Solaris Trusted Extensions Administrator's Procedures

Beta

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 819–7309–01
September 2006

060927@15490

Contents

Tables

Figures

Preface

This *Solaris Trusted Extensions Administrator's Procedures* guide provides procedures for managing users, zones, devices, and hosts that are labeled with Solaris™ Trusted Extensions software.

Note – This Solaris release supports systems that use the SPARC® and x86 families of processor architectures: UltraSPARC®, SPARC64, AMD64, Pentium, and Xeon EM64T. The supported systems appear in the *Solaris 10 Hardware Compatibility List* at `http://www.sun.com/bigadmin/hcl`. This document cites any implementation differences between the platform types.

In this document these x86 related terms mean the following:

- "x86" refers to the larger family of 64-bit and 32-bit x86 compatible products.
- "x64" points out specific 64-bit information about AMD64 or EM64T systems.
- "32-bit x86" points out specific 32-bit information about x86 based systems.

For supported systems, see the *Solaris 10 Hardware Compatibility List*.

Who Should Use This Book

This book is used by administrators who are able to assume an administrative role. This book describes how to do administrative tasks that are particular to Trusted Extensions.

Administrators should be familiar with Solaris administration. In addition, you should understand the following:

- Basic concepts and procedures for using a host that is configured with Trusted Extensions, as described in the *Solaris Trusted Extensions User's Guide*.
- How administrative tasks are divided among roles at your site.

How the Solaris Trusted Extensions Books Are Organized

The Solaris Trusted Extensions 1.0 documentation set supplements the documentation for the Solaris Express release. Obtain a copy of both sets for a complete understanding of Solaris Trusted Extensions. The Solaris Trusted Extensions documentation set consists of the following books.

Book Title	Topics	Audience
Solaris Trusted Extensions Transition Guide	Provides an overview of the differences between Trusted Solaris 8 software, Solaris Express software, and Solaris Trusted Extensions 1.0 software.	All
Solaris Trusted Extensions Reference Manual	Provides Solaris Trusted Extensions man pages.	All
Solaris Trusted Extensions User's Guide	Describes the basic features of Solaris Trusted Extensions. This book contains a glossary.	End users, administrators, and developers
Solaris Trusted Extensions Release Notes	Lists known problems and describes workarounds for Solaris Trusted Extensions 1.0 software.	Administrators, developers
Solaris Trusted Extensions Installation and Configuration	Describes how to plan for, install, and configure Solaris Trusted Extensions.	Administrators, developers
Solaris Trusted Extensions Administrator's Procedures	Provides detailed information for performing specific administration tasks.	Administrators, developers
Solaris Trusted Extensions Developer's Guide	Describes how to develop applications with Solaris Trusted Extensions.	Developers, administrators
Solaris Trusted Extensions Label Administration	Provides information on specifying label components in the label encodings file.	Administrators
Compartmented Mode Workstation Labeling: Encodings Format	Describes the syntax used in the label encodings file. The syntax enforces the various rules for well-formed labels for a system.	Administrators

Documentation, Support, and Training

The Sun web site provides information about the following additional resources:

- Documentation (`http://www.sun.com/documentation/`)
- Support (`http://www.sun.com/support/`)
- Training (`http://www.sun.com/training/`)

Typographic Conventions

The following table describes the typographic changes that are used in this book.

TABLE P–1 Typographic Conventions

Typeface or Symbol	Meaning	Example
`AaBbCc123`	The names of commands, files, and directories, and onscreen computer output	Edit your `.login` file. Use `ls -a` to list all files. `machine_name% you have mail.`

`AaBbCc123`	What you type, contrasted with onscreen computer output	`machine_name%` **`su`**
		`Password:`

aabbcc123	Placeholder: replace with a real name or value	The command to remove a file is `rm` *filename*.
AaBbCc123	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the *User's Guide*. Perform a *patch analysis*. Do *not* save the file. [Note that some emphasized items appear bold online.]

Shell Prompts in Command Examples

The following table shows the default system prompt, role prompt, and superuser prompt for the C shell, Bourne shell, and Korn shell.

TABLE P–2 Shell Prompts

Shell	Prompt
C shell prompt	`machine_name%`
C shell superuser prompt	`machine_name#`
Profile shell prompt	`$`
Bourne shell and Korn shell prompt	`$`
Bourne shell and Korn shell superuser prompt	`#`

CHAPTER 1

Trusted Extensions Administration Concepts

This chapter introduces you to administering a system that is configured with Solaris Trusted Extensions software.

- “Solaris OS With Trusted Extensions Software” on page 19
- “Basic Concepts of Trusted Extensions” on page 21

Solaris OS With Trusted Extensions Software

Trusted Extensions software adds labels to a system that is running the Solaris Operating System (Solaris OS). Labels implement *mandatory access control* (MAC). MAC, along with DAC, discretionary access control, protect system subjects and objects. Trusted Extensions software provides interfaces to handle label configuration, label assignment, and label policy.

Similarities Between Trusted Extensions and the Solaris OS

Trusted Extensions software uses rights profiles, roles, auditing, privileges, and other security features of the Solaris OS. You can use Solaris Secure Shell (SSH), BART, the Solaris cryptographic framework, IPsec, and IPfilter with Trusted Extensions.

- As in the Solaris OS, users can be limited to those applications that are necessary for performing their jobs. Other users can be authorized to do more.
- As in the Solaris OS, capabilities that were formerly assigned to superuser are available to separate, discrete “roles.”
- As in the Solaris OS, privileges protect processes. Zones are also used to separate processes.
- As in the Solaris OS, events on the system can be audited.
- Trusted Extensions uses the system configuration files of the Solaris OS, such as `policy.conf` and `exec_attr`.

Differences Between Trusted Extensions and the Solaris OS

Trusted Extensions software extends the Solaris OS. The following list provides an overview. For a quick reference, see Appendix A.

- Trusted Extensions controls access to data with special security tags that are called *labels*. Labels provide *mandatory access control* (MAC). MAC protection is in addition to UNIX® file permissions, or discretionary access control (DAC). Labels are directly assigned to users, zones, devices, windows, and network endpoints. Labels are implicitly assigned to processes, files, and other system objects.

  MAC cannot be overridden by ordinary users. Trusted Extensions requires ordinary users to operate in labeled zones. By default, no users or processes in labeled zones can override MAC.

  As in the Solaris OS, the ability to override security policy can be assigned to specific processes or users when MAC can be overridden. For example, users can be authorized to change the label of a file. Such an action upgrades or downgrades the sensitivity of the information in that file.
- Trusted Extensions adds to existing configuration files and commands. For example, Trusted Extensions adds audit events, authorizations, privileges, and rights profiles.
- Trusted Extensions can require features from the Solaris OS that are optional when Trusted Extensions is not configured. For example, zones and roles are required on a system that is configured with Trusted Extensions.
- Trusted Extensions recommends features that are optional on a Solaris system. For example, Trusted Extensions recommends that you turn the root user into the `root` role.
- Trusted Extensions can change the default behavior of the Solaris OS. For example, on a system that is configured with Trusted Extensions, auditing is enabled by default. Device allocation is required.
- Trusted Extensions can narrow the options that are available in the Solaris OS. For example, on a system that is configured with Trusted Extensions, the NIS+ naming service is not supported. Also, in Trusted Extensions, all zones are labeled zones. Unlike the Solaris OS, labeled zones must use the same pool of user and group IDs. Additionally, in Trusted Extensions, labeled zones can share one IP address.
- Trusted Extensions provides additional graphical user interfaces (GUIs) and command line interfaces (CLIs). For example, Trusted Extensions provides a Device Allocation Manager to administer devices. The `updatehome` command is used to place startup files in an ordinary user's home directory at every label.
- Trusted Extensions requires the use of particular GUIs for administration. For example, on a system that is configured with Trusted Extensions, the Solaris Management Console is used to administer users, roles, and the network. Similarly, the Admin Editor is used to edit system files.
- Trusted Extensions limits what users can see. For example, a device that cannot be allocated by a user cannot be seen by that user.
- Trusted Extensions limits users' desktop options. For example, users are allowed a limited time of workstation inactivity before the screen locks.

Basic Concepts of Trusted Extensions

Trusted Extensions software adds labels to a Solaris system. Labeled desktops and trusted applications, such as the Label Builder and the Device Allocation Manager, are also added. The concepts in this section are basic to understanding Trusted Extensions, both for users and administrators. Users are introduced to these concepts in the *Solaris Trusted Extensions User's Guide*.

Trusted Extensions Protections

Trusted Extensions software adds to the protection of the Solaris OS. The Solaris OS protects access to the system with user accounts that require passwords. You can require that passwords be changed regularly, be of a certain length, and so on. Roles require additional passwords to perform administrative tasks. Additional authentication limits the damage that can be done by an intruder who guesses the root password, because roles cannot be used as login accounts. Trusted Extensions software restricts users and roles to an approved label range. This label range limits the information that users and roles can access.

Trusted Extensions software displays the Trusted Path symbol, an unmistakable, tamper-proof emblem that appears at the left of the trusted stripe. In CDE, the stripe is on the bottom of the screen. The Trusted Path symbol indicates to users when they are using security-related parts of the system. If this symbol does not appear when the user is running a trusted application, that version of the application should be checked immediately for authenticity.

Most security-related software, the Trusted Computing Base (TCB), runs in the global zone. Ordinary users cannot enter the global zone or view its resources. Users are able to interact with TCB software, as in when they change passwords. The Trusted Path symbol is displayed whenever the user is interacting with the TCB.

Trusted Extensions and Access Control

Trusted Extensions software protects information and other resources through both discretionary access control (DAC) and mandatory access control (MAC). DAC is the traditional UNIX permission bits and access control lists that are set at the discretion of the owner. MAC is a mechanism that the system enforces automatically. MAC controls all transactions by checking the labels of processes and data in the transaction.

A user's *label* represents the sensitivity level at which the user is permitted to and chooses to operate. The label determines the information that the user is allowed to access. Both MAC and DAC can be overridden by special permissions that are in the Solaris OS. *Privileges* are special permissions that can be granted to processes. *Authorizations* are special permissions that can be granted to users and roles by an administrator.

As administrator, you need to train users on the proper procedures for securing their files and directories, according to your site's security policy. Furthermore, you should instruct any users who are allowed to upgrade or downgrade labels as to when it is appropriate to change a label.

Roles and Trusted Extensions

On a system that is running Solaris software without Trusted Extensions, roles are optional. On a system that is configured with Trusted Extensions, roles are required. The system is administered by the System Administrator role and the Security Administrator role. In some cases, the root role is used.

As in the Solaris OS, rights profiles are the basis of a role's capabilities. Trusted Extensions provides two rights profiles, Information Security and User Security. These two profiles define the Security Administrator role.

The programs and tools that are available to a role in Trusted Extensions have a special property, the *trusted path attribute*. This attribute indicates that the program is part of the TCB. The trusted path attribute is available when a program is launched from the global zone.

For information about roles, see Part III, "Roles, Rights Profiles, and Privileges," in *System Administration Guide: Security Services*.

Labels in Trusted Extensions Software

Labels and clearances are the heart of mandatory access control (MAC) in Trusted Extensions. They determine which users can access which programs, files, and directories. Labels and clearances consist of one *classification* component and zero or more *compartment* components. The classification component indicates a hierarchical level of security such as `TOP SECRET` or `CONFIDENTIAL`. The compartment component represents a group of users who might need access to a common body of information. Some typical types of compartments are projects, departments, or physical locations. Labels are readable by authorized users, but internally, labels are manipulated as numbers. The numbers and their readable versions are defined in the `label_encodings` file.

Trusted Extensions mediates all attempted security-related transactions. The software compares the labels of the accessing entity, typically a process, and the entity being accessed, usually a filesystem object. The software then permits or disallows the transaction depending on which label is *dominant*. Labels are also used to determine access to other system resources, such as allocatable devices, networks, frame buffers, and other hosts.

Dominance Relationships Between Labels

One entity's label is said to *dominate* another label if the following two conditions are met:

- The classification component of the first entity's label is equal to or higher than the second entity's classification. The security administrator assigns numbers to classifications in the `label_encodings` file. These numbers are compared when determining dominance.
- The set of compartments in the first entity includes all of the second entity's compartments.

Two labels are said to be *equal* if they have the same classification and the same set of compartments. If the labels are equal, they dominate each other and access is permitted.

If one label has a higher classification or if it has the same classification and its compartments are a superset of the second label's compartments or both, the first label is said to *strictly dominate* the second label.

Two labels are said to be *disjoint* or *noncomparable* if neither label dominates the other label.

The following table presents examples of label comparisons for dominance. In the example, NEED_TO_KNOW is a higher classification than INTERNAL. There are three compartments: Eng, Mkt, and Fin.

TABLE 1–1 Examples of Label Relationships

Label 1	Relationship	Label 2
NEED_TO_KNOW Eng Mkt	(strictly) dominates	INTERNAL Eng Mkt
NEED_TO_KNOW Eng Mkt	(strictly) dominates	NEED_TO_KNOW Eng
NEED_TO_KNOW Eng Mkt	(strictly) dominates	INTERNAL Eng
NEED_TO_KNOW Eng Mkt	dominates (equals)	NEED_TO_KNOW Eng Mkt
NEED_TO_KNOW Eng Mkt	is disjoint with	NEED_TO_KNOW Eng Fin
NEED_TO_KNOW Eng Mkt	is disjoint with	NEED_TO_KNOW Fin
NEED_TO_KNOW Eng Mkt	is disjoint with	INTERNAL Eng Mkt Fin

Administrative Labels

Trusted Extensions provides two special administrative labels that are used as labels or clearances: ADMIN_HIGH and ADMIN_LOW. These labels are used to protect system resources and are intended for administrators rather than ordinary users.

ADMIN_HIGH is the highest label. ADMIN_HIGH dominates all other labels in the system and is used to protect system data, such as administration databases or audit trails, from being read. You must be in the global zone to read data that is labeled ADMIN_HIGH.

ADMIN_LOW is the lowest label. ADMIN_LOW is dominated by all other labels in a system. Mandatory access control does not permit users to write data to files with labels lower than the subject's label. Thus, a file at the label ADMIN_LOW can be read by ordinary users, but cannot be modified. ADMIN_LOW is typically used to protect public executables that are shared, such as files in /usr/bin.

Label Encodings File

All label components for a system, that is, classifications, compartments, and the associated rules, are stored in an `ADMIN_HIGH` file, the `label_encodings` file. This file is located in the `/etc/security/tsol` directory. The security administrator sets up the `label_encodings` file for the site. A label encodings file contains:

- Component definitions – Definitions of classifications, compartments, labels, and clearances, including rules for required combinations and constraints
- Accreditation range definitions – Specification of the clearances and minimum labels that define the sets of available labels for the entire system and for ordinary users
- Printing specifications – Identification and handling information for print banners, trailers, headers, footers, and other security features on printer output
- Customizations – Local definitions including label color codes, and other items

For more information, see the `label_encodings`(4) man page. Detailed information can be found in *Solaris Trusted Extensions Label Administration* and in *Compartmented Mode Workstation Labeling: Encodings Format*.

Label Ranges

A *label range* is the set of potentially usable labels at which users can operate. Users and resources both have label ranges. Resources that can be protected by label ranges include such things as allocatable devices, networks, interfaces, frame buffers, and commands or actions. A label range is defined by a clearance at the top of the range and a minimum label at the bottom.

A range is not necessarily all combinations of labels that fall between a maximum and minimum label. Rules in the label encodings file can disqualify certain combinations. A label must be *well-formed*, that is, permitted by all applicable rules in the label encodings file, in order to be included in a range.

On the other hand, a clearance does not have to be well-formed. Suppose, for example, that a label encodings file prohibits any combination of compartments `Eng`, `Mkt`, and `Fin` in a label. `INTERNAL Eng Mkt Fin` would be a valid clearance but not a valid label. As a clearance, this combination would let a user access files that are labeled `INTERNAL Eng`, `INTERNAL Mkt`, and `INTERNAL Fin`.

Account Label Range

When you assign a clearance and a minimum label to a user, you define the upper and lower boundaries of the *account label range* in which that user is permitted to operate. The following equation describes the account label range, using ≤ to indicate dominated by or the same as:

minimum label ≤ permitted label ≤ clearance

Thus, the user is permitted to operate at any label that is dominated by the clearance as long as that label dominates the minimum label. When a user's clearance or minimum label is not expressly set, the defaults that are defined in the `label encodings` file take effect.

Users can be assigned a clearance and a minimum label that enable them to operate at more than one label, or at a single label. When a user's clearance is equal to the user's minimum label, the user can operate at only one label.

Session Range

The *session range* is the set of labels that is available to a user during a Trusted Extensions session. The session range must be within the user's account label range and the label range set for the system. If the user selects single-label session mode, the session range is limited to that label. If the user selects multilabel mode, then the label that is selected becomes the session clearance. The session clearance defines the upper boundary of the session range. The user's minimum label defines the lower bound. The user begins the session in a workspace at the minimum label. During the session, the user can switch to a workspace at any label in the session range.

What Labels Protect and Where Labels Appear

Labels appear on the desktop and on output that is executed on the desktop, such as printer output.

Applications
: Applications start processes. These processes run at the label of the workspace where the application is started. An application in a labeled zone, as a file, is labeled at the label of the zone.

Devices
: Data flowing through devices is controlled through device allocation and device label ranges. To use a device, users must be within the label range of the device, and be authorized to allocate the device.

Filesystem mount points
: Every mount point has a label. The label is viewable by using the `getlabel` command.

Network interfaces
: IP addresses (hosts) have templates which describe their label range. Unlabeled hosts also have a default label.

Printers and printing
: Printers have label ranges. Labels are printed on body pages. Labels, handling information, and other security information is printed on the banner and trailer pages. To configure printing in Trusted Extensions, see Chapter 15 and “Labels on Printed Output” in *Solaris Trusted Extensions Label Administration*.

Processes
: Processes are labeled. Processes run at the label of the workspace where the process originates. The label of a process is visible by using the `plabel` command.

Users
: Users are assigned a label and a label range. The label of the user's workspace indicates the label of the user's processes.

Windows
: Labels are visible at the top of windows on the desktop. The label of the desktop is also indicated by color. In CDE, the color shows on the desktop switch, and above window title bars.

When a window is moved to a workspace at another label, the window maintains its original label.

Zones

Every zone has a unique label. The files and directories that are owned by a zone are at the label of the zone. For more information, see the `getzonepath`(1) man page.

◆ ◆ ◆ CHAPTER 2

Trusted Extensions Administration Tools

This chapter lists the tools that are available in Trusted Extensions, the location of the tools, and the databases on which the tools operate.

- "Administration Tools for Trusted Extensions" on page 27
- "Trusted CDE Actions" on page 28
- "Device Allocation Manager" on page 31
- "Solaris Management Console Tools" on page 32
- "Command Line Tools" on page 34
- "Remote Administration" on page 37

Administration Tools for Trusted Extensions

Administration on a system that is configured with Trusted Extensions uses many of the same tools that are available in the Solaris OS. Trusted Extensions offers security-enhanced tools as well. Administration tools are available only to roles in a role workspace.

Within a role workspace, you can access four types of trusted applications. The following table summarizes these administrative tools.

TABLE 2–1 Trusted Extensions Administrative Tools

Tool	Description	For More Information
In CDE, actions in the Trusted_Extensions folder in the Application Manager folder	Used to edit local files that the Solaris Management Console does not manage, such as `/etc/system`. Some actions run scripts, such as the Install Zone action.	See "Trusted CDE Actions" on page 28 and "How to Launch CDE Administrative Actions in Trusted Extensions" on page 43.
Device Allocation Manager	Used to administer the label ranges of devices, and to allocate or deallocate devices.	See "Device Allocation Manager" on page 31 and "Handling Devices in Trusted Extensions (Task Map)" on page 189.

TABLE 2–1 Trusted Extensions Administrative Tools *(Continued)*

Tool	Description	For More Information
Solaris Management Console	Used to configure users, roles, rights, hosts, zones, and networks. Can update local files or LDAP databases. Can also launch the `dtappsession` legacy application.	For basic functionality, see Chapter 2, “Working With the Solaris Management Console (Tasks),” in *System Administration Guide: Basic Administration*. For information that is specific to Trusted Extensions, see “Solaris Management Console Tools” on page 32.
Solaris Management Console commands, such as `smuser`(1M) and `smtnzonecfg`(1M)	Is the command line interface for the Solaris Management Console.	For a list, see Table 2–4.
Trusted Extensions commands	Used to perform tasks that are not covered by Solaris Management Console tools or CDE actions.	For the list of administrative commands, see Table 2–5. Also, see individual man pages in the *Solaris Trusted Extensions Reference Manual*.

Trusted CDE Actions

The following graphic and tables list the CDE actions that roles in Trusted Extensions can run. These trusted CDE actions are available from the Trusted_Extensions folder.

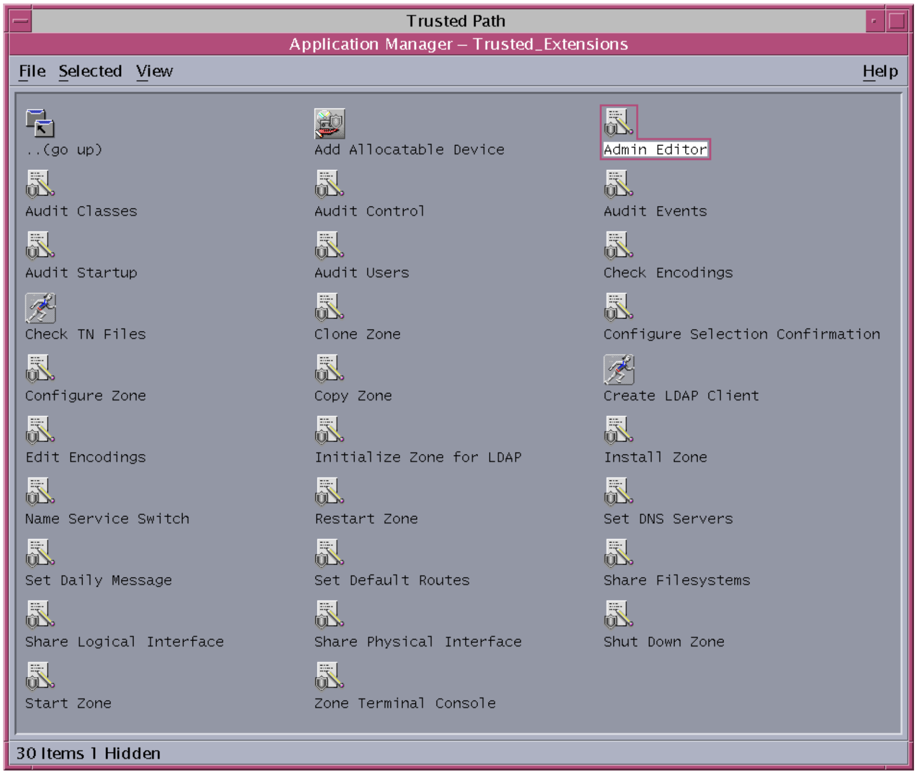


TABLE 2–2 Administrative Actions in CDE, Their Purpose, and Associated Rights Profiles

Action Name	Purpose of Action	Default Rights Profile
`Add Allocatable Device`	Creates devices by putting entries in device databases. See `add_allocatable`(1M).	Device Security
`Admin Editor`	Edits the specified file. See "How to Edit Administrative Files in Trusted Extensions" on page 44.	Object Access Management
`Audit Classes`	Edits the `audit_class` file. See `audit_class`(4).	Audit Control
`Audit Control`	Edits the `audit_control` file. See `audit_control`(4).	Audit Control
`Audit Events`	Edits the `audit_event` file. See `audit_event`(4).	Audit Control
`Audit Startup`	Edits the `audit_startup.sh` script. See `audit_startup`(1M).	Audit Control
`Check Encodings`	Runs the `chk_encodings` command on specified encodings file. See `chk_encodings`(1M).	Object Label Management
`Check TN Files`	Runs the `tnchkdb` command on `tnrhdb`, `tnrhtp`, and `tnzonecfg` databases. See `tnchkdb`(1M).	Network Management

TABLE 2–2 Administrative Actions in CDE, Their Purpose, and Associated Rights Profiles *(Continued)*

Action Name	Purpose of Action	Default Rights Profile
`Configure Selection Confirmation`	Edits `/usr/dt/config/sel_config` file. See `sel_config`(4).	Object Label Management
`Create LDAP Client`	Makes the global zone an LDAP client of an existing LDAP directory service.	Information Security
`Edit Encodings`	Edits the specified `label_encodings` file and runs the `chk_encodings` command. See `chk_encodings`(1M).	Object Label Management
`Name Service Switch`	Edits the `nsswitch.conf` file. See `nsswitch.conf`(4).	Network Management
`Set DNS Servers`	Edits the `resolv.conf` file. See `resolv.conf`(4).	Network Management
`Set Daily Message`	Edits the `/etc/motd` file. At login, the contents of this file display in the Last Login dialog box.	Network Management
`Set Default Routes`	Specifies default static routes.	Network Management
`Share Filesystem`	Edits the `dfstab` file. Does not run the `share` command. See `dfstab`(4).	File System Management

The following actions are used by the install team during zone setup. Some of these actions can be used for maintenance and troubleshooting.

TABLE 2–3 Installation Actions in CDE, Their Purpose, and Associated Rights Profiles

Action Name	Purpose of Action	Default Rights Profile
`Clone Zone`	Creates a labeled zone from a ZFS snapshot of an existing zone.	Zone Management
`Copy Zone`	Creates a labeled zone from an existing zone.	Zone Management
`Configure Zone`	Associates a label with a zone name.	Zone Management
`Initialize Zone for LDAP`	Initializes the zone for booting as an LDAP client.	Zone Management
`Install Zone`	Installs the system files that a labeled zone requires.	Zone Management
`Restart Zone`	Restarts a zone that has already been booted.	Zone Management
`Share Logical Interface`	Sets up one interface for the global zone and a separate interface for the labeled zones to share.	Network Management
`Share Physical Interface`	Sets up one interface that is shared by the global zone and the labeled zones.	Network Management
`Shut Down Zone`	Shuts down an installed zone.	Zone Management
`Start Zone`	Boots an installed zone and starts the services for the zone.	Zone Management
`Zone Terminal Console`	Opens a console to view processes in an installed zone.	Zone Management

Device Allocation Manager

A *device* is either a physical peripheral that is connected to a computer or a software-simulated device called a pseudo-device. Because devices provide a means for the import and export of data to and from a system, devices must be controlled to properly protect the data. Trusted Extensions uses device allocation and device label ranges to control data flowing through devices.

Examples of devices that have label ranges are frame buffers, tape drives, diskette and CD-ROM drives, printers, and USB devices.

Users allocate devices through the Device Allocation Manager. The Device Allocation Manager mounts the device, runs a clean script to prepare the device and performs the allocation. When finished, the user deallocates the device through the Device Allocation Manager, which runs another clean script and unmounts and deallocates the device.

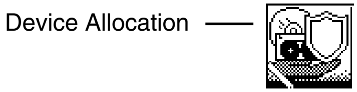
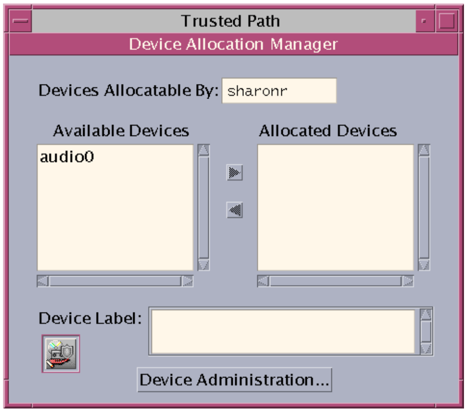


FIGURE 2–1 Device Allocation Manager Icon

Administrators manage devices by using the Device Administration tool from the Device Allocation Manager.



Devices Allocatable By: sharonr

Available Devices

audio0

Allocated Devices

Device Label:

FIGURE 2–2 Device Allocation Manager GUI

For more information on device protection in Trusted Extensions, see Chapter 17.

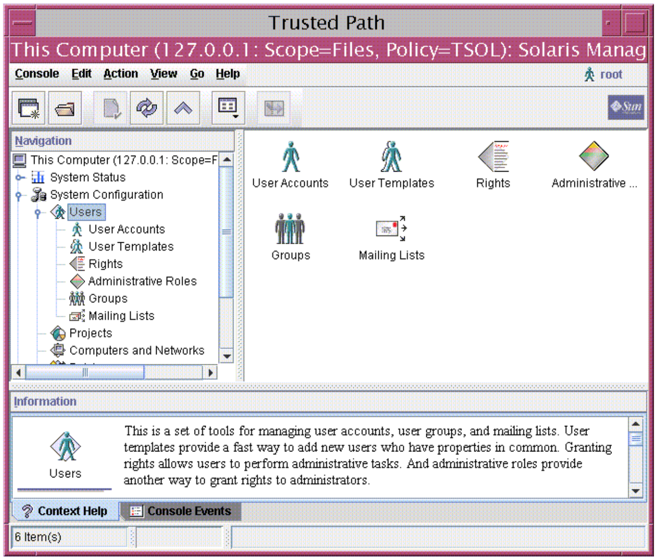
Solaris Management Console Tools

The Solaris Management Console provides access to families of GUI-based administration tools. These tools enable you to edit items in various configuration databases. In Trusted Extensions, the Solaris Management Console is the administrative interface for users, roles, and the trusted network databases.

- Trusted Extensions modifies the Solaris Management Console Users tool set. For an introduction to the tool set, see Chapter 2, "Working With the Solaris Management Console (Tasks)," in *System Administration Guide: Basic Administration*.
- Trusted Extensions adds the Security Templates tool and the Trusted Network Zones tool to the Computers and Networks tool set.

Solaris Management Console tools are stored in sets that are referred to as *toolboxes*. To administer Trusted Extensions, Trusted Extensions provides toolboxes whose `Policy=TSOL`. You can access tools according to scope, that is, according to naming service. The available scopes are local host and LDAP.

The Solaris Management Console is shown in the following figure. A `Scope=Files` Trusted Extensions toolbox is loaded and the Users tool set is open.



This is a set of tools for managing user accounts, user groups, and mailing lists. User templates provide a fast way to add new users who have properties in common. Granting rights allows users to perform administrative tasks. And administrative roles provide another way to grant rights to administrators.

FIGURE 2–3 Typical Trusted Extensions Toolbox in the Solaris Management Console

Trusted Extensions Tools in the Solaris Management Console

Trusted Extensions adds configurable security attributes to three tools:

- **User Accounts tool –** Is the administrative interface to change user's label, change users' views of labels, and to control account usage.
- **Administrative Roles tool –** Is the administrative interface to change a role's label range and screen-locking behavior when idle.
- **Rights tool** –– Includes CDE actions that can be assigned to rights profiles. Security attributes can be assigned to these actions.

Trusted Extensions adds two tools to the Computers and Networks tool set:

- **Security Templates tool –** Is the administrative interface for managing the label aspects of hosts and networks. This tool modifies the `tnrhtp` and `tnrhdb` databases, enforces syntactic accuracy, and updates the kernel with the changes.
- **Trusted Network Zones tool –** Is the administrative interface for managing the label aspects of zones. This tool modifies the `tnzonecfg` database, enforces syntactic accuracy, and updates the kernel with the changes.

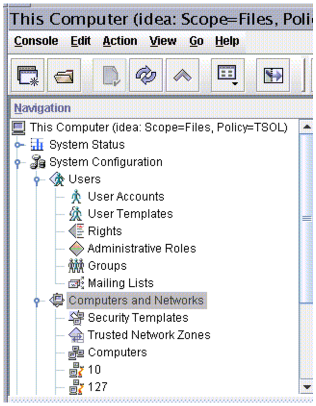


FIGURE 2–4 Computers and Networks Tool Set in the Solaris Management Console

Security Templates Tool

A *security template* describes a set of security attributes in a template. The Security Templates tool enables you to conveniently assign a set of security attributes to a group of hosts. These attributes control how data is packaged, transmitted, and interpreted. Hosts that are assigned to a template have identical security settings.

The hosts are defined in the Computers tool. The security attributes of the hosts are assigned in the Security Templates tool. The Modify Template dialog box describes the template:

- **General tab –** Describes the template. Includes its name, host type, default label, domain of interpretation (DOI), accreditation range, and set of discrete sensitivity labels.
- **Hosts Assigned to Template tab –** Lists all the hosts on the network that you have assigned to this template.

Trusted networking and security templates are explained in more detail in Chapter 12.

Trusted Network Zones Tool

The Trusted Network Zones tool identifies the zones on your system. Initially, the global zone is listed. When you add the zones and their labels, the zone names display in the pane. Zone creation usually occurs during system configuration. Label assignment, multilevel port configuration, and label policy is configured in this tool.

Solaris Management Console Documentation

The main source of documentation for the Solaris Management Console is its online help. Context-sensitive help is tied to the currently selected feature and is displayed in the information pane. Expanded help topics are available from the Help menu or by clicking links in the context-sensitive help. Further information is provided in Chapter 2, “Working With the Solaris Management Console (Tasks),” in *System Administration Guide: Basic Administration* and in “Using the Solaris Management Tools With RBAC (Task Map)” in *System Administration Guide: Basic Administration*.

Command Line Tools

Commands that are specific to Trusted Extensions are in the *Solaris Trusted Extensions Reference Manual*. When Trusted Extensions modifies Solaris commands, the modified commands are in the *Solaris Reference Manual*. The `man` command finds all the commands.

The following table lists commands that are unique to Trusted Extensions. The commands are listed in man page format.

TABLE 2–4 List of User and Administrative Trusted Extensions Commands

Man Page	Trusted Extensions Modification	Documented Use
`add_allocatable(1M)`	Enables a device to be allocated by adding the device to device allocation databases. By default, removable devices are allocatable.	Use the Device Allocation Manager. “How to Configure a Device” on page 191

TABLE 2–4 List of User and Administrative Trusted Extensions Commands *(Continued)*

Man Page	Trusted Extensions Modification	Documented Use
atohexlabel(1M)	Translates a label into hexadecimal format.	"How to Get the Hexadecimal Equivalent for a Label" on page 58
chk_encodings(1M)	Checks the integrity of the label_encodings file.	"How to Debug a label_encodings File" in *Solaris Trusted Extensions Label Administration*
dtappsession(1)	Opens a remote CDE session by using the Application Manager.	Chapter 8
getlabel(1)	Displays the label of the selected files or directories.	"How to Display the Labels of Mounted Files" on page 106
getzonepath(1)	Displays the full pathname of a specific zone.	"Acquiring a Sensitivity Label" in *Solaris Trusted Extensions Developer's Guide*
hextoalabel(1M)	Translates a hexadecimal label into its readable equivalent.	"How to Get a Readable Label From Its Hexadecimal Form" on page 59
plabel(1)	Displays the label of the current process.	
remove_allocatable(1M)	Prevents allocation of a device by removing its entry from device allocation databases.	Use the Device Allocation Manager. "How to Configure a Device" on page 191
setlabel(1)	Relabels the selected item. Requires the solaris.label.file.downgrade or solaris.label.file.upgrade authorization. These authorizations are in the Object Label Management rights profile.	For a procedure that uses the mouse rather than the command line, see "How to Use Two File Managers to Relabel a File" on page 111.
smtnrhdb(1M)	Manages entries in the tnrhdb database locally or in a naming service database.	For procedures that use the Solaris Management Console, see "Configuring Trusted Network Databases (Tasks)" on page 138.
smtnrhtp(1M)	Manages entries in the tnrhtp database locally or in a naming service database.	
smtnzonecfg(1M)	Manages entries in the local tnzonecfg database.	For a procedure that uses the Solaris Management Console, see "How to Create a Multilevel Port for a Zone" on page 112.
tnchkdb(1M)	Checks the integrity of the tnrhdb and tnrhtp databases.	"How to Check the Syntax of Trusted Network Databases" on page 151.
tnctl(1M)	Caches network information in the kernel.	"How to Synchronize Kernel Cache With Network Databases" on page 153.
tnd(1M)	Executes the trusted network daemon.	"How to Change the tnd Polling Interval" on page 154.

TABLE 2–4 List of User and Administrative Trusted Extensions Commands *(Continued)*

Man Page	Trusted Extensions Modification	Documented Use
`tninfo`(1M)	Displays kernel-level network information and statistics.	"How to Compare Network Database Information With Kernel Cache" on page 151.
`updatehome`(1M)	Updates `.copy_files` and `.link_files` for the current label.	"How to Configure Startup Files for Users" on page 74.

The following table lists Solaris commands that are modified or are extended by Trusted Extensions. The commands are listed in man page format.

TABLE 2–5 List of User and Administrative Commands That Trusted Extensions Modifies

Man Page	Purpose of Command	Documented Use
`allocate`(1)	Adds options to clean the allocated device, and to allocate a device to a specific zone. In Trusted Extensions, ordinary users do not use this command.	Use the Device Allocation Manager. "How to Allocate a Device" in *Solaris Trusted Extensions User's Guide*.
`deallocate`(1)	Adds options to clean the device, and to deallocate a device from a specific zone. In Trusted Extensions, ordinary users do not use this command.	"How to Allocate a Device" in *Solaris Trusted Extensions User's Guide*.
`list_devices`(1)	Adds the `-a` option to display device attributes, such as authorizations and labels. Adds the `-d` option to display the default attributes of an allocated device type. Adds the `-z` option to display available devices that can be allocated to a labeled zone.	
`tar`(1)	Adds the `-T` option to archive and extract files and directories that are labeled.	"How to Back Up Files" on page 119 and "How to Restore Files" on page 120.
`auditconfig`(1M)	Adds the `windata_down` and `windata_up` audit policy options.	"How to Configure Audit Policy" in *System Administration Guide: Security Services*.
`auditreduce`(1M)	Adds the `-l` option to select audit records by label.	"How to Select Audit Events From the Audit Trail" in *System Administration Guide: Security Services*.
`automount`(1M)	Modifies the names and contents of `auto_home` maps to account for zone names and zone visibility from higher labels.	"Changes to the Automounter in Trusted Extensions" on page 117.
`ifconfig`(1M)	Adds the `all-zones` option to make an interface available to every zone on the system.	"How to Verify That a Host's Interfaces Are Up" on page 155.
`netstat`(1M)	Adds the `-R` option to display extended security attributes for sockets and routing table entries.	"How to Debug the Trusted Extensions Network" on page 156.

TABLE 2–5 List of User and Administrative Commands That Trusted Extensions Modifies *(Continued)*

Man Page	Purpose of Command	Documented Use
route(1M)	Adds the -secattr option to display the security attributes of the route: min_sl, max_sl, doi, and cipso.	"How to Configure Routes With Security Attributes" on page 149.

Remote Administration

You can remotely administer a system that is configured with Trusted Extensions by using the ssh command, the dtappsession program, or the Solaris Management Console. If site security policy permits, you can make adjustments to log in from a non-Trusted Extensions host, although this configuration is less secure. For more information, see Chapter 8.

◆ ◆ ◆ CHAPTER 3

Getting Started as a Trusted Extensions Administrator

This chapter introduces you to administering a system that is configured with Trusted Extensions.

Security Requirements When Administering Trusted Extensions

In Trusted Extensions, roles are the conventional way to administer the system. Typically, superuser is not used. Roles are created as they are in the Solaris OS, and most tasks are done by roles. In Trusted Extensions, the `root` user is not used to perform administrative tasks.

The following roles are typical of a Trusted Extensions site:

- `root` role – Created by the install team
- Security Administrator role – Created during or after initial configuration by the install team
- System Administrator role – Created by the Security Administrator role

As in the Solaris OS, you might also create a Primary Administrator role, an Operator role, and so on. With the exception of the `root` role, the roles that you create can be administered in a naming service.

As in the Solaris OS, only users who have been assigned a role can assume the role. In Trusted Extensions, you can assume a role from a desktop menu, the Trusted Path menu.

Role Creation in Trusted Extensions

To administer Trusted Extensions, you create roles that divide system and security functions. The install team created the Security Administrator role during configuration. For details, see "Create the Security Administrator Role" in *Solaris Trusted Extensions Installation and Configuration*

The process of creating a role in Trusted Extensions is identical to the Solaris OS process. As described in Chapter 2, the Solaris Management Console is the GUI for managing roles in Trusted Extensions.

- For an overview of role creation, see Chapter 10, "Role-Based Access Control (Reference)," in *System Administration Guide: Security Services* and "Using RBAC (Task Map)" in *System Administration Guide: Security Services*.
- To create a powerful role that is equivalent to superuser, see "Creating the Primary Administrator Role" in *System Administration Guide: Basic Administration*. At sites that use Trusted Extensions, the Primary Administrator role might violate security policy. These sites would turn `root` into a role, and create a Security Administrator role.
- To create the `root` role, see "How to Make root User Into a Role" in *System Administration Guide: Security Services*.
- To create roles by using the Solaris Management Console, see "How to Create and Assign a Role By Using the GUI" in *System Administration Guide: Security Services*.

Role Assumption in Trusted Extensions

Unlike the Solaris OS, Trusted Extensions provides an Assume *Rolename* Role menu item from the Trusted Path menu. After confirming the role password, the software activates a role workspace with the trusted path attribute. Role workspaces are administrative workspaces. Such workspaces are in the global zone.

Getting Started as a Trusted Extensions Administrator (Task Map)

You should familiarize yourself with the following procedures before administering Trusted Extensions.

Task	Description	For Instructions
Log in	Log in securely.	"Starting in Trusted Extensions (Tasks)" in *Solaris Trusted Extensions User's Guide*
Change your password	Unlike the Solaris OS, Trusted Extensions requires users and roles to use the Trusted Path menu to change passwords. The menu has a Change Password option.	"How to Change Your Password" in *Solaris Trusted Extensions User's Guide*

Task	Description	For Instructions
Create useful roles	Create administrative roles for your site. When you create roles in LDAP, this is a one-time task. The Security Administrator role is a useful role.	"Role Creation in Trusted Extensions" on page 39 "Create the Security Administrator Role" in *Solaris Trusted Extensions Installation and Configuration*
(Optional) Make root a role	Turn the root user into the `root` role. This task is done once per system.	"How to Make root User Into a Role" in *System Administration Guide: Security Services*
Assume a role	Enter the global zone in a role.	"How to Enter the Global Zone in Trusted Extensions" on page 41
Administer users, roles, rights, zones, and networks	Use the Solaris Management Console to manage the distributed system.	"How to Launch the Solaris Management Console" on page 42
Administer locally by using CDE actions	Use the administrative actions in the Trusted_Extensions folder.	"How to Launch CDE Administrative Actions in Trusted Extensions" on page 43
Edit an administrative file in CDE	Use the Admin Editor action.	"How to Edit Administrative Files in Trusted Extensions" on page 44
Administer device allocation in CDE	Use the Device Allocation Manager – Device Administration GUI.	"Managing Devices in Trusted Extensions (Tasks)" on page 190

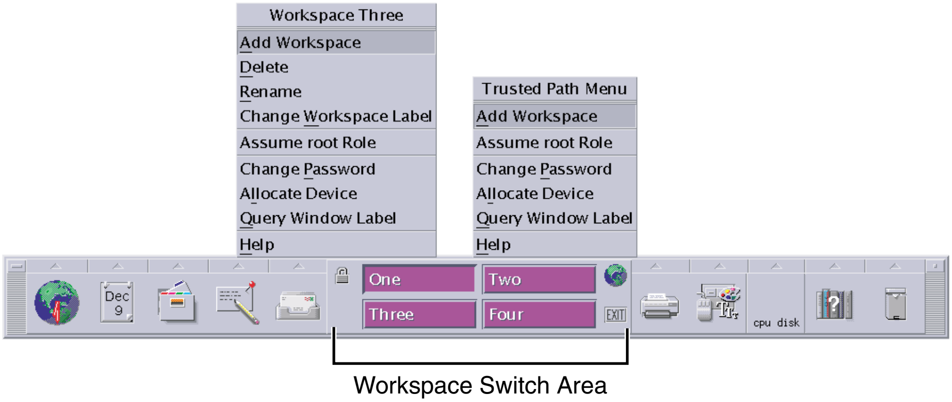
▼ How to Enter the Global Zone in Trusted Extensions

By assuming a role, you enter the global zone in Trusted Extensions. Administration of the entire system is possible only from the global zone. Only superuser or a role can enter the global zone.

After assuming a role, the role can create a workspace at a user label to edit administration files in a labeled zone.

Before You Begin You have created one or more roles, or you plan to enter the global zone as root. For pointers, see "Role Creation in Trusted Extensions" on page 39.

1 In CDE, click mouse button 3 over the workspace switch area.



2 **Choose Assume *rolename* role from the Trusted Path menu.**

For troubleshooting purposes, you can also enter the global zone by starting a Failsafe session in CDE. For details, see "How to Log In to a Failsafe Session in CDE" on page 77.

3 **Type the role password when prompted.**

The workspace changes to the role workspace. In CDE, the workspace switch button changes to the color of the role desktop, and the title bar above a window shows `Trusted Path`.

You leave a role workspace in Trusted Extensions by using the mouse to choose an ordinary user workspace. You can also delete the last role workspace to exit a role.

▼ How to Launch the Solaris Management Console

The first time on a system that you launch the Solaris Management Console, a delay occurs while the tools are registered and various directories are created. This delay typically occurs during system configuration.

Before You Begin You must have assumed a role. For details, see "How to Enter the Global Zone in Trusted Extensions" on page 41.

To use the LDAP toolbox, you must have completed "Configuring the Solaris Management Console for LDAP (Tasks)" in *Solaris Trusted Extensions Installation and Configuration*.

1 **Launch the tool.**

In CDE, you have three choices.

- **Use the `smc` command in a terminal.**

```
$ /usr/sbin/smc &
```

- **From the Tools pull-up menu on the Front Panel, click the Solaris Management Console icon.**

- **In the Trusted_Extensions folder, double-click the Solaris Management Console icon.**

2 Choose Console -> Open Toolbox.

3 From the list, choose a Trusted Extensions toolbox of the appropriate scope.

A Trusted Extensions toolbox has `Policy=TSOL` as part of its name. The Files scope updates local files on the current computer. The LDAP scope updates LDAP directories on the Sun Java System Directory Server. The toolbox names appear similar to the following:

```
This Computer (this-host: Scope=Files, Policy=TSOL)
This Computer (this-host: Scope=LDAP, Policy=TSOL)
```

4 (Optional) Save the current toolbox to reduce reloading time.

a. Choose Console ->Preferences.

b. On the Console tab, click the Use Current Toolbox button.

c. Click OK.

5 Navigate to the desired Solaris Management Console tool.

For tools that Trusted Extensions has modified, click System Configuration.

6 When prompted, provide a password.

Refer to the online help for additional information about Solaris Management Console tools. For an introduction to the tools that Trusted Extensions modifies, see "Solaris Management Console Tools" on page 32.

7 To close the GUI, choose Exit from the Console menu.

▼ How to Launch CDE Administrative Actions in Trusted Extensions

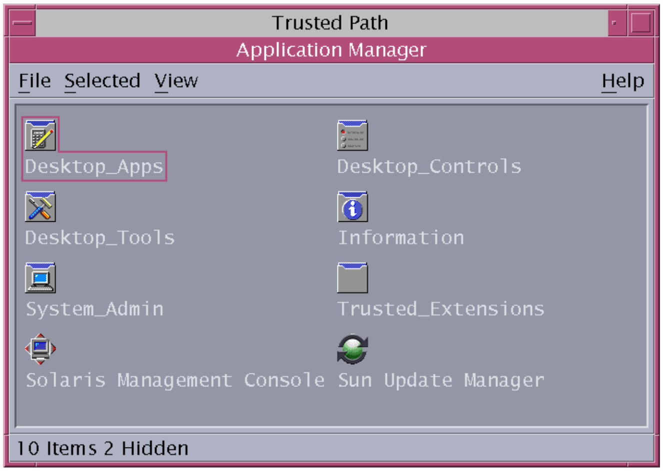
1 Assume a role.

For details, see "How to Enter the Global Zone in Trusted Extensions" on page 41.

2 In CDE, bring up the Workspace menu.

Click mouse button 3 on the background.

a. Click Applications, then click the Application Manager menu item.



The Trusted_Extensions folder is in the Application Manager.

b. **Open the Trusted_Extensions folder.**

c. **Double-click the icon for the desired administrative action.**

For a list of administrative actions, see "Trusted CDE Actions" on page 28.

▼ How to Edit Administrative Files in Trusted Extensions

Before You Begin You must be in a role workspace.

◗ **Double-click the Admin Editor action.**

The Admin Editor action is in the Trusted_Extensions folder. This trusted editor incorporates auditing. This editor also prevents the user from executing shell commands and from saving to any file name other than the original file being edited.

- To create a new file, type the full path name for the new file.

  When you save the file, the editor creates the file with the specified path name.
- To edit an existing file, type the full path name for the existing file.

CHAPTER 4

Security Requirements on a Trusted Extensions System

This chapter describes configurable security features on a system that is configured with Trusted Extensions.

Configurable Solaris Security Features

Trusted Extensions uses the same security features that the Solaris OS provides, and adds some features. For example, the Solaris OS provides `eeprom` protection, password requirements, system protection by locking out a user, protection from keyboard shutdown, and strong password algorithms.

Trusted Extensions differs from the Solaris OS in the actual procedures to modify these security defaults. In Trusted Extensions, you typically administer in a role. Local settings are modified by using the Admin Editor, a trusted editor. Changes that affect the network of users, roles, and hosts are made in the Solaris Management Console.

Trusted Extensions Interfaces for Configuring Security Features

Where Trusted Extensions requires a particular interface to modify security settings, and that interface is optional in the Solaris OS, procedures are provided in this book. Where Trusted Extensions requires the use of the Admin Editor to edit local files, no separate procedures are provided in this book. For example, the procedure, "How to Prevent Account Locking for Individuals" on page 82, describes how to update a user's account by using the Solaris Management Console to prevent the account from being locked. However, the procedure for setting a system-wide password lock policy is not provided in this book. You follow the Solaris instructions, except that in Trusted Extensions, you use the Admin Editor to modify the system file.

Extension of Solaris Security Mechanisms

The following Solaris security mechanisms are extensible in Trusted Extensions as they are in the Solaris OS.

- **Audit events and classes –** Adding audit events and audit classes is described in Chapter 29, “Managing Solaris Auditing (Tasks),” in *System Administration Guide: Security Services*.
- **Rights profiles –** Adding rights profiles is described in Part III, “Roles, Rights Profiles, and Privileges,” in *System Administration Guide: Security Services*.
- **Roles –** Adding roles is described in Part III, “Roles, Rights Profiles, and Privileges,” in *System Administration Guide: Security Services*.
- **Authorizations –** For an example of adding a new authorization, see “Customizing Device Authorizations in Trusted Extensions (Tasks)” on page 199.

As in the Solaris OS, privileges cannot be extended.

Trusted Extensions Security Features

Trusted Extensions provides unique security features.

- Labels – Subjects and objects are labeled. Processes are labeled. Zones and the network are labeled.
- Device Allocation Manager – By default, devices are protected by allocation requirements. A Device Allocation Manager GUI is the interface for administrators and for ordinary users.
- Change Password menu item – The Trusted Path menu enables you to change your user password, and the password of the role that you have assumed.

Security Requirements Enforcement

To ensure that the security of the system is not compromised, administrators need to protect passwords, files and audit data. Computer users need to be trained to do their part. To be consistent with the requirements for an evaluated configuration, follow the guidelines in this section.

Users and Security Requirements

Each site's security administrator ensures that users are trained. The security administrator should hand off the following rules to new employees and remind existing employees of these rules on a regular basis.

- Anyone who knows your password can access the same information that you can without being identified and therefore without being accountable.
- Do not tell anyone else the password.
- Do not write the password down or include it in an email message.
- Choose passwords that are hard to guess.
- Do not leave your computer unattended without locking the screen or logging off.
- Be aware that sender information in email can be forged.
- Remember that administrators do not rely on email to send instructions to users. Do not ever follow emailed instructions from an administrator without first double-checking with the administrator.
- Do not send your password to anyone by email.
- Because you are responsible for the access permissions on files and directories that you create, make sure that the permissions on your files and directories are set appropriately. Do not allow unauthorized users to read a file, to change a file, to list the contents of a directory, or to add to a directory.

Your organization might want to provide additional suggestions beyond the suggestions that are listed in this section.

Email Usage

It is poor practice to use email to instruct users to take an action.

Tell users not to trust email with instructions that purport to come from an administrator. This prevents the possibility that spoofed email messages could be used to fool users into changing a password to a certain value or divulging the password, which could subsequently be used to log in and compromise the system.

Password Enforcement

The System Administrator role must specify a unique user name and a unique user ID when creating a new account. When choosing the name and ID for a new account, the administrator must ensure that both the user name and associated UID are not duplicated anywhere on the network and have not been previously used.

The Security Administrator role is responsible for specifying the original password for each account and for handing off the passwords to new accounts. This administrator should take into account the following information when administering passwords:

- Make sure that the accounts for users who are able to assume the Security Administrator role are configured so that the account cannot be locked. This ensures that at least one account can always log in and assume the Security Administrator role to reopen everyone's account if it ever happens that all other accounts are locked.
- Hand over the password to an account in such a way that the password cannot be eavesdropped by anyone else.
- Change an account's password if there is any suspicion that the password has been discovered by anyone who should not know it.
- Never reuse user names or UIDs over the lifetime of the system. Ensuring that user names and UIDs are not reused prevents possible confusion about the following:
  - Which actions were performed by which user when audit records are analyzed
  - Which user owns which files when archived files are restored

Information Protection

Administrators are responsible for correctly setting up and maintaining DAC and MAC protections for security-critical files. Critical files include the `shadow`(4) file containing encrypted passwords, the local `prof_attr`(4), `exec_attr`(4), and `user_attr`(4) databases, and the audit trail.

⚠ **Caution** – Because the protection mechanisms for LDAP entries are not subject to the access control policy enforced by the Trusted Extensions software, the default LDAP entries should not be extended, and their access rules should not be modified.

Password Protection

In local files, passwords are protected from viewing by DAC and from modifications by both DAC and MAC. Passwords for local accounts are maintained in the `/etc/shadow` file, that is readable only by root. For more information, see the `shadow`(4) man page.

Group Administration

The System Administrator role needs to verify on the local system and on the network that all groups have a unique group ID (GID).

When a local group is deleted from the system, the System Administrator role must ensure the following:

- All objects with the GID of the deleted group must be deleted or be assigned to another group.
- All users who have the deleted group as their primary group must be reassigned to another primary group.

User Deletion Practices

When an account is deleted from the system, the System Administrator role and the Security Administrator role must take the following actions:

- The account's home directory must be deleted.
- Any processes or jobs belonging to the deleted account must be removed:
  - Any objects owned by the account must be deleted or the ownership must be assigned to another user.
  - Any `at` or `batch` jobs scheduled on behalf of the user must be deleted. For details, see the `at`(1) and `crontab`(1) man pages.
- The user (account) name and UID must be retired and not reused.

Rules When Changing the Level of Security for Data

By default, ordinary users can perform cut and paste, copy and paste, and drag and drop operations on both files and selections. The source and target must be at the same label.

To change the label of files or of information within files requires authorization. When users are authorized to change the security level of data, a Selection Manager mediates the transfer. The `sel_config` file controls file relabeling actions, and cutting and copying information to a different label. The `/usr/dt/bin/sel_mgr` application controls dragging and dropping between windows. As the following tables illustrate, the `sel_mgr` is more restrictive than the `sel_config` file.

The following table summarizes the rules for file relabeling. The rules cover copying, cutting, pasting, dragging, and dropping files.

TABLE 4–1 Conditions for Moving Files to a New Label

Transaction Description	Label Relationship	Owner Relationship	Required Authorization
Copy/Cut and paste, or drag and drop of files between File Managers	Same label	Same UID	None
	Downgrade	Same UID	`solaris.label.file.downgrade`
	Upgrade	Same UID	`solaris.label.file.upgrade`
	Downgrade	Different UIDs	`solaris.label.file.downgrade`
	Upgrade	Different UIDs	`solaris.label.file.upgrade`

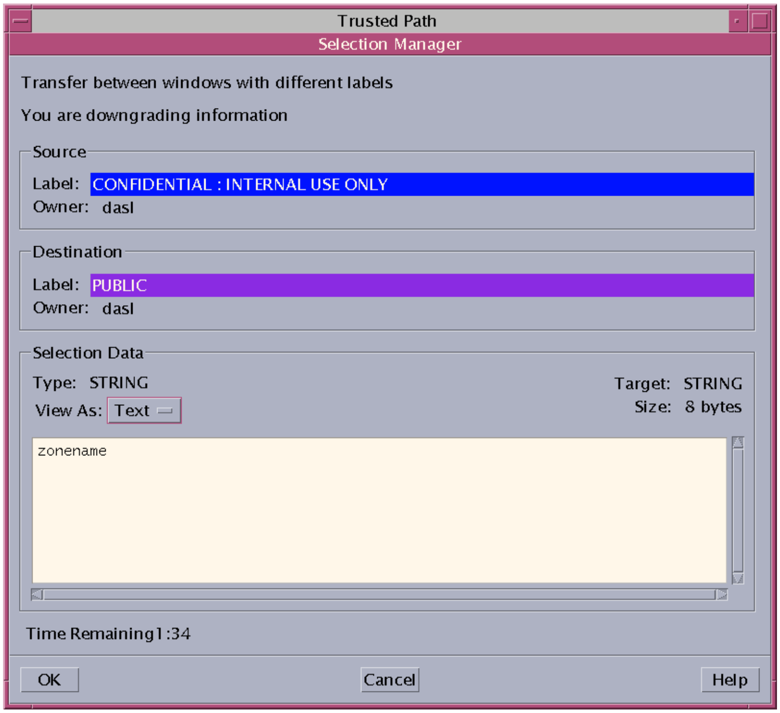
Different rules apply to selections within a window or file. Drag and drop of *selections* always requires equality of labels and ownership. Drag and drop between windows is mediated by the `/usr/dt/bin/sel_mgr` application, not by the `sel_config` file.

The rules for changing the label of selections are summarized in the following table.

TABLE 4–2 Conditions for Moving Selections to a New Label

Transaction Description	Label Relationship	Owner Relationship	Required Authorization
Copy/Cut and paste of selections between windows	Same label	Same UID	None
	Downgrade	Same UID	`solaris.label.win.downgrade`
	Upgrade	Same UID	`solaris.label.win.upgrade`
	Downgrade	Different UIDs	`solaris.label.win.downgrade`
	Upgrade	Different UIDs	`solaris.label.win.upgrade`
Drag and drop of selections between windows	Same label always required	Same UID always required	None applicable

Trusted Extensions provides a selection confirmer to mediate label changes. This window appears when an authorized user attempts to change the label of a file or selection. The user has 120 seconds to confirm the operation. To change the security level of data without the selection confirmer requires the `solaris.label.win.noview` authorization, in addition to the relabeling authorizations.



Trusted Path

Selection Manager

Transfer between windows with different labels

You are downgrading information

Source

Label: CONFIDENTIAL : INTERNAL USE ONLY

Owner: dasl

Destination

Label: PUBLIC

Owner: dasl

Selection Data

Type: STRING

View As: Text

Target: STRING

Size: 8 bytes

```
zonename
```

Time Remaining1:34

By default, the selection confirmer displays whenever data is being transferred to a different label. If a selection requires several transfer decisions, the automatic reply mechanism provides a way to reply once to the several transfers. For more information, see the `sel_config`(4) man page, and the following section.

`sel_config` File

The `sel_config` file is consulted to determine the behavior of the selection confirmer when an operation would upgrade or downgrade a label.

The `/usr/dt/config/sel_config` file defines the following:

- A list of selection types to which automatic replies are given
- Whether certain types of operation should be automatically confirmed
- Whether a selection confirmer dialog should be displayed

In CDE, the Security Administrator role can change the defaults by using the Configure Selection Confirmation action in the Trusted_Extensions folder. The new settings become effective at the next login. If you are in Java DS when modifying the file, do not use the CDE action. Copy the `sel_config` file to the `/etc/dt/config` directory. Then, customize that copy as you would customize any other CDE configuration file.

Customization of Solaris Trusted Extensions (CDE)

In Solaris Trusted Extensions (CDE), users can add actions to the Front Panel and customize the Workspace menu. Trusted Extensions software limits users' ability to add programs and commands to CDE.

Workspace Menu Customization

The Workspace Menu is the menu that appears when you click mouse button 3 on the background of the workspace. Ordinary users can customize the menu, and add items to the menu.

The following conditions apply when a user is allowed to work at multiple labels:

- The user must have a home directory in the global zone.
- The user must use the Customize Menu and Add Item to Menu options in an ordinary user workspace. The user can create a different customization per label.
- When the user assumes a role, changes to the Workspace Menu persist.
- Changes that are made to the Workspace Menu are stored in the user's home directory at the current label. The customized menu file is `.dt/wsmenu`.
- The user's rights profile must enable the user to run the action.

  Any option that is added to the Workspace Menu must be handled by one of the user's rights profiles. Otherwise, the option will fail when invoked and an error message is displayed.

  For example, anyone with the Run action can double-click the icon for any executable and run it, even if the action or any commands that the action invokes are not in one of the account's rights profiles. By default, roles do not have the Run action. Therefore, any item that requires the Run action fails when executed by a role.

Front Panel Customization

Anyone can drag and drop a pre-existing action from the Application Manager to the Front Panel as long as the account doing the modification has the action in its profile. Actions in the `/usr/dt/` or `/etc/dt/` directories can be added to the Front Panel, but applications in the `$HOME/.dt/appconfig` directories cannot. While users can use the Create Action action, they cannot write into any of the directories where the system-wide actions are stored, so they cannot use the actions.

In Trusted Extensions, the actions' search path has been changed. Actions in any individual's home directory are processed last instead of first. Therefore, no one can customize existing actions.

The Security Administrator role has the Admin Editor action, so can make any needed modifications to the `/usr/dt/appconfig/types/C/dtwm.fp` file and the other configuration files for the Front Panel subpanels.

◆ ◆ ◆ CHAPTER 5

Administering Security Requirements in Trusted Extensions

This chapter contains tasks that are commonly performed on a system that is configured with Trusted Extensions.

Common Tasks in Trusted Extensions (Tasks)

The following procedures include procedures that set up a working environment for administrators of Trusted Extensions.

Task	Description	For Instructions
Change the workspace menu	Customize the Workspace Menu at your session clearance.	"How to Customize the Workspace Menu" on page 54
Save your desktop configuration	Configure the windows in your user workspaces, and save the layout.	"How to Save Your Desktop Layout in CDE" on page 55
Work at a different label	Find or create a workspace at a specific label.	"How to Work at a Different Label" on page 56
	Keep your current workspace, and add a new workspace at a different label.	"How to Add a Workspace at a Particular Label" on page 55
	Relabel your current workspace.	"How to Assign a Different Label to a Workspace" on page 56
Change the editor program for the Admin Editor	Specify the editor that the Admin Editor action opens.	"How to Assign the Editor of Your Choice as the Admin Editor" on page 57
Change the password for root	Specify a new password for the `root` user, or for the `root` role.	"How to Change the Password for `root`" on page 58
Change the password for a role	Specify a new password for your current role.	Example 5–2

Task	Description	For Instructions
Determine the hexadecimal number for a label	Find the internal representation for a text label.	"How to Get the Hexadecimal Equivalent for a Label" on page 58
Determine the text representation for a label	Display the text representation for a hexadecimal label.	"How to Get a Readable Label From Its Hexadecimal Form" on page 59
Edit system files	Administer Solaris or Trusted Extensions system files.	"How to Change Security Defaults in System Files" on page 60
Allocate a device	Use a device to add information to or remove information from the system.	"How to Allocate a Device" in *Solaris Trusted Extensions User's Guide*
Administer remotely	Administer Solaris or Trusted Extensions hosts from a remote host.	Chapter 8

▼ How to Customize the Workspace Menu

Note – In CDE, users and roles can customize the Workspace Menu for each distinct label.

1. **In a role workspace, start to customize the menu.**

   - **To add one or more items to the menu, choose the Add Item to Menu ... item.**

     A dialog box with a Browse button appears.

   - **To modify the menu or menu properties, choose Customize Menu ... item.**

     A File Manager appears.

2. **If you are adding items, do the following:**

   a. **Find your program.**

   The Browse button shows the files that are available for this workspace at this label.

   b. **Select the program.**

   c. **Repeat for all programs that you want to add to the menu.**

   d. **Close the window.**

   The items are added to the top of the menu.

3 **If you are modifying the menu, do the following:**

- **To remove menu items, use mouse button 3 over the item.**

  Select Put in Trash.

- **To change properties, such as permissions, use mouse button 3 over the item.**

  Select Properties. Modify permissions. You can also view file information and file sensitivity label.

4 **Confirm the menu changes or cancel.**

- **To cancel your changes, choose File –> Close.**

- **To confirm your changes, choose File –> Update Workspace Menu,**

  The Workspace Menu reflects your changes.

▼ How to Save Your Desktop Layout in CDE

Users can save desktop configurations. Roles cannot save desktop configurations.

1 **On your CDE desktop, lay out your workspaces.**

2 **Open the Style Manager from the Workspace Menu or from the Front Panel.**

The Style Manager requires the trusted path. Therefore, do not run it from the Application Manager.

3 **Save the Startup.**

▼ How to Add a Workspace at a Particular Label

You can add a workspace at your minimum label, or at the label of an existing workspace.

1 **To create a workspace at your minimum label, do the following:**

- **Click mouse button 3 over the Workspace Switch Area.**

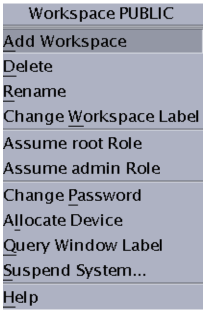
- **Choose Add Workspace.**

  The workspace is created at your minimum label.

2 **To create a workspace at the label of an existing workspace, do the following:**

- **Click mouse button 3 over the workspace button.**

- **Choose Add Workspace.**



The workspace is created at the label of the workspace button.

How to Assign a Different Label to a Workspace

Before You Begin You must be logged in to a multilevel session.

1. **Click mouse button 3 over the workspace button.**

2. **Choose Change Workspace Label.**

3. **Choose a label from the label builder.**

   The workspace label is changed to the new label. Windows and applications that were invoked before the label change continue to run at the previous label.

How to Work at a Different Label

The ability to set workspace labels in Trusted Extensions provides a convenient means of working at different labels within the same session.

Before You Begin You must be logged in to a multilevel session.

1. **To work in the same workspace, see "How to Assign a Different Label to a Workspace" on page 56.**

2. **To work in a different workspace, see "How to Add a Workspace at a Particular Label" on page 55.**

▼ How to Assign the Editor of Your Choice as the Admin Editor

The Admin Editor action uses the value of the `$EDITOR` environment variable as its editor.

Before You Begin You must be in a role in the global zone.

1 **Determine the value of the `$EDITOR` variable.**

```
# echo $EDITOR
```

The following are editor possibilities. The `$EDITOR` variable might also not be set.

- `/usr/dt/bin/dtpad` – Is the editor that CDE provides.
- `/usr/bin/gedit` – Is the editor that Java DS or GNOME provides.
- `/usr/bin/vi` – Is the visual editor.

2 **Set the value of the `$EDITOR` variable.**

- **To set the value permanently, modify the value in the shell initialization file for the role.**
- **To set the value for the current shell, set the value in the terminal window.**

  For example, in a Korn shell, use the following commands:

  ```
  # setenv EDITOR=pathname-of-editor
  # export $EDITOR
  ```

  In a C shell, use the following command:

  ```
  # setenv EDITOR=pathname-of-editor
  ```

Example 5–1 Specifying the Editor for the Admin Editor Action

The Security Administrator role wants to use `vi` when editing system files. The user who has assumed the role modifies the `.kshrc` initialization file in the role's home directory.

```
$ cd /home/secadmin
$ vi .kshrc

# Interactive shell
set -o vi
...
export EDITOR=vi
```

The next time that any user assumes the Security Administrator role, `vi` is the Admin Editor.

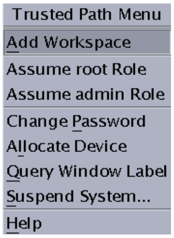
▼ How to Change the Password for `root`

The Security Administrator role can change any account's password at any time by using the Solaris Management Console. However, the Solaris Management Console cannot change the password of a "system account". A system account is an account whose UID is below 100. `root` is a system account because its UID is 0.

1 **Become root.**

If your site has made superuser into the `root` role, assume the `root` role.

2 **Choose Change Password from the Trusted Path menu.**



3 **Change the password and confirm the change.**

Example 5–2 Changing the Password for a Role

A role that is defined in LDAP uses the Trusted Path menu to change the password for the role. The password is then changed in LDAP for all users who attempt to assume the role.

▼ How to Get the Hexadecimal Equivalent for a Label

This procedure provides an internal hex representation of a label. This representation is safe for storing in a public directory. For more information, see the `atohexlabel`(1M) man page.

Before You Begin You must be in the Security Administrator role in the global zone. For details, see "How to Enter the Global Zone in Trusted Extensions" on page 41.

◗ **To get a hexadecimal value for a label, do one of the following.**

- **To get the hexadecimal value for a sensitivity label, pass the label to the command.**

```
$ atohexlabel "CONFIDENTIAL : NEED TO KNOW"
0x0004-08-68
```

- **To get the hexadecimal value for a clearance, use the `-c` option.**

  ```
  $ atohexlabel -c "CONFIDENTIAL NEED TO KNOW"
  0x0004-08-68
  ```

Example 5–3 Using the `atohexlabel` Command

When you pass a valid label in hexadecimal format, the command returns the argument.

```
$ atohexlabel 0x0004-08-68
0x0004-08-68
```

When you pass an administrative label, the command returns the argument.

```
$ atohexlabel admin_high
ADMIN_HIGH
atohexlabel admin_low
ADMIN_LOW
```

Troubleshooting The error message `atoxhexlabel parsing error found in <string> at position 0` indicates that the `<string>` argument that you passed to `atohexlabel` was not a valid label or clearance. Check your typing, and check that the label exists in your installed `label_encodings` file.

▼ How to Get a Readable Label From Its Hexadecimal Form

This procedure provides a way to repair labels that are stored in internal databases. For more information, see the `hextoalabel`(1M) man page.

Before You Begin You must be in the Security Administrator role in the global zone.

◗ **To get the text equivalent for an internal representation of a label, do one of the following.**

- **To get the text equivalent for a sensitivity label, pass the hexadecimal form of the label.**

  ```
  $ hextoalabel 0x0004-08-68
  CONFIDENTIAL : NEED TO KNOW
  ```

- **To get the text equivalent for a clearance, use the `-c` option.**

  ```
  $ hextoalabel -c 0x0004-08-68
  CONFIDENTIAL NEED TO KNOW
  ```

▼ How to Change Security Defaults in System Files

In Trusted Extensions, the security administrator changes or accesses default security settings on a computer.

Files in the `/etc/security` and `/etc/default` directories contain security settings. On a Solaris system, superuser can edit these files. For Solaris security information, see Chapter 3, "Controlling Access to Systems (Tasks)," in *System Administration Guide: Security Services*.

Caution – Relax system security defaults only if site security policy allows you to.

Before You Begin You must be in the Security Administrator role in the global zone.

◗ **Use the Admin Editor action to edit the file.**

For details, see "How to Edit Administrative Files in Trusted Extensions" on page 44.

File	Task	Further Information
`/etc/default/login`	Reduce the allowed number of password tries	Example under "How to Monitor All Failed Login Attempts" in *System Administration Guide: Security Services*. `passwd`(1) man page.
`/etc/default/kbd`	Disable keyboard shutdown	"How to Disable a System's Abort Sequence" in *System Administration Guide: Security Services*. **Note** – On hosts that are used by administrators for debugging, the default setting for `KEYBOARD_ABORT` allows access to the `kadb` kernel debugger. For more on the debugger, see the `kadb`(1M) man page.
`/etc/security/policy.conf`	Require a more powerful algorithm for user passwords Remove a basic privilege from all users of this host Restrict users of this host to Basic Solaris User authorizations	`policy.conf`(4) man page.

File	Task	Further Information
`/etc/default/passwd`	Require users to change passwords frequently Require users to create maximally different passwords Require a longer user password Require a password that cannot be found in your dictionary	`passwd`(1) man page.

CHAPTER 6

Users, Rights, and Roles in Trusted Extensions

This chapter describes essential decisions to make before creating regular users, and provides additional background for managing user accounts. The chapter assumes that the install team has set up roles, and has set up a limited number of user accounts. These users can assume the roles that configure and administer Trusted Extensions. For details, see *Solaris Trusted Extensions Installation and Configuration*.

User Security Features in Trusted Extensions

Trusted Extensions software adds the following security features to users, roles, or profiles:

- A user has a label range within which the user can use the system.
- A role has a label range within which the role can perform its administrative tasks.
- A Trusted Extensions rights profile can include actions. Like commands, actions can have security attributes.
- Commands and actions in a Trusted Extensions rights profile have a label attribute. The command or action must be performed within a label range, or at a particular label.
- Trusted Extensions software adds privileges and authorizations to the set of authorizations and privileges that are defined by the Solaris OS.

Administrator Responsibilities for Users

The System Administrator role creates user accounts. The Security Administrator role sets up the security aspects of an account.

If you are using the Sun Java™ System Directory Server for the LDAP naming service, check that the install team configured the `tsol_ldap.tbx` toolbox. For the procedure, see "Configuring the Solaris Management Console for LDAP (Tasks)" in *Solaris Trusted Extensions Installation and Configuration*.

For details on setting up users and roles, use the following references.

- *System Administration Guide: Basic Administration*
- "Setting Up User Accounts (Task Map)" in *System Administration Guide: Basic Administration*
- Part III, "Roles, Rights Profiles, and Privileges," in *System Administration Guide: Security Services*

System Administrator Responsibilities for Users

In Trusted Extensions, the System Administrator role is responsible for determining who can access to the system. The System Administrator is responsible for the following tasks:

- Adding and deleting users
- Adding and deleting roles
- Modifying user and role configurations, other than security attributes

Security Administrator Responsibilities for Users

In Trusted Extensions, the Security Administrator role is responsible for all security attributes of a user or role. The Security Administrator is responsible for the following tasks:

- Assigning and modifying the security attributes of a user, a role, or a profile
- Creating and modifying rights profiles
- Assigning rights profiles to a user or role
- Assigning privileges to a user, a role, or a profile
- Assigning authorizations to a user, a role, or a profile
- Removing privileges from a user, a role, or a profile
- Removing authorizations from a user, a role, or a profile

In general, the Security Administrator creates rights profiles. However, if a profile needs capabilities that the Security Administrator role cannot grant, then superuser or the Primary Administrator role creates the profile.

Before creating a rights profile, the security administrator should analyze whether any of the commands or actions in the new profile need privilege or authorization to be successful. The man pages for individual commands list the privileges and authorizations that a command might need. For examples of actions that require privileges and authorizations, see the `exec_attr` database.

Decisions to Make Before Creating Users

The following decisions and setup affect what users are able to do in Trusted Extensions and how much effort is required. Some decisions are the same as the decisions that you would make when installing the Solaris OS. However, decisions that are specific to Trusted Extensions can affect site security and ease of use.

- Decide whether to change default user security attributes in the `policy.conf` file. User defaults in the `label_encodings` file were configured by the install team. For a description of the defaults, see "Default User Security Attributes" on page 65.
- Decide which startup files, if any, should be copied from or linked from each user's minimum-label home directory to the user's higher-level home directories. For the procedure, see "How to Configure Startup Files for Users" on page 74.
- Decide if users can access peripheral devices, such as the microphone, CD-ROM drive, and JAZ drive.

  If access is permitted to some users, decide if your site requires additional authorizations to satisfy site security. For the default list of device-related authorizations, see Example 17–8. For a finer-grained set of device authorizations, see "Customizing Device Authorizations in Trusted Extensions (Tasks)" on page 199.

Default User Security Attributes

Settings in the `label_encodings` and the `policy.conf` files together define default security attributes for user accounts. The values that you set explicitly for a user override these system values. Some of the values that are set in these files also apply to role accounts. For security attributes that you can explicitly set, see "Configurable User Attributes" on page 66.

Label Encodings File Defaults

The `label_encodings` file defines a user's minimum label, clearance, and default label view. For details about the file, see the `label_encodings`(4) man page. Your site's `label_encodings` file was installed by your install team. Their decisions were based on "Devising a Label Strategy" in *Solaris Trusted Extensions Installation and Configuration*, and examples from *Solaris Trusted Extensions Label Administration*.

Label values that the Security Administrator explicitly sets for individual users in the Solaris Management Console are derived from the `label_encodings` file. User values override the values in the `label_encodings` file.

`policy.conf` File Defaults

The Solaris `/etc/security/policy.conf` file contains the default security settings for the system. In addition, Trusted Extensions provides two keywords. You can add these keyword-value pairs to the file if you want to change the system-wide value. These keywords are enforced by CDE.

TABLE 6–1 Trusted Extensions Security Defaults in `policy.conf` File

Keyword	Default Value	Possible Values	Notes
IDLECMD	LOCK	LOCK | LOGOUT	Does not apply to roles.
IDLETIME	30	0 to 120 minutes	Does not apply to roles.

The authorizations and rights profiles that are defined in the `policy.conf` file are *in addition* to any authorizations and profiles that are assigned to individual accounts. For the other fields, the individual user's value overrides the system value.

"Planning User Security" in *Solaris Trusted Extensions Installation and Configuration* includes a table of every `policy.conf` keyword. See also the `policy.conf`(4) man page.

Configurable User Attributes

The Solaris Management Console 2.1 is your tool for creating and modifying user accounts. For multilevel users, you might also want to set up `.copy_files` and `.link_files` files in each user's minimum label home directory.

The User Accounts tool in the Solaris Management Console works as it does in the Solaris OS. There are two exceptions:

- Trusted Extensions adds attributes to user accounts.
- Home directory server access does not work as smoothly in Trusted Extensions as it does in the Solaris OS.
  - You create the home directory server entry the same as you do on a Solaris system.
  - Then, you do additional steps to mount the home directory at every user label.

As described in "How to Add a User With the Solaris Management Console's Users Tool" in *System Administration Guide: Basic Administration*, a Wizard enables you to create user accounts quickly. After using the Wizard, each account's security information must be entered.

For more information about `.copy_files` and `.link_files` files, see "`.copy_files` and `.link_files` Files" on page 68.

User Attributes That Are Assigned After Creation

Security information must still be entered by the Security Administrator, as the following table shows. For information about the files that contain default values, see "Default User Security Attributes" on page 65.

TABLE 6–2 User Security Attributes That You Assign After Creation

User Attribute	Location of Default Value	Is Action Required	Effect of Action
Password	None	Required	User has password
Roles	None	Optional	User can assume a role
Authorizations	`policy.conf` file	Optional	User has additional authorizations
Rights Profiles	`policy.conf` file	Optional	User has additional rights profiles
Labels	`label_encodings` file	Optional	User has different default label or accreditation range
Privileges	`policy.conf` file	Optional	User has different set of privileges
Account Usage	`policy.conf` file	Optional	User has different setting for computer when it is idle
Audit	`audit_control` file	Optional	User is audited differently from the system audit settings

Security Attribute Assignment to Users

The Security Administrator role assigns security attributes to users in the Solaris Management Console after the user is created. If the administrator has set up correct defaults, assigning security attributes is needed only for users who are exceptions to the defaults.

Assigning Passwords

The Security Administrator role assigns passwords to users after the user has been created. The password is created by the administrator. After assignment, users can change their passwords.

As in the Solaris OS, users can be forced to change their passwords at regular intervals. The password aging options limit how long any intruder who is able to guess or steal a password could potentially access the system. Establishing a minimum length of time to elapse before change also prevents a user with a new password from reverting immediately to the old password. For details, see the `passwd(1)` man page.

Note – The passwords for users who can assume roles should not be subject to any password aging constraints.

Assigning Roles

A user is not required to have a role. A single user can be assigned more than one role if that assignment is consistent with your site's security policy.

Assigning Authorizations

As in the Solaris OS, assigning authorizations directly to a user adds those authorizations to existing authorizations. In Trusted Extensions, you add the authorizations to a rights profile, then assign the profile to the user.

Assigning Rights Profiles

As in the Solaris OS, the order of profiles is important. The profile mechanism uses the first instance of the command or action in an account's profile set.

You can use the sorting order of profiles to your advantage. If you want a command to run with different security attributes from those defined for it in an existing profile, create a new profile with the desired assignments for the command. Then, insert that new profile before the existing profile.

Note – Rights profiles that include administrative actions or administrative commands should not be assigned to an ordinary user account. The profile would not work, because an ordinary user cannot enter the global zone.

Changing Privilege Default

The default privilege set can be too liberal for many sites. To restrict the privilege set for any ordinary user on a system, change the `policy.conf` file setting. To change the privilege set for individual users, use the Solaris Management Console. For an example, see "How to Modify a User's Set of Privileges" on page 81.

Changing Label Defaults

Changing a user's label defaults creates an exceptions to the user defaults in the `label_encodings` file.

Changing Audit Defaults

As in the Solaris OS, assigning audit classes to a user creates exceptions to the audit classes that are assigned in the `/etc/security/audit_control` file on the system. For more about auditing, see Chapter 18.

`.copy_files` and `.link_files` Files

In Trusted Extensions, files are automatically copied from the skeleton directory *only* into the zone that contains the account's minimum label. To ensure that zones at higher labels can use startup files, either the user or the administrator must create the files `.copy_files` and `.link_files`.

The Trusted Extensions files `.copy_files` and `.link_files` help to automate the copying or linking of startup files into every label of an account's home directory. Whenever a user creates a workspace at a new label, the `updatehome` command reads the contents of `.copy_files` and `.link_files` at the account's minimum label. The command then copies or links every listed file into the higher-labeled workspace.

The `.copy_files` file is useful when a user wants a slightly different startup file at different labels. Copying is desirable, for example, when users use different mail aliases at different labels. The `.link-files` file is useful when a startup file should be identical at any label that it is invoked. Linking is desirable, for example, when using one printer for all labeled print jobs. For sample files, see "How to Configure Startup Files for Users" on page 74.

The following is a list of some startup files that you might want users to be able to link or to copy to higher labels:

`.acrorc`

`.aliases`

`.cshrc`

`.dtprofile`

`.emacs`

`.login`

`.mailrc`

`.mime_types`

`.newsrc`

`.profile`

`.signature`

`.soffice`

`.Xdefaults`

`.Xdefaults-`*hostname*

CHAPTER 7

Managing Users, Rights, and Roles in Trusted Extensions

This chapter provides the procedures for configuring and managing users, user accounts, and rights profiles.

- "Customizing the User Environment for Security (Tasks)" on page 71
- "Managing Users and Rights With Solaris Management Console (Tasks)" on page 77

Customizing the User Environment for Security (Tasks)

The following table describes common tasks that are performed when customizing a system for all users, or when customizing an individual user's account.

Task	Description	For Instructions
Change label attributes	Modify label attributes, such as minimum label and default label view, for a user account.	"How to Modify Default User Label Attributes" on page 72
Change Trusted Extensions policy for all users of a system	Log the user out after a set amount of time that the computer is idle.	Example 7–1
Shorten the idle time	Turn on the screensaver after a short amount of time.	Example 7–1
Change every user's privilege set	Remove unnecessary privileges from all ordinary users of a system.	Example 7–2
Hide labels from users	Prevent labels from being visible on a single-label system.	Example 7–3
Remove labels from printer output	Remove labels from printouts at a public kiosk.	Example 7–4
Configure initialization files for users	Configure startup files, such as `.cshrc`, `.copy_files`, and `.soffice` for all users.	"How to Configure Startup Files for Users" on page 74

Task	Description	For Instructions
Lengthen the timeout for file relabeling	Configure some applications to enable authorized users to relabel files.	"How to Lengthen the Timeout When Relabeling Information" on page 76
Log in to a Failsafe Session in CDE	Fix faulty user initialization files	"How to Log In to a Failsafe Session in CDE" on page 77

▼ How to Modify Default User Label Attributes

You can change default user label attributes during the configuration of the first system. The changes must be copied to every machine.

Before You Begin You must be in the Security Administrator role in the global zone. For details, see "How to Enter the Global Zone in Trusted Extensions" on page 41.

1. **Review the default user attribute settings in the `/etc/security/tsol/label_encodings` file.**

   For the defaults, see "Label Encodings File Defaults" on page 65.

2. **Modify the user security attributes in the `label_encodings` file.**

   Use the Edit Label Encodings action. For details, see "How to Launch CDE Administrative Actions in Trusted Extensions" on page 43.

   The `label_encodings` file should be the same on all hosts.

3. **Distribute a copy of the file to every Trusted Extensions host.**

▼ How to Modify `policy.conf` Defaults

Changing the `policy.conf` defaults in Trusted Extensions is similar to changing any security-relevant system file in the Solaris OS. In Trusted Extensions, you use a trusted editor to modify system files.

Before You Begin You must be in the Security Administrator role in the global zone. For details, see "How to Enter the Global Zone in Trusted Extensions" on page 41.

1. **Review the default settings in the `/etc/security/policy.conf` file.**

   Read the file for the settings. For Trusted Extensions keywords, see Table 6–1.

2. **Modify the settings.**

   For a description of the general method for system files, see "How to Change Security Defaults in System Files" on page 60.

Example 7–1 Changing the Computer's Idle Settings

In this example, the security administrator wants idle computers to return to the login screen. The default locks an idle computer. Therefore, the Security Administrator role adds the IDLECMD keyword=value pair to the /etc/security/policy.conf file, as follows:

```
IDLECMD=LOGOUT
```

The administrator also wants computers to be idle a shorter amount of time before logout. Therefore, the Security Administrator role adds the IDLETIME keyword=value pair to the policy.conf file, as follows:

```
IDLETIME=10
```

The computer now logs out the user after the computer is idle for ten minutes.

Example 7–2 Modifying Every User's Basic Privilege Set

In this example, the security administrator of a Sun Ray™ installation does not want ordinary users to see the processes of other Sun Ray users. Therefore, on every system that is configured with Trusted Extensions, the administrator removes proc_info from the basic set of privileges. The PRIV_DEFAULT setting in the /etc/policy.conf file is modified as follows:

```
PRIV_DEFAULT=basic,!proc_info
```

Example 7–3 Hiding Labels on a System

In this example, the security administrator changes the default setting in a system's policy.conf file to hide labels. Any user on this system would not see labels, unless the user was specifically configured to be able to see labels. This setting is reasonable on a single-label system, or on a system that is available to the general public.

```
# /etc/security/policy.conf
...
LABELVIEW=hidesl
```

To configure a user to override this setting, see "How to Hide Labels From a User" on page 83.

Example 7–4 Assigning Printing–Related Authorizations to All Users of a Computer

In this example, the security administrator enabled a public kiosk computer to print without labels by typing AUTHS_GRANTED= solaris.print.unlabeled in the computer's /etc/security/policy.conf file. At the next boot, print jobs by all users of this kiosk printed without page labels.

The administrator then decided to save paper by removing banner and trailer pages. She first ensured that the Always Print Banners checkbox in the Print Manager was not marked with a check.

She then modified the `policy.conf` entry to read: `AUTHS_GRANTED=` `solaris.print.unlabeled,solaris.print.nobanner` and rebooted. Now, all print jobs are unlabeled, and have no banner or trailer pages.

▼ How to Configure Startup Files for Users

Users can put a `.copy_files` file and `.link_files` file into their home directory at the label that corresponds to their minimum sensitivity label. Users can also modify the existing `.copy_files` and `.link_files` files at the users' minimum label. This procedure is for the administrator role to automate the setup for a site.

Before You Begin You must be in the System Administrator role in the global zone. For details, see "How to Enter the Global Zone in Trusted Extensions" on page 41.

1. **Create two Trusted Extensions startup files.**

   You are going to add `.copy_files` and `.link_files` to your list of startup files.

2. **Customize the `.copy_files` file.**

   a. **Use the Admin Editor.**

      For details, see "How to Edit Administrative Files in Trusted Extensions" on page 44.

   b. **Type the full pathname to the `.copy_files` file in the Admin Editor.**

   c. **Type into `.copy_files`, one file per line, the files to be copied into the user's home directory at all labels.**

      Use "`.copy_files` and `.link_files` Files" on page 68 for ideas. For sample files, see Example 7–5.

3. **Customize the `.link_files` file.**

   Follow the steps for `.copy_files`. In `.link_files`, type the files to be linked from the user's minimum-label home directory to higher labels.

4. **Customize the other startup files for your users.**

   - For a discussion of what to include in startup files, see "Customizing a User's Work Environment" in *System Administration Guide: Basic Administration*.
   - For details, see "How to Customize User Initialization Files" in *System Administration Guide: Basic Administration*.
   - For an example, see Example 7–5.

5. **(Optional) Create a `skelP` subdirectory for users whose default shell is a profile shell.**

6 **Copy the customized startup files into the appropriate skeleton directory.**

7 **Use the appropriate `skel`*X* pathname when you create the user.**

Example 7–5 Customizing Startup Files for Users

In this example, the security administrator configured files for every user's home directory. The files were in place before any user logged in. The files are at the user's minimum label. At this site, the users' default shell is the C shell.

The security administrator created a `.copy_files` and a `.link_files` file.

```
# .copy_files for regular users
# Copy these files to my home directory in every zone
.mailrc
.mozilla
.soffice
:wq

# .link_files for regular users with C shells
# Link these files to my home directory in every zone
.cshrc
.login
.Xdefaults
.Xdefaults-hostname
:wq

# .link_files for regular users with Korn shells
# Link these files to my home directory in every zone
.ksh
.profile
.Xdefaults
.Xdefaults-hostname
:wq
```

In the shell initialization files, the administrator ensured that the users' print jobs would go to a labeled printer.

```
# .cshrc file
setenv PRINTER conf-printer1
setenv LPDEST  conf-printer1

# .ksh file
export PRINTER conf-printer1
export LPDEST  conf-printer1
```

The administrator modified the `.Xdefaults-`*home-directory-server* file to force the `dtterm` command to source the `.profile` file for a new terminal.

```
# Xdefaults-HDserver
Dtterm*LoginShell: true
```

The customized files were copied to the appropriate skeleton directory.

```
$ cp .copy_files .link_files .cshrc .login .profile \
.mailrc .Xdefaults .Xdefaults-home-directory-server \
/etc/skelC
$ cp .copy_files .link_files .ksh .profile \
.mailrc .Xdefaults .Xdefaults-home-directory-server \
/etc/skelK
```

▼ How to Lengthen the Timeout When Relabeling Information

In Trusted Extensions, a Selection Manager mediates transfers of information between labels. The Selection Manager appears for drag and drop operations, and for cut and paste operations. Some applications require that you set a suitable timeout so that the Selection Manager has time to intervene. A value of 2 minutes is sufficient.

Note – Do not change the default timeout value on an unlabeled system.

Before You Begin You must be in the System Administrator role in the global zone. For details, see "How to Enter the Global Zone in Trusted Extensions" on page 41.

1 For the StarOffice™ or OpenOffice application, do the following:

a. Navigate to the file *office-installation-directory*/`VCL.xcu`**.**

where *office-install-directory* is the StarOffice or OpenOffice installation directory, *office-top-dir*/`share/registry/data/org/openoffice`

b. Change the `SelectionTimeout` **property value to 120.**

Use the Admin Editor action. For details, see "How to Edit Administrative Files in Trusted Extensions" on page 44.

The default value is 3 seconds. A value of 120 sets the timeout to 2 minutes.

2 For users of applications that rely on the GNOME ToolKit (GTK) library, do one of the following:

- **Have each user change the selection timeout property value to 2 minutes.**

  Each user would perform the steps that you would perform, as described in the following step.

- **Change the selection timeout property value to 2 minutes.**

  Most Java DS applications use the GTK library. Web browsers such as Mozilla™, Firefox, and Thunderbird use the GTK library.

  By default, the selection timeout value is 300, or 5 seconds. A value of 7200 sets the timeout to 2 minutes.

  a. **Create a GTK startup file.**

     Name the file `.gtkrc-mine`. The `.gtkrc-mine` file belongs in the user's home directory at the minimum label.

  b. **Add the selection timeout value to the file.**

     ```
     # .gtkrc-mine
     *gtk-selection-timeout: 7200
     ```

     As in the Solaris OS, the `gnome-settings-daemon` reads this file on startup.

- **(Optional) Add the `.gtkrc-mine` file to the list in each user's `.link_files` file.**

  For details, see "How to Configure Startup Files for Users" on page 74.

▼ How to Log In to a Failsafe Session in CDE

In Trusted Extensions, failsafe login is protected. If an ordinary user has customized shell initialization files and now cannot log in, you can use Failsafe Login to fix the user's files.

Before You Begin You must know the root password.

1. **As in the Solaris OS, choose Options –> Failsafe Session on the login screen.**

2. **At the prompt, have the user provide the user's name and password.**

3. **At the prompt for the root password, provide the password for `root`.**

   You can now debug the user's initialization files.

Managing Users and Rights With Solaris Management Console (Tasks)

In Trusted Extensions, you must use the Solaris Management Console to administer users, authorizations, rights, and roles. To manage users and their security attributes, assume the Security Administrator role.

Task	Description	For Instructions
Change a user's label range	Modify the labels at which a user can work. This task constrains the user to a different range within the range that the `label_encodings` file permits.	"How to Modify a User's Label Range in the Solaris Management Console" on page 78
Create an authorizations profile	Several authorizations exist that might be useful for ordinary users. Create a profile for users who qualify to have these authorizations.	"How to Create a Convenient Authorizations Rights Profile" on page 79
Modify a user's default privilege set.	Remove a privilege from the user's basic set.	"How to Modify a User's Set of Privileges" on page 81
Prevent account locking for particular users	Users who can assume a role should have account locking turned off.	"How to Prevent Account Locking for Individuals" on page 82
Hide labels on a user's screen	On a single-label system, you might want a user to not pay attention to labels.	"How to Hide Labels From a User" on page 83
Change view of `ADMIN_HIGH` and `ADMIN_LOW` for ordinary users	On a single-label system, you might want a user to not pay attention to labels.	"How to Change the Administrative Label View for a User" on page 83
Enable a user to relabel data	Authorize a user to downgrade information or upgrade information.	"How to Enable a User to Change the Security Level of Data" on page 84
Remove a user	Remove a user from the system.	"How to Remove a User" on page 85
Handle other tasks	Use the Solaris Management Console to handle tasks that are not specific to Trusted Extensions.	"How to Handle Other Tasks in the Solaris Management Console" on page 85

▼ How to Modify a User's Label Range in the Solaris Management Console

You might want to add to a user's label range to give the user read access to an administrative application. A user who can log into the global zone can run, for example, the Solaris Management Console. The user would not be able to change the contents, but could view the contents.

Alternatively, you might want to restrict the user's label range. A guest user, for example, might be limited to one label.

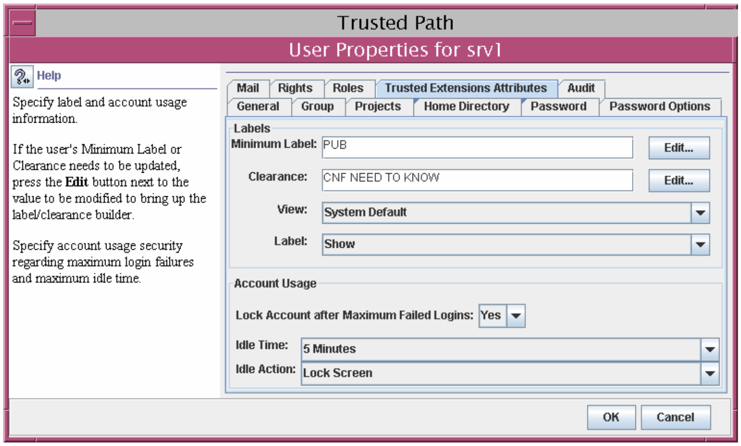
Before You Begin You must be in the Security Administrator role in the global zone.

1. **Open a Trusted Extensions toolbox in the Solaris Management Console.**

   Use the toolbox of the appropriate scope. For details, see "How to Launch the Solaris Management Console" on page 42.

2. **Select the individual user from User Accounts.**

3. **Click the Trusted Extensions Attributes tab.**



Specify label and account usage information.

If the user's Minimum Label or Clearance needs to be updated, press the **Edit** button next to the value to be modified to bring up the label/clearance builder.

Specify account usage security regarding maximum login failures and maximum idle time.

Labels

Minimum Label: PUB

Clearance: CNF NEED TO KNOW

View: System Default

Label: Show

Account Usage

Lock Account after Maximum Failed Logins: Yes

Idle Time: 5 Minutes

Idle Action: Lock Screen

- **To extend the user's label range, choose a higher clearance.**

  You can also lower the minimum label.

- **To restrict the label range to one label, make the clearance equal to the minimum label.**

4 **Click OK to save the changes.**

▼ How to Create a Convenient Authorizations Rights Profile

Where site security policy permits, you might want to create a rights profile that contains authorizations for users who can do limited administrative tasks that require authorization. To enable every user of a particular system to be authorized, see "How to Modify `policy.conf` Defaults" on page 72.

Before You Begin You must be in the Security Administrator role in the global zone.

1 **Open a Trusted Extensions toolbox in the Solaris Management Console.**

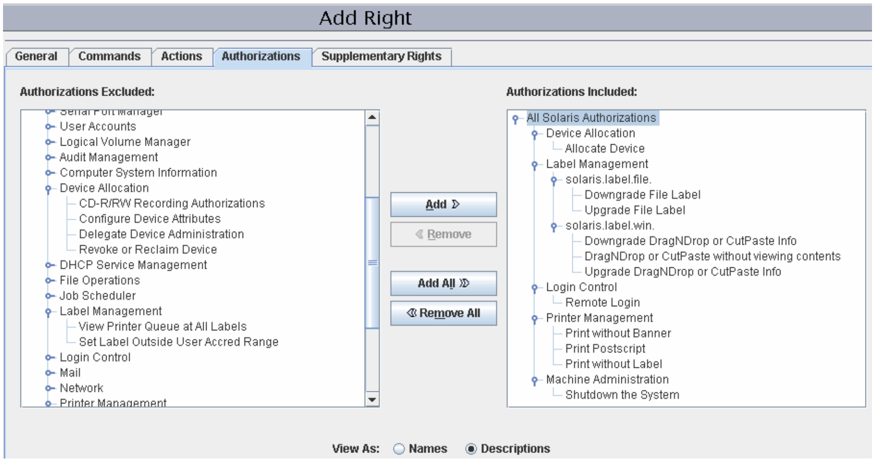
Use the toolbox of the appropriate scope. For details, see "How to Launch the Solaris Management Console" on page 42.

2 **Under System Configuration, select Rights.**

3 **Create a rights profile that contains one or more of the following authorizations.**

For a step by step procedure, see "How to Create or Change a Rights Profile" in *System Administration Guide: Security Services*.

By default, Solaris users can read and write to a CD-ROM. However, in Trusted Extensions, only users who can allocate a device can access the CD-ROM drive. To allocate the drive for use requires authorization. Therefore, to read and write to a CD-ROM in Trusted Extensions, a user needs the Allocate Device authorization.



Authorizations Excluded:

- Serial Port Manager
- User Accounts
- Logical Volume Manager
- Audit Management
- Computer System Information
- Device Allocation
  - CD-R/RW Recording Authorizations
  - Configure Device Attributes
  - Delegate Device Administration
  - Revoke or Reclaim Device
- DHCP Service Management
- File Operations
- Job Scheduler
- Label Management
  - View Printer Queue at All Labels
  - Set Label Outside User Accred Range
- Login Control
- Mail
- Network
- Printer Management

Authorizations Included:

- All Solaris Authorizations
  - Device Allocation
    - Allocate Device
  - Label Management
    - solaris.label.file.
      - Downgrade File Label
      - Upgrade File Label
    - solaris.label.win.
      - Downgrade DragNDrop or CutPaste Info
      - DragNDrop or CutPaste without viewing contents
      - Upgrade DragNDrop or CutPaste Info
  - Login Control
    - Remote Login
  - Printer Management
    - Print without Banner
    - Print Postscript
    - Print without Label
  - Machine Administration
    - Shutdown the System

View As: ○ Names ● Descriptions

- Allocate Device - Authorizes a user to allocate a peripheral device, such as a microphone.
- Downgrade DragNDrop or CutPaste Info - Authorizes a user to select information from a higher-level file and place the information in a lower-level file.
- Downgrade File Label - Authorizes a user to lower the security level of a file
- DragNDrop or CutPaste without viewing contents - Authorizes a user to move information without viewing the information that is being moved.
- Print Postscript - Authorizes a user to print PostScript files.
- Print without Banner - Authorizes a user to print hardcopy without a banner page.
- Print without Label - Authorizes a user to print hardcopy that does not display labels.
- Remote Login - Authorizes a user to remotely log in.
- Shutdown - Authorizes a user to shut down the computer and to shut down a zone.
- Upgrade DragNDrop or CutPaste Info - Authorizes a user to select information from a lower-level file and place the information in a higher-level file.
- Upgrade File Label - Authorizes a user to heighten the security level of a file.

4 **Assign the profile to a user, or to a role.**

For assistance, follow the online help. For a step by step procedure, see "How to Change the RBAC Properties of a User" in *System Administration Guide: Security Services*.

Example 7–6 Assigning a Printing-Related Authorization to a Role

In the following example, the Security Administrator allows a role to print jobs without labels on body pages.

On the Solaris Management Console, navigate to Administrative Roles. Make sure that the desired print-related authorization is contained in one of the role's rights profiles.

▼ How to Modify a User's Set of Privileges

Site security might require that users be permitted fewer privileges than users are assigned by default. For example, at a site that uses Trusted Extensions on Sun Rays, you might want to prevent users from viewing other users' processes on the Sun Ray server.

Before You Begin You must be in the Security Administrator role in the global zone.

1. **Open a Trusted Extensions toolbox in the Solaris Management Console.**

   Use the toolbox of the appropriate scope. For details, see "How to Launch the Solaris Management Console" on page 42.

2. **Under System Configuration, navigate to Users.**

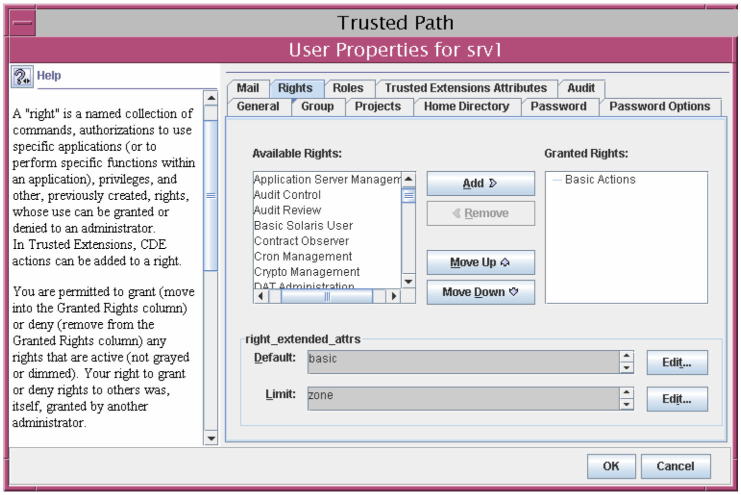
3. **For each user, remove one or more of the privileges in the `basic` set.**

   Caution – Do not remove `proc_exec` or `proc_fork` unless you know exactly why you can do this successfully.

   a. **Double-click the icon for the user.**

   b. **Select the Rights tab.**

   c. **Click the Edit button to the right of the `right_extended_attr` field.**



User Properties for srv1

A "right" is a named collection of commands, authorizations to use specific applications (or to perform specific functions within an application), privileges, and other, previously created, rights, whose use can be granted or denied to an administrator.
In Trusted Extensions, CDE actions can be added to a right.

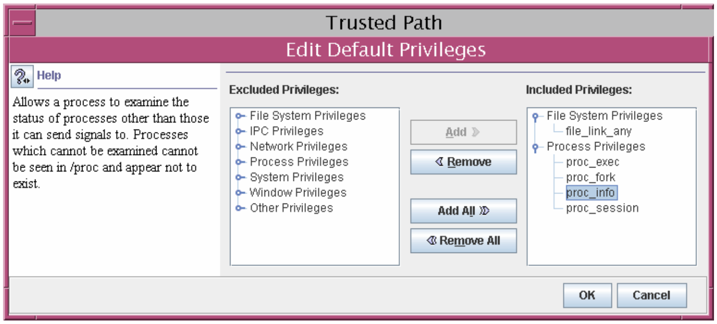
You are permitted to grant (move into the Granted Rights column) or deny (remove from the Granted Rights column) any rights that are active (not grayed or dimmed). Your right to grant or deny rights to others was, itself, granted by another administrator.

Available Rights:
Application Server Managem
Audit Control
Audit Review
Basic Solaris User
Contract Observer
Cron Management
Crypto Management
DAT Administration

Granted Rights:
Basic Actions

right_extended_attrs
Default: basic
Limit: zone

d. **Remove `proc_info` or `proc_sessions` from the user's basic privilege set.**



Edit Default Privileges

Allows a process to examine the status of processes other than those it can send signals to. Processes which cannot be examined cannot be seen in /proc and appear not to exist.

Excluded Privileges:
File System Privileges
IPC Privileges
Network Privileges
Process Privileges
System Privileges
Window Privileges
Other Privileges

Included Privileges:
File System Privileges
file_link_any
Process Privileges
proc_exec
proc_fork
proc_info
proc_session

4. **Save your changes.**

▼ How to Prevent Account Locking for Individuals

Trusted Extensions extends the user security features in the Solaris Management Console to include account locking. Account locking should be turned off for users who can assume a role.

Before You Begin You must be in the Security Administrator role in the global zone.

1. **Bring up the Solaris Management Console.**

   Use the toolbox of the appropriate scope. For details, see "How to Launch the Solaris Management Console" on page 42.

2 **Navigate from System Configuration to the User Accounts tool.**

Type the role password when prompted.

3 **Open the User Properties tool for an individual user.**

4 **Under the Trusted Extensions Attributes tab, prevent account locking.**

In the Account Usage section, select `No` from the pull-down menu next to `Lock account after maximum failed logins`.

▼ How to Hide Labels From a User

Hiding labels is useful at a site where users can work at a single label only. An organization might not want ordinary users to see labels or to be aware of mandatory access controls. Ordinary users can then work whose desktop closely resembles the Java DS or the CDE desktop on a Solaris system.

Before You Begin You must be in the Security Administrator role in the global zone.

1 **Open a Trusted Extensions toolbox in the Solaris Management Console.**

Use the toolbox of the appropriate scope. For details, see "How to Launch the Solaris Management Console" on page 42.

2 **Select the individual user from User Accounts.**

3 **Click the Trusted Extensions Attributes tab.**

4 **Choose Hide from the Label: selection list.**

This setting overrides the value of `LABELVIEW` in the system's `policy.conf` file.

For details, see "Default User Security Attributes" on page 65.

▼ How to Change the Administrative Label View for a User

In Trusted Extensions, users rarely see the administrative label names `ADMIN_HIGH` or `ADMIN_LOW`. Windows that implement security-relevant actions, such as the Device Allocation Manager, display the label `Trusted Path`. Also, Trusted Extensions interfaces such as `getlabel` do not check for label view. Therefore, hiding the names of administrative labels affects very little that an ordinary user can see.

Before You Begin You must be in the Security Administrator role in the global zone.

1 **Open a Trusted Extensions toolbox in the Solaris Management Console.**

Use the toolbox of the appropriate scope. For details, see "How to Launch the Solaris Management Console" on page 42.

2 **Select the individual user from User Accounts.**

3 **Click the Trusted Extensions Attributes tab.**

4 **Choose External from the View: selection list.**

This setting overrides the value of `Default Label View` in the system's `label_encodings` file.

▼ How to Enable a User to Change the Security Level of Data

An ordinary user or a role is given authorization to change the labels of files and directories. The user or role, in addition to having the authorization, must be configured to work at more than one label.

Caution – Changing the security level of data is a privileged operation. This task is for trusted users only.

Before You Begin You must be in the Security Administrator role in the global zone.

1 **Follow the procedure "How to Create a Convenient Authorizations Rights Profile" on page 79 to create a rights profile.**

The following authorizations enable a user to relabel a file.

- Downgrade File Label
- Upgrade File Label

The following authorizations enable a user to relabel information within a file.

- Downgrade DragNDrop or CutPaste Info
- DragNDrop or CutPaste Info Without Viewing
- Upgrade DragNDrop or CutPaste Info

2 **Use the Solaris Management Console to assign the profile to the appropriate users and roles.**

For assistance, follow the online help. For a step by step procedure, see "How to Create or Change a Rights Profile" in *System Administration Guide: Security Services* and "How to Change the RBAC Properties of a User" in *System Administration Guide: Security Services*.

▼ How to Remove a User

When a user is deleted from the system, the administrator must ensure that the user's home directory and any objects owned by that user are also deleted. As an alternative to deleting objects that are owned by the user, the administrator might change the ownership of these objects to a valid user.

The administrator must also ensure that all batch jobs that are associated with the deleted user are also deleted. The administrator must ensure that there are no objects or processes belonging to a deleted user that remain on the system.

Before You Begin You must be in the System Administrator role.

1 **Open a Trusted Extensions toolbox in the Solaris Management Console.**

Use the toolbox of the appropriate scope. For details, see "How to Launch the Solaris Management Console" on page 42.

2 **Navigate to User Accounts.**

a. **Click Users, and provide a password if prompted.**

b. **Double-click User Accounts.**

3 **Select the user to be deleted and click the Delete button.**

Users might have home directories and mail files to delete. These should be archived and be deleted manually.

▼ How to Handle Other Tasks in the Solaris Management Console

Before You Begin You must be superuser, or in a role in the global zone.

◗ **Follow Solaris procedures to handle tasks in the Solaris Management Console.**

Task	For Instructions
Perform administrative tasks by using the Solaris Management Console	Chapter 2, "Working With the Solaris Management Console (Tasks)," in *System Administration Guide: Basic Administration*
Create users	"Using the Solaris Management Tools With RBAC (Task Map)" in *System Administration Guide: Basic Administration*

Task	For Instructions
Create roles	"How to Create and Assign a Role By Using the GUI" in *System Administration Guide: Security Services*
Modify roles	"How to Change the Properties of a Role" in *System Administration Guide: Security Services*
Create or modify a profile	"How to Create or Change a Rights Profile" in *System Administration Guide: Security Services*
Change other security attributes of a user	"How to Change the RBAC Properties of a User" in *System Administration Guide: Security Services*
Audit the actions of a role	"How to Audit Roles" in *System Administration Guide: Security Services*
List the rights profiles by using `smprofile list -D`*name-service-type*`:/`*server-name*`/`*domain-name*	*Chapter 9, "Using Role-Based Access Control (Tasks)," in System Administration Guide: Security Services* or the `smprofile`(1M) man page

CHAPTER 8

Remote Administration in Trusted Extensions

This chapter describes how to use Trusted Extensions administrative tools to administer a remote host.

Secure Remote Administration in Trusted Extensions

By default, Trusted Extensions does not allow remote administration. Remote administration would present a significant security risk if remote untrusted systems could administer a system that is configured with Trusted Extensions. Therefore, the system is initially installed without the ability for it to be remotely administered.

Until the network is configured, all remote hosts are assigned the `admin_low` security template. Therefore, the CIPSO protocol is not used or accepted for any connections. While in this initial state, the system is protected from remote attacks by several mechanisms. Mechanisms include netservices settings, default login policy, and PAM policy.

- When the `netservices` Service Management Facility (SMF) profile is set to `limited`, no remote services except secure shell are enabled. However, the `ssh` service cannot be used for remote logins because of the login and PAM policies.
- The `root` account cannot be used for remote logins because the default policy for `CONSOLE` in `/etc/default/login` prevents remote logins by `root`.
- Two PAM settings also affect remote logins.

  The `pam_roles` module always rejects local logins from accounts of type `role`. By default, this module also rejects remote logins. However, the system can be configured to accept remote logins by specifying `allow_remote` in the system's `pam.conf` entry.

Additionally, the pam_tsol_account module rejects remote logins into the global zone unless the CIPSO protocol is used. The intent for this policy is that remote administration should be done using another Trusted Extensions system.

To enable remote login functionality, both systems must assign their peer to a CIPSO security template. If this approach is not practical, the network protocol policy can be relaxed by specifying the allow_unlabeled option in the pam.conf file. If either one of these two policies are relaxed, the default network template should be changed so that arbitrary machines cannot access the global zone. The admin_low template should be used sparingly, and the tnrhdb database should be modified so that the wildcard address 0.0.0.0 does not default to the admin_low label. For details, see "Administering Remotely (Tasks)" on page 90 and "How to Limit the Hosts That Can Be Contacted on the Trusted Network" on page 147.

Methods for Administering Remote Systems

Typically, administrators use the rlogin and ssh commands to administer remote systems from the command line. The Solaris Management Console can also be used. In CDE, the dtappsession program can launch Trusted CDE actions remotely.

- The root user can log in to a remote host from a terminal. For the steps, see "How to Log In Remotely From the Command Line" on page 92. This method works as it does on a Solaris system. This method is insecure.
- A role can log in to a remote host from a terminal in their role workspace. For the steps, see "How to Log In Remotely From the Command Line" on page 92.
- Administrators can launch a Solaris Management Console server that is running on a remote host. For the steps, see "How to Remotely Administer With the Solaris Management Console" on page 94.
- Actions in the Trusted_Extensions folder can be started remotely by using the dtappsession command. For the steps, see "How to Remotely Administer With dtappsession" on page 92.

Remote Login by a Role

As in the Solaris OS, a setting in the /etc/default/login file on each host must be changed to allow remote logins. In Trusted Extensions, the Security Administrator role is responsible for the change. For the procedure, see "How to Enable root to Log In Remotely" on page 90.

On both Trusted Extensions and Solaris hosts, remote logins might or might not require authorization. "Remote Login Management" on page 89 describes the conditions and types of logins that require authorization. By default, roles have the Remote Login authorization.

Remote Role-Based Administration From Unlabeled Hosts

In Trusted Extensions, users assume roles through the Trusted Path menu. The roles then operate in protected trusted workspaces. By default, roles cannot be assumed outside of the trusted path. If site policy permits, the Security Administrator role can change the default policy. Administrators on unlabeled hosts that are running Solaris Management Console 2.1 client software can then administer trusted hosts.

- To change the default, see "How to Enable Remote Administration by a Role" on page 91.
- To administer remotely, see "How to Log In Remotely From the Command Line" on page 92.

This policy change only applies when the user on the remote unlabeled computer has a user account on the Trusted Extensions host. The Trusted Extensions user account must have the ability to assume the administrative role. The role can then also use the Solaris Management Console to administer the remo

Caution – If remote administration from a non-Trusted Extensions host is enabled, the administrative environment is less protected than a Trusted Extensions administrative workspace. Be cautious when entering passwords and other secure data. As a precaution, shut down all untrusted applications before starting the Solaris Management Console.

Remote Login Management

A remote login between two Trusted Extensions hosts is considered to be an extension of the current login session.

An authorization is not required when the `rlogin` command does not prompt for a password. If an `/etc/hosts.equiv` or a `.rhosts` file in the user's home directory on the remote host lists either the username or the host from which the remote login is being attempted, no password is required. For more information, see the `rhosts`(4) and `rlogin`(1) man pages.

For all other remote logins, including logins with the `ftp`(1) command, the Remote Login authorization is required.

To create a profile that includes the Remote Login authorizations, see "Managing Users and Rights With Solaris Management Console (Tasks)" on page 77.

Administering Remotely (Tasks)

The following table describes how to administer a remote Trusted Extensions host.

Task	Description	For Instructions
Enable root to remotely log in to a Trusted Extensions host	Enable the `root` user to work remotely from a labeled host.	"How to Enable `root` to Log In Remotely" on page 90
Enable a role to remotely log in to a Trusted Extensions host	Allow any role to work remotely from a labeled host.	"How to Enable Remote Administration by a Role" on page 91
Enable remote login from an unlabeled host to a Trusted Extensions host	Allow any user or role to work remotely from an unlabeled host.	"How to Enable Remote Logins From an Unlabeled Host" on page 91
Log in remotely to a Trusted Extensions host	Log in as a role to a Trusted Extensions.	"How to Log In Remotely From the Command Line" on page 92
Administer a host remotely	Use the `dtappsession` command to administer the remote host with Trusted_Extensions actions.	"How to Remotely Administer With `dtappsession`" on page 92
	Use the Solaris Management Console to administer the remote host.	"How to Remotely Administer With the Solaris Management Console" on page 94
Enable specific users to log in to the global zone	Use user and network tools in the Solaris Management Console to enable specific users to access the global zone.	"How to Enable Specific Users to Log in Remotely to the Global Zone" on page 94

▼ How to Enable `root` to Log In Remotely

As in the Solaris OS, root can log in remotely from a labeled host when the `CONSOLE` entry is disabled. To log in from an unlabeled host, the `allow_unlabeled` option must be added to the `pam.conf` file.

Before You Begin You must be in the Security Administrator role in the global zone.

1 **Edit the `/etc/default/login` file.**

Use the Admin Editor action. For details, see "How to Edit Administrative Files in Trusted Extensions" on page 44.

a. **Insert a pound sign (#) to comment out the line:** `CONSOLE=/dev/console`.

```
#CONSOLE=/dev/console
```

b. **Save and quit the file.**

2 **To use ssh, you must permit root logins.**

Modify the /etc/ssh/sshd_config file.

```
PermitRootLogin yes
```

3 **To log in from an unlabeled host, you relax PAM policy.**

For the procedure, see "How to Enable Remote Logins From an Unlabeled Host" on page 91.

▼ How to Enable Remote Administration by a Role

Follow this procedure to enable a role, including the root role, to remotely log in from a labeled host.

Before You Begin You must be in the Security Administrator role in the global zone.

1 **Launch a terminal.**

Bring up the Workspace Menu by clicking with mouse button 3 on the screen background. Select Tools → Terminal.

2 **Relax the PAM policy for pam_tsol_account.so.1.**

a. **Edit the /etc/pam.conf file.**

b. **Add allow_remote to the account module for pam_roles.so.1.**

```
other    account    required    pam_roles.so.1    allow_remote
```

The fields are separated by tabs.

c. **Save and quit the file.**

3 **To log in as the root role, complete "How to Enable root to Log In Remotely" on page 90.**

4 **To log in from an unlabeled host, you relax PAM policy.**

For the procedure, see "How to Enable Remote Logins From an Unlabeled Host" on page 91.

▼ How to Enable Remote Logins From an Unlabeled Host

For any user or role to log in from an unlabeled host, the allow_unlabeled option must be added to the pam.conf file.

Before You Begin You should have completed "How to Limit the Hosts That Can Be Contacted on the Trusted Network" on page 147.

You must be in the Security Administrator role in the global zone.

- **Relax the policy for `pam_tsol_account.so.1` in the `/etc/pam.conf` file.**

  a. **Add `allow_unlabeled` to the `account` module for `pam_tsol_account.so.1`.**

  ```
  other    account    required    pam_tsol_account.so.1    allow_unlabeled
  ```

  The fields are separated by tabs.

  b. **Save and quit the file.**

How to Log In Remotely From the Command Line

Note – The `telnet` cannot be used for remote role assumption because this command cannot pass the primary and role identities to the `pam_roles` module.

Before You Begin The role must have the Remote Login authorization. By default, this authorization is in the Remote Administration and Maintenance and Repair rights profiles.

The Security Administrator has completed the procedure "How to Enable Remote Administration by a Role" on page 91 on every host that is going to be remotely administered. If the computer can be administered from an unlabeled host, the procedure "How to Enable Remote Logins From an Unlabeled Host" on page 91 has also been completed.

- **From a role workspace, log in to the remote host.**

  Use the `rlogin` command, the `ssh` command, or the `ftp` command.

  - If the `rlogin` command is used to log in, all commands that are in the current role's rights profiles are available.
  - If the `ftp` command is used, see the `ftp(1)` man page for the commands that are available.

How to Remotely Administer With `dtappsession`

The `dtappsession` program enables an administrator to administer a remote system that is running CDE. You can also invoke `dtappsession` from a Solaris Management Console that is administering a remote system. For details, see "How to Remotely Administer With the Solaris Management Console" on page 94.

`dtappsession` is useful when a remote host does not have a monitor. For example, `dtappsession` is often used when to administer domains on large servers. For more information, see the `dtappsession(1)` man page.

Before You Begin On a labeled host, you must be in an administrative role in the global zone. On an unlabeled host, you must assume a role that is defined on the remote host. You must then run the remote login from the role's profile shell.

1 **(Optional) Create a workspace that is dedicated to the remote session.**

To avoid confusion between the remote CDE applications and any local applications, dedicate an administrative role workspace to this procedure.

2 **Log in to the remote host.**

You can use the `rlogin` command or the `ssh` command.

```
$ ssh remote-host
```

3 **Start remote administration.**

In the terminal, type the `dtappsession` command followed by the name of the local host.

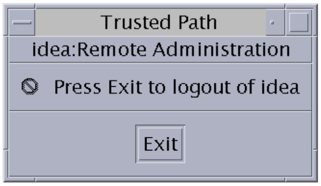
```
$ /usr/dt/bin/dtappsession local-host
```

An Application Manager that is running on the remote host displays on the local host. Also, an Exit dialog box appears.

4 **Administer the remote host.**

If you invoked the remote session from CDE, you can use actions in the Trusted_Extensions folder.

5 **When finished, click the `Exit` button.**



Trusted Path

idea:Remote Administration

Press Exit to logout of idea

Caution – Closing the Application Manager does not end the login session and is not recommended.

6 **In the terminal, exit the remote login session.**

```
$ exit
$ hostname
local-host
```

▼ How to Remotely Administer With the Solaris Management Console

The Solaris Management Console provides a remote administration interface.

Before You Begin You must be in an administrative role in the global zone.

1 Launch the Solaris Management Console.

For details, see "How to Launch the Solaris Management Console" on page 42.

a. From the Console menu, choose Open Toolbox.

b. In the Open Toolbox dialog box, select the appropriate server.

2 Administer the remote system.

To run `dtappsession`, double-click the Legacy Applications icon in the left panel.

▼ How to Enable Specific Users to Log in Remotely to the Global Zone

You might want to do this procedure for a tester who is using a labeled host remotely. For security, the host should be running a disjoint label from other users.

Before You Begin You should have a very good reason why this user can log in to the global zone.

You must be in the Security Administrator role in the global zone.

1 For specific users to log in to the global zone, assign to them an administrative label range.

Use the Solaris Management Console to assign a clearance of `ADMIN_HIGH` and a minimum label of `ADMIN_LOW` to each user. For details, see "How to Modify a User's Label Range in the Solaris Management Console" on page 78.

2 To enable remote login from a labeled zone into the global zone, do the following.

a. Enable remote login to the global zone.

Use the Solaris Management Console to add a multilevel port to the global zone. Port 513 over the TCP protocol enables remote login. For an example, see "How to Create a Multilevel Port for a Zone" on page 112.

b. Read the `tnzonecfg` changes into the kernel.

```
# tnctl -fz /etc/security/tsol/tnzonecfg
```

c. **Restart the remote login service.**

```
# svcadm restart svc:/network/login:rlogin
```

◆ ◆ ◆ CHAPTER 9

Trusted Extensions and LDAP

This chapter describes the use of the Sun Java System Directory Server (LDAP service) for a system that is configured with Trusted Extensions.

Using a Naming Service

To achieve uniformity of user, host, and network attributes within a security domain with multiple Trusted Extensions computers, a naming service is used for distributing most configuration information. LDAP is an example of a naming services. The `nsswitch.conf` file determines which naming service is used. LDAP is the recommended naming service for Trusted Extensions.

The Sun Java System Directory Server can provide the LDAP naming service for Trusted Extensions and Solaris clients. The server must include Trusted Extensions network databases, and the Trusted Extensions clients must connect to the server over a multilevel port. The security administrator specifies the multilevel port when configuring Trusted Extensions.

Trusted Extensions adds two trusted network databases to the LDAP server: `tnrhdb` and `tnrhtp`. They are administered by using Security Templates tool in the Solaris Management Console.

- For information about the use of the LDAP naming service in the Solaris Operating System, see *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*.
- Setting up the Sun Java System Directory Server for Trusted Extensions clients is described in *Solaris Trusted Extensions Installation and Configuration*. Trusted Extensions computers can be clients of a Solaris LDAP server by using an LDAP proxy server that is configured with Trusted Extensions.

Note – Systems that are configured with Trusted Extensions cannot be clients of NIS or NIS+ masters.

Non-Networked Trusted Extensions Computers

If a naming service is not used at a site, administrators have the responsibility to ensure that configuration information for users, hosts, and networks is identical on all hosts. A change that is made on one host must be made on all hosts.

On a non-networked Trusted Extensions computer, configuration information is maintained in the `/etc`, `/etc/security`, and `/etc/security/tsol` directories. Actions in the Trusted_Extensions folder enable you to modify some configuration information. The Security Templates tool in the Solaris Management Console enables you to modify network database parameters. Users, roles, and rights are modified in the User Accounts, Administrative Roles, and Rights tools. A toolbox on This Computer with `Scope=Files, Policy=TSOL` stores configuration changes locally.

Trusted Extensions LDAP Databases

Trusted Extensions extends the directory server's schema to accommodate the `tnrhdb` and `tnrhtp` databases. Trusted Extensions defines two new attributes, `ipTnetNumber` and `ipTnetTemplateName`, and two new object classes, `ipTnetHost` and `ipTnetTemplate.`

The attribute definitions are as follows:

```
ipTnetNumber
   ( 1.3.6.1.1.1.1.34 NAME 'ipTnetNumber'
     DESC 'Trusted network host or subnet address'
     EQUALITY caseExactIA5Match
     SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
     SINGLE-VALUE )

ipTnetTemplateName
   ( 1.3.6.1.1.1.1.35 NAME 'ipTnetTemplateName'
     DESC 'Trusted network template name'
     EQUALITY caseExactIA5Match
     SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
     SINGLE-VALUE )
```

The object class definitions are as follows:

```
ipTnetTemplate
   ( 1.3.6.1.1.1.2.18 NAME 'ipTnetTemplate' SUP top STRUCTURAL
     DESC 'Object class for Trusted network host templates'
     MUST ( ipTnetTemplateName )
     MAY ( SolarisAttrKeyValue ) )

ipTnetHost
   ( 1.3.6.1.1.1.2.19 NAME 'ipTnetHost' SUP top AUXILIARY
     DESC 'Object class for Trusted network host/subnet address
```

```
           to template mapping'
     MUST ( ipTnetNumber $ ipTnetTemplateName ) )
```

The cipso template definition in LDAP is similar to the following:

```
ou=ipTnet,dc=example,dc=example1,dc=exampleco,dc=com
 objectClass=top
 objectClass=organizationalUnit
 ou=ipTnet

 ipTnetTemplateName=cipso,ou=ipTnet,dc=example,dc=example1,dc=exampleco,dc=com
 objectClass=top
 objectClass=ipTnetTemplate
 ipTnetTemplateName=cipso
 SolarisAttrKeyValue=host_type=cipso;doi=1;min_sl=ADMIN_LOW;max_sl=ADMIN_HIGH;

 ipTnetNumber=0.0.0.0,ou=ipTnet,dc=example,dc=example1,dc=exampleco,dc=com
 objectClass=top
 objectClass=ipTnetTemplate
 objectClass=ipTnetHost
 ipTnetNumber=0.0.0.0
 ipTnetTemplateName=internal
```

Using the LDAP Naming Service

The LDAP naming service is managed in Trusted Extensions as it is managed in the Solaris OS. the following is a sample of useful commands, and references to more detailed information.

- For strategies to solve LDAP configuration problems, see Chapter 13, "LDAP Troubleshooting (Reference)," in *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*.
- To troubleshoot client-to-server LDAP connection problems that are affected by labels, see "How to Debug a Client Connection to the LDAP Server" on page 158.
- To troubleshoot other client-to-server LDAP connection problems, see Chapter 13, "LDAP Troubleshooting (Reference)," in *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*.
- To display LDAP entries from an LDAP client:

  ```
  $ ldaplist -l
  $ ldap_cachemgr -g
  ```
- To display LDAP entries from an LDAP server:

  ```
  $ ldap_cachemgr -g
  $ idsconfig -v
  ```
- To list the hosts that LDAP manages:

```
$ ldaplist -l hosts       Long listing
$ ldaplist hosts       One-line listing
```

- To list information in the Directory Information Tree (DIT) on LDAP:

```
$ ldaplist -l services | more
dn: cn=apocd+ipServiceProtocol=udp,ou=Services,dc=exampleco,dc=com
  objectClass: ipService
  objectClass: top
  cn: apocd
  ipServicePort: 38900
  ipServiceProtocol: udp

...
$ ldaplist services name
dn=cn=name+ipServiceProtocol=udp,ou=Services,dc=exampleco,dc=com
```

- To display the status of the LDAP service on the client:

```
# svcs -xv network/ldap/client
svc:/network/ldap/client:default (LDAP client)
 State: online since date
   See: man -M /usr/share/man -s 1M ldap_cachemgr
   See: /var/svc/log/network-ldap-client:default.log
Impact: None.
```

- To start and stop the LDAP client:

```
# svcadm enable network/ldap/client

# svcadm disable network/ldap/client
```

- To start and stop the LDAP server:

```
# installation-directory/slap-LDAP-server-hostname/start-slapd

# installation-directory/slap-LDAP-server-hostname/stop-slapd
```

CHAPTER 10

Managing Zones in Trusted Extensions

This chapter describes how non-global zones work on a system that is configured with Trusted Extensions.

Zones in Trusted Extensions

A properly configured Trusted Extensions host consists of a global zone, which is the operating system instance, and one or more labeled non-global zones. During configuration, Trusted Extensions attaches a unique label, which creates a labeled zone. The labels come from the `label_encodings` file. The installers can create a zone for each label, but are not required to. It is possible to have more labels than labeled zones on a system. It is not possible to have more labeled zones than labels.

On a Trusted Extensions system, the file systems of a zone are usually mounted as a loopback file system (`lofs`). All writable files and directories in a labeled zone are at the label of the zone. By default, a user can view files that are in a zone at a lower level than the user's current level. This configuration enables users to view their home directories at lower labels than the label of the current workspace. Although users can view files, they cannot modify those files. Users can only modify files from a process that is the same label as the file.

In Trusted Extensions, the global zone is an administrative zone. The labeled zones are for ordinary users. Users can work in a zone whose label is within the user's accreditation range.

Every zone has an associated IP address and security attributes. A zone can be configured with multilevel ports (MLPs). Also, a zone can set a policy for Internet Control Message Protocol (ICMP) broadcasts, such as `ping`.

Zones in Trusted Extensions are built on the Solaris zones product. For details, see Part II, "Zones," in *System Administration Guide: Solaris Containers-Resource Management and Solaris Zones*. In particular, patching and package installation issues affect Trusted Extensions. For details, see

Chapter 24, "About Packages and Patches on a Solaris System With Zones Installed (Overview)," in *System Administration Guide: Solaris Containers-Resource Management and Solaris Zones* and Chapter 28, "Troubleshooting Miscellaneous Solaris Zones Problems," in *System Administration Guide: Solaris Containers-Resource Management and Solaris Zones*.

Zones and IP Addresses in Trusted Extensions

Your installation assigned IP addresses to the global zone and the labeled zones. There are three possible configurations.

- The machine has one IP address for the global zone and all labeled zones.

  This configuration is useful on a machine that uses DHCP software to get its IP address. All the labeled zones can share one address with the global zone. If no users are expected to log in, an LDAP server might have this configuration.
- The machine has one IP address for the global zone, and one address that is shared by all zones, including the global zone. Any zone can have a combination of a unique address and a shared address.

  This configuration is useful on a machine that ordinary users are going to log in to. This configuration saves IP addresses. It can also be used for a printer or an NFS server.
- The machine has one IP address for the global zone, and each labeled zone has a unique IP address.

  This configuration is useful for providing access to separate physical networks of single-level systems. Each zone would typically have a different physical network as well as a unique IP address.

Zones and Multilevel Ports

By default, a zone cannot send packets to and receive packets from any other zone. Multilevel ports (MLPs) enable particular services on a port to accept requests within a range of labels or from a set of labels. These privileged services can reply at the label of the request. For example, you might want to create a privileged web browser port, that can listen at all labels, but whose replies are restricted by label. By default, labeled zones have no MLPs.

The range of labels or set of labels that constrains the packets that the MLP can accept is configured by the zone's IP address. The IP address is assigned a remote host template in the `tnrhdb` database. The label range or set of labels in the remote host template constrains the packets that the MLP can accept.

- On a system where the global zone has an IP address, and each labeled zone has a unique IP address an MLP for a particular service can be added to every zone. For example, the system could be configured so that the `ssh` service, over TCP port `22`, is an MLP in the global zone, and in every labeled zone.

- In a typical configuration, the global zone is assigned one IP address, and labeled zones share a second IP address. When an MLP is added to a shared interface, the service packet is routed to the labeled zone where the MLP is defined. The packet is accepted only if the remote host template for the labeled zone includes the label of the packet. If the range is `ADMIN_LOW` to `ADMIN_HIGH`, then all packets are accepted. A narrower range would discard packets that are not within the range.

  At most, one zone can define a particular port to be an MLP on a shared interface. In the preceding scenario, where the `ssh` port is configured as a shared MLP in a non-global zone, no other zone can receive `ssh` connections on the shared address. However, the global zone could define the `ssh` port as a private MLP for receipt of connections on its zone-specific address.
- On a system where the global zone and the labeled zones share an IP address, an MLP for the `ssh` service could be added to one zone. If the MLP for `ssh` is added to the global zone, then no labeled zone can add an MLP for the `ssh` service. Similarly, if the MLP for the `ssh` service is added to a labeled zone, then the global zone cannot be configured with an `ssh` MLP.

For an example of adding MLPs to labeled zones, see Example 13–14.

Zones and ICMP in Trusted Extensions

Networks transmit broadcast messages and send ICMP packets to machines on the network. On a multilevel system, these transmissions could flood the system at every label. The network policy for labeled zones, by default, requires that ICMP packets be received only at the matching label.

Zone Administration Utilities in Trusted Extensions

Some zone administration can be done from the command line. However, the simplest way to administer zones is to use the GUIs that Trusted Extensions provides.

- Configuring the security attributes of zones is done by using the Trusted Network Zones tool in the Solaris Management Console. For a description of the tool, see “Trusted Network Zones Tool” on page 34. For examples of zone configuration and creation, see Chapter 4, “Configuring Trusted Extensions,” in *Solaris Trusted Extensions Installation and Configuration*, and “How to Create a Multilevel Port for a Zone” on page 112.
- The shell script, `/usr/sbin/txzonemgr`, provides a menu-based wizard for creating, installing, initializing, and booting zones. If you are administering zones from Java DS, use `txzonemgr` rather than CDE actions. `txzonemgr` uses the `zenity` command. For details, see the `zenity`(1) man page.
- Configuring and creating zones can be done from desktop utilities. In CDE, actions in the Trusted_Extensions folder are used. In Java DS, applets can be used. For a description of the actions, see “Trusted CDE Actions” on page 28. For procedures that use the actions, see “How to Launch CDE Administrative Actions in Trusted Extensions” on page 43.

Managing Zones (Tasks)

The following table describes zone management that is specific to Trusted Extensions.

Task	Description	For Instructions
View all zones	At any label, see the zones that are dominated by the current zone.	"How to Display Ready or Running Zones" on page 105
View mounted directories	At any label, see the directories that are dominated by the current label.	"How to Display the Labels of Mounted Files" on page 106
Enable ordinary users to view an `/etc` file	Loopback mount a directory or file from the global zone that is not visible by default in a labeled zone.	"How to Mount a File That is Usually Not Visible From a Labeled Zone" on page 107
Prevent ordinary users from viewing a lower-level home directory from a higher label.	By default, lower-level directories are visible from higher-level zones. When you disable the mounting of one lower-level zone, you disable all mounts of lower-level zones.	"How to Disable the Mounting of Lower-Level Files" on page 108
Configure a zone to enable changing the labels on files.	Labeled zones have limited privileges. By default, labeled zones do not have the privilege that enables an authorized user to relabel a file. You modify the zone configuration to add the privilege.	"How to Enable Files to be Relabeled From a Labeled Zone" on page 109
Move a file or directory into or out of a labeled zone	Change a file or directory's level of security by changing its label.	"How to Use Two File Managers to Relabel a File" on page 111
Configure a new zone	Create a zone at a label that is not currently being used to label a zone on this system.	"Specify Zone Names and Zone Labels" in *Solaris Trusted Extensions Installation and Configuration* Then, follow the procedure that the install team used to create the other zones. For the steps, see "Creating the Labeled Zones (Tasks)" in *Solaris Trusted Extensions Installation and Configuration*.
Create a multilevel port for an application.	Multilevel ports are useful for programs that require a multilevel feed into a labeled zone.	"How to Create a Multilevel Port for a Zone" on page 112
Troubleshoot NFS mount and access problems	Debug general access issues for mounts, and possibly for zones.	"How to Troubleshoot Mount Failures" on page 122

▼ How to Display Ready or Running Zones

This procedure creates a shell script that displays the labels of the current zone and of all zones that the current zone dominates.

Before You Begin You must be in the System Administrator role in the global zone.

1. **Use the Admin Editor action to create the `getzonelabels` script.**

   Supply the pathname to the script, such as `/usr/local/scripts/getzonelabels`.

2. **Supply the following content and save the file.**

```
#!/bin/sh
#
echo "NAME\t\tSTATUS\t\tLABEL"
echo "====\t\t======\t\t====="
myzone=‘zonename‘
for i in ‘/usr/sbin/zoneadm list -p‘ ; do
        zone=‘echo $i | cut -d ":" -f2‘
        status=‘echo $i | cut -d ":" -f3‘
        path=‘echo $i | cut -d ":" -f4‘
        if [ $zone != global ]; then
                if [ $myzone = global ]; then
                        path=$path/root/tmp
                else
                        path=$path/export/home
                fi
        fi
        label=‘/usr/bin/getlabel -s $path |cut -d ":" -f2-9‘
        if [ ‘echo $zone|wc -m‘ -lt 8 ]; then
                echo "$zone\t\t$status\t$label"
        else
                echo "$zone\t$status\t$label"
        fi
done
```

3. **Test the script in the global zone.**

   When run from the global zone, the script displays the labels of all ready or running zones. Here is the global zone output for the zones that were created from the default `label_encodings` file:

```
# getzonelabels
NAME            STATUS          LABEL
====            ======          =====
global          running         ADMIN_HIGH
needtoknow      running         CONFIDENTIAL : NEED TO KNOW
restricted      running         CONFIDENTIAL : RESTRICTED
internal        running         CONFIDENTIAL : INTERNAL
public          running         PUBLIC
```

Example 10–1 Displaying the Labels of All Ready or Running Zones

In the following example, a user runs the `getzonelabels` script in the `internal` zone.

```
# getzonelabels
NAME            STATUS          LABEL
====            ======          =====
internal        running         CONFIDENTIAL : INTERNAL
public          running         PUBLIC
```

▼ How to Display the Labels of Mounted Files

This procedure creates a shell script that displays the mounted file systems of the current zone. When run from the global zone, the script displays the labels of all mounted file systems in every zone.

Before You Begin You must be in the System Administrator role in the global zone.

1. **Use the Admin Editor action to create the `getmounts` script.**

   Supply the pathname to the script, such as `/usr/local/scripts/getmounts`.

2. **Supply the following content and save the file.**

```
#!/bin/sh
#
for i in ‘/usr/sbin/mount -p | cut -d " " -f3‘ ; do
        /usr/bin/getlabel $i
done
```

3. **Test the script in the global zone.**

```
% /usr/local/scripts/getmounts
/:      ADMIN_LOW
/dev:   ADMIN_LOW
/kernel:        ADMIN_LOW
/lib:   ADMIN_LOW
/opt:   ADMIN_LOW
/platform:      ADMIN_LOW
/sbin:  ADMIN_LOW
/usr:   ADMIN_LOW
/var/tsol/doors:        ADMIN_LOW
/zone/needtoknow/export/home:   CONFIDENTIAL : NEED TO KNOW
/zone/internal/export/home:     CONFIDENTIAL : INTERNAL USE ONLY
/zone/restricted/export/home:   CONFIDENTIAL : RESTRICTED
/proc:  ADMIN_LOW
/system/contract:       ADMIN_LOW
/etc/svc/volatile:      ADMIN_LOW
/etc/mnttab:    ADMIN_LOW
```

```
/dev/fd:        ADMIN_LOW
/tmp:           ADMIN_LOW
/var/run:       ADMIN_LOW
/zone/public/export/home:  PUBLIC
/root:          ADMIN_LOW
```

Example 10–2 Displaying the Labels of File Systems in a `restricted` Zone

When run from a labeled zone by an ordinary user, the `getmounts` script displays the labels of all the mounted file systems in that zone. On a system that created zones for every label in the default `label_encodings` file, the following is the output from the `restricted` zone:

```
% /usr/local/scripts/getmounts
/:      CONFIDENTIAL : RESTRICTED
/dev:   CONFIDENTIAL : RESTRICTED
/kernel:        ADMIN_LOW
/lib:   ADMIN_LOW
/opt:   ADMIN_LOW
/platform:      ADMIN_LOW
/sbin:  ADMIN_LOW
/usr:   ADMIN_LOW
/var/tsol/doors:        ADMIN_LOW
/zone/needtoknow/export/home:   CONFIDENTIAL : NEED TO KNOW
/zone/internal/export/home:     CONFIDENTIAL : INTERNAL USE ONLY
/proc:  CONFIDENTIAL : RESTRICTED
/system/contract:       CONFIDENTIAL : RESTRICTED
/etc/svc/volatile:      CONFIDENTIAL : RESTRICTED
/etc/mnttab:    CONFIDENTIAL : RESTRICTED
/dev/fd:        CONFIDENTIAL : RESTRICTED
/tmp:   CONFIDENTIAL : RESTRICTED
/var/run:       CONFIDENTIAL : RESTRICTED
/zone/public/export/home:       PUBLIC
/home/gfaden:   CONFIDENTIAL : RESTRICTED
```

▼ How to Mount a File That is Usually Not Visible From a Labeled Zone

This procedure enables a user in a specified labeled zone to see files that are not exported from the global zone by default.

Before You Begin You must be in the System Administrator role in the global zone.

1. **Halt the zone whose configuration you want to change.**

```
# zoneadm -z zone-name halt
```

2 Loopback mount a file or directory.

For example, enable ordinary users to view a file in the `/etc` directory.

```
# zonecfg -z zone-name
 add filesystem
 set special=/etc/filename
 set directory=/etc/filename
 set type=lofs
 end
 exit
```

3 3. Start the zone.

```
# zoneadm -z zone-name boot
```

Example 10–3 Loopback Mounting the `/etc/passwd` file

In this example, the security administrator wants to enable testers and programmers to check that their local passwords are set. The `sandbox` zone is first halted, then it is configured to loopback mount the `passwd` file, then the zone is restarted.

```
# zoneadm -z sandbox halt
# zonecfg -z sandbox
 add filesystem
    set special=/etc/passwd
    set directory=/etc/passwd
    set type=lofs
    end
 exit
# zoneadm -z sandbox boot
```

▼ How to Disable the Mounting of Lower-Level Files

By default, users can view lower-level files. Use this procedure to prevent the viewing of all lower-level files from a particular zone.

Before You Begin You must be in the System Administrator role in the global zone.

1 Halt the zone whose configuration you want to change.

```
# zoneadm -z zone-name halt
```

2 Configure the zone to prevent viewing.

Remove the `net_mac_aware` privilege from the zone.

```
# zonecfg -z zone-name
 set limitpriv=default,!net_mac_aware
 exit
```

3 **3. Start the zone.**

```
# zoneadm -z zone-name boot
```

For a description of the `net_mac_aware` privilege, see the `privileges`(5) man page.

Example 10–4 Preventing Users From Viewing Lower-Level Files

In this example, the security administrator wants to prevent users on this machine from being confused. Therefore, users can only see files at the label at which the users are working. So, the security administrator prevents the viewing of all lower-level files. On this system, users cannot see publicly available files unless they are working at the `PUBLIC` label.

```
# zoneadm -z restricted halt
# zonecfg -z restricted
 set limitpriv=default,!net_mac_aware
 exit
# zoneadm -z restricted boot

# zoneadm -z needtoknow halt
# zonecfg -z needtoknow
 set limitpriv=default,!net_mac_aware
 exit
# zoneadm -z needtoknow boot

# zoneadm -z internal halt
# zonecfg -z internal
 set limitpriv=default,!net_mac_aware
 exit
# zoneadm -z internal boot
```

Because `PUBLIC` is the lowest label, the security administrator does not run the commands for the `PUBLIC` zone.

▼ How to Enable Files to be Relabeled From a Labeled Zone

This procedure is a prerequisite for a user to be able to relabel files.

Before You Begin You must be in the Security Administrator role in the global zone.

1 **Halt the zone whose configuration you want to change.**

```
# zoneadm -z zone-name halt
```

2 **Configure the zone to allow relabeling.**

Add the appropriate privileges to the zone. The windows privileges enable users to use drag and drop, and cut and paste.

- **To enable downgrades, add the `file_downgrade_sl` privilege to the zone.**

  ```
  # zonecfg -z zone-name
   set limitpriv=default,win_dac_read,win_mac_read,win_dac_write,
   win_mac_write,win_selection,file_downgrade_sl
   exit
  ```

- **To enable upgrades, add the `sys_trans_label` and `file_upgrade_sl` privileges to the zone.**

  ```
  # zonecfg -z zone-name
   set limitpriv=default,win_dac_read,win_mac_read,win_dac_write,
   win_mac_write,win_selection,sys_trans_label,file_upgrade_sl
   exit
  ```

- **To enable both upgrades and downgrades, add the both privileges to the zone.**

  ```
  # zonecfg -z zone-name
   set limitpriv=default,win_dac_read,win_mac_read,win_dac_write,
   win_mac_write,win_selection,sys_trans_label,file_downgrade_sl,
   file_upgrade_sl
   exit
  ```

3 **3. Start the zone.**

```
# zoneadm -z zone-name boot
```

For the user and process requirements, see the `setflabel`(3TSOL) man page.

Example 10–5 Enabling Upgrades From the `internal` Zone

In this example, the security administrator wants to enable authorized users on this machine to upgrade files. When you upgrade information, you protect the information at a higher level of security. In the global zone, you use the following zone administration commands.

```
# zoneadm -z internal halt
# zonecfg -z internal
 set limitpriv=default,sys_trans_label,file_upgrade_sl
 exit
# zoneadm -z internal boot
```

Authorized users can now upgrade `internal` information to `restricted` from this zone.

Example 10–6 Enabling Downgrades From the `restricted` Zone

In this example, the security administrator wants to enable authorized users on this machine to downgrade files. Because the administrator does not add windows privileges to the zone, authorized users cannot use the File Manger to relabel files. They can use the `setlabel` command.

When you downgrade information, you allow users at a lower level of security to access the files. In the global zone, you use the following zone administration commands.

```
# zoneadm -z restricted halt
# zonecfg -z restricted
 set limitpriv=default,file_downgrade_sl
 exit
# zoneadm -z restricted boot
```

Authorized users can now downgrade `restricted` information to `internal` or `public` from this zone by using the `setlabel` command.

▼ How to Use Two File Managers to Relabel a File

Before You Begin The zone of the source file must be configured with the appropriate privileges. For details, see "How to Enable Files to be Relabeled From a Labeled Zone" on page 109.

You must be authorized to downgrade or upgrade labels. For the required authorizations, see "Rules When Changing the Level of Security for Data" on page 49. To assign a required authorization, see "How to Create a Convenient Authorizations Rights Profile" on page 79.

1. **In CDE, open File Managers in workspaces at two different labels.**

2. **Move the target File Manager to the workspace of the source file.**

   Place the windows side by side.

3. **In the source File Manager, navigate to the directory of the file that is to be moved.**

4. **In the target File Manager, navigate to the directory where you plan to place the file.**

5. **Drag the file with the mouse from the source File Manager to the target File Manager.**

6. **Read the information in the Selection Confirmer.**

   - **If the information is correct, click OK.**

   - **Otherwise, cancel the operation.**

▼ How to Create a Multilevel Port for a Zone

This procedure is part of a larger project. You create a Multilevel Port (MLP) in a labeled zone to enable a particular port to communicate with the zone for a specific reason. In this procedure, a web proxy communicates with the zone. The Solaris Management Console is used to add the MLP.

Before You Begin You must be in the Security Administrator role in the global zone.

1 **Launch the Solaris Management Console.**

For details, see "How to Launch the Solaris Management Console" on page 42.

2 **Choose the Files toolbox.**

The title of the toolbox includes `Scope=Files, Policy=TSOL`.

3 **Add the proxy host and the webservices host to the list of computers.**

a. **Navigate to the Computers and Networks tool.**

In the Computers tool, click the Action menu and choose Add Computer. For the proxy host and the webservice host, do the following.

b. **Add the host name and IP address.**

c. **Apply the changes.**

4 **Configure the zones and the MLPs.**

a. **Navigate to the Trusted Network Zones tool.**

b. **If the zone names do not appear in the list, click the Action menu and choose Add Zone Configuration.**

c. **Assign a label to each zone.**

d. **In the MLP Configuration for Local IP Addresses, specify the appropriate port/protocol field.**

e. **Apply the settings.**

5 **Customize a template for each zone.**

a. **Navigate to the Security Templates tool.**

Click the Action menu and choose Add Template. For each host, create a template. Assign the template a recognizable name.

   b. **Create the template.**

      Use the host name for the template name.

   c. **Specify CIPSO for the Host Type**

   d. **Use the label of the zone for the Minimum Label and for the Maximum Label.**

   e. **Assign the zone label to the Security Label Set.**

   f. **Select the Hosts Explicitly Assigned tab.**

   g. **In the Add an Entry section, add the IP address that is associated with the zone.**

   h. **Apply the settings.**

6 **Close the Solaris Management Console.**

7 **3. Start the zones.**

```
# zoneadm -z zone-name boot
```

8 **In the global zone, add routes for the new addresses.**

If the zones have a shared IP address, do the following.

```
# route add proxy labeled-zones-IP-address
# route add webservice labeled-zones-IP-address
```

CHAPTER 11

Mounting Files in Trusted Extensions

This chapter describes how LOFS mounts and NFS mounts work on a system that is configured with Trusted Extensions.

File Systems in Trusted Extensions

Trusted Extensions software supports the same file systems and file system management commands as the Solaris OS. In addition, Trusted Extensions attaches a unique label to every non-global zone. All the files and directories that belong to that zone are mounted at the label of the zone. Any shared file systems that belong to other zones or to NFS servers are mounted at the label of the owner. Trusted Extensions prevents any mounts that would violate the MAC policies for labeling. For example, a zone's label must dominate all of its mounted file system labels, and only equally labeled file systems can be mounted read-write.

File System Mounts in Trusted Extensions

Mounting file systems on a system that is configured with Trusted Extensions is similar to mounting file systems on a Solaris system. For permanent mounts, you enter the standard mounting information in the `vfstab` file on the client and the sharing information in the `dfstab` file on the server. For dynamic mounts, use the `mount`(1M) command.

Labels affect which file systems can be mounted. File systems are shared and are mounted at a particular label. For a Trusted Extensions client to write to a file system that is NFS-mounted, the file system must be mounted read-write *and* be at the same label as the client. If you are setting up a mount between two Trusted Extensions hosts, the server and the client must have compatible remote host templates of type cipso. If you are setting up a mount between a Trusted Extensions host and an unlabeled host, file systems that are at the single label that is specified for the unlabeled host in the tnrhdb file can be mounted.

Labels also affect which file systems can be viewed. By default, lower-level objects are not available in a user's environment. Therefore, in the default configuration, an ordinary user cannot view files that are in a zone at a lower level than the user's current level. If site security permits, you can make these lower-level objects visible to the user. Files that are mounted with LOFS can be viewed, but cannot be modified. For details on NFS mounts, see "Access to NFS Mounted Directories in Trusted Extensions" on page 116.

The /export/home pathname is the exception to lower-level visibility. By default, users can see their lower-level home directories from a higher label. For details, see "Home Directory Creation in Trusted Extensions" on page 117.

The mount policy in Trusted Extensions has no MAC overrides. MAC policies enforce the default configuration, and are invisible to ordinary users. Ordinary users cannot see objects unless they have MAC access to them.

Access to NFS Mounted Directories in Trusted Extensions

By default, NFS-mounted file systems are visible at the label of the exported file system. If the file system is exported read-write, users at that label can write to the files. With the exception of home directories, NFS mounts that are at a lower label than the user's current session require administrative intervention to be visible to the user.

To make lower-level directories that are NFS-mounted visible to users in a higher-level zone, the administrator of the global zone on the NFS server exports the parent directory. The parent directory is exported at its label. On the client side, the administrator of the global zone must configure each zone that can mount the exported directory. Each zone must have the privilege net_mac_aware.

Server configuration

On the NFS server, you export the parent directory in a dfstab file. The dfstab file must be modified in the labeled zone of the directory that is being exported. For an example, see "How to Share File Systems" on page 120.

Client configuration

The net_mac_aware privilege must be specified in the zone configuration file that is used during initial zone configuration. So, a user who is permitted to see all lower-level home directories must have the net_mac_aware privilege in every zone except the lowest one. For an example, see "How to Mount File Systems" on page 121.

EXAMPLE 11–1 Providing Access to Lower-Level Home Directories

On the home directory server, the administrator modifies the dfstab file in every labeled zone. The dfstab file exports the /export/home directory with rw access, so that when the directory is mounted

EXAMPLE 11–1 Providing Access to Lower-Level Home Directories *(Continued)*

at the same label, the home directory is writable. To export the /export/home directory of PUBLIC, the administrator creates a workspace at the PUBLIC label on the home directory server, and modifies the dfstab file in that zone.

On the client, the administrator of the global zone checks that every labeled zone except the lowest label has the net_mac_aware privilege. This privilege permits the mount. This privilege can be specified by using the zonecfg command during zone configuration. The lower-level home directory can be viewed only. MAC protects the files in the directory from modification. The files can only be modified by a process at the same label of the files.

Home Directory Creation in Trusted Extensions

Home directories are a special case in Trusted Extensions. You need to make sure that the home directories are created in every zone that a user can use. Also, the home directory mount points must be created in the zones on the user's system. For NFS-mounted home directories to work correctly, the conventional location for directories, /export/home, must be used. The automounter has been modified to handle home directories in every zone, that is, at every label. For details, see "Changes to the Automounter in Trusted Extensions" on page 117.

Home directories are created when users are created. In Trusted Extensions, the Solaris Management Console is used to create users, so the Solaris Management Console creates the home directories. However, the console creates the home directories in the global zone of the home directory server. On that server, the directories are mounted by LOFS. Home directories are automatically created by the automounter if they are specified as LOFS mounts. However, the automounter cannot automatically create home directories on remote NFS servers. Either the user must first log in to the NFS server or administrative intervention is required. To create home directories for users, see "Enable Users to Access Their Home Directories" in *Solaris Trusted Extensions Installation and Configuration*.

Changes to the Automounter in Trusted Extensions

In Trusted Extensions, each label requires a separate home directory mount. The automount command has been modified to handle these labeled automounts. For each zone, autofs mounts an auto_home_*zone-name* file. For example, the following is the entry for the global zone in the auto_home_global file:

```
+auto_home_global
*       -fstype=lofs    :/export/home/&
```

When a zone is booted that permits lower-level zones to be mounted, the following occurs. The home directories of lower-level zones are mounted read-only under

/zone/<*zonename*>/export/home. The auto_home_<*zonename*> map specifies the /zone path as the source directory for an lofs remount onto /zone/<*zonename*>/home/<*username*>.

For example, the following is an auto_home_public entry in an auto_home_*zone-at-higher-label* map that is generated from a higher level zone:

```
+auto_home_public
*       -fstype=lofs    :/zone/public/export/home/&
```

The corresponding entry in the public zone is:

```
auto_home_public
*       -fstype=lofs    :/export/home/&
```

When a home directory is referenced and the name does not match any entries in the auto_home_<*zonename*> map, the map tries to match this loopback mount specification. The software creates the home directory when two conditions are met.

1. When the map finds the match of the loopback mount specification
2. When the home directory name matches a valid user whose home directory does not yet exist in *zonename*

For details on changes to the automounter, see the automount(1M) man page.

Trusted Extensions Software and NFS Protocols

Trusted Extensions software recognizes the NFS protocols that the Solaris Operating System (Solaris OS) supports: NFS Version 2 (V2), NFS Version 3 (V3), and NFS Version 4 (V4).

Trusted Extensions software recognizes labels on NFS Version 4 (V4) only. Therefore, any file system that is being mounted from a NFS server that is running Trusted Extensions software must be mounted with *vers=4* and *proto=tcp* mount options.

A host that is configured with Trusted Extensions can also share its own file systems with unlabeled hosts. A file or directory that is exported to an unlabeled host is *writable* if its label equals the label that is associated with the remote host in its trusted networking database entries. A file or directory that is exported to an unlabeled host is *readable* only if its label is dominated by the label that is associated with the remote host.

Communications with computers that are running a release of Trusted Solaris software is possible only at a single label. The Trusted Extensions system and the Trusted Solaris system must assign to the other system a template with the unlabeled host type. The unlabeled host types must specify the same single label. As an unlabled NFS client of a Trusted Solaris server, the label of the client cannot be ADMIN_LOW.

The NFS protocol that is used is independent of the type of the local file system. Rather, the protocol depends on the type of the sharing computer's operating system. The file system type that is specified to the `mount` command or in the `vfstab` for remote file systems is always NFS.

Managing Files and File Systems (Tasks)

The following table describes common tasks that are performed when managing files and file systems and the procedures associated with the tasks.

Task	Description	For Instructions
Back up files	Protect your data by backing it up.	"How to Back Up Files" on page 119
Restore data	Retrieve data from a backup.	"How to Restore Files" on page 120
Share the contents of a directory	Allow the contents of a directory to be shared among users.	"How to Share File Systems" on page 120
Mount the contents of a directory	Allow the contents of a directory to be mounted.	"How to Mount File Systems" on page 121
Create home directory mount points	Create mount points for every user at every label. This enables users to access their home directory on a system that is not the NFS home directory server.	"Enable Users to Access Their Home Directories" in *Solaris Trusted Extensions Installation and Configuration*
Hide lower-level information from a user who is working at a higher label	Prevent the viewing of lower-level information from a higher-level window.	"How to Disable the Mounting of Lower-Level Files" on page 108
Troubleshoot file system mounting problems	Resolve problems with mounting a file system.	"How to Troubleshoot Mount Failures" on page 122

▼ How to Back Up Files

1 **Assume the Operator role.**

This role includes the Media Backup rights profile.

2 **Use one of the following backup methods:**

- `/usr/lib/fs/ufs/ufsdump` for major backups
- `/usr/sbin/tar cT` for small backups
- A script calling either of the above commands

  For example, the Budtool backup application calls the `ufsdump` command. For details on the `T` option to the `tar` command, see the `tar`(1) man page.

▼ How to Restore Files

1. **Assume the System Administrator role.**

   This role includes the Media Restore rights profile.

2. **Use one of the following methods:**

   - `/usr/lib/fs/ufs/ufsrestore` for major restores
   - `/usr/sbin/tar xT` for small restores
   - A script calling either of the above commands

   For details on the `T` option to the `tar` command, see the `tar`(1) man page.

Caution – Only these commands preserve labels.

▼ How to Share File Systems

As in the Solaris OS, the Mounts and Shares tool in the Solaris Management Console cannot be used to mount or share file systems in labeled zones. Use the `dfstab` file in the labeled zone, and the `shareall` and `mount` commands to export and mount file systems in labeled zones.

Caution – Do not use proprietary names for shared file systems. The names of shared file systems are visible to every user.

Before You Begin You must be on the file server, in the zone at the label of the files that you want to share. You must be the superuser, or in the System Administrator role.

1. **Create a workspace at the label of the file system that is going to be shared.**

2. **Create a new `dfstab` file in that zone.**

   Each zone shares its exported directories at the label of the zone.

   a. **Go to the zone's `/etc/dfs` directory.**

   ```
   # cd /zone/zone-name/etc/dfs
   ```

   b. **Open the Admin Editor.**

   c. **Type the full pathname of the `dfstab` file into the editor.**

   ```
   # /zone/zone-name/etc/dfs/dfstab
   ```

d. **Add an entry to share the exported directory.**

   For example, the following entry shares an application's files at the label of the containing zone.

   ```
   share -F nfs -o ro /viewdir/viewfiles
   ```

3 **Repeat the preceding steps for every zone that is going to share a directory.**

4 **In the global zone, run the `shareall` command.**

The actual sharing occurs when each zone is brought into the `ready` or `running` state.

5 **Display the status of the labeled zones.**

```
# zoneadm list -v
```

6 **For each zone, share the directories.**

As root in the global zone, run one of the following commands for each zone. Each zone can share its directories in any of these ways.

- **If the zone is not in the running state and you do not want users to log in to the server at the label of the zone, set the zone state to `ready`.**

  ```
  # zoneadm -z zonename ready
  ```

- **If the zone is not in the running state and users are allowed to log in to the server at the label of the zone, boot the zone.**

  ```
  # zoneadm -z zonename boot
  ```

- **If the zone is already running, reboot the zone.**

  ```
  # zoneadm -z zonename reboot
  ```

7 **To enable the client to mount the exported files, see "How to Mount File Systems" on page 121.**

▼ How to Mount File Systems

Unlabeled and labeled hosts can be mounted on a Solaris Trusted Extensions labeled host. The assigned label of the remote host must be identical to the zone in which the file system is being mounted.

As in the Solaris OS, the Mounts and Shares tool in the Solaris Management Console cannot be used to mount or share file systems in labeled zones. Use the `dfstab` file in the labeled zone, and the `shareall` and `mount` commands to export and mount file systems in labeled zones.

Before You Begin You must be on the client system, in the zone at the label of the files that you want to mount. You must be the superuser, or in the System Administrator role. The zone must be configured with the `net_mac_aware` privilege.

1 **Create a workspace at the label of the file system that is going to be mounted.**

2 **In that workspace, run the `mount` command.**

You can also modify the local `vfstab` file to mount the file system across reboots.

▼ How to Share Files for User Access at a Higher Label

◗ **Follow the procedure, "How to Mount File Systems" on page 121.**

▼ How to Troubleshoot Mount Failures

Before You Begin You must be in the zone at the label of the files that you want to mount. You must be the superuser, or in the System Administrator role.

1 **Check the security attributes of the NFS server.**

Use the Security Templates tool in the Solaris Management Console, at the appropriate scope.

a. **Check that the IP address of the NFS server is an assigned host in one of the security templates.**

The address might be directly assigned, or be assigned indirectly through a wildcard mechanism. The address can be in a labeled template, or in an unlabeled template.

b. **Check the label that the template assigns to the NFS server.**

c. **Check that the label is consistent with the label at which you are trying to mount the files.**

2 **To mount file systems from an NFS server that is running earlier versions of Trusted Solaris software, do the following:**

- **For a Trusted Solaris 1 NFS server, use the `vers=2` and `proto=udp` options to the `mount` command.**

- **For a Trusted Solaris 2.5.1 NFS server, use the `vers=2` and `proto=udp` options to the `mount` command.**

- **For a Trusted Solaris 8 NFS server, use the `vers=3` and `proto=udp` options to the `mount` command.**

To mount file systems from any of these servers, the server must be assigned to an unlabeled template.

CHAPTER 12

Trusted Networking

This chapter describes trusted networking concepts and mechanisms in Trusted Extensions.

The Trusted Network

Trusted Extensions assigns security attributes to zones, hosts, and networks. These attributes ensure that the following security features are enforced on the network:

- Data is properly labeled in network communications.
- Mandatory access control (MAC) rules are enforced when data is sent or is received across a local network and when file systems are mounted.
- MAC rules are enforced when data is routed to distant networks.
- MAC rules are enforced when data is routed to zones.

In Trusted Extensions, network packets are protected by Mandatory Access Control (MAC). Labels are used for MAC decisions. Data is labeled explicitly or implicitly with a sensitivity label. A label has an ID field, a classification or "level" field, and a compartment or "category" field. Data must pass an accreditation check. This check determines if the label is well-formed, and if the label lies within the accreditation range of the receiving host. Well-formed packets that are within the receiving host's accreditation range are granted access.

IP packets that are exchanged between trusted systems can be labeled. Trusted Extensions supports Commercial IP Security Option (CIPSO) labels. A CIPSO label on a packet serves to classify, segregate, and route IP packets. Routing decisions compare the sensitivity label of the data with the destination.

Typically on a trusted network, the label is generated by a sending host and consumed by a receiving host. However, a trusted router can also add or strip labels while forwarding packets in a trusted network. A sensitivity label is mapped to a CIPSO label before transmission. The CIPSO label is embedded in the IP packet. Typically. a packet sender and the packet's receiver operate at the same label.

Trusted networking software ensures that the Trusted Extensions security policy is enforced even when the subjects (processes) and objects (data) are located on different hosts. Trusted Extensions networking preserves MAC across distributed applications.

Trusted Extensions Data Packets

Trusted Extensions data packets include a CIPSO label option. The data packets can be sent over IPv4 or IPv6 networks.

In the standard IPv4 format, the IPv4 header with options is followed by a TCP or UDP or SCTP header and then the actual data. The Trusted Extensions version of an IPv4 packet uses the CIPSO option in the IP header for security attribute.

IPv4 Header With CIPSO Option	TCP or UDP or SCTP	Data

In the standard IPv6 format, an IPv6 header with extensions is followed by a TCP or UDP or SCTP header and then the actual data. The Trusted Extensions IPv6 packet includes a multilevel security option in the header extensions.

IPv6 Header With Extensions	TCP or UDP or SCTP	Data

Trusted Network Communications

Trusted Extensions supports labeled and unlabeled hosts on a trusted network. LDAP is a fully supported naming service. Various commands and GUIs enable the network to be administered.

Systems that run Trusted Extensions software support network communications between Trusted Extensions hosts and any of the following types of computers:

- Other computers that are running Trusted Extensions
- Computers that are running operating systems that do not recognize security attributes, but do support TCP/IP, such as Solaris systems, other UNIX systems, Microsoft Windows, and Mac OS computers
- Computers that are running other trusted operating systems that recognize CIPSO labels

As in the Solaris OS, network communications and services can be managed by a naming service. Trusted Extensions adds the following interfaces to Solaris network interfaces:

- Trusted Extensions adds three network configuration databases, tnzonecfg, tnrhdb, and tnrhtp. For details, see "Network Configuration Databases in Trusted Extensions" on page 126.
- The Trusted Extensions version of the naming service switch file, nsswitch.conf, includes entries for the tnrhtp and tnrhdb databases. These entries can be modified to suit each site's configuration.

  Trusted Extensions uses the LDAP naming service to centrally manage configuration files that define hosts, networks and users. The default nsswitch.conf entries for the trusted network databases for the LDAP naming service is shown below.

  ```
  # Trusted Extensions
  tnrhtp: files ldap
  tnrhdb: files ldap
  ```

  The LDAP naming service on a Sun Java System Directory Server is the only fully supported naming service in Trusted Extensions. For the use of LDAP on a system that is configured with Trusted Extensions, see Chapter 9.
- Trusted Extensions adds tools to the Solaris Management Console. The console is used to centrally manage zones, hosts, and networks. The network tools are described in "Solaris Management Console Tools" on page 32.

  The *Solaris Trusted Extensions Installation and Configuration* guide describes how to define zones and hosts when configuring the network. For additional details, see Chapter 13.
- Trusted Extensions adds commands to administer trusted networking. Trusted Extensions also adds options to the Solaris network commands. For a description of the commands, see "Network Commands in Trusted Extensions" on page 126.

Network Configuration Databases in Trusted Extensions

Trusted Extensions loads three network configuration databases into the kernel. These databases are used in accreditation checks as data is transmitted from one host to another.

- `tnzonecfg` – This local database stores zone attributes that are security-related. The attributes for each zone specify the zone label and the zone's access to single-level and multilevel ports. Another attribute handles responses to control messages, such as `ping`. The labels for zones are defined in the `label_encodings` file. For more information, see the `label_encodings`(4) and `tnzonecfg`(4) man pages. For a discussion of multilevel ports, see "Zones and Multilevel Ports" on page 102.
- `tnrhtp` – This database stores templates that describe the security attributes of hosts and gateways. `tnrhtp` can be a local database, or stored on the LDAP server. Hosts and gateways use the attributes of the destination host and next-hop gateway to enforce MAC when sending traffic. When receiving traffic, hosts and gateways use the attributes of the sender. For details of the security attributes, see "Trusted Extensions Security Attributes" on page 127. For more information, see the `tnrhtp`(4) man page.
- `tnrhdb` – This database holds the IP addresses and network prefixes (fallback mechanism) that correspond to all hosts that are allowed to communicate. `tnrhdb` can be a local database, or stored on the LDAP server. Each host or network prefix is assigned a security template from the `tnrhtp` database. The attributes in the template define the attributes of the assigned host. For more information, see the `tnrhdb`(4) man page.

The Solaris Management Console has been extended to handle these databases. For details, see "Solaris Management Console Tools" on page 32.

Network Commands in Trusted Extensions

Trusted Extensions adds commands to administer trusted networking:

- `tnchkdb` – This command is used to verify the correctness of the trusted network databases. The `tnchkdb` command is used whenever you change a security template (`tnrhtp`), a security template assignment (`tnrhdb`), or the configuration of a zone (`tnzonecfg`). The Solaris Management Console tools run this command automatically when a database is modified. For details, see the `tnchkdb`(1M) man page.
- `tnctl` – This command can be used to update the trusted network information in the kernel. `tnctl` is also a system service. A restart with the command `svcadm restart /network/tnctl` refreshes the kernel cache from the trusted network databases on the local system. The Solaris Management Console tools run this command automatically when a database is modified in the Files scope. For details, see the `tnctl`(1M) man page.
- `tnd` – This daemon pulls `tnrhdb` and `tnrhtp` information from the LDAP directory. `tnd` is started at boot time as a service, as in `svcadm start /network/tnd`. This command also can be used for debugging and for changing the polling interval. For details, see the `tnd`(1M) man page.

- `tninfo` – This command displays the details of the current state of the trusted network kernel cache. The output can be filtered by host name, by zone, and by security template. For details, see the `tninfo`(1M) man page.

Trusted Extensions adds options to the Solaris network commands.

- `ifconfig` – The `all-zones` interface flag for this command makes the specified interface available to every zone on the system. The appropriate zone to deliver data to is determined by the label that is associated with the data. For details, see the `ifconfig`(1M) man page.
- `netstat` – The `-R` option extends Solaris `netstat` usage to display Trusted Extensions-specific information, such as security attributes for multilevel sockets and routing table entries. The extended security attributes include the label of the peer, and whether the socket is specific to a zone, or available to several zones. For details, see the `netstat`(1M) man page.
- `route` – The `-secattr` option extends Solaris `route` usage to display the security attributes of the route. The value of the option has the following format:

  ```
  min_sl=label,max_sl=label,doi=integer,cipso
  ```

  The `cipso` keyword is optional and set by default. For details, see the `route`(1M) man page.
- `snoop` – As in the Solaris OS, the `-v` option to this command can be used to display the IP headers in detail. The headers contain label information.

Trusted Extensions Security Attributes

Network administration in Trusted Extensions is based on the concept of security families. A security family is a set of hosts that have common protocols and identical security attributes.

Security attributes are administratively assigned to computers, both hosts and routers, by means of templates. The Security Administrator role administers templates and assigns them. If a computer does not have an assigned template, no communications are allowed with that computer.

Every template includes the following:

- A `Host Type` of either Unlabeled or CIPSO. The protocol that is used for network communication is determined by the host type of the template.

  The host type is used to determine whether to use CIPSO options, and affects MAC. See “Host Type in Security Templates” on page 129.
- A set of security attributes that are applied per host type.

For more detail about host types and security attributes, see “Network Security Attributes in Trusted Extensions” on page 128.

Network Security Attributes in Trusted Extensions

Trusted Extensions ships with a default set of security templates. When a template is assigned to a host, the security values in the template are applied to the host. In Trusted Extensions, both unlabeled hosts and labeled hosts on the network are assigned security attributes by means of a template. Hosts that are not assigned a security template cannot be reached. The templates can be stored locally, or in the LDAP naming service on the Sun Java System Directory Server.

Templates can be assigned directly or indirectly to a host. Direct assignment assigns a template to a particular IP address. Indirect assignment assigns a template to a network address that includes the host. Hosts that do not have a security template cannot communicate with hosts that are configured with Trusted Extensions. For an explanation of direct assignment and indirect assignment, see "Trusted Extensions Network Fallback Mechanism" on page 131.

Templates are modified or are created by using the Security Templates tool in the Solaris Management Console. The Security Templates tool enforces the required fields in the templates. The enforcement is based on host type.

Each host type has its own set of additional required and optional security attributes. The following security attributes are specified in networking templates:

- Host type – Defines whether the packets are labeled with CIPSO security labels, or are not labeled.
- Default label – Defines the level of trust of the unlabeled host. Packets that are sent by an unlabeled host are read at this label by the receiving Trusted Extensions host or gateway.
- DOI – An integer that identifies the domain of interpretation. The DOI is used to indicate which set of label encodings applies to a network communication or network entity. Labels with different DOIs, even if otherwise identical, are disjoint. For unlabeled hosts, the DOI applies to the default label.
- Minimum label – Defines the bottom of the label accreditation range. Hosts and next-hop gateways do not receive packets that are below the minimum label that is specified in their template.
- Maximum label – Defines the top of the label accreditation range. Hosts and next-hop gateways do not receive packets that are higher than the maximum label that is specified in their template.
- Security label set – Optional. Specifies a discrete set of security labels for a security template. In addition to their accreditation range that is determined by the maximum and minimum label, hosts that are assigned to a template with a security label set can send and receive packets that match any one of the labels in the label set. Four is the maximum number of labels that can be specified.

The Default Label attribute is specific to the unlabeled host type. For details, see the `tnrhtp`(4) man page and the following sections.

Host Type in Security Templates

Trusted Extensions support two host types in the trusted network databases. The first column shows the name used in the Security Templates host type menu in the Solaris Management Console. Trusted Extensions classifies host types according to the networking protocols, as indicated in the second column of the table.

CIPSO host type

The CIPSO host type is intended for hosts that run trusted operating systems. Trusted Extensions supplies one template for this host type, `cipso`.

Common IP Security Option (CIPSO) protocol is used to specify security labels that are passed in the IP options field. CIPSO labels are derived automatically from the data's label. Tag type 1 is used to pass the CIPSO security label. This label is then used to make security checks at the IP level and to label the data in the network packet.

Unlabeled host type

The unlabeled host type is intended for those hosts that use standard networking protocols but do not support CIPSO options. Trusted Extensions supplies one template for this host type, `admin_low`.

This host type is assigned to hosts that run the Solaris OS or other unlabeled operating systems. This host type gives a default label and default clearance to apply to communications with the unlabeled host. Also, a label range or a list of discrete labels can be set to allow the sending of packets to an unlabeled gateway for forwarding.

Default Security Templates

Templates are created according to host type. A host that is running Solaris Trusted Extensions 1.0 and compatible releases can be assigned any template that has the CIPSO host type. The `cipso` template is provided for CIPSO host types.

Trusted Extensions supports communications with hosts that run operating systems that do not recognize labels, such as the Solaris OS. A host that does not recognize labels is an unlabeled host. The `admin_low` template is provided as an example for unlabeled hosts.

Caution – The `admin_low` template provides an example for constructing unlabeled templates with site-specific labels. While the `admin_low` template is required for installation of Trusted Extensions, the security settings might not be appropriate for normal system operations. It is recommended that the provided templates be retained without modification for system maintenance and support reasons.

Default Label in Security Templates

Templates for the unlabeled host type specify a Default Label. This label is used to control communications with hosts whose operating systems are not aware of labels, such as Solaris systems. The Default Label that is assigned reflects the level of trust that is appropriate for the host and its users.

Because communications with unlabeled hosts are essentially limited to the default label, these hosts are also referred to as single-label hosts.

Domain of Interpretation (DOI) in Security Templates

Organizations that use the same DOI agree among themselves to interpret label information, and other security attributes, in the same way. When Trusted Extensions performs a label comparison check, the check includes checking that the DOI is equal.

Label Range in Security Templates

The Minimum Label and Maximum Label attributes are used to establish the label range for labeled and unlabeled hosts. These attributes are used to do the following:

- To set the range of labels that can be used when communicating with a remote CIPSO host.

  In order for a packet to be sent to a destination host, the label of the packet must be within the label range assigned to the destination host in the security template for that host.
- To set a label range for packets that are being forwarded through a CIPSO gateway or an unlabeled gateway.

  The label range can be specified in the template for an unlabeled host type. The label range enables the host to forward packets that are not necessarily at the label of the host, but that are within a specified label range.

Security Label Set in Security Templates

The security label set defines at most four discrete labels at which packets can be accepted, forwarded, or sent by the remote host. This attribute is optional. By default, no security label set is defined.

Trusted Extensions Network Fallback Mechanism

The `tnrhdb` database can assign a security template to a particular host either directly or indirectly. Direct assignment is by the host's IP address. Indirect assignment is by a fallback mechanism. The trusted network software first looks for an entry that specifically assigns the host's IP address to a template. If it does not find a specific entry for the host, the software looks for the "longest prefix of matching bits". You can indirectly assign a host to a security template when the IP address of the host falls within "longest prefix of matching bits" of an IP address with a fixed prefix length.

In IPv4, you can make an indirect assignment by subnet. When you make an indirect assignment by using 4, 3, 2,or 1 trailing zero (0) octets, the software calculates a prefix length of 0, 8, 16, or 24, respectively. Entries 3 through 6 in the following table illustrate this fallback mechanism.

You can also set a fixed prefix length by adding a slash (/) followed by the number of fixed bits. IPv4 network addresses can have a prefix length between 1 and 32. IPv6 network addresses can have a prefix length between 1 and 128. The following table provides fallback examples. If an address within the set of fallback addresses is directly assigned, the fallback mechanism is not used for that address.

TABLE 12–1 `tnrhdb` Fallback Mechanism Entries

IP Version	`tnrhdb` Entry	Addresses Covered
IPv4	192.168.118.128.57:cipso	192.168.118.57
	192.168.118.128/26:cipso	From 192.168.118.0 through 192.168.118.63
	192.168.118.0:cipso	Starting with 192.168.118.
	192.168.0.0:cipso	Starting with 192.168.
	192.0.0.0:cipso	Starting with 192.
	0.0.0.0:cipso	All addresses on network
IPv6	2001\:DB8\:22\:5000\:\:21f7:cipso	2001:DB8:22:5000::21f7
	2001\:DB8\:22\:5000\:\:0/52:cipso	From 2001:DB8:22:5000::0 through 2001:DB8:22:5fff:ffff:ffff:ffff:ffff
	0\:\:0/0:cipso	All addresses on network

For more information on prefix lengths in IPv4 and IPv6 addresses, see "Designing Your CIDR IPv4 Addressing Scheme" in *System Administration Guide: IP Services* and "IPv6 Addressing Overview" in *System Administration Guide: IP Services*.

Overview of Routing in Trusted Extensions

In Trusted Extensions, routes between hosts on different networks must maintain security at each step in the transmission. Trusted Extensions adds extended security attributes to the routing protocols in the Solaris OS. Unlike the Solaris OS, this Trusted Extensions release does not support dynamic routing. For details on specifying static routing, see the `-p` option in the `route`(1M) man page.

Gateways and routers route packets. In this discussion, the terms gateway and router are used interchangeably.

For communications between hosts on the same subnet, accreditation checks are performed at endpoints only because no routers are involved. Label range checks are performed at the source. If the receiving host is running Trusted Extensions software, label range checks are also performed at the destination.

When the source and destination hosts are on different subnets, the packet is sent from the source host to a gateway. The label range of the destination and of the first-hop gateway is checked at the source when selecting a route. The gateway forwards the packet to the network where the destination host is connected. A packet might go through a number of gateways before reaching the destination.

Background on Routing

On Trusted Extensions gateways, label range checks are performed in certain cases. A Trusted Extensions computer that is routing a packet between two unlabeled hosts compares the default label of the source host to the default label of the destination host. When the unlabeled hosts share a default label, the packet is routed.

Each gateway maintains a list of routes to all destinations. Standard Solaris routing makes choices to optimize the route. Extensions in Trusted Extensions software check security requirements that apply to the route choices. The Solaris choices that do not satisfy security requirements are skipped.

Routing Table Entries in Trusted Extensions

The routing table entries in Trusted Extensions can incorporate security attributes. Security attributes can include a `cipso` keyword. Security attributes must include a maximum label, a minimum label, and a DOI.

For entries that do not provide security attributes, the attributes in the gateway's security template are used.

Trusted Extensions Accreditation Checks

Trusted Extensions software determines the suitability of a route for security. The software runs a series of tests called *accreditation checks* on the source host, destination host, and the intermediate gateways.

Note – In the following discussion, an accreditation check for a label range also means a check for a security label set.

The accreditation check verifies the label range and CIPSO label information. The security attributes for a route are obtained from the routing table entry, or from the security template of the gateway if the entry has no security attributes.

For incoming communications, the Trusted Extensions networking software obtains labels from the packets themselves whenever possible. Obtaining labels from packets is only possible when the messages are sent from systems that support labels. When the label is not all available from a packet, a default label is assigned to the message from trusted networking database files. These labels are then used during accreditation checks. Trusted Extensions enforces several checks on outgoing messages, forwarded messages, and incoming messages.

Source Accreditation Checks

The following accreditation checks are performed on the sending process or sending zone:

- For all destinations, the label of the data must be within the label range of the next hop in the route, that is, the first hop. And, the label must be in the first-hop gateway's security attributes.
- For all destinations, the DOI of an outgoing packet must match the DOI of the destination host. The DOI must also match the DOI of all hops along the route, including its first-hop gateway.
- When the destination host is an unlabeled host, one of the following conditions must be satisfied:
  - The sending host's label must match the destination host's default label.
  - The sending host is privileged to do cross-label communication and sender's label dominates destination's default label.
  - The sending host is privileged to do cross-label communication and sender's label is `ADMIN_LOW`. That is, the sender is sending from the global zone.

Note – A first-hop check occurs when a message is being sent from a host on one network to a host on another network, through a gateway.

Gateway Accreditation Checks

On a Trusted Extensions gateway computer, accreditation checks are performed for the next-hop gateway:

- If the incoming packet is unlabeled, the packet inherits the source host's default label from the `tnrhdb` entry. Otherwise, the packet receives the indicated CIPSO label.
- Forwarding a packet proceeds similar to source accreditation:
  - For all destinations, the label of the data must be within the label range of the next hop. And, the label must be in the security attributes of the next-hop host.

- For all destinations, the DOI of an outgoing packet must match the DOI of the destination host. The DOI must also match the DOI of the next-hop host.
- The label of an unlabeled packet must match the destination host's default label.
- The label of a CIPSO packet must be within the destination host's label range.

Destination Accreditation Checks

When a Trusted Extensions host receives data, the trusted network software performs the following checks:

- If the incoming packet is unlabeled, the packet inherits the source host's default label from the `tnrhdb` entry. Otherwise, the packet receives the indicated CIPSO label.
- The label and DOI for the packet must be consistent with the destination zone or destination process' label and DOI. The exception is when a process is listening on a port. The listening process can receive a packet if the process is privileged to do cross-label communications, and either is in the global zone, or has a label that dominates the packet's label.

Administration of Routing in Trusted Extensions

Trusted Extensions supports several methods for routing communications between networks. In the Security Administrator role, you can set up routes that enforce the degree of security required by your site's security policy.

For example, sites can restrict communications outside of the local network to a single label. This label is applied to publicly-available information. Labels such as `UNCLASSIFIED` or `PUBLIC` can indicate public information. To enforce the restriction, these sites assign a single-label template to the network interface that is connected to the external network. For more details about TCP/IP and routing, see the following sections in the *System Administration Guide: IP Services*:

- "Planning for Routers on Your Network" in *System Administration Guide: IP Services*
- "Configuring Systems on the Local Network " in *System Administration Guide: IP Services*
- "Major TCP/IP Administrative Tasks (Task Map)" in *System Administration Guide: IP Services*
- "Preparing Your Network for the DHCP Service (Task Map)" in *System Administration Guide: IP Services*

Choosing Routers in Trusted Extensions

Trusted Extensions hosts offer the highest degree of trust as routers. Other types of routers might not recognize Trusted Extensions security attributes. Without administrative action, packets can be routed through routers that do not provide MAC security protection.

- CIPSO routers drop packets when they do not find the right type of information in the IP options section of the packet. For example, a CIPSO router drops a packet if it does not find a CIPSO option in the IP options when one is required, or when the DOI in the IP options is not consistent with the destination's accreditation.
- Other types of routers that are not running Trusted Extensions software can be configured to either pass the packets or drop the packets that include the CIPSO option. Only CIPSO-aware gateways such as Trusted Extensions provides can use the contents of the CIPSO IP option to enforce MAC.

To support trusted routing, the Solaris Express routing tables are extended to include Trusted Extensions security attributes. The attributes are described in "Routing Table Entries in Trusted Extensions" on page 132. Trusted Extensions supports static routing, where the administrator creates routing table entries manually. For details, see the `-p` option in the `route`(1M) man page.

The routing software tries to find a route to the destination host in the route tables. When the host is not explicitly named, the routing software looks for an entry for the subnetwork where the host resides. When neither the host nor the network where the host resides is defined, the host sends the packet to a default gateway, if one has been defined. Multiple default gateways can be defined, and each is treated equally.

In this release of Trusted Extensions, the security administrator sets up routes manually, and then manually makes changes to the routing table when conditions change. For example, many sites have a single gateway that communicates with the outside world. In these cases, the single gateway can be statically defined as the *default* on each host on the network. Dynamic routing support will be available in future releases of Trusted Extensions.

Gateways in Trusted Extensions

An example of routing in Trusted Extensions is shown in the following figure. The routing diagram and routing table show three potential routes between Host 1 and Host 2.

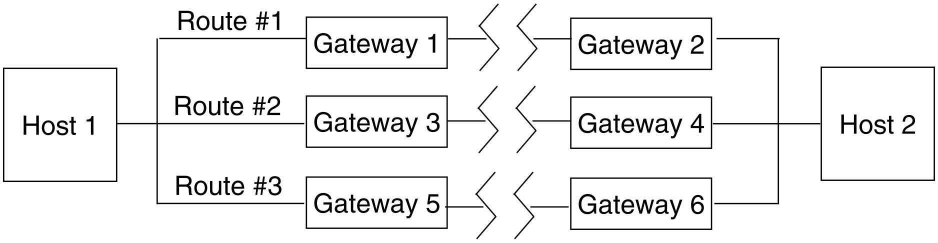


FIGURE 12–1 Typical Trusted Extensions Routes and Routing Table Entries

Route	First-Hop Gateway	Min Label	Max Label	DOI
1	Gateway 1	`CONFIDENTIAL`	`SECRET`	1
2	Gateway 3	`ADMIN_LOW`	`ADMIN_HIGH`	1
3	Gateway 5			

- Route #1 can transmit packets within the a label range of `CONFIDENTIAL` to `SECRET`.
- Route #2 can transmit packets from `ADMIN_LOW` to `ADMIN_HIGH`.
- Route #3 does not specify routing information. Therefore, its security attributes are derived from the template in the `tnrhtp` database for Gateway #5.

Routing Commands in Trusted Extensions

To show labels and extended security attributes for sockets, Trusted Extensions modifies Solaris network commands.

- The `netstat -rR` command displays the security attributes in routing table entries.
- The `netstat -aR` command displays the security attributes for sockets.
- The `route -p` command with the `add` or `delete` option changes the routing table entries.

For details, see the `netstat`(1M) and the `route`(1M) man pages.

For examples, see “How to Configure Routes With Security Attributes” on page 149.

CHAPTER 13

Managing Networks in Trusted Extensions

This chapter provides implementation details and procedures for securing a Trusted Extensions network.

- "Managing the Trusted Network (Task Map)" on page 137
- "Configuring Trusted Network Databases (Tasks)" on page 138
- "Configuring Routes and Checking Network Information in Trusted Extensions (Tasks)" on page 149
- "Troubleshooting the Trusted Network (Tasks)" on page 154

Managing the Trusted Network (Task Map)

The following table points to the task maps for common networking procedures.

Task	Description	For Instructions
Configure network databases	Create remote host templates, and assign hosts to the templates.	"Configuring Trusted Network Databases (Tasks)" on page 138
Configure routing, and check network databases and network information in the kernel	Configure static routes that enable labeled packets to reach their destination through labeled and unlabeled gateways. Display the state of your network.	"Configuring Routes and Checking Network Information in Trusted Extensions (Tasks)" on page 149
Troubleshoot networking problems	Steps to take when diagnosing network problems with labeled packets.	"Troubleshooting the Trusted Network (Tasks)" on page 154

Configuring Trusted Network Databases (Tasks)

Trusted Extensions software includes the `tnrhtp` and `tnrhdb` databases. These databases provide labels for remote hosts that contact the system. The Solaris Management Console provides the GUI to administer these databases.

Task	Description	For Instructions
Determine if your site requires customized security templates	Evaluate the existing templates for the security requirements of your site.	"How to Determine If You Need Site-Specific Security Templates" on page 139
Access the Security Templates tool in the Solaris Management Console	Access the tool for modifying trusted network databases.	"How to Open the Trusted Networking Tools" on page 139
Work with security templates	Modify the definitions of security attributes in your trusted network by modifying the trusted network databases.	"How to Construct a Remote Host Template" on page 140
Create an unlabeled security template	Create security templates for unlabeled computers and networks.	Example 13–1
Restrict a security template to a single label	Create a security template for hosts that restrict communication between other hosts to a single label.	Example 13–1
Create a template for a single-label gateway	Create a security template for hosts that operate as single-label gateways.	Example 13–2
Create a CIPSO template with a restricted label range	Create security templates for labeled computers and networks.	Example 13–3
Add hosts to the known network	Add computers and networks that you plan to add to the trusted network.	"How to Add Hosts to the System's Known Network" on page 144
Provide remote host access by using wildcard entries	Allow hosts within a range of IP addresses to communicate with this system by indirectly assigning each host to the same security template.	Example 13–7 Example 13–8 Example 13–9
Change the `admin_low` wildcard entry in the `tnrhdb` file	Increase security by replacing the wildcard entry with specific addresses for the host to contact at boot time.	"How to Limit the Hosts That Can Be Contacted on the Trusted Network" on page 147
	Increase security by replacing the wildcard entry with a network of labeled hosts as the default.	Example 13–10
Assign security templates	Associate a template with an IP address or list of contiguous IP addresses.	"How to Assign a Template to a Group of Hosts" on page 145

▼ How to Determine If You Need Site-Specific Security Templates

Before You Begin You must be in the global zone in a role. The Security Administrator role is responsible for creating security templates.

1. **Familiarize yourself with the Trusted Extensions templates.**

   Read the `tnrhtp` file on a local host. The comments in the file are helpful. You can also see the security attribute values in the Security Templates tool in the Solaris Management Console.

   - The default templates match any installation. The label range for each template is `ADMIN_LOW` to `ADMIN_HIGH`.
   - The `cipso` template defines a CIPSO host type whose DOI is `1`. The label range for the template is `ADMIN_LOW` to `ADMIN_HIGH`.
   - The `admin_low` template defines an unlabeled host whose DOI is `1`. Its default label is `ADMIN_LOW`. The label range for the template is `ADMIN_LOW` to `ADMIN_HIGH`. In the default configuration, the address `0.0.0.0` is assigned to this template. Therefore, all non-CIPSO hosts are treated as hosts that operate at the `ADMIN_LOW` security label.

2. **Keep the default templates.**

   For support purposes, do not delete or modify the default templates.

3. **Create new templates if you want to do any of the following:**

   - Limit the label range of a host or a group of hosts.
   - Create a single-label host.
   - Create a host that recognizes a few discrete labels.
   - Use a different DOI than `1`.
   - Require a default label for unlabeled hosts that is not `ADMIN_LOW`.

   For details, see "How to Construct a Remote Host Template" on page 140.

▼ How to Open the Trusted Networking Tools

Before You Begin You must be in the global zone in a role that can modify network security. For example, roles that are assigned the Information Security or Network Security rights profiles can modify security settings. The Security Administrator role includes these profiles.

To use the LDAP toolbox, you must have completed "Configuring the Solaris Management Console for LDAP (Tasks)" in *Solaris Trusted Extensions Installation and Configuration*.

1. **Use the Security Templates tool.**

   a. **Launch the Solaris Management Console.**

```
# /usr/sbin/smc &
```

b. **Open the Trusted Extensions toolbox for your computer.**

   Use the Console → Open Toolbox menu item.

   i. **Select a toolbox whose `Policy=TSOL`.**

   ii. **Click Open.**

c. **Open System Configuration.**

d. **Click the Computers and Networks tool.**

   Provide a password when prompted.

2 **Use the appropriate tool.**

- To modify a template, use the Security Templates tool.

  All currently-defined templates display in the right-hand pane. When you select or create a template, online help is available in the left-hand pane.
- To assign a host to a template, use the Security Templates tool.
- To create a host that can be assigned to a template, use the Computers and Networks tool.
- To assign a label to a zone, use the Trusted Network Zones tool. For more on zones in Trusted Extensions, see Chapter 10.

▼ How to Construct a Remote Host Template

Before You Begin You must be in the global zone in a role that can modify network security. For example, roles that are assigned the Information Security or Network Security rights profiles can modify security settings. The Security Administrator role includes these profiles.

1 **Open the Security Templates tool.**

See "How to Open the Trusted Networking Tools" on page 139 for the steps in detail.

2 **Double-click Security Templates under Computers and Networks in the Solaris Management Console.**

The existing templates are displayed in the View pane. These templates describe the security attributes for hosts that this machine can contact. These machines include CIPSO hosts that are running Trusted Extensions and unlabeled hosts.

3 **Examine the `cipso` template.**

a. **Examine the General tab.**

   Look at the values for Host Type, Default Label, DOI, and Minimum/Maximum Label.

b. **Examine the Hosts tab.**

   See what computers and network are already assigned this template.

4 **Examine the `admin_low` template.**

Look at the tabs as you did for the `cipso` template.

5 **(Optional) Modify an existing template.**

Double-click the template, and use the online help for assistance. You can change the assigned computers or change the assigned networks.

Note – Do not modify the definition of the default templates. Retain them for support purposes.

---

6 **(Optional) Create a template.**

If the provided templates do not sufficiently describe the computers that can be in communication with this computer, choose Add Template from the Action menu.

Use the online help for assistance. Before continuing, create all the templates that your site requires.

Example 13–1 Creating a Security Template That Has a Single Label

In this example, the security administrator wants to create a gateway that can only pass packets at a single label, `PUBLIC`. In the Solaris Management Console, the administrator creates a template. Still in the Security Templates tool, she assigns the gateway host to the template.

1. The gateway host and IP address are added to the Computers and Networks tool.

   ```
   gateway-1
   192.168.131.75
   ```

2. The template is created in the Security Templates tool.

   The values in the template are the following:

   ```
   template: CIPSO_PUBLIC
   host_type: CIPSO
   doi: 1
   min_sl: PUBLIC
   max_sl: PUBLIC
   ```

   The tool supplied the hexadecimal value for `PUBLIC`, `0X0002-08-08`.

3. The `gateway-1` host is assigned to the template by its name and IP address.

   ```
   gateway-1
   192.168.131.75
   ```

On a local host, the `tnrhtp` entry would appear similar to the following:

```
cipso_public:host_type=cipso;doi=1;min_sl=0X0002-08-08;max_sl=0X0002-08-08;
```

On a local host, the tnrhdb entry would appear similar to the following:

```
# gateway-1
192.168.131.75:cipso_public
```

Example 13–2 Creating a Security Template For an Unlabeled Router

Any IP router can forward messages with CIPSO labels even though the router does not explicitly support labels. Such an unlabeled router needs a default label to define the level at which connections to the router, perhaps for router management, should be handled. In this example, the security administrator creates a router that can forward traffic at any label, but all direct communication with the router is handled at the default label, PUBLIC.

In the Solaris Management Console, the administrator creates a template. Still in the Security Templates tool, she assigns the gateway host to the template.

1. The router and its IP address are added to the Computers and Networks tool.

   ```
   router-1
   192.168.131.82
   ```

2. The template is created in the Security Templates tool.

   The values in the template are the following:

   ```
   Template Name: UNL_PUBLIC
   Host Type: UNLABELED
   DOI: 1
   Default Label: PUBLIC
   Minimum Label: ADMIN_LOW
   Maximum Label: ADMIN_HIGH
   ```

   The tool supplies the hexadecimal value for the labels.

3. The router-1 router is assigned to the template by its name and IP address.

   ```
   router-1
   192.168.131.82
   ```

Example 13–3 Creating a Security Template That Has a Limited Label Range

In this example, the security administrator wants to create a gateway that restricts packets to a narrow label range. In the Solaris Management Console, the administrator creates a template. Still in the Security Templates tool, she assigns the gateway host to the template.

1. The host and its IP address are added to the Computers and Networks tool.

   ```
   gateway-ir
   192.168.131.78
   ```

2. The template is created in the Security Templates tool.

   The values in the template are the following:

   ```
   Template Name: CIPSO_IUO_RSTRCT
   Host Type: CIPSO
   DOI: 1
   Minimum Label: CONFIDENTIAL : INTERNAL USE ONLY
   Maximum Label: CONFIDENTIAL : RESTRICTED
   ```

   The tool supplies the hexadecimal value for the labels.

3. The `gateway-ir` gateway is assigned to the template by its name and IP address.

   ```
   gateway-ir
   192.168.131.78
   ```

Example 13–4 Creating a Security Template That Has a Sensitivity Label Set

In this example, the security administrator wants to create a security template that recognizes two labels only. In the Solaris Management Console, the administrator creates a template. Still in the Security Templates tool, she assigns the gateway host to the template.

1. Each host and IP address that is going to use this template is added to the Computers and Networks tool.

   ```
   host-slset1
   192.168.132.21

   host-slset2
   192.168.132.22

   host-slset3
   192.168.132.23

   host-slset4
   192.168.132.24
   ```

2. The template is created in the Security Templates tool.

   The values in the template are the following:

   ```
   Template Name: CIPSO_PUB_RSTRCT
   Host Type: CIPSO
   DOI: 1
   Minimum Label: PUBLIC
   Maximum Label: CONFIDENTIAL : RESTRICTED
   SL Set: PUBLIC, CONFIDENTIAL : RESTRICTED
   ```

   The tool supplies the hexadecimal value for the labels.

3. The range of IP addresses are assigned to the template by using the Wildcard button and a prefix.

   ```
   192.168.132.0/17
   ```

Example 13–5 Creating an Unlabeled Template at the Label PUBLIC

In this example, the security administrator allows a subnetwork of Solaris computers to be treated as PUBLIC in the trusted network. The template has the following values:

```
Template Name: public
Host Type: Unlabeled
Default Label: Public
Minimum Label: Public
Maximum Label: Public
DOI: 1

Wildcard Entry: 10.10.0.0
Prefix: 16
```

All machines on the 10.10.0.0 subnetwork are treated as being labeled PUBLIC.

Example 13–6 Creating a Labeled Template for Developers

In this example, the security administrator creates a SANDBOX template. This template is assigned to machines that are used by developers of trusted software. The two machines that are assigned this template can create and test labeled programs. However, their tests do not affect the other labeled systems. The label SANDBOX is disjoint from the other labels on the network.

```
Template Name: cipso_sandbox
Host Type: CIPSO
Minimum Label: SANDBOX
Maximum Label: SANDBOX
DOI: 1

Hostname: DevMachine1
IP Address: 196.168.129.129

Hostname: DevMachine2
IP Address: 196.168.129.102
```

The users of these machines can communicate with each other at the label SANDBOX.

▼ How to Add Hosts to the System's Known Network

This tool is identical to the tool in the Solaris OS. This procedure is provided for convenience. After the hosts are known, you then assign the hosts to a security template.

Before You Begin You must be in an administrator who can manage networks. For example, roles that include the Network Management or System Administrator rights profiles can manage networks.

1. **Open the Solaris Management Console.**

   For details, see "How to Open the Trusted Networking Tools" on page 139.

2. **Navigate from System Configuration to the Computers tool.**

   When prompted, confirm that you want to see all computers on the network.

3. **Add a host that this computer can contact.**

   You should add every host that his system might contact, including any static routers, and any audit servers.

   a. **Choose Add Computer from the Action menu.**

   b. **Click Apply to add the host.**

   c. **Click OK when the entries are complete.**

4. **Add a group of hosts that this computer can contact.**

   Follow the online help to add groups of computers by using a wildcard.

▼ How to Assign a Template to a Group of Hosts

Before You Begin You must be in the global zone in a role. The Security Administrator role modifies security templates and host assignments.

All hosts that you want to assign to a template must exist in the Computers and Networks tool. For details, see "How to Add Hosts to the System's Known Network" on page 144.

1. **In the Security Administrator role, open the Security Templates tool.**

   For details, see "How to Open the Trusted Networking Tools" on page 139.

2. **Double-click the template name.**

3. **Click the Hosts Assigned to Template tab.**

4. **To add a single host, do the following:**

   a. **Type its name in the Hostname field.**

   b. **Type its address in the IP Address field.**

   c. **Click the Add button.**

d. **Click OK to save the changes.**

5 **To add a group of hosts with contiguous addresses, do the following:**

a. **Click Wildcard.**

b. **Type an IP address in the IP Address field.**

c. **Type a prefix that describes the group of contiguous addresses in the Prefix field.**

d. **Click the Add button.**

e. **Click OK to save the changes.**

Example 13–7 Adding an IPv4 Network as a Wildcard Entry

In the following example, a security administrator wanted several IPv4 subnetworks to be assigned to the same security template. In the Hosts Assigned to Template tab, the administrator added the following wildcard entries:

```
IP Address: 192.168.113.0
IP address: 192.168.75.0
```

Example 13–8 Adding a List of IPv4 Hosts as a Wildcard Entry

In the following example, a security administrator wanted contiguous IPv4 addresses that were not along octet boundaries to be assigned to the same security template. In the Hosts Assigned to Template tab, the administrator added the following wildcard entries:

```
IP Address: 192.168.113.100
Prefix Length: 25
```

This wildcard entry covers the address range of `192.168.113.0` to `192.168.113.127`. The address includes `192.168.113.100`.

Example 13–9 Adding a List of IPv6 Hosts as a Wildcard Entry

In the following example, a security administrator wanted contiguous IPv6 addresses to be assigned to the same security template. In the Hosts Assigned to Template tab, the administrator added the following wildcard entries:

```
IP Address: 2001:a08:3903:200::0
Prefix Length: 56
```

This wildcard entry covers the address range of `2001:a08:3903:200::0` to `2001:a08:3903:2ff:ffff:ffff:ffff:ffff`. The address includes `2001:a08:3903:201:20e:cff:fe08:58c`.

▼ How to Limit the Hosts That Can Be Contacted on the Trusted Network

This procedure protects labeled hosts from being contacted by arbitrary unlabeled hosts. The default template when Trusted Extensions is installed defines every host on the network. Use this procedure to enumerate specific unlabeled hosts.

The local `tnrhdb` file on each computer is used to contact the network at boot time. By default, every host that is not provided with a CIPSO template is defined by the `admin_low` template. This template assigns every computer that is not otherwise defined, that is, `0.0.0.0`, to be an unlabeled computer with the default label of `admin_low`.

Caution – Security Issue – The default `admin_low` template can be a security risk on a Trusted Extensions network. If site security requires strong protection, the Security Administrator role can remove the `0.0.0.0` entry after the computer is installed. The entry must be replaced with entries for every computer that the host contacts during boot.

For example, DNS servers, home directory servers, audit servers, broadcast and multicast addresses, and routers would need to be in the local `tnrhdb` file after the `0.0.0.0` wildcard entry is removed.

Before You Begin You must be in the Security Administrator role in the global zone.

All hosts that are to be contacted at boot must exist in the Computers and Networks tool.

1. **Open the Security Templates tool in the Files scope.**

   The Files scope protects the system during boot. To access the Security Templates tool, see "How to Open the Trusted Networking Tools" on page 139.

2. **Open the `admin_low` template.**

   Click the Hosts Assigned to Template tab. Every host that is added can be contacted during boot at the label `ADMIN_LOW`.

   a. **Add every individual unlabeled host that should be contacted at boot time.**

      Include every on-link router that is not running Trusted Extensions, through which this host must communicate.

   b. **Add ranges of hosts that should be contacted at boot time.**

   c. **Remove the `0.0.0.0` entry.**

3 **Open the `cipso` template.**

Click the Hosts Assigned to Template tab. Every labeled host that is added can be contacted during boot.

a. **Add every individual labeled host that should be contacted at boot time.**

- Include the LDAP server.
- Include every on-link router that is running Trusted Extensions, through which this host must communicate
- Make sure that all network interfaces are assigned to the template.
- Include broadcast addresses.

b. **Add groups of hosts that should be contacted at boot time.**

4 **Verify that the host assignments allow the computer to boot.**

Example 13–10 Changing the Label of the `0.0.0.0` `tnrhdb` Entry

In this example, the security administrator creates a public gateway system. She removed the `0.0.0.0` entry from the `admin_low` template, and assigns the entry to an unlabeled template that is named `public`. The system then recognizes any computer that is not listed in its `tnrhdb` file as an unlabeled system with the security attributes of the `public` security template.

The following describes an unlabeled template that was created specifically for public gateways.

```
Template Name: public
Host Type: Unlabeled
Default Label: Public
Minimum Label: Public
Maximum Label: Public
DOI: 1
```

Example 13–11 Enumerating Computers to Contact During Boot in the `tnrhdb`

The following example shows the local `tnrhdb` file with entries for an LDAP client with two network interfaces. The client communicates with another network and routers.

```
127.0.0.1:cipso            Loopback address
192.168.112.111:cipso       Interface 1 of this host
192.168.113.111:cipso       Interface 2 of this host
10.6.6.2:cipso              LDAP server
192.168.113.6:cipso         Audit server
192.168.112.255:cipso       Subnet broadcast address
192.168.113.255:cipso       Subnet broadcast address
192.168.113.1:cipso         Router
```

```
192.168.117.0:cipso        Another Trusted Extensions network
192.168.112.12:public      Specific network router
192.168.113.12:public      Specific network router
224.0.0.2:public           Multicast address
255.255.255.255:admin_low       Broadcast address
```

Configuring Routes and Checking Network Information in Trusted Extensions (Tasks)

Task	Description	For Instructions
Configure static routes	Manually describe the best route from one host to another.	"How to Configure Routes With Security Attributes" on page 149
Check database accuracy	Use the `tnchkdb` command to check the syntactic validity of the local network databases.	"How to Check the Syntax of Trusted Network Databases" on page 151
Compare the network database entries with the entries in the kernel cache	Use the `tninfo` command to see if the kernel cache has been updated with the latest database information.	"How to Compare Network Database Information With Kernel Cache" on page 151
Synchronize the kernel cache with the network databases	Use the `tnctl` command to update the kernel cache with up-to-date network database information on a running system.	"How to Synchronize Kernel Cache With Network Databases" on page 153

▼ How to Configure Routes With Security Attributes

Before You Begin You must be in the Security Administrator role in the global zone.

1 Add every destination host and gateway that you are using in routing.

The addresses are added to the local `/etc/hosts` file, or to its equivalent on the LDAP server. Use the Computers and Networks tool in the Solaris Management Console. For details, see "How to Open the Trusted Networking Tools" on page 139.

2 Assign each destination host, network, and gateway to a security template.

The addresses are added to the local `/etc/security/tsol/tnrhdb` file, or to its equivalent on the LDAP server. Use the Security Templates tool in the Solaris Management Console. For details, see "How to Add Hosts to the System's Known Network" on page 144 and "How to Assign a Template to a Group of Hosts" on page 145.

3 Set up the routes.

In a terminal, use the `route add` command to specify routes. Use the `-secattr` flag to specify security attributes.

The first entry sets up a default route. The entry specifies a gateway's address, 192.168.113.1, to use when no specific route is defined for either the host or the packet's destination.

```
route add default gateway-address -interface|-static

route add default 192.168.113.1  -static
```

For details, see the route(1M) man page.

4 **Set up one or more network entries.**

In the following list of commands, the second line shows a network entry. The third line shows a network entry with a label range of PUBLIC to INTERNAL.

```
default 192.168.113.36  1
net 192.168.102.0 gateway-101
net 192.168.101.0 gateway-102 -secattr min_sl="PUBLIC",
max_sl="CONFIDENTIAL: INTERNAL USE ONLY",doi=1
```

5 **Set up one or more host entries.**

The new fourth line shows a host entry for the gateway gateway-pub. gateway-pub has a label range of PUBLIC to PUBLIC.

```
default 192.168.113.36
net 192.168.102.0 gateway-101
net 192.168.101.0 gateway-102 -secattr min_sl="PUBLIC",
max_sl="CONFIDENTIAL : INTERNAL USE ONLY",doi=1
host 192.168.101.3 gateway-pub -secattr min_sl="PUBLIC",
max_sl="PUBLIC",doi=1
```

Example 13–12 Adding a Route With a Label Range of C to TS

The following route command adds to the routing table the hosts at 192.168.115.0 and 192.168.118.39 with a label range from C to TS, and a DOI of 1.

```
$ route add net 192.168.115.0 192.168.118.39 \
-secattr min_sl=c,max_sl=ts,doi=1

add net 192.168.115.0: gateway 192.168.118.39
```

The result of the added hosts is shown with the netstat -rR command. In the following excerpt, the other routes are replaced by ellipses (...).

```
$ netstat -rRn
...
192.168.115.0        192.168.118.39        UG       0      0
        min_sl=C,max_sl=TS,DOI=1,CIPSO
...
```

How to Check the Syntax of Trusted Network Databases

This command checks that the syntax of each network database is accurate. The Solaris Management Console runs this command automatically when you use the Security Templates tool or the Trusted Network Zones tool.

Before You Begin You must be in the global zone in a role that can check network settings. The Security Administrator role and the System Administrator role can check these settings.

Run the `tnchkdb` command.

```
$ tnchkdb
checking /etc/security/tsol/tnrhtp ...
checking /etc/security/tsol/tnrhdb ...
checking /etc/security/tsol/tnzonecfg ...
```

Example 13–13 Testing the Syntax of a Trial Network Database

In this example, the security administrator is testing a network database file for possible use. Initially, the administrator uses the wrong option. The results of the check are printed on the line for the `tnrhdb` file:

```
$ tnchkdb -h /opt/secfiles/trial.tnrhtp
checking /etc/security/tsol/tnrhtp ...
checking /opt/secfiles/trial.tnrhtp ...
line 12: Illegal name: min_sl=ADMIN_LOW;max_sl=ADMIN_HIGH
line 14: Illegal name: min_sl=ADMIN_LOW;max_sl=ADMIN_HIGH
checking /etc/security/tsol/tnzonecfg ...
```

When the security administrator checks the file by using the `-t` option, the command confirms that the syntax of the alternate file is accurate:

```
$ tnchkdb -t /opt/secfiles/trial.tnrhtp
checking /opt/secfiles/trial.tnrhtp ...
checking /etc/security/tsol/tnrhdb ...
checking /etc/security/tsol/tnzonecfg ...
```

How to Compare Network Database Information With Kernel Cache

The databases might contain information that is not cached in the kernel. This procedure checks that the information is identical. The Solaris Management Console runs this command automatically when you use the Security Templates tool or the Trusted Network Zones tool.

Before You Begin You must be in the global zone in a role that can check network settings. The Security Administrator role and the System Administrator role can check these settings.

- **Run the `tninfo` command.**
  - `tninfo -h` *hostname* displays the IP address and template for the specified host.
  - `tninfo -t` *templatename* displays the following information:

```
template: template-name
host_type: one of CIPSO or UNLABELED
doi: 1
min_sl: minimum-label
hex: minimum-hex-label
max_sl: maximum-label
hex:maximum-hex-label
```

  - `tninfo -m` *zone-name* displays the multilevel port (MLP) configuration of a zone.

Example 13–14 Displaying Multilevel Ports on a Host

In this example, a system is configured with several labeled zones. All zones share the same IP address. Some zones are also configured with zone-specific addresses. In this configuration, the TCP port for web browsing, port `8080`, is an MLP on a shared interface in the public zone. The administrator has also set up `telnet`, TCP port `23`, to be an MLP in the public zone. Because these two MLPs are on a shared interface, no other zone, including the global zone, can receive packets on the shared interface on ports `8080` and `23`.

In addition, the TCP port for `ssh`, port `22`, is a per-zone MLP in the public zone. The public zone's `ssh` service can receive any packets on its zone-specific address within the address's label range.

The following are the MLPs for the public zone.

```
$ tninfo -m public
private: 22/tcp
shared:  23/tcp;8080/tcp
```

The following are the MLPs for the global zone. Note that ports `23` and `8080` cannot be MLPs in the global zone, because the global zone shares the same address with the public zone.

```
$ tninfo -m global
private: 111/tcp;111/udp;514/tcp;515/tcp;631/tcp;2049/tcp;
         6000-6003/tcp;38672/tcp;60770/tcp;
shared:  6000-6003/tcp
```

▼ How to Synchronize Kernel Cache With Network Databases

When the kernel has not been updated with trusted network database information, you have several ways to update the kernel cache. The Solaris Management Console runs this command automatically when you use the Security Templates tool or the Trusted Network Zones tool.

Before You Begin You must be in the Security Administrator role in the global zone.

◗ **Run one of the following commands:**

- **Restart the `tnctl` service.**

  ⚠ **Caution** – Do not use this method on systems that get their trusted network databases information from an LDAP server.

  ```
  $ svcadm restart  svc:/network/tnctl
  ```

  This command reads all information from the local trusted network databases into the kernel.

- **Update the kernel cache for a single host.**

  ```
  $ tnctl -h hostname
  ```

  This command reads only the information from the chosen option into the kernel. For details on the options, see the `tnctl`(1M) man page.

- **Change the `tnd` polling interval.**

  This does not update the kernel cache. However, you can shorten the polling interval to update the kernel cache more frequently. For details, see "How to Change the `tnd` Polling Interval" on page 154.

- **Refresh the `tnd`.**

  This triggers an immediate update of recent trusted network database entry changes in the kernel.

  ```
  $ svcadm refresh svc:/network/tnd
  ```

- **Restart the `tnd`.**

  This is rarely a good idea. This can interrupt communications that are currently succeeding. If you plan to restart the `tnd`, use the `svcadm` command.

  ```
  $ svcadm restart svc:/network/tnd
  ```

Example 13–15 Updating Network Information in the Kernel

In this example, the administrator updates the trusted network with a public print server, and then checks that the kernel settings are correct.

```
$ tnctl -h public-print-server
$ tninfo -h public-print-server
IP Address: 192.168.103.55
Template: PublicOnly
$ tninfo -t PublicOnly
=================================
Remote Host Template Table Entries
---------------------------------
template: PublicOnly
host_type: CIPSO
doi: 1
min_sl: PUBLIC
hex: 0x0002-08-08
max_sl: PUBLIC
hex: 0x0002-08-08
```

Troubleshooting the Trusted Network (Tasks)

Task	Description	For Instructions
Modify the network polling interval	Change the time to lapse between polls of the system's network traffic.	"How to Change the tnd Polling Interval" on page 154
Determine why two hosts cannot communicate	Check that the interfaces on a single system are up.	"How to Verify That a Host's Interfaces Are Up" on page 155
	Use debugging tools when two machines cannot reach each other.	"How to Debug the Trusted Extensions Network" on page 156
Determine why an LDAP client cannot reach the LDAP server	Troubleshoot loss of connection between an LDAP server and a client	"How to Debug a Client Connection to the LDAP Server" on page 158

▼ How to Change the tnd Polling Interval

By default, the tnd polls the local trusted network databases and the LDAP service every thirty minutes for changes. When you are setting up the network, or debugging, you might want to shorten the polling interval. Because tnd is a service, you use the Service Management Facility (SMF) to change the interval. For more information, see the smf(5) man page.

Before You Begin You must be in the Security Administrator role in the global zone.

1 **Change the polling interval for the trusted network daemon.**

```
$ svccfg -s tnd
  listprop tnd/poll_interval
     integer 1800
  setprop tnd/poll_interval = 0-to-1800-seconds
  quit
```

2 **Then, restart the daemon.**

```
$ svcadm restart tnd
```

Example 13–16 Updating Network Information in the Kernel

In this example, the security administrator is setting up a network. No users are on the network. Because many changes are going to be made to the network databases, the administrator ensures that the kernel cache is updated from network database changes every two minutes. This poll interval is retained across boots.

```
$ svccfg -s tnd
  listprop tnd/poll_interval
     integer 1800
  setprop tnd/poll_interval = 120
  listprop tnd/poll_interval
     integer 120
  quit
$ svcadm restart tnd
```

When the network has been set up, the administrator sets the poll interval to its original value.

```
$ svccfg -s tnd
  setprop tnd/poll_interval = 1800
  listprop tnd/poll_interval
     integer 1800
quit
$ svcadm restart tnd
```

▼ How to Verify That a Host's Interfaces Are Up

Use this procedure if your system does not communicate with others as expected.

Before You Begin You must be in the global zone in a role that can check network settings. The Security Administrator role and the System Administrator role can check these settings.

1 **Verify that the host's network interface is up.**

```
# ifconfig -a
...
hme0: flags=1000843<BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
```

```
         inet 192.168.0.11 netmask ffffff00 broadcast 192.168.0.255
hme0:3 flags=1000843<BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
         inet 192.168.0.12 netmask ffffff00 broadcast 192.168.0.255
```

This system has two network interfaces, hme0 and hme0:3. Neither interface is up.

2 **If the interface is not up, bring the interface up.**

Then, check that the interface is up.

```
# ifconfig hme0 up
# ifconfig -a
...
hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,...
hme0:3 flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,..
```

▼ How to Debug the Trusted Extensions Network

To debug two machines that should be communicating and are not, you can use Trusted Extensions and Solaris debugging tools. For example, Solaris network debugging commands such as snoop and netstat are available. For details, see the snoop(1M) and netstat(1M) man pages. For commands that are specific to Trusted Extensions, see Table 2–4.

For contacting labeled zones, see “Managing Zones (Tasks)” on page 104.

For debugging NFS mounts, see “How to Troubleshoot Mount Failures” on page 122.

For debugging LDAP communication, see “How to Debug a Client Connection to the LDAP Server” on page 158.

Before You Begin You must be in the global zone in a role that can check network settings. The Security Administrator role and the System Administrator role can check these settings.

1 **Check that the hosts that cannot communicate are using the same naming service.**

a. **Check the nsswitch.conf file on each machine.**

b. **Check the values for Trusted Extensions databases in the nsswitch.conf file.**

```
# Trusted Extensions
tnrhtp: files ldap
tnrhdb: files ldap
```

c. **If the values are different, fix the nsswitch.conf file to contain the correct values.**

To modify these entries, the System Administrator role uses the Name Service Switch action. For details, see “How to Launch CDE Administrative Actions in Trusted Extensions” on page 43. This action preserves the required DAC and MAC file permissions.

2 Check that the LDAP naming service is configured.

```
ldaplist -l
```

3 Check that both hosts are in the LDAP naming service.

```
ldaplist -l hosts | grep hostname
```

4 Check that each host is defined correctly.

- Check that each host is assigned to a security template that is compatible with the security template of the other host.
- On an unlabeled computer, check that the default label assignment is correct.
- Check that the assignment in each host machine's kernel cache matches the assignment on the network, and on the other host.
- Check that the multilevel ports (MLPs) are correctly configured.

To get security information for the source, destination, and gateway hosts in the transmission, use the `tninfo` command.

a. Display the IP address and assigned security template for a given host.

```
$ tninfo -h hostname
IP Address: IP-address
Template: template-name
```

b. Display a template definition by using `tninfo -t`.

```
$ tninfo -t template-name
template: template-name
host_type: one of CIPSO or UNLABELED
doi: 1
min_sl: minimum-label
hex: minimum-hex-label
max_sl: maximum-label
hex: maximum-hex-label
```

c. Display the MLPs for a zone by using `tninfo -m`.

```
$ tninfo -m zone-name
private: ports-that-are-specific-to-this-zone-only
shared: ports-that-the-zone-shares-with-other-zones
```

5 Fix any incorrect information.

- To change or check network security information, use the Solaris Management Console tools. For details, see "How to Open the Trusted Networking Tools" on page 139
- To update the kernel cache, restart the `tnctl` service on the host whose information is out of date. Allow some time for this process to complete. Then refresh the `tnd`. If refresh fails, try restarting the `tnd`. For details, see "How to Synchronize Kernel Cache With Network Databases" on page 153.

Rebooting clears the kernel cache. At boot time, the cache is populated with database information. The `nsswitch.conf` file determines whether local databases or LDAP databases are used to populate the kernel.

6 **Collect information to help you in debugging.**

- **Use the `get` subcommand to the `route` command.**

  ```
  $ route get [ip] -secattr sl=label,doi=integer
  ```

  For details, see the `route`(1M) man page.

- **Use the `snoop -v` command.**

  The `-v` option displays the details of packet headers, including label information. This command provides a lot of detail, so you might want to restrict the packets that the command examines. For details, see the `snoop`(1M) man page.

- **Use the `-R` option with the `netstat -a|-r` command.**

  The `-aR` option displays extended security attributes for sockets. The `-rR` option displays routing table entries. For details, see the `netstat`(1M) man page.

▼ How to Debug a Client Connection to the LDAP Server

Misconfiguration of the client entry on the LDAP server can prevent the client from communicating with the server. Similarly, misconfiguration of files on the client can also prevent communication. Check the following entries and files when attempting to debug a client-server communication problem.

Before You Begin You must be in the Security Administrator role in the global zone on the LDAP client.

1 **Check that the remote host template of the LDAP server is correct.**

Check also that the remote host template for the gateway to the LDAP server is correct.

```
# tninfo -h LDAP-server
# route get LDAP-server
# tninfo -h gateway-to-LDAP-server
```

If a remote host template assignment is incorrect, assign the host to the correct template in the Solaris Management Console.

2 **Check the `/etc/hosts` file.**

Your machine, the interfaces for the labeled zones on your machine, the gateway to the LDAP server, and the LDAP server should be listed in the file. You might have more entries.

Look for duplicate entries. Remove any entries that are labeled zones on other systems. For example, if `Lserver` is the name of your LDAP server, and `LServer-zones` is the shared interface for the labeled zones, remove `LServer-zones` from `/etc/hosts`.

Correct the file.

3 If you are using DNS, check that the entries in the `resolv.conf` file are accurate.

```
# more resolv.conf
search list of domains
domain domain-name
nameserver IP-address

...
nameserver IP-address
```

Correct the file.

4 Check that the `tnrhdb` and `tnrhtp` entries in the `nsswitch.conf` file are accurate.

5 Check that the client is correctly configured on the server.

```
# ldaplist -l tnrhdb client-IP-address
```

6 Check that the interfaces for your labeled zones are correctly configured on the LDAP server.

```
# ldaplist -l tnrhdb client-zone-IP-address
```

7 Verify that you can `ping` the LDAP server from all currently running zones.

```
# ldapclient list
...
NS_LDAP_SERVERS= LDAP-server-address
# zlogin zone-name1 ping LDAP-server-address
LDAP-server-address is alive
# zlogin zone-name2 ping LDAP-server-address
LDAP-server-address is alive
...
```

8 Then, configure LDAP and reboot.

a. Run the Create LDAP Client action.

This action re-clients the global zone to the LDAP server.

b. In every labeled zone, re-client the zone to the LDAP server.

```
# zlogin zone-name1
# ldapclient init \
-a profileName=profileName \
-a domainName=domain \
-a proxyDN=proxyDN \
-a proxyPassword=password LDAP-Server-IP-Address
# exit
# zlogin zone-name2 ...
```

c. **Then, halt all zones, lock the file systems, and reboot.**

If you are using ZFS, halt the zones and lock the file systems before rebooting. If you are not using ZFS, you can reboot without halting the zones and locking the file systems.

```
# zoneadm list
# zoneadm -z zonename halt
# lockfs -fa
# reboot
```

CHAPTER 14

Multilevel Mail in Trusted Extensions

This chapter covers security and multilevel mailers on systems that are configured with Trusted Extensions.

Multilevel Mail Service

Trusted Extensions provides multilevel mail for any mail application. When ordinary users launch their mailer, the application opens at the user's current label. If users are operating in a multilevel system, they might want to link or copy their mailer initialization files. For details, see "How to Configure Startup Files for Users" on page 74.

Trusted Extensions Mail Features

In Trusted Extensions, the System Administrator role sets up and administers mail servers according to instructions in the Solaris *System Administration Guide: Advanced Administration* and *System Administration Guide: IP Services*. In addition, the security administrator determines how Trusted Extensions mail features should be configured. The following aspects of managing mail are specific to Trusted Extensions.

- The `.mailrc` is at a user's minimum label.

  Therefore, users who work at multiple labels do not have a `.mailrc` at the higher labels unless they copy or link the `.mailrc` file to each higher directory.

  The Security Administrator role or the individual user can add the `.mailrc` file to either `.copy_files` or `.link_files`. For a description of these files, see the `updatehome`(1M) man page. For configuration suggestions, see "`.copy_files` and `.link_files` Files" on page 68.
- Your mail reader can run at every label on a system. Some configuration is required to connect a mail client to the server.

  For example, to use Mozilla mail for multilevel mail requires that you configure a Mozilla mail client at each label to specify the mail server. The mail server could be the same or different for each label, but the server must be specified.

- The Mailing Lists tool in the Solaris Management Console manages mail aliases.

  Depending on the scope of the selected Solaris Management Console toolbox, you can update the local `/etc/aliases` file or the entry on the Sun Java System Directory Server (the LDAP entry).
- Trusted Extensions software checks host and user labels before sending or forwarding mail.
  - The software checks that the mail is within the accreditation range of that host, as described in the following list and in Chapter 13.
  - The software checks that the mail is between the account's clearance and minimum label.
  - Users can read email that is received within their accreditation range. During a session, users can read mail only at their current label.

    To contact an ordinary user by email, an administrative role must send mail from a workspace that is at a label that the user can read. The user's default label is usually a good choice.

CHAPTER 15

Managing Labeled Printing

This chapter describes how to use Trusted Extensions software to configure labeled printing. It also describes how to configure print jobs without the labeling options.

Labels, Printers, and Printing

Trusted Extensions software uses labels to control printer access. Labels are used to control access to printers and to information about queued print jobs. The software also labels printed output. Body pages are labeled, and mandatory banner and trailer pages are labeled. Banner and trailer pages can also include handling instructions.

The System Administrator role handles basic printer administration. The Security Administrator role manages printer security, which includes labels. Printer security also includes how the labeled output is handled. The administrators follow basic Solaris printer administration procedures, then assign labels to the print servers and printers.

Trusted Extensions software supports both single-level and multilevel printing. Multilevel printing is implemented in the global zone only. To use the global zone's print server, a labeled zone must have a host name that is different from the global zone. One way to get a distinct host name, is to assign an IP address to the labeled zone. The address would be distinct from the global zone's IP address.

Restricting Access to Printers and Print Job Information

Users and roles on a system that is configured with Trusted Extensions software create print jobs at the label of their session. The print jobs can print only on printers that recognize that label. The label must be in the printer's label range.

Users and roles can view print jobs whose label is the same as the label of the session. In the global zone, a role can view jobs whose labels are dominated by the label of the zone.

Printers that are configured with Trusted Extensions software print labels on the printer output. Printers that are managed by unlabeled print servers do not print labels on the printer output. Such printers have the same label as their unlabeled server. For example, a Solaris print server can be assigned an arbitrary label in the `tnrhdb` of the LDAP naming service. Users can then print jobs at that arbitrary label on the Solaris printer. Like Trusted Extensions printers, those Solaris printers can only accept print jobs from users who are working at the label that has been assigned to the print server.

Labeled Printer Output

Trusted Extensions prints security information on body pages and banner and trailer pages. The information comes from the `label_encodings` file and from the `tsol_separator.ps` file.

The Security Administrator role can do the following to modify defaults that set labels and add handling instructions to printer output:

- Localize or customize the text on the banner and trailer pages.
- Specify alternate labels to be printed on body pages or in the various fields of the banner and trailer pages.
- Change or omit any of the text or labels.

The Security Administrator can also configure users to use printers that do not print labels on the output. Users can also be authorized to selectively not print banners or labels on printer output.

Labeled Body Pages

By default, the Protect As classification is printed at the top and bottom of every body page. The “Protect As” classification is the dominant classification when the classification from the job’s label is compared to the `minimum protect as classification`. The `minimum protect as classification` is defined in the `label_encodings` file.

For example, if the user is logged in to an Internal Use Only session, then the user’s print jobs are at that label. If the `minimum protect as classification` in the `label_encodings` file is Public, then the Internal Use Only label is printed on the body pages.

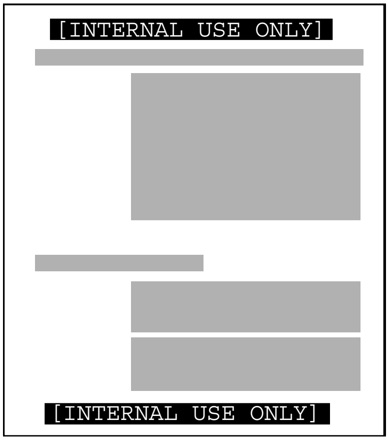
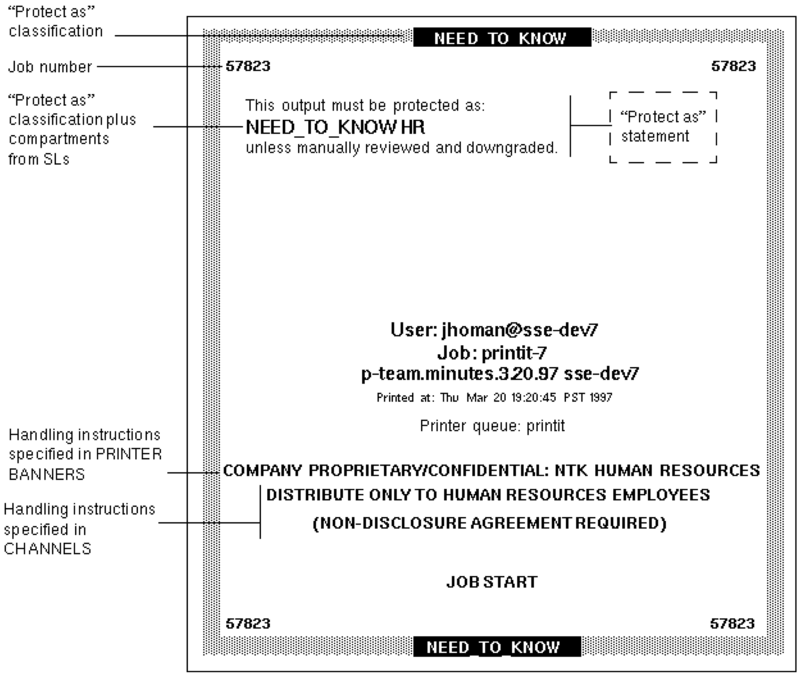


FIGURE 15–1 Job's Label Printed at the Top and Bottom of a Banner Page

Labeled Banner and Trailer Pages

The following figures show a default banner page and a default trailer page. Callouts identify the various sections. Note that the trailer page uses a different outer line.

The text, the labels, and the warnings that appear on print jobs are configurable. The text can also be replaced with text in another language for localization.



NEED TO KNOW

57823 57823

This output must be protected as:
NEED_TO_KNOW HR
unless manually reviewed and downgraded.

User: jhoman@sse-dev7
Job: printit-7
p-team.minutes.3.20.97 sse-dev7
Printed at: Thu Mar 20 19:20:45 PST 1997

Printer queue: printit

COMPANY PROPRIETARY/CONFIDENTIAL: NTK HUMAN RESOURCES
DISTRIBUTE ONLY TO HUMAN RESOURCES EMPLOYEES
(NON-DISCLOSURE AGREEMENT REQUIRED)

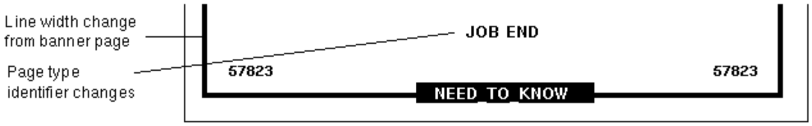
JOB START

57823 57823

NEED_TO_KNOW

"Protect as" classification
Job number
"Protect as" classification plus compartments from SLs
"Protect as" statement
Handling instructions specified in PRINTER BANNERS
Handling instructions specified in CHANNELS

FIGURE 15–2 Typical Banner Page of a Labeled Print Job



JOB END

57823 57823

NEED_TO_KNOW

Line width change from banner page
Page type identifier changes

FIGURE 15–3 Differences on a Trailer Page

The following table shows aspects of trusted printing that the Security Administrator can change by modifying the `/usr/lib/lp/postscript/tsol_separator.ps` file.

Note – To localize or internationalize the printed output, see the comments in the `tsol_separator.ps` file.

TABLE 15–1 Configurable Values in the `tsol_separator.ps` File

Output	Default Value	How Defined	To Change
`PRINTER BANNERS`	`/Caveats Job_Caveats`		See "Specifying Printer Banners" in *Solaris Trusted Extensions Label Administration*.
`CHANNELS`	`/Channels Job_Channels`		See "Specifying Channels" in *Solaris Trusted Extensions Label Administration*.
Label at the top of banner and trailer pages	`/HeadLabel Job_Protect def`	See `/PageLabel` description.	See `/PageLabel` change. Also see "Specifying the Protect As Classification" in *Solaris Trusted Extensions Label Administration*.
Label at the top and bottom of body pages	`/PageLabel Job_Protect def`	Compares the label of the job to the `minimum protect as classification` in the `label_encodings` file. Prints the more dominant classification. Contains compartments if the print job's label has compartments.	Change the `/PageLabel` definition to specify another value. Or, type a string of your choosing. Or, print nothing at all.
Text and label in the "Protect as" statement	`/Protect Job_Protect def`	See `/PageLabel` description.	See `/PageLabel` change.
	`/Protect_Text1 () def`	Text to appear above label.	Replace `()` in `Protect_Text1` and `Protect_Text2` with text string
	`/Protect_Text2 () def`	Text to appear below label.	

PostScript Printing of Security Information

Labeled printing in Trusted Extensions relies on features from Solaris Express printing. In the Solaris OS, printer model scripts handle banner page creation. To implement labeling, a printer model script first converts the print job to a PostScript™ file. Then, the PostScript file is manipulated to insert labels on body pages, and to create banner and trailer pages.

Solaris Express print model scripts can also translate PostScript into the native language of a printer. If a printer accepts PostScript, then Solaris software sends the job to the printer. If a printer does not accept PostScript, then the software converts the PostScript to a raster image. The raster image is then converted to the appropriate printer format.

Because PostScript is used to print label information, users cannot print PostScript files by default. This restriction prevents a knowledgeable PostScript programmer from creating a PostScript file that modifies the labels on the printer output.

The Security Administrator role can override this restriction by assigning the Print PostScript authorization to role accounts and to trustworthy users. The Security Administrator role should do so only if the account can be trusted not to spoof the labels on printer output. Also, the printing of PostScript files must be consistent with the site's security policy.

Printer Model Scripts

A model script enables a particular model of printer to provide banner and trailer pages. Trusted Extensions provides four model scripts:

- `tsol_standard` - For directly attached PostScript printers, for example, printers attached by a parallel port
- `tsol_netstandard` - For network–accessible PostScript printers
- `tsol_standard_foomatic` - For directly attached printers that do not print PostScript
- `tsol_netstandard_foomatic` - For network–accessible printers that do not print PostScript

The `foomatic` models are used when a printer driver name begins with `Foomatic`. Foomatic drivers are PostScript Printer Drivers (PPD). By default, "Use PPD" is specified in the Print Manager when you add a printer. A PPD is then used to translate banner and trailer pages into the language of the printer.

Additional Conversion Filters

A conversion filter converts text files to PostScript. The filter's programs are trusted programs that are run by the printer daemon. Files that are converted to PostScript by any installed filter programs can be trusted to have authentic labels and banner and trailer page text.

Solaris software provides most conversion filters that a site would need. A site's System Administrator role can install additional filters. These filters can then be trusted to have authentic labels and banner and trailer pages. To add conversion filters, see Chapter 5, "Managing Character Sets, Filters, Forms, and Fonts (Tasks)," in *System Administration Guide: Advanced Administration*.

Interoperability With Trusted Solaris 8 Printing

Trusted Solaris 8 and Trusted Extensions systems that have compatible `label_encodings` files and that identify each other as using a CIPSO template can use each other for remote printing.

Originating System	Print Server System	Action	Results
Trusted Extensions	Trusted Solaris 8	In the Trusted Extensions `tnrhdb`, assign a template with the appropriate label range to the Trusted Solaris 8 print server. The label could be CIPSO or unlabeled.	Trusted Solaris 8 printer can print jobs from a Trusted Extensions computer within the printer's label range.
Trusted Extensions	Trusted Solaris 8	On the Trusted Extensions computer, create a profile that adds the needed authorizations. Assign the profile to users.	Trusted Extensions users can list or cancel print jobs that they sent to a Trusted Solaris 8 printer. Users cannot view or remove jobs at a different label.
Trusted Solaris 8	Trusted Extensions	In the Trusted Solaris 8 `tnrhdb`, assign a template with the appropriate label range to the Trusted Extensions print server. The label could be CIPSO or unlabeled.	Trusted Extensions printer can print jobs from a Trusted Solaris 8 computer within the printer's label range.
Trusted Solaris 8	Trusted Extensions	On the Trusted Solaris 8 computer, create a profile that adds the needed authorizations. Assign the profile to users.	Trusted Solaris 8 users can list or cancel print jobs that they sent to a Trusted Extensions printer. Users cannot view or remove jobs at a different label.

Trusted Extensions Print Interfaces (Reference)

User commands are extended to conform with Trusted Extensions security policy.

- `cancel` – Caller must be equal to cancel a job. By default, ordinary users can cancel only their own jobs.
- `lp` – Trusted Extensions adds the `-o nolabels` option. Users must be authorized to print with no labels. Similarly, users must be authorized to use the `-o nobanner` option.
- `lpstat` – Caller must be equal to get the status of a job. By default, ordinary users can see only their own print jobs.

Several administrative commands are extended to conform with Trusted Extensions security policy. As in the Solaris OS, these commands can only be run by a role that includes the Printer Management profile.

- `lpmove` – Caller must be equal to move a job. By default, ordinary users can move only their own print jobs.
- `lpadmin` – In the global zone, this command works for all jobs. In a labeled zone, caller must dominate to view a job, and be equal to change a job.

  Trusted Extensions adds printer model scripts to the `-m` option. Trusted Extensions adds the `-o nolabels` option.
- `lpsched` – In the global zone, this command is always successful. As in the Solaris OS, use the `svcadm` command to enable, disable, start, or restart the print service. In a labeled zone, caller must be equal to change the print service. For details of the service management facility, see the `smf`(5), `svcadm`(1M), and `svcs`(1) man pages.

Trusted Extensions adds the `solaris.label.print` authorization to the Printer Management rights profile. The `solaris.print.unlabeled` authorization is required to print body pages without labels.

Managing Printing in Trusted Extensions (Task Map)

Trusted Extensions procedures for configuring printing are completed after doing Solaris printer setup. The following task map points to the major tasks that manage labeled printing.

Task	Description	For Instructions
Configure printers for labeled output	Enable users to print to a Trusted Extensions printer. The print jobs are marked with labels.	"Configuring Labeled Printing (Tasks)" on page 170
Remove visible labels from printer output	Enable users to print at a specific label to a Solaris printer. The print jobs are not marked with labels. Or, prevent labels from printing on a Trusted Extensions printer.	"Reducing Printing Restrictions in Trusted Extensions (Tasks)" on page 176

Configuring Labeled Printing (Tasks)

The following table describes common configuration procedures that are related to labeled printing.

Note – Printer clients can only print jobs within the label range of the Trusted Extensions print server.

Task	Description	For Instructions
Launch the Print Manager	Use this GUI to identify the printer to the network or to the local computer. The System Administrator launches the GUI in an administrative role workspace.	Chapter 3, "Setting Up Printers (Tasks)," in *System Administration Guide: Advanced Administration*
Configure printing from the global zone	Create a multilevel print server in the global zone.	"How to Configure a Multilevel Print Server and Its Printers" on page 171
Configure printing from a labeled zone	Create a single–label print server for a labeled zone.	"How to Configure a Zone for Single-Label Printing" on page 172
Configure a multilevel print client	Connect a Trusted Extensions host to a printer	"How to Enable a Trusted Extensions Client to Access a Printer" on page 173
Restrict the label range of a printer	Limit a Trusted Extensions printer to a narrow label range.	"How to Configure a Restricted Label Range for a Printer" on page 175

▼ How to Configure a Multilevel Print Server and Its Printers

Printers that are managed by a Trusted Extensions print server print labels on body pages, banner pages, and trailer pages. Such printers can print jobs within the label range of the print server. Any Trusted Extensions host that can reach the print server can use the printers that are connected to that server.

Before You Begin Determine the print server for your Trusted Extensions network. You must be in the System Administrator role in the global zone on this print server.

1. **Enable multilevel printing.**

   Create a multilevel port (MLP) for the print server by adding the port to the global zone.

   a. **Launch the Solaris Management Console.**

   For details, see "How to Launch the Solaris Management Console" on page 42.

   b. **Choose the Files toolbox.**

   The title of the toolbox includes `Scope=Files, Policy=TSOL`.

   c. **Configure the global zone with the print server port, 515/tcp.**

   i. **Navigate to the Trusted Network Zones tool.**

   ii. **In the Multilevel Ports for Zone's IP Addresses, add 515/tcp.**

   iii. **Click OK.**

2 **Define the characteristics of the connected printers.**

a. **Launch the Print Manager.**

b. **Determine the printer driver of a connected printer.**

In the Print Manager, you supply the answers to the first two fields, and the Print Manager supplies the driver name.

`Printer Make`	*manufacturer*
`Printer Model`	*manufacturer-part-number*
`Printer Driver`	*automatically filled in*

3 **Assign a printer model script.**

For every printer that is connected to the print server, assign the appropriate model script. The model script activates the banner and trailer pages for the specified printer.

For your choices, see "Printer Model Scripts" on page 168. If the driver name for the printer starts with `Foomatic`, then specify one of the foomatic model scripts.

```
$ lpadmin -p printer -m model
```

If the default printer label range of `ADMIN_LOW` to `ADMIN_HIGH` is acceptable for every printer, then your label configuration is done. Otherwise, go to "How to Configure a Restricted Label Range for a Printer" on page 175.

To prevent labeled printer output, see "Reducing Printing Restrictions in Trusted Extensions (Tasks)" on page 176.

4 **Finish other configuration.**

For details, see Chapter 3, "Setting Up Printers (Tasks)," in *System Administration Guide: Advanced Administration*.

5 **(Optional) Enable other computers to use this zone as a print server.**

For details, see "How to Enable a Trusted Extensions Client to Access a Printer" on page 173.

▼ How to Configure a Zone for Single-Label Printing

Before You Begin The zone must not be sharing an address with the global zone. You must be in the System Administrator role in the global zone.

1 **Add a workspace.**

2 **Change the label of the new workspace.**

Change the label to the label of the zone that will be the print server for that label.

3 Define the characteristics of the connected printers.

a. At the label of zone, launch the Print Manager.

By default, "Use PPD" is marked with a check. The system finds the appropriate driver for the printer.

b. (Optional) To specify a particular printer driver, do the following.

Remove the check from "Use PPD".

Then, supply the answers to the first two fields, and the Print Manager supplies the driver name.

```
Printer Make      manufacturer
Printer Model     manufacturer-part-number
Printer Driver    automatically filled in
```

4 Assign a printer model script.

For every printer that is connected to the zone, assign the appropriate model script. The model script activates the banner and trailer pages for the specified printer.

For your choices, see "Printer Model Scripts" on page 168. If the driver name for the printer starts with `Foomatic`, then specify one of the foomatic model scripts.

```
$ lpadmin -p printer -m model
```

To prevent labeled printer output, see "How to Prevent Labels on Printer Output" on page 177.

5 (Optional) Enable identically-labeled zones to use this zone as a print server.

For details, see "How to Enable a Trusted Extensions Client to Access a Printer" on page 173.

6 Finish other configuration.

Complete the setup of the printer. For details, see Chapter 3, "Setting Up Printers (Tasks)," in *System Administration Guide: Advanced Administration*.

▼ How to Enable a Trusted Extensions Client to Access a Printer

Initially, only the zone in which a print server was configured can print to the printers of that print server. Other zones and other computers must explicitly add access to those printers.

- A labeled zone must add access to the printers that are connected to its global zone.
- A labeled zone must add access to a printer that a remote zone at the same label is configured for.
- Global zones must add access to the printers that are connected to a global zone on a different computer.
- A labeled zone must add access to the printers that are connected to a global zone on a different computer.

Before You Begin A print server has been configured with a label range or a single label, and the printers that are connected to it have been configured. For details, see the following:

- "How to Configure a Multilevel Print Server and Its Printers" on page 171
- "How to Configure a Zone for Single-Label Printing" on page 172
- "How to Assign a Label to an Unlabeled Print Server" on page 177

You must be in the System Administrator role in the global zone, or be able to assume the role.

1 Do one or more of the following procedures.

You can also use the Print Manager instead of the `lpadmin` command. For details, see Step 2.

- **Configure a labeled zone to use its global zone for printer access.**

  a. **Assume the System Administrator role.**

  b. **Change the label of the role workspace to the label of the labeled zone.**

  c. **Add access to the printer.**

  ```
  lpadmin -s printer
  ```

- **Configure the global zone on a computer that is not a print server to use another computer's global zone for printer access.**

  a. **On the computer that does not have printer access, assume the System Administrator role.**

  b. **Add access to the printer that is connected to the Trusted Extensions print server.**

  ```
  lpadmin -s printer
  ```

- **Configure a labeled zone to use another computer's labeled zone for printer access.**

  The labels of the zones must be identical.

  a. **On the computer that does not have printer access, assume the System Administrator role.**

  b. **Change the label of the role workspace to the label of the labeled zone.**

  c. **Add access to the printer that is connected to the print server of the remote labeled zone.**

  ```
  lpadmin -s printer
  ```

- **Configure a labeled zone to use an arbitrarily labeled print server for printer access.**

  The label of the zone must be identical to the label of the print server.

  a. **On the computer that does not have printer access, assume the System Administrator role.**

  b. **Change the label of the role workspace to the label of the labeled zone.**

c. **Add access to the printer that is connected to the arbitrarily labeled print server.**

   `lpadmin -s` *printer*

2 **(Optional) Use the Print Manager to enable printer access.**

Rather than run the `lpadmin` command, use the Printers –> Add Access to Printer menu item. The Print Manager must be launched in the same zone at the same label as the `lpadmin` command.

▼ How to Configure a Restricted Label Range for a Printer

The default printer label range is `ADMIN_LOW` to `ADMIN_HIGH`. This procedure narrows the label range for a printer that is controlled by a Trusted Extensions print server.

Before You Begin You must be in the Security Administrator role in the global zone.

1 **Launch the Device Allocation Manager.**

- **Choose the Allocate Device option from the Trusted Path menu.**
- **In CDE, launch the Device Allocation Manager action from the Tools subpanel on the Front Panel.**

2 **Bring up the Device Allocation: Administration dialog box.**

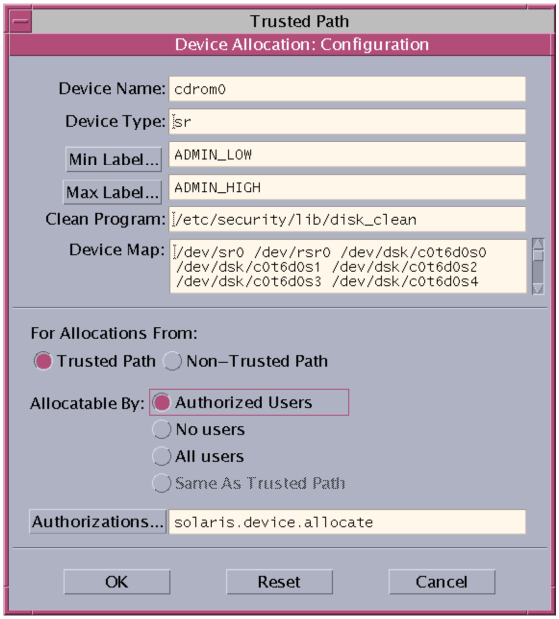
Click the Device Administration button.

3 **Type a name for the new printer.**

If the printer is attached to your system, find the name of the printer.

4 **Bring up the Device Allocation: Configuration dialog box.**

Click the Configure button.



Trusted Path

Device Allocation: Configuration

Device Name: cdrom0

Device Type: sr

Min Label... ADMIN_LOW

Max Label... ADMIN_HIGH

Clean Program: /etc/security/lib/disk_clean

Device Map:

```
/dev/sr0  /dev/rsr0  /dev/dsk/c0t6d0s0
/dev/dsk/c0t6d0s1  /dev/dsk/c0t6d0s2
/dev/dsk/c0t6d0s3  /dev/dsk/c0t6d0s4
```

For Allocations From:

● Trusted Path ○ Non–Trusted Path

Allocatable By:

● Authorized Users

○ No users

○ All users

○ Same As Trusted Path

Authorizations... solaris.device.allocate

5 Change the printer's label range.

a. Choose the minimum label by clicking the Min Label button.

b. Choose the maximum label by clicking the Max Label button.

6 Save the changes.

a. Click the OK button on the Configuration dialog box.

b. Click the OK button on the Administration dialog box.

Then close the Device Allocation Manager.

Reducing Printing Restrictions in Trusted Extensions (Tasks)

The following tasks are optional. These procedures reduce the printing security that Trusted Extensions software provides by default.

Task	Description	For Instructions
Configure a printer to not label output	Prevent security information from printing on body pages, and remove banner and trailer pages.	"How to Prevent Labels on Printer Output" on page 177
Configure printers at a single label without labeled output	Enable users to print at a specific label to a Solaris printer. The print jobs are not marked with labels.	"How to Assign a Label to an Unlabeled Print Server" on page 177
Remove visible labeling of body pages	Modify the `tsol_separator.ps` file to prevent labeled body pages on all print jobs that are sent from a Trusted Extensions host.	"How to Remove Page Labels From All Print Jobs" on page 178
Suppress banner and trailer pages	Authorize individual users to print jobs without banner and trailer pages.	"How to Suppress Banner and Trailer Pages for Specific Users" on page 179
Enable trusted users to print jobs without labels	Authorize individual users or all users of a particular system to print jobs without labels.	"How to Enable Specific Users to Suppress Page Labels" on page 179
Enable the printing of PostScript files	Authorize individual users or all users of a particular system to print PostScript files.	"How to Enable Users to Print PostScript Files" on page 180
Assign printing authorizations	Enable users to bypass default printing restrictions.	"How to Create a Convenient Authorizations Rights Profile" on page 79 "How to Modify `policy.conf` Defaults" on page 72

▼ How to Prevent Labels on Printer Output

Printers that do not have a printer model script do not print banner or trailer pages. The body pages also do not include labels.

Before You Begin You must be in the Security Administrator role in the global zone.

At the appropriate label, do one of the following:

- **From the print server at the appropriate label, stop banner printing altogether.**

```
% lpadmin -p printer -o nobanner=never
```

- **Set the printer model script to a Solaris script.**

```
% lpadmin -p printer -m  \
-m { standard | netstandard | standard_foomatic | netstandard_foomatic }
```

▼ How to Assign a Label to an Unlabeled Print Server

Printers that are connected to an unlabeled server can print jobs only at the arbitrary label that has been assigned to the print server. Jobs print without labels or trailer pages and might print without banner pages. If a job prints with a banner page, the page does not contain any security information.

A Trusted Extensions computer can be configured to send jobs to a printer that is managed by an unlabeled print server. Users can print jobs on the unlabeled printer at the label that the Security Administrator assigns to the print server. A Solaris print server is an unlabeled print server that can be assigned a label for Trusted Extensions access to the printer at that label.

Before You Begin You must be in the Security Administrator role in the global zone.

1. **Open the Solaris Management Console in the desired scope.**

   For details, see "How to Launch the Solaris Management Console" on page 42.

2. **Navigate to Computers and Networks.**

   Provide a password when prompted.

3. **Assign an unlabeled template to the print server.**

   Choose a label. Users who are working at that label can send print jobs to the Solaris printer at the label of the print server. Pages do not print with labels, and banner and trailer pages are also not part of the print job.

Example 15–1 Sending Public Print Jobs to an Unlabeled Printer

Files that are available to the general public are candidates for printing to an unlabeled printer. In this example, marketing writers need to produce documents that do not have labels printed on the top and bottom of the pages.

The Security Administrator role assigns an unlabeled host type template to the Solaris print server. The arbitrary label of the template is `PUBLIC`. The printer `pr-nolabel1` is connected to this print server. Print jobs from users in a `PUBLIC` zone print on the `pr-nolabel1` printer with no labels. Depending on the settings for the printer, the jobs might or might not have banner pages. The banner pages do not contain security information.

▼ How to Remove Page Labels From All Print Jobs

This procedure prevents all print jobs on a Solaris Trusted Extensions 1.0 printer from having visible labels on the body pages of the print job.

Before You Begin You must be in the Security Administrator role in the global zone.

1. **Edit the `/usr/lib/lp/postscript/tsol_separator.ps` file.**

   Use the Admin Editor action. For details, see "How to Edit Administrative Files in Trusted Extensions" on page 44.

2 **Find the definition of** `/PageLabel`**.**

Find the following lines:

```
%% To eliminate page labels completely, change this line to
%% set the page label to an empty string: /PageLabel () def
/PageLabel Job_PageLabel def
```

Note – The value `Job_PageLabel` might be different at your site.

---

3 **Replace the value of** `/PageLabel` **with a set of empty parentheses.**

```
/PageLabel () def
```

▼ How to Enable Specific Users to Suppress Page Labels

This procedure enables an authorized user or role to print jobs on a Trusted Extensions printer without labels on the top and bottom of each body page. Page labels are suppressed for all labels at which the user can work.

Before You Begin You must be in the Security Administrator role in the global zone.

1 **Determine who is permitted to print jobs without page labels.**

2 **Authorize those users and roles to print jobs without page labels.**

Assign a profile that includes the Print without Label authorization to those users and roles. For details, see "How to Create a Convenient Authorizations Rights Profile" on page 79.

3 **Instruct the user or role to specify no labels when submitting print jobs.**

Use the `lp -o nolabels` command.

```
% lp -o nolabels staff.mtg.notes
```

▼ How to Suppress Banner and Trailer Pages for Specific Users

Before You Begin The Always Print Banner checkbox on the Print Manager dialog box is not marked with a check.

□ Always Print Banner

You must be in the Security Administrator role in the global zone.

1 **Create a profile that includes the Print without Banner authorization.**

Assign the profile to each user or role that is allowed to print without banner and trailer pages.

For details, see "How to Create a Convenient Authorizations Rights Profile" on page 79.

2 **Instruct the user or role to submit jobs by using the `lp` command with the option `-o nobanner`.**

```
% lp -o nobanner staff.mtg.notes
```

▼ How to Enable Users to Print PostScript Files

Before You Begin You must be in the Security Administrator role in the global zone.

◗ **Use one of the following three methods to enable users to print PostScript files.**

- **To enable anyone to print PostScript files from a system, modify the `/etc/default/print` file.**

  a. **Create or modify the `/etc/default/print` file.**

  Use the Admin Editor. For details, see "How to Edit Administrative Files in Trusted Extensions" on page 44.

  b. **Type the following entry.**

  ```
  PRINT_POSTSCRIPT=1
  ```

  c. **Save and close the file.**

- **To enable anyone to print PostScript files from a system, modify the `/etc/security/policy.conf` file.**

  a. **Modify the `policy.conf` file.**

  Use the Admin Editor. For details, see "How to Edit Administrative Files in Trusted Extensions" on page 44.

  b. **Add the `solaris.print.ps` authorization.**

  ```
  AUTHS_GRANTED=other-authorizations,solaris.print.ps
  ```

  c. **Save and close the file.**

- **To enable a user or role to print PostScript files, give just those users and roles the authorization.**

  Assign a profile that includes the Print Postscript authorization to those users and roles. For details, see "How to Create a Convenient Authorizations Rights Profile" on page 79.

  These authorized users and roles can print PostScript files from any system.

Example 15–2 Enabling PostScript Printing From a Public System

In the following example, the Security Administrator has constrained a public kiosk to operate at the PUBLIC label. The system also has a few icons that open topics of interest. These topics can be printed.

The Security Administrator created an /etc/default/print file on the system. The file has one entry to enable printing PostScript files. No user needs a Print Postscript authorization.

```
# /etc/default/print

# PRINT_POSTSCRIPT=0
PRINT_POSTSCRIPT=1
```

CHAPTER 16

Devices in Trusted Extensions

This chapter describes device protection in Trusted Extensions.

Device Protection With Trusted Extensions Software

On a Solaris system, devices can be protected by allocation and by authorization. By default, devices are available to ordinary users without an authorization. A system that is configured with Trusted Extensions software uses the device protection mechanisms of the Solaris OS.

However, Trusted Extensions requires that a device be allocated for use, and by default, requires that the user be authorized to use the device. In addition, devices are protected by labels. Trusted Extensions provides a graphical user interface for administrators to manage devices. The same interface is used by users to allocate devices.

For information on device protection in the Solaris OS, see Chapter 4, "Controlling Access to Devices (Tasks)," in *System Administration Guide: Security Services*. This chapter covers the extensions to device protection that Trusted Extensions provides.

On a system that is configured with Trusted Extensions, two roles protect devices.

- The system administrator controls access to peripheral devices.

  The System Administrator role makes a device allocatable. Devices that the system administrator makes nonallocatable cannot be used by anyone. Allocatable devices can be allocated only by authorized users.
- The Security Administrator role restricts the labels at which a device can be accessed and sets device policy. The security administrator decides who is authorized to allocate a device.

The following are the main features of device control with Trusted Extensions software:

- An unauthorized user on a default Trusted Extensions system cannot allocate devices such as tape drives, CD-ROM drives, or diskette drives.

  An ordinary user with the Allocate Device authorization can import or export information at the label at which the user allocates the device.
- Users invoke the Device Allocation Manager to allocate devices when logged in directly. To allocate a device remotely, you must be able to the global zone. Typically, only roles can log in to the global zone.
- The label range of each device can be restricted by the security administrator. Ordinary users are limited to accessing devices whose label range includes the labels at which the users are allowed to work. The default label range is `ADMIN_LOW` to `ADMIN_HIGH`.
- Label ranges can be restricted for both allocatable and nonallocatable devices. Nonallocatable devices are devices such as framebuffers and printers.

Device Label Ranges

To prevent users from copying sensitive information, each allocatable device has a label range. To use an allocatable device, the user must be currently operating at a label within the device's label range. If the user is not, allocation is denied. The user's current label is applied to data imported or exported while the device is allocated to the user. The label of exported data is displayed when the device is deallocated. The user should physically label the medium that contains the exported data.

Effects of Label Range on a Device

To restrict direct login access through the console, the Security Administrator role can set a restricted label range on the framebuffer.

For example, a restricted label range might be specified to limit access to a publicly accessible computer. The label range enables users to access the computer only at a label within the framebuffer's label range.

When a host has a local printer, a restricted label range on the printer limits the jobs that can be printed on the printer.

Device Access Policies

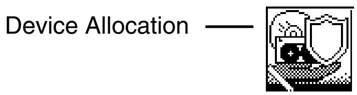
Trusted Extensions follows the same device policy as the Solaris OS. The Security Administrator role can change default policies and define new policies. The `getdevpolicy` command retrieves information about device policy, and the `update_drv` command changes device policy. For more information, see "Configuring Device Policy (Task Map)" in *System Administration Guide: Security Services*. See also the `getdevpolicy`(1M) and `update_drv`(1M) man pages.

Device-Clean Scripts

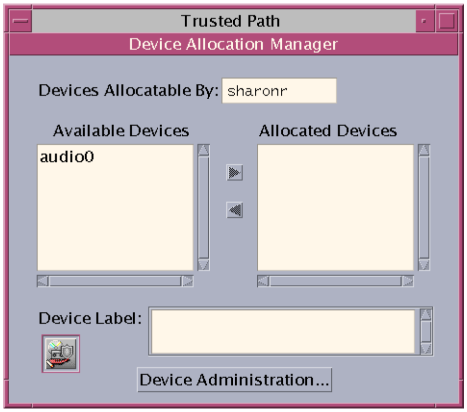
A device-clean script is run when a device is allocated or deallocated. The Solaris OS provides scripts for tape, CD-ROM, and diskette drives. If your site adds allocatable device types to the system, the added devices might need scripts. To see existing scripts, go to the `/etc/security/lib` directory. For more information, see “Device-Clean Scripts” in *System Administration Guide: Security Services*.

For Trusted Extensions software, device-clean scripts must satisfy certain requirements. The requirements are described on the `device_clean`(5) man page.

Device Allocation Manager GUI



The Device Allocation Manager is used by administrators to administer allocatable and nonallocatable devices. The Device Allocation Manager is also used by ordinary users to allocate and deallocate devices. The users must have the Allocate Device authorization. In a Solaris Trusted Extensions (CDE) workspace, the Device Allocation Manager is launched from the Front Panel. The following figure shows a Device Allocation Manager that was opened by a user who can allocate the `audio` device.



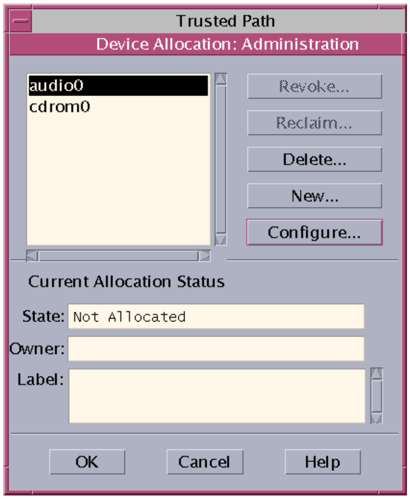
Devices Allocatable By: sharonr

Available Devices: audio0

FIGURE 16–1 Device Allocation Manager Opened by a User

Users see an empty list when the users are not authorized to allocate devices. Or, an empty list might mean that the allocatable devices are currently allocated by another user or are in an error state. If a user cannot see a device in the Available Devices list, the user needs to contact the responsible administrator.

The Device Administration feature is available to roles that have either one or both of the authorizations that are needed to administer devices. The administration authorizations are Configure Device Attributes, and Revoke or Reclaim Device. The following figure shows a Device Allocation Administration dialog box.



audio0
cdrom0

State: Not Allocated
Owner:
Label:

Enforcement of Device Security

The security administrator decides who can allocate devices. The security administrator should make sure that any user who is authorized to use devices is trained. The user is trusted to do the following:

- Properly label and handle any media containing exported sensitive information so that the information does not become available to anyone who should not see it.

  For example, if information at a label of `NEED TO KNOW ENGINEERING` is stored on a diskette, the person who exports the information must physically label the disk with the `NEED TO KNOW ENGINEERING` label. The diskette must be stored where it is accessible only to members of the engineering group with a need to know.

- Ensure that labels are properly maintained on any information being imported (read) from media on these devices.

  An authorized user should allocate the device at the label that matches the label of the information being imported. For example, if a user allocates a diskette drive at `PUBLIC`, the user should only import information labeled `PUBLIC`.

The Security Administrator role also is responsible for enforcing proper compliance with security requirements.

Devices in Trusted Extensions (Reference)

Trusted Extensions device protection uses Solaris interfaces and Trusted Extensions interfaces.

For Solaris command line interfaces, see “Device Protection (Reference)” in *System Administration Guide: Security Services*.

Note – In Trusted Extensions, users cannot use the `allocate` and `deallocate` commands. Users must use the Device Allocation Manager.

Administrators who do not have access to the Device Allocation Manager can administer allocatable devices by using the command line. The `allocate` and `deallocate` commands have administrative options. For examples, see “Forcibly Allocating a Device” in *System Administration Guide: Security Services* and “Forcibly Deallocating a Device” in *System Administration Guide: Security Services*.

For Trusted Extensions command line interfaces, see the `add_allocatable`(1M) and `remove_allocatable`(1M) man pages.

CHAPTER 17

Managing Devices for Trusted Extensions

This chapter describes how to administer and use devices on a system that is configured with Trusted Extensions.

Handling Devices in Trusted Extensions (Task Map)

The following task map points to the administrative and user tasks when handling peripheral devices.

Task	Description	For Instructions
Administer devices	Configure devices for ordinary users.	"Managing Devices in Trusted Extensions (Tasks)" on page 190
Customize device authorizations	The Security Administrator role creates new authorizations, adds them to the device, places them in a profile and assigns this profile to the user.	"Customizing Device Authorizations in Trusted Extensions (Tasks)" on page 199
Use devices	Use a device as a role or as an ordinary user.	"Using Devices in Trusted Extensions (Tasks)" on page 204

Managing Devices in Trusted Extensions (Tasks)

Use the following task map to protect devices at your site.

Task	Description	For Instructions
Set or modify device policy	Change the privileges that are required to access a device.	"Configuring Device Policy (Task Map)" in *System Administration Guide: Security Services*
Authorize users to allocate a device	The Security Administrator role assigns a profile with the Allocate Device authorization to the user.	"How to Authorize Users to Allocate a Device" in *System Administration Guide: Security Services*
	The Security Administrator role assigns a profile with the site-specific authorizations to the user.	"Customizing Device Authorizations in Trusted Extensions (Tasks)" on page 199
Configure a device	Choose security features to protect the device.	"How to Configure a Device" on page 191
Revoke or reclaim a device	Use the Device Allocation Manager to make a device available for use.	"How to Revoke or Reclaim a Device" on page 194
	Use Solaris commands to make a device available for use.	"Forcibly Allocating a Device" in *System Administration Guide: Security Services* "Forcibly Deallocating a Device" in *System Administration Guide: Security Services*
Prevent access to an allocatable device	Provide fine–grained access control to a device.	Example 17–4
	Deny everyone access to an allocatable device.	Example 17–1
Protect printers and framebuffers	Ensure that nonallocatable devices are not allocatable.	"How to Protect Nonallocatable Devices" on page 195
Configure serial login device	Enable logins by serial port.	"How to Configure a Serial Line for Logins" on page 196
Enable a CD player program to be used	Enable an audioplayer program to launch automatically when a music CD is inserted.	"How to Configure an Audio Player Program for Use" on page 197
Prevent the File Manager from displaying	Prevent the File Manager from popping up after a device has been allocated.	"How to Prevent File Manager Display After Device Allocation" on page 198
Use a new device-clean script	Place a new script in the appropriate places.	"How to Add a Device Clean Script" on page 198

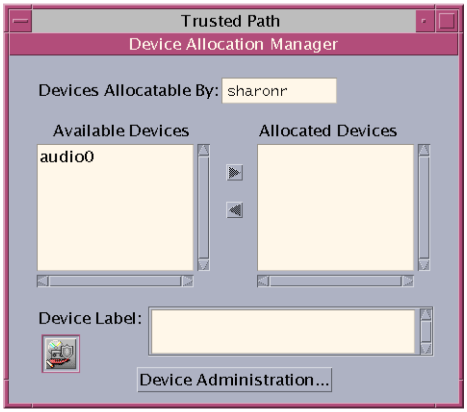
▼ How to Configure a Device

By default, an allocatable device has a label range from `ADMIN_LOW` to `ADMIN_HIGH`, and must be allocated for use. Also, users must be authorized to allocate the device. These defaults can be changed.

Before You Begin You must be in the Security Administrator role in the global zone.

1 **Open the Device Allocation Manager.**

Click the Device Allocation icon on the Tools subpanel.



Trusted Path
Device Allocation Manager

Devices Allocatable By: sharonr

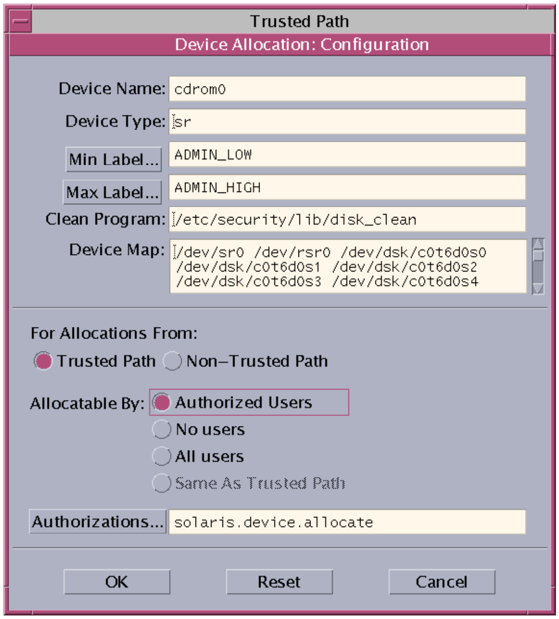
Available Devices: audio0

Allocated Devices:

Device Label:

2 **View the default security settings.**

Click Device Administration, then highlight the device. The following figure shows a CD-ROM device with default security settings.



Device Name: cdrom0

Device Type: sr

Min Label... ADMIN_LOW

Max Label... ADMIN_HIGH

Clean Program: /etc/security/lib/disk_clean

Device Map:

```
/dev/sr0 /dev/rsr0 /dev/dsk/c0t6d0s0
/dev/dsk/c0t6d0s1 /dev/dsk/c0t6d0s2
/dev/dsk/c0t6d0s3 /dev/dsk/c0t6d0s4
```

For Allocations From:

● Trusted Path ○ Non–Trusted Path

Allocatable By:
● Authorized Users
○ No users
○ All users
○ Same As Trusted Path

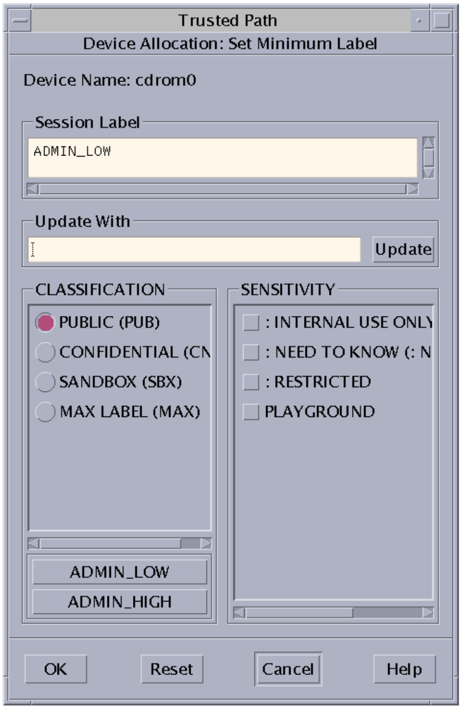
Authorizations... solaris.device.allocate

3 (Optional) Restrict the label range on the device.

The label range for all devices is `ADMIN_LOW` to `ADMIN_HIGH`.

a. Set the minimum label.

Click the Min Label... button. Choose the minimum label from the label builder.



Trusted Path

Device Allocation: Set Minimum Label

Device Name: cdrom0

Session Label: ADMIN_LOW

Update With:

CLASSIFICATION
- ● PUBLIC (PUB)
- ○ CONFIDENTIAL (CN
- ○ SANDBOX (SBX)
- ○ MAX LABEL (MAX)

SENSITIVITY
- □ : INTERNAL USE ONLY
- □ : NEED TO KNOW (: N
- □ : RESTRICTED
- □ PLAYGROUND

b. **Set the maximum label.**

Click the Max Label... button. Choose the minimum label from the label builder.

4 **Specify if the device can be allocated locally.**

In the Allocations From Trusted Path, choose an option from the Allocatable By list. By default, the Authorized Users option is checked. Therefore, the device is allocatable and users must be authorized.

- **To make the device nonallocatable, click No Users.**

  When configuring a printer, frame buffer, or other device that should not be allocatable, make sure to select No Users.

- **To make the device allocatable, but not require authorization, click All Users.**

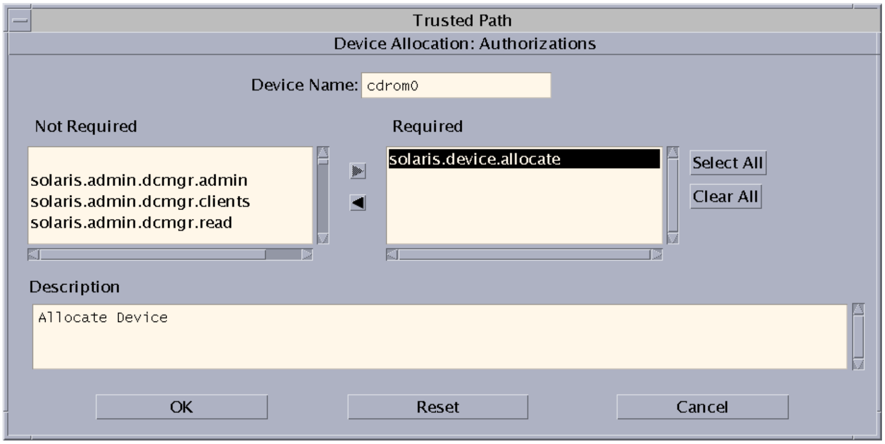
5 **Specify if the device can be allocated remotely.**

In the Allocations From Non-Trusted Path, choose an option from the Allocatable By list. By default, the Same As Trusted Path option is checked.

- **To require user authorization, choose Allocatable by Authorized Users.**

- **To make the device nonallocatable by remote users, click No Users.**
- **To make the device allocatable by anyone, click All Users.**

6 **If the device is allocatable, *and* your site has created new device authorizations, select the appropriate authorization.**



Device Name: cdrom0

Not Required
solaris.admin.dcmgr.admin
solaris.admin.dcmgr.clients
solaris.admin.dcmgr.read

Required
solaris.device.allocate

Description
Allocate Device

To create and use site-specific device authorizations, see "Customizing Device Authorizations in Trusted Extensions (Tasks)" on page 199.

7 **Save your changes.**

How to Revoke or Reclaim a Device

If a device is not listed, it might be already allocated or it might be in an allocate error state. The system administrator can recover the device for use.

Before You Begin You must be in the System Administrator role in the global zone. This role has the `solaris.device.revoke` authorization.

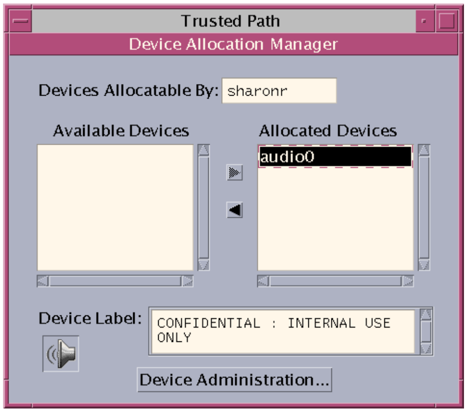
1 **Open the Device Allocation Manager.**

Click the Device Allocation icon on the Tools subpanel.

2 **Click the Device Administration button.**

3 **Check the status of a device.**

Highlight the device name and look at the State: field. In the following figure, the `audio` device is already allocated to a user.



Devices Allocatable By: sharonr

Allocated Devices: audio0

Device Label: CONFIDENTIAL : INTERNAL USE ONLY

- **If the State field is Allocate Error State, click the Reclaim button.**
- **If the State Field is Allocated, do one of the following:**
  - Ask the user in the Owner field to deallocate the device.
  - Force deallocation of the device by clicking the Revoke button .

4 **Close the Device Allocation Manager.**

▼ How to Protect Nonallocatable Devices

The No Users option is used most often for the framebuffer and printer, which do not have to be allocated to be used.

Before You Begin You must be in the Security Administrator role in the global zone.

1 **Open the Device Allocation Manager.**

Click the Device Allocation icon on the Tools subpanel.

2 **Click the Device Administration button.**

3 **Select the new printer or framebuffer.**

a. **For Allocatable By, click No Users.**

b. **(Optional) Restrict the label range by setting the minimum and maximum label.**

Example 17–1 Preventing Remote Allocation of the Audio Device

The No Users option is used most often for the framebuffer and printer, which do not have to be allocated to be used. But as shown in the following example, No Users prevents remote users from hearing conversations around a remote machine.

The Security Administrator role configures the audio device in the Device Allocation Manager as follows:

```
Device Name: audio
For Allocations From: Trusted Path
Allocatable By: Authorized Users
Authorizations: solaris.device.allocate

Device Name: audio
For Allocations From: Non-Trusted Pathh
Allocatable By: No Users
```

▼ How to Configure a Serial Line for Logins

Before You Begin You must be in the Security Administrator role in the global zone.

1 Launch the Solaris Management Console with the Files scope.

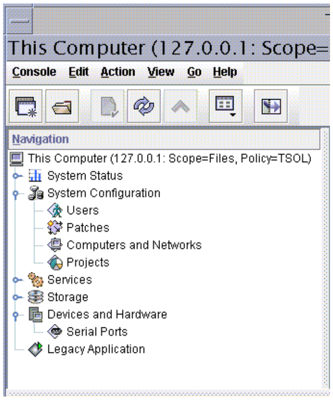


FIGURE 17–1 Solaris Management Console Tools

2 Navigate to Serial Ports.

Click Devices and Hardware, then Serial Ports. Provide a password when prompted. Follow the online help to configure the serial port.

3 Click the Device Allocation icon on the Tools subpanel on the Front Panel.

The default label range is `ADMIN_LOW` to `ADMIN_HIGH`.

4 **Save your changes.**

Example 17–2 Restricting the Label Range of a Serial Port

After creating a serial login device, the Security Administrator role restricts the label range of the serial port to a single label, `Public`. The administrator sets the following values in the Device Administration dialog boxes.

```
Device Name: /dev/term[a|b]
Device Type: tty
Clean Program: /bin/true
Device Map: /dev/term[a|b]
Minimum Label: Public
Maximum Label: Public
Allocatable By: No Users
```

▼ How to Configure an Audio Player Program for Use

The following procedure enables an audio player to launch automatically when a user inserts a music CD. For the user's procedure, see "How to Listen to a Music CD" on page 204.

Before You Begin You must be in the System Administrator role in the global zone.

1 **Edit the** `/etc/rmmount.conf` **file.**

Use the Admin Editor action. For details, see "How to Edit Administrative Files in Trusted Extensions" on page 44.

2 **Add your site's CD player program to the** `cdrom` **action in the file.**

`action` *medium* `action_`*program*`.so` *path-to-program*

Example 17–3 Configuring an Audio Player Program for Use

In the following example, the system administrator makes the `workman` program available to all users of this system. The `workman` program is an audio player program.

```
# /etc/rmmount.conf file
action cdrom action_workman.so /usr/local/bin/workman
```

▼ How to Prevent File Manager Display After Device Allocation

Before You Begin You must be in the System Administrator role in the global zone.

1. **Edit the `/etc/rmmount.conf` file.**

   Use the Admin Editor action. For details, see "How to Edit Administrative Files in Trusted Extensions" on page 44.

2. **Find the `filemgr` actions.**

```
action cdrom action_filemgr.so
action floppy action_filemgr.so
```

3. **Comment out the appropriate action.**

   The following example shows the `action_filemgr.so` commented out for both the `cdrom` and `diskette` devices.

```
# action cdrom action_filemgr.so
# action floppy action_filemgr.so
```

▼ How to Add a Device Clean Script

If no `device_clean` script is specified at the time the device is created, the default is `/bin/true`.

Before You Begin Have ready a script that purges all usable data from the physical device and that returns `0` for success. For devices with removable media, the script attempts to eject the media if the user does not do so. The script puts the device into the allocate error state if the medium is not ejected. For details of the requirements, see `device_clean`(5).

You must be in the System Administrator role in the global zone.

1. **Copy the script into `/etc/security/lib`.**

2. **In the Device Administration GUI, type the full path to the script.**

   a. **Open the Device Allocation Manager.**

   b. **Click the Device Administration button.**

   c. **Highlight the name of the affected device and click the Configure… button.**

   d. **Type the full path to the script in the Clean Program field.**

   e. **Close the Device Allocation Manager.**

Customizing Device Authorizations in Trusted Extensions (Tasks)

Use the following task map to change device authorizations at your site.

Task	Description	For Instructions
Create new device authorizations	Create site-specific authorizations.	"How to Create New Device Authorizations" on page 199
Add authorizations to a device	Add site-specific authorizations to selected devices.	"How to Add Site-Specific Authorizations to a Device" on page 202
Assign device authorizations to users and roles	Enable users and roles to use the new authorizations.	"How to Assign Device Authorizations" on page 202

▼ How to Create New Device Authorizations

If no authorization is specified at the time a device is created, by default all users can use the device. If an authorization is specified, then by default only authorized users can use the device.

Before You Begin You must be in the Security Administrator role in the global zone.

1 Open the `auth_attr` file for editing.

Use the Admin Editor action. For details, see "How to Edit Administrative Files in Trusted Extensions" on page 44.

2 Create a heading for the new authorizations.

Use the reverse-order Internet domain name of your organization followed by optional additional arbitrary components. Separate components by dots. End heading names with a dot.

```
domain-suffix.domain-prefix.:::Company Header::help=Company.html
```

3 Add new authorization entries.

Enter the authorizations, one authorization per line. The lines are split for display purposes.

```
domain-suffix.domain-prefix.grant:::Grant All Company Authorizations::
help=CompanyGrant.html
domain-suffix.domain-prefix.grant.device:::Grant Company Device Authorizations::
help=CompanyGrantDevice.html
domain-suffix.domain-prefix.device.allocate.tape:::Allocate Tape Device::
help=CompanyTapeAllocate.html
domain-suffix.domain-prefix.device.allocate.floppy:::Allocate Floppy Device::
help=CompanyFloppyAllocate.html
```

4 **Save and close the file.**

```
:wq
```

5 **If you are using LDAP as your naming service, update the `auth_attr` entries on the Sun Java System Directory Server (LDAP server).**

For information, see the `ldapaddent`(1M).

6 **Add the new authorizations to the appropriate profiles. Then assign the profiles to users and roles.**

Use the Solaris Management Console. Assume the Security Administrator role, then follow the Solaris procedure "How to Create or Change a Rights Profile" in *System Administration Guide: Security Services*

7 **Use the authorization to restrict access to tapes and diskette drives.**

Add the new authorizations to the list of required authorizations in the Device Allocation Manager. For the procedure, see "How to Add Site-Specific Authorizations to a Device" on page 202.

Example 17–4 Creating More Fine-Grained Device Authorizations

A security administrator for NewCo constructed more fine-grained device authorizations for the company.

First, the administrator included a header for all of the authorizations for `newco.com` in the `auth_attr` file.

```
# auth_attr file
com.newco.:::NewCo Header::help=Newco.html
```

Next, the administrator added authorization entries to the file:

```
com.newco.grant:::Grant All NewCo Authorizations::
help=NewcoGrant.html
com.newco.grant.device:::Grant NewCo Device Authorizations::
help=NewcoGrantDevice.html
com.newco.device.allocate.tape:::Allocate Tape Device::
help=NewcoTapeAllocate.html
com.newco.device.allocate.floppy:::Allocate Floppy Device::
help=NewcoFloppyAllocate.html
```

The lines are split for display purposes.

The entries created the following authorizations:

- An authorization to grant all NewCo's authorizations
- An authorization to grant NewCo's device authorizations
- An authorization to allocate a tape device
- An authorization to allocate a diskette device

Example 17–5 Creating Trusted Path and Non-Trusted Path Authorizations

By default, the Allocate Devices authorizations enables allocation from the trusted path and from outside the trusted path.

In the following example, site security policy required restricting remote CD-ROM allocation. The security administrator created the `com.someco.device.cdrom.local` authorization. This authorization is for CD-ROM devices that are allocated with the trusted path. The `com.someco.device.cdrom.remote` authorization is for those few users who are allowed to allocate a CD-ROM outside the trusted path.

The security administrator added the authorizations to the `auth_attr` database, added the authorizations to the devices, and then placed the authorizations in profiles. The profiles were assigned to users who were allowed to allocate devices.

- `auth_attr` database entries:

  ```
  com.someco.:::SomeCo Header::help=Someco.html
  com.someco.grant:::Grant All SomeCo Authorizations::
  help=SomecoGrant.html
  com.someco.grant.device:::Grant SomeCo Device Authorizations::
  help=SomecoGrantDevice.html
  com.someco.device.cdrom.local:::Allocate Local CD-ROM Device::
  help=SomecoCDAllocateLocal.html
  com.someco.device.cdrom.remote:::Allocate Remote CD-ROM Device::
  help=SomecoCDAllocateRemote.html
  ```

- Device Allocation Manager assignment:

  Trusted Path enables authorized users to use the Device Allocation Manager when allocating the local CD-ROM.

  ```
  Device Name: cdrom_0
  For Allocations From: Trusted Path
  Allocatable By: Authorized Users
  Authorizations: com.someco.device.cdrom.local
  ```

  Non-Trusted Path enables users to allocate a device remotely by using the `allocate` command.

  ```
  Device Name: cdrom_0
  For Allocations From: Non-Trusted Path
  Allocatable By: Authorized Users
  Authorizations: com.someco.device.cdrom.remote
  ```

- Rights profile entries:

  ```
  # Local Allocator profile
  com.someco.device.cdrom.local

  # Remote Allocator profile
  com.someco.device.cdrom.remote
  ```

- Authorized users:

  ```
  # List of profiles for ordinary authorized user
  Local Allocator Profile
  ...

  # List of profiles for role or authorized user
  Remote Allocator Profile
  ...
  ```

To prevent all access to an allocatable device without using authorizations, see Example 17–1.

▼ How to Add Site-Specific Authorizations to a Device

Before You Begin You must be in the Security Administrator role, or in a role that includes the Configure Device Attributes authorization. You must have already created site-specific authorizations, as described in "How to Create New Device Authorizations" on page 199.

1. **Follow the "How to Configure a Device" on page 191 procedure.**

2. **Click the Authorizations... button to add the appropriate site-specific authorizations.**

3. **Save your changes.**

   For a complete example, see Example 17–5.

▼ How to Assign Device Authorizations

The Allocate Device authorization enables users to allocate a device. The Allocate Device authorization, and the Revoke or Reclaim Device authorization are appropriate for administrative roles.

Before You Begin You must be in the Security Administrator role in the global zone.

1. **Launch the Solaris Management Console in the appropriate scope.**

   For details, see "How to Launch the Solaris Management Console" on page 42.

2. **Navigate to the User Account tool.**

   Provide a password when prompted.

3. **Click the Rights tab.**

4 **Assign to the user a rights profile that contains the Allocate Device authorization.**

The All Authorizations profile includes administrator authorizations, such as the Revoke or Reclaim Device authorization. This profile is not suitable for an ordinary user. This profile might not be suitable for all roles.

If the existing profiles are not appropriate, the Security Administrator role can create a new profile. For an example, see "How to Create a Convenient Authorizations Rights Profile" on page 79. The Security Administrator role creates and reconfigures profiles.

Example 17–6 Assigning Device Authorizations to a Role

In this example, the Security Administrator role chooses Administrative Roles in the User Accounts tool and navigates to the Rights tab. The administrator assigns one or more of the following rights profiles.

The following profiles enable a role to allocate devices:

- All Authorizations
- Device Management
- Media Backup
- Media Restore
- Object Label Management
- Software Installation

The following profiles enable a role to revoke or reclaim devices:

- All Authorizations
- Device Management

The following profiles enable a role to create or configure devices:

- All Authorizations
- Device Security

Example 17–7 Assigning New Device Authorizations

In this example, the security administrator configures the new device authorizations. The Security Administrator role does the following:

1. Creates new device authorizations, as in "How to Create New Device Authorizations" on page 199.
2. In the Device Allocation Manager, adds the new device authorizations to the tape and diskette devices.
3. Places the new authorizations in the rights profile, NewCo Allocation.
4. Adds the NewCo Allocation profile to the profiles of users and roles who are authorized to allocate tapes and diskette drives.

Authorized users and roles can now use the tape drives and diskette drives on this system.

Using Devices in Trusted Extensions (Tasks)

In Trusted Extensions, all roles are authorized to allocate a device. Like users, roles must use the Device Allocation Manager. The `allocate` command does not work in Trusted Extensions. The following task map points to convenient procedures for using devices.

Task	For Instructions
Allocate and deallocate a device	"How to Allocate a Device" in *Solaris Trusted Extensions User's Guide*
Listen to music	"How to Listen to a Music CD" on page 204
Load a diskette for reading or writing	"How to Load Removable Media That Has a File System" on page 204

▼ How to Listen to a Music CD

Before You Begin The user must be authorized to allocate the audio and CD-ROM devices.

1. **Allocate the audio device and the CD-ROM drive.**

   Use the Device Allocation Manager. For instructions, see "How to Allocate a Device" in *Solaris Trusted Extensions User's Guide*.

2. **Insert the music CD.**

3. **When you are asked, do not mount the CD-ROM.**

   - If the administrator has defined an `audio` action in `rmmount.conf`, the `audio` action executes.

     To set up an audio action for users, see "How to Configure an Audio Player Program for Use" on page 197.

   - By default, no `audio` action is specified.

     To play the CD, invoke an audioplayer application.

Troubleshooting If you are authorized to allocate the audio and CD-ROM devices, and one of the devices is not listed, the device might be already allocated. Or, the device might be in an allocate error state. Contact the system administrator.

▼ How to Load Removable Media That Has a File System

Before You Begin You must be authorized to allocate the device. All roles are authorized to allocate devices. To grant users the Allocate Device authorization, see "How to Create a Convenient Authorizations Rights Profile" on page 79.

1. **Log in as an ordinary user.**

2 **Allocate the appropriate device in a workspace at the appropriate label.**

- **To perform an administrative task, you assume a role.**
  - **To load the file system on the media into the global zone, allocate the media in the global zone.**
  - **To load the file system on the media into a labeled zone, create a workspace at the appropriate label.**

    Then, allocate the device.
- **To load a file system that is not part of an administrative task, create a workspace at the appropriate label.**

  Then, allocate the device.

Use the Device Allocation Manager. For instructions, see "How to Allocate a Device" in *Solaris Trusted Extensions User's Guide*.

3 **Insert the media.**

4 **Answer the question about whether to mount the diskette:**

- **To read the device, answer** `yes`**.**

  After the file system on the media is mounted as part of device allocation, a File Manager appears. The current directory is set to the mount point.
- **To create or modify the file system on the media, answer** `no`**.**

  Programs to format media can create a new file system only if the file system on the media is not mounted.

Example 17–8 Loading Audit Configuration Files

In this example, roles are not yet configured on the system. The `root` user needs to copy configuration files onto removeable media. The contents of the media are then going to be copied onto other systems. These files are to be copied to each system that is configured with Trusted Extensions software.

So, the `root` user allocates the `floppy_0` device in the Device Allocation Manager, and responds `yes` to the mount query. Then `root` user inserts the diskette with the configuration files, and copies them to the disk. The diskette is labeled `Trusted Path`.

To read from the media, the root user allocates the device on the receiving host, then downloads the contents.

If the configuration files are on a tape, the `root` user allocates the `mag_0` device. If the configuration files are on a CD-ROM, the `root` user allocates the `cdrom_0` device.

Troubleshooting If you are authorized to allocate a device, but the device is not listed, the device might be already allocated. Or, the device might be in an allocate error state. Contact the system administrator.

CHAPTER 18

Trusted Extensions Auditing

This chapter describes the additions to auditing that Trusted Extensions provides.

Trusted Extensions and Auditing

On a system that is configured with Trusted Extensions software, auditing is configured and is administered similarly to auditing on a Solaris system. However, there are some differences.

- Trusted Extensions software adds audit classes, audit events, audit tokens, and audit policies to the system.
- By default, auditing is enabled in Trusted Extensions software.
- As in the Solaris OS, the `cnt` audit policy is enabled. The `ahlt` policy is disabled.
- Per-zone auditing is not supported. All audit configuration is done in the global zone, for all zones.
- Trusted Extensions provides administrative tools to administer users' audit characteristics and to edit audit files.
- Two roles, System Administrator and Security Administrator, configure and administer auditing in Trusted Extensions.

  The security administrator plans what to audit and any site-specific event-to-class mappings. As in the Solaris OS, the system administrator plans disk space for the audit files, creates an audit administration server, and installs audit configuration files.

Audit Management by Role in Trusted Extensions

Auditing in Trusted Extensions requires the same planning as it does in the Solaris OS. For details of planning, see Chapter 28, "Planning for Solaris Auditing," in *System Administration Guide: Security Services*.

Role Setup for Audit Administration

In Trusted Extensions, auditing is the responsibility of two roles. The System Administrator role sets up the disks and the network of audit storage. The Security Administrator decides what is to be audited, and enters the information in the audit configuration files. As in the Solaris OS, the site creates the roles in software. The profiles for these two roles are provided. The install team created the Security Administrator role during initial configuration. For details, see "Create the Security Administrator Role" in *Solaris Trusted Extensions Installation and Configuration*.

Note – Administrators should understand that a system only records the security-relevant events that the audit configuration files configure the machine to record (that is, by preselection). Therefore, any subsequent audit can only consider the events that have been recorded. If auditing is not configured to record the certain security-relevant events for the system, those events are not audited. This can mean that attempts to breach the security of the system go undetected, or that the administrator is unable to detect the user who is responsible for an attempted breach of security. Administrators should regularly analyze audit trails to check for breaches of security.

Audit Tasks in Trusted Extensions

The procedures to configure and manage auditing in Trusted Extensions differ slightly from Solaris procedures:

- Audit configuration is done in the global zone by one of two administrative roles. For details, see the following sections.
- Trusted Extensions administrators use CDE actions that invoke a trusted editor when editing audit configuration files. For the list of actions, see "Trusted CDE Actions" on page 28.
- Trusted Extensions administrators use the Solaris Management Console to configure individual users. Users' audit characteristics can be specified in this tool. Specifying user characteristics is only required when the user's audit characteristics differ from the audit characteristics of the systems on which the user works. For an introduction to the tool, see "Solaris Management Console Tools" on page 32.

Audit Tasks of the Security Administrator Role

The following tasks are security-relevant, and are therefore the responsibility of the Security Administrator role. Follow the Solaris instructions, and use the Trusted Extensions administrative tools.

Task	Solaris Instructions	Trusted Extensions Instructions
Configure audit files	"Configuring Audit Files (Task Map)" in *System Administration Guide: Security Services*	Use the Audit actions. For details, see "How to Launch CDE Administrative Actions in Trusted Extensions" on page 43.
(Optional) Change default audit policy	"How to Configure Audit Policy" in *System Administration Guide: Security Services*	Use the Audit Startup action.
Disable and re-enable auditing	"How to Disable Auditing" in *System Administration Guide: Security Services*	The `bsmunconv` command must be run in the global zone.
Manage auditing	"Solaris Auditing (Task Map)" in *System Administration Guide: Security Services*	Use the Admin Editor. Ignore per-zone audit tasks.

Audit Tasks of the System Administrator Role

The following tasks are the responsibility of the System Administrator role. Follow the Solaris instructions, and use the Trusted Extensions administrative tools.

Task	Solaris Instructions	Trusted Extensions Instructions
Create audit partitions, create an audit administration server, export audit partitions, and mount audit partitions. Create an `audit_warn` alias.	"Configuring and Enabling the Auditing Service" in *System Administration Guide: Security Services*	Perform all administration in the global zone. Use the Admin Editor action.
(Optional) Distribute audit configuration files		"Copying To and From Portable Media" in *Solaris Trusted Extensions Installation and Configuration*
Manage auditing	"Solaris Auditing (Task Map)" in *System Administration Guide: Security Services*	Ignore per-zone audit tasks.
Select audit records by label	"How to Select Audit Events From the Audit Trail" in *System Administration Guide: Security Services*	Use the `auditreduce` command with the `-l` option.

Trusted Extensions Audit Reference

Trusted Extensions software adds audit classes, audit events, audit tokens, and audit policy options to the Solaris OS. Several auditing commands are extended to handle labels. Trusted Extensions audit records include a label, as is shown in the following figure.

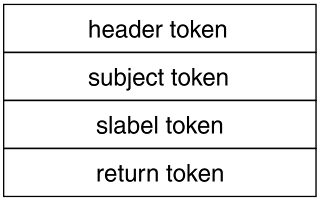


FIGURE 18–1 Typical Audit Record on a Labeled System

Trusted Extensions Audit Classes

The audit classes that Trusted Extensions software adds to the Solaris OS classes are listed alphabetically in the following table. The classes are listed in the file `/etc/security/audit_class`. For more information about audit classes, see the `audit_class`(4) man page.

TABLE 18–1 X Server Audit Classes

Short Name	Long Name	Audit Mask
xc	X - Object create/destroy	0x00800000
xp	X - Privileged operations	0x00400000
xs	X - Operations that fail silently	0x01000000
xx	X - All X events in the xc, xp, and xs classes	0x01c00000

The X server audit class events are mapped to these classes according to the following criteria:

- **xp –** This class audits for use of privilege. Privilege use can be successful or unsuccessful. For example, `ChangeWindowAttributes()` is audited when a client attempts to change the attributes of another client's window. This class also includes administrative routines, such as `SetAccessControl()`.
- **xc –** This class audits routines that do not return X error messages to clients on failure when security attributes cause the failure. For example, `GetImage()` does not return a `BadWindow` error if it cannot read from a window for lack of privilege. It does not read from that window, but creates no error message.

  These events should be selected for audit on success only. When xc events are selected for failure, the audit trail fills up with uninteresting records.
- **xs –** This class audits server objects for creation or for destruction. For example, this class audits `CreateWindow()`.
- **xx –** This class includes all the other X audit classes.

Trusted Extensions Audit Events

Trusted Extensions software adds audit events to the system. The new audit events and the audit classes to which the events belong are listed in the `/etc/security/audit_event` file. The audit event numbers for Trusted Extensions are between 9000 and 10000. For more information about audit events, see the `audit_event`(4) man page.

Trusted Extensions Audit Tokens

The audit tokens that Trusted Extensions software adds to the Solaris OS are listed alphabetically in the following table. The tokens are also listed on the `audit.log`(4) man page.

TABLE 18–2 Trusted Extensions Audit Tokens

Token Name	Description
"`label` Token" on page 211	Sensitivity label
"`xatom` Token" on page 212	X window atom identification
"`xclient` Token" on page 212	X client identification
"`xcolormap` Token" on page 213	X window color information
"`xcursor` Token" on page 213	X window cursor information
"`xfont` Token" on page 213	X window font information
"`xgc` Token" on page 214	X window graphical context information
"`xpixmap` Token" on page 214	Xwindow pixel mapping information
"`xproperty` Token" on page 214	X window property information
"`xselect` Token" on page 215	X window data information
"`xwindow` Token" on page 215	X window window information

`label` Token

The `label` token contains a sensitivity label. The fields are:

- A token ID
- A sensitivity label

The following figure shows the token format.

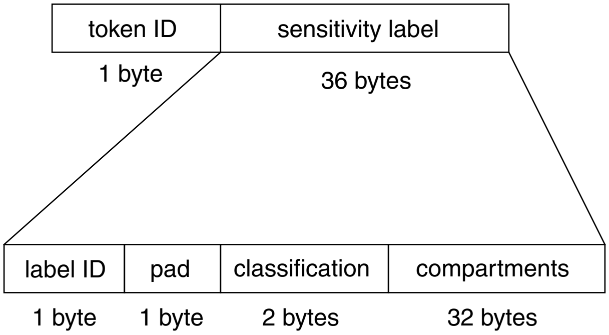


FIGURE 18–2 `label` Token Format

A `label` token is displayed by `praudit` as follows:

sensitivity label,ADMIN_LOW

`xatom` Token

The `xatom` token contains information concerning an X atom. This token contains the following fields:

- A token ID
- The string length
- A text string that identifies the atom

An `xatom` token is displayed by `praudit` as follows:

X atom,_DT_SAVE_MODE

`xclient` Token

The `xclient` token contains information concerning the X client. This token contains the following fields:

- A token ID
- The client ID

An `xclient` token is displayed by `praudit` as follows:

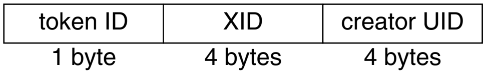
X client,15

xcolormap Token

The xcolormap token contains information about the colormaps. This token contains the following fields:

- A token ID
- The X server identifier
- The creator's user ID

The following figure shows the token format.



token ID	XID	creator UID
1 byte	4 bytes	4 bytes

FIGURE 18–3 Format for xcolormap, xcursor, xfont, xgc, xpixmap, and xwindow Tokens

An xcolormap token is displayed by praudit as follows:

X color map,0x08c00005,srv

xcursor Token

The xcursor token contains information about the cursors. This token contains the following fields:

- A token ID
- The X server identifier
- The creator's user ID

Figure 18–3 shows the token format.

An xcursor token is displayed by praudit as follows:

X cursor,0x0f400006,srv

xfont Token

The xfont token contains information about the fonts. The fields are:

- A token ID
- The X server identifier
- The creator's user ID

Figure 18–3 shows the token format.

An xfont token is displayed by praudit as follows:

X font,0x08c00001,srv

`xgc` Token

The `xgc` token contains information about the xgc. The fields are:

- A token ID
- The X server identifier
- The creator's user ID

Figure 18–3 shows the token format.

An `xgc` token is displayed by `praudit` as follows:

Xgraphic context,0x002f2ca0,srv

`xpixmap` Token

The `xpixmap` token contains information about the pixel mappings. This token contains the following fields:

- A token ID
- The X server identifier
- The creator's user ID

Figure 18–3 shows the token format.

An `xpixmap` token is displayed by `praudit` as follows:

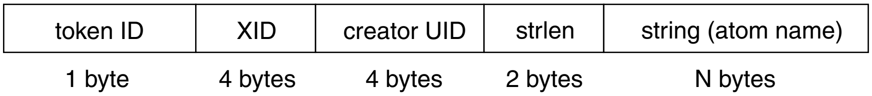
X pixmap,0x08c00005,srv

`xproperty` Token

The `xproperty` token contains information about various properties of a window. This token contains the following fields:

- A token ID
- The X server identifier
- The creator's user ID
- A string length
- A string (atom name)

The following figure shows an `xproperty` token format.



token ID	XID	creator UID	strlen	string (atom name)
1 byte	4 bytes	4 bytes	2 bytes	N bytes

FIGURE 18–4 `xproperty` Token Format

An `xproperty` token is displayed by `praudit` as follows:

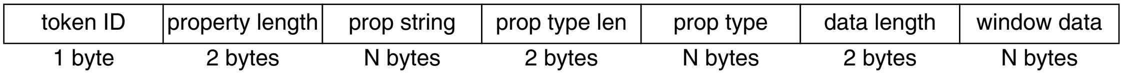
X property,0x000075d5,root,_MOTIF_DEFAULT_BINDINGS

`xselect` Token

The `xselect` token contains the data that is moved between windows. This data is a byte stream with no assumed internal structure, and a property string. This token contains the following fields:

- A token ID
- The length of the property string
- The property string
- A length for the property type
- The property type string
- A length field that gives the number of bytes of data
- A byte string that contains the data

The following figure shows the token format.



token ID	property length	prop string	prop type len	prop type	data length	window data
1 byte	2 bytes	N bytes	2 bytes	N bytes	2 bytes	N bytes

FIGURE 18–5 `xselect` Token Format

An `xselect` token is displayed by `praudit` as follows:

X selection,

`xwindow` Token

The `xwindow` token contains information about a window. The fields are:

- A token ID
- The X server identifier
- The creator's user ID

Figure 18–3 shows the token format.

An `xwindow` token is displayed by `praudit` as follows:

X window,0x07400001,srv

Trusted Extensions Audit Policy

Trusted Extensions adds the following audit policies to existing Solaris auditing policies:

```
$ auditconfig -lspolicy
...
windata_down Include downgraded window information in audit records
```

```
windata_up   Include upgraded window information in audit records
...
```

Extensions to Auditing Commands in Trusted Extensions

The `auditconfig`, `auditreduce`, and `bsmrecord` commands are extended to handle Trusted Extensions information.

- The `auditconfig` command includes the Trusted Extensions audit policies. For details, see the `auditconfig`(1M) man page.
- The `auditreduce` command adds the `-l` option for filtering records according to label. For details, see the `auditreduce`(1M) man page.
- The `bsmrecord` command includes the Trusted Extensions audit events. For details, see the `bsmrecord`(1M) man page.

CHAPTER 19

Software Management in Trusted Extensions

This chapter contains information about ensuring that third-party software runs in a trustworthy manner on a system that is configured with Trusted Extensions.

- "Adding Software to Trusted Extensions" on page 217
- "Trusted Processes in the Window System" on page 220
- "Managing Software (Tasks)" on page 221

Adding Software to Trusted Extensions

Any software that can be added to a Solaris system can be added to a system that is configured with Trusted Extensions. Additionally, programs that use Trusted Extensions APIs can be added. Adding software to a Trusted Extensions system is similar to adding software to a Solaris system that is running non-global zones.

For example, packaging issues affect systems that have installed non-global zones. Package parameters define the following:

- The zone scope of the package. The scope determines the type of zone in which an individual package can be installed.
- The visibility of the package. Visibility determines whether a package is required to be installed on all zones, and be identical in all zones.
- The limitation of the package. One limitation is whether a package must be installed in the current zone only.

In Trusted Extensions, programs are typically installed in the global zone for use by ordinary users in labeled zones. For details on installing packages in zones, see Chapter 24, "About Packages and Patches on a Solaris System With Zones Installed (Overview)," in *System Administration Guide: Solaris Containers-Resource Management and Solaris Zones* and the `pkgadd`(1M) man page.

At a Trusted Extensions site, the system administrator and the security administrator work together to install software. The security administrator evaluates software additions for adherence to security policy. When the software requires privileges or authorizations to succeed, the Security Administrator role assigns an appropriate rights profile to the users of that software.

To import software from removable media requires authorization. An account with the device allocation authorization can import or export data from removable media. Data includes programs. An ordinary user can only import data at a label within that user's clearance.

The System Administrator role is responsible for adding the programs that the security administrator approves.

Solaris Security Mechanisms

Trusted Extensions uses the same security mechanisms as the Solaris OS. The mechanisms include the following:

- **Authorizations –** Users of a program can be required to have a particular authorization. For information on authorizations, see "Solaris RBAC Elements and Basic Concepts" in *System Administration Guide: Security Services*. Also, see the `auth_attr`(4) and the `getauthattr`(3SECDB) man pages.
- **Privileges –** Programs and processes can be assigned privileges. For information on privileges, see Chapter 8, "Using Roles and Privileges (Overview)," in *System Administration Guide: Security Services*. Also, see the `privileges`(5) man page.

  The `ppriv` command provides a debugging utility. For details, see the `ppriv`(1) man page. For using this utility with programs that work in non-global zones, see "Using the ppriv Utility" in *System Administration Guide: Solaris Containers-Resource Management and Solaris Zones*.
- **Right Profiles –** Rights profiles collect security attributes in one place for assignment to users or roles. For information on rights profiles, see "RBAC Rights Profiles" in *System Administration Guide: Security Services*. Trusted Extensions adds CDE actions to the type of executables that can be given security attributes.
- **Trusted libraries –** Dynamically-shared libraries that are used by setuid, setgid, and privileged programs can be loaded only from trusted directories. As in the Solaris OS, the `crle` command is used to add a privileged program's shared library directories to the list of trusted directories. For details, see the `crle`(1) man page.

Evaluating Software for Security

When software has been assigned privileges or when the software runs with an alternate UID or GID, the software becomes *trusted*. Trusted software can bypass aspects of the Trusted Extensions security policy. Be aware that you can make software trusted even though it might not be worthy of trust. The Security Administrator role should not give any privileges to software until careful scrutiny has revealed that the software uses the privileges in a trustworthy manner.

- **Programs that require no security attributes –** Some programs run at a single level and require no privileges. These programs can be installed in a public directory, such as `/usr/local`. For access, assign the programs as commands in the rights profiles of users and roles.

- **Programs that run as root –** Some programs execute with `setuid 0`. Such programs can be assigned an effective UID of root in a rights profile. The Security Administrator role then assigns the profile to an administrative role.

  If the application can use privileges in a trustworthy manner, assign the needed privileges to the application, and do not execute the program as root.
- **Programs that require privileges –** Some programs might need privileges for reasons that are not obvious. Even if a program is not performing any function that seems to violate system security policy, the program might be doing something internally that violates security. For example, the program could be using a shared log file, or the program could be reading from `/dev/kmem`. For security concerns, see the `mem`(7D) man page.

  Sometimes an internal policy override is not particularly important to the application's correct operation. Rather, the override provides a convenient feature for users. If your organization has access to the source code, check if you can remove the operations that require policy overrides without affecting the performance of the application.

Developer Responsibilities When Creating Trusted Programs

Even though a program's developer can manipulate privilege sets in the program's source code, if the Security Administrator role does not assign the required privileges, the program will fail. The developer and security administrator cooperate when creating trusted programs.

A developer who writes a trusted program must do the following:

1. Understand where the program requires privileges to do its work.
2. Know and follow techniques, such as privilege bracketing, for safely using privileges in programs.
3. Be aware of the security implications when assigning privileges to a program. Make sure that the program does not violate security policy.
4. Compile the program by using shared libraries that are linked to the program from a trusted directory.

   For additional information, see *Solaris Security for Developers Guide*. For examples of code for Trusted Extensions, see *Solaris Trusted Extensions Developer's Guide*.

Security Administrator Responsibilities for Trusted Programs

The security administrator is responsible for testing and evaluating new software. After determining that the software is trustworthy, the Security Administrator role configures rights profiles and other security-relevant attributes for the program.

1. Make sure the programmer and the program distribution process is trusted.
2. From one of these sources, find out which privileges are required by the program:
   1. Ask the programmer.
   2. Search the source code for any privileges that the program expects to use.
   3. Search the source code for any authorizations that the program requires of its users.

   d. Use the debugging options to the ppriv command to search for use of privilege. For examples, see the ppriv(1) man page.

3. Examine the source code to make sure that the code behaves in a trustworthy manner when using the privileges that the program needs to operate.

   If the program fails to use privilege in a trustworthy manner, and you can modify the program's source code, then modify the code. A security consultant or programmer who is knowledgeable about security can modify the code. Modifications might include privilege bracketing or checking for authorizations.

   The assignment of privileges should not be automatic. A program that fails due to lack of privilege can be assigned privileges. Alternatively, the Security Administrator role might decide to assign an effective UID or GID to make the privilege unnecessary.

Trusted Processes in the Window System

In Solaris Trusted Extensions (CDE), the following window system processes are trusted:

- Front Panel
- Subpanels of the Front Panel
- Workspace Menu
- File Manager
- Application Manager

The window system's trusted processes are available to everyone, but access to administrative actions is restricted to roles in the global zone.

In the File Manager, if an action is not in one of the account's profiles, the icon for the action is not visible. In the Workspace Menu, if an action is not in one of the account's profiles, the action is visible, but an error displays if the action is invoked.

In CDE, the window manager, dtwm, calls the Xtsolusersession script. This script works with the window manager to invoke actions that are launched from the window system. The Xtsolusersession script consults the account's rights profiles when the account attempts to launch an action. In either case, if the action is in an assigned rights profile, the action is run with the security attributes that are specified in the profile.

Adding Trusted CDE Actions

The process of creating and using CDE actions is similar in Trusted Extensions to the process in the Solaris OS. Adding actions is described in the Chapter 4, "Adding and Administering Applications," in *Solaris Common Desktop Environment: Advanced User's and System Administrator's Guide*.

As in the Solaris OS, the use of actions can be controlled by the rights profile mechanism. In Trusted Extensions, a number of actions have been assigned security attributes in the rights profiles of administrative roles. The Security Administrator role can also use the Rights tool to assign security attributes to new actions.

The following table summarizes the main differences that are encountered in creating and using actions in the Solaris Trusted Extensions system.

TABLE 19–1 Constraints on CDE Actions in Solaris Trusted Extensions

Solaris CDE Actions	Solaris Trusted Extensions (CDE) Actions
New actions can be created by anyone within the originator's home directory. A new action is automatically usable by its creator.	An action is usable only if the action is a rights profile that is assigned to the user. The search path for actions has been changed. Actions in any individual's home directory are processed last instead of first. Therefore, no one can customize existing actions. Users can create a new action in their home directory, but the action might not be usable. Users with the All profile can use an action that they create. Otherwise, the Security Administrator role must add the name of the new action to one of the account's rights profiles. To launch the action, the user uses the File Manager. The System Administrator role can place actions in public directories.
Actions can be dragged and dropped to the Front Panel.	The Front Panel is part of the trusted path. The window manager recognizes only the administratively-added actions that are located in the `/usr/dt` and `/etc/dt` subdirectories. Even with the All profile, a user cannot drag a new action to the Front Panel. Action from a user's home directory are not recognized by the window manager. The manager only looks in the public directories.
Actions can do privileged operations if they are run by root.	Actions can do privileged operation if the actions have been assigned privileges in a rights profile that has been assigned to a user.
Actions are not managed by the Solaris Management Console.	Actions are assigned to rights profiles in the Rights tool of the Solaris Management Console. If new actions are added, the Security Administrator role can make the new actions available.

Managing Software (Tasks)

Managing software in Trusted Extensions is similar to managing software on a Solaris system that has non-global zones installed. For details about zones, see Part II, "Zones," in *System Administration Guide: Solaris Containers-Resource Management and Solaris Zones*.

▼ How to Add a Software Package

Before You Begin You must be in a role that can allocate a device.

1 Start from the appropriate workspace.

- **To install a software package in the global zone, stay in the global zone.**

- **To install a software package in a labeled zone, create a workspace at that label.**

  For details, see "How to Work at a Different Label" on page 56.

2 **Allocate the CD-ROM device.**

For details, see "How to Allocate a Device" in *Solaris Trusted Extensions User's Guide*.

3 **Install the software.**

4 **Deallocate the device when you are finished.**

▼ How to Install a Java Jar File

This procedure downloads a Java jar file to the global zone. From the global zone, the administrator can make it available to ordinary users.

Before You Begin The security administrator has verified that the source of the Java program is trustworthy, that the method of delivery is secure, and that the program can run in a trustworthy manner.

You are in the System Administrator role in the global zone.

1 **Download the Java jar file to the `/tmp` directory.**

For example, if you are selecting software from the http://www.sunfreeware.com, download the link that is described with Web Start Wizard form of *application*.

2 **Open a File Manager to the `/tmp` directory.**

The Software Installation profile includes the Open action for Java code.

3 **Double-click the downloaded file.**

4 **Answer the questions in the dialog boxes to install the software.**

5 **Read the installation log.**

Example 19–1 Downloading a Java Jar File to a User Label

To limit the security risk, the system administrator downloads the software to a single label within an ordinary user's accreditation range. Then, the security administrator tests the jar file at that label. The administrator then downgrades the label to `ADMIN_LOW`, and installs it on an NFS server to make it available to all users.

1. In the System Administrator role, create a workspace at a user label.
2. In that workspace, download the Java jar file.
3. In that workspace, test the file.

4. Then, change the label of the file to `ADMIN_LOW`.
5. The System Administrator role then copies the file to an NFS server whose label is `ADMIN_LOW`.

APPENDIX A

Quick Reference to Trusted Extensions Administration

Trusted Extensions interfaces extend the Solaris OS. This appendix provides a quick reference of the differences. For a detailed list of interfaces, including library routines and system calls, see *Solaris Trusted Extensions Transition Guide*.

Administrative Interfaces in Trusted Extensions

Trusted Extensions provides interfaces for its software. The following interfaces are available only when Trusted Extensions software is running.

Admin Editor action

In Trusted Extensions, this editor is used to edit system files. In CDE, the Admin Editor action is in the Trusted_Extensions folder in the Application Manager.

Other Trusted CDE actions

The Trusted_Extensions folder in the Application Manager contains CDE actions that configure files, install and boot zones, and simplify other Trusted Extensions tasks. For the tasks that these actions perform, see "Trusted CDE Actions" on page 28. CDE online help also describes the actions.

Device Allocation Manager

In Trusted Extensions, this GUI is the interface to administering devices. The Device Administration dialog box is used by administrators to configure devices.

The Device Allocation Manager is used by roles and by ordinary users to allocate devices. The GUI is available from the Trusted Path menu.

Label Builder

This application appears when the user can choose a label or a clearance. This application also appears when a role is assigning labels to zones, users, or roles.

Selection Manager

This application appears when an authorized user or authorized role attempts to upgrade or downgrade information.

Trusted Path menu

This menu handles interactions with the trusted computing base (TCB). For example, this menu has a Change Password menu item. In CDE, you get the Trusted Path menu from the workspace switch area. This menu is also a user interface.

Administrative commands

Trusted Extensions provides commands to get labels and do other tasks. For a list of the commands, see "Command Line Tools" on page 34.

Extensions of Existing Solaris Interfaces

Trusted Extensions adds to existing configuration files, commands, and GUIs.

Administrative commands

Trusted Extensions adds options to selected Solaris commands. For a list, see Table 2–5.

Configuration files

Trusted Extensions adds two privileges, `net_mac_aware` and `net_mlp`. For the use of `net_mac_aware`, see "Access to NFS Mounted Directories in Trusted Extensions" on page 116.

Trusted Extensions adds authorizations to the `auth_attr` database. For a list, see "Additional Rights and Authorizations in Trusted Extensions" in *Solaris Trusted Extensions Transition Guide*.

Trusted Extensions adds executables, including CDE actions, to the `exec_attr` database.

Trusted Extensions modifies existing profiles in the `prof_attr` database. It also adds profiles to the database.

Trusted Extensions adds CDE actions to the executables that can be privileged in the `exec_attr` database.

Trusted Extensions adds fields to the `policy.conf` database. For the fields, see "`policy.conf` File Defaults" on page 66.

Trusted Extensions adds audit tokens, audit events, audit classes, and audit policy options. For a list, see "Trusted Extensions Audit Reference" on page 209.

Solaris Management Console

Trusted Extensions adds a Security Templates tool to the Computers and Networks tool set.

Trusted Extensions adds a Trusted Network Zones tool to the Computers and Networks tool set.

Trusted Extensions adds a Trusted Extensions Attributes tab to the Users tool and the Administrative Roles tool.

Tighter Security Defaults in Trusted Extensions

Trusted Extensions establishes tighter security defaults than the Solaris OS.

Auditing

By default, auditing is enabled.

An administrator can turn off auditing, but auditing is typically required at sites that install Trusted Extensions.

Devices

By default, device allocation is enabled.

By default, device allocation requires authorization. Therefore, by default, ordinary users cannot use removable media.

An administrator can remove the authorization requirement. Removing the device allocation requirement is similar to turning off auditing.

Printing

Ordinary users can print only to printers that include the user's label in the printer's label range.

By default, print output has trailer and banner pages. These pages, and the body pages, include the label of the print job.

By default, users cannot print PostScript files.

Roles

Roles are available in the Solaris OS, but their use is optional. In Trusted Extensions, roles are required for proper administration.

Making `root` a role is possible in the Solaris OS. In Trusted Extensions, `root` is made a role to better audit who is acting as superuser.

Limited Options in Trusted Extensions

Trusted Extensions narrows the range of Solaris configuration options.

Desktop

Trusted Extensions offers two desktops, the Solaris Trusted Extensions (CDE) and the Java DS.

Naming service

The LDAP naming service is supported. All zones must be administered from one naming service.

Zones

The global zone is an administrative zone. Only the `root` user or a role can enter the global zone. Therefore, administrative interfaces that are available to ordinary Solaris users are not available to ordinary Trusted Extensions users. For example, in Trusted Extensions, ordinary users cannot bring up the Solaris Management Console.

Non-global zones are labeled zones. Users work in labeled zones.

All zones must be administered from one naming service.

Index