SunScreen SKIP User's Guide, Release 1.1

Sun Microsystems, Inc.
901 N. San Antonio Road
Palo Alto, CA 94303-4900
U.S.A.

Part No: 805-5743b
June 18 1998

Contents

Preface

Welcome to Sunscreen™ SKIP. The purpose of this guide is to provide you with the information that you need to be able to set up and manage *SunScreen SKIP* on your system.

Who Should Use This Guide

This guide is written for people familiar with Solaris™ Versions 2.4, 2.5, and 2.5.1 or Solaris for the Intel Platform who wish to run IP-level encryption on their system.

Before You Read This Guide

This guide assumes that you are familiar with TCP/IP, networking, and public-key and shared-key cryptography.

How This Guide Is Organized

The *SunScreen SKIP* User's Guide is divided into the following chapters:

Chapter 1, "Installing SunScreen SKIP," describes how to install the *SunScreen SKIP* software from the CD-ROM onto your Solaris Versions 2.4, 2.5, or 2.5.1 or Solaris for

the Intel platform system. This chapter also describes how to protect your locally stored secrets with a passphrase.

Chapter 2, “Installing Keys and Certificates,” details how to create and install keys and certificates on your system. If you installed Unsigned Diffie-Hellman Key during installation, you may skip this chapter.

Chapter 3, “Managing SunScreen SKIP Through skiptool,” describes how to use the skiptool graphical user interface (GUI) to monitor the network, how to configure SKIP, how to enable SKIP, how to verify SKIP installation and setup, how to view statistics, and how to manage keys.

Chapter 4, “Managing SunScreen SKIP Through the Command-Line Interface,” describes how to use the command-line interface as superuser or root.

Chapter 5, “Usage Examples,” describes examples of the usage of *SunScreen SKIP* in several network configurations.

Appendix A, “Quick-Start Guide,” covers installing the SKIP binaries or adding the packages with pkgadd, and setting up IP-level encryption between two hosts.

Appendix B, “SunScreen SKIP Theory of Operations,” is an overview of what SKIP provides to users and how *SunScreen SKIP* fits in with other security products that use SKIP.

Appendix C, “Glossary,” covers those terms that are specific or unique to Sun and the SunScreen line of products.

What Is New in This Release

SunScreen SKIP, Release 1.1, is the upgrade for SKIP for Solaris, Release 1.0. The following is a list of the new features for *SunScreen SKIP*, Release 1.1.

1. The random number generator has been changed so that using this line `rng_dev_audio 1` in the `skipd.conf:` file will cause the random number generator to use `/dev/audio` for enhanced entropy collection. This is the default.
2. Local identities can now be protected with a passphrase; that is, `/etc/opt/SUNWicg/skip/localid/0.secret, 1.secret` through `<n>.secret` are DES encrypted).

   You can protect with a passphrase, change the passphrase, or remove (delete) the passphrase:

   ```
   skiplocal passwd, skiplocal rmpasswd
   ```

   If you protect your local identities with a passphrase, these commands will prompt for `passwd` when invoked:

`skiplocal keygen, skiplocal add.` The daemon `skipd` also requires the passphrase.

When rebooting the system, if passphrase protection is used, no encrypted connections can be supported until the key manager, `skipd`, is reinitialized with the `skipd_restart` command, which will prompt for the passphrase.

3. Support for tunnel addresses has been added to `skiphost –a` (add and SCL entry) by means of the parameter `–A`, which takes the tunnel address as its argument.

   In the `skipd.conf` file, the line `cdp_server =` has been added, which means by default the host specified as the tunnel address will be asked for the certificate.
4. `skiphost` no longer supports plumb and unplumb (`–p, -u`) as options.
5. `print_cert` and man page are now available. This command will print contents of a certificate found in the certificate file specified
6. `skipif` with the arguments– `-l -v` now lists Access Control Lists on an interface
7. `skipdb` and `skiplocal` now use the keyword `udh` in preference to `dhpublic` when referring to Unsigned Diffie-Hellman certificates.
8. `skipdb`, `skiplocal`, and `skipca` now use the keyword `rm` in preference to `del` when removing items from their respective databases.

What Has Been Fixed

All of the outstanding problems from SKIP for Solaris, Release 1.0 and Release 1.03, have been fixed.

Related Books and Publications

It may be helpful to refer to the following books when installing the *SunScreen SKIP*:

- Applied Cryptography Bruce Schneier John Wiley & Sons, 1994, ISBN 0-471-59756-2
- Building Internet Firewalls D. Brent Chapman and Elizabeth D. Zwicky O'Reilly &Associates, 1995, ISBN 1-56592-124-0
- Firewalls and Internet Security Bill Cheswick and Steve Bellovin Addison-Wesley, 1994, ISBN 0-201-63357-4
- Handbook of Computer-Communications Standards Volume 3: The TCP/IP Protocol Suite William Stallings, Macmillan, 1990

- Internetworking with TCP/IP, 2nd Edition Douglas E. Comer, Prentice Hall, 1995, ISBN 0-13-216987-8
- Network and Internetwork Security Principles and Practice William Stallings, Prentice Hall, 1995, ISBN 0-02-415483-0
- Practical UNIX Security Simson Garfinkel and Gene Spafford O'Reilly & Associates, 1991
- TCP/IP Illustrated, Volume 1 The Protocols W. Richard Stevens Addison-Wesley, 1994, ISBN 0-201-63346-9
- TCP/IP Network Administration Craig Hunt O'Reilly & Associates, 1992

What Typographic Changes and Symbols Mean

The following table describes the type changes and symbols used in this book.

Typeface or Symbol	Meaning	Example
AaBbCc123	The names of application or program groups, book titles, new words or terms, or words to be emphasized	Open the SunScreen SPF-100 program group.Select the Configure application.Read Chapter 6 in *User's Guide*. These are called *class* options.
AaBbCc123	The name of a menu item, button, or key.	Select Exit from the File pull-down menu.Press the F1 key for help.Click on the Done button.

Keys, Certificates, and Algorithms

Upgrade packages for U.S. Domestic and U.S. Export keys, certificates, and algorithms from SunCA (Sun Microsystems' Certificate Authority) are intended to be used with *SunScreen SKIP*, Release 1.1, as well as with *SKIP for Solaris*, Release 1.0.

U.S. customers and companies and some foreign customers and companies may order additional keys, certificates, and algorithms in stronger encryption strengths.

To place an order with ICG please follow the directions below.

1. **Complete a Purchase Order for the product.**
   Please include the following information:
   - Ship-to address
   - Bill-to address
   - Contact Name
   - Telephone
   - Product Name
   - Part Number
   - Quantity
   - Purchase Order Number

2. **Fax your Purchase Order to 415-336-0074.**
   You will receive confirmation when your order ships with an airbill number.

3. **If you cannot fax your Purchase Order, please send it to the following address:**
   Internet Commerce Group Sun Microsystems, Inc. Mail Stop PAL-01-550 2550 Garcia Avenue Mountain View, CA 94043-1100
   Telephone Numbers:
   1-800-820-9995 (U.S. Customers)
   415-336-0018 (Foreign Customers)
   415-336-0074 (fax)

CHAPTER 1

Installing SunScreen SKIP

An Overview of *SunScreen SKIP*

SunScreen SKIP is Sun Microsystems' implementation of Simple Key-Management for Internet Protocols (SKIP).

SunScreen SKIP is replacement software and upgrade software for any previous version of *SKIP for Solaris*.

This chapter provides instructions for installing *SunScreen SKIP* on Solaris, Versions 2.4, 2.5, or 2.5.1 and *Solaris* for the Intel Platform. Once *SunScreen SKIP* is installed, configured, and enabled on the systems requiring its services, IP-layer encryption can begin. *SunScreen SKIP* runs without further administration effort until new systems need to be added or certificate management is required. This chapter also describes how you can protect your locally stored secrets with a password.

Hardware and Software Requirements

Supported Platforms

SunScreen SKIP is supported on the following platforms:

- Any Sun SPARC workstation running Solaris, Versions 2.4, 2.5, or 2.5.1.
- Any Intel-based PC that is compatible with and running Solaris for the Intel Platform, Versions 2.4 or 2.5.

Hardware Requirements

The hardware requirements are as follows:

- A minimum of 16–MB of RAM is required, 32–MB of RAM is recommended.
- A minimum of 6–MB of free disk space is required for installation, 3–MB of disk space is permanently used.
- One or more supported network interfaces.
- A CD-ROM drive.
- A floppy drive, if planning to install SunCA certificates.

Operating System Requirements

To run *SunScreen SKIP*, you must

1. **Install the Solaris SunCore™ software group.**

   This software group contains the minimum software required to boot and run the Solaris operating system. It includes some networking software and the drivers necessary to run the OpenWindows environment; it does not include the OpenWindows software.

2. **Additionally, install the following packages**:

system	SUNWadmr	System & Network Administration Root
system	SUNWcar	Core Architecture, (Root)
system	SUNWcsd	Core Solaris Devices
system	SUNWcsr	Core Solaris, (Root)
system	SUNWcsu	Core Solaris, (Usr)
system	SUNWdfb	Dumb Frame Buffer Device Drivers
system	SUNWesu	Extended System Utilities
system	SUNWkvm	Core Architecture, (Kvm)
system	SUNWlibC	SPARCompilers Bundled libC
system	SUNWlibms	SPARCompilers Bundled shared libm
system	SUNWtoo	Programming Tools

system	SUNWvolr	Volume Management, (Root)
system	SUNWvolu	Volume Management, (Usr)

3. **If you plan to use the `skiptool` GUI, install the packages for OpenWindows.**

Protocol Compatibility

SunScreen SKIP supports the following protocol versions:

- SKIP, Version 1, for SunScreen SPF-100/100G compatibility.
- Any platform that has implemented SKIP as described in the ICG Technical Reports listed in Section 1.1.2, including the SunScreen product line, except SunScreen SPF-100, which only implements SKIP, Version 1 (see above).
- Raw mode (also known as ESP/AH, manual keying, or S/WAN) for compliance with RFC 1825: Security Architecture for the Internet Protocol.
- *SunScreen SKIP*, Release 1.1, is the upgrade for SKIP for Solaris, Release 1.0.

Installation Procedure

Before installing *SunScreen SKIP*, Release 1.1, be sure that you have the CD-ROM for the base software and any encryption upgrade CD-ROMs or diskettes to which you are entitled.

For the new user, this chapter tells about

1. **Installing *SunScreen SKIP*. (“Installing the Software” on page 8)**
2. **Generating and installing an Unsigned Diffie-Hellman (UDH) key pair, if you are using UDH. (“Installing SKIP Unsigned Diffie-Hellman (UDH) Certificates” on page 11)**
3. **Installing *SunScreen SKIP* on your network interface. (“Installing Your Network Interface” on page 14)**
4. **Rebooting your system. (“Rebooting Your System” on page 15)**
5. **Protecting your locally stored secrets with a passphrase. (“Activating Your Passphrase” on page 16)**

For the user who is upgrading from any version of *SKIP for Solaris* to this release, this chapter tells about

1. **Upgrading to *SunScreen SKIP*. (“Upgrading From Earlier Versions of *SKIP for Solaris*” on page 7)**
   - Removing any old version of *SKIP for Solaris*
   - Preserving or removing previous configurations
   - Installing *SunScreen SKIP*
2. **Generating and installing an Unsigned Diffie-Hellman (UDH) key pair. (“Installing SKIP Unsigned Diffie-Hellman (UDH) Certificates” on page 11)**
3. **Installing *SunScreen SKIP* on your network interface. (“Installing Your Network Interface” on page 14)**
4. **Rebooting your system. (“Rebooting Your System” on page 15)**
5. **Protecting your locally stored secrets with a passphrase. (“Activating Your Passphrase” on page 16)**

Installing the Software for the First Time

This section provides instructions for installing *SunScreen SKIP* on Solaris for SPARC Platforms, Versions 2.4, 2.5, or 2.5.1 and Solaris for the Intel Platform.

To install and run the software, you must be able to become root on your local system and know the IP address of the machine on which SKIP is to be installed. Ask your systems administrator for the IP address of your machine. To install the software for the first time or if you are installing it without saving the configurations, follow these steps:

1. **Open a terminal window and become root.**
2. **Mount the CD-ROM through the file manager by typing**

```
volcheck
```

Note - If you are not using `vold` on your system, type

```
# mount -F hsfs -oro /dev/dsk/c0t6d0s0 /mnt
```

The device name or the mount point or both depends on your local system configuration.

---

3. **Go to the directory on the CD-ROM for your OS. (The examples assume a machine with only one CD-ROM.)**

   Solaris for the SPARC Platform:

```
cd /cdrom/cdrom0/sparc
```

   Solaris for the Intel Platform:

```
cd /cdrom/cdrom0/x86
```

   Note - If you have mounted the CD-ROM manually, replace `/cdrom/cdrom0` with `/mnt`.

4. **Type the standard Solaris operating system** `pkgadd` **command to add all packages**:

```
pkgadd  -d ‘pwd‘
```

5. **You will be prompted with the following menu of packages to install.**

```
  1 SICGbdcdr SKIP Bulk Data Crypt 1.1-FCS Software
   (sparc) 1.1-FCS
  2 SICGcrc2 SKIP RC2 Crypto Module 1.1-FCS Software
   (sparc) 1.1-FCS
  3 SICGcrc4  SKIP RC4 Crypto Module 1.1-FCS Software
   (sparc) 1.1-FCS
  4 SICGes SKIP End System 1.1-FCS Software
   (sparc) 1.1-FCS
  5 SICGkeymg SKIP Key Manager Tools 1.1-FCS Software
   (sparc) 1.1-FCS
  6 SICGkisup SKIP I-Support module 1.1-FCS Software
   (sparc) 1.1-FCS
  Select package(s) you wish to process (or ’all’ to process all
packages). (default: all) [?,??,q]:
```

   Select `a` (all). As the prompts appear, answer questions with `Y` (yes) followed with a `<Return>` if you wish to add the package.

6. **When you get back to the same menu of packages, type** `q` **followed by a** `<Return>` **to quit** `pkgadd`**.**

7. **To eject the CD-ROM from the CD-ROM drive, type**

```
cd / eject cdrom0
```

or eject the CD-ROM from the CD-ROM drive through the file manager.

Note - If you are not using `vold` on your system, unmount your CD-ROM by typing

```
# cd /
# umount/mnt
# eject cdrom0
```

---

8. **To add** `/opt/SUNWicg/bin` **to your PATH variable in the Bourne shell, type**

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

9. **To add** `/opt/SUNWicg/man` **to your MANPATH variable in the Bourne shell, type**

```
MANPATH=/opt/SUNWicg/man:$MANPATH
export MANPATH
```

10. **It will be helpful to add** `/opt/SUNWicg/bin` **to the PATH variable in your initialization file (such as:** `.profile`**,** `.cshrc`**, or** `.login` **file), and** `/opt/SUNWicg/man` **to the MANPATH variable in the same file.**

    Now you are ready to generate and install SKIP Unsigned Diffie-Hellman (UDH) certificates (*Section* “Installing SKIP Unsigned Diffie-Hellman (UDH) Certificates” on page 11) or to install SunCA certificates (*Chapter 2*) and to install *SunScreen SKIP* on your network interface (*Section* “Installing Your Network Interface” on

page 14). After you have completed these two procedures, you must reboot your system (*Section* “Rebooting Your System” on page 15).

You may use SKIP Unsigned Diffie-Hellman certificates and SunCA keys and certificates at the same time on *SunScreen SKIP*.

Upgrading From Earlier Versions of *SKIP for Solaris*

▼ Removing the Earlier Versions of the Software

To remove any version of *SKIP for Solaris*, become root and use the `pkginfo` and `pkgrm` packages shown in the following steps.

1. **Type**

```
pkginfo | grep SICG
```

to list the SKIP packages that were installed:

```
1 SICGbdcdr SKIP Bulk Data Crypt 1.0.3-FCS Software
 (sparc) 1.0.3-FCS
2 SICGcrc2 SKIP RC2 Crypto Module 1.0.3-FCS Software
 (sparc) 1.0.3-FCS
3 SICGcrc4 SKIP RC4 Crypto Module 1.0.3-FCS Software
 (sparc) 1.0.3-FCS
4 SICGes SKIP End System 1.0.3-FCS Software
 (sparc) 1.0.3-FCS
5 SICGkeymg SKIP Key Manager Tools 1.0.3-FCS Software
 (sparc) 1.0.3-FCS
6 SICGkisup SKIP I-Support module 1.0.3-FCS Software
 (sparc) 1.0.3-FCS
```

2. **Type**

```
pkgrm SIGbdcdr SICGcrc2 SICGcrc4 SICGes SICGkeymg SICGisup
```

and answer **Y** (yes) to questions that the `pkgrm` program asks. The `pkgrm` program ends with the statement:

```
Removal of <SICGkisup> was successful.
```

Note - This is valid only for this example. If moduli of other sizes were used, then the last package remove would be different.

3. **To remove the "`/etc/opt/SUNWicg/skip`" directory and any configurations that were installed, type**

```
rm -rf /etc/opt/SUNWicg/skip
```

Caution - If you want to preserve previous configurations (access control list [ACL] files, certificates, and the key manager configuration file), do not remove the `/etc/opt/SUNWicg/skip` directory.

4. **To reboot the machine, type**

```
init 6
```

▼ Installing the Software

Become root on your local system and then follow these steps:

1. **Open a terminal window and become root.**
2. **Mount the CD-ROM through the file manager or by typing**

```
volcheck
```

Note - If you are not using `vold` on your system, type

`# mount -F hsfs -oro /dev/dsk/c0t6d0s0/mnt`

The device name or the mount point or both depends on your local system configuration.

3. **Go to the directory on the CD-ROM for your OS:**

   Solaris for the SPARC Platform:

```
cd /cdrom/cdrom0/sparc
```

   Solaris for the Intel Platform:

```
cd /cdrom/cdrom0/x86
```

Note - If you have mounted the CD-ROM manually, replace `/cdrom/cdrom0` with `/mnt`.

---

4. **To use the standard Solaris operating system** `pkgadd` **command to add all packages, type**

```
pkgadd  -d ‘pwd‘
```

5. **You will be prompted with the following menu of packages to install.**

```
  1 SICGbdcdr SKIP Bulk Data Crypt 1.1-FCS Software
   (sparc) 1.1-FCS
  2 SICGcrc2 SKIP RC2 Crypto Module 1.1-FCS Software
   (sparc) 1.1-FCS
  3 SICGcrc4  SKIP RC4 Crypto Module 1.1-FCS Software
   (sparc) 1.1-FCS
  4 SICGes SKIP End System 1.1-FCS Software
   (sparc) 1.1-FCS
  5 SICGkeymg SKIP Key Manager Tools 1.1-FCS Software
   (sparc) 1.1-FCS
  6 SICGkisup SKIP I-Support module 1.1-FCS Software
   (sparc) 1.1-FCS
Select package(s) you wish to process (or ’all’ to process all
packages). (default: all) [?,??,q]:
```

   Select a (all) or the number of the package. As the prompts appear, answer questions with Y (yes) followed with a <Return>, if you wish to add the package.

   When you get back to the same menu of packages, type q followed by a <Return> to quit `pkgadd`.

6. **When you get back to the same menu of packages, type q to quit.**

7. **To eject the CD-ROM from the CD-ROM drive, type**

```
cd /
eject cdrom0
eject cdrom0
```

or eject the CD-ROM through the file manager.

Note - If you are not using `vold` on your system, unmount your CD-ROM by typing

```
# cd /
# umount/mnt
# eject cdrom0
```

Now you are ready to generate and install SKIP Unsigned Diffie-Hellman (UDH) certificates if you are going to use SKIP UDH certificates.

You may use SKIP UDH certificates and SunCA keys and certificates at the same time on *SunScreen SKIP*.

You are also ready to install *SunScreen SKIP* on any new or different network interface, if you need to. Generate and install the SKIP UDH certificates (*Section* "Installing SKIP Unsigned Diffie-Hellman (UDH) Certificates" on page 11) and install *SunScreen SKIP* on the network interface (*Section* "Installing Your Network Interface" on page 14) before you reboot your system.

Note - If you are going to use the same keys and certificates and network interface that you used in SKIP for Solaris, Release 1.0, you only need to reboot your system according to the instructions in "Rebooting Your System" on page 15. This is only true if you did not remove the `/etc/opt/SUNWicg/skip` directory.

Installing SKIP Unsigned Diffie-Hellman (UDH) Certificates

Once the *SunScreen SKIP* software has been installed, you must install at least one local identity (public-private key pair) for this host.

The procedure below creates a SKIP UDH certificate, which is the one you will most likely use. For a more detailed discussion of SKIP UDH certificates, see Appendix C.

Chapter 2 discusses keys, certificates, and hashes in greater detail. If you are installing other kinds of keys and certificates, see the documentation that is supplied with them or contact the vendor. If you are installing keys and certificates from Sun Microsystems' Internet Commerce Group (ICG), see Chapter 3.

The `skiplocal` command creates and manages all local key types, including UDH certificates, on your system. You can have more than one UDH certificate on your system. Your local identities can also be of different lengths (moduli), depending on the version of *SunScreen SKIP* that you have. The default will always be the largest modulus you can generate.

Note - Local secret is the term used for an encryption certificate and key.

♦ **To generate an UDH key pair locally, type**

```
skiplocal keygen
```

Note - If you have local identities of different strengths, such 512 (Global), 1024 (Export), and 2048 (U.S. and Canada Only), use the argument `-m` followed immediately with the bit size of the modulus without an intervening space (Figure 1–1).

When generating an unsigned certificate, no authority exists to certify the identities. This means that each party must verify the name of the certificate over the telephone or some other trusted channel. Without verification through a secure channel, you have no way of knowing if the certificate belongs to the correct party or not.

In Figure 1–1 the `skiplocal keygen` command was used to generate a local key pair, in this case with a 512–bit modulus.

```
# skiplocal keygen -m 512
generating local secret with 512 modulus size
It would help the quality of the random numbers if you would
type 50-100 random keys on the keyboard.  Hit return when
you are done.
100
Format: Hashed Public Key (MD5)
Name/Hash: 9e 23 db 35 a2 c2 d8 17 20 19 21 99 3d c9 06 e1
Not valid Before: Sun Aug 25 17:00:00 1996
Not valid After: Sat Aug 25 17:00:00 2001
g: 2
p:
f52aff3ce1b1294018118d7c84a70a72d676c40319c807297aca950cd9969fab
d00a509b0246d3083d66a45d419f9c7cbd894b221926baaba25eca55e92a055f
public key:
0b5522b769b3d2b8098e69312a941ce7e6de9e1635ca09dd780b328db7114173
9e9bb46a3d0d183372d98d7c2a0d850b70fad05edaaaa865ae5dddf618cadbff
Added local identity slot 0
```

Figure 1–1 512-bit Modulus

In Figure 1–2 the `skiplocal export` command is used to print out the local system's current information in a form that can be sent (for example, via e-mail) to other users who wish to communicate with you.

Caution - The defaults proposed by `skiplocal export` work well if you and the party with whom you wish to communicate have one key and one network interface. If you have some other configuration, you should not use `skiplocal export`.

A safer solution than using `skiplocal export` is to have each user run `skiptool` and then call each other on the telephone and type the other person's key ID in the Remote Key ID field in the add window (See Chapter 3).

```
On local machine (mysun)

# skiplocal export                          displays ACL entry
in export format
skiphost -a mysun -R
0x24be59e388dadfa6814885d1e5f79de9 -r 8 -s 8 -k des-ede-k3 -t \
des-cbc -m md5
# skiplocal export | mail username@host

Mails above text to the username@host


On peer machine (host)

# skiphost -a mysun -R
0x24be59e388dadfa6814885d1e5f79de9 -r 8 -s 8 -k des-ede-k3 -t \
des-cbc -m md5

Copied from mail message sent by mysun

Adding mysun:                 SKIP params:
    IP mode:                      tunneling
    Tunnel address                mysun
    Kij  alg:                     DES-EDE-K3
    Crypt alg:                    DES-CBC
    MAC alg:                      MD5
    Receiver NSID                 MD5 (DH Pub. Value)
    Receiver key id           0x24be59e388dadfa6814885d1e5f79de9
    Sebder NSID                   MD5 (DH Pub. Value)

done.
```

Figure 1–2 Sending and Loading an ACL Entry

Caution - Even when using `skiplocal export`, make sure you both verify the key ID over the telephone with the other party to make sure no one is impersonating them.

In Figure 1–3, the `skiplocal list` command is used to list the current local identities.

```
# skiplocal list
Local ID Slot Name: 0Type: Software Slot
    NSID: 8 MKID (name): 24be59e388dadfa6814885d1e5f79de9
    Not Valid Before: Tue Aug  6 17:00:00 1996
    Not Valid After: Mon Aug  6 17:00:00 2001
    Modulus size: 2048 bits

Local ID Slot Name: 1Type: Software Slot
    NSID: 8 MKID (name): 8ace505b602127f38e08f74f13d0c915
    Not Valid Before: Sun Aug 25 17:00:00 1996
    Not Valid After: Sat Aug 25 17:00:00 2001
    Modulus size: 2048 bits

Local ID Slot Name: 2Type: Software Slot
    NSID: 8 MKID (name): 9e23db35a2c2d817201921993dc906e1
    Not Valid Before: Sun Aug 25 17:00:00 1996
    Not Valid After: Sat Aug 25 17:00:00 2001
    Modulus size: 512 bits

#
```

Figure 1–3 Listing All Local Identities

For more information on the `skiplocal` command, refer to the man pages for *SunScreen SKIP*.

Note - If you installed an UDH certificate during installation, the information in Chapter 2 will not apply to you unless you also plan to install SunCA keys and certificates. You may use SKIP UDH certificates and SunCA keys and certificates at the same time on *SunScreen SKIP*.

Installing Your Network Interface

The `skipif` command is used to install *SunScreen SKIP* on a network interface.

- **If you are adding *SunScreen SKIP* to a machine with only one interface, make sure that you are root and type**

```
skipif -a
```

- **If you are adding *SunScreen SKIP* to a machine with multiple interfaces, make sure that you are root and type**

```
skipif -i <networkinterface> -a
```

Note - Replace <*networkinterface*> with the interface that you wish to specify. If you do not specify the network interface, it attaches to the first network interface that it finds.

- **You can add SKIP on more than one interface. In that case, you need to run the** `skipif -a -i` **<*interface*> command for each interface on which you want to use SKIP.**

- **If you want to use SKIP on all the network interfaces present in the system, simply use the** `skipif -a -i` **all command.**

Rebooting Your System

After you have installed the software, generated and installed the local identities, and installed the network interface, you must reboot your system.

- **To reboot the machine, type**

```
init 6
```

Passphrase Protection

SunScreen SKIP includes a new, optional feature that allows you to protect your locally stored secrets with a passphrase. A passphrase differs from a password in that it is longer and capitalization counts. It permits you to assign a global passphrase that will be used to encrypt all of your SKIP secret values. Your passphrase should be one that you can remember, but that is hard to guess. You can change the passphrase or delete it at any time. After you set, change, or delete your passphrase, you should run

```
skipd_restart
```

to reinitialize your key manager.

Caution - Once you have protected your secret values with a passphrase, each time that you reboot you will *not* be able to run *SunScreen SKIP*-encrypted connections because your system cannot get to your locally stored secrets with the passphrase. You must run

 # skipd_restart
which will then prompt you for your passphrase.

Caution - If you forget your passphrase, there is no way to discover it or recover it. Your protected locally stored secrets will no longer be available. If you do not know the passphrase and you want to reinstall or upgrade the software, you must first remove the old software and its locally stored secrets. See Section 2.2.2 *Upgrading the Software*. The old locally stored secrets will remain encrypted with the old passphrase and will be unavailable.

Once you set a passphrase, you will be prompted for it each time you add a new local identity (through `skiplocal add` or `skiplocal keygen`).

Activating Your Passphrase

To activate your passphrase, use the following procedure:

1. **Type**

```
skiplocal passwd
```

2. **You will be prompted as follows:**

```
You are now assigning a global passphrase which will be used to
encrypt all of your SKIP secret values. Please choose a passphrase
which you will remember, but will be hard for someone else to guess
New global passphrase: <type a new passphrase>
again: <type the new passphrase>
```

3. **To reinitialize your key manager, type**

```
skipd_restart
```

Changing Your Passphrase

To change your passphrase, use the following procedure:

1. **Type**

```
skiplocal passwd
```

2. **You will be prompted as follows:**

```
You are now changing the global passphrase which is used
to encrypt your SKIP secrets
 Global passphrase: <type a old passphrase>
 New Passphrase: <type a new passphrase>
 again: <type the new passphrase>
```

3. **To reinitialize your key manager, type**

```
skipd_restart
```

Removing Your Passphrase

To remove your passphrase, use the following procedure:

1. **Type**

```
skiplocal rmpasswd
```

2. **You will be prompted as follows:**

```
You are now removing the global passphrase which will be used
to encrypt all of your SKIP secrets
 Global passphrase: <type your passphrase>
```

If it matches, all locally stored secrets are decrypted and stored and the passphrase feature is disabled.

3. To reinitialize your key manager, type

```
skipd_restart
```

You can use `delpasswd` as an alias for `rmpasswd`.

CHAPTER 2

Installing Keys and Certificates

It you have installed *SunScreen SKIP* on your machine, you must set it up so that it can talk to other systems. This chapter tells you how to install keys and certificates on your system.

Note - If you installed an UDH certificate during installation, the information in this chapter will not apply to you unless you also plan to install SunCA keys and certificates.

There are two kinds of certificates that you can use with *SunScreen SKIP*:

- UDH
- SunCA

Which certificates you choose to use is determined by the security policy of your company.

At the end of the installation process in Chapter 1 (“Installation Procedure”), you created a SKIP UDH certificate using the `skiplocal` command.

Note - You must be root to use the command-line commands.

You may use the `install_skip_keys` command to install SunCA keys and certificates on *SunScreen SKIP* at the same time. This section shows how to install certificates signed by the SunCA.

Keys and Certificates

Keys

Traditional cryptography relies on the sender and receiver of a message knowing and using the same secret key. When both sender and receiver use the same secret key, the system is referred to as a symmetric or single-key crypto system. The problems with using the same secret key are: how is one selected, how do the parties inform each other of the secret key if they are not physically in the same location, how do they change keys from time to time, and how is the secret key kept secure.

Public-key cryptography was proposed as a solution to the problems found in traditional, symmetric key cryptography. In public-key cryptography, each person, host, or network participating in a coded exchange, receives a pair of keys: one public and one private. The private key is kept a secret and the public key is published so that anyone who wishes to communicate confidentially with a person or an entity can do so by encoding their message using the public key. The confidential message can then only be decoded by the private key, which is kept in the sole possession of the intended recipient.

SKIP is a public-key, certificate-based, key-management scheme. It uses certified Diffie-Hellman public values to eliminate the need for prior communications between two entities wishing to exchange encrypted data.

There are times when it is useful to allow a system to have more than one pair of public-private keys. For example, different key sizes may be required when communicating with subsidiaries in other countries because of U.S. or local regulations. To meet these user requirements, *SunScreen SKIP*'s implementation permits a system to possess as many local keys as required. Public-private key pairs like UDH keys can be used for authentication.

Certificates

To ensure that a public key is authentic (that is, it has not been tampered with by an unauthorized user and does indeed belong to the claimant), the public key is normally signed by a Certification Authority (CA). The result, a digital document called a certificate, can be freely passed around the network. Its authenticity can be verified by anyone holding the CA's signature information; that is, the CA's public key.

Before any form of encrypted communication can begin, the parties involved in the transaction must exchange certificates. This is a manual procedure in that the certificate and possibly the key are provided by the certifying agency on physical media: tape, diskette, or CD-ROM. The user must load them into the system through a command-line interface.

Key and Certificate Management

Secure key management is a necessary requirement for any cryptographic product. Users must be able to obtain keys as required for their security needs, have a method of looking up other's public keys, publicize their own keys, and determine that a key is valid. Certificates are used for this purpose.

Certificates must be unforgettable, obtainable in a secure manner, and processed in such a way that an unauthorized user cannot misuse them. This means that the network manager must handle the following issues:

- Loss or compromise of a private key
- Verifiable signature after key expiration
- Expiration dates
- Secure storage of private keys

Adding Certificates or Local Identities with `install_skip_keys`

The `install_skip_keys` command is used to install key packages that have been received from a key server or from one of the SunCAs. If used with `-icg`, it means that the SunCA or the SunCAglobal CA certified the keys. The SunCA certifies 1024–bit and 2048–bit modulus certificates, and the SunCAglobal certifies 512–bit certificates.

To communicate with a SunScreen SPF-100 or SunScreen SPF-100 G, you need to use SunCA or SunCAglobal certificates.

Note - The `install_skip_keys` command is not used to add someone else's certificate. It is only used to install local identities for CA key packages.

The Figure 2–1 shows installing a SunCAglobal key and certificate from diskettes. After installing the key and certificate, because you have added a new local identity, you must either run the `skipd_restart` command or reboot your system to initialize the key manager.

```
# install_skip_keys -icg /floppy/unnamed_floppy
Added CA certificate as ca-slot 0
Added local identity slot 3
added 0a1030cc to database
/opt/SUNWicg/bin/install_skip_keys: you should now reboot the
machine to initialize SKIP.
```

Figure 2–1 Installing a SunCA Global Key and Certificate from Diskette

For more information on `install_skip_keys`, see the man pages.

CHAPTER 3

Managing SunScreen SKIP Through skiptool

Now that you have installed *SunScreen SKIP* and the local keys on your machine, you must set it up so that it can talk to other systems. You do this primarily through the graphical user interface (GUI).

This chapter tells

- How to use the Graphical User Interface (GUI) or `skiptool` (“Using the Graphical User Interface (skiptool)” on page 24)
- How to configure SKIP (“Configuring *SunScreen SKIP*” on page 24)
- How to enable SKIP (“Enabling SKIP” on page 46)
- How to verify SKIP installation and set up (“Verifying the SKIP Installation and Set Up” on page 48)
- How to view statistics (“Viewing *SunScreen SKIP* Statistics ” on page 50)
- How to manage keys (“Key Management with `skiptool`” on page 61)

Note - If you are using *SunScreen SKIP* as part of another application (such as *SunScreen EFS*), much of the configuration has already been done. Refer to the appropriate application manual for specific instructions on what needs to be configured through the *SunScreen SKIP* interfaces.

Using the Graphical User Interface (skiptool)

SKIP provides two interfaces for configuring and managing *SunScreen SKIP*: `skiptool`, the GUI; and `skiphost`, the command-line interface. The command-line interface is discussed in Chapter 4. The easiest way to set up your ACL is through `skiptool`. The GUI 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 unencrypted [clear]), as well as determine how to deal with unauthorized hosts that try to connect to your system. It also allows you to view the following statistics:

- Network Interface Statistics
- SKIP Header Statistics
- Key Statistics
- Encryption Statistics (for Versions 1 and 2)
- Authentication Statistics

To run `skiptool`, you must be able to become root on your system. In addition, enable access for any client to the X server for Solaris 2.x systems by entering the `xhost +<localhost>` command; for example,

```
% xhost +mysun
```

before you become root.

Configuring *SunScreen SKIP*

You can configure only one interface at a time using `skiptool`. If you have more than one network interface, you must configure each separately.

Configuring *SunScreen SKIP* requires completing several steps:

1. **Adding authorized systems (“Adding Authorized Systems” on page 28)**
   - Communicating in the clear (*Off*) (“Communicating In the Clear (Off)” on page 43)
   - Communicating using SKIP Version 1 (“Communicating Using SKIP Version 1” on page 43)
   - Communicating using SKIP (“Communicating Using SKIP” on page 43)

- Communicating using ESP/AH (“Communicating Using ESP/AH” on page 45. This is typically used in test mode.)

2. **Adding excluded systems, if any (“Adding Excluded Systems” on page 45)**

3. **Setting up the behavior for unauthorized systems (“Behavior for Authorized Systems” on page 46)**

4. **Enabling SKIP (*Access* control button is enabled) (“Enabling SKIP” on page 46)**

5. **Verifying the installation and set up (“Verifying the SKIP Installation and Set Up” on page 48)**

   There are two optional steps that are helpful in troubleshooting and in tuning key usage, respectively.

1. **Viewing *SunScreen SKIP* statistics (“Viewing *SunScreen SKIP* Statistics ” on page 50)**

2. **Key management with `skiptool` (“Key Management with `skiptool`” on page 61)**

   Each step is described in further detail in the following sections.

   When `skiptool` is started just after the initial installation of the software, the following defaults are in effect:

   - Access control is disabled
   - Unauthorized systems are set at No Access

▼ Starting skiptool

To start `skiptool`, complete the following steps:

1. **Open a window, and type**

```
% xhost +mysun
```

2. **Become root, and type**

```
# skiptool&
```

If you are configuring a system that has multiple network interfaces, you can specify the interface following the `skiptool` command; for example, `skiptool le1`.

The main window of `skiptool` appears, as shown in Figure 3–1.

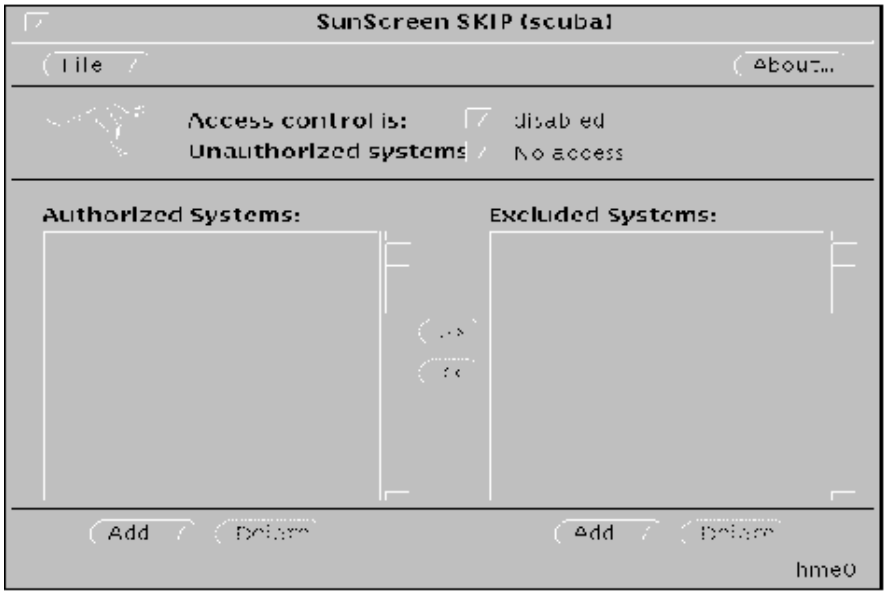

Figure 3–1 `skiptool` Main Window

The skiptool Main Window

The `skiptool` main window has several important features:

- The File button
- The Access Control buttons
- The Authorized and Excluded Systems lists
- The Add and Delete (Management) buttons

File Menu

The file menu has five submenus:

Load—Loads current ACL from the kernel. This is useful if you have modified the ACL through other tools and want to update the configuration in `skiptool`.

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: (1) Network Interface Stats, (2) SKIP Header Stats, (3) Encryption Stats (Version 1), (4) Encryption Stats (Version 2), (5) Key Stats, or (6) Authentication Stats.

Save—Makes the configuration permanent. Before saving, it prompts you to add any systems that are in use, that have access, and that are not currently on the authorized list. The next time that you reboot this configuration is used. Quitting and restarting `skiptool` will not affect either saved or unsaved changes in configuration. (Another way to save the current ACL is to use the command-line tool `skipif` with the `-s` option.)

Note - If you do not save the changes in the configuration, you can use them until the next time you reboot your machine when they will no longer be in effect.

Exit—Closes all open windows and quits *SunScreen SKIP*. The Statistics window will not close when you quit `skiptool`.

Access Control Buttons

Access Control button—This button toggles to enable or disable SKIP. When SKIP is enabled, the ACL rules apply. (For example, you could have only the “default” entry in the authorized systems list and some entries in the excluded systems list. In this case, any host except those that are in the excluded systems list could connect.) When SKIP is disabled, any system can connect, if the “default” entry is configured in the clear.

Unauthorized System button—This button is used to set the policy regarding unauthorized systems.

Note - If a default authorized host entry exists, this policy does not take effect. The default entry has the name “default” and the ACL looks for this entry (in authorized or excluded host lists) if it cannot find a given entry that matches the host or network criteria.

The policy can be

No Access—Does not allow unauthorized hosts 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—Any host that sends packets to this system is automatically added to the authorized systems list.

Note - It is recommended that you do not change the value from "No Access."

Authorized Systems/Excluded Systems Lists

Authorized Systems—A list of systems that are authorized to have access to this host. System types are host, network, or nomadic. Secure systems are denoted by a padlock or the Sun Microsystems' logo next to the system name, depending on the type of security being used.

Excluded Systems—A list of systems that are specifically denied access to your system. When you move or add a system to the excluded list, it is immediately excluded.

`skiptool` allows you to move systems from the list of authorized systems to the list of excluded systems and *vice versa* with the arrows between the two lists.

Management Buttons

These buttons enable 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 pop-up menu where the system type to be added to the ACL is selected:

Host—Adds an individual host, either with or without security.

Network—Adds a network, either with or without security.

Nomadic—Adds a nomadic identity, with SKIP Version 1 or SKIP Version 2 security.

Delete—Deletes the selected system from the list. When an item is deleted, the deletion occurs immediately and cannot be undone.

You may also move ACL entries from one list to another with the arrow buttons. These arrow buttons make it easy to add or delete system when troubleshooting.

Caution - If you add or delete ACL entries from one list to another, the addition or deletion takes effect immediately.

▼ Adding Authorized Systems

Any remote host with which you want to communicate (send or receive data) must be configured using the Add pop-up window.

An authorized host may or may not be using encryption. The *Add* pop-up window provides four options:

- Off or not using encryption
- Using SKIP encryption
- Using SKIP Version 1 encryption
- Using ESP/AH (manual keying)

You add hosts to the authorized systems list using the Add button, located at the bottom left of the main window of `skiptool`.

The valid types of remote hosts that you can add to your ACL are

- Host
- Network
- Nomadic

Caution - When setting up *SunScreen SKIP*, be sure to include any NFS servers and NIS or DNS name servers on the authorized systems list, otherwise your system may hang.

Note - To avoid problems such as this, a safe approach at the beginning is to add the clear "default" entry. Once you become more comfortable with SKIP configuration, you can remove it.

To determine the servers your system communicates with, use the following commands:

- **For NFS servers, type**

```
mount
```

- **For NIS servers, type**

```
ypwhich
```

- For DNS servers, consult your system administrator
- It might be useful to verify the current routing entries used by the local system. To verify the current routing entries, type `netstat -rn` and add specific network ACL entries.

If you do not specify a system that you currently have in use when you enable access control, a menu will come up and ask if you want to add the system. It also checks for multicast routers that are being used for others and adds them to the proposed list of systems to add.

Regardless of the type of system that you are adding to the ACL, you must implement the same policy on both your machine and the entity with which you wish to communicate securely over the intranetworks or internetworks. If you do not configure both systems properly, the packets are silently dropped and it appears as if that particular host does not exist. `skiplog` is useful in diagnosing this situation.

When you click on the Add button, the Add pop-up window appears. From the menu in this window, you select the type of connection: Host, Network, or Nomadic. Next, use the pull-right menu to set the security level. After you have selected the level of security, the appropriate 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. Table 3–1shows what type of encryption can be used with hosts, networks, or nomadic systems. The procedures in the sections following the table detail how to set up each encryption option.

TABLE 3–1 Type of Security Available, by Type of System

Type of System	**Type of Security**			
	Off (none)	**SKIP**	**SKIP (Version 1)**	**ESP/AH (manual keying)**
Host	X	X	X	X
Network	X	X	X	X
Nomadic	—	X	X	—

Adding a Host or Network with No Encryption

This procedure is used to allow a host or network access to your system without using any encryption.

1. **Click and hold 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 or Network. (*Nomadic* does not offer this option.)**

3. **Pull right on the type of connection and select Off.**

   The Add Host properties or Add Network properties dialog box will appear (Figure 3–2).

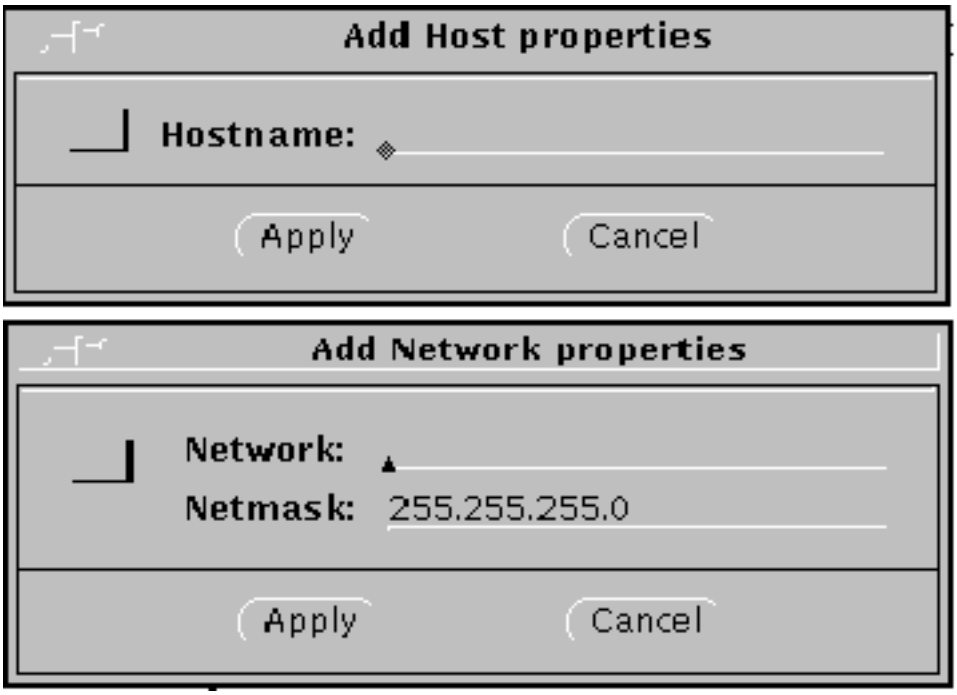

Figure 3–2 Add Host/Properties—No Encryption

4. **In the Add Host or Network properties window, enter the name or IP address of the host system to be added to your ACL.**

   In the case of a network, you must define the network with the IP address and the netmask.

5. **Click the Apply button.**

Setting Up Security for a Host, Network, or Nomadic System

These procedures enable a host, network, or nomadic system access to your system according to the encryption rules set up using one of the procedures below. Remember, both your system and the other system need to use the same properties in order to communicate.

Explanations of the Dialog Box Parameters

The three encryption dialog boxes (SKIP, SKIP Version 1, and ESP/AH) use common set-up parameters, as you can see in Figure 3–3 through Figure 3–10. Explanations of the parameters follow the figures. The procedure follows the explanations.

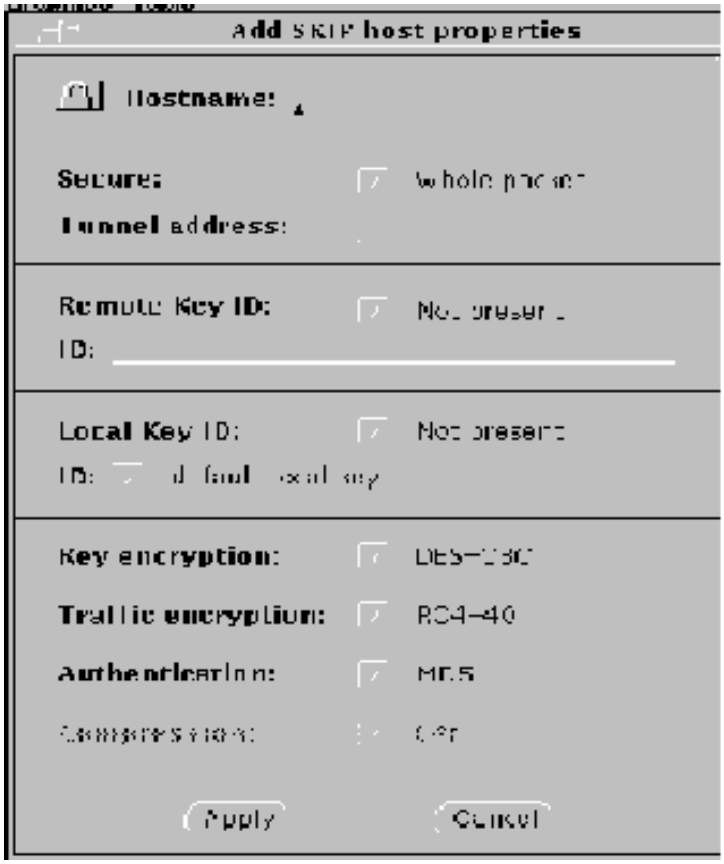

Figure 3–3 Host—Add SKIP Host Properties

Figure 3–4 Host—Add SKIP Version 1 Properties

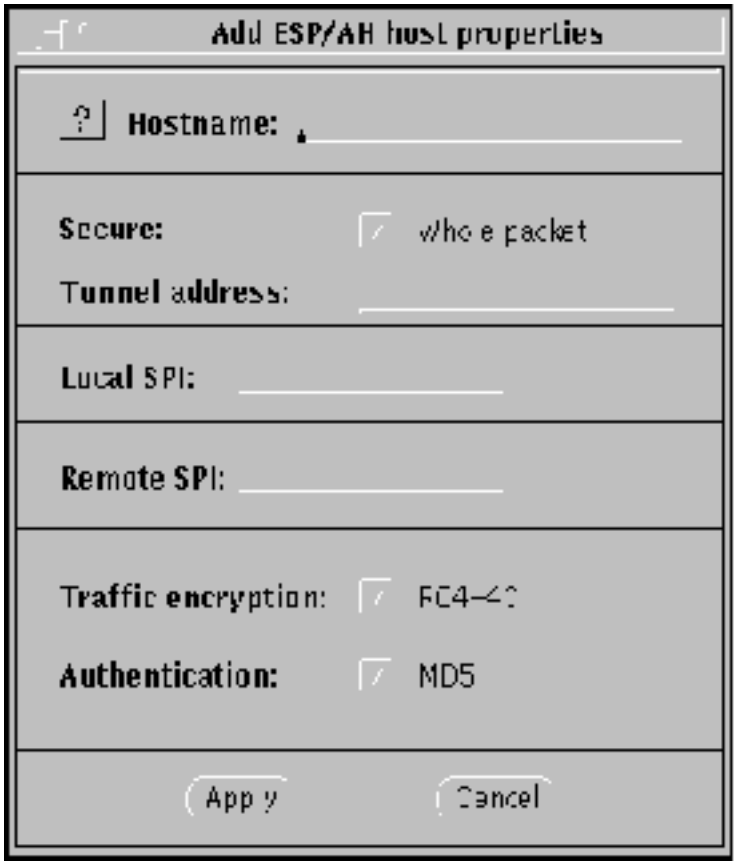

Figure 3–5 Host—Add ESP / AH Host Properties

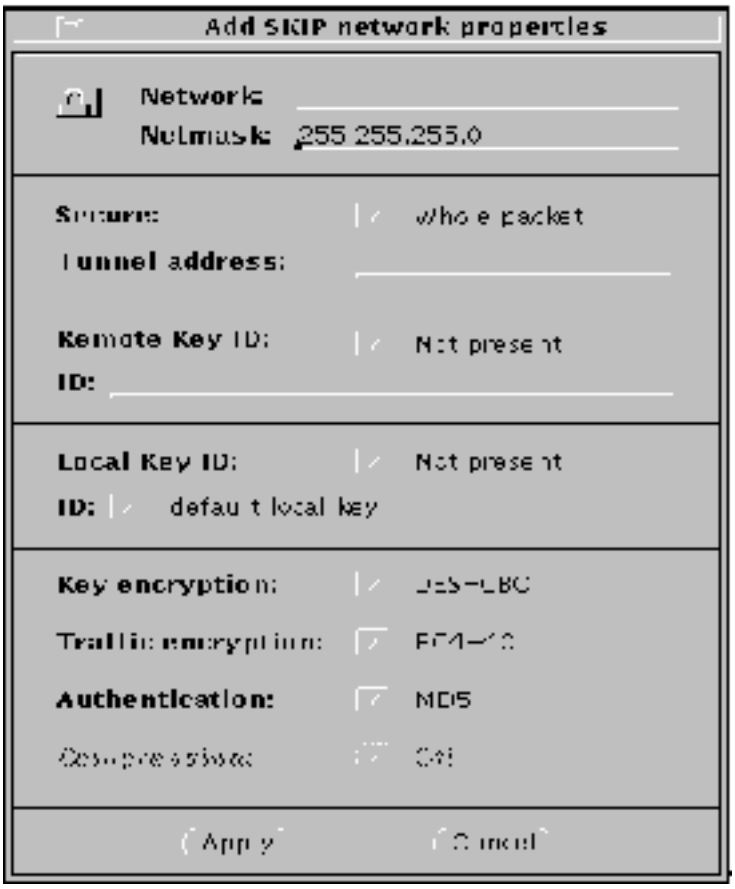

Figure 3–6 Network—Add SKIP Network Properties

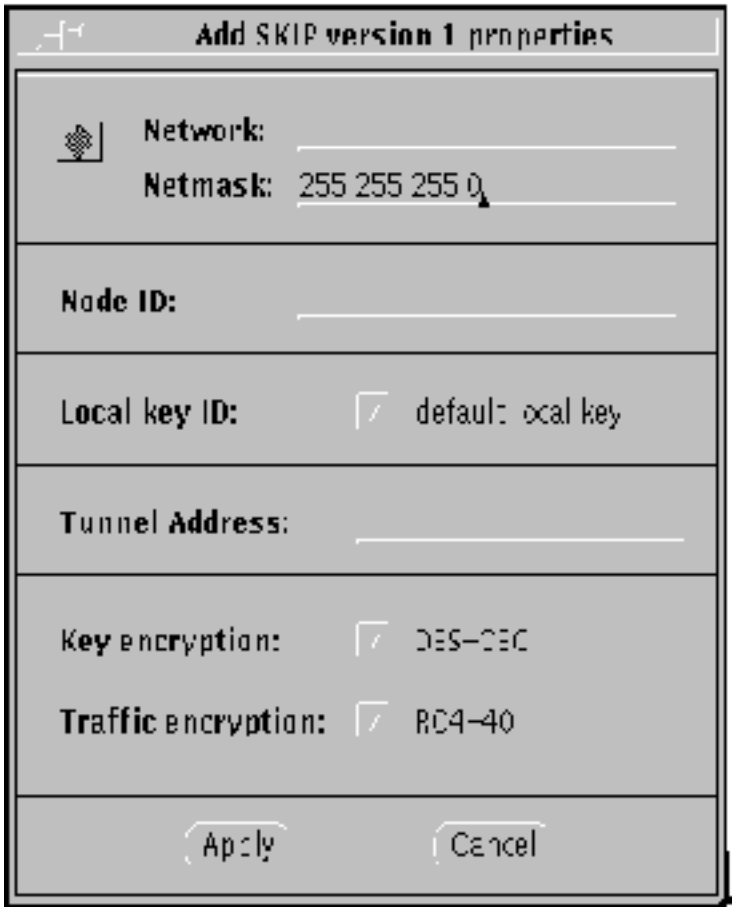

Figure 3–7 Network—Add SKIP Version 1 Properties

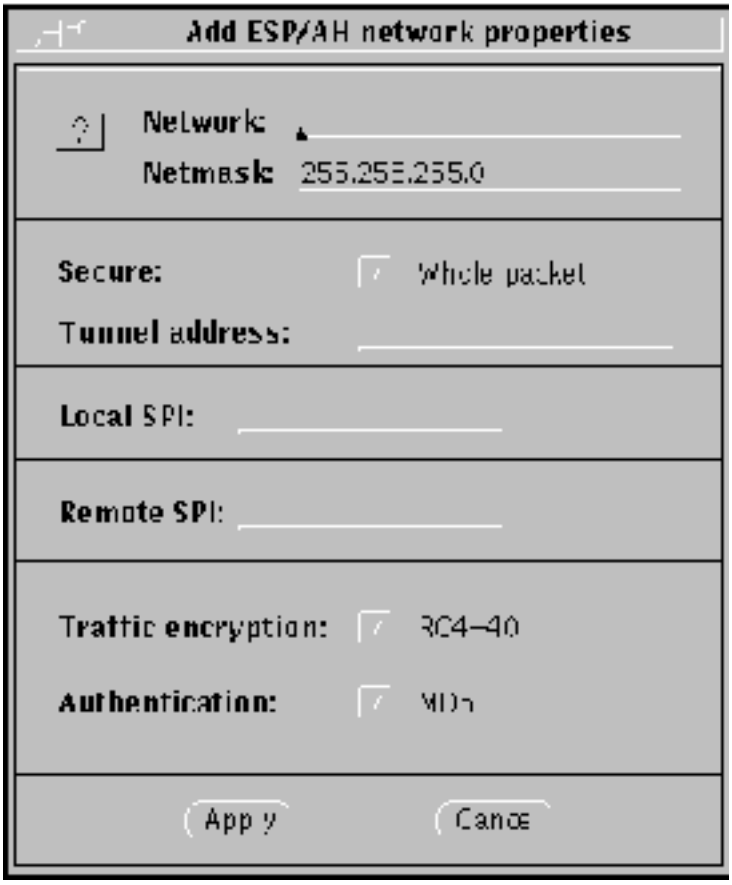

Figure 3–8 Network—Add ESP/AH (Manual Keying) Network Properties

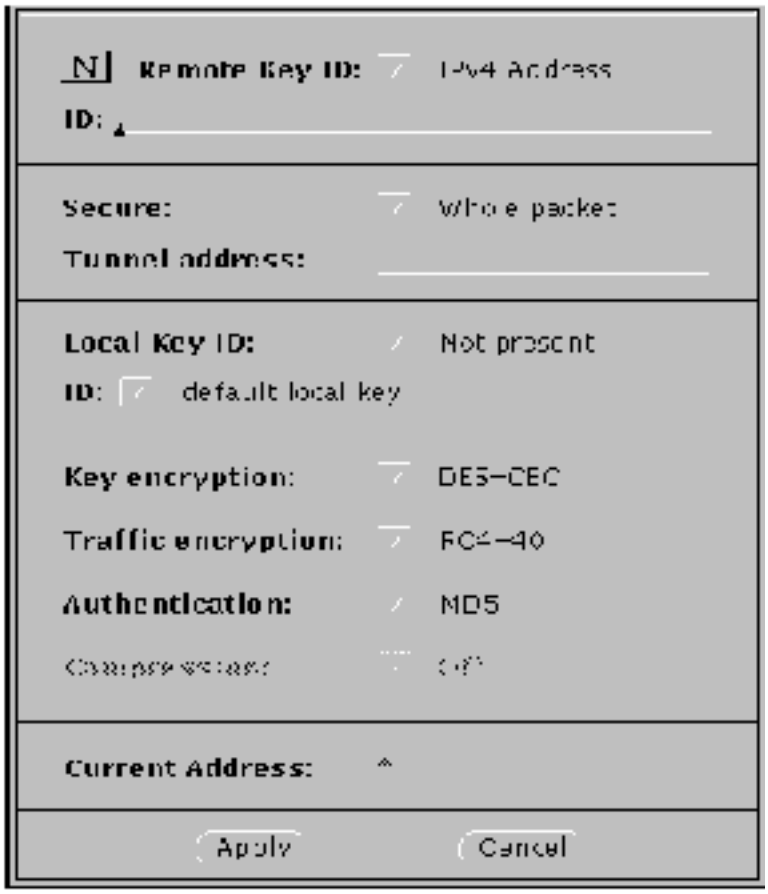

Figure 3–9 Nomadic—Add SKIP Properties (Nomadic)

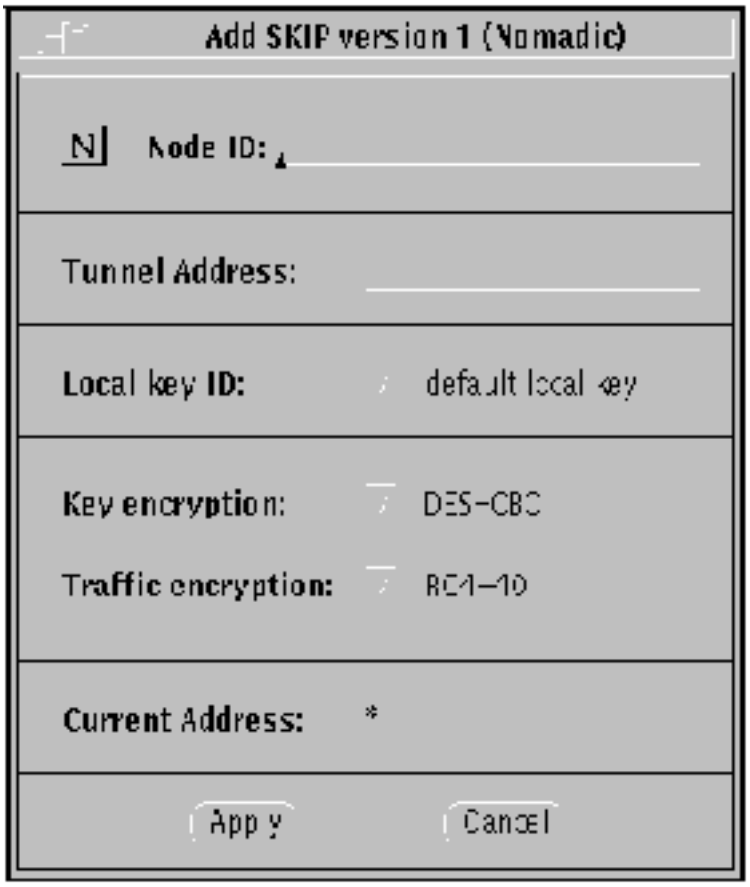

Figure 3–10 Nomadic—Add SKIP Version 1 (Nomadic)

- Hostname / Network / Nomadic. Enter the name of the host or nomadic system, or the IP address of the host or network.
- Netmask. (network only) Enter the netmask of the network. The default (255.255.255.0) is already entered.
- Secure button. (SKIP and ESP / AH only) Set to either Whole packet ("tunnel mode") or Data only ("transport mode"). Whole packet is recommended because it offers a greater degree of security.
- Node ID. (SKIP Version 1 only) This is the IPv4 key ID.
- Tunnel Address. Use the tunnel address as the destination IP address. Tunnel address is generally used for clients of 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 must remain unknown or hidden from the rest of the world. This is called topology hiding. This field is not available if you select Data only.
- Local / Remote SPI. (ESP / AH only) You need to provide some sort of identifier for the local and remote systems when using manual keying. These are converted to

hexadecimal numbers by SKIP. The Local security parameters index (SPI) is your machine, and the Remote SPI is the destination machine. Alternatively, you can enter the Local/Remote SPI values directly in hexadecimal by typing an eight-digit hexadecimal quantity with the prefix “0x.”

- Remote Key ID button. (SKIP only) Select whether you want the remote system’s key ID included in SKIP packets and, if so, the namespace that key ID occupies. Selecting Not Present means that the receiver key ID will not be sent.

  The following namespaces are listed in this menu:

  - Not Present
  - IPv4 Address
  - MD5 (DH Public Value)

  Not Present is the default. It uses the IP address of the remote system to identify its certificate. If a remote system has a key ID other than that identified by its IP address, set the namespace and indicate the remote system’s key ID in the ID field.

- Remote Key ID field. (SKIP only) The namespace indicated in the Remote Key ID field is determined by the type of certificate (Table 3–2) that you are using or have obtained for this system:

TABLE 3–2 Remote Key ID Field

Certificate Type	Remote Key ID Field
CA (Sun or other)	IPv4
Self-generated unsigned key	MD5 (DH Public Value)

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 (such as 0x0a000000). It must contain the appropriate key ID for the system being authorized based upon the selection made with the **Remote Key ID** button. 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.

- Local Key ID and ID buttons. Use the Local Key ID button to indicate whether you want your local system to send its key ID in the SKIP packet and, if so, the namespace that key occupies. If you select Not Present, the sender’s key ID is not sent in the packet and the remote system uses the local system’s IP address to decide what key to use.

Note - If you have installed new local keys after you have started `skiptool`, `skiptool` will not list them. You must restart the key manager with the `skipd_restart` command to list them and rerun `skiptool`.

All the local-key times installed for this host are listed. Select the namespace for the local key that is to be used for communication with the above host. Once you have selected the namespace, click on the **ID** field to select the key to be used, in hexadecimal, for communication with this host.

- Key Encryption button. Selecting this button lists the available key encryption algorithms. The algorithms available are determined by the system type and the selected encryption method selected.
- 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, the version of *SunScreen SKIP*, and the method of encryption selected.
- Authentication button. Use the authentication button to select the type of authentication for the packets. Currently, *SunScreen SKIP* supports only one type of authentication—MD5. You can also select None for no authentication.
- Compression button. Compression is not available at this time.

▼ Adding Authorized Systems with Encryption

1. **Click and hold on the Add button at the bottom of the authorized systems list on `skiptool`'s Main Window.**

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

3. **Pull right on the type of connection and select the type of encryption that you want to use.**
   - If the remote host system also uses SKIP and the traffic between your systems is to be encrypted, select SKIP.
   - For systems using Sun Microsystems' SunScreen SPF-100, select SKIP Version 1.
   - 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 ACL.**

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

6. **Each type of encryption requires that certain options be set.**

The parameters selected are determined by the type of system being authorized and your security policies. The options to be considered are based on the method of encryption selected. They are

- For systems using SKIP: Tunnel address, Remote Key ID, Local Key ID. If you leave the tunnel address blank, it will default to the peer's address.
- For SKIP Version 1: Key ID, Tunnel address.
- For ESP / AH systems: Tunnel address, Local SPI, Remote SPI.

7. **Select the appropriate algorithms buttons for Key encryption, Traffic encryption, and Authentication.**

   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 key encryption algorithms. The algorithm available is determined by the type of system and selected method of encryption.
   - Traffic Encryption button: Selecting this button lists the algorithms available for encryption between your system and the remote system. The algorithms that are available for key and traffic encryption depend on the packages that were installed on the system, such as core product and key upgrades. The algorithms available determine the type of system and the method of encryption selected.
   - Authentication button: Use this button to select the type of authentication for the packets.
   - Compression button: Compression is not currently supported.

8. **Click Apply to add the host to the authorized systems list.**

   Refer to the previous section for descriptions of the fields and buttons.

   Repeat Steps 1 though 8 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 with which you wish to communicate through encrypted channels. If the configuration on your system does not match that of the party with which you wish to communicate, the packets are silently dropped. It will simply appear as though that host no longer exists.

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 to allow hosts that are not listed in the ACL to communicate in the clear. It may, however, be used to create a default encryption rule.

Note - If the default ACL remains and is set to Off, it is unnecessary to add any entity with the Off security option. Further, if the default ACL remains and is set to Off, the option set by the Unauthorized Systems button never goes into effect because all systems are considered as authorized.

Communicating In the Clear (Off)

Typically, the NIS and DNS servers to which your servers have access are set up as communicating with your system in the *clear* or *unencrypted*. In addition, any host that does not use an encryption package must be set up to communicate with you in the clear.

▼ Communicating Using SKIP Version 1

Complete the following steps 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. **Enter the Node ID.**

   This is the IPv4 key ID.

3. **Local Key ID and ID buttons.**

   Use the Local Key ID button to indicate whether you want your local system to send its key ID in the SKIP packet.

4. **Set the *Tunnel Address*, if you are using topology hiding.**

   Tunnel addressing is generally used for clients of 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 Key encryption algorithms are DES-CBC and RC2-40. Available Traffic encryption algorithms are RC4-40 and RC2-40.**

▼ Communicating Using SKIP

Complete the following steps 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 (‘tunnel mode”) or Data only (“transport mode”).**

   Whole packet is recommended because it offers a greater degree of security.

3. **Set the Tunnel address, if you are using topology hiding.**

   Tunnel addressing is generally used for clients of 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.**

   If so, what namespace does that key occupy. By selecting Not Present, the receiver key ID is not sent.

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

Certificate Type	Remote Key ID Field
CA (Sun or other)	IPv4
Self-generated unsigned key	MD5 (DH Public Value)

5. **The following namespaces are used in this menu**:

Not present	IPv4 Address	MD5 (DH public Value)

6. **If the Remote Key ID field has been set to something other than Not Present, enter the key ID in hexadecimal format in the ID field (0x0a000000).**

   It must contain the appropriate key ID for the system that is 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 keyID on the diskette or from the Local key ID field of the other host.

7. **Select the appropriate key and traffic algorithms for the Key and Traffic encryption buttons.**

   Available Key encryption is None, DES_CBC, and RC2-40. Available Traffic encryption is None, RC4-40, and RC2-40.

8. **Authentication button.**

Use the authentication button to select the type of authentication for the packets. Currently, *SunScreen SKIP* supports only one type of authentication—MD5. You can also select None for no authentication.

9. **Compression button.**

   Compression is not available at this time.

Communicating Using ESP/AH

ESP/AH (also called 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` files must be configured.

▼ Adding Excluded Systems

If the default entry remains on the authorized systems list, then any remote host with which you want to prevent communication must be configured using the Add button located under the excluded systems list. When setting up an excluded system, you only need to enter the hostname for hosts and network number for networks. For nomadic systems you need to specify the key IDs.

If the state of the host or network changes to an authorized system, you must delete the system from the excluded systems list and add it to the authorized systems list.

The easiest way to exclude a system is to move it from the authorized systems list with the arrow button to the excluded systems list. The arrow buttons make it easy to add or delete systems when troubleshooting and the host is already present in the authorized systems list. If the host does not already exist on one of the lists, it is simpler to add it directly on the excluded systems list so that you can move it easily with the arrow button when you wish to add it to the authorized systems list.

Note - If you move an encrypted host from the authorized systems list to the excluded systems list with the arrow button, *SunScreen SKIP* retains the encryption parameters so that if you later move this host back to the authorized systems list, its parameters are restored.

You can also complete the following steps to exclude a system:

1. **Click on the Add button at the bottom of the excluded systems list on skiptool's 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 that you want to deny access to your system.**

   If you are excluding a nomadic system, also enter the key ID.

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

Caution - If you add or delete ACL entries from one list to another, the addition or deletion takes effect immediately.

Behavior 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 obtain access to your system. An unidentified system is unrecognized by SKIP; that is, it is not on either the authorized systems list or the excluded systems list.

Use the Unauthorized Systems button on the main window to select the action SKIP should take when an unidentified system attempts access. When you remove a default entry from these lists, SKIP will take one of the following three actions:

- No Access
- Ask for confirmation
- Add automatically

It is recommended that you leave this entry in the default selection of No Access for greater security.

If you quit skiptool or if you reboot your system, the selection will revert to No Access.

Note - If a default ACL entry is on the authorized systems list, this option does not take effect.

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

Enabling SKIP

The last step in setting up *SunScreen SKIP* is to enable access control for the system. Enable *SunScreen 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 with which you are talking in the clear. It detects the NFS, X Windows, NIS, and DNS

servers with which you are communicating and offers the possibility of adding the systems automatically to the ACL when you select Add from the Required Systems window (Figure 3–11). Choosing Cancel can hang your system or prevent your access to the system or network the next time you try to log in because certain necessary servers may not have been added. To prevent this, select disable after canceling.

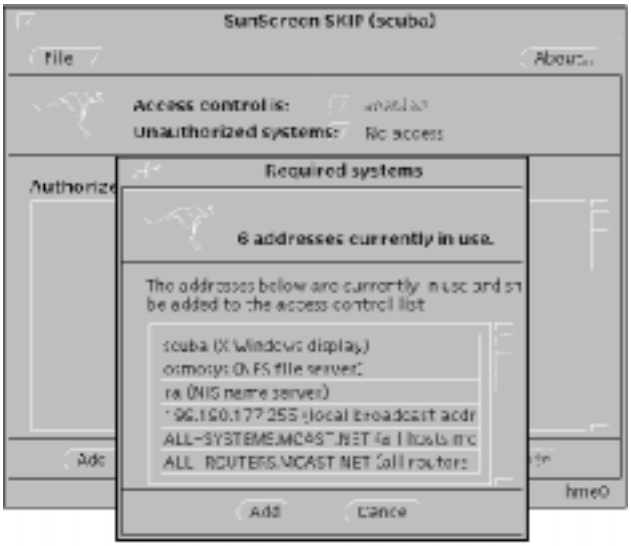

Figure 3–11 Enabling SKIP

Understanding the Symbols in the Authorized Systems List

The authorized systems area lists all the hosts that are allowed access. The excluded systems area shows all those known hosts that 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 Version 1 (Security = SKIP version 1).
- A box with a question mark "?" 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 that it is using either SKIP or SKIP Version 1 as the security method.

Iconify *SunScreen SKIP*

Once you have enabled *SunScreen SKIP*, it is no longer necessary to keep the window open. At this time, you may wish to iconify the main window. The `skiptool` icon (Figure 3–12) shows SKIP's status. If you have set unauthorized systems to *No Access*, you can quit `skiptool`.

Figure 3–12 SKIP Icon Showing Both the Enabled and Disabled States

If you quit the application, SKIP stays in whatever mode it was last in (enabled or disabled).

Unauthorized Systems automatically changes to No Access, since there is no longer any way to notify you if an unauthorized system attempts to gain access.

Verifying 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. `skiplog` will log this event.

To verify that *SunScreen 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 *SunScreen SKIP* enabled and be using the same key and traffic encryption algorithms as your system.

   If you have the remote site's certificate, you can immediately start sending encrypted IP. Otherwise, SKIP will need to fetch the remote machine's certificate. By default, this is done by asking the remote site for its 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 when you ping.

Note - The initial ping can fail because the key manager's computation may exceed the time-out value of some of the IP protocols, such as ping.

---

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

   If encryption is not taking place between your system and a system on your authorized 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 the screen. Try to restart the key manager by using the `skipd_restart` command.
   - Verify that SKIP is installed, configured, enabled, and has the certificate of the remote system.
   - Verify the key ID of the remote system in the log file `/var/log/skipd.log` to see if the key manager has set the key ID to what you think it should be. If it is not the correct key ID, get certificates for the correct key ID.
   - Verify that both machines have the same key encryption, traffic encryption, and authentication algorithms. You can check which ACL entry will be used when communicating with a remote host by using `skiphost <hostname/IP address>` command. This command will check default entries, as well as network entries.
   - Certificate Discovery works by sending UDP requests to port 1640 of the server. If you are connecting through a firewall, check with your system administrator that UDP messages are allowed to pass on port 1640. These ports are required for the certificate discovery protocol (CDP). As a workaround, you can manually distribute keys. Also, make sure that the SKIP protocol 57 (decimal number) and the SKIP Version 1 protocol 79 (decimal number) are permitted to pass through the firewall.
   - Some routers also filter packets. Check on the router and its configuration.
   - 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 that machine clocks be synchronized within one hour. Make sure they are synchronized. Messages in `/var/log/skipd.log` will indicate this situation. You may use the UNIX command `rdate` (1M) to synchronize the clocks.

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

Viewing *SunScreen SKIP* Statistics

SunScreen SKIP provides two methods of viewing statistics: `skiptool` and `skipstat`. `skiptool` is the GUI. `skipstat` is the command-line interface for viewing SKIP statistics and is discussed in Chapter 4. The method you choose is a matter of personal preference since both interfaces provide the same data. The GUI display has a yellow label with the word “UPDATED” in front of fields whose values have changed since the last “sampling.” This feature is not available through `skipstat`.

The following statistics are available in *SunScreen SKIP*:

- Network Interface Statistics
- SKIP Header Statistics
- Key Statistics
- Encryption Statistics (for Versions 1 and 2)
- Authentication Statistics

The Statistics Window

You can view the Network Interface, SKIP Header, Key, Encryption (Versions 1 and 2), and Authentication statistics in real-time by selecting SKIP Statistics from the File menu (File —> SKIP Statistics) on the `skiptool` main window (Figure 3–13).

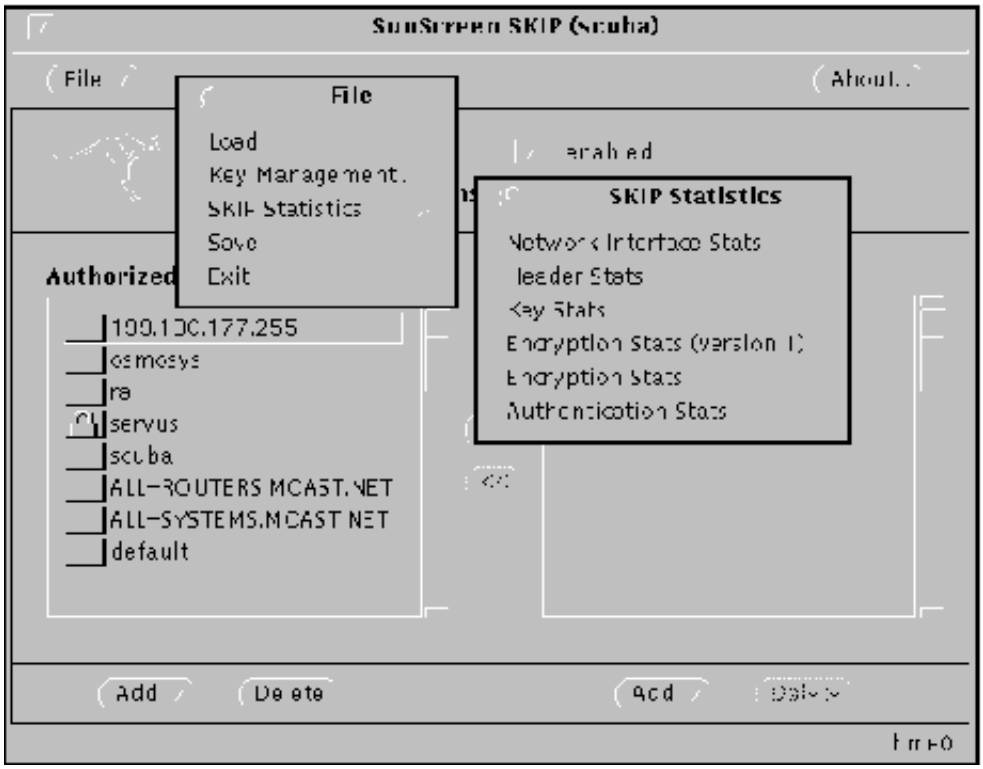

Figure 3–13 Bringing Up a Statistics Window

Each of the statistics available for *SunScreen SKIP* is described on the following pages. Sample data with field descriptions illustrate the information available for monitoring *SunScreen SKIP*'s performance. The fields on the statistics screens are updated approximately every 3 seconds. A status change is indicated by a yellow label with the word "UPDATED" next to the fieldname.

SKIP Statistics

SKIP Interface StatisticsSelecting File —> SKIP Statistics —> Network Interface Stats displays the SKIP Interface Statistics window (Figure 3–14).

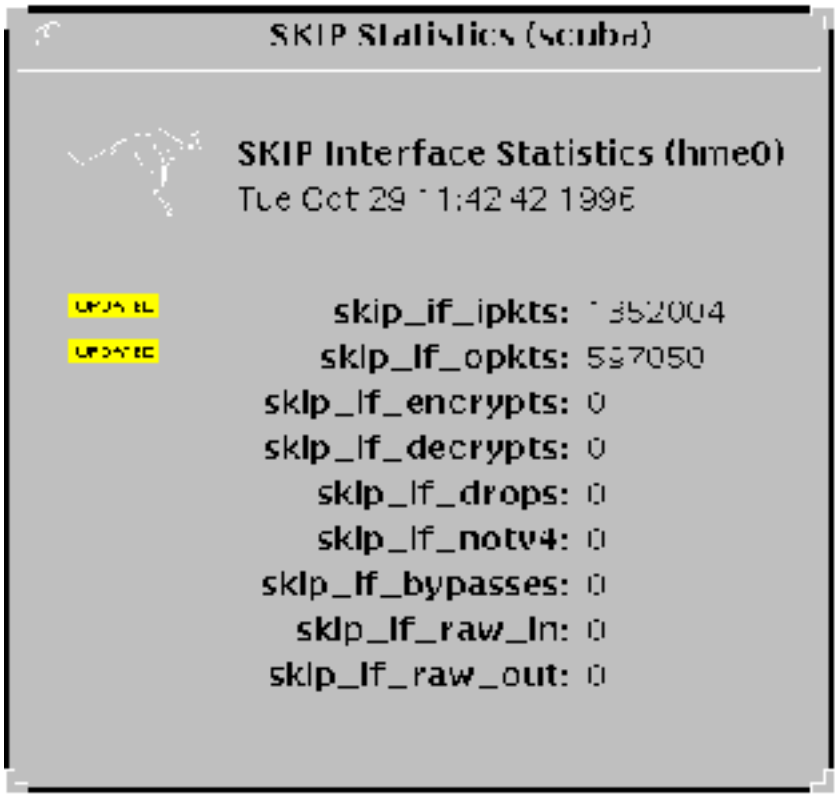

Figure 3–14 SKIP Interface Statistics Window

A brief description of each field is given below:

`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 Header Statistics window (Figure 3–15). In the field descriptions below, V1 refers to SKIP Version 1.

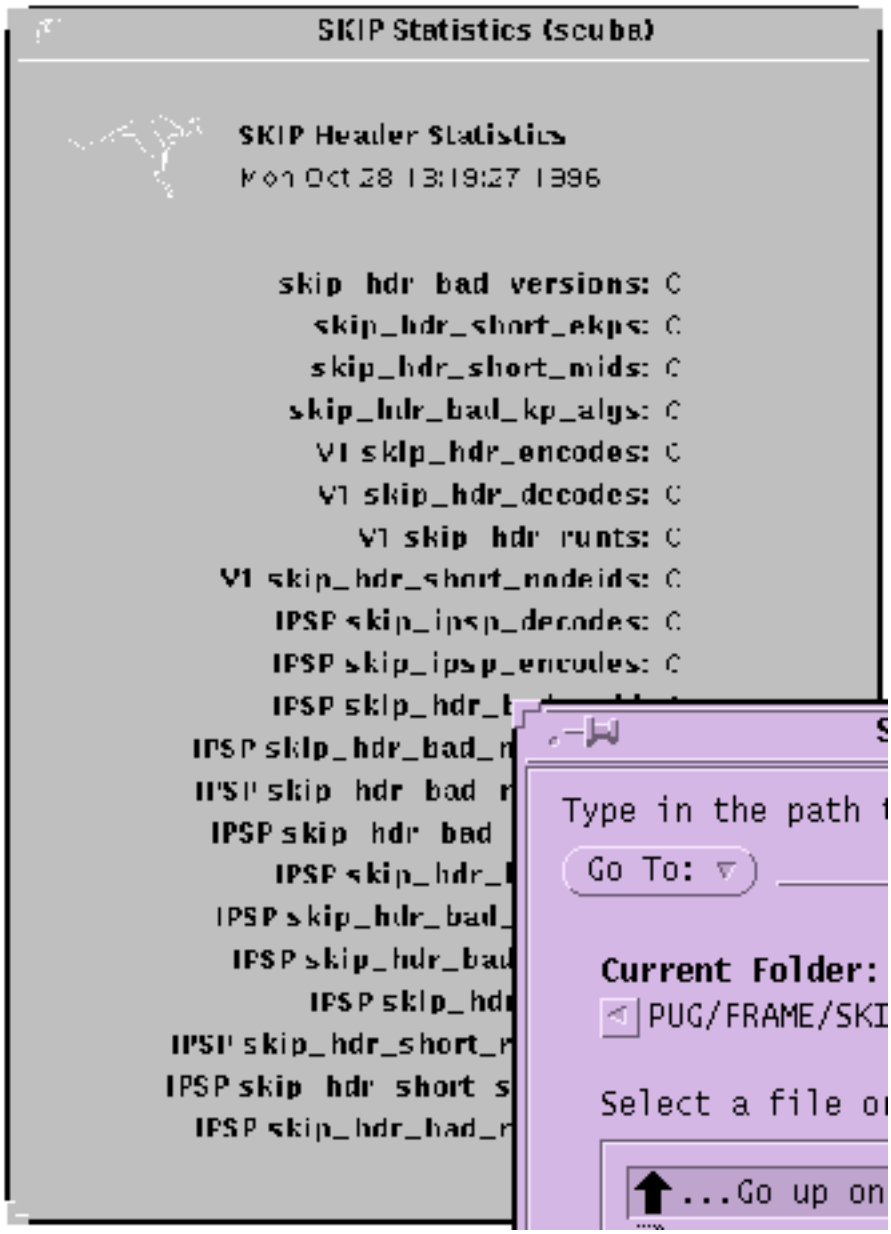

Figure 3–15 SKIP Header Statistics Window

A brief description of each field in SKIP Header Statistics window is given below:

`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 key ID.
`IPSP skip_ipsp_decodes`	The number of SKIP headers decoded.
`IPSP skip_ipsp_encodes`	The number of SKIP headers encoded.
`IPSP skip_hdr_bad_nsid`	The number of headers with a bad SKIP 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 MAC size.
`IPSP skip_hdr_bad_mac_val`	The number of headers with an authentication error in the MAC value.
`IPSP skip_hdr_bad_next`	The number of headers with a bad SKIP next protocol field.
`IPSP skip_hdr_bad_esp_spi`	The number of headers with a bad SKIP SPI field.
`IPSP skip_hdr_bad_ah_spi_`	The number of bad AH/SPI headers (manual keying).
`IPSP skip_hdr_bad_iv`	The number of headers with a bad SKIP initialization vector.

`IPSP skip_hdr_short_r_mkeyid`	The number of headers with a short SKIP receiver key ID.
`IPSP skip_hdr_short_s_mkeyid`	The number of headers with a short SKIP sender key ID.
`IPSP skip_hdr_bad_r_mkeyid`	The number of headers with a bad SKIP receiver key ID.

SKIP Key Statistics

Selecting File —> SKIP Statistics —> Key Stats displays the Key Statistics window (Figure 3–16).

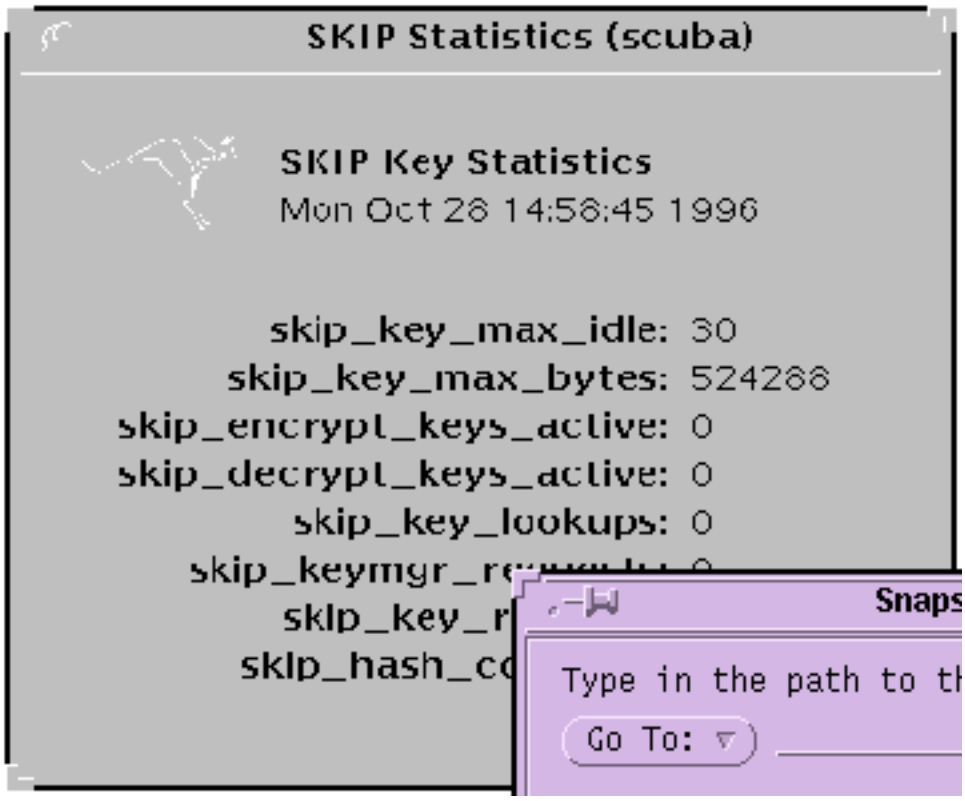

Figure 3–16 SKIP Key Statistics Window

A brief description of each field on the Key Statistics window is given below:

`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 (Version 1) Algorithm Statistics

Selecting File —> SKIP Statistics —> Encryption Stats (Version 1) displays the Algorithm Statistics window for SKIP Version 1 (Figure 3–17).

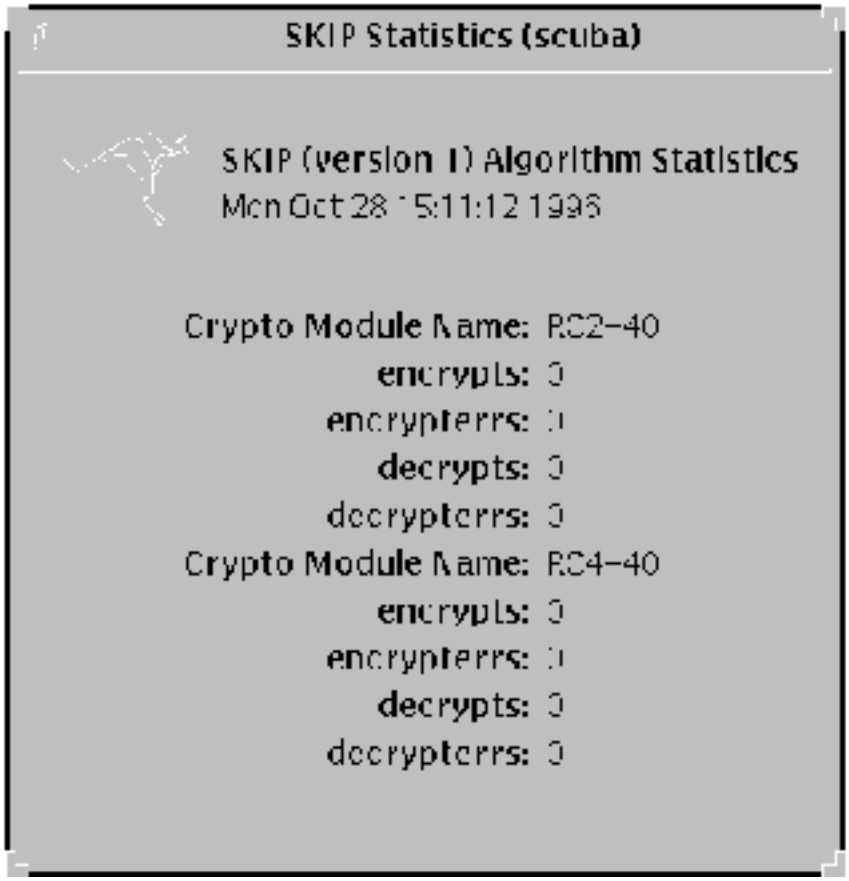

Figure 3–17 Encryption Statistics Window—SKIP Version 1

SKIP Algorithm Statistics

Selecting File —> SKIP Statistics —> Encryption Stats displays the Algorithm Statistics window shown in Figure 3–18.

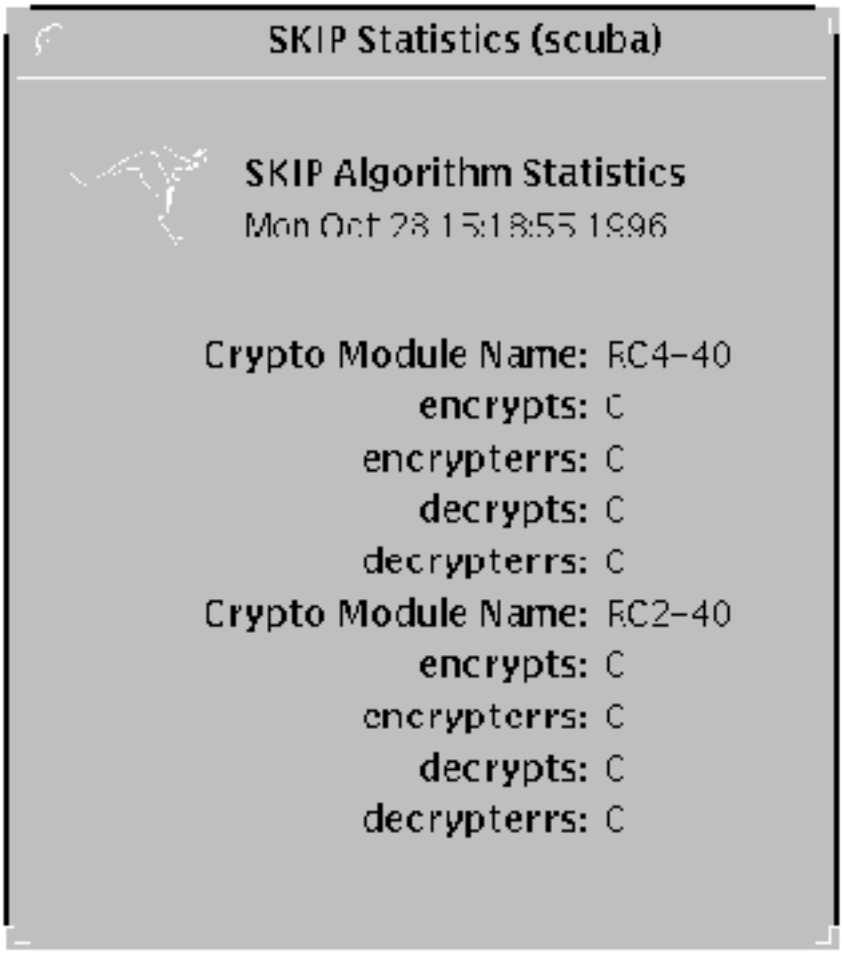

Figure 3–18 Encryption Statistics Window

One set of statistics is displayed for each different traffic and key encryption module. A brief description of each field is give below:

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 Statistics

Selecting File —> SKIP Statistics —> Authentication Stats displays the Authentication Statistics window (Figure 3–19), which provides information on MACs (Message Authentication Code).

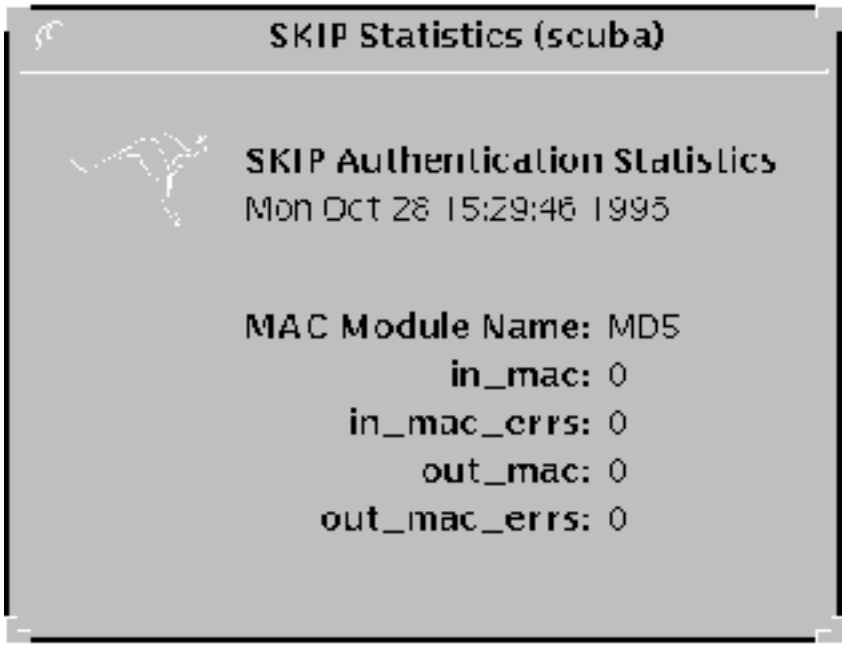

Figure 3–19 Authentication Statistics Window

A brief description of each field on the Authentication Stats window is given below:

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.

Key Management with `skiptool`

The Key Management Parameters window (Figure 3–20) is displayed by selecting File —> Key Management. 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 per encrypt key.

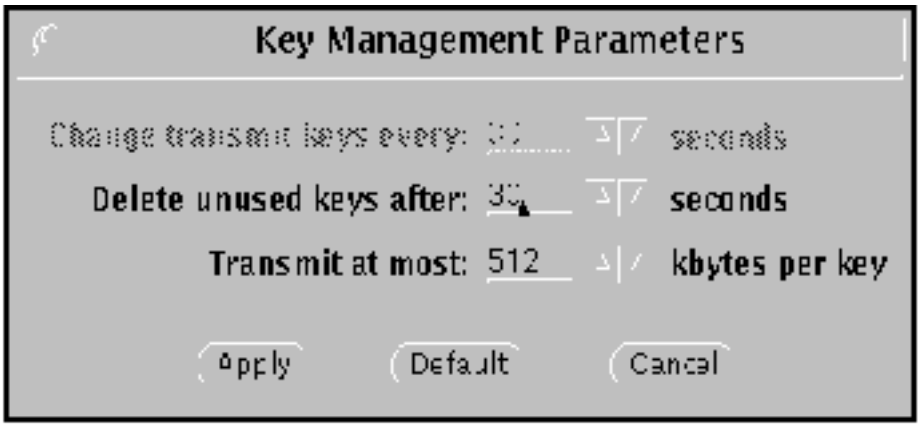

Figure 3–20 Key Management Parameters Window

The Key Management Parameters window has four major components.

Change transmit keys every:

The system uses the delete unused key parameter to decide when to change active encrypt keys.

Delete unused keys after:

This button sets the number of seconds an unused traffic 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:

This button sets 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 can 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 enable 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.

CHAPTER 4

Managing SunScreen SKIP through the Command-Line Interface

This chapter describes how to use the command-line interface.

To use the command-line interface, you must be logged in as root.

SKIP Command-Line Interface

The *SunScreen SKIP* command-line interface commands follow, including a brief description of what they do. Many of these commands duplicate what can also be done using the GUI, while others are enabling commands for other commands. For a more complete discussion of the command-line interface, refer to the man pages for *SunScreen SKIP*.

`print_cert`	Prints a certificate to standard output.
`certreq`	Requests and retrieves a certificate from a key server or other host.
`install_skip_keys`	Installs a private key and certificate received from a key server or from the SunCA.
`skipca`	Manages the SKIP Certificate Authorities Database. It is used to add, delete, or list CAs.

`skipd`	It is not a user command, but a system process not normally start by the user.The skipd daemon is started at system boot, and restarted when necessary with the skipd_restart command. Only one key manager may be running at a time. The key manager must be started by root.
`skipd_restart`	Kills the existing running SKIP key-management daemon (`skipd`) and starts a new one. It is used after any changes in key configurations to make them permanent.
`skipdb`	Administers the SKIP database of certificates. SKIP stores the long-term certificates in the database so that the key manager can have access to them.
`skiphost`	Lists, adds, or deletes host, network, or nomadic (mobile) system information from SKIP's ACL. `skiphost` can be also used to enable or disable SKIP.
`skipif`	Adds or removes SKIP from the network interfaces. It is also used to save ACL status.
`skiplocal`	Used to manage the SKIP local keys for the workstation. It is used to add, delete, or print local keys.
`skiplog`	Displays security events for the local system.
`skipstat`	Displays statistical information about the use of SKIP on the local system.

Using the Command-Line Interface

print_cert: Printing a Certificate to Standard Output

`print_cert` prints the contents of the certificate found in the certificate file specified. You can specify the type of certificate—the types of certificates supported are X.509 and UDH. The default is X.509.

certreq: Retrieving a Certificate From a Key Server

`certreq` is a maintenance command. It requests and retrieves a certificate from a key server or other host. You must specify the key ID and key server. This command is a debugging tool and is not meant for general use. The interface is cryptic and there is no way to specify a host name or IP address instead of the key ID, even if the key ID is identical to the IP address.

install_skip_keys: Installing Keys and Certificates From a Certificate Authority

`install_skip_keys` installs keys received from a key server (default) or from the SunCA (if `-icg` is specified). If you are installing a key package from a key server, the filename specifies the name of that package. The key file is a pretty good privacy (PGP) or an encoded file containing: a Diffie-Hellman private key, a Diffie-Hellman signed public key, the common Diffie-Hellman parameters used by the certificate issuer, the certificate issuer's signed public key, and a MD5 checksum of the other four files. The filename is an encoded `tar` file usually received from a key server or other certificate issuer.

If you are installing a SunCA certificate, the filename is the name of the directory that contains the files. This is usually a diskette, so the path will often be similar to

```
/floppy/floppy0
```

`install_skip_keys` verifies the MD5 checksums of the individual files with the checksum file. If they match, the files are copied into place.

The key manager must be restarted (see `skipd_restart`) in order for it to recognize the new keys.

Currently, the name of the certificate is hard coded into the code. Certificates are expected to come from the SKIP experimental Zero Assurance Certificate Issuer or the SunCA. Even if they do not, the certificate will have to be called `ZeroAssurance_Cert`. This release does not support multiple certificate issuers.

skipca: Setting Up Trusted CAs

Certificates are the digital documents that 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 CA as trusted and to manage that database. `skipca` options are add, extract, init, list, delete, create, and revoke CA certificates.

You must either reboot the system or restart the key manager with `skipd_restart` before any changes will take effect.

This command has broad security implications. By designating a CA, you are trusting the identity of all certificates signed by that CA. Since root CA certificates are self-signed, there is no automated way to verify that a CA certificate actually comes from that CA. Before adding a CA certificate, you *must* be absolutely certain that the certificate is valid. Validity may be checked by having the CA publish the hash of its certificate publicly and comparing that hash with the hash obtained from the certificate.

skipdb: Managing Keys and Certificates

`skipdb` is used to manage certificates. Long-term certificates are stored in a database for access by the key manager. The `skipdb` command allows the manual administration of the certificate database.

X.509 certificates without proper signatures will not be added to the `skipdb` database. The CA's certificate must be added to the CA certificate database using the `skipca` command before adding certificates signed by that CA to the `skipdb` database.

Unsigned public keys will be added with the appropriate hash of the contents as the name.

skipd_restart: Activating the Changes

`skipd_restart` reinitializes the SKIP key manager in order for the changes that you made though `skipca`, `skipdb`, and `skiplocal` to take effect.

skiphost: Setting Up the ACL

The functionality of `skiphost` is the same as the `skiptool` GUI.

Use `skiphost` to list, add, and delete host, network, or nomadic (mobile) systems from the ACL, as well as to enable and disable SKIP. Without arguments, it lists the state of the SKIP interface and authorized or unauthorized hosts, networks, and nomadic systems for the default interface.

The ACL allows the user to configure which remote systems can obtain access to the local host and the type of access granted. Access control is usually based on the IP address of the remote host or by the remote system's key ID.

Remote systems can be specified either as individual hosts, networks, or nomadic systems.

Hosts are specified by their host name or IP address.

Networks of subnetworks are specified by a network address plus a mask similar to that used in subnetworking.

Nomadic systems can be specified in SKIP and in SKIP Version 1. They are specified by a key identifier (that is, any IP address with the key ID "*x*").

The order of processing ACL entries is as follows. A search is made for an ACL entry specifying the remote host. If one exists, it will be used.

If no entry containing the IP address can be found, then a search is made for a nomadic ACL entry containing the sender's key ID in the SKIP protocol header. If one is found and the packet is correctly authenticated, then the sender's IP address is stored for future reference.

If no corresponding ACL entry can be found for a remote system, the default is used. The default may be configured to allow access or to deny access. This method is similar to the method used by the IP when it is deciding how to route a packet to a destination (that is, host routes take precedence over network routes, and, in the absence of anything better, the default route is used).

When applying access control, the system treats the lists of authorized and excluded systems as a global list and always selects the best match.

A default entry can be specified to indicate all other hosts not specifically covered by other access-control entries.

Note - Before you enable 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.

In order to set up SKIP, `skiphost` must be run multiple times: one time for each host being set up in the ACL, then one final time to enable SKIP.

See "Enabling SKIP" on page 46 for information on enabling SKIP.

See the man pages for more detail.

skipif: Managing Network Interfaces

`skipif` is used to add SKIP to or delete SKIP from network interfaces. `skipif` is also used to save SKIP's ACL for a given network interface so that it is permanent across system reboots. In addition, `skipif` is used to list the network interfaces present in the system and optionally to print the current access control configuration for each network interface.

SKIP's ACL for each network interface is stored as a text file (as a series of `skiphost` commands to be executed during SKIP start-up). SKIP's ACL files are under the `/etc/opt/SUNWicg/skip` directory and the ACL file name for a given

interface is `acl.<interface name>` (for example, `acl.le0`, `acl.hme0`, and `acl.qe1`). If an incorrect or incomplete ACL prevents the system from operating, it may be necessary to modify the file manually or remove the appropriate file. Some non-LAN interfaces (PPP, for example) will not be configured at boot time even if an ACL exists for these interfaces. It is the responsibility of the user in the interface configuration procedure to use the SKIP configuration file for this interface.

skipif notifies the user if it is necessary to reboot the system so that any changes will take effect.

See the man pages for more detail.

skiplocal: Managing Local Identities

`skiplocal` is the utility for managing SKIP identities on a workstation. A host may wish to have multiple identities if it must interoperate with other hosts that have incompatible Diffie-Hellman parameters (for instance, a U.S. host may wish to communicate with other U.S. hosts with a 1024–bit modulus, but must also communicate with a host outside the U.S. that is limited to a 512–bit modulus). Each local identity has a secret, a certificate, and a unique name. The name is extracted from the certificate and used as a local identity. `skiplocal` is the primary tool for administering local identities. With `skiplocal`, you can create, delete, and list local identities based on the command option specified.

You can use `skiplocal` to set or remove a passphrase that is used to encrypt SKIP locally stored secrets.

Caution - Beware of electronically transmitting access control commands to remote hosts. For complete security, the receiving system must verify the remote key ID out of band.

Note - After adding a local ID, the key manager must be restarted using `skipd_restart`, in order for any changes to take effect.

Caution - `skiplocal export` does not work well for communicating with multiple keys. Since the local system does not know which key on the remote system should be used, incorrect bindings can occur. Therefore, it is recommended that the `skiplocal export` command be used carefully.

See the man pages for more detail.

skiplog: Viewing Security Events

`skiplog` displays security events for the local system. It displays the types of events presented below. In all cases, the date and time of the event, as well as the IP address information, are logged.

Unknown Source—A packet was received from a system that is not currently in the ACL. The packet is dropped.

Unknown Destination—The local system sent a packet to a system that is not currently in the ACL. The packet is dropped.

Excluded Source—A packet was received from a system explicitly excluded by the ACL. The packet is dropped.

Excluded Destination—The local system sent a packet to a system that was explicitly excluded by the ACL. The packet is dropped.

Bad Parameters—A packet was received that contained security parameters that were incompatible with the ACL entry.

Note - Only one instance of `skiplog` may be active for a given network interface. `skiptool`'s "Ask for Confirmation" and "Add Automatically" options may not be active at the same time as `skiplog` for a given network interface.

See the man pages for more detail.

skipstat: Viewing *SunScreen SKIP* Statistics

`skipstat` is the command-line interface for viewing SKIP statistics. Because `skipstat` is a command-line interface, the information that is displayed does not update on screen with the results of the latest sampling as `skiptool` does.

The following statistics are available in *SunScreen SKIP*:

- SKIP Network Interface Statistics
- SKIP Header Statistics
- SKIP Key Statistics
- SKIP Encryption Statistics (for Versions 1 and 2)
- SKIP Authentication Statistics

The following is a breakdown of `skipstat` output for each of the main options:

SKIP Network Interface Statistics

Command: `skipstat -I<interface>`

SKIP interface (le0) statistics:

`skip_if_ipkts:`	number of packets received by interface
`skip_if_opkts:`	number of packets sent by interface
`skip_if_encrypts:`	number of packets encrypted
`skip_if_decrypts:`	number of packets decrypted
`skip_if_drops:`	number of packets dropped
`skip_if_notv4:`	number of non-IPV4 packets
`skip_if_bypasses:`	number of certificate packets
`skip_if_raw_in:`	number of raw packets received
`skip_if_raw_out:`	number of raw packets sent

SKIP Header Statistics:

Command: `skipstat -h`

Note - In the description below, V1 refers to SKIP's *SunScreen SPF-100* and *SPF-100G* compatibility mode (based on an earlier version of the SKIP protocol).

`skip_hdr_encodes:`	number of SKIP V1 headers encoded
`skip_hdr_decodes:`	number of SKIP V1 headers decoded
`skip_ipsp_encodes:`	number of SKIP V2 headers encoded
`skip_ipsp_decodes:`	number of SKIP V2 headers decoded

Header decode error statistics:

`skip_hdr_bad_versions:`	invalid protocol version
`skip_hdr_short_ekps:`	short eKp fields
`skip_hdr_short_mids:`	short MID fields
`skip_hdr_bad_kp_algs:`	unknown crypto algorithms
`skip_hdr_runts:`	short SKIP V1 packets
`skip_hdr_short_nodeids:`	short SKIP V1 node ids
`skip_hdr_bad_nsid:`	bad V2 namespace ID
`skip_hdr_bad_mac_alg:`	bad MAC algorithm
`skip_hdr_bad_mac_size:`	bad MAC data size
`skip_hdr_bad_mac_val:`	bad MAC value
`skip_hdr_bad_next:`	bad V2 next protocol field
`skip_hdr_bad_esp_spi:`	bad V2 encryption SPI field
`skip_hdr_bad_ah_spi:`	bad V2 MAC SPI field
`skip_hdr_bad_iv:`	bad V2 initialization vector
`skip_hdr_short_r_mkeyid:`	short V2 receiver key ID
`skip_hdr_short_s_mkeyid:`	short V2 sender key ID
`skip_hdr_bad_r_mkeyid:`	bad V2 receiver key ID

Key Statistics

Command: `skipstat -k`

`skip_key_max_idle:`	unused key time-out
`skip_key_max_bytes:`	maximum bytes to encrypt
`skip_encrypt_keys_active:`	encrypt keys in cache
`skip_decrypt_keys_active:`	decrypt keys in cache
`skip_key_lookups:`	key cache lookups
`skip_keymgr_requests:`	key cache misses
`skip_key_reclaims:`	cache entries reclaimed
`skip_hash_collisions:`	hash table collisions

SKIP Encryption Statistics:

Command: `skipstat -c` (requires the version of SKIP as part of the argument

Cryptographic algorithm stats (SKIP Version 1)

Crypto Module Name: DES-CBC

`encrypts:`	number of successful encryptions
`encrypterrs:`	number of failed decryptions
`decrypts:`	number of successful decryptions
`decrypterrs:`	number of failed decryptions

Cryptographic algorithm stats (SKIP)

Crypto Module Name: DES-EDE-K3-CBC

`encrypts:`	number of successful encryptions
`encrypterrs:`	number of failed decryptions
`decrypts:`	number of successful decryptions
`decrypterrs:`	number of failed decryptions

SKIP Authentication Statistics

Command: `skipstat -m`

MAC algorithm statistics (SKIP)

MAC Module Name: MD5

`in_mac:`	number of received MAC calculation
`in_mac_errs:`	number of failed received MAC calculation
`out_mac:`	number of successful sent MAC calculation
`out_mac_errs:`	number of failed sent MAC calculation

For more information using `skipstat`, refer to the man pages for *SunScreen SKIP*.

CHAPTER 5

Usage Examples

This chapter describes sample topologies for systems and networks using *SunScreen SKIP*.

All topologies require that

1. Keys be generated or installed.
2. Key IDs (and certificates) be exchanged between hosts.
3. On both hosts, ACL entries be configured with matching algorithms, key IDs, and protocol versions.
4. SKIP be enabled.

The topologies explained here are the following:

1. Setting up an encrypted connection between two hosts.
2. Setting up an encrypted connection between a host and a SunScreen SPF-100.
3. Setting up an encrypted connection from a host to an encrypting gateway, SunScreen EFS, or SunScreen SPF-200.
4. Setting up a host as a nomadic encrypting gateway.
5. Using tunnel addresses.

Setting Up an Encrypted Connection Between Two or More Hosts

Figure 5–1 depicts the configuration in which a host has an encrypted connection to another host. This is the simplest case.

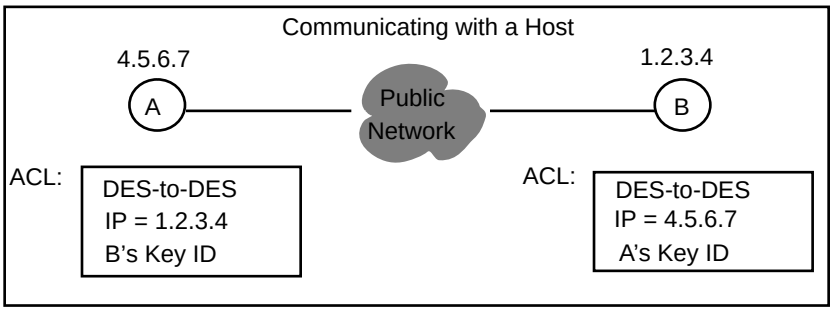

Figure 5–1 Communicating with a Host

Figure 5–1 is an example of host-to-host communication using UDH keys and SKIP.

All the hosts must:

- Share the same key types, such as UDH, SunCA X.509, or the like, and of the same encryption strength. If X.509 certificates and keys are used, the certificates and keys for both hosts must be from the same vendor.
- Exchange certificates.
- Have the same algorithm to use that includes authentication, key encryption, and traffic encryption.
- Enable SKIP.

A machine must also have a local identity. Hosts can have many identities, but the user must choose one with which to communicate to the other host. This local identity consists of the local key type (NSID) and the local key name.

The hosts must exchange key IDs. The safest method of exchanging UDH key IDs is to have each user run `skiptool`, then call each other on the telephone and type the other's UDH key ID in the Remote Key ID field in the Add window.

UDH key IDs can be exchanged and added to the ACL of each using the `skiplocal export` command. In this case, both system administrators should telephone one another and confirm the key ID.

The address of each host with which a host wants to communicate must be in its ACL.

Setting Up an Encrypted Connection Between a Host and a *SunScreen SPF-100*

Figure 5–2 depicts the configuration of an encrypted connection between a host and a SunScreen SPF-100.

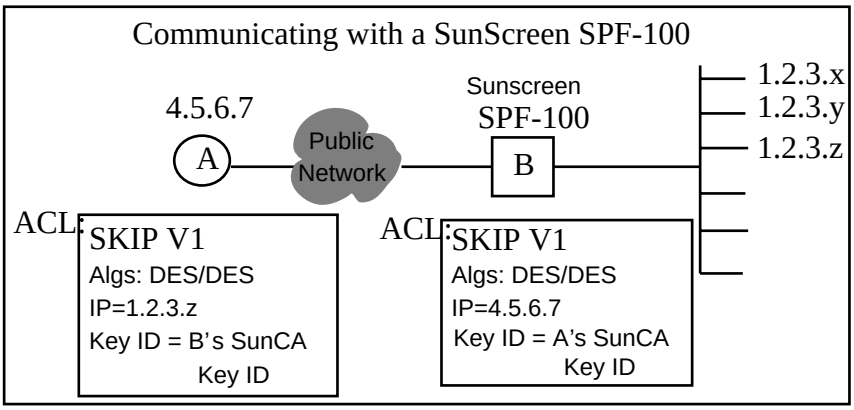

Figure 5–2 Communicating with a SunScreen SPF-100

In this case, both the host and the SunScreen SPF-100 must

1. Install a SunCA X.509 key of the same encryption strength.
2. Manually exchange certificates.
3. Use SKIP protocol Version 1.
4. Have an IP address or remote name.
5. Use the same algorithm that includes authentication, key encryption, and traffic encryption.
6. Enable SKIP.

A machine must also have a local identity. Hosts can have many identities, but the user must choose one with which to communicate to the remote host. This local identity consists of the local key type and the local key name.

X.509 certificates and keys must be used when speaking to a SunScreen SPF-100. The physical diskettes containing the public keys must be physically exchanged.

The only method of exchanging key IDs is to have each user run `skiptool`, then call each other on the telephone and type the other's key ID in the Remote Key ID field in the Add window.

The ACL for both the host and the SunScreen SPF-100 must be configured with each other's address. The host must also include the addresses of any networks and hosts attached to the SunScreen SPF-100 in its ACL. The SunScreen SPF-100 does not really use ACL: It uses packet filtering rules. These rule must be set to "match" the ACL on the host running *SunScreen SKIP*.

Setting Up an Encrypted Connection From a Host to an Encrypting Gateway, or *SunScreen EFS*

Figure 5–3 depicts the configuration in which a host is communicating with an encrypting gateway.

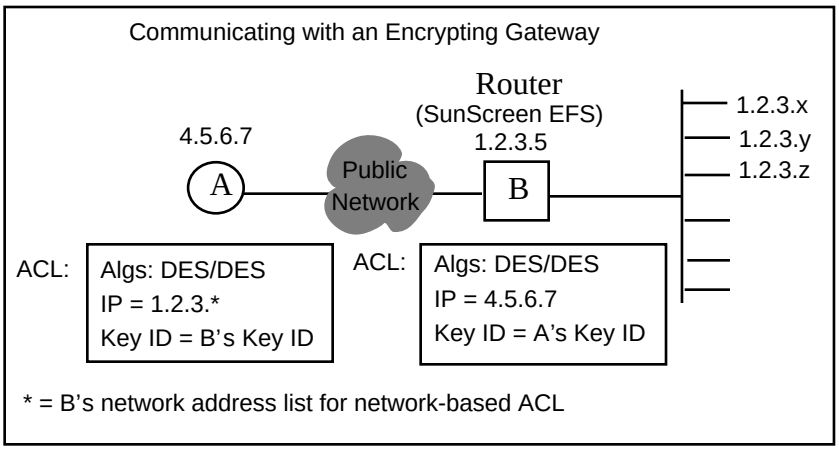

Figure 5–3 Communicating with an Encrypting Gateway

In this case, both the host and the encrypting gateway, whether it be a gateway, or a SunScreen EFS must

1. Have the same key type, such as UDH, SunCA X.509, or the like, and of the same encryption strength. If X.509 certificates and keys are used, the certificates and keys for both hosts must be from the same vendor.
2. Exchange names or certificates.
3. Use the same version of the SKIP protocol.
4. Have an IP address or remote name.
5. Use the same algorithm that includes authentication, key encryption, and traffic encryption.
6. Enable SKIP.

A machine must also have a local identity. Hosts can have many identities, but the user must choose one with which to communicate to the remote host. This local identity consists of the local key type and the local key name.

Both machines install or generate their keys and exchange namespace/key ID information. This should be done over the telephone or some other media.

The user should type the encrypting gateway's information into the Add System box of skiptool. The user should also set the Tunnel Address field of this box to be the IP address of the intermediate system. This enables certificate discovery to ask the correct host for its certificate.

For example: You are contacting a gateway that has three networks attached to it (networks 199.190.177, 199.190.176, and 199.190.176) and these networks are to remain hidden. It also has a local host attached to it. The ACL in the host should be set up as in Table 5–1.

TABLE 5–1

Host	Algorithm	Tunnel Address	Remote Key
199.190.177.*	V2 DES/DES	Gateway	Gateway's
199.190.176.*	V2 DES/DES	Gateway	Gateway's
199.190.176.*	V2 DES/DES	Gateway	Gateway's
Local host	V2 DES/DES	Gateway	Gateway's
Default	V2 DES/DES	Gateway	Gateway's

The user can configure a default so that everything is sent to the gateway where it will be decrypted and sent to the proper recipient in the clear. The recipients of the packets will not be aware of any encryption. The gateway will handle all the encryption and decryption of packets from and to everything behind it.

Setting Up a Nomadic Encrypting Gateway

Figure 5–4 depicts the configuration in which a host is communicating with an encrypting gateway that receives packets from an encrypting nomadic system.

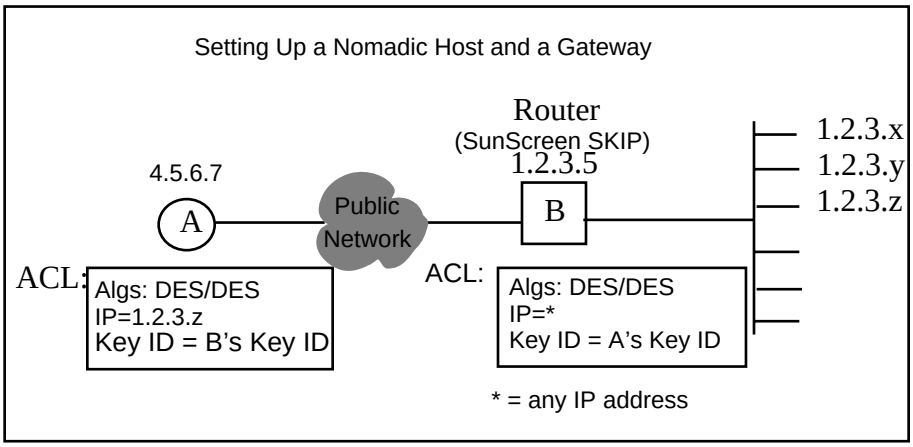

Figure 5–4 Setting Up a Nomadic Host and a Gateway

A nomadic encrypting gateway is an encrypting gateway that encrypts and decrypts packets from hosts whose IP address is not known ahead of time (for instance, hosts who receive the IP address dynamically). This is the same as configuring a host-to-host configuration except that the ACL does not have a specific address for the nomadic system. The address in the ACL is * and gets the temporary address from the nomadic system when it contacts the host. The host can only contact the nomadic system when it knows its address. Every time the nomadic system moves and then reconnects with the host, it will have a new address.

Using Tunnel Addresses

Figure 5–5 depicts the configuration in which a host is communicating with a hidden system through a tunnel address to an encrypting gateway. The hidden system also uses a tunnel address from the encrypting gateway to the host.

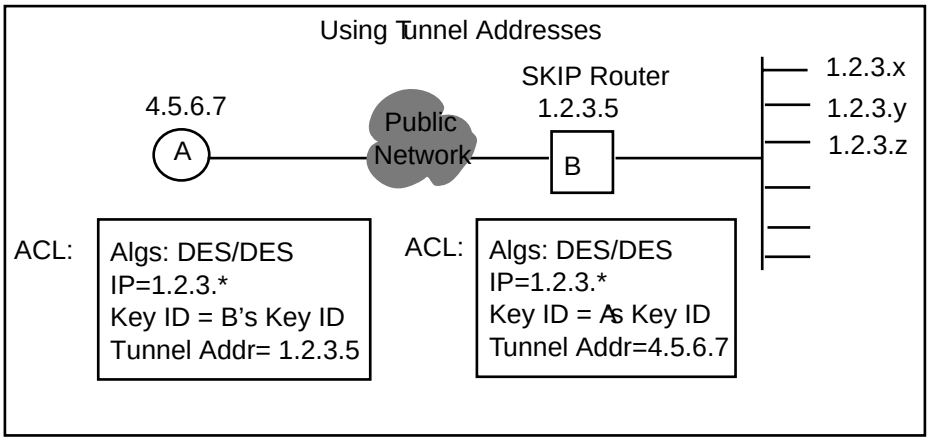

Figure 5–5 Using Tunnel Addresses

In tunneling, the packets are sent from the host to the gateway. The packets are encrypted such that the gateway decrypts them and sends them to their final destination in the clear.

When setting up tunneling, the user must add the address for the gateway into the host's ACL because there is no way that the host can discover the gateway's certificate.

APPENDIX **A**

Quick-Start Guide

This appendix is a quick-start guide for *SunScreen SKIP*. It covers installing the SKIP binaries or adding the packages with `pkgadd`, and setting up IP-level encryption between two hosts. These instructions assume that only one network interface is active on each machine.

For complete documentation, refer to the *SunScreen SKIP* documentation and the SKIP man pages.

Installing SKIP Binaries

1. **Mount the CD-ROM and type**

```
volcheck
```

Note - If you are not using `vold` on your system, type

```
# mount -F hsfs -oro /dev/dsk/c0t6d0s0/mnt
```

The device name or the mount point or both depends on your local system configuration.

2. **Go to the directory on the CD-ROM for your OS**

   Solaris for the SPARC Platform:

```
cd /cdrom/cdrom0/sparc
```

Solaris for the Intel Platform:

```
cd /cdrom/cdrom0/x86
```

Note - If you have mounted the CD-ROM manually, replace `/cdrom/cdrom0` with `/mnt`.

---

3. **Use the standard Solaris operating system** `pkgadd` **command to add all packages**

```
pkgadd  -d ‘pwd‘
```

4. **Add** `/opt/SUNWicg/bin` **to your PATH variable**

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

5. **Generate a secret and a public certificate locally by issuing the command**

```
skiplocal keygen
```

6. **Add SKIP to your network interface by issuing the command**

```
skipif -a
```

7. **Reboot the machine.**

8. **Enable SKIP and configure IP encryption with one other host**

```
PATH=$PATH:/opt/SUNWicg/bin; export PATH
skiphost -a default default IP traffic is unencrypted
skiplocal export prints the skiphost command
                 others need to run to talk to us
skiplocal export | mail Friend@remote.host
```

`Friend@remote.host` should issue these commands as well. Once the corresponding mail is received, verify out-of-band (for example, over the telephone) that the received mail matches the mail that was sent. Then execute the received `skiphost` command.

```
skiphost -o on enable SKIP
```

▼ Is It Working?

At this point, encryption should be enabled with the remote host. Traffic will be exchanged with all other hosts in the clear.

1. `ping` **the other host to make sure everything is working**

```
ping host
```

2. **View the key manager log file to see if the certificate exchange and the shared-secret computation succeeded**

```
tail /var/log/skip.log
```

3. **If you have** `snoop`**,** `tcpdump`**,** `etherfind`**, or some other packet dumping utility, you can verify that encrypted packets are using protocol 57.**

▼ Examining the Local SKIP Configuration

`skiphost`	list the SKIP access control entries
`skiplocal list`	list the set of local identities
`skipdb list`	list the certificates in our database
`skipca list`	list the Certificate Authorities we trust

SKIP configuration files are stored in the `/etc/opt/SUNWicg/skip` directory.

APPENDIX **B**

SunScreen SKIP Theory of Operations

An Overview of *SunScreen SKIP*

SunScreen SKIP is Sun Microsystems' implementation of Simple Key-Management for Internet Protocols (SKIP) for use on computers running Solaris, Versions 2.4, 2.5, and 2.5.1 or Solaris for the Intel Platform Edition, Versions 2.4 and 2.5. *SunScreen SKIP* is part of the SunScreen product line, offered by Sun Microsystems.

SKIP is an IP-layer encryption package that provides a system with the ability to encrypt any protocol within the TCP/IP suite efficiently. Once installed, systems running SKIP can encrypt all traffic to any SKIP-enabled product including SunScreen products.

SKIP Is Unique

SKIP is independent of any application and can be used with many applications, such as FTP, Mosaic, and Telnet. SKIP was invented by Ashar Aziz at Sun Microsystems, Inc. SKIP uses the principles of Diffie-Hellman Key Exchange to generate unique keys that only the source and destination nodes can use.

The Engineering Data About SKIP

The following is a series of the technical reports that are available from the Internet Commerce Group of Sun Microsystems, Inc.

1. A. Aziz, T. Markson, and H. Prafullchandra, *Simple Key-Management For Internet Protocols (SKIP)*, ICG Technical Report Series, Internet Commerce Group, Sun Microsystems, Inc., October 1996.
2. A. Aziz, T. Markson, H. Prafullchandra and G. Caronni, *Certificate Discovery Protocol*, ICG Technical Report Series, Internet Commerce Group, Sun Microsystems, Inc., October 1996.
3. A. Aziz, T. Markson, and H. Prafullchandra, *Encoding of an Unsigned Diffie-Hellman Public Value*, ICG Technical Report Series, Internet Commerce Group, Sun Microsystems, Inc., October 1996.
4. A. Aziz, T. Markson, and H. Prafullchandra, *SKIP Extensions for IP Multicast*, ICG Technical Report Series, Internet Commerce Group, Sun Microsystems, Inc., October 1996.
5. A. Aziz, T. Markson, and H. Prafullchandra, *SKIP Algorithm Discovery Protocol*, ICG Technical Report Series, Internet Commerce Group, Sun Microsystems, Inc., October 1996.
6. A. Aziz, T. Markson, and H. Prafullchandra, *X.509 Encoding of Diffie-Hellman Public Values*, ICG Technical Report Series, Internet Commerce Group, Sun Microsystems, Inc., October 1996.
7. A. Aziz, *SKIP Extension for Perfect Forward Secrecy (PFS)*, ICG Technical Report Series, Internet Commerce Group, Sun Microsystems, Inc., October 1996.

How SKIP Has Evolved

Sun Microsystems, Inc. is continuously developing SKIP. The first products to use this technology were the SunScreen SPF-100 and SunScreen SPF-100G, which were developed from the October 1995 draft of SKIP, also known as SKIP, Version 1.

Since the October 1995 draft, SKIP and the other related protocols have evolved so that now a whole set of new features is available. This new protocol is known as Version 2.

To maintain backwards compatibility, products such as *SunScreen SKIP* have a Version 1 mode that enables them to communicate with products like a SunScreen SPF-100 unit, which uses the earlier version.

The tools for configuring and managing *SunScreen SKIP* use the “SKIP Version 1” label to denote SunScreen SPF-100 compatibility and the “SKIP” label for the new definition of the protocol.

SunScreen SKIP Security Services

SunScreen SKIP implements security services through four major components:

1. Bulk Data Crypt for key caching, bulk data encryption, and authentication.

2. Cryptographic Modules that support the most rigorous symmetric key cryptography and authentication methods currently available.
3. Key and Certificate Management tools that provide automatic management of certificates and the generation of random traffic encryption keys.
4. SKIP End System for the system administrator's use in controlling access to corporate resources on the network.

Relating SKIP to Data Encryption Concepts

Once installed, any two (or more) systems running SKIP have the ability to encrypt all traffic between or among them transparently.

SunScreen SKIP Services

SunScreen SKIP provides networks with four security services:

1. Access Control to protect corporate network and data resources from unauthorized use.
2. Encryption and Decryption to ensure the confidentiality of information sent over the network.
3. Authentication to ensure the integrity of the information transferred from one group to another within a the network.
4. Key and Certificate Management to provide efficient, cost-effective administration of the basic building blocks of a security policy.

Access Control List (ACL) Using *SunScreen SKIP*

The ACL feature allows you to limit and control who uses your host systems and applications through your network. Each entity—host, network, or nomadic system, with which you communicate over your network when using SKIP—must be identified and authenticated so that access to your system is controlled. Clear-text hosts are not authenticated. Once communication is established, data can be exchanged in the clear, integrity protected, or encrypted.

SunScreen SKIP can provide mobile remote (nomadic) users with access through the ability to separate an entity from its physical address by means of a key identifier (key ID).

SunScreen SKIP's ACL is based on the requesting system's IP address, if this is fixed and known, or on the key ID, if the SKIP user is nomadic (that is, does not have a fixed address).

When a system tries to connect to a host running *SunScreen SKIP*, the order of processing for the host is as follows:

1. Search for an entry specifying a remote host by IP address. If the entry exists and it meets the established criteria, the host allows traffic from the remote host; otherwise, it continues to the next search action. If the entry exists, but does not meet the established criteria (in case of incoming traffic), the connection (packet) is refused.
2. Search for a network entry that matches the remote entry. If the entry exists and it meets the established criteria, the host allows traffic from the remote host; otherwise, it continues to the next search action.
3. If an entry for a host or network is not found, search for a nomadic ACL entry containing the sender's key identifier in the SKIP protocol header. If the entry is found and the packet is authenticated, the host stores the sender's IP address until it is replaced with a new value. If it exists and meets these criteria, the host allows traffic from the remote host; otherwise, the host continues to the next search action.
4. Finally, if no match can be found, a catchall ACL entry (named "default") will be used if it is present.

Note - These rules may be used to prevent a host or network from obtaining access to the system.

Here is an example of how ACL works. Suppose there are two network nodes A and B that wish to communicate securely using DES-to-DES encryption. The three cases that can occur are

Figure B–1 Example of Case One—Access Control

1. Nodes A and B (Figure B–1) may be on a network where their IP addresses are known and can be identified as specific host entries in their ACLs.

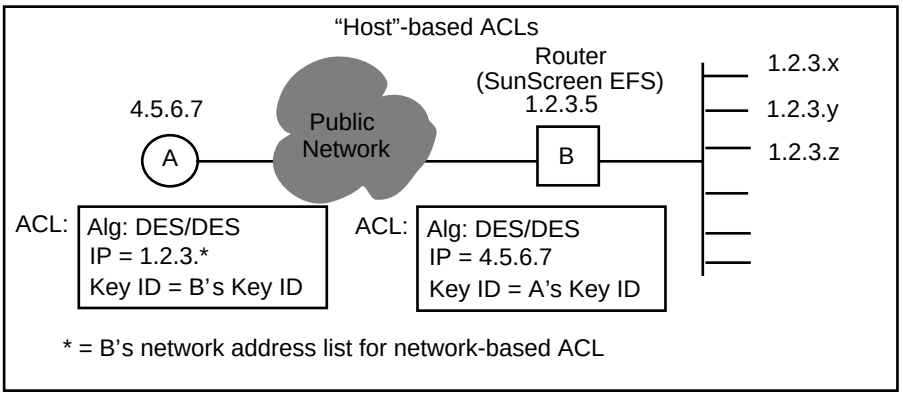

Figure B–2 Example of Case Two—Access Control

2. Node B may be a router (Figure B–2) that has a list of IP addresses, one of which is the host, with which Node A wishes to communicate with as part of a network ACL.

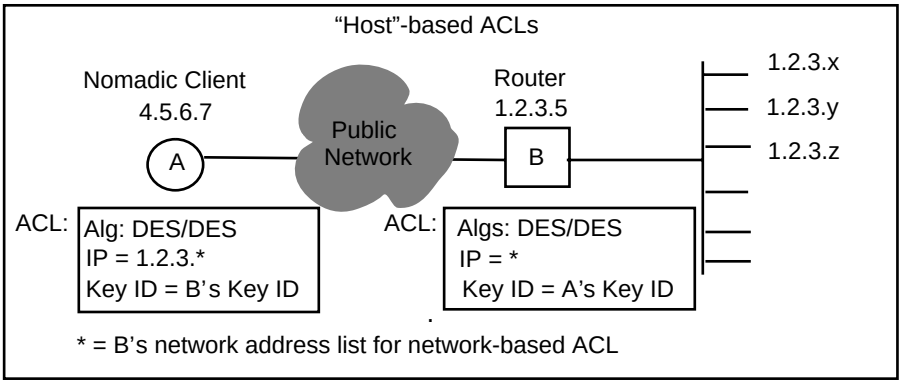

Figure B–3 Example of Case Three—Access Control

3. Node A may be nomadic. Node A will find the sender's key identifier in the SKIP protocol header when it searches for an ACL entry. The packet is authenticated and the sender's IP address is stored until the nomadic entity tries to communicate with the host node again.

The ACL searches through these cases to authenticate the nodes. Node B can have the same options as Node A.

In Case One, or the host ACL, the ACL search for each node finds that they are to use DES/DES to communicate and that they have each other's IP addresses and certificates.

In Case Two, or network ACL, Node B is a router; Node A has the same information as before, but instead of the IP address of 1.2.3.5, it has the router's network address list (1.2.3.*) so it can communicate with any of the nodes in that list.

The main difference is that Node A does not have the certificates of the list of individual hosts on Node B's network, it just has Node B's certificate. So, the whole set of addresses behind Node B is protected; data are encrypted up to Node B and then are sent in the clear behind Node B to the individual hosts on Node B's network.

In Case Three, or the nomadic ACL, a nomadic ACL entry containing the sender's key identifier in the SKIP protocol header is found. The router has an address of "*" for the nomadic system. The entry is found and the packet is authenticated, it stores the sender's IP address until the nomadic entity tries to communicate with the host node again.

Transport and Tunnel Modes

Each IP packet can be encrypted or authenticated in two ways:

1. Transport Mode—Only the data part of the IP packet can be encrypted.
2. Tunnel Mode—The whole IP packet is protected.

Topology Hiding

SKIP supports topology hiding through the use of a *tunnel address*. The tunnel address field contains the IP address of the host that serves as the intermediary between any or all hosts or systems on a network whose topology is to remain hidden from the rest of the world. The source host is not hidden; only the destination address can be hidden (that is, replaced with a tunnel address that the user specified). To hide the topology, the remote system must be configured using Tunnel Mode and the same router must be used for the tunnel destination as the original destination.

Public-Key Cryptography and Diffie-Hellman Certificates

Public-Private Keys

Cryptography takes the original message, and produces an encoded version by using a special piece of information known to the sender and receiver. The original message is called the plaintext, the special information is called the key, and the resulting message is called the ciphertext. Cryptosystems work by taking the digital representation of the plaintext and manipulating it mathematically under the control of the digital key to produce the ciphertext.

Public-key cryptography was invented by Whitfield Diffie and Martin Hellman. It takes a message encrypted in one shared secret and decrypts it in another. The keys are mathematically related in such a way that a knowledge of one key does not make it possible to figure out the other key. This permits the one key, the public key, to be

made widely known, while the corresponding private key is known only to a single user. The two keys together are called a key pair.

The Diffie-Hellman key produces shared secret keys directly from private and public components that are not in themselves keys. The advantage of a public-key system is that the secret components do not have to be shared to exchange information securely. The private portion is never given out to anyone, and it cannot feasibly be calculated from the public portion.

Certificates

To know that the key pair being used in the transaction is actually the key pair for that user, a special sort of signed record is used called a certificate. A certificate contains information identifying the user: distinguished name, public key, and expiration date; for example, digitally signed by a trusted network entity called a certification authority (CA).

Certification Authority

The CA's public key is known to every user of the network. This permits anyone wishing to authenticate a certificate to follow the same procedure for authenticating a message. The CA's public key is available in certificate format so that it, too, can be verified. The major commercial application for CAs is to authenticate businesses and the employees of those businesses. SunCA and SunCAglobal are Sun Microsystems' certification authority in the Internet Commerce Group (ICG).

Unsigned Diffie-Hellman (UDH) Keys

Unsigned Diffie-Hellman (UDH) keys derive their certificate name from the *hash* of the *public key*. By communicating the hash over the telephone or some other trusted method of communication with another party, user's can securely communicate without requiring the infrastructure of a CA. How their certificate name is derived and how this is placed in the SKIP packet are shown in Figure B–4.

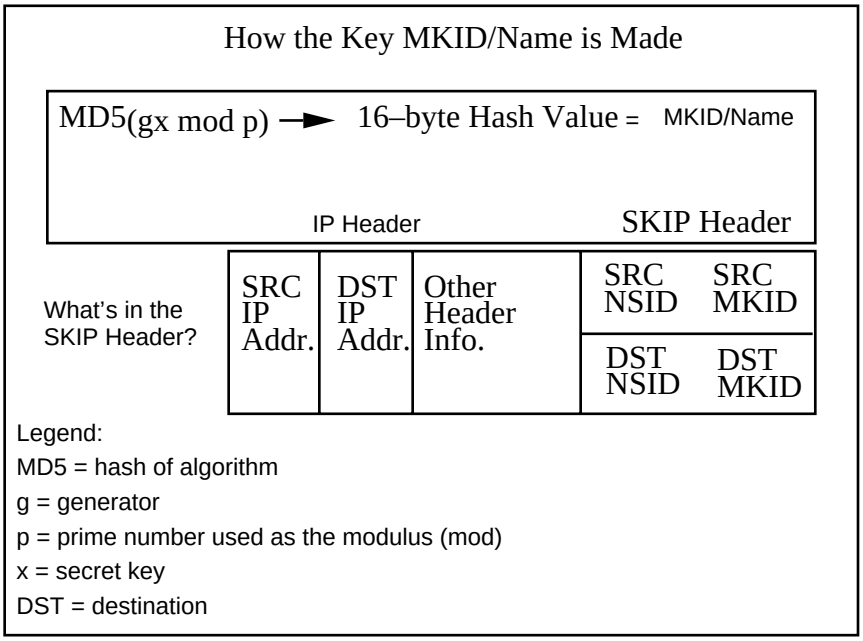

Figure B–4 Unsigned Diffie-Hellman Keys

The advantages of UDH pairs of public and private keys over signed certificates are as follows:

- The private key is a 256–bit secret that is generated by the user. This secret never leaves the user's machine so there is no longer any need to keep the floppy diskettes containing the certificate locked up somewhere nor to trust that the packages have not been tampered with. All that is necessary is to trust that the user's machine is kept secure.
- Since the private key is always in the user's machine, the physical handling of the key is eliminated so there is no need for a CA and no need to package and ship the private keys.
- Since a key cannot be duplicated and is not registered by a CA, it does not need to be formally revoked; if there is reason to believe a key has been compromised, just generate a new key and repeat the distribution and authentication of your public value and its ID.
- The authentication of the public keys is simplified because it does not have to be done secretly, merely securely.

An example of this last point might be as follows: Two users call each other on the telephone and exchange public keys using the recognition of each other's voice as the authentication mechanism. Because these are hashes of public keys, it does not matter if anyone overhears the conversation.

The disadvantage of UDH keys is that the names must be communicated out of band, such as over the telephone by PGP.

SKIP must be able to name or identify a given key. It does this by using a name or identifier drawn from a namespace identifier (NSID). It also must be able to figure out which certificate name to use when communicating with a remote system. This information can be derived from its IP address or explicitly stated in the protocol via NSID / key ID.

If NSID is set to 0, the IP address is used for key lookup. By default, the NSID is set at 0 and a key ID is not sent; however, with the key ID 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, various people have the ability to communicate with each other and corporate using encryption.

The Namespace Identifiers (NSID)

SunScreen SKIP provides users with the ability to separate the identity of an entity from its physical address. This means that each person (sender or receiver) participating in a transfer of encrypted data over a computer network can be identified by a namespace identifier / key identifier (NSID / key ID) pair.

NSIDs are a part of SKIP; these identifiers are used to identify the keys being used. The NSIDs supported by *SunScreen SKIP* are

- NSID 0 (No Key ID present, figure out which key to use from the IP address)
- NSID 1 (IPv4 address key ID, for hosts whose key IDs do not match their IP address, such as hosts that use signed SunCA keys)
- NSID 8 (MD5 hash of Diffie-Hellman Public Value Key ID present, for UDH keys that are not signed by any CA)

The first two are nearly identical in that they both use signed X.509 keys, with one very important difference. SKIP packets that use NSID 1 include the key ID in the packet. SKIP packets that use NSID 0 figure out which key to use.

With SunCA keys, for example, it is necessary to put the key identifier into the SKIP header because the IP address may not correspond to the identifier in the certificate. If there is a SunCA key identifier of “0a000101” for a certificate, it becomes “10.0.1.1” in IP address terminology.

Further, if your IP address is “192.12.10.49,” then you would have to include your key identifier in the SKIP header because it does not equal your IP address. But with NSID 0, which also uses X.509 certificates, it is guaranteed that the key identifier is the IP address; therefore, the key identifier does not have to be sent.

Using NSID 0 results in a small gain in efficiency by not having to send the key identifier. This is what is meant by “No Key ID present” in the NSID 0 bullet above. This approach reduces the amount of packet expansion because of SKIP.

Traffic Encryption

Traffic is encrypted using conventional symmetric key cryptography, such as RC2, RC4, DES, and the like. The user installs *SunScreen SKIP*, which has the algorithm packages that are required. Traffic encryption keys are changed based on the volume of data and the length of time a key is used.

There is a tool with a GUI to control how often you want the traffic encryption keys changed. As shipped, the default is to change traffic keys after every 512K bytes of data or after being used for 30 seconds; traffic keys are deleted after being unused for 30 seconds. You can change these values to meet the security needs of your site. This tool is discussed in detail in Chapter 3.

It is important to change the traffic encryption keys frequently enough so that cracking a key will leave little data, and yet not so frequently so that reconfiguring the keys incurs excessive overhead.

Authentication of SKIP Packets

Authentication is used to guarantee the integrity of SKIP packets. In this process, AH authentication protects the integrity of the packets and the mutual authentication of the sender and the receiver.

AH stands for Authentication Header. This header has been defined for the authentication of IP packets by the IPSEC working group of the ETIF. Packet authentication is performed with a keyed hash function to create a MAC that guarantees the integrity of the packet. When the sender transmits a packet, it calculates a hash of the IP packet along with a key and includes it in the packet. When the packet is received, the receiver calculates the hash over the IP packet and the key as well. If the value that the receiver calculates is the same as the one that the sender included in the packet, the packet has been authenticated. If someone modified the packet in transmission, the value that the receiver calculates will not match the one that the sender calculated and the packet fails authentication and is discarded.

Key and Certificate Management with SKIP

Keys and certificates are handled by the key manager. Details of the implementation are presented above. Local-key (that is, your own key) information is managed using the `skiplocal` command, CA information is managed using the *skipca* command, and peer certificate information is managed by the `skipdb` 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. *SunScreen SKIP* operates a source function on the low-order bits of the Diffie-Hellman key agreement algorithm to yield the key. The key is encrypted using conventional symmetric key cryptography.
- The traffic encryption algorithm. A number of conventional symmetric-key algorithms are supported, such as DES, RC2, and RC4. A random traffic key is used as a key to encrypt data. The algorithms supported are automatically installed by *SunScreen SKIP* and appear in skiptool.
- The traffic authentication algorithm. Currently, only the keyed MD5 algorithm is used.

As stated earlier, certificates are the digital documents that testify to the binding of a public key to an individual (or other entity) to prevent someone else from impersonating you. For two hosts that are running a security package to communicate, they must exchange certificates or public keys. Common methods of exchange for these items are

1. Certificate Discovery Protocol (CDP)—Hosts running SKIP request each other's certificates through a clear channel. A host can also ask a certificate server for a certificate.
2. Manual Exchange—This procedure is manual in that the certificate and possibly the key are provided by the certifying agency on physical media: tape, diskette or CD-ROM. They must be loaded into the system by the user through the command line provided by the vendor.

SKIP supports the common methods of certificate and key exchange. By default, the key manager asks the host with which it is trying to communicate for its certificate or public key.

It is useful to allow a system to have more than one pair of public-private keys. For example, keys of different sizes may be required because of U.S. export controls or local laws or regulations when communicating with subsidiaries in other countries.

To meet these requirements, *SunScreen SKIP* implementation allows a system to possess as many pairs of keys as required. Similarly, the *SunScreen SKIP* can also be configured with the details of several CAs so that certificates signed by different CAs can be checked for authenticity.

For more information on configuring certificate-fetching protocols and certificate management, see the man pages for `skipd`, `skipdb`, and `skipca`.

Certificate Discovery Protocol (CDP)

Certificate Discovery Protocol (CDP) greatly simplifies the management of secure communications because it eliminates the manual exchange of certificates. CDP can be used to exchange X.509 or UDH certificates.

What Are the Operation Requirements of CDP?

To work, the hosts on both sides of a communication must support CDP and both users must agree to use it.

Caution - SunScreen SPF-100 does not support certificate discovery, you cannot use it to communicate between a machine that is running *SunScreen SKIP* and a SunScreen SPF-100.

If both hosts can use CDP and both users agree to it, then the users merely exchange certificate identifiers and allow CDP to do the work instead of exchanging their public keys. This is a simpler solution than manually exchanging certificates.

As an example, if for X.509 certificates, your certificate number is “0a000100” and another user’s public certificate number or master key identifier is “0a000102,” you can exchange these numbers and enter them into your respective ACL when you set up your ACL with the other user’s host for access.

You can do the same for UDH certificates, namely, by exchanging hash values.

Then, when communication between the two is attempted, even though your *SunScreen SKIP* program does not have the peer’s certificate in its certificate database, your host can request that the certificate be sent automatically from the other host and can put it into its certificate database since it knows the certificate’s master key ID.

How Do You Configure CDP?

The only configuration required is to enter the host with which you wish to communicate into your ACL, along with its certificate number or master key ID. If the two hosts attempt to communicate, the fact that there is no corresponding certificate for the key ID in the certificate database automatically activates CDP. If you are communicating to hosts through an encrypting gateway, you must configure the encrypting gateway’s IP address as the tunnel address. This alerts *SunScreen SKIP* to query the gateway for its certificate.

There is a `skip.conf` file that stores configuration data. You can set its values through the `skip_conf` command.

More information on the `skip_conf` command can be found in the man pages.

How Long Are Certificates Cached?

Once the certificates have been transferred and entered into the certificate database of the hosts of the users that wish to communicate, they are cached until they expire or until they are replaced.

The SKIP Encryption Algorithm

SKIP uses the knowledge of its own secret key or private component and the destination's public component to calculate a unique key that can only be used between them.

Each side's public component can be defined as g^x mod p, where x is the private component. In this system, g is the generator and p is a prime number that is used as the modulus (mod). g and p are fixed values known to both parties.

The first node is called Node I. Node I has a public component K_i and a private component *i*. The second node is called Node J. Node J has a public component K_j and a private component *j*.

Every node's public component is distributed in the form of a certificate. They are connected by an unsecure network.

Because Node I knows its own private component and Node J's public component, it can use the two components to compute a unique key that only the two of them can know.

Note - This shared secret is implicit. It does not need to be communicated explicitly to either principal. Each principal can compute this secret based on knowledge of the other principal's identity and public-key certificate. The shared secret is computed using the well-known Diffie-Hellman algorithm.

This mutually authenticated long-term secret is used to derive a key, which is denoted K_{ij} in SKIP Version 1 and K_{ijn} in SKIP, *n* is a number derived from an ever increasing counter that is called the "n counter."

Note - In SKIP, the master key is not used directly, but it is hashed together with some other data to produce the key.

The key is derived by taking the low-order key size bits of g^{ij} mod p. The key K_{ij} or K_{ijn} is used as a master or key-encrypting key to provide IP packet-based encryption and authentication. An individual IP packet is encrypted (or authenticated) using a randomly generated packet key denoted as K_p.

The packet key is in turn encrypted using K_{ij} or K_{ijn}. Since K_{ij} or K_{ijn} can be cached for efficiency, it allows traffic (that is, packet) keys to be modified very rapidly (if necessary even on a per-packet basis) without incurring the computational overhead of a public-key operation.

Furthermore, since the keys are communicated in the packets themselves, there is no need to incur the overhead and complexity of a pseudo-session layer underneath IP. Figure B–5 shows an encrypted IP packet, using the two-step encryption procedure described above.

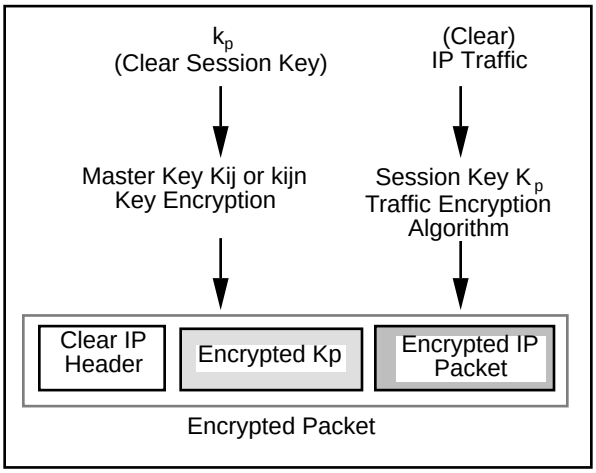

Figure B–5 Encrypted Packet

When a node receives this encrypted packet, it looks up the sender's certificate. Using this and the receiving node's long-term private key, the receiving node can compute K_{ij} or K_{ijn}. Using K_{ij} or K_{ijn}, the receiving node can decrypt K_p and, therefore, decrypt the packet.

Although there is a packet key in each packet, it is not necessary to change the key in every packet. The keys can be changed as frequently as desired based on key-management policies enforced at the site.

Zero-Message Master-Key Update

The preceding section describes how the nodes can compute one long-term key, K_{ij} or K_{ijn}. Changing this key requires issuing a new certificate to one or the other principal.

There are two desirable reasons for updating the master key. The first is that it minimizes the exposure of any given key-encrypting key, making cryptanalysis more difficult. Second, updating the master key prevents reusing compromised traffic keys (K_p). Should a traffic key used for packet authentication ever be compromised (for whatever reason), then it cannot be used to send forged traffic since the encryption of K_p under the current K_{ij} or K_{ijn} is not known.

The master key is updated by sending a counter (say n) in the packet that only increments and is never decremented. The key K_{ij} becomes a function of this counter *n*, as follows:

```
Kijn = h(Kij, n)
```

where *h* is a pseudo-random function such as MD5.

A second feature of the incrementing counter is that it prevents coarse-grained playback of traffic. Once the master keys are updated, traffic that has been encrypted or authenticated with the help of earlier master keys cannot be played back.

In SKIP, the n-counter increments once an hour. It began at zero on January 1, 1995, 00:00:00 GMT.

Summary

This appendix discussed the ideas essential to understanding how SKIP works in more detail. It described how *SunScreen SKIP* handles keys and certificates with and without a CA; examined how the encryption algorithm operates; listed what important services *SunScreen SKIP* provides; and presented an overall view of the *SunScreen SKIP* architecture.

Glossary

3DES	Also called triple-DES or DES-EDE-IT. It means encryption is performed on a block three times with the two keys: first with the first key, then with the second key, and finally with the first key again. The resulting key length is 112–bits. See DES and EDE.
ACL	Access control list. Limits and controls who uses a host system or applications through communications link
address	In networking, a unique code that identifies a node to the network.
ADP	Algorithm discovery protocol. Enables one entity to inform another of the capabilities it supports.
AH	Authentication header. A mechanism for providing strong integrity and authentication for IP datagrams. It may also provide nonrepudiation, depending on which cryptographic algorithm is used and how keying is performed. It does not provide confidentiality or protection from traffic analysis.
algorithm	A sequence of steps designed to solve a problem or execute a process such as drawing a curve from a set of control points, or encrypting a block of data.
alias	Used with the Log Browser to refer to a textual representation of a numerical filter parameter, such as a port, IP address, or error code.
API	Application programmer's interface. A set of calling conventions defining how a service is invoked through a software package.
argument	An item of information following a command. It may, for example, modify the command or identify a file to be affected.

attack An attempted cryptanalysis or an attempt to compromise system security.

authentication The property of knowing that the claimed sender is in fact the actual sender.

block Groups of bits are called blocks.

block cipher or block algorithm An encryption algorithm that encrypts while blocks at once. (See stream ciphers)

Bourne shell The shell used by the standard Bell Labs UNIX.

broadcast A packet delivery system where a copy of a given packet is given to all hosts attached to the network.

button A one-choice element of a control area or a menu that starts an activity. Buttons execute commands (command buttons), display pop-up windows (window buttons), and display menus (menu buttons).

CA Certification authority. A trusted network entity that digitally signs a certificate containing information identifying the user; such as, the user's name, public key, and the key's expiration date.

cache A buffer of high-speed memory used to store frequently accessed memory or values. A cache increases effective memory transfer rates and processor speed.

CBC Cipher block chaining (see also DES). A mode used to chain a feedback mechanism, which essentially means the previous block is used to modify the encryption of the next block.

CDP Certificate discovery protocol. A request/response protocol used by two parties to transfer certificates.

CD-ROM Compact disc, read-only memory. A form of storage characterized by high capacity (roughly 600 megabytes) and the use of laser optics rather than magnetic means for reading data.

certificate A certificate is a data structure that binds the identity of an entity with a public-key value. SunScreen uses X.509 certificates.

CFB Cipher feedback. Uses a block cipher (such as DES) to implement a stream cipher.

cipher	A cryptographic algorithm used for encryption or decryption.
ciphertext	An encrypted message.
CLI	Command line interface
command	In a graphical user interface (GUI), a button, menu item, or controls.
command button	The button used to execute application commands.
compiler	A translation program that converts a high-level computer language (such as FORTRAN) into machine language.
confidentiality	The property of communicating such that the intended recipients know what is being sent, but unintended parties cannot determine what is sent.
controls	Objects in a menu that are used to perform an action.
cookie	(In cryptography) A cookie is a pseudo-random number used to prevent denial-of-service attacks.
cryptanalysis	The art and science of breaking ciphertext.
cryptography	The art and science of keeping messages secure.
C shell	The standard shell provided with Berkeley standard versions of UNIX.
daemon	A process that runs in the background to perform a task on behalf of the system.
data compression	Application of an algorithm to reduce the bit rate of a digital signal.
data encrypting key	A key used to encipher and decipher data intended for programs that perform encryption.
decoder	A facility that takes data that have been encoded, or compressed, by an encoder and decompresses them.
decryption	The process of turning ciphertext back into plaintext.

DES A commonly used, highly sophisticated algorithm developed by IBM for the U.S. National Bureau of Standards for encrypting and decrypting data. See *CBC*.

DH Diffie-Hellman. A classic cryptographic construction that uses exponentiations over a prime field.

digital signatures The bit string attached to the document to authenticate it when signed.

diskette A 3.5–inch removable storage medium supported by some Sun systems.

DN Distinguished name. A numeric string representation of a list of IP addresses or equivalent identifier for principals in the network, such as IP nodes or users.

DNS Domain name system. The distributed name/address mechanism used in the Internet.

DSA Digital signature algorithm. Each DSA is responsible for the directory information for a single organization or organizational unit.

dynamic packet screening Examines traffic to be either allowed or rejected.

dynamic translation A *NAT* address translation that converts a set of internal private addresses into external public addresses. It allows internal hosts to contact external hosts, but it cannot be used to allow external hosts to contact internal hosts.

EDE Encrypt-decrypt-encrypt (See 3DES)

EFS Encryption Firewall Server. A software solution that can reside on any Sun machine running Solaris 2.4 or 2.5. It can secure all the servers on a corporate intranet. A corporation may have any number of database servers—one each for marketing, accounting, and engineering divisions, for example. Each server's data should be protected by EFS. The majority of break-ins that companies experience happen from within the company's own network. This product locks down each server. Since it works at the network IP layer, it can "talk" to any other machine and thus can be placed in "front" of any competitor's machine to protect it.

EKE Encrypted key exchange

encapsulation	The technique used by layered protocols in which a layer adds header information to the protocol data unit from the layer above. In Internet terminology, for example, a packet would contain a header from the physical layer, followed by a header from the network layer (IP), followed by a header from the transport layer (TCP), followed by the application protocol data.
encryption	A mechanism commonly used to provide confidentiality.
encryption key	A value that controls how information is enciphered or deciphered. Often called the public key. (See data encrypting key)
entity	Terminology for a layer protocol machine. An entity within a layer performs the functions of the layer within a single computer system, accessing the layer entity below and providing services to the layer entity above at local service access points.
ESP	Encapsulating security payload. A mechanism for providing integrity and confidentiality to IP datagrams. In some circumstances it can also provide authentication to IP datagrams, depending on which algorithm or algorithm mode is used. It does not provide nonrepudiation and protection from traffic analysis.
Ethernet	A type of local area network that enables communication between machines connected directly together through cables.
FDDI	Fiber distributed data interface. A high-speed networking standard. The underlying medium is fiber optics, and the topology is a dual-attached, counter-rotating token ring. FDDI networks can often be spotted by the orange fiber “cable.”
filters	Allow selection of a subset of packets based on specific attributes of the logged packets.
Filter Catalog	Used with the Log Browser as part of the hierarchical structure of saved filters. Filter groups are saved in filter catalogs.
Filter Directory Service	Used with the Log Browser as the hierarchical structure into which filters are grouped and saved.
Filter Group	Used with the Log Browser and refers to a set of filters created by the administrator, then saved so they can be applied to multiple log files.

GUI Graphical user interface. Provides the user with a method of interacting with the computer and its special applications, usually via a mouse or other selection device. The GUI usually includes such things as windows, an intuitive method of manipulating directories and files, and icons.

hash A message digest or cryptographic checksum.

header file A file of information, identified at the beginning of the program, that contains the definitions of data types and variables used by the functions in the program.

hidden file A special type of file, such as `.login`, that does not show up in normal file listings. Special files usually pertain to system configuration.

host computer The primary or controlling computer in a multiple computer installation.

hung A condition in which the system is frozen and unresponsive to commands.

IANA Internet Assigned Numbers Authority. SKIP was assigned the protocol decimal number 57. SKIP Version 1 was assigned protocol decimal number 79 by IANA.

ICG Internet Commerce Group. A business unit of Sun Microsystems, Inc., that is committed above all else to developing solutions to communicate securely over unsecured public networks. Formed in 1994, ICG already has three strong SunScreen security product lines that stand at the head of the class. Each depends on the public-key cryptography invented by Sun's Distinguished Engineer Whitfield Diffie, along with Stanford's Martin Hellman. Building upon public-key cryptography, ICG developed SKIP—Simple Key-management for Internet Protocols—the premier protocol that makes key management easier to use than previous innovations. SKIP is the central cryptographic protocol upon which ICG draws in its products.

ICMP Internet control message protocol

icon (1) An on-screen symbol that simplifies access to a program, command, or data file. (2) A small pictorial representation of a base window. Displaying objects as icons conserves space on the screen while keeping the window available for easy access.

IDEA	International data encryption algorithm
integrity	The property of ensuring that data are transmitted from the source to destination without undetected alteration.
IP	Internet Protocol. The network layer protocol for the Internet protocol suite.
IPSEC	IP security
ISDN	Integrated Services Digitial Network
IV	Initialization vector
kernel	The core of the operating system software. The kernel manages the hardware and supplies fundamental services such as filing that the hardware does not provide.
Key and Certificate Diskette	Diskettes that contain both the private key and the certificate containing the public key. The identifier for this certificate is on the label. The information is extremely sensitive and should be kept secure.
key encrypting key	A key used to encipher and decipher other keys, as part of a key management and distribution system.
keyspace	The range of possible values of the key.
layer	A set of structures and routines that handle a particular class of events. For example, in the seven-layer International Organization of Standardization's open systems interconnection model, the network layer is responsible for routing the signals to their intended recipients.
locally stored secret	The secret key that corresponds to a public key certificate. Used to encrypt and decrypt messages.
Log Browser	The main window for examining log files.
MAC	Message authentication code. The term "MAC" is synonymous with the term "authentication data."
man pages	Stands for manual pages, the UNIX on-line documentation.

MD Message digest. An authentication code that cryptographically guarantees that data have not been forged or tampered with.

MD5 A message digest one-way hash function designed by Ron Rivest. The algorithm produces a 128–bit hash, or message digest, of the input message.

MDC Message digest cipher

menu button A multiple-choice control that has a *menu mark* and is used to display a menu.

menu mark A hollow triangle in the border of a button or following a menu item that has a *submenu* attached to it. The triangle points to where the menu or submenu is displayed.

MIC Message integrity check

MI Message indicator

MKID Master Key-ID. A generic term used to identify a particular key. MKIDs effectively decouple the identification of a master key for purposes of key lookup and access control from issues of network topology, routing, and IP addresses.

modulus An arithmetic operation used in programming whose result is the remainder of a division operation. The plural is moduli.

MSP Message security protocol. An X.400-compatible application-level protocol for securing electronic mail that was developed by NSA.

MTU Maximum transmission unit

multicast A special form of *broadcast* where copies of the packet are delivered to only a subset of all possible destinations.

NAT Network Address Translation. An address translation function used in SKIP where packets passing through a box have their addresses changed (or translated) between sets of addresses to hide internal addresses such that they cannot be used as an attack point. It is also useful on the Internet as you must use registered addresses so no two systems use the same address. However, many internal networks were built without registering their addresses because they were built before the Internet was considered vital to business. Address translation can be used to translate unregistered (that is,

	illegal) addresses into a smaller set of registered addresses, thus allowing internal systems with unregistered addresses to access systems on the Internet.
network	The hardware connecting various systems enabling them to communicate.
network administrator	The person who maintains a network.
network layer	The third of the seven layers in the International Organization for Standardization's open systems interconnection model for standardizing computer-to-computer communications.
network mask	A number used by software to separate the local subnet address from the rest of a given Internet protocol address.
NeWS	Network extensible window system that Sun developed and licenses. It is based on Abobe's PostScript.
NFS	A distributed file system developed by Sun that enables a set of computers to cooperatively access each other's files in a transparent manner.
NIS	Network information service. A distributed network database containing key information about the systems and the users on the network. The NIS database is stored on the master server and all the slave servers.
node	A point at which subsidiary parts originate or center.
nonrepudiation	The property of a receiver being able to prove that the sender of some data did in fact send the data even though the sender might later desire to deny ever having sent these data.
NSA	National Security Agency. The United States of America's official cryptographic organ.
NSID	Name-space identifier. Used to identify a naming scheme for a key.
OFB	Output feedback
one-way hash	A cryptographically secure hash function that cannot be reversed. (See MD5, SHA, hash)
OSPF	Open shortest path first

packet A group of information in a fixed format that is transmitted as a unit over communications lines.

passphrase A passphrase is longer than a password. Letters in both upper and lower case can be used, as well as special characters and numbers.

password A security measure used to restrict access to computer systems and sensitive files. A password is a unique string of characters that a user types in as an identification code. The system compares the code against a stored list of authorized passwords and users. If the code is legitimate, the system allows the user access, at whatever security level has been approved for the owner of the password.

peer Any functional unit in the same layer as another *entity*.

peer-to-peer communication Interaction between devices that operate on the same communications level on a network based on a layered architecture.

PFS Perfect forward secrecy. Ephemeral Diffie-Hellman key exchange used in conjunction with the SKIP key distributions protocol provides PFS where required.

PGP Pretty Good Privacy. A public-domain encryption program that uses IDEA for data encryption, *RSA* for key management, and MD5 as a one-way hash function.

ping Packet Internet groper. A program used to test reachability of destinations by sending them an Internet control message protocol (ICMP) echo request and waiting for a reply.

plaintext An unencrypted message.

PMSP Preliminary Message Security Protocol. Used for “unclassified but sensitive” messages (this protocol is also called “Mosaic”).

pop-up window A window that displays to perform a specific function and then is dismissed.

private key Often called the decryption key and sometimes called the secret key.

protocol A protocol is a series of steps, involving two or more parties, designed to accomplish a task.

POSIX An acronym created from the phrase “portable operating system interface,” which is an IEEE standard that defines a set of operating-system services. Programs that adhere to the POSIX standard can be easily ported from one system to another.

pseudo-random Something that is statistically random.

Public Certificate Diskette Contains only the certificate containing the public key. The identifier for this certificate is on the label.

public key Often called the encryption *key*.

public-key certificate Someone’s public key, signed by a trustworthy person.

public-key cryptography Also known as asymmetric key cryptography. In public-key cryptosystems, everyone has two related complementary keys, a publicly revealed key and a secret key (also frequently called a private key). Each key unlocks the code that the other key makes. Knowing the public key does not help you deduce the corresponding secret key. The public key can be published and widely disseminated across a communications network. This protocol provides privacy without the need for the same kind of secure channels that a conventional cryptosystem requires.

push To add a new element to a stack, a data structure generally used to hold, temporarily, pieces of data being transferred or the partial result of an arithmetic operation.

query The process by which a master station asks a slave station to identify itself and give its status.

quit To stop in an orderly manner; to execute the normal shutdown of a program and return control to the operating system.

radio button In graphical user interfaces, a means of selecting one of several mutually exclusive options, usually within an option-selection area such as a dialog box. The presence of radio buttons in a list of options means that only one of the options can be selected at any given time. Visually, a radio button is a small circle that, when selected, has a smaller, filled circle inside it.

RC2 and RC4 RC2 and RC4 are variable-key-size encryption algorithms designed by Ron Rivest for RSA Data Security, Inc. Apparently, “RC” stands for “Ron’s Code.” RC2 is a variable-key-size block cipher, designed to be a replacement for DES. RC4 is a variable-key-size stream

	cipher that is, according to the company, ten times faster than DES. Both algorithms are quite compact, and their speed is independent of the key's size. It is notable, however, that neither RC2 nor RC4 has survived the 20 years of intense cryptanalysis that DES has. See DES.
RC2-40 and RC4-40	A globally exportable encryption algorithm from RSA, Inc.
robust	Reliable or dependable. Not prone to error. Usually used in reference to an application program.
root user name	SunOS user name that grants special privileges to the person who logs in with that ID. The user who can supply the correct password for the root user name is given superuser privileges for the particular machine.
router	A system responsible for making decisions about which of several paths network (or Internet) traffic will follow. To do this it uses a routing protocol to gain information about the network, and algorithms to choose the best route based on several criteria known as "routing metrics."
rules	There are three types of rules: Encryption, Pass (in the clear), and Fail. An encryption rule determines how data are secured and always takes precedent over pass or fail rules. Pass rules take precedence over fail rules.
RSA	The most popular public-key algorithm named after the three inventors, Ron Rivest, Adi Shamir, and Leonard Adleman.
SDNS	Secure Data Network System
secret key	See private key
security association	The set of security information relating to a given network connection or set of connections.
session key	A common cryptographic technique to encrypt each individual conversation between two people with a separate key.
SHA	Secure hash algorithm
shared-key cryptography	Also known as symmetric key cryptography. Shared-key cryptography is cryptography where each party must have the same key to encrypt or decrypt ciphertext.

SKCS Symmetric Key CryptoSystem

SKID Secret-key identification

SKIP Simple Key-management for Internet Protocols. SKIP is a public key certificate-based key-management scheme that provides key-management for Internet protocols. SKIP uses certified Diffie-Hellman public values, which obviates the need for pseudo-session state establishment and for prior communications between two participating ends in order to acquire and change traffic encryption keys.

SKIP addresses the problems inherent in companies that have employees telecommuting from home, a sales force on the road working from laptops, or customers purchasing their products off the Web. The SunScreen SKIP allows employees, partners, and consumers to communicate with encryption, while protecting their data as they go out on the Internet. At this point, SunScreen SKIP works with Sun Solaris™ 2.4, 2.5, and 2.5.1 and Solaris for the Intel Platform.

SNMP Simple network management protocol. The network management protocol of choice for TCP/IP-based internets.

source code The uncompiled version of a program written in a language such as C or Pascal. The source code must be translated to machine language by a program known as the compiler before the computer can execute the program.

SPARC A RISC processor.

special characters Or, metacharacters, is a character having a special meaning to UNIX. For example, the UNIX shell interprets the ? character to stand for any single character.

SPI Security parameters index. An unstructured opaque index that is used in conjunction with the destination address to identify a particular security association.

stack A list constructed and maintained so that the next item to be retrieved and removed is the most recently stored item still in the list.

static translation A NAT address translation that provides fixed translation between an external public address and internal private (possibly illegal)

address. It provides a way for external hosts to initiate connections to internal hosts at the expense of "using up" an external address.

stream algorithm or stream cipher — A symmetric algorithm that operates on the plaintext a single bit (or byte) at a time. (See block cipher)

submenu — A menu that displays additional choices that is displayed through a menu item on a menu.

SunScreen — The name of a family of security products produced by the Internet Commerce Group. SunScreen is a dedicated hardware security solution enabling companies to connect securely to and conduct business privately over an unsecured public network.

SunScreen SPF-100 — Winner of LAN magazine's 1996 Product-of-the-Year Award in the firewall category, the SunScreen SPF-100 acts as a traditional firewall, while securing communications over the Internet by engaging in encryption, authentication and key agreement procedures. One of the best uses of the SunScreen SPF-100 is as an Internet gateway which protects a corporate network from break-ins. The SunScreen SPF-100 also encrypts data sent out on the Internet or intranet and protects it. It is a complete hardware/software solution. The SunScreen SPF-100 is a stealthy machine that encrypts and decrypts without being detected. In short, the SunScreen SPF-100 is invisible on the network, and you can't break something you can't see.

superuser — A special user who has privileges to perform all administrative tasks on the system. Also known as root.

Telnet — The virtual terminal protocol in the Internet suite of protocols. Enables users of one host to log into a remote host and interact as normal terminal users of that host.

TIFF — Tag image file format

TCP/IP — Transport control protocol/interface program. The protocol suite originally developed for the Internet. It is also called the Internet protocol suite. SunOS networks run on TCP/IP by default.

token — A unique structured data object or message that circulates continuously among the nodes of a token ring and describes the current state of the network. Before any node can send a message, it must first gain control of the token.

token ring network An LAN formed in a ring (closed loop) topology that uses *token* passing as a means of regulating traffic on the line.

topology hiding The tunnel address 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 must remain unknown or hidden from the rest of the world.

traffic analysis The analysis of network traffic flow for the purpose of deducing information that is useful to an adversary. Examples of such information are frequency of transmission, the identities of the conversing parties, sizes of packets, flow identifiers used, and the like.

transport mode Encrypts only IP packet data, but not the headers.

tunneling The process of encrypting an entire IP packet, and wrapping it in another (unencrypted) IP packet. The source and destination addresses on the inner and outer packets may be different.

tunnel address The address to which tunnels packets are sent. This will be the destination address on the outer (unencrypted) IP packet.

tunnel mode The process of tunneling, as opposed to “transport mode.”

user ID A number that identifies a user to the system.

UDH Unsigned Diffie-Hellman. The UDH public value can only be used when entities are named using the message digest (*hash*) of their DH public value, and these names are securely communicated.

UDP User datagram protocol. All CDP communication uses UDP.

unicast A packet sent to a single destination.

VPN Virtual private network

window In applications and graphical interfaces, a portion of the screen that can contain its own document or message. In window-based programs, the screen can be divided into several windows, each of which has its own boundaries and can contain a different document (or another view into the same document).

window button A button used to display a window containing additional controls.