



SOLARIS - A FEATURE OPERATING SYSTEM

NARAYANAN S.¹, ESWAR S.² AND KALIDASBABU M.³

¹Dept. of. ECE, VSVN Polytechnic College, Virudhunagar, TN, India.

²Mphasis an HP company, Chennai, TN, India.

³Dept. of. Computer Engg., VSVN Polytechnic College, Virudhunagar, TN, India.

*Corresponding Author: Email-

Received: January 12, 2012; Accepted: February 15, 2012

Abstract- Solaris is system software belongs to UNIX family. Solaris is a high security operating system. This paper shows the overview of the Solaris operating system. It also elaborates the features of the Solaris briefly. It also deals with the latest File system called ZFS in UNIX. It also explores the security system in Solaris than any other flavor of UNIX.

Keywords- ZFS, Solaris, UNIX, Security, Virtualization, File system.

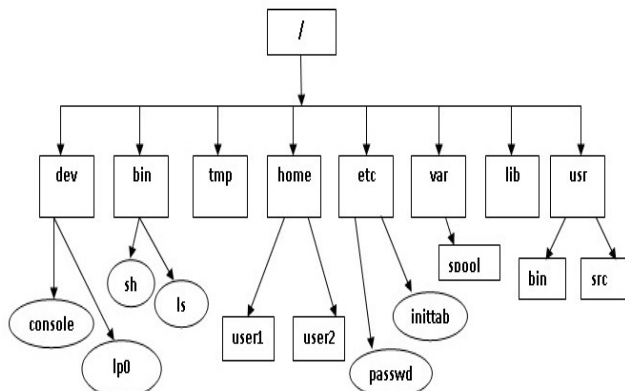
Citation: Narayanan S., Eswar S. and Kalidasbabu M. (2012) Solaris - A Feature Operating System. Journal of Information Systems and Communication ISSN: 0976-8742 & E-ISSN: 0976-8750, Volume 3, Issue 1, pp.- 247-250.

Copyright: Copyright©2012 Narayanan S., et al. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Solaris- An Overview

Solaris is an operating system introduced by Sun Microsystems Ltd. Solaris belongs to Unix Operating system family which uses shell as interpreter. The shell translates the user commands to a language that is understood by the kernel of the Operating system. The Kernel controls all the hardware such as CPU, Storage devices and I/Os.

File system in Solaris [7].



Benefits

Virtualization: Optimize resource utilization to deliver predictable service levels with Solaris Containers.

Networking: Attain near-wire-speed throughput with the open, programmable Solaris networking stack [1].

Security: Implement a secure foundation for deploying services with Solaris leading-edge security features.

Availability: Increase uptime with the predictive self healing feature.

Features of Solaris

Security Solaris includes some of the world's most advanced security features, such as Process and User Rights

Management: Trusted Extensions for Mandatory Access Control, the Cryptographic Framework and Secure by Default Networking that allow you to safely deliver new Solutions consolidate with security and protect mission-critical data [1].

Performance: Solaris delivers indisputable performance advantages for database, Web, and Java technology-based services, as well as massive scalability, shattering world records by delivering unbeatable price/performance advantages.

Networking: With its optimized network stack and support for today's advanced network computing protocols, Solaris delivers high-performance networking to most applications without modification.

Data Management: Solaris offers dramatic advances in file system and volume management by delivering virtually unlimited

capacity and near-zero administration [2].

Interoperability: Understanding that businesses today rely on a mix of technologies from a variety of vendors, Solaris provides tools to enable seamless interoperability with hundreds of heterogeneous hardware and software platforms [7].

Observability: The Solaris release gives you Observability into your system with tools such as Solaris Dynamic Tracing (DTrace), which enables real-time application debugging and optimization.

Platform Choice: Solaris is fully supported on more than 900 SPARC-based and x64/x86-based systems from top manufacturers, including systems from Sun, Dell, HP, and IBM [2].

Virtualization: The Solaris OS includes industry-first virtualization features such as Solaris Containers, which let you consolidate, isolate, and protect thousands of applications on a single server [7].

Availability: Solaris features, such as Predictive Self Healing, support automatic diagnosis and recovery from hardware and application faults, maximizing system uptime.

Support & Services: Offering a broad portfolio of world-class services, Sun can help you extract maximum value from the Solaris Operating System.

ZFS Administration

ZFS is a combined file system and logical volume manager designed by Sun Microsystems.

The features of ZFS include data integrity verification against data corruption modes (like bit rot), support for high storage capacities, integration of the concepts of file system and volume management, snapshots and copy-on-write clones, continuous integrity checking and automatic repair, RAID-Z and native NFSv4 ACLs [4]. ZFS is implemented as open-source software, licensed under the Common Development and Distribution License (CDDL). The ZFS name is a trademark of Oracle.

Comparison of Solaris to other operating systems [6]:

	Solaris	Linux	Windows
Creator	Sun Microsystems Inc	Red Hat Enterprises	Microsoft Corporation
First Release	1992	1992	1993
Kernel Type	Monolithic with modules	Monolithic with modules	Hybrid
Supported Architecture	x86, x86-64, SPARC	x86, x86-64, PPC, SPARC, Alpha,	x86, x86-64, IA-64
File system supported	UFS, ZFS, ext2, FAT, ISO 9660, UDF, NFS, QFS	ext2, ext3, ext4, ReiserFS, FAT, ISO 9660, UDF, NFS	NTFS, FAT, ISO 9660, UDF
GUI Default On	Yes	Depends on flavor	Yes
Encrypted File System	Yes	Yes	Yes
Cost availability	Commercial; a free 90 days evaluation version exists. For open source its free	Free	\$469 Web Server; other editions dependent on number of CALs purchased

Features of ZFS:

- Recursive ZFS Snapshots
- Double Parity RAID-Z (raidz2)
- Hot Spares for ZFS Storage Pool Devices
- Replacing a ZFS File System with a ZFS Clone (ZFS promote)

- Upgrading ZFS Storage Pools (zpool upgrade)
- Using ZFS to Clone Non-Global Zones and Other Enhancements
- Recovering Destroyed Storage Pools
- ZFS is integrated with Fault Manager
- New zpool clear Command
- Compact NFSv4ACLFormat
- File System Monitoring Tool (fsstat)
- ZFS Web-Based Management

Working of ZFS

ZFS uses the concept of storage pools to manage physical storage. Historically, file systems were constructed on top of a single physical device. To address multiple devices and provide for data redundancy, the concept of a Volume manager was introduced to provide the image of a Single device so that file systems would not have to be modified to take advantage of multiple devices [5]. This design added another layer of complexity and ultimately prevented certain file system advances, because the file system had no control over the physical placement of data on the virtualized volumes.

ZFS eliminates the volume management altogether. Instead of forcing you to create virtualized volumes, ZFS aggregates devices into a storage pool. The storage pool describes the physical characteristics of the storage (device layout, data redundancy, and so on,) and acts as an arbitrary data store from which file systems can be created. File systems are no longer constrained to individual devices, allowing them to share space with all file systems in the pool. You no longer need to predetermine the size of a file system, as file systems grow automatically within the space allocated to the storage pool [5].

When new storage is added, all file systems within the pool can immediately use the additional space without additional work. In many ways, the storage pool acts as a virtual memory system. When a memory DIMM is added to a system, the operating system doesn't force you to invoke some commands to configure the memory and assign it to individual processes. All processes on the system automatically use the additional memory.

Advantages of ZFS File system Transactional Semantics

ZFS is a transactional file system, which means that the file system state is always consistent on disk. Traditional file systems overwrite data in place, which means that if the machine loses power.

For example, between the times a data block is allocated and when it is linked into a directory, the file system will be left in an inconsistent state. Historically, this problem was solved through the use of the fsck command.

This command was responsible for going through and verifying file system state, making an attempt to repair any inconsistencies in the process. This problem caused great pain to administrators and was never guaranteed to fix all possible problems.

The journaling process records action in a separate journal, which can then be replayed safely if a system crash occurs. This process introduces unnecessary overhead, because the data needs

to be written twice, and often results in a new set of problems, such as when the journal can't be replayed properly[7].

With a transactional file system, data is managed using copy on write semantics. Data is never overwritten, and any sequence of operations is either entirely committed or entirely ignored. This mechanism means that the file system can never be corrupted through accidental loss of power or a system crash. So, no need for an fsck equivalent exists [5].

While the most recently written pieces of data might be lost, the file system itself will always be consistent.

In addition, synchronous data (written using the O_DSYNC flag) is always guaranteed to be written before returning, so it is never lost.

Checksums and Self-Healing Data

With ZFS, all data and metadata is check summed using a user-selectable algorithm. Traditional file systems that do provide check summing have performed it on a per-block basis, out of necessity due to the volume management layer and traditional file system design [4]. The traditional design means that certain failure modes, such as writing a complete block to an incorrect location, can result in properly check summed data that is actually incorrect. ZFS checksums are stored in a way such that these failure modes are detected and can be recovered from gracefully. All check summing and data recovery is done at the file system layer, and is transparent to applications [4].

In addition, ZFS provides for self-healing data. ZFS supports storage pools with varying levels of data redundancy, including mirroring and a variation on RAID-5. When a bad data block is detected, ZFS fetches the correct data from another replicated copy, and repairs the bad data, replacing it with the good copy.

Unparalleled Scalability

ZFS has been designed from the ground up to be the most scalable file system ever. The file system itself is 128-bit, allowing for 256 quadrillion zeta bytes of storage. All metadata is allocated dynamically, so no need exists to pre-allocate inodes or otherwise limit the scalability of the file system when it is first created.

All the algorithms have been written with scalability in mind. Directories can have up to 248 (256 trillion) entries and no limit exists on the number of file systems or number of files that can be contained within a file system.

ZFS Snapshots

A snapshot is a read-only copy of a file system or volume. Snapshots can be created quickly and easily.

Initially, snapshots consume no additional space within the pool. As data within the active dataset changes, the snapshot consumes space by continuing to reference the old data. As a result, the snapshot prevents the data from being freed back to the pool.

Simplified Administration

ZFS provides a greatly simplified administration model. Through the use of hierarchical file system layout, property inheritance, and auto management of mount points and NFS share semantics, ZFS makes it easy to create and manage file systems without needing multiple commands or editing configuration files.

You can easily set quotas or reservations, turn compression on or off, or manage mount points for numerous file systems with a

single command. Devices can be examined or repaired without having to understand a separate set of volume manager commands. You can take an unlimited number of instantaneous snapshots of file systems. You can backup and restore individual file systems.

ZFS manages file systems through a hierarchy that allows for this simplified management of properties such as quotas, reservations, compression, and mount points [4].

In this model, file systems become the central point of control. File systems themselves are very cheap (equivalent to a new directory), so you are encouraged to create a file system for each user, project, workspace, and so on. This design allows you to define fine-grained management points.

Solaris security

To be effective, data security features must be designed into the heart of an operating system (OS), rather than simply wrapped around its core components.

Solaris combines multiple security technologies allowing it to monitor file integrity, verify user and process rights, and retain a detailed audit trail of all system events.

Solaris's networking configuration design and cryptographic capabilities also significantly reduce the system's risk exposure [3].

Benefits

- Verifies system integrity via file verification features.
- Reduces risk, granting only the privileges required.
- Defends against system attacks via secure by default networking profile, IP filter firewall, and TCP wrappers.
- Simplifies administration by using open, standards-based cryptographic and key management frameworks in Solaris for encryption.
- Controls access to data based on its sensitivity through Solaris with trusted extensions labeled security technology.
- Evaluated against stringent independent testing profiles.

File Integrity

System administrators can detect possible attacks on their systems by monitoring for changes to file information. Solaris utilizes binaries that are digitally signed, so that administrators can easily track changes. All patches or enhancements are embedded with digital signatures, eliminating the false positives associated with

Upgrading or patching file integrity-checking software. Solaris also provides the basic audit reporting tool for integrity checking of customer files.

User and Process Rights Management

Hackers often attempt to exploit root accounts because those accounts are empowered with complete access to UNIX systems. Solaris offers unique user and process rights management technology that reduces risks by granting users and applications only the minimum capabilities needed to perform their duties. Unlike other solutions, Solaris requires no application changes to take advantage of these security enhancements. Solaris applications running on 64-bit SPARC, AMD, and Intel processors are also automatically protected from a form of intrusion known as "stack smashing" by a non executable stack feature, with no application changes required and no degradation of system performance [3].

Network Service Protection

Solaris provides protection against inappropriate use of network resources through its secure by default networking configuration, which disables many unused network services to reduce exposure to attack. With secure by default networking, an administrator can enable or disable individual network services or change how they listen for network connections.

Solaris also ships with IP filter firewall software preinstalled. This integrated firewall can reduce the number of network services that are exposed to attack. It also provides protection against maliciously crafted networking packets.

Cryptographic Services and Encrypted Communication

For high-performance, system wide cryptographic routines, the cryptographic framework in Solaris adds a standards-based common API that provides a single point of administration for cryptographic routines and digital certificate lifecycle management.

Solaris' key management framework provides a single set of administrative commands for digital certificate creation requests, manipulation, and loading. These pluggable frameworks balance loads across hardware accelerators and software implementations, increasing encrypted network traffic throughput. They're available to applications written to use the PKCS #11, OpenSSL, and Java Cryptography Extension APIs [3].

The IPSec/IKE and secure shell protocols in Solaris guard against theft of sensitive material by encrypting communications. IPSec/IKE complies with industry standards to provide data encryption between two or more systems over the network, without any application modification.

The secure shell protocol is a specific set of utilities modified to allow for encrypted remote access and file transfer between two systems.

Flexible Enterprise Authentication

Solaris delivers a number of flexible authentication features, including support for the Pluggable Authentication Module (PAM), which makes it possible to add authentication services to the OS dynamically. Third-party vendors provide many PAM modules, and customers can create their own to meet specific security needs.

The Kerberos service in Solaris delivers Kerberos-enabled remote applications such as rsh, rcp, telnet, secure shell, and NFS file sharing. Kerberos-based protocols allow for standards-based enterprise single sign-on (SSO), authorization, and encrypted communication. Lightweight Directory Access Protocol (LDAP) client-side authentication and interoperability enhancements enable enterprise wide, secure, standards-based access to your servers and applications. To enable easier integration with existing environments, Solaris provides NIS and NIS+ to LDAP gateways.

All Solaris user and process rights management information can also be stored and managed centrally using LDAP-based directory server software.

System-specific user IDs now have strong password encryption options, including MD5 and Blowfish, as well as account lockout, password history and complexity checking, long password format, and a banned passwords list.

Repeatable Security Hardening and Monitoring

New features in Solaris make it easier than ever to minimize and harden a system.

The reduced networking Meta cluster install option creates a minimized Solaris image, ready for administrators to add functionality and services in direct support of their system's purpose. In addition, the Secure by Default networking configuration disables many unused network services, while configuring all other services for local system-only communications [3].

Solaris Service Manager can be used to control exactly which services run, who can manage those services, and what privileges those services run with.

The freely available security toolkit for Solaris assists in the process of installing and maintaining a minimized and hardened OS security configuration. The toolkit integrates with the Solaris Jumpstart installation process for repeatable secure

Mandatory Access Control, Labeling, and Security Certification

Solaris with trusted extensions solves the problem of controlling access to sensitive data by implementing sensitivity labels for access control to files, printers, networks, windows, applications, and devices.

Solaris with trusted extensions is the only labeled OS feature to support full enterprise-class solutions, giving customers multilevel desktops through the GNOME-based Java Desktop System or CDE, simple deployment, and centralized user ID management.

Solaris 11/06 is currently in evaluation at EAL4+, one of the highest levels of Common Criteria Certification, with three Protection Profiles:

- Labeled Security Protection Profile (LSPP),
- Controlled Access Protection Profile (CAPP), and
- Role-Based Access Control Protection Profile (RBACPP).

Conclusion

Solaris offers many new technologies to enhance the overall security of the IT infrastructure. Innovation in rights management for processes and users, N1 Grid Containers, cryptographic infrastructure with already strong features in authentication and access control has the potential to make Solaris the most secure operating platform in the market. Solaris OS security, system administrators have new possibilities open to them to solve problems that previously would have taken many more systems, complex add-on products, changes in networking topology or other such compromises. Explore additional Sun documentation and articles for more ideas on how to use the Solaris OS to creatively solve your business and security issues.

References

- [1] <http://docs.sun.com/doc/819-5461>.
- [2] Solaris 10, *The Complete Reference by Watters* 8-22.
- [3] Solaris 10, *The Complete Reference by Watters* 169-190.
- [4] <http://docs.sun.com/doc/817-1985>.
- [5] Solaris 10, *The Complete Reference by Watters* 340-356.
- [6] en.wikipedia.org/wiki/Comparison_of_operating_systems.
- [7] http://download.oracle.com/docs/cd/E23823_01/.