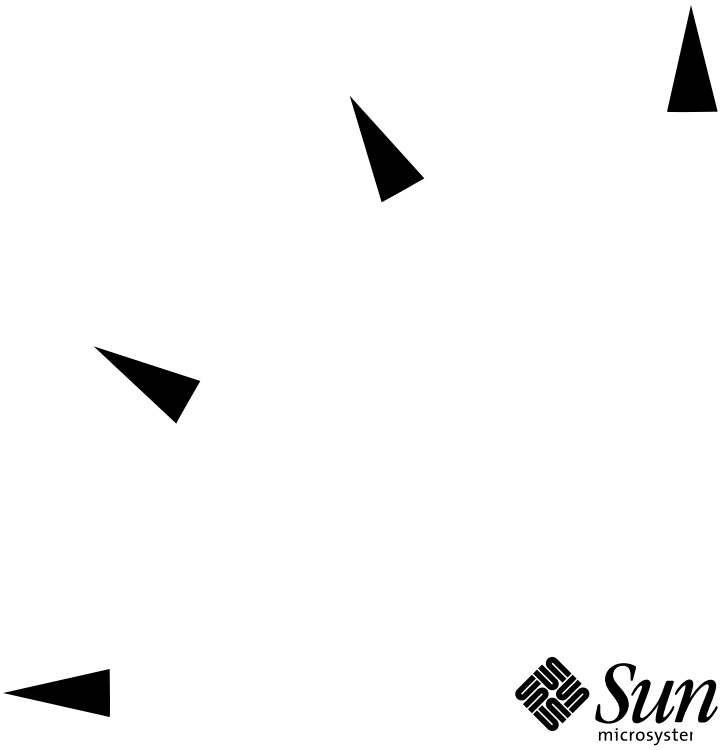
Trusted Solaris 2.5.1 Transition Guide



Sun Microsystems Federal, Inc.
A Sun Microsystems, Inc. Business
901 San Antonio Road
Palo Alto, CA 94303
U.S.A.

Part No. 805-8030-10
Revision A, August 1998

THE NETWORK IS THE COMPUTER™

Contents

Tables

Preface

This guide presents an overview of the differences between the Trusted Solaris 2.5.1 operating environment and its base, the Solaris 2.5.1 operating environment, and the changes and enhancements of the Trusted Solaris 2.5.1 environment over the Trusted Solaris 1.2 environment. See the Trusted Solaris 2.5.1 document set and man pages for how to use the new Trusted Solaris 2.5.1 features.

Who Should Use This Book

This book is for users, programmers, and administrators familiar with previous versions of the Trusted Solaris environment, and for users, programmers, and administrators familiar with the Solaris 2.5.1 environment.

Related Materials

The Trusted Solaris 2.5.1 document set is supplemental to the Solaris 2.5.1 document set. You should obtain a copy of both sets for a complete understanding of the Trusted Solaris 2.5.1 environment. Books of particular help when addressing transition include:

Trusted Solaris Books

- *Trusted Solaris User's Guide:* Describes basic features of the Trusted Solaris environment that are common to all users of the system. Includes a glossary.

- *Trusted Solaris Administration Overview:* Explains basic concepts and terminology commonly used throughout the Trusted Solaris 2.5.1 environment.
- *Trusted Solaris Administrator's Procedures:* Explains administrative procedures in the Trusted Solaris 2.5.1 environment that differ from the Solaris 2.5.1 environment.

Base Solaris Books

- *Solaris 2.5 Introduction*: Tabulates differences between Solaris 1.0 plus its compatible versions, and later Solaris releases.
- *Solaris 1.x to 2.x Transition Guide*: Details differences in structure and use between Solaris 1.0 plus its compatible versions, and later Solaris releases.

How This Book is Organized

This guide is divided into a high-level overview, changes that affect users and administrators, and changes that affect programmers. The appendixes summarize privileges, authorizations, and changed programming interfaces.

Chapter 1– Overview gives an overall view of the differences between the Trusted Solaris 2.5.1 environment and its Solaris 2.5.1 base, and between the Trusted Solaris 2.5.1 environment and the Trusted Solaris 1.2 version. All users should read this chapter.

Chapter 2– User and Administrator Transition describes how the Trusted Solaris 2.5.1 environment differs from the Solaris 2.5.1 environment and the Trusted Solaris 1.2 environment from the perspective of a desktop user, a system administrator, and a security administrator. It describes desktop changes, administration utilities changes, and other changes in the Trusted Solaris environment. All users and administrators should read this chapter.

Chapter 3 – Programmer Transition describes how the Trusted Solaris 2.5.1 environment differs from the Solaris 2.5.1 environment and the Trusted Solaris 1.2 environment from the perspective of a developer. It lists changes in location for man pages, header files, and databases. It also lists modified programming interfaces (APIs), commands, and macros. Developers should read this chapter.

Appendix A – Trusted Solaris Programming Interfaces lists preserved, removed, and changed Trusted Solaris 2.5.1 programming interfaces (APIs), commands, and macros.

Appendix B – Privileges and Authorizations Transition describes how Trusted Solaris 2.5.1 privileges and authorizations differ from their Trusted Solaris 1.2 implementation.

Typographic Changes and Symbols

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

Table P-1 Typographic Conventions

Typeface or Symbol	**Meaning**	**Example**
`AaBbCc123`	The names of commands, files, and directories; on-screen computer output	Edit your `.login` file. Use `ls -a` to list all files. `system% You have mail.`
`AaBbCc123`	What you type, contrasted with on-screen computer output	`system% su` `Password:`
AaBbCc123	Command-line placeholder or variable name. Replace with a real name or value	To delete a file, type `rm` *filename*. The *errno* variable is set.
AaBbCc123	Book titles, new words or terms, or words to be emphasized	Read Chapter 6 in *User's Guide*. These are called *class* options. You *must* be root to do this.
Code samples are in `code font` and may display the following:		
`%`	UNIX C shell prompt	`machine%`
`$`	UNIX Bourne shell, Korn shell, and Profile shell prompt	`$`
`#`	root prompt, all shells	`#`

Overview 1

The Trusted Solaris 2.5.1 operating environment is built on

- the Solaris 2.5.1 environment plus hardware additions and patches (This is sometimes called “the base” or “base Solaris”.)
- the Common Desktop Environment (CDE) 1.1, and
- the Solstice AdminSuite™ 2.3 databases.

Like the Trusted Solaris 1.2 version, Trusted Solaris 2.5.1 runs on SPARC hardware. Unlike the Solaris 2.5.1 environment, it does not run on the x86 or PowerPC platforms.

The Trusted Solaris 2.5.1 environment is an upgrade to Trusted Solaris 2.5 and to the Trusted Solaris 1.2 version, which was built on Solaris 1.1. It has been enhanced to accommodate different commercial and government environments. Also, many Trusted Solaris security features can be customized to accommodate site-specific requirements for daily operations.

This table lists major topics for the chapter.

1

Main Differences

The main difference between the Solaris 2.5.1 environment and Trusted Solaris 2.5.1 is that the Trusted Solaris environment is more trusted. To achieve trust, it breaks up superuser, it restricts the operation of security-relevant programs, and it adds security features, such as labels, to files and processes. Since the user's desktop is part of Trusted Solaris security, CDE is bundled with the product, and is the required desktop for users, administrators, and developers. No other desktop is supported.

There are many differences between the Trusted Solaris 1.2 operating environment and Trusted Solaris 2.5.1. Among them are the desktop environment, the administration of Trusted Solaris databases using the NIS+ name service, execution profiles, label configuration, auditing record format, and reworked authorizations and privileges. Solstice AdminSuite tools are installed as part of the product and provide the administrative interface for the Trusted Solaris network of users and workstations.

Differences from Solaris 2.5.1 Release

The Solaris 2.5.1 features listed in the following table are not available in the Trusted Solaris 2.5.1 environment.

Table 1-1 Differences from Solaris 2.5.1 Release

Solaris 2.5.1 Feature	Trusted Solaris Information on this Feature
NIS (Yellow Pages)	NIS+ is the required name service.
cache file system (cachefs)	Cachefs™ and AutoClient™ are not supported.
UNIX-to-UNIX Copy Program (UUCP)	Network protocols TCP and UDP are supported.
Federated Naming Service (FNS)	Not supported.
Accounting	Not supported.
Quotas	Not supported.
Volume Management	To mount a volume, allocate the device. Requires administrative authorization.
OpenLook Window Manager	Trusted CDE Window Manager is required (dtwm).

Table 1-1 Differences from Solaris 2.5.1 Release

Solaris 2.5.1 Feature	Trusted Solaris Information on this Feature
remote filesystem mounting at installation	File systems are mounted after installation.
upgrade installation option	You can upgrade from Trusted Solaris 2.5; you cannot upgrade from Trusted Solaris 1.2.
`sys-unconfig(1M)`	Does not reset Trusted Solaris configuration files.
`bsmconv(1M)` and `bsmunconv(1M)` scripts not used	Modify `/etc/init.d/audit` script and set audit_load kernel switch to disable and enable auditing.

Differences from Trusted Solaris 1.2 Release

The Trusted Solaris 1.2 features listed in the following table are not available in Trusted Solaris 2.5.1 environment.

Table 1-2 Trusted Solaris 1.2 Features Not Supported

Trusted Solaris 1.2 Feature	Further Information
per file auditing	Use the public object flag.
consistency checking	Actions and other features ensure consistency.
Trusted NeWSprint	Printing is now based on System V, Release 4 (SVR4) printing.

Differences from Trusted Solaris 2.5 Release

The Trusted Solaris 2.5.1 release adds the following features to the Trusted Solaris operating environment.

- The `autofs` file system is supported; file systems can be automounted.
- Dynamic routing is available.
- Security attributes specified at mount time cannot override attributes already specified for a file system.

- Two filesystem-specific mount options used with -o (or placed in the `/etc/vfstab` file) are supported: `devices|nodevices privs|noprivs`. `nodevices` restricts devices to standard directory locations. `noprivs` prevents privileged applications from operating from non-standard locations.
- The policy for writing to unlabeled machines is write up, not write equal.
- Customers can install their own randomword function in the PAM (Pluggable Authentication Module).
- Information Labels (ILs) are off by default when installing a workstation.
- MAXBADLOGINS is set to 3: administrative intervention is required to unlock a user's account after 3 unsuccessful login attempts.
- Authorization is required to shut down a workstation; Stop-A is controlled by the abort_enable setting in `/etc/system`.
- X window policy is configurable via the file, `config.privs`.
- Administrative roles have the profile shell as their default shell.
- "Custom *administrative* Role" profiles have been added for site modifications to administrative roles (admin, oper, root, secadmin).
- Basic NIS+ commands have been added to System_Admin actions.
- There are new audit events and audit classes, such as the class `pm`, process modify. Audit class masks have been changed.
- CDE workspace menu can be customized by a user in the `.dtwmrc` file.
- An untrusted client window can no longer occupy a trusted role workspace.
- Users must explicitly click to expand a minimized Front Panel.
- File Manager, Selection Manager, and Label Builder interfaces are modified.
- Calendar Manager is polyinstantiated, like Mail and other front panel actions.
- Desktop access tool for people with disabilities id on the Trusted Desktop subpanel above the Style Manager on the Front Panel.
- Device Manager mounts a device when it is allocated.
- MIT Magic Cookie is written at the user's clearance.
- Terminal login is supported.

- Dates after 1999 (2000-on) are correctly interpreted throughout the environment.

See also the *Trusted Solaris 2.5.1 Release Notes* for bugs fixed between releases.

Trusted Solaris Security Features

The security features listed in the following table are implemented in the Trusted Solaris environment, but not in the Solaris 2.5.1 environment.

Table 1-3 Trusted Solaris 2.5.1 Security Features

Trusted Solaris 2.5.1 Feature	**Further Information**
Labels and Clearances	Labels and clearances label information (including processes) and regulate who can access the information. Based on the label encodings file.
Administrative Roles	Roles divide the responsibilities of superuser. Users assume a role; a role cannot log in.
Authorizations	Authorizations enable a user or role to perform security-relevant tasks.
Device Allocation	Devices are accessible to users when authorized by the security administrator. Devices must be explicitly allocated and deallocated for use.
Privileges	Privileges enable commands and actions to perform security-relevant tasks.
Trusted Networking	Trusted Solaris security attributes are applied to all network packets.
Trusted Path	The trusted path is a secure communication path between a user or process and the system's trusted software.

The security features listed in the following table are implemented in the Trusted Solaris 2.5.1 environment, but not in the Solaris 2.5.1 environment nor in the Trusted Solaris 1.2 environment.

Table 1-4 New Trusted Solaris 2.5.1 Security Features

Trusted Solaris 2.5.1 Feature	**Further Information**
Execution Profiles	Profiles describe the commands and actions a user (or role) can execute, any privileges or security restrictions on those actions and commands, and authorizations granted to a user (or role). Profiles package what a user is allowed to do in the environment.
Label Configuration	At installation, labels can be configured.
Label Visibility	Labels can be hidden (from windows, from icons) on a per workstation and per user basis.
Common Desktop Environment	CDE is installed from the Trusted Solaris CD-ROM and is enabled by the program. It has been enhanced to handle and enforce Trusted Solaris security features.
Multilevel home directories	Users' home directories are created multilevel.
Multilevel CDE workspaces	CDE workspaces are multilevel. Applications operate at the label of the invoking (workspace) process.
Multilevel (Polyinstantiated) Ports	Multiple instances of network ports with the same port number can exist at different labels.

Transition from Trusted Solaris 1.2 Release

The following table gives a broad view of the main changes between the Trusted Solaris 1.2 and Trusted Solaris 2.5.1 releases.

Table 1-5 Main Changes Between Trusted Solaris 1.2 and Trusted Solaris 2.5.1 Releases

Area of System	**Further Information**
Adding software	Trusted Solaris 2.5 software packages now have security attributes.
Administrative utilities	See manpage sections (1tsol) and (1mtsol) rather than (1t), (8t) or (8).
	Distributed databases (users, roles, profiles, remote hosts) are handled through NIS+ tables.
	Solstice AdminSuite is the graphic user interface to distributed databases.
	/etc/rc.conf file replaced by /etc/system file.
	/etc/m6.routes file replaced by /etc/defaultrouter or /etc/tsolgateways file.
	Dynamic routing enabled using /etc/tunnel file.
	openwin-menu file replaced by /usr/dt/config/C/sys.dtwmrc
	CMWXdefaults file replaced by various files in /usr/dt and /usr/openwin/server/tsol
Auditing	Auditing is enabled by default; audit record format is incompatible with TS1.2 records.
Desktop	CDE, not OpenWindows.
Devices	Configurable security policy file, `device_policy(4TSOL)`
File system, local and mounted	File systems now have filesystem-wide security attributes, additions to the Solaris 2.5.1 file system attributes.
Installation	Installation is similar to installing Solaris 2.5.1. The installation GUI is Motif-based.

Table 1-5 Main Changes Between Trusted Solaris 1.2 and Trusted Solaris 2.5.1 Releases

Area of System	Further Information
	CDE is installed from the Trusted Solaris CD-ROM.
	Network installation requires the install and boot server to be the same workstation.
	Auditing is enabled during installation.
	Solstice AdminSuite is installed from the Trusted Solaris CD-ROM.
Name service	Only the NIS+ name service is supported. The NIS compatibility package is not supported.
Network interface	Network interfaces can be secured using the local database, `tnidb(4TSOL)`.
Network remote hosts	Remote host databases renamed `tnrhdb` and `tnrhtp`, the latter providing templates and wildcard fallback feature. They are NIS+ tables, managed through the Solstice AdminSuite Database Manager.
Network routing	For static routing, use `/etc/defaultrouter` or `/etc/tsolgateways`.
	Dynamic routing is now implemented. For tweaking, use the file `/etc/tunnel`.
NIS+ databases	Trusted Solaris databases `tsoluser`, `tsolprof`, `tnrhdb`, and `tnrhtp` are added to the regular NIS+ databases when populating the NIS+ tables, and are in the nsswitch.conf file. Trusted Solaris configuration information can be added to the `bootparams` NIS+ database for network installation.

Table 1-5 Main Changes Between Trusted Solaris 1.2 and Trusted Solaris 2.5.1 Releases

Area of System	Further Information
Role access	Roles are users who cannot log in. Roles have their own workspaces and are assigned their own passwords by the security administrator. Roles do not have reserved UIDs.
User workspace	The Common Desktop Environment is the required desktop for users and administrators. Openwin programs will run.
Programming interfaces	X11R4 protocols replaced by X11R5 protocols. Xnews server replaced by Xsun server. Tnet protocols replaced by TSIX protocols. Privileges and authorizations reworked.

1

User and Administrator Transition 2

This chapter describes the Trusted Solaris 2.5.1 changes that affect users and administrators. Administrators can set up a user's interface so that labels are not visible, only those applications and commands that help get the user's job done are available, and every label in the user's accreditation range is accessible during one CDE session.

The Trusted Solaris 2.5.1 environment has been changed and enhanced for users and administrators in the following areas.

2

Installation

Installing the Trusted Solaris 2.5.1 operating environment interactively has the look and feel of installing the Solaris 2.5.1 operating environment interactively. However, NIS+ is the only name service choice, dataless clients are not supported, and remote file systems cannot be mounted at installation time. Labels are configured during installation (see “Label Configuration” on page 18), and trusted versions of the Common Desktop Environment (CDE) 1.1 and Solstice AdminSuite 2.3 are installed from the Trusted Solaris 2.5.1 CD-ROM.

Trusted Solaris 2.5.1 network installation enables getting Trusted Solaris label configuration settings from the `bootparams` database on the install/boot server. The `bootparams` database is a NIS+ table, administered through the Database Manager in Solstice AdminSuite.

The `sys-unconfig(1M)` command has not been extended to reset Trusted Solaris configuration files.

Upgrade Option

The upgrade option during installation applies to upgrading from Trusted Solaris 2.5 to Trusted Solaris 2.5.1 only.

The upgrade script attempts to change the root’s login shell from `/bin/sh` to `/bin/pfsh` without administrative intervention. To take advantage of other changes in Trusted Solaris 2.5.1, such as dynamic routing, requires administrative intervention after installation.

Login

The Trusted Solaris 2.5.1 CDE login screen offers fewer options than its base counterpart. There is no Command Line Login. The Session options are limited to Common Desktop Environment, User’s Last Desktop, Remote Login, and Failsafe Session. The Failsafe session brings up a CDE workspace where none of the user’s files that usually run at login, such as `.login`, `.profile`, or `.cshrc`, are run. Logins must be enabled before users can log in, and during login, the user is shown messages and label information. Users who are allowed to work at multiple labels have the option of restricting their session to a single label, or to lowering their clearance.

Administration

The Trusted Solaris 2.5.1 environment is configured and administered using trusted versions of CDE and Solstice AdminSuite. As in Trusted Solaris 1.2 versions, there is a trusted, restricted shell, `pfsh` (profile shell), for command line input and a trusted editor for editing local administrative files. Following the CDE desktop metaphor, the trusted editor is invoked through CDE actions.

Databases

The following table shows the additions, removals, and new locations of databases used for Trusted Solaris administration.

Table 2-1 Trusted Solaris 2.5.1 Databases

Trusted Solaris 1.2 Name and Location	Trusted Solaris 2.5.1 Equivalent and Location
`/etc/exports`	`/etc/dfs/dfstab`
`/etc/fstab`	`/etc/vfstab`
`/etc/m6.routes`	`/etc/tsolgateways, /etc/defaultrouter`
`/etc/rc.boot, rc.single, etc.`	`/etc/rc`*`n`*`.d/`*`scripts`*
`/etc/rc.conf`	`/etc/system`

2

Table 2-1 Trusted Solaris 2.5.1 Databases

Trusted Solaris 1.2 Name and Location	Trusted Solaris 2.5.1 Equivalent and Location
/etc/security/audit_~	/etc/security/audit_~
/etc/security/fstab.adjunct	/etc/security/tsol/vfstab_adjunct
/etc/security/label_encodings	/etc/security/tsol/label_encodings
/etc/security/tcb_static, tcb_dynamic	No longer supported.
/etc/security/TNETCONFDB	No longer supported.
/etc/security/TNETIDB	/etc/security/tsol/tnidb
/etc/security/TNETRHDB	/etc/security/tsol/tnrhdb
Not in 1.2. Provides host type templates.	/etc/security/tsol/tnrhtp
Not in 1.2. For tunneling (dynamic routing).	/etc/security/tsol/tunnel
Not in 1.2. For bootstrapping the network info.	/etc/security/tsol/boot
/etc/security/authoriz, tfm_role.~	/etc/security/tsol/tsolprof
/etc/security/passwd.adjunct	/etc/security/tsol/tsoluser
/etc/security/user.cmw_labels	/etc/security/tsol/tsoluser
/etc/security/device_allocate	/etc/security/device_allocate
/etc/security/device_maps	/etc/security/device_maps
Not in 1.2. For device policy.	/etc/security/tsol/device_policy
Not in 1.2. For privilege debugging.	/etc/syslog.conf
Not in 1.2. For privilege debugging.	/var/log/privdebug.log
/etc/security/ctab_list	No longer supported.
/etc/tfm/files	/etc/security/tsol/tsoluser
/etc/tfm/nis	Replaced by NIS+ tables.
/etc/tfm/updaters	No longer supported.
/usr/openwin/lib (OpenWindows)	/usr/dt (CDE)
/usr/openwin/lib/CMWXdefaults	/usr/openwin/server/tsol/~.atoms /usr/openwin/server/tsol/config.privs /usr/dt/app-defaults/C/Dtwm /usr/dt/app-defaults/C/Dt /usr/dt/app-defaults/C/Sel_Mgr

Table 2-1 Trusted Solaris 2.5.1 Databases

Trusted Solaris 1.2 Name and Location	Trusted Solaris 2.5.1 Equivalent and Location
	`/usr/dt/config/sel_config`
`/usr/openwin/lib/openwin-menu`	`/usr/dt/config/C/sys.dtwmrc`
Not in 1.2. New in Solaris 2.5.1 environment.	`/proc`

NIS+ Name Service

The NIS+ name service is the only name service supported by the Trusted Solaris 2.5.1 operating environment. Therefore, a Trusted Solaris 2.5.1 NIS+ root master cannot administer Trusted Solaris 1.2 workstations or Solaris 2.5.1 workstations, although it can communicate with them. The Trusted Solaris NIS+ implementation does not offer NIS (Yellow Pages) compatibility.

Administrative Graphic User Interfaces

A trusted version of Solstice AdminSuite is the graphic user interface (GUI) for administering distributed (and local) Trusted Solaris databases. It shows up as a folder named Solstice_Apps in the Application Manager, a front panel icon in CDE. Trusted administrative interfaces to local databases not administered by Solstice AdminSuite show up as actions in the System_Admin folder in the Application Manager. Only a user in an administrative role with the appropriate authorizations can use the actions in the Solstice_Apps and System_Admin folders. The Trusted Solaris 1.2 Trusted Management Facility is no longer supported.

Trusted Solstice AdminSuite Administration

The Solstice_Apps folder holds the database managers, such as User Manager, for entering information into NIS+ tables. Trusted Solaris 2.5.1 adds four databases: `tsolprof(4TSOL)`, `tsoluser(4TSOL)`, `tnrhdb(4TSOL)`, and `tnrhtp(4TSOL)` to the databases populated by the `nispopulate` command. It also adds a *keyword:value* entry to the `bootparams` database, used during network installation. Online help for the modified and new databases is available; the reference (glossary) has been updated.

Passwords are changed using a password GUI available from the Trusted Path menu. Expired passwords can also be changed via `rlogin(1)`, `telnet(1)` and `login(1TSOL)`. A password generator can automatically generate possible passwords for users. Customers may substitute their own randomword generator in the PAM (Pluggable Authentication Module). Setting a password using the `nispasswd` or the `passwd` command on the command line is not supported.

The User Manager has been modified to incorporate Trusted Solaris security attributes. The User Manager and its database, `/etc/security/tsol/tsoluser`, replace the Trusted Solaris 1.2 `/etc/security/user.cmw_labels` file and its `/etc/tfm/nis` table equivalent.

The Profile Manager with its database, `/etc/security/tsol/tsolprof`, is new to Solstice_Apps and to Trusted Solaris software. It replaces the functionality provided in Trusted Solaris 1.2 software by `/etc/security` files and its `/etc/tfm/nis` tables that listed what roles could use what commands.

Three Trusted Solaris databases are added to the Database Manager's databases, `/etc/security/tsol/tnrhdb`, `/etc/security/tsol/tnrhtp`, and `/etc/security/tsol/tnidb`. The first two replace the Trusted Solaris 1.2 `/etc/security/TNETRHDB` file. The second file, `tnrhtp(4TSOL)`, creates templates that describe host types. The third file, `tnidb(4TSOL)`, is a local database that controls the workstation's network interface.

The following Solstice AdminSuite managers remain unchanged in Solstice_Apps; they do not detect or add Trusted Solaris security attributes to their entries:

- Group Manager
- Serial Port Manager
- Storage Manager

New – Profile Manager

The Profile Manager is new to Solstice_Apps and to Trusted Solaris software. It handles the execution profile [`tsolprof(4TSOL)`] database. See Table 1-4 on page 6 for the definition of execution profile. Trusted Solaris 2.5.1 software provides execution profiles for normal users and administrative roles. A security administrator can add to and modify the ones provided.

The administrator's trusted shell, called `tsh` in the Trusted Solaris 1.2 operating environment, is called `pfsh(1MTSOL)` (profile shell) in the Trusted Solaris 2.5.1 operating environment. It is assigned to a user or role as the login shell via the User Manager, and assigned as an assumable shell via an execution profile.

The security administrator uses the User Manager invoked from Solstice_Apps to assign execution profiles to users and roles.

New – User Manager

The User Manager divides a user account into six areas. User account information in the areas Identity and Home is set by the system administrator (`admin` role). Authorizations are also required to modify Identity and Home fields, and are provided to the `admin` role. Security information in the areas Labels, Profiles, Roles, and Idle is set by the security administrator (`secadmin` role). The required authorizations are provided to the `secadmin` role. As in Trusted Solaris 1.2, the security administrator cannot administer secadmin's security information, and the system administrator cannot modify sysadmin's Home or Identity information.

Modified – Database Manager

The Database Manager handles the regular NIS+ databases, such as `hosts` and `bootparams`, and the new network databases, `tnidb(4TSOL)`, `tnrhdb(4TSOL)` and `tnrhtp(4TSOL)`. The templates in `tnrhtp` simplify network administration by providing generic and per-domain host type descriptions.

Trusted Local Files Administration

The System_Admin folder in the Application Manager holds actions that affect the local workstation but are not handled by Solstice_Apps, such as editing the `vfstab(4TSOL)` and `vfstab_adjunct(4TSOL)` files. The Edit Encodings action enables the security administrator to edit and check the syntax of the `label_encodings(4TSOL)` file. Administrative files without a corresponding action, such as the `system(4)` file or a new file, are edited using the action Admin Editor. The Admin Editor action invokes a trusted editor, configurable by the security administrator to be `adminvi(1MTSOL)`, `dtpad(1)`, or other editors modified to be trustworthy.

2

Labels

As in the Trusted Solaris 1.2 environment, it is best to run all workstations with the same `label_encodings(4TSOL)` file. Labels are configurable, and label visibility can be set on a per workstation or per user basis, as discussed in "Window and Icon Labeling" on page 19.

Label Configuration

The installation program on the Trusted Solaris CD-ROM prompts for the site's label configuration. The choices are:

- to have a single sensitivity label, often called a "system high" configuration, or to have multiple sensitivity labels (also known as "multilevel")
- if multilevel, whether to hide upgraded names, that is, to hide file names whose sensitivity label is higher than the sensitivity label of its directory
- to enable information labels (sometimes called advisory labels)
- if enabled, whether to float information labels, that is, add the information label of a process to the information label of the object it is affecting
- if enabled, whether to reset the information label to `admin_low` when starting a process (upon a call to `exec`)

The installation program places the label configuration information in the `/etc/system` file. In Trusted Solaris 1.2, configuration choices were put in the `/etc/rc.conf` file.

Changes from Trusted Solaris 1.2 Labels

The administrative labels called `SYSTEM_HIGH` and `SYSTEM_LOW` in Trusted Solaris 1.2 are now called `ADMIN_HIGH` and `ADMIN_LOW`. The security administrator can assign them different names.

The "Bound Translation By *" options in the label encodings file have been removed. Label translations are always bounded by the sensitivity label; they cannot be bounded by clearance.

Common Desktop Environment

The Trusted Common Desktop Environment (CDE) is an enhanced version of CDE 1.1. Its window manager, `dtwm`, is the required desktop window manager for Trusted Solaris 2.5.1. Trusted Solaris 1.2 used OpenWindows 3.0 for the desktop; the OpenLook Window Manager (olwm) is no longer supported.

Trusted CDE supports a multilevel front panel, a trusted path menu, multiple workspaces for users and roles, window and icon labeling, a trusted stripe, trusted front panel applications and other desktop applications, multilevel ToolTalk and window properties, and trusted drag 'n drop. Access to commands, CDE actions, and applications are restricted by execution profiles.

See "User's Workspace" on page 33 for information on using the multilevel front panel, the trusted path menu, multiple workspaces for users and roles, and trusted front panel applications. Administrative issues for CDE are covered below. Help on Trusted Solaris modifications of CDE is available in the CDE online help.

Defaults

The default setting in `/etc/dt/config/sys.dtprofile` has been changed so that `.login` or `.profile` *are* sourced for users or roles whose default shell is the profile shell (`pfsh`). By default, the "Introducing the Desktop" online help does not appear at first login.

Window and Icon Labeling

If sensitivity labels and information labels were enabled during label configuration, they do not have to be visible as window and icon labels. The security administrator can hide either or both labels, for windows or icons or both. When all labels are hidden, the trusted screen stripe is iconified.

Label visibility is set for each user in the User Manager. Labels that are not enabled in the `system(4)` file cannot be made visible in the User Manager.

Administrative labels (`admin_high` and `admin_low`) can be displayed as the highest and lowest user labels, respectively. Called "label view", this is set for each workstation in the `label_encodings(4TSOL)` file and individually for each user in the User Manager.

Trusted Screen Stripe

In Trusted Solaris 2.5.1, the trusted screen stripe can be fully visible or iconified. When a user's labels are hidden, the screen stripe is iconified. When labels are visible, the screen stripe is visible.

Desktop Applications

CDE and OpenWindows applications should run with little modification. Possible modifications applications might need would be: multilevel directory for program files, polyinstantiation property in `inetd.conf`, and root permission during installation.

In CDE, actions can be dropped directly on the Front Panel. The Trusted Solaris 2.5.1 environment provides two modifications. First, actions that are not in a user's profile cannot be added to the Front Panel. Second, only applications in the directory `/usr/dt/~` or `/etc/dt/~` can be added to the Front Panel. Applications in the `~/.dt/appconfig` directories cannot be added to the Front Panel. Actions in the System_Admin and Solstice_Apps folders can be dropped onto the Front Panel by an administrative role; however, they will not work for a normal user.

Tooltalk

Tooltalk is multilevel. A separate Tooltalk session exists for each label and user ID pair. The File Manager, Application Manager, Calendar Manager, and mail notification have been extended in the Trusted Solaris 2.5.1 environment.

Trusted Drag 'n Drop

A user can have windows with different sensitivity labels in a workspace. Inter-window data moves between windows with different sensitivity labels must be authorized.

Cut-and-paste movement of data between windows of different sensitivity labels is restricted to authorized users, and a mediation window pops up to confirm upgrades and downgrades of data. Drag 'n drop movement between applications is constrained to the applications having the same sensitivity label and user ID. However, the front panel icons and File Manager handle multilevel data drag 'n drop as follows:

- Front panel icons – The individual icons in the front panel act as multilevel drop sites for drag-and-drop movements of data. For example, data dropped onto the trash can icon is delivered to the appropriate trash folder based on the sensitivity label and user ID of the source data.
- File Manager – The File Manager allows authorized users to relabel files via drag'n drop, and to add privileges to executable files. It also properly handles multilevel directories.

Windows

In the Trusted Solaris 2.5.1 operating environment, window system objects (window properties, selections, and tooltalk sessions) are instantiated at more than one sensitivity label (that is, they are multilevel), and access is segregated by their sensitivity label. For example, a property on the root window can exist at two different sensitivity labels as separate and distinct objects. Only windows with the same sensitivity label are available to clients to avoid sensitivity label collisions when using selections to transfer data from window to another window. Polyinstantiation is configurable by modifying the entries in files in the directory `/usr/openwin/server/tsol`.

Window Policy

X Window server policy is configurable by modifying the file `/usr/openwin/server/tsol/config.privs`.

Network

The Trusted Solaris 2.5.1 networking subsystem consists of a set of extensions to the Solaris 2.5.1 TCP/IP network. The network supports communications with other labeled and unlabeled hosts, multilevel ports, security attributes on the network interface, and centralized network administration using the enhanced Database Manager in Solstice_Apps. Network database formats have changed from Trusted Solaris 1.2 formats.

Host Types

Trusted Solaris 2.5.1 recognizes hosts of the following types on a distributed, heterogenous network:

- *Unlabeled* – Data can be received from and sent to workstations running operating systems that do not recognize labels.
- *Trusted Solaris 1.2 (MaxSix 1.0+ Protocol)* – Trusted Solaris 2.5.1 supports network connectivity with Trusted Solaris 1.2 workstations.

  The tsol protocol uses the MaxSix 1.0 protocol to enable Trusted Solaris 2.5.1 applications to communicate with Trusted Solaris 1.2 applications without change to the application.
- *Sun_tsol* – The protocol used between Trusted Solaris 2.5.1 hosts.

Note – In Trusted Solaris 1.2, the default set of supported security attributes is larger. Trusted Solaris 2.5.1 hosts using the `sun_tsol` type do not transmit supplementary groups or audit information in the network packet. Both are available through an out-of-band protocol when requested by the TSIG library interfaces.

- *CIPSO* – The Common Internet Protocol Security Option (CIPSO) specifies security labels that are passed in the internet protocol (IP) options field.
- *TSIX(RE) 1.1. – SATMP* – The Security Attribute Token Mapping protocol (SATMP) is used by trusted systems to exchange attribute-token mappings.
- *TSIX(RE) 1.1. – SAMP* – The Security Attribute Modulation protocol (SAMP) passes tokenized security attributes associated with a sending process on every write operation made to the senders's transport layer protocol.
- *RIPSO* – The Revised Internet Protocol Security Option (RIPSO) is described in the IETF RFC 1108. It specifies a Department of Defense (DoD) IP labeling method to incorporate labels into IP packets.

Multilevel Ports - New

In the Trusted Solaris 2.5.1 operating environment, all network endpoints are either multilevel or single-level ports to support unmodified applications in a trusted environment. The Trusted Solaris 1.2 environment supported only single-level.

- Multilevel port – A network endpoint associated with all sensitivity labels.
- Single-level port – A network endpoint associated with one sensitivity label.

A multilevel port can receive data with any sensitivity label, while a single-level port can receive only data with the matching label. A network packet carrying a particular label is delivered to a matching single-level port first. If no such single-level port exists, it is then delivered to a matching multilevel port. If no such multilevel port exists, the packet is dropped.

Network Interfaces and Routing

The network interface layer accepts packets from the internet layer and places them on the appropriate physical network. As in Trusted Solaris 1.2, Trusted Solaris 2.5.1 provides static routing. As in Solaris 2.5.1, Trusted Solaris 2.5.1 provides dynamic routing.

Table 2-2 Routing File Changes

Trusted Solaris 1.2 Routing	**Solaris 2.5.1 Routing**	**Trusted Solaris 2.5.1 Routing**
`/etc/m6.routes`	`/etc/defaultrouter`	`/etc/defaultrouter` `/etc/tsolgateways`
	dynamic routing	`/etc/tunnel`

Network Interface Database

The network interface database, `tnidb(4TSOL)`, is stored in a different location and has a different format from the Trusted Solaris 1.2 `tnidb`. Data must be manually ported.

- `/etc/security/tsol/tnidb` – Network interface database. It is always a local file, and is administered through the Database Manager, as described in “Administrative Graphic User Interfaces” on page 15. A `tnidb` entry describes each network interface in terms of the following:
  - Host type
  - Label range for incoming and outgoing data
  - For incoming unlabeled data, if unspecified in the remote host database: CMW label, clearance, effective user and group ID, effective privileges

Remote Host Databases

Network database formats have changed and must be ported manually. The remote host databases are administered through the Database Manager, as described in "Administrative Graphic User Interfaces" on page 15. The Trusted Solaris 2.5.1 network uses the following files or NIS+ tables:

- `/etc/security/tsol/tnrhdb` – Remote Host Database that maps known hosts to template names. Like other NIS+ files, this database can be a local file or a NIS+ table. It specifies IP address and template (`tnrhtp`) name.
- `/etc/security/tsol/tnrhtp` – Remote Host Template Database. Like other NIS+ files, this database can be a local file or a NIS+ table. It describes templates for use by `tnrhdb`. There are six basic templates based on host type:
  - `sun_tsol` type for communicating with other Trusted Solaris 2.5.1 hosts
  - `unlabeled` type for communicating with unlabeled workstations, such as Solaris 2.5.1 hosts
  - `tsix` type for communicating with workstations whose vendors are using TSIX
  - `msix` type for communicating with Trusted Solaris 1.2 hosts
  - `cipso` type for communicating with hosts sending CIPSO packets
  - `ripso` type for communicating with hosts sending RIPSO packets

  Each host type can be further characterized by attributes specific to the host type. For example, the `sun_tsol` host type has an effective privileges attribute while the `unlabeled` host type does not.

Network Utility Commands

The Trusted Network uses the following utilities:

- `tnchkdb(1MTSOL)` – Checks database file syntax.
- `tnctl(1MTSOL)` – Controls single element of trusted network, sets daemon debugging, and loads a single interface, host, or template.
- `tnd(1MTSOL)` – Initializes trusted networking and loads databases, loads all the databases into the Trusted Solaris 2.5.1 kernel for their use, and manages the kernel database tables.
- `tninfo(1MTSOL)` – Displays information about trusted networking interface, host, template, information, and trusted network statistics.

New snoop Command

The `snoop(1MTSOL)` command displays network packets based on specified options and filters. It can display different levels of detail based on options specified.

In Trusted Solaris 2.5.1, three types of packets are recognized: `unlabeled`, `tsol`, and `tsix`. A new filtering primitive, `sectype`, is added to filter packets based on security type. The syntax is

```
sectype type
```

where *type* is `unlabeled`, `tsol`, or `tsix`. For Trusted Solaris (`tsol`) packets, the security attributes are displayed.

Distributed Services

A distributed service refers to a component of the system that can perform services for clients on different machines (or same machine) as the service provider. Trusted Solaris 2.5.1 supports most of the UDP, TCP, and TI-RPC services provided in Solaris 2.5.1. These services include the following:

- Various Internet services coordinated by `inetd`, including `rlogin`, `ftp`, `telnet`, `rsh`, `rcp` and others.
- The NIS+ database is now used instead of the NIS database for distributed network administration. There is now a template file for describing remote hosts.
- Domain Name Service (DNS).

Access Control Lists

Access Control Lists (ACLs) are part of the Solaris 2.5.1 operating environment. The Trusted Solaris 2.5.1 operating environment supports the same format and interfaces for ACLs that are found in the Solaris 2.5.1 operating environment, and extends the support for ACLs to include the `tmpfs` file system.

2

File Systems

The Solaris 2.5.1 file system architecture has a mechanism, the shadow inode, to store extended attributes such as ACLs. Trusted Solaris 2.5.1 builds upon this architecture to store its extended security attributes. File systems now have Trusted Solaris 2.5.1 security attributes.

File systems support attributes for file system objects, as in Trusted Solaris 1.2, and for local file systems and mounted file systems. When a file system object is not given explicit security attribute values, it inherits the local file system attribute values. When the local file system does not have an attribute value, the mount attribute is used. File system security attributes can be set in the `/etc/security/tsol/vfstab_adjunct` file. Also, the `mount(1MTSOL)` command has been extended to include file system security attributes.

As in Trusted Solaris 1.2 software, file systems can be single-label or multilevel. A single-label file system supports only the standard Solaris 2.5.1 object attributes. The system administrator either sets the Trusted Solaris security attributes in the `vfstab_adjunct(4TSOL)` file, or mounts the single-label file system using the `mount(1MTSOL)` command. Otherwise, Trusted Solaris security attributes are absent altogether.

Unlike Trusted Solaris 1.2 file systems, Trusted Solaris 2.5.1 file systems have more security attributes, security attributes are stored filesystem-wide, and there are commands and files to handle filesystem-wide security for local and mounted file systems.

Note – Because Trusted Solaris 2.5.1 does not recognize Trusted Solaris security attributes on file systems mounted from NFS Version 2 servers, all such servers need security attributes specified in `/etc/security/vfstab_adjunct` or with the `mount(1MTSOL)` -S option. Also, specify vers=2 with the `mount(1MTSOL)` command or in the `vfstab(4TSOL)` file. The entry in `/etc/security/tsol/tnrhdb` for the mounted server should be unlabeled.

The Trusted Solaris 2.5.1 operating environment supports Solaris 2.5.1 file system security attributes in addition to its own, as shown in the following table.

Table 2-3 File System Attributes in Trusted Solaris 2.5.1

Solaris 2.5.1 Attributes	**and Trusted Solaris 2.5.1 Attributes**
Access Control Lists	CMW label
Discretionary Access Control permission bits	File system label range
User and Group ID	Forced and allowed privilege sets
	Audit preselection attributes
	Multilevel directory (MLD) prefix

File System Attribute Commands and Files

Trusted Solaris 2.5.1 commands that handle file system security attributes affect all security attributes, including those defined in the Solaris 2.5.1 operating environment. Trusted Solaris 2.5.1 files that handle file system security attributes affect only Trusted Solaris 2.5.1 security attributes.

Trusted Solaris 2.5.1 file system attributes can be applied when the file system is mounted through the new `vfstab_adjunct(4TSOL)` file. The `vfstab(4TSOL)` file handles Solaris 2.5.1 security attributes (such as ACLs), There are also options to the `mount(1MTSOL)` command that apply Trusted Solaris file system security attribute values for the duration of the mount operation.

New - File System Object Attributes

File system objects have flags that indicate whether the object uses multilevel directory semantics and whether the object is public.

New - File Manager Sets Security Attributes

An administrator can set a file's security attributes in the File Manager.

The following file system commands and files are either new or modified for Trusted Solaris 2.5.1.

Table 2-4 File System Command and File Changes

Operation	Interfaces	Origin/Change
Get and set file security attribute flags for individual files	`getfattrflag(1TSOL)`, `setfattrflag(1TSOL)`	New in 2.5
Get and set file system security attributes	`getfsattr(1MTSOL)`, `setfsattr(1MTSOL)`, `getfsattr_ufs(1MTSOL)`	New in 2.5
Get and set file privilege sets	`getfpriv(1TSOL)`, `setfpriv(1TSOL)`	Replace 1.2 `lspriv` and `chpriv` commands
Test file privilege sets	`testfpriv(1TSOL)`	New in 2.5
Get multilevel directory adornment	`adornfc(1TSOL)`, `getmldadorn(1TSOL)`, `mldpwd(1TSOL)`, `mldrealpath(1TSOL)`	Modified 1.2 (parameter list change)
Mount filesystem	`/etc/security/tsol/vfstab_adjunct` [`vfstab_adjunct(4TSOL)`]	New in 2.5; replaces 1.2 `fstab.adjunct` file
Mount filesystem	`mount(1MTSOL)`, `mountd(1MTSOL)`	Modified Solaris 2.5.1 (more parameters)
Share and unshare filesystem	`share(1MTSOL)`, `share_nfs(1MTSOL)`, `unshare(1MTSOL)`, `unshare_nfs(1MTSOL)`, `shareall(1MTSOL)`, `unshareall(1MTSOL)`	Modified Solaris 2.5.1
Display filesystem information	`dfmounts(1MTSOL)`, `dfshares(1MTSOL)`, `showmount(1MTSOL)`, `sharetab(4TSOL)`, `rmtab(4TSOL)`	Modified Solaris 2.5.1

Multilabel File Systems

The following file system types support multiple labels:

- `ufs` – Standard Solaris UNIX file system with Trusted Solaris attributes
- `tnfs` – Remote file system protocol for UNIX file system access with Trusted Solaris attributes (based on networked file system (NFS) V3 and supporting V3 over TCP).
- `tmpfs` – In-memory temporary file system
- `swapfs` – Swap file system

- `lofs` – Loopback file system
- `autofs` – Automount file system

To allow easy migration of UNIX file system (UFS) disks between Solaris 2.5.1 and Trusted Solaris 2.5.1, there are two ways to configure a Solaris 2.5.1 UFS disk mounted on a Trusted Solaris 2.5.1 workstation.

1. Set up security attribute defaults and store security attributes on the file system. See the `setfsattr(1MTSOL)` man page.
2. Set up security attribute defaults for the file system at mount time. This enables the UFS disk to be reintegrated on a Solaris 2.5.1 workstation later. See the `mount(1MTSOL)` and the `vfstab_adjunct(4TSOL)` man pages.

Multilevel Directories

Multilevel directories (MLDs) allow data to be stored at different sensitivity labels within the same directory. Multilevel directories are supported only by `ufs` with Trusted Solaris 2.5.1 attributes, `tnfs`, `autofs`, `lofs`, and `tmpfs` file system types. As in Trusted Solaris 1.2, MLDs appear in the file system as ordinary directories with a Trusted Solaris 2.5.1 attribute identifying them as MLDs.

Users' home directories are automatically created as multilevel directories in the parent directory, such as `/export/home` or `/home`. In the Trusted Solaris 1.2 operating environment, the `/tmp` directory was created as a multilevel directory. The Window Manager and CDE File Manager are aware of MLDs as described in "Trusted Drag 'n Drop" on page 20.

In Trusted Solaris 1.2, single-level directory (SLD) names were the hexadecimal equivalent of the label names. In Trusted Solaris 2.5.1, SLD names are sequentially assigned within each MLD as they are created. The first SLD created in an MLD is numbered 0, the second is numbered 1, and so forth. The numbers have no relation to the label of the directory, therefore the label is truly hidden.

2

Devices

As in Trusted Solaris 1.2 software, devices have a label range and require authorization to operate. The Trusted Solaris 1.2 GUI, `alloctool`(1T), is replaced by the Device Manager, available from the subpanel (its title is Trusted Desktop) of the Style Manager.

Unlike Solaris 2.5.1 device allocation, Trusted Solaris device allocation is not affected by the `bsmconv(1MTSOL)` and `bsmunconv(1MTSOL)` scripts. Device allocation is configurable by the appropriate administrator. The following table lists the administrative commands for device allocation.

Table 2-5 Device Command and File Changes

Operation	Interfaces	Origin/Change
Inform system of new device driver, remove driver	`add_drv(1MTSOL)`, `rem_drv(1MTSOL)`	Solaris 2.5
Allocate and deallocate device	`allocate(1MTSOL)`, `deallocate(1MTSOL)`	Solaris 2.5, TS1.2
Configure/query STREAMS modules pushed when device opened	`autopush(1MTSOL)`	Solaris 2.5
Add and remove local devices as allocatable, including printers	`add_allocatable(1MTSOL)`, `remove_allocatable(1MTSOL)`	New in 2.5.1
Create and query configurable device policy table in kernel	`devpolicy(1MTSOL)`	New in 2.5
Store device configuration information	`/etc/security/tsol/device_policy` `device_policy(4TSOL)`	New in 2.5
Device clean scripts	`device_clean(1MTSOL)`	Solaris 2.5, TS1.2
List allocatable devices	`list_devices(1MTSOL)`	Solaris 2.5, TS1.2
Get device information from device_maps file	`dminfo(1MTSOL)`	Solaris 2.5, TS1.2
Configure /dev directory	`drvconfig(1MTSOL)`	Solaris 2.5

Auditing

Auditing in the Trusted Solaris 2.5.1 environment is similar in concept to Trusted Solaris 1.2 auditing. It extends the Solaris 2.5.1 Basic Security Module (BSM) to include Trusted Solaris audit tokens, audit events, audit classes, and audit policy flags. Auditing is turned on by default in the Trusted Solaris 2.5.1 environment. It can be disabled by commenting out the audit daemon in the `/etc/init.d/audit` script, which is called by the `/etc/rc2.d` run level script, `S99audit`,

The following table shows the auditing compatibility between Trusted Solaris 2.5.1, Solaris 2.5.1, and Trusted Solaris 1.2 versions. A Trusted Solaris 2.5.1 audit server can read Solaris 2.5.1 audit records, but cannot read Trusted Solaris 1.2 audit records.

Table 2-6 Interoperability Between Audit Trail Formats

		Audit Record Format		
		TS 1.X	**Solaris 2.5.1**	**TS 2.5**
	TS 2.5	N	Y	Y
Audit Servers	**Solaris 2.5**	N	Y	N
	TS 1.X	Y	N	N

The next table summarizes the changes in audit commands from the Solaris 2.5.1 implementation. The security policy is similar to Trusted Solaris 1.2 policy.

Table 2-7 Modified from Base Solaris

Modifications	**Audit Command**
The calling process needs `sys_audit` to query the state of the system.	`audit(1MTSOL)`
	`audit_warn(1MTSOL)`
	`audit_startup(1MTSOL)`
	`auditd(1MTSOL)`
Policy flags added	`auditconfig(1MTSOL)`
	`auditstat(1MTSOL)`

Table 2-7 Modified from Base Solaris

Modifications	Audit Command

The calling process needs `sys_audit` to get or set the operational state of the audit module or the audit behavior of any audited process.
Auditing is enabled by default. Devices have been removed from `bsmconv` scripts.
	`auditreduce(1MTSOL)`
	`praudit(1MTSOL)`
	`bsmconv(1MTSOL)`
	`bsmunconv(1MTSOL)`

The following table summarizes the changes in audit files from the Solaris 2.5.1 BSM and from Trusted Solaris 1.2 auditing.

Table 2-8 Changes in Audit Files from BSM and Trusted Solaris 1.2 Auditing

Modifications	Audit File

Calling process needs `sys_audit`, `proc_audit_tcb`, or `proc_audit_appl` to read the files.	`audit_control(4TSOL)`
	`audit_data(4TSOL)`
	`audit_user(4TSOL)`
	`audit.log(4TSOL)`

audit classes and umasks changed	`audit_class(4TSOL)`
Trusted Solaris events added	`audit_event(4TSOL)`
Use `audit_user(4TSOL)` files	`passwd.adjunct(5)`
Enable auditing - not bsmconv(4)	`/etc/init.d/audit`
Disable auditing - not bsmunconv(4)	`/etc/init.d/audit`

By default, a regular user is not assigned a profile that enables getting or setting audit parameters. The administrative role `secadmin` has the Audit Control profile, whose commands inherit the `sys_audit` privilege for audit operations. The administrative role `admin` has the Audit Review profile, whose commands `auditreduce(1MTSOL)` and `praudit(1MTSOL)` enable audit trail analysis.

Printing

The Solstice AdminSuite Printer Manager in Trusted Solaris 2.5.1 can administer all print domains. Note that this is *not* the Printer Manager that appears by default on the Front Panel. The Printer Manager is available to the security administrator; it is in the Solstice_Apps folder.

The Trusted Solaris 2.5.1 operating environment labels printed output to meet the needs of sites where sensitivity information requires special handling. A banner (first) and trailer (last) page are introduced to surround the printed output and provide information on the sensitivity label, information label, and handling caveats of the printed output and job ID. Information labels (ILs) are also printed at the top and bottom of each printed page. While an authorized user can turn off IL printing, that event is auditable and not recommended.

The Solaris 2.5.1 printer output system has been modified in Trusted Solaris 2.5.1 to provide the printed output labeling and segregate the print queues by sensitivity label. This functionality provides the ability to print to any PostScript printer. However, the user must have the authorization TSOL_AUTH_PRINT_POSTSCRIPT, because the labeling mechanism for PostScript files lacks the integrity of the plain text file labeling mechanism.

A Solaris 2.5.1 workstation can be used as an unlabeled print server for Trusted Solaris 2.5.1 print jobs, but not the other way around.

Mail

Trusted Solaris 2.5.1 adds several privacy options to `sendmail(1MTSOL)` for security. Unlike Trusted Solaris 1.2, roles do not need to be added to an aliases file because roles are users, so at creation get a home directory and mail like other users.

The default multilevel mail utility is `dtmail(1x)`. To use another mailer, such as the OpenWindows `mailtool`, see the *Trusted Solaris Administrator's Procedures* manual.

User's Workspace

The Trusted Solaris 2.5.1 operating environment requires the use of the Common Desktop Environment (CDE). Trusted CDE supports a multilevel front panel, a trusted path menu, multiple workspaces for users and roles,

window and icon labeling, a trusted stripe, trusted front panel applications, other desktop applications, and trusted drag 'n drop. Access to commands, CDE actions, and applications are restricted by execution profiles.

Front Panel

In the Trusted Solaris 2.5.1 CDE, the Front Panel is a special multilevel window that provides mechanisms for starting applications at different labels. It contains a trusted File Manager, multilevel mail, a trusted editor, and a trusted Application Manager, plus other applications. Applications launched from the front panel are labeled with the current workspace's label. A normal user does not have the authorization to change the contents of the front panel, or to place privileged applications in the Application Manager.

Trusted Path Menu

The Switch Area menu of the front panel in base CDE has been replaced by the Trusted Path (TP) menu in Trusted Solaris 2.5.1. Right-clicking in the central area of the front panel brings up the menu, which contains menu items to assume a role, change passwords, change the sensitivity label of the workspace, and other actions that affect the security of the workstation.

Multiple Workspaces

Trusted CDE supports multiple workspaces. By default, each user gets four workspaces named One, Two, Three, and Four. Upon login, each workspace is labeled with the user's minimum sensitivity label. Applications launched from the front panel are started at this label. In a multilevel session, the sensitivity label of a workspace can be changed from the Trusted Path menu. As in base CDE, Trusted CDE user workspaces can be renamed, customized, deleted, and saved.

Role Workspaces

The Trusted Path menu also provides the mechanism for assuming roles. The menu presents a list of roles available to the user. If there is a list, the user selects a role. Once the role password is authenticated, a new workspace is created for that role with the role's environment, minimum sensitivity label, user and group IDs, and environment variables. As with other workspaces,

CDE actions taken in a role workspace are restricted to those defined in the execution profiles assigned to the role. User workspace windows (untrusted client windows) cannot occupy a workspace of an administrative role. And unlike user workspaces, role workspaces are not saved in their entirety. Role applications are not saved, though workspace appearance, set through the Style Manager, can be saved.

Window and Icon Labeling

In the Trusted Solaris 2.5.1 environment, window and icon labels can be hidden.

Trusted Screen Stripe

In the Trusted Solaris 2.5.1 environment, when labels are hidden, the screen stripe is iconified. When labels are visible, the screen stripe is visible.

Trusted Front Panel Actions

The Application Manager and File Manager, desktop applications invoked from the front panel, have been modified to be trusted. They work seamlessly at multiple labels in one user session, as do most applications, such as multilevel mail.

- The Application Manager and File Manager have been modified to consult the `tsolprof(4TSOL)` database before presenting or invoking actions. A user can invoke only those CDE actions defined in the profiles assigned to the user or role that the user has assumed. The restrictive and enabling attributes associated with profiles are also applied to the invocation of CDE actions.

  The folders Solstice_Apps, for administering the distributed system, and System_Admin, for administering the local workstation, have been added to the Application Manager. A trusted editor is available from the action Admin Editor in the System_Admin folder. See “Trusted Solstice AdminSuite Administration” on page 15 and “Trusted Local Files Administration” on page 17 for details.

- The File Manager provides interfaces for getting and setting Trusted Solaris file attributes such as labels and privileges. Users in an administrative role with the appropriate authorizations can use the interfaces. It has also been made aware of multilevel directories; all users can use the multilevel directories feature.
- The mail icon for `dtmail(1X)` in the front panel has been modified to support multilevel mail. It displays the arrival of mail for users and active roles at each sensitivity label. The mailboxes are accessible from a subpanel menu above the mail icon. A mail reader corresponding to the sensitivity label of the incoming mail can be launched by clicking on the icon corresponding to that label.
- The Calendar Manager has been modified to support multilevel operation.
- The Style Manager control now has a Trusted Desktop subpanel. The Device Allocation Manager, new to Trusted Solaris, and the Desktop Access tool for users with disabilities, are on the subpanel.

Desktop Applications

CDE and OpenWindows applications should run with little modification.

Trusted Drag ‘n Drop

A user can have windows with different sensitivity labels in a workspace. Inter-window data moves between windows with different sensitivity labels must be authorized.

Cut-and-paste movement of data between windows of different sensitivity labels is restricted to authorized users, and a mediation window pops up to confirm upgrades and downgrades of data.

The front panel icons and File Manager handle multilevel data drag ’n drop transparently. However, drag ’n drop movement between applications is constrained to the applications having the same sensitivity label and user ID.

Interoperation

The Trusted Solaris 2.5.1 environment interoperates with other network protocols, file systems, and Trusted Solaris 1.2 windows and OpenWindows applications.

Network Protocols

Network connectivity between Solaris 2.5.1 workstations and Trusted Solaris 2.5.1 workstations by way of TCP, UDP, and IP is supported at a single sensitivity label. Various Internet Services available in the Solaris 2.5.1 operating environment, such as `rsh`, `rcp`, `rlogin`, `telnet`, and `ftp` are supported at a single sensitivity label in either direction between Trusted Solaris 2.5.1 and Solaris 2.5.1 workstations.

The Trusted Solaris 2.5.1 networking protocol is backwards compatible with the Trusted Solaris 1.2 networking protocol. Therefore, nonprivileged multilevel network interoperability between the two environments is supported.

- Network file system – NFS operation is supported between Trusted Solaris 2.5.1 and Trusted Solaris 1.2 workstations at a single sensitivity label.
- Internet services – Internet services including `rlogin`, `telnet`, and `ftp` are supported between Trusted Solaris 2.5.1 and Trusted Solaris 1.2 workstations at a single sensitivity label.
- Diskless Boot – Diskless boot is not supported between a Trusted Solaris 2.5.1 workstation and a Trusted Solaris 1.2 or Solaris 2.5.1 workstation.

Note – To communicate with Trusted Solaris 1.2 workstations, `root` on the Trusted Solaris 2.5.1 workstation must be a member of the group `wheel`. `wheel` in Trusted Solaris 1.2 is GID 0, but `root` group in Trusted Solaris 2.5.1 is GID 0, so give Trusted Solaris 2.5.1 `wheel` a different GID.

Privilege and Authorization Interoperability

The mapping between Trusted Solaris 1.2 authorization names and authorization IDs, and privilege names and privilege IDs have not been preserved. For this reason, the Trusted Solaris 2.5.1 operating environment supports only nonprivileged interoperation with Trusted Solaris 1.2 hosts where no authorization checking takes place.

File System Interoperability

The Trusted Solaris 2.5.1 operating environment supports most Solaris 2.5.1 file systems and maintains the file system interfaces. Solaris 2.5.1 file system objects are also supported.

File system mounting between Trusted Solaris 2.5.1 hosts and Solaris 2.5.1 hosts at a single label is supported.

For each file system type, Trusted Solaris 2.5.1 security attributes are specified when the file system is mounted. A single-label file system supports objects only at the label at which the file system was mounted.

- `ufs` – Standard Solaris UNIX file system without Trusted Solaris attributes
- `nfs` – Remote file system protocol (V2, V3, and V3 over TCP).
- `pcfs` – MS-DOS formatted file system
- `hsfs` – High Sierra file system for CD-ROM drives

NFS client and server functionality at a single sensitivity label is supported in either direction between Solaris 2.5.1 and Trusted Solaris 2.5.1 workstations. This includes secure NFS as well as NFS Version 3 (V3), V3 over TCP, and Version 2.

Window Interoperability

Windows can be remotely displayed between Solaris 2.5.1 and Trusted Solaris 2.5.1 workstations at a single sensitivity label. Authentication can occur based on host, secure remote procedure call (RPC) credential, or the default user ID set up for that workstation.

Nonprivileged, multilevel window interoperability is also supported between Trusted Solaris 2.5.1 and Trusted Solaris 1.2 workstations. (See “Privilege and Authorization Interoperability” on page 37 for why privileges are not compatible).

Data Interchange

The `tar(1TSOL)` command with the `-T` option (to preserve security attributes) is the recommended method for porting data from a Trusted Solaris 1.2 workstation to a Trusted Solaris 2.5.1 workstation. Files can be ported along with their attribute information, excluding privileges (see “Privilege and Authorization Interoperability” on page 37).

Dumping and restoring filesystems [`ufsdump(1M)`, `ufsrestore(1M)`] between Trusted Solaris 2.5.1 and other operating environments is not supported.

Note – The `tar(1TSOL)` command is limited by directory depth: the directory name cannot be longer than 155 characters, and the filename cannot be longer than 100 characters.

The following table shows the effects of commands for transfering files to a Trusted Solaris 2.5.1 workstation.

Table 2-9 Moving Data to a Trusted Solaris 2.5.1 Workstation

Moving Files from ...	Using command ...	Results in ...
Trusted Solaris 2.5.1 Workstation	`tar(1TSOL)` cT	files with tsol attributes
	`ufsdump(1M)`	files with all security attributes, when restored using `ufsdump(1M)`
	`cpio(1)` `pax(1) -x`	files with no security attributes
Solaris 2.5.1 Workstation	`tar(1) cp`	files with ACLs, without other security attributes

	`tar(1) c`
	`cpio(1)`
	`pax(1) -x`
		files with no security attributes

Trusted Solaris 1.2 Workstation	`tar(1) cs` `tar(1)` csa	files with security attributes minus privilege attributes

	`tar(1) c`
	`cpio(1)`
	`pax(1) -x`
		files with no security attributes

The following table shows the commands available in the Trusted Solaris 2.5.1 environment, and their effects when transfering Trusted Solaris 2.5.1 files to other operating environments.

Table 2-10 Moving Data from a Trusted Solaris 2.5.1 Workstation

Moving Files to ...	Using command ...	Results in ...
Trusted Solaris 2.5.1 Workstation	`tar(1TSOL) cT`	files with all security attributes
	`ufsdump(1M)`	files with all security attributes, when restored using `ufsrestore(1M)`
	`cpio(1)` `pax(1) -x`	files with no security attributes
Solaris 2.5.1 Workstation	`tar(1TSOL) cp`	files with ACLs
	`tar(1TSOL) c`	files with no security attributes
	`cpio(1)` `pax(1) -x`	files with no security attributes
Trusted Solaris 1.2 Workstation	`tar(1TSOL)` c `tar(1TSOL)` cT	unpredictable
	`cpio(1)` `pax(1) -x`	files with no security attributes

Porting Applications

CDE applications are based on Motif 1.2.4. However, the Trusted Solaris 2.5.1 operating environment supports and allows interoperability with OpenWindows applications based on the XView and OpenLook Interface Toolkit (OLIT) toolkits. The current version of these toolkits is 3.5.1.

Most third-party applications will run in the Trusted Solaris 2.5.1 environment without modification. In the Trusted Solaris 2.5.1 environment, multiple instances of third-party applications run at different sensitivity labels without privileges or errors. Motif-based applications should run with little modification.

Trusted Solaris 1.2 trusted applications must be re-coded for privilege and authorization changes in Trusted Solaris 2.5.1. It may be best to re-work the applications in Motif or as CDE actions.

Trusted Solaris 2.5.1 software packages now have Trusted Solaris security attributes as well as Solaris 2.5.1 attributes. The following security attributes can be set on files at package installation time:

- CMW labels
- ACLs
- Allowed and forced privileges
- Public object flag
- MLD flag

Privilege Debugging

Programmers and administrators can test applications that require privileges without using auditing to detect a program's use of privilege. The AUE_MAC audit event, used in Trusted Solaris 1.2 for debugging MAC use, is no longer supported. Running applications in privilege debugging mode logs privilege failures while allowing applications to succeed. The log lists the privileges the program needs for successful operation. A role with appropriate access turns on privilege debugging mode by:

1. Setting the `tsol_privs_debug` switch in the `/etc/system` file to the value `1`.
2. Uncommenting the `kern.debug` line in the `/etc/syslog.conf` file.
3. Rebooting, entering an administrative role workspace, and issuing the command `runpd(1MTSOL)` in a profile shell [`pfsh(1MTSOL)`] to start the program in debug mode.
   The `/var/log/privdebug.log` file logs the program's privilege requirements; there is also screen output.

New and Replaced Commands

The Trusted Solaris 2.5.1 operating environment has changed or removed commands available in the Trusted Solaris 1.2 releases, as well as added commands.

New User Commands

There are new user commands to do the following:

- Get and set file system attributes.
- Get and set file privilege sets.
- Get process attribute flags, process clearance, process CMW label, and process effective privilege set.
- Test file privilege sets and process effective privilege set.
- Obtain information on kernel level network information and statistics.

New System Administration Commands

There are new system administration commands to do the following:

- Start a restricted text editing environment.
- Start the profile and system shells.
- Address resolution display.
- Add and remove local devices, including printers.
- Convert ASCII labels to hexadecimal and back.
- Manage auditing, streams, and device allocation.
- Get and set file system and process attributes and attribute flags.
- Set security attributes on a new file system.
- Configure network interface parameters.
- Link files and directories, mount and unmount file systems, check and repair file systems, and make local resources available for remote filesystem mounting.
- Show network status, create NIS+ tables, initialize NIS+ domain, capture and inspect network packets, send one-way stream of packets to host, check syntax of trusted network database, configure network daemon control parameters, start the trusted network daemon, write an audit record, and handle token maps.
- Set echo requests, set system date from remote host, manipulate routing tables, run a command for privilege debugging, send mail over the internet,

Administrative and User Command Summary

Administrative commands are described in *Trusted Solaris Administrator's Procedures*, and user commands are described in the *Trusted Solaris User's Guide*. The following table lists the status of Trusted Solaris 1.2 administrative and user commands in the Trusted Solaris 2.5.1 release. Administrative and user commands new in the Trusted Solaris 2.5.1 release are listed in Table 2-12 on page 45.

Note – Auditing and ACLs have been modified for the Trusted Solaris 2.5.1 release based on Solaris 2.5.1 source code. Therefore, commands in these areas are likely to be different from their Trusted Solaris 1.2 versions.

Table 2-11 Trusted Solaris 2.5.1 Equivalents of Trusted Solaris 1.2 Commands

Trusted Solaris 1.2 User Command	Trusted Solaris 2.5.1 Equivalent
`acledit(1T)`	Trusted File Manager interface.
`add_ctab_obj(8T)`	Removed.
`adornfc(1T)`	`adornfc(1TSOL)`
`allocate(1T)`	`allocate(1MTSOL)`
`ccheck(8T)`	Removed.
`chk_encodings(8T)`	`chk_encodings(1MTSOL)`
`chpriv(8T)`	`setfpriv(1TSOL)` and File Manager
`dbck(8T)`	`tnchkdb(1MTSOL)`
`deallocate(1T)`	`deallocate(1MTSOL)`
`del_ctab_obj(8T)`	Removed.
`dminfo(1T)`	`dminfo(1MTSOL)`
`getfacl(1T)`	`getfacl(1)`
`getlabel(1T)`	`getlabel(1TSOL)` -h option gets label of a symbolic link
`getmldadorn(1T)`	`getmldadorn(1TSOL)`
`getplabel(1T)`	`plabel(1TSOL)`
`getppriv(1T)`	`ppriv(1TSOL)`
`getsldname(1T)`	`getsldname(1TSOL)` -s option gets label of SLD

Table 2-11 Trusted Solaris 2.5.1 Equivalents of Trusted Solaris 1.2 Commands

Trusted Solaris 1.2 User Command	Trusted Solaris 2.5.1 Equivalent
`install_ctabs(8T)`	Removed.
`labeld(8T)`	Removed.
`list_devices(1T)`	`list_devices(1MTSOL)`
`lspriv(1T)`	`getfpriv(1TSOL)`
`mkctab(8T)`	Removed.
`mkdb(8T)`	Removed.
`mldpwd(1T)`	`mldpwd(1TSOL)`
`mldrealpath(1T)`	`mldrealpath(1TSOL)` -s option gets label of SLD
`newfs(8)`	`newsecfs(1MTSOL)`
`peeraudit(8C)`	`rpc.getpeerinfod(1MTSOL)`
`privenable(8T)`	No longer documented.
`postmaster(1T)`	Removed.
`rpc.labeld(8T)`	Removed.
`rtp(8T)`	Removed.
`secuname(1T)`	`uname(1TSOL)`
`setfacl(1T)`	`setfacl(1)`
`setlabel(1T)`	`setlabel(1TSOL)`; -h option sets label of a symbolic link
`setlow(8T)`	`setlabel(1TSOL)` “admin_low[admin_low]” *file*
`sireport(8T)`	Removed.
`sync_ctab(8T)`	Removed.
`systsh` (trusted system shell)	`sysh(1MTSOL)`
`tar(1T) -s option`	`tar(1TSOL)` -T option
`tcb_verify(8T)`	Removed.
`tfmedit, tfmvi(1T)`	`adminvi(1MTSOL)`
`tnet_kstats(8T)`	`tninfo(1MTSOL)`

Table 2-11 Trusted Solaris 2.5.1 Equivalents of Trusted Solaris 1.2 Commands

Trusted Solaris 1.2 User Command	Trusted Solaris 2.5.1 Equivalent
`tnetd(8T)`	`tnd(1MTSOL)`
`tnetd_ctl(8T)`	`tnctl(1MTSOL)`
`tsh` (trusted administrative shell)	`pfsh(1MTSOL)`

Table 2-12 Trusted Solaris 2.5.1 New User and Administrative Commands

New Trusted Solaris Command	Notes
`add_allocatable(1MTSOL)`	Add device
`atohexlabel(1MTSOL)`	ASCII to hex label translation
`device_clean(1MTSOL)`	Clean devices after deallocation and before allocation
`devpolicy(1MTSOL)`	Load device policy into the kernel
`dl_booting(1MTSOL)`	Inform kernel of diskless booting state
`dl_restore(1MTSOL)`	Inform kernel of normal booting state
`getfattrflag(1TSOL)`	Get security attribute flags for a file
`getfpriv(1TSOL)`	Get file privilege sets
`getfsattr(1MTSOL)`	Get file system security attributes
`getfsattr_ufs(1MTSOL)`	Get underlying file system security attributes
`hextoalabel(1MTSOL)`	Hex to ASCII label translation
`ipcrm(1TSOL)`	Remove SVIPC IDs from namespace
`ipcs(1TSOL)`	Get status of SVIPC mechanisms
`newsecfs(1MTSOL)`	Create file system with security attributes
`pattr(1TSOL)`	Get process attribute flags
`pclear(1TSOL)`	Get process clearance
`pfsh(1MTSOL)`	Profile shell
`ppriv(1TSOL)`	Get process effective privilege set

Table 2-12 Trusted Solaris 2.5.1 New User and Administrative Commands

New Trusted Solaris Command	Notes
`pprivtest(1TSOL)`	Test file privilege sets and process effective privilege set
`remove_allocatable(1MTSOL)`	Remove allocatable device
`rpc.getpeerinfod(1MTSOL)`	Daemon that transmits audit and process information
`rpc.tbootparamd(1MTSOL)`	Trusted Solaris boot parameter daemon
`runpd(1MTSOL)`	Run a command for debugging the command's privilege requirements
`setfattrflag(1TSOL)`	Set security attribute flags for a file
`setfpriv(1TSOL)`	Set file privilege sets
`setfsattr(1MTSOL)`	Set security attributes on new file system
`tbootparam(1MTSOL)`	Inform rpc.tbootparamd that host is now labeled
`testfpriv(1TSOL)`	Test privilege sets associated with a file
`tnchkdb(1MTSOL)`	Check syntax of trusted network database files
`tnctl(1MTSOL)`	Configure Trusted Solaris network daemon
`tnd(1MTSOL)`	Trusted Solaris network daemon
`tninfo(1MTSOL)`	Obtain kernel level network information and statistics
`tokmapctl(1MTSOL)`	Configure token mapping daemon
`tokmapd(1MTSOL)`	Token mapping daemon
`writeaudit(1MTSOL)`	Write an audit record

Programmer Transition 3

Programmers familiar with the Trusted Solaris 1.2 operating environment will find the concepts of the Trusted Solaris 2.5.1 operating environment familiar, but will notice extensive differences in the implementation. Programmers familiar with the Solaris 2.5.1 operating environment will recognize the implementation but may be less familiar with the concepts.

The following table lists the major topics for this chapter.

3

Highlights

The Trusted Solaris 2.5.1 operating environment is standards-based. Windowing is based on Motif 1.2.4, networking is based on TSIX (RE1.2), the desktop is based on CDE 1.1, and the environment is based on the Solaris 2.5.1 operating environment, which is moving toward 1170 compliance. The NIS+ name service, a distributed database, contains network information; Solstice AdminSuite is its administrative front end.

The following table gives the highlights of Trusted Solaris 2.5.1 features that affect programmers.

Table 3-1 Highlights of Trusted Solaris 2.5.1 Features

Area	Highlighted Differences
1.2 non-privileged applications	Should run with little or no porting.
1.2 privileged applications	May require porting to handle privilege and authorization changes.
Solaris 2.5.1 applications	Should run with little or no porting.
Trusted Solaris 2.5.1 applications	Can wrap commands and executables in CDE actions and add Trusted Solaris attributes to application package. Applications should port easily to other platforms.
1.2 Boot scripts	Require porting to the Solaris 2.5.1 boot style.
1.2 Audit scripts	Require porting to read the Trusted Solaris 2.5.1 audit format. 1.2 audit records are not readable by Trusted Solaris 2.5.1 audit system and vice versa.
Access Control Lists (ACLs)	Initially implemented in the Trusted Solaris 1.2 environment. ACLs are now part of the Solaris 2.5.1 operating environment. Trusted Solaris 2.5.1 ACLS are modified versions of Solaris 2.5.1 ACLs, not of 1.2 ACLs. Significant interface changes.
Auditing	New audit tokens, audit events, audit classes, and audit policies.
Authorizations	Changes in names; assigned to users and roles via execution profiles. Significant interface changes.
Databases	Many Trusted Solaris 1.2 databases are no longer supported; others have been moved and/or renamed. Databases have been added. Significant interface changes.
Desktop	Now CDE, the common desktop environment for UNIX platforms. Significant interface changes.

Table 3-1 Highlights of Trusted Solaris 2.5.1 Features

Area	Highlighted Differences
Desktop applications	Applications (including OpenWindows applications) can be wrapped in a CDE action, and invoked from the front panel.
Devices	New policy implementation using the file `device_policy(4TSOL)`.
File system attributes	File systems now have security attributes.
Home directories	Home directories are multilevel directories.
Interoperability	Trusted Solaris 2.5.1 is designed to promote interoperability by conforming to standards. Where 1.2 did not conform to the same standards, interoperability is limited.
Labels	Labels are now configurable, including visibility, whether to enable ILs, and whether to float ILs. Minor interface changes.
Man pages	Trusted Solaris 2.5.1 man page sections end in tsol, such as man3tsol for library routines.
Network	Now based on TSIX (RE 1.2), that enables application portability to other UNIX platforms. Significant interface changes.
Ports	Now multilevel.
Packaging applications	Now have Trusted Solaris attributes on packages.
Printing	Now based on SVR4 standard, not BSD. Trusted NeWSprint no longer used. Significant interface changes.
Privileges	Changes in names, semantics, and assigning them to executables. Also, privilege debugging is now available. Significant interface changes.
Processes	Now multithreaded, with symmetric multiprocessing. Trusted Solaris 2.5.1 security attributes added to Solaris 2.5.1 attributes. Significant interface changes.
Streams	New trusted streams.
User information	Now in two databases, `tsoluser` and `tsolprof`, managed by NIS+. Significant interface changes.
Window Manager	Now `dtwm`, desktop window manager, the Motif base for CDE.
Window system	Now based on the Solaris 2.5.1 X11R5-based window system. New interfaces.
X Toolkits	Supports Motif, OpenLook, XView.

3

Man Pages and Header File Locations

Table 3-2 gives the Trusted Solaris 2.5.1 locations for man pages and header files. Trusted Solaris 1.2 man pages appended a `T` to the man page name. Trusted Solaris 2.5.1 man pages have a `TSOL` extension and `tsol` is appended to the man page name. The Intro man pages explain Trusted Solaris concepts and provide a summary of the Trusted Solaris man pages.

Table 3-2 Man Page and Header File Locations

What	Trusted Solaris 1.2 Location	Trusted Solaris 2.5.1 Location
Trusted Solaris man pages	`/usr/man/man*t`	`/usr/man/man*tsol`
Trusted Solaris user header files	`/usr/include/cmw/*.h`	`/usr/include/tsol/*.h`
Label builder header	`/usr/include/cmw/lbuild.h`	`/usr/dt/include/Dt/ModLabel.h`
Auditing header	`/usr/include/bsm/auditwrite.h`	`/usr/include/bsm/auditwrite.h`
Label clipping header	No equivalent.	`/usr/dt/include/Dt/label_clipping.h`
RPC header	No equivalent.	`/usr/include/rpc/rpc.h`
IPC header	No equivalent.	`/usr/include/sys/ipc/ipc/h`
SV Message Queue header	No equivalent.	`/usr/include/sys/msg.h`
SV Semaphores header	No equivalent.	`/usr/include/sys/sem.h`
SV Shared memory header	No equivalent.	`/usr/include/sys/shm.h`
Trusted streams header	No equivalent.	`/usr/include/sys/tsol/stream.h`
TSIX library header	No equivalent.	`/usr/include/tsix/t6attrs.h`
System header files	No equivalent.	`/usr/include/sys/tsol/*.h`
X11 header	`/usr/openwin/include/cmw/Xcmw.h`	`/usr/openwin/include/tsol/Xtsol.h`

Database Locations

The following table gives the directory locations for databases. Database man pages are stored in the `.../man4tsol` directory. For a complete list of databases, see Table 2-1 on page 13.

Table 3-3 Trusted Solaris 2.5.1 Database Directories

Directory	Database Description
/	Dot files, such as .login and .profile, are not read at startup.
/etc	Contains standard Solaris files, such as `nsswitch.conf`, some extended for Trusted Solaris 2.5.1, such as `system`., and new databases, such as `tsolgateways`
/etc/rc*n*.d	Contains scripts called during booting.
/etc/inet.d	Contains daemon information. May need to be altered for daemons that handle multilevel applications.
/etc/security	Contains audit configuration files.
/etc/security/tsol	Contains Trusted Solaris databases.
/usr/dt/	Contains CDE and window manager databases.

New Network Database Interfaces – `tnrhdb` *and* `tnrhtp`

The `tnrhdb(4TSOL)` database is the equivalent of the Trusted Solaris 1.2 TNETRHDB. However, it simplifies entries for a host or domain by pointing to a template of security attributes found in the `tnrhtp(4TSOL)` database. The `tnidb(4TSOL)` database is the (local) network interface database. Its format is similar to `tnrhdb`. The programming interfaces that affect hosts are listed in Table 3-4 on page 52.

A `tnrhdb` entry has:

- host IP address
- template name (from `tnrhtp`)

A `tnrhtp` entry for an unlabeled host consists of:

- template name
- host type
- default label
- default clearance

- default UID
- default GID
- forced privileges

Entries for labeled hosts have fields specific to those host types. See the `tnrhtp(4TSOL)` man page for details. Host types are discussed in "Network" on page 21.

The system call in the following table affects network host labeling in the kernel:

Table 3-4 Kernel Host Information API

Operation	Interaface
Change the view of a host between labeled and unlabeled	`chstate(2TSOL)`

System Packaging Tools

The system packaging tools have been enhanced in the Trusted Solaris 2.5.1 release with Trusted Solaris 2.5.1 attributes. The system packaging tools are based on the Solaris tools that provide both the means to package components and the foundation for the patch mechanism.

Changes to Programming Interfaces

Applications coded for the Trusted Solaris 1.2 environment may need recoding because of changes to the Trusted Solaris 1.2 application programming interfaces (APIs). The Trusted Solaris 2.5.1 and 1.2 releases are not binary compatible, and a binary compatibility package is not provided.

For programming interfaces that have not changed from the Solaris 2.5.1 operating environment, see *Solaris 1.x to Solaris 2.x Transition Guide* and the Solaris 2.5.1 document set.

For full alphabetical listing of programming interfaces that have changed from the Solaris 2.5.1 operating environment, are unique to Trusted Solaris releases, or have been removed in this version of the Trusted Solaris operating environment, see Appendix A, "Trusted Solaris Programming Interfaces."

Devices

Device allocation programming interfaces have been removed from Trusted Solaris 2.5.1 and not replaced. The library routines allowed a program to read the information in the `device_allocate(4TSOL)` and `device_maps(4TSOL)` files. The removed interfaces are listed in Table A-7 on page 95. In the Trusted Solaris 2.5.1 environment, the user interface for device allocation is restricted to the device allocation GUI program launched from the Trusted Path menu.

Device security policy is configurable in the `device_policy(4TSOL)` file.

Unlike devices in the Solaris 2.5.1 environment, Trusted Solaris 2.5.1 devices are not affected by the `bsmconv(1M)` and `bsmunconv(1M)` scripts.

Label Interfaces

The labels and clearance API has had a few minor parameter list and binary format changes. Recompiling should handle any differences between Trusted Solaris 1.2 and Trusted Solaris 2.5.1 labels, although some recoding may be necessary.

- There are now 256 bits available for compartments and an additional 256 bits available in information labels for markings, up from 128 in Trusted Solaris 1.2. There are 32767 bits set aside for classifications, with ADMIN_LOW defined as 0 and ADMIN_HIGH defined as 32767.
- The Trusted Solaris 1.2 label view environment variable, `SUNW_CMW_LABEL_VIEW`, is now the label view process attribute flag `PAF_LABEL_VIEW`.

The following table lists the new library routines for labels and clearances. Table 3-6 lists the modified library routines for labels and clearances.

Table 3-5 New API for Labels and Clearances

Operation	Interfaces
Symbolic link CMW label	`lsetcmwlabel(2TSOL)`

Re-entrant binary to hexadecimal conversion	`bcleartoh_r(3TSOL)`
	`bcltoh_r(3TSOL)`
	`biltoh_r(3TSOL)`
	`bsltoh_r(3TSOL)`

Space for hexadecimal value	`h_alloc(3TSOL)` `h_free(3TSOL)`
ASCII and hexadecimal conversion	`atohexlabel(1MTSOL)` `hextoalabel(1MTSOL)`

X Window string conversion	`Xbcleartos(3TSOL)`
	`Xbcltos(3TSOL)`
	`Xbiltos(3TSOL)`
	`Xbsltos(3TSOL)`

Table 3-6 Modified Label Commands and API

Operation	Interfaces	Modification
File CMW label	`fsetcmwlabel(2TSOL)` `setcmwlabel(2TSOL)`	Constant parameter now in list and new structure
Process CMW label	`setcmwplabel(2TSOL)`	New structure

Hexadecimal to binary conversion	`htobcl(3TSOL)`
	`htobclear(3TSOL)`
	`htobil(3TSOL)`
	`htobsl(3TSOL)`
		Constant parameters now in list

Binary to ASCII conversion	`bcleartos(3TSOL)` `bcltobanner(3TSOL)`	

Table 3-6 Modified Label Commands and API

Operation	Interfaces	Modification

ASCII to binary conversion	`sbcleartos(3TSOL)`
	`sbcltos(3TSOL)`
	`sbiltos(3TSOL)`
	`sbsltos(3TSOL)`
	`setbltype(3TSOL)`
	`stobcl(3TSOL)`
	`stobil(3TSOL)`	

Comparison	`bilconjoin(3TSOL)`
	`bildominates(3TSOL)`
	`bilequal(3TSOL)`	

Bounds	`blmaximum(3TSOL)` `blminimum(3TSOL)`	
Get ASCII color name	`bltocolor(3TSOL)`	
Find label type	`bltype(3TSOL)`	
Get `label_encodings` file version	`labelvers(3TSOL)`	

Label Encodings File Interfaces

The following table lists the interfaces to access `label_encodings` file information:

Table 3-7 New API for Labels and Clearances

Operation	Interfaces
Get ASCII color name for a binary label.	`bltocolor(3TSOL)`
Get ASCII color name for a binary label, reentrant.	`bltocolor_r(3TSOL)`
Get maximum string length specifications.	`labelinfo(3TSOL)`
Get the label encodings file version information.	`labelvers(3TSOL)`

Label Builder Interfaces

Label builder interfaces in the Trusted Solaris 2.5.1 release are based on Motif 1.2.4 rather than Open Look Interface Toolkit (OLIT). The `event_handler()` routine is gone due to implementation changes in the label building interfaces. The man page directory has changed to `man3tsol`. The next table lists the label builder interfaces.

Table 3-8 Label Builder Interfaces

Operation	Interfaces
Create a label builder.	`tsol_lbuild_create(3TSOL)`
Get values.	`tsol_lbuild_get(3TSOL)`
Set values.	`tsol_lbuild_set(3TSOL)`
Destroy values.	`tsol_lbuild_destroy(3TSOL)`

Privilege Interfaces

Because the privilege and authorization sets have changed greatly between the Trusted Solaris 1.2 versions and the Trusted Solaris 2.5.1 release, privileges and authorizations cannot be exchanged between releases. Therefore, only nonprivileged interoperation is supported; no authorization checking takes place. There is no compatibility library. See Appendix B, “Privileges and Authorizations Transition” for lists of removed and added privileges.

- File and process privilege set interfaces have new names.
- There are new interfaces for translating privilege manifest constant names.
- The Trusted Solaris 1.2 Trusted Path privilege `sys_trusted_path` is now a process attribute flag.
- Privilege sets are now stored in a structure instead of an array. The `priv_set_t` data type is now a structure instead of an array.
- The mapping between privilege names and privilege IDs of the Trusted Solaris 1.2 releases has not been preserved.

- Process privilege set updating has been changed when a new program is executed with `exec(2TSOL)`. The change provides better security and improved privilege inheritance. The formulas for each release are shown. In the formulas, E means effective privilege, P is permitted, I is inherited, S is Saved, F is forced, and A is allowed privilege sets.
  - Trusted Solaris 1.2 privilege set updating

    $E_2 = P_2 = I_2 = (I_1$ union F) intersected by A.
    $S_2 = I_1$ intersected by A
  - Trusted Solaris 2.5.1 privilege set updating

    $E_2 = P_2 = (I_1$ union F) intersected by A.
    $S_2 = I_1$ intersected by A
    $I_2 = I_1$. A process with an empty allowed set can pass inheritable privileges.

See "Why Inheritable Privileges Are Important" in Chapter 16 of the *Trusted Solaris Administrator's Procedures*.

The following table lists the new and modified Trusted Solaris 2.5.1 privilege interfaces.

Table 3-9 Privilege API Changes

Operation	Interfaces	Origin/Change

Get and set file privilege sets	`getfpriv(2TSOL)`
	`setfpriv(2TSOL)`
	`fgetfpriv(2TSOL)`
	`fsetfpriv(2TSOL)`
		Modified 1.2 (new structure)

Get and set process privilege sets	`getppriv(2TSOL)` `setppriv(2TSOL)`	
Convert privileges	`priv_to_str(3TSOL)` `str_to_priv(3TSOL)`	

Set process privilege sets	`set_effective_priv(3TSOL)`
	`set_inheritable_priv(3TSOL)`
	`set_permitted_priv(3TSOL)`
		New in 2.5.1

Convert privileges	`priv_set_to_str(3TSOL)`
	`str_to_priv_set(3TSOL)`
	`get_priv_text(3TSOL)`	

Privilege Macros

The man page `priv_macros(5TSOL)` describes the Trusted Solaris 2.5.1 privilege macros. Some macros have the same name as in the Trusted Solaris 1.2 operating environment, but their parameter lists or data types are slightly different.

Table 3-10 Changes to Privilege Macros

Present in Both Releases	Removed from Trusted Solaris 2.5.1	Added in Trusted Solaris 2.5.1
PRIV_ASSERT	PRIV_COPY	PRIV_ISFULL
PRIV_ISASSERT		PRIV_FILL
PRIV_EMPTY		PRIV_EQUAL
PRIV_ISEMPTY		
PRIV_CLEAR		
PRIV_INTERSECT		
PRIV_UNION		
PRIV_XOR		
PRIV_INVERSE		
PRIV_ISSUBSET		

Privilege Debugging

In the Trusted Solaris 2.5.1 operating environment, privilege debugging mode logs privilege failures while allowing applications to succeed. The log lists the privileges the program needs for successful operation. An administrative role with appropriate access enables privilege debugging by:

1. Setting the `tsol_privs_debug` switch in the `/etc/system` file to the value `1`. The Admin Editor action is used to modify the file.
2. Uncommenting the `kern.debug` line in the `/etc/syslog.conf` file. The Admin Editor action is used to modify the file.
3. Rebooting and issuing the command `runpd(1MTSOL)` from an administrative role to start the program in debug mode. The results print to the screen, as well as to `/var/log/privdebug.log`, an ADMIN_HIGH file owned by root.

Authorization Interfaces

Because the authorization set has changed greatly between the Trusted Solaris 1.2 versions and the Trusted Solaris 2.5.1 release, authorizations cannot be exchanged between releases. There is no compatibility library.

- Authorization sets are now stored in a structure.
- The mapping between authorization names and authorization IDs of the Trusted Solaris 1.2 operating environment has not been preserved.

The following table lists the new authorization interfaces.

Table 3-11 Trusted Solaris 2.5.1 Authorization API Changes

Operation	Interfaces	Origin/Change
Check authorizations	`chkauth(3TSOL)`	Replaces `check_authorization`(3T)
Convert authorizations	`auth_to_str(3TSOL)`	New

	`str_to_auth(3TSOL)`
	`auth_set_to_str(3TSOL)`
	`str_to_auth_set(3TSOL)`	

Free authorization set	`free_auth_set(3TSOL)`	
Get authorization description	`get_auth_text(3TSOL)`	

The removed interfaces are listed in Table A-7 on page 95.

Process Attributes and Flags Interfaces

The Trusted Solaris 2.5.1 operating environment has extended the Solaris 2.5.1 process attributes as shown in Table 3-12. These attributes are used to make access control decisions when a process tries to read or write data. For

multithreaded applications, all threads share a single view of the process security attributes. Table 3-13 on page 60 lists the new programming interfaces for manipulating these attributes.

Table 3-12 Security Attributes Available in the Trusted Solaris 2.5.1 Environment

Base Solaris Security Attributes	Trusted Solaris Security Attributes
Effective user ID	Sensitivity label
Effective group ID	Information label
Process ID	Process clearance
Network session ID	Effective privilege set
Supplementary group IDs	Process attribute flags
Audit ID	
Audit information (process preselection mask, audit terminal ID, and audit session ID)	

As in the Solaris 2.5.1 implementation, all threads of a program share a single view of process security attributes. When a thread makes a change to process security attributes, the new attributes are picked up by other threads the next time those threads enter kernel mode (such as making a system call). If a thread is in the middle of a system call while other threads are changing process security attributes, the new attributes would not be seen by the thread until it finishes the current system call. That is, process security attributes of a thread are kept stable during the course of a system call.

Table 3-13 Process Command and API Changes

Operation	Interfaces	Change
Process CMW Label	`setcmwplabel(2TSOL)`	New structure
Process clearance	`pclear(1TSOL)`	New
Process CMW label	`plabel(1TSOL)`	New
Process privilege sets	`ppriv(1TSOL)` `pprivtest(1TSOL)`	New

Table 3-13 Process Command and API Changes

Operation	Interfaces	Change
Process security attributes	`pattr(1TSOL)`	New
Process privilege sets	`getppriv(2TSOL)` `setppriv(2TSOL)`	New
Process security attribute flags	`getpattr(2TSOL)` `setpattr(2TSOL)`	New

File System Attributes and Flags Interfaces

The Trusted Solaris 2.5.1 operating environment extends the Solaris 2.5.1 file system attributes to include Trusted Solaris security attributes. Since privilege sets have also changed from the Trusted Solaris 1.2 operating environment, there are several changes to file system interfaces.

The next table lists the new or modified Trusted Solaris 2.5.1 interfaces to access file system security attributes and flags.

Table 3-14 Unique to Trusted Solaris 2.5.1 File System Interfaces

Operation	File System API	Origin/Change
Get and set file system security attributes	`fgetfsattr(2TSOL)` `getfsattr(2TSOL)`	New
Get single-level directory names	`fgetsldname(2TSOL)` `getsldname(2TSOL)`	Supports new sld names
Get multilevel directory adornment	`fgetmldadorn(2TSOL)` `getmldadorn(2TSOL)`	Parameter list change

The following table lists the Trusted Solaris 2.5.1 interfaces that are new or modified for files.

Table 3-15 Unique to Trusted Solaris 2.5.1 File Interfaces

Operation	File System API	Origin/Change
Get and set security attribute flags	`fgetfattrflag(2TSOL)` `getfattrflag(2TSOL)`	New

Get and set file privilege sets	`getfpriv(2TSOL)`
	`fgetfpriv(2TSOL)`
	`setfpriv(2TSOL)`
	`fsetfpriv(2TSOL)`
		New structure

Get multilevel directory file statistics	`mldstat(2TSOL)` `mldlstat(2TSOL)`	Parameter list change

Access Control Lists

Trusted Solaris 2.5.1 Access Control Lists (ACLs) are implemented as Solaris 2.5.1 ACLs are. The Solaris 2.5.1 operating environment has made a number of changes to the ACL application programming interfaces; the ACLs in Trusted Solaris 2.5.1 are based on the re-implementation, not on the ACL implementation shipped with Trusted Solaris 1.2 releases. There are no plans to provide a compatibility library.

The Trusted Solaris 2.5.1 operating environment has extended the support for ACLs to include the `tmpfs` file system.

The following table lists the Trusted Solaris 2.5.1 ACL interfaces that are new or modified from the Trusted Solaris 1.2 implementation.

Table 3-16 ACL Interface Changes

Operation	ACL Interface
Get or set file's ACL	`acl(2TSOL)`, `facl(2TSOL)`
Check validity	`aclcheck(3)`
Convert to internal representation	`acltomode(3)`
Sort ACL	`aclsort(3)`
Convert an ACL to permission bits	`acltopbits(3)`
Convert to external representation	`acltotext(3)`

ACLs are a file system attribute. Trusted Solaris 2.5.1 programming interfaces to retrieve file system attributes were described in “File System Attributes and Flags Interfaces” on page 61.

User Database Interfaces – `tsolprof` *and* `tsoluser`

The Trusted Solaris 2.5.1 operating environment has new interfaces to access user entries in the `tsoluser(4TSOL)` database and execution profile entries in the `tsolprof(4TSOL)` database. The following table lists the library routines for accessing user information.

Table 3-17 Library Routines to Access Execution Profile and User Information

Execution Profile Library Routines	User Library Routines
`getprofent(3TSOL)`	`getuserent(3TSOL)`
`getprofentbyname(3TSOL)`	`getuserentbyname(3TSOL)`
`getprofstr(3TSOL)`	`getuserentbyuid(3TSOL)`
`getprofstrbyname(3TSOL)`	
`endprofent(3TSOL)`	`enduserent(3TSOL)`
`endprofstr(3TSOL)`	
`free_profent(3TSOL)`	`free_userent(3TSOL)`
`free_profstr(3TSOL)`	
`setprofent(3TSOL)`	`setuserent(3TSOL)`
`setprofstr(3TSOL)`	

Execution Profile Database Entries

Execution profiles define the authorizations, commands, and actions a particular user or role can use. Profiles replace a number of Trusted Solaris 1.2 databases, such as `authoriz`, `ctab_list`, `tcb_dynamic`, `tcb_static`, `tfm_role.cmds`, `tfm_role.fncs`, `user.cmw_labels`, and `user.login_ctrl`.

Note – In the Trusted Solaris 2.5.1 release, there is no consistency checking. However, profiles reduce reliance on forced privileges for security-relevant files.

The `tsolprof(4TSOL)` database entries consist of multiple lines of text, containing at least the following five fields:

- profile
- description
- authorizations
- actions
- commands

The action field consists of:

- action name
- class
- type
- argument mode
- argument count
- privileges
- effective user id
- effective group id
- minimum sensitivity label
- maximum sensitivity label

The command field consists of

- directory
- file name
- privileges
- effective user id
- effective group id
- minimum sensitivity label
- maximum sensitivity label

User Database Entries

The `tsoluser(4TSOL)` database entries consist of multiple lines of text with the following fields:

- username

- user account status (locked, open)
- number of failed logins
- method of password generation
- associated execution profiles
- assumable roles
- minutes a workstation can remain idle
- what to do when idle time is reached
- label view – whether internal (display `ADMIN_HIGH` and `ADMIN_LOW`) or external (demote `ADMIN_HIGH` to next lowest external label and promote `ADMIN_LOW` next highest external label).
- label translation – When on, label translation means the `flags= keyword` option is in use in the `label_encodings` file.
- allowed low login labels
- allowed high login labels
- type of user (regular, admin role, non-admin role)
- three reserved fields.

Auditing

The Trusted Solaris 2.5.1 auditing interfaces are very similar in concept to the audit interfaces provided in the Trusted Solaris 1.x operating environment, but are very similar in implementation to those provided in SunOS 5.x. Trusted Solaris 2.5.1 extends the Solaris 2.5.1 Basic Security Module (BSM) to include audit classes, events, and tokens that are specific to Trusted Solaris, and includes Trusted Solaris security attributes in its audit records. It also has the framework for per-file auditing, but does not support it in this release.

The following table lists the auditing interfaces that are modified from the Solaris 2.5.1 BSM.

Table 3-18 Auditing Modifications from the Base

Trusted Solaris Changes	Auditing API
Calling process needs `proc_audit_tcb` or `proc_audit_appl`	`audit(2TSOL)`
Trusted Solaris audit flags added	`auditon(2TSOL)`

Table 3-18 Auditing Modifications from the Base

Trusted Solaris Changes	Auditing API
Calling process needs `sys_audit`	`auditsvc(2TSOL)`
To get or set the audit ID, the calling process needs `sys_audit`. To get the audit ID, the calling process needs `proc_audit_appl` (if the process generates non-TCB events). `or proc_audit_TCB` (if the process generates TCB events).	`getauid(2TSOL)`
Calling process needs `sys_audit`	`setauid(2TSOL)`

The following table lists the auditing interface changes from the Trusted Solaris 1.2 operating environment.

Table 3-19 Auditing Modifications from Trusted Solaris 1.2

Trusted Solaris Changes	Auditing API
Removed from Trusted Solaris 2.5.1	`auditstat(2)`
Calling process needs `proc_audit_appl` to construct user-level audit records and `proc_audit_tcb` to construct TCB audit events.	`auditwrite(3TSOL)`
Calling process needs `proc_audit_appl` to get `peer`'s attributes.	`getpeerinfo(3TSOL)`
Calling process needs `sys_audit` and `proc_audit_appl` or `proc_audit_tcb` to get audit information.	`getaudit(2TSOL)`
Calling process needs `sys_audit` to set audit information.	`setaudit(2TSOL)`

Trusted X Window System

The Trusted Solaris 2.5.1 X window system is based on and generally compatible with the Solaris 2.5.1 X11R5-based window system. The Trusted Solaris 2.5.1 window system adds access control to all X resources that are accessible through the X protocol.

Although restrictions apply to virtually every protocol request, the security policy is transparent to most commercial applications. Inappropriate access is denied, and the event is auditable.

X Protocol Extensions

The Trusted Solaris 2.5.1 operating environment has extended the X Protocols to restrict data flow according to security policy. Most access operations will function normally; however, some operations, such creating a new colormap, require privilege.

The Trusted Solaris 1.2 Trusted X server interfaces, based on the X11R4 library and the Xnews server, have been replaced by the Trusted X11R5 server library, which implements the Xsun server. The interfaces do not map one-to-one. Applications coded for Trusted Solaris 1.2 Trusted X server interfaces must be recoded. There are no plans to provide a compatibility library.

The following table lists interfaces accessing X window system security attributes. The removed interfaces are listed in Table A-7 on page 95.

Table 3-20 X WIndow System Interfaces Unique to the Trusted Solaris 2.5.1 release

Operation	Trusted Solaris 2.5.1 X Protocol Extensions
Test for Trusted Path window	`XTSOLIsWindowTrusted(3X11TSOL)`
Create Trusted Path window	`XTSOLMakeTPWindow(3X11TSOL)`
Shut down the workstation	`XTSOLShutdown(3X11TSOL)`
Get client connection attributes	`XTSOLgetClientAttributes(3X11TSOL)`
Get property attributes	`XTSOLgetPropAttributes(3X11TSOL)`
Get property CMW label	`XTSOLgetPropLabel(3X11TSOL)`
Get property user ID	`XTSOLgetPropUID(3X11TSOL)`
Get window, colormap, or pixmap attributes	`XTSOLgetResAttributes(3X11TSOL)`
Get window, colormap, or pixmap CMW label	`XTSOLgetResLabel(3X11TSOL)`
Get window, colormap, or pixmap user ID	`XTSOLgetResUID(3X11TSOL)`
Get screen stripe height	`XTSOLgetSSHeight(3X11TSOL)`
Get window input information label	`XTSOLgetWindowIIL(3X11TSOL)`
Get server user ID	`XTSOLgetWorkstationOwner(3X11TSOL)`
Set polyinstantiation information	`XTSOLsetPolyInstInfo(3X11TSOL)`
Set property CMW label	`XTSOLsetPropLabel(3X11TSOL)`
Set property user ID	`XTSOLsetPropUID(3X11TSOL)`

Table 3-20 X WIndow System Interfaces Unique to the Trusted Solaris 2.5.1 release

Operation	Trusted Solaris 2.5.1 X Protocol Extensions
Set resource CMW	`XTSOLsetResLabel(3X11TSOL)`
Set resource user ID	`XTSOLsetResUID(3X11TSOL)`
Set screen stripe height	`XTSOLsetSSHeight(3X11TSOL)`
Set session clearance	`XTSOLsetSessionHI(3X11TSOL)`
Set minimum label	`XTSOLsetSessionLO(3X11TSOL)`
Set window input information label	`XTSOLsetWindowIIL(3X11TSOL)`
Set server user ID	`XTSOLsetWorkstationOwner(3X11TSOL)`

Inter-Process Communication (IPC)

Trusted Solaris 2.5.1 has two primary interprocess communication mechanisms: System V Inter-Process Communications (SVIPC) which provides only `intra`-machine paths, and networking, that provides both *intra*- and *inter*-machine communication paths.

System V IPC

The Trusted Solaris 2.5.1 operating environment enhances Solaris 2.5.1 system calls to handle labels and other security features. The Trusted Solaris 1.2 SVIPC system calls are unchanged in this release, except that `aclipc`(2T) system call is not included and `semopl(2TSOL)` has been added.

Note – Since the privilege mechanism is different from the Trusted Solaris 1.2 implementation, read the man pages for details.

The following table lists the Solaris 2.5.1 interfaces enhanced for security. Trusted Solaris 1.2 SVIPC interfaces are listed in Table A-1 on page 76.

Table 3-21 SVIPC Interfaces Enhanced in the Trusted Solaris 2.5.1 Releases

Enhancements	SVIPC Interfaces	
Checks that object is at subject's sensitivity label. If not, checks that the calling process asserts the appropriate privilege override.	`msgctl(2TSOL)`	`semget(2TSOL)`
	`msgget(2TSOL)`	`semop(2TSOL)`
	`msgrcvl(2TSOL)`	`shmat(2TSOL)`
	`msgsndl(2TSOL)`	`shmctl(2TSOL)`
	`semctl(2TSOL)`	`shmdt(2TSOL)`
		`shmget(2TSOL)`

Endpoint Communications Interfaces

The Berkeley sockets and Transport Layer (TLI) interfaces have been modified to recognize and create multilevel ports.

Table 3-22 Solaris 2.5.1 Socket Interfaces Modified for Security and Multilevel Ports

Trusted Solaris 2.5.1 Changes	Changed Interfaces
Modified to identify multilevel ports. Binding process needs `net_mac_read` privilege.	`accept(3NTSOL)`
	`bind(3NTSOL)`
	`connect(3N)`
	`getsockopt(3N)`
	`recv(3N)`
	`recvfrom(3XN)`
	`recvmsg(3XN)`
Modified to send packets at same label as preceding packet. Calling process needs `net_reply_equal` privilege.	`send(3NTSOL)`
	`sendmsg(3NTSOL)`
	`sendto(3NTSOL)`
	`socket(3N)`
	`socketpair(3N)`

Table 3-23 Solaris 2.5.1 TLI Interfaces Modified for Security and Multilevel Ports

Trusted Solaris 2.5.1 Changes	Changed Interfaces
Modified to identify multilevel ports. Binding process needs `net_mac_read` privilege.	`t_accept(3NTSOL)`
	`t_bind(3NTSOL)`
	`t_alloc(3N)`
	`t_close(3N)`
	`t_connect(3N)`
	`t_error(3N)`
	`t_free(3N)`
	`t_getinfo(3N)`
	`t_getstate(3N)`
	`t_listen(3N)`
	`t_look(3N)`
	`t_open(3N)`
	`t_optmgmt(3NTSOL)`
	`t_rcvconnect(3N)`
	`t_rcvdis(3N)`
	`t_rcvrel(3N)`
	`t_rcvudata(3N)`
Modified to identify multilevel ports. Binding process needs `net_mac_read` privilege.	`t_snd(3NTSOL)`
	`t_snddis(3N)`
	`t_sndrel(3N)`
	`t_sndudata(3N)`
	`t_strerror(3N)`
	`t_sync(3N)`
	`t_unbind(3N)`

Trusted Security Information Exchange Library (TSIX)

Trusted Solaris 1.2 trusted network interfaces have been replaced by the Trusted Security Information eXchange (TSIX (RE 1.2)) library for transporting security attribute information over communication endpoints. The old and new interfaces do not map one-to-one. Trusted Solaris 1.2 applications must be recoded to use the TSIX library. There are no plans to provide a compatibility library.

The TSIX library promotes portability of trusted applications. Though not required for interoperability, it greatly assists third-party software developers who write distributed applications for multiple platforms. See Table 3-12 for the lists of Solaris 2.5.1 and Trusted Solaris 2.5.1 security attributes.

The following table lists the new TSIX library routines. The removed interfaces are listed in Table A-7 on page 95.

Table 3-24 Network Interfaces Unique to Trusted Solaris 2.5.1

Operation	**2.5.1 Network Interfaces**
Allocate space	`t6alloc_blk(3NTSOL)`
Duplicate attributes	`t6dup_blk(3NTSOL)`
Enable security options	`t6ext_attr(3NTSOL)`
Free space	`t6free_blk(3NTSOL)`
Get security attributes	`t6get_attr(3NTSOL)`
Get endpoint attributes	`t6get_endpt_default(3NTSOL)`
Get endpoint attributes	`t6get_endpt_mask(3NTSOL)`
Return new attributes only	`t6new_attr(3NTSOL)`
Examine attributes	`t6peek_attr(3NTSOL)`
Receive data and attributes	`t6recvfrom(3NTSOL)`
Send data and attributes	`t6sendto(3NTSOL)`
Set security attributes	`t6set_attr(3NTSOL)`
Set endpoint default attributes	`t6set_endpt_default(3NTSOL)`
Get size of one attribute	`t6size_attr(3NTSOL)`
Examine attributes on last data byte.	`t6last_attr(3NTSOL)`
Copy attributes.	`t6copy_blk(3NTSOL)`

Table 3-24 Network Interfaces Unique to Trusted Solaris 2.5.1

Operation	2.5.1 Network Interfaces
Compare attributes.	`t6cmp_blk(3NTSOL)`
Clear security attributes	`t6clear_blk(3NTSOL)`
Set the endpoint attribute mask.	`t6set_endpt_mask(3NTSOL)`
Backwards compatibility.	`t6ext_attr(3NTSOL)`

RPC Interfaces

Trusted Solaris 2.5.1 RPC library routines have been modified to handle the transportation of security attribute information. Data structures have been modified to contain the attributes, and RPC calls have been modified to transport the attributes in the structure. See the `rpc(3NTSOL)` man page for more information.

New Multilevel Ports

In the Trusted Solaris 2.5.1 operating environment, all communication endpoints are either multilevel or single-level ports to support unmodified applications in a trusted environment. In the Trusted Solaris 1.2 operating environment, only single-level ports were supported.

- Multilevel port – A network endpoint associated with all sensitivity labels.
- Single-level port – A network endpoint associated with one sensitivity label.

A multilevel or single-level port is created when an endpoint is bound. If the binding process has `net_mac_read`, the port is a multilevel port. Otherwise, it is a single-level port.

A multilevel port can receive data with any sensitivity label, while a single-level port can receive only data with the matching label.

Trusted Streams

The Solaris 2.5.1 `streamio(7)` device network interface is modified. It runs below other Trusted Solaris 2.5.1 networking interfaces and is not usually used directly. The Trusted Streams interfaces let you send and receive security

attribute information across a stream. They are new in the Trusted Solaris 2.5.1 operating environment. Table A-2 on page 78 lists the modified and new interfaces.

CDE Desktop Applications

Applications are launched from the CDE desktop as CDE actions. The Trusted Solaris 2.5.1 operating environment fully supports applications written with CDE interfaces and the following X Toolkits:

- Motif 1.2.4
- XView (current version is 3.5.1)
- OpenLook Interface Toolkit (OLIT) (current version is 3.5.1)

Note – Any toolkit that conforms to the Inter-Client Communications Conventions Manual (ICCCM) standards should function properly. However, the Motif toolkit provides maximum portability.

CDE applications shipped with the Trusted Solaris 2.5.1 release are based on Motif 1.2.4. However, Trusted Solaris 2.5.1 supports and allows interoperability with OpenWindows applications based on the XView and OpenLook Interface Toolkit (OLIT) toolkits.

Motif-based applications should run in the Trusted Solaris 2.5.1 environment with little or no modification. See the *Solaris Common Desktop Environment: Motif Transition Guide*, PN 802-2962-10 (June 1995) manual for Motif to Solaris porting issues. Colormaps, for example, are handled differently.

Possible Trusted Solaris modifications to existing applications include: creating a multilevel directory for program files, adding properties created by the application to the file `/usr/openwin/server/tsol/property.atoms` so that the properties can be polyinstantiated, and root (UID=0) permission during installation.

Note – Use `runpd(1MTSOL)` to check for privilege requirements. See “Privilege Debugging” on page 58 for details.

3

CDE enables you to wrap commands and executables in CDE actions that can be invoked by clicking on an icon in the front panel. CDE action creation is unmodified in the Trusted Solaris 2.5.1 environment. See the CDE document set for programming guidelines. See the *Trusted Solaris Administrator's Procedures* for restrictions on CDE action assignment.

For	**See: *Common Desktop Environment Documentation***
Programming style	`Common Desktop Environment: Style Guide and Certification Checklist`
Online help	`Common Desktop Environment: Help System Author's and Programmer's Guide`
Creating actions	*Solaris Common Desktop Environment: Advanced User's and System Administrator's Guide* *Solaris Common Desktop Environment: Motif Transition Guide* *Solaris Common Desktop Environment: Programmer's Guide*

Trusted Solaris Programming Interfaces

A

The tables list in alphabetical order all the system calls, functions, and library routines in the Trusted Solaris 2.5.1 release that are new or are modified versions of Solaris 2.5.1 interfaces. Changes from Trusted Solaris 1.2 interfaces are also listed.

A

System Calls Unique to Trusted Solaris Operating Systems

The system calls listed in Table A-1 were added to the Trusted Solaris environment in either the 1.2 or the 2.5 and 2.5.1 releases. The column on the right indicates whether the system call is new to or has changed in the Trusted Solaris 2.5.1 release. Where there is no change the column is empty.

Note – The man page section of system calls has changed from 2T in the Trusted Solaris 1.2 release to 2TSOL in the Trusted Solaris 2.5.1 release.

New – This system call is new in the Trusted Solaris 2.5.1 release.

Description – Changes from the Trusted Solaris 1.2 system call.

Table A-1 Trusted Solaris-specific System Calls

Trusted Solaris System Calls	Change from 1.2 to 2.5.1, if any
`chstate(2TSOL)`	New
`fgetcmwfsrange(2TSOL)`	
`fgetcmwlabel(2TSOL)`	
`fgetfattrflag(2TSOL)`	New
`fgetfpriv(2TSOL)`	New
`fgetfsattr(2TSOL)`	New
`fgetmldadorn(2TSOL)`	
`fgetsldname(2TSOL)`	
`fsetcmwlabel(2TSOL)`	Constant parameter now in list and new structure.
`fsetfattrflag(2TSOL)`	New
`fsetfpriv(2TSOL)`	New data structure and new name in 2.5.1
`getclearance(2TSOL)`	
`getcmwfsrange(2TSOL)`	
`getcmwlabel(2TSOL)`	
`getcmwplabel(2TSOL)`	
`getfattrflag(2TSOL)`	New
`getfpriv(2TSOL)`	New data structure and new name in 2.5.1

Table A-1 Trusted Solaris-specific System Calls

Trusted Solaris System Calls	Change from 1.2 to 2.5.1, if any
`getfsattr(2TSOL)`	New
`getmldadorn(2TSOL)`	
`getmsgqcmwlabel(2TSOL)`	
`getpattr(2TSOL)`	New
`getppriv(2TSOL)`	New data structure and new name in 2.5.1
`getsemcmwlabel(2TSOL)`	
`getshmcmwlabel(2TSOL)`	
`getsldname(2TSOL)`	
`lgetcmwlabel(2TSOL)`	
`lsetcmwlabel(2TSOL)`	New
`mldgetfattrflag(2TSOL)`	New
`mldsetfattrflag(2TSOL)`	New
`mldstat(2TSOL)`	Constant parameter now in list.
`mldlstat(2TSOL)`	Constant parameter now in list.
`msggetl(2TSOL)`	
`msgrcvl(2TSOL)`	
`msgsndl(2TSOL)`	
`preadl(2TSOL)`	New
`readl(2TSOL)`	New
`readlink(2TSOL)`	New
`readv(2TSOL)`	New
`readvl(2TSOL)`	New
`secconf(2TSOL)`	New
`semgetl(2TSOL)`	
`semopl(2TSOL)`	New
`setclearance(2TSOL)`	
`setcmwlabel(2TSOL)`	Constant parameter now in list and new structure.

A

Table A-1 Trusted Solaris-specific System Calls

Trusted Solaris System Calls	Change from 1.2 to 2.5.1, if any
`setcmwplabel(2TSOL)`	New structure for 2.5.1.
`setfattrflag(2TSOL)`	New
`setfpriv(2TSOL)`	New data structure and new name in 2.5.1
`setpattr(2TSOL)`	New
`setppriv(2TSOL)`	New
`shmgetl(2TSOL)`	
`writel(2TSOL)`	New
`writev(2TSOL)`	New
`writevl(2TSOL)`	New

Trusted Streams Functions

Table A-2 Trusted Streams Interfaces

Modified Solaris 2.5.1 Interfaces	New Trusted Streams Interfaces
`linkb(9FTSOL)`	`tsol_get_strattr(9FTSOL)`
`put(9FTSOL)`	`tsol_linkb(9FTSOL)`
`putbq(9FTSOL)`	`tsol_putctl(9FTSOL)`
`putctl(9FTSOL)`	`tsol_putctl1(9FTSOL)`
`putctl1(9FTSOL)`	`tsol_putnextctl(9FTSOL)`
`putnext(9FTSOL)`	`tsol_putnextctl1(9FTSOL)`
`putnextctl(9FTSOL)`	`tsol_set_strattr(9FTSOL)`
`putnextctl1(9FTSOL)`	
`putq(9FTSOL)`	
`qreply(9F)`	

Solaris 2.5.1 System Calls Enhanced for Security

The Solaris 2.5.1 system calls listed in Table A-3 are enhanced for security policy. The original Trusted Solaris modification release is indicated.

Table A-3 Modified Solaris 2.5.1 System Calls

Interface	Release Modified
access(2TSOL)	2.5.1
acl(2TSOL)	1.2
adjtime(2TSOL)	2.5.1
audit(2TSOL)	2.5.1
auditctl(2TSOL)	1.2
auditon(2TSOL)	2.5.1
auditsvc(2TSOL)	2.5.1
chdir(2TSOL)	2.5.1
chmod(2TSOL)	2.5.1
chown(2TSOL)	2.5.1
chroot(2TSOL)	2.5.1
close(2)	2.5.1
creat(2TSOL)	2.5.1
dup(2)	2.5.1
exec(2TSOL)	2.5.1
execl(2TSOL)	2.5.1
execle(2TSOL)	2.5.1
execlp(2TSOL)	2.5.1
execv(2TSOL)	2.5.1
execve(2TSOL)	2.5.1
execvp(2TSOL)	2.5.1
facl(2TSOL)	1.2
fchdir(2TSOL)	2.5.1
fchmod(2TSOL)	2.5.1

Table A-3 Modified Solaris 2.5.1 System Calls

Interface	Release Modified
fchown(2TSOL)	2.5.1
fchroot(2TSOL)	2.5.1
fcntl(2TSOL)	2.5.1
fork(2TSOL)	2.5.1
fork1(2TSOL)	2.5.1
fpathconf(2TSOL)	2.5.1
fstat(2TSOL)	2.5.1
fstatvfs(2TSOL)	2.5.1
getaudit(2TSOL)	1.2
getauid(2TSOL)	2.5.1
getdents(2TSOL)	2.5.1
getgroups(2TSOL)	2.5.1
getpgrp(2TSOL)	2.5.1
getsid(2TSOL)	2.5.1
ioctl(2)	2.5.1
kill(2TSOL)	2.5.1
lchown(2TSOL)	2.5.1
link(2TSOL)	2.5.1
llseek(2TSOL)	2.5.1
lseek(2TSOL)	2.5.1
lstat(2TSOL)	2.5.1
memcntl(2)	2.5.1
mkdir(2TSOL)	2.5.1
mknod(2TSOL)	2.5.1
mmap(2)	2.5.1
mount(2TSOL)	2.5.1
mprotect(2)	2.5.1

Table A-3 Modified Solaris 2.5.1 System Calls

Interface	Release Modified
msgctl(2TSOL)	2.5.1
msgget(2TSOL)	2.5.1
msggetl(2TSOL)	2.5.1
msgop(2TSOL)	2.5.1
msgrcv(2TSOL)	2.5.1
msgsnd(2TSOL)	2.5.1
msgsndl(2TSOL)	2.5.1
nice(2TSOL)	2.5.1
open(2TSOL)	2.5.1
pathconf(2TSOL)	2.5.1
pipe(2)	2.5.1
pread(2TSOL)	2.5.1
priocntl(2TSOL)	2.5.1
priocntlset(2TSOL)	2.5.1
processor_bind(2TSOL)	2.5.1
ptrace(2)	2.5.1
pwrite(2TSOL)	2.5.1
pwritel(2TSOL)	2.5.1
p_online(2TSOL)	2.5.1
read(2TSOL)	2.5.1
readvl(2TSOL)	2.5.1
rename(2TSOL)	2.5.1
rmdir(2TSOL)	2.5.1
semctl(2TSOL)	2.5.1
seek(2TSOL)	2.5.1
semget(2TSOL)	2.5.1
semgetl(2TSOL)	2.5.1

Table A-3 Modified Solaris 2.5.1 System Calls

Interface	Release Modified
`semop(2TSOL)`	2.5.1
`semopl(2TSOL)`	2.5.1
`setaudit(2TSOL)`	1.2
`setauid(2TSOL)`	2.5.1
`setegid(2TSOL)`	2.5.1
`seteuid(2TSOL)`	2.5.1
`setgid(2TSOL)`	2.5.1
`setgroups(2TSOL)`	2.5.1
`setrlimit(2TSOL)`	2.5.1
`setuid(2TSOL)`	2.5.1
`shmat(2TSOL)`	2.5.1
`shmctl(2TSOL)`	2.5.1
`shmdt(2TSOL)`	2.5.1
`shmget(2TSOL)`	2.5.1
`shmgetl(2TSOL)`	2.5.1
`sigsend(2TSOL)`	2.5.1
`sigsendset(2TSOL)`	2.5.1
`stat(2TSOL)`	2.5.1
`statvfs(2TSOL)`	2.5.1
`stime(2TSOL)`	2.5.1
`symlink(2TSOL)`	2.5.1
`sync(2)`	2.5.1
`sysfs(2)`	2.5.1
`sysinfo(2TSOL)`	2.5.1
`uadmin(2TSOL)`	2.5.1
`ulimit(2TSOL)`	2.5.1
`umask(2)`	2.5.1

Table A-3 Modified Solaris 2.5.1 System Calls

Interface	Release Modified
`umount(2TSOL)`	2.5.1
`unlink(2TSOL)`	2.5.1
`utimes(2TSOL)`	2.5.1
`vfork(2TSOL)`	2.5.1
`vhangup(2)`	2.5.1
`write(2TSOL)`	2.5.1

A

Removed Trusted Solaris 1.2 System Calls

The Trusted Solaris 1.2 system calls listed in Table A-4 are not supported in the Trusted Solaris 2.5.1 release. A Trusted Solaris 2.5.1 replacement system call is listed if there is one.

Table A-4 System Calls Removed Between Trusted Solaris Releases

Programming Interface	Trusted Solaris 2.5.1 Equivalent
`aclipc(2T)`	Removed.
`adituser(2T)`	Removed.
`getkernstate(2T)`	Removed.
`setkernstate(2T)`	Removed.
`getprocpriv(2T)`	`getppriv(2TSOL)`
`setprocpriv(2T)`	`setppriv(2TSOL)`
`getfilepriv(2T)`	`getfpriv(2TSOL)`
`setfilepriv(2T)`	`setfpriv(2TSOL)`
`mkfso(2T)`	Removed.
`revoke(2T)`	Removed.
`secuname(2T)`	Removed.
`setuseraudit(2)`	Removed.

Library Routines Unique to Trusted Solaris

The library routines listed in Table A-5 on page 85 were added to either the 1.2 or the 2.5 and 2.5.1 releases. The column on the right indicates whether the library routine is new to or has changed as follows:

No change for 2.5.1 – This Trusted Solaris 1.2 library routine is unchanged in the Trusted Solaris 2.5.1 release.

New in 2.5.1 – This Trusted Solaris 2.5.1 library routine is new.

Parameter list change – This Trusted Solaris 1.2 library routine is also in the Trusted Solaris 2.5.1 release, but there have been some changes to its parameter list.

Table A-5 Trusted Solaris 2.5.1 Library Routines

Library Routine	Changed from Release?
`adornfc(3TSOL)`	1.2
`auditwrite(3TSOL)`	Parameter list change. The *aw_errlist* and *aw_nerr* variables are removed.
`auth_set_to_str(3TSOL)`	New in 2.5.1 authorizations
`auth_to_str(3TSOL)`	New in 2.5.1 authorizations
`bclearhigh(3TSOL)`	No change for 2.5.1 labels
`bclearlow(3TSOL)`	No change for 2.5.1 labels
`bcleartoh(3TSOL)`	No change for 2.5.1 labels
`bcleartoh_r(3TSOL)`	New in 2.5.1.
`bcleartos(3TSOL)`	Constant parameters now in list.
`bclearundef(3TSOL)`	No change for 2.5.1 labels
`bclearvalid(3TSOL)`	No change for 2.5.1 labels
`bclhigh(3TSOL)`	No change for 2.5.1 labels
`bcllow(3TSOL)`	No change for 2.5.1 labels
`bcltobanner(3TSOL)`	Constant parameters now in list.
`bcltoh(3TSOL)`	No change for 2.5.1 labels
`bcltoh_r(3TSOL)`	New in 2.5.1. labels. Re-entrant
`bcltoil(3TSOL)`	No change for 2.5.1 labels
`bcltos(3TSOL)`	No change for 2.5.1 labels
`bcltosl(3TSOL)`	No change for 2.5.1 labels
`bclundef(3TSOL)`	No change for 2.5.1 labels
`bilconjoin(3TSOL)`	Constant parameters now in list.
`bildominates(3TSOL)`	Constant parameters now in list.
`bilequal(3TSOL)`	Constant parameters now in list.
`bilhigh(3TSOL)`	No change for 2.5.1.

Table A-5 Trusted Solaris 2.5.1 Library Routines

Library Routine	Changed from Release?
`billow(3TSOL)`	No change for 2.5.1.
`biltoh(3TSOL)`	No change for 2.5.1.
`biltoh_r(3TSOL)`	New in 2.5.1.
`biltolev(3TSOL)`	No change for 2.5.1.
`biltos(3TSOL)`	No change for 2.5.1.
`bilundef(3TSOL)`	No change for 2.5.1.
`bilvalid(3TSOL)`	No change for 2.5.1.
`bimdominates(3TSOL)`	No change for 2.5.1.
`bimequal(3TSOL)`	No change for 2.5.1.
`blcompare(3TSOL)`	No change for 2.5.1.
`bldominates(3TSOL)`	No change for 2.5.1.
`blequal(3TSOL)`	No change for 2.5.1.
`blinrange(3TSOL)`	No change for 2.5.1.
`blinset(3TSOL)`	No change for 2.5.1.
`blmaximum(3TSOL)`	Constant parameters now in list.
`blminimum(3TSOL)`	Constant parameters now in list.
`blmanifest(3TSOL)`	No change for 2.5.1.
`blminmax(3TSOL)`	No change for 2.5.1.
`blportion(3TSOL)`	No change for 2.5.1.
`blstrictdom(3TSOL)`	No change for 2.5.1.
`bltocolor(3TSOL)`	Constant parameters now in list.
`bltos(3TSOL)`	New in 2.5.1.
`bltype(3TSOL)`	Constant parameters now in list.
`blvalid(3TSOL)`	Constant parameters now in list.
`bslhigh(3TSOL)`	No change for 2.5.1.
`bsllow(3TSOL)`	No change for 2.5.1.
`bsltoh(3TSOL)`	No change for 2.5.1.

Table A-5 Trusted Solaris 2.5.1 Library Routines

Library Routine	Changed from Release?
bsltoh_r(3TSOL)	New in 2.5.1.
bsltos(3TSOL)	No change for 2.5.1.
bslundef(3TSOL)	No change for 2.5.1.
bslvalid(3TSOL)	No change for 2.5.1.
btohex(3TSOL)	New in 2.5.1.
chkauth(3TSOL)	New in 2.5.1.
endprofent(3TSOL)	New in 2.5.1.
endprofstr(3TSOL)	New in 2.5.1
enduserent(3TSOL)	New in 2.5.1
free_auth_set(3TSOL)	New in 2.5.1
free_profent(3TSOL)	New in 2.5.1
free_profstr(3TSOL)	New in 2.5.1
free_userent(3TSOL)	New in 2.5.1
fsync(3C)	New in 2.5.1
get_auth_text(3TSOL)	New in 2.5.1
get_priv_text(3TSOL)	New in 2.5.1
getcil(3TSOL)	No change for 2.5.1.
getcsl(3TSOL)	No change for 2.5.1.
getpeerinfo(3TSOL)	Replaces 1.2 implementation.
getprofent(3TSOL)	New in 2.5.1
getprofentbyname(3TSOL)	New in 2.5.1
getprofstr(3TSOL)	New in 2.5.1
getprofstrbyname(3TSOL)	New in 2.5.1
getuserent(3TSOL)	New in 2.5.1
getuserentbyname(3TSOL)	New in 2.5.1
getuserentbyuid(3TSOL)	New in 2.5.1
getvfsaent(3TSOL)	New in 2.5.1

Table A-5 Trusted Solaris 2.5.1 Library Routines

Library Routine	Changed from Release?
getvfsafile(3TSOL)	New in 2.5.1
h_alloc(3TSOL)	New in 2.5.1
h_free(3TSOL)	New in 2.5.1
hextob(3TSOL)	New in 2.5.1
htobcl(3TSOL)	Constant parameters now in list.
htobclear(3TSOL)	Constant parameters now in list.
htobil(3TSOL)	Constant parameters now in list.
htobsl(3TSOL)	Constant parameters now in list.
labelinfo(3TSOL)	No change for 2.5.1.
labelvers(3TSOL)	Constant parameters now in list.
mldgetcwd(3TSOL)	New in 2.5.1.
mldrealpath(3TSOL)	No change for 2.5.1.
mldrealpathl(3TSOL)	New in 2.5.1.
priv_set_to_str(3TSOL)	New in 2.5.1
priv_to_str(3TSOL)	No change for 2.5.1
putprofent(3TSOL)	New in 2.5.1
putprofstr(3TSOL)	New in 2.5.1
sbcleartos(3TSOL)	Constant parameters now in list.
sbcltos(3TSOL)	Constant parameters now in list.
sbiltos(3TSOL)	Constant parameters now in list.
sbsltos(3TSOL)	Constant parameters now in list.
set_effective_priv(3TSOL)	No change in 2.5.1.
set_inheritable_priv(3TSOL)	No change in 2.5.1.
set_permitted_priv(3TSOL)	No change in 2.5.1.
setbltype(3TSOL)	Constant parameters now in list.
setcil(3TSOL)	No change in 2.5.1.
setcsl(3TSOL)	No change in 2.5.1.

Table A-5 Trusted Solaris 2.5.1 Library Routines

Library Routine	Changed from Release?
setprofent(3TSOL)	New in 2.5.1
setprofstr(3TSOL)	New in 2.5.1
setuserent(3TSOL)	New in 2.5.1
stobcl(3TSOL)	Constant parameters now in list.
stobclear(3TSOL)	Constant parameters now in list.
stobil(3TSOL)	Constant parameters now in list.
stobl(3TSOL)	Constant parameters now in list.
stobsl(3TSOL)	Constant parameters now in list.
str_to_auth(3TSOL)	New in 2.5.1
str_to_auth_set(3TSOL)	New in 2.5.1
str_to_priv(3TSOL)	New structure in parameter list.
str_to_priv_set(3TSOL)	New in 2.5.1
t6alloc_blk(3NTSOL)	New in 2.5.1
t6clear_blk(3NTSOL)	New in 2.5.1
t6cmp_blk(3NTSOL)	New in 2.5.1
t6copy_blk(3NTSOL)	New in 2.5.1
t6dup_blk(3NTSOL)	New in 2.5.1
t6ext_attr(3NTSOL)	New in 2.5.1
t6free_blk(3NTSOL)	New in 2.5.1
t6get_attr(3NTSOL)	New in 2.5.1
t6get_endpt_default(3NTSOL)	New in 2.5.1
t6get_endpt_mask(3NTSOL)	New in 2.5.1
t6new_attr(3NTSOL)	New in 2.5.1
t6peek_attr(3NTSOL)	New in 2.5.1
t6recvfrom(3NTSOL)	New in 2.5.1
t6sendto(3NTSOL)	New in 2.5.1
t6set_attr(3NTSOL)	New in 2.5.1

Table A-5 Trusted Solaris 2.5.1 Library Routines

Library Routine	Changed from Release?
t6set_endpt_default(3NTSOL)	New in 2.5.1
t6size_attr(3NTSOL)	New in 2.5.1
t6last_attr(3NTSOL)	New in 2.5.1
t6set_endpt_mask(3NTSOL)	New in 2.5.1
t6ext_attr(3NTSOL)	New in 2.5.1
tsol_lbuild_create(3TSOL)	Now Motif-based
tsol_lbuild_get(3TSOL)	Now Motif-based
tsol_lbuild_set(3TSOL)	Now Motif-based
tsol_lbuild_destroy(3TSOL)	New in 2.5.1
Xbcleartos(3TSOL)	New in 2.5.1
Xbcltos(3TSOL)	New in 2.5.1
Xbiltos(3TSOL)	New in 2.5.1
Xbsltos(3TSOL)	New in 2.5.1
XTSOLIsWindowTrusted(3X11TSOL)	New in 2.5.1
XTSOLMakeTPWindow(3X11TSOL)	New in 2.5.1
XTSOLgetClientAttributes(3X11TSOL)	New in 2.5.1
XTSOLgetPropAttributes(3X11TSOL)	New in 2.5.1
XTSOLgetPropLabel(3X11TSOL)	New in 2.5.1
XTSOLgetPropUID(3X11TSOL)	New in 2.5.1
XTSOLgetResAttributes(3X11TSOL)	New in 2.5.1
XTSOLgetResLabel(3X11TSOL)	New in 2.5.1
XTSOLgetResUID(3X11TSOL)	New in 2.5.1
XTSOLgetSSHeight(3X11TSOL)	New in 2.5.1
XTSOLgetWindowIIL(3X11TSOL)	New in 2.5.1
XTSOLgetWorkstationOwner(3X11TSOL)	New in 2.5.1
XTSOLsetPolyInstInfo(3X11TSOL)	New in 2.5.1
XTSOLsetPropLabel(3X11TSOL)	New in 2.5.1

Table A-5 Trusted Solaris 2.5.1 Library Routines

Library Routine	Changed from Release?
XTSOLsetPropUID(3X11TSOL)	New in 2.5.1
XTSOLsetResLabel(3X11TSOL)	New in 2.5.1
XTSOLsetResUID(3X11TSOL)	New in 2.5.1
XTSOLsetSSHeight(3X11TSOL)	New in 2.5.1
XTSOLsetSessionHI(3X11TSOL)	New in 2.5.1
XTSOLsetSessionLO(3X11TSOL)	New in 2.5.1
XTSOLsetWindowIIL(3X11TSOL)	New in 2.5.1
XTSOLsetWorkstationOwner(3X11TSOL)	New in 2.5.1
XTSOLShutdown(3X11TSOL)	New in 2.5.1

Solaris 2.5.1 Library Routines Enhanced for Security

The Solaris 2.5.1 library routines listed in the following table were enhanced for security policy in the indicated Trusted Solaris release.

Table A-6 Modified Solaris 2.5.1 Library Routines

Library Routine	Release Modified
aclcheck(3)	base and 1.2
acltomode(3)	2.5.1
aclsort(3)	base and 1.2
acltopbits(3)	2.5.1
acltotext(3)	base and 1.2
aiocancel(3)	2.5.1
aioinit(3TSOL)	2.5.1
aionotify(3TSOL)	2.5.1
aioread(3)	2.5.1
aiostart(3TSOL)	2.5.1

Table A-6 Modified Solaris 2.5.1 Library Routines

Library Routine	Release Modified
aiowait(3)	2.5.1
aiowrite(3TSOL)	2.5.1
auditwrite(3TSOL)	2.5.1
au_preselect(3TSOL)	2.5.1
endac(3TSOL)	2.5.1
endauclass(3TSOL)	2.5.1
endauevent(3TSOL)	2.5.1
endauuser(3TSOL)	2.5.1
getacdir(3TSOL)	2.5.1
getacflg(3TSOL)	2.5.1
getacinfo(3TSOL)	2.5.1
getacmin(3TSOL)	2.5.1
getacna(3TSOL)	2.5.1
getauclassent(3TSOL)	2.5.1
getauclassent_r(3TSOL)	2.5.1
getauclassnam(3TSOL)	2.5.1
getauclassnam_r(3TSOL)	2.5.1
getauditflags(3TSOL)	2.5.1
getauditflagsbin(3TSOL)	2.5.1
getauditflagschar(3TSOL)	2.5.1
getauevent(3TSOL)	2.5.1
getauevent_r(3TSOL)	2.5.1
getauevnam(3TSOL)	2.5.1
getauevnam_r(3TSOL)	2.5.1
getauevnonam(3TSOL)	2.5.1
getauevnum(3TSOL)	2.5.1
getauevnum_r(3TSOL)	2.5.1

Table A-6 Modified Solaris 2.5.1 Library Routines

Library Routine	Release Modified
getauuserent(3TSOL)	2.5.1
getauuserent_r(3TSOL)	2.5.1
getauusernam(3TSOL)	2.5.1
getfauditflags(3TSOL)	2.5.1
reboot(3C)	2.5.1
sethostname(3C)	2.5.1
settimeofday(3C)	2.5.1
setpriority(3C)	2.5.1
send(3NTSOL)	2.5.1
sendmsg(3NTSOL)	2.5.1
sendto(3NTSOL)	2.5.1
setac(3TSOL)	2.5.1
setauclass(3TSOL)	2.5.1
setauevent(3TSOL)	2.5.1
setauuser(3TSOL)	2.5.1
connect(3N)	2.5.1
getpeername(3N)	2.5.1
getsockname(3N)	2.5.1
getsockopt(3N)	2.5.1
listen(3NTSOL)	2.5.1
recv(3N)	2.5.1
recvfrom(3XN)	2.5.1
recvmsg(3XN)	2.5.1
reject(3N)	2.5.1
setsockopt(3NTSOL)	2.5.1
shutdown(3N)	2.5.1
socket(3N)	2.5.1

Table A-6 Modified Solaris 2.5.1 Library Routines

Library Routine	Release Modified
socketpair(3N)	2.5.1
accept(3NTSOL)	2.5.1
bind(3NTSOL)	2.5.1
send(3NTSOL)	2.5.1
sendmsg(3NTSOL)	2.5.1
sendto(3NTSOL)	2.5.1
t_accept(3NTSOL)	2.5.1
t_alloc(3N)	2.5.1
t_bind(3NTSOL)	2.5.1
t_close(3N)	2.5.1
t_connect(3N)	2.5.1
t_error(3N)	2.5.1
t_free(3N)	2.5.1
t_getinfo(3N)	2.5.1
t_getstate(3N)	2.5.1
t_listen(3N)	2.5.1
t_look(3N)	2.5.1
t_open(3N)	2.5.1
t_optmgmt(3NTSOL)	2.5.1
t_rcvconnect(3N)	2.5.1
t_rcvdis(3N)	2.5.1
t_rcvrel(3N)	2.5.1
t_rcvudata(3N)	2.5.1
t_snd(3NTSOL)	2.5.1
t_snddis(3N)	2.5.1
t_sndrel(3N)	2.5.1
t_sndudata(3N)	2.5.1

Table A-6 Modified Solaris 2.5.1 Library Routines

Library Routine	Release Modified
t_strerror(3N)	2.5.1
t_sync(3N)	2.5.1
t_unbind(3N)	2.5.1
clock_settime(3RTSOL)	2.5.1

Removed Trusted Solaris 1.2 Library Routines and Functions

The Trusted Solaris 1.2 library routines listed in the following table were removed or replaced in the Trusted Solaris 2.5.1 release.

Table A-7 Removed Trusted Solaris 1.2 Library Routines

Library Routine	Module	Trusted Solaris 2.5.1 Equivalent
	label builder	labelbuilder(3TSOL)
enddaent(3T)	Devices	
enddmapent(3T)	Devices	
event_handler(3T)	label builder	
getdaent(3T)	Devices	
getdanam(3T)	Devices	
getdatype(3T)	Devices	
getdmapdev(3T)	Devices	
getmapent(3T)	Devices	
getdmapnam(3T)	Devices	
getmaptype(3T)	Devices	
setdafile(3T)	Devices	
setdmapent(3T)	Devices	
setdmapfile(3T)	Devices	
check_authorization(3T)	Authorizations	chkauth(3TSOL)
getauthorizent(3T)	Authorizations	

A

Table A-7 Removed Trusted Solaris 1.2 Library Routines

Library Routine	Module	Trusted Solaris 2.5.1 Equivalent
getauthoriznam(3T)	Authorizations	
setauthorizent(3T)	Authorizations	
setauthorizent(3T)	Authorizations	
endauthorizent(3T)	Authorizations	
fgetauthorizent(3T)	Authorizations	
tnet_attr_alloc(3T)	Network	t6alloc_blk(3NTSOL)
tnet_copy_attr(3T)	Network	
tnet_create_attr_buf(3T)	Network	
tnet_def_attrs(3T)	Network	
tnet_dup_attr(3T)	Network	t6dup_blk(3NTSOL)
tnet_ext_attrs(3T)	Network	t6ext_attr(3NTSOL)
tnet_last_attrs(3T)	Network	
tnet_new_attrs_only(3T)	Network	t6new_attr(3NTSOL)
tnet_peek(3T)	Network	t6peek_attr(3NTSOL)
tnet_receive(3T)	Network	t6recvfrom(3NTSOL)
tnet_recv_attrs(3T)	Network	
tnet_send(3T)	Network	t6sendto(3NTSOL)
tnet_send_oob(3T)	Network	
tnet_tcb(3T)	Network	
XCMWperformAuthentication	X Protocol	
XCMWsetTPathIndicator	X Protocol	
XCMWregisterTrustedClient	X Protocol	
XCMWgetPropLabel	X Protocol	XTSOLgetPropLabel(3X11TSOL)
XCMWgetPropUID	X Protocol	XTSOLgetPropUID(3X11TSOL)
XCMWgetResLabel	X Protocol	XTSOLgetResLabel(3X11TSOL)
XCMWgetResUID	X Protocol	XTSOLgetResUID(3X11TSOL)
XCMWgetSSHeight	X Protocol	XTSOLgetSSHeight(3X11TSOL)

Table A-7 Removed Trusted Solaris 1.2 Library Routines

Library Routine	Module	Trusted Solaris 2.5.1 Equivalent
XCMWgetWindowIIL	X Protocol	XTSOLgetWindowIIL(3X11TSOL)
XCMWgetWorkstationOwner	X Protocol	XTSOLgetWorkstationOwner(3X11TSOL)
XCMWsetPropLabel	X Protocol	XTSOLsetPropLabel(3X11TSOL)
XCMWsetPropUID	X Protocol	XTSOLsetPropUID(3X11TSOL)
XCMWsetResLabel	X Protocol	XTSOLsetResLabel(3X11TSOL)
XCMWsetResUID	X Protocol	XTSOLsetResUID(3X11TSOL)
XCMWsetSSHeight	X Protocol	XTSOLsetSSHeight(3X11TSOL)
XCMWsetSessionHI	X Protocol	XTSOLsetSessionHI(3X11TSOL)
XCMWsetSessionLO	X Protocol	XTSOLsetSessionLO(3X11TSOL)
XCMWsetWindowIIL	X Protocol	XTSOLsetWindowIIL(3X11TSOL)
XCMWsetWorkstationOwner	X Protocol	XTSOLsetWorkstationOwner(3X11TSOL)
XCMWgetAuthenticationStatus	X Protocol	
XCMWreadTrustedMessage	X Protocol	
XCMWsendTrustedMessage	X Protocol	
XCMWsetTWinCMWLabel	X Protocol	
XCMWsetTWinIIL	X Protocol	

A

Privileges and Authorizations Transition

B

There is not a one-to-one correspondence between Trusted Solaris 1.2 and Trusted Solaris 2.5.1 privileges and authorizations. The libraries have been moved and renamed and descriptions have been added.

The following table lists the major topics for this chapter.

B

Location of Privilege and Authorization Information

	Trusted Solaris 1.2 Release	Trusted Solaris 2.5.1 Release
Authorizations		
declaration	`/usr/include/cmw/authoriz.h`	`/usr/include/tsol/auth_names.h`
description		`/usr/lib/tsol/locale/C/auth_name`
		`auth_desc(4TSOL)`
Privileges		
declaration	`/usr/include/cmw/priv_name.h`	`/usr/include/sys/tsol/priv_names.h`
description		`/usr/lib/tsol/locale/C/priv_name`
		`priv_desc(4TSOL)`

Changes to Privileges

Trusted Solaris 2.5.1 privileges offer fine-grained control over security. Where the underlying system has changed, as in network and windows, their privileges have changed from the Trusted Solaris 1.2 implementation. The Trusted Solaris 1.2 mapping between privilege names and privilege IDs has not been preserved.

Table B-1 on page 101 lists the privileges that have been retained between the releases.

Table B-2 on page 102 lists the Trusted Solaris 1.2 privileges that have been removed, and Trusted Solaris 2.5.1 equivalents for the privileges, if any.

Table B-3 on page 103 lists the privileges that have been added in the Trusted Solaris 2.5.1 release.

Retained Privileges

Table B-1 Privileges Retained Between the Two Releases

Privileges Present in Both Releases	
PRIV_FILE_CHOWN	PRIV_FILE_MAC_WRITE
PRIV_FILE_DAC_EXECUTE	PRIV_FILE_NOFLOAT
PRIV_FILE_DAC_READ	PRIV_FILE_OWNER
PRIV_FILE_DAC_SEARCH	PRIV_FILE_SETDAC
PRIV_FILE_DAC_WRITE	PRIV_FILE_SETID
PRIV_FILE_DOWNGRADE_IL	PRIV_FILE_MAC_SEARCH
PRIV_FILE_DOWNGRADE_SL	PRIV_FILE_SETPRIV
PRIV_FILE_LOCK	PRIV_FILE_UPGRADE_IL
PRIV_FILE_MAC_READ	PRIV_FILE_UPGRADE_SL
PRIV_IPC_DAC_READ	PRIV_IPC_MAC_WRITE
PRIV_IPC_DAC_WRITE	PRIV_IPC_NOFLOAT
PRIV_IPC_MAC_READ	PRIV_IPC_OWNER
PRIV_NET_BROADCAST	PRIV_NET_SETID
PRIV_NET_PRIVADDR	PRIV_NET_SETPRIV
PRIV_NET_RAWACCESS	
PRIV_PROC_AUDIT_APPL	PRIV_PROC_NOFLOAT
PRIV_PROC_AUDIT_TCB	PRIV_PROC_OWNER
PRIV_PROC_CHROOT	PRIV_PROC_SETCLR
PRIV_PROC_DUMPCORE	PRIV_PROC_SETID
PRIV_PROC_MAC_READ	PRIV_PROC_SETIL
PRIV_PROC_MAC_WRITE	PRIV_PROC_SETSL
PRIV_PROC_NODELAY	PRIV_PROC_TRANQUIL
PRIV_SYS_AUDIT	PRIV_SYS_MINFREE
PRIV_SYS_BOOT	PRIV_SYS_MOUNT
PRIV_SYS_CONSOLE	PRIV_SYS_NET_CONFIG

B

Table B-1 Privileges Retained Between the Two Releases

Privileges Present in Both Releases	
PRIV_SYS_DEVICES	PRIV_SYS_NFS
PRIV_SYS_FS_CONFIG	PRIV_SYS_SUSER_COMPAT
PRIV_SYS_IPC_CONFIG	PRIV_SYS_TRANS_LABEL
PRIV_SYS_MAXPROC	

Removed Privileges

Table B-2 Privileges Removed Between the Two Releases

In Trusted Solaris 1.2 but not 2.5.1	Equivalents
PRIV_IPC_SETIL	PRIV_IPC_UPGRADE_IL PRIV_IPC_DOWNGRADE_IL
PRIV_NET_ALLOWACCESS	
PRIV_NET_BOOT	
PRIV_NET_MAC_OVERRIDE	
PRIV_NET_MAC_WRITE	
PRIV_NET_NOAUTH	
PRIV_NET_SESSION	
PRIV_NET_SETIL	
PRIV_PROC_PTRACE	None required, because process tracing is now subject to policy.
PRIV_SYS_ACCT	
PRIV_SYS_DIRLINK	
PRIV_SYS_NICE	
PRIV_SYS_RESOURCE	
PRIV_SYS_REVOKE	Removed in release 1.2.
PRIV_SYS_TIME	
PRIV_SYS_TRUSTED_PATH	The trusted path is now a process attribute flag.

Table B-2 Privileges Removed Between the Two Releases

In Trusted Solaris 1.2 but not 2.5.1	Equivalents
PRIV_WIN_ADMIN	See Table B-3 for a list of the new windows privileges.
PRIV_WIN_BYPASS_SEL_AGNT	
PRIV_WIN_MOWNER	
PRIV_WIN_MSL	
PRIV_WIN_NEWS	
PRIV_WIN_ROWNER	
PRIV_WIN_RSL	
PRIV_WIN_SETIL	

New Trusted Solaris 2.5.1 Privileges

For complete descriptions of the privileges, see the file `/usr/lib/tsol/locale/C/priv_name` or the `priv_desc(4TSOL)` man page.

For how to use privileges in code, see the *Trusted Solaris Developer's Guide*.

For how to test privileges in applications, see "Privilege Debugging" on page 41.

Table B-3 New Trusted Solaris 2.5.1 Privileges

New Trusted Solaris 2.5.1 Privileges	Notes
PRIV_FILE_AUDIT	
PRIV_IPC_DOWNGRADE_IL	Replacements for PRIV_IPC_SETIL
PRIV_IPC_UPGRADE_IL	
PRIV_NET_DOWNGRADE_IL	The added network privileges manage information labels, information label floating, and bypass mandatory access control (MAC) checks on data transmitted across the network.
PRIV_NET_DOWNGRADE_SL	
PRIV_NET_MAC_READ	
PRIV_NET_NOFLOAT	A combination of *net_mac_read* and *net_reply_equal* allows unmodified programs to successfully receive and reply at all sensitivity labels.
PRIV_NET_REPLY_EQUAL	
PRIV_NET_SETCLR	

Table B-3 New Trusted Solaris 2.5.1 Privileges

New Trusted Solaris 2.5.1 Privileges	Notes
PRIV_SYS_CONFIG	
PRIV_WIN_COLORMAP	The addition of Trusted X11 window server to Trusted Solaris 2.5.1 causes a major change in the window privileges to enable access to X11 resources, colormaps, and font paths, and to control information labels, information label floating, and mandatory access controls (MAC).
PRIV_WIN_CONFIG	
PRIV_WIN_DAC_READ	
PRIV_WIN_DAC_WRITE	
PRIV_WIN_DEVICES	
PRIV_WIN_DGA	
PRIV_WIN_DOWNGRADE_IL	
PRIV_WIN_DOWNGRADE_SL	
PRIV_WIN_FONTPATH	
PRIV_WIN_MAC_READ	
PRIV_WIN_MAC_WRITE	
PRIV_WIN_NOFLOAT	
PRIV_WIN_SELECTION	
PRIV_WIN_UPGRADE_IL	
PRIV_WIN_UPGRADE_SL	

Changes to Authorizations

The Trusted Solaris 2.5.1 environment has replaced the Trusted Solaris 1.2 administrative authorizations with a more extensive set. Most administrative tasks require a specific authorization.

Authorization administration is no longer in a file. Authorizations are part of an execution profile. Execution profiles are provided; more can be constructed in the Profile Manager. Execution profiles are assigned to users in the User Manager; upon login, a user has all the authorizations in all of the execution profiles assigned to the user. Authorizations that administrative roles require have been placed in the roles' execution profiles.

The security administrator (`secadmin` role) can assign a regular user an authorization:

- by assigning the user an administrative role whose profiles include the authorization,
- by assigning the user an existing execution profile that contains the authorization, or
- by creating one or more execution profiles with the authorizations appropriate for that user, then assigning the profiles.

Users get all the authorizations associated with their execution profiles upon login. Users who have been assigned a role get all the authorizations associated with that role's execution profiles when they assume the role. Table B-6 on page 106 lists the authorizations available in the Trusted Solaris 2.5.1 environment.

Replacements for Trusted Solaris 1.2 Authorizations

In the Trusted Solaris 1.2 environment there were two types of authorizations: administrative and user. Administrative authorizations allowed users to assume any of four different administrative roles. The following table shows the default Trusted Solaris 1.2 role authorizations and their Trusted Solaris 2.5.1 equivalents.

Table B-4 Replacements for Trusted Solaris 1.2 Default TFM_ROLE Authorizations

Trusted Solaris 1.2 Authorization	**Trusted Solaris 2.5.1 Equivalent**
TFM_ROLE_admin	`admin` role assigned in the User Manager
TFM_ROLE_isso	`secadmin` role assigned in the User Manager
TFM_ROLE_oper	`oper` role assigned in the User Manager
TFM_ROLE_root	`root` role assigned in the User Manager

Trusted Solaris 1.2 user authorizations allowed some users to perform security-relevant tasks without assuming an administrative role. The authorizations were declared in `/usr/include/cmw/authoriz.h`; no descriptions were provided. The Trusted Solaris 1.2 mapping between authorization names and authorization IDs has not been preserved.

Table B-5 shows the user authorizations and their equivalents. See Table B-6 on page 106 for a complete list of Trusted Solaris 2.5.1 authorizations. The file `/usr/lib/tsol/locale/C/auth_name` contains descriptions, as does the `auth_desc(4TSOL)` man page.

Table B-5 Replacements for Trusted Solaris 1.2 Default User Authorizations

Trusted Solaris 1.2 Authorization	Trusted Solaris 2.5.1 Equivalent
SunCMW_boot_system	TSOL_AUTH_ENABLE_LOGIN
SunCMW_terminal_login	TSOL_AUTH_TERMINAL_LOGIN
SunCMW_remote_login	TSOL_AUTH_REMOTE_LOGIN
SunCMW_downgrade_sensitivity_label	TSOL_AUTH_FILE_DOWNGRADE_SL TSOL_AUTH_WIN_DOWNGRADE_SL
SunCMW_outside_accreditation_range	TSOL_AUTH_SYS_ACCRED_SET
SunCMW_set_single_level_device	TSOL_AUTH_ALLOCATE
SunCMW_upgrade_sensitivity_label	TSOL_AUTH_FILE_UPGRADE_SL TSOL_AUTH_WIN_UPGRADE_SL

Table B-6 Trusted Solaris 2.5.1 Authorizations

Authorization Type	Trusted Solaris 2.5.1x Authorization
Administrative Tools Each administrative tool authorization controls a button in the Solstice_Apps User Manager. Roles that are not authorized to change information see a grayed button.	TSOL_AUTH_USER_IDENT TSOL_AUTH_USER_PASSWORD TSOL_AUTH_USER_SELF TSOL_AUTH_USER_LABELS TSOL_AUTH_USER_AUDIT TSOL_AUTH_USER_PROFILES TSOL_AUTH_USER_IDLE TSOL_AUTH_USER_ROLES TSOL_AUTH_USER_HOME
Database Control Each database control authorization controls a button in the Solstice_Apps Database Manager. Roles that are not authorized to change information see a grayed button.	TSOL_AUTH_DB_ALIASES TSOL_AUTH_DB_AUTO_HOME TSOL_AUTH_DB_BOOTPARAMS TSOL_AUTH_DB_ETHERS TSOL_AUTH_DB_GROUP

Table B-6 Trusted Solaris 2.5.1 Authorizations

Authorization Type	Trusted Solaris 2.5.1x Authorization
	TSOL_AUTH_DB_HOSTS
	TSOL_AUTH_DB_LOCALE
	TSOL_AUTH_DB_NETGROUP
	TSOL_AUTH_DB_NETMASKS
	TSOL_AUTH_DB_NETWORKS
	TSOL_AUTH_DB_PASSWD
	TSOL_AUTH_DB_PROTOCOLS
	TSOL_AUTH_DB_RPC
	TSOL_AUTH_DB_SERVICES
	TSOL_AUTH_DB_TIMEZONE
	TSOL_AUTH_DB_TNIDB
	TSOL_AUTH_DB_TNRHDB
	TSOL_AUTH_DB_TNRHTP
Device Control	TSOL_AUTH_ALLOCATE
	TSOL_AUTH_SHUTDOWN
	TSOL_AUTH_CONFIG_DEVICE
	TSOL_AUTH_REVOKE_DEVICE
File Control	TSOL_AUTH_FILE_AUDIT
	TSOL_AUTH_FILE_DOWNGRADE_SL
	TSOL_AUTH_FILE_UPGRADE_SL
	TSOL_AUTH_FILE_OWNER
	TSOL_AUTH_FILE_CHOWN
	TSOL_AUTH_FILE_SETPRIV
File Management	TSOL_AUTH_BYPASS_FILE_VIEW
Label Control	TSOL_AUTH_SYS_ACCRED_SET
Login Control	TSOL_AUTH_ENABLE_LOGIN
	TSOL_AUTH_REMOTE_LOGIN

Table B-6 Trusted Solaris 2.5.1 Authorizations

Authorization Type	Trusted Solaris 2.5.1x Authorization
	TSOL_AUTH_TERMINAL_LOGIN
`at` Job Control	TSOL_AUTH_AT_ADMIN
	TSOL_AUTH_AT_USER
`cron` Job Control	TSOL_AUTH_CRON_ADMIN
	TSOL_AUTH_CRON_USER
Print Control	TSOL_AUTH_PRINT_ADMIN
	TSOL_AUTH_PRINT_NOBANNER
	TSOL_AUTH_PRINT_POSTSCRIPT
	TSOL_AUTH_PRINT_UNLABELED
Window Control	TSOL_AUTH_WIN_DOWNGRADE_SL
	TSOL_AUTH_WIN_UPGRADE_SL
	TSOL_AUTH_OCCUPY_WORKSPACE

Index

D

E

R

S

T

U

W

X