Minimizing the Solaris™ Operating Environment for Security

Updated for Solaris 9 Operating Environment

By Alex Noordergraaf, Enterprise Server Products Group, Security Architect

Sun BluePrints™ OnLine—November 2002

Please
Recycle

Adobe PostScript™

Minimizing the Solaris™ Operating Environment for Security

Updated for Solaris 9 Operating Environment

This article provides tips, instructions, and preferred practices for minimizing the Solaris Operating Environment (Solaris OE) to increase system security. It focuses on operating system (OS) installation practices for minimizing and automating Solaris OE installations. This article provides a simple, reproducible, and secure application installation methodology.

The minimal OS requirements of a server vary depending on the applications, OS release, and hardware. This article describes the methodology for OS minimization and gives specific recommendations for packages needed to support the Sun™ ONE Web Server. You can use the methodology and recommendations in this article to assist you in determining the minimum OS modules that must be installed on your server.

This article contains the following topics:

Updates

In applying the methodology and updates in this article, we used the Netra™ t1 running Solaris 9 OE 64-bit as the test platform.

This article includes revisions and content changes for the following:

- Updated for Solaris 9 OE (5/02), 64-bit mode
- Updated scripts for Solaris™ Security Toolkit software version 0.3.9
- Added support for Sun ONE Web Server 6.0 (formerly iPlanet™ Web Server)
- Removed Solaris 2.5.1 OE

The scripts released with this article are updated to be compatible with the Solaris Security Toolkit software version 0.3.9. Please refer to "References and Related Resources" on page 33, for additional information about the Solaris Security Toolkit software.

All Solaris OE package listings support the Sun ONE Web Server version 6.0sp2 product release.

No changes were made to the methodology presented in the previous version of this article.

Note – All references to Solaris 2.5.1 OE were removed. Solaris 2.5.1 OE is no longer supported on either the hardware or software described in this update.

Introduction

Securing computer systems against unauthorized access is one of the most pressing issues facing today's data center administrators. Recent studies suggest that the number of unauthorized accesses continue to rise, as do the monetary losses associated with security breaches.

One way to reduce system vulnerabilities is to minimize the amount of software on a server. Fewer software components on a server means fewer security holes to detect and protect. The majority of system intrusions are accomplished through exploiting security holes in the OS. Minimizing the OS modules installed on a server can greatly improve overall system security by reducing the vulnerabilities. This process is done within the framework of a jumpstart installation, which makes it possible to completely automate the installation process. Automation is particularly important in a data center environment, where machines typically number in the hundreds.

Assumptions and Limitations

The processes in this methodology are orientated toward a typical lights-out data center environment. We make the following assumptions about server configurations:

■ JumpStart™ software is available for system installations.

■ JumpStart software is configured properly for hands-off system installation and configuration. (Refer to "References and Related Resources" on page 33 for additional information.)

■ The Solaris Security Toolkit software is used as the framework to deploy the security scripts. (Refer to "References and Related Resources" on page 33 for additional information.)

■ Terminal consoles (character-based) are used for console access.

■ No video cards are used on any of the systems.

■ No X Window server software is required on the server.

The software builds are performed on Sun4U™ systems that are based on UltraSPARC™ II and III technology (Netra™, Ultra Enterprise™, and Sun Fire™ servers) and use PCI interface cards. Additional packages are required to support other hardware platforms not documented in this article.

Minimized systems may or may not support specific applications. Care should be taken when developing minimized Solaris OE configurations to ensure that required applications run properly.

_____

Choosing an Installation Cluster

The Solaris OE installation process requires the selection of one of four installation clusters:

■ Core – Installs the smallest Solaris OE image. Only packages that may be required for any SPARC™ or Solaris OE Intel Platform Edition system are installed.

■ End User – Builds on the Core cluster by installing the window managers included with the Solaris OE (OpenWindows™ and CDE).

■ Developer – Includes additional libraries, header files, and software packages that may be needed on systems used as compile and development servers.

■ Entire Distribution – Also referred to as the OEM cluster, includes all Solaris OE software on the installation CDs.

Each installation cluster represents a group of packages. The cluster groups simplify the installation of the OS for the mass market. Because each of these installation clusters contains support for a variety of hardware platforms (for example, microSPARC™, UltraSPARC™, UltraSPARC II, and more) and software requirements (for example, NIS, NIS+, DNS, OpenWindows, Common Desktop Environment [CDE], development, computer aided design [CAD], and more), far more packages are installed than are normally used and required on a single Solaris OE.

The size of the clusters varies significantly. The following table shows the increasing packages and size of each cluster:

Cluster	Solaris 2.6 OE Packages/Size	Solaris 7 OE Packages/Size	Solaris 8 OE Packages/Size	Solaris 9 OE Packages/Size
Core	33/38 MB	39/52 MB	62/61 MB	94/116 MB
End User	131/182 MB	142/242 MB	313/471 MB	346/525 MB
Entire Distribution	204/369 MB	235/493 MB	390/679 MB	468/850 MB
OEM	261/461 MB	387/692 MB	459/711 MB	544/985 MB

Note – The package and size information was obtained through installations performed on a Netra t1 and Netra t 1125. Results for other hardware platforms may vary.

Experience shows that, in many cases, a secure web server requires less than 20 Solaris OE packages and uses as little as 36 MBytes of disk space.

Installing unnecessary services, packages, and applications can severely compromise system security. One example of this is the OpenWindows Calendar Manager Server Daemon (`rpc.cmsd`), which is unnecessary on many data center systems. This daemon is installed and started by default when the End User, Developer, or Entire Distribution cluster is chosen during the installation process.

There have been known bugs filed against and fixed within the `rpc.cmsd` subsystem of OpenWindows/CDE, and at least two CERT/CC advisories (CA-99-08, CA-96-09) issued. Scanners for `rpc.cmsd` are included in the most common scanning tools available on the Internet.

The best protection against unknown `rpc.cmsd` vulnerabilities is to not install the daemon at all, thus preventing it from being accidentally enabled. This problem is well known in the computer industry; there are hundreds of similar examples. Not surprisingly, almost every security reference book addresses the need to perform "minimal OS installations" [Garfinkel]. Unfortunately, this is easier said than done. Other than the occasional firewall, no software applications are shipped with lists of their package requirements, and there is no easy way of determining this information other than through trial and error.

Because it is so difficult to determine the minimal set of necessary packages, system administrators commonly just install the Entire Distribution cluster. While this may be the easiest to do from the short-term perspective of getting a system up and running, it makes it considerably more difficult to secure the system.

The remainder of this article presents a methodology for determining the minimal set of packages required to successfully install and run a particular application—the Sun ONE Web Server in this case.

Methodology Summary

The following tasks summarize the methodology.

1. Verify that the Solaris OE versions on the JumpStart boot server are correct.

2. Install the Core Solaris OE cluster, plus any other required packages.

3. Install all patches.

4. Remove all unnecessary packages.

5. Use JumpStart software with the Solaris Security Toolkit software to configure the OS for the data center environment.

6. Install and configure the software applications you want to run on the system.

7. Check the logs for errors; if necessary, fix the errors and repeat the installation process.

8. Test the software installation.

Detailed procedures and information for performing each of these processes is in the following section, "Minimizing and Automating Installations" on page 6.

Minimizing and Automating Installations

The primary goal of this article is to create a simple, reproducible, and secure application installation methodology. A secondary goal is to automate the OS and software installation process.

The following sections provide procedures for each of the tasks:

- "Verifying OS Software" on page 7
- "Installing Solaris OE Core Cluster" on page 8
- "Installing Patches" on page 9
- "Removing Unnecessary Packages" on page 9
- "Using JumpStart Software to Configure the OS" on page 10

- "Installing and Configuring Software Packages" on page 10
- "Checking For Errors" on page 10
- "Testing Software Installation" on page 11

Verifying OS Software

Verify that the Solaris OE versions installed on the JumpStart boot server are the versions that you require. For the purposes of the testing performed for this article, we use the following Solaris OE versions:

- Solaris 2.6 OE (5/98)
- Solaris 7 OE (11/99)
- Solaris 8 OE (2/02)
- Solaris 9 OE (5/02)

Installing and Configuring a JumpStart Server

The installation and configuration of a JumpStart server is beyond the scope of this article. For information, please refer to the following:

- Sun BluePrints™ book *JumpStart Technology: Effective Use in the Solaris Operating Environment* (`http://www.sun.com/blueprints/pubs.html`)
- Sun BluePrints OnLine article *Building a JumpStart Infrastructure*, (`http://www.sun.com/blueprints/0401/BuildInf.pdf`)
- FAQ available from SunSolve OnLine℠ (`http://sunsolve.sun.com`)
- *Solaris Advanced Installation Guide* (`http://docs.sun.com`)

Kernel Patching the Boot Image

Depending on the Solaris OE version used on the JumpStart server and the hardware platform being installed, some kernel patching of the boot image may be required. If you encounter unexpected results during the installation, refer to SunSolve OnLine (`http://sunsolve.sun.com`) to determine if any patches are required.

For information on how to kernel patch the boot image, refer to the Sun BluePrints *JumpStart™ Technology: Effective Use In the Solaris™ Operating Environment.*

Using Solaris Security Toolkit Scripts

The JumpStart framework and automation capabilities of the Solaris Security Toolkit software simplify Step 2 through Step 6 of the methodology as described in "Methodology Summary" on page 6. You can download the scripts used to validate and test this methodology from `http://www.sun.com/blueprints/tools/`. The following list describes the scripts specific to the Sun ONE Web Server installation supported by Solaris OE versions 2.6 through 9.

- `install-iPlanetWS.driver` – Provides a framework, based on the Solaris Security Toolkit software, in which all other scripts are run.

- `install-iPlanetWS.fin` – Extracts and installs the Sun ONE Web Server software onto the server.

- `minimize-iPlanetWS.fin` – Removes unnecessary Solaris OE packages according to the Solaris OE being installed.

- `minimal-iPlanetWS-Solaris9-64bit.profile` – Defines which cluster and packages should be installed for a 64-bit Solaris 9 OE installation.

- `minimal-iPlanetWS-Solaris8-32bit.profile` – Defines which cluster and packages should be installed for a 32-bit Solaris 8 OE installation.

- `minimal-iPlanetWS-Solaris8-64bit.profile` – Defines which cluster and packages should be installed for a 64-bit Solaris 8 OE installation.

- `minimal-iPlanetWS-Solaris7-32bit.profile` – Defines which cluster and packages should be installed for a 32-bit Solaris 7 OE installation.

- `minimal-iPlanetWS-Solaris7-64bit.profile` – Defines which cluster and packages should be installed for a 64-bit Solaris 7 OE installation.

- `minimal-iPlanetWS-Solaris26.profile` – Defines which cluster and packages should be installed for a Solaris 2.6 OE installation.

Installing Solaris OE Core Cluster

The initial installation should include only the Solaris OE Core cluster and a few other packages that contain critical functionality. In JumpStart server terminology, the Core cluster is referred to as the `SUNWCreq` cluster. For your initial Core cluster, be advised that each OS version requires additional packages. Refer to "Minimizing the Sun ONE Web Server" on page 11 for version details.

The profile, which is used by JumpStart to define which OS cluster and packages are installed, must specify both the Solaris OE install cluster and any additional packages required. Sample profiles are available in the Solaris Security Toolkit software.

Installing Patches

Before making any other changes to the system, it is critical to install on your server all recommended, security, and software vendor patches. This step is especially important when the goal is to minimize the number of installed packages, because some patches may install unwanted packages.

Sun recommends installing the Recommended and Security Patch Cluster for the Solaris OE version being installed. It contains all recommended and security patches. Access to these patch clusters does not require a service contract.

You can find the Recommended and Security Patch Cluster, the Patches Containing Security Fixes, and the Kernel Update Patches on the SunSolve OnLine Web site at: `http://sunsolve.sun.com`.

The kernel update patch 106541 for Solaris 7 OE is an example of why patches must be installed before you perform any minimization or security hardening. The `README` and `pkgmap` of this patch shows that the following files are updated when the patch is installed:

- `/etc/rc2.d/S71rpc`
- `/etc/syslog.conf`
- `/etc/init.d/rpc`

The presence of any of these files may either enable a service that has previously been disabled (`rpc`, automounter, or volume manager) or overwrite a file with specific configuration information in it (`syslog.conf`).

Caution – Once package removal and system configuration has begun, patch installation should be done *only* after the `README` and `pkgmap` of a package is reviewed for possible conflicts.

Removing Unnecessary Packages

After you install and patch the Solaris OE, remove unnecessary packages. The package removal process deletes all packages not explicitly required by either the OS or the software package being installed.

In the test environment, we use headless Sun4U systems based on SPARC technology and using PCI- based I/O cards. We remove more than half of the 62 packages included in the Solaris 9 OE Core cluster. The number of packages to remove depends upon the system being used. This package removal is automated with the `minimize-iPlanetWS.fin` script. This script is both application and OS-specific, because each software package and OS has slightly different requirements.

For package listings of all Solaris OE versions, refer to "Minimizing the Sun ONE Web Server" on page 11. Different hardware architectures, environments, and software packages may require other packages for your installations.

Additional configuration and hardening of the OS is not covered in this article. Refer to "References and Related Resources" on page 33 for Sun BluePrints OnLine articles that cover these topics.

Using JumpStart Software to Configure the OS

Due to the repetitive nature of installations in this methodology, the basic network configuration steps for a server are automated. These include both required network and operating systems configurations. We use the Solaris Security Toolkit software to automate the creation of files such as `/etc/defaultrouter` and to configure services, thereby simplifying system configuration.

Installing and Configuring Software Packages

The final step in the methodology is to install and configure as much of the software package as possible. The level of automation implemented depends on how the software package is installed and the time available to automate the process. In the case of the Sun ONE Web Server software, which uses a `curses`-based installation process, the only automated task is to extract the source packages into an appropriate directory. Once extracted, the installation routines must be run manually to configure the server. The Solaris Security Toolkit `install-iPlanetWS.fin` script copies and extracts the software package into the `/opt` directory of the client.

Checking For Errors

Before installing software applications, it is important that you examine the installation logs on the server for any errors or configuration problems.

The JumpStart logs are located in the `/var/sadm/system/logs` directory. The `begin.log` contains all pre-OS installation operations and the `finish.log` contains all post-OS installation steps. Usually the `finish.log` contains the most pertinent messages. If you find errors, correct them and repeat the installation. Repeat this process until you resolve all errors.

Testing Software Installation

The software installation is tested manually by running the setup routine and selecting a default configuration for both the administrative and production web server ports. After a default configuration is defined, the `startconsole` command is used to start up the administration server. For additional information on how to automate the configuration process, refer to the product documentation.

In our test environment, this command, while successfully starting the Sun ONE Web Server software, also attempted to launch a local Netscape™ Communicator session. It failed because Netscape Communicator was not installed locally on the system. Rather than managing the installation locally, we used a remote Netscape Communicator session to configure the web server through the administration port.

Minimizing the Sun ONE Web Server

Using the process, procedures, and scripts in this article, we determine the minimal OS installation for Sun ONE Web Server software. We install the minimum Solaris OE Core cluster in the test environment. Beyond this, each Solaris OE installation is different. Refer to the following sections for details about the packages required for:

- Solaris 9 OE, 64-bit
- Solaris 8 OE, both 32-bit and 64-bit
- Solaris 7 OE, both 32-bit and 64-bit
- Solaris 2.6 OE

The packages we remove are specific to each version of the operating system.

Solaris 9 OE

This section presents the minimum packages required to successfully install and run a 64-bit Solaris 9 OE environment. In addition, the packages specifically required for Sun ONE Web Server are listed. The package listing is presented with explanations of why the packages are recommended or required.

Note – This section was written for Solaris 9 OE (5/02), which was the first version released. Subsequent updates of Solaris 9 OE may introduce additional packages that may require modification of the minimization scripts. Use the information provided as a template and customize it to the specific OS and patch version you are minimizing.

Solaris 9 OE—64 Bit

The following packages must be available to run Solaris 9 OE in 64-bit mode based on a Sun4U system:

Package Type	Description
SUNWcar	Core Architecture, (Root)
SUNWcarx	Core Architecture, (Root) (64-bit)
SUNWcsd	Core Solaris Devices
SUNWcsl	Core Solaris, (Shared Libs)
SUNWcslx	Core Solaris Libraries (64-bit)
SUNWcsr	Core Solaris, (Root)
SUNWcsu	Core Solaris, (Usr)
SUNWcsxu	Core Solaris (Usr) (64-bit)
SUNWhmd	SunSwift Adapter Drivers
SUNWhmdx	SunSwift Adapter Drivers (64-bit)
SUNWkvm	Core Architecture, (Kvm)
SUNWkvmx	Core Architecture (Kvm) (64-bit)
SUNWloc	System Localization
SUNWlocx	System Localization (64-bit)
SUNWnamos	Northern America OS Support

Altogether, a minimum of 18 packages are required to boot a Solaris 9 OE system running in a 64-bit mode.

The Netra t1 installation process requires the following additional packages as part of its installation process, because it is a Sun4U system with IDE disk and PCI based I/O cards:

Package Type	Description
SUNWidecr	IDE device drivers
SUNWider	IDE Device Driver, (Root)
SUNWidecx	IDE Device Driver, (Root)
SUNWpd	PCI Drivers
SUNWpdx	PCI Drivers (64-bit)

The following packages are recommended to simplify administration and support, because they contain required utilities such as awk and patchadd:

Package Type	Description
SUNWesu	Extended System Utilities
SUNWswmt	Install and Patch Utilities

To summarize, a Netra t1 requires 22 Solaris packages to run a relatively usable version of Solaris OE.

Solaris 9 OE—Sun ONE Web Server

To successfully install and run the Sun ONE Web Server software on a minimized system, the following additional packages are required:

Package Type	Description
SUNWlibms	Forte Developer Bundled shared libm
SUNWlmsx	Forte Developer Bundled 64-bit shared libm
SUNWlibC	Sun Workshop Compilers Bundled libC
SUNWlibCx	Sun WorkShop Bundled 64-bit libC

Solaris 9 OE—Infrastructure Services

The previous list of Solaris 9 OE packages required for Sun ONE Web Server does not include support for some services and protocols that may be required in a data center environment. We recommend that you add the following packages:

Package Type	Description
SUNWsshcu	SSH Common, (Usr)
SUNWsshdr	SSH Server, (Root)
SUNWsshdu	SSH Server, (Usr)
SUNWsshr	SSH Client and utilities, (Root)
SUNWsshu	SSH Client and utilities, (Usr)
SUNWzlib	The Zip compression library
SUNWzlibx	The Zip compression library (64-bit)

To provide support for SSH X Tunneling, add the following packages:

Package Type	Description
SUNWdtbax	CDE application basic runtime environment (64-bit)
SUNWmfrun	Motif RunTime Kit
SUNWxwplt	X Window System platform software
SUNWxwplx	X Window System library software (64-bit)
SUNWxwrtl	X Window System & Graphics Runtime Library Links in /usr/lib
SUNWxwrtx	X Window System Runtime Compatibility Package (64-bit)
SUNWxwice	X Window System Inter-Client Exchange (ICE) Components
SUNWxwicx	X Window System ICE library (64-bit)

To provide support for `ping`, add the following packages:

Package Type	Description
SUNWbip	Basic IP commands (Usr)

To provide FTP support, modify the `minimize-iPlanetWS.fin` script so that the following packages are not removed:

Package Type	Description
SUNWftpr	FTP Server, (Root)
SUNWftpu	FTP Server, (Usr)

To provide Telnet support, modify the `minimize-iPlanetWS.fin` script so that the following packages are not removed:

Package Type	Description
SUNWtnetc	Telnet Command (client)
SUNWtnetd	Telnet Server Daemon (Usr)
SUNWtnetr	Telnet Server Daemon (Root)

To provide Network Time Protocol (NTP) support, add the following packages:

Package Type	Description
SUNWntpr	NTP, (Root)
SUNWntpu	NTP, (Usr)

To provide Simple Mail Transport Protocol (SMTP) capabilities, modify the `minimize-iPlanetWS.fin` script so that the following packages are not removed:

Package Type	Description
SUNWsndmu	Sendmail user
SUNWsndmr	Sendmail root

To provide support for `truss`, add the following packages:

Package Type	Description
SUNWtoo	Programming Tools
SUNWtoox	Programming Tools (64-bit)

To provide support for `gzip`, add the following package:

Package Type	Description
SUNWgzip	GNU Zip (gzip) compression utility

To support `snoop`, modify the `minimize-iPlanetWS.fin` script so that the following packages are not removed:

Package Type	Description
SUNWrcmdc	Remote Network Client Commands

Note – The two Federated Name Service packages (`SUNWfns` and `SUNWfnsx`) are no longer required to support `snoop` in Solaris 9 OE.

Solaris 8 OE

This section presents the minimum packages required to successfully install and run 32-bit and 64-bit Solaris 8 OE environments. In addition, the packages required for Sun ONE Web Server are listed. The package listing is presented with explanations of why the packages are recommended or required.

Note – This section was written for Solaris 8 OE (3/00), which was the first version released. Subsequent releases of Solaris 8 OE introduced additional packages that require modifications to the minimization scripts. Use the information provided as a template and customize it to the specific OS and patch version you are minimizing.

Solaris 8 OE—32 Bit

Solaris 8 OE running in 32-bit mode requires the following packages:

PackageType	Description
SUNWcar	Core Architecture, (Root)
SUNWcsd	Core Solaris Devices
SUNWcsl	Core Solaris, (Shared Libs)
SUNWcsr	Core Solaris, (Root)
SUNWcsu	Core Solaris, (Usr)
SUNWesu	Extended System Utilities
SUNWhmd	SunSwift SBus Adapter Drivers
SUNWkvm	Core Architecture, (Kvm)
SUNWlibms	Sun WorkShop Bundled shared libm
SUNWloc	System Localization
SUNWnamos	Northern America OS Support
SUNWpd	PCI Drivers
SUNWswmt	Install and Patch Utilities

The Netra t1 installation process requires the following additional packages as part of its installation process, because it is a Sun4U system with IDE disk:

Package Type	Description
SMEvplr	SME platform links
SMEvplu	SME usr/platform links
SUNWensqr	Ensoniq ES1370/1371/1373 Audio Device Driver (32-bit), (Root)
SUNWglmr	Symbios 875/876 SCSI device driver, (Root)
SUNWidecr	IDE device drivers
SUNWider	IDE Device Driver, (Root)

Solaris 8 OE—64 Bit

In addition to the 32-bit packages required for Solaris 8 OE, additional 64-bit packages must be added. The 64-bit packages are required only on systems running a 64-bit kernel. To run Sun ONE Web Server in 64-bit mode, the following packages must be added:

Package Type	Description
SUNWcarx	Core Architecture, (Root) (64-bit)
SUNWcslx	Core Solaris Libraries (64-bit)
SUNWcsxu	Core Solaris (Usr) (64-bit)
SUNWesxu	Extended System Utilities (64-bit)
SUNWhmdx	SunSwift SBus Adapter Drivers (64-bit)
SUNWkvmx	Core Architecture (Kvm) (64-bit)
SUNWlmsx	Sun WorkShop Bundled 64-bit shared libm
SUNWlocx	System Localization (64-bit)
SUNWnamox	Northern America 64-bit OS Support
SUNWpdx	PCI Drivers (64-bit)

Altogether, a minimum of 22 packages are required to boot a Solaris 8 OE system running in a 64-bit mode.

Solaris 8 OE—Sun ONE Web Server

To successfully install and run the Sun ONE Web Server on a minimized system, the following additional packages are required:

Package Type	Description
SUNWlibC	Sun Workshop Compilers Bundled libC
SUNWlibCx	Sun WorkShop Bundled 64-bit libC

As previously mentioned, the 64-bit package is required only on a system running in 64-bit mode.

Solaris 8 OE—Infrastructure Services

The previous listing of Solaris 8 OE packages required for Sun ONE Web Server does not include support for some services and protocols that may be required in a data center environment.

To provide Network Time Protocol (NTP) support, add the following packages:

Package Type	Description
SUNWntpr	NTP, (Root)
SUNWntpu	NTP, (Usr)

To provide Simple Mail Transport Protocol (SMTP), add the following packages:

Package Type	Description
SUNWsndmu	Sendmail user
SUNWsndmr	Sendmail root

To provide support for `truss`, add the following packages:

Package Type	Description
SUNWtoo	Programming Tools
SUNWtoox	Programming Tools (64-bit)

To provide support for `gzip`, add the following package:

Package Type	Description
SUNWgzip	GNU Zip (gzip) compression utility

To provide support for `snoop`, add the following packages:

Package Type	Description
SUNWfns	Federated Naming System
SUNWfnsx	Federated Naming System (64-bit)

To provide support for OpenSSH X Tunneling, add the following packages:

Package Type	Description
SUNWxcu4	XCU4 Utilities
SUNWxcu4x	XCU4 Utilities (64-bit)
SUNWxwplt	X Window System platform software
SUNWxwplx	X Window System library software (64-bit)
SUNWxwrtl	X Window System & Graphics Runtime Library Links in /usr/lib
SUNWxwrtx	X Window System Runtime Compatibility Package (64-bit)
SUNWswmt	Install and Patch Utilities
SUNWxwice	ICE components
SUNWxwicx	X Window System ICE library (64-bit)

Solaris 7 OE

This section presents the minimum packages required to successfully install and run a 32-bit and a 64-bit Solaris 7 OE.

Solaris 7 OE—32 Bit

The following Solaris 7 OE packages are required to run Sun ONE Web Server:

Package Type	Description
SUNWcar	Core Architecture, (Root)
SUNWcsd	Core Solaris Devices
SUNWcsl	Core Solaris, (Shared Libs)
SUNWcsr	Core Solaris, (Root)
SUNWcsu	Core Solaris, (Usr)
SUNWesu	Extended System Utilities
SUNWhmd	SunSwift SBus Adapter Drivers

Package Type	Description
SUNWkvm	Core Architecture, (Kvm)
SUNWlibC	Sun Workshop Compilers Bundled libC
SUNWlibms	Sun WorkShop Bundled shared libm
SUNWloc	System Localization
SUNWpd	PCI Drivers
SUNWploc	Partial Locales
SUNWswmt	Install and Patch Utilities

The total disk space used for these packages is less than 40 MBytes.

For Netra t1, install the following required packages:

Package Type	Description
SMEvplr	SME platform links
SMEvplu	SME usr/platform links
SUNWide	IDE device drivers
SUNWidecr	IDE Device Driver, (Root)
SUNWider	IDE device drivers
SUNWpci	PCI Simba device drivers

Solaris 7 OE—64 Bit

For 64-bit Solaris 7 OE, install the following packages in addition to the packages specified in the 32-bit mode section:

Package Type	Description
SUNWcarx	Core Architecture, (Root) (64-bit)
SUNWcslx	Core Solaris Libraries, (64-bit)
SUNWcsxu	Core Solaris, (Usr) (64-bit)
SUNWesxu	Extended System Utilities (64-bit)
SUNWhmdx	SunSwift SBus Adapter Drivers (64-bit)
SUNWkvmx	Core Architecture, (Kvm) (64-bit)

Package Type	Description
SUNWlibCx	Sun WorkShop Bundled 64-bit libC
SUNWlmsx	Sun WorkShop Bundled 64-bit shared libm
SUNWlocx	System Localization (64-bit)
SUNWpdx	PCI Drivers (64-bit)

Solaris 7 OE—Infrastructure Services

The previous listings of Solaris 7 OE packages do not include support for some services and protocols that may be required in a data center environment. Depending on the package, you may either need to add packages to the profile or modify the minimize-iPlanetWS.fin script to not remove certain packages. The most commonly used services are listed as follows.

To provide Network Time Protocol (NTP) support, add the following packages:

Package Type	Description
SUNWntpr	NTP, (Root)
SUNWntpu	NTP, (Usr)

To provide Simple Mail Transport Protocol (SMTP), add the following packages:

Package Type	Description
SUNWsndmu	Sendmail user
SUNWsndmr	Sendmail root

To provide support for truss, add the following packages:

Package Type	Description
SUNWtoo	Programming Tools
SUNWtoox	Programming Tools (64-bit)

To provide support for `snoop`, add the following packages:

Package Type	Description
SUNWfns	Federated Naming System
SUNWfnsx	Federated Naming System (64-bit)

To provide support for OpenSSH X Tunneling, add the following packages:

Package Type	Description
SUNWxcu4	XCU4 Utilities
SUNWxwplt	X Window System platform software
SUNWxwplx	X Window System library software (64-bit)
SUNWxwrtl	X Window System & Graphics Runtime Library Links in /usr/lib
SUNWxwrtx	X Window System Runtime Compatibility Package (64-bit)
SUNWxwice	ICE components
SUNWxwicx	X Window System ICE library (64-bit)

Solaris 2.6 OE

The Solaris 2.6 OE software package listing for Sun ONE Web Server contains the following packages:

Package Type	Description
SUNWcar	Core Architecture, (Root)
SUNWcsd	Core Solaris Devices
SUNWcsr	Core Solaris, (Root)
SUNWcsu	Core Solaris, (Usr)
SUNWesu	Extended System Utilities
SUNWglmr	Symbios 875/876 SCSI device driver, (Root)
SUNWhmd	SunSwift SBus Adapter Drivers

Package Type	Description
SUNWkvm	Core Architecture, (Kvm)
SUNWlibC	SPARCompilers Bundled libC
SUNWlibms	Sun WorkShop Bundled shared libm
SUNWloc	System Localization
SUNWpd	PCI Drivers
SUNWploc	Partial Locales
SUNWswmt	Install and Patch Utilities

The total disk space used by these packages is approximately 40 Mbytes.

The following packages are added, by the Solaris OE installation program, for the Netra t1 system:

Package Type	Description
SMEvplr	SME platform links
SMEvplu	SME usr/platform links
SUNWide	IDE device drivers
SUNWidecr	IDE Device Driver, (Root)
SUNWider	IDE device drivers
SUNWpci	PCI Simba device drivers

Solaris 2.6 OE—Infrastructure Services

The previous listing of Solaris 2.6 OE packages does not include support for some services and protocols that may be required in a data center environment. The most commonly used services are listed as follows.

To provide Network Time Protocol (NTP) support, add the following packages:

Package Type	Description
SUNWntpr	NTP, (Root)
SUNWntpu	NTP, (Usr)

To provide support for `truss` add, the following package:

Package Type	Description
SUNWtoo	Programming Tools

To provide support for `snoop`, add the following package:

Package Type	Description
SUNWfns	Federated Naming System

To provide support for OpenSSH X Tunneling, add the following packages:

Package Type	Description
SUNWxcu4	XCU4 Utilities
SUNWxwplt	X Window System platform software
SUNWxwrtl	X Window System & Graphics Runtime Library Links in /usr/lib

Presenting a Case Study

This section presents a case study of the minimization process we use on the test environment of Solaris 9 OE 64-bit on a Netra t1. Each step of our process is documented and described to provide information about the challenges encountered and how we addressed them.

In this case study, the Core OS packages are as follows:

Package Type	Description
SUNWcar	Core Architecture, (Root)
SUNWcarx	Core Architecture, (Root) (64-bit)
SUNWcsd	Core Solaris Devices
SUNWcsl	Core Solaris, (Shared Libs)
SUNWcslx	Core Solaris Libraries (64-bit)
SUNWcsr	Core Solaris, (Root)
SUNWcsu	Core Solaris, (Usr)
SUNWcsxu	Core Solaris (Usr) (64-bit)
SUNWkvm	Core Architecture, (Kvm)
SUNWkvmx	Core Architecture (Kvm) (64-bit)

Although these packages comprise the Core cluster packages, they are not sufficient to successfully boot the system from its internal hard drives. The Netra t1 is a PCI-based system using IDE disks. We install the following packages so that the system can boot the OS from internal disks:

Package Type	Description
SUNWidecr	IDE device drivers
SUNWidecx	IDE device drivers- 64bit
SUNWider	IDE Device Driver, (Root)
SUNWpd	PCI Drivers
SUNWpdx	PCI Drivers (64-bit)

At this point, the system boots Solaris 9 OE, but still produces the following errors:

```
ifconfig: plumb: hme0: no such interface
moving addresses from failed IPv4 interfaces: hme0 (couldn't move,
no alternative interface).
```

Based on the error messages displayed during system boot, the network interfaces built into the system are not being configured properly through `plumb`. To resolve this error, we install the following package:

Package Type	Description
SUNWhmd	SunSwift SBus Adapter Drivers
SUNWhmdx	SunSwift Adapter Drivers (64-bit)

The system now boots properly and configures the internal network interfaces. However, errors are produced when the system attempts to use `awk` and `nawk`. The error messages are as follows:

```
/etc/rc2.d/S69inet: /usr/bin/awk: not found
/etc/rc2.d/S69inet: /usr/bin/awk: not found
```

A quick check of a complete Solaris 9 OE installation using either of the two following commands determines which package contains the /usr/bin/awk command.

```
# /usr/sbin/pkgchk -l -p /usr/bin/awk
Pathname: /usr/bin/awk
Type: regular file
Expected mode: 0555
Expected owner: root
Expected group: bin
Expected file size (bytes): 85176
Expected sum(1) of contents: 59286
Expected last modification: Feb 12 09:05:16 PM 2002
Referenced by the following packages:
        SUNWesu
Current status: installed
```

```
# grep /usr/bin/awk /var/sadm/install/contents
/usr/bin/awk f none 0555 root bin 85176 59286 1013565916 SUNWesu
/usr/bin/oawk=../../usr/bin/awk l none SUNWesu
```

We add the following package required for awk and nawk support:

Package Type	Description
SUNWesu	Extended System Utilities

The system now boots properly and configures the internal network interfaces, however, startup errors are produced similar to the following:

```
ld.so.1: awk: fatal: libm.so.1: open failed: No such file or
directory
ld.so.1: awk: fatal: libm.so.1: open failed: No such file or
directory
ld.so.1: awk: fatal: libm.so.1: open failed: No such file or
directory
```

Using the grep command, we obtain the following:

```
# grep libm.so.1 /var/sadm/install/contents
/usr/lib/libm.so=libm.so.1 s none SUNWlibms
/usr/lib/libm.so.1 f none 0755 root bin 110116 53593 1008109356
SUNWlibms
/usr/lib/sparcv9/libm.so=libm.so.1 s none SUNWlmsx
/usr/lib/sparcv9/libm.so.1 f none 0755 root bin 125992 55763
1008109377 SUNWlmsx
```

To resolve these error messages, we add the following packages:

Package Type	Description
SUNWlibms	Forte Developer Bundled shared libm
SUNWlmsx	Forte Developer Bundled 64-bit shared libm

The system now boots properly and configures the internal network interfaces. However, localization errors are produced similar to the following:

```
couldn't set locale correctly
couldn't set locale correctly
```

To resolve these error messages, we add the following packages:

Package Type	Description
SUNWloc	System Localization
SUNWlocx	System Localization (64-bit)
SUNWnamos	Northern America OS Support

These packages are required when performing a system installation with the `system_locale` defined in the Solaris Security Toolkit software, provided for Solaris 9 OE `sysidcfg`:

```
system_locale=en_US
```

Other packages may be required when using locales other than North America.

Having `ping` available simplifies network and system configuration troubleshooting, so we install the following package:

Package Type	Description
SUNWbip	Basic IP commands (Usr)

We want the system to have `patchadd` and `unzip`, so we install the following package:

Package Type	Description
SUNWswmt	Install and Patch Utilities

When we run the Sun ONE Web Server install script, the following error message is generated:

```
# /opt/iWS6.0sp2/setup
ld.so.1: /opt/iWS6.0sp2/setup: fatal: libC.so.5: open failed: No
such file or directory
Killed
```

Using the grep command, we obtain the following:

```
# grep libC.so.5 /var/sadm/install/contents
/usr/4lib/libC.so.5.0 f none 0755 root bin 147456 26195 822002211
SUNWscbcp
/usr/lib/libC.so.5 f none 0755 root bin 332616 64299 1006882613
SUNWlibC
```

To resolve these error messages, we add the following packages:

Package Type	Description
SUNWlibC	Sun Workshop Compilers Bundled libC
SUNWlibCx	Sun WorkShop Bundled 64-bit libC

While re-installing the system, we add the following SSH packages into the profile:

Package Type	Description
SUNWsshcu	SSH Common, (Usr)
SUNWsshdr	SSH Server, (Root)
SUNWsshdu	SSH Server, (Usr)
SUNWsshr	SSH Client and utilities, (Root)
SUNWsshu	SSH Client and utilities, (Usr)

On system startup the following error message is generated.

```
ld.so.1: /usr/lib/ssh/sshd: fatal: libz.so.1: open failed: No such
file or directory
```

Using the `grep` command, we obtain the following:

```
# grep libz.so.1 /var/sadm/install/contents
/usr/lib/libz.so=./libz.so.1 s none SUNWzlib
/usr/lib/libz.so.1 f none 0755 root bin 60228 6405 1013760222
SUNWzlib
/usr/lib/sparcv9/libz.so=libz.so.1 s none SUNWzlibx
/usr/lib/sparcv9/libz.so.1 f none 0755 root bin 70856 4579
1013760225 SUNWzlibx
```

We add the following packages to resolve the previous error message:

Package Type	Description
SUNWzlib	The Zip compression library
SUNWzlibx	The Zip compression library (64-bit)

At this point, Sun ONE Web Server and Secure Shell function properly. The complete package listing is as follows for our Netra t1 test platform, which has a total of 34 Solaris OE packages:

Package Type	Description
SUNWbip	Basic IP commands (Usr)
SUNWcar	Core Architecture, (Root)
SUNWcarx	Core Architecture, (Root) (64-bit)
SUNWcsd	Core Solaris Devices
SUNWcsl	Core Solaris, (Shared Libs)
SUNWcslx	Core Solaris Libraries (64-bit)
SUNWcsr	Core Solaris, (Root)
SUNWcsu	Core Solaris, (Usr)
SUNWcsxu	Core Solaris (Usr) (64-bit)
SUNWesu	Extended System Utilities
SUNWhmd	SunSwift Adapter Drivers
SUNWhmdx	SunSwift Adapter Drivers (64-bit)
SUNWidecr	IDE device drivers
SUNWidecx	IDE device drivers- 64bit

Package Type	Description
SUNWider	IDE Device Driver, (Root)
SUNWkvm	Core Architecture, (Kvm)
SUNWkvmx	Core Architecture (Kvm) (64-bit)
SUNWlibC	Sun Workshop Compilers Bundled libC
SUNWlibCx	Sun WorkShop Bundled 64-bit libC
SUNWlibms	Forte Developer Bundled shared libm
SUNWlmsx	Forte Developer Bundled 64-bit shared libm
SUNWloc	System Localization
SUNWlocx	System Localization (64-bit)
SUNWnamos	Northern America OS Support
SUNWpd	PCI Drivers
SUNWpdx	PCI Drivers (64-bit)
SUNWsshcu	SSH Common, (Usr)
SUNWsshdr	SSH Server, (Root)
SUNWsshdu	SSH Server, (Usr)
SUNWsshr	SSH Client and utilities, (Root)
SUNWsshu	SSH Client and utilities, (Usr)
SUNWswmt	Install and Patch Utilities
SUNWzlib	The Zip compression library
SUNWzlibx	The Zip compression library (64-bit)

References and Related Resources

Publications

- Howard, John S., and Alex Noodergraaf. *JumpStart™ Technology: Effective Use In the Solaris™ Operating Environment*, The Official Sun Microsystems Resource Series, Prentice Hall, October 2001.
  `http://www.sun.com/blueprints/pubs.html`
- Garfinkel, Simon and Spafford, Gene. *Practical Unix and Internet Security*, Published by O'Reilly & Associates; 04/1996; ISBN: 1565921488.
- Noordergraaf, Alex. "Building a JumpStart Infrastructure," Sun BluePrints OnLine, April 2001.
  `http://www.sun.com/blueprints/0401/BuildInf.pdf`
- Noordergraaf, Alex and Brunette, Glenn. "The Solaris Security Toolkit - Installation, Configuration, and Usage Guide: Updated for version 0.3," Sun BluePrints OnLine, June 2001, `http://sun.com/blueprints/0601/jass_config_install-v03.pdf`
- Noordergraaf, Alex and Brunette, Glenn. "The Solaris Security Toolkit - Quick Start: Updated for version 0.3," Sun BluePrints OnLine, June 2001, `http://sun.com/blueprints/0601/jass_quick_start-v03.pdf`
- Noordergraaf, Alex and Watson, Keith. *Solaris Operating Environment Security - Updated for the Solaris 8 Operating Environment*, Sun BluePrints OnLine, April 2001. `http://www.sun.com/blueprints/0401/security-updt1.pdf`
- *Solaris Advanced Installation Guide* (`http://docs.sun.com`)

Web Sites

- CERT/CC `http://www.cert.org` is a federally funded research and development center working with computer security issues.
- FAQ (frequently asked questions) available from SunSolve™ OnLine (`http://sunsolve.sun.com`)
- Security Focus `http://www.securityfocus.org` is a web site dedicated to discussing security topics.

- Rootshell.com web site `http://www.rootshell.com` provides a searchable list of vulnerabilities posted to full-disclosure mailing lists.
- Attrition site `http://www.attrition.org` maintains an archive of defaced sites for those interested in other's experiences.

About the Author

Alex Noordergraaf has over 10 years experience in the areas of computer and network security. As the Security Architect of the Enterprise Server Products (ESP) group at Sun Microsystems, he is responsible for providing technical leadership to define the security of Sun's next generation servers while addressing security for current products. He is the driving force behind the very popular freeware Solaris Security Toolkit. Prior to his role in ESP, he was a Senior Staff Engineer in the Enterprise Engineering (EE) group of Sun Microsystems, where he developed, documented, and published security best practices through the Sun BluePrints program. Published topics include: Sun Fire Midframe 15K system security, secure N-tier environments, Solaris OE minimization, Solaris OE network settings, and Solaris OE security. He has co-authored two Sun BluePrint books "Jumpstart Technology- Effective Use in the Solaris Operating Environment" and "Enterprise Security Solaris Operating Environment, Security Journal."

Prior to his role in EE, he was a Senior Security Architect with Sun Professional Services where he worked with many Fortune 500 companies on projects that included security assessments, architecture development, architectural reviews, and policy/procedure review and development. He developed and delivered an enterprise security assessment methodology and training curriculum to be used worldwide by SunPS. His customers included major telecommunication firms, financial institutions, ISPs, and ASPs. Before joining Sun, Alex was an independent contractor specializing in network security. His clients included BTG, Inc. and Thinking Machines Corporation.