SKIP Installation and User's Guide Source Release 1.0

sun microsystems

Table of Contents

Introducing SKIP 1
SKIP Services 1
Sun ICG SKIP Product Family 6
SKIP security services 6
Installing SKIP 7
Hardware and Software Requirements 7
General Installation Procedures 8
Configuring SKIP 12
Implementing SKIP Services 12
Enabling SKIP 28
Verify the SKIP Installation and Set Up 31
Troubleshooting 32
Viewing SKIP Statistics (skiptool and skipstat) 33
Managing Keys and Certificates 44
Installing Certificates 47

1

Introducing SKIP

SKIP (Simple Key Management for Internet Protocols) is an IP layer encryption package. It provides a system with the ability to efficiently encrypt any protocol within the TCP/IP protocol suite. Once installed, any two (or more) systems running SKIP have the ability to transparently encrypt and/or authenticate all traffic between them.

1.1 SKIP Services

SKIP provides four network security services:

- Access Control for protecting your corporate data resources from unauthorized use.
- Encryption and decryption services to ensure the confidentiality of information sent over the network.
- Authentication to ensure the integrity of the information transferred from one group to another within your corporation.
- Key and certificate management which provides for efficient, cost-effective administration of the basic building blocks of your security policy.

1.1.1 Access Control

Use access control on your network to limit and control who uses your host systems and applications through your communications links. Each entity-- host, network, or nomadic system, you communicate over your network with must be identified and/or authenticated so that access to your system is controlled. Once communication is established, data may be exchanged in the clear or encrypted.

SKIP's access control is based on the requesting system's IP address. To provide access for mobile remote users, SKIP has provided users with the ability to separate an entity from it's physical address through the use of a key identifier. When a system tries to connect to a host running SKIP, the order of processing is as follows:

1. Search for an entry specifying a remote host. If the entry exists use it, otherwise continue to the next search argument.

2. Search for a network entry which matches the remote entry. If the entry exists use it, otherwise continue to the next search action.

3. If an IP address entry is not found, search for a nomadic ACL entry containing the sender key identifier present in the SKIP protocol header. If the entry is found and the packet is authenticated, store the sender's IP address for future reference.

4. If a corresponding ACL entry is still not found, use the default system entry. This entry is configured by the user to either allow or deny access to unknown hosts.

Note: These rules may also be used to exclude a host or network from accessing the system.

Tunneling and Transport modes

Each IP packet can be encrypted/authenticated using SKIP in two ways:

- only the data part of the IP packet is encrypted. This is called the “transport mode”.
- the whole the IP packet is encrypted. This is called the “tunnel mode”

Topology Hiding

SKIP supports topology hiding through the use of a **Tunnel address** . The **Tunnel address** field contains the IP address of the host which serves as the intermediary between any or all hosts/systems on a network whose topography is to remain hidden from the rest of the world. The remote system needs to be configured using the “tunnel” mode.

1.1.2 Encryption/Decryption (Confidentiality)

The SKIP kernel does traffic encryption and decryption. It views the key manager as a process that it gives encrypted traffic keys to and gets back decrypted traffic keys.

SKIP provides users with the ability to separate the identity of an entity from it’s physical address. This means that each person (sender/receiver) participating in a transfer of encrypted data over a computer network can be identified by an NSID/Local KeyID pair. The KeyID is used for key lookup, not for routing. By default, the NSID is set to 0 and a KeyID is not sent. However, with the KeyID feature activated, key names are no longer tied to IP addresses. This means that regardless of their physical location on the network or on the Internet, sales, marketing, and support personnel have the ability to communicate with each other and corporate using encryption. Corporate data remains secured regardless of the location of a sales rep or system engineer at customer sites or trade shows.

The Name Space Identifiers (NSID). The NSID's supported by SKIP are:

- NSID 0 (Not present)
- NSID 1 (IPv4 address)
- NSID 8 (MD5 of Diffie-Hellman Public Values)

- **The traffic encryption algorithm**. Traffic is encrypted using conventional symmetric key cryptography. A random traffic key is used as a key to encrypt data. The algorithms supported at your site are automatically installed by SKIP and appear in skiptool.

1.1.3 *Authentication (Integrity)*

Authentication is the process of verifying that the user requesting access is who they say they are. In SKIP, authentication is implemented through the Keyed MD5 algorithm and applies to the whole IP packet.

1.1.4 *Key and Certificate Management*

The SKIP key management system is based on public key cryptography, that is, each participant holds a pair of keys: his/her public key and a private key. To ensure that the public keys are authentic, that is, they have not been tampered with by an attacker and do indeed belong to the claimant, the public key is signed by a Certification Authority (CA). The result, a Certificate, is freely passed around the network. Its authenticity can be verified by anyone holding the CA's signature information, that is, its public key.

Keys and certificates are handled by the key manager. Local key information is managed using the skiplocal command and CA information is managed using the skipca command.

The algorithms used by SKIP are:

- **The long term secret key algorithm**. The Diffie-Hellman Key Agreement algorithm is used.

- **The key encryption algorithm.**The low order bits of the Diffie-Hellman secret key agreement function as the key. Keys are encrypted using conventional symmetric key cryptography.

As stated earlier, certificates are the digital documents which testify to the binding of a public key to an individual or other entity for the purpose of preventing someone else from impersonating you. In order for two hosts running a security package to communicate, they must exchange certificates. Common methods of certificate exchange are:

1. Certificate Discovery Protocol - Hosts running SKIP request each other's certificates through a clear channel. A host may also ask a Directory Server for a certificate.

2. Certificate Authority (CA). This procedure is manual in that certificate and possibly the key are provided by the certifying agency on a physical media: tape, diskette or CD-ROM. They must be loaded into the system by the user through a command line.

3. Manually generate, add, and exchange certificates by using a command line interface provided by a vendor.

SKIP supports all of the common methods of certificate exchange. For more information on configuring certificate fetching protocols and certificate management, please see the manual pages for skipd, skipdb, skiplocal and skipca. By default, the key manager asks the host its trying to communicate with for it's certificate.

There are times when it is useful to allow a system to have more than one public/private key pair. For example, different key sizes may be required when communicating with subsidiaries in other countries due to local regulations. To meet these user requirement's, Sun's SKIP implementation allows a system to possess as many local keys as required. Similarly, the SKIP system can also be configured with the details of several Certification Authorities so that certificates signed by different CAs can be checked for authenticity.

1.2 Sun ICG SKIP Product Family

SKIP is available as an unsupported free product for SunOS 4.1.3 and FreeBSD. Sun Microsystems Internet Commerce Group also sells the Sunscreen(tm) line of products which include *Sunscreen SKIP for Solaris* (sparc and x86) and several firewall solutions. All SKIP products from ICG including the Free Source interoperate. For more information about Sunscreen(tm) products, visit our web page:

http://www.incog.com.

For more information on SKIP, please see the SKIP web page:

http://skip.incog.com

1.3 SKIP security services

SKIP implements security services through these four major components:

- Bulk Data Crypt for key caching, bulk data encryption, and authentication.
- Cryptographic Modules which support the most rigorous symmetric key cryptography and authentication methods currently available.
- Key and Certificate Management tools which provide automatic management of certificates and the generation of random traffic encryption keys.
- SKIP End System for the system administrator's use in controlling access to corporate resources on the network.

Together, these components provide authentication and privacy at the network layer without the need to modify your applications. This offers the system administrator a method of selectively controlling access to corporate data--encrypting and authenticating network traffic as required.

2

Installing SKIP

This section provides instructions for installing SKIP on Solaris 1.x, FreeBSD 2.1.0 and 2.1.5 systems. Installation on FreeBSD is identical to installation of Solaris 1.x. Once SKIP is installed, configured, and enabled on the systems requiring its services, IP layer encryption begins. SKIP runs without further administration effort until new systems need to be added or certificate management is required.

2.1 Hardware and Software Requirements

2.1.1 Supported Platforms

The SKIP source reference is supported on the following platforms:

- Sun SPARCstation 1, 1+, 2, Classic, LX, 5, 10 or 20 running Solaris 1.1.1 (SunOS 4.1.3 U1).
- Any x86 personal computer running FreeBSD 2.1.0 and 2.1.5.

2.1.2 Hardware Requirements

- A minimum of 6 MB free disk space is required for installation. 3 MB of disk space is permanently used.

2.2 General Installation Procedures

The exact steps in installing SKIP are determined by the media used for the installation, the hardware platform, and the operating system in use on the system which is to use SKIP. However, the general installation steps are:

1. Unpack the software.
2. Install the software.
3. Obtain a certificate from the Certificate Issuer or generate an unsigned key.
4. Install the keys.
5. Delete the temporary files.
6. Update environment variables (PATH and MANPATH)
7. Install SKIP on the network interface.
8. Re-boot your system.

The installation process is described in detail on the following pages.

2.2.1 Solaris 1.x and FreeBSD 2.1.0 and 2.1.5 Installation

This section provides instructions for installing SKIP on Solaris 1.x systems (SunOS 4.1.3 u1) and FreeBSD. In order to install and run the software, you must be able to become root on your local system.Get the software from http://skip.incog.com.
Unpack the Software

The software must be unpacked before it can be installed. To extract the files, complete the following steps:

1. Become root on your local system.
2. Change to the `/usr` directory. Verify that approximately 7 Mbytes of free space exist in the `/usr` directory.

3. Type the following command, exactly as it is written below, substituting the name of the directory where you put skip.tar.Z for *directoryname*:

```
example# zcat directoryname/skip.tar.z|tar -xvf -
```

The files are extracted into /usr/skip and /etc/skip.

Install the Software

After the files have been de-compressed, SKIP is ready to be installed. Use the procedure below to install the software:

1. Open a terminal window and become root.
2. Change to the `/usr/skip` directory:

```
example# cd /usr/skip
```

3. Use the `install.skip` command to install SKIP:

```
example# ./install.skip
```

4. The program asks you a number of questions. For most of these questions, the default answer can be used. You may have to rebuild a new kernel for the support of loadble modules
5. If the install program needs to reboot your system, then complete the following steps:
   a. Log in once more as root.
   b. Switch to the directory where you stored the certificate for the system.
6. Add ***/usr/skip/bin*** to your path by typing:

```
example# set path = ( /usr/skip/bin $path )
```

Generate and Install Unsigned Keys

When generating an unsigned key, no authority exists to certify identities. This means that each party in the communication must verify the name of the key (the -R field in the skiphost command or the remote ID in the skiptool menu), over the phone or another trusted channel. Otherwise, a third party may impersonate the system/person you wish to reach with impunity. Without verification through a secure channel, you have no way of knowing if the key belongs to the correct party or not. If you wish to generate a secret and unsigned dhpublic certificate (NSID 8) locally, complete the following steps:

1. Generate your key and install it as a local key id:

```
# skiplocal keygen
```

Communicate your keyid to the party you wish to talk to. Use an out of band communication like the phone. The *skiplocal list* command will list your local keys. Get the other parties keyid and enter it into the Add System menu of skiptool. A shortcut command called *skiplocal export* exists which will display what your system thinks the other side should use to add you to their Access Control List. You can mail the output of this command to the party you wish to communicate with and they can do the same. Each of you can cut and paste the other's line into the shell to add the other to the Access Control list. Since your system does not know what local keyids and network interfaces the other system has, this command will only work when both parties have one network interface and one key.

Caution - Even when using *skiplocal export*, make sure you both verify the other party's keyid over the phone with the other party to make sure someone is not impersonating them.

Install SKIP on your network interface

The skipif (see skipif(1)) command is used to install SKIP on a network interface. After you reboot the system, SKIP will have been installed on this interface. If you want to add SKIP to a machine with one network interface you can just use the skipif command:

```
example# skipif -a
```

On a machine with multiple network interfaces, you should specify the interface :

```
example# skipif -i zp0 -a
```

Finishing Up

After the software installation is completed, a few file management and clean-up procedures should take place:

1. Add ***/usr/skip/bin*** to the path in your ***.login*** file.
2. Add the man pages to the ***.login*** file by entering:

   setenv MANPATH /usr/share/man:/usr/skip/man

3. Still as root, reboot your system by typing:

   ```
   example# reboot
   ```

After the system has rebooted, login as root.

3

Configuring SKIP

3.1 Implementing SKIP Services

3.1.1 Key and Certificate Management Command Line Tools

Setting Up Trusted CAs: skipca

As stated earlier, certificates are the digital documents which testify to the binding of a public key to an individual or other entity for the purpose of preventing someone else from impersonating you. In order for two hosts running a security package to communicate, they must exchange certificates. The skipca command line interface is used to designate a Certificate Authority as trusted and manage that database. skipca options: add, list, delete, create, and revoke certificates.

X509 Certificates without proper signatures are not be added to the database. Therefore, the CAs certificate must be added to the CA Certificate database using the *skipca* command prior to adding certificates signed by that CA to the database.

Managing SKIP Local Identities: skiplocal

There are times when it is useful to allow a system to have more than one public/private key pair. For example, different key sizes may be required when communicating with subsidiaries in other countries due to local regulations. To meet these user requirement’s, Sun's SKIP implementation allows a system to possess as many local keys as required. Similarly, the SKIP system can also be configured with the details of several Certification Authorities so that certificates signed by different CAs can be checked for authenticity. *skiplocal* is the command line interface utilitized when managing multiple local identities for a system.

Managing Certificates: skipdb and skipd

skipdb and *skipd* are used to manage certificates. Long term certificates are stored in a database for access by the key manager. The *skipdb* command line interface allows the manual administration of the certificate database.

skipd services encryption and decryption requests from the kernel and acts as a certificate server. The encryption/decryption requests are received and answered via /dev/skip_key. When acting as a certificate server, *skipd* answers CDP requests from remote hosts as well as issuing requests for certificates from remote Certificate Discovery servers when necessary.

Activating the Changes: skipd_restart

In order for the changes made by *skipca*, `skipdb`, and `skiplocal` to take effect, *skipd_restart* must be run to reinitialize the key manager. To run *skipd_restart*, login as root and enter:

```
# skipd_restart
```

3.1.2 Access Control

SKIP provides two interfaces for setting up your Access Control List: skiptool and skiphost. The easiest way to set up your access control list is through the Graphical User Interface (GUI), *skiptool*. If you prefer command line tools, refer to the *skiphost* command described in the next section.

Configuring skiptool

skiptool is the graphical user interface which allows you to enable and disable access to your machine, set the type of encryption used for hosts or network connections to your system (encrypted or clear) as well as determine how to deal with unauthorized hosts which try to connect to your system. To use SKIP, you must be root on your system. In addition, for Solaris 2.x systems, access to the X server must be enabled for any client by entering the `xhost +` command. To configure SKIP, complete the following steps:

1. Login as root and enter your password.
2. Start SKIP.
3. Add authorized systems:
   - Communicating in the clear
   - Communicating using SKIP
   - Communicating with systems using SKIP V1
   - Communicating using AH/ESP.
4. Add the excluded systems.
5. Set up the rules for unauthorized systems.
6. Enable SKIP (Access control button is enabled).
7. Iconify the SKIP window.
8. Verify the installation and set up.
9. Save thecurrent configuration

Each step is described in further detail on the following pages.

Starting skiptool

To run skiptool, you must be able to become root on your system. When skiptool is started for the first time, the following defaults are in effect:

- Access control is disabled.
- Unauthorized systems are set at *No access*.
- A default system entry resides on the Authorized Systems list.

Leave these options as they are initially set for now. When your access control list is complete, these default settings are modified.

To start SKIP, complete the following steps:

1. Open a window and become root.

As root, type **`skiptool&`**. If you are configuring a system which has multiple network interfaces, you may specify the interface following the skiptool command. For example, **`skiptool zp0.`** The main menu may be seen in figure 3-1.

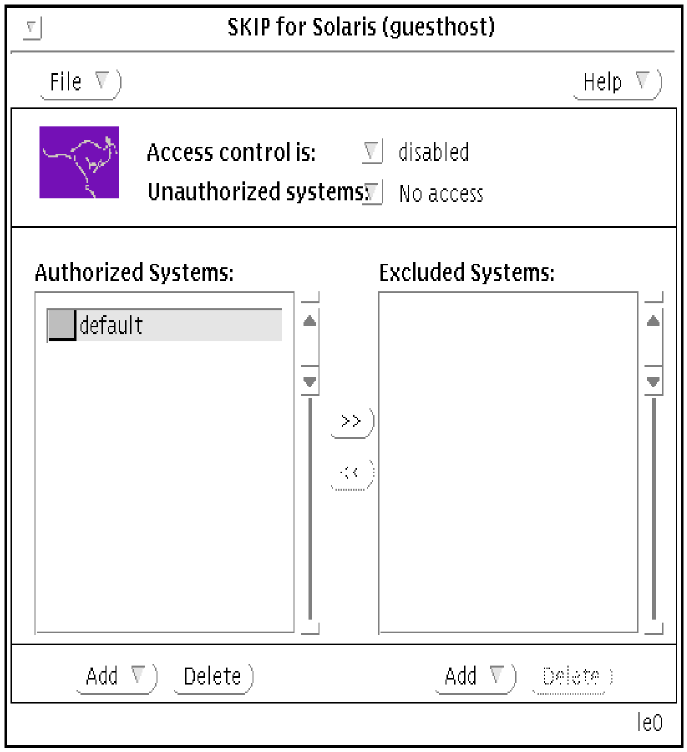

Figure 3-1 skiptool Main Menu

skiptool Main Menu

The components available from the skiptool Main Menu are:

Component	Submenu	Definition
File Menu	The file menu has three submenus:	
	Load	Loads the access control list from the kernel.
	Key Management	Defines the parameters for key usage, including when to delete an unused key (in seconds), and how much to transmit per key (in Kbytes).
	SKIP Statistics	Brings up one of six Statistics windows: Interface Stats, Header Stats, Encryption Stats (SKIP V1 or IPSP), Key Stats, or Authentication.
	Save	Saves the configuration to a file. Prompts for addition of any systems which have access and are not currently on the authorized list before doing a save operation. The next time a reboot occurs, this configuration is used.
	Exit	Closes all SKIP open windows and quits the application.

Component	Submenu	Definition
Access Control Button	None	This button toggles whether SKIP is enabled or disabled. When enabled, only those systems listed in the Authorized Systems List can connect. When disabled, any system can connect.
Unauthorized System Policy Button	This button is used to set the policy regarding Unauthorized Systems. *Note that if a default authorized host entry exists, this policy does not take effect.* The policy can be:	
	No access	Don't allow unauthorized systems to connect.
	Ask For Confirmation	Every time an unauthorized host connects, a pop up window appears on which the user determines whether or not that particular connection should be allowed.
	Add Automatically	Add any host to the Authorized System List which sends packets to this system.

Component	Submenu	Definition
Authorized Systems List	*None*	A list of systems authorized to access this host. System types are: host, network or nomadic. Secure systems are denoted by the lock or the Sun Microsystems logo next to the system name, depending on the type of security they are using.
Excluded Systems List	*None*	A list of systems which are specifically denied access to your system.
Management Buttons	These buttons allow you to add or delete a system from the access list. The buttons are available for both Authorized and Excluded systems.	
	Add	Brings up the Add popup menu where the system type to be added to the access control list is selected: • Host • Network • Nomadic.
	Delete	Deletes the selected system from the list. A delete operation occurs immediately and cannot be undone.

Adding Authorized Systems

Any remote host that you want to communicate with (send/receive data) must be configured using the *Add* popup window. An authorized host may or may not be using encryption. The *Add* popup window provides four options: not using encryption, using SKIP encryption, using SKIP V1 compatible encryption, and using ESP/AH (manual keying).

Hosts are added to the Authorized System list using the **Add** button, located at the bottom left of the Authorized Systems list on the skiptool Main Window. When setting up SKIP, be sure to include any NFS servers and NIS name servers on the Authorized Systems list, otherwise your system hangs. To determine the servers your system communicates with, use the following commands:

NFS servers	**mount**
NIS servers	**ypwhich**

It could be usefull also to verify the current routing entries used by the local system with "netstat -rn" and add specific network ACL entries.

When adding entities to your access control list, valid types are:

- Host
- Network
- Nomadic

Regardless of the system type you are adding to the access control list, the same policy must be implemented on both your machine and the entity you wish to communicate securely with over the intra or internetworks. If both systems are not configured properly, the packets are silently dropped, appearing as if that particular host does not exist.

When you click on the **Add** button, the *Add* popup window appears. Here you select the Type of connection: Host, Network, or Nomadic. Next, set the Security level. After any level of security has been selected, the *Properties* window becomes available. The *Add system Properties* window is used to set up the options for the type of encryption used by the host, network, or nomadic system being authorized.   On the *Add system Properties* window, the following button is not implemented at this time: Compression.

To authorize a system, complete the following steps:

1. Click on the **Add** button at the bottom of the Authorized Systems list on the skiptool Main Window.

2. Select the type of connection being authorized: **Host**, **Network,** or **Nomadic.**

3. Set the type of **Security** to be implemented for this system**.** This is a four-position button:
   - To add hosts without encryption, select OFF.
   - If the remote host system also has SKIP and the traffic between your systems is to be encrypted, select SKIP.
   - For systems using the Sun Microsystem's SunScreen(tm) SPF-100 product, select SKIP V1
   - If ESP/AH (manual keying) is to be used, click on ESP/AH.

4. On the *Add properties* window, enter the name or IP address of the host system to be added to your access control list.

5. If this system is not using any form of encryption, you are finished with the authorization process. Click the **Apply** button. If you are authorizing a host system which uses encryption, continue with the rest of the procedure.

After the type of **Security** has been selected, set the appropriate encryption fields for the entity being authorized.

6. Determine whether **Whole packets ("transport mode")** or **Data only ("tunnel mode")** is secure by clicking on the appropriate selection for the **Secure** button.

7. Each type of encryption requires that certain options be set. The parameters selected are determined by the system type being authorized and your security policies. The options to be considered based upon the encryption method selected are:

   - **For systems using SKIP: Tunnel address, Remote Key ID**, **Local Key ID**
   - For SKIP V1: Node ID, Tunnel Address
   - For ESP/AH systems: Tunnel address, Local SPI, Remote SPI

   Refer to the appropriate subsection for a complete discussion of these options.

8. Select the appropriate algorithms for **Key encryption**, **Traffic encryption,** and **Authentication** buttons. The options available for each system are based upon the method of encryption selected from the Security pop-up menu:

   - **Key Encryption Button**. *Selecting this button lists the available session key encryption algorithms. The algorithms available are determined by the system type and the selected encryption method.*
   - **Traffic Encryption Button.** Select the algorithm for encrypting the traffic between your system and the remote system. The algorithms available are determined by the system type and the initial encryption method selected in Step 5.
   - **Authentication Button.**

     MD5

     None
   - **Compression Button**. Not available at this time.

9. Click **Apply** to add the host to the Authorized systems list.

Repeat steps 1 though 9 for all encrypted hosts. Remember that your policy options for each system entered on your ACL must be the same as those entered on the system entity you wish to communicate with through encrypted channels. If the configuration on your system does not match that of the party you wish to communicate with, the packets are silently dropped. It will simply appear as though that host no longer exists. Be sure to verify your installation after enabling SKIP.

3.1.3 Communicating in the Clear (OFF)

Typically, the NIS and DNS servers your system accesses are set up as communicating with your system in the clear. In addition, any host that does use an encryption package must be set up to communicate with you in the clear.

3.1.4 Communicating Using SKIP

Any host that you want to send encrypted SKIP traffic to must be configured using the *Add* popup window. When SKIP **Security** is selected, the **Tunnel address**, **Remote Key ID, Local Key ID, Key encryption, Traffic encryption** and **Authentication** fields must be set up from the Properties window.   In addition, if the **Remote/Local Key ID** fields are set to other than *Not Prese*nt, the **ID** field must also be configured. Complete the following procedure to set these fields for encrypted traffic between your server and the system to be authorized:

1. After selecting the type of system and setting the security to SKIP, enter the **Hostname**.

2. Set the **Secure** button to either **Whole packet** or **Data only**. It is recommended that the whole packet be secured (encrypted).

3. Set the **Tunnel address**: if topology hiding is in use. Tunnel addressing is generally used for encrypted gateways where the IP address of the host entered here serves as the

intermediary for any or all hosts on a network whose topography is to remain unknown or hidden from the rest of the world.

4. Use the **Remote Key ID** button to select whether you would like the remote system's keyid included in SKIP packets and, if so, what namespace that keyid lives in. By selecting Not Present, the sending of the receiver keyid is disabled.

Not Present is the default. It uses the IP address of the remote system to identify its certificate. If a remote system has a Node ID other than that identified by its IP address, set the namespace and indicate the remote system's keyid in the **ID Field**. The namespace indicated in the Remote Key ID field is determined by the type of certificate used/obtained for this system:

Certificate Type	Remote Key ID field
O assurance certificate	Not present
CA (Sun or other)	IPv4
Self-generated unsigned key	MD-5 DH Public Value

The following namespaces are listed in this menu:

Not present	IPv4 Address	MD5 (DH pub.Value)	

5. If the **Remote Key ID** field has been set to other than *Not Present*, enter the Key ID in hexadecimal format in the **ID**: field (0x0a000000). It must contain the appropriate Node ID for the system being authorized based upon the selection made in the **Remote Key ID** field. Depending on the type of certificate, this information may be obtained from the master key id on the diskette or from the local key id field of the other host.

6. Use the **Local Key ID** button to indicate whether you want your local system to send it's keyid in the SKIP packet and, if so, what namespace that key lives in. If you select *Not Present*, a sender Key ID is not sent in the packet. All the local key times installed for this host are listed. Select the namespace for the local key to be used for communication with the above host. Once the namespace is selected, click on the **ID** field to select the key to be used, in hexadecimal, for communicate with this host.

For a more complete discussion about keys and namespaces, refer to the *advanced.TOPICS* document in `/usr/skip/doc`.

7. Now select the key encryption, traffic encryption, and authentication algorithms you wish to use for communication with the remote system.

3.1.5 Communicating with Sunscreen(TM) Products

Sun Microsystems has created a security product family called Sunscreen(TM). All products in the Sunscreen(TM) family can securly communicate with this package using SKIP protocols.

To communicate with a Sunscreen(TM) SPF-100 you must use the SKIP V1 Protocol and install the proper key. Call SunService for help in obtaining the key. Appendix A of this manual describes how to install the key.

When accessing a SunScreen, the Node ID, Tunnel address, Key encryption, and Traffic encryption fields must be set up.

1. After selecting the type of system and setting the security to SunScreen Compatibility, enter the **Hostname**.

2. Enter the **Node ID** for this host.

3. The **Local key ID** field is completed automatically for you by SKIP.

4. Set the **Tunnel address**: if topology hiding is in use. Tunnel addressing is generally used for encrypted gateways where the IP address of the host entered here serves as the

intermediary for any or all hosts on a network whose topography is to remain unknown or hidden from the rest of the world.

5. Select the appropriate key and traffic algorithms for the **Key** and **Traffic encryption** buttons. Available encryption methods are:

3.1.6 Communicating Using AH/ESP

Manual keying is typically used in test mode only. It is not recommended for day to day operations. To configure a host with which you are using manual keying, both skiptool and the raw_keys file must be configured.

3.1.7 Adding Excluded Systems

If the default entry remains on the **Authorized systems** list, than any remote host that you want to prevent communication with must be configured using the *Add button* located under the **Excluded Systems:** list. An excluded host may or may not be using encryption. When setting up an excluded system, it is only necessary to enter the Hostname. If the state of the host or network changes to an authorized system, you must delete the system from the Excluded list and add it to the Authorized systems list. To exclude a system, complete the following steps:

1. Click on the **Add** button at the bottom of the Excluded Systems list on the skiptool Main Window.

2. Select the system type: Host, Network or Nomadic.

3. In the **Hostname** field on the *Exclude system* window, enter the name or IP address of the host system you want to deny access to your system.

4. Click **Apply** on the Exclude system window.

3.1.8 Rules for Authorized Systems

Once you have entered the authorized systems and the excluded systems, you need to determine what should happen when unidentified systems attempt to access your system. An unidentified system is unknown by SKIP, that is, it is not on either the **Authorized System**s list or the **Excluded Systems** list.

Use the **Unauthorized Systems** button on the main window to select what action SKIP should take when an unidentified system attempts access. There are three possible actions which SKIP may take when an unidentified system attempts to connect to your system when the default entry has been removed from these lists:

- No access
- Ask for confirmation
- Add automatically

It is recommended that this entry not be modified from the default selection of No access.

Note - If a Default authorized system resides on either the **Authorized** or **Excluded Systems** lists, this option does not take affect.

Once SKIP has been configured on your system, you are ready to configure it on the other systems which you will be communicating with either in the clear or through one of the encryption methods available in SKIP. Once both parties have installed and configured SKIP, SKIP should be enabled and your data protected.

3.1.9 Using skiphost

skiphost is the command line interface used to list, add, and delete entities from the access control list as well as enable SKIP. It's functionality is the same as the skiptool GUI. In addition, skiphost is used to remove SKIP from a network interface if required.

To configure SKIP using the *skiphost* command, one entry is required for each system/host being set up in the access control list. Before enabling SKIP, any hosts needed for operation of the local system must be present in the ACL. Verify that any NFS file servers, NIS servers, or any local broadcast addresses for your network are on the ACL. Then *skiphost* must be run one final time to enable SKIP. See the man page for *skiphost* for details

3.2 Enabling SKIP

The last step in setting up SKIP is to enable access control for the system. Enable SKIP by selecting *enabled* from the **Access Control Button** on the main window. When SKIP is enabled for the first time, it checks for all systems that you are talking with in the clear. It detects the NIS and DNS servers you are communicating with and automatically adds their addresses to the access control list when you select *Add* from the **Required systems** window. Choosing *Cancel* may hang your system or prevent your access to the system/network the next time you try to login.

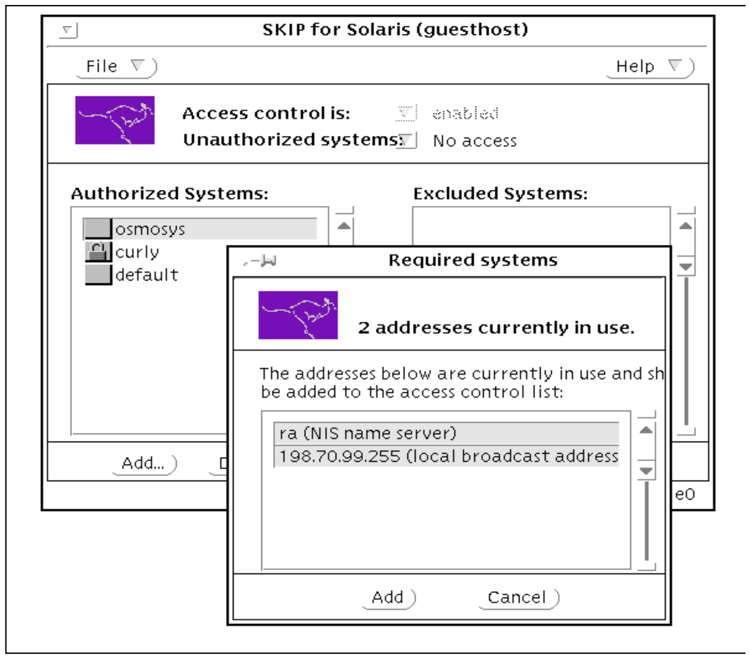

Figure 3-2 Enabling SKIP

This windows now adds also the “ALL-SYSTEMS” and “ALL-ROUTERS” multicast addresses.

The **Authorized Systems** area lists all the hosts allowed access. The **Excluded Systems** area shows all those known hosts which are explicitly denied access.The graphic preceding the host name or IP address depicts what type of security is being used with that host.

- A blank box preceding the host name indicates no encryption (Security = *off*).
- A box with a lock in it indicates that the system is using SKIP as the encryption method (Security = *SKIP*).
- A box with the Sun Microsystems logo in it indicates that the system is using SKIP V1 Compatibility mode (Security = *SKIP v1*).

- A box with   indicates that the system is using manual keying (Security = *ESP/AH*).
- A box with an N indicates a system that is Nomadic, that is, it is identified by its Key Id not its IP address, and it's using either SKIP or SKIP V1 Compatibility mode as the encryption method.

3.2.1 Default System Entry

The default system entry is used when no other more specific ACL entry matches a host. Often, this entry is set to clear traffic to allow hosts which aren't in listed in the ACL to communicate in the clear. It may, however, be used to create a default encryption rule, as well.

Note that if the default system entry remains, it is unnecessary to add any entity with the OFF security option. Further, if the default system entry remains, the option set by the Unauthorized Systems button never goes into effect.

3.2.2 Iconify the SKIP Application

Once SKIP is enabled, it is no longer necessary to keep the window open. At this time, you may wish to iconify the main window. The skiptool icon shows SKIP's status.

If you quit the application, SKIP stays in whatever mode it was last in (enabled or disabled). The **Unauthorized Systems** policy automatically changes to *No Access*, since there is no longer any way to notify you if an unauthorized system is attempting to gain access.

Figure 3-3 SKIP Icon showing both the Enabled and Disabled states

SKIP: Enabled

SKIP: Disabled

3.3 Verify the SKIP Installation and Set Up

Once you have configured and enabled SKIP, it is time to determine that it is working properly. If the configurations on the systems do not match, that is the encryption algorithms used, it will appear as if the other part of the communication equation does not exist. SKIP silently drops the packets.

To verify that SKIP is operating properly on your system, complete one or more of the following procedures:

1. Ping the remote system. The remote system must have SKIP enabled, and be using the same key and traffic encryption algorithms as your system.

   If you have the remote site's certificate, you immediately start sending Encrypted IP. Otherwise, your keymanager will need to do a remote certificate fetch. By default, this is done by asking the remote site for it's certificate over a clear channel. If you have configured other hosts to act as key servers, they will be asked for the certificate. See the man pages for *skipd* and *skipd.conf* for details. If there are no problems at the remote site, you receive replies to your pings.

2. Run snoop or tcpdump on your local system or a sniffer to see that packets are being encrypted.

3.4 Troubleshooting

If encryption is not taking place between your system and a system on your **Authorize**d **Systems** list or you cannot connect to that system, check the following items:

- Is SKIP enabled? Check the Access Control button. Set it to enabled.
- Verify that a certificate exists for each system you wish to communicate with on your Authorized systems list. Use the skipdb command to check for the certificate of the remote system by dumping the database to stdout. Try to restart the key manager by using the skipd_restart command .
- Verify that SKIP is installed, configured, enabled, and has the certificate on the remote system.
- Verify the KeyID of the remote system in the log file */var/log/skipd.log* to see if the keymanager has set the keyID to what you think it should be. If it is not the correct keyID, get certificates for the correct keyIDd.
- Verify that both machines have the same key encryption, traffic encryption and authentication algorithms.
- If you are commucating through a firewall, check with your system administrator that ports for UDP 1639 & 1640 are are allowed to pass. These ports required for the Certificate Discovery protocol. As a workaround, you may manually distribute keys. Also, make sure that the SKIP protocol (57) is permitted to pass through the firewall.
- Verify that the cdp server specified in skipd.conf is correct and has been authorized in skiptool. If the cdp_server entry is = or @, it is specifying the tunnel address or host address, respectively.
- SKIP requires machine clocks to be syncronized within one hour. Make sure they are in sync. Messages in /var/log/skipd.log will indicate this situation.

- If the *skiplocal export* command has been used to communicate keyids when one or both of the systems have multiple keys or multiple network interfaces, the keyid may have been bound to the wrong network interface or local keyid. Use skiptool or skiphost to add the remote host after verifying keyids over the phone.
- Use *skiplog* to verify configuration mismatches

3.5 Viewing SKIP Statistics (skiptool and skipstat)

SKIP provides two methods of viewing statistics: skiptool and skipstat. skiptool is the GUI format you have already been introduced to in the previous section and skipstat is the command line interface for viewing SKIP statistics. The method chosen is a matter of personal preference since both interfaces provide the same data. SKIP provides the following statistics:

- Network Interface
- Header Statistics
- Key Statistics
- Encryption Statistics
- Authentication Stats

3.5.1 *Using skiptool*

You can view the Network Interface, Header, Encryption (SKIP V1/IPSP), Authentication, and Key statistics in real-time by selecting **SKIP Statistics** from the **File** menu on the skiptool Main Window. The fields on the statistics screens are updated approximately every 3 seconds. A status change is indicated by the word *UPDATED* next to the fieldname. Figure 3-4 shows the menu path to a Statistics window.

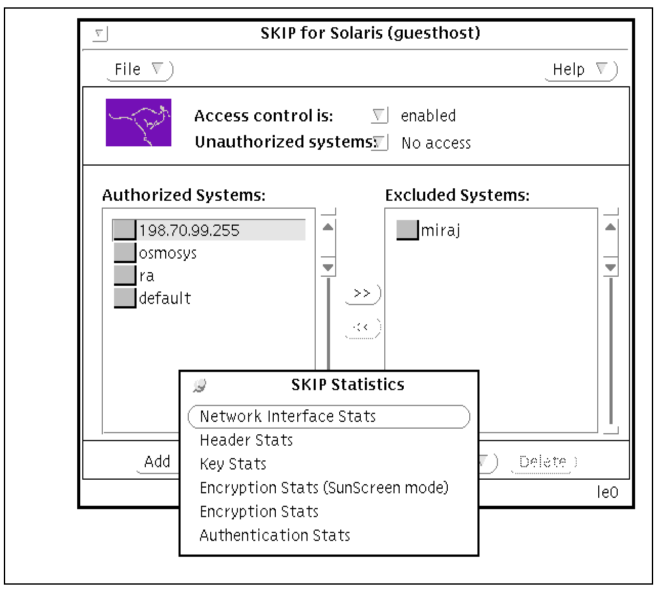

Figure 3-4 Bringing up a Statistics Window

Each of the statistics available for SKIP is described on the following pages. Sample data with field descriptions illustrate the information available for monitoring SKIP performance.

SKIP Network Interface Statistics

Selecting **File - SKIP Statistics-Network Interface Stats** displays the SKIP Network Interface Statistics window.

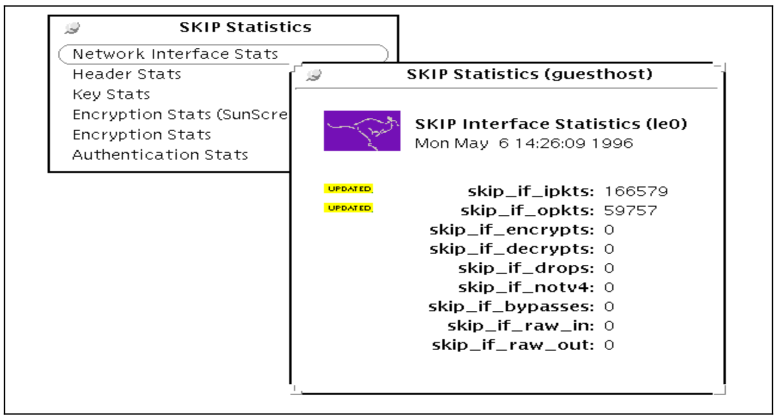

Figure 3-5 SKIP Interface Statistics Window

The fields on the SKIP Interface Statistics window show the number of:

skip_if_ipkts	Packets received by the interface.
skip_if_opkts	Packets sent by the interface.
skip_if_encrypts	Packets encrypted.
skip_if_decrypts	Packets decrypted.
skip_if_drops	Packets dropped.
skip_if_notv4	Packets that are not IPv4 packets.
skip_if_bypasses	The number of certificate packets.
skip_if_raw_in	Raw AH and ESP packets received by the interface.
skip_if_raw_out	Raw AH and ESP packets sent by the interface.

SKIP Header Statistics

Selecting **File - SKIP Statistics-Header Stats** displays the SKIP Header Statistics window.In the field descriptions below, V1 refers to SKIP v1 compatibility mode.

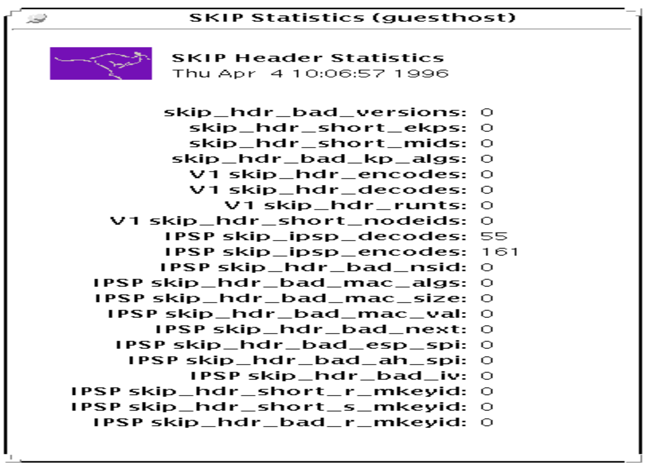

Figure 3-6 SKIP Header Statistics Window

The fields on the SKIP Header Statistics window are:

skip_hdr_bad_versions	The number of headers with invalid protocol versions.
skip_hdr_short_ekps	The number of headers with short eKp fields.
skip_hdr_short_mids	The number of headers with short MID fields.
skip_hdr_bad_kp_algs	The number of headers with unknown cryptographic algorithms.

V1 skip_hdr_encodes	The number of SKIP V1 headers encoded.
V1 skip_hdr_decodes	The number of SKIP V1 headers decoded.
V1 skip_hdr_runts	The number of headers with short SKIP V1 packets.
V1 skip_hdr_short_nodeids	The number of headers with short SKIP V1 node ids.
IPSP skip_ipsp_decodes	The number of SKIP V2 headers decoded.
IPSP skip_ipsp_encodes	The number of SKIP V2 headers encoded.
IPSP skip_hdr_bad_nsid	The number of headers with a bad V2 name space id.
IPSP skip_hdr_bad_mac_algs	The number of headers with unknown or bad authentication algorithms.
IPSP skip_hdr_bad_mac_size	The number of headers with an authentication error in the
IPSP skip_hdr_bad_mac_val	The number of headers with an authentication error in the
IPSP skip_hdr_bad_next	The number of headers with a bad V2 next protocol field.
IPSP skip_hdr_bad_esp_spi	The number of headers with a bad V2 SPI field.
IPSP skip_hdr_bad_ah_spi_	
IPSP skip_hdr_bad_iv	The number of headers with a bad V2 initialization vector.

IPSP skip_hdr_short_r_mkeyid	The number of headers with a short V2 receiver key id.
IPSP skip_hdr_short_s_mkeyid	The number of headers with a short V2 sender key id.
IPSP skip_hdr_bad_r_mkeyid	The number of headers with a bad V2 receiver key id.

SKIP Key Statistics

Selecting **File - SKIP Statistics-Key Stats** displays the SKIP Key Statistics window (Figure 3-7).

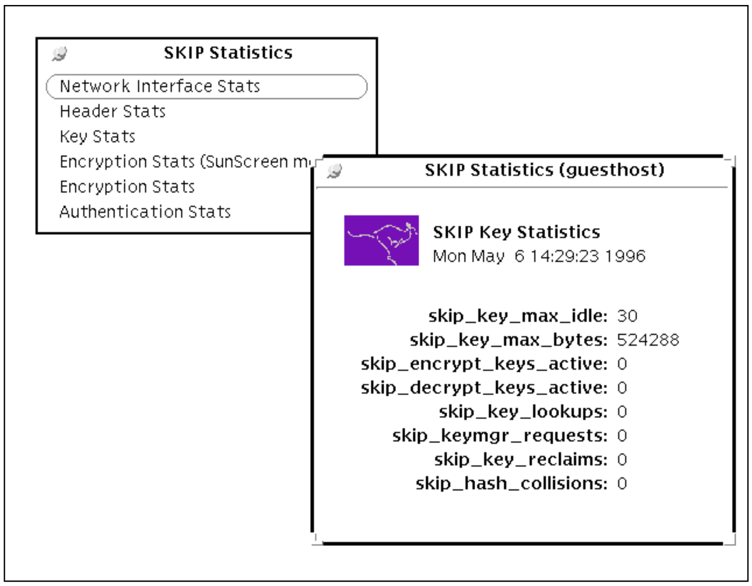

Figure 3-7 SKIP Key Statistics

The fields on the SKIP Key Statistics window are:

skip_key_max_idle	The time, in seconds until an unused key is reclaimed.
skip_key_max_bytes	Maximum number of bytes to encrypt before discarding a key.
skip_encrypt_keys_active	Number of encryption keys in the cache.

skip_decrypt_keys_active	Number of decryption keys in the cache.
skip_key_lookups	The total number of key cache lookups.
skip_keymgr_requests	The total number of key cache misses (key not found).
skip_key_reclaims	The total number of key entries reclaimed.
skip_hash_collisions	The total number of table collisions.

SKIP Encryption Statistics

Selecting **File - SKIP Statistics-Encryption Stats (SKIP V1 or IPSP)** displays the SKIP Algorithm Statistics window.

One set of statistics is displayed for each different traffic and key encryption module. The fields are:

Crypto Module Name	The name of the cryptographic module for which the statistics are being displayed.
encrypts	Number of successful encryptions.
encrypterrs	Number of failed encryptions.
decrypts	Number of successful decryptions.
decrypterrs	Number of failed decryptions.

SKIP Authentication Stats

Selecting **File - SKIP Statistics-Authentication Stats** displays the SKIP Authentication Stats window which provides information on MAC's (Message Authentication Code).

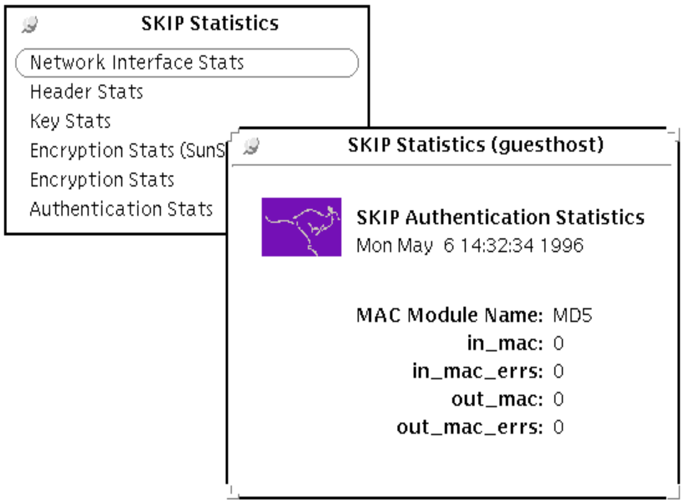

Figure 3-8 Authentication Stats

The fields on the SKIP Authentication Stats window are:

MAC_Module_Name	MAC method used for authentication.
in_mac	Number of received MAC calculations that succeeded.
in_mac_errs	Number of received MAC calculations that failed.
out_mac	Number of sent MAC calculations that succeeded.
out_mac_errs	Number of sent MAC calculations that failed.

3.5.2 *Using skipstat*

The alternative method of viewing SKIP statistics is to use the *skipstat* command line interface. *skipstat* is a part of the *skiptool* GUI as well. skipstat provides statistical data on the following items for the local system:

- Cryptographic algorithms supported on the local system
- Statistics for the cryptographic algorithms
- MAC algorithms
- Interface
- Key
- Header
- Statistics for the specified network interface

3.6 *Managing Keys and Certificates*

Secure methods of key management are a necessity for users. Users must be able to easily obtain and use key pairs with the knowledge that these pairs have not been comprised. In order to meet these requirements SKIP provides both a GUI and a command line interface. skiptool provides the GUI Key Management parameters window and the command line interface provides print_cert, skipdb, and certreq.

3.6.1 Key Management with skiptool

The *Key Management Parameters* window is displayed by selecting **Key Management** from the **File** pulldown menu. Key management parameters are global, that is, one set of key management parameters governs the activity of all keys on a particular system. They determine when a key is deleted based upon use and the maximum number of bytes transmitted. The *Key Management Parameters* window has four major components, as shown in Figure 3-9.

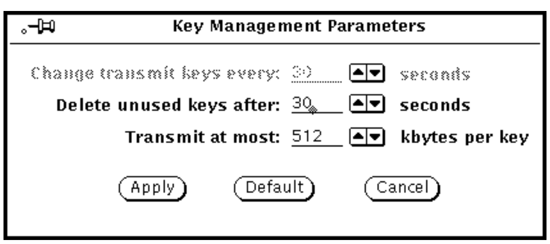

Figure 3-9 Key Management Parameters Window

The Key Management Parameters window components are:

Change transmit keys every:	This button sets how often the transmit key should be changed. It is not currently implemented.
Delete unused keys after	Set the number of seconds an unused key is kept before it is deleted. The number may be changed by either typing in a new number, or clicking on the up and down arrows until the desired number is reached. Default value = 30 seconds. Valid range: 5 seconds to 10,000 seconds.
Transmit at most	Set the maximum amount of information that can be transmitted using a particular key. When the set amount is reached, the key is changed. The number may be changed by either typing in a new number, or clicking on the up and down arrows until the desired number is reached. Default value=512 Kbytes per key. Valid range: 1 Kbyte per key to 10,000 Kbytes per key.
Management Buttons	These three buttons allow you to apply the new values, return to the default values, or dismiss the window without changes. • Apply - Makes the changes made in the window active. • Default - Returns the values in the window to the default values (30 seconds and 512 Kbytes). • Cancel - Dismisses the window without changing anything.

3.6.2 *Key and Certificate Management Using the Command Line Interface*

When issuing any of the commands used in key and certificate management, you must be logged in as root. In addition, if you are running Solaris 2.x, you must issue the *xhost* + command to allow access to the X server. The key and management commands provided by SKIP are:

Command	Description
skipdb	*Administer* the SKIP database of certificates. SKIP uses the database to store long term certificates so that they may be accessed by the key manager.
print_cert	Print a user readable certificate.
certreq	Request and retrieve a certificate from a keyserver or other host.
skipca	Add, delete or list the CA.
skiplocal	Add, delete or print local keys.

X509 Certificates without proper signatures are not be added to the database. Therefore, the CAs certificate must be added to the CA Certificate database using the *skipca* command prior to adding certificates signed by that CA to the database.

Further information to be supplied.

AG.

Installing Certificates

The following procedure provides instructions for installing certificates from Sun Microsystems Internet Commerce Group's Certificate Authority.

A.1 Installing an ICG Certificate

Remember that this is a DOS diskette using DOS filenaming conventions. To load your certificate, complete the following steps:

1. Insert the diskette and type *volcheck*.
2. Install the certificate and keys by typing:

   ```
   example# install_skip_keys -icg /floppy/floppy0
   ```

   This script unpacks the certificate file, verifies the MD-5 checksums, and installs the certificates.
3. Eject the floppy using the eject command:

   ```
   example#  eject floppy
   ```
4. Store the floppy in a secure location.
5. Reboot or use *skipd_restart* to activate the keys.