Crypto Accelerator 4000 `vca` Devices With the `vca.conf` File

If you omit the device path name (parent name, node name, and the unit address), the variable is set for all instances of all Sun Crypto Accelerator 4000 Ethernet devices.

▼ To Set Parameters for All vca Devices With the vca.conf File

1. **Add a line in the `vca.conf` file to change the value of a parameter for all instances by entering** *parameter=value;*.

   The following example sets the `adv-autoneg-cap` parameter to `1` for all instances of all Sun Crypto Accelerator 4000 Ethernet devices:

```
adv-autoneg-cap=1;
```

Example vca.conf File

The following is an example `vca.conf` file:

```
#
# Copyright 2005 Sun Microsystems, Inc.  All rights reserved.
# Use is subject to license terms.
#
#ident  "@(#)vca.conf   1.5     04/08/10 SMI"

#
# Make sure we forceattach the driver so it attaches very early in the
# boot process.
#
ddi-forceattach=1;
```

Cryptographic and Ethernet Driver Operating Statistics

This section describes the statistics presented by the `kstat`(1M) command.

Cryptographic Driver Statistics

TABLE 3-11 describes the cryptographic driver statistics.

TABLE 3-11 Cryptographic Driver Statistics

Parameter	Description	Stable or Unstable
`vs-mode`	The values are `FIPS`, `standard`, or `unitialized`. `FIPS` indicates that the board is in FIPS mode. `standard` indicates that the board is not in FIPS mode. `unitialized` indicates that the board is not initialized.	Stable
`vs-status`	The values are `ready`, `faulted`, or `failsafe`. `ready` indicates that the board is operating normally. `faulted` indicates that the board is not operating. `failsafe` indicates failsafe mode, which is the original factory state of the board.	Stable

Ethernet Driver Statistics

TABLE 3-12 describes the Ethernet driver statistics.

TABLE 3-12 Ethernet Driver Statistics

Parameter	Description	Stable or Unstable
`ipackets`	Number of inbound packets.	Stable
`ipackets64`	64-bit version of `ipackets`.	Stable
`ierrors`	Total packets received that could not be processed because they contained errors (long).	Stable
`opackets`	Total packets requested to be transmitted on the interface.	Stable
`opackets64`	Total packets requested to be transmitted on the interface (64-bit).	Stable
`oerrors`	Total packets that were not successfully transmitted because of errors (long).	Stable
`rbytes`	Total bytes successfully received on the interface.	Stable
`rbytes64`	Total bytes successfully received on the interface (64-bit).	Stable
`obytes`	Total bytes requested to be transmitted on the interface.	Stable

TABLE 3-12 Ethernet Driver Statistics *(Continued)*

Parameter	Description	Stable or Unstable
`obytes64`	Total bytes requested to be transmitted on the interface (64-bit).	Stable
`multircv`	Multicast packets successfully received, including group and functional addresses (long).	Stable
`multixmt`	Multicast packets requested to be transmitted, including group and functional addresses (long).	Stable
`brdcstrcv`	Broadcast packets successfully received (long).	Stable
`brdcstxmt`	Broadcast packets requested to be transmitted (long).	Stable
`norcvbuf`	Times that a valid incoming packet was known to be discarded because a buffer could not be allocated for the receive packet (long).	Stable
`noxmtbuf`	Packets discarded on output because transmit buffer was busy, or no buffer could be allocated for transmit (long).	Stable

TABLE 3-13 describes the transmit and receive MAC counters.

TABLE 3-13 TX and RX MAC Counters

Parameter	Description	Stable or Unstable
`tx-collisions`	16-bit loadable counter increments for every frame transmission attempt that resulted in a collision.	Stable
`tx-first-collisions`	16-bit loadable counter increments for every frame transmission that experienced a collision on the first attempt, but was successfully transmitted on the second attempt.	Unstable
`tx-excessive-collisions`	16-bit loadable counter increments for every frame transmission that has exceeded the Attempts Limit.	Unstable

TABLE 3-13 TX and RX MAC Counters *(Continued)*

Parameter	Description	Stable or Unstable
`tx-late-collisions`	16-bit loadable counter increments for every frame transmission that has experienced a collision. The parameter indicates the number of frames that the TxMAC has dropped due to collisions that occurred after transmitting at least the Minimum Frame Size number of bytes. Usually this is an indication that at least one station on the network violates the maximum allowed span of the network.	Unstable
`tx-defer-timer`	16-bit loadable timer increments when the TxMAC is deferring to traffic on the network while it is attempting to transmit a frame. The time base for the timer is the media byte clock divided by 256.	Unstable
`tx-peak-attempts`	8-bit register indicates the highest number of consecutive collisions per successfully transmitted frame, that have occurred since this register was last read. The maximum value that this register can attain is 255. A maskable interrupt is generated to the software if the number of consecutive collisions per successfully transmitted frame exceeds 255. This register is automatically cleared at 0 after it is read.	Unstable
`tx-underrun`	16-bit loadable counter increments after a valid frame has been received from the network.	Unstable
`rx-length-err`	16-bit loadable counter increments after a frame, whose length is greater than the value that was programmed in the Maximum Frame Size Register, has been received from the network.	Unstable

TABLE 3-13 TX and RX MAC Counters *(Continued)*

Parameter	Description	Stable or Unstable
`rx-alignment-err`	16-bit loadable counter increments when an alignment error is detected in a receive frame. An alignment error is reported when a receive frame fails the cyclic redundancy checksum (CRC) checking algorithm, *and* the frame contains a noninteger number of bytes (that is, the frame size in bits is not equal to zero).	Unstable
`rx-crc-err`	16-bit loadable counter increments when a receive frame fails the CRC checking algorithm, *and* the frame contains an integer number of bytes (that is, the frame size in bits modulo 8 is equal to zero).	Unstable
`rx-code-violations`	16-bit loadable counter increments when an Rx_Err indication is generated by the XCVR over the MII, while a frame is being received. This indication is generated by the transceiver when it detects an invalid code in the received data stream. A receive code violation is not counted as an FCS or an Alignment error.	Unstable
`rx-overflows`	Number of Ethernet frames dropped due to lack of resources.	Unstable
`rx-no-buf`	Number of times the hardware cannot receive data because there is no more receive buffer space.	Unstable
`rx-no-comp-wb`	Number of times the hardware cannot post completion entries for received data.	Unstable
`rx-len-mismatch`	Number of received frames where the asserted length does not match the actual frame length.	Unstable

The following Ethernet properties (TABLE 3-14) are derived from the intersection of device capabilities and the link partner capabilities.

TABLE 3-14 Current Ethernet Link Properties

Parameter	Description	Stable or Unstable
`ifspeed`	1000, 100, or 10 Mbps	Stable
`link-duplex`	0=half, 1=full	Stable
`link-pause`	Current pause setting for the link, see "Flow Control Parameters" on page 33	Stable
`link-asmpause`	Current pause setting for the link, see "Flow Control Parameters" on page 33	Stable
`link-up`	1=up, 0=down	Stable
`link-status`	1=up, 0=down	Stable
`xcvr-inuse`	Type of transceiver in use: 1=internal MII, 2=external MII, 3=external PCS	Stable

TABLE 3-15 describes the read-only Media Independent Interface (MII) capabilities. These parameters define the capabilities of the hardware. The Gigabit Media Independent Interface (GMII) supports all of the following capabilities.

TABLE 3-15 Read-Only `vca` Device Capabilities

Parameter	Description	Stable or Unstable
`cap-autoneg`	0 = Not capable of autonegotiation 1 = Autonegotiation capable	Stable
`cap-1000fdx`	Local interface full-duplex capability 0 = Not 1000 Mbps full-duplex capable 1 = 1000 Mbps full-duplex capable	Stable
`cap-1000hdx`	Local interface half-duplex capability 0 = Not 1000 Mbps half-duplex capable 1 = 1000 Mbps half-duplex capable	Stable
`cap-100fdx`	Local interface full-duplex capability 0 = Not 100 Mbps full-duplex capable 1 = 100 Mbps full-duplex capable	Stable
`cap-100hdx`	Local interface half-duplex capability 0 = Not 100 Mbps half-duplex capable 1 = 100 Mbps half-duplex capable	Stable
`cap-10fdx`	Local interface full-duplex capability 0 = Not 10 Mbps full-duplex capable 1 = 10 Mbps full-duplex capable	Stable

TABLE 3-15 Read-Only `vca` Device Capabilities *(Continued)*

Parameter	Description	Stable or Unstable
`cap-10hdx`	Local interface half-duplex capability 0 = Not 10 Mbps half-duplex capable 1 = 10 Mbps half-duplex capable	Stable
`cap-asm-pause`	Local interface flow control capability 0 = Not asymmetric pause capable 1 = Asymmetric pause (from the local device) capable (See "Flow Control Parameters" on page 33)	Stable
`cap-pause`	Local interface flow control capability 0 = Not Symmetric pause capable 1 = Symmetric pause capable (See "Flow Control Parameters" on page 33)	Stable

Reporting the Link Partner Capabilities

TABLE 3-16 describes the read-only link partner capabilities.

TABLE 3-16 Read-Only Link Partner Capabilities

Parameter	Description	Stable or Unstable
`lp-cap-autoneg`	0 = No autonegotiation 1 = Autonegotiation	Stable
`lp-cap-1000fdx`	0 = No 1000 Mbps full-duplex transmission 1 = 1000 Mbps full-duplex	Stable
`lp-cap-1000hdx`	0 = No 1000 Mbps half-duplex transmission 1 = 1000 Mbps half-duplex	Stable
`lp-cap-100fdx`	0 = No 100 Mbps full-duplex transmission 1 = 100 Mbps full-duplex	Stable
`lp-cap-100hdx`	0 = No 100 Mbps half-duplex transmission 1 = 100 Mbps half-duplex	Stable
`lp-cap-10fdx`	0 = No 10 Mbps full-duplex transmission 1 = 10 Mbps full-duplex	Stable

TABLE 3-16 Read-Only Link Partner Capabilities *(Continued)*

Parameter	Description	Stable or Unstable
`lp-cap-10hdx`	0 = No 10 Mbps half-duplex transmission 1 = 10 Mbps half-duplex	Stable
`lp-cap-asm-pause`	0 = Not asymmetric pause capable 1 = Asymmetric pause towards link partner capability (See "Flow Control Parameters" on page 33)	Stable
`lp-cap-pause`	0 = Not symmetric pause capable 1 = Symmetric pause capable (See "Flow Control Parameters" on page 33)	Stable

If the link partner is not capable of autonegotiation (when `lp-cap-autoneg` is 0), the remaining information described in TABLE 3-16 is not relevant and the parameter value is 0.

If the link partner is capable of autonegotiation (when `lp-cap-autoneg` is 1), then the speed and mode information is displayed when you use autonegotiation and the link partner capabilities.

TABLE 3-17 describes the driver-specific parameters.

TABLE 3-17 Driver-Specific Parameters

Parameter	Description	Stable or Unstable
`lb-mode`	Copy of the loopback mode the device is in, if any.	Unstable
`promisc`	When enabled, the device is in promiscuous mode. When disabled, the device is not in promiscuous mode.	Unstable
`mac-mtu`	The MAClayer MTU size. Normally this is set to 1518. With jumbo frames enabled, this is set to 9194.	Unstable
Ethernet Transmit Counters		
`tx-wsrv`	Count of the number of times the transmit ring is full.	Unstable
`tx-msgdup-fail`	Attempt to duplicate packet failure.	Unstable
`tx-allocb-fail`	Attempt to allocate memory failure.	Unstable
`tx-queue0`	Number of packets queued for transmission on the first hardware transmit queue.	Unstable
`tx-jumbo-pkt`	Number of jumbo frame packets transmitted.	Unstable

TABLE 3-17 Driver-Specific Parameters *(Continued)*

Parameter	Description	Stable or Unstable
`tx-queue1`	Number of packets queued for transmission on the second hardware transmit queue.	Unstable
`tx-queue2`	Number of packets queued for transmission on the third hardware transmit queue.	Unstable
`tx-queue3`	Number of packets queued for transmission on the fourth hardware transmit queue.	Unstable
Ethernet Receive Counters		
`rx-hdr-pkts`	Number of packets received that were less than 256 bytes.	Unstable
`rx-mtu-pkts`	Number of packets received that were greater than 256 bytes and less than 1514 bytes.	Unstable
`rx-split-pkts`	Number of packets that were split across two pages.	Unstable
`rx-jumbo-pkts`	Number of jumbo frame packets received.	Unstable
`rx-nocanput`	Number of packets dropped due to failures on delivery to the IP stack.	Unstable
`rx-msgdup-fail`	Number of packets that could not be duplicated.	Unstable
`rx-allocb-fail`	Number of block allocation failures.	Unstable
`rx-new-pages`	Number of pages that were replaced during reception.	Unstable
`rx-new-hdr-pages`	Number of pages that were filled with packets less than 256 bytes that were replaced during reception.	Unstable
`rx-new-mtu-pages`	Number of pages that were filled with those packets greater than 256 bytes and less than 1514 that got replaced during reception.	Unstable
`rx-new-nxt-pages`	Number of pages that contained packets that were split across pages that were replaced during reception.	Unstable
`rx-page-alloc-fail`	Number of page allocation failures.	Unstable
`rx-mtu-drops`	Number of times a whole page of packets greater than 256 bytes and less than 1514 was dropped because the driver was unable to map a new one to replace the page.	Unstable

TABLE 3-17 Driver-Specific Parameters *(Continued)*

Parameter	Description	Stable or Unstable
`rx-hdr-drops`	Number of times a whole page of packets less than 256 bytes was dropped because the driver was unable to map a new one to replace the page.	Unstable
`rx-nxt-drops`	Number of times a page with a split packet was dropped because the driver was unable to map a new one to replace the page.	Unstable
`rx-rel-flow`	Number of times the driver was told to release a flow.	Unstable
Ethernet PCI Properties		
`rev-id`	Revision ID of the Sun Crypto Accelerator 4000 Ethernet device useful for recognition of a device being used in the field.	Unstable
`pci-err`	S um of all PCI errors.	Unstable
`pci-rta-err`	Number of target aborts received.	Unstable
`pci-rma-err`	Number of master aborts received.	Unstable
`pci-parity-err`	Number of PCI parity errors detected.	Unstable
`pci-drto-err`	Number of times the delayed transaction retry time-out was reached.	Unstable
`dma-mode`	Used by the Sun Crypto Accelerator 4000 driver (`vca`).	Unstable

▼ To Check Link Partner Settings

- **As superuser, type the `kstat vca:`*N* command:**

```
# kstat vca:N
module: vca                             instance: 0
name:   vca0                            class:    misc
```

Where *N* is the instance number of the `vca` device. This number should reflect the instance number of the board for which you are running the `kstat` command.

IPsec In-Line Acceleration Statistics

TABLE 3-18 describes the kernel statistics that are incremented when the board is configured for in-line IPsec hardware acceleration. See "Enabling In-Line IPsec Acceleration" on page 59 for instructions on how to configure the board to use the in-line IPsec configuration.

TABLE 3-18 Cryptographic Driver Statistics for In-Line IPsec Acceleration

Parameter	Description	Stable or Unstable
`ipsec_ierrrors`	Total IPsec packets received that could not be processed because they contained errors (long)	Stable
`ipsec_ipackets`	Number of inbound IPsec packets	Stable
`ipsec_ipackets64`	Number of inbound IPsec packets (64-bit)	Stable
`ipsec_obytes`	Total IPsec bytes requested to be transmitted on the interface	Stable
`ipsec_obytes64`	Total IPsec bytes requested to be transmitted on the interface (64-bit)	Stable
`ipsec_oerrors`	Total IPsec packets that were not successfully transmitted because of errors (long)	Stable
`ipsec_opackets`	Total IPsec packets requested to be transmitted on the interface	Stable
`ipsec_opackets64`	Total IPsec packets requested to be transmitted on the interface (64-bit)	Stable
`ipsec_rbytes`	Total IPsec bytes successfully received on the interface	Stable
`ipsec_rbytes64`	Total IPsec bytes successfully received on the interface (64-bit)	Stable
`sadb_cache_misses`	Number of firmware cache misses	Stable
`sadb_cache_overflows`	Number of firmware cache overflows	Stable
`sadb_entries`	Number of entries in the SADB driver	Stable
`sadb_operations`	Number of SADB operations sent from Solaris IPsec to the driver	Stable
`ipsec_status`	Inline IPsec configuration status: `unconfigured` = not configured in the `vca.conf` file `configured` = configured in the `vca.conf` file `enabled` = enabled by IPsec	

Note – The IPsec kernel statistics listed in TABLE 3-18 are only incremented for IPsec packets that are actually processed in-line by the hardware. Receive packets of less than 256 bytes are not processed in-line and the IPsec kernel statistics will not be incremented for these packets. These kernel statistics also do not apply to out-of-band IPsec traffic (See "IPsec Hardware Acceleration Configuration" on page 58). If `snoop` is enabled, these counters are not incremented. Out-of-band packets will increment the regular network kernel statistics and any applicable cryptographic statistics, that is, `3desbytes` and `3desjobs`.

Network Configuration

This section describes how to edit the network host files after the adapter has been installed on your system.

Configuring the Network Host Files

After installing the driver software, you must create a `hostname.vca`*N* file for the adapter's Ethernet interface. Note that in the file name `hostname.vca`*N*, *N* corresponds to the instance number of the `vca` interface you plan to use. You must also create both an IP address and a host name for its Ethernet interface in the `/etc/hosts` file.

1. **Locate the correct `vca` interfaces and instance numbers in the `/etc/path_to_inst` file.**

   Refer to the online manual pages for `path_to_inst`(4).

```
# grep vca /etc/path_to_inst
"/pci@8,600000/network@1" 0 "vca"
```

   The instance number in this example is `0`.

2. **Use the `ifconfig`(1M) command to set up the adapter's `vca` interface.**

   Use the `ifconfig` command to assign an IP address to the network interface. Type the following at the command line, replacing *ip-address* with the adapter's IP address:

```
# ifconfig vcaN plumb ip-address up
```

   Refer to the `ifconfig`(1M) man page and the Solaris documentation for more information.

   - If you want a setup that will remain the same after you reboot, create an `/etc/`*hostname*`.vca`*N* file, where *N* corresponds to the instance number of the `vca` interface you plan to use.

     To use the `vca` interface of the example shown in Step 1, create an `/etc/`*hostname*`.vca`*N* file, where *N* corresponds to the instance number of the device which is 0 in this example. If the instance number were 1, the file name would be `/etc/`*hostname*`.vca1`.

   - Do not create an `/etc/`*hostname*`.vca`*N* file for a Sun Crypto Accelerator 4000 interface you plan to leave unused.
   - The `/etc/`*hostname*`.vca`*N* file must contain the host name for the appropriate `vca` interface.
   - The host name must have an IP address and must be listed in the `/etc/hosts` file.
   - The host name must be different from any other host name of any other interface, for example, `/etc/`*hostname*`.vca0` and `/etc/`*hostname*`.vca1` cannot share the same host name.

   The following example shows the `/etc/`*hostname*`.vca`*N* file required for a system named `zardoz` that has a Sun Crypto Accelerator 4000 board (`zardoz-11`).

```
# cat /etc/hostname.hme0
zardoz
# cat /etc/hostname.vca0
zardoz-11
```

3. **Create an appropriate entry in the `/etc/hosts` file for each active `vca` interface.**

   For example:

```
# cat /etc/hosts
#
# Internet host table
#
127.0.0.1     localhost
129.144.10.57 zardoz    loghost
129.144.11.83 zardoz-11
```

IPsec Hardware Acceleration Configuration

The board has two configurations of IPsec hardware acceleration: in-line and out-of-band. Both configurations accelerate IPsec cryptographic operations. However, because each method offers different advantages, overall system requirements should be evaluated to determine the appropriate configuration.

Out-of-band is the default IPsec configuration, and is optimized for performance on a multiprocessor system. This configuration offloads DES and 3DES cryptographic functions to the board, and is the preferred configuration on a multiprocessor system for which host processing power is not an issue.

In-line IPsec configuration augments out-of-band functionality with authentication support (MD5 and SHA1), and offloads portions of the host packet processing to the board. By handling the additional packet processing, the board significantly reduces host CPU usage.

Note – Out-of-band might provide greater IPsec throughput than in-line on multiprocessor systems that only require DES or 3DES encryption algorithms.

Enabling Out-of-Band IPsec Acceleration

Out-of-band is the default configuration for the board. No IPsec configuration or tuning is required to use the board for out-of-band IPsec acceleration in Solaris 9. You simply install the Sun Crypto Accelerator 4000 packages and reboot.

Enabling In-Line IPsec Acceleration

To configure in-line acceleration, you must change configuration files in both the Solaris software and the `vca` driver.

▼ To Enable In-Line IPsec Hardware Acceleration

1. **Enable in-line acceleration in the `vca` driver by adding the following entry to the `/kernel/drv/vca.conf` configuration file:**

```
inline-ipsec=1;
```

For the change in the `/kernel/drv/vca.conf` file to take effect, you must reboot the system.

Once in-line acceleration has been enabled, the Solaris software IPsec policies can be configured for the interface with the standard IPsec configuration procedures. For information on configuring IPsec policies in Solaris refer to the *IPsec and IKE Administration Guide* available at: `http://docs.sun.com`

In-line acceleration can be used to accelerate both AH and ESP algorithms; however, multiple nested transforms (including AH+ESP) cannot be performed on the board. If multiple transforms are applied, only the outermost transform is performed in-line. The remaining transforms are performed by the Solaris IPsec configuration.

When the board is configured for IPsec in-line acceleration, additional statistics presented by the `kstat`(1M) command will be incremented. See TABLE 3-18 for descriptions of the IPsec in-line acceleration `kstat` statistics.

Jumbo Frames Configuration

Enable jumbo frame support by adding the following entry to the `/kernel/drv/vca.conf` file:

```
accept-jumbo=1;
```

For the change in the `/kernel/drv/vca.conf` file to take effect, you must reboot the system.

When jumbo frames are enabled, the `mac-mtu` kstat variable is set to 9194 instead the default of 1518.

Cryptographic Configuration

Enabling AES Encryption/Decryption

To enable AES, the following must be added to `/kernel/drv/vca.conf`:

```
enable-aes=1;
```

For the change in the `/kernel/drv/vca.conf` file to take effect, you must reboot the system.

This entry enables the board to perform AES cryptographic operations. Please note that AES is implemented in firmware and is intended only to be used for applications that require AES with sensitive keys. For AES operations, the board will not perform nearly as well as the host system.

CHAPTER 4

Administering the Sun Crypto Accelerator 4000 Board

This chapter provides an overview of administering the board with the `vcaadm`, `vcad`, and `vcadiag` utilities. The following sections are included:

- "Using the vcaadm Utility" on page 61
- "Managing the vcad Service" on page 92
- "Managing the vcad Service" on page 92
- "Using the vcadiag Utility" on page 95
- "Assigning Different MAC Addresses to Multiple Boards Installed in the Same Server" on page 98

Note – `vcaadm` 2.0 administers boards running either Version 1.1 or 2.0 firmware and is fully backwards compatible.

Using the `vcaadm` Utility

The `vcaadm` utility offers a command-line interface to the Sun Crypto Accelerator 4000 board. Only users designated as security officers are permitted to use the `vcaadm` utility. When you first connect to a Sun Crypto Accelerator 4000 board with `vcaadm`, you are prompted to create an initial security officer and password.

To access the `vcaadm` utility easily, place the Sun Crypto Accelerator 4000 tools directory in your search path, for example:

```
$ PATH=$PATH:/opt/SUNWconn/bin
$ export PATH
```

The `vcaadm` command-line syntax is:

- `vcaadm [-H]`
- `vcaadm [-V]`
- `vcaadm [-y] [-h` *hostname*`] [-p` *port*`] [-d` *device*`] [-f` *filename*`] [-l` *pkcs11-library*`] [-t` *pkcs11-token*`]`
- `vcaadm [-y] [-h` *hostname*`] [-p` *port*`] [-d` *device*`] [-s` *sec-officer*`] [-l` *pkcs11-library*`] [-t` *pkcs11-token*`]` *command*

Note – When using the `-d` attribute, `vca`*N* is the board's device name, where the *N* corresponds to the Sun Crypto Accelerator 4000 device instance number.

TABLE 4-1 shows the options for the `vcaadm` utility.

TABLE 4-1 `vcaadm` Options

Option	Meaning
`-H`	Displays help files for `vcaadm` commands and exits.
`-d vca`*N*	Connects to the Sun Crypto Accelerator 4000 board that has *N* as the driver instance number. For example, `-d vca1` connects to device `vca1` where `vca` is a string in the board's device name and 1 is the instance number of the device. This value defaults to `vca0` and must be in the form of `vca`*N*, where *N* corresponds to the device instance number.
`-f` *filename*	Interprets one or more commands from *filename* and exits.
`-h` *hostname*	Connects to the Sun Crypto Accelerator 4000 board on *hostname*. The value for *host* can be a host name or an IP address, and defaults to the loopback address.
`-l` *pkcs11-library*	Indicates which PKCS#11 library implementation to use. By default this is `/usr/lib/libpkcs11.so`.
`-p` *port*	Connects to the Sun Crypto Accelerator 4000 board on *port*. The value for *port* defaults to 6870.
`-s` *sec-officer*	Logs in as a security officer named *sec-officer.*
`-t` *pkcs11-token*	Indicates which PKCS#11 token to use. By default `vcaadm` will select the first token it finds that can do all the cryptographic opertaions `vcaadm` needs.
`-V`	Displays version information for `vcaadm`.
`-y`	Forces a yes answer to any command that would normally prompt for a confirmation.

Note – The name *sec-officer* is used throughout this user's guide as an example security officer name.

Modes of Operation

`vcaadm` can run in one of three modes. These modes differ mainly in how commands are passed into `vcaadm`. The three modes are Single-Command mode, File mode, and Interactive mode.

Note – To use `vcaadm`, you must authenticate as security officer. How often you need to authenticate as security officer is determined by which operating mode you are using.

Single-Command Mode

In Single-Command mode, you must authenticate as security officer for every command. Once the command is executed, you are logged out of `vcaadm`.

When entering commands in Single-Command mode, you specify the command to be run after all the command-line switches are specified. For example, in Single-Command mode, the following command would show all the users in a given keystore and return the user to the command shell prompt.

```
$ vcaadm show user
Security Officer Name: sec-officer
Security Officer Password:
```

The following command performs a login as the security officer, *sec-officer*, and creates the user *web-admin* in the keystore.

```
$ vcaadm -s sec-officer create user web-admin
Security Officer Password:
Enter new user password:
Confirm password:
User web-admin created successfully.
```

Note – The first password is for the security officer, followed by the password and confirmation for the new user *web-admin*.

All output from Single-Command mode goes to the standard output stream. This output can be redirected using standard UNIX shell-based methods.

File Mode

In File mode, you must authenticate as security officer for every file you run. You are logged out of `vcaadm` after the commands in the command file are executed.

To enter commands in File mode, you specify a file from which `vcaadm` reads one or more commands. The file must be ASCII text, consisting of one command per line. Begin each comment with a pound sign (#) character. If the File mode option is set, `vcaadm` ignores any command-line arguments after the last option. The following example runs the commands in the `deluser.scr` file and answers all prompts in the affirmative:

```
$ vcaadm -f deluser.scr -y
```

Interactive Mode

In Interactive mode, you must authenticate as security officer every time you connect to a board. This is the default operating mode for `vcaadm`. To log out of `vcaadm` in Interactive mode, use the `logout` command. Refer to "Logging In and Out With vcaadm" on page 64.

Interactive mode presents the user with an interface similar to `ftp`(1), where commands can be entered one at a time. The `-y` option is not supported in Interactive mode.

Logging In and Out With `vcaadm`

When you use `vcaadm` from the command line and specify *host*, *port*, and *device* using the `-h`, `-p`, and `-d` attributes respectively, you are immediately prompted to log in as security officer if a successful network connection was made.

The `vcaadm` utility establishes an encrypted network connection (channel) between the `vcaadm` application and the Sun Crypto Accelerator 4000 firmware running on a specific board.

During setup of the encrypted channel, boards identify themselves by their hardware Ethernet address and an RSA public key. A trust database (`$HOME/.vcaadm/trustdb`) is created the first time `vcaadm` connects to a board. This file contains all of the boards that are currently trusted by the security officer.

Note – The Sun Crypto Accelerator 4000 board is preprogrammed with a unique remote access key for connecting to an uninitialized board. The fingerprint for this remote access key is printed on the board and must be verified when logging into a board for the first time to ensure a secure channel is established with the correct board.

Logging In to a Board With `vcaadm`

If the security officer connects to a new board, `vcaadm` notifies the security officer and prompts with the following options:

```
1. Abort the connection
2. Trust the board for this session only
3. Replace the trusted key with the new key
```

If the security officer connects to a board that has a remote access key that has been changed, `vcaadm` will notify the security officer and prompt the following three options:

```
1. Abort the connection
2. Trust the connection one time only (no changes to trust
   database)
3. Replace the old public key bound to this hardware ethernet
   address with the new public key
```

Logging In to a New Board

Note – The remaining examples in this chapter were created with the Interactive mode of `vcaadm`.

When connecting to a new board, `vcaadm` must create a new entry in the trust database. The following is an example of logging in to a new board.

```
# vcaadm -h hostname
Warning: MAC ID and Public Key Not Found
-----------------------------------------------------
The MAC ID and public key presented by this board were
not found in your trust database.

MAC ID: 08:00:20:EE:EE:EE
Key Fingerprint: 29FC-7A54-4014-442F-7FD9-5FEA-8411-CFB4
-----------------------------------------------------
Please select an action:

1. Abort this connection
2. Trust the board for this session only
3. Trust the board for all future sessions

Your Choice -->
```

Logging In to a Board With a Changed Remote Access Key

When connecting to a board that has a changed remote access key, `vcaadm` must change the entry corresponding to the board in the trust database. The following is an example of logging in to a board with a changed remote access key.

```
# vcaadm -h hostname
Warning: Public Key Conflict
-----------------------------------------------------
The public key presented by the board you are connecting
to is different than the public key that is trusted for
this MAC ID.

MAC ID: 08:00:20:EE:EE:EE
New Key Fingerprint: 29FC-7A54-4014-442F-7FD9-5FEA-8411-CFB4
Trusted Key Fingerprint: A508-38D1-FED8-8103-7ACC-0D19-C9C9-11F2
-----------------------------------------------------
Please select an action:

1. Abort this connection
2. Trust the board for this session only
3. Replace the current trusted key with the new key

Your Choice -->
```

`vcaadm` *Prompt*

The `vcaadm` prompt in Interactive mode is displayed as follows:

```
vcaadm{vcaN@hostname, sec-officer}> command
```

The following table describes the `vcaadm` prompt variables:

TABLE 4-2 `vcaadm` Prompt Variable Definitions

Prompt Variable	Definition
`vca`*N*	`vca` is a string that represents the Sun Crypto Accelerator 4000 board. *N* is the device instance number (unit address) that is in the device path name of the board. Refer to "To Set Driver Parameters Using a vca.conf File" on page 43 for details on retrieving this number for a device.
hostname	The name of the host for which the Sun Crypto Accelerator 4000 board is physically connected. *hostname* may be replaced with the physical host's IP address.
sec-officer	The name of the security officer that is currently logged in to the board.

Logging Out of a Board With `vcaadm`

If you are working in Interactive mode, you might want to disconnect from one board and connect to another board without completely exiting `vcaadm`. To disconnect from a board and log out, but remain in Interactive mode, use the `logout` command:

```
vcaadm{vcaN@hostname, sec-officer}> logout
vcaadm>
```

In the previous example, notice that the vcaadm> prompt no longer displays the device instance number, hostname, or security officer name. To log in to another device, type the connect command with the following optional parameters.

TABLE 4-3 connect Command Optional Parameters

Parameter	Meaning
dev vca*N*	Connect to the Sun Crypto Accelerator 4000 board with the driver instance number of *N*. For example -d vca1 connects to the device vca1; this defaults to device vca0.
host *hostname*	Connect to the Sun Crypto Accelerator 4000 board on *hostname* (defaults to the loopback address). *hostname* may be replaced with the physical host's IP address.
port *port*	Connect to the Sun Crypto Accelerator 4000 board on port *port* (defaults to 6870).

Example:

```
vcaadm{vcaN@hostname, sec-officer}> logout
vcaadm> connect host hostname dev vca2
Security Officer Login: sec-officer
Security Officer Password:
vcaadm{vca2@hostname, sec-officer}>
```

vcaadm does not let you issue the connect command if you are already connected to a Sun Crypto Accelerator 4000 board. You must first log out and then issue the connect command.

Each new connection causes vcaadm and the target Sun Crypto Accelerator 4000 firmware to renegotiate new session keys to protect the administrative data that is sent.

Entering Commands With vcaadm

The vcaadm utility has a command language that must be used to interact with the Sun Crypto Accelerator 4000 board. Commands are entered using all or part of a command (enough to uniquely identify that command from any other command). Entering sh instead of show would work, but re is ambiguous because it could be reset or rekey.

The following example shows entering commands using entire words:

```
vcaadm{vcaN@hostname, sec-officer}> show user
User                                    Status
-------------------------------------------------
web-admin                               Enabled
Tom                                     Enabled
-------------------------------------------------
```

The same information can be obtained in the previous example using partial words as commands, such as `sh us`.

An ambiguous command produces an explanatory response:

```
vcaadm{vcaN@hostname, sec-officer}> re
Ambiguous command: re
```

Getting Help for Commands

`vcaadm` has built-in help functions. To get help, you must enter a question mark (?) character following the command you want more help on. If an entire command is entered and a "?" exists anywhere on the line, you get the syntax for the command, for example:

```
vcaadm{vcaN@hostname, sec-officer}> create ?
Sub-Command                     Description
-------------------------------------------------
so                              Create a new security officer
user                            Create a new user

vcaadm{vcaN@hostname, sec-officer}> create user ?
Usage: create user [<username>]

vcaadm{vcaN@hostname, sec-officer}> set ?
Sub-Command                     Description
-------------------------------------------------
lock                            Lock master key (Prevents key backup)
multiadmin                      Configure Multi-Admin mode
passreq                         Set password security level
password                        Change password for security officer
timeout                         Set firmware auto-logout timer
```

You can also enter a question mark at the `vcaadm` prompt to see a list of all of the `vcaadm` commands and their description, for example:

```
vcaadm{vcaN@hostname, sec-officer}> ?
Sub-Command                         Description
-----------------------------------------------------
backup                              Backup master key
connect                             Begin admin session with firmware
create                              Create users and accounts
delete                              Delete users and accounts
diagnostics                         Run diagnostic tests
disable                             Disable a user
enable                              Enable a user
exit                                Exit vcaadm
loadfw                              Load new firmware
logout                              Logout current session
quit                                Exit vcaadm
rekey                               Generate new system keys
reset                               Reset the hardware
set                                 Set operating parameters
show                                Show system settings
zeroize                             Delete all keys and reset board
```

Note – When not in `vcaadm` Interactive mode, the "?" character could be interpreted by the shell in which you are working. In this case, ensure that you use the command shell escape character before the question mark. For example in the C shell, you would need to type \?

Quitting the `vcaadm` Utility in Interactive Mode

Two commands allow you to exit from `vcaadm`: `quit` and `exit`. The Ctrl-D key sequence also exits from `vcaadm`.

Initializing the Board With `vcaadm`

The first step in configuring a Sun Crypto Accelerator 4000 board is to initialize it. When you initialize a board it is necessary to create a keystore. (See "Concepts and Terminology" on page 114.) When you first connect to a Sun Crypto Accelerator 4000 board with `vcaadm`, you are prompted to initialize the board with a new keystore or an existing keystore, which is stored in a backup file. `vcaadm` prompts you for all the required information for either type of board initialization.

▼ To Initialize the Board With a New Keystore

1. **Initialize the board with the `vcaadm` command.**
   - If the board is installed locally, enter `vcaadm` at the system prompt.
   - If the system is remote, enter `vcaadm -h` *hostname*
2. **Enter 2 then 1 as shown in the following example:**

```
# vcaadm -h hostname
Please select an action:

1. Abort this connection
2. Trust the board for this session only.
3. Replace the trusted key with the new key.

Your Choice --> 2
This board is uninitialized.
You will now initialize the board. You may either
completely initialize the board and start with a new
keystore or initialize the board to use an existing
keystore, providing a backup file in the process.

1. Initialize the board with a new keystore
2. Initialize the board to use an existing keystore

Your Choice (0 to exit) --> 1
```

3. **Create a keystore name (See "Naming Requirements" on page 75.):**

```
Keystore Name: keystore-name
```

4. **Select FIPS 140-2 mode or non-FIPS mode.**

   When in FIPS mode the board is FIPS 140-2, level 3 compliant. FIPS 140-2 is a Federal Information Processing standard that requires tamper-resistance and a high level of data integrity and security. Refer to the FIPS 140-2 document located at `http://www.nist.gov/dmvp`

```
Run in FIPS 140-2 mode? (Y/Yes/N/No) [No]: y
```

5. **Create an initial security officer name and password (See "Naming Requirements" on page 75.):**

```
Initial Security Officer Name: sec-officer
Initial Security Officer Password:
Confirm Password:
```

Note – Before an essential parameter is changed or deleted, or before a command is executed that may have drastic consequences, `vcaadm` prompts you to enter `Y`, `Yes`, `N`, or `No` to confirm. These values are not case sensitive; the default is `No`.

---

6. **Verify the configuration information:**

```
Board initialization parameters:
-----------------------------------------------------
Initial Security Officer Name: sec-officer
Keystore name: keystore-name
Run in FIPS 140-2 Mode: Yes
-----------------------------------------------------

Is this correct? (Y/Yes/N/No) [No]: y
Initializing crypto accelerator board... This may take a few
minutes...Done.
```

Initializing the Board to Use an Existing Keystore

If you are adding multiple boards to a single keystore, you might want to initialize all of the boards to use the same keystore information. In addition, you might want to restore a Sun Crypto Accelerator 4000 board to the original keystore configuration. This section describes how to initialize a board to use an existing keystore which is stored in a backup file.

You must first create a backup file of an existing board configuration before performing this procedure. Creating and restoring a backup file requires a password to encrypt and decrypt the data in the backup file. (See "Backing Up the Master Key" on page 80.)

Note – To initialize a board from a previous backup, both the master key backup file and the encrypted keystore file are required. The encrypted keystore file must exist in the keystore directory (`/etc/opt/SUNWconn/vca/keydata/` by default) prior to initializing the board to use that keystore. If the keystore file is not present, it must be restored from a previous archive.

▼ To Initialize the Board to Use an Existing Keystore

1. **Initialize the board with the `vcaadm` command.**
   - If the board is installed locally, enter `vcaadm` at the system prompt.
   - If the system is remote, enter `vcaadm -h` *hostname*

1. **Enter 2 as shown in the following example:**

```
# vcaadm -h hostname
This board is uninitialized.
You will now initialize the board.  You may
either completely initialize the board and
start with a new keystore or restore the board
using a backup file.

1. Initialize the board with a new keystore
2. Initialize the board to use an existing keystore

Your Choice (0 to exit) --> 2
```

2. **Enter the path and password to the backup file:**

Note – If the backup file was created in Multi-Admin mode, authentication is required by multiple security officers assigned the Multi-Admin role.

```
Enter the path to the backup file: /tmp/board-backup
Password for restore file:
```

3. **Verify the configuration information:**

```
Board restore parameters:
-----------------------------------------------------------------
Path to backup file: /tmp/board-backup
Keystore name: sca4000-keystore
Requires Multi-Admin auth: No
-----------------------------------------------------------------

Is this correct? (Y/Yes/N/No) [No]: y
Restoring data to crypto accelerator board...
```

Managing Keystores With `vcaadm`

A keystore is a repository for key material. Associated with a keystore are security officers and users. Keystores not only provide storage, but a means for key objects to be owned by user accounts. This enables keys to be hidden from applications that do not authenticate as the owner. Keystores have three components:

- **Key objects –** Long-term keys that are stored for applications such as the Sun ONE Web Server.
- **User accounts –** These accounts provide applications a means to authenticate and access specific keys.
- **Security officer accounts –** These accounts provide access to key management functions through `vcaadm`.

Note – A single Sun Crypto Accelerator 4000 board must have exactly one keystore. Multiple boards can be configured to collectively work with the same keystore to provide additional performance and fault-tolerance.

Naming Requirements

Security officer names, user names, and keystore names must meet the following requirements:

TABLE 4-4 Security Officer Name, User Name, and Keystore Name Requirements

Name Requirement	Description
Minimum length	At least one character
Maximum length	63 characters for user names and 32 characters for keystore names
Valid characters	Alphanumeric, underscore (_), dash (-), and dot (.)
First character	Must be alphabetic

Password Requirements

Password requirements vary based on the current `set passreq` setting (`low`, `med`, or `high`).

Setting the Password Requirements

Use the `set passreq` command to set the password requirements for the Sun Crypto Accelerator 4000 board. This command sets the password character requirements for any password prompted by `vcaadm`. There are three settings for password requirements, as shown in the following table:

TABLE 4-5 Password Requirement Settings

Password Setting	Requirements
`low`	Does not require any password restrictions. This is the default while the board is in non-FIPS mode.
`med`	Requires six characters minimum: Three characters must be alphabetic and one character must be nonalphabetic. This is the default setting while the board is in FIPS 140-2 mode and is the minimum password requirement allowed in FIPS 140-2 mode.
`high`	Requires eight characters minimum: Three characters must be alphabetic, and one character must be nonalphabetic. This is not a default setting and must be configured manually.

To change the password requirements, enter the `set passreq` command followed by `low`, `med`, or `high`. The following commands set the password requirements for a Sun Crypto Accelerator 4000 board to `high`:

```
vcaadm{vcaN@hostname, sec-officer}> set passreq high

vcaadm{vcaN@hostname, sec-officer}> set passreq
Password security level (low/med/high): high
```

Populating a Keystore With Security Officers

There may be more than one security officer for a keystore. Security officer names are known only within the domain of the Sun Crypto Accelerator 4000 board and do not need to be identical to any user name on the host system.

When creating a security officer, the name is an optional parameter on the command line. If the security officer name is omitted, `vcaadm` prompts you for the name. (See "Naming Requirements" on page 75.)

```
vcaadm{vcaN@hostname, sec-officer}> create so Alice
Enter new security officer password:
Confirm password:
Security Officer Alice created successfully.

vcaadm{vcaN@hostname, sec-officer}> create so
New security officer name: Bob
Enter new security officer password:
Confirm password:
Security Officer Bob created successfully.
```

Populating a Keystore With Users

These user names are known only within the domain of the Sun Crypto Accelerator 4000 board and do not need to be identical to the UNIX user name for the web server process.

When creating a user, the user name is an optional parameter on the command line. If the user name is omitted, `vcaadm` prompts you for the user name. (See "Naming Requirements" on page 75.)

```
vcaadm{vcaN@hostname, sec-officer}> create user web-admin
Enter new user password:
Confirm password:
User web-admin created successfully.

vcaadm{vcaN@hostname, sec-officer}> create user
New user name: Tom
Enter new user password:
Confirm password:
User Tom created successfully.
```

Users must use this password when authenticating during a web server startup.

Caution – Users must remember their password so they can access their keys. There is no way to retrieve a lost password.

Note – The user account is logged out if no commands are entered for more than five minutes. This is a tunable option. See "Setting the Auto-Logout Time" on page 88 for details.

Listing Users and Security Officers

To list users or security officers associated with a keystore, enter the `show user` or `show so` commands.

```
vcaadm{vcaN@hostname, sec-officer}> show user
User                                          Status
-----------------------------------------------------
web-admin                                     Enabled
Tom                                           Enabled
-----------------------------------------------------


vcaadm{vcaN@hostname, sec-officer}> show so
Security Officer                              Multi-Admin Role
-------------------------------------------------------------------
sec-officer1                                  Enabled
sec-officer2                                  Enabled
sec-officer3                                  Enabled
sec-officer4                                  Disabled
-------------------------------------------------------------------
```

Changing Passwords

Only security officer passwords may be changed with `vcaadm`. Security officers can change are their own password. Use the `set password` command to change security officer passwords.

```
vcaadm{vcaN@hostname, sec-officer}> set password
Enter new security officer password:
Confirm password:
Security Officer password has been set.
```

User passwords may be changed through the PKCS#11 interface with the Sun ONE Web Server `modutil` utility. Refer to the Sun ONE Web Server documentation for details.

Enabling or Disabling Users

Note – Security officers cannot be disabled. Once a security officer is created, it is enabled until it is deleted.

By default each user is created in the enabled state. Users may be disabled. Disabled users cannot access their key material with the PKCS#11 interface. Enabling a disabled user restores access to all of that user's key material.

When enabling or disabling a user, the user name is an optional parameter on the command line. If the user name is omitted, `vcaadm` prompts you for the user name.

```
vcaadm{vcaN@hostname, sec-officer}> disable user Tom
User Tom disabled.
vcaadm{vcaN@hostname, sec-officer}> disable user
User name: web-admin
User web-admin disabled.
```

To disable a user account, enter the `disable user` command.

To enable an account, enter the `enable user` command.

```
vcaadm{vcaN@hostname, sec-officer}> enable user Tom
User Tom enabled.

vcaadm{vcaN@hostname, sec-officer}> enable user
User name: web-admin
User web-admin enabled.
```

Deleting Users

Issue the `delete user` command and specify the user to be deleted. When deleting a user, the user name is an optional parameter on the command line. If the user name is omitted, `vcaadm` prompts you for the user name.

```
vcaadm{vcaN@hostname, sec-officer}> delete user web-admin
Delete user web-admin? (Y/Yes/N/No) [No]: y
User web-admin deleted successfully.

vcaadm{vcaN@hostname, sec-officer}> delete user
User name: Tom
Delete user Tom? (Y/Yes/N/No) [No]: y
User Tom deleted successfully.
```

Deleting Security Officers

Issue the `delete so` command and specify the security officer to be deleted. When deleting a security officer, the security officer name is an optional parameter on the command line. If the security officer name is omitted, `vcaadm` prompts you for the security officer name.

```
vcaadm{vcaN@hostname, sec-officer}> delete so Bob
Delete Security Officer Bob? (Y/Yes/N/No) [No]: y
Security Officer Bob deleted.

vcaadm{vcaN@hostname, sec-officer}> delete so
Security Officer name: Alice
Delete Security Officer Alice? (Y/Yes/N/No) [No]: y
Security Officer Alice deleted.
```

Backing Up the Master Key

Keystores are stored on the disk and encrypted in a master key. This master key is stored in the Sun Crypto Accelerator 4000 firmware and can be backed up by a security officer.

To back up the master key, use the `backup` command. The `backup` command requires a path name to a backup file where the backup will be stored. This path name can be placed on the command line or if omitted, `vcaadm` prompts you for the path name.

A password must be set for the backup data. This password is used to encrypt the master key in the backup file.

Note – If the following command is executed in Multi-Admin mode, authentication is required by multiple security officers assigned the Multi-Admin role.

```
vcaadm{vcaN@hostname, sec-officer}> backup /opt/SUNWconn/vca/backups/bkup.data
Enter a password to protect the data:
Confirm password:
Backup to /opt/SUNWconn/vca/backups/bkup.data successful.
```

Caution – Choose a password that is very difficult to guess when making backup files, because this password protects the master key for your keystore. You must also remember the password you enter. Without the password, you cannot access the master key backup file. There is no way to retrieve the data protected by a lost password.

Note – To initialize a board from a previous backup, both the master key backup file and the encrypted keystore file are required. After performing a master key backup, both the master key backup file and the current keystore file must be archived for future initialization operations. By default the encrypted keystore file is located in the `/etc/opt/SUNWconn/vca/keydata/` directory.

Locking the Keystore to Prevent Backups

A site might have a strict security policy that does not permit the master key for a Sun Crypto Accelerator 4000 board to leave the hardware. This can be enforced using the `set lock` command.

Caution – Once this command is issued, all attempts to back up the master key will fail. This lock persists even if the master key is rekeyed. The only way to clear this setting is to zeroize the Sun Crypto Accelerator 4000 board with the `zeroize` command. (See "Performing a Software Zeroize on the Board" on page 91.)

```
vcaadm{vcaN@hostname, sec-officer}> set lock
WARNING: Issuing this command will lock the
         master key.  You will be unable to back
         up your master key once this command
         is issued.  Once set, the only way to
         remove this lock is to zeroize the board.
Do you wish to lock the master key? (Y/Yes/N/No) [No]: y
The master key is now locked.
```

Multi-Admin Authentication

The `vcaadm` utility includes a special mode of operation called Multi-Admin mode. In this mode, certain commands require multiple security officers to authenticate and approve the command before it can complete successfully. Security officers must be in the Multi-Admin role before they can authenticate Multi-Admin commands.

When a Multi-Admin command is issued, no other general administration on the board can take place until either the command times out, is cancelled by the security officer who started the command, or the command completes successfully. A timeout from 1 to 15 minutes must be set at or before Multi-Admin mode is enabled. See "Setting a Multi-Admin Command Timeout" on page 83 for more information. Also security officers must set the number of Multi-Admin role members required to authenticate any Multi-Admin command.

When a Multi-Admin command is initiated, the `vcaadm` session from which it is started will wait until one of three conditions occur: The command completes successfully, the command fails, or the command times out. Other Multi-Admin role members will log in to the device using their respective `vcaadm` sessions. During Multi- Admin mode commands, these role members can only authenticate the command in progress. If the initiating security officer's `vcaadm` session terminates unexpectedly, the security officer can log back into the device and cancel the command. Otherwise, the board cannot be administered normally until the command times out.

Managing Multi-Admin Mode With `vcaadm`

This section describes how to configure and manage Multi-Admin mode with the `vcaadm` utility. First, you must identify your security officers and place them in the multi-admin role. You must have enough security officers in that role to satisfy the minimum number set with the `set multiadmin minauth` command. If the number of multi-admin role members is below the minimum threshold, you cannot enable multi-admin mode.

Assigning Security Officers the Multi-Admin Role

To assign the Multi-Admin role to a security officer, use the `enable authmember` *so-name* command. If executed in Multi-Admin mode, this command requires authentication by multiple security officers assigned the Multi-Admin role.

The following command assigns a security officer the Multi-Admin role.

```
vcaadm{vcaN@hostname, sec-officer}> enable authmember sec-officer
Added multi-admin role to Security Officer sec-officer.
```

Removing a Security Officer From the Multi-Admin Role

To remove a security officer from the Multi-Admin role, use the `disable authmember` *so-name* command. If executed in Multi-Admin mode, this command requires authentication by multiple security officers assigned the Multi-Admin role.

```
vcaadm{vcaN@hostname, sec-officer}> disable authmember sec-officer
Removed multi-admin role from Security Officer rew.
```

This command removes security officers from the Multi-Admin role only if they are in addition to the minimum required. This command exits if only a minimum number of security officers are assigned the Multi-Admin role. See "Setting the Minimum Number of Security Officers Required to Authenticate Multi-Admin Commands" on page 83.

Setting the Minimum Number of Security Officers Required to Authenticate Multi-Admin Commands

To set the minimum number of required security officers to authenticate Multi-Admin commands, use the `set multiadmin minauth` *minimum-role-members* command. The *minimum-role-members* value must at least two and less than or equal to the total number of security officers on the system. In addition, if Multi-Admin mode is already enabled the new value cannot exceed the number of members with the Multi-Admin role. If executed in Multi-Admin mode, this command requires authentication by multiple security officers assigned the Multi-Admin role.

The following command sets the minimum number of required security officers to authenticate Multi-Admin commands.

```
vcaadm{vcaN@hostname, sec-officer}> set multiadmin minauth 3
Multi-admin mode now requires 3 security officers to authenticate.
```

Setting a Multi-Admin Command Timeout

To change the timeout for commands that require Multi-Admin mode authentication, use the `set multiadmin timeout` *minutes* command. The *minutes* value must be between 1 and 1440 minutes (1 day). If a value larger than 1440 is specified, the value will be set to 1440. If executed in Multi-Admin mode, this command requires authentication by multiple security officers assigned the Multi-Admin role.

The following command changes the timeout for commands that require Multi-Admin mode authentication.

```
vcaadm{vcaN@hostname, sec-officer}> set multiadmin timeout 3
New multi-admin timeout value is 3 minutes.
```

Enabling Multi-Admin Mode

To enable Multi-Admin mode, use the `enable multiadmin` command. When enabled, certain commands require multiple security officers to authenticate before the command can complete successfully. When this command is executed, the security officer is presented with the current Multi-Admin mode settings and is given the opportunity to change these settings before the command completes. This command does not accept the `-y` (yes to all) flag.

The following command enables Multi-Admin mode.

```
vcaadm{vcaN@hostname, sec-officer}> enable multiadmin
WARNING: This command will place the device in multi-
         admin mode.  This mode will require multiple
         security officers to authenticate for certain
         commands to be executed.

Enable Multi-Admin Mode? (Y/Yes/N/No) [No]: y

Multi-Admin mode parameters:
---------------------------------------------------------------
Minimum number of admins: 3
Multi-Admin command timeout: 3 minutes
---------------------------------------------------------------

Is this correct? (Y/Yes/N/No) [No]: y
The board is now in multi-admin mode.
```

Disabling Multi-Admin Mode

To disable Multi-Admin mode, use the `disable multiadmin` command. This command requires authentication by multiple security officers assigned the Multi-Admin role.

The following command disables Multi-Admin mode.

```
vcaadm{vcaN@hostname, sec-officer}> disable multiadmin
```

Adding Additional Security Officers to the Multi-Admin Role

With the minimum number of required security officers set to three, adding additional security officers requires the authorization of three different security officers, including the initiating security officer, to authenticate before this command can complete.

Execute the following command on the initiating security officer's `vcaadm` session.

```
vcaadm{vca0@localhost, sec-officer1}> enable authmember sec-officer4
NOTICE: Please wait while the other required 2 administrators
        authenticate this command.  This command will time out
        in 3 minutes.

Update: Authenticated security officers: sec-officer1
Update: Authenticated security officers: sec-officer1 sec-officer3
Update: Authenticated security officers: sec-officer1 sec-officer3 sec-officer2
Added multi-admin role to Security Officer sec-officer4.
```

Other security officers must log in from their respective `vcaadm` sessions.

```
# vcaadm
Security Officer Login: sec-officer3
Security Officer Password:
NOTICE: A Multi-Admin command is currently in progress.
        You are a member of the Multi-Admin role and
        may approve this command.
Command: enable authmember sec-officer4
Initiating SO: sec-officer1

Authorize this command? (Y/Yes/N/No) [No]: y
Authorization successful
```

```
# vcaadm
Security Officer Login: sec-officer2
Security Officer Password:
NOTICE: A Multi-Admin command is currently in progress.
        You are a member of the Multi-Admin role and
        may approve this command.
Command: enable authmember sec-officer4
Initiating SO: sec-officer1

Authorize this command? (Y/Yes/N/No) [No]: y
Authorization successful
```

Cancelling a Multi-Admin Command Originated by the Initiating Security Officer

In this example the following command is cancelled. This command must be entered on the initiating security officer's `vcaadm` session.

```
vcaadm{vca0@localhost, sec-officer1}> disable authmember sec-officer4
NOTICE: Please wait while the other required 2 administrators
        authenticate this command.  This command will time out
        in 3 minutes.

Update: Authenticated security officers: sec-officer1
```

To cancel the command, the initiating security officer must either close the current `vcaadm` session or log in with a second `vcaadm` session.

```
# vcaadm
Security Officer Login: sec-officer1
Security Officer Password:
NOTICE: A Multi-Admin command is currently in progress.
        Since you are the admin that initiated this
        command, you have the option of cancelling it.
        If you choose not to cancel the command, you
        will be logged out and the board will continue
        with the command.

Cancel this command? (Y/Yes/N/No) [No]: y
Authorization successful
```

If the `vcaadm` session from which the command was initiated is still active, the following message is displayed.

```
Failed to remove role from Security Officer sec-officer4: Command cancelled
```

Allowing a Multi-Admin Command to Time Out

In this example, the following command is issued by the security officer.

```
vcaadm{vca0@localhost, sec-officer1}> disable authmember sec-officer4
WARNING: Issuing this command will remove the
         multi-admin role from this security
         officer.  Once complete, this security
         officer will not be able to validate multi-
         admin commands.

Proceed with change? (Y/Yes/N/No) [No]: y
NOTICE: Please wait while the other required 2 administrators
        authenticate this command.  This command will time out
        in 3 minutes.

Update: Authenticated security officers: sec-officer1
Update: Authenticated security officers: sec-officer1 sec-officer2
Failed to remove role from Security Officer sec-officer4: Multi-Admin command
timeout
```

Attempting to Log in to a Board During a Multi-Admin Command as a Security Officer not in the Multi-Admin Role

Log in by non-multi-admin security officer.

```
# vcaadm
Security Officer Login: new-sec-officer
Security Officer Password:
You have authenticated successfully but this board is
currently waiting for all needed approvals for a
Multi-Admin mode command.  Since you are not a member
of the Multi-Admin role, you will not be able to
administer this board until this command has completed.

Connection closed.
```

Attempting to Execute a Multi-Admin Command Without Multi-Admin Role Permissions

In this example, the following command is executed by a security officer without multi-admin role permissions.

```
vcaadm{vca0@localhost, new-so}> disable multiadmin
WARNING: Issuing this command will take the board
         out of multi-admin mode and return it to the
         single-administrator mode of authentication.

Proceed with change? (Y/Yes/N/No) [No]: y
Failed disabling Multi-admin mode: Unauthorized command
```

Managing Boards With `vcaadm`

This section describes how to manage Sun Crypto Accelerator 4000 boards with the `vcaadm` utility.

Setting the Auto-Logout Time

To customize the amount of time before a security officer is automatically logged out of the board, use the `set timeout` command. To change the auto-logout time, enter the `set timeout` command followed by the number of minutes before a security officer is automatically logged out. A value of 0 disables the automatic logout feature. The maximum delay is 1,440 minutes (1 day). A newly initialized board defaults to 5 minutes.

The following command changes the auto-logout time for a security officer to 10 minutes:

```
vcaadm{vcaN@hostname, sec-officer}> set timeout 10
```

Displaying Board Status

To get the current status of a Sun Crypto Accelerator 4000 board, issue the `show status` command. This command displays the hardware and firmware versions for that board, the MAC address of the network interface, the status (Up versus Down, speed, duplex, and so on) of the network interface, and the keystore name and ID.

```
vcaadm{vcaN@hostname, sec-officer}> show status
Board Status
----------------------------------------------------------------
Version Info:
* Hardware Version: 2.0
* Firmware Version: 2.0
* Bootstrap Firmware Version: VCA Crypto Accelerator 1.0  March 2003
* Current Firmware Version: VCA Crypto Accelerator 2.0  March 2003

Network Settings:
* MAC Address: 00:03:ba:0e:9a:32
* Interface Information: Link down

Keystore Info:
* Keystore Name: sca4000-keystore
* Keystore ID: 8327aec84176e959
* Keystore Lock: Disabled
* FIPS 140-2 Mode: Disabled

Security Settings:
* Login Session Timeout (in minutes): 5
* Password Policy Security Level: LOW
* Number of Master Key Backups: 0
* Multiadmin Mode: Disabled
* Minimum Number of Authenticators: 2
* Multiadmin Timeout: 5 Minutes
----------------------------------------------------------------
```

Loading New Firmware

You can update the firmware for the Sun Crypto Accelerator 4000 board as new features are added. To load firmware, issue the `loadfw` command and provide a path to the firmware file.

A successful update of the firmware requires you to manually reset the board with the `reset` command. When you reset the board, the currently logged in security officer is logged out.

```
vcaadm{vcaN@hostname, sec-officer}> loadfw /opt/SUNWconn/cryptov2/firmware/sca4000fw
Security Officer Login: sec-officer
Security Officer Password:
WARNING: This command will load new firmware onto the
          the target device.  You must issue a reset
          command and log back into the target device in
          order to use the new firmware.

Proceed with firmware update? (Y/Yes/N/No) [No]: y
```

Resetting the Board

In certain situations, it might be necessary to reset the board. To do this, you must issue the `reset` command. You are asked if this is what you wish to do. Resetting a Sun Crypto Accelerator 4000 board might temporarily cease the acceleration of cryptography on the system unless there are other active Sun Crypto Accelerator 4000 boards able to take over the load. Also, this command automatically logs you out of `vcaadm`, so you must reconnect to the device by logging back into `vcaadm` if you wish to continue administering it.

```
vcaadm{vcaN@hostname, sec-officer}> reset
WARNING: Issuing this command will reset the
         the board and close this connection.

Proceed with reset? (Y/Yes/N/No) [No]: y
Reset successful.
```

Rekeying the Board

If your security policy changes, you might want to use new keys as the master key or remote access key. The `rekey` command enables you to regenerate either of these keys, or both.

Rekeying the master key also causes the keystore to be reencrypted under the new key, and invalidates older backed up master key files with the new keystore file. Make a backup of the master key whenever it is rekeyed. If you have multiple Sun Crypto Accelerator 4000 boards using the same keystore, you need to backup this new master key and restore it to the other boards.

Rekeying the remote access key logs the security officer out, forcing a new connection that uses the new remote access key.

You may specify one of three key types when issuing the `rekey` command:

TABLE 4-6 Key Types

Key Type	Action
`master`	Rekey the master key.
`remote`	Rekey the remote access key. Logs the security officer out.
`all`	Rekeys both master and remote access keys.

The following is an example of entering a key type of `all` with the `rekey` command:

```
vcaadm{vcaN@hostname, sec-officer}> rekey

Key type (master/remote/all): all
WARNING: Rekeying the master key will render all old board backups
         useless with the new keystore file. If other boards use this
         keystore, they will need to have this new key backed up and
         restored to those boards. Rekeying the remote access key will
         terminate this session and force you to log in again.

Rekey board? (Y/Yes/N/No) [No]: y
Rekey of master key successful.
Rekey of remote access key successful.  Logging out.
```

Performing a Software Zeroize on the Board

There are two methods of clearing a board of all its key material. The first method is with a hardware jumper (shunt); this form of zeroizing returns the board to its original factory state (Failsafe mode). (See "Zeroizing the Sun Crypto Accelerator 4000 Hardware to the Factory State" on page 163.) The second method is to use the `zeroize` command.

Note – The `zeroize` command removes the key material, and leaves any updated firmware intact. This command also logs the security officer out upon successful completion.

To perform a software zeroize on a board with the `zeroize` command, enter the command and confirm it:

```
vcaadm{vcaN@hostname, sec-officer}> zeroize
WARNING: Issuing this command will zeroize all keys
         on the board.  Once zeroized, these keys
         cannot be recovered unless you have
         previously backed up your master key.

Proceed with zeroize? (Y/Yes/N/No) [No]: y
All keys zeroized successfully.
```

Using the `vcaadm diagnostics` Command

Diagnostics can be performed from the `vcaadm` utility and from the SunVTS software. The `diagnostics` command in `vcaadm` covers three major categories in the Sun Crypto Accelerator 4000 hardware: general hardware, cryptographic subsystem, and network subsystem. Tests for general hardware cover DRAM, flash memory, the PCI bus, the DMA controller, and other hardware internals. Tests for the cryptographic subsystem cover random number generators and cryptographic accelerators. Tests on the network subsystem cover the `vca` device.

```
vcaadm{vcaN@hostname, sec-officer}> diagnostics
Performing diagnostic tests...Done.
Diagnostic Results
-----------------------------------------------------
General Hardware:               PASS
Cryptographic Subsystem:        PASS
Network Subsystem:              PASS
-----------------------------------------------------
```

Managing the `vcad` Service

The service management facility (SMF) allows you to start and stop the `vcad` service. On boot, the service will normally be in a running (online) state. The state of the `vcad` service can be checked using the `svcs` command:

```
# svcs device/vcad
```

See the svcs(1) command for more display options.

You can use the svcadm command to start, stop, and restart the vcad service:

```
svcadm enable device/vcad
svcadm disable device/vcad
svcadm restart device/vcad
```

The vcad command configures and starts the vcad daemon, which provides cryptographic keystore services for vcaadm(1M) and other cryptographic applications. The vcad daemon also handles reading and writing of keystore data for the driver and hardware.

To access the vcad command easily, place the Sun Crypto Accelerator 4000 tools directory in your search path, for example:

```
$ PATH=$PATH:/opt/SUNWconn/cryptov2/sbin/
$ export PATH
```

The command-line syntax for the vcad command is:

`/opt/SUNWconn/cryptov2/sbin/vcad [-FV] [-f` *config-file*`]`

Note – Whenever possible, use the Service Management Facility to start and stop vcad rather than the command-line.

TABLE 4-7 describes the supported options for the vcad command.

TABLE 4-7 vcad Command Options

Option	Description
`-f` *config-file*	Specifies the location of the configuration file. The default location for this configuration file is `/etc/opt/SUNWconn/vca/vcad.conf`. If this option is used and the file cannot be opened, vcad does not start.
`-F`	Performs vcad in the foreground and sends log output to stderr. This behavior overrides a *logfile* chosen with the `-L` flag.
`-V`	Displays the version information for vcad.

vcad Configuration File

The vcad daemon obtains operating parameters from a configuration file. By default the daemon looks for this configuration file in /etc/opt/SUNWconn/vca/vcad.conf, although you may specify other files with the -f flag of the vcad command when invoking the vcad daemon. If the -f flag is not used and the default configuration file cannot be found or read, the vcad daemon attempts to start using all default values. In this case a warning message is sent to the standard error output.

The configuration file contains one directive per line. Each directive must have a value associated with it. Comments may be used and must start with the pound sign (#). Directive names are case-insensitive, but their values might be case-sensitive. See the descriptions of each directive in TABLE 4-8 for more detail.

Configuration file directives may be superseded by the use of a command-line option for the same operating parameter. For example, you can supersede the "Port" configuration file directive with the -p option. Operating parameters that are not specified with a command-line option or a configuration file directive use a built-in default value. TABLE 4-8 describes the supported command-line directives for the vcad command.

TABLE 4-8 Command-Line Directives for the vcad Command

Directives	Description
DebugLevel *level*	Enables the user to set the one of three debug levels in the configuration file. These three levels, from least verbose to most, are Notice, Info, and Debug. Notice level is the default.
HostBind *host/IP*	Tells vcad to bind and listen on the specified IPv4 or IPv6 address, or the IP address that host resolves to. Multiple HostBind directives enable vcad to listen on more than one address. If no HostBind entries are in a configuration file, the default behavior is to listen on all interfaces for connections.
KeyStoreDir *directory*	Enables the administrator to select an alternate directory for the storage of keystore files. This directory must have read and write permission for the user for which vcad runs (See the User directive). The default location for the keystore directory is /etc/opt/SUNWconn/vca/keydata.
LogFile *logfile*	Uses *logfile* as the location where all logging data is to be written. By default, logging data is written to syslog. If the -F (run in foreground) command-line flag is used, this directive is ignored and vcad logging data is sent to the standard error device.

TABLE 4-8 Command-Line Directives for the `vcad` Command *(Continued)*

Directives	Description
`MaxData` *size*	Sets the maximum allowable data to be sent in a single command to be size bytes. By default this value is 4 MB (4194304 bytes). If the data sent exceeds this value, `vcad` returns an error to the client and closes the connection.
`Port` *port*	Sets the port on which `vcad` listens. The default port vcad listens on is 6870.
`Timeout` *seconds*	Enables the administrator to set a timeout value for command data once the first byte of that data has been received. This timeout value prevents stalled reads from locking access to specific cards. This timeout does not apply to `vcad` when it is waiting for a connected client to send a new command. Firmware timeout values cover this issue. (See "Setting the Auto-Logout Time" on page 88.) The default timeout is 300 seconds (five minutes).
`User` *username*	Sets vcad to run as username. The daemon attempts to set its real user ID to the UID associated with username. The default value for this directive is the user who started the vcad process.

Using the `vcadiag` Utility

The `vcadiag` utility provides a command-line interface to the Sun Crypto Accelerator 4000 board that enables superusers to perform administrative tasks without authenticating as security officer. Command-line options determine the actions that `vcadiag` performs.

To access the `vcadiag` utility easily, place the Sun Crypto Accelerator 4000 tools directory in your search path, for example:

```
$ PATH=$PATH:/opt/SUNWconn/cryptov2/sbin/
$ export PATH
```

The `vcadiag` command-line syntax is:

- `vcadiag [-D] vca`*N*
- `vcadiag [-F] vca`*N*
- `vcadiag [-K] vca`*N*
- `vcadiag [-L] vca`*N*
- `vcadiag [-R] vca`*N*
- `vcadiag [-S] vca`*N*
- `vcadiag [-U]` *fw-file device*

- `vcadiag [-V]`
- `vcadiag [-Z] vca`*N*

Note – When using the `[-DFKLRSZ]` options, `vca`*N* is the board's device name where the *N* corresponds to the Sun Crypto Accelerator 4000 device instance number.

TABLE 4-9 describes the supported options for the `vcadiag` utility.

TABLE 4-9 `vcadiag` Options

Option	Meaning
`-D vca`*N*	Performs diagnostics on the Sun Crypto Accelerator 4000 board.
`-F vca`*N*	Displays the public key fingerprint used by the Sun Crypto Accelerator 4000 board for securing administration sessions.
`-K vca`*N*	Displays the public key and the public key fingerprint used by the Sun Crypto Accelerator 4000 board for securing administration sessions.
`-L vca`*N*	Force the driver to load if not present.
`-R vca`*N*	Resets the board.
`-S vca`*N*	Check device status for possible DR. This command only verifies whether the board is in use as a crypto service provider.
`-U` *fw-file device*	Load the firmware file *fw-file* onto device. This command works only when the board is uninitialized. To upgrade firmware on an initialized board, use the `vcaadm`(1m) command.
`-V`	Display the version information for `vcadiag`
`-Z vca`*N*	Zeroizes the board.

The following is an example of the `-D` option:

```
# vcadiag -D vca0
Running vca0 on-board diagnostics.
Diagnostics on vca0 PASSED.
```

The following is an example of the `-F` option:

```
# vcadiag -F vca0
5f26-b516-83b4-d254-a75f-c70d-0544-4de6
```

The following is an example of the `-K` option:

```
# vcadiag -K vca0
Device: vca0
Key Length: 1024 bits
Key Fingerprint: 5f26-b516-83b4-d254-a75f-c70d-0544-4de6
Modulus:
        b7215a99 8bb0dfe9 389363a0 44dac2b0 7c884161
        20ee8c8b d751437d 4e6a5cdb 76fdcb2a ad353c0b
        248edc1d 3c76591d dbca5997 f6ee8022 e8bb5a6d
        465a4f8c 601d46be 573e8681 506e5d8d f240a0db
        11d5c095 2d237061 df27b2de c353900f f531092b
        7d9a755b c5d79782 95a1180b e17303bb aca939ef
        006c73f7 74469031
Public Exponent:
        00010001
```

The following is an example of the `-R` option:

```
# vcadiag -R vca0
Resetting device vca0, this may take a minute.
Please be patient.
Device vca0 reset ok.
```

The following is an example of the `-S` option:

```
# vcadiag -S vca0
Device vca0 free.
```

The following is an example of the `-U` option:

```
# vcadiag -U fw-file vca0
Updating firmware on vca0, this may take a few minutes.
Please be patient.
Firmware update on vca0 complete.
Reset required to activate new firmware.
```

The following is an example of the `-V` option:

```
# vcadiag -V
vcadiag (Sun Crypto Accelerator 4000) 2.0
Copyright 2004 Sun Microsystems, Inc.
All rights reserved.
Use is subject to license terms.
```

The following is an example of the `-Z` option:

```
# vcadiag -Z vca0
Zeroizing device vca0, this may take a few minutes.
Please be patient.
Device vca0 zeroized.
```

Assigning Different MAC Addresses to Multiple Boards Installed in the Same Server

There are two methods to assign different MAC addresses to multiple boards in a single server. The first method is at the operating-system level, and the second is at the OpenBoot PROM level.

▼ To Assign Different MAC Addresses From a Terminal Window

1. **Enter the following command:**

```
# eeprom local-mac-address\?=true
```

Note – With the "`local-mac-address?`" parameter set to `true`, all unintegrated network interface devices use the local MAC address assigned to the product at the manufacturing facility.

---

2. **Reboot the system.**

▼ To Assign Different MAC Addresses From the OpenBoot PROM Level

1. **Enter the following command at the OpenBoot PROM `ok` prompt:**

```
ok setenv local-mac-address? true
```

Note – With the "`local-mac-address?`" parameter set to `true`, all nonintegrated network interface devices use the local MAC address assigned to the product at the manufacturing facility.

---

2. **Boot the operating system.**

CHAPTER 5

Building PKCS#11 Applications for Use With the Sun Crypto Accelerator 4000 Board

This chapter describes the board's implementation of the PKCS#11 interface and describes how to build customized PKCS#11 applications to be used with the board. This chapter includes the following sections:

- "Board Administration" on page 102
- "Slot Description" on page 102
- "PKCS#11 and FIPS Mode" on page 107
- "Developing Applications to Use PKCS#11" on page 107

The Sun Crypto Accelerator 4000 board is registered in the Solaris Cryptographic Framework as a hardware provider. Thus, the board can be administered using the system commands. Refer to *Solaris Cryptographic Services* section in the *Solaris 10 System Administration Guide: Security Services* document.

The Solaris Cryptographic Framework provides a PKCS#11 library through which the Sun Crypto Accelerator 4000 board is accessed. By default, the library is located at `/usr/lib` for 32-bit mode and `/usr/lib/sparcv9` for 64-bit mode.

```
/usr/lib/libpkcs11.so
/usr/lib/sparcv9/libpkcs11.so
```

Board Administration

PKCS#11 has a limited administrative facility with just two functions: `C_InitToken`, which initializes the token; and `C_InitPin`, which sets user PINs. The board does not use this facility, and instead uses the `vcaadm` utility. See Chapter 4.

When the board is first initialized, `vcaadm` prompts you to set up a security officer account. This security officer is not related to the PKCS#11 security officer, and cannot authenticate to a board through the PKCS#11 interface.

Also during board initialization, `vcaadm` prompts for the keystore name. The keystore name is used as the slot description and the token label for the Keystore slot. See "Keystore Slot" on page 103.

After the board is initialized, the security officer can create one or more users using the `vcaadm` utility. Users created by the security officer authenticate to a board through the PKCS#11 interface. Since PKCS#11 is designed for a single-user system, the `C_Login` entry point does not take the username as a parameter. To differentiate users, a PIN must be given as a string of the form `username:password`. For example, if the password of user `webserv` is `abc123`, the PIN used through the PKCS#11 `C_Login` entry point is `webserv:abc123`.

Slot Description

There are four kinds of slots available through the Solaris PKCS#11 library.

- Keystore slot

  Keystore slot groups together the multiple hardware providers that share a common keystore to support availability and load balancing. The Keystore slot description and the token label for the Sun Crypto Accelerator 4000 board are made up of the keystore name padded with spaces.

- Sun Metaslot

  Sun Metaslot uses all of the cryptographic engines on the system, including the Sun Crypto Accelerator 4000; thus, it provides the maximum functionality. By default, Sun Metaslot uses the Solaris Softtoken keystore; however it can be configured to use the Sun Crypto Accelerator 4000 keystore. See "Sun Metaslot" on page 103.

- Hardware slot

Hardware slot is a slot which is bound to and dedicated to a hardware device. There should be one hardware slot per Sun Crypto Accelerator 4000 board. The hardware slot's description and the token label for board are in the following format: `vca/`*instance* `Crypto Accel 2.0`. This slot is useful for diagnosis since it is directly associated with a board.

- Sun Softtoken slot

  Sun Softtoken slot is a software cryptographic provider with an on-disk keystore.

The following subsections provide details on the Keystore slot, Sun Metaslot, and Hardware slot.

Keystore Slot

The Keystore slot has the advantage of hardware redundancy and load balancing when there are more than one Sun Crypto Accelerator 4000 board on the system with the same keystore. For example, when there are two boards with the same keystore with the name of `ks`, a slot with the slot description and token label of `ks` is used as the Keystore slot.

When the Keystore slot is used, a crypto job may be sent to either board based on the board state. If one board is fully tasked, the job is sent to the other board. Also, if one board is not available due to a hardware failure, the job is sent to the other board.

With Keystore slot, both sensitive session keys and sensitive token keys are kept secure on the board. Thus, the secure key value is never revealed clear on the host memory. If the security of sensitive session keys are required, the Keystore slot is preferred over Sun Metaslot.

Sun Metaslot

The Sun Metaslot takes advantage of the Sun Crypto Accelerator 4000 board for cryptographic acceleration along with all other cryptographic providers available on the system. Sun Metaslot uses the board for the mechanisms it supports, and it uses other slots including the Solaris software implementation for the mechanisms not supported by the board. Sun Metaslot also supports failover. For more details, please refer to the Sun Metaslot documentation.

Configuring Sun Metaslot to Use the Sun Crypto Accelerator 4000 Keystore

Through Sun Metaslot, only one keystore can be accessed. By default Sun Metaslot uses the Solaris Softtoken keystore. To access the Sun Crypto Accelerator 4000 keystore through Sun Metaslot, you must use one of the following configurations.

- Configure Sun Metaslot to use the Sun Crypto Accelerator 4000 keystore system-wide using `cryptoadm`(1M).

  Enter the following command to use the Sun Crypto Accelerator 4000 keystore. For the example in this section, `ks` is the name of the Sun Crypto Accelerator 4000 keystore.

```
% cryptoadm enable metaslot token=ks
```

  This command forces a global change throughout the system, which causes all applications on the system to use the Sun Crypto Accelerator 4000 keystore by default.

- Configure Sun Metaslot to use the Sun Crypto Accelerator 4000 keystore with an environment variable.

  Sun Metaslot can be configured to use the board's keystore on a per application basis by setting an environment variable. The variable should be set to the name of the Sun Crypto Accelerator 4000 keystore.

```
% METASLOT_OBJECTSTORE_TOKEN=ks
% export METASLOT_OBJECTSTORE_TOKEN
```

  The environment variable overwrites the system-wide configuration.

Configuring Secure Failover for Sun Metaslot

Sun Metaslot supports failover by automatically migrating keys from the Sun Crypto Accelerator 4000 keystore to other slots. By doing so, the keys securely stored on the board may be revealed on the host memory. To protect the secure keys, enter the following command.

```
% cryptoadm disable metaslot auto-key-migrate
```

The auto key migration can also be disabled on a per application basis by setting the following environment variable.

```
% METASLOT_AUTO_KEY_MIGRATE=false
% export METASLOT_AUTO_KEY_MIGRATE
```

When the auto key migration is disabled, sensitive token keys are not automatically migrated to other slots. With this configuration, if an operation with a sensitive token key fails on the Sun Crypto Accelerator 4000 board, the request does not failover to other slots, and the operation fails.

When this variable is not set, the sensitive token key is migrated to other slots that support the operation, and the request is processed in a failover slot. If the job fails over to a software slot, such as Sun Softtoken, the key could be revealed on the host memory.

Note – This configuration applies to the sensitive token keys only. Other keys, such as non-sensitive keys and sensitive session keys will still be automatically migrated for failover.

To verify the current system-wide configuration, enter the following command.

```
% cryptoadm list -v metaslot
```

The following output shows that the Sun Metaslot is enabled, the automatic key migration is disabled, and the keystore slot, `ks`, is used for the persistent object store.

```
% cryptoadm list -v metaslot
System-wide Meta Slot Configuration:
------------------------------------
Status: enabled
Sensitive Token Object Automatic Migrate: disabled
Persistent object store token: ks

Detailed Meta Slot Information:
-------------------------------
actual status: enabled.
Description: Sun Metaslot

Token Present: True
Token Label: Sun Metaslot
Manufacturer ID: Sun Microsystems, Inc.
Model: 1.0
Serial Number:
Hardware Version: 0.0
Firmware Version: 0.0
UTC Time:
PIN Length: 0-253
Flags: CKF_RNG CKF_LOGIN_REQUIRED CKF_USER_PIN_INITIALIZED
CKF_TOKEN_INITIALIZED CKF_SO_PIN_LOCKED
```

Hardware Slot

The Hardware slot is dedicated to a single board; thus, it does not allow hardware redundancy or load balancing. For the typical application, either the Keystore slot or Sun Metaslot is preferred for this reason. This slot, however, is useful for diagnosis. Like the Keystore slot, both sensitive session keys and sensitive token keys are kept secure on the board.

PKCS#11 and FIPS Mode

When put in FIPS mode by the SO (using `vcaadm`), the Sun Crypto Accelerator 4000 board is compliant with Federal Information Processing Standard FIPS 140-2 level 3. Detailed information on FIPS 140-2 can be found at: `http://www.nist.gov/dmvp`

Operating the board in FIPS mode causes the following changes in the board's operation:

- Only FIPS-approved mechanisms are made available by the board, itself.
- All keys and critical security parameters cross the PCI bus in encrypted form.
- Certain additional integrity checks are done at startup and when keys and random numbers are generated.
- Random numbers are generated by a FIPS-approved algorithm that combines saved state and true random data (entropy) from a thermal-noise-based generator using hashing and arithmetic. 512 bits from the thermal-noise-based generator are used for every 160 bits of output data. (In non-FIPS mode, 512 bits from the thermal-noised-based generator are SHA-1 hashed to 160 bits.)

FIPS mode applies only to the Sun Crypto Accelerator 4000 board itself. As stated above, when the board is put in FIPS mode, only FIPS-approved mechanisms are provided by the board. Notably, MD5, and RC2 are not FIPS-approved.

However, because the FIPS regulations apply only to the hardware, software implementation of the non-FIPS-approved mechanisms will still be available through the Sun Metaslot.

Developing Applications to Use PKCS#11

The necessary header files are in `/usr/include/security`; add this directory to the include path and include `cryptoki.h`. The lower-level include files, `pkcs11.h`, `pkcs11f.h`, and `pkcs11t.h` are also available in the directory. These files are identical to those available at the PKCS#11 web site (`http://www.rsasecurity.com/rsalabs/PKCS`).

The PKCS#11 libraries are: `/usr/lib/libpkcs11.so` (32-bit mode) and `/usr/lib/sparcv9/libpkcs11.so` (64-bit mode)

The Solaris PKCS#11 library can be linked as an ordinary library, or it can be dynamically opened with `dlopen` (3DL).

When linking as an ordinary library, use the following command:

```
% cc [flags] files... -L /usr/lib -R /usr/lib -lpkcs11 [other libraries...]
```

Sun Crypto Accelerator 4000 PKCS#11 Implementation Specifics

The PKCS#11 administrative functions `C_InitToken` and `C_InitPin` are not implemented. The `C_Login` function with the `CKU_SO` (security officer) flag is rejected.

In PKCS#11, public token objects are token objects that are visible and deletable without authentication. Because the users known by the Sun Crypto Accelerator 4000 software are unrelated to Solaris users, and because the software does not ascertain user identity until `C_Login` succeeds, these objects would need to be globally visible to all users, and therefore deletable by any user. Because this behavior is not acceptable, public token objects are not allowed. Any attempt to create a public token object will fail.

The number of session objects is limited by virtual memory only. Token objects must all fit in the RAM on the board, and the driver limits the size of the keystore to 16 Mbytes. However, the fields of the `CK_TOKEN_INFO` structure (returned by the `C_GetTokenInfo` function) that indicate maximum memory sizes are all set to `CK_EFFECTIVELY_INFINITE`. The `C_GetObjectSize` function is not implemented.

The optional dual operation functions (`C_DigestEncryptUpdate`, `C_DecryptDigestUpdate`, `C_SignEncryptUpdate`, and `C_DecryptVerifyUpdate`) are not implemented, and the `CKF_DUAL_OPERATIONS_FLAG` in the flags field returned by `C_GetTokenInfo` is false.

`C_GetOperationState` and its companion function `C_SetOperationState` are not supported.

Since the Sun Crypto Accelerator 4000 board can only operate SHA-1 and MD5 in a single part and the PKCS#11 interface requires both single part and multi part for the hash operations, `CKM_SHA_1` and `CKM_MD5` are not available from the user level of the PKCS#11 application. However, those mechanisms are available for the kernel consumers, such as IPsec.

Since AES has become a standard algorithm for secure bulk encryption, the Sun Crypto Accelerator 4000 board supports the CKM_AES_CBC mechanism with sensitive keys. AES support is implemented in the firmware and provides less performance than supported by the host system. AES is only intended to be used by applications that require the added security of having keys kept in the Sun Crypto Accelerator 4000 hardware. Thus, unless you are concerned about the security of the AES keys, AES algorithm for the board should be turned off. (See Chapter 3)

The tokens provided by the Sun Crypto Accelerator 4000 system are considered unremovable. Thus the CKF_REMOVABLE_DEVICE flag returned by CK_GetSlotInfo is false. However, the board can be dynamically reconfigured when there is no PKCS#11 application that has an active session on the board.

The C_WaitForSlotEvent function is not implemented, and the Sun Crypto Accelerator 4000 system never calls the callback function passed as the Notify parameter to C_OpenSession. The software never surrenders control back to the calling application with the pApplication parameter of C_OpenSession.

The Sun Crypto Accelerator 4000 board contains a high-quality true random number generator. It does not need to be seeded, and in fact, C_SeedRandom will be rejected.

The Sun Crypto Accelerator 4000 software defines the default values for some attributes as listed in the following table. Some permission flags such as CKA_LOCAL and CKA_ALWAYS_SENSITIVE are not implemented or enforced as noted.

TABLE 5-1 PKCS#11 Attributes and Default Values

Attribute	Value
CKA_AC_ISSUER	empty string
CKA_ALWAYS_SENSITIVE	always false
CKA_APPLICATION	empty string
CKA_ATTR_TYPES	empty string
CKA_AUTH_PIN_FLAGS	false
CKA_DECRYPT	true (not enforced)
CKA_DERIVE	false (not enforced)
CKA_ENCRYPT	true (not enforced)
CKA_END_DATE	empty string
CKA_EXTRACTABLE	true
CKA_HAS_RESET	false
CKA_ID	empty string
CKA_ISSUER	empty string
CKA_LABEL	empty string

TABLE 5-1 *(Continued)*PKCS#11 Attributes and Default Values

Attribute	Value
CKA_LOCAL	always false
CKA_MODIFIABLE	true
CKA_NEVER_EXTRACTABLE	always false
CKA_OBJECT_ID	empty string
CKA_OWNER	empty string
CKA_PRIVATE	same as CKA_TOKEN
CKA_RESET_ON_INIT	false
CKA_SECONDARY_AUTH	false
CKA_SENSITIVE	opposite of CKA_EXTRACTABLE
CKA_SERIAL_NUMBER	empty string
CKA_SIGN	true (not enforced)
CKA_SIGN_RECOVER	true (not enforced)
CKA_START_DATE	empty string
CKA_SUBJECT	empty string
CKA_TOKEN	false
CKA_TRUSTED	false
CKA_UNWRAP	true (not enforced)
CKA_VERIFY	true (not enforced)
CKA_VERIFY_RECOVER	true (not enforced)
CKA_WRAP	true (not enforced)

The CKA_TOKEN attribute defaults to false. The CKA_PRIVATE attribute defaults to the same value as CKA_TOKEN. An attempt to set both CKA_TOKEN and CKA_PRIVATE to false will fail since Sun Crypto Accelerator 4000 does not support public token objects.

The CKA_EXTRACTABLE attribute defaults to true. The CKA_SENSITIVE attribute defaults to the opposite of CKA_EXTRACTABLE. An attempt to set both CKA_SENSITIVE and CKA_EXTRACTABLE to false will fail with CKR_TEMPLATE_INCONSISTENT.

Inconsistent attributes are generally not detected. For example, even if CKA_VALUE_LENGTH is specified in the template when the CKK_DES key is created with C_CreteObject, Sun Crypto Accelerator 4000 software will not return an error code. The inconsistent attribute CKA_VALUE_LENGTH is simply ignored by the software.

The error codes returned by the software are not always as specific as what might be expected. In particular, `CKR_MECHANISM_INVALID` is returned for many errors where other values might seem more appropriate. The return code `CKR_HOST_MEMORY` usually means that an internal call to the `malloc`(3c) command failed. After this error is returned, an important state has probably not been properly saved, and attempting to continue, except by calling `C_Finalize`, could be ineffective.

The mutex callback function pointers that can be passed to `C_Initialize` are ignored.

As required by the PKCS#11 standard, all token object handles become invalid when the user calls the `C_Logout` function or closes the last PKCS#11 session. The software purges the token objects from the software's cache. A subsequent successful `C_Login` function brings in all the then-current token objects. Note that this log in could be for a different user and thus bring in a different set of token objects. However, even if this login is for the same user, the token objects might not get the same handles as they had before.

CHAPTER 6

Installing and Configuring Sun ONE Server Software

This chapter describes how to configure the Sun Crypto Accelerator 4000 board for use with Sun ONE servers. This chapter includes the following sections:

- "Administering Security for Sun ONE Web Servers" on page 113
- "Before Configuring Sun ONE Web Servers" on page 116
- "Overview of Enabling Sun ONE Web Servers" on page 119
- "Installing and Configuring Sun ONE Web Server 6.1" on page 119

Note – The Sun ONE servers described in this manual were previously named iPlanet™ Servers.

Note – All Sun ONE server software is supported for use with the board. The example in this section covers configuring the Sun ONE Web Server only. Refer to the Sun ONE documentation for details on how to install and configure Sun ONE server software.

Administering Security for Sun ONE Web Servers

This section provides an overview of the security features of the Sun Crypto Accelerator 4000 board as it is administered with Sun ONE applications.

Note – To manage keystores, you must have access to the system administrator account for your system.

Concepts and Terminology

Keystores and users must be created for applications that communicate with the Sun Crypto Accelerator 4000 board through a PKCS#11 interface, such as the Sun ONE Applications.

Note – The Apache Web Server (Chapter 7) does not use the keystore or user account features described in this chapter.

Within the context of the Sun Crypto Accelerator 4000 board, users are owners of cryptographic keying material. Each key is owned by a single user. Each user may own multiple keys. A user might want to own multiple keys to support different configurations, such as a production key and a development key (to reflect the organizations the user is supporting).

Note – The term user or user account refers to Sun Crypto Accelerator 4000 users created in `vcaadm`, not traditional UNIX user accounts. There is no fixed mapping between UNIX user names and Sun Crypto Accelerator 4000 user names.

A keystore is a repository for key material. Associated with a keystore are security officers and users. Keystores provide not only storage, but a means for key objects to be owned by user accounts. This enables keys to be hidden from applications that do not authenticate as the owner. Keystores have three components:

- Key objects – Long-term keys that are stored for applications such as the Sun ONE Web Server.
- User accounts – Accounts that provide applications a means to authenticate and access specific keys
- Security officer accounts – Accounts that provide access to key management functions through vcaadm.

Note – A single Sun Crypto Accelerator 4000 board must have exactly one keystore. Multiple Sun Crypto Accelerator 4000 boards can be configured to collectively work with the same keystore to provide additional performance and fault-tolerance.

A typical installation contains a single keystore with three users. For example, such a configuration could consist of a single keystore *keystore-name* and three users within that keystore, `webserv`, `dirserv`, and `mailserv`. This would enable the three users to own and maintain access control of their server keys within that single keystore. FIGURE 6-1 illustrates an overview of a typical installation.

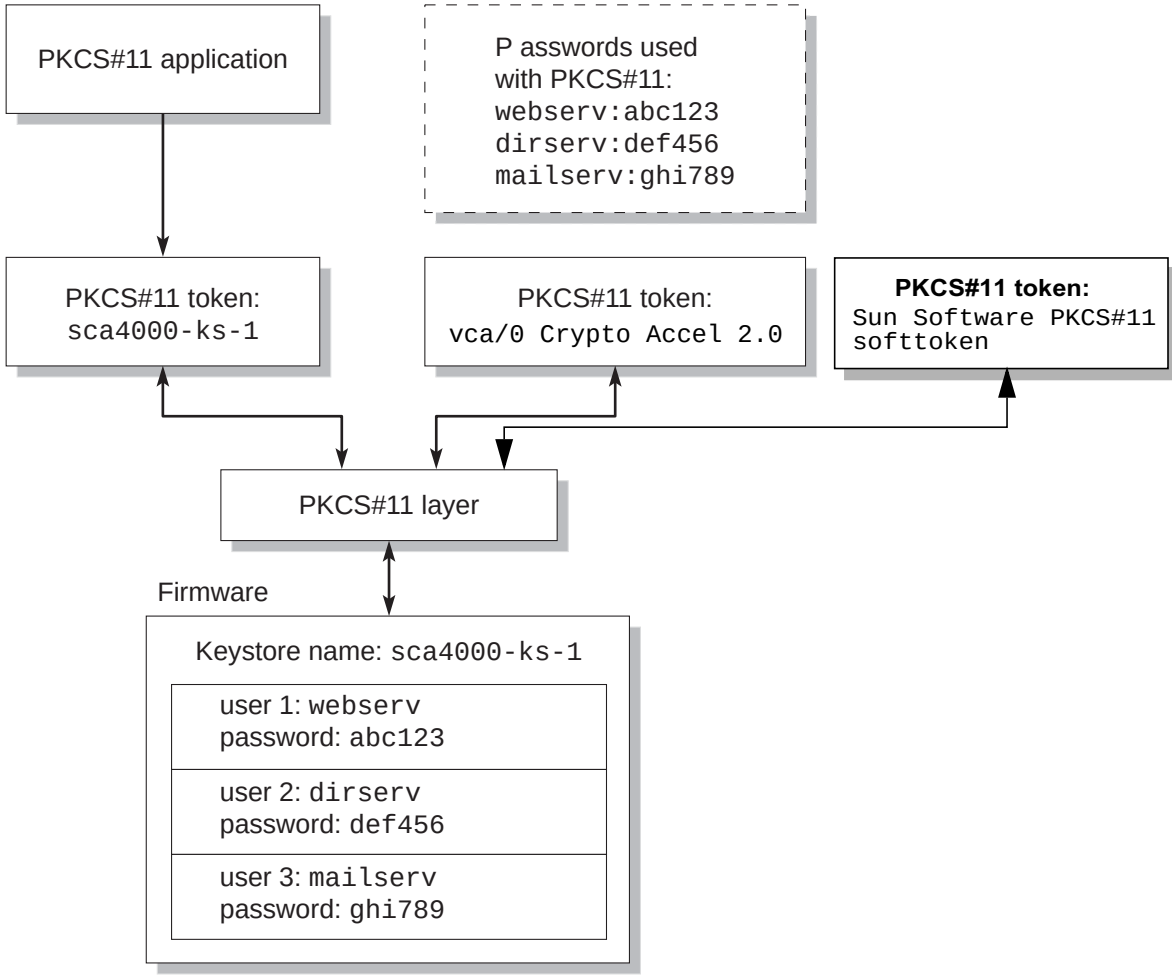

FIGURE 6-1 Keystore and Users Overview

An administrative tool, `vcaadm`, is used to manage Sun Crypto Accelerator 4000 keystores and users. See "Managing Keystores With vcaadm" on page 74.

Slots and Tokens

As discussed in Chapter 5, there are four kinds of slots presented through the Solaris Cryptographic Framework's PKCS#11 interface.

The Sun Crypto Accelerator 4000 Keystore slot can also be used for Sun ONE applications. Through a Keystore slot, asymmetric operations are the only mechanisms accelerated by the Sun Crypto Accelerator 4000 board. When there are more than two boards using the same keystore, Keystore slot provides additional performance and fault-tolerance.

Alternatively, the Sun Crypto Accelerator Hardware Slot can be used for Sun ONE applications. When the Hardware Slot is used, there is no failover support.

Example:

If there are two boards, `vca0` and `vca1`, each is assigned a keystore name (`engineering` and `finance`), five slots are presented to the Sun ONE application.

- `engineering`
- `vca/0 Crypto Accel 2.0`
- `finance`
- `vca/1 Crypto Accel 2.0`
- `Sun Software PKCS#11 softtoken`

If the server certificate resides in the `finance` keystore, the possible slots to be used for the Sun ONE application is as follows:

1. **`finance` (the Keystore slot)**
2. `vca/1` **`Crypto Accel 2.0` (the hardware slot)**

Before Configuring Sun ONE Web Servers

This section describes assigning passwords, how to populate a keystore, and how to enable the Sun ONE Web Server.

You are asked for several passwords in the course of enabling a Sun ONE Web Server, all of which are described in TABLE 6-2. These passwords are referred to throughout this chapter.

TABLE 6-1 Passwords Required for Sun ONE Web Servers

Type of Password	Description
Sun ONE Web Server Administration Server	Required to start up the Sun ONE Web Server Administration Server. This password was assigned during the Sun ONE Web Server setup.
Web Server Trust Database	Required to start the internal cryptographic module when running in secure mode. This password was assigned when creating a trust database through the Sun ONE Web Server Administration Server. This password is also required when requesting and installing certificates into the internal cryptographic module.
Security Officer	Required when performing `vcaadm` privileged operations.
username:password	Required to start the Sun Crypto Accelerator 4000 module when running in secure mode. This password is also required when requesting and installing certificates into the internal cryptographic module *keystore-name*. This password consists of the *username* and *password* of a keystore user that was created in `vcaadm`. The keystore *username* and *password* are separated by a colon (:).

Populating a Keystore

Before you can enable the board for use with a Sun ONE Web Server, you must first initialize the board and populate the board's keystore with at least one user. The keystore for the board is created during the initialization process. You can also initialize Sun Crypto Accelerator 4000 boards to use an existing keystore. See "Initializing the Board With vcaadm" on page 70.

Note – Only one keystore per Sun Crypto Accelerator 4000 board can be configured and you must configure one keystore per board. You can configure multiple Sun Crypto Accelerator 4000 boards to collectively work with the same keystore to provide additional performance and fault-tolerance.

▼ To Populate a Keystore

1. **If you have not already done so, place the Sun Crypto Accelerator 4000 tools directory in your search path, for example:**

```
$ PATH=$PATH:/opt/SUNWconn/bin
$ export PATH
```

2. **Access the `vcaadm` utility with the `vcaadm` command or enter `vcaadm -h` *hostname* to connect `vcaadm` to a board on a remote host. See "Using the vcaadm Utility" on page 61.**

```
$ vcaadm -h hostname
```

3. **Populate the board's keystore with users.**

   These user names are known only within the domain of the Sun Crypto Accelerator 4000 board and do not need to be identical to the UNIX user name that the web server process is using. Before attempting to create the user, remember that you must first log in as a `vcaadm` security officer.

4. **Create a user with the `create user` command.**

```
vcaadm{vca0@hostname, sec-officer}> create user username
Initial password:
Confirm password:
User username created successfully.
```

   The username and password created here collectively make the *username:password* (See TABLE 6-1). You must use this password when authenticating during a web server startup. This is the keystore password for a single user.

Caution – Users must remember this *username:password*. Without this password, users cannot access their keys. There is no way to retrieve a lost password.

5. **Exit `vcaadm`.**

```
vcaadm{vca0@hostname, sec-officer}> exit
```

Overview of Enabling Sun ONE Web Servers

To enable Sun ONE Web Servers you must complete the following procedures, that the rest of the chapter explains in detail.

1. **Install the Sun ONE Web Server.**
2. **Create a trust database.**
3. **Request a certificate.**
4. **Install the certificate.**
5. **Configure the Sun ONE Web Server.**

Caution – These procedures must be followed in the order given. Failure to do so could result in an incorrect configuration.

Installing and Configuring Sun ONE Web Server 6.1

This section describes how to install and configure Sun ONE Web Server 6.1 to use the board. You must perform these procedures in order. Refer to the Sun ONE Web Server documentation for more information about installing and using Sun ONE Web Servers. This section includes the following procedures:

- "To Install Sun ONE Web Server 6.1" on page 120
- "Configuring Sun ONE Web Server 6.1" on page 120
- "To Create a Trust Database" on page 121
- "To Register the Board With the Web Server" on page 122
- "To Generate a Server Certificate" on page 123
- "To Install the Server Certificate" on page 126
- "To Enable the Web Server for SSL" on page 127

▼ To Install Sun ONE Web Server 6.1

1. **Download the Sun ONE Web Server 6.1 software.**

   You can find the web server software at the following URL:
   `http://www.sun.com/`

2. **Change to the installation directory and extract the web server software.**

3. **Install the web server with the setup script from the command-line.**

   The default path name for the server is: `/opt/SUNWwbsvr/`.

   This chapter refers to the default paths. If you decide to install the software in a different location, be sure to note where you installed it.

   ```
   % ./setup
   ```

4. **Answer the prompts from the installation script.**

   Except for the following prompts, you can accept the defaults:

   a. **Agree to accept the license terms by typing** `yes`.

   b. **Enter a fully qualified domain name.**

   c. **Enter the Sun ONE Web Server 6.1 Administration Server password twice.**

   d. **Press Return when prompted.**

Configuring Sun ONE Web Server 6.1

These procedures create a trust database for the web server instance; register the board with the web server; generate and install a server certificate; and enable the web server for SSL.

The Sun ONE Web Server Administration Server must be up and running during the configuration process. This example uses the Sun Crypto Accelerator 4000 Keystore slot.

▼ To Create a Trust Database

1. **Start the Sun ONE Web Server 6.1 Administration Server.**

   To start a Sun ONE Web Server 6.1 Administration Server, use the following command (instead of running `startconsole` as `setup` requests):

```
% /opt/SUNWwbsvr/https-admserv/start
Sun ONE Web Server 6.1 B08/22/2003 12:37
info: CORE3016: daemon is running as super-user
info: CORE5076: Using [Java HotSpot(TM) Server VM, Version
1.4.1_03] from [Sun Microsystems Inc.]
info: WEB0100: Loading web module in virtual server [vs-admin] at
[/admin-app]
info: HTTP3072: [LS ls1] http://hostname.domain:8888 ready to
accept requests
startup: server started successfully
```

   The response provides the URL for connecting to your servers.

2. **Start the Administration GUI by opening up a web browser and typing:**

```
http://hostname.domain:admin-port
```

   In the authentication dialog box, enter the Sun ONE Web Server 6.1 Administration Server user name and password you selected while running `setup`.

Note – If you used the default settings during Sun ONE Web Server setup, enter `admin` for the User ID or the Sun ONE Web Server 6.1 Administration Server user name.

---

3. **Click OK.**

   The Sun ONE Web Server 6.1 Administration Server window is displayed.

4. **Create the trust database for the web server instance.**

   You might want to enable security on more than one web server instance. If so, repeat the following Step a through Step d for each web server instance.

Note – If you want to run SSL on the Sun ONE Web Server 6.1 Administration Server as well, the process of setting up a trust database is similar. Refer to the *iPlanet Web Server, Enterprise Edition Administrator's Guide* at `http://docs.sun.com` for more information.

---

a. **Click the Servers tab in the Sun ONE Web Server 6.1 Administration Server dialog box.**

b. **Select a server and click the Manage button.**

c. **Click the Security tab near the top of the page and click the Create Database link.**

d. **Enter a password (web server trust database, see** TABLE 6-1**) in the two dialog boxes and click OK.**

Choose a password of at least eight characters. This will be the password used to start the internal cryptographic modules when the Sun ONE Web Server runs in secure mode.

▼ To Register the Board With the Web Server

1. **Register the Solaris PKCS#11 library in the security module database of the Sun ONE Web Server using the `modutil` utility.**

Note – `modutil` is a utility developed by Mozilla and is available with the Sun ONE distribution. By default, the `modutil` is located at `/opt/SUNWwbsvr/bin/https/admin/bin` directory. It uses the NSS libraries located at `/opt/SUNWwbsvr/bin/https/lib`, and should be included in the environment variable, `$LD_LIBRARY_PATH`.

```
% modutil -dbdir /opt/SUNWwbsvr/alias -nocertdb -add "Solaris Cryptographic
Framework" -libfile /usr/lib/libpkcs11.so
```

2. **Certain Sun ONE applications ask for a password for every known PKCS#11 token. To limit the slots presented to those required to start the web server, disable all slots, except for one slot used by the Sun ONE application.**

```
% modutil -dbdir /opt/SUNWwbsvr/alias -nocertdb -disable "Solaris Cryptographic
Framework"
% modutil -dbdir /opt/SUNWwbsvr/alias -nocertdb -enable "Solaris Cryptographic
Framework" -slot "keystore-name"
```

▼ To Generate a Server Certificate

1. **Restart the Sun ONE Web Server 6.1 Administration Server by typing the following commands:**

```
% /opt/SUNWwbsvr/https-admserv/stop
% /opt/SUNWwbsvr/https-admserv/start
```

   The response provides the URL for connecting to your servers.

2. **Start the Administration GUI by opening up a web browser and typing:**

```
http://hostname.domain:admin-port
```

   In the authentication dialog box enter the Sun ONE Web Server 6.1 Administration Server user name and password you selected while running `setup`.

Note – If you used the default settings during Sun ONE Web Server setup, enter **`admin`** for the user ID or the Sun ONE Web Server 6.1 Administration Server user name.

---

3. **Click OK.**

   The Sun ONE Web Server 6.1 Administration Server window is displayed.

4. **To request the server certificate, select the Servers tab near the top of Sun ONE Web Server 6.1 Administration Server window. Then select a server from the drop-down menu and click the Manage button.**

   The Sun ONE Web Server 6.1 Server Manager window is displayed.

5. **Select the Security tab near the top of the Sun ONE Web Server 6.1 Server Manager window. Then click the Request a Certificate link on the left panel.**

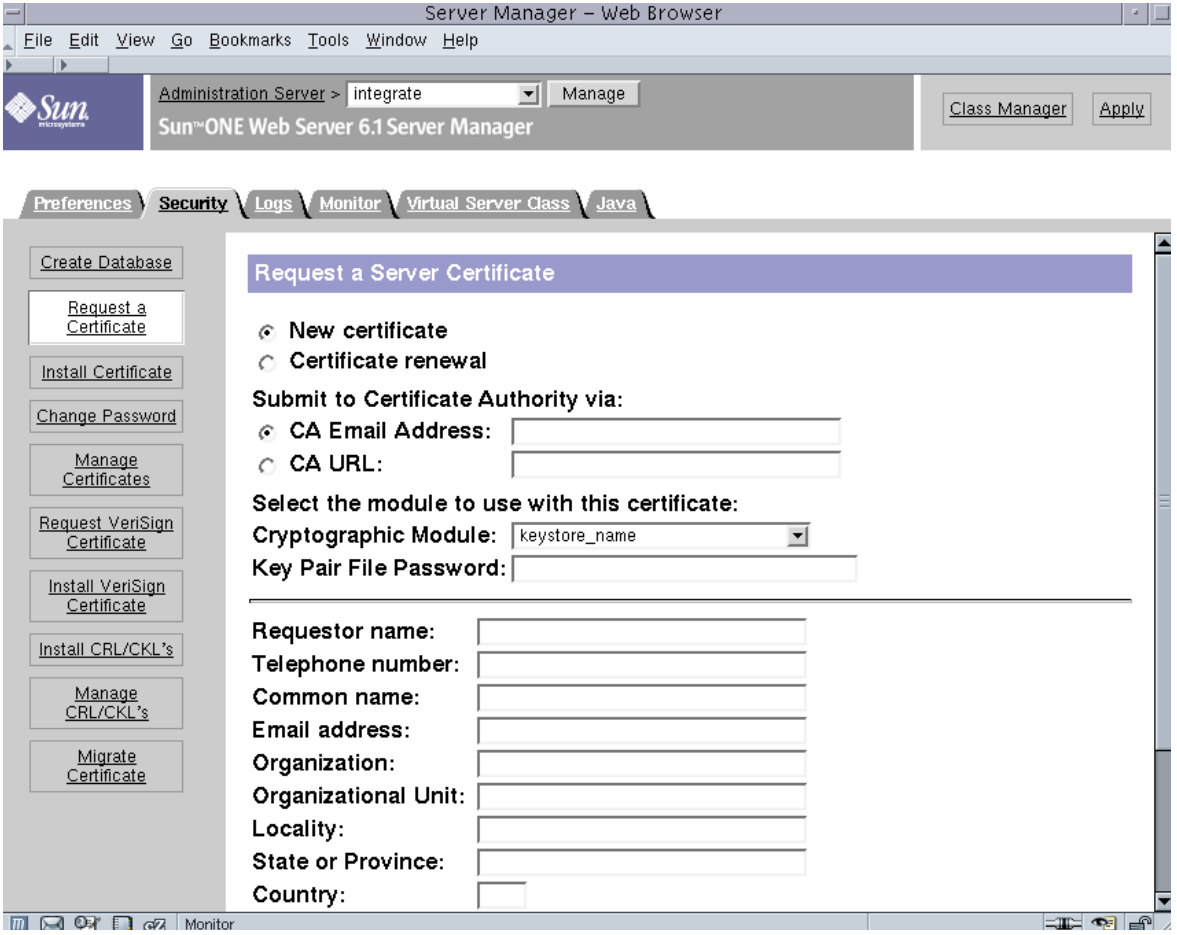

FIGURE 6-2 Sun ONE Web Server 6.1 Administration Server Request a Server Certificate Dialog Box With *keystore-name* Selected

6. **Fill out the form to generate a certificate request, using the following information:**

   a. **Select a New Certificate.**

      If you can directly post your certificate request to a web-capable certificate authority or registration authority, select the CA URL link. Otherwise, select CA Email Address and enter an email address where you would like the certificate request to be sent.

b. Select the Cryptographic Module you want to use.

Each slot has its own entry in this pull-down menu. For this example, the *keystore-name* is chosen.

c. In the Key Pair File Password dialog box, provide the password for the user that will own the key.

This password is the *username:password* (TABLE 6-1).

d. Type the appropriate information for the requestor information fields in TABLE 6-2**.**

TABLE 6-2 Requestor Information Fields

Field	Description
Requestor Name	Contact information for the requestor
Telephone Number	Contact information for the requestor
Common Name	Web site domain that is typed in a visitor's browser
Email Address	Contact information for the requestor
Organization	Company name
Organizational Unit	(Optional) Department of the company
Locality	(Optional) City, county, principality, or country
State	(Optional) Full name of the state
Country	Two-letter ISO code for the country (for example, the United States is US)

e. Click OK to submit the information.

7. Use a certificate authority to generate the certificate.

- If you choose to post your certificate request to a CA URL, the certificate request is automatically posted there.
- If you choose the CA Email Address, copy the certificate request that was emailed to you with the headers and hand it off to your certificate authority.

8. Once the certificate is generated, copy it, along with the headers, to the clipboard.

Note – The certificate is different from the certificate request and is usually presented to you in text form. Keep this data on the clipboard for Step 4 of "To Install the Server Certificate" on page 126.

▼ To Install the Server Certificate

Once your request has been approved by a certificate authority and a certificate has been issued, you must install the certificate in the Sun ONE Web Server.

1. **Click the Security tab near the top of the Sun ONE Web Server 6.1 Server Manager window.**
2. **On the left panel, click the Install Certificate link.**

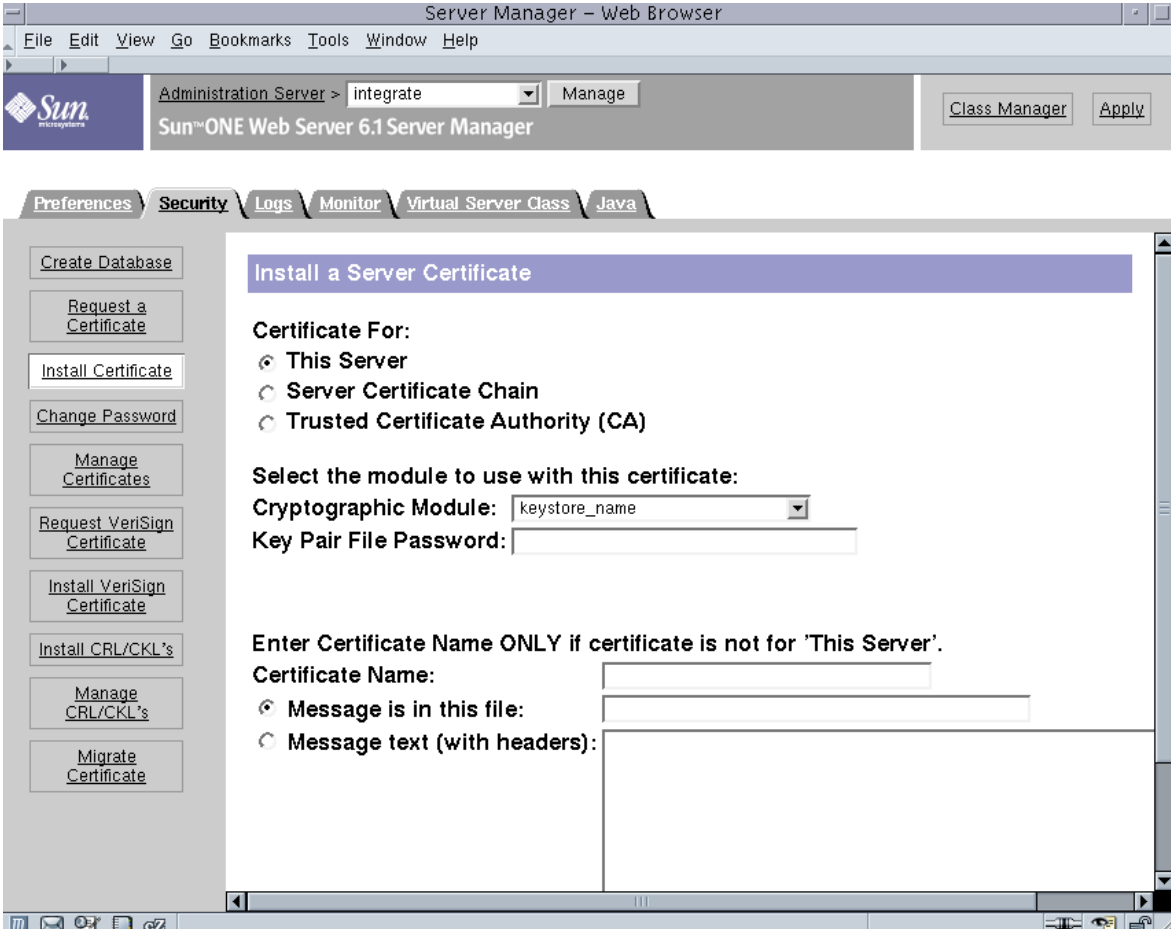

FIGURE 6-3 Sun ONE Web Server 6.1 Administration Server Install a Server Certificate Dialog Box

3. **Fill out the form to install your certificate:**

TABLE 6-3 Fields for the Certificate to Install

Fields	Description
Certificate For	This server
Cryptographic Module	Each slot has its own entry in this pull-down menu. Ensure that you select the correct slot name. For this example, use *keystore-name*.
Key Pair File Password	This password is the *username:password* (TABLE 6-1)
Certificate Name	In most cases, you can leave this blank. If you provide a name, it alters the name the web server uses to access the certificate and key when running with SSL support. The default for this field is `Server-Cert`.

4. **Paste the certificate you copied from the certificate authority (in Step 8 of the "To Generate a Server Certificate" on page 123) into the Message text box.**

   You are shown some basic information about the certificate.

5. **Click OK.**

6. **If everything looks correct, click the Add Server Certificate button.**

   On-screen messages tell you to restart the server. This is not necessary because the web server instance has been shut down the entire time.

   You are also notified that in order for the web server to use SSL, the web server must be configured to do so. Use the following procedure to configure the web server.

   Now that your web server and the Server Certificate are installed, you must enable the web server for SSL.

▼ To Enable the Web Server for SSL

1. **Select the Preferences tab near the top of the page.**

2. **Select the Edit Listen Sockets link on the left panel.**

   The main panel lists all the listen sockets set for the web server instance.

   a. **Click the link under Listen Socket ID for the listen socket you wish to configure.**

   b. **Alter the following fields:**

   - **Port**: Set to the port on which you will be running your SSL-enabled web server (usually this is port 443).
   - **Security**: Set to Enabled.

c. **Click OK to apply these changes.**

3. **Click the link under Listen Socket ID again for the listen socket you wish to configure.**

4. **Enter the *username:password* to authenticate to the keystore on the system.**

5. **If you want to change the default set of ciphers, select the cipher suites under the Ciphers heading.**

   A dialog box is displayed for changing the cipher settings. You can select either Cipher Default settings, SSL2, or SSL3/TLS. If you select the Cipher Default, you are not shown the default settings. The other two choices require you to select the algorithms you want to enable in a pop-up dialog box. Refer to your Sun ONE documentation on cipher selection.

6. **Select the certificate for the keystore followed by**: `Server-Cert` **(or the name you chose).**

   Only keys that the appropriate keystore user owns appear in the Certificate Name field. This keystore user is the user that is authenticated with the *username:password*.

7. **When you have chosen a certificate and confirmed all the security settings, click OK.**

8. **Select the Apply link in the far upper right corner to apply these changes before you start your server.**

9. **Select the Load Configuration Files link to apply the changes.**

   You are redirected to a page that allows you to start your web server instance.

   If you click the Apply Changes button when the server is off, an authentication dialog box prompts you for the *username:password*. This window is not resizable, and you might have a problem submitting the change.

   There are two workarounds for this problem:

   - Select Load Configuration Files instead.
   - Start up the web server first, and click Apply Changes.

10. **In the Sun ONE Web Server 6.1 Administration Server window, select the On/Off link on the left side of the window.**

11. **Enter the passwords for the servers and click Server On.**

    You are prompted for one or more passwords. At the Module Internal prompt, provide the password for the web server trust database.

    At the Module *keystore-name* prompt, enter the *username:password*.

    Enter the *username:password* for other keystores as prompted.

12. **Verify the new SSL-enabled web server at the following URL:**

    `https://`*hostname.domain:server-port*`/`

Note – The default *server-port* is 443.

Configuring Sun ONE Web Servers to Start Up Without User Interaction on Reboot

You can enable the Sun ONE Web Servers to perform an unattended startup at reboot with an encrypted key.

▼ To Create an Encrypted Key for Automatic Startup of Sun ONE Web Servers on Reboot

1. **Navigate to the `config` subdirectory for your Sun ONE Web Server instance—for example, `/opt/SUNWwbsvr/https-webserver-instance-name/config`.**

2. **Create a `password.conf` file with only the following lines (See** TABLE 6-1 **for password definitions):**

```
internal:trust-db-password
token-label:username:password
```

3. **Set the file ownership of the password file to the UNIX user ID that the web server runs as, and set the file permissions to be readable only by the owner of the file:**

```
# chown web-server-UNIX-user-ID password.conf
# chmod 400 password.conf
```

CHAPTER 7

Installing and Configuring Apache Web Server Software

This chapter explains how to configure and enable the Sun Crypto Accelerator 4000 board for use with Apache Web Servers. This chapter includes the following sections:

- "Creating a Private Key and Certificate" on page 131
- "Enabling Apache Web Servers" on page 133

Creating a Private Key and Certificate

The following procedure describes how to create the private key and certificate required to enable Apache Web Servers to use the Sun Crypto Accelerator 4000 board. If you already have a private key and certificate, go to "Enabling Apache Web Servers" on page 133.

▼ To Create a Private Key and Certificate

1. **Generate an RSA private key in Privacy-Enhanced Mail (PEM) format.**

```
% /usr/sfw/bin/openssl genrsa -des3 -out /etc/apache/ssl.key/server.key 1024
```

2. **Create your PEM passphrase.**

   This passphrase protects the key material. Be sure to select a strong passphrase, but one that you can remember. If you forget the passphrase, you will be unable to access your keys.

```
Enter PEM pass phrase:
Verifying password - Enter PEM pass phrase:
```

Caution – You must remember the passphrase you enter. Without the passphrase, you cannot access your keys. There is no way to retrieve a lost passphrase.

3. **Create a certificate request using the keys you just created.**

```
% /usr/sfw/bin/openssl req -new -key server.key -out certreq.csr
```

   You must first enter the passphrase to access your keys. Then provide the appropriate information for the following fields:

   - Country Name: The two-letter ISO code for the country, which is asserted on the certificate and is a required field (for example, the United States is US)
   - State or Province Name: (Optional) The full name of the state in this field (or type "." and press Return).
   - Locality: (Optional) City, county, principality, or country, which is also asserted on the certificate if provided
   - Organization Name: A value for the Organization to be asserted on the certificate
   - Organizational Unit Name: (Optional) A value for the Organizational Unit that will be asserted on the certificate
   - SSL Server Name: Web site Domain that is typed in a visitor's browser
   - Email Address: Contact information for requestor

The following is an example of how the certificate fields are entered:

```
Enter PEM pass phrase:
You are about to be asked to enter information that will be incorporated into
your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [US]:US
State or Province Name (full name) [Some-State]:.
Locality Name (eg, city) []:.
Organization Name (eg, company) []:Fictional Company, Inc.
Organizational Unit Name (eg, section) []:Online Sales Division
Common Name (eg, YOUR name) []:www.fictional-company.com
Email Address []:admin@fictional-company.com

Please enter the following 'extra' attributes to be sent with your certificate
request
A challenge password []:
An optional company name []: Fictional Comany, Inc.
```

4. **Hand off the `certreq.csr` file to your certificate authority.**
5. **Once the certificate is signed by the certificate authority, go back to the previous section to setup the Apache Web Server.**

Enabling Apache Web Servers

Apache Web Server and mod_ssl are provided with the Solaris 10 Operating System. The following instructions are for these specific releases of Apache Web Server. Refer to the Apache Web Server documentation for more information.

▼ To Enable the Apache Web Server

1. **Create an `httpd` configuration file.**

   For Solaris systems, the `httpd.conf-example` file is usually in `/etc/apache`. You can use this file as a template and copy it as follows:

   ```
   % cp /etc/apache/httpd.conf-example /etc/apache/httpd.conf
   ```

2. **Replace `ServerName` with your server name in the `http.conf` file.**

3. **If you have a private key and certificate, go to Step 4. If you do not have a private key and certificate, go to "Creating a Private Key and Certificate" on page 131.**

4. **Rename the private key as `server.key` and place it in the `/etc/apache/ssl.key` directory. Rename the private certificate as `server.crt` and place it in the `/etc/apache/ssl.crt` directory.**

5. **Start the Apache Web Server.**

   This assumes the Apache binary directory is `/usr/apache/bin`; if this is not the Apache binary directory, type in the correct directory.

   ```
   % /usr/apache/bin/apachectl startssl
   ```

6. **Enter you PEM passphrase if prompted for it.**

7. **Verify the SSL enabled web server with a browser pointing to the following URL:**

   ```
   https://ServerName:ServerPort/
   ```

   Note – The default port is 443.

8. **Verify that the Sun Crypto Accelerator 4000 Board is being used.**

   ```
   % kstat -n vca0
   ```

   Verify that the `rsaprivate` field is being incremented in the statistics.

CHAPTER 8

Diagnostics and Troubleshooting

This chapter describes diagnostic tests and troubleshooting for the Sun Crypto Accelerator 4000 software. This chapter includes the following sections:

- "Diagnostic Software" on page 135
- "Using kstat to Determine Cryptographic Activity" on page 137
- "Using the OpenBoot PROM FCode Self-Test" on page 138
- "Sun's Predictive Self-Healing" on page 141
- "Troubleshooting the Sun Crypto Accelerator 4000 Board" on page 141

Diagnostic Software

The Sun Crypto Accelerator 4000 software provides three interactive utilities for running diagnostics on the board. The first of these utilities, SunVTS, focuses on the system level network and cryptographic functionality of the Sun Crypto Accelerator 4000 subsystem (driver, firmware, and hardware). The other two utilities, `vcaadm` and `vcadiag`, perform low level diagnostics on individual hardware components of the Sun Crypto Accelerator 4000 board.

Performing SunVTS Diagnostics

SunVTS is Sun's Validation Test Suite software. The core SunVTS wrapper provides test control and a user interface to a suite of system level tests. These tests are delivered with packages `SUNWvts` and `SUNWvtsts` to make up a bundle that is contained on the Solaris 10 Software DVDs, and also available for download at `http://www.sun.com/oem/vts`.

The Sun Crypto Accelerator 4000 board can be tested by three SunVTS tests that are bundled with the core SunVTS software beginning with SunVTS 6.0 Patch Set 1 (PS1) released with Solaris 10. The first two SunVTS tests, `nettest` and `netlbtest`, operate on the Ethernet circuitry of the board. The third SunVTS test, `cryptotest`, provides diagnostics of the cryptographic circuitry of the board.

For the `nettest` and `netlbtest`, refer to the SunVTS 6.0 Test Reference Manual, User's Guide, and Quick Reference Card for instructions on how to perform and monitor these diagnostics tests. These documents are available on the Solaris 10 Documentation DVD and in the Solaris on Sun Hardware Documentation Set at `http://docs.sun.com`. For the `cryptotest`, also refer to the *SunVTS 6.0 Patch Set 1 Documentation Supplement* available at: `http://www.sun.com/products-n-solutions/hardware/docs/Software/Diagnostics/index.html`

Performing `vcaadm` Diagnostics

The `vcaadm` utility is used by a security officer to test an initialized card and is the recommended interactive diagnostic application. Both `vcaadm` and `vcadiag` invoke the same diagnostics routines on the card, but the `vcaadm` utility provides more information regarding any failures encountered. Details on how to run the `vcaadm` utility are provided in Chapter 4 of this document, and an example of how to run diagnostics using `vcaadm` is provided in "Using the vcaadm diagnostics Command" on page 92.

Performing `vcadiag` Diagnostics

The `vcadiag` interface allows the security administrator to perform diagnostics on both an initialized and uninitialized board. The `vcadiag` interface provides less information regarding diagnostic failures then the `vcaadm` interface and is primarily intended to provide a general pass/fail status to someone other than a board security officer. To run `vcadiag` diagnostics, the user invokes the `vcadiag` command with the `-D` parameter. Details on how to run the `vcadiag` utility are provided in Chapter 4, and an example of how to run diagnostics using `vcadiag` is provided in "Using the vcadiag Utility" on page 95.

Using `kstat` to Determine Cryptographic Activity

The Sun Crypto Accelerator 4000 board does not contain lights or other indicators to reflect cryptographic activity on the board. To determine whether cryptographic work requests are being performed on the board, use the `kstat`(1M) command to display the device usage. The following excerpt shows the various kstats that can be used to determine cryptographic activity.

```
# kstat vca:0
--------------
# kstat vca:0
module: vca                             instance: 0
name:   vca0                            class:    net
        3desbytes                       0
        3desjobs                        0
        aesbytes                        0
        aesjobs                         0
        dsasign                         0
        dsaverify                       0
        keygenbytes                     0
        keygenjobs                      0
        md5bytes                        0
        md5jobs                         0
        rngbytes                        280
        rngjobs                         14
        rsaprivate                      0
        rsapublic                       0
        sha1bytes                       0
        sha1jobs                        0
---------------
```

Note – In the previous example, `0` is the instance number of the `vca` device. This number should reflect the instance number of the board for which you are performing the `kstat` command.

Displaying the `kstat` information indicates whether cryptographic requests or "jobs" are being sent to the Sun Crypto Accelerator 4000 board. A change in the *jobs* values over time indicates that the board is accelerating cryptographic work requests

sent to the Sun Crypto Accelerator 4000 board. If cryptographic work requests are not being sent to the board, verify your web server configuration per the web server specific configuration.

Using the OpenBoot PROM FCode Self-Test

The following tests are available to help identify problems with the adapter if the system does not boot.

You can invoke the FCode self-test diagnostics by using the `test` or `test-all` commands from the OpenBoot PROM `ok` prompt. If you encounter an error while performing diagnostics, appropriate messages will be displayed. Refer to the *OpenBoot Command Reference Manual* for more information on the `test` and `test-all` commands.

The FCode self-test exercises most functionality subsection by subsection and ensures the following:

- Connectivity during adapter board installation
- Verification that all components required for a system boot are functional

▼ Performing the Ethernet FCode Self-Test Diagnostic

To perform the Ethernet diagnostics, you must first bring the system to a stop at the OpenBoot PROM `ok` prompt after issuing a reset. If you do not reset the system, the diagnostic tests might cause the system to hang.

For more information about the OpenBoot commands in this section, refer to the *OpenBoot Command Reference Manual*.

1. **Shut down the system.**

   Use the standard shutdown procedures described in the *Solaris Handbook for Sun Peripherals*.

2. **At the OpenBoot PROM `ok` prompt, set the `auto-boot?` configuration variable to `false`.**

```
ok setenv auto-boot? false
```

3. **Reset the system.**

```
ok reset-all
```

4. **Type `show-nets` to display the list of devices and enter a selection:**

   You see a list of devices, similar to the example below, specific to the adapter:

```
ok show-nets
a) /pci@8,600000/network@1
b) /pci@8,700000/network@5,1
q) NO SELECTION
Enter Selection, q to quit: a
/pci@8,600000/network@1 has been selected.
Type ^Y ( Control-Y ) to insert it in the command line.
e.g. ok nvalias mydev ^Y for creating devalias mydev for
/pci@8,600000/network@1
```

Note – To perform the following self-test with the `test` command, the Ethernet port must be connected to a network.

---

5. **Perform the self-test using the `test` command:**

   The following tests are performed when the `test` command is executed:

   - `vca` register test (happens only when `diag-switch?` is true)
   - Internal loopback test
   - Link up/down test

Note – The Sun Crypto Accelerator 4000 UTP adapter self-test for a 1000 Mbps connection is not supported for use with an external loopback cable because the link-clock cannot be reconciled. For this test, the local and remote ports must reconcile as clock master and clock slave. If an external loopback cable is used, both the local and remote ports are identical. So, the single port cannot be both a clock master and a clock slave, because this causes the PHY link-up to fail. For a Sun Crypto Accelerator 4000 UTP adapter self-test for a 1000 Mbps connection to work, a remote 1000BASE-T port must be connected.

Type the following:

```
ok test device-path
```

If the `test` passes, you see the following messages:

```
ok test /pci@8,600000/network@1
Testing /pci@8,600000/network@1
Register tests: passed
Internal loopback test: passed
/pci@8,600000/network@1: 100 Mbps half duplex link up
```

If the board is not connected to a network, you see the following messages:

```
ok test /pci@8,600000/network@1
Testing /pci@8,600000/network@1
Register tests: passed
Internal loopback test: passed
/pci@8,600000/network@1: link down
```

6. **After testing the adapter, type the following to return the OpenBoot PROM `ok` prompt interface to standard operating mode:**

```
ok setenv diag-switch? false
```

7. **Set the `auto-boot?` configuration parameter to `true`.**

```
ok setenv auto-boot? true
```

8. **Reset and reboot the system.**

Sun's Predictive Self-Healing

Solaris 10 introduces a new architecture for building and deploying systems and services capable of Predictive Self-Healing. The `vca` driver delivers an error telemetry for diagnosis of hardware and software problems by the Solaris Fault Manager, `fmd`(1M).

When problems are detected by the `vca` driver or Sun Crypto Accelerator 4000 firmware, error reports are sent to the fault manager daemon for diagnosis and logging. The `fmdump`(1M) utility can be used to view the list of problems diagnosed by the fault manager, along with their Universal Unique Identifiers (UUIDs) and knowledge article message identifiers. The `fmadm`(1M) utility can be used to view the resources on the system believed to be faulty. The `fmstat`(1M) utility can be used to report statistics kept by the fault manager. The fault manager is started automatically when Solaris boots, so it is not necessary to use the `fmd` command directly. Refer to the man pages for more details regarding the use of these tools.

The fault manager also sends a message to the `syslogd`(1M) service to notify an administrator that a problem has been detected. The message directs administrators to a knowledge article at `http://www.sun.com/msg/`, which explains more about the problem impact and appropriate responses. A brief description of the problem and the action required by the administrator is also provided in the message.

Troubleshooting the Sun Crypto Accelerator 4000 Board

This section describes the commands available at the OpenBoot PROM level for troubleshooting the board. Refer to the *OpenBoot Command Reference Manual* for more information on the commands described in the following subsections.

show-devs

To determine whether the Sun Crypto Accelerator 4000 device is listed in the system: from the OpenBoot PROM `ok` prompt, type `show-devs` to display the list of devices. You see lines in the list of devices, similar to the examples below, specific to the board:

```
ok show-devs
.
.
/chosen
/packages
/upa@8,480000/SUNW,ffb@0,0
/pci@8,600000/network@1
/pci@8,600000/SUNW,qlc@4
/pci@8,600000/SUNW,qlc@4/fp@0,0
.
.
```

In the preceding example, the `/pci@8,600000/network@1` entry identifies the device path to the board. There will be one such line for each board in the system.

.properties

To determine whether the Sun Crypto Accelerator 4000 device properties are listed correctly: from the `ok` prompt, type `.properties` to display the list of properties.

```
ok .properties
assigned-addresses       82000810 00000000 00102000 00000000 00002000
                          81000814 00000000 00000400 00000000 00000100
                          82000818 00000000 00200000 00000000 00200000
                          82000830 00000000 00400000 00000000 00100000
d-fru-len                00 00 00 00
d-fru-off                00 00 e8 00
d-fru-dev                eeprom
s-fru-len                00 00 08 00
s-fru-off                00 00 e0 00
s-fru-dev                eeprom
compatible               70 63 69 38 30 38 36 2c 62 35 35 35 2e 31 30 38
reg                      00000800 00000000 00000000 00000000 00000000
                          02000810 00000000 00000000 00000000 00002000
                          02000814 00000000 00000000 00000000 00000100
                          02000818 00000000 00000000 00000000 00200000
                          02000830 00000000 00000000 00000000 00100000
address-bits             00 00 00 30
max-frame-size           00 00 40 00
network-interface-type   ethernet
device-type              network
name                     network
local-mac-address        00 03 ba 0e 99 ca
version                  Sun PCI Crypto Accelerator 4000 1000Base-T
FCode 2.11.13 03/03/04
phy-type                 mif
board-model              501-6039
model                    SUNW,pci-vca
fcode-rom-offset         00000000
66mhz-capable
fast-back-to-back
devsel-speed             00000001
class-code               00100000
interrupts               00000001
max-latency              00000040
min-grant                00000040
subsystem-vendor-id      0000108e
subsystem-id             00003de8
revision-id              00000002
device-id                0000b555
vendor-id                00008086
```

watch-net

To monitor a network connection: from the `ok` prompt, type the `apply watch-net` command with the device path:

```
ok apply watch-net /pci@8,600000/network@1
/pci@8,600000/network@1: 1000 Mbps full duplex link up
Watch ethernet packets
'.' is a good packet and 'X' is a bad packet
Press any key to stop
.....X...X......X.....
```

The system monitors network traffic, displaying "." each time it receives an error-free packet and "X" each time it receives a packet with an error that can be detected by the network hardware interface.

APPENDIX **A**

Specifications

This appendix lists the specifications for the Sun Crypto Accelerator 4000 MMF and UTP adapters. It contains the following sections:

- "Sun Crypto Accelerator 4000 MMF Adapter" on page 145
- "Sun Crypto Accelerator 4000 UTP Adapter" on page 148

Sun Crypto Accelerator 4000 MMF Adapter

This section provides the specifications for the Sun Crypto Accelerator 4000 MMF adapter.

Connectors

FIGURE A-1 shows the connector for the Sun Crypto Accelerator 4000 MMF adapter.

FIGURE A-1 Sun Crypto Accelerator 4000 MMF Adapter Connector

TABLE A-1 lists the characteristics of the SC connector (850 nm).

TABLE A-1 SC Connector Link Characteristics (IEEE P802.3z)

Characteristic	62.5 Micron MMF	50 Micron MMF
Operating range	Up to 260 meters	Up to 550 meters

Physical Dimensions

TABLE A-2 Physical Dimensions

Dimension	Measurement	Metric Measurement
Length	12.283 inches	312.00 mm
Width	4.200 inches	106.68 mm

Performance Specifications

TABLE A-3 Performance Specifications

Feature	Specification
PCI clock	33/66 MHz max
PCI data burst transfer rate	Up to 64-byte bursts
PCI data/address width	32/64-bit
PCI modes	Master/slave
1 Gbps, 850 nm	1000 Mbps (full duplex)

Power Requirements

TABLE A-4 Power Requirements

Specification	Measurement
Maximum power consumption	6.25 W @ 5V 12.75 W @ 3.3V
Voltage tolerance	5V +/- 5% 3.3V +/- 5%

Interface Specifications

TABLE A-5 Interface Specifications

Feature	Specification
PCI clock	33 MHz or 66 MHz
Host interface	PCI 2.1 with support for 33 MHz or 66 MHz clock rate and 3.3V or 5V power
PCI bus width	32 bits or 64 bits

Environmental Specifications

TABLE A-6 Environmental Specifications

Condition	Operating Specification	Storage Specification
Temperature	0˚ to +55˚ C, +32˚ to +131˚ F	-40˚ to +75˚ C, -40˚ to +167˚ F
Relative humidity	5 to 85% noncondensing	0 to 95% noncondensing

Sun Crypto Accelerator 4000 UTP Adapter

This section provides the specifications for the Sun Crypto Accelerator 4000 UTP adapter.

Connectors

FIGURE A-1 shows the connector for the Sun Crypto Accelerator 4000 UTP adapter.

FIGURE A-2 Sun Crypto Accelerator 4000 UTP Adapter Connector

TABLE A-7 lists the characteristics of the Cat-5 connector used by the Sun Crypto Accelerator 4000 UTP adapter.

TABLE A-7 Cat-5 Connector Link Characteristics

Characteristic	Description
Operating range	Up to 100 meters

Physical Dimensions

TABLE A-8 Physical Dimensions

Dimension	Measurement	Metric Measurement
Length	12.283 inches	312.00 mm
Width	4.200 inches	106.68 mm

Performance Specifications

TABLE A-9 Performance Specifications

Feature	Specification
PCI clock	33/66 MHz max
PCI data burst transfer rate	Up to 64-byte bursts
PCI data/address width	32/64-bit
PCI modes	Master/slave
1 Gbps	1000 Mbps (Full Duplex)
100 Mbps	100 Mbps (Full and Half Duplex)
10 Mbps	10 Mbps (Full and Half Duplex)

Power Requirements

TABLE A-10 Power Requirements

Specification	Measurement
Maximum power consumption	6.25 W @ 5V 12.75 W @ 3.3V
Voltage tolerance	5V +/- 5% 3.3V +/- 5%

Interface Specifications

TABLE A-11 Interface Specifications

Feature	Specification
PCI clock	33 MHz or 66 MHz
Host interface	PCI 2.1 with support for 33 MHz or 66 MHz clock rate and 3.3V or 5V power
PCI bus width	32 bits or 64 bits

Environmental Specifications

TABLE A-12 Environmental Specifications

Condition	Operating Specification	Storage Specification
Temperature	0˚ to +55˚ C, +32˚ to +131˚ F	-40˚ to +75˚ C, -40˚ to +167˚ F
Relative humidity	5 to 85% noncondensing	0 to 95% noncondensing

APPENDIX **B**

Software Licenses

This appendix provides the Sun Binary Code License Agreement and third-party software notices and licenses.

Note – The third-party licenses and notices provided in this appendix are included exactly as they are provided by the owners of the software licenses and notices.

Copyright

Products bearing SPARC trademarks are based upon architecture developed by Sun Microsystems, Inc. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

This product is covered and controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

License Agreement

SUN ONE (TM) SUN CRYPTO ACCELERATOR 4000

Sun Microsystems, Inc. Binary Code License Agreement

READ THE TERMS OF THIS AGREEMENT AND ANY PROVIDED SUPPLEMENTAL LICENSE TERMS (COLLECTIVELY "AGREEMENT") CAREFULLY BEFORE OPENING THE SOFTWARE MEDIA PACKAGE. BY OPENING THE SOFTWARE MEDIA PACKAGE, YOU AGREE TO THE TERMS OF THIS AGREEMENT. IF YOU ARE ACCESSING THE SOFTWARE ELECTRONICALLY, INDICATE YOUR ACCEPTANCE OF THESE TERMS BY SELECTING THE "ACCEPT" BUTTON AT THE END OF THIS AGREEMENT. IF YOU DO NOT AGREE TO ALL THESE TERMS, PROMPTLY RETURN THE UNUSED SOFTWARE TO YOUR PLACE OF PURCHASE FOR A REFUND OR, IF THE SOFTWARE IS ACCESSED ELECTRONICALLY, SELECT THE "DECLINE" BUTTON AT THE END OF THIS AGREEMENT.

1. LICENSE TO USE. Sun grants you a non-exclusive and non-transferable license for the internal use only of the accompanying software and documentation and any error corrections provided by Sun (collectively "Software"), by the number of users and the class of computer hardware for which the corresponding fee has been paid.

2. RESTRICTIONS. Software is confidential and copyrighted. Title to Software and all associated intellectual property rights is retained by Sun and/or its licensors. Except as specifically authorized in any Supplemental License Terms, you may not make copies of Software, other than a single copy of Software for archival purposes. Unless enforcement is prohibited by applicable law, you may not modify, decompile, or reverse engineer Software. You acknowledge that Software is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility. Sun disclaims any express or implied warranty of fitness for such uses. No right, title or interest in or to any trademark, service mark, logo or trade name of Sun or its licensors is granted under this Agreement.

3. LIMITED WARRANTY. Sun warrants to you that for a period of ninety (90) days from the date of purchase, as evidenced by a copy of the receipt, the media on which Software is furnished (if any) will be free of defects in materials and workmanship under normal use. Except for the foregoing, Software is provided "AS IS". Your exclusive remedy and Sun's entire liability under this limited warranty will be at Sun's option to replace Software media or refund the fee paid for Software.

4. DISCLAIMER OF WARRANTY. UNLESS SPECIFIED IN THIS AGREEMENT, ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT THESE DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

5. LIMITATION OF LIABILITY. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR SPECIAL, INDIRECT, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF OR RELATED TO THE USE OF OR INABILITY TO USE SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. In no event will Sun's liability to you, whether in contract, tort (including negligence), or otherwise, exceed the amount paid by you for Software under this Agreement. The foregoing limitations will apply even if the above stated warranty fails of its essential purpose.

6. TERMINATION. This Agreement is effective until terminated. You may terminate this Agreement at any time by destroying all copies of Software. This Agreement will terminate immediately without notice from Sun if you fail to comply with any provision of this Agreement. Upon Termination, you must destroy all copies of Software.

7. EXPORT REGULATIONS. All Software and technical data delivered under this Agreement are subject to US export control laws and may be subject to export or import regulations in other countries. You agree to comply strictly with all such laws and regulations and acknowledge that you have the responsibility to obtain such licenses to export, re-export, or import as may be required after delivery to you.

8. U.S. GOVERNMENT RESTRICTED RIGHTS. If Software is being acquired by or on behalf of the U.S. Government or by a U.S. Government prime contractor or subcontractor (at any tier), then the Government's rights in Software and accompanying documentation will be only as set forth in this Agreement; this is in accordance with 48 CFR 227.7201 through 227.7202-4 (for Department of Defense (DOD) acquisitions) and with 48 CFR 2.101 and 12.212 (for non-DOD acquisitions).

9. GOVERNING LAW. Any action related to this Agreement will be governed by California law and controlling U.S. federal law. No choice of law rules of any jurisdiction will apply.

10. SEVERABILITY. If any provision of this Agreement is held to be unenforceable, this Agreement will remain in effect with the provision omitted, unless omission would frustrate the intent of the parties, in which case this Agreement will immediately terminate.

11. INTEGRATION. This Agreement is the entire agreement between you and Sun relating to its subject matter. It supersedes all prior or contemporaneous oral or written communications, proposals, representations and warranties and prevails over any conflicting or additional terms of any quote, order, acknowledgment, or other communication between the parties relating to its subject matter during the term of this Agreement. No modification of this Agreement will be binding, unless in writing and signed by an authorized representative of each party.

For inquiries please contact: Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, CA 95054

Sun Microsystems, Inc.
Supplemental Terms for Sun Crypto Accelerator 4000

These Supplemental Terms for the Sun Crypto Accelerator 4000 supplement the terms of the Binary Code License Agreement ("BCL"). Capitalized terms not defined herein shall have the meanings ascribed to them in the BCL. These Supplemental Terms will supersede any inconsistent or conflicting terms in the BCL. Use of the Software constitutes acceptance of the BCL as supplemented hereby.

1. Trademarks and Logos. You acknowledge and agree as between you and Sun that Sun owns the SUN, SOLARIS, JAVA, JINI, FORTE, STAROFFICE, SunVTS, AnswerBook2, Sun Enterprise, Sun Enterprise Volume Manager and iPLANET trademarks and all SUN, SOLARIS, JAVA, JINI, FORTE, STAROFFICE, SunVTS, AnswerBook2, Sun Enterprise, Sun Enterprise Volume Manager and iPLANET-related trademarks, service marks, logos and other brand designations ("Sun Marks"), and you agree to comply with the Sun Trademark and Logo Usage Requirements currently located at http://www.sun.com/policies/trademarks. Any use you make of the Sun Marks inures to Sun's benefit.

2. Source Code. Software may contain source code that is provided solely for reference purposes pursuant to the terms of this Agreement. Source code may not be redistributed unless expressly provided for in this Agreement.

3. Termination for Infringement.Â Either party may terminate this Agreement immediately should any Software become, or in either party's

```
opinion be likely to become, the subject of a claim of infringement of
any intellectual property right.
```

Third Party License Terms

OPENSSL LICENSE ISSUES

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

OpenSSL License

Copyright (c) 1998-2001 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)"

4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.

5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.

6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Original SSLeay License

Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com) All rights reserved.

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com). The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are aheared to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed.

If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)" The word 'cryptographic' can be left out if the rouines from the library being used are not cryptographic related :-).

4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution licence [including the GNU Public Licence.]

``Ian Fleming was a UNIX fan!
How do I know? Well, James Bond
had the (license to kill) number 007,
i.e. he could execute anyone.''
-- Unknown

MOD_SSL LICENSE

The mod_ssl package falls under the Open-Source Software label because it's distributed under a BSD-style license. The detailed license information follows.

Copyright (c) 1998-2000 Ralf S. Engelschall. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes software developed by Ralf S. Engelschall <rse@engelschall.com> for use in the mod_ssl project (http://www.modssl.org/)."

4. The names "mod_ssl" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact rse@engelschall.com.

5. Products derived from this software may not be called "mod_ssl" nor may "mod_ssl" appear in their names without prior written permission of Ralf S. Engelschall.

6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes software developed by Ralf S. Engelschall <rse@engelschall.com> for use in the mod_ssl project (http://www.modssl.org/)."

THIS SOFTWARE IS PROVIDED BY RALF S. ENGELSCHALL ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL RALF S. ENGELSCHALL OR HIS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

APPENDIX **C**

Manual Pages

This appendix provides descriptions of the Sun Crypto Accelerator 4000 commands and utilities provided in the board's software, and lists the online manual pages for each.

The online manual pages can be viewed with the following command:

```
man -M /opt/SUNWconn/man pagename
```

TABLE C-1 lists and describes the available online manual pages.

TABLE C-1 Sun Crypto Accelerator 4000 Online Manual Pages

man page	Description
vca(7d)	Leaf driver that provides access control to the underlying hardware cryptographic accelerator
vcad(1m)	Daemon that provides keystore services
vcaadm(1m)	Utility that manipulates the configuration, account, and keying databases associated with the board
vcadiag(1m)	Utility that allows superusers to reset boards, zeroize key material, and perform basic diagnostics

APPENDIX **D**

Zeroizing the Hardware

This appendix describes how to perform a hardware zeroize of the Sun Crypto Accelerator 4000 board, which returns the board to the factory state. When the board is returned to the factory state, it is in Failsafe mode.

Caution – You should perform a hardware zeroize only if it is absolutely necessary. If you need to remove all key material only, perform a software zeroize with the `zeroize` command in the `vcaadm` program. See "Performing a Software Zeroize on the Board" on page 91 for details on the `zeroize` command. Also refer to the online manual pages for `vcadiag`(4) for removing all key material.

Note – Performing a hardware zeroize on the board removes the Sun Crypto Accelerator 4000 firmware. You will have to reinstall the firmware which is provided with the Sun Crypto Accelerator 4000 software.

Zeroizing the Sun Crypto Accelerator 4000 Hardware to the Factory State

In some situations, it might become necessary to return a board to `failsafe` mode, and clear it of all key material and configuration information. This can only be done by using a standard SCSI hardware jumper (shunt).

Note – You can use the `zeroize` command with the `vcaadm` program to remove all key material from a Sun Crypto Accelerator 4000 board. However, the `zeroize` command leaves any updated firmware intact. See "Performing a Software Zeroize on the Board" on page 91. Also refer to the `vcadiag`(4) online manual pages.

▼ To Zeroize the Sun Crypto Accelerator 4000 Board With a Hardware Jumper

1. **Power off the system.**

Note – For some systems, you can use dynamic reconfiguration (DR) to remove and replace the board as necessary for this procedure instead of powering off the system. Refer to the documentation delivered with your system for the correct DR procedures.

Caution – The board must not receive any electrical power while adjusting the jumper.

2. **Remove the computer cover to get access to the jumper, which is located at the top middle of the board.**

3. **Place the jumper on pins 1 and 2 of the jumper block.**

   Pins 1 and 2 are the pins closest to the bracket. There are four sets of two pins. Place the jumper on the 1 and 2 pin set as shown in FIGURE D-1.

Caution – The board does not function with the jumper on pins 1 and 2.

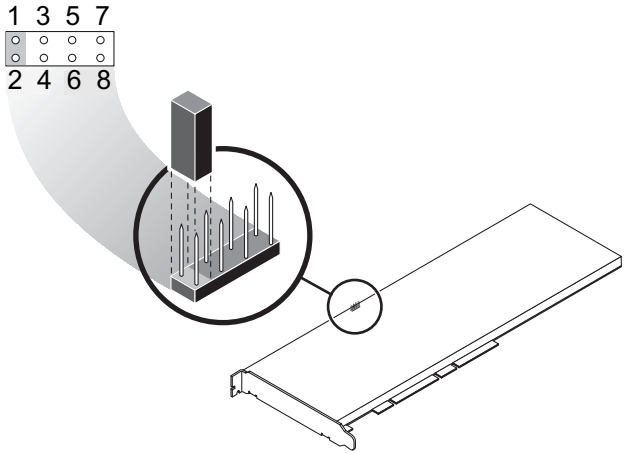

FIGURE D-1 Hardware Jumper Block Pins

4. **Power on the system.**

Caution – When you power on the system after adjusting the hardware jumper, all firmware, key material, and configuration information is deleted. This process returns the board to the factory state and places the board in Failsafe mode.

5. **Power off the system.**
6. **Remove the jumper from pins 1 and 2 of the jumper block and store the jumper in the original location.**
7. **Power on the system.**
8. **Connect to the Sun Crypto Accelerator 4000 board with `vcaadm`.**

   `vcaadm` prompts you for a path to upgrade the firmware.
9. **Type `/opt/SUNWconn/cryptov2/firmware/sca4000fw` as the path for installing the firmware.**

   The firmware is automatically installed and you are logged out of `vcaadm`.
10. **Reconnect to Sun Crypto Accelerator 4000 board with `vcaadm`.**

    `vcaadm` prompts you to either initialize the board with a new keystore, or initialize the board to use an existing keystore. See "Initializing the Board With vcaadm" on page 70.

APPENDIX **E**

Mechanisms and Restriction

This appendix lists the PKCS#11 mechanisms supported by the Sun Crypto Accelerator 4000 board.

TABLE E-1 lists the mechanisms supported by the board.

TABLE E-1 Supported PKCS#11 Mechanisms

Mechanisms	Key Range	Note
CKM_DES_CBC	8 bytes	
CKM_DES3_CBC	16 or 24 bytes	
CKM_AES_CBC	16, 24, or 32 bytes	Enable only when configured
CKM_RSA_X_509	256-2048 bits	
CKM_RSA_PKCS	256-2048 bits	
CKM_DSA	512-1024 bits	
CKM_DES_CBC_PAD	8 bytes	Wrap/Unwrap Only
CKM_DES3_CBC_PAD	16 or 24 bytes	Wrap/Unwrap Only
CKM_RC2_CBC_PAD	1-128 bytes	Wrap/Unwrap Only
CKM_AES_CBC_PAD	16, 24, or 32 bytes	Wrap/Unwrap Only Enable only when configured
CKM_DES_KEY_GEN	8 bytes	
CKM_DES2_KEY_GEN	16 bytes	
CKM_DES3_KEY_GEN	24 bytes	
CKM_AES_KEY_GEN	16, 24, or 32 bytes	Enable only when configured
CKM_RSA_PKCS_KEY_PAIR_GEN	256-2048 bits	
CKM_DSA_KEY_PAIR_GEN	512-1024 bits	
CKM_SHA_1	N/A	Available for kernel applications only
CKM_MD5	N/A	Available for kernel applications only

Index

Symbols

$HOME/.vcaadm/trustdb, 65
.properties command, 143
/etc/driver_aliases file, 43
/etc/hostname.vcaN file, 57
/etc/hosts file, 58
/etc/opt/SUNWconn/vca/keydata, 20
/etc/path_to_inst file, 43
/opt/SUNWconn/cryptov2/firmware/sca4000fw, 165
/opt/SUNWconn/cryptov2/lib, 20
/opt/SUNWconn/cryptov2/sbin, 20

Numerics

16-bit loadable counter increments, 47
8-bit vectors, 36

A

administrative commands, 20
adv-asmpause-cap, 33
adv-asmpause-cap parameter, 33
adv-autoneg-cap, 30
adv-autoneg-cap parameter, 30
advertised link parameters, 31
algorithms, 6
alias read, 35
Apache Web Servers
  creating a certificate, 131
  enabling, 133
assigning an IP address, 57
auto-boot? configuration variable, 138, 140
autonegotiation, 29, 33
  disabling, 42
  pause capability, 33
  setting, 29, 42
  transmit and receive, 33

B

blanking register for alias read, 35
blanking values, 30, 35

C

commands
  .properties, 143
  driver.conf, 43
  ifconfig, 57
  kstat, 45, 54
  modinfo, 26
  pkgadd, 25
  prtconf, 43
  prtdiag, 26
  setenv auto-boot?, 138
  show-devs, 142
  show-nets, 139
  watch-net, 144
  zeroize, 164
configuration, network, 56
configuring device driver parameters, 29
configuring the network host files, 56
cryptographic algorithm acceleration, 4
cryptographic and Ethernet driver operating statistics, 45

cryptographic driver operating statistics, 45
cryptographic driver statistics, 46

D

deleting security officers, 80
detecting 8-bit vectors, 36
device path names, 44
diagnostic support, 4
diagnostics tests, 136
`diag-switch?` configuration variable, 139
directories and files, 20
  hierarchy of, 20
displaying board status, 89
driver parameters, 29
  configuring, 29
  forced mode, 30
  parameters and settings, 30
  values and definitions, 30
driver statistics, 46
`driver.conf` file, 43
`driver_aliases` file, 43
driver-specific parameters, 52
drop parameters, 36
dynamic reconfiguration, 10

E

early detecting 8-bit vectors, 36
early drop parameters, 36
editing the network host files, 56
enabling
  Apache Web Servers, 133
entropy, 11
  high-quality, 11
  low-quality, 11
`etc/hostname.vcaN` file, 57
`etc/hosts` file, 58
`etc/path_to_inst` file, 43
Ethernet
  driver operating statistics, 45
  driver statistics, 46
  FCode self-test diagnostic, 138
  MMF, 29
  PCI properties, 54
  properties, 50
  receive counters, 53
  transmit counters, 52
  UTP, 29
example `vca.conf` file, 45

F

factory state, 163, 167
Failsafe mode, 163, 167
FCode self-test, 138
FIFO occupancy, 36
files and directories
  installation, 18, 25
FIPS 140-2 mode, 71
firmware, 165
flow control, 33
  frames, 33
  keywords, 33
forced mode of operation, 30
forced mode parameter, 34
Frame Based Link Level Flow Control Protocol, 33

G

gap parameters, 34
Gigabit forced mode parameter, 34
Gigabit media independent interface (GMII), 50

H

hardware, 11
hardware and software requirements, 11
hardware zeroize, 163, 167
high availability, 10
high-quality entropy, 11
host files, 56
`hostname.vcaN` file, 57
`hosts` file, 58
hot-plug, 10

I

IEEE 802.3x, 33
`ifconfig` command, 57
`infinit-burst`, 31
`infinit-burst` parameter, 31
initializing the board, 21
installation
  directories and files, 20

files and directories, 18, 25
software packages, 25
interface
Gigabit media independent, 50
media independent, 50
vca interface, 57
interpacket gap parameters, 34
interrupt blanking values, 30, 35
interrupt parameters, 35
ipg0, 34
ipg0 parameter, 34
ipg1, 34
ipg1 parameter, 34
ipg2, 34
ipg2 parameter, 34

K

key objects, 74
keystore data, 20
keystores, 71, 72
managing with vcaadm, 74
kstat command, 45, 54

L

link capabilities, 32
link parameters, 31
link partner, 29, 33, 50, 54
checking, 54
settings, 54
link-master, 30
link-master parameter, 30
load balancing, 11
load sharing, 11
locking to prevent backups, 81
long-term keys, 11

M

man page descriptions, 161
media independent interface (MII), 50
MMF, 29
mode, FIPS 140-2, 71
modinfo command, 26
Multi-Admin, 81
managing with vcaadm, 82

N

name property, 29
naming requirements, 75
ndd utility, 38
network configuration, 56
network host files, 56

O

OBP commands
.properties, 143
reset-all, 139
setenv auto-boot?, 138
setenv diag-switch?, 140
show-devs, 142
show-nets, 139
test device_path, 140
watch-net, 144
OBP configuration variables
auto-boot?, 138, 140
diag-switch?, 139
OBP PROM, 138, 142
occupancy, FIFO, 36
online manual pages, 161
vca(7d), 161
vcaadm(1m), 161
vcad(1m), 161
vcadiag(1m), 161
OpenBoot PROM, 138, 142
OpenBoot PROM FCode self-test, 138
operating environment, 11
operating statistics, 45
operational mode parameters, 31, 32
opt/SUNWconn/cryptov2/firmware/sca4000fw, 165
optimize throughput, 11
optional packages, 25
descriptions, 18, 25

P

packages
optional, 25
required, 25
parameter values
how to modify and display, 39
parameters, 31
8-bit vectors, 36

adv-asmpause-cap, 33
adv-autoneg-cap, 30
driver-specific, 52
early detecting 8-bit vectors, 36
early drop, 36
flow control, 33
forced mode, 34
Gigabit forced mode parameter, 34
infinit-burst, 31
interpacket gap, 34
interrupt, 35
ipg0, 34
ipg1, 34
ipg2, 34
link, 31
link capabilities, 32
link-master, 30
operational mode, 32
pause-off-threshold, 30
PCI bus interface, 37
RX random early detecting 8-bit vectors, 36
rx-intr-pkts, 30, 35
rx-intr-time, 35
setting for all vcadevices, 45
setting with vca.conf file, 43, 45
parameters and settings, 30
password requirements, 75
passwords
vcaadm, 75
path names, 44
path_to_inst file, 43
pause capability, 33
pause-off-threshold, 30
pause-off-threshold parameter, 30
PCI adapters, 29
PCI bus interface parameters, 37
pci name property, 29
PKCS#11 interface, 78
pkgadd command, 25
platforms, 11
product features, 1
properties
Ethernet, 50
Ethernet PCI, 54
protocols and interfaces, 2
prtconf command, 43
prtdiag command, 26

Q

quitting vcaadm, 70

R

random early detecting 8-bit vectors, 36
random early drop parameters, 36
read-only link partner capabilities, 51
read-only vca device capabilities, 50
read-write flow control, 33
receive counters, 53
receive interrupt blanking values, 30, 35
receive MAC counters, 47
receive random early detecting 8-bit vectors, 36
register for alias read, 35
request coalescing, 11
required packages, 25
RX blanking register for alias read, 35
RX MAC counters, 47
RX random early detecting 8-bit vectors, 36
rx-intr-pkts, 30, 35
rx-intr-pkts parameter, 30, 35
rx-intr-time, 35
rx-intr-time parameter, 35

S

security officer accounts, 74
security officers, 76
self-test, 138
server certificate, 123
setenv auto-boot?, 138
setting vca driver parameters
using ndd, 38, 43
using vca.conf, 38, 43
show-devs command, 142
show-nets command, 139
software packages, 25
Solaris operating environments, 11
specifications, 146, 147, 148, 149, 150, 151
MMF adapter, 146, 147, 148
characteristics, 146
environmental specifications, 148
interface specifications, 148

performance specifications, 147
power requirements, 147
UTP adapter, 148, 149, 150, 151
characteristics, 149
connectors, 148
environmental specifications, 151
interface specifications, 151
performance specifications, 150
physical dimensions, 150
power requirements, 150
SSL acceleration, 6
SSL algorithms, 5
standard Ethernet frame sizes, 2
standards and protocols, 2
Sun ONE Web Servers
Sun ONE Web Server 6.0
creating a trust database, 121
generating a server certificate, 123
installing, 120
installing a server certificate, 126
support libraries, 20
supported
algorithms, 6
cryptographic algorithms, 4
hardware, 11
operating environments, 11
platforms, 11
software, 11
Solaris operating environments, 11
SSL algorithms, 6

T

transmit and receive pause capability, 33
transmit counters, 52
transmit MAC counters, 47
troubleshooting, 141
trust database
creating
Sun ONE Web Server 6.0, 121
vcaadm, 65
TX and RX MAC counters, 47
TX MAC counters, 47

U

UNIX `pci` name property, 29
URL
for Sun ONE software, 120
user accounts, 74
utilities, 20
UTP, 29

V

values and definitions, 30
`vca` driver parameters
configuring, 29
forced mode, 30
parameters and settings, 30
values and definitions, 30
`vca` interface, 57
`vca.conf` file, 43
`vca.conf` file, example, 45
`vcaadm`
backups, 80
changing passwords, 78
character requirements, 75
command-line syntax, 62
deleting users, 79
diagnostics command, 92
enabling and disabling users, 78
entering commands, 68
file mode, 64
getting help, 69
initializing the board, 70
interactive mode, 64
listing security officers, 78
listing users, 78
loading new firmware, 89
locking to prevent backups, 81
logging in and out, 64
managing boards, 88
modes of operation, 63
naming requirements, 75
options, 62
password requirements, 75
populating a keystore
with security officers, 76
with users, 76
prompt, 67
quitting, 70
rekeying a board, 90
resetting a board, 90
setting auto-logout, 88
user name requirements, 75

using, 61
utility, 61
vcadiag
command-line syntax, 95
examples, 96, 98
options, 96
using, 95
utility, 95
vectors, 36

W

watch-net command, 144

Z

zeroize command, 164
zeroizing the hardware, 163, 167