ORACLE SOLARIS 11 SPARC
SECURITY TECHNICAL IMPLEMENTATION GUIDE (STIG) OVERVIEW

Version 3, Release 2

30 January 2025

Developed by Oracle and DISA for the DOD

Trademark Information

Names, products, and services referenced within this document may be the trade names, trademarks, or service marks of their respective owners. References to commercial vendors and their products or services are provided strictly as a convenience to our users, and do not constitute or imply endorsement by the Defense Information Systems Agency (DISA) of any nonfederal entity, event, product, service, or enterprise.

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

1. INTRODUCTION

1.1 Executive Summary

The Oracle Solaris 11 STIG provides the technical security policies, requirements, and implementation details for applying security concepts to Solaris 11 systems.

1.2 Authority

Department of Defense Instruction (DODI) 8500.01 requires that "all IT [information technology] that receives, processes, stores, displays, or transmits DOD information will be […] configured […] consistent with applicable DOD cybersecurity policies, standards, and architectures." The instruction tasks that DISA "develops and maintains control correlation identifiers (CCIs), security requirements guides (SRGs), security technical implementation guides (STIGs), and mobile code risk categories and usage guides that implement and are consistent with DOD cybersecurity policies, standards, architectures, security controls, and validation procedures, with the support of the NSA/CSS [National Security Agency/Central Security Service], using input from stakeholders, and using automation whenever possible." This document is provided under the authority of DODI 8500.01.

Although the use of the principles and guidelines in these SRGs/STIGs provides an environment that contributes to the security requirements of DOD systems, applicable NIST SP 800-53 cybersecurity controls must be applied to all systems and architectures based on the Committee on National Security Systems (CNSS) Instruction (CNSSI) 1253.

1.3 Vulnerability Severity Category Code Definitions

Severity Category Codes (referred to as CAT) are a measure of vulnerabilities used to assess a facility or system security posture. Each security policy specified in this document is assigned a Severity Category Code of CAT I, II, or III.

Table 1-1: Vulnerability Severity Category Code Definitions

Category	DISA Category Code Guidelines
CAT I	Any vulnerability, the exploitation of which will **directly and immediately** result in loss of Confidentiality, Availability, or Integrity.
CAT II	Any vulnerability, the exploitation of which **has a potential** to result in loss of Confidentiality, Availability, or Integrity.
CAT III	Any vulnerability, the existence of which **degrades measures** to protect against loss of Confidentiality, Availability, or Integrity.

1.4 STIG Distribution

Parties within the DOD and federal government's computing environments can obtain the applicable STIG from the DOD Cyber Exchange website at https://cyber.mil/. This site contains

the latest copies of STIGs, SRGs, and other related security information. Those without a Common Access Card (CAC) that has DOD Certificates can obtain the STIG from https://public.cyber.mil/.

1.5   SRG Compliance Reporting

All technical NIST SP 800-53 requirements were considered while developing this STIG. Requirements that are applicable and configurable will be included in the final STIG. A report marked Controlled Unclassified Information (CUI) will be available for items that did not meet requirements. This report will be available to component authorizing official (AO) personnel for risk assessment purposes by request via email to: disa.stig_spt@mail.mil.

1.6   Document Revisions

Comments or proposed revisions to this document should be sent via email to the following address: disa.stig_spt@mail.mil. DISA will coordinate all change requests with the relevant DOD organizations before inclusion in this document. Approved changes will be made in accordance with the DISA maintenance release schedule.

1.7   Other Considerations

DISA accepts no liability for the consequences of applying specific configuration settings made on the basis of the SRGs/STIGs. It must be noted that the configuration settings specified should be evaluated in a local, representative test environment before implementation in a production environment, especially within large user populations. The extensive variety of environments makes it impossible to test these configuration settings for all potential software configurations.

For some production environments, failure to test before implementation may lead to a loss of required functionality. Evaluating the risks and benefits to a system's particular circumstances and requirements is the system owner's responsibility. The evaluated risks resulting from not applying specified configuration settings must be approved by the responsible AO. Furthermore, DISA implies no warranty that the application of all specified configurations will make a system 100 percent secure.

Security guidance is provided for the DOD. While other agencies and organizations are free to use it, care must be given to ensure that all applicable security guidance is applied at both the device hardening level and the architectural level due to the fact that some settings may not be configurable in environments outside the DOD architecture.

1.8   Product Approval Disclaimer

The existence of a STIG does not equate to DOD approval for the procurement or use of a product.

STIGs provide configurable operational security guidance for products being used by the DOD. STIGs, along with vendor confidential documentation, also provide a basis for assessing compliance with cybersecurity controls/control enhancements, which supports system assessment and

authorization (A&A) under the DOD Risk Management Framework (RMF). Department of Defense AOs may request available vendor confidential documentation for a product that has a STIG for product evaluation and RMF purposes from disa.stig_spt@mail.mil. This documentation is not published for general access to protect the vendor's proprietary information.

AOs have the purview to determine product use/approval in accordance with (IAW) DOD policy and through RMF risk acceptance. Inputs into acquisition or pre-acquisition product selection include such processes as:

- National Information Assurance Partnership (NIAP) evaluation for National Security Systems (NSS) (https://www.niap-ccevs.org/) IAW CNSSP #11.
- National Institute of Standards and Technology (NIST) Cryptographic Module Validation Program (CMVP) (https://csrc.nist.gov/groups/STM/cmvp/) IAW Federal/DOD mandated standards.
- DODIN Approved Products List (APL) (https://aplits.disa.mil/processAPList.action) IAW DODI 8100.04.

2. CONCEPTS AND TERMINOLOGY CONVENTIONS

2.1 Introduction

Oracle Solaris is a robust, mature 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 third-party vendors.

Oracle Solaris 11 provides a solid foundation for company data and applications by protecting data on disk and in transit. Oracle Solaris resource management and Oracle Solaris Zones provide features that separate and protect applications from misuse. This containment, together with least privilege implemented through privileges and the role-based access control (RBAC) feature of Oracle Solaris, reduce the security risk of intruder or regular user actions. Authenticated and encrypted protocols such as IP security (IPsec) provide virtual private networks (VPNs) across the Internet, as well as tunnels within a LAN or a WAN for safe data delivery. Additionally, the auditing feature of Oracle Solaris ensures that records are kept of any activity of interest. Oracle Solaris 11 security services provide defense-in-depth by offering layers of protection for the system and the network. Oracle Solaris protects the kernel by limiting, within kernel utilities, what privileged actions the utility can perform. The default network configuration provides data protection on the system and across the wire. IPsec, the IP Filter feature of Oracle Solaris, and Kerberos can provide additional protections.

Oracle Solaris security services include:

- Protecting the kernel – Kernel daemons and devices are protected by file permissions and by privileges.
- Protecting memory – Address space layout is randomized for userland processes.
- Protecting logins – Logons require passwords. Passwords are strongly encrypted. Remote logons are initially limited to an encrypted and authenticated channel through the Secure Shell feature of Oracle Solaris. The root account cannot log on directly.
- Protecting data – Data on disk is protected by file permissions. Additional layers of protection can be configured. For example, the user can use access control lists (ACLs), place data in a zone, encrypt a file, encrypt an Oracle Solaris ZFS dataset, create a read-only ZFS dataset, and mount file systems so that setuid programs cannot be run and executable files cannot be executed.

2.2 Audience

The intended audience for this document includes trained Oracle Solaris 11 administrators and security auditing staff whose responsibilities include:

- Installing and configuring Solaris.
- Securing Solaris.

- Maintaining and reporting on Solaris security configurations.
- Auditing compliance with security guidelines.

2.3   Security Evaluations

Oracle Solaris 11.1 has completed evaluation by the Canadian Common Criteria Scheme at Evaluation Assurance Level 4 (EAL4) and augmented by flaw remediation (EAL4+). EAL4+ is the highest level of evaluation that can be achieved for commercial software. EAL4 is also the highest level of evaluation mutually recognized by 26 countries under the Common Criteria Recognition Arrangement (CCRA).

The evaluation was being conducted against the Operating System Protection Profile (OS PP) and includes the following four optional extended packages:

- Advanced Management (AM).
- Extended Identification and Authentication (EIA).
- Labeled Security (LS).
- Virtualization (VIRT).

Oracle Solaris 11.1 also completed evaluation against the NIST FIPS 140-2 encryption standard.

2.4   Oracle Solaris Versions

Oracle Solaris 11 was introduced in November of 2011. It incorporates existing Solaris 10 features such as zones, ZFS, and Trusted Extensions with new technologies including Image Packaging System, Alternate Boot Environments, and network virtualization.

The Oracle Solaris 11 Security Technical Implementation Guide was written by Oracle with support from DISA Field Security Office and requires the 11.1 version released in 2012.

2.5   Oracle Solaris 11 References

This document provides a brief overview of Solaris security capabilities as they relate to the DISA Security Guidelines. For standard Oracle Solaris 11 information, the user should refer to the Solaris 11.1 documentation set at: http://docs.oracle.com/cd/E26502_01/

For additional detail on Oracle Solaris 11.1 security, the user should refer to the Solaris Security guide at: http://docs.oracle.com/cd/E26502_01/html/E29014/index.html

2.6   Oracle Solaris 11 STIG Organization

The Oracle Solaris 11 STIG is organized in a number of sections focusing on various aspects of the Solaris 11 system:

1. Auditing is designed to ensure that actions taken by users and applications are recorded for later review.

2. System Packaging provides controls for which Solaris packages are installed and that they are validated and maintained in the proper configuration.
3. System Services determine what network and operations services are enabled on the system.
4. User Accounts are designed to allow authorized users access to the system with the proper credentials and capabilities.
5. Network Access Controls affect the appearance of the system to external networks.
6. Encryption controls ensure that proper standards are implemented and data is protected on internal and external storage.
7. Permissions and ownership protect file system objects from unauthorized access.
8. System Level Configuration describes controls that affect the Solaris kernel and hardware platforms.
9. Operation Actions are those procedures that must be performed by the administrative staff.
10. Zones describe specific security settings relating to the creation and configuration of Solaris zones virtualization technology.

2.7   Solaris Security Concepts

An overview is provided for some basic security concepts in Oracle Solaris 11 including:

- Naming Services.
- Roles and Profiles.
- Auditing.
- System Packages.
- System Services.
- File Systems.
- Zones.
- Trusted Extensions.

2.8   Naming Services

Solaris 11 supports several naming services managed using the Service Management Facility. Naming services are used to store user accounts, roles, authorizations, groups, projects, and other system information. In the DOD environment, only the files and LDAP naming services are supported. NIS is not allowed in the DOD environment and NIS+ is no longer supported in Solaris 11. All the instructions for account management in the STIG assume that the "files" naming service is being used. If LDAP is being used, account information is stored in a specific LDAP schema and managed separately. Building and managing and LDAP repository is beyond the scope of the STIG.

Example: Determine the naming service in use

```
# svccfg -s system/name-service/switch listprop config/default
config/default astring files
```

2.9 Roles and Profiles

In conventional UNIX systems, the root user, also referred to as the superuser, is all-powerful. Programs that run as root, or setuid programs, are all-powerful. The root user has the ability to read and write to any file, run all programs, and send kill signals to any process. Effectively, anyone who can become a superuser can modify a site's firewall, alter the audit trail, read confidential records, and shut down the entire network. A setuid program that is hijacked can do anything on the system.

Role-based access control (RBAC) provides a more secure alternative to the all-or-nothing superuser model. With RBAC, the organization can enforce security policy at a more fine-grained level. RBAC uses the security principle of **least privilege**. Least privilege means that a user has precisely the amount of privilege that is necessary to perform a job. Regular users have enough privilege to use their applications, check the status of their jobs, print files, create new files, and so on. Capabilities beyond regular user capabilities are grouped into rights profiles. Users who are expected to do jobs that require some of the capabilities of a superuser assume a role that includes the appropriate rights profile.

RBAC collects superuser capabilities into **rights profiles**. These rights profiles are assigned to special user accounts that are called **roles**. A user can then assume a role to do a job that requires some of a superuser's capabilities. Predefined rights profiles are supplied with Oracle Solaris software. The organization creates the roles and assigns the profiles.

Rights profiles can provide broad capabilities. For example, the System Administrator rights profile enables an account to perform tasks that are not related to security, such as printer management and cron jobs. Rights profiles can also be narrowly defined. For example, the Cron Management rights profile manages at and cron jobs. When you create roles, the roles can be assigned broad capabilities or narrow capabilities or both.

You cannot, in a default Solaris 11 installation, log on directly as root. The initial installation requires the creation of a username and allows that user to assume the root role using the su command. A user with the root role can grant the ability to assume the root role to other users.

2.10 Profile Usage in the STIG

The Oracle Solaris 11 STIG documents which profiles are the minimum required to perform certain actions. While the root role will enable the administrator to perform all the required actions, assumption of the root role provides the user with excessive privileges and may result in undesired results. It is recommended to only provide the user with the least privilege required to accomplish the changes required.

The pfexec command allows the user to execute a command with whatever profile capabilities he or she has without requiring an additional password or assumption of the root role. Alternatively, the user can be configured with a "profile shell" such as pfbash (rather than standard bash) in /etc/passwd. If the user's shell is pfbash, the pfexec command is not required. If the user assumes the root role, pfexec is not required.

Oracle's guidance is to use the pfedit command when editing all system security relevant files. The pfedit command allows authorized users to edit system configuration files. The invoking user must

have the authorization solaris.admin.edit/*path_to_file* or the blanket authorization solaris.admin.edit.

The pfedit command creates a copy of files owned by the invoking user. It then invokes an editor on that file using the ID and privileges of the invoking user. In the case of a successful update, an attempt to make unauthorized use, or if an error occurs, an audit record is generated to capture the subject, the file name, the authorization used, the file change if any, and the success or failure of the operation.

2.11 Profile Examples

Change the user's shell to pfbash.

```
# pfexec usermod -s /usr/bin/pfbash [username]
```

Determine the user's current profile.

```
# profiles [username]
```

List all available profiles along with their capabilities.

```
# profiles -la
```

Add or remove a profile to a user.

```
# pfexec usermod -P +"Name Service Security" [username]
# pfexec usermod -P -"Name Service Security" [username]
```

Enable a user with the ability to assume the root role.

```
# pfexec usermod -R root [username]
```

Edit a file using the pfedit command to ensure audit records are produced documenting the file changes.

```
# pfedit [filename]
```

Determine the authorizations provided by a specific profile.

```
# profiles -p "profile name" info
```

Determine the complete set of authorizations for a user.

```
# auths [username]
```

List the roles available on the system.

```
# getent user_attr |grep "type=role"
```

2.12 Auditing

Auditing is the collecting of data about the use of system resources. The audit data provides a record of security-related system events. This data can then be used to assign responsibility for actions that take place on a host. Successful auditing starts with two security features: identification and

authentication. At each logon, after a user supplies a username and authentication succeeds, a unique and immutable **audit user ID** is generated and associated with the user, and a unique audit session ID is generated and associated with the user's process. The audit session ID is inherited by every process that is started during that logon session. When a user switches to another user, all user actions are tracked with the same audit user ID. Note that by default, certain actions, such as booting and shutting down the system, are always audited.

The audit service allows the operator to:

- Monitoring security-relevant events that take place on the host.
- Recording the events in a network-wide audit trail.
- Detecting misuse or unauthorized activity.
- Reviewing patterns of access and the access histories of individuals and objects.
- Discovering attempts to bypass the protection mechanisms.
- Discovering extended use of privilege that occurs when a user changes identity.

The "Audit configuration", "Audit Review", and "Audit Control" profiles are required to completely control the auditing system. Audit configuration allows the user to add and remove auditing actions using the auditconfig command while the Audit Control profile allows the user to start and stop the audit system using the audit command. The Audit Review profile allows usage of the auditreduce and praudit reporting commands.

All audit flags must be enabled in a single auditconfig command.

The STIG instructions describe how to configure the correct audit flags, policies, and log file settings.

2.13   Reviewing Audit Logs

STIG guidance ensures that auditing is enabled and the proper audit flags, options, and log files are configured. Audit data is stored in a binary file. The auditreduce command can be used to select specific data from the binary file while the praudit command converts the binary data to a human readable format. The STIG instructions describe how to configure the audit log locations, permissions, encryption, warning, and quotas. Proper management of the audit logs size and content is important to system security and availability. The Audit Review profile is required to execute the auditreduce and praudit commands.

2.14   Auditing Examples

Enable the audit system

```
# pfexec audit –s
```

Terminate the audit system

```
# pfexec audit –t
```

Restart the audit system and open a new audit file

```
# pfexec audit -n
```

Determine whether auditing is enabled

```
#pfexec auditconfig -getcond
```

Review the entire audit trail with one long entry per line

```
# pfexec auditreduce | praudit -l
```

Review only "lo" (login/logout) audit records with one long entry per line

```
# pfexec auditreduce -c lo | praudit -l
```

Review only "lo" (login/logout) audit records with output in XML format

```
# pfexec auditreduce -c lo | praudit -x
```

Enable the recommended audit flags. All flags must be enabled in a single command. Both attributable and nonattributable actions should be enabled.

```
# pfexec auditconfig -setflags cusa,ps,fd,fa,ft,fm
# pfexec auditconfig -setnaflags cusa,ps,fd,fa,ft,fm
```

2.15   System Packaging

Oracle Solaris 11 is distributed using IPS packages. IPS packages are stored in IPS package repositories, which are populated by IPS publishers. IPS packages are installed into Oracle Solaris 11 images. A subset of the capabilities that are available through the IPS command-line interface is available through the Package Manager graphical user interface. IPS packages are managed using the pkg command.

The Software Installation profile is required to use the pkg command to install, remove, and manage packages. The system security posture can be improved by removing unneeded packages, however, IPS tracks dependencies and will not allow you to remove a package that other packages are dependent upon.

System updates (formerly known as patches) are installed using the pkg update command. If required by the package, an alternate boot environment will automatically be created using a ZFS snapshot and the updates will safely be installed into the alternate boot environment. A reboot will be required to activate this type of update.

The default package repository (known as a publisher) for updates is stored online at pkg.oracle.com. ISO images of the repository can be installed on your local network or in the local file systems for updates that do not require internet access.

2.16   System Package Examples

Determine packages installed on the system

```
# pkg list
```

Uninstall or install a package

```
# pfexec pkg uninstall [package name]
# pfexec pkg install [package name]
```

Update a package or the entire system

```
# pfexec pkg update [package name]
# pfexec pkg update
```

Verify and fix a damaged or modified package (leave off the package name to verify or fix all packages on the system)

```
# pkg verify [pkg name]
# pfexec pkg fix [pkg name]
```

Determine the source of all repository publishers

```
# pkg publisher
```

Search for locally installed packages

```
# pkg search -l [pattern]
```

Determine the currently available boot environments

```
# beadm list
```

Activate a boot environment for next reboot event

```
# pfexec beadm activate [BE name]
```

2.17   System Services

Oracle Solaris 11 system services are managed using the Service Management Facility (SMF). SMF makes it easier to manage applications and system services. The framework infrastructure augments the traditional UNIX startup scripts, init run levels, and configuration files. SMF provides a mechanism to define the relationships between applications or services, so that dependent services can automatically be restarted when necessary. Information needed to manage each service is stored in the service configuration repository, which provides a simplified way to manage each service.

SMF defines a set of actions that can be invoked on a service by an administrator. These actions, which can be manually manipulated by the svcadm command, include enable, disable, refresh, restart, and mark. Each service is managed by a service restarter, which carries out the administrative actions. In general, the restarters carry out actions by executing methods for a service. Methods for each service are defined in the service configuration repository. These methods allow the restarter to move the service from one state to another state.

The service configuration repository provides a per-service snapshot at the time that each service is successfully started so that fallback is possible. In addition, the repository provides a consistent and

persistent way to enable or disable a service, as well as a consistent view of service states. This capability helps you debug service configuration problems.

Each service instance is named with a Fault Management Resource Identifier or FMRI. The FMRI includes the service name and the instance name. For example, the FMRI for the rlogin service is svc:/network/login:rlogin, where network/login identifies the service and rlogin identifies the service instance. Equivalent formats for an FMRI are as follows:

- `svc://localhost/system/system-log:default`
- `svc:/system/system-log:default`
- `system/system-log:default`

The Oracle Solaris 11 STIG defines which services must be disabled and also requires that unnecessary services should be disabled.

2.18   System Services Examples

List all the running services.

```
# svcs -a
```

List services on which a service is dependent.

```
# svcs -d [service name]
```

List services dependent on a particular service.

```
# svcs -D [service name]
```

Enable or disable a service.

```
# pfexec svcadm enable [service name]
# pfexec svcadm disable [service name]
```

Direct a service to refresh its configuration settings.

```
# pfexec svcadm refresh [service name]
```

Restart a service.

```
# pfexec svcadm restart [service name]
```

2.19   Network Access Controls

Network access controls affect the interaction of the Solaris 11 system with the outside world over network interface cards. Solaris 11 supports both IPv4 and IPv6 controls. There are multiple levels of control available including:

- Network Interface and Virtual Network interfaces (vNIC) managed using the dladm command.
- TCP/IP level controls managed using the ipadm command.

- Firewall protections controlled by the ipfilters firewall tool.
- TCP Wrappers managed using SMF commands included svcadm and svccfg.
- Network encryption facilities via IPsec and IKE standards.

2.20 Network Interfaces

Solaris 11 supports both physical NICs and virtual NICs. These can have system-assigned names or user-assigned "vanity names." The interface names are important because they are referenced by other tools, such as the ipfilters firewall, IPsec, IP multi-pathing, and ipadm commands. The dladm command is used to create, manipulate, and configure network interfaces, including quality-of-service controls and anti-spoofing parameters. Virtual network interfaces (vNIC) are automatically created when a nonglobal zone is created.

2.21 TCP/IP Level Control

The ipadm command is introduced to replace the ifconfig command for interface configuration. The command also replaces the ndd command to configure protocol properties. The ipadm is used to create IP address, as well as configure TCP/IP communications options defined in the STIG to enhance network security.

The route and routeadm commands are used to control TCP/IP routing.

2.22 Firewall Protection

The IP Filter feature of Oracle Solaris is a firewall that provides stateful packet filtering and network address translation (NAT). IP Filter also includes stateless packet filtering and the ability to create and manage address pools.

Packet filtering provides basic protection against network-based attacks. IP Filter can filter by IP address, port, protocol, network interface, and traffic direction. IP Filter can also filter by an individual source IP address, a destination IP address, a range of IP addresses, or address pools.

- IP Filter is managed by the SMF service svc:/network/ipfilter.
- IP Filter requires direct editing of configuration files.
- IP Filter is installed as part of Oracle Solaris.
- To administer IP Filter, you must assume the root role or able to assume a role that includes the IP Filter Management rights profile.

2.23 TCP Wrappers

TCP wrappers add a measure of security for service daemons, such as ftpd, by standing between the daemon and incoming service requests. TCP wrappers log successful and unsuccessful connection attempts. Additionally, TCP wrappers can provide access control, allowing or denying the connection depending on where the request originates. You can use TCP wrappers to protect

daemons such as SSH, Telnet, and FTP. Specific hosts can be allowed or denied access using the /etc/hosts.allow and /etc/hosts.deny files.

2.24 Network Encryption

IPsec and IKE protect network transmissions between nodes and networks that are jointly configured with IPsec and IKE. Use the ipsecinit.conf file and the ipsecconf command to configure IPsec policies. Configuring a properly secured and authenticated IPsec connection is a detailed process and beyond the process of the current STIG guidance.

DOD information system environments require the use of DOD public key infrastructure (PKI) for authentication, identity, signature, and encryption processes. Oracle Solaris 11 natively does not provide this capability. Commercial off-the-shelf solutions (COTS), such as ActivIdentity clients, are available to leverage for meeting this requirement. Other PKI solutions embedded into the web, database, or application server software, which reside outside of the control of the base operating system, can also be configured to authenticate via the DOD PK-Enabled infrastructure.

2.25 Network Command Examples

List physical network devices.

```
# dladm show-phys
```

List IP addresses associated with interfaces.

```
# ipadm show-addr
```

List current TCP/IP/UDP properties.

```
# ipadm show-prop
```

List routing configuration.

```
# routeadm
```

Enable TCP Wrappers.

```
# inetadm -M tcp_wrappers=TRUE
```

List the status of services managed by inetd.

```
# inetadm
```

2.26 File Systems

In Solaris 11, the ZFS file system is used for all system software and configuration file storage. It is closely integrated with the Image Package System and Alternate Boot Environment capabilities to provide safe, flexible system updates. Although UFS is still supported for customer data, ZFS is recommended because of its advanced data protection, snapshot, compression, de-duplication and encryption features. ZFS uses the concept of **storage pools** to manage physical storage.

The zpool and zfs commands are used to create and manage data pools and filesystems. The /etc/vfstab is no longer used to configure ZFS file system mounts. ZFS file systems are mounted upon creation.

ZFS file systems support standard UNIX-style file permissions for user, group, other providing read, write, execute controls, as well as access control lists.

Two rights profiles are provided to enable management of ZFS pools and file systems.

- ZFS Storage Management – Provides the privilege to create, destroy, and manipulate devices within a ZFS storage pool.
- ZFS File system Management – Provides the privilege to create, destroy, and modify ZFS file systems.

2.27 Sample ZFS Commands

Create a mirrored ZFS pool of two disks.

```
# pfexec zpool create [poolname] mirror c1t0d0 c2t0d0
```

Create and mount a ZFS file system.

```
# pfexec zfs create [poolname]/[filesystem name]
```

Determine the status of existing zpools or file systems.

```
# zpool status
# zfs list
```

Get properties for a ZFS filesystem.

```
# zfs get all [poolname]/[filesystem name]
```

Enable compression on a ZFS filesystem.

```
# pfexec zfs set compression=on [poolname]/[filesystem name]
```

2.28 Zones

The Oracle Solaris Zones software partitioning technology enables the organization 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, you can configure a read-only zone root.

A nonglobal zone can be thought of as a box. One or more applications can run in this box without interacting with the rest of the system. Zones isolate software applications or services by using flexible, software-defined boundaries. Applications that are running in the same instance of the Oracle Solaris operating system can then be managed independently of one other. Thus, different versions of the same application can be run in different zones, to match the requirements of the organization's configuration.

A process assigned to a zone can manipulate, monitor, and directly communicate with other processes that are assigned to the same zone. The process cannot perform these functions with processes that are assigned to other zones in the system or with processes that are not assigned to a zone. Processes that are assigned to different zones are only able to communicate through network APIs.

IP networking can be configured in two different ways, depending on whether the zone has its own exclusive IP instance or shares the IP layer configuration and state with the global zone. Exclusive-IP is the default type. For more information about IP types in zones, see Zone Network Interfaces. For configuration information, see How to Configure the Zone.

Every Oracle Solaris system contains a **global zone**. The global zone has a dual function. The global zone is both the default zone for the system and the zone used for system-wide administrative control. All processes run in the global zone if no **nonglobal** zones, referred to simply as zones, are created by the **global administrator** or a user with the Zone Security profile.

The global zone is the only zone from which a nonglobal zone can be configured, installed, managed, or uninstalled. Only the global zone is bootable from the system hardware. Administration of the system infrastructure, such as physical devices, routing in a shared-IP zone, or dynamic reconfiguration (DR), is only possible in the global zone. Appropriately privileged processes running in the global zone can access objects associated with other zones.

Because nonglobal zones have limited privileges, certain STIG items are only applicable to the global zone. Zones are configured with the zonecfg command and managed with the zoneadm command.

2.29   Immutable Zones

A zone with a read-only zone root is called an Immutable Zone. An Oracle Solaris 11 Immutable Zone preserves the zone's configuration by implementing read-only root file systems for nonglobal zones. This zone extends the zones secure runtime boundary by adding additional restrictions to the runtime environment. Unless performed as specific maintenance operations, modifications to system binaries or system configurations are blocked.

By default, the zonecfg file-mac-profile property is not set in a nonglobal zone. A zone is configured to have a writable root dataset. In a Solaris read-only zone, the file-mac-profile property is used to configure a read-only zone root. A read-only root restricts access to the runtime environment from inside the zone. Through the zonecfg utility, the file-mac-profile can be set to one of the following values. All of the profiles except none will cause the /var/pkg directory and its contents to be read-only from inside the zone.

- none:

- - Standard, read-write, nonglobal zone, with no additional protection beyond the existing zones boundaries. Setting the value to none is equivalent to not setting file-mac-profile property.
- strict:
  - Read-only file system, no exceptions.
  - IPS packages cannot be installed.
  - Persistently enabled SMF services are fixed.
  - SMF manifests cannot be added from the default locations.
  - Logging and auditing configuration files are fixed. Data can only be logged remotely.
- fixed-configuration:
  - Permits updates to /var/* directories, with the exception of directories that contain system configuration components.
  - IPS packages, including new packages, cannot be installed.
  - Persistently enabled SMF services are fixed.
  - SMF manifests cannot be added from the default locations.
  - Logging and auditing configuration files can be local. syslog and audit configuration are fixed.
- flexible-configuration:
  - Permits modification of files in /etc/* directories, changes to root's home directory, and updates to /var/* directories.
  - IPS packages, including new packages, cannot be installed.
  - Persistently enabled SMF services are fixed.
  - SMF manifests cannot be added from the default locations.
  - Logging and auditing configuration files can be local. syslog and audit configuration can be changed.

Current STIG guidance does not require the use of one of the stricter policies, but the customer may choose to use them if it meets the requirements of their operations and application.

2.30 Trusted Extensions

Configuration of Oracle Solaris 11 Trusted Extensions requires complex organizational security policy decisions and is beyond the scope of this STIG. While Oracle Solaris 11 Trusted Extensions are not addressed in the STIG, an overview is provided here for customers who require a higher level of data and network separation.

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. Unlike most other multilevel operating systems, Trusted Extensions includes a multilevel desktop.

Trusted Extensions meets the requirements of the Common Criteria Labeled Security Protection Profile (LSPP), the Role-Based Access Protection Profile (RBAPP) and the Controlled Access Protection Profile (CAPP). However, the Trusted Extensions implementation is unique in its ability to provide high assurance, while maximizing compatibility and minimizing overhead.