SUN BLUEPRINTS APPROVED

Securing the Sun Fire™ Midframe System Controller

By Alex Noordergraaf - Enterprise Engineering and Tony M. Benson - Enterprise Server Products

Sun BluePrints™ OnLine - September 2001

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.: 816-1942-10
Revision 1.2, 10/02/01
Edition: September, 2001

Please
Recycle

Adobe PostScript™

Securing the Sun Fire™ Midframe System Controller

The System Controller (SC) of a Sun Fire™ Midframe system, controls the assignment of resources within the Sun Fire frame, or platform. This includes which domains are on or off, and which components (such as CPUs, IO cards, and memory) are associated with domains. All of the server's configuration is stored on the System Controller. The security of the System Controller is critical to the overall integrity of the entire Sun Fire platform.

This article provides recommendations on how to securely deploy the Sun Fire System Controller (SC). These recommendations apply to environments concerned with security and particularly those where the uptime requirements of the SC and/ or the information on the Sun Fire server is critical to the organization.

There are a variety of issues involved in securing the Sun Fire SC. The most significant is its use of insecure administrative protocols. In addition, it is also sensitive to a variety of network-based attacks such as Denial of Service (DoS). This Sun BluePrints™ OnLine article provides specific recommendations on how to secure SC. The major components of these recommendations are:

- Creating a private SC network
- Using a dedicated server to control access to the private SC network
- Securing this dedicated server with the Solaris™ Security Toolkit software
- Using a network terminal server supporting SSH
- Configuring the SC for maximum security

System Controller (SC) Overview

The Sun Fire SC is an embedded, Real Time Operating System (RTOS) based system that is built into the Sun Fire frame. It 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 is used to store the RTOS while the other contains the SC application itself.

Additional information on the SC can be found in the *Sun Fire 6800/4810/4800/3800 Platform Administration Manual* and the *Sun Fire 6800/4810/4800/3800 System Controller Command Reference Manual*. Refer to the References section at the end of this document for the URLs of these documents.

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. In addition, if these security recommendations are not implemented, the SC is an extremely easy candidate for network based attacks such as the previously mentioned Denial of Service (DoS) attack or session sniffing and/or hijacking.

Given that only one password, belonging to the platform administrator, is needed to effectively control the machine, it is critical that the insecure protocols required to manage the SC be limited to a private and highly-secured network. To limit these protocols to one network segment, a gateway system is needed to provide an access and control point. This gateway system would have at least two network interfaces. One interface would connect 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 (e.g., SSH, IPsec, SNMPv2usec) can be installed. Administrators could then log into the MSP using the encrypted protocols. The insecure and non-encrypted protocols would only be used then on the private SC network. If the private SC network is built on physically separate network devices (i.e., no VLANs) there is little exposure for network sniffing or other network based attacks.

In this way, the SC can still be managed remotely, but the passwords and access information that would allow a hostile user to take over the platform, are not transmitted clear-text across a public network. These recommendations for the placement are built on top of the recommendations made in the Sun BluePrints OnLine article titled *Building Secure N-Tier Environments (October 2000)*. Refer to the Bibliography for the URL of this article.

Midframe Service Processor

The Midframe Service Processor (MSP) is responsible for providing a variety of services to the Sun Fire SC including, but not limited to:

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

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

Hardware Requirements

Specific recommendations of the hardware requirements cannot be made because they depend extensively on the number of SC’s being supported by the MSP, in addition to the software being run on the MSP. For example, if the MSP is only running 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 will be running additional monitoring and management software for several hundred SCs, then a significantly larger server is required.

The minimum hardware recommended for an MSP is listed below:

- `sun4u` architecture
- 8 GB disk
- 128 MB RAM
- CD-ROM drive
- SunSwift™ card or ideally a QuadFast Ethernet card
- Solaris™ 8 Operating Environment (Solairs OE)

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

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

This does not mean that additional software cannot be installed 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 as it controls access and the flow of information to and from the SC. The MSP should be managed based on the specific 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 be recommended to use the MSP as a general purpose Web server. In addition, the potential security impact of additional installed software should be evaluated to validate that the overall security of the MSP is not adversely affected.

The most secure MSP has the least software installed in addition to the fewest services and administrator accounts. The more secure the MSP, the better protected the Sun Fire SC will be.

Mapping of MSP to SC

Depending on the architecture of an environment it may be desirable to support several SC's from one MSP. This is recommended, from a security perspective, so 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 datacenter. In this situation one MSP, or pair of MSPs, would be appropriate to manage as many of these Sun Fire database servers as needed. Alternatively, this administrative domain must not include the Internet-accessible Web servers that access the database servers. The Web servers, as they are exposed to a significantly greater risk of misuse, are in a different administrative domain and should be managed by a separate MSP.

Network Topology

The sample network topology discussed in this section involves one Sun Fire 6800 server, two SC's and one MSP. Other architectures should be extrapolated from this basic design. The systems in this topology are as follows:

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

FIGURE 1 is a logical diagram and does not include all of the components required to actually make this environment function. Specifically, the network switches required are not discussed. It is recommended that separate network switches be used for the private SC network and not VLANs on a larger switch. Whatever switch is used for the private SC network, it should be managed and, more importantly, monitored as all other switches are in the environment.

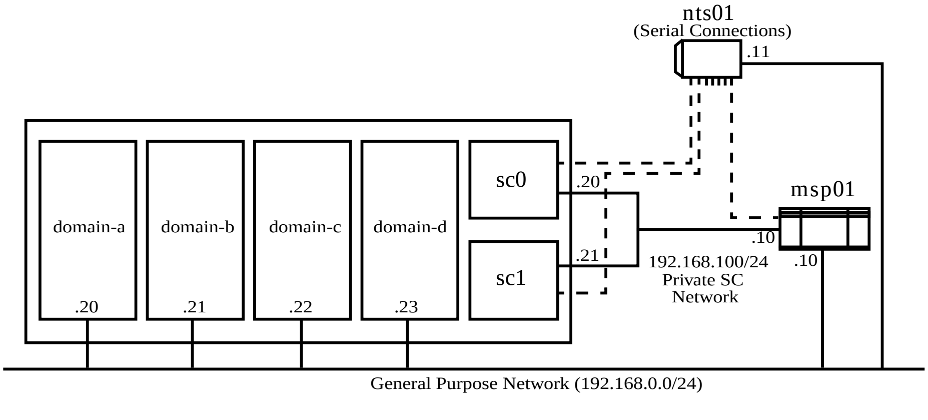

FIGURE 1 Sample SC Network Topology

The above network diagram illustrates the separate networks used to isolate the SC from general network traffic. In the example the general network, or 192.168.0/24 is not routed to the private SC network at 192.168.100/24 as IP Forwarding is disabled on the MSP.

Two access mechanisms are available to connect to the SC in this network architecture. First, an administrator can SSH to the MSP, or `msp01` in the diagram, and then TELNET from it to the SC. Secondly, the serial connection accessible from the network terminal server, or `nts01` in the diagram, can be used as an alternative access mechanism to the SC. In this topology even if the MSP is not available the SC is still accessible through the network terminal server.

The configuration of the MSP will be discussed in greater detail in the *MSP Security* section below. The security options in the SC will be discussed immediately after the *MSP Security* section.

Serial Port Access to SC

It is strongly recommended that a terminal server be used which supports the use of SSH to encrypt the session. This is strongly recommended because the terminal server is not on the private SC network, but on the general purpose intranet. Correspondingly, if TELNET is used to access the terminal server, then all passwords will be passed over the general purpose network, in clear text. This will undo many of the security measures designed into this architecture. Terminal servers supporting SSH are available from Cisco (`http://www.cisco.com`) and Perle (`http://www.perle.com`)

Control-A and Control-X

There are special commands that 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 when entered at the serial port. If entered within the first 30 seconds after an SC reboot, the Control-X key sequence performs a soft reboot of the SC. This soft reboot is similar to the issuance of a reset from the OpenBoot™ PROM on the Ultra Enterprise™ servers. The Control-A key sequence creates a RTOS shell.

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

The special capabilities of these key sequences are disabled 30 seconds after the Sun copyright message is printed. Once the capability is disabled, Control-A and Control-X operate as normal control keys with no special privileges.

The security of the SC could be compromised by unauthorized access to the RTOS shell. Correspondingly, access to the serial ports of the SC should be carefully controlled.

Appendix A contains a procedure, documented in the README file contained in patch `800054-01`, on how to use the Control-A and Control-X commands to reset the platform administrators password.

Write protect jumper

The SC contains several 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. When changing the setting to the write-protect setting, it is recommended that the jumper be left, on the board, but only plugged into one of the pins on the jumper to avoid misplacing the jumper.

In the write-enable position, the RTOS image may be updated using the `flashupdate` command, as described in the *Sun Fire 6800/4810/4800/3800 Platform Administration Manual*. In order to change the position of the write-protect jumper, the SC must be removed from the chassis.

If the RTOS write protect jumper is moved to the write-protect position, the following features are disabled:

- Attempts to `flashupdate` the RTOS image.
- The ability to use the keyboard commands, Control-A and Control-X during the first 30 seconds after an SC reboot.

Note – Removal of the SC should be carried out by qualified personnel to avoid the risk of damage to the SC or chassis. During removal and re-insertion of the SC, there is a risk of damage to the SC hardware and the chassis. To minimize this risk, and corresponding system downtime, it is required that only appropriately trained personnel perform this procedure. The procedure for removal and replacement of the SC is documented in the *Sun Fire 6800/4810/4800/3800 Platform Administration Manual*.

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

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. In configurations with a single SC, this results in platform downtime. For this reason, it is recommended that the platform be configured with a redundant SC to minimize Sun Fire frame downtime.

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

Spacebar

If the space bar is pressed while connecting through the network terminal server to the serial port of the SC, during the Power On Self Test (POST) process, the system enters an interactive mode called SCPOST. In this mode the user has a variety of commands and options available. No password is required to enter this mode.

Two of the 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 user (knowledgeable of SC memory addresses) accesses the interactive SCPOST facility, the SC platform and/or domain passwords could be modified.

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

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, in addition to the 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 these risks.

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

Additional redundancy can also 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, just not powered on. This minimizes most of the downtime associated with fixing the primary system as a replacement system is already configured and available and just needs to be powered on once the failed system has been 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. A detailed discussion of how this type of a configuration could impact the security posture of the SC is beyond the scope of this article.

MSP Security

The MSP is the gateway between general purpose internal networks and the private SC network. As such, it controls access between the general purpose networks and the private SC network. In order to effectively protect itself against unauthorized access, it must be configured securely; specifically, it must be appropriately hardened and have encrypted access mechanisms installed.

Note – The process described in this section is based on an interactive Solaris OE installation and not a Solaris JumpStart™ installation. Similar tasks, using the Solaris Security Toolkit software (e.g., `jass`) can also be performed in a JumpStart environment.

MSP Performance and Software Requirements

The performance and storage requirements for the MSP, depend on many variables. The configuration discussed in this article has the following software installed:

- Solaris 8 OE installed with the End User Cluster
- Latest patch cluster from SunSolve^{SM} Online Web site
- OpenSSH

Based on these requirements a low-end `sun4u` system such as a Netra T1, Ultra 1, or Ultra 5 systems, has the required performance. As with any system installation, the latest Security and Recommended Patch Cluster, available from the SunSolve Online Web site, should be installed on the MSP as it is being built.

Note – The MSP can be built either through an interactive CD-based or Solaris JumpStart installation. The Solaris Security Toolkit software can be used in either type of installation. Refer to *The Solaris Security Toolkit - Quick Start: updated for version 0.3 (June 2001)* article (referenced in the Bibliography).

The recommended Solaris OE cluster is End User. While it would be possible to install the MSP with significantly fewer Solaris OE packages this is not a supported configuration.

OpenSSH Installation

Administrator access to the SC through TELNET sessions and platform/ administrator shells must be encrypted. This requirement, for secured environments, is one of the major reasons for the presence of the MSP. The most commonly used mechanism to encrypt administrator traffic is SSH, as implemented by either freeware OpenSSH or commercial SSH products.

A Sun BluePrints OnLine article discussing how to compile and deploy OpenSSH titled: *Building and Deploying OpenSSH on the Solaris Operating Environment (July 2001)* is available at:

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

Information on where to obtain the commercial versions of SSH is provided in the References section.

Apache Installation

The Apache Web server is used, by the SC, to perform Solaris™ Web Start Flash updates of the SC EEPROMs, in addition to providing `restoreconfig` with a transport mechanism to restore to SC backups created with `dumpconfig`. The MSP is built using the Solaris OE End User cluster. The Apache distribution available in Solaris 8 OE is not installed with this cluster. So, it is necessary to manually install the three Apache packages required. The three required Solaris 8 OE Apache packages are as follows:

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

They can be found on any Solaris 8 OE 2 of 2 CD dated 4/01 in the following directory:

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

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

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

This `tar` file can then be moved to the MSP, extracted, and installed with the following commands:

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

Answer yes to all the questions asked. Once the installation has completed the `pkginfo | grep Apache` command should list the three Apache packages.

Next an appropriate user and group ID must be created for Apache to run as. First create a new group by adding the following line to the `/etc/group` file:

```
mspstaff::15:
```

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

Create a user account for the Apache daemon; this example uses `msphttp`:

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

Note – Administrators who are going to need access to files shared by Apache must be added to the `mspstaff` group by adding their user IDs to the end of the `mspstaff` entry in the `/etc/group` file.

Before starting the Apache daemon, it must be configured. Only a few steps are required to do that. First, create an `httpd.conf` file using the following command:

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

Next, open the `/etc/apache/httpd.conf` file in an editor and search for the following line:

```
#Listen 12.34.56.78:80
```

Add the following line immediately after it—where the IP address used, is the IP address of the MSP on the private SC network:

```
Listen 192.168.100.10:80
```

This will configure Apache to only respond to connection requests from the private SC network. Apache will not provide an HTTP services to the general purpose network. This is important as other systems must not be able to access the information which will be made available, over HTTP, to the SC.

A few other Apache configuration modifications are still required. Next, the Apache server must be told what name to use. Since the name of the MSP on the private SC network may not be resolvable, this configuration uses the IP address of that interface. Search for the following line in the `/etc/apache/httpd.conf` file:

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

Add the following line immediately after it—where the IP address used, is the IP address of the MSP on the private SC network:

```
ServerName 192.168.100.10
```

Also, the Apache server must be told what directory structure to make available. This is called the `DocumentRoot` and should be the top-most directory of where the Flash archives and backup files will be kept. Search for the following line in the `/etc/apache/httpd.conf` file:

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

Add the following line immediately after it—where the directory used is the topmost directory of what will be made 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 those 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, a user ID and group ID were created for this purpose. They were `msphttp` and `mspstaff`, respectively. Now that Apache is installed, it can be configured to use that user ID and group ID by making the following change in the `httpd.conf` file:

```
User msphttp
Group mspstaff
```

To allow this configuration to work, change the ownerships of the Apache log file directory with the following command:

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

Create the `/msp` directory on the MSP; use a partition with adequate free space. In the following example, the directory was created on the `/`, or root, filesystem of `msp01`:

```
# mkdir /msp
```

Next, the ownerships and permissions of the /msp directory must be set to the msphttp user ID and mspstaff group ID with the following commands:

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

Now the Web server can be started with the following command:

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

The Apache Web server is now ready to function as a restoreconfig server.

MSP Hardening

At this point, the MSP has had Solaris 8 OE End User cluster installed, been patched with the latest Security and Recommended Patch Cluster from SunSolve Online Web site, either a freeware or commercial version of SSH installed, and had the Apache Web server installed and configured. The next step for the MSP is for it to be hardened. This hardening is critical to the security of the SC as the default configuration of Solaris OE will not provide the required protection for the MSP.

This article focuses on hardening, or configuring the Solaris OE for maximal security. Minimization, or the removal of non-essential Solaris OE components, will not be discussed in this article.

The recommended Solaris OE installation, used for the MSP is the End User Cluster, and not Developer, Entire, or OEM installation clusters. This significantly reduces the number of Solaris OE packages installed on the MSP.

The Solaris Security Toolkit software, or jass, will be used to secure the MSP. The Toolkit implements the recommendations made in the Sun BluePrints OnLine security articles. These recommendations are documented in these previously published Sun BluePrints OnLine articles:

- *Solaris Operating Environment Security: Updated for the Solaris 8 Operating Environment (April 2001)*
- *Solaris Operating Environment Network Settings for Security: Updated for Solaris 8 Operating Environment (December 2000)*
- *The Solaris Security Toolkit - Installation, Configuration, and Usage Guide: Updated for version 0.3 (June 2001)*

All of these articles are referenced in the Bibliography, and are available from the Sun BluePrints OnLine Web site at:

```
http://www.sun.com/blueprints/browsesubject.html#security
```

The Solaris Security Toolkit Software

The Solaris Security Toolkit software, also known as `jass`, provides a flexible and extensible mechanism to minimize, harden, and secure Solaris OE systems. The primary goal behind the development of this toolkit is to simplify and automate the process of securing Solaris OE systems.

The Solaris Security Toolkit software will be used to automate the security modifications to the MSP. This Toolkit is freely available from:

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

Documentation on the Toolkit is included in the Toolkit `Documentation` directory or from the Sun Web site listed above. An MSP specific driver is included in the Toolkit distribution (version 0.3.1) to perform the hardening tasks described in this section. This driver, `sunfire_mf_msp-secure.driver`, creates a secured and supported configuration of the MSP based on its Solaris 8 OE installation. While the final configuration is supported, the Solaris Security Toolkit software itself, is not a supported Sun product.

The goal of the hardening is to disable all unnecessary Solaris OE services and enable all off-by-default optional Solaris OE security features. None of the standard Solaris OE services are required—not even SNMP. This provides for an extremely secure Solaris OE configuration, as only SSH is available on the general network interface, after the Toolkit run. If SSH was not installed before the Toolkit run, no services will be available on any MSP interface and only the serial console will be available as a login point.

The actual hardening process using `jass` is detailed below.

Solaris Security Toolkit Installation

First, the Solaris Security Toolkit software must be downloaded and installed on the MSP.

The instructions included use filenames which are only correct for this release of the Toolkit. Use the following procedure to download and install the Toolkit:

1. **Download the source file** (`SUNWjass-0.3.1.pkg.Z`).

   The source file is located at:

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

2. **Uncompress the package on the server using the** uncompress **command as shown**:

```
# uncompress SUNWjass-0.3.1.pkg.Z
```

3. **Install the Toolkit onto the server using the** pkgadd **command as shown:**

```
# pkgadd -d SUNWjass-0.3.1.pkg SUNWjass
```

Executing this command creates the SUNWjass directory in /opt. This subdirectory will contain all the Toolkit directories and associated files. The script make-pkg, included in version 0.3.1 of the Toolkit administrators can create custom packages using a different installation directory.

Recommended and Security Patch Installation

The Solaris Security Toolkit software will be used to install the most recent *Recommended and Security Patch* cluster available from the SunSolve Online Web site. To install these patches with the Toolkit, they must be downloaded and stored uncompressed in the /opt/SUNWjass/Patches directory on the MSP.

Downloading the Solaris OE Recommended and Security Patch cluster does not require a SunSolve support contract. To download the latest cluster, go to the SunSolve Online Web site at http://sunsolve.sun.com and click on the "Patches" link which is on the top of the left navigation bar.

Next, select the appropriate Solaris OE version in the "Recommended Solaris Patch Clusters" box. In this example, Solaris 8 OE will be used. Once the appropriate Solaris OE version is selected, select the best download option, either HTTP or FTP, with the associated radio button and then click on the "Go" button.

This should bring up a “Save As” window on your browser. Save the file locally in preparation to uploading it to the MSP.

Once downloaded, move the file securely to the MSP with `scp`, or `ftp` if `scp` is not available. The `scp` command used should appear similar to the following:

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

Next, the file must be moved to the `/opt/SUNWjass/Patches` directory and uncompressed. The following commands perform those tasks:

```
# 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
[. . .]
```

Note – If the *Recommended and Security Patches* are not loaded into the appropriate directory, an error will be produced during the execution of the Toolkit.

Solaris Security Toolkit Execution

The Solaris Security Toolkit version 0.3.1 has a pre-built driver for hardening an MSP which is called `sunfire_mf_msp-secure.driver`. This driver should be run on the MSP as follows:

```
# pwd
/opt/SUNWjass
# ./jass-execute -d sunfire_mf_msp-secure.driver

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

============================================================
JASS Version:  0.3.1
Node name:     baked
Host ID:       808cf880
Host address:  192.168.100.10
MAC address:   8:0:20:8c:f8:80
Date:          Mon Jul 16 13:54:49 PDT 2001
============================================================
[...]
```

The Toolkit will perform approximately one hundred different security modifications to the MSP.

Note – The actions performed by each of the scripts is described in *The Solaris Security Toolkit - Internals: updated for version 0.3 (June 2001)* Sun BluePrints OnLine article. The Toolkit hardening described is performed in standalone, not JumpStart mode, as the MSP was built using an interactive Solaris OE installation. For details on the differences between standalone and JumpStart installation modes refer to the Toolkit documentation.

Once the MSP is hardened, has the appropriate version of SSH installed, and has been rebooted, the only services running are listed below:

```
# 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
```

The Ultra 1/200E server (running Solaris 8 OE 4/01 release) on which these recommendations were tested, the number of TCP IPv4 services listed by `netstat` went from 31, prior to the Toolkit run, to one. Similarly, the number of UDP IPv4 services listed by `netstat` went from 57 to zero. By reducing the number of services available, the exposure points of this system are reduced significantly.

MSP SYSLOG Configuration

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

Each SYSLOG selector specifies the facility (e.g., `kern`, `daemon`, `auth`, `user`, etc.) and level at which a message was logged. There are five available levels ranging from most serious (`emerg`) to the least serious (`debug`). The facility is used to group log messages together by subsystem. For instance, all kernel messages are grouped together through the facility `kern`. The facilities available include, but are not limited to, the following:

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

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

It is also possible to substitute a wildcard (*) for the facility name in the syslog.conf file. This is particularly useful when all messages (i.e., *.debug), or all messages at one level or higher must be logged (i.e., *.kern).

Each SYSLOG message also 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 is used to indicate a message contains debugging information and may not be particularly important. The complete list of levels available in the syslog.conf includes, but is not limited to, the following:

- emerg
- crit
- err
- notice
- debug

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

While a wildcard can be used to define a facility, it cannot be used to define a level. Hence, the entry *.debug is acceptable; however, the corresponding entry of auth.* is not correct and must not be used.

For the MSP, the recommended configuration is to have all SYSLOG traffic, from the SC stored locally in both the standard file /var/adm/messages, in addition to a separate Sun Fire only file.

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

The recommended syslog.conf should look something 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 critical kernel messages are also displayed on the console.

If an automated log parsing tool such as logcheck or swatch is to be 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 the following lines should be added to those listed above:

```
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.

SC Application Security Settings

While configuring the platform and domains of the SC, steps must be taken to configure it securely. Some of these tasks are performed as the platform administrator, while others are performed as the appropriate domain administrator.

These security modifications should be implemented 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. At the time of writing, the most current SC firmware patch is 111346-02. Always use the most recent update available from SunSolve Online Web site at:

```
    http://sunsolve.sun.com
```

This article focuses on those SC configuration changes required to secure the SC. Normal administrative issues may be discussed if they are impacted by a security modification. For full details on configuration of the SC, refer to the various System Controller manuals listed in the Bibliography.

Platform Administrator

The first steps taken to secure the SC are as follows:

- Configure network settings
- Configure Loghost
- Set SNMP community strings
- Set Access Control Lists (ACLs) for hardware
- Set platform password
- Set passwords for platform and domain shells
- Set SNMP domains

Most of these operations are performed through the `setupplatform` command. This command should be run either in an interactive mode where it asks specific questions or by specifying the configuration modification required. For the purposes of this discussion, the command is run in the latter mode by using the `-p` option.

Network Settings

The first step in setting up an SC is to enable networking. This defines whether the system uses dynamic IP address assigned through DHCP, what its hostname will be, its IP address, DNS server, and other network specific information. In this secured topology static IP addresses are used. 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 setup and manage the DHCP server is only appropriate if there are many SCs to be configured. If DHCP is used, the DHCP server should be configured to only provide services for the private SC network and no other network segments.

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

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

The following command was used to enter these 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.
```

Configuring Platform Loghost

The second setup involved in configuring the SC is defining the 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 on to a central location for storage, reconciliation, and review for unusual activity. When defining the Loghost, care must be taken to define it through the use of IP addresses if DNS is not being used. In this example, DNS is not being used, so the IP address is entered.

In addition to specifying the name/IP address of the Loghost, the facility level included in the SYSLOG messages, can also be specified. The SYSLOG protocol provides eight user-defined facility levels: `local0` through `local7`, in addition to the eighteen system defined facilities. 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 will be the SYSLOG server, so its IP address should be entered in the following manner with the corresponding SYSLOG facility level (`local0`) for the platform:

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

Loghosts
--------
Loghost [oslab-mon]: 192.168.100.10:local0
```

Details on how to configure the SYSLOG service on the MSP were provided in the *MSP Security* section above.

Note – There is a bug in the `showplatform` and `showdomain` commands. In this bug, which has bugid 4421267, a previously entered facility value for a define Loghost is not displayed by the `showplatform` and `showdomain` commands when validating that one has been entered correctly. For example - if the above example was followed and a `showplatform -p loghost` command issued, the default value listed would be 192.168.100.10 and not the correct value of 192.168.100.10:local0. The fix for this is incorporated in SCapp version 5.12.5.

Setting Platform Password

The next step 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 MD5 to generate a hash of the password entered. Correspondingly, all characters entered are significant. A recommended password length is at least 16 characters. Passwords should be comprised of at least lowercase, uppercase, numeric, and punctuation marks. The following command is used to set the platform shell password:

```
ds7-sc0:SC> password

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

A minimum password length of 16 characters is recommended to promote the use of pass-phrases instead of passwords. Given the capabilities of current systems to either brute-force or guess encrypted passwords, an 8 character length string is no longer, and has not been for some time, secure. Given that the SC supports the use of longer passwords, their use is strongly recommended.

Note – If the platform administrator’s password is lost, there is a documented procedure on clearing the password. This procedure is described in Appendix A of this article.

Defining Domain Passwords

A domain shell is always present for a domain, whether or not any hardware is actually defined for that domain. Because of this, and to avoid potential unauthorized reallocation of hardware to an unused domain, all domain shells should have passwords defined. The passwords for each domain should be different from each other, the platform shell, and the Solaris OE images running on the domains. A robust mechanism of password management is recommended to track all of these passwords.

Note – All domain shells should have passwords set—regardless of whether or not they are used and have hardware assigned.

A domain’s password can be set either from the shell of that specific domain or the platform shell with the `password` command. The below example sets the domain passwords from the platform shell. A domain password has the same restrictions as the platform password—which are, in effect, none. As with the platform password, a minimum password length should be 16 mixed-case alphanumeric characters. The following command was used to set the password, from the platform shell, of `domain-a`:

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

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

The same command, with the appropriate domain name, would be used to set the passwords for domains `b` through `d`.

Note – If a password has already been defined for either a platform or domain shell, the password command requires its entry before allowing a new password to be entered. The platform administrator cannot, without knowing the old password, reset a domain password.

Note – The only supported mechanism by which domain passwords can be forcibly reset is the `setdefaults` command. This command resets the SCs configuration back to factory defaults. All changes made to the SC, since it was shipped from the factory will be lost including all settings described in this article. This command should be used with care.

SNMP Configuration

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, suffered from security issues as they didn't address issues such as authentication, data integrity checks, and encryption. Updated versions of the protocol have been proposed, such as SNMPv2usec and SNMPv3, but have not been fully approved by the organization that controls these standards, the IETF. Additional references to SNMPv2usec and SNMPv3 information can be found in the *References* section. 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 compatability.

The Sun Fire SC only supports the use of SNMPv1. Due to this limitation there are two possible recommendations.

The first alternative is for those customers who wish to use Sun Management Center (Sun MC)3.0 software to manage and maintain their SunFire Midframe systems. To use Sun MC 3.0 securely it is recommended that, in addition to using its SNMPv2usec capabilities, all of its management traffic be isolated to a physically isolated and dedicated management network. This recommendation of isolating management traffic to a physically separate and highly protected network segment 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 SunFire Midframe SC. This software can be installed on either the SunMC server or a separate server. In either case the system on which the platform agent software is installed cannot be connected to the public intranet so as to limit access to the platform agent software which is why the software should not be installed on the MSP. If isolating the Sun MC server to completely separate and isolated networks is not possible, then the platform agent software should be installed on a separate system. This server would require at least two network interfaces. One would connect to the private SC network while the other would connect 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. Any additional server used, and the Sun MC server, should be appropriately hardened and secured.

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

The SNMP daemon on the SC is disabled in the following manner:

```
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.
```

Setting Access Control Lists (ACLs) for Hardware

The next step is to define the ACLs for each domain. Obviously, this step is only important if the Sun Fire server will have multiple domains and their resources are restricted in some way. Only if these conditions are met should ACLs be implemented. By default, all hardware present in the system is accessible to all domains. In this example a Sun Fire Midframe 6800 server is divided into three domains—where each domain will have one CPU and I/O board.

The platform administrator shell should be used to assign the different CPU and I/O boards into the appropriate domain. ACLs only apply to the domain shells and not the platform shell. The platform shell’s ability to assign and reassign hardware components is not restricted by ACLs. It is recommended that the platform administrator account be used only to initially assign hardware components to the appropriate domain. Once 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 these recommendations.

First, determine what boards are present with the following command:

```
ds7-sc0:SC> showboard

Slot    Pwr Component Type               State     Status
----     --  -------------                ----     -----
/N0/SB0  On  CPU Board                   Available Passed
/N0/SB2  On  CPU Board                   Available Passed
/N0/SB3  On  CPU Board                   Available Passed
/N0/IB6  On  PCI I/O Board               Available Passed
/N0/IB7  On  PCI I/O Board               Available Passed
/N0/IB8  On  PCI I/O Board               Available Passed
```

Next, assign these resources to the appropriate domains with the following commands:

```
ds7-sc0:SC> addboard -d a /N0/SB0 /N0/IB6
ds7-sc0:SC> addboard -d b /N0/SB2 /N0/IB8
ds7-sc0:SC> addboard -d c /N0/SB3 /N0/IB7
```

The `addboard` command now produces 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
```

There are now three domains, `a` through `c`, defined on this Sun Fire server each with one CPU and I/O board.

Rebooting System Controller

If needed, the SC should be rebooted at this time. The SC only has to be rebooted if a console message was generated to that effect. If in doubt, the SC should be rebooted to ensure the changes take effect.

A message similar to the following would have been displayed if the SC must be rebooted:

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

An SC is rebooted with the following command from the platform shell:

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

Once rebooted the `showplatform` command can be run to validate that all the modifications have taken effect.

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

Domain Administrator

Once all of the platform shell configuration modifications have been performed, the domain-specific configuration modifications can be implemented. Most of the recommended changes are performed in the platform shell. Only a few domain-specific changes are required in the domain shells. These modifications include defining the following:

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

Each of these must be defined individually for each domain.

The samples below only make these changes on one domain; specifically, all these changes are performed on `domain-a`. Before attempting to execute the command below, first log into the appropriate domain shell.

Setting the Loghost

Similarly to the *Configuring Platform Loghost* configuration option described above; a Loghost must be defined for each of the domains individually. In addition, a facility unique to the frame should also be used. By having separate definitions of Loghost for each domain and platform shells, separate SYSLOG servers can be used to collect this information. In this secured network environment, there is only one system available to collect and parse the SYSLOG data—the MSP. The use of the facility option helps differentiate SYSLOG messages coming from the four different domains and platform shells.

The following command is used to set the `domain-a` shell Loghost to be the MSP:

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

Loghosts
--------
Loghost [0.0.0.0]: 192.168.100.10:local1
```

In this 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` should use `local2`, `domain-c` `local3`, and `domain-d` `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 is forwarded. The Solaris OE SYSLOG server should be defined, as normal, in the `/etc/syslog.conf` configuration file of the Solaris OE.

Details on how to configure the SYSLOG service on the MSP are provided in the *MSP Security* section below.

Setting the Domain SNMP Information

Each domain has unique SNMP configurations which must be configured separately. Some of the domain SNMP information may be the same (i.e., 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. These two community strings provide the mechanism by which individual domains are accessed.

In this secured configuration the SNMP daemon has been disabled in the platform shell. Correspondingly, it is not necessary to set the Public and Private community strings as SNMP will not be used.

Domain Security Settings

This section discusses the security configuration options available within each domain.

The `setkeyswitch` Command

The `setkeyswitch` command provides functionality similar to the physical key setting on the Ultra Enterprise server line. As with the Ultra Enterprise systems, when the server is functioning, the `keyswitch` should be in the `secure` setting. With the Sun Fire servers, 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 used by an administrator who has domain shell access on the SC. If the administrator has that access then using `setkeyswitch` to change from `secure` to `on` is straightforward. Other administrators, without domain and/or platform access will not be able to perform this command. The following `setkeyswitch` command sets `domain-a` into secure mode:

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

Two other Sun Fire domain features are also disabled by the `setkeyswitch secure` option. When a domain is running in `secure` mode, it will ignore `break` and `reset` commands from the SC. This is not only an excellent precaution from a security perspective, but it will also ensure that an accidently issued `break`/`reset` will not halt a running domain.

Note – There is a bug in the currently released version of Solaris 8 OE running on Sun Fire domains that affects the behavior of the `setkeyswitch secure` mode. The bugid is 4417940. When a domain is in `secure` mode it will queue any `break` or `reset` commands sent to it. These `break` and `reset` commands are not processed until the domain is in `on` mode. Hence, if the domain is in `secure` mode, a break can be issued and the domain will ignore it. Sometime later, when `setkeyswitch` is used to set the domain in `on` mode, the domain will immediately halt. Depending on how much time separated the issuance of the `break` and the `setkeyswitch` modification, it may be extremely difficult to determine what happened. In addition, the domain will have suffered from unscheduled downtime. The fix for this bug has been integrated into Solaris 8 OE Update 6, which is scheduled for release in late 2001. Due to the nature of this bug, systems with high uptime requirements should not use the `setkeyswitch secure` option until they are running a Solaris 8 OE which incorporates the fix for this bug.

Other System Controller Security Issues

This section discusses how to securely backup and restore the SC, in addition to other SC security options. In this section, the MSP is used as the `dumpconfig`, `restoreconfig` and `flashupdate` server.

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 intended 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 set period of time. Passwords are generated internally by Sun on an as needed basis, and as such are not generally available.

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

`dumpconfig` and `restoreconfig`

The `dumpconfig` and `restoreconfig` commands are 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 `dumpconfig` command utilizes the FTP protocol to save the current platform and domain configurations to a server. In this case, the server is the MSP. The `restoreconfig` command utilizes either the FTP or HTTP protocol to restore a previously saved configuration to the SC from the server.

All stored platform and domain configuration information is included in the dump file. This includes the MD5 hash of the platform and domain administrator passwords, 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 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 constrained to the private network and minimizing the 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 the previous sections describing platform and domain password setup.

The MSP is configured to respond to HTTP, but does not normally respond to FTP, since the FTP service is disabled during MSP setup. Refer to the section *MSP Security*. In order to perform a dumpconfig, the FTP service needs to be enabled on the MSP. On satisfactory completion of the dumpconfig command, the FTP service should be disabled on the MSP. 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 dumpconfig and restoreconfig operations.

The Apache Web server on the MSP was 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. However, since 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 7pm would be stored into a directory called 2001071619.

In order to enable the FTP service on the MSP, first log on to the MSP using Secure Shell, and su to root. Edit the file /etc/inetd.conf, and uncomment the following FTP entry:

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

Having done this, 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
```

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 above, the following directory is created:

```
# mkdir /msp/2001071619
# chown msphttp:mspstaffmsphttp /msp/2001071619
# chmod 770 /msp/2001071619
```

At the SC, dump the configuration, using FTP with a user name and password. This should appear 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
```

When this 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
```

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
```

Confirm that the FTP service is disabled be executing the following commands:

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

When it is necessary to restore configuration settings, first ensure that the platform and domain administration passwords contained in the chosen dump file are known by the platform and domain administrators. In order to avoid the necessity of enabling the FTP service on the MSP for this operation it is recommended that the restoreconfig operation be carried out using HTTP. As with the dumpconfig operation, a user ID and password will be used for this operation, and the user ID is only used for dumpconfig and restoreconfig operations.

flashupdate

The flashupdate feature is used to update 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 may be on a server or another board of the same type. This section refers to updates executed from an image on a server. The MSP is used as the server for flashupdate images.

In order to avoid the necessity of enabling FTP on the MSP for this operation, it is recommended that the `flashupdate` operation be carried out 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.

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.

Only use the `flashupdate` command on the RTOS if you need to. If a RTOS flash fails, then a service call to Sun will be needed 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.

Download the latest `flashupdate` for the SC from the Product Patches section of the SunSolve Online Web site. 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
```

Confirm that the checksum of the file matches the checksum listed on the SunSolve Online Web site with 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 also 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 (May 2001)* available at:

`http://www.sun.com/blueprints/0501/Fingerprint.pdf`

Unpack the files containing the patch. These should be placed in a subdirectory under the Apache 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
```

The instructions in the file `Install.info` should be followed. In this example, `sc-app` and `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. 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
```

Note – The Solaris OE image running in each domain should have been halted gracefully, through a `shutdown` command, before issuing the `setkeyswitch off` command described above.

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.

Conclusion

This article described, and provided examples of, how to build a secure environment around the Sun Fire SC. Exposure points of the SC were described and recommendations were made on how they can be mitigated. The recommendations made in this article include building a separate and private SC network, to which the insecure protocols required to manage an SC are restricted. A system, referred to as the MSP was introduced as the secure gateway into this private SC network. A detailed, supported, and secured MSP configuration was described in detail and an automated implementation mechanism discussed.

Acknowledgements

The authors would like to thank everyone on the Sun MC team involved in crafting the *SNMP Configuration* section above including (in alphabetical order): Raju Alluri, Daniel J. Carroll, Ravi Chhabria, Andres Gomez-Rivas, Ade Hamza, Evert Hoogendoorn, and Govindarajan Rangarajan.

References

- Commercial versions of SSH are available from:

`http://www.ssh.com`

`http://www.fsecure.com`

- SNMPv2usec information

  RFC 1909 An Administrative Infrastructure for SNMPv2

  RFC 1910 User Based Security Model for SNMPv2

- 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`

Bibliography

- Noordergraaf, Alex, *Building JumpStart Architectures,* Sun BluePrints OnLine, April 2001, `http://sun.com/blueprints/0401/BuildInf.pdf`
- 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 - 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 Brunette, Glenn, *The Solaris Security Toolkit - Release Notes: Updated for version 0.3,* Sun BluePrints OnLine, June 2001, `http://sun.com/blueprints/0601/jass_release_notes-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`
- 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`

Appendix A: Recovering A Lost Platform Administrators Password

If the platform administrator's password is lost, the following procedure can be used to clear the password. This procedure was first documented in patch 800054-01.

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 (before the Control-A key is disabled), press Control-A. This will give you the RTOS prompt.

```
->
```

3. Make a note of the current boot flags settings. This will be used to restore the boot flags to the original value.

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

Save the `0x` number for step 9. below.

4. Change the boot flags to disable autoboot.

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

5. Reboot the System Controller by pressing Control-X. Once reset, it will stop at the RTOS prompt.

6. 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
```

7. The System Controller will output the following messages:

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

8. Wait until the above messages have been displayed.

9. Restore the bootflags to the original value using the `setBootFlags`() command. Use the value returned from step #3 above.

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

10. Reboot the System Controller by pressing Control-X.

Once rebooted, the platform administrator’s password will be cleared.

11. Log on to the System Controller Platform Shell. This will not prompt for a password. Set the new Platform password as described in the above section SC Application Security Settings.

Author's Bio: Alex Noordergraaf:

Alex Noordergraaf has more than nine years experience in the area of Computer and Network Security. As a Senior Security Architect in the Enterprise Engineering group of Sun Microsystems, he is developing, documenting, and publishing security best practices through the Sun BluePrints OnLine program. Articles completed include recommendations on Solaris OE Security settings, Solaris OE Minimization, and Solaris OE Network settings.

Prior to his role in Enterprise Engineering, 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. In addition to providing billable services to customers, he developed and delivered an Enterprise Security Assessment methodology and training curriculum to be used worldwide by SunPS. His customers have included major telecommunication firms, financial institutions, ISPs, and ASPs.

Author's Bio: Tony 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.