Solaris 8
Security Target

Issue:	1.0
Date:	28 July 2000
Reference:	S8.0_101 / ts2_101
Author:	David Lodge
Status:	Definitive

Abstract: This document is the Security Target for the EAL4 Common Criteria v2.1 evaluation of Solaris 8 developed by Sun Microsystems, Inc.

This document was prepared by:

Logica CLEF (LFL),
Logica UK Limited,
Cobham Park,
Downside Road,
Cobham,
Surrey.
KT11 3LG

on behalf of:

SUN Microsystems, Inc.
901 San Antonio Road,
Palo Alto,
CA 94303-4900
USA

Contents

References

Standards & Criteria

[CC] Common Criteria for Information Technology Security Evaluation, CCIMB-99-031, Version 2.1, August 1999

[CAPP] Controlled Access Protection Profile, Issue 1.d, 8 October 1999

[ST-ITSEC] Solaris 2.6SE Security Target
Ref: S2.6_101, Version 1.0, 17/06/98.

This Page Intentionally Left Blank

1 Introduction

1.1 ST Identification

Title: Solaris 8 Security Target

Keywords: Solaris 8, general-purpose operating system, POSIX, UNIX.

This document is the security target for the CC evaluation of the Solaris 8 operating system product, and is conformant to the Common Criteria for Information Technology Security Evaluation [CC].

1.2 ST Overview

This security target documents the security characteristics of the Solaris 8 operating system.

Solaris is a highly-configurable UNIX-based operating system which has been developed to meet the requirements of the C2 class of the U.S. Department of Defence (DoD) Trusted Computer System Evaluation Criteria (TCSEC), including the use of Access Control Lists. These broad requirements are described for the Common Criteria scheme in [CAPP], the Controlled Access Protection Profile.

A Solaris 8 system consists of a number of workstations and servers linked together to form a single distributed system. Users share the resources of multiple workstations and servers connected together in a single, distributed Trusted Computing Base (TCB).

1.3 CC Conformance

This ST is conformant to the Controlled Access Protection Profile version 1.d [CAPP].

This ST is CC *Part 2 extended* and *Part 3 conformant*, with a claimed Evaluation Assurance Level of EAL4 (see section 7.3.3).

1.4 Structure

The structure of this document is as defined by [CC] Part 1 Annex C.

- Section 2 is the TOE Description.
- Section 3 provides the statement of TOE security environment.
- Section 4 provides the statement of security objectives.
- Section 5 provides the statement of IT security requirements.
- Section 6 provides the TOE summary specification, which includes the detailed specification of the IT Security Functions.
- Section 7 provides the rationale for the security objectives, security requirements, TOE summary specification and PP claims against [CAPP].

1.5 Terminology

This section contains definitions of technical terms that are used with a meaning specific to this document. Terms defined in the [CC] are not reiterated here, unless stated otherwise.

Administrative User: This term refers to an administrator of a Solaris 8 system. Some administrative tasks requires use of the *root* username and password so that they can become the superuser (with a user ID of 0) while other tasks can be performed by specified users only.

Authentication data: This includes a user identifier, password and authorisations for each user of the product.

Object: In Solaris 8, objects belong to one of four categories: file system objects, other kernel objects (such as processes, programs and interprocess communication), window system objects and miscellaneous objects.

Product: The term product is used to define all hardware and software components that comprise the distributed Solaris 8 system.

Public object: A type of object for which all subjects have read access, but only the TCB has write access.

Role: A role represents a set of actions that an authorised user, upon assuming the role, can perform.

Security Attributes: As defined by functional requirement FIA_ATD.1, the term 'security attributes' includes the following as a minimum: user identifier; group memberships; user authentication data; and security-relevant roles.

Subject: There are two classes of subjects in Solaris 8:

- untrusted internal subject - this is a Solaris 8 process running on behalf of some user, running outside of the TCB (for example, with no privileges).
- trusted internal subject - this is a Solaris 8 process running as part of the TCB. Examples are service daemons and the processes implementing the windowing system.

System: Includes the hardware, software and firmware components of the Solaris 8 product which are connected/networked together and configured to form a usable system.

Target of Evaluation (TOE): The TOE is defined as the Solaris 8 operating system, running and tested on the hardware and firmware specified in this Security Target. The BootPROM firmware forms part of the TOE Environment (see section 5.4).

User: Any individual/person who has a unique user identifier and who interacts with the Solaris 8 product.

1.6 Document Layout

IT security functions are assigned a unique reference identifier of the form Name.1 to enable ease of reference. For example, DAC.1, Audit.1.

Naming has been preserved from [ST-ITSEC] in order to preserve consistency and traceability. ITSFs added from [ST-ITSEC] have been assigned a new identifier, consistent with existing naming conventions. Most of the IT SFs are identical to those contained in [ST-ITSEC], the Security Target for the previous ITSEC E3 evaluation of Solaris 2.6SE.

This Page Intentionally Left Blank

2 TOE Description

2.1 Introduction

The TOE description aims to aid the understanding of the TOE's security requirements and provides a context for the evaluation. It defines the scope and boundaries of the TOE, both physically and logically, and describes the environment into which the TOE will fit.

2.2 Intended Use

Solaris 8 is a highly-configurable UNIX-based operating system which has been developed to meet "System High" Operation including the use of Access Control Lists;

A Solaris 8 system consists of a number of workstations and servers linked together to form a single distributed system. Users share the resources of multiple workstations and servers connected together in a single, distributed Trusted Computing Base (TCB).

2.3 Evaluated Configurations

2.3.1 Target of Evaluation

This section defines the software that comprise the ToE and the Workstations/ Servers and Peripherals that the software runs on.

2.3.1.1 Workstations/Servers

The target of evaluation is a (distributed) operating system product running on:

- the Sun SPARCstation, using the Ultra-I and Ultra-II family of processors[1]; and

1. The range includes UltraSPARC workstation (from the Ultra 1 through to the Ultra 80), workgroup servers (E250, E450 etc) and Midrange servers (E3500, E6500 etc) using one or more UltraSPARC I, II or IIi processors. The High End servers (E10000 based) are not included due to the domain separation capability. Non UltraSPARC processor machines such as the SparcStation 20 using superSparc processor are excluded.

- 'IBM compatible' PCs using the Intel Pentium II/III family of processors.

Each workstation requires a minimum of 96MByte of RAM, and a colour bitmap monitor. For diskfull configurations a SCSI disk of at least 2GB is required, whilst for diskless machines an attached disk is not required (although the machine needs to be connected to a SPARCsystem acting as a boot server via Ethernet).

2.3.1.2 Software

The Target of Evaluation is based on the following system software:

- Solaris 8, First Customer Shipment (FCS), February 2000.
- AdminSuite, First Customer Shipment (FCS), verison 3.0.1
- The following patches: 108875-07, 108879-02 for Sparc and 108876-07, 108881-02 for Intel.

The TOE documentation is supplied on CD-ROM.

2.3.2 File systems

The following filesystem types are supported:

- the standard Solaris UNIX filesystem, `ufs`, without the Trusted Solaris attributes;
- the standard remote filesystem access protocol, `nfs` (V2 and V3);
- the MS-DOS formatted filesystem `pcfs`; and
- the High Sierra filesystem for CD-ROM drives, `hsfs`.

In addition to the above file systems a number of "internal" filesystems are supported:

- The file descriptor file system, `fd`, allows programs to access their own file descriptors through the file name space, such as `/dev/stdin` corresponding to `/dev/fd0`.
- The names file system, `namefs` (or `namfs`) allows the arbitrary mounting of any file descriptor on top of another file name.

- The doors file system, `doorfs` allows fast control transfer between processes on the same machine.

- The process file system, `procfs` (`/proc`), provides access to the process image of each process on the machine as if the process were a "file". Process access decisions are enforced by DAC attributes inferred from the underlying process' DAC attributes.

2.3.3 Configurations

The evaluated configurations are defined as follows.

- any of the product installation configurations may be selected with the exception of the 'core system support' option which does not provide the required windowing environment;

- the CDE windowing environment must be used in preference to OpenWindows;

- role based access control features are not completely implemented in the product and in general, must not be used. The one exception to this in the configuration of AdminSuite which does use role based access. In this case, only non-root user accounts must be used to provide authorisation to this facility.

- Solaris 8 supports the use of IPv4 and IPv6;

- support for DHCP is not included;

- 32 bit and 64 bit architectures are included (32 bit only for Intel);

- Web Based Enterprise Management Services (WBEM) are not included;

- both network and CD installations are supported;

- the default configuration for identification and authentication only. Support for other authentication options e.g. smartcard authentication, is not included in the evaluation configuration;

- if the system console is used, it must be connect directly to the worksation and afforded the same physical protection as the workstation.

The product comprises one or more of the above listed workstations (and optional peripherals) running the above listed system software (a workstation running the above listed software is referred to as a "TOE workstation" below).

If the product is configured with more than one TOE workstation, they are linked by Ethernet LANs, which may be joined by bridges/routers or by TOE workstations which act as routers/gateways.

No other processors may be connected to the Ethernet network, except as noted below.

If the product is configured with more than one TOE workstation, then the NIS+ service must be used and any NIS+ server(s) must be TOE workstations.

.

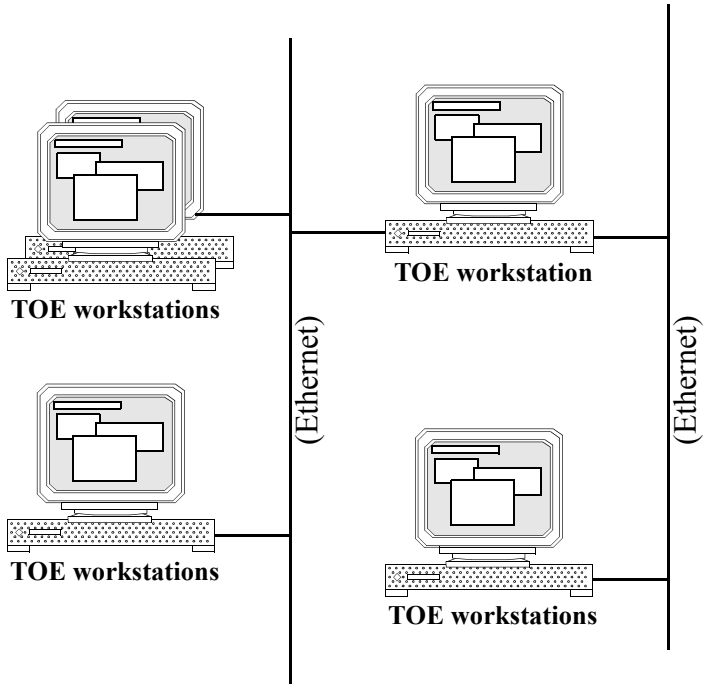

Table 1: Typical Evaluation Configuration

2.4 Summary of Security Features

The primary security features of the product are:

- Discretionary Access Control, supported by object reuse functionality;
- Identification and Authentication; and
- Auditing.

These primary security features are supported by domain separation and reference mediation, which ensure that the features are always invoked and cannot be bypassed.

2.4.1 DAC

Discretionary Access Control (DAC) restricts access to objects, such as files and is based on Access Control Lists (ACLs) and the standard UNIX permissions for user, group and other users.

2.4.2 Identification and Authentication

Solaris 8 provides identification and authentication based upon user passwords.

2.4.3 Auditing

Solaris 8 can collect extensive auditing information about security related actions taken or attempted by users, ensuring that users are accountable for their actions. For each such action or event an audit record is generated containing: date & time of the event, user, security attributes and success or failure. This audit trail can be analysed to identify attempts to compromise security and determine the extent of the compromise.

This Page Intentionally Left Blank

3 TOE Security Environment

3.1 Introduction

The statement of TOE security environment describes the security aspects of the environment in which the TOE is intended to be used and the manner in which it is expected to be employed.

To this end, the statement of TOE security environment identifies the lists the assumptions made on the operational environment (including physical and procedural measures) and the intended method of use of the for the product, defines the the threats that the product is designed to counter, and the organisational security policies with which the product is designed to comply.

3.2 Threats

The assumed security threats are listed below.

The **IT assets** to be protected comprise the information stored, processed or transmitted by the TOE. The term “information” is used here to refer to all data held within a workstation, including data in transit between workstations.

The TOE counters the general threat of unauthorised access to information, where “access” includes disclosure, modification and destruction.

The **threat agents** can be categorised as either:

- unauthorised users of the TOE, i.e. individuals who have not been granted the right to access the system; or
- authorised users of the TOE, i.e. individuals who have been granted the right to access the system.

The threat agents are assumed to originate from a well managed user community in a non-hostile working environment, and hence the product protects against threats of inadvertent or casual attempts to breach the system security. The TOE is not intended to be applicable to circumstances in which protection is required against determined attempts by hostile and well funded attackers to breach system security.

The threats listed below are grouped according to whether or not they are countered by the TOE. Those that are not countered by the TOE are countered by environmental or external mechanisms.

3.2.1 Threats countered by the TOE

[T.ACCESS_INFO] An authorised user of the TOE accesses information without having permission from the person who owns, or is responsible for, the information.

In this context 'access' is to be interpreted as observing information for which the user has no 'need to know', even though that user may have sufficient clearance to see the information.

[T.ACCESS_TOE] An unauthorised user of the TOE gains access to the system, thereby gaining unauthorised access to information.

An unauthorised user of the TOE could gain access to the system by impersonating an authorised user, or by gaining access to an unattended workstation at which an authorised user is logged on. Failure to detect the fact that an attack is taking place, or that many attempts have taken place over a period of time, may result in the attack eventually succeeding, resulting in the attacker gaining unauthorised access to information.

[T.MODIFY] Unauthorised modification or destruction of information by an authorised user of the TOE.

In this context 'unauthorised' means not having the explicit or implicit permission of the designated owner of the information.

[T.ADMIN_RIGHTS] Unauthorised use of facilities which require administration rights by an authorised user of the TOE.

Unauthorised use of such facilities by a user who cannot be trusted not to misuse them (whether intentionally or accidentally) could be exploited to gain unauthorised access to information.

3.2.2 Threats to be countered by measures within the TOE environment

The following threats apply in environments where specific threats to distributed systems need to be countered.

[T.TRANSIT] Data transferred between workstations is disclosed to or modified by unauthorised users or processes either directly or indirectly (e.g. through spoofing of workstation identity).

3.3 Organisational Security Policies

The TOE complies with the following organisational security policies:

[P.AUTH] Only those users who have been authorised to access the information within the system may access the system.

[P.DAC] The right to access specific data objects is determined on the basis of:

a) the owner of the object; and

b) the identity of the subject attempting the access; and

c) the implicit and explicit access rights to the object granted to the subject by the object owner.

[P.ACCOUNTABLE] The users of the system shall be held accountable for their actions within the system.

3.4 Assumptions

This section indicates the minimum physical and procedural measures required to maintain security of the Solaris 8 product. It is not a complete list, as specific measures may be required for different configurations and sites.

3.4.1 Physical Aspects

[A.PROTECT] It is assumed that all software and hardware, including network and peripheral cabling is approved for the transmittal of the most sensitive data held by the system. Such items are assumed to be physically protected against threats to the confidentiality and integrity of the data transmitted.

3.4.2 Personnel Aspects

[A.ADMIN] It is assumed that there are one or more competent individuals who are assigned to manage the TOE and the security of the information it contains.

Such personnel are assumed not to be careless, wilfully negligent or hostile.

3.4.3 Connectivity Aspects

[A.NIS_DOMAINS] It is assumed that, if the product comprises more than one workstation, all workstations are administered from a central point within each NIS+ domain.

NIS+ allows the creation of multiple administrative domains, thus allowing administrators to control local resources and user accounts, yet making it possible for users and resources to operate seamlessly over the entire organisation.

[A.BRIDGES&ROUTERS] All bridges and routers are assumed to correctly pass data without modification.

4 Security Objectives

4.1 Security Objectives for the TOE

[O.AUTHORISATION] The TOE must ensure that only authorised users gain access to the TOE and its resources.

[O.DAC] The TOE must provide its users with the means of controlling and limiting access to the objects and resources they own or are responsible for, on the basis of individual users or identified groups of users, and in accordance with the set of rules defined by the P.DAC security policy.

[O.AUDIT] The TOE must provide the means of recording any security relevant events, so as to:

a) assist an administrator in the detection of potential attacks or misconfiguration of the TOE security features that would leave the TOE susceptible to attack; and

b) hold users accountable for any actions they perform that are relevant to security.

[O.RESIDUAL_INFO] The TOE must ensure that any information contained in a protected resource is not released when the resource is recycled.

[O.MANAGE] The TOE must allow administrators to effectively manage the TOE and its security functions, and must ensure that only authorised administrators are able to access such functionality.

[O.ENFORCEMENT] The TOE security policy is enforced in a manner which ensures that the organisational policies are enforced in the target environment i.e. the integrity of the TSF is protected.

4.2 Security Objectives for the TOE Environment

[O.ADMIN] Those responsible for the TOE are competent and trustworthy individuals, capable of managing the TOE and the security of the information it contains.

[O.ACCOUNTABLE]Those responsible for the TOE must ensure that:

a) The product is configured such that only the approved group of users for which the system was accredited may access the system.

b) Each individual user is assigned a unique user ID.

[O.AUDITDATA] Those responsible for the TOE must ensure that the audit functionality is used and managed effectively. In particular:

a) Procedures must exist to ensure that the audit trail for the product (i.e., all networked components containing an audit trail) is regularly analysed and archived, to allow retrospective inspection.

b) The auditing system must be configured such that the loss of audit data is minimised upon:

- planned or unplanned shutdown; or
- lack of available audit storage (in particular administrators should ensure that the AUDIT_CNT flag is correctly set as identified in the Administration documentation supplied with the TOE, and that remote partitions are mounted with the appropriate option [noac] so that audit information is not lost when the partition fills).

c) The auditing system must be configured such that bad authentication data will not be stored in the audit trail (in particular, administrators should ensure that the PASSWD flag is correctly set as identified in the Administration documentation supplied with the TOE).

d) The media on which audit data is stored must not be physically removable from the workstation by unauthorised users.

[O.AUTHDATA] Those responsible for the TOE must ensure that user authentication data is stored securely and not disclosed to unauthorised individuals. In particular:

a) Procedures must be established to ensure that user passwords generated by an administrator during user account creation or modification are distributed in a secure manner, as appropriate for the clearance of the system.

b) The media on which authentication data is stored must not be physically removable from the workstation by unauthorised users.

c) Users must not disclose their passwords to other individuals.

[O.BOOT] Hardware and firmware within the IT environment shall ensure that the correct copy of the Solaris 8 operating system is "booted" during system start-up.

Note: The above applies to Sparc workstations and servers. For Intel platforms, the above may be achieved through the PC BIOS (i.e. firmware), but administrators should also take precautions to prevent booting from the floppy drive, CD device or over the network where this is considered a threat.

[O.CONSISTENCY] Administrators of the TOE must establish and implement procedures to ensure the consistency of the security-related data across all distributed components that are networked to form a single system (e.g. authentication data). In particular, if the product comprises more than one workstation, all such workstations are administered from a central point within each NIS+ domain.

[O.INSTALL] Those responsible for the TOE must establish and implement procedures to ensure that the hardware, software and firmware components that comprise the networked product are distributed, installed and configured in a secure manner.

[O.INFO_PROTECT]Those responsible for the TOE must establish and implement procedures to ensure that information is protected in an appropriate manner. In particular:

a) DAC protections on security critical files (such as audit trails and authentication databases) shall always be set up correctly.

b) All network and peripheral cabling must be approved for the transmittal of the most sensitive data held by the system. Such physical links are assumed to be adequately protected against threats to the confidentiality and integrity of the data transmitted.

[O.MAINTENANCE]Administrators of the TOE must ensure that the comprehensive diagnostics facilities provided by the product are invoked at every scheduled preventative maintenance period.

[O.RECOVER] Those responsible for the TOE must ensure that procedures and/or mechanisms are provided to assure that, after system failure or other discontinuity, recovery without a protection (i.e., security) compromise is obtained.

[O.SOFTWARE_IN] Those responsible for the TOE shall ensure that the system shall be configured so that only an administrator can introduce new software into the system.

[O.SERIAL_LOGIN] Those responsible for the TOE shall implement procedures to ensure that users clear the screen before logging off where serial login devices (e.g.VT100) are used.

The following security objective applies in environments where specific threats to distributed systems need to be countered, as described in section 3. Typically this objective is met by cryptographic protection of network connections.

[O.PROTECT] Those responsible for the TOE must ensure that procedures and/ or mechanisms exist to ensure that data transferred between workstations is secured from disclosure, interruption or tampering.

5 Security Requirements

5.1 TOE Security Functional Requirements

5.1.1 Requirements Taken from Protection Profile(s)

The security functional requirements for the TOE are as defined in [CAPP] with refinements to SFRs that are left to the security target author. The following table lists the classes, families, components and elements defined in [CAPP]. These all apply to the TOE, but the elements that are to be tailored for this security target are indicated by a * after the element's name.

CLASS	FAMILY	COMPONENT	ELEMENT	[CAPP] paragraph
FAU	FAU_GEN	FAU_GEN.1	FAU_GEN.1.1 FAU_GEN.1.2	5.1.1.1 5.1.1.2
		FAU_GEN.2	FAU_GEN.2.1	5.1.2.1
	FAU_SAR	FAU_SAR.1	FAU_SAR.1.1 FAU_SAR.1.2	5.1.3.1 5.1.3.2
		FAU_SAR.2	FAU_SAR.2.1	5.1.4.1
		FAU_SAR.3	FAU_SAR.3.1*	5.1.5.1
	FAU_SEL	FAU_SEL.1	FAU_SEL.1.1*	5.1.6.1
	FAU_STG	FAU_STG.1	FAU_STG.1.1 FAU_STG.1.2	5.1.7.1 5.1.7.2
		FAU_STG.3	FAU_STG.3.1*	5.1.8.1
		FAU_STG.4	FAU_STG.4.1*	5.1.9.1
FDP	FDP_ACC	FDP_ACC.1	FDP_ACC.1.1*	5.2.1.1
	FDP_ACF	FDP_ACF.1	FDP_ACF.1.1* FDP_ACF.1.2* FDP_ACF.1.3* FDP_ACF.1.4*	5.2.2.1 5.2.2.2 5.2.2.3 5.2.2.4
	FDP_RIP	FDP_RIP.2	FDP_RIP.2.1	5.2.3.1
	FDP_RIP	FDP_RIP.2 (Note1)	FDP_RIP.2.1	5.2.3.2

Table 2: Security Functional Requirements

CLASS	FAMILY	COMPONENT	ELEMENT	[CAPP] paragraph
FIA	FIA_ATD	FIA_ATD.1	FIA_ATD.1.1*	5.3.1.1
	FIA_SOS	FIA_SOS.1	FIA_SOS.1.1	5.3.2.1
	FIA_UAU	FIA_UAU.1	FIA_UAU.1.1* FIA_UAU.1.2	5.3.3.1 5.3.3.2
		FIA_UAU.7	FIA_UAU.7.1	5.3.4.1
	FIA_UID	FIA_UID.1	FIA_UID.1.1* FIA_UID.1.2	5.3.5.1 5.3.5.2
	FIA_USB	FIA_USB.1	FIA_USB.1.1;1* FIA_USB.1.1;2* FIA_USB.1.1;3*	5.3.6.1 5.3.6.2 5.3.6.3
FMT	FMT_MSA	FMT_MSA.1	FMT_MSA.1.1*	5.4.1.1
		FMT_MSA.3	FMT_MSA.3.1 FMT_MSA.3.2*	5.4.2.1 5.4.2.2
	FMT_MTD	FMT_MTD.1	FMT_MTD.1.1;1 FMT_MTD.1.1;2 FMT_MTD.1.1;3 FMT_MTD.1.1;4 FMT_MTD.1.1;5	5.4.3.1 5.4.4.1 5.4.5.1 5.4.6.1 5.4.6.2
	FMT_REV	FMT_REV.1	FMT_REV.1.1;1 FMT_REV.1.2;1* FMT_REV.1.1;2 FMT_REV.1.2;2*	5.4.7.1 5.4.7.2 5.4.8.1 5.4.8.2
	FMT_SMR	FMT_SMR.1	FMT_SMR.1.1* FMT_SMR.1.2	5.4.9.1 5.4.9.2
FPT	FPT_AMT	FPT_AMT.1	FPT_AMT.1.1*	5.5.1.1
	FPT_RVM	FPT_RVM.1	FPT_RVM.1.1	5.5.2.1
	FPT_SEP	FPT_SEP.1	FPT_SEP.1.1 FPT_SEP.1.2	5.5.3.1 5.5.3.2
	FPT_STM	FPT_STM.1	FPT_STM.1.1	5.5.4.1

Table 2: Security Functional Requirements

5.1.2 Protection Profile SFRs Tailored for This Security Target

The elements in [CAPP] that are tailored for this security target are indicated by a

* after the element's name in the table above. These tailored elements are given below, with the new material underlined. The remaining SFRs in the table above are to be used for this security target exactly as they appear in [CAPP].

5.1.2.1 Security Audit (FAU)

5.1.5.1 The TSF shall provide the ability to perform searches of audit data based on the following attributes: FAU_SAR.3.1

a) User identity;

b) type of audit event and audit class.

5.1.6.1 The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes: FAU_SEL.1.1

a) User identity;

b) audit class.

5.1.8.1 The TSF shall generate an alarm to the authorized administrator if the audit trail exceeds or meets 100% occupancy. FAU_STG.3.1

Note: An alarm is generated once 100% of the allocated audit space is reached. This disk space may be exceeded in certain circumstances e.g. by auditable actions taken by authorised administrators.

5.1.9.1 The TSF shall be able to prevent auditable events, except those taken by the authorized administrator, if the audit trail is full. FAU_STG.4.1

5.1.2.2 User Data Protection (FDP)

5.2.1.1 The TSF shall enforce the Discretionary Access Control Policy on processes acting on the behalf of users, Class A objects and all operations among subjects and objects covered by the DAC policy. FDP_ACC.1.1

5.2.2.1 The TSF shall enforce the Discretionary Access Control Policy to objects based on the following: FDP_ACF.1.1

a) The user identity and group membership(s) associated with a subject; and

b) The access control attributes associated with an object: ACL, permission bits

5.2.2.2 The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: FDP_ACF.1.2

IF the object has an explicit ACL, THEN:

- access granted to the object's owner is based on the user::rwx permissions
- access granted to individuals specified in the ACL is based on the bitwise AND operation of the user:[specified]:rwx and mask:rwx permissions
- access granted to subjects who belong to the object's group is based on the bitwise AND operation of the group::rwx and the mask:rwx entries
- access granted to subjects who belong to groups specified in the ACL is based on the bitwise AND operation of the group:[specified]:rwx and mask:rwx permissions
- access granted to all other subjects is based on the object's *other* permissions

ELSE

- access granted to the object's owner is based on the object *user* rwx permissions
- access granted to subjects who belong to the object's group is based on the object *group* rwx permissions
- access granted to all other subjects is based on the object *other* rwx permissions

5.2.2.3 The TSF shall explicitly authorize access of subjects to objects based on the following additional rule: FDP_ACF.1.3

a) If a subject has an effective UID of 0, the TSF shall authorize access of the subject to any given Class A object, even if such access is disallowed by FDP_ACF.1.2.

5.2.2.4 The TSF shall explicitly deny access of subjects to objects based on no additional rules. FDP_ACF.1.4

5.1.2.3 Identification and Authentication (FIA)

5.3.1.1 The TSF shall maintain the following list of security attributes belonging to individual users: FIA_ATD.1.1

a) User Identifier;

b) Group Memberships;

c) Authentication Data;

d) Security-relevant Roles; and

e) login shell.

5.3.3.1 The TSF shall allow the following TSF-mediated actions on behalf of the user to be performed before the user is authenticated

a) select language;

b) select desktop or console login;

c) select remote host for login;

d) help for login function. FIA_UAU.1.1

5.3.5.1 The TSF shall allow the following TSF-mediated actions on behalf of the user to be performed before the user is identified.

a) select language;

b) select desktop or console login;

c) select remote host for login;

d) help for login function. FIA_UID.1.1

5.3.6.1 The TSF shall associate the *following* user security attributes with subjects acting on the behalf of that user: FIA_USB.1.1;1

a) The audit user identity;

b) The effective user identity;

c) The effective group identities;

d) The real user identity and real group identities.

5.3.6.2 The TSF shall enforce the following rules on the initial association of user security attributes with subjects acting on the behalf of a user: FIA_USB.1.1;2

a) Upon successful identification and authentication, the real and effective and audit user identities shall be those specified via the User Identifier attribute held by the TSF for the user.

b) Upon successful identification and authentication, the real and effective group identities shall be those specified via the Group Memberships attributes held by the TSF for the user.

5.3.6.3 The TSF shall enforce the following rules governing changes to the user security attributes associated with subjects acting on the behalf of a user: FIA_USB.1.1;3

a) The effective user identity associated with a subject can be changed to another user's identity via a command, provided that
the effective user identity was 0, or
successful authentication as the new user identity has been achieved;

b) When executing a file which has the set UID permission bit set, the effective user identity associated with the subject shall be changed to that of the owner of the file;

c) When executing a file which has the set GID permission bit set, the effective group identity associated with the subject shall be changed to that of the group attribute of the file.

Application Note: The DAC policy is enforced based on the effective UID as described above. All auditable events are recorded with the audit ID, which contains the identity of the user at identification time. In this manner, all auditable events can be traced back to the person initially identified to the TOE and are not associated to another person who may at some time identify them self as the alternate identity.

5.1.2.4 Security Management (FMT)

5.4.1.1 The TSF shall enforce the Discretionary Access Control Policy to restrict the ability to modify the access control attributes associated with a named object to the subject that owns the object and a subject with an effective UID of 0.FMT_MSA.1.1;1

5.4.2.2 The TSF shall allow the authorised administrators and users authorised by the Discretionary Access Control Policy to modify object security attributes to specify alternative initial values to override the default values when an object or information is created.FMT_MSA.3.2

5.4.7.2 The TSF shall enforce the rules: FMT_REV.1.2;1

a) The immediate revocation of security-relevant authorizations; and

b) Administrative users shall be able to revoke security-relevant authorisations by completely deleting user security attributes, or by modifying the user identity, user name, primary group, secondary group and login shell, or by setting a new password. Such revocation is to take effect when the user next authenticates to the system.

5.4.8.2 The TSF shall enforce the rules: FMT_REV.1.2;2

a) The access rights associated with an object shall be enforced when an access check is made.

5.4.9.1 The TSF shall maintain the roles: FMT_SMR.1.1

a) authorized administrator;

b) users authorised by the Discretionary Access Control Policy to modify object security attributes;

c) users authorised to modify their own authentication data.

5.1.2.5 Protection of the TOE Security Functions (FPT)

5.5.1.1 The TSF shall run a suite of tests at the request of an authorized administrator to demonstrate the correct operation of the security assumptions provided by the abstract machine that underlies the TSF. FPT_AMT.1.1

5.2 Strength of Function

The claimed minimum strength of function is *SOF-medium*.

5.3 TOE Security Assurance Requirements

The target evaluation assurance level for the product is EAL4 [CC]. No augmented assurance requirements are defined.

5.4 Security Requirements for the IT Environment

The IT environment is required to meet the objectives described in Section 4.2. All but one of these objectives is met by procedural measures, however O.BOOT is met by the OpenBoot PROM for Sparcstations. The functionality provided by this firmware is specified as follows:

The OpenBoot PROM on Sparc workstations shall restrict the ability to modify the behaviour of the boot strapping process to users who know the valid PROM password.[FMT_MOF.1]

Refinement:

a) In fully secure mode, the valid password is required in order to boot the workstation;

b) In command-secure mode, the valid password is required in order to boot a non-default operating system;

c) In fully secure and command-secure modes, the valid password is required in order to configure PROM operating modes, PROM passwords or boot parameters.

6 TOE Summary Specification

6.1 IT Security Functions

The ITSFs to which the claimed Strength of Function (SoF) rating applies are as follows:

- IA.1
- IA.11

6.1.1 Discretionary Access Control (DAC)

Policy

The security-related software shall define and control access between named users and named objects (e.g., files and programs) in the data processing system. All named users and named objects shall be uniquely identifiable over all the workstations in the system.

Within Solaris, DAC is applied in two different ways depending on the type of object. This security target therefore defines two classes of object, Class A and Class B.

- Class A objects are filesystem and System V IPC objects
- Class B objects are process objects and X-window objects.

Discretionary Access Control - Class A Objects

The enforcement mechanisms for Class A objects shall allow users to specify and control sharing of those objects, initially generated by the user, by named users (group control is optional) using the specific designations of read, write, execute/ search.

The discretionary access control mechanism shall, either by explicit user action or by default, provide that objects are protected from unauthorised access.

These access controls shall be capable of including or excluding access down to the level of a single user.

Access permission to a Class A object by users not already possessing access permission shall only be assigned by an authority responsible and authorised to grant access.

Subjects have a number of IDs associated with them:-

- effective user ID, real user ID, saved user ID;
- effective group ID, real group ID, saved group ID, supplemental groups; and
- audit user ID.

The Solaris 8 discretionary access controls use the effective user ID and effective group ID for policing a subject's access rights to a Class B object.

Self/Group/Public/ACL Permissions

The product shall implement a discretionary access control mechanism that controls the access of subjects to named Class A objects. The discretionary access control mechanism shall associate with each Class A object an owner identification, a group identification, a set of access permissions and/or an access control list (ACL).

DAC.1 Subject to DAC.8 the access permissions on a Class A object can be modified only by a subject that owns the object.

DAC.2 No subject may change the owner or group of a Class A object unless it has a uid of 0, or optionally is the owner of the object.

Note that Solaris 8 can be configured to allow the modification of owner and group of a Class A object by the owner, or can be configured to be POSIX compliant whereby only the root user (uid 0) can modify ownership, and the owner can change the group only to one which they are a member of. The functioning of both of these modes should be assessed during the evaluation of the TOE.

DAC.3 Subject to DAC.1, a subject may assign any combination of the following access modes to a Class A object:-

- read, write, execute/search

to:-

- the owner of the object (self);
- any member of the owning group (group); and
- any user other than the owner or a member of the owning group (other).

DAC.4 Subject to DAC.1, an Access Control List (ACL) can be created for a *ufs* or *nfs* filesystem object to specify a set of allowable access modes (as per DAC.3) for individually named users or groups. If an ACL entry for a user or group contains no access modes, the specified user or group is specifically excluded from accessing the object. Users not listed anywhere in an ACL (either through explicit user ACL entries or through any applicable group ACL entries) shall have their access to the object determined by the "Other" ACL entry.

Note that the scope of the above Security Function is limited to regular files held on ufs and nfs filesystems. This includes hard links but excludes device special files, pipes and symbolic links. However, the regular files referenced by symbolic links can still be controlled by ACLs.

DAC.6 Whenever a subject requests access to a Class A object, the access permissions for that object shall be checked to determine whether the user who owns the subject can access the object in the requested mode. Where an ACL is defined for an object, it shall be used instead of the object's permission bits.

DAC.7 When a subject creates a filesystem object, the user ID of the subject is assigned to the object, and the user's umask restricts the initial access permissions of the object. The TOE default is that a user's umask is set to prevent any user other than the owner having write access to the object.

DAC.8 Subjects may only override discretionary access control if they have a uid of 0.

6.1.2 Object Reuse

OR.1 When an object is initially assigned, allocated or reallocated to a subject from the system's pool of unused objects, the security-related software shall assure that the object contains no data for which the subject is not authorised.

OR.2 When memory objects are allocated for use by a subject at run-time, the memory shall contain no data from a previous subject.

Any portion of a file object that has not been previously written to shall either:

- not be readable by any subject; or
- shall be cleared before it can be read.

OR.3 The TOE shall revoke all access rights held by a subject to the information contained within a storage object, before reuse by other subjects.

6.1.3 Identification and Authentication

Password Authentication

IA.1 The product shall require users to identify and successfully authenticate themselves, using a user name and a password, before performing any other actions.

IA.2 Upon successful identification and authentication, the real and audit user IDs and the real group IDs of the user's subjects shall be those specified by the authentication data.

Password Protection

The authentication data shall not contain a clear text version of each user's password, but rather a one-way encrypted value based on the user's password. When a user enters his password, it is used to construct an encrypted value and is compared against the encrypted value in the authentication data.

IA.9 On entry, passwords shall not be displayed in cleartext.

IA.10 User passwords are always stored in encrypted form.

Note: this ITSF does not apply to BOOTPROM passwords (which are not user passwords, and are beyond the scope of this security target).

IA.11 The authentication data shall be protected so that it cannot be written other than as follows:

- by administrative users who may
    - create, delete user identities,
    - modify the name, primary group, secondary group, login shell;
    - set passwords if required; and
- by a user supplying a new password.

6.1.4 Audit

Audit Events

Audit.1 The use of the identification and authentication mechanisms is auditable. The following information is recorded for each event audited:-

- date;
- time;
- user identity - audit ID and effective user ID (if successful);
- security attributes of the user (if successful)
- identification of the workstation or terminal used; and
- success or failure of the event.

Audit.2 Attempts to access to objects are auditable. The following information is recorded for each event audited:-

- date;
- time;
- user identity - audit ID and effective user ID;
- name of the object;
- type of access attempted; and
- success or failure of the attempt.

Audit.3 The creation of an object is auditable. The following information is recorded for each event audited:

- date;
- time;
- user identity - audit ID and effective user ID;
- name of the object.

Audit.4 The creation of a subject to run on behalf of a user is auditable. The following information is recorded for each event audited:

- date;
- time;
- user identity - audit ID and effective user ID;
- success or failure of the attempt;

Audit.5 The creation, deletion, disabling or enabling of user accounts is auditable. The following information is recorded for each event audited:

- date;
- time;
- identity of the user implementing the change - audit ID and effective user ID;
- name of the user account being modified; and
- type of action.

Audit.6 Attempts to assign or modify security attributes are auditable. The following information is recorded for each event audited:

- date;
- time;
- identity of the user implementing the change - audit ID and effective user ID;
- name of the user account or object being modified;
- type of attribute; and
- success or failure of the attempt.

Audit.7 The assuming of uid 0 is auditable. The following information is recorded for each event audited:

- date;
- time;
- user identity - audit ID and effective user ID;
- name of the object involved (if any).

Audit.8 Security relevant events affecting the operation of the auditing functions are auditable. The following information is recorded for each event audited:

- date;
- time;
- user identity (if relevant) - audit ID and effective user ID; and
- type of event.

Audit.10 The creation or deletion of a logical device for storage media is auditable. The following information is recorded for each event audited:

- date;
- time;
- user identity (if relevant) - audit ID and effective user ID;
- name of the object and device; and
- type of action.

Audit.11 Start-up and shutdown of the system is auditable. The following information is recorded for each event audited:

- date;
- time;
- user identity (mandatory for shutdown only) - audit ID and effective user ID; and
- type of event.

Audit.12 The date and time information recorded in audit records shall be reliable.

Protection of Audit Information

Audit.14 Audit data shall be protected so that access to it is limited to administrative users.

Audit.15 Password data (in clear or encrypted form) is never recorded in the audit log.

Selective Audit Data Collection/Reduction

Audit.16 Only administrative users may define classes of audit event.

Audit.17 Only administrative users shall be able to define the default system audit-mask that defines which audit classes are recorded by default.

Audit.18 Only administrative users shall be able to define a per-user audit-mask that defines which audit classes are recorded for that user. For a given user, the system shall audit those classes that are in the default system audit mask or the per-user audit mask.

Audit.19 Audit reduction software shall be available to allow administrative users to selectively retrieve audit data based on, at a minimum, the identity of users, the type of audit event, and the audit class.

Audit Data Storage

Audit.20 Each workstation of the (distributed) product may store audit data locally or on another workstation of the product that can act as an audit server.

Audit.21 If another workstation of the product is being used as an audit server, and this audit server becomes unavailable, the (local) workstation shall either:

- automatically switch over to storing audit data locally,

or

- suspend operation until the audit server is again available,

or

- suspend operation until an alternative workstation of the product takes over as an audit server;

or

- if no workstation is able to store audit data then no further auditable events shall occur (ie., all auditable actions will be suspended).

Audit.22 Facilities are available to allow administrative users to archive and maintain the audit logs. Only such users may use these facilities to archive and maintain the audit logs.

Audit.23 The system shall notify an administrator of audit trail saturation.

6.1.5 Enforcement Functions

ENF.1 The TOE shall validate all actions between subjects and objects that require policy enforcement, before allowing the action to succeed.

ENF.2 The TOE shall maintain a domain ‘kernel space’ for its own trusted execution. This shall be kept separate from untrusted subjects which operate in a separate domain ‘user space’.

ENF.3 The TOE shall allow an administrator to perform a self test to ensure that the underlying TSF is enforcing process separation.

6.2 Required Security Mechanisms

6.2.1 Identification and Authentication

The TOE uses a username and password mechanism to provide authentication of users. The construction of passwords is sufficient to meet the requirements of a strength of function of Medium. This mechanism supports the IT SFs IA.1 and IA.11.

Passwords are encrypted using a proprietary one way hashing algorithm, however the assessment of algorithmic strength does not form part of the evaluation.

6.3 Assurance Measures

Assurance measures will be adopted to address each of the EAL4 assurance requirements, as summarised in Table B.1 in [CC, Part 3] and as summarised below.

Assurance components	Description of how requirement will be met
ACM_AUT.1 Partial CM automation	The requirement for partial CM automation will have been implicitly examined under the previous ITSEC E3 evaluation of Solaris 2.4SE and 2.6SE. The method of achieving this will be highlighted to the evaluation team in the [unchanged] Software Development Framework [SDF] document
ACM_CAP.4 Generation support and acceptance procedures	This requirement will have been examined under the previous ITSEC E3 evaluations of Solaris 2.4SE and 2.6SE. Full reuse of results will therefore be claimed.
ACM_SCP.2 Problem tracking CM coverage	Problem tracking is largely covered under the previously evaluated configuration management procedures. The tracking of security flaws will be documented separately and submitted to the evaluation.
ADO_DEL.2 Detection of modification	Full reuse of results of the ITSEC E3 Solaris 2.6SE evaluation will be claimed for this assurance requirement.
ADO_IGS.1 Installation, generation, and start-up procedures	Full reuse of results of the ITSEC E3 Solaris 2.6SE evaluation will be claimed for this assurance requirement.

Table 3: How Assurance Requirements Will Be Met

Assurance components	Description of how requirement will be met
ADV_FSP.2 Fully defined external interfaces	The Solaris 8 MAN pages, which are relevant to the implementation of the security functions, will be provided to the evaluation and assessed against this assurance requirement.
ADV_HLD.2 Security enforcing high-level design	The Architectural Design document, previously evaluated against ITSEC E3 for the Solaris 2.6SE product, will be amended for Solaris 8 and submitted to the evaluation for assessment against this requirement.
ADV_IMP.1 Subset of the implementation of the TSF	The source code for Solaris 8 will be provided to the evaluation.
ADV_LLD.1 Descriptive low-level design	The Detailed Design document, previously evaluated against ITSEC E3 for the Solaris 2.6SE product, will be amended for Solaris 8 and submitted to the evaluation for assessment against this requirement.
ADV_RCR.1 Informal correspondence demonstration	This correspondence information will be contained in the functional specification and design documents. The functional specification will map ITSFs to MAN pages. The HLD will map ITSFs to the HLD, and the LLD will map ITSFs and source code modules to the LLD basic components
ADV_SPM.1 Informal TOE security policy model	The requirements for an Informal Security Policy Model (ISPM) are met by the information presented in sections 1-7 of this document.
AGD_ADM.1 Administrator guidance	The Solaris 8 operational documentation relevant to an administrator will be submitted to the evaluation and assessed against this requirement.
AGD_USR.1 User guidance	The Solaris 8 operational documentation relevant to an end user will be submitted to the evaluation and assessed against this requirement.
ALC_DVS.1 Identification of security measures	Full reuse of results of the ITSEC E3 Solaris 2.6SE evaluation will be claimed for this assurance requirement.

Table 3: How Assurance Requirements Will Be Met

Assurance components	Description of how requirement will be met
ALC_LCD.1 Developer defined life-cycle model	The Life Cycle definition is documented in the [SDF]. This will be submitted to the evaluation against this requirement.
ALC_TAT.1 Well-defined development tools	The tools used in the development of Solaris 8 are the same as for Solaris 2.4SE and 2.6SE. Full reuse of results will therefore be claimed.
ATE_COV.2 Analysis of coverage	The analysis of test coverage will be presented to the evaluation in a form similar to that provided to the Solaris 2.6SE evaluation. The existing coverage is against both High and Low level designs and should therefore be to a sufficient depth.
ATE_DPT.1 Testing: high-level design	As for ATE_COV.2
ATE_FUN.1 Functional testing	The test documentation provided to the evaluation will be in a format similar to that provided to the Solaris 2.6SE evaluation. The tests will be run on a range of platforms including: - small, medium and large Ultra II workstations as representative examples of the UltraSparc II processor range of workstations. - an Intel platform.
ATE_IND.2 Independent testing - sample	The resources provided to the CLEF for functional testing will be made available for them to perform additional, independent testing.
AVA_MSU.2 Validation of analysis	The Ease of Use analysis, previously submitted for the ITSEC E3 evaluation of Solaris 2.6SE, will be updated for Solaris 8 and submitted to the evaluation against this requirement.
AVA_SOF.1 Strength of TOE security function evaluation	The Strength of Mechanism analysis, previously submitted for the ITSEC E3 evaluation of Solaris 2.6SE, will be updated for Solaris 8 and submitted to the evaluation against this requirement.
AVA_VLA.2 Independent vulnerability analysis	The construction and operational vulnerability analyses, previously submitted for the ITSEC E3 evaluation of Solaris 2.6SE, will be updated for Solaris 8 and submitted to the evaluation against this requirement. In addition, evidence of Sun's continuing search for vulnerabilities and the resolution of them in the Solaris product, will be provided.

Table 3: How Assurance Requirements Will Be Met

7 Rationale

This chapter presents the evidence used in the Security Target evaluation. This evidence supports the claims that the ST is a complete and cohesive set of requirements, that a conformant TOE would provide an effective set if IT security countermeasures within the security environment, and that the TOE summary specification addresses the requirements. The rationale also demonstrates that any PP conformance claims are valid.

7.1 Correlation of Threats, Policies, Assumptions and Objectives.

The correlation between threats, organisational policies, assumptions and objectives is detailed in the following sections, and is summarised below.

Objectives:	O.Authorisation	O.DAC	O.Audit	O.Residual_Info	O.Manage	O.Enforcement	O.Admin	O.Accountable	O.AuditData	O.AuthData	O.Boot	O.Consistency	O.Install	O.Info_Protect	O.Maintenance	O.Recover	O.Software_in	O.Serial_Login	O.Protect
Threats																			
T.Access_Info	✓	✓	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓		✓		✓	✓
T.Access_TOE	✓		✓		✓	✓	✓	✓	✓	✓	✓	✓	✓						
T.Modify	✓	✓	✓			✓	✓				✓	✓	✓	✓					✓
T.Admin_Rights	✓		✓		✓	✓	✓	✓		✓	✓	✓	✓				✓		
T.Transit													✓	✓					✓
P.Auth	✓				✓	✓				✓									
P.DAC		✓		✓	✓	✓													
P.Accountable	✓		✓		✓	✓	✓	✓	✓										
A.Protect														✓					✓
A.Admin							✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
A.NIS_DOMAINS													✓						
A.Bridges&Routers														✓					✓

The threats, objectives and correlation between them have been derived from the previous Solaris ITSEC security target, and provides the primary focus for the reader in portraying the intended purpose and use of the Solaris product.

The OSPs are derived from the [CAPP] and are included to indicate how the OSPs

relate to the TOE security objectives and the primary non-IT security objectives. The OSPs are generally more abstract than the threats and so the correlation between similar threats and OSPs to objectives is not necessarily the same.

The environmental objectives O.ADMIN, O.BOOT, O.INSTALL and O.CONSISTENCY are general objectives which are help counter all the threats (with the exception of T.TRANSIT in some cases) as follows:

- O.ADMIN: Those responsible for administering the TOE must be competent and trustworthy in order to manage the security functions effectively. Effective management is necessary in order that the threats are not inadvertently or deliberately realised;
- O.BOOT and O,.INSTALL ensure that the correct copy of the operating system is installed and subsequently booted in a secure manner, and is hence relevant to help counter all the threats;
- O.CONSISTENCY is required to ensure that data is set up and maintained in a consistent manner across all workstations in the distributed system. Erroneous or duplicate entries in the authentication information may allow any of the threats to be realised.

7.2 Security Objectives Rationale

This section demonstrates that the security objectives stated in Section 4 above are traceable to all of the aspects identified in the TOE security environment and are suitable to cover them.

7.2.1 Complete Coverage - Threats

This section provides evidence demonstrating coverage of the threats by both the IT and Non-IT security objectives. The table is followed by a discussion of the coverage for each threat.

[T.ACCESS_INFO] *An authorised user of the TOE accesses information without having permission from the person who owns, or is responsible for, the information.*

Security objective O.DAC counters this threat directly by ensuring the means are provided by which users can securely implement compartmentalisation of information in order to counter this threat. O.RESIDUAL_INFO helps counter the threat

by ensuring that once an object has passed outside the control of DAC, that residual information contained in it is not passed to other users.

Security objective O.AUTHORISATION supports O.DAC in countering this threat by ensuring that an authorised user cannot impersonate another authorised user, thereby undermining the intent of O.DAC.

O.AUDIT helps counter this threat by ensuring that repeated [unsuccessful] attempts to access information to which the user is not granted permission, can be detected, thereby allowing the administrator to take action before the attack is successful.

O.MANAGE and O.ENFORCEMENT counter this threat by ensuring:
- privileged actions are controlled; and
- the access controls cannot be bypassed.

Support is also provided by the following security objectives for the environment:

a) O.ADMIN - to administer the controls over access to information;
b) O.BOOT - to ensure that information cannot be accessed by booting an alternative operating system;
c) O.AUTHDATA is require to protect the information which would otherwise enable attackers to gain access to the TOE;
d) O.PROTECT - to ensure that data transmitted over network cabling is appropriately protected;
e) O.RECOVER - to ensure that information cannot be accessed by terminating the operation of a workstation (whether intentional or not);
f) O.SERIAL_LOGIN - to ensure that information is not seen by users who do not have a need to know when serial devices are being used;

[T.ACCESS_TOE] *An unauthorised user of the TOE gains access to the system, thereby gaining unauthorised access to information.*

O.AUTHORISATION ensures that all users identify themselves to the system, and that their claimed identity is authenticated before being granted access to the system. This therefore prevents unauthorised users gaining access to the system.

O.AUDIT provides support in the form of auditing attempts to access the TOE. The auditing of unsuccessful attempts to login help to detect and hence counter the threat of repeated attacks on the access functions.

O.MANAGE and O.ENFORCEMENT support this threat by ensuring:
- the database of authorised users is properly managed and maintained;
- the authorisation functions are always invoked and cannot be bypassed;
- the auditing functions are set up appropriately to detect repeated attempts to login.

Support is also provided by the following security objectives for the environment:

a) O.ADMIN - to ensure that the introduction of new user identities is a restricted operation and performed only by the users responsible.
b) O.ACCOUNTABLE - to ensure that unauthorised users are not provided with accounts enabling them to access the TOE;
c) O.AUDITDATA - which ensures that bad passwords, which might be used to determine valid passwords, are not stored in the audit trail, and hence not known to any users.
d) O.AUTHDATA - which ensures that valid authentication data is not disclosed to unauthorised individuals;
e) O.CONSISTENCY - which ensures that access is granted to individuals on a basis consistent across all workstation. This avoids possible duplication of authentication data.

[T.MODIFY] *Unauthorised modification or destruction of information by an authorised user of the TOE.*

The security objective O.DAC provides the means to ensure that users can protect the integrity of the information they own or are responsible for.

Security objective O.AUTHORISATION supports O.DAC in countering this threat by ensuring that an authorised user cannot impersonate another authorised user, thereby undermining the intent of O.DAC.

O.AUDIT helps counter this threat by ensuring that repeated [unsuccessful] attempts to modify information to which the user is not granted permission, can be detected, thereby allowing the administrator to take action before the attack is successful.

O.ENFORCEMENT supports this threat by ensuring the access control functions are always invoked and cannot be bypassed.

Support is also provided by the following security objectives for the environment:

a) O.INFO_PROTECT and O.PROTECT - ensures that information transmitted over the network is not accessible to other authorised users of the TOE and hence the data cannot be modified or destroyed;
b) O.ADMIN ensures that the default access permissions are set appropriately so that access is granted, by default, to a restricted set of users.

[T.ADMIN_RIGHTS] *Unauthorised use of facilities which require administration rights by an authorised user of the TOE.*

O.AUTHORISATION ensures that only authorised users can access the TOE, and provides for identification of users to determine the administration right assigned to the user.

O.AUDIT discourages the unauthorised use of administrator facilities by ensuring that any such breach of security policy can be detected.

O.MANAGE and O.ENFORCEMENT support this threat by ensuring:
- the database of authorised administrators is properly managed and maintained;
- the administration functions are always checked when invoked and cannot be bypassed;
- the auditing functions are set up appropriately to detect repeated attempts to use the administration functions by non-administrative users.

O.AUTHDATA ensures user's authentication data is kept secure. This prevents an authorised user impersonating an administrator to gain unauthorised access to administrator facilities. O.CONSISTENCY ensures that a single set of administration rights exist across the TOE, thereby avoiding errors caused by duplication or erroneous entries in the authorisation data. O.ACCOUNTABLE ensure that users are uniquely identified and the use of privileged facilities can be controlled amongst the user community.

O.SOFTWARE_IN ensures that only administrators can introduce software into the TOE and hence counters the threat of malicious software being introduced. The introduction of some software e.g. compilers, may provide enhanced facilities to an attacker which could be used to mount a successful attack on the TOE and hence make unauthorised use of administration facilities.

The following threats apply in environments where specific threats to distributed systems need to be countered. Typically such threats are countered by cryptographic or physical protection of network connections.

[T.TRANSIT] *Data transferred between workstations is disclosed or modified to unauthorised users or processes either directly or indirectly (e.g. through spoofing of workstation identity).*

Administrators must ensure that data transferred between workstations i.e. along network cabling, is suitably protected against physical or other (e.g. tempest) attacks which may result in the disclose, modification or delay of information transmitted between workstations. Objective O.PROTECT ensures this is achieved. Because such issues need to be considered at installation time, objectives O.INSTALL and O.INFO_PROTECT are also applicable.

7.2.2 Complete Coverage - Policy

This section provides evidence demonstrating coverage of the Organisational Security Policy by both the IT security objectives. The table is followed by a discussion of the coverage for each Security Policy.

[P.AUTH] Only those users who have been authorised to access the information within the system may access the system.

This policy is implemented through the objective O.AUTHROISATION which ensures that only authorised users are allowed access to the system. O.MANAGE and O.ENFORCEMENT support this policy by ensuring that the set of authorised users is effectively managed and that the authorisation functions are always invoked and cannot be bypassed.

O.AUTHDATA supports this policy by ensuring that authorisation data is constructed in a manner commensurate with the protection required for the information on the TOE and that passwords are not disclosed since doing so would compromise the policy.

[P.DAC] The right to access specific data objects is determined on the basis of:

a) the owner of the object; and
b) the identity of the subject attempting the access; and
c) the implicit and explicit access rights to the object granted to the subject by the object owner.

P.DAC is implemented through the objective O.DAC which provides the means of controlling access between objects and subjects on the attributes defined by the policy, and is supported by O.RESIDUAL_INFO objective which ensures that

information will not given to users which do not have a need to know, when resources are reused. O.ENFORCEMENT supports this policy by ensuring that the access control functions are always invoked and cannot be bypassed. O.MANAGE supports this policy by requiring authorized administrator be able to manage the functions.

[P.ACCOUNTABLE] The users of the system shall be held accountable for their actions within the system.

Accountability is implemented primarily through the objective O.AUDIT which ensures uses' security relevant events can be recorded so as to be able to hold users accountable for their actions. An unauthorised user can not be held accountable for their actions and O.AUTHORISATION therefore supports this policy by ensuring that only authorised users are allowed access. O.MANAGE and O.ENFORCEMENT support this policy by ensuring that an effective set of actions are audited in order to detect attempted breaches of the security policy and that the auditing functions are always invoked and cannot be bypassed.

O.ADMIN, O.ACCOUNTABLE and O.AUDITDATA ensure that the administrator manages the auditing security functions effectively.

7.2.3 Complete Coverage - Environmental Assumptions

This section provides evidence demonstrating coverage of the environmental assumptions by security objectives. The table is followed by a discussion of the coverage for each environmental assumption.

[A.PROTECT] *It is assumed that all network and peripheral cabling is approved for the transmittal of the most sensitive data held by the system. Such physical links are assumed to be adequately protected against threats to the confidentiality and integrity of the data transmitted.*

The environmental objective O.PROTECT ensures that network cabling is suitably protected against threats of modification, tampering or interruption of the data transmitted via this medium. O.INFO_PROTECT ensures that, where the cabling is carrying classified information, that the infrastructure has been approved.

[A.ADMIN] *It is assumed that there are one or more competent individuals who are assigned to manage the TOE and the security of the information it contains. Such personnel are assumed not to be careless, wilfully negligent or hostile.*

This assumption is met primarily by O.ADMIN, and supported by all the other environmental objectives which ensure that the administrative functions are performed in an manner effective in maintaining the security functions of the TOE.

[A.NIS_DOMAINS] *It is assumed that, if the product comprises more than one workstation, all workstations are administered from a central point within each NIS+ domain.*

Note: NIS+ allows the creation of multiple administrative domains, thus allowing administrators to control local resources and user accounts, yet making it possible for users and resources to operate seamlessly over the entire organisation.

NIS+ is installed and configured at installation time, and therefore objective O.INSTALL ensures this assumption is upheld.

[A.BRIDGES&ROUTERS] *All bridges and routers are assumed to correctly pass data without modification.*

As for A.Protect, this assumptions is met by O.PROTECT and O.INFO_PROTECT; bridges and routers are part of the cabling infrastructure.

7.3 Security Requirements Rationale

This section demonstrates that the set of security requirements is suitable to meet and is traceable to the set of security objectives.

7.3.1 Complete Coverage - Objectives

This section demonstrates that the functional components selected for the TOE provide complete coverage of the defined security objectives. The mapping of components to security objectives is depicted in the following table.

O.AUTHORISATION

The TSF must ensure that only authorised users gain access to the TOE and its resources.

Users authorised to access the TOE are defined using an identification and authentication process [CAPP, 5.3.5 and 5.3.3]. To ensure authorised access to the TOE, authentication data is protected [CAPP, 5.3.1, 5.3.4, 5.4.6]. The strength of the authentication mechanism must be sufficient to ensure unauthorised users cannot

pose as authorised users with reasonable time, effort and other constraints [CAPP, 5.3.2].

O.DAC

The TSF must provide its users with the means of controlling and limiting access to the objects and resources they own or are responsible for, on the basis of individual users or identified groups of users, and in accordance with the set of rules defined by the P.DAC security policy.

Discretionary access control must have a defined scope of control [CAPP, 5.2.1]. The rules of the DAC policy must be defined [CAPP, 5.2.2]. The security attributes of objects used to enforce the DAC policy must be defined [CAPP, 5.2.2]. The security attributes of subjects used to enforce the DAC policy must be defined [CAPP, 5.3.1 and 5.3.6]. Authorised users must be able to control who has access to objects [CAPP, 5.4.1] and be able to revoke that access [CAPP, 5.4.8]. Protection of named objects must be continuous, starting from object creation [CAPP, 5.4.2].

Security Objective	Functional Component
O.AUTHORISATION	5.3.1 User Attribute Definition (FIA_ATD.1) 5.3.2 Strength of Authentication Data (FIA_SOS.1) 5.3.3 Authentication (FIA_UAU.1) 5.3.4 Protected Authentication Feedback (FIA_UAU.7) 5.3.5 Identification (FIA_UID.1) 5.4.6 Management of Authentication Data (FMT_MTD.1)
O.DAC	5.2.1 Discretionary Access Control Policy (FDP_ACC.1) 5.2.2 Discretionary Access Control Functions (FDP_ACF.1) 5.3.1 User Attribute Definition (FIA_ATD.1) 5.3.6 User-Subject Binding (FIA_USB.1) 5.4.1 Management of Object Security Attributes (FMT_MSA.1) 5.4.2 Static Attribute Initialization (FMT_MSA.3) 5.4.8 Revocation of Object Attributes (FMT_REV.1)
O.AUDIT	5.1.1 Audit Data Generation (FAU_GEN.1) 5.1.2 User Identity Generation (FAU_GEN.2) 5.1.3 Audit Review (FAU_SAR.1) 5.1.4 Restricted Audit Review (FAU_SAR.2) 5.1.5 Selectable Audit Review (FAU_SAR.3) 5.1.6 Selective Audit (FAU_SEL.1) 5.1.7 Guarantees of Audit Data Availability (FAU_STG.1) 5.1.8 Action in Case of Possible Audit Data Loss (FAU_STG.3) 5.1.9 Prevention of Audit Data Loss (FAU_STG.4) 5.3.6 User-Subject Binding (FIA_USB.1) 5.4.3 Management of the Audit Trail (FMT_MTD.1) 5.4.4 Management of Audited Events (FMT_MTD.1) 5.5.4 Reliable Time Stamps (FPT_STM.1)
O.RESIDUAL_INFO	5.2.3 Object Residual Information Protection (Note.1) 5.2.4 Subject Residual Information Protection (FDP_RIP.2)
O.MANAGE	5.1.3 Audit Review (FAU_SAR.1) 5.1.5 Selectable Audit Review (FAU_SAR.3) 5.1.6 Selective Audit (FAU_SEL.1) 5.1.8 Action in Case of Possible Audit Data Loss (FAU_STG.3) 5.1.9 Prevention of Audit Data Loss (FAU_STG.4) 5.4.3 Management of the Audit Trail (FMT_MTD.1) 5.4.4 Management of Audited Events (FMT_MTD.1) 5.4.5 Management of User Attributes (FMT_MSA.1) 5.4.6 Management of Authentication Data (FMT_MTD.1) 5.4.7 Revocation of User Attributes (FMT_REV.1) 5.4.9 Security Management Roles (FMT_SMR.1)
O.ENFORCEMENT	5.5.1 Abstract Machine Testing (FPT_AMT.1) 5.5.2 Reference Mediation (FPT_RVM.1) 5.5.3 Domain Separation (FPT_SEP.1)

O.AUDIT

The TOE must provide the means of recording any security relevant events, so as to (a) assist an administrator in the detection of potential attacks or misconfiguration of the TOE security features that would leave the TOE susceptible to attack; and (b) hold users accountable for any actions they perform that are relevant to security.

Security-relevant actions must be defined, auditable [CAPP, 5.1.1], and capable of being associated with individual users [CAPP, 5.1.2 and 5.3.6]. The audit trail must be protected so that only authorized users may access it [CAPP, 5.1.4]. The TSF must provide the capability to audit the actions of an individual user [CAPP, 5.1.5, 5.1.6 and 5.3.6]. The audit trail must be complete [CAPP, 5.1.7 and 5.1.9]. The time stamp associated must be reliable [CAPP, 5.5.4]. An authorised administrator must be able to review [CAPP, 5.1.3] and manage [CAPP, 5.1.8, 5.4.3 and 5.4.4] the audit trail.

O.RESIDUAL_INFO

The TSF must ensure that any information contained in a protected resource is not released when the resource is recycled.

Residual information associated with defined objects in the TOE must be purged prior to the reuse of the object containing the residual information [CAPP, 5.2.3, 5.3.4].

O.MANAGE

The TSF must allow administrators to effectively manage the TOE and its security functions, and must ensure that only authorised administrators are able to access such functionality.

The TSF must provide for an authorised administrator to manage the TOE [CAPP, 5.4.9]. The administrator must be able to administer user accounts [CAPP, 5.4.5, 5.4.6, 5.4.7]. The administrator must be able to review manage the audit trail [CAPP, 5.1.3, 5.1.5, 5.1.6, 5.1.8, 5.1.9, 5.4.3 and 5.4.4].

O.ENFORCEMENT

The TOE security policy is enforced in a manner which ensures that the organisational policies are enforced in the target environment i.e. the integrity of the TSF is protected.

The TSF must make and enforce the decisions of the TSP [CAPP, 5.5.2]. It must be protected from interference that would prevent it from performing its functions [CAPP, 5.5.3]. Additionally, the TOE must provide the capability to demonstrate correct operation of the TSF's underlying abstract machine [CAPP, 5.5.1]. The correctness of this objective is further met through the assurance requirements defined in this PP.

This objective provides global support to other security objectives for the TOE by protecting the parts of the TOE which implement policies and ensures that policies are enforced.

7.3.2 Requirements are Mutually Supportive and Internally Consistent

	FAU_GEN.1	FAU_SAR.1	FAU_STG.1	FDP_ACC.1	FDP_ACF.1	FDP_IFC.1	FDP_IFF.1	FIA_ATD.1	FIA_UAU.1	FIA_UID.1	FMT_MSA.1	FMT_MSA.3	FMT_MTD.1	FMT_SMR.1	FPT_STM.1
FAU_GEN.1															x
FAU_GEN.2	x									x					i
FAU_SAR.1	x														i
FAU_SAR.2	i	x													i
FAU_SAR.3	i	x													i
FAU_SEL.1	x									i			x	i	i
FAU_STG.1	x														i
FAU_STG.3	i		x												i
FAU_STG.4			x												i
FDP_ACC.1				i	x					i	i	i		i	
FDP_ACF.1				x	i					i	i	x		i	
FDP_RIP.2															
Note 1															
FIA_ATD.1															
FIA_SOS.1															
FIA_UAU.1										x					
FIA_UAU.7									x	i					
FIA_UID.1															
FIA_USB.1								x							
FMT_MSA.1				o	?	o	?			i				x	
FMT_MSA.3				?	?	?	?			i	x			x	
FMT_MTD.1										i				x	
FMT_REV.1										i				x	
FMT_SMR.1										x					
FPT_AMT.1															
FPT_RVM.1															
FPT_SEP.1															
FPT_STM.1															

All dependencies are satisfied. The key to the symbols used are:

x required dependency
i inferred dependency
o optional dependency (there may be cases where one of two or more optional dependencies are required)
? inferred, optional, dependency

The above correlation is taken directly from [CAPP] with corrections where

[CAPP] is incorrect. The set of IT security requirements for the TOE are the same as those for [CAPP] and hence the above table taken from [CAPP] is appropriate, nd the justification in [CAPP, 7.2.1] applies to show that the TOE has IT security requirements that together form a mutually supportive and internally consistent whole.

7.3.3 Justification for Choice of Assurance Requirements

This security target has been based largely on [CAPP]. It specifies security requirements for a product which is to be used in an environment with a moderate level of risk to the assets. In such environments, an assurance level of at least EAL3 is recommended as stated in [CAPP]. This security target claims an assurance level of EAL4, which also meets these requirements.

7.3.4 Strength of Function Claim is Consistent with Security Objectives

The claimed strength of function rating is SOF-medium. This is consistent with [CAPP] which states that a 'one off' probability of guessing the password shall be 1,000,000. This is specified in SFR FIA_SOS.1 which is in turn consistent with the security objectives described in section 7.3.

7.4 TOE Summary Specification Rationale

This section demonstrates that the TOE security functions and assurance measures are suitable to meet the TOE security requirements.

7.4.1 IT Security Functions Satisfy Functional Requirements

This section demonstrates that the combination of the specified TOE IT security functions work together so as to satisfy the TOE security functional requirements. The table below shows the TOE security functions which together satisfy each security functional requirement. They are grouped under the relevant TOE security

objective.

Security Functional Requirement	TOE Security Function(s)
Audit Data Generation (FAU_GEN.1.1)	Audit.1 to Audit.11, Audit.15[a]
Audit Data Generation (FAU_GEN.1.2)	Audit.1,2,3,4,5,6,8,11,21
User Identity Generation (FAU_GEN.2.1)	Audit.1 to Audit.11
Audit Review (FAU_SAR.1.1)	Audit.19
Audit Review (FAU_SAR.1.2)	Audit.19
Restricted Audit Review (FAU_SAR.2.1)	Audit.14
Selectable Audit Review (FAU_SAR.3.1)	Audit.19
Selective Audit (FAU_SEL.1.1)	Audit.17, Audit.18
Protected Audit Trail Storage (FAU_STG.1.1)	Audit.14
Protected Audit Trail Storage (FAU_STG.1.2)	Audit.14
Action in Case of Possible Audit Data Loss (FAU_STG.3.1)	Audit.23
Prevention of Audit Data Loss (FAU_STG.4.1)	Audit.20, Audit.21
Discretionary Access Control Policy (FDP_ACC.1.1)	DAC.6
Discretionary Access Control Functions (FDP_ACF.1.1)	DAC.3, DAC.4
Discretionary Access Control Functions (FDP_ACF.1.2)	DAC.6
Discretionary Access Control Functions (FDP_ACF.1.3)	DAC.8
Discretionary Access Control Functions (FDP_ACF.1.4)	DAC.3, DAC.4, DAC.6.
Object Residual Information Protection (FDP_RIP.2.1)	OR.1, OR.2, OR.3
Subject Residual Information Protection (Note 1)	OR.1, OR.2, OR.3
User Attribute Definition (FIA_ATD.1.1)	IA.11
Strength of Authentication Data (FIA_SOS.1.1)	IA.1, IA.11[b]
Authentication (FIA_UAU.1.1)	IA.1
Authentication (FIA_UAU.1.2)	IA.1
Protected Authentication Feedback (FIA_UAU.7.1)	IA.9

Table 4: SFR - IT SF Mapping

Security Functional Requirement	TOE Security Function(s)
Identification (FIA_UID.1.1)	IA.1
Identification (FIA_UID.1.2)	IA.1
User-Subject Binding (FIA_USB.1.1;1)	IA.2
User-Subject Binding (FIA_USB.1.1;2)	IA.2
User-Subject Binding (FIA_USB.1.1;3)	IA.2
Management of Object Security Attributes (FMT_MSA.1.1;1)	DAC.1, DAC.2
Static Attribute Initialization (FMT_MSA.3.1)	DAC.7
Static Attribute Initialization (FMT_MSA.3.2)	DAC.7
Management of the Audit Trail (FMT_MTD.1.1;1)	Audit.14
Management of Audited Events (FMT_MTD.1.1;2)	Audit.16, 17, 18
Management of User Attributes (FMT_MTD.1.1;3)	IA.11
Management of Authentication Data (FMT_MTD.1.1;4)	IA.11
Management of Authentication Data (FMT_MTD.1.1;5)	IA.10, IA.11
Revocation of User Attributes (FMT_REV.1.1;1)	IA.11
Revocation of User Attributes (FMT_REV.1.1;2)	DAC.1, DAC.2
Revocation of Object Attributes (FMT_REV.1.2;1)	IA.11
Revocation of Object Attributes (FMT_REV.1.2;2)	DAC.6
Security Management Roles (FMT_SMR.1.1)	DAC.1, DAC.2, IA.11
Security Management Roles (FMT_SMR.1.2)	DAC.2, DAC.8, IA.11, Audit.14, 16, 17, 18, 19, 22, 23
Abstract Machine Testing (FPT_AMT.1.1)	ENF.3
Reference Mediation (FPT_RVM.1.1)	ENF.1
Domain Separation (FPT_SEP.1.1)	ENF.2
Domain Separation (FPT_SEP.1.2)	ENF.2
Reliable Time Stamps (FPT_STM.1.1)	Audit.12

Table 4: SFR - IT SF Mapping

a. FAU_GEN.1.1 implicitly includes the requirement not to store password information in the audit trail as required by IT SF Audit.15.

b. Supplying a new password is stated in ITSF IA.11, and it is the process through which a user enters a new password that enforces the construction of the password and hence the probability of guessing the correct password.

7.4.2 Justification for Compliance of Assurance Measures

Section 6.3 shows that all assurance requirements are met by an appropriate assurance measure.

7.5 PP Claims and Rationale

7.5.1 PP Reference

The TOE meets all of the requirements of the Controlled Access Protection Profile, which is defined in [CAPP].

7.5.2 PP Tailoring

The security functional requirements for the TOE are as defined in [CAPP] with refinements as necessary and appropriate for a Security Target. These refinements are detailed in section 5.1.2.

7.5.3 PP Additions

There are no additional security functional requirements for the TOE beyond that defined in [CAPP]. There is one additional security requirement for the IT environment which is detailed in section 5.4. This relates to the requirements placed on the OpenBoot PROM in support of protecting the workstation in the environment.

There are no additional TOE security objectives to those contained in [CAPP]. The security objectives for the TOE environment in this security target may be regarded as additional to those contained in [CAPP], although they are deemed to be broadly equivalent, and refined due to the specific environment assumed for the Solaris 8 product.

7.5.4 PP Rationale

The objectives used in this Security Target are derived from [CAPP]. The differences are minor and result from refinements appropriate to a Security Target where a specific product and the assumed environment are being described.

The SFRs used in this Security Target are derived from [CAPP], and have been

refined as required for inclusion in a Security Target.

The rationale presented in this document describing why the SFRs are appropriate to meet the security objectives has been taken from [CAPP] also. Because of the similarities between the objectives and SFRs contained in this Security Target and in [CAPP], the justification provided in [CAPP] is also appropriate for this Security Target.

8 Security Policy Model

For the purpose of an evaluation under the Common Criteria, the requirements for an Informal Security Policy Model (ISPM) are met by the information presented in sections 1-7. The material presented in this section is provided for guidance where a greater depth of knowledge is required on the underlying principles of security.

8.1 Axioms

8.1.1 Axiom P–1: Discretionary Access Control

No Solaris user shall be permitted to access (view or modify) any information contained in the Solaris 8 system unless that access has been explicitly granted by an authorized Solaris 8 system user, or the access is inherent in the specification of the Solaris 8 system interface and violates no other policy axioms.

8.1.2 Axiom P–2: Identification and Authentication

No use of the Solaris system shall be permitted for anyone except users who have properly identified and authenticated themselves to the Solaris 8 system and have been previously authorized such access by a Solaris 8 administrator.

8.1.3 Axiom P–3: Auditing

On direction from a Solaris administrator, the Solaris 8 system shall maintain a record (audit trail) of all specified security-relevant actions taken by specified Solaris 8 system users.

8.1.4 Axiom P–4: Reference Monitor Principle

No Solaris system user shall, without authorization, be permitted to affect the operation of the Solaris 8 system such that any of these policy axioms is violated.

8.2 Policies

8.2.1 Discretionary Access Control

The discretionary policy is based on subject identity and the *permissions* and *ownership* of objects. Most objects have the *owner-uid*, *owner-gid*, and *permissions* attributes. Subjects (processes) have the *effective-uid, effective-gid,* and *supple-*

mental groups attributes. In most cases, these object and subject attributes are interpreted as described below; for some objects, the policy is extended and other attributes may be considered as well. These exceptions are described in the "Interpretations" section for each object type.

8.2.1.1 Permissions

There are three access restriction modes for named objects: read, write, and execute:

- Read permission grants access to view the object's contents.
- Write permission grants access to modify the object's contents.
- Execute permission for file objects grants access to execute the file as a program; for directory objects, it grants access to use the directory in a pathname; and for most other object types, it is ignored.

For some objects, separate access modes are defined for the object's owner, individually named users, owning group, individually named groups, and all others. Appropriately privileged subjects may override the mode restrictions.

8.2.1.2 Ownership

The *owner-uid* attribute of the object controls discretionary access for modifying most attributes of the object. If the process's *effective-uid* matches the object's *owner-uid*, the process is considered the *owner* of the object and can modify the object's attributes.

In general, the *owner-uid* is set when the object is created, and cannot be changed, except by a subject privileged to change the ownership *(file_owner*). The *owner-gid* is set when the object is created, and can only be changed by the *owner*, and a subject privileged to change the ownership (*file_owner*). An unprivileged subject can only set the *owner-gid* to a group which the owner is a member of. The permissions are set when the object is created (granting access for the *owner* only unless the subject's default has been changed), and can only be changed by the *owner*, and a subject privileged to override the ownership check (*file_owner*).

Ownership is interpreted in more different ways for different object types than permissions (which have a fairly constant meaning for all object types).

8.2.1.3 Discretionary Access Control Policy

Rule **D–1** (owner controls permissions):

Each named object has an owner, which is typically the associated owner of the subject responsible for creation of the object. No subject **S** except the owner (or a subject authorized to act as the owner), unless **privileged(S,** override the ownership check**)**, may grant or rescind discretionary access to an object.

Rule **D–2** (precedence):
For objects with multiple access mode categories, the access rights of the subject are granted based on that subject's user or group and the access mode in the order (as applicable) owner, individually named user, owning group together with any other matching individually named groups, the combination of all matching individually named groups, all others, even though the subject's user or group may be a member of another category.

Rule **D–3** (inheritance):
A subject may inherit access to an object if the subject responsible for the inheriting subject's creation already had access to the object at the time of the inheriting subject's creation.

Rule **D–4** (preserving access):
A subject may request access to an object directly from the TCB. After a subject is granted access to the object in a particular mode, such access rights cannot be revoked unless the subject specifically relinquishes access.

Rule **D–5** (permanent propagation to others):
Subjects may permanently propagate their identity and access rights to other users by attaching indicators termed set-user-ID to executable objects.
Subjects may permanently propagate their group access rights to other users by attaching indicators termed set-group-ID to executable objects.

Rule **D–6** (temporary propagation to others):
Subjects may temporarily propagate their access rights to other users by passing open file descriptors across Unix Domain sockets or STREAMS.

8.2.2 Identification and Authentication

Identification and Authentication policy is applied to users, in order to allow creation of subjects with appropriate attributes on behalf of those users.

Unprivileged processes (subjects) in a system are created only on behalf of users who have been properly authenticated by the TCB. The authentication mechanism uses passwords to validate user identity.

Each distinct user name has associated with it a distinct numeric user ID.

After a user is authenticated, all subjects created to act on his behalf are tagged with an Audit User ID identifying that user. The Audit User ID is included in all audit records for those subjects, and is carried along with all operations performed by (or on behalf of) those subjects.

8.2.2.1 Identification and Authentication Policy

Rule **A–1** (identification):
Before performing any security-related actions, users are required to identify themselves.

Rule **A–2** (authentication):
The identity of users is authenticated via user-specific authentication data that is maintained by the TCB.

Rule **A–3** (protection):
The authentication data is protected so that it cannot be accessed by any unauthorized user.

8.2.3 Reference Monitor

The Reference Monitor principle requires that a system provide a Reference Validation Mechanism (RVM) or mechanisms which meet the three Reference Monitor requirements.

8.2.3.1 Reference Monitor Policy

Rule **R–1** (Always invoked):
The RVM validates all references to the protected resources in the system, permitting or refusing the attempted reference according to the system's security policy. The system must not provide interfaces which explicitly bypass the security policy.

Rule **R–2** (Tamperproof):
The RVM must be protected from tampering or other external interference. The system must not contain flaws which permit the RVM to be disabled or otherwise misled.

Rule **R–3** (Small enough to be analyzed):
The RVM must be simple enough that its adherence to the other two rules is evident from inspection. It must be well structured and easily understandable.

Solaris provides a weak implementation of the Reference Monitor principle [Anderson82]. The Solaris Reference Monitor includes the entire Solaris Trusted Computing Base (the TCB is partially defined by the main body of this document). It is a weak implementation of the analysis principle because the RVM is very large and not particularly well structured. Although the intent of the design is that the RVM applies an appropriate policy to controlling all references, and that there are no flaws in the system which permit tampering with the RVM's protected code and data, the internal architecture of the standard system is too complex to provide strong assurance that those requirements are met.

The TCB is always invoked by unprivileged processes when referring to system resources. Although back doors for direct access to internal TCB data structures exist (such as device special files allowing access to kernel memory and raw disk), access to these is prohibited by the basic access control mechanisms, and available only to privileged subjects. Since the only way for unprivileged subjects to manipulate protected resources is through well-defined system call or window system protocol request, and privileged subject interfaces, the first RVM requirement is satisfied.

The TCB is designed to be tamper proof. The kernel part of the TCB is completely isolated from unprivileged access by protections on the kernel program files and special files. The privileged subjects are isolated by access control on their program files, by internal checks to refuse invalid requests, and by the basic kernel-provided process isolation mechanisms. The access control mechanism used to protect TCB programs and data involves DAC, which is used to protect all TCB programs and data files.

Many privileged functions are not provided by the kernel, but rather by independent, isolated, privileged processes which simply service user requests across a well-defined communication interface. It is only the possibility of implementation errors in the TCB software, and the difficulty, caused by the TCB's size and complexity, of demonstrating that no errors exist, that cause this implementation not to meet the TCSEC B2 assurance requirements of a true RVM.

This Page Intentionally Left Blank