Sun™ Crypto Accelerator 1000 Board Installation and User's Guide

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054 U.S.A.
650-960-1300

Part No. 816-2450-10
February 2002, Revision A

Send comments about this document to: docfeedback@sun.com

Declaration of Conformity

EMC

Compliance Model Number: DEIMOS
Product Family Name: Sun Crypto Accelerator 1000 (X6762A)

European Union

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

EN55022:1998/CISPR22:1997		Class A
EN55024:1998	Required Limits (as applicable):	
	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 60950:1999, 3rd Edition

Supplementary Information

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

/S/

Dennis P. Symanski DATE
Manager, Compliance Engineering
Sun Microsystems, Inc.
901 San Antonio Road, MPK15-102
Palo Alto, CA 94303-4900 U.S.A.
Tel: 650-786-3255
Fax: 650-786-3723

/S/

Peter Arkless DATE
Quality Manager
Sun Microsystems Scotland, Limited
Springfield, Linlithgow
West Lothian, EH49 7LR
Scotland, United Kingdom
Tel: 0506-670000 Fax: 0506-760011

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。

Contents

1. Product Overview 1

Hardware Overview 1

Product Features 2

Dynamic Reconfiguration and High Availability Considerations 3

Load Sharing 4

Hardware and Software Requirements 4

Required Patches 5

2. Installing and Removing the Sun Crypto Accelerator 1000 Board 7

Handling the Board 7

Installing the Board 8

▼ To Install the Hardware 8

Installing the Sun Crypto Accelerator 1000 Software 9

▼ To Install the Software 9

Directories and Files 11

Removing the Software 13

▼ To Delete Realms 13

▼ To Remove the Software 14

3. Enabling the Board for iPlanet Web Servers 15

Passwords 15

Creating and Populating a Realm 16

▼ To Create and Populate a Realm 16

Overview for Enabling iPlanet Web Servers 18

4. Installing and Configuring iPlanet Web Server 4.1 19

Installing iPlanet Web Server 4.1 19

▼ To Install iPlanet Web Server 4.1 19

▼ To Create a Trust Data Base 20

▼ To Generate a Server Certificate 23

▼ To Install the Server Certificate 25

Configuring iPlanet Web Server 4.1 26

▼ To Configure the iPlanet Web Server 4.1 26

5. Installing and Configuring iPlanet Web Server 6.0 29

Installing iPlanet Web Server 6.0 29

▼ To Install iPlanet Web Server 6.0 29

▼ To Create a Trust Data Base 30

▼ To Generate a Server Certificate 33

▼ To Install the Server Certificate 35

Configuring iPlanet Web Server 6.0 36

▼ To Configure the iPlanet Web Server 6.0 36

6. Enabling Apache Web Servers 39

Enabling Apache Web Servers 39

▼ To Enable the Apache Web Server 39

Creating a Certificate 42

▼ To Create a Certificate 42

7. Diagnostics and Troubleshooting 47

SunVTS Diagnostic Software 47

▼ To Run `dcatest` 48

Test Parameter Options for `dcatest` 49

`dcatest` Command-Line Syntax 50

Troubleshooting the Sun Crypto Accelerator 1000 51

A. Administering the Sun Crypto Accelerator 1000 Board With iPlanet Web Servers 53

Concepts and Terminology 53

Realms, Users, and the iPlanet Web Server 54

Tokens and Slot Files 54

Slot Files 55

Using `secadm` 56

Modes of Operation 57

Entering Commands With `secadm` 58

Authentication using `secadm` 58

Getting Help for Commands 60

Quitting the `secadm` Program 61

Setting Up and Managing Realms 61

Creating a Realm 62

Setting the Current Working Realm 63

Listing Realms 64

Listing Realm Classes 65

Deleting a Realm 65

Setting Up and Managing User Accounts 65

Creating Users 66

Listing Users 66

Changing User Passwords 66

Enabling or Disabling Users 67

Deleting Users 68

B. Manual Pages 69

C. SSL Configuration Directives for Apache Web Servers 71

D. Building Applications for Use With Sun Crypto Accelerator 1000 Board 79

E. Sun Crypto Accelerator 1000 Board Specifications 81

Physical Dimensions 81

Interface Specifications 82

Power Requirements 82

Environmental Specifications 83

F. Third-Party Licenses 85

Tables

TABLE 1-1 Supported SSL Algorithms 3

TABLE 1-2 Hardware and Software Requirements 4

TABLE 1-3 Required Patches for Sun Crypto Accelerator 1000 Software 5

TABLE 1-4 Recommended Patches for Sun Crypto Accelerator 1000 Software 5

TABLE 2-1 Sun Crypto Accelerator 1000 Directories 11

TABLE 3-1 Passwords Required for iPlanet Web Servers 16

TABLE 7-1 Test Parameter Options for `dcatest` 49

TABLE 7-2 `dcatest` Subtests 49

TABLE 7-3 `dcatest` Command-Line Syntax 50

TABLE A-1 `secadm` Options 56

TABLE A-2 Command Matrix 59

TABLE B-1 `man` pages for Sun Crypto Accelerator 1000 69

TABLE C-1 SSL Protocols 72

TABLE C-2 Available SSL Ciphers 73

TABLE C-3 SSL Aliases 74

TABLE C-4 Special Characters to Configure Cipher Preference 75

TABLE C-5 SSL Verify Client Levels 76

TABLE C-6 SSL Log Level Values 77

TABLE C-7 Available SSL Options 78

TABLE E-1 Physical Dimensions 81

TABLE E-2 Interface Specifications 82

TABLE E-3 Power Requirements 82

TABLE E-4 Environmental Specifications 83

Preface

The *Sun Crypto Accelerator 1000 Board Installation and User's Guide* provides a description of the features of the Sun™ Crypto Accelerator 1000 board and describes how to install and use the board in your system.

This book assumes that you are a system administrator familiar with the Solaris operating environment.

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*
- AnswerBook2™ online documentation for the Solaris™ operating environment
- Other software documentation that you received with your system

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*.

Shell Prompts

Shell	Prompt
C shell	*machine_name*%
C shell superuser	*machine_name*#
Bourne shell and Korn shell	$
Bourne shell and Korn shell superuser	#

Accessing Sun Documentation Online

A broad selection of Sun system documentation is located at:

`http://www.sun.com/products-n-solutions/hardware/docs`

A complete set of Solaris documentation and many other titles are located at:

`http://docs.sun.com`

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. You can email your comments to Sun at:

`docfeedback@sun.com`

Please include the part number (816-2450-10) of your document in the subject line of your email.

CHAPTER 1

Product Overview

This chapter describes the Sun Crypto Accelerator 1000 Board.

This chapter contains the following sections.

- "Hardware Overview" on page 1
- "Hardware and Software Requirements" on page 4

Hardware Overview

The Sun Crypto Accelerator 1000 board is a short PCI board that functions as a cryptographic co-processor to accelerate public key and symmetric cryptography. This product has no external interfaces. The board communicates with the host through the internal PCI bus interface. The purpose of this board is to accelerate a variety of computationally intensive cryptographic algorithms for security protocols in eCommerce applications.

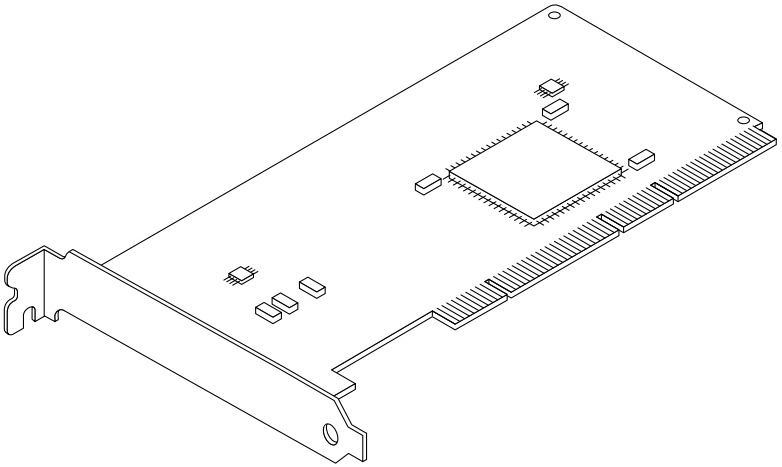


FIGURE 1-1 Sun Crypto Accelerator 1000 Board

Product Features

The Sun Crypto Accelerator 1000 is a cryptographic accelerator board to enhance the performance of SSL on Sun platforms. The Sun Crypto Accelerator 1000 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. Additionally for hardware acceleration, there is a non-trivial cost of moving data from the user application space to the hardware acceleration device, and moving the results back to the user application.

Note that a few cryptographic algorithms (for example, ARCFOUR) can be performed by highly tuned software as quickly as they can be performed in dedicated hardware. The Sun Crypto Accelerator 1000 product examines each cryptographic request and determines the best location for the acceleration (host processor or Sun Crypto Accelerator 1000), to achieve maximum throughput. Load distribution is based on cryptographic algorithm, current job loading, and data size.

TABLE 1-1 shows which accelerated algorithms may be off-loaded to hardware and which software algorithms are provided for iPlanet and Apache web servers.

TABLE 1-1 Supported SSL Algorithms

	IPlanet Web Servers		Apache Web Servers	
Algorithm	**Hardware**	**Software**	**Hardware**	**Software**
RSA	X	X	X	X
DSA	X	X	X	X
Diffie-Hellman			X	X
DES	X	X	X	X
3DES	X	X	X	X
ARCFOUR		X		X

Dynamic Reconfiguration and High Availability Considerations

The Sun Crypto Accelerator 1000 hardware and associated software provides the capability to work effectively on Sun platforms supporting Dynamic Reconfiguration (DR) and hot-plugging. In the instance where a DR or hot-plug operation takes place, the Sun Crypto Accelerator 1000 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 1000 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 1000 hardware failure, the software layer detects the failure and removes the failed card from the list of available hardware cryptographic accelerators. Sun Crypto Accelerator 1000 adjusts the scheduling algorithms to accommodate the reduction in hardware resources. Subsequent cryptographic requests will be scheduled to the remaining cards.

Additionally, the Sun Crypto Accelerator 1000 software libraries provide the capability to perform all cryptographic operations in software. This supports DR or hot-plug removal of all Sun Crypto Accelerator 1000 boards within a system domain with no adverse functional consequences. There will be a significant performance penalty incurred until the Sun Crypto Accelerator 1000 hardware is restored to the configuration.

Note that the Sun Crypto Accelerator 1000 hardware provides a source for high quality entropy for the generation of long-term keys. If all the Sun Crypto Accelerator 1000 boards within a domain or system are removed, long-term keys are generated with lower quality entropy.

Load Sharing

The Sun Crypto Accelerator 1000 software distributes 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. Requests are queued to the first 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 1-2 provides a summary of the hardware and software requirements for the Sun Crypto Accelerator 1000 board.

TABLE 1-2 Hardware and Software Requirements

Hardware and Software	Requirements
Hardware	Sun Blade™ 1000 Sun Enterprise™ 220R, 250, 420R, 450 Sun Fire™ 280R, V480, V880, 4800, 4810, 6800 Sun Netra™ T1 AC200/DC200, Netra 20, Netra t 1400/1405 Sun Ultra™ 60, 80
Operating Environment	Solaris 8 7/01 or a subsequent compatible release
PCI slots	32-bit or 64-bit 33 MHz or 66 MHz
Software	iPlanet™ Web Server 4.1 SP9, 6.0 SP1, or Apache Web Server 1.3.12 Any required patches to run the iPlanet or Apache web servers

Note – The service pack numbers (SP9 or SP1) are implied whenever iPlanet Web Server 4.1 or 6.0 is mentioned.

Required Patches

The following patches may be required to run the Sun Crypto Accelerator 1000 on your system. Solaris Updates contain patches to previous releases. Use the `showrev -p` command to determine whether the listed patches have already been installed.

If necessary, you can download the patches from the following website: `http://sunsolve.sun.com`.

Install the latest version of the patches. The dash number (-01, for example) becomes higher with each new version of the patch. If the version on the web site is higher than that shown in the following tables, it is simply a later version.

If the patch you need is not available on SunSolve[SM], contact your local sales or service representative.

The following tables list required and recommended patches to use with this product. TABLE 1-3 lists and describes required patches.

TABLE 1-3 Required Patches for Sun Crypto Accelerator 1000 Software

Patch-ID	Description
110383-01	`libnvpair`
108528-05	`KU-05` (`nvpair` support)
112438-01	`/dev/random`

Note – If you plan to use the Apache 1.3.12 web server, you must also install Patch Number 109234-02.

TABLE 1-4 lists and describes recommended patches.

TABLE 1-4 Recommended Patches for Sun Crypto Accelerator 1000 Software

Patch_ID	Description
108528-13	`KU-13` (`nvpair` security fixes)

CHAPTER 2

Installing and Removing the Sun Crypto Accelerator 1000 Board

This chapter describes how to install the Sun Crypto Accelerator 1000 hardware and software.

This chapter includes the following sections

- "Handling the Board" on page 7
- "Installing the Board" on page 8
- "Directories and Files" on page 11

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 1000 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 documentaion 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 (preferrably 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 1000 board by its edges only, take it out of the plastic bag and insert it into the PCI slot, and then secure the screw on the rear bracket.**
6. **Replace the computer cover, reconnect the power cord, and power on the system.**
7. **Verify that the board is properly installed by issuing the `show-devs` command at the `ok` prompt:**

```
ok show-devs
. . .
/pci@1f,2000/pci108e,5455@1
/pci@1f,4000/pci108e,5455@5
. . .
```

The lines `/pci@1f,2000/pci108e,5455@`*n* show that the board is installed and recognized by the system.

Installing the Sun Crypto Accelerator 1000 Software

The Sun Crypto Accelerator 1000 software is included on the *Sun Crypto Accelerator 1000* CD. You may need to download patches from the SunSolve web site. See “Required Patches” on page 5 for more information.

▼ To Install the Software

1. **Insert the** *Sun Crypto Accelerator 1000* **CD into a CD-ROM drive that is connected to your system.**
   - If your system is running Sun Enterprise Volume Manager™, it should automatically mount the CD-ROM to the `/cdrom/cdrom0` directory.
   - If your system is not running Volume Manager, mount the CD-ROM as follows:

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

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

File or Directory	Contents	
`Copyright`	U.S. copyright file	
`FR_Copyright`	French copyright file	
`Docs`	Sun Crypto Accelerator 1000 User's Guide	
`Packages`	Contains the Sun Crypto Accelerator 1000 software packages:	
	`SUNWcrypr`	Cryptography Kernel Components
	`SUNWcrypu`	Cryptographic Administration Utility and Libraries
	`SUNWcrysu`	SSL Support for Apache (optional)
	`SUNWcrypm`	Cryptographic Administration Manual Pages
	`SUNWdcar`	DCA Crypto Accelerator (Root)
	`SUNWdcamn`	DCA Crypto Accelerator Manual Page
	`SUNWdcav`	SunVTS Test of DCA Crypto Accelerator (optional)
	`SUNWcrysl`	SSL Development Tools and Libraries for Apache (optional)

Install the `SUNWcrysu` package only if you plan to use Apache as your web server.

Install the `SUNWcrysl` package only if you plan to relink to another (unsupported) version of Apache web server.

Install the `SUNWdcav` package only if the you plan to perform the SunVTS™ tests. You must have SunVTS 4.4, 4.5, or 4.6 installed to install the `SUNWdcav` package.

2. **Install the software packages by typing:**

```
# cd /cdrom/cdrom0/Packages
# pkgadd -d .
```

3. **To verify that the software is installed properly, run the `pkginfo` command.**

```
# pkginfo SUNWcrypr SUNWcrypu SUNWcrysl SUNWcrysu SUNWcrypm SUNWdcar SUNWdcamn
SUNWdcav
system  SUNWcrypr   Cryptography Kernel Components
system  SUNWcrypu   Cryptographic Administration Utility and Libraries
system  SUNWcrysl   SSL Development Tools and Libraries
system  SUNWcrysu   SSL Support for Apache
system  SUNWcrypm   Cryptographic Administration Manual Pages
system  SUNWdcar    DCA Crypto Accelerator (Root)
system  SUNWdcamn   DCA Crypto Accelerator Manual Page
system  SUNWdcav    SunVTS Test of DCA Crypto Accelerator
```

4. **(Optional)To ensure that the driver attached, you can run the `prtconf` command.**

```
# prtconf
pci108e,5455, instance #0
pci108e,5455, instance #1
```

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

```
# modinfo | grep Crypto
130 1033e946   6df0  79   1  cryptio (Cryptographic IOCTL v1.58)
131 1030240c   2d93   -   1  kcl (Cryptographic Library v1.64)
132 10313ac8   131e   -   1  kcpi (Crypto Provider Interface v1.27)
135 103178be   8684  82   1  dca (PCI Crypto Accelerator v1.156)
```

However, until you have actually used the Sun Crypto Accelerator 1000 to perform cryptographic operations, `kcl` and `cryptio` may not be loaded or appear.

Directories and Files

TABLE 2-1 shows the directories created by the default installation of the Sun Crypto Accelerator 1000 software.

TABLE 2-1 Sun Crypto Accelerator 1000 Directories

Directory	Contents
`/etc/opt/SUNWconn/crypto/realms`	Realm and user data
`/opt/SUNWconn/crypto/bin`	Application executables
`/opt/SUNWconn/crypto/lib`	Application libraries
`/opt/SUNWconn/crypto/sbin`	Statically linked executables

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

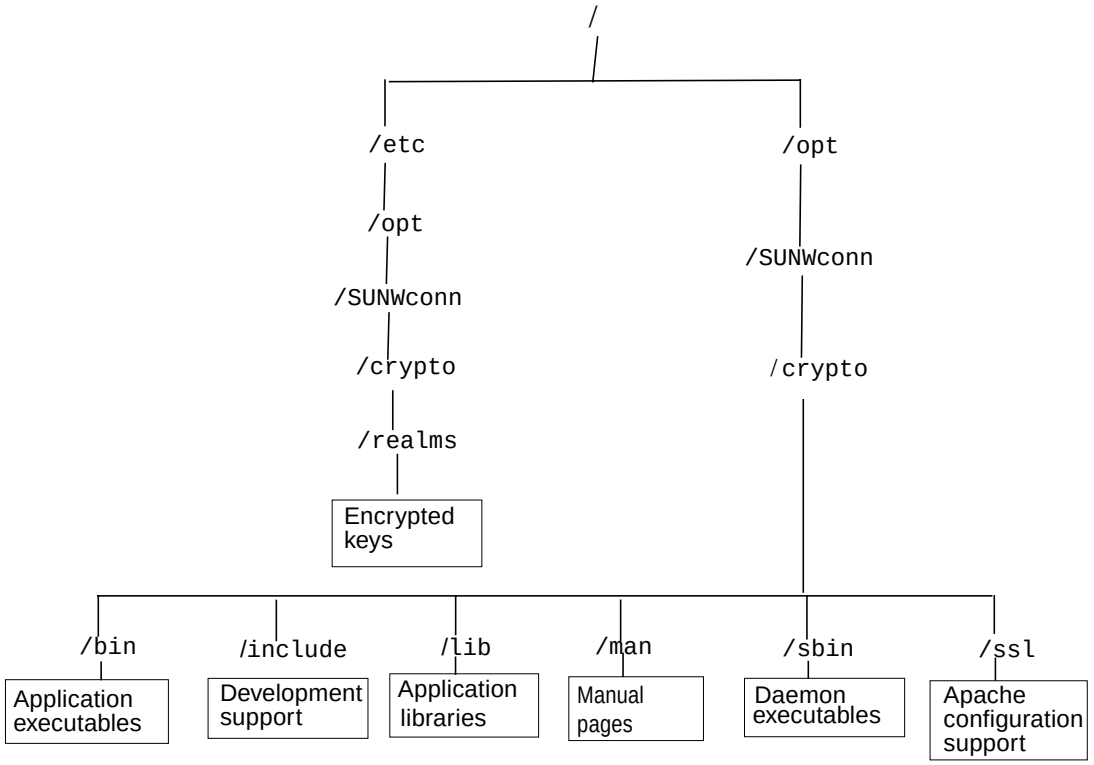


FIGURE 2-1 Sun Crypto Accelerator 1000 Directories and Files

Removing the Software

If you have created realms, you must delete the realms before removing the software. If you did not create realms, you can safely ignore the following procedure. You cannot delete a realm that is currently in use. To free references to realms, you may have to shut down the web server and/or administration server.

Caution – Before removing Sun Crypto Accelerator 1000 software you must disable any web servers you have enabled for use with the Sun Crypto Accelerator 1000 board. Failure to do so will leave those web servers nonfunctional.

▼ To Delete Realms

1. **As superuser, access the** `secadm` **utility:**

```
# /opt/SUNWconn/crypto/bin/secadm
secadm>
```

2. **Use the** `secadm` **utility to delete each realm.**

```
secadm> delete realm=realm-name
Delete realm realm-name? [Y/N]: Y
System Administrator Login Required
Login: root
Password:
Realm realm-name deleted successfully.
```

This removes all site specific realm data, including keying material.

▼ To Remove the Software

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

  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 SUNWcrysl SUNWdcav SUNWdcar SUNWcrysu SUNWcrypu SUNWcrypr
SUNWdcamn SUNWcrypm
```

Note – After installing or removing the SunVTS test for the Sun Crypto Accelerator 1000 (`SUNWdcav`), if SunVTS is already running it may be necessary to have SunVTS reprobe the system to update the available tests. See your SunVTS documentation for more information.

CHAPTER 3

Enabling the Board for iPlanet Web Servers

This chapter explains how to enable the Sun Crypto Accelerator 1000 board for use with iPlanet web servers.

This chapter includes the following sections

- "Passwords" on page 15
- "Overview for Enabling iPlanet Web Servers" on page 18
- "Creating and Populating a Realm" on page 16

Passwords

You will be asked for several passwords in the course of enabling an iPlanet web server (iWS). TABLE 3-1 provides a description of each. These passwords will be referred to throughout this chapter. If there is any confusion about which password should be used, refer to TABLE 3-1.

TABLE 3-1 Passwords Required for iPlanet Web Servers

Type of Password	Description
iWS administration server	Required to start up the iPlanet administration server. This password was assigned during iPlanet setup.
Web server trust database	Required to start the internal cryptographic modules when running in secure mode, when a certificate is requested, and when a certificate is installed. In the iPlanet web server this password is also referred to as the key pair file password and the Module internal password.
System Administrator	Required when performing `secadm` priviledged operations. This is the UNIX host password.
user@realm-name	Required to start the Sun Crypto Accelator 1000 module when running in secure mode. This password was assigned when creating a user for a realm using `secadm`.

Creating and Populating a Realm

Before you can enable the board for use with iPlanet web servers, you must first set up and populate realms. If you have not already done so, you must set up at least one realm and one user. See Appendix A for more information on realms.

▼ To Create and Populate a Realm

1. **If you have not already done so, place the Sun Crypto Accelerator 1000 tools directory in your search path, for example:**

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

2. **Access the `secadm` utility:**

```
$ secadm
```

3. **Use the** `secadm` **utility to create a new realm**:

```
secadm> create realm=realm-name
System Administrator Login Required
Login: root
Password:
Realm realm-name created successfully.
```

4. **Populate the realm with users.**

   These usernames are known only within the domain of the Sun Crypto Accelerator 1000 and do not need to be identical to the UNIX username that the web server process actually runs as. Before attempting to create the user, remember that you must first set the current working realm and log in as the system administrator.

   Before you create the users you must set the realm where the users will be created.

```
secadm> set realm=realm-name
secadm{realm-name}> su
System Administrator Login Required
Login: root
Password:
secadm{root@realm-name}#
```

   a. **If you only need one realm user, you can avoid setting up a slot file by using the user name "nobody." (See "Slot Files" on page 55 for more information.)**

```
secadm{root@realm-name}# create user=nobody
Initial password:
Confirm password:
User nobody created successfully.
```

      You must use this password when authenticating during a web server startup. This is the *user@realm-name* password.

Caution – You must remember the password you enter. Without the password, you cannot access your keys. There is no way to retrieve a lost password.

5. **Exit secadm.**

```
secadm> exit
```

Overview for Enabling iPlanet Web Servers

To enable iPlanet web servers you must complete the following procedures, which are explained in detail in the next two chapters.

1. Install the iPlanet web server
2. Create a trust database.
3. Request a certificate.
4. Install the certificate.
5. Configure the iPlanet web server.

Caution – These procedures must be followed in the order given. Failure to do so may result in an incorrect configuration.

- If you are using iPlanet Web Server 4.1, go to Chapter 4.
- If you are using iPlanet Web Server 6.0, go to Chapter 5.

CHAPTER 4

Installing and Configuring iPlanet Web Server 4.1

This chapter explains how to install and configure iPlanet Web Servers 4.1

This chapter includes the following sections

- "Installing iPlanet Web Server 4.1" on page 19
- "Configuring iPlanet Web Server 4.1" on page 26

Installing iPlanet Web Server 4.1

The following sections describe how to install iPlanet Web Server 4.1. These procedures must be done in the order given. Refer to the iPlanet web server documentation for more information about using iPlanet web servers.

▼ To Install iPlanet Web Server 4.1

1. **Install the iPlanet Web Server 4.1 software.**

   You can find the web server software at the following URL:

   `http://www.iplanet.com`

2. **Install the web server.**

   Instructions are included for one example, you may decide to configure your web server differently. The default path name for the server is:
   `/usr/netscape/server4`

   Accept the default path during the iPlanet web server installation. This book refers to these default paths. If you decide to install it in a different location, be sure to note where you installed it.

3. **Run the setup program.**

4. **Answer the prompts in the installation script.**

   Except for the following prompts you can accept the default for ease of use.

   a. **Agree to accept the license terms by typing** `yes`**.**

   b. **Enter a fully qualified** *hostname.domain***.**

   c. **Enter the iWS administration server password twice.**

   d. **Press Return when prompted.**

▼ To Create a Trust Data Base

1. **Start the administration server.**

   - To start an iPlanet Web Server 4.1, use the following command (instead of running `startconsole` as setup requests):

```
# /usr/netscape/server4/https-admserv/start
iPlanet-WebServer-Enterprise/4.1SP9 BB1-08/23/2001 05:50
startup: listening to http://hostname.domain, port 8888 as root
```

   A response message provides the URL to connect to in order to administer your servers.

2. **Start the iPlanet administration server by opening up a web browser and entering:**

```
http://hostname.domain:admin_port
```

   A pop-up window asks for user ID and password. Enter the iWS administration server username and password you selected while running setup.

Note – Enter the word `admin` for the User ID or the iWS adminstration server username if you used the default settings during iPlanet web server setup.

---

3. **Click OK.**

4. **Create the trust database for the web server instances.**

   You may want to enable security on more than one web server instance. Repeat this process for each web server instance.

Note – If you want to run SSL on the administration server as well, the process of setting up a trust database is similar. Refer to the iPlanet documentation for more information.

---

   a. **Click the Servers tab in the administration server.**

   b. **Select a server and click the Manage button.**

   c. **Click the Security tab near the top of the page and select the Create Database option.**

   d. **Enter a password (web server trust database) 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 iPlanet web server runs in secure mode.

5. **Execute the following script to enable the Sun Crypto Accelerator 1000 board:**

```
# /opt/SUNWconn/crypto/bin/sslconfig
```

   This script prompts you to choose a web server. It installs the Sun Crypto Accelerator 1000 cryptographic modules for iPlanet web server or Apache web server. The script then updates the configuration files to enable the Sun Crypto Accelerator 1000 board.

6. **Type `1` to configure your iPlanet web server to use SSL and press Enter.**

```
Sun Crypto Accelerator Installation
------------------------------------------------------
This script will install the Sun Crypto Accelerator
cryptographic modules for iPlanet Web Server
or Apache.

Please select the type of web server you wish to configure
to use the Sun Crypto Accelerator:
------------------------------------------------------
1. Configure iPlanet Web Server for SSL
2. Configure Apache for SSL
3. Work with iPlanet and Apache keys
Your selection (0 to quit): 1
```

7. **Enter the path of the web server root directory when prompted and press Enter.**

```
Please enter the full path of the web server
root directory [/usr/netscape/server4]: /usr/netscape/server4
```

8. **Type `y` and press Enter when prompted, if you want proceed.**

```
This script will update your iPlanet Web Server installation
in /usr/netscape/server4 to use the Sun Crypto Accelerator
You will need to restart your admin server after this has
completed.
Ok to proceed? y

Using database directory /usr/netscape/server4/alias...
Module "Sun Crypto Accelerator" added to database.
/usr/netscape/server4 has been configured to use
the Sun Crypto Accelerator.


<Press ENTER to continue>
```

9. **Type `0` to quit.**

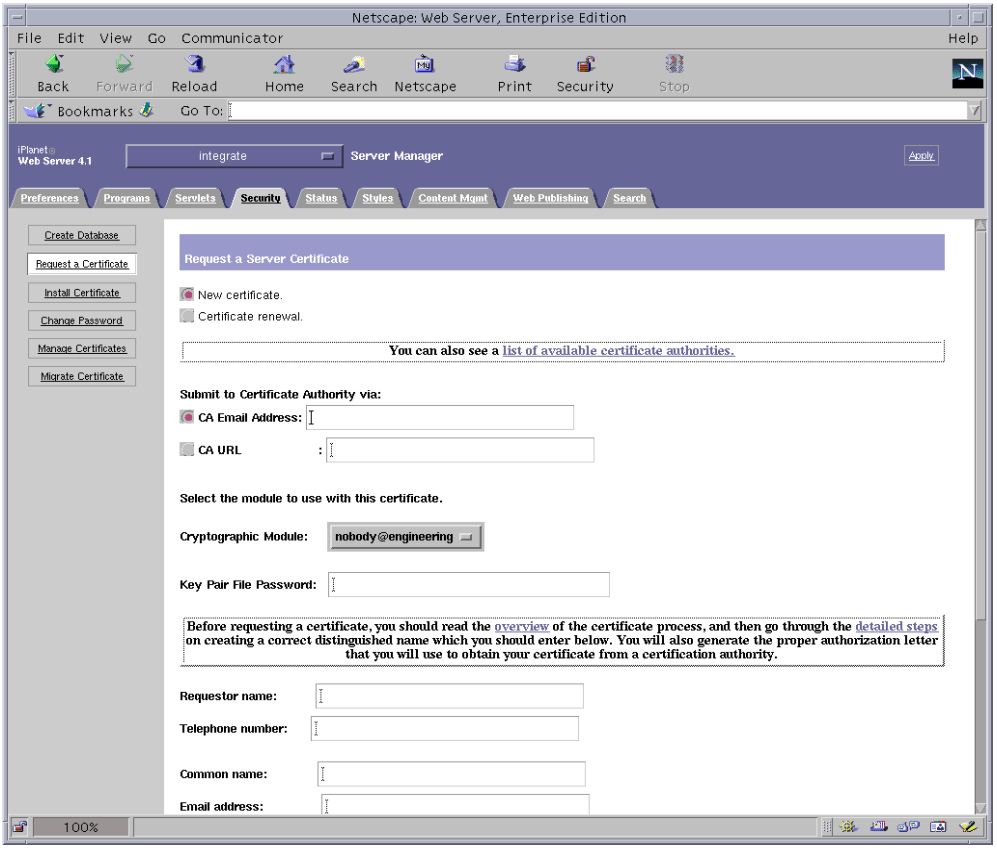
▼ To Generate a Server Certificate

1. **Restart the administration server by typing the following commands:**

```
# /usr/netscape/server4/https-admserv/stop
# /usr/netscape/server4/https-admserv/start
```

2. **To request the server certificate, click the Security tab near the top of this page.**
   The Create Trust Database window is displayed.

3. **Select the Request Certificate link on the left side of the page.**



4. **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 option. Otherwise, choose the CA Email Address and choose an email address where you would like the certificate request to be emailed to.

b. Select the Cryptographic Module you want to use.

Each realm has its own entry in this pull-down menu. Be sure that you select the correct realm. To use the Sun Crypto Accelerator 1000, you must select a module in the form of *user@realm-name*.

c. In the Key Pair File Password dialog box, provide the password for the *user@realm-name* that will own the key.

d. Provide the appropriate information for the following fields:

- Requestor Name: Contact information for the requestor
- Telephone Number: Contact information for the requestor
- Common Name: Website Domain that is typed in a visitor's browser *hostname.domain*
- Email Address: Contact information for requestor
- Organization: A value for the Organization to be asserted on the certificate
- Organizational Unit: (Optional) A value for the Organizational Unit that will be asserted on the certificate
- Locality: (Optional) City, county, principality, or country, which is also asserted on the certificate if provided
- State: (Optional) The full name of the state in this field
- Country: The two-letter ISO code for the country, which is asserted on the certificate and is a required field

e. Once you have entered all this information, click the OK button to submit it.

5. 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 the certificate request that was mailed to you with the headers and hand it off to your certificate authority.

6. Once the certificate is generated, copy it, along with the headers, to the clipboard.

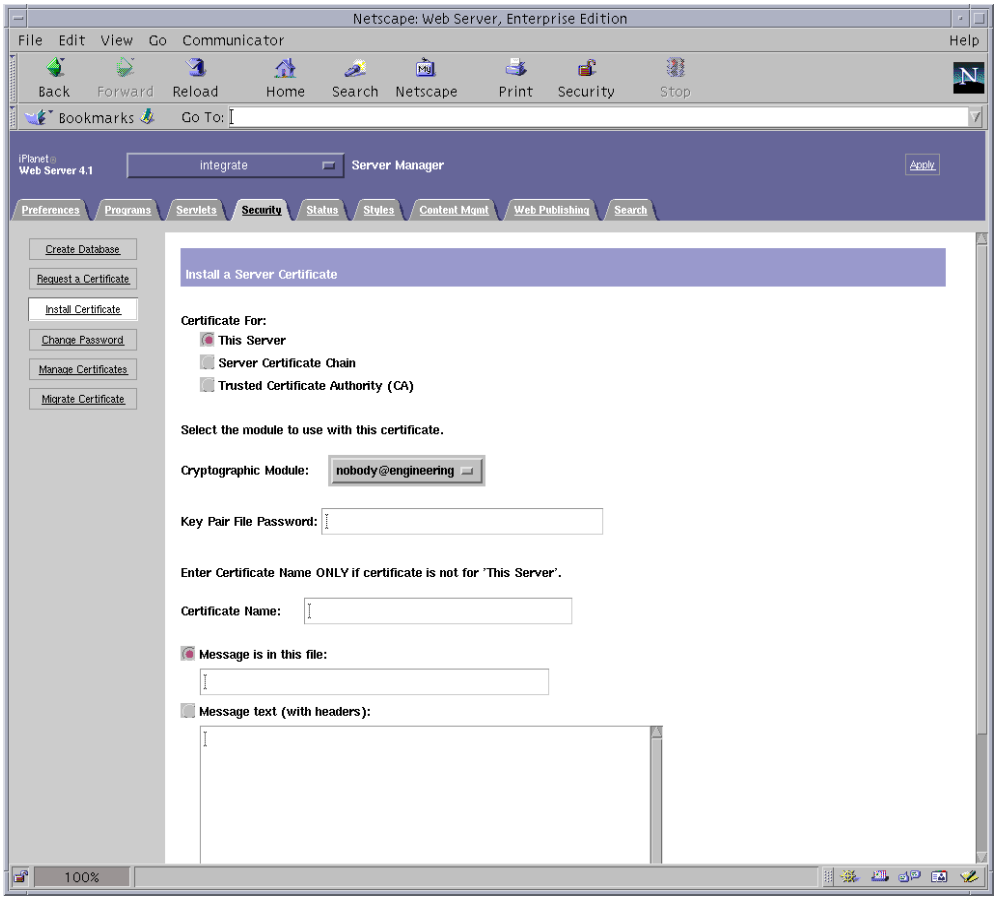
Note that the certificate is different from the certificate request and usually presented to you in text form.

▼ To Install the Server Certificate

1. **Select the Install Certificate link on the left side of the page.**

   Once your request has been approved by a certificate authority and a certificate has been issued, you must install it in the iPlanet web server.

2. **Select the Security tab and on the left frame, choose the Install Certificate option.**



3. **Fill out the form to install your certificate:**
   - Certificate For: This Server
   - Cryptographic Module: Select the appropriate *user@realm-name* name.
   - Key Pair File Password: Provide the password for the *user@realm-name* that owns the key that was generated earlier.

- Certificate Name: In most cases, you can leave this blank. If you choose to provide a name, it will alter the name the web server uses to access the certificate and key when running with SSL support.

4. **Choose Message text (with headers) and paste the certificate you copied earlier.**
5. **Click the OK button at the bottom of the page, pasting the certificate you copied from the certificate authority into the Message box.**

   You are shown some basic information about the certificate.

6. **If everything looks correct, click the Add Server Certificate button.**

   On-screen messages tell you to restart the server. This is not necessary as 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.

Configuring iPlanet Web Server 4.1

Now that your web server and the Server Certificate are installed, you must configure the web server for SSL.

▼ To Configure the iPlanet Web Server 4.1

1. **From the main administration page, choose the web server instance you want to work with and click Manage.**

   By default you should be on the Preferences tab at the top of the page. If you are not, click that tab.

2. **Click the Preferences tab near the top of the page. Select the Encryption On/Off link on the left side of the page. Set encryption to On.**

   The port field in the dialog box should update to the default SSL port number 443. Alter the port number if necessary.

3. **Click the OK button.**
4. **Apply these changes by clicking the Save button.**

   The web server is now configured to run in secure mode.

5. **Edit the** `/usr/netscape/server4/https-`*hostname*`/config/magnus.conf` **file by adding the following line:**

```
CERTDefaultNickname user@realm-name:Server-Cert
```

   Where *hostname* is the name of the web server.

   By default, the certificate you generated in Step 2 and Step 3 is named `Server-Cert`. If your certificate has a different name, substitute the name of the certificate for `Server-Cert`.

6. **Select the server you want to administer and click the Apply button in the far upper right corner of the page.**

   This action applies the changes through the administration server.

7. **Click the Load Configuration Files button to apply the changes you just made to the** `magnus.conf` **file.**

   If you click the Apply Changes button when the server is off, a pop-up window shows up for password prompt. This window is not resizable, and you might have problem submitting the change. There are two workarounds for this problem:

   - Click the Load Configuration Files instead.
   - Start up the web server first, and click on the Apply Changes button.

8. **On the web server page, select the On/Off link on the left side of the page. Enter the passwords for the servers and click the OK button.**

   You will be prompted for one or more passwords. At the Module Internal prompt, provide the password for the web server Trust Database.

   At the Module *user@realm-name* prompt, enter the password you set when you created *user* in the *realm-name* using `secadm`.

9. **Verify the new SSL-enabled web server with a browser by going to the following URL:**

   `https://`*hostname.domain*`:`*server_port*`/`

   Note that the default *server_port* is 443.

CHAPTER 5

Installing and Configuring iPlanet Web Server 6.0

This chapter explains how to enable the Sun Crypto Accelerator 1000 board for use with iPlanet 6.0 Web Servers.

This chapter includes the following sections

- "Installing iPlanet Web Server 6.0" on page 29
- "Configuring iPlanet Web Server 6.0" on page 36

Installing iPlanet Web Server 6.0

The following sections describe how to install and configure iPlanet web servers. These procedures must be done in the order given. Refer to the iPlanet web server documentation for more information about using iPlanet web servers.

▼ To Install iPlanet Web Server 6.0

1. **Install the iPlanet Web Server 6.0 software.**

   You can find the web server software at the following URL:

   `http://www.iplanet.com`

2. **Install the web server.**

   Instructions are included for one example, you may decide to configure your web server differently. The default path name for the server is: `/usr/iplanet/servers`

   Accept the default path during the iPlanet web server installation. This book refers to these default paths. If you decide to install it in a different location, be sure to note where you installed it.

3. **Run the setup program.**

4. **Answer the prompts in the installation script.**

   Except for the following prompts you can accept the default for ease of use.

   a. **Agree to accept the license terms by typing** `yes`**.**

   b. **Enter a fully qualified** *hostname.domain***.**

   c. **Enter the iWS administration server password twice.**

   d. **Press Return when prompted.**

▼ To Create a Trust Data Base

1. **Start the administration server.**

   To start an iPlanet web server, use the following command (instead of running `startconsole` as `setup` requests):

```
# /usr/iplanet/servers/https-admserv/start
iPlanet-WebServer-Enterprise/6.0SP1 B08/20/2001 00:58
warning: daemon is running as super-user
[LS ls1] http://hostname.domain/port 8888 ready to accept requests
startup: server started successfully
```

   A response message provides the URL to connect to in order to administer your servers.

2. **Start the iPlanet administration server by opening up a web browser and entering:**

```
http://hostname.domain:admin_port
```

   A pop-up window asks for user ID and password. Enter the iWS administration server username and password you selected while running setup.

Note – Enter the word `admin` for the User ID or the iWS administration server username if you used the default settings during iPlanet web server setup.

3. **Click OK.**

4. **Create the trust database for the web server instances.**

   You may want to enable security on more than one web server instance. Repeat this process for each web server instance.

Note – If you want to run SSL on the administration server as well, the process of setting up a trust database is similar. Refer to the iPlanet documentation for more information.

   a. **Click the Servers tab in the administration server.**

   b. **Select a server and click the Manage button.**

   c. **Click the Security tab near the top of the page and select the Create Database option.**

   d. **Enter a password (web server trust database) 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 iPlanet web server runs in secure mode.

5. **Execute the following script to enable the Sun Crypto Accelerator 1000 board:**

```
# /opt/SUNWconn/crypto/bin/sslconfig
```

   This script prompts you to choose a web server. It installs the Sun Crypto Accelerator 1000 cryptographic modules for iPlanet Web Server or Apache Web Server. The script then updates the configuration files to enable the Sun Crypto Accelerator 1000 board.

6. **Type `1` to configure your iPlanet web server to use SSL and press Enter.**

```
Sun Crypto Accelerator Installation
------------------------------------------------------
This script will install the Sun Crypto Accelerator
cryptographic modules for iPlanet Web Server
or Apache.

Please select the type of web server you wish to configure
to use the Sun Crypto Accelerator:
------------------------------------------------------
1. Configure iPlanet Web Server for SSL
2. Configure Apache for SSL
3. Work with iPlanet and Apache keys
Your selection (0 to quit): 1
```

7. **Enter the path of the web server root directory when prompted and press Enter.**

```
Please enter the full path of the web server
root directory [/usr/iplanet/servers]: /usr/iplanet/servers
```

8. **Type `y` and press Enter when prompted, if you want proceed.**

```
This script will update your iPlanet Web Server installation
in /usr/iplanet/servers to use the Sun Crypto Accelerator
You will need to restart your admin server after this has
completed.
Ok to proceed? y

Using database directory /usr/iplanet/servers/alias...
Module "Sun Crypto Accelerator" added to database.
/usr/iplanet/servers has been configured to use
the Sun Crypto Accelerator.


<Press ENTER to continue>
```

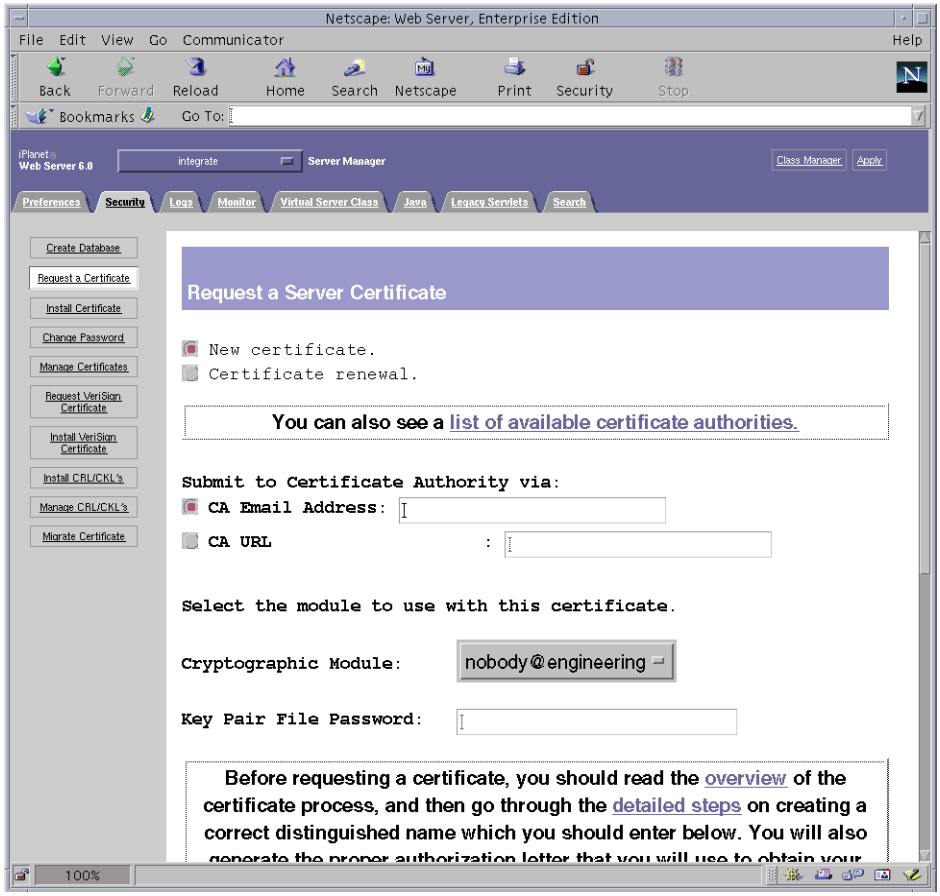
9. **Type `0` to quit.**

▼ To Generate a Server Certificate

1. **Restart the administration server by typing the following commands:**

```
# /usr/iplanet/servers/https-admserv/stop
# /usr/iplanet/servers/https-admserv/start
```

2. **To request the server certificate, click the Security tab near the top of this page.**
   The Create Trust Database window is displayed.
3. **Select the Request Certificate link on the left side of the page.**



4. **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 option. Otherwise, choose the CA Email Address and choose an email address where you would like the certificate request to be emailed to.

b. Select the Cryptographic Module you want to use.

Each realm has its own entry in this pull-down menu. Be sure that you select the correct realm. To use the Sun Crypto Accelerator 1000, you must select a module in the form of *user@realm-name*.

c. In the Key Pair File Password dialog box, provide the password for the *user@realm-name* that will own the key.

d. Provide the appropriate information for the following fields:

- Requestor Name: Contact information for the requestor
- Telephone Number: Contact information for the requestor
- Common Name: Website Domain that is typed in a visitor's browser *hostname.domain*
- Email Address: Contact information for requestor
- Organization: A value for the Organization to be asserted on the certificate
- Organizational Unit: (Optional) A value for the Organizational Unit that will be asserted on the certificate
- Locality: (Optional) City, county, principality, or country, which is also asserted on the certificate if provided
- State: (Optional) The full name of the state in this field
- Country: The two-letter ISO code for the country, which is asserted on the certificate and is a required field

e. Once you have entered all this information, click the OK button to submit it.

5. 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 mailed to you with the headers and hand it off to your certificate authority.

6. Once the certificate is generated, copy it, along with the headers, to the clipboard.

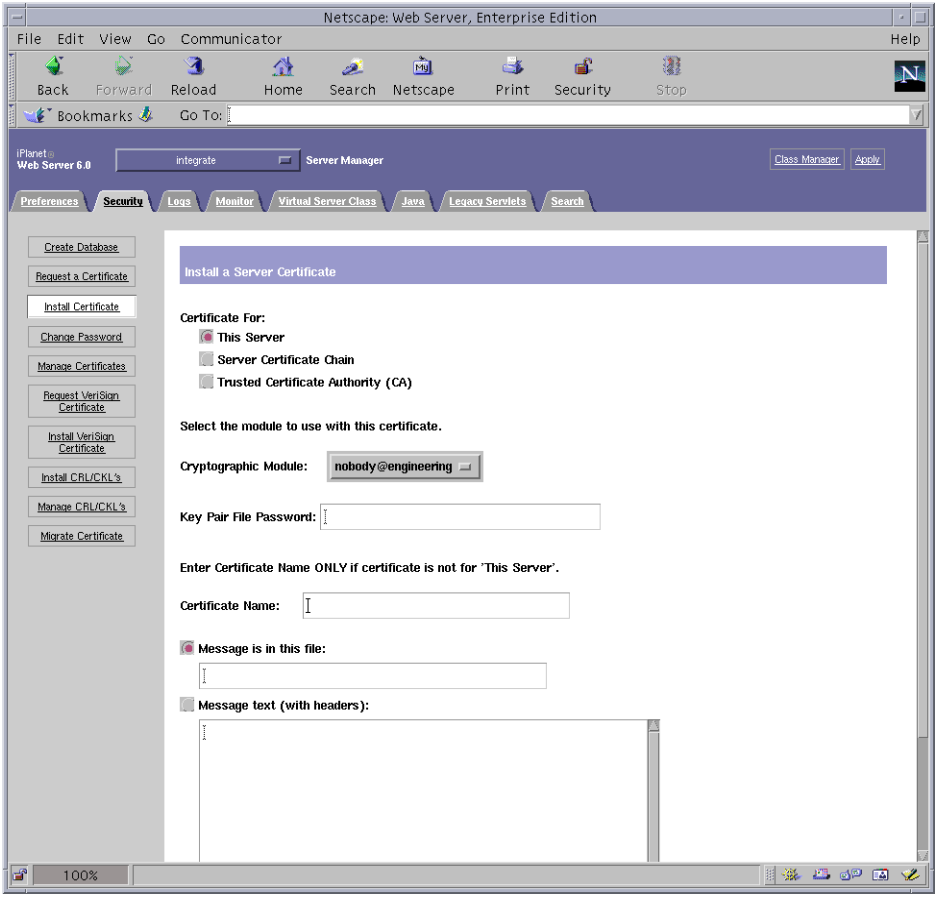
Note that the certificate is different from the certificate request and usually presented to you in text form.

▼ To Install the Server Certificate

1. **Select the Install Certificate link on the left side of the page.**

   Once your request has been approved by a certificate authority and a certificate has been issued, you must install it in the iPlanet web server.

2. **Select the Security tab and on the left frame, choose the Install Certificate option.**



3. **Fill out the form to install your certificate:**
   - Certificate For: This Server
   - Cryptographic Module: Select the appropriate *user@realm-name*.
   - Key Pair File Password: Provide the password for the *user@realm-name* that owns the key that was generated earlier.

- Certificate Name: In most cases, you can leave this blank. If you choose to provide a name, it will alter the name the web server uses to access the certificate and key when running with SSL support.

4. **Choose Message text (with headers) and paste the certificate you copied earlier.**
5. **Click the OK button at the bottom of the page, pasting the certificate you copied from the certificate authority into the Message box.**

   You are shown some basic information about the certificate.

6. **If everything looks correct, click the Add Server Certificate button.**

   On-screen messages tell you to restart the server. This is not necessary as 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.

Configuring iPlanet Web Server 6.0

Now that your web server and the Server Certificate are installed, you must configure the web server for SSL.

▼ To Configure the iPlanet Web Server 6.0

1. **Click the Preferences tab near the top of the page. Select the Edit Listen Sockets option on the left frame.**

   The main frame lists all the listen sockets set for the web server instance.

   a. **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 On.

   b. **Click the OK button to apply these changes.**

      In the security field of the Edit Listen Sockets page, there should now be an Attributes link.

2. **Click the Attributes link.**
3. **Enter** *user@realm-name* **password to authenticate to the** *user@realm-name* **on the system.**

4. **Select SSL settings from the pop-up window.**

   You can choose Cipher Default settings, SSL2, or SSL3/TLS. The Default choice does not show the default settings. The other two choices require you to select the algorithms you want to enable.

5. **Select the certificate for the *user@realm-name* followed by `:Server-Cert` (or the name you chose if it is different).**

   Only keys that the appropriate *user@realm-name* owns appear in the Certificate Name field.

6. **When you have chosen a certificate and confirmed all the security settings, click the OK button.**

7. **Click the Apply link in the far upper right corner, to apply these changes before you start your server.**

8. **Click 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, a pop-up window shows up for password prompt. This window is not resizable, and you might have problem submitting the change.

   There are two workarounds for the problem noted above:

   - Click the Load Configuration Files instead.
   - Start up the web server first, and click on the Apply Changes button.

9. **Provide the requested passwords in the dialog boxes to start the server.**

   You will be prompted for one or more passwords. At the Module Internal prompt, provide the password for the web server Trust Database.

   At the Module *user@realm-name* prompt, enter the password you set when you created *user* in the *realm-name* using `secadm`.

10. **Verify the new SSL-enabled web server with a browser by going to the following URL:**

    `https://`*hostname.domain*`:`*server_port*`/`

    Note that the default *server_port* is 443.

CHAPTER 6

Enabling Apache Web Servers

This chapter explains how to enable the Sun Crypto Accelerator 1000 board for use with Apache web servers.

This chapter includes the following sections

- "Enabling Apache Web Servers" on page 39
- "Creating a Certificate" on page 42

Enabling Apache Web Servers

Apache Web Server 1.3.12 is provided with the Solaris 8 7/01 operating environment. The following instructions are for that specific release of Apache Web Server. Refer to the Apache web server documentation for more information about using Apache web servers.

▼ 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 httpd.conf-example /etc/apache/httpd.conf
   ```

   Replace the `ServerName` in the file with your server name.

2. **Start** `sslconfig`**.**

```
# /opt/SUNWconn/crypto/bin/sslconfig
```

3. **Select** 2 **to configure your Apache web server to use SSL:**

```
Sun Crypto Accelerator Installation
------------------------------------------------------------
This script will install the Sun Crypto Accelerator
cryptographic modules for iPlanet Web Server
or Apache.

Please select the type of web server you wish to configure
to use the Sun Crypto Accelerator:
------------------------------------------------------------
1. Configure iPlanet Web Server for SSL
2. Configure Apache for SSL
3. Work with iPlanet and Apache keys

Your selection (0 to quit):
```

4. **Provide the directory where the Apache binaries exist.**

   On Solaris systems, this is usually `/usr/apache`.

```
Please enter the directory where the Apache
binaries and libraries exist [/usr/apache]: /usr/apache
```

5. **Provide the location of the configuration files for Apache.**

   On Solaris systems, this is usually `/etc/apache`.

```
Please enter the directory where the Apache
configuration files exist [/etc/apache]: /etc/apache
```

6. **Create an RSA keypair for your system.**

   If you choose not to, you must go back later and use `sslconfig` to generate keys.

```
Do you wish to create a new RSA keypair and certificate request?
[Y/N]:
```

   If you answer No to this question, skip to "To Create a Certificate" on page 42.

7. **Provide the directory for storing the keys.**

   If this directory does not exist, it is created.

```
Where would you like the keys stored? [/etc/apache/keys]:
/etc/apache/keys
```

8. **Choose a base name for the key material.**

   This name is appended with different suffixes to distinguish key files, certificate request files and later on, certificate files from one another.

```
Please choose a base name for the key and request file:
```

9. **Provide a key length between 512 and 2048 bits.**

   For most web server applications, 1024 bits is sufficiently strong, but you can opt for stronger keys if you prefer.

```
What size would you like the RSA key to be [1024]? 1024
Generating RSA private key, 1024 bit long modulus
..........++++++
......++++++
e is 65537 (0x10001)
```

10. **Create your PEM pass phrase.**

    This pass phrase is used to protect the key material. Be sure to select a strong pass phrase, but one that you can remember. If you forget the password, you will be unable to access your keys.

```
Enter PEM pass phrase:
Verifying password - Enter PEM pass phrase:
```

Caution – You must remember the pass phrase you enter. Without the pass phrase, you cannot access your keys. There is no way to retrieve a lost pass phrase.

Creating a Certificate

The following procedure describes how to create the certificate required to enable Apache web servers to use the Sun Crypto Accelerator 1000 board.

▼ To Create a Certificate

1. **Create a certificate request using the keys you just created.**

   You must first enter the password 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
   - 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
   - Organizational 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: Website Domain that is typed in a visitor's browser
   - Email Address: Contact information for requestor

```
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
SSL Server Name (eg, www.company.com) []:www.fictional-company.com
Email Address []:admin@fictional-company.com
```

2. **Modify the** `/etc/apache/httpd.conf` **file as directed.**

   You are shown information concerning your key and certificate files. You are also instructed on how to modify the `/etc/apache/httpd.conf` file for use with the Sun Crypto Accelerator 1000.

```
The keyfile is stored in /etc/apache/keys/ap6-key.pem.
The certificate request is in /etc/apache/keys/ap6-certreq.pem.

You will need to edit /etc/apache/httpd.conf for the following items:

You must specify the ports that Apache will listen to for
SSL connections, as well as for non-SSL connections.  One
way to accomplish this is to add the following lines in
the Listen section:

Listen 80
Listen 443

In the LoadModule section, add the following:

LoadModule ssl_module /usr/apache/libexec/mod_ssl.so.1.3.12

In the AddModule section, add the following:

AddModule mod_ssl.c
```

3. **If you chose not to set up a** `VirtualHost` **the** `SSLEngine`**,** `SSLCertificateFile`**, and** `SSLCertificateKeyFile` **directives must be placed in** `httpd.conf` **file, just above the** `SSLPassPhraseDialog` **directive.**

```
You may need a virtual host directive similar to
what is shown below:

<VirtualHost _default_:443>
        SSLEngine on
        SSLCertificateFile /etc/apache/keys/ap6-cert.pem
        SSLCertificateKeyFile /etc/apache/keys/ap6-key.pem
</VirtualHost>

You must add the following line after all of your VirtualHost
definitions:

SSLPassPhraseDialog exec:/opt/SUNWconn/crypto/bin/sslpassword

Other SSL-related directives and their explanations
can be found in the Sun Crypto Accelerator documentation.

Other Apache-related directives may need to be configured
in order to start your Apache web server.  Please refer
to your Apache documentation.

<Press ENTER to continue>
```

If you answered no to the question in Step 6, you will also be given additional information on how to generate key material later:

```
Since you did not create keys, you will need to
make sure that you have a key file and a certificate
file in place before enabling SSL for Apache.

You can create a new key file and certificate request
by selecting the "Generate a keypair and request a
certificate for Apache" option after choosing
"Work with iPlanet and Apache keys" from the
sslconfig main menu.
```

4. **Select** `0` **to quit when you finish with** `sslconfig`**.**

5. **Copy your certificate request with the headers from** `/etc/apache/keys/`*base_name*`-certreq.pem` **(where base_name was set in Step 8)and hand it off to your certificate authority.**

6. **Once the certificate is generated, create the certificate file** `/etc/apache/keys/`*base_name*`-cert.pem` **and paste your certificate into it.**

7. **Start the Apache web server.**

   This assumes your Apache binary directory is `/usr/apache/bin`. If this is not your binary directory, type in the correct directory.

```
# /usr/apache/bin/apachectl start
```

8. **Enter your PEM pass phrase when prompted for it.**

9. **Verify the new SSL-enabled web server with a browser by going to the following URL:**

   `https://`*server_name*`:`*server_port*`/`

   Note that the default *server_port* is 443.

CHAPTER 7

Diagnostics and Troubleshooting

This chapter describes diagnostic tests and troubleshooting for the Sun Crypto Accelerator 1000 software. It includes the following sections:

- "SunVTS Diagnostic Software" on page 47
- "Troubleshooting the Sun Crypto Accelerator 1000" on page 51

SunVTS Diagnostic Software

The SunVTS test `dcatest`, delivered in package `SUNWdcav` on the *Sun Crypto Accelerator 1000* CD, operates with the core SunVTS test control and user interface, delivered in packages `SUNWvts` and `SUNWvtsx` on the Solaris Supplement CD, to provide diagnostics for the Sun Crypto Accelerator 1000 board.

Refer to the SunVTS documentation for instructions on how to run and monitor these diagnostics tests. These documents are available on the Solaris on Sun Hardware AnswerBook, which is provided on the Solaris Supplement CD for the Solaris release on your system.

Note – SunVTS can be used only if you have installed the SunVTS packages from the Solaris Supplement CD.

▼ To Run dcatest

1. **As superuser, start SunVTS.**

```
# /opt/SUNWvts/bin/sunvts
```

   Refer to the *SunVTS User's Guide* for detailed instructions on starting SunVTS.

   The following instructions assume that you have started SunVTS using the CDE user interface.

2. **On the SunVTS Diagnostic main window, set the System Map to Logical mode.**
3. **Disable all tests by clearing their check boxes.**
4. **Select the check box for OtherDevices then select the plus box for OtherDevices to display all tests in the OtherDevices group.**
5. **Clear check boxes in the OtherDevices group that are not named** `dcatest`**.**
   - If a `dcatest` is displayed then go to Step 6.
   - If a `dcatest` is not displayed, probe the system to find it by selecting Reprobe system in the Commands drop down menu..

   Refer to the SunVTS documentation for the exact procedure. When the probe completes and a `dcatest` is displayed, continue to Step 6.

6. **Click one of the instances of** `dcatest` **then right-click and drag to display the Test Parameter Options.**

   These options, which only pertain to the `dcatest`, are described in "Test Parameter Options for dcatest" on page 49.

7. **After you have made all selections, click Within Instance Apply to change the selected instance of** `dcatest`**, or click Across All Instances Apply to change all checked instances of** `dcatest`**.**

   This action removes the pop-up and returns you to the Sun Diagnostic main window.

8. **Click one of the instances of** `dcatest` **then right-click and drag to display the Test Execution Options.**

   An alternate method of displaying Test Execution Options is to click the Options pop-up then click Test Executions. These options are generic SunVTS controls that affect all tests. Refer to the SunVTS documentation for detailed information.

9. **When you have made all selections, click Apply to remove the pop-up window and return to the Sun Diagnostic main window.**
10. **Click Start to run the selected tests.**

11. Click Stop to stop all tests.

Test Parameter Options for dcatest

TABLE 7-1 displays the Test Parameter Options for dcatest as detailed in Step 6 in "To Run dcatest" on page 48. The board type being tested is indicated in the Configuration area of the pop-up.

TABLE 7-1 Test Parameter Options for dcatest

Option Label	Description
Test_Sel	A decimal value that specifies the combination of sub-tests to be run. A value of 0 (zero) selects all tests. Each sub-test is assigned a power-of-2 number. An individual sub-test can be selected by entering the number assigned to the sub-test. Multiple sub-tests can be selected by entering the sum of the numbers assigned to the desired sub-tests. The default setting is zero.
Info_Print	Enables or disables the printing of Information (INFO type) messages. The default setting is Enable.

TABLE 7-2 describes the dcatest subtests.

TABLE 7-2 dcatest Subtests

Test Name	Number	Description
ALL	0	All tests are executed.
SHOWINFO	1	Prints an INFO type message showing provider and device under test information.
3DES	2	Tests 3DES Bulk Encryption.
RSA	4	Tests RSA Public and Private Keys.
DSA	32	Tests DSA Signature Verification.
Random	64	Tests Random and Pseudo-Random Number Generation. Prints an INFO type message showing numbers generated.

Messages generated by sub-tests are displayed in the Test Messages area of the SunVTS Diagnostic main window. Messages generated by the sub-tests are grouped by type:

- INFO type messages which provide non-critical information if Info_Print option is enabled in the Test Parameters pop-up are printed in the Test Messages area and recorded in the Information Log.
- FATAL error type messages are always displayed and are recorded in the Test Error Log and the Information Log.
- VERBOSE type messages which track progress through sub-tests are only displayed if the VERBOSE option is enabled in the Test Execution pop-up window. VERBOSE messages are not recorded in any log.

  A quiet mode of testing that displays and logs `dcatest` FATAL error messages can be selected by disabling VERBOSE and `Info_Print` options.

`dcatest` Command-Line Syntax

If you choose to run `dcatest` from the command line instead of the CDE environment then all arguments must be specified in the command-line string.

In 32-bit mode, the path to `dcatest` is `/opt/SUNWvts/bin/`. In 64-bit mode, the path to `dcatest` is `/opt/SUNWvts/bin/sparcv9/`.

The following example shows the syntax for the 32-bit mode command:

```
/opt/SUNWvts/bin/dcatest -f [Standard Command-Line Arguments]
[-o [dev=dcan][,testsel=n][,infodis]]
```

Refer to the *SunVTS Test Reference Manual* for a definition of the standard command-line arguments. Since the `dcatest` is a Functional Mode test, `-f` must be included. Include `-u` to display a usage message, or `-v` for VERBOSE messages. Items enclosed in square brackets above denote optional entries. Omission of an option produces the default behavior for that option, as stated in TABLE 7-3.

TABLE 7-3 `dcatest` Command-Line Syntax

Argument	Description
`dev=dca`*n*	Specifies the instance of the device to test such as dca0 or dca2. Defaults to dca0 if not included.
`testsel=`*n*	Specifies the sub-tests to be executed where n can be a number from 0 to 127. Defaults to zero if not included.
`infodis`	Included if INFO type messages are to be disabled. Defaults to Info_Print Enabled, if not included.

Troubleshooting the Sun Crypto Accelerator 1000

To determine whether the Sun Crypto Accelerator 1000 device is listed in the system, from the OpenBoot PROM (OBP) prompt, type `show-devs` to display the list of devices. You should see lines in the list of devices, similar to the examples below, specific to the Sun Crypto Accelerator 1000 board:

```
ok show-devs
 . . .
/pci@1f,0/pci@1/pci108e,5455@2
 . . .
```

In the above example, the `pci108e,5455` identifies the device path to the Sun Crypto Accelerator 1000 board. There is no firmware on this board so OBP level diagnostics are not available.

The Sun Crypto Accelerator 1000 does not contain lights or other indicators to reflect cryptographic activity on the board. In order to determine whether cryptographic work requests are actually being performed on the board, use the `kstat`(1M) command to display the device usage:

```
# kstat -m dca -i 0 -n dca0

module: dca                             instance: 0
name:   dca0                        class:    misc
    3desbytes                           3040
    3desjobs                            5
    crtime                              65.342725895
    dsasign                             0
    dsaverify                           0
    rngbytes                            10592
    rngjobs                             187
    rngsha1bytes                        16328
    rngsha1jobs                         327
    rsaprivate                          9
    rsapublic                           0
    snaptime                            106956.467004482
```

Displaying the `kstat` information indicates whether cryptographic requests or "jobs" are being sent to the Sun Crypto Accelerator 1000 board. A change in the "jobs" values over time indicates that the board is accelerating cryptographic work requests sent to the Sun Crypto Accelerator 1000 board. If cryptographic work requests are not being sent to the board, verify your web server configuration per the web server specific configuration.

It is not always possible to determine where a cryptographic request has been performed, and like cryptographic requests may be performed at a different location, subject to the subsystem loading at the time of request submission.

Do not attempt to interpret the kernel / driver statistic values returned by `kstat`(1M). These values are maintained within the driver to facilitate field support. The meanings and actual names may change over time.

APPENDIX **A**

Administering the Sun Crypto Accelerator 1000 Board With iPlanet Web Servers

This appendix provides an overview of the security features of the Sun Crypto Accelerator 1000 board as it is administered with iPlanet web servers.

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

This appendix includes the following sections:

- "Concepts and Terminology" on page 53
- "Setting Up and Managing Realms" on page 61
- "Setting Up and Managing User Accounts" on page 65

Concepts and Terminology

Realms and users must be created for applications that communicate with the Sun Crypto Accelerator 1000 through a PKCS#11 interface, such as iPlanet Web Server.

Users within the context of the Sun Crypto Accelerator 1000 are unique owners of cryptographic keying material. Each user may own multiple keys. A user may want to own multiple keys to support different configurations, such as a "production" key and a "development" key (to reflect the user's different organizations), or may require multiple keys to facilitate a high availability (HA) configuration. Note that the term "user" or "user account" refers to Sun Crypto Accelerator 1000 users, not traditional UNIX user accounts. There is no fixed mapping between UNIX usernames and Sun Crypto Accelerator 1000 usernames.

Realms are logical partitions of users and their keying material. Realms provide the ability to contain multiple users. An advantage of partitioning users by realms is that a unique namespace is maintained for each realm. This allows realm contents to be managed separately.

A typical installation contains a single realm with a single user. For example, such a configuration might consist of a single realm “webserver” and a single user within that realm, “nobody.” This would allow the user “nobody” to own and maintain access control of the server keys within the single realm.

The flexibility exists to construct additional realms to partition users and keying material. A more complex configuration might consist of multiple realms, for example, “finance”, “legal,” and “engineering”. Each realm maintains a unique namespace. For example, the user “webserv” in the finance realm is a different user account than “webserv” in the engineering realm.

An administrative tool, `secadm`, is used to manage Sun Crypto Accelerator 1000 realms and users.

Realms, Users, and the iPlanet Web Server

When the iPlanet Web Server needs to reference a key managed by the Sun Crypto Accelerator 1000, it uses a “token name” to indicate that the key is managed by the hardware and not its internal software database.

The Sun Crypto Accelerator 1000 creates its token names by combining a user account and a realm name together with the “@” symbol. In the typical installation example above, a single realm, “webserver,” was created with a single user, “nobody.” The token label that iPlanet Web Server would use to reference keys owned by user “nobody” in the realm “webserver” would be “nobody@webserver”. The password for user nobody (which is set when the user is created using `secadm`) must be used when requesting a certificate, installing a certificate, or authenticating to start the iPlanet Web Server.

Tokens and Slot Files

iPlanet web servers access key material through tokens, also known as slots. Slot files are a technique for Sun Crypto Accelerator 1000 administrators to selectively present only specific tokens to a given application.

If no slot file exists, the Sun Crypto Accelerator 1000 software presents a default set of tokens to iPlanet web servers. In this case, one token is presented per realm, with a token name of `nobody@`*realm-name*.

Example

There are three realms, engineering, finance, and legal. The following tokens are presented to the iPlanet web server:

- nobody@engineering
- nobody@finance
- nobody@legal

However, for any of these names to be usable, a user “nobody” must exist in each of these realms.

Slot Files

To override the default case, a slot file must exist. Slot files are text files that contain one or more token names, one per line. An iPlanet web server presents only the tokens listed in this file. The methods of specifying slot files are as follows (in order of precedence):

1. The file `$HOME/.SUNWconn_crypto_slots`

   This file must exist in the home directory of the UNIX user that iPlanet web server runs as. iPlanet web server may run as a UNIX user who has no home directory and in which case this approach may not be feasible.

2. The file `/etc/opt/SUNWconn/crypto/slots`

   The `/etc/opt/SUNWconn/crypto/slots` file is a global default, and is used if a `.SUNWconn_crypto_slots` file does not exist in the user’s home directory.

Following is an example of the contents in a slot file:

```
webserv@engineering
webserv@finance
```

If none of the above files are found, then the default method described in “Tokens and Slot Files” on page 54 is used.

See Chapter 3 for more information on token names as they pertain to iPlanet web server configuration.

Using secadm

The secadm program offers a command-line interface to the Sun Crypto Accelerator 1000.

To access the secadm program easily, place the Sun Crypto Accelerator 1000 tools directory in your search path, for example:

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

The secadm command syntax is:

secadm [-h]

secadm [-y] [-f *filename*]

secadm [-y] [-r *realm-name*] [-u *username* | -s *admin-name*] *command*

The command is located in the /opt/SUNWconn/crypto/bin/ directory.

TABLE A-1 shows the options for the secadm tool

TABLE A-1 secadm Options

Option	Meaning
-h	Display command help for secadm and exit.
-f *filename*	Read in one or more commands from *filename* and exit.
-r *realm-name*	Used in single-command mode only. The -r option tells secadm to execute the supplied command in realm *realm-name*.
-s *admin-name*	Used in single-command mode only. The -s option tells secadm to log in as a system administrator using *admin-name* as the login name. *admin-name* must be a UID 0 (zero) UNIX user (such as root). The login will take place before the supplied command is executed.
-u *username*	Used in single-command mode only. The -u option tells secadm to log in as *username*. The login takes place before the supplied command is executed.
-y	Forces a “yes” answer on any command that would normally prompt for a confirmation.

Modes of Operation

`secadm` can run in one of three modes. These modes differ mainly in how commands are passed into `secadm`. The three modes are single-command mode, file mode, and interactive mode. Each mode requires a different password.

Single-Command Mode

In single-command mode, the user specifies the command to be run by `secadm` after all the command-line switches are specified. For example, the following command would show all realms in existence and return the user to the command shell prompt.

```
$ secadm show realm
```

The following command performs a login as the system administrator, and creates the user webserv in the realm engineering.

```
$ secadm -r engineering -s root create user=webserv
Password:
Initial password:
Confirm password:
User webserv created successfully.
```

Note that the password entered at the “Password:” prompt requires the System Administrator password, while the password entered at the “Initial password:” and “Confirm password:” prompts require the password for the newly created user.

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, the user specifies a file from which `secadm` reads one or more commands. The file must be ASCII text, consisting of one command per line. Begin each comments a “#” character. If the file mode option is set, `secadm` ignores any command line arguments after the last option. The following example runs the commands in `deluser.scr` and answers all prompts in the affirmative:

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

Interactive Mode

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.

Entering Commands With `secadm`

The `secadm` program has a command language that must be used to interact with the Sun Crypto Accelerator 1000 board. Commands are entered using all or part of a word (enough to uniquely identify that word from any other possibilities). Using “sh” instead of “show” would work, but “lo” is ambiguous because it could be “login” or “logout”.

The following example shows entering commands using entire words:

```
secadm{root@engineering}# show user
User                                    Status
------------------------------------------------------
webserv                                 enabled
alice                                   enabled
bob                                     enabled
------------------------------------------------------
```

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

An ambiguous command produces an explanatory response:

```
secadm{root@engineering}# lo
Ambiguous command: lo
```

Authentication using `secadm`

Many commands, particularly those that deal with user accounts and keys, require you to authenticate as a system administrator or as a user. System administrators must authenticate to the Sun Crypto Accelerator 1000 to perform actions such as creating realms, creating user accounts, enabling and disabling user accounts, and deleting realms and user accounts. Authentication as a user is necessary in order to

change a user's password or to list key objects owned by that user. TABLE A-2 shows which commands can be used by the system administrator and which can be used by the user.

TABLE A-2 Command Matrix

Command	Authenticate	Credential Held	Authenticated User
`create user=`*username*	No	Yes	System administrator
`create realm=`*realm-name*	Yes	No	System administrator
`delete user=`*username*	No	Yes	System administrator
`delete realm=`*realm-name*	Yes	No	System administrator
`disable user=`*username*	No	Yes	System administrator
`enable user=`*username*	No	Yes	System administrator
`exit`	No	No	All
`login`	Yes	No	User
`logout`	No	No	All
`passwd`	Yes	Yes	User
`set realm=`*realm-name*	No	No	All
`show class`	No	No	All
`show key`	No	Yes	User
`show realm`	No	No	All
`show user`	No	Yes	System administrator
`su`	Yes	No	System administrator
`quit`	No	No	All
`unset realm`	No	No	All

To authenticate as a system administrator, you must provide a UNIX username that is UID 0 (such as root), and provide the password when prompted. Users require the password that was set for them when the user was created. When logging in as either a system administrator or a user, you must select a realm first.

To log in as a user, type:

```
secadm{realm-name}> login user=username
```

To log in as a system administrator, type:

```
secadm{realm-name}> su
```

When logged in as a user or a system administrator, the secadm prompt shows the currently logged in user. A user login and a system administrator login are differentiated by the last character in the prompt. Users have an angle-bracket (>) while system administrators have a pound sign (#). If you are currently logged in as a user or system administrator and try to log in as another user or system administrator, your current credentials will be lost when the new login is successful For example:.

```
secadm> set realm=engineering
secadm{engineering}> login user=webserv
Password:
secadm{webserv@engineering}> su
System Administration Login Required
Login: root
Password:
secadm{root@engineering}# logout
secadm{engineering}>
```

Getting Help for Commands

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

```
secadm> create ?
Usage: create {user=<username> | realm=<realm-name>}

secadm> show ?
Sub-Command                    Description
-----------------------------------------------------
class                          Show all realm classes
key                            Show all key objects in a realm
realm                          Show all realms
user                           Show all system accounts
```

Entering a “?” gives you the list of valid command words, for example:

```
secadm> ?
Sub-Command                    Description
---------------------------------------------------
create                         Create users and accounts
delete                         Delete users and accounts
disable                        Disable a user
enable                         Enable a user
exit                           Exit secadm
login                          Login as a user
logout                         Logout current session
passwd                         Change password for a user
set                            Set current working realm
show                           Show system settings
su                             Authenticate as the System Administrator
quit                           Exit secadm
unset                          Unset secadm operating parameters
```

If you want to get help in command-line mode, you must remember that in some cases the “?” character is interpreted by the shell you are working in. Make sure you use the command shell escape character before the question mark.

Quitting the `secadm` Program

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

Setting Up and Managing Realms

A realm is a repository for key material. Associated with a realm are administrators and users. Realms not only provide storage, but a means for key objects to be owned by user accounts. This allows keys to be hidden from applications that do not authenticate as the owner. Realms have two components:

- Key objects: These are long term keys that are stored for applications such as iPlanet Web Server.
- User accounts: These accounts provide applications a means to authenticate and access specific keys.

While only one realm is necessary, there can be multiple realms, and each realm has its own set of user accounts. For example, if an application authenticates as user `webserv` and needs to access keys in a realm, then the user account `webserv` must exist in that realm.

Creating a Realm

Creating a realm creates the directories, files, and other resources necessary to store long-term key objects. To create a realm, the administrator must use the `create realm` command and provide the name of the realm to be created. Regardless of any currently held credentials, the system administrator must authenticate for this command to be completed successfully. When prompted for the password, enter the UNIX system administrator password. For example:

```
secadm> create realm=engineering
System Administrator Login Required
Login: root
Password:
Realm engineering successfully created.
```

You can name realms to suit your use. For example, you may want to set up realms for different departments, such as finance and engineering. In such case, you would name the realms `finance` and `engineering`. For example:

```
secadm> create realm=finance
System Administrator Login Required
Login: root
Password:
Realm finance successfully created
```

Setting the Current Working Realm

`secadm` can only manage keys and user accounts in one realm at a time. Most commands that deal with realms and user accounts require you to select a realm first. To select a realm, issue the `set realm` command, as shown in the following example:

```
secadm> set realm=finance
secadm{finance}>
```

When you have selected the realm, the `secadm` prompt shows the realm name in curly brackets.

If you no longer want to work in the realm you are currently in, you can either set the current working realm to a new value or unset the realm. Changing or unsetting the current working realm will also automatically log out any currently authenticated user or system administrator in that realm. For example:

```
secadm{finance}> set realm=engineering
secadm{engineering}> unset realm
secadm>
```

Populating the Realm with Users

These usernames are known only within the domain of the Sun Crypto Accelerator 1000 and do not need to be identical to the UNIX username that the web server process actually runs as. Before attempting to create the user, remember that you must first select the correct realm and login as the system administrator. For example:

```
secadm> set realm=engineering
secadm{engineering}> su
System Administrator Login Required
Login: root
Password:
secadm{root@engineering}#
```

If you only need one realm user, you can avoid setting up a slot file by using the realm name “nobody.” The following example creates the user “nobody” in the realm “engineering” and sets the password for “nobody@engineering”, defined as *user@realm-name* in TABLE 3-1.

```
secadm{root@engineering}# create user=nobody
Initial password:
Confirm password:
User nobody successfully created.
```

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

Caution – You must remember the password you enter. Without the password, you cannot access your keys. There is no way to retrieve a lost password.

Listing Realms

You can list information on a realm by issuing the `show realm=`*realm-name* command.

```
secadm> show realm
Realm Name
-----------------------------------------------------
engineering
finance
-----------------------------------------------------
```

Listing Realm Classes

Realm classes are key management modules that control how realms manage key objects, user accounts, and authentication data. The only realm class currently supported by the Sun Crypto Accelerator 1000 is the `SUNW_filesys` realm class. To list all realm classes supported, use the `show class` command.

```
secadm> show class
Realm Class
-----------------------------------------------------
SUNW_filesys
-----------------------------------------------------
```

Deleting a Realm

You can delete a realm by issuing the `delete realm` command and providing the name of the realm to be removed. When you issue the command, `secadm` prompts you for a yes/no confirmation to remove the realm. As with creating a realm, the system administrator must authenticate before this command is executed. In addition, you cannot delete a realm that is in use. To free references to realms, you may have to shut down the web server and/or administration server.

Setting Up and Managing User Accounts

User accounts provide a way for applications to authenticate to the Sun Crypto Accelerator 1000 and also allow a means of separating keys within a realm. Keys owned by one user account are not accessible to applications that are unauthenticated or authenticate to that realm as another user. For all these commands, a realm must be selected and the system administrator must be logged into that realm using the `secadm su` command.

Creating Users

- **Issue the** `create user` **command to create a user.**

  This command requires the username in the form `create user=`*username*.

```
secadm{root@engineering}# create user=username
Initial password:
Confirm password:
User username created successfully.
```

Caution – You must remember the password you enter. Without the password, you cannot access your keys. There is no way to retrieve a lost password.

Listing Users

Only the system administrator can list the users in a realm. The system administrator must issue the `show user` command. This command only lists the users in the currently selected realm.

- **Issue the** `show user` **command.**

```
secadm{root@engineering}# show user
User                                    Status
-----------------------------------------------------------
webserv                                 enabled
alice                                   enabled
bob                                     enabled
-----------------------------------------------------------
```

Changing User Passwords

Only the individual logged-in user using the `secadm login` command can change that user's password. You must know your current password before you can set a new password.

- **Issue the** `passwd` **command.**

```
secadm{username@realm-name}> passwd
Enter current password:
Enter Password:
Confirm Password:
Password successfully changed for user username.
```

Caution – You must remember the password you enter. Without the password, you cannot access your keys. There is no way to retrieve a lost password.

Enabling or Disabling Users

Only system administrators have the ability to enable or disable users. By default each user is created in the enabled state.

- **To disable a user account enter the** `disable user=`*username* **command**.

```
secadm{root@engineering}# disable user=username
User is now disabled.
```

All attempts to authenticate to a disabled user account will fail. None of the keys are altered in any way, however. When the account is re-enabled all the keys that are owned by that user are once again accessible by the authenticated application.

- **To enable an account, enter** `enable user=`*username* **command.**

```
secadm{root@engineering}# enable user=username
User is now enabled.
```

Deleting Users

- **Issue the** `delete user` **command by specifying the user to be deleted.**

  The system administrator must provide the user account name to be deleted.

  Keys associated with users are deleted when the command is issued. `secadm` prompts the system administrator for a yes/no confirmation before deleting the user.

```
secadm{root@engineering}# delete user=username
Delete user webserv? [Y/N]: y
User username deleted successfully.
```

APPENDIX **B**

Manual Pages

This appendix provides descriptions of the `man` pages included with Sun Crypto Accelerator 1000 software.

The man pages can be viewed using the command:

```
man -M /opt/SUNWconn/man page
```

TABLE B-1 lists and describes the available `man` pages.

TABLE B-1 `man` pages for Sun Crypto Accelerator 1000

man page	Description
`cryptio(7d)`	The `cryptio` device driver provides access control to the underlying hardware cryptographic accelerator. The `cryptio` driver requires the presence of layered software for applications and kernel clients to access the provided services.
`dca(7d)`	The `dca` device driver is a Sun cryptographic provider leaf driver that provides access control to the underlying hardware cryptographic accelerator. The `dca` driver requires the presence of layered software for applications and kernel clients to access the provided services.
`kcl(7d)`	The `kcl` device driver is a multithreaded loadable kernel module providing support for Sun cryptographic provider drivers. The `kcl` driver requires the presence of layered software for applications and kernel clients to access the provided services.
`kcpi(7d)`	The `kcpi` device driver is a multithreaded loadable kernel module providing support for Sun cryptographic provider drivers. The `kcpi` driver requires the presence of layered software for applications and kernel clients to access the provided services.

TABLE B-1 `man` pages for Sun Crypto Accelerator 1000

`man` page	Description
`secadm(1m)`	`secadm` is the administration utility for the Sun Crypto Accelerator. The `secadm` command is used to manually manipulate the configuration, account, and keying databases associated with the Sun Crypto Accelerator. `secadm` handles sensitive cryptographic key information.
`secd(1m)`	The `secd` daemon provides administrative access services to the `secadm` application.
`sslconfig(1m)`	`sslconfig` is the configuration utility for the Sun Crypto Accelerator 1000.

APPENDIX **C**

SSL Configuration Directives for Apache Web Servers

This appendix lists directives for configuring SSL support for Apache web servers with Sun Crypto Accelerator 1000 software. Configure directives in your `http.conf` file. Refer to the Apache documentation for more information.

1. `SSLPassPhraseDialog exec:`*program*

   Context: global

   This directive informs the Apache web server that the specified *program* should be executed to collect the password for key file. *program* should print the collected password to standard output.

   If multiple key files are present, and have common passwords, then *program* will only be executed once (each collected password is tried before running *program* again.)

   program is executed with two arguments, the first is the name of the server, in the form *servername*:*port*, for example, `www.fictional-company.com:443`. (Port 443 is the typical port for SSL based web servers.) The second is a the type of key in the key file (*keytype*). *keytype* can be either RSA or DSA.

Note – Because this program can be executed during system startup, be sure to design it to cope with the situation where the console is not a `tty` device (that is a `tty`(3c) will return false).

   The supplied program `/opt/SUNWconn/crypto/bin/sslpassword` can be used for the *program* executable. This program automatically prompts for the password, suppressing the display of the password as it is entered.

   The supplied `sslpassword` program also automatically searches for passwords in files, which can be used to avoid user interaction when the web server starts up. Passwords for key files are searched for in files named

`/etc/apache/`*servername*`:`*port*`.`*keytype*`.pass`. If this file is not present, then the file `/etc/apache/default.pass` is used. The content of these password files is just the unencrypted password on a line by itself.

Note – Password files should be protected by permissions so that only the UNIX user that the web server runs as can read the file. This should be the same user as configured with the standard Apache `User` directive.

If not specified, the default behavior is to use an internal prompting mechanism. Sun customers are advised to avoid the default, and use the supplied `sslpassword` program instead, to avoid problems with interaction at system startup.

2. `SSLEngine` (`on|off`)

   Context: global, virtual host

   This directive is used to enable the SSL protocol. It is typically used in a virtual host to enable SSL on a subset of servers. One form commonly used is:

```
<VirtualHost _default_:443>
SSLEngine on
</VirtualHost>
```

which configures the use of SSL for any servers listening on port 443 (the standard HTTPS port). If not present, this is turned off by default.

3. `SSLProtocol [+-]`*protocol*

   Context: global, virtual host

   This directive configures the protocol(s) that the server should use for SSL transactions.

   The available protocols are listed and described in TABLE C-1:

TABLE C-1 SSL Protocols

Protocol	Description
`SSLv2`	the original defacto standard SSL protocol from Netscape
`SSLv3`	an updated version of the SSL protocol, it is supported by most popular web browsers
`TLSv1`	an update to `SSLv3` currently undergoing IETF standardization, with minimal browser support at the time of this writing
`all`	enable all protocols

Using the plus (+) or minus (-) signs, protocols can be added or removed. For example, to disable support for SSLv2, the following directive could be used:

```
SSLProtocol all -SSLv2
```

which is also equivalent to

```
SSLProtocol +SSLv3 +TLSv1
```

4. `SSLCipherSuite` *cipher-spec*

   Context: global, virtual host, directory, `.htaccess`

   The SSLCipherSuite directive is used to configure which SSL ciphers are available for use and their preference. In global context or virtual host context, it is used during the initial SSL handshake. In per-directory context, it forces an SSL renegotiation to use the named ciphers. The renegotation takes place after the request is read, but before the response is sent.

   The *cipher-spec* is a colon delimited list of the ciphers described in TABLE C-2.

TABLE C-2 Available SSL Ciphers

Cipher-Tag	Protocol	Key Exchange	Auth.	Encryption	MAC	Type
DES-CBC3-SHA	SSLv3	RSA	RSA	3DES (168 bit)	SHA1	
DES-CBC3-MD5	SSLv2	RSA	RSA	3DES (168 bit)	MD5	
RC4-SHA	SSLv3	RSA	RSA	ARCFOUR (128 bit)	SHA1	
RC4-MD5	SSLv3	RSA	RSA	ARCFOUR (128 bit)	MD5	
RC4-MD5	SSLv2	RSA	RSA	ARCFOUR (128 bit)	MD5	
RC2-CBC-MD5	SSLv2	RSA	RSA	ARCTWO (128 bit)		
DES-CBC-SHA	SSLv3	RSA	RSA	DES (56 bit)	SHA1	
RC4-64-MD5	SSLv2	RSA	RSA	ARCFOUR (64 bit)	MD5	
DES-CBC-MD5	SSLv2	RSA	RSA	DES (56 bit)	MD5	
EXP-DES-CBC-SHA	SSLv3	RSA (512 bit)	RSA	DES (40 bit)	SHA1	export
EXP-RC2-CBC-MD5	SSLv2	RSA (512 bit)	RSA	ARCTWO (40 bit)	SHA1	export
EXP-RC2-CBC-MD5	SSLv3	RSA (512 bit)	RSA	ARCTWO (40 bit)	SHA1	export
EXP-RC4-MD5	SSLv3	RSA (512 bit)	RSA	ARCFOUR (40 bit)	MD5	export
EXP-RC4-MD5	SSLv2	RSA (512 bit)	RSA	ARCFOUR (40 bit)	MD5	export
NULL-SHA	SSLv3	RSA	RSA	None	SHA1	

TABLE C-2 Available SSL Ciphers

Cipher-Tag	Protocol	Key Exchange	Auth.	Encryption	MAC	Type
NULL-MD5	SSLv3	RSA	RSA	None	MD5	
ADH-DES-CBC3-SHA	SSLv3	DH	None	3DES (168 bit)	SHA1	
ADH-DES-CBC-SHA	SSLv3	DH	None	DES (56 bit)	SHA1	
ADH-RC4-MD5	SSLv3	DH	None	ARCFOUR (128 bit)	MD5	
EDH-RSA-DES-CBC3-SHA	SSLv3	DH	RSA	3DES (168 bit)	SHA1	
EDH-DSS-DES-CBC3-SHA	SSLv3	DH	DSS	3DES (168 bit)	SHA1	
EDH-RSA-DES-CBC-SHA	SSLv3	DH	RSA	DES (56 bit)	SHA1	
EDH-DSS-DES-CBC-SHA	SSLv3	DH	DSS	DES (56 bit)	SHA1	
EXP-EDH-RSA-DES-CBC-SHA	SSLv3	DH (512 bit)	RSA	DES (40 bit)	SHA1	export
EXP-EDH-DSS-DES-CBC-SHA	SSLv3	DH (512 bit)	DSS	DES (40 bit)	SHA1	export
EXP-ADH-DES-CBC-SHA	SSLv3	DH (512 bit)	None	DES (40 bit)	SHA1	export
EXP-ADH-RC4-MD5	SSLv3	DH (512 bit)	None	ARCFOUR (40 bit)	MD5	export

In TABLE C-2, DH refers to Diffie-Hellman and DSS refers to the Digital Signature Standard.

TABLE C-3 lists and describes the aliases that provide macro-like groupings.

TABLE C-3 SSL Aliases

Alias	Description
SSLv2	all SSL version 2.0 ciphers
SSLv3	all SSL version 3.0 ciphers
EXP	all export-grade ciphers
EXPORT40	all 40-bit export ciphers
EXPORT56	all 56-bit export ciphers
LOW	lower strength ciphers (DES, 40-bit RC4)
MEDIUM	all 128-bit ciphers
HIGH	all ciphers using Triple DES
RSA	all ciphers using RSA key exchange
DH	all ciphers using Diffie-Hellman key exchange
EDH	all ciphers using Ephemeral Diffie-Hellman key exchange

TABLE C-3 SSL Aliases

Alias	Description
ADH	all ciphers using anonymous Diffie-Hellman key exchange
DSS	all ciphers using DSS authentication
NULL	all ciphers using no encryption

The preference of ciphers can be configured using the special characters listed and described in TABLE C-4.

TABLE C-4 Special Characters to Configure Cipher Preference

Character	Description
<none>	add cipher to list
!	remove a cipher from the list entirely -- it cannot be added again
+	add cipher to list, and pull to current location (possibly demoting it)
-	remove cipher from list (can be added later in list)

The default value of *cipher-spec* is

```
SSLCipherSuite ALL:!ADH:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP
```

The default configures all ciphers except anonymous (unauthenticated) Diffie-Hellman, giving preference to ARCFOUR and RSA, and then higher grades of encryption over the lower grades.

5. `SSLCertificateFile` *file*

   Context: global, virtual host

   This directive specifies the location of the PEM-encoded X.509 certificate file for this server.

6. `SSLCertificateKeyFile` *file*

   Context: global, virtual host

   This directive specifies the location of the PEM-encoded private key file for this server, corresponding to the certificate configured with the `SSLCertificateFile` directive.

7. `SSLCertificateChainFile` *file*

   Context: global, virtual host

   This directive specifies the location of a file containing the PEM-encoded certificates making up the certification path of the server. It may be used to assist clients in verifying the server's certificate when the server's certificate is not directly signed by an authority that the client recognizes.

Certificates in the chain are assumed to be valid for client authentication as well, when client authentication (`SSLVerifyClient`) is used.

8. `SSLCACertificateFile` *file*

   Context: global, virtual host

   This directive specifies the location of a file containing the concatenation of the certificates for certification authorities (CAs) used for client authentication.

9. `SSLCARevocationFile` *file*

   Context: global, virtual host

   This directive specifies the location of a file containing the concatenation of the certificate revocation lists of CAs used for client authentication.

10. `SSLVerifyClient` *level*

    Context: global, virtual host, directory, `.htaccess`

    This directive configures the authentication of clients to the server. (Note that this is not normally needed for eCommerce applications, but has use in other applications.)

    Values for *level* are listed and described in TABLE C-5.

TABLE C-5 SSL Verify Client Levels

Level	Description
`none`	no client certificate is required
`optional`	the client may present a valid certificate
`require`	the client *must* present a valid certificate
`optional_no_ca`	the client may present a certificate, but it need not be valid

Typically either `none` or `require` will be used. The default is `none`.

11. `SSLVerifyDepth` *depth*

    Context: global, virtual host, directory, `.htaccess`

    This directive specifies the maximum certificate chain depth that the server will allow for client certificates. A value of 0 means that only self-signed certificates are eligible, whereas a value of 1 means that client certificates must be signed by a CA known directly to the server (via the `SSLCACertificateFile`). Larger values permit delegation of the CA.

12. `SSLLog` *filename*

    Context: global, virtual host

This directive specifies a log file where SSL-specific information will be logged. If not specified (default), then no SSL-specific information will be logged.

13. `SSLLogLevel` *level*

Context: global, virtual host

This directive specifies the verbosity of the information logged in the SSL log file. Values for *level* are listed and described in TABLE C-6.

TABLE C-6 SSL Log Level Values

Value	Description
`none`	no logging, but error messages are still sent to the standard Apache error log
`warn`	include warning messages
`info`	include information messages
`trace`	include trace messages
`debug`	include debugging messages

14. `SSLOptions` [`+-`] *option*

Context: global, virtual host, directory, `.htaccess`

This directive configures SSL specific options. Options can be added to the current configuration by prefixing them with a plus sign (+), or removed using a minus sign (-). If no plus or minus sign is present, then the most closely scoped set of options is used.

Options are listed and described in TABLE C-7.

TABLE C-7 Available SSL Options

Options	Description
`StdEnvVars`	standard set of SSL related CGI/SSI environment variables are created—there is a performance penalty for this.
`ExportCertData`	causes the `SSL_SERVER_CERT`, `SSL_CLIENT_CERT` and `SSL_CLIENT_CERT_CHAIN`*n* (*n* = 0, 1, ...) environment variables to be exported. These variables contain PEM-encoded certificates for the client and server.
`FakeBasicAuth`	the Distinguished Name (DN) of the client certificate is translated into an HTTP Basic Authentication Username, and is "faked" to have authentication. This allows the use of standard Apache access control mechanisms with SSL client authentication without prompting the user for a password. Entries for these users in the Apache password files must use the encrypted password `xxj31ZMTZzkVA`, which is just an encrypted form (`crypt`(3c)) of the word "password."
`StrictRequire`	forces a forbidden access due to `SSLRequireSSL` to be denied, even in the presence of other directives, such as `Satisfy Any`, which might override this.

15. SSLRequireSSL

Context: directory, `.htaccess`

This directive forbids access in a given directory unless HTTPS is used. It can be used to guard against misconfigurations that might otherwise leave a directory's contents available to unauthenticated and unencrypted accesses.

APPENDIX **D**

Building Applications for Use With Sun Crypto Accelerator 1000 Board

This appendix discusses the software supplied with the Sun Crypto Accelerator 1000, which can be used to build some OpenSSL-compatible applications to take advantage of the cryptographic acceleration features of the Sun Crypto Accelerator.

Note – This information on building applications to use the Sun Crypto Accelerator 1000 software and hardware is provided strictly as-is, and is not an officially supported part of this product. This information is provided in the hope it may be useful, but without any warranty. If you require a Sun-supported solution, please contact Sun Professional Services to learn about your options.

You must first install the `SUNWcrysl` package, which contains the required header files and libraries.

Your application must be configured to include OpenSSL headers from `/opt/SUNWconn/crypto/include`, such as with the compiler flag:

```
-I /opt/SUNWconn/crypto/include
```

Additionally, the linker must be directed to include references to the appropriate libraries. Most OpenSSL-compatible applications will reference either or both of the `libcrypto.a` and `libssl.a` libraries. The Sun cryptographic libraries must be included as well. The following linker flags will accomplish this:

```
-L/opt/SUNWconn/crypto/lib -R/opt/SUNWconn/crypto/lib \
-lcrypto -lssl -lcryptography -lnvpair
```

Note that not all OpenSSL applications will benefit from being compiled in this fashion (as opposed to being built with the stock OpenSSL library, which can be downloaded from `www.openssl.org`).

APPENDIX **E**

Sun Crypto Accelerator 1000 Board Specifications

This chapter outlines the various specifications of the Sun Crypto Accelerator 1000 Board.

This appendix includes the following sections:

- "Physical Dimensions" on page 81
- "Interface Specifications" on page 82
- "Power Requirements" on page 82
- "Environmental Specifications" on page 83

Physical Dimensions

TABLE E-1 Physical Dimensions

Dimension	Measurement	Metric Measurement
Length	6.875 inches	174.625 mm
Width	4.2 inches	106.680 mm

Interface Specifications

TABLE E-2 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

Power Requirements

TABLE E-3 Power Requirements

Specification	Measurement
Maximum power consumption	10W @ 5V 700mW @ 3.3V
Voltage tolerance	5V +/- 5% 3.3V +/- 5%
Operational current	2A @ 1.8V 150mA @ 3.3V

Environmental Specifications

TABLE E-4 Environmental Specifications

Condition	Operating Specification	Storage Specification
Temperature	0° to 70°C, 32° to 160°F	-65°C to +150°C, -85° to 300° F
Relative humidity	5 to 85% non-condensing	0 to 95% non-condensing

APPENDIX **F**

Third-Party Licenses

Some portions of Software are provided with notices and/or licenses from other parties which govern the use of those portions.

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

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.

Index

A

administering iPlanet web servers, 53
algorithms, 3
Apache SSL directives, 71

D

dcatest, 48
  command-line syntax, 50
  parameter options, 49
  sub-tests, 49
diagnostics tests, 47
directories
  hierarchy of, 11
Dynamic Reconfiguration, 3

E

enabling
  Apache web servers, 39
  iPlanet web servers, 15

F

files and directories, 10

H

High Availability, 3
hot-plug, 3

K

key length, 41

L

load sharing, 4

P

passwords
  list required for iPlanet web servers, 15
patches
  recommended, 5
  required, 5

R

realms, 53
  creating, 62
  deleting, 65
  listing, 64
  setting, 63
requirements
  hardware, 4
  software, 4
RSA keypair, 40

S

secadm, 56
server certificate, 23, 33
slot files, 54
software packages, 10
statistic values, 52
SunVTS, 47

U

URL
  for iPlanet software, 19, 29
  for openssl, 80
user password
  changing, 66
users, 53
  creating, 65
  deleting, 68
  enabling or disabling, 67
  listing, 66