Oracle® Solaris 11 Security Guidelines

Part No: E36837-02
August 2014

ORACLE

Contents

Tables

Using This Documentation

- **Overview** – Provides an overview of Oracle Solaris security features and the guidelines for using those features to harden and protect an installed system and its applications.
- **Audience** – System administrators, security administrators, application developers, and auditors who develop, deploy, or assess security on Oracle Solaris 11 systems.
- **Required knowledge** – Site security requirements.

Product Documentation Library

Late-breaking information and known issues for this product are included in the documentation library at http://www.oracle.com/pls/topic/lookup?ctx=E36784.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Feedback

Provide feedback about this documentation at http://www.oracle.com/goto/docfeedback.

♦ ♦ ♦ CHAPTER 1

About Oracle Solaris Security

Oracle Solaris is a robust, premier enterprise operating system that offers proven security features. With a sophisticated network-wide security system that controls the way users access files, protect system databases, and use system resources, Oracle Solaris 11 addresses security requirements at every layer. While traditional operating systems can contain inherent security weaknesses, the flexibility of Oracle Solaris 11 enables it to satisfy a variety of security objectives from enterprise servers to desktop clients. Oracle Solaris is fully tested and supported on a variety of SPARC and x86-based systems from Oracle and on other hardware platforms from third-party vendors.

What's New in Security Features in Oracle Solaris 11.2

This section highlights information for existing customers about important new security features in this release.

- You can assess the compliance of your systems to security standards with the new `compliance` command. It enables you to assess and report the compliance of your system to industry standard security benchmarks, including PCI-DSS. For details, see "Oracle Solaris 11.2 Security Compliance Guide " and the `compliance`(1M) man page.
- The Cryptographic Framework feature of Oracle Solaris is validated at FIPS 140-2, Level 1 for userland and kernel functions in the Oracle Solaris 11.1 SRU 5.5 and Oracle Solaris 11.1 SRU 3 releases.

    - For a list of Oracle FIPS 140-validated products, see Oracle FIPS 140 Software Validations (http://www.oracle.com/technetwork/topics/security/fips140-software-validations-1703049.html).
    - For information about enabling FIPS 140 mode on your system, see "Using a FIPS 140 Enabled System in Oracle Solaris 11.2 ".
- Oracle Solaris 11.1 is certified under the Canadian Common Criteria Scheme. See "Oracle Solaris 11 Common Criteria EAL4+ Certification" on page 27.
- The audit service can use the Oracle Audit Vault to store, review, and analyze audit records. See "Using Oracle Audit Vault and Database Firewall for Storage and Analysis of Audit Records" in "Managing Auditing in Oracle Solaris 11.2 ".
- Verified boot protects the boot process from threats on Oracle SPARC T5 Series servers and Oracle SPARC T7 Series servers. For more information, see "Using Verified Boot" in "Securing Systems and Attached Devices in Oracle Solaris 11.2 ".
- You can secure Automatic Installation (AI) installations with certificates and keys for the install server, for specified client systems, for all clients of a specified install service, and for any other AI clients. Secure AI protects the transmission of Oracle Solaris packages to your systems. See "Increasing Security for Automated Installations" in "Installing Oracle Solaris 11.2 Systems ".
- A new group installation package is available, `pkg:/group/system/solaris-minimal-server`. For a description and a comparison of group package contents, see "Oracle Solaris 11.2 Package Group Lists ".
- You can install Kerberos clients by using AI, so that the client is a Kerberized system at first boot. See "How to Configure Kerberos Clients Using AI" in "Installing Oracle Solaris 11.2 Systems ".
- In this release, physical global zones, called Immutable Global Zones, and virtual global zones, called Oracle Solaris Kernel Zones, can be read-only. Immutable global zones are slightly more powerful than Kernel Zones, but neither can permanently change the hardware or configuration of the system. Read-only zones boot faster and are more secure than zones that allow writes.

  For maintenance, immutable global zones define a special set of processes, called the Trusted Computing Base (TCB) that can be configured through a protected login called the Trusted Path. For more information, see Chapter 12, "Configuring and Administering Immutable Zones," in "Creating and Using Oracle Solaris Zones ". For information about zone configuration resources, see "Introduction to Oracle Solaris Zones ". See also the `mwac`(5) and `tpd`(5) man pages.

  Oracle Solaris Kernel Zones are useful for deploying a compliant system. For example, you can configure a compliant system, create a Unified Archive, then deploy the image as a kernel zone. For more information, see the `solaris-kz`(5) man page, "Creating and Using Oracle Solaris Kernel Zones ", "Oracle Solaris Zones Overview" in "Introduction to Oracle Solaris 11.2 Virtualization Environments ", and "Using Unified Archives for System Recovery and Cloning in Oracle Solaris 11.2 ".
- New features in user and process rights include the following:
    - Time-based and location-based access control to PAM services

- Authorization Roles Managed on RBAC (ARMOR) predefined roles
- Rights profiles that force users to provide a password before running a privileged action
- The Network Observability and System Observability rights profiles for running the diagnostic commands `ipstat`, `tcpstat`, `snoop`, and `intrstat` with privilege and without being `root`

For details, see "What's New in Rights in Oracle Solaris 11.2" in "Securing Users and Processes in Oracle Solaris 11.2 ".

- IKE Version 2 (IKEv2) provides the latest IKE protocol for automatic key management of IPsec-protected network packets. For details, see "What's New in Network Security in Oracle Solaris 11.2" in "Securing the Network in Oracle Solaris 11.2 ".
- The Oracle Hardware Management Pack (HMP) supplies command-line tools for configuring and updating firmware. For information about how to use HMP securely with other Oracle hardware products such as network switches and network interface cards, see "Oracle Hardware Management Pack for Oracle Solaris Security Guide ".

Oracle Solaris 11 Security After Installation

Oracle Solaris is installed "secure by default" (SBD). This security posture protects the system from intrusion and monitors login attempts, among other security features.

System Access Is Limited and Monitored

Initial user and `root` role accounts – The initial user account can log in from the console. This account is assigned the `root` role. The password for the initial user and the `root` accounts is identical at installation.

- After logging in, the initial user can assume the `root` role to further configure the system. Upon assuming the role, the user is prompted to change the `root` password. Note that no role can log in directly, including the `root` role.
- The initial user is assigned defaults from the `/etc/security/policy.conf` file. The defaults include the Basic Solaris User rights profile and the Console User rights profile. These rights profiles enable users to read and write to a CD or DVD, run any command on the system without privilege, and stop and restart their system when sitting at the console.
- The initial user account is also assigned the System Administrator rights profile. Therefore, without assuming the `root` role, the initial user has some administrative rights, such as the right to install software and manage the naming service.

Password requirements – User passwords must be at least six characters long, and have at least two alphabetic characters and one non-alphabetic character. Passwords are hashed by

using the SHA256 algorithm. When changing their password, all users including the root role must conform to these password requirements.

Limited network access – After installation, the system is protected from intrusion over the network. Remote login by the initial user is allowed over an authenticated, encrypted connection with the ssh protocol. This is the only network protocol that accepts incoming packets. The ssh key is wrapped by the AES128 algorithm. With encryption and authentication in place, the user can reach the remote system without interception, modification, or spoofing.

Recorded login attempts – The audit service is enabled for all login/logout events (login, logout, switching user, starting and stopping an ssh session, and screen locking) and for all non-attributable (failed) logins. Because the root role cannot log in, the name of the user who is acting as root is recorded in the audit trail. The initial user can review the audit logs by a right granted through the System Administrator rights profile.

Kernel, File, and Desktop Protections Are in Place

After the initial user is logged in, the kernel, file systems, system files, and desktop applications are protected by file permissions, privileges, and user rights. User rights are also known as *role-based access control* (RBAC).

Kernel protections – Many daemons and administrative commands are assigned just the privileges that enable them to succeed. Many daemons are run from special administrative accounts that do not have root (UID=0) privileges, so they cannot be hijacked to perform other tasks. These special administrative accounts cannot log in. Devices are protected by privileges.

File systems – By default, all file systems are ZFS file systems. The user's umask is 022, so when a user creates a new file or directory, only the user is allowed to modify it. Members of the user's group are allowed to read and search the directory, and read the file. Logins that are outside the user's group can list the directory and read the file. The default directory permissions are drwxr-xr-x (755). The file permissions are -rw-r--r-- (644).

System files – System configuration files are protected by file permissions. Only the root role or a user who is assigned the right to edit a specific system file can modify a system file.

Desktop applets – Desktop applets are protected by rights management. Therefore, administrative actions, such as the addition of remote printers in Print Manager, are restricted to users and roles who have administrative rights for printing.

Oracle Hardware Management Package

The Oracle Hardware Management Package provides a set of utilities for configuring, managing, and monitoring Oracle servers. This value-add set of tools for Oracle hardware is always available. It can automatically deliver certain hardware-related information to

ILOM to complete the view that it has of system hardware. For information about the utilities and security, see the Systems Management and Diagnostics Documentation"> (http://www.oracle.com/goto/ohmp/docs).

Oracle Solaris Configurable Security

In addition to the solid foundation that Oracle Solaris security defaults provide, the security posture of a Oracle Solaris system is highly configurable to satisfy a range of security requirements.

The following sections provide a short introduction to the security features of Oracle Solaris. The descriptions include references to more detailed explanations and to procedures in this guide and other Oracle Solaris system administration guides that demonstrate these features.

Protecting Data

Oracle Solaris protects data from booting through installation, use, and archiving.

File Permissions and Access Control Entries

The first line of defense for protecting objects in a file system are the default UNIX permissions that are assigned to every file system object. UNIX permissions support assigning unique access rights to the owner of the object, to a group assigned to the object, as well as to anyone else. Additionally, the default file system, ZFS, supports access control lists (ACLs), which more finely control access to individual or groups of file system objects.

For more information, see the following:

- For an overview of file permissions, see "Using UNIX Permissions to Protect Files" in "Securing Files and Verifying File Integrity in Oracle Solaris 11.2 ".
- For an overview and examples of protecting ZFS files, see Chapter 7, "Using ACLs and Attributes to Protect Oracle Solaris ZFS Files," in "Managing ZFS File Systems in Oracle Solaris 11.2 " and the man pages.
- For instructions about setting ACLs on ZFS files, see the `chmod`(1) man page.

Cryptographic Services

The Cryptographic Framework feature of Oracle Solaris and the Key Management Framework (KMF) feature of Oracle Solaris provide central repositories for cryptographic services and key

management. Hardware, software, and end users have seamless access to optimized algorithms. KMF provides a unified interface for otherwise different storage mechanisms, administrative utilities, and programming interfaces for various public key infrastructures (PKIs).

The Cryptographic Framework provides a common store of algorithms and PKCS #11 libraries to handle cryptographic requirements. The PKCS #11 libraries are implemented according to the RSA Security Inc. PKCS #11 Cryptographic Token Interface (Cryptoki) standard. Cryptographic services, such as encryption and decryption for files, are available to regular users.

KMF provides tools and programming interfaces for centrally managing public key objects, such as X.509 certificates and public/private key pairs. The formats for storing these objects can vary. KMF also provides a tool for managing policies that define the use of X.509 certificates by applications. KMF supports third-party plugins.

For more information, see the following:

- Selected man pages include `cryptoadm`(1M), `encrypt`(1), `mac`(1), `pktool`(1), and `kmfcfg`(1).
- For an overview of cryptographic services, see Chapter 1, “Cryptographic Framework,” in “Managing Encryption and Certificates in Oracle Solaris 11.2 ” and Chapter 4, “Key Management Framework,” in “Managing Encryption and Certificates in Oracle Solaris 11.2 ”.
- For examples of using the Cryptographic Framework, see Chapter 3, “Cryptographic Framework,” in “Managing Encryption and Certificates in Oracle Solaris 11.2 ” and the man pages.
- To enable the Cryptographic Framework FIPS 140 provider, see “How to Create a Boot Environment with FIPS 140 Enabled” in “Managing Encryption and Certificates in Oracle Solaris 11.2 ” .

Oracle Solaris ZFS File System

ZFS is the default file system for Oracle Solaris 11. The ZFS file system fundamentally changes the way Oracle Solaris file systems are administered. ZFS is robust, scalable, and easy to administer. Because file system creation in ZFS is lightweight, you can easily establish quotas and reserved space. UNIX permissions and ACLs protect files, and you can encrypt the entire dataset at creation. Oracle Solaris rights management supports the delegated administration of ZFS datasets, that is, users who are assigned a limited set of privileges can administer ZFS datasets.

For more information, see the following:

- “User Rights Management” in “Securing Users and Processes in Oracle Solaris 11.2 ”
- Chapter 1, “Oracle Solaris ZFS File System (Introduction),” in “Managing ZFS File Systems in Oracle Solaris 11.2 ”

- “Oracle Solaris ZFS and Traditional File System Differences” in “Managing ZFS File Systems in Oracle Solaris 11.2 ”
- Chapter 5, “Managing Oracle Solaris ZFS File Systems,” in “Managing ZFS File Systems in Oracle Solaris 11.2 ”
- “How to Remotely Administer ZFS With Secure Shell” in “Managing Secure Shell Access in Oracle Solaris 11.2 ”
- Selected man pages include `zfs`(1M) and `zfs`(7FS).

Java Cryptography Extension

Java provides the Java Cryptography Extension (JCE) for developers of Java applications. For more information, see Java SE Security (http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136007.html).

Protecting and Isolating Applications

Applications can be entry points for malware and malicious users. In Oracle Solaris, these threats are mitigated by the use of privileges and the containment of applications within zones. Applications can run with just the privileges that the application needs, so a malicious user does not have root privileges to access the rest of the system. Zones can limit the extent of an attack. Attacks on applications in a non-global zone can affect processes in that zone only, not the zone's host system.

Address space layout randomization (ASLR) and the Service Management Facility (SMF) are additional features that protect applications. ASLR makes it difficult for intruders to hijack an executable, and SMF features enable administrators to restrict starting, stopping, and using an application.

Privileges in Oracle Solaris

Privileges are fine-grained, discrete rights on processes that are enforced in the kernel. Oracle Solaris defines over 80 privileges, ranging from basic privileges like `file_read` to more specialized privileges like `proc_clock_highres`. Privileges can be granted to a process, a user, or a role. Many Oracle Solaris commands and daemons run with just the privileges that are required to perform their task. Privilege-aware programs can prevent intruders from gaining more privileges than the program itself uses.

The use of privileges is also called *process rights management*. Privileges enable organizations to specify, hence limit, which privileges are granted to services and processes that run on their systems.

For more information, see the following:

- "Process Rights Management" in "Securing Users and Processes in Oracle Solaris 11.2 "
- Chapter 2, "Developing Privileged Applications," in "Developer's Guide to Oracle Solaris 11 Security "
- Selected man pages include `ppriv`(1) and `privileges`(5).

Oracle Solaris Zones

The Oracle Solaris Zones software partitioning technology enables you to maintain the one-application-per-server deployment model while simultaneously sharing hardware resources.

Zones are virtualized operating environments that enable multiple applications to run in isolation from each other on the same physical hardware. This isolation prevents processes that run within a zone from monitoring or affecting processes that run in other zones, viewing each other's data, or manipulating the underlying hardware. Zones also provide an abstraction layer that separates applications from physical attributes of the system on which they are deployed, such as physical device paths and network interface names.

In Oracle Solaris 11.2, you can configure immutable root file systems.

For more information, see the following:

- "Configuring Read-Only Zones" in "Creating and Using Oracle Solaris Zones "
- "Introduction to Oracle Solaris Zones "
- Selected man pages include `brands`(5), `zoneadm`(1M), and `zonecfg`(1M).

Address Space Layout Randomization

Address space layout randomization (ASLR) randomizes the addresses that are used by a given program. ASLR can prevent certain types of attacks that are based on knowing the exact location of certain memory ranges, and can detect the attempt when it likely stops the program. For more information, see "Address Space Layout Randomization" in "Securing Systems and Attached Devices in Oracle Solaris 11.2 " and "How to Verify That ASLR Is Enabled" on page 31.

Service Management Facility

Services are persistently running applications. A service can represent a running application, the software state of a device, or a set of other services. The Service Management Facility (SMF) feature of the Oracle Solaris is used to add, remove, configure, and manage services. SMF uses rights management to control access to service management functions on the system. In

particular, SMF uses authorizations to determine who can manage a service and what functions that person can perform.

SMF enables organizations to control access to services, as well as to control how those services are started, stopped, and refreshed.

For more information, see the following:

- "Managing System Services in Oracle Solaris 11.2 "
- "How to Assign Specific Privileges to the Apache Web Server" in "Securing Users and Processes in Oracle Solaris 11.2 "
- Selected man pages include `svcadm`(1M), `svcs`(1), and `smf`(5).

Protecting Users and Assigning Additional Rights

Users are assigned a basic set of privileges, rights profiles, and authorizations from the `/etc/security/policy.conf` file, similar to the initial user as described in "System Access Is Limited and Monitored" on page 13. These rights are configurable. You can deny basic rights and increase the rights for a user.

Oracle Solaris protects users with flexible complexity requirements for passwords, authentication that is configurable for different site requirements, and user rights management, which uses rights profiles, authorizations, and privileges to limit and distribute administrative rights to trusted users. Additionally, special shared accounts called *roles* assign the user just those administrative rights when the user assumes the role. The Authorization Rules Managed On RBAC (ARMOR) package provides predefined roles.

Passwords and Password Constraints

Strong user passwords help defend against attacks involving brute force guessing.

Oracle Solaris has a number of features that you can use to configure user passwords to your site requirements. You can specify password length, content, frequency of change, and modification requirements, and keep a password history. A dictionary of passwords to be avoided is provided. Several possible password hash algorithms are available. The default is SHA256.

For more information, see the following:

- "Maintaining Login Control" in "Securing Systems and Attached Devices in Oracle Solaris 11.2 "
- "Securing Logins and Passwords" in "Securing Systems and Attached Devices in Oracle Solaris 11.2 "
- Selected man pages include `passwd`(1) and `crypt.conf`(4).

Pluggable Authentication Modules

The Pluggable Authentication Module (PAM) framework enables administrators to coordinate and configure user authentication requirements for accounts, credentials, sessions, and passwords without modifying the services that require authentication.

The PAM framework enables organizations to customize the user authentication experience as well as account, session, and password management functionality. System entry services such as `login` and `ssh` use the PAM framework to secure all entry points for the freshly installed system. PAM enables the replacement or modification of authentication modules in the field to secure the system against any newly found weaknesses without requiring changes to any system services that use the PAM framework.

Oracle Solaris delivers a broad set of PAM modules and configurations to meet most site policies. For more information, see the following:

- Chapter 1, "Using Pluggable Authentication Modules," in "Managing Kerberos and Other Authentication Services in Oracle Solaris 11.2 "
- "Writing Applications That Use PAM Services" in "Developer's Guide to Oracle Solaris 11 Security "
- `pam.conf`(4) man page

User Rights Management

User rights in Oracle Solaris are governed by the security principle of least privilege. Organizations can selectively grant administrative rights to users or roles according to the unique needs and requirements of the organization. They can also deny rights to users when required. Rights are implemented as privileges on processes and authorizations on users or SMF methods. Rights profiles provide a convenient way to collect privileges and authorizations into a bundle of related rights.

For more information, see the following:

- "Securing Users and Processes in Oracle Solaris 11.2 "
- Selected man pages include `auths`(1), `privileges`(5). `profiles`(1), `rbac`(5), `roleadd`(1M), `roles`(1), and `user_attr`(4).

Securing Network Communications

Network communications can be protected by features such as firewalls, TCP wrappers on networked applications, and encrypted and authenticated remote connections.

Packet Filtering

Packet filtering provides basic protection against network-based attacks. Oracle Solaris includes the IP Filter feature and TCP wrappers.

Firewall

The IP Filter feature of Oracle Solaris creates a firewall to ward off network-based attacks.

Specifically, IP Filter provides stateful packet filtering capabilities and can filter packets by IP address or network, port, protocol, network interface, and traffic direction. It also includes stateless packet filtering and the capability to create and manage address pools. In addition, IP Filter also has the capability to perform network address translation (NAT) and port address translation (PAT).

For more information, see the following:

- For an overview of IP Filter, see Chapter 4, “About IP Filter in Oracle Solaris,” in “Securing the Network in Oracle Solaris 11.2 ”.
- For examples of using IP Filter, see Chapter 5, “Configuring IP Filter,” in “Securing the Network in Oracle Solaris 11.2 ” and the man pages.
- For information and examples about the syntax of the IP Filter policy language, see the `ipnat`(4) man page.
- Selected man pages include `ipfilter`(5), `ipf`(1M), `ipnat`(1M), `svc.ipfd`(1M), and `ipf`(4).

TCP Wrappers

TCP wrappers provide access control for internet services. When various internet (`inetd`) services are enabled, the `tcpd` daemon checks the address of a host requesting a particular network service against an ACL. Requests are granted or denied accordingly. TCP wrappers also log host requests for network services in `syslog`, which is a useful monitoring function.

The Secure Shell (`ssh`)and `sendmail` features of Oracle Solaris are configured to use TCP wrappers. Network services that have a one-to-one mapping to executable files, such as `proftpd` and `rpcbind`, are candidates for TCP wrappers.

TCP wrappers support a rich configuration policy language that enables organizations to specify security policy not only globally but on a per-service basis. Further access to services can be permitted or restricted based upon host name, IPv4 or IPv6 address, netgroup name, network, and even DNS domain.

For information about TCP wrappers, see the following:

- "How to Use TCP Wrappers" on page 45
- For information and examples of the syntax of the access control language for TCP wrappers, see the `hosts_access`(4) man page.
- Selected man pages include `tcpd`(1M) and `inetd`(1M).

Remote Access

Remote access attacks can damage a system and a network. Oracle Solaris provides defense in depth for network transmissions. Defense features include encryption and authentication checks for data transmission, login authentication, the disabling of unnecessary remote services.

IPsec and IKE

IP security (IPsec) protects network transmissions by authenticating the IP packets, by encrypting them, or by doing both. Because IPsec is implemented well below the application layer, Internet applications can take advantage of IPsec without requiring modifications to their code.

IPsec and its automatic key exchange protocol, IKE, use algorithms from the Cryptographic Framework. Additionally, the Cryptographic Framework provides a central keystore. When IKE is configured to use the metaslot, organizations have the option of storing the keys on disk, on an attached hardware keystore, or in a software keystore called *softtoken*.

IPsec and IKE require configuration, so are installed but not enabled by default. When properly administered, IPsec is an effective tool in securing network traffic.

For more information, see the following:

- Chapter 6, "About IP Security Architecture," in "Securing the Network in Oracle Solaris 11.2 "
- Chapter 7, "Configuring IPsec," in "Securing the Network in Oracle Solaris 11.2 "
- "IPsec and FIPS 140" in "Securing the Network in Oracle Solaris 11.2 "
- Chapter 8, "About Internet Key Exchange," in "Securing the Network in Oracle Solaris 11.2 "
- Chapter 9, "Configuring IKEv2," in "Securing the Network in Oracle Solaris 11.2 "
- Selected man pages include `ipsecconf`(1M) and `in.iked`(1M).

Secure Shell

By default, the Secure Shell feature of Oracle Solaris is the only active remote access mechanism on a newly installed system. All other network services are either disabled or in listen-only mode.

Secure Shell creates an encrypted communications channel between systems. Secure Shell can also be used as an on-demand virtual private network (VPN) that can forward X Window system traffic or can connect individual port numbers between a local system and remote systems over an authenticated and encrypted network link.

Thus, Secure Shell prevents a would-be intruder from being able to read an intercepted communication and prevents an adversary from spoofing the system.

For more information, see the following:

- Chapter 1, "Using Secure Shell," in "Managing Secure Shell Access in Oracle Solaris 11.2 "
- "Secure Shell and FIPS 140" in "Managing Secure Shell Access in Oracle Solaris 11.2 "
- Selected man pages include `ssh`(1), `sshd`(1M), `sshd_config`(4), and `ssh_config`(4).

Kerberos Service

The Kerberos feature of the Oracle Solaris enables single sign-on and secure transactions, even over heterogeneous networks where systems run different operating systems and run the Kerberos service.

Kerberos is based on the Kerberos V5 network authentication protocol that was developed at the Massachusetts Institute of Technology (MIT). The Kerberos service offers strong user authentication, as well as integrity and privacy. Using the Kerberos service, you can log in once and access other systems, execute commands, exchange data, and transfer files securely. Additionally, the service enables administrators to restrict access to services and systems.

For more information, see the following:

- "Managing Kerberos and Other Authentication Services in Oracle Solaris 11.2 "
- "FIPS 140 Algorithms and Kerberos Encryption Types" in "Managing Kerberos and Other Authentication Services in Oracle Solaris 11.2 "
- Selected man pages include `kadmin`(1M), `kdcmgr`(1M), `kerberos`(5), `kinit`(1), and `krb5.conf`(4).

Maintaining System Security

Oracle Solaris provides the following features to maintain the security of a system:

- Verified boot – Secures the boot process. Verified boot is disabled by default.
- Package verification – Verifies that the installed packages are identical to the packages in the source repository.
- Audit service – Audits access and use of the system. Auditing is enabled by default.

- File integrity verification – BART manifests can list every file on the system, and comparisons of manifests are used to verify that file integrity is maintained.
- Log files – SMF provides log files for every service. The `syslog` utility provides a central file for naming and configuring logs for system services and can optionally notify administrators of critical events. Other features, such as auditing, also create their own logs.
- Compliance reports – Oracle Solaris provides several security benchmarks against which to assess your system. These assessments produce reports that help you evaluate the security posture of the system.

Verified Boot

Verified boot is an Oracle Solaris feature that secures a system's boot process. This feature protects the system from threats such as the installation of unauthorized kernel modules and trojan applications. By default, verified boot is disabled.

For more information, see Chapter 2, “Protecting Oracle Solaris Systems Integrity,” in “Securing Systems and Attached Devices in Oracle Solaris 11.2 ”.

Package Integrity Verification

After installing or updating packages, you can run the `pkg verify` command to ensure that the packages on your system are identical to the packages from the source repository.

For more information, see the `pkg`(1) man page and “How to Verify Your Packages” on page 31.

Audit Service

Oracle Solaris provides an audit service that collects data about system access and use. The audit data provides a reliable time-stamped log of security-related system events. This data can then be used to assign responsibility for actions that take place on a system.

Auditing is a basic requirement for security evaluation, validation, compliance, and certification bodies. Auditing can also provide a deterrent to potential intruders.

For more information, see the following:

- For a list of audit-related man pages, see Chapter 7, “Auditing Reference,” in “Managing Auditing in Oracle Solaris 11.2 ”.
- For guidelines, see “How to Audit Significant Events in Addition to Login/Logout” on page 42 and the man pages.

- For an overview of auditing, see Chapter 1, “About Auditing in Oracle Solaris,” in “Managing Auditing in Oracle Solaris 11.2 ”.
- For auditing tasks, see Chapter 3, “Managing the Audit Service,” in “Managing Auditing in Oracle Solaris 11.2 ”.

File Integrity Verification

The BART feature of Oracle Solaris enables you to comprehensively validate systems by performing file-level checks of a system over time. After installation, the `pkg verify` command confirms that your source and destination package contents are identical. After package verification, BART manifests can easily and reliably gather information about the files on a system.

BART is a useful tool for integrity management on one system or on a network of systems. A system's files can be compared to the system's original files, and to other system's files. The reports might indicate that a system has not been patched, an intruder has installed unapproved files, or an intruder has changed the permissions or contents of sensitive files, such as the `root`-owned files.

For more information, see the following:

- For guidelines, see “Verifying File Integrity by Using BART” on page 54, “Verifying File Integrity by Using BART” on page 54, and the man pages.
- For an overview of BART, see Chapter 2, “Verifying File Integrity by Using BART,” in “Securing Files and Verifying File Integrity in Oracle Solaris 11.2 ”.
- For examples of using BART, see “About Using BART” in “Securing Files and Verifying File Integrity in Oracle Solaris 11.2 ” and the man pages.
- Selected man pages include `bart`(1M), `bart_rules`(4), and `bart_manifest`(4).

Log Files

The Service Management Facility (SMF) feature of the Oracle Solaris logs the status of its services per service. Many services, such as auditing and Secure Shell, write their own logs. The `syslog` or `rsyslog` daemon writes a centralized log that can inform and warn administrators of critical conditions in many services. For example, auditing can be configured to write summarized auditing records to `syslog`. See the `syslogd`(1M) and `syslog.conf`(4) man pages.

Compliance to Security Standards

The `compliance assess` command provides a snapshot of your system's security posture. The reports from the assessments suggest specific changes to your system to satisfy industry

security benchmarks. For more information, see "Oracle Solaris 11.2 Security Compliance Guide " and the `compliance`(1M) man page.

Labeled Security

Labeled security in Oracle Solaris is provided by the Trusted Extensions feature.

Trusted Extensions Feature in Oracle Solaris

The Trusted Extensions feature of Oracle Solaris is an optionally enabled layer of secure labeling technology that enables data security policies to be separated from data ownership. Trusted Extensions supports both traditional discretionary access control (DAC) policies based on ownership, as well as label-based mandatory access control (MAC) policies. Unless the Trusted Extensions layer is enabled, all labels are equal so the kernel is not configured to enforce the MAC policies. When the label-based MAC policies are enabled, all data flows are restricted based on a comparison of the labels associated with the processes (subjects) requesting access and the objects containing the data.

The Trusted Extensions implementation is unique in its ability to provide high assurance, while maximizing compatibility and minimizing overhead. Trusted Extensions is part of the "Oracle Solaris 11 Common Criteria EAL4+ Certification" on page 27.

Trusted Extensions meets the requirements of the Common Criteria Labeled Security Package (LSP). See "Oracle Solaris 11 Common Criteria EAL4+ Certification" on page 27.

For more information, see the following:

- For information about configuring and maintaining Trusted Extensions, see "Trusted Extensions Configuration and Administration ".
- Selected man pages include `trusted_extensions`(5), `labeladm`(1M), and `labeld`(1M).

Labeled Filesystem

By default, filesystems are assigned a single label in a zone at that same label. You can create a multilevel ZFS dataset, mount it on a Trusted Extensions system, and with appropriate permissions, upgrade and downgrade the files in that dataset. For more information, see "Multilevel Datasets for Relabeling Files" in "Trusted Extensions Configuration and Administration ".

Labeled Network Communications

Trusted Extensions labels network communications. Data flows are restricted based on a comparison of the labels associated with the originating network endpoint and the receiving network endpoint. Gateways and in-between hops must also be labeled to allow the passage of information at the label of the communication. NFS and multilevel ZFS datasets provide additional features on a network.

For more information, see the following:

- "Configuring the Network Interfaces in Trusted Extensions" in "Trusted Extensions Configuration and Administration "
- Chapter 15, "Trusted Networking," in "Trusted Extensions Configuration and Administration "
- Chapter 16, "Managing Networks in Trusted Extensions," in "Trusted Extensions Configuration and Administration "

Trusted Extensions Multilevel Desktop

Unlike most other multilevel operating systems, Trusted Extensions includes a multilevel desktop. Users can be configured to see only their allowed labels. Each label can be configured to require a separate password.

For more information, see "Trusted Extensions User's Guide ". To configure users, see Chapter 11, "Managing Users, Rights, and Roles in Trusted Extensions," in "Trusted Extensions Configuration and Administration ".

Oracle Solaris 11 Common Criteria EAL4+ Certification

Oracle Solaris 11 is certified under the Canadian Common Criteria Scheme at Evaluation Assurance Level 4 (EAL4) and augmented by flaw remediation (EAL4+). EAL4 is the highest level of evaluation mutually recognized by 26 countries under the Common Criteria Recognition Arrangement (CCRA).

The certification is for the Operating System Protection Profile (OSPP) and includes the following extended packages:

- Advanced Management
- Extended Identification and Authentication
- Labeled Security
- Virtualization

For information about the certification, see:

- Oracle Security Evaluations Matrix (http://www.oracle.com/technetwork/topics/security/security-evaluations-099357.html)
- The Common Criteria Recognition Arrangement (http://www.commoncriteriaportal.org/ccra/)
- Operating System Protection Profile (http://www.commoncriteriaportal.org/files/ppfiles/pp0067b_pdf.pdf)

Site Security Policy and Practice

For a secure system or network of systems, your site must have a security policy in place with security practices that support the policy. If you are developing programs or installing third-party programs, you must develop and install those programs securely.

For more information, review the following:

- Importance of Software Security Assurance (http://www.oracle.com/us/support/assurance/overview/index.html)
- Appendix A, “Secure Coding Guidelines for Developers,” in “Developer’s Guide to Oracle Solaris 11 Security ”
- Appendix A, “Site Security Policy,” in “Trusted Extensions Configuration and Administration ”
- “Security Requirements Enforcement” in “Trusted Extensions Configuration and Administration ”
- Keeping Your Code Secure (http://blogs.oracle.com/maryanndavidson/entry/those_who_can_t_do)

♦ ♦ ♦ CHAPTER 2

Configuring Oracle Solaris Security

This chapter describes the actions to take to configure security on your system. The chapter covers installing packages, configuring the system itself, then configuring various subsystems and additional applications that you might need, such as IPsec.

Installing the Oracle Solaris OS

The Oracle Solaris OS is installed by selecting a set of packages called a *group* from a package repository. Different groups supply packages for different uses, such as multipurpose servers, minimally installed systems, and desktop systems. Packages are signed and their secure transfer can be verified.

When you install the Oracle Solaris OS, choose the media that installs the appropriate *group* package, as follows:

- **Oracle Solaris Large Server –** Both the default manifest in an Automated Installer (AI) installation and the text installer install the `group/system/solaris-large-server` group, which provides an Oracle Solaris large server environment.
- **Oracle Solaris Small Server –** The Automated Installer (AI) installation and the text installer optionally install the `group/system/solaris-small-server` group, which provides a useful command-line environment to which you can add packages.
- **Oracle Solaris Minimal Server –** The Automated Installer (AI) installation and the text installer optionally install the `group/system/solaris-minimal-server` group, which provides a minimal command-line environment to which you can add just the packages that you want.

- **Oracle Solaris Desktop –** The Live Media installs the `group/system/solaris-desktop` group, which provides an Oracle Solaris 11 desktop environment.

  To create a desktop system for centralized use, add the `group/feature/multi-user-desktop` group to the desktop server. For more information, see the article: "Optimizing the Oracle Solaris 11 Desktop for a Multiuser Environment ".

For an automated installation using the Automated Installer (AI), see Part III, "Installing Using an Install Server," in "Installing Oracle Solaris 11.2 Systems ".

To guide your media choice, see the following installation and package content guides:

- "Installing Oracle Solaris 11.2 Systems "
- "Creating a Custom Oracle Solaris 11.2 Installation Image "
- "Adding and Updating Software in Oracle Solaris 11.2 "
- "Oracle Solaris 11.2 Package Group Lists "

Initially Securing the System

The following tasks are best performed in order. At this point, the Oracle Solaris is installed and only the initial user who can assume the `root` role has access to the system.

TABLE 2-1 Securing the System Task Map

Task	Description	For Instructions
1. Verify the packages on the system.	Checks that the packages from the installation source are identical to the installed packages.	"How to Verify Your Packages" on page 31
2. Ensure that executables are protected.	Checks that ASLR is enabled.	"How to Verify That ASLR Is Enabled" on page 31
3. Safeguard the hardware settings on the system.	Protects hardware by requiring a password to change hardware settings. On an x86, access to the GRUB menu is controlled. On a SPARC, the `eeprom` command protects the hardware.	"Controlling Access to System Hardware" in "Securing Systems and Attached Devices in Oracle Solaris 11.2 "
3. Disable unneeded services.	Prevents processes that are not part of the system's required functions from running.	"How to Disable Unneeded Services" on page 32
5. Prevent the workstation owner from powering down the system.	Prevents the Console User from shutting down or suspending the system.	"How to Remove Power Management Capability From Users" on page 32
6. Create a login warning message that reflects your site's security policy.	Notifies users before and after authentication that the system is monitored.	"How to Place a Security Message in Banner Files" on page 33 "How to Place a Security Message on the Desktop Login Screen" on page 34

How to Verify Your Packages

Immediately after installation, validate the installation by verifying your packages.

Before You Begin You must assume the `root` role. For more information, see "Using Your Assigned Administrative Rights" in "Securing Users and Processes in Oracle Solaris 11.2 ".

1. **Review the installation log.**
2. **Run the `pkg verify` command.**

   To keep a record, send the command output to a file.

   ```
   # pkg verify > /var/pkgverifylog
   ```

3. **Review the log for any errors.**
4. **If you find errors, reinstall from the media or fix the errors.**

See Also For more information, see the pkg(1) and pkg(5) man pages. The man pages contain examples of using the `pkg verify` command.

How to Verify That ASLR Is Enabled

By default, executable instructions that are tagged are written to unconnected address spaces to reduce the ability of intruders to inject instructions on the executable stack.

Before You Begin You must assume the `root` role. For more information, see "Using Your Assigned Administrative Rights" in "Securing Users and Processes in Oracle Solaris 11.2 ".

1. **Verify that ASLR is enabled.**

   ```
   # sxadm info
   EXTENSION       STATUS                  CONFIGURATION
   aslr            enabled (all)           enabled (all)
   ```

   The value `all` is stronger than the default, and could result in errors in applications that rely on a consecutive stack in memory. For example, databases might rely on a consecutive stack in memory.

2. **If ASLR is disabled, enable the default value and verify that it is in effect.**

   ```
   # sxadm delcust aslr
   # sxadm info
   EXTENSION       STATUS                  CONFIGURATION
   aslr            enabled (tagged-files)  system default (default)
   ```

See Also For debugging purposes, you can turn off ASLR by calling the `sxadm` command on a particular binary. For examples, see the `sxadm`(1M) man page.

How to Disable Unneeded Services

Use this procedure to disable services that are not required on this system.

Before You Begin You must assume the `root` role. For more information, see "Using Your Assigned Administrative Rights" in "Securing Users and Processes in Oracle Solaris 11.2 ".

1. **List the online network services.**

```
# svcs | grep network
online         Sep_07   svc:/network/loopback:default
online         Sep_07   svc:/network/http:apache22
online         Sep_07   svc:/network/nfs/server:default
...
online         Sep_07   svc:/network/ssh:default
```

2. **Disable the services that are not required by this system.**

   For example, if the system is not an NFS server or a web server and their services are online, disable them.

```
# svcadm disable svc:/network/nfs/server:default
# svcadm disable svc:/network/http:apache22
```

See Also For more information, see Chapter 1, "Introduction to the Service Management Facility," in "Managing System Services in Oracle Solaris 11.2 " and the `svcs`(1) man page.

How to Remove Power Management Capability From Users

Use this procedure to prevent users on the console of a system from suspending the system or powering it down. This software solution is not effective if the system hardware can be unplugged by the console user.

Before You Begin You must assume the `root` role. For more information, see "Using Your Assigned Administrative Rights" in "Securing Users and Processes in Oracle Solaris 11.2 ".

1. **Review the contents of the Console User rights profile.**

```
% profiles -p "Console User" info
```

```
name=Console User
desc=Manage System as the Console User
auths=solaris.system.shutdown,solaris.device.cdrw,
            solaris.smf.manage.vbiosd,solaris.smf.value.vbiosd
profiles=Suspend To RAM,Suspend To Disk,Brightness,CPU Power Management,
                Network Autoconf User
help=RtConsUser.html
```

2. **Create a rights profile that includes any rights in the Console User profile that you want users to retain.**

   For instructions, see "How to Create a Rights Profile" in "Securing Users and Processes in Oracle Solaris 11.2 ".

3. **Comment out the Console User rights profile in the `/etc/security/policy.conf` file.**

   ```
   #CONSOLE_USER=Console User
   ```

4. **Assign the rights profile that you created in Step 2.**

   - If you have many users that share a rights profile, setting this value in a rights profile can be a scalable solution.

     ```
     # usermod -P shared-profile username
     ```

   - You can also assign the profile per system in the `policy.conf` file.

     ```
     # pfedit /etc/security/policy.conf...
     #PROFS_GRANTED=Basic Solaris User
     PROFS_GRANTED=shared-profile,Basic Solaris User
     ```

See Also For more information, see "policy.conf File" in "Securing Users and Processes in Oracle Solaris 11.2 " and the `policy.conf`(4) and `usermod`(1M) man pages.

▼ How to Place a Security Message in Banner Files

Use this procedure to create security messages in two banner files that reflect your site's security policy. The `/etc/issue` file displays before authentication, while the `/etc/motd` file displays after authentication.

Note - The sample messages in this procedure do not satisfy U.S. government requirements and likely do not satisfy your security policy. Consult with your company's legal counsel about the content of the security message.

Before You Begin You must become an administrator who is assigned the Administrator Message Edit rights profile. For more information, see "Using Your Assigned Administrative Rights" in "Securing Users and Processes in Oracle Solaris 11.2 ".

1. **Create the `/etc/issue` file and add a security message.**

```
# pfedit /etc/issue
ALERT   ALERT   ALERT   ALERT   ALERT

This machine is available to authorized users only.

If you are an authorized user, continue.

Your actions are monitored, and can be recorded.
```

The `login` command displays the contents of `/etc/issue` before authentication, as do the `ssh`, `telnet`, and FTP services. To display the contents of `/etc/issue` at desktop login, see "How to Place a Security Message on the Desktop Login Screen" on page 34.

For more information, see the `issue`(4) and `pfedit`(1M) man pages.

2. **Add a security message to the `/etc/motd` file.**

```
# pfedit /etc/motd
This system serves authorized users only. Activity is monitored and reported.
```

In Oracle Solaris, the user's initial shell displays the contents of the `/etc/motd` file.

▼ How to Place a Security Message on the Desktop Login Screen

Choose from several methods to create a security message for users to review before authentication, after authentication, or both. The `/etc/issue` file displays before authentication, while the `/etc/motd` file displays after authentication.

For more information, click the System → Help menu from the desktop to bring up the GNOME Help Browser. You can also use the `yelp` command. Desktop login scripts are discussed in the `GDM Login Scripts and Session Files` section of the `gdm`(1M) man page.

Note - The sample message in this procedure does not satisfy U.S. government requirements and likely does not satisfy your security policy. Consult with your company's legal counsel about the content of the security message.

Before You Begin To create a file, you must assume the `root` role. To modify an existing file, you must become an administrator who is assigned the `solaris.admin.edit/`*path-to-existing-file* authorization.

1. **Place a security message on the desktop login screen before authentication by using one of the following options.**

   The options that create a dialog box before authentication use the security message in the `/etc/issue` file from Step 1 of "How to Place a Security Message in Banner Files" on page 33.

   - **OPTION 1: Modify a GDM initialization script to display the security message in a dialog box.**

     The `/etc/gdm` directory contains three initialization scripts that display the security message before authentication and after authentication.

     ```
     # pfedit /etc/gdm/Init/Default
     /usr/bin/zenity --text-info --width=800 --height=300 \
     --title="Security Message"  --filename=/etc/issue
     ```

     For information about editing system files as a non-`root` user, see the `pfedit`(1M) man page.

   - **OPTION 2: Modify the login window to display the security message above the entry field.**

     The login window expands to fit your message. This method does not point to the `/etc/issue` file. You must type the text into the GUI.

Note - The login window, `gdm-greeter-login-window.ui`, is overwritten by the `pkg fix` and `pkg update` commands. To preserve your changes, copy the file to a configuration files directory, and merge its changes with the new file after upgrading the system. For more information, see the `pkg`(5) man page.

---

   a. **Change directory to the login window user interface.**

      ```
      # cd /usr/share/gdm
      ```

   b. **(Optional) Save a copy of the original login window UI.**

      ```
      # cp gdm-greeter-login-window.ui /etc/gdm/gdm-greeter-login-window.ui.orig
      ```

   c. **Add a label to the login window by using the GNOME Toolkit interface designer.**

      The `glade-3` program opens the GTK+ interface designer. You type the security message into a label that displays above the user entry field.

```
# /usr/bin/glade-3 /usr/share/gdm/gdm-greeter-login-window.ui
```

To review the guide for the interface designer, click Development in the GNOME Help Browser. The `glade-3`(1) man page is listed under Applications in the Manual Pages.

d. (Optional) Save a copy of the modified login window UI.

```
# cp gdm-greeter-login-window.ui /etc/gdm/gdm-greeter-login-window.ui.site
```

2. Place a security message on the desktop login screen after authentication by using one of the following options.

The file that creates a dialog box after authentication use the security message in the `/etc/motd` file from Step 2 of "How to Place a Security Message in Banner Files" on page 33.

- **OPTION 1: Place a security message on the desktop after authentication.**

```
# pfedit /etc/gdm/PreSession/Default
/usr/bin/zenity --text-info --width=800 --height=300 \
--title="Security Message"  --filename=/etc/motd
```

Note - The dialog box can be covered by windows in the user's workspace.

- **OPTION 2: Create a desktop file that displays the security message in an additional window after authentication.**

```
# pfedit /usr/share/gdm/autostart/LoginWindow/banner.desktop
[Desktop Entry]
Type=Application
Name=Banner Dialog
Exec=/usr/bin/zenity --text-info --width=800 --height=300 \
--title="Security Message" \
--filename=/etc/motd
OnlyShowIn=GNOME;
X-GNOME-Autostart-Phase=Application
```

To reach the workspace after being authenticated in the login window, the user must close the security message window. For the options to the `zenity` command, see the `zenity`(1) man page.

Example 2-1 Creating a Short Warning Message at Desktop Login

In this example, the administrator types a short message as an argument to the `zenity` command in the desktop file. The administrator also uses the `--warning` option, which displays a warning icon with the message.

```
# pfedit /usr/share/gdm/autostart/LoginWindow/bannershort.desktop
```

```
[Desktop Entry]
Type=Application
Name=Banner Dialog
Exec=/usr/bin/zenity --warning --width=800  --height=150 --title="Security Message" \
--text="This system serves authorized users only. Activity is monitored and reported."
OnlyShowIn=GNOME;
X-GNOME-Autostart-Phase=Application
```

Securing Users

At this point, only the initial user who can assume the `root` role has access to the system. The following tasks are best performed in order before regular users can log in.

TABLE 2-2 Securing Users Task Map

Task	Description	For Instructions
Require strong passwords and regular password changes.	Strengthens the default password constraints on each system.	"How to Set Stronger Password Constraints" on page 38
Configure restrictive file permissions for regular users.	Sets a more restrictive value than `022` for file permissions for regular users.	"How to Set a More Restrictive `umask` Value for Regular Users" on page 41.
Set account locking for regular users.	On systems that are not used for administration, sets account locking system-wide and reduces the number of logins that activate the lock.	"How to Set Account Locking for Regular Users" on page 39
Preselect the `cusa` audit class for all users.	Provides better monitoring and recording of potential threats to the system.	"How to Audit Significant Events in Addition to Login/Logout" on page 42
Create roles.	Distributes discrete administrative tasks to several trusted users so that no one user can damage the system. You can use predefined ARMOR roles, create your own roles, or extend ARMOR with your own roles.	"Managing User Accounts by Using the CLI" in "Managing User Accounts and User Environments in Oracle Solaris 11.2 " "Assigning Rights to Users" in "Securing Users and Processes in Oracle Solaris 11.2 "
Reduce the number of visible GNOME desktop applications.	Prevents users from using desktop applications that can affect security.	See Chapter 11, "Disabling Features in the Oracle Solaris Desktop System," in "Oracle Solaris 11.2 Desktop Administrator's Guide ".
Limit a user's privileges.	Removes basic privileges that users do not need.	"How to Remove Unneeded Basic Privileges From Users" on page 43

How to Set Stronger Password Constraints

Use this procedure if the defaults do not satisfy your site security requirements. The steps are in the order of the variable entries in the `/etc/default/passwd` file.

Before You Begin You must become an administrator who is assigned the `solaris.admin.edit/etc/default/passwd` authorization. For more information, see "Using Your Assigned Administrative Rights" in "Securing Users and Processes in Oracle Solaris 11.2 ".

- **Use the `pfedit` command to make the following changes in the `/etc/default/passwd` file:**

  a. **Require users to change their passwords every four months, but not more frequently than every three weeks.**

  ```
  ## /etc/default/passwd
  ##
  #MAXWEEKS=
  #MINWEEKS=
  MAXWEEKS=13
  MINWEEKS=3
  ```

  b. **Require a password of at least eight characters.**

  ```
  #PASSLENGTH=6
  PASSLENGTH=8
  ```

  c. **Keep a password history.**

  ```
  #HISTORY=0
  HISTORY=10
  ```

  d. **Require a minimum difference from the last password.**

  ```
  #MINDIFF=3
  MINDIFF=4
  ```

  e. **Require at least one uppercase letter.**

  ```
  #MINUPPER=0
  MINUPPER=1
  ```

  f. **Require at least one digit.**

  ```
  #MINDIGIT=0
  MINDIGIT=1
  ```

See Also

- For the list of variables that constrain password creation, see the `passwd`(1) man page.

- For the password constraints in effect after installation, see "System Access Is Limited and Monitored" on page 13.

▼ How to Set Account Locking for Regular Users

Use this procedure to lock regular user accounts after a certain number of failed login attempts.

Note - Roles are shared accounts. Do not set account locking for users who can assume roles or roles because one locked user can lock out the role.

Before You Begin Do not set this protection system-wide on a system that you use for administrative activities. Rather, monitor the administrative system for unusual use and keep it available for administrators.

You must assume the `root` role. For more information, see "Using Your Assigned Administrative Rights" in "Securing Users and Processes in Oracle Solaris 11.2 ".

1. **Set the `LOCK_AFTER_RETRIES` security attribute to `YES`.**

   Choose the scope of the attribute value.

   - **Set system-wide.**

     This protection applies to any user who attempts to use the system.

     ```
     # pfedit /etc/security/policy.conf
     ...
     #LOCK_AFTER_RETRIES=NO
     LOCK_AFTER_RETRIES=YES
     ...
     ```

   - **Set per user.**

     This protection applies only to the user for whom you run this command. If you have many users, this is not a scalable solution.

     ```
     # usermod -K lock_after_retries=yes username
     ```

   - **Create and assign a rights profile.**

     This protection applies to any user or system where you assign this rights profile.

     a. **Create the rights profile.**

        ```
        # profiles -p shared-profile -S ldap
        shared-profile: set lock_after_retries=yes
        ```

```
...
```

For more information on creating rights profiles, see "Creating Rights Profiles and Authorizations" in "Securing Users and Processes in Oracle Solaris 11.2 ".

b. Assign the rights profile to users or system-wide.

If you have many users that share a rights profile, setting this value in a rights profile can be a scalable solution.

```
# usermod -P shared-profile username
```

You can also assign the profile per system in the `policy.conf` file.

```
# pfedit /etc/security/policy.conf
...
#PROFS_GRANTED=Basic Solaris User
PROFS_GRANTED=shared-profile,Basic Solaris User
```

2. Set the `RETRIES` security attribute to `3`.

Choose the scope of the attribute value.

- **Set system-wide.**

```
# pfedit /etc/default/login
...
#RETRIES=5
RETRIES=3
...
```

- **Set per user.**

```
# usermod -K lock_after_retries=3 username
```

- **Create and assign a rights profile.**

Follow the steps in Step 1.3 and to create a rights profile that includes `lock_after_retries=3`.

See Also

- For a discussion of user and role security attributes, see Chapter 8, "Reference for Oracle Solaris Rights," in "Securing Users and Processes in Oracle Solaris 11.2 ".
- Selected man pages include `policy.conf`(4), `profiles`(1), `user_attr`(4), and `usermod`(1M).

How to Set a More Restrictive umask Value for Regular Users

The umask utility sets the file permission bits of user-created files. If the default umask value, 022, is not restrictive enough, set a more restrictive mask by using this procedure.

Before You Begin You must become an administrator who is authorized to edit the skeleton files. The root role is assigned these authorizations. For more information, see "Using Your Assigned Administrative Rights" in "Securing Users and Processes in Oracle Solaris 11.2 ".

1. **View the sample files that Oracle Solaris provides for user shell defaults.**

```
# ls -1a /etc/skel
.bashrc
.profile
local.cshrc
local.login
local.profile
```

2. **Set the umask value in the /etc/skel files that you are going to assign to users.**

   Choose one of the following values:

   - umask 026 – Provides moderate file protection

     (751) – r for group, x for others
   - umask 027 – Provides strict file protection

     (750) – r for group, no access for others
   - umask 077 – Provides complete file protection

     (700) – No access for group or others

See Also For more information, see the following:

- "Managing User Accounts by Using the CLI" in "Managing User Accounts and User Environments in Oracle Solaris 11.2 "
- "Default umask Value" in "Securing Files and Verifying File Integrity in Oracle Solaris 11.2 "
- Selected man pages include useradd(1M) and umask(1).

How to Audit Significant Events in Addition to Login/Logout

Use this procedure to audit administrative commands, system access, and other significant events as specified by your site security policy.

Note - The examples in this procedure might not be sufficient to satisfy your security policy.

Before You Begin You must assume the `root` role. For more information, see "Using Your Assigned Administrative Rights" in "Securing Users and Processes in Oracle Solaris 11.2 ".

1. **Audit all uses of privileged commands by users who are assigned administrative rights profiles and roles.**

   Add the `cusa` audit class to their preselection mask.

   ```
   # usermod -K audit_flags=cusa:no username
   ```

   ```
   # rolemod -K audit_flags=cusa:no rolename
   ```

   The audit classes that the `cusa` meta-class includes are listed in the `/etc/security/audit_class` file.

2. **Record the arguments to audited commands.**

   ```
   # auditconfig -setpolicy +argv
   ```

3. **(Optional) Record the environment in which audited commands are executed.**

   ```
   # auditconfig -setpolicy +arge
   ```

Note - This policy option can be useful when troubleshooting.

See Also

- For information about audit policy, see "Audit Policy" in "Managing Auditing in Oracle Solaris 11.2 ".
- For examples of setting audit flags, see "Configuring the Audit Service" in "Managing Auditing in Oracle Solaris 11.2 " and "Troubleshooting the Audit Service" in "Managing Auditing in Oracle Solaris 11.2 ".
- `auditconfig`(1M) man page

How to Remove Unneeded Basic Privileges From Users

Under particular circumstances, some basic privileges can be removed from a regular or guest user's basic set. For example, Sun Ray users might be prevented from examining the status of processes that they do not own.

Before You Begin You must assume the `root` role. For more information, see "Using Your Assigned Administrative Rights" in "Securing Users and Processes in Oracle Solaris 11.2 ".

1. **List a full definition of the basic privilege set.**

   The following three basic privileges are likely candidates for removal.

```
% ppriv -lv basic
file_link_any
 Allows a process to create hardlinks to files owned by a uid
 different from the process' effective uid.
...
proc_info
 Allows a process to examine the status of processes other
 than those it can send signals to.  Processes which cannot
 be examined cannot be seen in /proc and appear not to exist.
proc_session
 Allows a process to send signals or trace processes outside its
 session.
...
```

2. **Choose the scope of the privilege removal.**

   - **Set system-wide.**

     Any user who attempts to use the system is denied these privileges. This method of privilege removal might be appropriate for a publicly available computer.

```
# pfedit /etc/security/policy.conf
...
#PRIV_DEFAULT=basic
PRIV_DEFAULT=basic,!file_link_any,!proc_info,!proc_session
```

   - **Remove privileges from individual users.**

     - **Prevent a user from linking to a file that the user does not own.**

```
# usermod -K 'defaultpriv=basic,!file_link_any' user
```

     - **Prevent a user from examining processes that the user does not own.**

```
# usermod -K 'defaultpriv=basic,!proc_info' user
```

- **Prevent a user from starting a second session, such as starting an `ssh` session from the user's current session.**

  ```
  # usermod -K 'defaultpriv=basic,!proc_session' user
  ```

- **Remove all three privileges from a user's basic set.**

  ```
  # usermod -K 'defaultpriv=basic,!file_link_any,!proc_info,!proc_session' user
  ```

- **Create and assign a rights profile.**

  This protection applies to any user or system where you assign this rights profile.

  a. **Create the rights profile.**

  ```
  # profiles -p shared-profile -S ldap
  shared-profile: set defaultpriv=basic,!file_link_any,!proc_info,!proc_session
  ...
  ```

  For more information on creating rights profiles, see “Creating Rights Profiles and Authorizations” in “Securing Users and Processes in Oracle Solaris 11.2 ”.

  b. **Assign the rights profile to users or system-wide.**

  If you have many users that share a rights profile, such as Sun Ray or remote users, setting this value in a rights profile can be a scalable solution.

  ```
  # usermod -P shared-profile username
  ```

  You can also assign the profile per system in the `policy.conf` file.

  ```
  # pfedit /etc/security/policy.conf
  ...
  #PROFS_GRANTED=Basic Solaris User
  PROFS_GRANTED=shared-profile,Basic Solaris User
  ```

See Also For more information, see Chapter 1, “About Using Rights to Control Users and Processes,” in “Securing Users and Processes in Oracle Solaris 11.2 ” and the `privileges`(5) man page.

Protecting the Network

At this point, you might have created users who can assume roles, and have created the roles.

From the following network tasks, perform the tasks that provide additional security according to your site requirements. These network tasks strengthen the IP, ARP, and TCP protocols.

TABLE 2-3 Configuring the Network Task Map

Task	Description	For Instructions
Disable the network routing daemon.	Limits access to systems by would-be network sniffers.	"How to Disable the Network Routing Daemon" in "Securing the Network in Oracle Solaris 11.2 "
Prevent the dissemination of information about the network topology.	Prevents the broadcast of packets.	"How to Disable Broadcast Packet Forwarding" in "Securing the Network in Oracle Solaris 11.2 "
	Prevents responses to broadcast echo requests and multicast echo requests.	"How to Disable Responses to Echo Requests" in "Securing the Network in Oracle Solaris 11.2 "
For systems that are gateways to other domains, such as a firewall or a VPN node, turn on strict source and destination multihoming.	Prevents packets that do not have the address of the gateway in their header from moving beyond the gateway.	"How to Set Strict Multihoming" in "Securing the Network in Oracle Solaris 11.2 "
Prevent Denial of Service (DoS) attacks by controlling the number of incomplete system connections.	Limits the allowable number of incomplete TCP connections for a TCP listener.	"How to Set Maximum Number of Incomplete TCP Connections" in "Securing the Network in Oracle Solaris 11.2 "
Prevent DoS attacks by controlling the number of permitted incoming connections.	Specifies the default maximum number of pending TCP connections for a TCP listener.	"How to Set Maximum Number of Pending TCP Connections" in "Securing the Network in Oracle Solaris 11.2 "
Return network parameters to their secure default values.	Increases security that was reduced by administrative actions.	"How to Reset Network Parameters to Secure Values" in "Securing the Network in Oracle Solaris 11.2 "
Add TCP wrappers to network services to limit applications to legitimate users.	Specifies systems that are allowed access to network services, such as FTP.	How to Use TCP Wrappers
Configure a firewall.	Uses the IP Filter feature to provide a firewall.	Chapter 4, "About IP Filter in Oracle Solaris," in "Securing the Network in Oracle Solaris 11.2 " Chapter 5, "Configuring IP Filter," in "Securing the Network in Oracle Solaris 11.2 "
Configure encrypted and authenticated network connections.	Uses IPsec and IKE to protect network transmissions between nodes and networks that are jointly configured with IPsec and IKE.	Chapter 7, "Configuring IPsec," in "Securing the Network in Oracle Solaris 11.2 " Chapter 9, "Configuring IKEv2," in "Securing the Network in Oracle Solaris 11.2 "

▼ How to Use TCP Wrappers

The following steps show three ways that TCP wrappers are used or can be used in Oracle Solaris.

Before You Begin You must assume the `root` role to modify a program to use TCP wrappers.

1. **You do not need to protect the `sendmail` application with TCP wrappers.**

By default, it is protected with TCP wrappers, as described in "Support for TCP Wrappers From Version 8.12 of sendmail" in "Managing sendmail Services in Oracle Solaris 11.2 ".

2. **To enable TCP wrappers for all `inetd` services, see "How to Use TCP Wrappers to Control Access to TCP Services" in "Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ".**

3. **Protect the FTP network service with TCP wrappers.**

   a. **Follow the instructions in the `/usr/share/doc/proftpd/modules/mod_wrap.html` module.**

      Because this module is dynamic, you must load it to use TCP wrappers with FTP.

   b. **Load the module by adding the following instructions to the `proftpd.conf` file:**

```
# pfedit /etc/proftpd.conf
<IfModule mod_dso.c>
    LoadModule mod_wrap.c
</IfModule>
```

   c. **Restart the FTP service.**

```
# svcadm restart svc:/network/ftp
```

Protecting File Systems

ZFS file systems are lightweight and can be encrypted, compressed, and configured with reserved space and disk space quotas.

The `tmpfs` file system can grow without bound. To prevent a denial of service (DoS) attack, complete "How to Limit the Size of the `tmpfs` File System" on page 47.

The following tasks configure a size limit for `tmpfs` and provide a glimpse of the protections that are available in ZFS, the default file system in Oracle Solaris. For additional information, see "Setting ZFS Quotas and Reservations" in "Managing ZFS File Systems in Oracle Solaris 11.2 " and the `zfs`(1M) man page.

TABLE 2-4 Protecting File Systems Task Map

Task	Description	For Instructions
Prevent DoS attacks by managing and reserving disk space.	Specifies the use of disk space by file system, by user or group, or by project.	"Setting ZFS Quotas and Reservations" in "Managing ZFS File Systems in Oracle Solaris 11.2 "

Task	Description	For Instructions
Guarantee a minimum amount of disk space to a dataset and its descendants.	Guarantees disk space by file system, by user or group, or by project.	"Setting Reservations on ZFS File Systems" in "Managing ZFS File Systems in Oracle Solaris 11.2 "
Encrypt data on a file system.	Protects a dataset with encryption and a passphrase to access the dataset at dataset creation.	"Encrypting ZFS File Systems" in "Managing ZFS File Systems in Oracle Solaris 11.2 " "Examples of Encrypting ZFS File Systems" in "Managing ZFS File Systems in Oracle Solaris 11.2 "
Limit the size of the tmpfs file system.	Prevents a malicious user from creating large files in /tmp to slow down the system.	"How to Limit the Size of the tmpfs File System" on page 47

▼ How to Limit the Size of the tmpfs File System

The size of the tmpfs file system is not limited by default. Therefore, tmpfs can grow to fill the available system memory and swap. Because the /tmp directory is used by all applications and users, an application can fill all available system memory. Similarly, an unprivileged user with malicious intent could cause a system slowdown by creating large files in the /tmp directory. To avoid a performance impact, you can limit the size of each tmpfs mount.

You might try several values to achieve best system performance.

Before You Begin To edit the vfstab file, you must become an administrator who is assigned the solaris.admin.edit/etc/vfstab authorization. To reboot the system, you must be assigned the Maintenance and Repair rights profile. The root role has all of these rights. For more information, see "Using Your Assigned Administrative Rights" in "Securing Users and Processes in Oracle Solaris 11.2 ".

1. Determine the amount of memory on your system.

Note - The SPARC T3 series system that is used for the following example has a solid state disk (ssd) for faster I/O and has eight 279.40 MB disks. The system has around 500 GB of memory.

```
% prtconf | head
System Configuration:  Oracle Corporation  sun4v
Memory size: 523776 Megabytes
System Peripherals (Software Nodes):

ORCL,SPARC-T3-4
scsi_vhci, instance #0
disk, instance #4
disk, instance #5
disk, instance #6
```

```
disk, instance #8
```

2. **Compute a memory limit for `tmpfs`.**

   Depending on the size of the system memory, you might want to compute a memory limit of around 20 percent for large systems and around 30 percent for smaller systems.

   So, for a smaller system, use `.30` as the multiplier.

   ```
   10240M x .30 ≈ 340M
   ```

   For a larger system, use `.20` as the multiplier.

   ```
   523776M x .20 ≈ 10475M
   ```

3. **Modify the `swap` entry in the `/etc/vfstab` file with the size limit.**

   ```
   # pfedit /etc/vfstab
   #device     device       mount       FS      fsck    mount mount
   #to mount   to fsck      point       type    pass    at boot options
   #
   ...
   #swap       -            /tmp        tmpfs   -       yes     -
   swap        -            /tmp        tmpfs   -       yes     size=10400m
   /dev/zvol/dsk/rpool/swap   -      -  swap    -       no      -
   ```

4. **Reboot the system.**

   ```
   # reboot
   ```

5. **Verify that the size limit is in effect.**

   ```
   % mount -v
   swap on /system/volatile type tmpfs
   read/write/setuid/devices/rstchown/xattr/dev=89c0006 on Tues Feb 4 14:07:27 2014
   swap on /tmp type tmpfs
   read/write/setuid/devices/rstchown/xattr/size=10400m/dev=89c0006 on Tues ...
   ```

6. **Monitor the memory usage and adjust it to the requirements of your site.**

   The `df` command is somewhat useful. The `swap` command provides the most useful statistics.

   ```
   % df -h /tmp
   Filesystem Size Used Available Capacity Mounted on
   swap          7.  4G     44M    7.4G 1%       /tmp

   % swap -s
   total: 190248k bytes allocated + 30348k reserved = 220596k used,
   7743780k available
   ```

   For more information, see the `tmpfs`(7FS), `mount_tmpfs`(1M), `df`(1M), and `swap`(1M) man pages.

Protecting and Modifying Files

By default, only the `root` role can modify system file permissions. Roles and users who are assigned the `solaris.admin.edit/`*path-to-system-file* authorization can modify that *system-file*. Only the `root` role can search for all files.

TABLE 2-5 Protecting and Modifying Files Task Map

Task	Description	For Instructions
Configure restrictive file permissions for regular users.	Sets a more restrictive value than `022` for file permissions for regular users.	"How to Set a More Restrictive `umask` Value for Regular Users" on page 41
Specify ACLs to protect files at a finer granularity than regular UNIX file permissions.	Extended security attributes can be useful in protecting files. For a caution about using ACLs, see Hiding Within the Trees (http://www.usenix.org/publications/login/2004-02/pdfs/brunette.pdf).	ZFS End-to-End Data Integrity (http://blogs.oracle.com/bonwick/entry/zfs_end_to_end_data)
Maintain system file integrity.	Finds rogue files through a script or by using BART.	"How to Find Files With Special File Permissions" in "Securing Files and Verifying File Integrity in Oracle Solaris 11.2 "

Securing System Access and Use

You can configure Oracle Solaris security features to protect your system use, including applications and services on the system and on the network.

TABLE 2-6 Securing System Access and Use Task Map

Task	Description	For Instructions
Prevent programs from exploiting an executable stack.	Sets a system variable that prevents the exploitation of buffer overflows that exploit the executable stack.	"Protecting Executable Files From Compromising Security" in "Securing Files and Verifying File Integrity in Oracle Solaris 11.2 "
Ensure that binaries that are tagged for address space layout randomization (ASLR) can use ASLR.	Enables ASLR for tagged binaries.	"How to Verify That ASLR Is Enabled" on page 31
Configure auditing.	Customizes audit configuration for coverage and file integrity.	"Using the Audit Service" on page 54
Protect core files that might contain sensitive information.	Creates a directory with limited access that is dedicated to core files.	"Enabling File Paths" in "Troubleshooting System Administration Issues in Oracle Solaris 11.2 " "Administering Your Core File Specifications" in "Troubleshooting System Administration Issues in Oracle Solaris 11.2 "

Task	Description	For Instructions
Protect a web server with SSL Kernel Proxy.	The Secure Sockets Layer (SSL) protocol can be used to encrypt and accelerate web server communications.	Chapter 3, “Web Servers and the Secure Sockets Layer Protocol,” in “Securing the Network in Oracle Solaris 11.2 ”
Create zones to contain applications.	Zones are containers that isolate processes. They can isolate applications and parts of applications. For example, zones can be used to separate a web site's database from the site's web server.	“Introduction to Oracle Solaris Zones ”
Manage resources in zones.	Zones provide a number of tools to manage zone resources.	“Administering Resource Management in Oracle Solaris 11.2 ”

Protecting a Legacy Service With SMF

You can limit application configuration to trusted users or roles by adding the application to the Service Management Facility (SMF) feature of Oracle Solaris, then requiring rights to start, refresh, and stop the service.

For information and procedures see the following:

- “Locking Down Resources by Using Extended Privileges” in “Securing Users and Processes in Oracle Solaris 11.2 ”
- Securing MySQL using SMF - the Ultimate Manifest (http://blogs.oracle.com/bobn/entry/securing_mysql_using_smf_the).
- Selected man pages include `smf`(5), `smf_security`(5), `svcadm`(1M), `svcbundle`(1M), and `svccfg`(1M).

Configuring a Kerberos Network

You can protect your network with the Kerberos service. This client-server architecture provides secure transactions over networks. The service offers strong user authentication, as well as integrity and privacy. Using the Kerberos service, you can log in to other systems, execute commands, exchange data, and transfer files securely. Additionally, the service enables administrators to restrict access to services and systems. As a Kerberos user, you can regulate other people's access to your account.

For information and procedures see the following:

- Chapter 3, “Planning for the Kerberos Service,” in “Managing Kerberos and Other Authentication Services in Oracle Solaris 11.2 ”
- Chapter 4, “Configuring the Kerberos Service,” in “Managing Kerberos and Other Authentication Services in Oracle Solaris 11.2 ”
- Selected man pages include `kadmin`(1M), `pam_krb5`(5), and `kclient`(1M).

Adding Labeled Multilevel Security

Trusted Extensions extends Oracle Solaris security by enforcing a label-based mandatory access control (MAC) policy. Sensitivity labels are automatically applied to all sources of data (networks, file systems, and windows) and consumers of data (user and processes). Access to all data is restricted based on the relationship between the label of the data (object) and the consumer (subject). The layered functionality consists of a set of label-aware services.

A partial list of Trusted Extensions services includes:

- Labeled networking
- Label-aware file system mounting and sharing
- Labeled desktop
- Label configuration and translation
- Label-aware system management tools
- Label-aware device allocation

The `system/trusted` and `system/trusted/trusted-global-zone` packages are sufficient for a headless system or a server that does not require a multilevel desktop. The `system/trusted/trusted-extensions` package provides the Oracle Solaris multilevel, trusted desktop environment.

Configuring Trusted Extensions

You must install the Trusted Extensions packages, then configure the system. When you install the `trusted-extensions` package, the system can run a desktop with a directly connected bitmapped display, such as a laptop or workstation. Network configuration is required to communicate with other systems.

For information and procedures see the following:

- Part I, “Initial Configuration of Trusted Extensions,” in “Trusted Extensions Configuration and Administration ”
- Part II, “Administration of Trusted Extensions,” in “Trusted Extensions Configuration and Administration ”

Configuring Labeled IPsec

You can protect your labeled packets with IPsec.

For information and procedures see the following:

- Chapter 6, “About IP Security Architecture,” in “Securing the Network in Oracle Solaris 11.2 ”

- “Administration of Labeled IPsec” in “Trusted Extensions Configuration and Administration ”
- “Configuring Labeled IPsec” in “Trusted Extensions Configuration and Administration ”

♦ ♦ ♦ CHAPTER 3

Maintaining and Monitoring Oracle Solaris Security

After initial installation and configuration, you can maintain and monitor the security posture of your system by following the procedures for:

- Regularly reviewing audit records
- Running package and file integrity checks
- Monitoring network activity
- Running compliance checks

Maintaining and Monitoring System Security

The following tasks maintain and monitor access and use of the system, data, and adherence to your site's security requirements.

TABLE 3-1 Maintaining and Monitoring the System Task Map

Task	Description	For Instructions
Verify the packages on the system.	Checks that the packages after an update are identical to the source packages.	"How to Verify Your Packages" on page 31
Verify file integrity.	After configuration, compares BART manifests at regular intervals to ensure that only files that should be changed are changed.	"Verifying File Integrity by Using BART" on page 54
Find rogue files.	Locates the potentially unauthorized use of the `setuid` and `setgid` permissions on programs.	"How to Find Files With Special File Permissions" in "Securing Files and Verifying File Integrity in Oracle Solaris 11.2 "
Review audit logs regularly.	Locates unusual access and use of the system.	"Using the Audit Service" on page 54
Review audit logs for login and logout events in real time.	Identifies attempted breaches near to the time that the attempts occur.	"Monitoring Audit Records in Real Time" on page 55
Run compliance tests.	Assesses the system's compliance to security benchmarks.	"Oracle Solaris 11.2 Security Compliance Guide " and the `compliance`(1M) man page

Verifying File Integrity by Using BART

BART is a rule-based file integrity scanning and reporting tool that uses cryptographic-strength hashes and file system metadata to report changes.

For information and procedures, see the following:

- "About BART" in "Securing Files and Verifying File Integrity in Oracle Solaris 11.2 "
- "About Using BART" in "Securing Files and Verifying File Integrity in Oracle Solaris 11.2 "
- "BART Manifests, Rules Files, and Reports" in "Securing Files and Verifying File Integrity in Oracle Solaris 11.2 "

For specific instructions on tracking changes to installed systems, see "How to Compare Manifests for the Same System Over Time" in "Securing Files and Verifying File Integrity in Oracle Solaris 11.2 ".

Using the Audit Service

Auditing keeps a record of how the system is being used. The audit service includes tools to assist with the analysis of the auditing data.

The audit service is described in "Managing Auditing in Oracle Solaris 11.2 ". For a list of the man pages and links to them, see "Audit Service Man Pages" in "Managing Auditing in Oracle Solaris 11.2 ".

The following audit service procedures are useful in many secure environments:

- Create separate roles to configure auditing, review auditing, and start and stop the audit service. Assign the roles to trusted users.

  Use the Audit Configuration, Audit Review, and Audit Control rights profiles as the basis for your roles.

  To create roles or use the predefined ARMOR roles, see "Assigning Rights to Users" in "Securing Users and Processes in Oracle Solaris 11.2 ".
- Audit all administrators with the `cusa` audit class.

  Events in the `cusa` audit class cover administrative actions that affect the system's security posture. For a description, see the `/etc/security/audit_class` file. For the procedure, see "How to Audit Significant Events in Addition to Login/Logout" on page 42.
- Send audit records to a central server.
  - Configure auditing to work with the Audit Remote Server (ARS).

    See "How to Send Audit Files to a Remote Repository" in "Managing Auditing in Oracle Solaris 11.2 ".

    - Schedule the secure transfer of complete audit files to an audit review file system on a separate ZFS pool.
- Monitor text summaries of selected audited events in the `syslog` utility

  Activate the `audit_syslog` plugin, then monitor the reported events.

  See “How to Configure syslog Audit Logs” in “Managing Auditing in Oracle Solaris 11.2 ”.
- Limit the size of audit files.

  Set the `p_fsize` attribute for the `audit_binfile` plugin to a useful size. Consider your reviewing schedule, disk space, and `cron` job frequency, among other factors.

  For examples, see “How to Assign Audit Space for the Audit Trail” in “Managing Auditing in Oracle Solaris 11.2 ”.
- Schedule the secure transfer of complete audit files to an audit review file system on a separate ZFS pool.
- Review complete audit files on the audit review file system.

Monitoring Audit Records in Real Time

The `audit_syslog` plugin enables you to record summaries of preselected audit events. To display the audit summaries in a terminal window as they are generated, run a command similar to the following:

```
# tail -0f  /var/adm/auditlog
```

To configure the audit log, see “How to Configure syslog Audit Logs” in “Managing Auditing in Oracle Solaris 11.2 ”.

Reviewing and Archiving Audit Logs

Audit records can be viewed in text format or in a browser in XML format.

For information and procedures see the following:

- “Audit Logs” in “Managing Auditing in Oracle Solaris 11.2 ”
- “Preventing Audit Trail Overflow” in “Managing Auditing in Oracle Solaris 11.2 ”
- “Displaying Audit Trail Data” in “Managing Auditing in Oracle Solaris 11.2 ”

♦ ♦ ♦ APPENDIX A

Bibliography for Oracle Solaris Security

The following references contain useful security information for Oracle Solaris systems. Security information from earlier releases of Oracle Solaris contain some useful and some outdated information.

Security References on the Oracle Technology Network

The following books and articles on the Oracle Technology Network web site contain descriptions of security on Oracle Solaris 11 systems:

- "Securing Systems and Attached Devices in Oracle Solaris 11.2 "
- "Securing Files and Verifying File Integrity in Oracle Solaris 11.2 "
- "Securing the Network in Oracle Solaris 11.2 "
- "Securing Users and Processes in Oracle Solaris 11.2 "
- "Managing Encryption and Certificates in Oracle Solaris 11.2 "
- "Managing Auditing in Oracle Solaris 11.2 "
- "Managing Kerberos and Other Authentication Services in Oracle Solaris 11.2 "
- "Managing Secure Shell Access in Oracle Solaris 11.2 "
- "Oracle Solaris 11.2 Security Compliance Guide "
- "Using a FIPS 140 Enabled System in Oracle Solaris 11.2 "

Oracle Solaris Security References in Third-Party Publications

The following books contain descriptions of security on Oracle Solaris 11 systems:

- *Security Configuration Benchmark For Solaris 11 11/11 Version 1.0.0 June 11th, 2012*

  This security benchmark is published by the Center for Internet Security (CIS) http://cisecurity.org/ for the security community. This document recommends security settings for the Oracle Solaris operating system. The targeted audience includes

system and application administrators, security specialists, auditors, support engineers, and installers and developers who develop, install, assess, or provide security solutions for Oracle Solaris. To obtain a copy, visit CIS Security Benchmarks (http://benchmarks.cisecurity.org/).

- *Oracle Solaris 11 System Administration: The Complete Reference*. Michael Jang, Harry Foxwell, Christine Tran, and Alan Formy-Duval. 2012. McGraw-Hill. ISBN 978007179042.

  This trade book includes security coverage of Oracle Solaris.
- *Oracle Solaris 11: First Look*. Philip P. Brown. 2013. Packt Publishing. ISBN 9781849688307.

  This trade book introduces administrators to Oracle Solaris and its security.
- *Oracle Solaris 11 System Administration*, Bill Calkins. 2013. Prentice Hall. ISBN 9780133007114.

  This trade book covers the new features of Oracle Solaris, including security features.