Securing the Sun Fire™ Midframe System Controller

Updated for SCapp 5.13, Solaris 8 (2/02), and Solaris 9

Alex Noordergraaf and Tony M. Benson, Enterprise Server Products

Sun BluePrints™ OnLine - June, 2002

http://www.sun.com/blueprints

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303 USA
650 960-1300 fax 650 969-9131

Part No.: 816-4940-10
Revision 01, 6/3/02
Edition: June 2002

Securing the Sun Fire Midframe System Controller

This article provides recommendations on how to securely deploy the Sun Fire™ midframe system controller (SC). These recommendations apply to environments where security is a concern, particularly environments where the uptime requirements of the SC and/or the information on the Sun Fire server is critical to the organization.

Many issues are involved in securing the Sun Fire SC. The most significant is its use of insecure administrative protocols. In addition, it is sensitive to some types of network-based attacks such as Denial of Service (DoS) attacks.

The recommendations in this article include building a separate and private SC network, to which the insecure protocols required to manage an SC are restricted. A midframe service processor (MSP) is the secure gateway into the private SC network. A detailed, supported, and secured MSP configuration is described.

This article contains the following topics:

Updates

This Sun BluePrints OnLine article is updated for the Solaris™ 8 (2/02) Operating Environment, version 5.13.0 of the SC application, and version 23 of the SC Real Time Operating System (RTOS). The recommendations in this article should apply to all SC application 5.13 releases.

The main changes are in the SC:

- The `peek` and `poke` commands available in the interactive SC power on self test (SCPOST) facility can now be disabled by a write-protect jumper on the SC board.
- The Telnet service can be disabled. If it is enabled, then a session idle timeout can be set.
- The `showplatform` and `showdomain` commands now indicate the `syslog` facility.
- BugId 4417940, which affected the operation of `setkeyswitch secure` mode, was fixed.
- Network ports 68, 111, and 1024 are disabled on the SC.
- Support for SC failover is introduced.
- Support for Simple Network Time Protocol (SNTP) is introduced to the SC.

Background Information

The following sections provide helpful information for understanding the SC, MSP, hardware and software requirements, and other topics. This section contains the following topics:

Assumptions and Limitations

In this article, our recommendations are based on several assumptions and limitations as to what can be done to secure a Sun Fire system controller (SC) using a midframe service processor (MSP) configuration.

Our recommendations assume a platform based on Solaris 8 Operating Environment (2/02), version 5.13.0 of the SC application, and version 23 of the SC Real Time Operating System (RTOS).

Solaris Operating Environment (Solaris OE) hardening can be interpreted in many ways. For purposes of developing a hardened MSP configuration, we address hardening all possible Solaris OE options. That is, anything that can be hardened is hardened. When there are good reasons for leaving services and daemons as they are, we do not harden or modify them.

Note – Be aware that hardening Solaris OE configurations to the level described in this article may not be appropriate for your environment. For some environments, you may want to perform fewer hardening operations than recommended. The configuration remains supported in these cases; however, additional hardening beyond what is recommended in this article is not supported.

The recommended Solaris OE cluster is End User. While it would be possible to install the MSP with significantly fewer Solaris OE packages, it is not a supported configuration. Only Solaris OE hardening tasks described in this article are supported configurations for the MSP.

Note – Standard security rules apply to hardening Sun Fire SCs and MSPs: *That which is not specifically permitted is denied*.

When addressing security of the MSPs, we focus on MSP functionality inherent in or required by MSP servers. We do not address security for non-MSP servers running Solaris 8 OE. For recommendations on generic Solaris OE security configuration, refer to other sources such as the security-related Sun BluePrints OnLine articles.

In this article, we omit additional software that you can install on the MSP, such as Sun℠ Remote Services Event Monitoring, Sun™ Remote Services Net Connect, and Sun™ Management Center software.

Qualified Software Versions

The configuration discussed in this article has the following software installed.

System Controller

- SC application version 5.13.0
- SC Real Time Operating System (RTOS) version 23

Midframe Service Processor

- Solaris 8 OE (2/02) installed with the End User Cluster
- Latest Security and Recommended Patch Cluster from SUNSOLVE^{SM} ONLINE Web site
- OpenSSH
- Solaris Security Toolkit version 0.3.6
- FixModes software
- MD5 software

Note – The use of Solaris 9 OE and its bundled version of Solaris™ Secure Shell is supported for use on the MSP.

Minimum MSP System Requirements

We cannot make specific recommendations of the hardware requirements because they depend extensively on the number of SCs supported by an MSP, in addition to the software being run on the MSP. For example, if the MSP is running only the software described in this article for several SCs, then a system such as the Netra™ T1 server would be recommended. Alternatively, if the MSP is running additional monitoring and management software for several hundred SCs, then a significantly larger server would be recommended.

The *minimum* hardware and software recommended for an MSP is as follows:

- Sun4U™ architecture
- 8-GByte disk
- 128-MByte RAM
- CD-ROM drive
- SunSwift™ card or, ideally, a Sun Quad FastEthernet™ card
- Solaris 8 OE

Performance and Storage Requirements

The performance and storage requirements for the MSP depend on many variables. Based on the configuration in this article, a low-end Sun4U system such as a Netra T1, Ultra 1, or Ultra 5 systems, has the required performance.

Obtaining Support

The Sun Fire SC and MSP configuration implemented by the Solaris Security Toolkit module (`sunfire_mf_msp-secure.driver`) is a Sun supported configuration. A hardened MSP is supported by Enterprise Services *only* if the security modifications are performed using the Solaris Security Toolkit. Support calls to Sun Enterprise Services are handled the same as other cases.

Note – The Solaris Security Toolkit itself is not a supported Sun product. Only configurations created with the Solaris Security Toolkit are supported.

To obtain Solaris Security Toolkit support, use the Solaris Security Forum link at the following web site:

```
http://www.sun.com/security/jass
```

System Controller (SC)

The Sun Fire system controller (SC) is an embedded device built into the Sun Fire midframe chassis which runs a Real Time Operating System (RTOS). This embedded device runs an application called `SCapp`. The device has limited processing and memory resources and no local non-volatile read / write storage such as hard drives, other than two erasable programmable read only memories (EPROMs). Of these two EPROMs, one stores the RTOS while the other contains the SC application.

Additional information about the SC is available in the *Sun Fire 6800/4810/4800/3800 Platform Administration Manual* and the *Sun Fire 6800/4810/4800/3800 System Controller Command Reference Manual*.

Currently, the SC does not support encrypted or strongly authenticated access and management mechanisms. All management traffic to the SC uses non-encrypted transport mechanisms such as TELNET, FTP, HTTP, and SNMPv1. These are insecure protocols and should not be transmitted across general purpose intranets. In secured environments with strict security policies requiring encryption and strong authentication, these non-secured protocols cannot be used. If these security recommendations are not observed, the SC is an extremely easy target for network-based attacks such as DoS, session sniffing, and / or hijacking.

Because only one password, belonging to the platform administrator, is needed to control the machine, it is critical that insecure protocols required to manage the SC be limited to a private and highly-secured network; referred to as the private SC network throughout the rest of this document. To limit these protocols to one network segment, a gateway system is needed to provide an access and control point. This gateway system should have at least two network interfaces. One interface connects to the private SC network, and the other to the general access intranet or management network.

This gateway system, referred to as the midframe service processor (MSP), is a server on which encrypted and strongly-authenticated management services (for example, SSH, IPsec, and SNMPv2usec) can be installed. Administrators log into the MSP using the encrypted protocols. The insecure and non-encrypted protocols should only be used on the private SC network. If the private SC network is built on physically separate network devices (for example, no VLANs) there is little exposure to network sniffing or other network-based attacks. The recommendations for the placement are built on top of the recommendations made in the Sun BluePrints OnLine article titled *Building Secure N-Tier Environments*.

Domain and SC Isolation and Communication

The Sun Fire midframe hardware architecture was designed to enforce strict separation between domains and limited communication between the domains and SC. However, there must exist a communication path between each domain and the SC so that the SC can provide a virtual console for each domain, access to the OpenBoot™ Prom (OBP), and a mechanism for services and daemons to communicate from the SC to the domains and domains to the SC. This communication path was carefully constructed to enforce the separation of domains and SC, and to ensure that information cannot be leaked between domains or from one domain to another through the SC. The following paragraphs provide additional information on how this communication path was designed and implemented to provide separation between the domains and SC.

The SP communicates with a domain and the domain with the SC via reading and writing to the static random access memories (SRAM) located on the Input/Output (I/O) and CPU boards.

The I/O board SRAM is accessible to CPUs in the domain through a PCI interface. Access to the SRAM on the CPU boards is provided by a local interface on those boards. It is not possible for a domain to use either of these mechanisms to access SRAM located on hardware in other domains. The SC is able to access all SRAMs in the Sun Fire midframe chassis over a separate hardware path called the console bus.

An entire SRAM is not dedicated to this communication channel. The SC specifies which SRAM and location within that SRAM is to be used during domain startup. Specifically, the SP provides this information to domain during its power on self test

(POST) sequence. POST then passes this information to the OpenBoot Prom (OBP) which then passes it on to Solaris OE. In this way the SC is able to define the SRAM to be used and the portion thereof.

Before passing SRAM information to OBP, the SC is responsible for initializing the data structures to be used. Different data structures are used for the portions of SRAM used to communicate between the SP and POST, the SP on OBP, and the SP and Solaris. These different memory structures are referred to as mailboxes. These mailboxes provide a bi-directional communication path between the different components on the domain and SP.

By implementing inter-chassis communications, strict separation is maintained between domains on a Sun Fire midframe. In addition, communication to the SP is strictly limited and does not provide a general purpose connection that could be used to either compromise the SP or leak information through the SP to another domain.

Failover

System controller failover is described in the *Sun Fire 6800/4810/4800/3800 Platform Administration Manual* and the *Sun Fire 6800/4810/4800/3800 System Controller Command Reference Manual*.

The configuration and operation of the SC for failover is not within the scope of this article. However, if the SC is configured for failover, then we recommend that you use SNTP for synchronization of the system clocks. Refer to “Use the SNTP Default Configuration” on page 20.

Terminal Server Usage

We strongly recommend that you use a terminal server that supports the use of SSH to encrypt sessions. This recommendation is made because the terminal server is not on the private SC network, but on the general purpose intranet. If Telnet is used to access the terminal server, then all passwords are passed over the general purpose network, in clear text. This insecure transmission defeats many of the security measures designed into the architecture. Terminal servers supporting SSH are available from Cisco Systems, Perle, and other vendors.

Special Commands for RTOS Shell Access

Two special commands can be issued to the SC, over its serial connection, while it is booting. These two key sequences, Control-A and Control-X, have special capabilities if entered at the serial port within the first 30 seconds after an SC reboot.

The Control-A key sequence displays an RTOS prompt. The Control-X key sequence performs a soft reboot of the SC. This soft reboot is similar to resetting from the OpenBoot PROM on Sun™ Enterprise servers.

Note – The Control-A and Control-X sequences are only accessible over the SC's serial connection. These special control sequences do not work from any Telnet connections to the SC.

The special capabilities of these key sequences are automatically disabled 30 seconds after the Sun copyright message is displayed. Once the capability is disabled, Control-A and Control-X operate as normal control keys.

The security of the SC could be compromised by unauthorized access to the RTOS shell. Access to the serial ports of the SC must be carefully controlled.

Interactive SC Power On Self-Test Mode

If you press the space bar while connecting through the network terminal server to the serial port of the SC, during the SC power on self test (SCPOST) process, the system enters an interactive mode. In this mode, the user has access to a variety of commands and options. No password is required to enter this mode.

Two commands available in the interactive SCPOST mode are `peek` and `poke`. The `peek` command allows a user to inspect the contents of SC memory. The `poke` command can alter the contents of SC memory. Thus, if a malicious user (who is knowledgeable of SC memory addresses) accesses the interactive SCPOST facility, the user could modify the SC platform and/or domain passwords.

This mode is only supported for Sun engineering staff use. End-user access and use of this mode is not supported and is strongly discouraged, because Sun Fire system components can be damaged while in this mode.

Engineering Mode

The platform administration shell can be operated in a special restricted mode known as *Engineering Mode*. Prior to patch 111346-02, this was referred to as *Expert Mode*. Engineering Mode is for use under guidance from Sun internal engineering staff, and is not supported for use under any other circumstance.

Access to *Engineering Mode* is protected by a password. These passwords are only good for a period of time. Passwords are generated internally by Sun on an as needed basis, and as such are not generally available.

Note – Improper use of *Engineering Mode* can damage hardware, override or change any aspect of SC behavior, and lead to breaches of platform security.

Service Mode

The platform administration shell can be operated in a special restricted mode known as *Service Mode*. This mode was introduced with version 5.13.0 of the SC application. *Service Mode* is for use by Sun service staff, and is not supported for use under any other circumstance.

Access to *Service Mode* is protected by a password. It does not share the same password as *Engineering Mode*, but the password management is similar. The password is only good for a period of time. Passwords are generated internally by Sun on an as needed basis, and as such are not generally available.

Note – Improper use of *Service Mode* can damage hardware, override or change aspects of SC behavior, and lead to breaches of platform security.

Write-Protect Jumper

The SC contains several erasable programmable read only memories (EPROMs)—one of which contains the RTOS image. This EPROM is associated with a write-protect jumper (labeled `J1303`). The jumper has two positions, write-protect and write-enable. The factory setting for this jumper is the write-enable position. The jumper is bridged in the write-enable position.

In the write-enable position, the RTOS image can be updated using the `flashupdate` command.

Some organizations may have security policies that require a high degree of protection against the risk of improper access to the RTOS. Where such a requirement exists, you can use the write-protect jumper to provide protection.

In the write-protect position, the following features are disabled:

- `flashupdate`
- Control-A and Control-X commands
- `peek` and `poke` commands in interactive SCPOST mode

Be aware of the following special considerations for using the write-protect jumper:

- To change the position of the write-protect jumper, the SC must be removed from the chassis. Only trained personnel are allowed to perform this procedure.
- When updates are required for the RTOS, it is necessary to power down and remove the SC to change the jumper configuration both before and after the RTOS update.
- During an RTOS update, while the EPROM is not write-protected, appropriate measures must be taken to avoid unauthorized access to the console serial port.
- It is recommended that the platform be configured with a redundant SC, using the SC failover feature to avoid Sun Fire frame downtime.

For instructions and additional information, refer to the *Sun Fire 6800/4810/4800/3800 Platform Administration Manual* and the *Sun Fire 6800/4810/4800/3800 System Controller Command Reference Manual*.

Midframe Service Processor (MSP)

A midframe service processor (MSP) is a separate component that you can use to provide services to the Sun Fire SC. In addition to other services, these services include the following:

- encrypted access point (for SSH, IPsec, or alternative)
- `SYSLOG` server
- flash update services
- `dumpconfig` and `restoreconfig` services
- secure choke point separating SC network traffic from general purpose intranet network traffic

We recommend that you configure the SC to use an external MSP server. For an example of the network topology of an SC and an MSP server, refer to FIGURE 1 on page 29.

An SC can function without an external server such as the MSP, however, some SC functionality and monitoring capabilities are not available. These include flash updates to the SC EPROMs, `SYSLOG` message logging, and configuration backup through `dumpconfig`. These functions are critical to the ongoing maintenance and management of a Sun Fire platform.

Because the MSP is used as a secure access mechanism between general purpose networks and private SC networks, the MSP should not be used for any other tasks. For example, an MSP should not be given additional tasks as a general purpose NFS server.

Note – The MSP should be dedicated to the task of isolating and protecting the SCs from malicious network and user access.

The most secure MSP has the least software installed and the fewest services and administrator accounts. The more secure the MSP, the better the protection provided for the Sun Fire SC.

This recommendation does not mean that you cannot install additional software on the MSP. However, any additional software should be restricted to that which is required to monitor and/or manage the MSP. The MSP is a critical system because it controls access and the flow of information to and from the SC. The MSP should be managed based on the requirements of the organization. For example, in an enterprise where enterprise backup software is used to backup systems, it would be appropriate and prudent to install the required software on the MSP. Conversely, it is not a good practice to use the MSP as a general purpose web server. Evaluate the potential security impact of additional software to ensure that the overall security of the MSP is not adversely affected.

Mapping to Multiple SCs

Depending on the architecture of an environment, it may be desirable to support several SCs from one MSP. This configuration is recommended, from a security perspective, as long as all the systems (MSP and SCs) are within one *administrative domain*.

An administrative domain is a group of systems that are managed by the same or cooperating organizations, perform similar functions, and operate at similar security levels. For example, an administrative domain may include all the database servers in a data center. In this situation, one MSP, or a pair of MSPs, would be appropriate to manage as many of the Sun Fire database servers as needed. This administrative domain must not include the Internet-accessible web servers that access the database servers. Because the web servers are exposed to a significantly greater risk of misuse, they are in a different administrative domain and should be managed by a separate MSP.

Fault Tolerance

The MSP topology described in this article places the MSP as a single point of failure for accessing the SC over Telnet connections, storing SYSLOG files, and other functions of the MSP. Single points of failure adversely affect uptime and should be avoided wherever possible. Several options are available to mitigate some of the risks.

The simplest option is use IP multipathing (IPMP). This option provides link-level redundancy for failures in the network cables, network switch port failures, or a failure of the QFE card port. This option does not protect against more significant hardware failures on the MSP.

Additional redundancy can be obtained by having a cold spare available to replace the MSP if a serious failure occurs. This spare system would be fully configured as the MSP, or `msp01` in this article; however, it would not be powered on. This configuration minimizes most of the downtime associated with fixing the primary system, because a replacement system is already configured and available; it just needs to be powered on when the failed system is powered off.

The most fault resistant configuration would be to cluster two MSPs. The clustering software could then automatically fail over the MSP services from one MSP server to the other in the event of a failure. To not lose access to log files, SYSLOG output, and other data files on the MSP, the two systems would have to share a disk subsystem. Obviously, while this system provides the highest availability, it is also the most complicated. Addressing how this type of a configuration could impact the security posture of the SC is beyond the scope of this article.

Securing the System Controller

When the platform and domains of the SC are configured, make sure to configure them securely. Some of the tasks are performed by the platform administrator, while others are performed by the appropriate domain administrator.

This article focuses on the SC configuration changes required to secure the SC. Normal administrative issues are addressed only when they are impacted by a security modification. For full details on configuring the SC, refer to the system controller publications listed in “Related Resources” on page 58.

Note – Implement the security modifications immediately after the Sun Fire RTOS and SC application has been flashed with the latest firmware updates and before any Sun Fire domains are configured or installed.

Always use the most recent updates available from SUNSOLVE ONLINE Web site.

Securing the SC consists of performing the following tasks:

- “Configuring Platform Administrator Settings” on page 14
- “Rebooting the SC to Implement Settings” on page 24
- “Configuring Domain Administrator Settings” on page 25

Caution – We recommend that you disable the SC failover mechanism before hardening the SCs. Re-enable failover only after you harden and test the entire configuration.

Configuring Platform Administrator Settings

Most of the platform administrator setting configurations are performed through the `setupplatform` command. You can run this command either in an interactive mode where it asks specific questions or a non-interactive mode by specifying the configuration modification required. For the purposes of this article, we run the command in non-interactive mode by using the `-p` option.

To secure the SC, perform the following tasks:

Configure Network Settings

The first task in setting up an SC is to enable networking. This task defines whether the system uses dynamic or static IP addresses, what its hostname is, its IP address, DNS server, and other network information.

In this secured topology, we use static IP addresses. Dynamic host configuration protocol (DHCP) is certainly an option and a DHCP server could be set up and populated with the appropriate MAC and hostname information for the SCs on the MSP. However, the effort required to set up and manage the DHCP server is appropriate only if there are many SCs to configure.

If you use DHCP, configure the DHCP server to provide services only for the private SC network and no other network segments.

All network traffic to the SC is routed through the MSP. Because IP forwarding is not enabled on the MSP, all the packets must be proxied through the MSP. As an additional security measure, this practice allows us to not specify a default router on the SC.

For network-based name resolution, the SC requires a DNS server. In this secured environment, this requirement is not necessary, because the only system the SC communicates with is the MSP. Consequently, no DNS server information is entered while configuring the SC.

We used the following command to enter the changes on the SC:

```
sc0:SC> setupplatform -p network

Network Configuration
---------------------
Is the system controller on a network? [yes]: yes
Use DHCP or static network settings? [dhcp]: static
Hostname [unknown]: ds7-sc0
IP Address [0.0.0.0]: 192.168.100.20
Netmask [0.0.0.0]: 255.255.255.0
Gateway [0.0.0.0]:
DNS Domain [none]: none
Primary DNS Server [0.0.0.0]:
Secondary DNS Server [0.0.0.0]:

Rebooting the SC is required for changes in network settings to
take effect.
```

Configure the Platform Loghost

The next task in configuring the SC is to configure the platform loghost to which all SYSLOG messages are forwarded. The SC has no local disk, so it cannot store these messages locally. They must be forwarded to a central location for storage, reconciliation, and review (for unusual activity). If DNS is not being used, you must take care to define the loghost through the IP addresses. In our example, DNS is not being used, so we enter the IP address.

In addition to specifying the name/IP address of the loghost, the facility level included in the SYSLOG messages can be specified. The SYSLOG protocol provides eight user-defined facility levels: `local0` through `local7`, in addition to the 18 system-defined facilities. However, only the user-defined facility levels can be used while customizing the SC's SYSLOG behavior.

All SC generated SYSLOG messages come from the same IP address—that of the SC. The different SYSLOG facilities must be used to distinguish between messages originated from the platform and each domain. For example, the platform would use the SYSLOG facility `local0`, while `domain-a` would use the SYSLOG facility `local1`, and so on.

The MSP is functioning as the SYSLOG server, so we enter its IP address in the following manner with the corresponding SYSLOG facility level (local0) for the platform:

```
ds7-sc0:SC> setupplatform -p loghost

Loghosts
--------
Loghost [ ]: 192.168.100.10
Log Facility [local0]: local0
```

Details on how to configure the SYSLOG service on the MSP are provided in "Configuring the MSP SYSLOG" on page 43.

Use the showplatform command to display the loghost and log facility for the platform:

```
ds7-sc0:SC> showplatform -p loghost

Loghost for Platform: 192.168.100.10
Log Facility for Platform: local0
```

Define Platform Password

The next task is to set the platform password. The only restrictions on SC platform and domain passwords are the character set supported by ASCII and the terminal emulator in use. The SC uses the MD5 software to generate a hash of the password entered. Correspondingly, all characters entered are significant.

A minimum password length of 16 characters is recommended to promote the use of pass-phrases instead of passwords. Passwords should be comprised of at least lowercase, uppercase, numeric, and punctuation marks. Given the capabilities of current systems to either brute-force access or guess encrypted passwords, an eight character length string is no longer secure.

The following command sets the platform shell password:

```
ds7-sc0:SC> password

Enter new password: xxxxxxxxxxxxxxxx
Enter new password again:  xxxxxxxxxxxxxxxx
```

Note – If a platform administrator's password is lost, refer to "Resetting a Platform Administrator's Lost Password" on page 53.

Define Domain Password

A domain shell is always present for a domain, whether any hardware is assigned to that domain. To avoid potential unauthorized reallocation of hardware to an unused domain, define passwords for all domain shells.

The passwords for each domain should be different from each other, the platform shell, and the Solaris OE images running on the domains. A reliable and secure method of password management is recommended to track all these passwords.

With the release of SCapp 5.13, the platform shell can reset domain passwords. Prior to this release, the only supported method to reset domain password was the `setdefaults` command.

You can set a domain's password from either the shell of the domain or the platform shell with the `password` command. As with the platform password, a minimum password length should be 16 mixed-case alphanumeric characters.

The following command sets the password of `domain-a` from the platform shell:

```
ds7-sc0:SC> password -d a

Enter new password: xxxxxxxxxxxxxxxx
Enter new password again: xxxxxxxxxxxxxxxx
```

Note – All domain shells should have passwords set—regardless of whether they are used and have hardware assigned.

The same command, with the appropriate domain name, sets the passwords for domains `b` through `d`.

If a password was defined for either a platform or domain shell, the password command requires its entry before allowing a new password to be entered. The only exception to this is that the platform administrator can change a domain password without knowing the old password with the release of 5.13 as follows:

```
ds7-sc0:SC> console d
Enter Password:
Connected to Domain D

Domain Shell for Domain D

ds76-sc0:D> disc
Connection closed.
ds7-sc0:SC> password -d d
Enter new password:
Enter new password again:
```

Choose Method for Managing Networked Devices

Simple Network Management Protocol (SNMP) is commonly used to monitor and manage networked devices and systems. Early versions of SNMP, such as SNMPv1 and SNMPv2, suffer from security issues because they don't address issues such as authentication, data integrity checks, and encryption. Updated versions of the protocol are proposed, such as SNMPv2usec and SNMPv3, yet are not fully approved by the IETF, the organization that controls these standards. For more information, refer to "Related Resources" on page 58.

While the full specification of SNMPv2usec does address many of the limitations of the SNMPv1 and v2 protocols, certain components of SNMPv2usec (such as encryption for privacy) are optional and not required for SNMPv2usec compatibility.

The Sun Fire SC only supports the use of SNMPv1. Due to this limitation, we make the following recommendations for choosing a method of monitoring and managing networked devices.

Using Sun Management Center Software

You can use Sun™ Management Center 3.0 (Sun MC) software to manage and maintain your Sun Fire midframe systems. To use Sun MC 3.0 securely, we recommend, in addition to using SNMPv2usec capabilities, that you isolate all of its management traffic to a physically isolated and dedicated management network. This recommendation is based on the network segmentation recommendations presented in the Sun BluePrints OnLine article titled *Building Secure N-Tier Environments*.

Sun MC requires platform agent software to manage the Sun Fire midframe SC. We recommend that you install the software on either the Sun MC server or a separate server. Do not connect the system to the public intranet. Limit access to the platform agent software by not installing it on the MSP.

If isolating the Sun MC server to a completely separate and isolated network is not possible, then install the platform agent software on a separate system. This server requires at least two network interfaces. One connects to the private SC network and the other connects to a private management network, connecting it to the Sun MC server.

Regardless of where the platform agent software is installed, the entire network from the SC to the Sun MC server must be a physically separated and dedicated network. Harden and secure all additional servers, including the Sun MC server.

Disabling SMNP

The alternative is to disable SNMP on the SC and not use any SNMP-based management products. This option provides protection against all possible SNMP-based attacks. It should be noted, however, that disabling these services on the SC prevents SNMP-based management tools from managing the SunFire SC.

Disable the SNMP daemon on the SC as follows:

```
ds7-sc0:SC> setupplatform -p snmp

SNMP
----
Platform Description [Serengeti-24 P1.2]:
Platform Contact [ppb]:
Platform Location []:
Enable SNMP Agent? [yes]: no

May 16 20:59:36 ds7-sc0 Chassis-Port.SC: Stopping SNMP agent.
```

Use the SNTP Default Configuration

The default SC configuration for SNTP is `off`, and we recommend that you configure it to `on`, so that you can use SNTP.

Simple Network Time Protocol (SNTP), described in RFC 2030, is an adaptation of the Network Time Protocol (NTP), described in RFC 1305, and is used to synchronize computer clocks. SNTP does not change the NTP specification; rather it clarifies certain design features of NTP to allow operation in a simple, stateless remote-procedure call (RPC) mode. SNTP clients such as the Sun Fire midframe SC can interoperate with existing NTP or SNTP clients and servers. SNTP is intended to be used only at the extremities of the time synchronization subnet.

A full description of how to architect and implement a time synchronization subnet is out of the scope of this document. We recommend that you understand the concepts described in the following Sun BluePrints OnLine articles:

- *Using NTP to control and Synchronize System Clocks - Part I: Introduction to NTP*
- *Using NTP to control and Synchronize System Clocks - Part II: Basic NTP Administration and Architecture*
- *Using NTP to Control and Synchronize System Clocks - Part III: NTP Monitoring and Troubleshooting*

If configured for SNTP, the SC sends a request to a designated SNTP or NTP unicast server and expects a reply from that server. The SC does not implement the optional authentication method specified in RFC 1305. The SC neither accepts remote administration commands via SNTP, nor does it accept any broadcast traffic.

Because the SC SNTP client uses port 123 UDP without authentication, it is not difficult to spoof the designated NTP or SNTP server; therefore, the SC is vulnerable to a port 123 DoS attack.

The use of RPC-based SNTP introduces another reason why the SCs must be isolated to a physically separate network. We recommend that the MSP be used as the SNTP server for the SC. However, it is important that the MSP be configured to secure its NTP traffic as described in the previously mentioned Sun BluePrints OnLine articles.

The configuration and operation of the SC for failover is not within the scope of this article. If you want to configure the SC for failover, then we recommend that you use SNTP for synchronization of the system clocks. For details, refer to the *Sun Fire 6800/4810/4800/3800 Platform Administration Manual* and the *Sun Fire 6800/4810/4800/3800 System Controller Command Reference Manual*.

Define Hardware Access Control Lists (ACLs)

This task applies and is important only if the Sun Fire server has multiple domains and their resources are restricted in some way. Only when these conditions are present should ACLs be implemented.

By default, all hardware present in the system is accessible to all domains. In our example, a Sun Fire 6800 server is divided into three domains—where each domain has one CPU and I/O board.

Use the platform administrator shell to assign the different CPU and I/O boards into the appropriate domains.

Note – ACLs only limit hardware assignments made while using the domain shells. Hardware assignments made while using the platform shell supersede all ACL definitions.

The capability of the platform shell to assign and reassign hardware components is not restricted by ACLs. We recommend that the platform administrator account be used initially only to assign hardware components to the appropriate domain. After hardware components are assigned to each domain, the administrators should log into the appropriate domain shell account to manage the hardware assigned to that domain. The remainder of this section provides a sample implementation of our recommendations.

First, we use the following command to determine which boards are present:

```
ds7-sc0:SC> showboard

Slot    Pwr Component Type              State     Status
----     --  -------------               ----      -----
SB0      On  CPU Board                   Available Passed
SB2      On  CPU Board                   Available Passed
SB3      On  CPU Board                   Available Passed
IB6      On  PCI I/O Board               Available Passed
IB7      On  PCI I/O Board               Available Passed
IB8      On  PCI I/O Board               Available Passed
```

We view the current set of ACLs defined on the system with the following commands:

```
ds7-sc0:SC> showplatform -p acl

ACL for Domain A: SB0 SB2 SB3 IB6 IB7 IB8
ACL for Domain B: SB0 SB2 SB3 IB6 IB7 IB8
ACL for Domain C: SB0 SB2 SB3 IB6 IB7 IB8
ACL for Domain D: SB0 SB2 SB3 IB6 IB7 IB8
```

We assign the resources to the appropriate domains with the following commands:

```
ds7-sc0:SC> addboard -d a SB0 IB6
ds7-sc0:SC> addboard -d b SB2 IB8
ds7-sc0:SC> addboard -d c SB3 IB7
```

We use the `showboard` command to produce the following output:

```
ds7-sc0:SC> showboard

Slot    Pwr Component Type        State   Status   Domain
----     --  -------------         ----    -----   ------
/N0/SB0  On  CPU Board           Assigned  Passed     A
/N0/SB2  On  CPU Board           Assigned  Passed     B
/N0/SB3  On  CPU Board           Assigned  Passed     C
/N0/IB6  On  PCI I/O Board       Assigned  Passed     A
/N0/IB7  On  PCI I/O Board       Assigned  Passed     C
/N0/IB8  On  PCI I/O Board       Assigned  Passed     B
```

As a final verification, we check the output from `setupplatform` and `showplatform` commands, which appears as follows for our example:

```
ds7-sc0:SC> setupplatform -p acl

ACLs
----
ACL for domain A [ SB0 SB2 SB3 IB6 IB7 IB8 ]: sb0 ib6
ACL for domain B [ SB0 SB2 SB3 IB6 IB7 IB8 ]: sb2 ib8
ACL for domain C [ SB0 SB2 SB3 IB6 IB7 IB8 ]: sb3 ib7
ACL for domain D [ SB0 SB2 SB3 IB6 IB7 IB8 ]:

ds7-sc0:SC> showplatform -p acl

ACL for Domain A: SB0 IB6
ACL for Domain B: SB2 IB8
ACL for Domain C: SB3 IB7
ACL for Domain D:
```

Now three domains, `a` through `c`, are defined on our Sun Fire server; each with one CPU and I/O board.

Note – Although a platform administrator can assign hardware into specific domains, it is up to domain administrators to use those resources appropriately and determine whether those resources are configured into a running domain.

Hardware already assigned to a running domain is not removed if its ACL is modified to restrict it from being used in that domain. Therefore, it is important to assign hardware into domains as soon as it is available in the chassis and before domain administrators assign it.

Configure Telnet

The Telnet service on the SC is enabled by default. You can define the session idle timeout period that applies to all Telnet connections to the SC. The default is no session idle timeout period. The Telnet configuration does not affect the operation of the platform console.

Based on the configuration in this article, we recommend that Telnet timeouts be enabled to a value appropriate for your organization. This practice allows Telnet sessions to be established from the MSP. Refer to the *Sun Fire 6800/4810/4800/3800 System Controller Command Reference Manual* for details on how to configure Telnet timeouts.

If the SC is on a general purpose network, then we recommend that you disable the Telnet service and restrict access to SSH-enabled terminal server access.

To disable the Telnet service, use the `setupplatform -p security` command as follows:

```
ds7-sc0:SC> settupplatform -p security
Security Options ---------------- Enable telnet servers? [yes]: no
Idle connection timeout (in minutes; 0 means no timeout) [0]:
ds7-sc0:SC>
```

For additional instructions, refer to the *Sun Fire 6800/4810/4800/3800 System Controller Command Reference Manual*.

Rebooting the SC to Implement Settings

If needed, reboot the SC to implement your configuration settings. The SC has to be rebooted only if a console message similar to the following is displayed:

```
Rebooting the SC is required for changes in network settings to
take effect.
```

To reboot the SC, enter the following command from the platform shell:

```
ds7-sc0:SC> reboot -y
```

Note – The SC can be rebooted while domains are up and running.

After rebooting the SC, use the `showplatform` command to validate that all the modifications are implemented.

Configuring Domain Administrator Settings

After all of the platform shell configuration modifications are made, implement the domain-specific configuration modifications. Most of the recommended changes are performed using the platform shell.

Only a few domain-specific changes require using domain shells. These modifications are as follows:

- Setting the Loghost and facility for each domain
- Setting the SNMP information

Each of these must be defined individually for each domain. The following samples show these changes for `domain-a`.

Define a Loghost

You must define a Loghost for each of the domains individually. The configuration is similar to that in the “Configure the Platform Loghost” on page 15. In addition, we recommend that you use a facility unique to the frame. By having separate definitions of Loghost for each domain and platform shell, you can use separate `SYSLOG` servers to collect information. In this secured network environment, only one system collects and parses the `SYSLOG` data—the MSP. The facility option helps differentiate `SYSLOG` messages coming from the four different domains and platform shells.

Before using the `setupdomain` command to define the Loghost for each domain, log into the appropriate domain shell.

We perform the following to set our example `domain-a` shell Loghost to be the MSP:

```
ds7-sc0:A> setupdomain -p loghost

Loghosts
--------
Loghost [ ]: 192.168.100.10
Log Facility for Domain A: local1
```

In our example, the Loghost definition defines a facility of `local1`. Previously, the platform shell used `local0`. This example is specific to `domain-a`. Correspondingly, `domain-b` uses `local2`, `domain-c` uses `local3`, and `domain-d` uses `local4`.

Note – The domain shell definition of Loghost has no effect on where the SYSLOG messages generated by a Solaris OE image running on that domain are forwarded. Define the Solaris OE SYSLOG server in the `/etc/syslog.conf` configuration file of the Solaris OE.

For information about how to configure the SYSLOG service on the MSP, refer to "Configuring the MSP SYSLOG" on page 43.

Use the `showdomain` command to display the Loghost and Log Facility for the domain:

```
ds7-sc0:A> showdomain -p loghost

Loghost for Domain A: 192.168.100.10
Log Facility for Domain A: local1
```

Configure Domain SNMP Information

Each domain has unique SNMP configurations that must be configured separately. Some of the domain SNMP information can be the same (for example, domain contact and trap host); however, the public and private community strings must be different for each domain. Different public and private community strings are required so that each domain can be accessed separately. The two community strings provide the mechanism by which individual domains are accessed.

In our secured configuration, the SNMP daemon was disabled in the platform shell. Correspondingly, it is unnecessary to set the public and private community strings, because we are not using SNMP.

If SNMP management or monitoring is used, then non-default SNMP community strings must be selected.

Configure Domain `setkeyswitch`

The `setkeyswitch` command provides functionality similar to the physical key setting on the Sun Enterprise server line. When a Sun Enterprise server is functioning, the `keyswitch` should be in the `secure` setting. With a Sun Fire server, there is no physical key to turn, so this functionality is provided with the `setkeyswitch` command from the platform and domain shells.

The recommended `setkeyswitch` setting for a running domain is `secure`. This setting is very similar to the `setkeyswitch on` position, with a few additional restrictions. Most importantly, in the `secure` setting, the ability to flash update the

CPU/Memory and I/O boards is disabled. Flash updating these boards should only be done by an administrator who has domain shell access on the SC. If the administrator has domain shell access, then using `setkeyswitch` to change from `secure` to `on` is straightforward. Administrators without domain and/or platform access cannot perform this command.

We use the following command to set our example `domain-a` into secure mode:

```
ds7-sc0:A> setkeyswitch secure
```

You can disable two other Sun Fire domain features by using the `setkeyswitch secure` option. When a domain is running in `secure` mode, it ignores `break` and `reset` commands from the SC. This practice is not only an excellent precaution from a security perspective, it also ensures that an accidently issued `break` or `reset` command does not halt a running domain.

Restricting SC OS Access

Some organizations have security policies that require a high degree of protection against the risk of improper access to the RTOS. Where such a requirement exists, you can use the write-protect jumper to provide protection. For more information about the jumper, refer to “Write-Protect Jumper” on page 9.

Although the jumper provides a higher degree of protection, be advised that using it requires additional maintenance effort. When updates are required for the RTOS, a qualified, trained person must power down the system and remove the SC to change the jumper configuration both before and after the RTOS update.

In configurations with a single SC, this task results in platform downtime. For this reason, we recommend that the platform be configured with a redundant SC, using the SC failover feature to avoid Sun Fire frame downtime.

For more details about configuring the SC failover feature, refer to the *Sun Fire 6800/4810/4800/3800 Platform Administration Manual* and the *Sun Fire 6800/4810/4800/3800 System Controller Command Reference Manual*.

During an RTOS update, while the EPROM is not write-protected, appropriate measures must be taken to avoid unauthorized access to the console serial port.

Building a Secure MSP

The MSP (midframe service processor) is the gateway between general purpose internal networks and the private SC network. As such, it controls access between these networks. To effectively protect it against unauthorized access, harden it and implement encrypted access mechanisms.

Hardening is critical to the security of the SC because the default configuration of Solaris OE does not provide the required protection for the MSP.

The recommended Solaris OE installation for the MSP is the End User Cluster rather than the Developer, Entire Distribution, or OEM Installation Clusters. Using the End User Cluster significantly reduces the number of Solaris OE packages installed on the MSP.

Hardening the MSP consists of performing the following tasks:

- "Configuring Network Topology" on page 29
- "Installing Apache Web Server" on page 30
- "Adding Security Software" on page 34
- "Installing Downloaded Software and Implementing Modifications" on page 40
- "Configuring the MSP `SYSLOG`" on page 43

In our example, we use the Solaris Security Toolkit software and the `FixModes` software to secure the MSP. The Solaris Security Toolkit implements recommendations made in the Sun BluePrints OnLine security articles. These recommendations are documented in the following articles:

- *Solaris Operating Environment Security: Updated for the Solaris 8 Operating Environment*
- *Solaris Operating Environment Network Settings for Security: Updated for Solaris 8 Operating Environment*
- *The Solaris Security Toolkit - Installation, Configuration, and Usage Guide: Updated for version 0.3*

Note – You can build the MSP either through an interactive CD-ROM-based or Solaris JumpStart installation. The Solaris Security Toolkit software can be used in either type of installation. Refer to the Sun BluePrints OnLine article *The Solaris Security Toolkit - Quick Start: Updated for Version 0.3*.

Configuring Network Topology

Configure the SC on a private SC network, using the MSP as a non-routing gateway to provide a secure access mechanism between general purpose networks and the private SC network.

In this section, we show a sample network topology containing one Sun Fire 6800 server, two SCs, and one MSP. You can extrapolate other architectures from this sample design. The systems in this topology are as follows:

- `msp01`
- `sc0`
- `sc1`
- `domain-a`
- `domain-b`
- `domain-c`
- `domain-d`
- `nts01`

FIGURE 1 shows a logical diagram and does not include all of the components required to make this sample environment function. Specifically, the network switches required are not addressed. We recommend that you use separate network switches for the private SC network instead of VLANs on a larger switch. Whichever switch you use for the private SC network, we recommend that the switch be managed and monitored the same way as other switches in the environment.

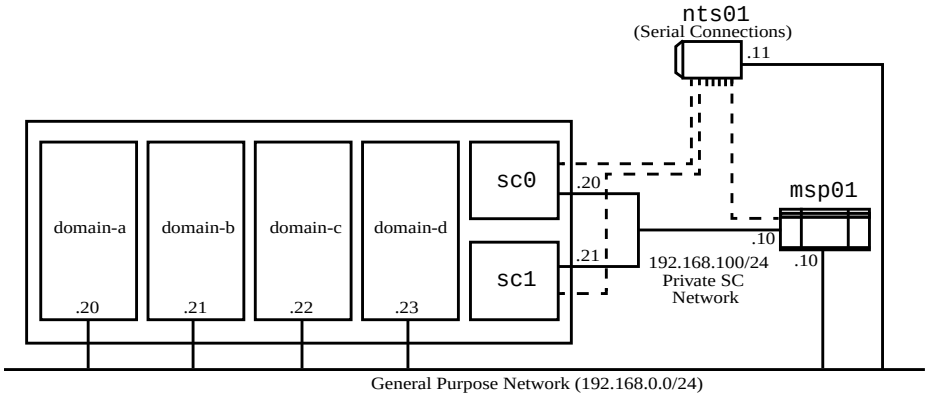

FIGURE 1 Sample Network Topology Configuration

The network diagram illustrates the separate networks we use to isolate the SC from general network traffic. The general network (192.168.0.0/24) is not routed to the private SC network (192.168.100.0/24), because IP Forwarding is disabled on the MSP.

Two access mechanisms are available to connect to the SC in this network architecture:

- An administrator can SSH to the MSP (`msp01` in the diagram), then Telnet from it to the SC.
- An administrator can use the serial connection accessible from the network terminal server (`nts01` in the diagram) as an alternative access mechanism to the SC. In this topology, even when the MSP is not available the SC is accessible through the network terminal server.

Installing Apache Web Server

In the configuration documented in this article, the MSP uses the Apache Web Server to perform Solaris Web Start Flash updates of the SC EPROMs and to provide `restoreconfig` with a transport mechanism to restore SC backups created with `dumpconfig`.

Other web servers can be used on the MSP, instead of the Apache Web server. However, only the recommended Apache configuration is described in this article.

The Apache distribution available in Solaris 8 OE is not installed with the End User Cluster, therefore, it may be necessary to manually install the three required Apache packages. If Apache is already installed on your MSP, some of the following steps may not be necessary.

▼ To Install the Apache Web Server

1. **Obtain the required packages from any Solaris 8 OE 2 of 2 CD-ROM, dated 4/01, in the following directory:**

```
# pwd
/cdrom/sol_8_401_sparc_2/Solaris_8/Product
```

The three required Solaris 8 OE Apache Web Server packages are as follows:

```
system      SUNWapchd      Apache Web Server Documentation
system      SUNWapchr      Apache Web Server (root)
system      SUNWapchu      Apache Web Server (usr)
```

2. **Create a** `tar` **file containing these three packages in the following manner:**

```
# tar -cvf /tmp/apache-pkgs.tar SUNWapchd SUNWapchr SUNWapchu
```

3. **Move the** `tar` **file to the MSP, extract it, and install it using the following commands:**

```
# tar -xf apache-pkgs.tar
# pkgadd -d . SUNWapchd SUNWapchr SUNWapchu
```

4. **Answer** `Yes` **to all the questions asked.**

5. **After the installation is completed, use the** `pkginfo | grep Apache` **command to verify that all three required Apache Web Server packages are present.**

   In the next steps, you'll create an appropriate user and group ID for Apache to run as.

6. **Create a new group by adding the following line to the** `/etc/group` **file:**

```
mspstaff::15:
```

   The example uses a group ID of 15 for `mspstaff`. If this group ID is already used in your environment, select a group ID that is not being used.

7. **Create a user account for the Apache daemon.**

   The following example uses `msphttp`:

```
# /usr/sbin/useradd -m -g mspstaff msphttp
11 blocks
```

8. **For all administrators who need access to files shared by Apache, add their user IDs to the end of the** `mspstaff` **entry in the** `/etc/group` **file.**

   Before starting the Apache daemon, you must configure it. Only a few steps are required to do that.

9. **Create an** `httpd.conf` **file using the following command:**

```
# pwd
/etc/apache
# cp httpd.conf-example httpd.conf
```

10. **Open the** `/etc/apache/httpd.conf` **file in an editor and search for the following line:**

```
#Listen 12.34.56.78:80
```

11. **Add the following line immediately after it.**

    Where the IP address is the IP address of the MSP on the private SC network:

```
Listen 192.168.100.10:80
```

    This step configures the Apache Web Server to respond only to connection requests from the private SC network and not to the general purpose network. This configuration is important because other systems must not be able to access the information that is made available over HTTP to the SC.

    A few other Apache configuration modifications are still required. The Apache Web Server must be told what name to use. Because the name of the MSP on the private SC network may not be resolvable, this configuration uses the IP address of that interface.

12. **Search for the following line in the** `/etc/apache/httpd.conf` **file:**

```
#ServerName new.host.name
```

13. **Add the following line immediately after it.**

    Where the IP address is the IP address of the MSP on the private SC network:

```
ServerName 192.168.100.10
```

    The Apache Web Server must be told what directory structure to make available. This directory is called the `DocumentRoot` and should be the top-most directory of where the Flash archives and backup files are kept.

14. **Search for the following line in the** `/etc/apache/httpd.conf` **file:**

```
DocumentRoot "/var/apache/htdocs"
```

15. **Replace it with the following line.**

    Where the directory is the top-most directory of what is available to the SC:

```
DocumentRoot "/msp"
```

    By default, the Apache Web Server runs as the user ID *nobody* and group ID *nobody*. On the MSP, this should be changed to a more restrictive configuration by creating a new user ID and group ID for the Apache Web server to better control access to the `/msp` directory. In this way, only administrators requiring access to the directory structure accessed by Apache can be added to the Apache group and, therefore, have access. Earlier in this section, you created a user ID and group ID for this purpose. They were `msphttp` and `mspstaff`, respectively.

16. **Now that Apache Web Server is installed, make the following change in the `httpd.conf` file to configure the server to use the user ID and group ID you entered earlier:**

```
User msphttp
Group mspstaff
```

17. **To allow this configuration to work, change the ownerships of the Apache Web Server log file directory using the following command:**

```
# chown -R msphttp:mspstaff /var/apache/logs
```

18. **Create the `/msp` directory on the MSP; use a partition with adequate free space.**

    In the following example, the directory is created on the `/`, or `root`, file system of `msp01`:

```
# mkdir /msp
```

19. **Set the ownerships and permissions of the `/msp` directory to the `msphttp` user ID and `mspstaff` group ID with the following commands:**

```
# chown msphttp:mspstaff /msp
# chmod 770 /msp
```

20. **Start the Apache Web Server with the following command:**

```
# /etc/init.d/apache start
httpd starting.
```

The Apache Web server is now ready to function as a `restoreconfig` server and can be used as a `flashupdate` server.

Adding Security Software

The next stage in hardening an MSP requires downloading and installing additional software security packages. This section covers the following tasks:

- "Install Solaris Security Toolkit Software" on page 34
- "Download Recommended Patch Cluster Software" on page 35
- "Download FixModes Software" on page 37
- "Download OpenSSH Software" on page 38
- "Download the `MD5` Software" on page 39

Note – Of the software described in this section, the Solaris Security Toolkit, Recommended and Security Patch Cluster, FixModes, and MD5 software are required. Instead of OpenSSH, you can substitute a commercial version of SSH, available from a variety of vendors. You must install an SSH product on the MSP.

Install Solaris Security Toolkit Software

The Solaris Security Toolkit software must be downloaded first, then installed on the MSP. Later, you'll use the Solaris Security Toolkit software to automate installing other security software and implementing the Solaris OE modifications for hardening the MSP.

The primary function of the Solaris Security Toolkit software is to automate and simplify building secured Solaris OE systems based on the recommendations contained in this and other security-related Sun BluePrints OnLine articles.

Note – The following instructions use filenames that are correct only for version 0.3.6 and later of the Solaris Security Toolkit software.

▼ To Download Solaris Security Toolkit Software

1. **Download the latest version of the source file.**

   At the time of this publication, the version is `SUNWjass-0.3.6.pkg.Z`. The source file is located at:

   ```
   http://www.sun.com/security/jass
   ```

2. **Extract the source file into a directory on the server using the `uncompress` command**:

   ```
   # uncompress SUNWjass-0.3.6.pkg.Z
   ```

3. **Install the Solaris Security Toolkit software onto the server using the `pkgadd` command**:

   ```
   # pkgadd -d SUNWjass-0.3.6.pkg SUNWjass
   ```

   Executing this command creates the `SUNWjass` subdirectory in `/opt`. This subdirectory contains all Solaris Security Toolkit directories and associated files. The script `make-pkg`—included in Solaris Security Toolkit software releases since version 0.3—allows administrators to create custom packages using a different installation directory.

Download Recommended Patch Cluster Software

Patches are regularly released by Sun to provide Solaris OE fixes for performance, stability, functionality, and security. It is critical to the security of a system that the most up-to-date patch is installed. To ensure that the latest Solaris OE Recommended and Security Patch Cluster is installed on the MSP, this section describes how to download the latest patch cluster.

Downloading the latest patch cluster does not require a SUNSOLVE ONLINE program support contract.

Note – Apply standard best practices to all patch installations. Before installing any patches, evaluate and test them on non-production systems or during scheduled maintenance windows.

▼ To Download Recommended Patch Cluster Software

1. **Download the latest patch from the SUNSOLVE ONLINE Web site at:**

   `http://sunsolve.sun.com`

2. **Click on the "Patches" link at the top of the left navigation bar.**

3. **Select the appropriate Solaris OE version in the "Recommended Solaris Patch Clusters" box.**

   In our example, we select Solaris 8 OE.

4. **Select the best download option, either HTTP or FTP, with the associated radio button, then click "Go."**

   A "Save As" dialog box is displayed in your browser window.

5. **Save the file locally.**

6. **Move the file securely to the MSP with the `scp` command, or `ftp` if `scp` is not available.**

   The `scp` command used should be similar to the following:

```
% scp 8_Recommended.zip msp01:/var/tmp
```

7. **Move the file to the `/opt/SUNWjass/Patches` directory and uncompress it as follows:**

```
# cd /opt/SUNWjass/Patches
# mv /var/tmp/8_Recommended.zip .
# unzip 8_Recommended.zip
Archive:     8_Recommended.zip
   creating: 8_Recommended/
  inflating: 8_Recommended/CLUSTER_README
  inflating: 8_Recommended/copyright
  inflating: 8_Recommended/install_cluster
[. . .]
```

Later, using the Solaris Security Toolkit software, you'll install the patch after downloading all the other security packages.

Note – If you do not place the *Recommended and Security Patches* software into the `/opt/SUNWjass/Patches` directory, a warning message displays when you execute the Solaris Security Toolkit software.

Download FixModes Software

FixModes is a software package that tightens the default Solaris OE directory and file permissions. Tightening these permissions can significantly improve overall security of the MSP. More restrictive permissions make it even more difficult for malicious users to gain privileges on a system.

▼ To Download FixModes Software

1. **Download the FixModes pre-compiled binaries from:**

   `http://www.sun.com/blueprints/tools/FixModes_license.html`

   The FixModes software is distributed as a precompiled and compressed `tar` file formatted for systems based on SPARC. The file name is `FixModes.tar.Z`.

2. **Once downloaded, move the file securely to the MSP with the `scp` command, or `ftp` if `scp` is not available.**

   The `scp` command used should be similar to the following

```
% scp FxiModes.tar.Z msp01:/var/tmp
```

3. **Save the file, `FixModes.tar.Z`, in the Solaris Security Toolkit `Packages` directory in `/opt/SUNWjass/Packages`.**

   The following commands perform these tasks:

```
# cd /opt/SUNWjass/Packages
# mv /var/tmp/FxiModes.tar.Z .
```

Caution – Leave the file in its compressed state.

Later, using the Solaris Security Toolkit software, you'll install the FixModes software after downloading all the other security packages.

Download OpenSSH Software

In any secured environment, the use of encryption in combination with strong authentication is required to protect user-interactive sessions. At a minimum, administrator access to the SC through Telnet sessions and platform/administrator shells must be encrypted. This requirement is one of the major reasons for the presence of the MSP.

The tool most commonly used to implement encryption is Secure Shell (SSH) software, whether a commercial or open source (freeware) version. To implement all the security modifications performed by the Solaris Security Toolkit software and recommended in this article, you must implement an SSH software product.

Information on where to obtain commercial versions of SSH is provided in "Related Resources" on page 58.

The Solaris Security Toolkit software disables all non-encrypted user-interactive services and daemons on the system, in particular daemons such as `in.rshd`, `in.telnetd`, and `in.ftpd`.

Note – The `in.ftpd` daemon is re-enabled during `dumpconfig`. Refer to "Backing Up and Restoring Configurations" on page 45.

Access to the system can be gained with SSH similarly to what is provided by RSH, Telnet, and FTP.

Note – If you choose to use an SSH product other than OpenSSH, install and configure it before or during the Solaris Security Toolkit software run.

▼ To Download OpenSSH Software

- **Obtain the following Sun BluePrints OnLine article and use the instructions in the article for downloading the software.**

  A Sun BluePrints OnLine article about how to compile and deploy OpenSSH titled: *Building and Deploying OpenSSH on the Solaris Operating Environment* is available at:

  ```
  http://www.sun.com/blueprints/0701/openSSH.pdf
  ```

  Later, using the Solaris Security Toolkit software, you'll install the OpenSSH software after downloading all the other security packages.

Caution – Do not compile OpenSSH on the MSP and do not install the compilers on the SC. Use a separate Solaris OE system—running the same Solaris OE version, architecture, and mode (for example, Solaris 8 OE, Sun4U, and 64 bit)—to compile OpenSSH. If you implement a commercial version of SSH, then no compiling is required.

Download the MD5 Software

The MD5 software validates MD5 digital fingerprints on the MSP. Validating the integrity of Solaris OE binaries provides a robust mechanism to detect system binaries that are altered or *trojaned* (hidden inside something that appears safe) by unauthorized users. By modifying system binaries, attackers provide themselves with back-door access onto a system; they hide their presence and cause systems to operate in unstable manners.

▼ To Install the MD5 Software (Intel and SPARC)

1. **Download the MD5 binaries from the following web site:**

   `http://www.sun.com/blueprints/tools/md5_license.html`

   The MD5 programs are distributed as a compressed `tar` file.

2. **Move the file `md5.tar.Z` securely to the MSP with the `scp` command, or `ftp` if `scp` is not available.**

   The `scp` command used should be similar to the following:

```
% scp md5.tar.Z msp01:/var/tmp
```

3. **Copy the file, `md5.tar.Z`, to the Solaris Security Toolkit `Packages` directory in `/opt/SUNWjass/Packages`**

Caution – Do not uncompress the `tar` archive.

After the MD5 software is saved to the `/opt/SUNWjass/Packages` directory, the execution of the Solaris Security Toolkit installs the software.

After the MD5 binaries are installed, you can use them to verify the integrity of executables on the system through the Solaris Fingerprint Database. More information on the Solaris Fingerprint Database is available in the Sun BluePrints OnLine article titled *The Solaris™ Fingerprint Database - A Security Tool for Solaris Software and Files*.

4. **(Optional) Download and install Solaris Fingerprint Database Companion and Solaris Fingerprint Database Sidekick software from the SUNSOLVE ONLINE Web site at:**

   `http://sunsolve.sun.com`

   We strongly recommend that you install these optional tools and use them with the MD5 software. These tools simplify the process of validating system binaries against the database of MD5 checksums. Use these tools frequently to validate the integrity of the Solaris OE binaries and files on the cluster nodes.

   These tools are described in the *The Solaris™ Fingerprint Database - A Security Tool for Solaris Software and Files* article.

Installing Downloaded Software and Implementing Modifications

The Solaris Security Toolkit version 0.3.6 and later provides a driver (`sunfire_mf_msp-secure.driver`) for automating the installation of security software and Solaris OE modifications. The driver performs the following tasks:

- Installs and executes the FixModes software to tighten file system permission
- Installs the `MD5` software
- Installs the Recommended and Security Patch Cluster software
- Implements almost 100 Solaris OE security modifications

Note – The actions performed by each of the scripts is described in the Sun BluePrints OnLine article "*The Solaris Security Toolkit - Internals: Updated for Version 0.3*." The hardening described is performed in standalone mode, not JumpStart mode, because the MSP was built using an interactive Solaris OE installation. For details on the differences between standalone mode and JumpStart mode, refer to the Solaris Security Toolkit documentation.

Note – During the installation and modifications implemented in this section, all non-encrypted access mechanisms to the MSP —such as Telnet, RSH, and FTP—are disabled. The hardening steps do not disable console serial access over SC serial ports.

▼ To Install Downloaded Software and Implement Changes

- **Execute the** `sunfire_mf_msp-secure.driver` **script as follows:**

```
# pwd
/opt/SUNWjass
# ./jass-execute -d sunfire_mf_msp-secure.driver
./jass-execute: NOTICE: Executing driver, sunfire_mf_msp-
secure.driver

============================================================
sunfire_mf_msp-secure.driver: Driver started.
============================================================

============================================================
JASS Version:  0.3.6
Node name:     mako-ss
Host ID:       80c68774
Host address:  192.168.100.10
MAC address:   8:0:20:c6:87:74
Date:          Wed Feb 20 09:47:43 PST 2002
============================================================
[...]
```

▼ To View the Contents of the Driver File

- **To view the contents of the driver file and obtain information about the Solaris OE modifications, refer to the Solaris Security Toolkit documentation available either in the** `/opt/SUNWjass/Documentation` **directory or through the web at:**

  `http:/www.sun.com/security/jass`

  For information about other scripts in the Solaris Security Toolkit software, refer to the Sun BluePrints OnLine article titled *Solaris Security Toolkit Internals: Updated for Version 0.3*.

▼ To Undo a Solaris Security Toolkit Run

Each Solaris Security Toolkit run creates a run directory in `/var/opt/SUNWjass/run`. The names of these directories are based on the date and time the run is initiated. In addition to displaying the output to the console, the Solaris Security Toolkit software creates a log file in the `/var/opt/SUNWjass/run` directory.

Caution – Do not modify the contents of the `/var/opt/SUNWjass/run` directories under any circumstances. Modifying the files can corrupt the contents and cause unexpected errors when you use Solaris Security Toolkit software features such as `undo`.

The files stored in the `/var/opt/SUNWjass/run` directory track modifications performed on the system and enable the `jass-execute` undo feature.

- **To undo a run or series of runs, use the `jass-execute -u` command.**

  For example, on a system where two separate Solaris Security Toolkit runs are performed, you could undo them by using the following command and options:

```
# pwd
/opt/SUNWjass
# ./jass-execute -u
Please select from one of these backups to restore to
1. September 25, 2001 at 06:28:12 (/var/opt/SUNWjass/run/
20010925062812)
2. April 10, 2002 at 19:04:36 (/var/opt/SUNWjass/run/
20020410190436)
3. Restore from all of them
Choice? 3
./jass-execute: NOTICE: Restoring to previous run
//var/opt/SUNWjass/run/20020410190436

============================================================
undo.driver: Driver started.
============================================================
[...]
```

Refer to the Solaris Security Toolkit documentation for details on the capabilities and options available in the `jass-execute` command.

Configuring the MSP SYSLOG

The MSP is configured to function as the SYSLOG repository for all SYSLOG traffic generated by the SC. The behavior of the SYSLOG daemon is controlled through the file `/etc/syslog.conf`; in this file, selectors and actions are specified.

Each SYSLOG selector specifies the facility (for example, `kern`, `daemon`, `auth`, and `user`) and level at which a message is logged. Five levels ranging from most serious (`emerg`) to least serious (`debug`) are available. The facility groups log messages together by subsystem. For instance, all kernel messages are grouped together through the facility `kern`. Some of the facilities available include:

- `kern`
- `daemon`
- `auth`
- `mail`
- `local0-7`

For a complete listing of SYSLOG facilities, refer to the `syslogd`(1m) man page.

Also, it is possible to substitute a wildcard (*) for the facility name in the `syslog.conf` file. This approach is particularly useful when all messages (for example, `*.debug`), or all messages at one level or higher, must be logged (for example, `*.kern`).

Each SYSLOG message includes a level. This level specifies the type of message being generated. The most critical level is `emerg`, which is only used on messages of particular importance. Correspondingly, the log level `debug` indicates that a message contains debugging information and may not be particularly important. Some of the levels available in the `syslog.conf` include:

- `emerg`
- `crit`
- `err`
- `notice`
- `debug`

For a complete listing of SYSLOG levels, refer to the `syslogd`(1m) man page.

Although you can use a wildcard to define a facility, you cannot use it to define a level. Hence, the entry `*.debug` is acceptable; however, the corresponding entry of `auth.*` is incorrect and cannot be used.

In the MSP configuration, we recommend for the secured configuration that all SYSLOG messages be stored both in the `/var/adm/messages` file and in a separate file containing only Sun Fire Midframe SYSLOG traffic.

Note – It is not recommended that the SYSLOG traffic be forwarded from the MSP to another SYSLOG server. If this were done, then a SYSLOG message after being forwarded from the MSP would identify itself as having been generated on the MSP and not the SC, as would actually be the case.

The recommended `syslog.conf` should be similar to the following:

```
*.debug                 /var/adm/messages
local0.debug            /var/adm/sc-messages-platform
local1.debug            /var/adm/sc-messages-domain-a
local2.debug            /var/adm/sc-messages-domain-b
local3.debug            /var/adm/sc-messages-domain-c
local4.debug            /var/adm/sc-messages-domain-d
kern.crit               console
```

This configuration logs all incoming messages to `/var/adm/messages`, all SC messages to `/var/adm/sc-messages-<name>`, and displays all critical kernel messages on the console.

If an automated log parsing tool such as `logcheck` or `swatch` is used, it may be appropriate to generate one file containing the SYSLOG messages from the platform and all the domains. If this consolidated file is required, then add the following lines to those listed previously:

```
local0.debug                /var/adm/sc-messages
local1.debug                /var/adm/sc-messages
local2.debug                /var/adm/sc-messages
local3.debug                /var/adm/sc-messages
local4.debug                /var/adm/sc-messages
```

This configuration logs all incoming SYSLOG messages to `/var/adm/sc-messages` for reconciliation by an automated tool.

This configuration is relatively generic and should only be considered a starting point for configuring the SYSLOG daemon on the MSP for an organization.

Note – It is critical the two columns be separated by tabs and not spaces. If spaces are used in an entry, the SYSLOG daemon will ignore that entry.

Backing Up, Restoring, and Updating the SC

This section provides information and recommendations for securely backing up and restoring the SC. In this section, the MSP is used as the `dumpconfig`, `restoreconfig` and `flashupdate` server.

Backing Up and Restoring Configurations

The `dumpconfig` command uses the FTP protocol to save the current platform and domain configurations to the MSP server. The `restoreconfig` command uses either the FTP or HTTP protocol to restore a previously saved configuration to the SC from the MSP server.

For complete descriptions and usage of the `dumpconfig` and `restoreconfig` commands, refer to the *Sun Fire 6800/4810/4800/3800 Platform Administration Manual* and the *Sun Fire 6800/4810/4800/3800 System Controller Command Reference Manual*.

All stored platform and domain configuration information is included in the dump file. This information includes the MD5 hash of the platform and domain administrator passwords, the OBP password, and the SNMP community strings.

The dump file is not encrypted. Hence the MD5 hash of the platform and domain administrator passwords and the non-encrypted OBP password and SNMP community strings are transmitted in clear text during the `dumpconfig` operation. For this reason, the dump files are saved on the MSP, thus ensuring that the insecure transmission of information is restricted to the private SC network, thus minimizing exposure to network snooping.

When a `restoreconfig` operation is carried out, the entire saved configuration is restored. This includes the platform administrator and domain administrator passwords. It is essential to ensure that the passwords are known before this operation is carried out. Refer to “Configuring Platform Administrator Settings” on page 14 and “Configuring Domain Administrator Settings” on page 25.

The Apache Web Server on the MSP is configured such that the `/msp` directory is made available to the SC. All backup and restore operations to the MSP must be contained in this directory. Because the backup files created during a `dumpconfig` are not differentiated by name or date, it is important that separate directories be created for each backup for version control and tracking. The recommended solution is to create a directory for each `dumpconfig` using the year, month, day, and hour. For example: the `dumpconfig` performed on July 16th, 2001 at 7 p.m. would be stored in a directory called 2001071619.

Backing Up Platform and Domain Configurations

Although the MSP is configured to respond to HTTP, it does not normally respond to FTP because the FTP service is disabled during MSP setup. To perform a `dumpconfig`, the FTP service needs to be enabled on the MSP.

After saving configurations, disable the FTP service again on the MSP. The MSP is configured such that a user ID and password are required for this operation, and the user ID should be used only for `dumpconfig` and `restoreconfig` operations.

▼ To Back Up Configurations on the MSP

1. **To enable the FTP service on the MSP, log in to the MSP using Secure Shell, then** `su` **to root.**

2. **Edit the file** `/etc/inetd.conf`**, and uncomment the following FTP entry:**

```
#ftp stream  tcp6    nowait  root  /usr/sbin/in.ftpd  in.ftpd -l
```

3. **Send the** `inetd` **daemon a SIGHUP signal with the following commands:**

```
# ps -ef | grep inetd
    root   221     1  0   Jun 08 ? 0:00 /usr/sbin/inetd -s -t
# kill -HUP 221
```

4. **Create a directory with the appropriate time and date stamp on the MSP.**

   Before the actual `dumpconfig` command can be run, a directory on the MSP must be created with the appropriate time and date stamp. Based on the example (July 16th, 2001 at 7 p.m. would be stored in a directory called 2001071619), the following directory would be created:

```
# mkdir /msp/2001071619
# chown mspHTTP:mspstaff /msp/2001071619
# chmod 770 /msp/2001071619
```

5. **At the SC, dump the configuration using FTP with a user name and password.**

Note – The following example assumes a user name “blueprints” and password “t00lk1t” on the MSP.

The command and results should be similar to the following:

```
ds7-sc0:SC> dumpconfig -f ftp://blueprints:t00lk1t@192.168.100.10/msp/2001071619
Created: ftp://blueprints:t00lk1t@192.168.100.10/msp/2001071619/ds7-sc0.nvci
Created: ftp://blueprints:t00lk1t@192.168.100.10/msp/2001071619/ds7-sc0.tod
```

6. **When the dump is complete, conclude the process by disabling the FTP entry in the** `/etc/inetd.conf` **by commenting out the following line in the** `/etc/inetd.conf`:

```
ftp  stream  tcp6  nowait  root /usr/sbin/in.ftpd in.ftpd -l
```

7. **Send the** `inetd` **daemon a SIGHUP signal in the following manner**:

```
# ps -ef | grep inetd
    root 221 1  0   Jun 08 ?   0:00 /usr/sbin/inetd -s -t
# kill -HUP 221
```

8. **Confirm that the FTP service is disabled by executing the following commands**:

```
# ftp localhost
ftp: connect: Connection refused
ftp> quit
```

Restoring Platform and Domain Configurations

When it is necessary to restore configuration settings, first ensure that at least the platform administration password contained in the chosen dump file is known by the platform administrators. With the release of 5.13, it is possible for a platform administrator to reset the domain passwords without knowing the old ones. Ideally, of course, both the domain and platform passwords contained in the dump file should be known.

To avoid the necessity of enabling the FTP service on the MSP for this operation, we recommend that you use `restoreconfig` operation using HTTP. As with the `dumpconfig` operation, a user ID and password are required for this operation, and the user ID should only be used for `dumpconfig` and `restoreconfig` operations.

Restoring configuration settings using HTTP is documented in *Sun Fire 6800/4810/4800/3800 Platform Administration Manual* and the *Sun Fire 6800/4810/4800/3800 System Controller Command Reference Manual*. The only considerations are ensuring that the appropriate backup directory is available through the web server and that the passwords used for the domain and platform administrators, in the configuration being restored, are known.

Updating the SC

The `flashupdate` feature updates the firmware running on the SC, the CPU/Memory boards, and the I/O assemblies. The update is initiated by using the `flashupdate` command on the SC. The source flash image can be on a server or another board of the same type. This section describes updates executed from an image on a server. The MSP is used as the server for `flashupdate` images.

To avoid the necessity of enabling FTP on the MSP for this operation, we recommend that you carry out the `flashupdate` operation using HTTP.

The MSP is configured such that a user ID and password are required for this operation, and the user ID should only be used for `flashupdate` operations.

Caution – It is important to be sure of the authenticity and integrity of the flash images before they are loaded from the server using the `flashupdate` command. Loading a corrupted or malicious image can cause damage to hardware, and may compromise security.

It is recommended that the RTOS be updated whenever SCapp is updated. When a new firmware is released, SCapp is normally only tested with the RTOS included with the firmware update. If an RTOS flash fails, then a service call to Sun is required to replace or repair the SC. In order to establish whether a RTOS flash is

necessary, refer to the product release notes accompanying the image and the `flashupdate` command documentation in the *Sun Fire 6800/4810/4800/3800 Platform Administration Manual*.

▼ To Update the SC

1. **Download the latest `flashupdate` for the SC from the Product Patches section of the SUNSOLVE ONLINE Web site at:**

   `http://sunsolve.sun.com`

2. **Make a note of the checksum listed for the patch in the Patch Checksums section of the SUNSOLVE ONLINE Web site, similar to the following:**

```
111346-02.zip
      MD5: 5e84f09ebf5743eb5426b5be6c6a777f
      SysV Sum: 7075   13729
      Sum: 43381   13729
```

3. **Confirm that the checksum of the file matches the checksum listed on the SUNSOLVE ONLINE Web site, using the following commands:**

```
# sum 111346-02.zip
7075 13729 111346-02.zip
# sum -r 111346-02.zip
43381  13729 111346-02.zip
```

A more robust file integrity check is to use the MD5 hash value listed for the patch. For more information about downloading and using MD5 hashes to verify patch integrity, refer to the Sun BluePrints OnLine article titled *The Solaris™ Fingerprint Database - A Security Tool for Solaris Software and Files.*

4. **Unpack the files containing the patch and place them in a subdirectory under the Apache Web Server document root directory** `/msp` **as follows:**

```
# cd /msp
# unzip 111346-02.zip
Archive:  111346-02.zip
   creating: 111346-02/
  inflating: 111346-02/Install.info
  inflating: 111346-02/VERSION.INFO
  inflating: 111346-02/copyright
  inflating: 111346-02/sgcpu.flash
  inflating: 111346-02/sgpci.flash
  inflating: 111346-02/sgrtos.flash
  inflating: 111346-02/sgsc.flash
  inflating: 111346-02/README.111346-02
```

5. **Follow the instructions in the** `Install.info` **file.**

   In our example, `sc-app`, `SB0`, `SB2`, `IB7`, and `IB9` are to be updated from version 5.11.6 to 5.11.7. The RTOS will be updated from release 17 to 17B. Not all system boards are powered up, so the `all` option cannot be used.

The following example downloads and installs the `flashupdate` file from the MSP:

```
ds7-sc0:SC> flashupdate -f http://blueprints:t00lk1t@192.168.100.10/
111346-02 SB0 SB2 IB7 IB9 scapp rtos

The RTOS flash image will be upgraded automatically during the next boot. The
ScApp flash image will be upgraded automatically during the next boot. After
this update you must reboot each active domain that you have upgraded. After
this update, the system controller will automatically reboot itself.
Do you want to continue? [no] y

Retrieving: http://blueprints:t00lk1t@192.168.100.10/111346-02/sgcpu.flash
Validating  ............ Done

Programming PROM 0 on /N0/SB0
Erasing     ............ Done
Programming ............ Done
Verifying   ............ Done

Programming PROM 1 on /N0/SB0
Erasing     ............ Done
Programming ............ Done
Verifying   ............ Done

Programming PROM 0 on /N0/SB2
Erasing     ............ Done
Programming ............ Done
Verifying   ............ Done

Programming PROM 1 on /N0/SB2
Erasing     ............ Done
Programming ............ Done
Verifying   ............ Done

Retrieving: http://blueprints:t00lk1t@192.168.100.10/111346-02/sgpci.flash
Validating  .... Done

Programming PROM 0 on /N0/IB7
Erasing     .... Done
Programming .... Done
Verifying   .... Done

Programming PROM 0 on /N0/IB9
Erasing     .... Done
Programming .... Done
Verifying   .... Done

Rebooting the SC to automatically update flash image.
```

The SC reboots, and the `flashupdate` proceeds as follows:

```
Copyright 2001 Sun Microsystems, Inc.  All rights reserved.

RTOS version: 17
ScApp version: 5.11.6
SC POST diag level: off

Auto Flashupdate

Retrieving: http://blueprints:t00lk1t@192.168.100.10/111346-
02/sgrtos.flash

Retrieving: http://blueprints:t00lk1t@192.168.100.10/111346-
02/sgsc.flash
Validating
.................................................. Done

Updating: RTOS
Erasing     .......... Done
Programming .......... Done
Verifying   .......... Done

Updating: ScApp from version 5.11.6 to version 5.11.7
Erasing
.................................................. Done
Programming
.................................................. Done
Verifying
.................................................. Done

Flashupdate completed successfully.
The SC is being rebooted to use the new images.
```

The SC then reboots with the new image.

6. **Halt the Solaris OE image running in each domain gracefully by using the `shutdown` command.**

   You must perform this step before issuing the `setkeyswitch off` command.

7. **For each domain affected by the updates, set the** `keyswitch` **to the** `off` **position by issuing the** `setkeyswitch off` **command from the domain shell.**

   In the following example, `domain-a` is affected:

```
ds7-sc0:A> setkeyswitch off

This will abruptly terminate Solaris in domain A.
Do you want to continue? [no] y
```

8. **Set the domain** `keyswitch` **to the** `on` **position using the following** `setkeyswitch on` **command:**

```
ds7-sc0:A> setkeyswitch on
```

   The `flashupdate` operation is now complete. At this point, the domain boots itself either to the OBP prompt or Solaris OE. In either case, the flashupdate operation is now complete.

Resetting a Platform Administrator's Lost Password

This section contains a procedure, documented in the README file of patch `800054-01`, on how to use the Control-A and Control-X commands to reset a platform administrator's password.

If a platform administrator's password is lost, use the following procedure to clear the original password.

Note – This procedure should be used only as a last resort, because it requires resetting BootFlags settings.

▼ To reset a platform administrator's password:

1. **Reboot the system controller.**

```
ds7-sc0:SC> reboot
reboot
Are you sure you want to reboot the system controller now? [no] y
```

2. **During the first 30 seconds, press Control-A.**

   You must enter the keyboard sequence within the first 30 seconds of reboot for the command to be accepted. The RTOS prompt is displayed.

```
->
```

3. **Make a note of the current BootFlags settings.**

   You will need these to restore the BootFlags to their original values.

```
-> getBootFlags()
value = 12 = 0xc
```

4. **Save the `0x` number for use later when you restore the BootFlags settings.**

5. **Change the BootFlags to disable autoboot.**

```
-> setBootFlags (0x10)
value = 12 = 0xc
```

6. **Reboot the system controller by pressing Control-X.**

   After you reset it, the RTOS prompt is displayed.

7. **Reset the system controller platform password by entering the following commands:**

```
-> kernelTimeSlice 5
value = 0 = 0x0
-> javaConfig
Loading JVM...done
value = 0 = 0x0
-> javaClassPathSet "/sc/flash/lib/scapp.jar:/sc/flash/lib/
jdmkrt.jar"
value = 30908120 = 0x1d79ed8
-> javaLoadLibraryPathSet "/sc/flash"
value = 33546104 = 0x1ffdf78 = userSigMon + 0x678
-> java "-Djava.compiler=NONE -Dline.separator=\r\n
sun.serengeti.cli.Password"
value = 0 = 0x0
```

The system controller displays the following messages:

```
-> Clearing SC Platform password...
Done. Reboot System Controller.
```

Wait until all of the messages are displayed.

8. **Restore the BootFlags settings to their original values using the** `setBootFlags`**() command. Use the value displayed for Step 3.**

```
-> setBootFlags (0xC)
value = 16 = 0x10
```

9. **Reboot the system controller by pressing Control-X.**

   When you reboot the SC, the platform administrator's password is cleared.

10. **Log on to the system controller Platform Shell.**

    The system does not prompt you for a password.

11. **Set the new Platform password.**

    For instructions, refer to "Define Platform Password" on page 16.

Verifying Hardening Results

After performing the procedures in this article to harden the SC and MSP, test the resulting configuration to verify that it is configured properly.

For the example configuration, our testing resulted in the following:

- TCP IPv4 services listed by `netstat` went from 31 to 1
- UDP IPv4 services listed by `netstat` went from 57 to 0

By reducing the number of services available, we reduced exposure points significantly.

Note – Earlier we recommended that you disable the SC failover mechanism before hardening the SCs. Re-enable failover only after you harden and test the entire configuration.

Verifying SC Hardening

After hardening the SC, review the settings to make sure that all the recommendations in “Securing the System Controller” on page 13 are in place.

Verifying MSP Hardening

After hardening the MSP, perform the following procedure to verify changes.

▼ To Verify MSP Hardening Results

1. **Reboot the MSP.**
2. **Validate that the number of daemons and services running on the MSP are significantly lower than before hardening.**

   Enable failover only after you harden and test the MSP.

After the MSP is hardened, the only services running in our sample configuration are as follows:

```
# netstat -a

UDP: IPv4
   Local Address         Remote Address     State
-------------------- -------------------- -------
      *.*                                  Unbound

TCP: IPv4
Local Address Remote Address Swind Send-Q Rwind Recv-Q  State
-------------------------------------------------------------
*.*               *.*         0     0   24576   0     IDLE
*.22              *.*         0     0   24576   0   LISTEN
```

3. **Enable SC failover.**

About the Authors

Alex Noordergraaf

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 the security of Sun midframe and high-end servers. He is the co-founder of the very popular freeware Solaris™ Security Toolkit. Before joining 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 security for Sun Fire servers, Sun™ Cluster software, Sun Fire™ midframe servers, Sun™ Enterprise 10000 servers, N-tier environments, the Solaris Operating Environment, and the Solaris OE network settings. He co-authored the Sun BluePrints publication, *JumpStart™ Technology: Effective Use in the Solaris™ Operating Environment*.

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[SM]. 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.

Tony M. Benson

Tony Benson has over twenty years experience of developing software solutions in the areas of military, aerospace and financial applications. As a Staff Engineer in the Enterprise Server Products group of Sun Microsystems, he is developing system management solutions for the Enterprise Server Product line.

Prior to his role in the Enterprise Server Products group, he developed secure, distributed revenue collection systems for a worldwide base of customers in the transit industry.

Related Resources

Publications

- Deeths, David and Brunette, Glenn. “Using NTP to Control and Synchronize System Clocks - Part I: Introduction to NTP,” Sun BluePrints OnLine, July 2001. `http://sun.com/blueprints/0701/NTP.pdf`
- Deeths, David and Brunette, Glenn. “Using NTP to Control and Synchronize System Clocks - Part II: Basic NTP Administration and Architecture,” Sun BluePrints OnLine, August 2001. `http://sun.com/blueprints/0801/NTPpt2.pdf`
- Deeths, David and Brunette, Glenn. “Using NTP to Control and Synchronize System Clocks - Part III: NTP Monitoring and Troubleshooting,” Sun BluePrints OnLine, September 2001. `http://sun.com/blueprints/0901/NTPpt3.pdf`
- Howard, John and Noordergraaf, Alex. *JumpStart™ Technology: Effective Use in the Solaris™ Operating Environment* (ISBN 0-13-062154-4), The Official Sun Microsystems Resource Series, Prentice Hall, September 2001.
- Noordergraaf, Alex. “Building Secure N-Tier Environments,” Sun BluePrints OnLine, October 2000. `http://sun.com/blueprints/1000/ntier-security.pdf`

- Noordergraaf, Alex. “Solaris Operating Environment Minimization for Security: Updated for the Solaris 8 Operating Environment,” Sun BluePrints OnLine, November 2000.
  http://sun.com/blueprints/1100/minimize-updt1.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_conf_install-v03.pdf
- Noordergraaf, Alex and Brunette, Glenn. “The Solaris Security Toolkit - Internals: 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://sun.com/blueprints/0401/security-updt1.pdf
- Reid, Jason M and Watson, Keith. “Building and Deploying OpenSSH in the Solaris Operating Environment,” Sun BluePrints OnLine, July 2001.
  http://sun.com/blueprints/0701/openSSH.pdf
- SNMPv2usec information: RFC 1909 An Administrative Infrastructure for SNMPv2 and RFC 1910 User Based Security Model for SNMPv2.
- Sun Microsystems, Inc., *Sun Fire 6800/4810/4800/3800 Platform Administration Manual* (805-7373-11), Sun Microsystems, Inc., April, 2001.
- Sun Microsystems, Inc., *Sun Fire 6800/4810/4800/3800 System Controller Command Reference Manual* (650-960-1300), Sun Microsystems, Inc., April, 2001.
- Vasanthan Dasan, Alex Noordergraaf, and Lou Ordorica. “The Solaris Fingerprint Database - A Security Tool for Solaris Operating Environment Files,” Sun BluePrints OnLine, May2001.
  http://www.sun.com/solutions/blueprints/0501/Fingerprint.pdf
- Watson, Keith and Noordergraaf, Alex. “Solaris Operating Environment Network Settings for Security: Updated for the Solaris 8 Operating Environment,” Sun BluePrints OnLine, December 2000.
  http://sun.com/blueprints/1200/network-updt1.pdf

Web Sites

- Commercial versions of SSH are available from:

  `http://www.ssh.com`

  `http://www.fsecure.com`

- SNMPv3 information:

  `http://www.ibr.cs.tu-bs.de/ietf/snmpv3/`

- SunFire documentation is available from:

  `http://www.sun.com/midframe`

- The Solaris Security Toolkit software is available from:

  `http://www.sun.com/security/jass`

- Cisco Systems (`http://www.cisco.com`) and Perle (`http://www.perle.com`)
- SUNSOLVE ONLINE Web site: `http://sunsolve.sun.com`