SUN BLUEPRINTS APPROVED

Securing the Sun Fire™ 12K and 15K System Controllers

Updated for SMS 1.2

Alex Noordergraaf and Dina K. Nimeh,
Enterprise Server Products

Sun BluePrints™ OnLine - July 2002

Sun microsystems

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.: 816-6694-10
Revision 1.1, 10/7/02
Edition: July 2002

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Please
Recycle

Adobe PostScript

Securing the Sun Fire™ 12K and 15K System Controllers

This article recommends how to securely deploy system controllers (SCs) on Sun Fire™ 12K and 15K systems. 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.

The SC controls the hardware components that comprise a Sun Fire 12K or 15K server. Because it is a central control point for the entire frame, it represents an attack point for intruders. To improve reliability, availability, and serviceability (RAS), the SC must be secured against malicious misuse and attack.

This article is one in a series that provides recommendations for enhancing the security of a Sun Fire system. After securing the SC, we recommend that you use the "Securing the Sun Fire 12 K and 15K Domain" article to secure the SC domains.

This article contains the following topics:

Background Information

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

- "Assumptions and Limitations" on page 2
- "Obtaining Support" on page 3
- "Understanding the System Controller" on page 4
- "Default SMS Configuration" on page 6
- "SC Network Interfaces" on page 13
- "Security Options in SMS 1.2" on page 16

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 SC, resulting in a supported configuration.

Our recommendations assume a platform based on Solaris 8™ Operating Environment (2/02) or Solaris 9 Operating Environment running System Management Services software version 1.2 on the SC. All of the Solaris Operating Environment (Solaris OE) components described in this article are included in these releases.

Note – The recommendations in this article are compatible with System Management Services (SMS) 1.1 on Solaris 8 (10/01), except where noted. The Solaris™ Security Toolkit software distinguishes which modifications are needed based on the version of SMS installed on your system.

Solaris Operating Environment (Solaris OE) hardening can be interpreted in many ways. For purposes of developing a hardened SC 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 here. The configuration remains supported in these cases; however, additional hardening beyond what is recommended in this article is not supported.

You can customize a copy of the Sun Fire 12K and 15K SC module of the Solaris Security Toolkit to disable certain hardening scripts. It is strongly recommended that any modifications to the default modules be made in copies of those files to simplify upgrades to newer Solaris Security Toolkit versions.

Solaris OE minimization (removing Solaris OE packages to minimize security exposures) is not a generally supported option on the SCs. However, it is possible to create a supported exception to this rule. If you are interested in minimizing the SC, contact your Sun account team for assistance.

Note – Standard security rules apply to the hardening of SCs: *That which is not specifically permitted is denied*.

In this article, we omit additional software that you can install on the SCs, such as Sun^{SM} Remote Services Event Monitoring, Sun^{SM} Remote Services Net Connect, and Sun™ Management Center software. We recommend that you carefully consider the security implications implicit with the installation of these types of software.

Obtaining Support

The SC configuration for Sun Fire systems implemented by the Solaris Security Toolkit module (`sunfire_15k_sc-secure.driver`) is a Sun supported configuration. A hardened SC is supported *only* if the security modifications are performed using the Solaris Security Toolkit. Support calls to Sun's support 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
```

Understanding the System Controller

Securing the system controller (SC) is the first priority in configuring Sun Fire systems to be resistant to unauthorized access and to function properly in hostile environments. Before securing the SC, it's important to understand the services and daemons that are running on the system. This section describes the software, services, and daemons specific to the SC. The functionality is described at a high-level with references to Sun documentation for more detailed information. This section provides administrators with a baseline of functionality required for the SC to perform properly.

The SC is a multi-function system board within the Sun Fire frame. This system is dedicated to running the SMS software. The SMS software is used to configure dynamic domains, provide console access to each domain, control whether a domain is powered on or off, and provide other functions critical to operating and monitoring Sun Fire systems.

The following list is an overview of the many services the SC provides for the Sun Fire systems:

- Manages the overall system configuration.
- Acts as a boot initiator for its domains.
- Serves as the `syslog` host for its domains; note that an SC can still be a `syslog` client of a LAN-wide `syslog` host.
- Provides a synchronized hardware clock source.
- Sets up and configures dynamic domains.
- Monitors system environmental information, such as power supply, fan, and temperature status.
- Hosts field-replaceable unit (FRU) logging data.
- Provides redundancy and automated SC failover in dual SC configurations.
- Provides a default name service for the domains based on virtual hostids, and MAC addresses for the domains.
- Provides administrative roles for frame management.

Redundant SCs

You can have up to two SCs within Sun Fire frames. Our security recommendations are the same for both SCs. The SC that controls the platform is referred to as the main SC, while the other SC acts as a backup and is called the spare SC. The software running on the SC monitors the SCs to determine when an automatic failover should be performed.

Note – For our sample configuration, the main SC is `sc0` and the spare SC is `sc1`. If no hardware failures are present and the SCs are booted at the same time, `sc0` always becomes the main SC.

We recommend that the two SCs have the same configuration. This duplication includes the Solaris OE, security modifications, patch installations, and all other system configurations.

The failover functionality between the SCs is controlled by the daemons running on the main and spare SCs. These daemons communicate across private communication paths built into the Sun Fire frames. Other than the communication of these daemons, there is no special trust relationship between the two SCs.

SMS Software

A significant aspect of SC security is access to applications that an administrator uses to manage Sun Fire systems. Some security issues associated with the SMS software are described in the *System Management Services (SMS) 1.2 Administrative Guide*. This article builds on the recommendations made in that guide.

Access to the SMS software is the core of the SC. Correspondingly, access to this software must be carefully controlled. Only authorized users should have access. The SMS software provides a mechanism, over and above the Solaris OE access controls, to limit access to the SMS software. These features are described in “Default SMS Configuration” on page 6.

Default SMS Configuration

This section describes the default SMS configuration installed on Sun Fire SCs.

Packages

A Sun Fire SC is based on Solaris 8 OE (10/01) for SMS 1.1, Solaris 8 OE (2/02) for SMS 1.2, or Solaris 9 OE for SMS 1.2, using the `SUNWCall` Solaris OE installation cluster.

The SMS software resides on the SC and oversees all SC operations. The entire SMS software bundle is comprised of the following packages, which are specific to the Sun Fire 12K and 15K SCs:

```
application SUNWSMSdf System Management Services Data Files
application SUNWSMSjh System Management Services On-Line Javahelp
application SUNWSMSlp System Management Services LPOST object files
application SUNWSMSmn System Management Services On-Line Manual Pages
application SUNWSMSob System Management Services OpenBoot PROM
application SUNWSMSod System Controller Open Boot Prom
application SUNWSMSop System Management Services Core Utilities
application SUNWSMSpd System Controller Power On Self Test
application SUNWSMSpo System Management Services POST Utilities
application SUNWSMSpp System Management Services picld(1M) Plug-in Module
application SUNWSMSr  System Management Services, (Root)
application SUNWSMSsu System Management Services User Environment
application SUNWufu   User Flash PROM Device Driver Header File
application SUNWufrx  User Flash PROM Device Driver (Root) (64-bit)
application SUNWscdvr Sun Fire 15000 System Controller drivers
```

Accounts and Security

The following users are added to the `/etc/passwd` file by the SMS software:

```
# grep sms /etc/passwd
sms-codd:x:10:2:SMS Capacity On Demand Daemon::
sms-dca:x:11:2:SMS Domain Configuration Agent::
sms-dsmd:x:12:2:SMS Domain Status Monitoring Daemon::
sms-dxs:x:13:2:SMS Domain Server::
sms-efe:x:14:2:SMS Event Front-End Daemon::
sms-esmd:x:15:2:SMS Environ. Status Monitoring Daemon::
sms-fomd:x:16:2:SMS Failover Management Daemon::
sms-frad:x:17:2:SMS FRU Access Daemon::
sms-osd:x:18:2:SMS OBP Service Daemon::
sms-pcd:x:19:2:SMS Platform Config. Database Daemon::
sms-tmd:x:20:2:SMS Task Management Daemon::
sms-svc:x:6:10:SMS Service User:/export/home/sms-svc:/bin/csh
```

Of these accounts, sms-svc is the only default account that administers the system. All the other accounts provide privileges for the daemons they are associated with. Never use these accounts to log into the system. You can secure them the same way as unused system accounts. These accounts are for the daemons running the SC as described in "Daemons" on page 9.

The following are newly added SMS /etc/shadow contents:

```
# grep sms /etc/shadow
sms-codd:NP:::::::
sms-dca:NP:::::::
sms-dsmd:NP:::::::
sms-dxs:NP:::::::
sms-efe:NP:::::::
sms-esmd:NP:::::::
sms-fomd:NP:::::::
sms-frad:NP:::::::
sms-osd:NP:::::::
sms-pcd:NP:::::::
sms-tmd:NP:::::::
sms-svc:lnrf21Ovf4G9s:11414::::::
```

All of these accounts, including the sms-svc account, are initially locked with "NP" as the encrypted password entry.

Caution – Set the password for the sms-svc user on both SCs immediately after installing the SMS software or first powering on the system.

The following entries are added to the `/etc/group` file by the SMS software:

```
# grep sms /etc/group
platadmn::15:sms-svc
platoper::16:sms-svc
platsvc ::17:sms-svc
dmnaadmn::18:sms-svc
dmnarcfg::19:sms-svc
dmnbadmn::20:sms-svc
dmnbrcfg::21:sms-svc
dmncadmn::22:sms-svc
dmncrcfg::23:sms-svc
dmndadmn::24:sms-svc
dmndrcfg::25:sms-svc
dmneadmn::26:sms-svc
dmnercfg::27:sms-svc
dmnfadmn::28:sms-svc
dmnfrcfg::29:sms-svc
dmngadmn::30:sms-svc
dmngrcfg::31:sms-svc
dmnhadmn::32:sms-svc
dmnhrcfg::33:sms-svc
dmniadmn::34:sms-svc
dmnircfg::35:sms-svc
dmnjadmn::36:sms-svc
dmnjrcfg::37:sms-svc
dmnkadmn::38:sms-svc
dmnkrcfg::39:sms-svc
dmnladmn::40:sms-svc
dmnlrcfg::41:sms-svc
dmnmadmn::42:sms-svc
dmnmrcfg::43:sms-svc
dmnnadmn::44:sms-svc
dmnnrcfg::45:sms-svc
dmnoadmn::46:sms-svc
dmnorcfg::47:sms-svc
dmnpadmn::48:sms-svc
dmnprcfg::49:sms-svc
dmnqadmn::50:sms-svc
dmnqrcfg::51:sms-svc
dmnradmn::52:sms-svc
dmnrrcfg::53:sms-svc
```

Groups provide the groundwork for delegation of domain and chassis administrative capabilities. They allow for separation of the administrative privileges and operator privileges for each domain and the entire frame. The *System Management Services (SMS) 1.2 Administrator Guide* contains detailed descriptions of which commands require a group’s privileges for executing.

Daemons

The SMS daemons are divided into the following three types, with sample `ps` output.

First are the platform or core SMS daemons run on both the main and spare SC:

```
root      8108   1  0 17:53:04 ?        0:01 mld
root      8123   1  0 17:53:05 ?       31:35 hwad
root      8126   1  0 17:53:05 ?        0:00 mand
sms-frad  331    1  0 12:41:21 ?        0:00 frad
root      8132   1  0 17:53:06 ?        0:03 fomd
root      4830   1  0 09:35:56 ?        0:00 ssd -i SMS software start-up initiated
-iSC POST results:  ’Power On Selftest n
```

Next are the SMS daemons that run only on the main SC:

```
sms-pcd   393     1  0 12:41:43 ?        0:03 pcd
sms-tmd   402     1  0 12:41:43 ?        0:00 tmd -t 12
sms-dsmd  405     1  0 12:41:44 ?        0:00 dsmd
sms-esmd  414     1  0 12:41:45 ?        0:05 esmd
sms-osd   419     1  0 12:41:46 ?        0:00 osd
root      8218    1  0 17:53:33 ?        0:00 kmd
sms-efe   475     1  0 12:41:47 ?        0:00 efe
sms-codd  483     1  0 12:41:48 ?        0:00 codd
```

Third are the SMS daemons that communicate to the domains, which run only on the main SC:

```
sms-dxs  4428   291  0 13:14:31 ?        0:00 dxs -d A
sms-dca  4429   291  0 13:14:31 ?        0:00 dca -d A
```

Note – The previous list of domain services is a sample of the services that may be encountered. Depending on how many domains are in use, more SMS daemons may be running.

These SMS daemons are started by `/etc/rc2.d/S99sms`.

The following paragraphs briefly describe the SMS daemons. For additional information on each of these daemons, refer to the *System Management Services (SMS) 1.2 Administrator Guide* and *System Management Services (SMS) 1.2 Reference Guide*.

dca

This daemon (domain configuration administration) supports remote dynamic reconfiguration (DR) by facilitating communication between applications and the dca daemon running on the domain. A separate instantiation of the dca daemon is run on the main SC for each domain running Solaris OE.

dsmd

This daemon (domain status monitoring daemon) monitors domain state, CPU reset conditions, and the Solaris OE heartbeat for all domains. This daemon notifies the dxs daemon and Sun Management Center software of all changes.

dxs

This daemon (domain x server) provides a variety of software support for a running domain including DR, hot-pluggable PCI I/O assembly (HPCI) support, domain driver requests and events, and virtual console support. One dxs daemon is started on the main SC for each running domain.

efe

This daemon (event front end) receives notification of events from various SMS daemons and forwards them to subscribed clients. With SMS 1.1 and 1.2, the only client that can subscribe is Sun Management Center 3.0 software.

esmd

This daemon (environmental status monitoring daemon) provides monitoring of the environment conditions of Sun Fire systems, including system cabinet conditions and fan tray and power supply temperatures. One instance of the esmd is run on the main SC.

fomd

This daemon (failover management daemon) is the center of the SC failover mechanism through its ability to detect faults on remote or local SCs and take appropriate action. One instance of fomd is run on the main and spare SCs. This daemon uses RPC services on the SC and is the reason why rpcbind is not disabled.

frad

This daemon (FRU access daemon) is the field-replaceable unit (FRU) access daemon for SMS. It is the mechanism by which access is provided to the serial electrically erasable programmable read-only memory (SEEPROMs) within the Sun Fire frame, to which the SC has access. The `frad` is run on the main and spare SCs.

hwad

This daemon (hardware access daemon) implements hardware access for SMS daemons used by the daemons to control, access, configure, and monitor hardware. The `hwad` is run on the main and spare SCs.

kmd

This daemon (key management daemon) manages the IPsec authenticated communication between the SC and domains. One instance of `kmd` is run on the main SC.

mand

This daemon (management network daemon) supports the Management Network (MAN). The role played by the `mand` daemon is specified by `fomd`. One instance of `mand` is run on both the main and spare SCs.

mld

This daemon (message logging daemon) accepts the output of all SMS daemons and processes and logs those messages based on its configuration files. One instance of `mld` is run on the main and spare SCs.

osd

This daemon (OpenBoot™ PROM support daemon) supports the OpenBoot PROM process running on a domain through the mailbox that resides on the domain. When the domain OpenBoot PROM writes requests to the mailbox, the `osd` daemon executes those requests. Only the main SC is responsible for booting domains. One instance of `osd` is run on the main SC.

pcd

This daemon (platform configuration database daemon) is responsible for managing and controlling access to platform and domain configuration information. The pcd is run only on the main SC.

ssd

This daemon (SMS startup daemon) starts, stops, and monitors all the key daemons and servers of SMS software. One instance of ssd is run on both the main and spare SCs.

tmd

This daemon (task management daemon) implements task management services for the SMS software such as scheduling. Currently, this daemon is used by setkeyswitch and other daemons to schedule hardware power-on self-test (HPOST) invocations. The main SC is responsible for these types of events, so one instance of tmd is run on the main SC.

SC Network Interfaces

There are several network interfaces used on an SC to communicate with the platform, domains, and other SCs. Most of these interfaces are defined as regular Ethernet network connections through `/etc/hostname`.* entries.

Main SC Network Interfaces

A typical main SC (`sc0` in our sample) has two files in `/etc` with contents similar to the following:

```
# more /etc/hostname.scman0
192.168.103.1 netmask + private up
# more /etc/hostname.scman1
192.168.103.33 netmask + private up
```

In addition, a typical main SC has corresponding entries in `/etc/netmasks`:

```
192.168.103.0   255.255.255.224
192.168.103.32  255.255.255.252
```

Note – Non-routed (RFC 1918) IP addresses are used in all SC examples. We recommend that you use these types of IP addresses when deploying Sun Fire SCs. The SMS software defines internal SC network connections to be private and not advertised.

Domain-to-SC Communication (`scman0`*) Interface*

The `/etc/hostname.scman0` entry sets up the I1 or domain-to-SC MAN. The first IP address in our example, 192.168.103.1, is controlled by the SMS software to always be available only on the main SC.

From a security perspective, the I1 MAN network between the domains and the SC, in addition to any network connection between the domains, may adversely impact domain separation. The hardware implementation of the I1 network within a Sun Fire 12K or 15K chassis addresses these concerns by permitting only SC-to-domain and domain-to-SC communication. The I1 MAN network is implemented as separate point-to-point physical network connections between the SCs and each of the 9 domains supported by a Sun Fire 12K system or 18 domains supported by a Sun Fire 15K system. Each of these connections terminates at separate I/O boards on each domain and SC.

On the SCs, these multiple separate networks are consolidated into one meta-interface to simplify administration and management. The I1 MAN driver software performs this consolidation and enforces domain separation and failovers to redundant communication paths.

Direct communication between domains over the I1 network is not permitted by the hardware implementation of the I1 network. By implementing the network in this manner, each SC-to-domain network connection is physically isolated from other connections.

Note – Although the `scman0` network supports regular Internet Protocol (IP)-based network traffic, it should be used only by Sun Fire management traffic. Any other use of this internal network may affect the reliability, availability, and serviceability (RAS) of the entire platform. Refer to the `scman` (7D) and `dman` (7D) man pages for more information.

SC-to-SC Communication (`scman1`*) Interface*

The `/etc/hostname.scman1` entry is used to configure the `I2` or SC-to-SC MAN. This network connection, on which both SCs have an IP address, is for the heartbeat connections between the two SCs.

Both of these network connections are implemented through the Sun Fire 12K or 15K internal MAN. No external wiring is used.

Spare SC Network Interfaces

The spare SC has the same physical network interfaces as the main SC. The `scman0` network interface is plumbed by the Solaris OE through the `/etc/hostname.scman0` file on the spare SC in the same manner, and with the same information as on the main SC. The difference between the main and spare SCs is that the interface is inactive on the spare. The spare SCs `scman0` port on the I/O hubs is disabled and `mand` does not provide path information to `scman0` on the spare.

The `scman1` interface, which is for SC-to-SC communication, has the following configuration information for this interface:

```
# more /etc/hostname.scman1
192.168.103.34 netmask + private up
```

In addition, the spare SC has the following corresponding `/etc/netmask` information:

```
192.168.103.32  255.255.255.252
```

Main and Spare Network Interface Sample Configurations

Putting them all together, our network configuration sample appears as follows on the main SC (`sc0`):

```
# ifconfig -a
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 1 inet
127.0.0.1 netmask ff000000

hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2 inet
10.1.72.80 netmask fffff800 broadcast 10.1.79.255 ether 8:0:20:a8:db:2e

scman0:flags=1008843<UP,BROADCAST,RUNNING,MULTICAST,PRIVATE,IPv4> mtu 1500 index 3
inet 192.168.103.1 netmask ffffffe0 broadcast 192.168.103.31 ether 8:0:20:a8:db:2e

scman1:flags=1008843<UP,BROADCAST,RUNNING,MULTICAST,PRIVATE,IPv4> mtu 1500 index 4
inet 192.168.103.33 netmask fffffffc broadcast 192.168.103.35 ether 8:0:20:a8:db:2e
```

Although the `scman0` network supports regular Internet Protocol (IP)-based network traffic, it should be used only by Sun Fire management traffic. Any other use of this internal network may affect the reliability, availability, and serviceability (RAS) of the entire platform. Refer to the `scman` (7D) and `dman` (7D) man pages for more information.

Use the following command to verify the status of the main SC:

```
# showfailover -r
MAIN
```

Putting them all together, our sample network configuration appears as follows on the spare SC (`sc1`):

```
# ifconfig -a
lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232 index 1
        inet 127.0.0.1 netmask ff000000

hme0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
        inet 10.1.72.81 netmask ffffff00 broadcast 10.1.72.255

scman0:flags=1008843<UP,BROADCAST,RUNNING,MULTICAST,PRIVATE,IPv4> mtu 1500 index 3
        inet 192.168.103.1 netmask ffffffe0 broadcast 192.168.103.31

scman1: flags=1008843<UP,BROADCAST,RUNNING,MULTICAST,PRIVATE,IPv4> mtu 1500 index 4
        inet 192.168.103.34 netmask fffffffc broadcast 192.168.103.35
```

Security Options in SMS 1.2

To improve network performance on the I1 MAN network, sequential MAC addresses are used by default on each of the up to 18 domains. With this configuration, it is straightforward to determine what the MAC address is of any given domain. It is possible, therefore, for a domain to broadcast gratuitous ARP information containing erroneous MAC addresses. The SC accepts these malicious MAC packets and uses them to misroute packets destined for domains. To protect against this type of ARP spoofing attack and other IP-based attacks, two options are available beginning with SMS 1.2:

- Disable ARP on the I1 MAN network between the SCs and domains.
- Disable all IP traffic between the SC and a domain by excluding that domain from the SC's MAN driver

We strongly recommend that you disable ARP on the MAN network in all multi-domain Sun Fire configurations. For multi-domain system configurations where domain separation is of critical concern, we also recommend disabling IP connectivity between the SC and all domains that require separation.

Disabling ARP on the MAN network provides some protection against ARP attacks, but it still leaves all other IP functionality present in the I1 network. If more stringent security is required, disabling all IP traffic between the SCs and one or more individual domains on the I1 network may be necessary. Instructions for implementing these two options are provided later in this article.

Note – Disabling ARP on the I1 MAN network impacts all domains and SCs within a Sun Fire 12K or 15K chassis. Implementing this option requires modifications to the SCs and all domains.

We recommend running software on the SC to monitor modifications to the ARP table, regardless of whether you use either of these options. One example of such software is `arpwatch`. Other similar software available either freely or commercially can be used. The freeware `arpwatch` software generates alerts based on ARP table modifications and is available from a variety of security sites on the Internet. As with any other non-SMS application to be run on an SC, the impact of `arpwatch` must be evaluated against the OpenSC guidelines presented in the *Sun Fire 15K Open System Controller (OpenSC)* white paper.

If a domain is excluded from the MAN network, the domain-to-SC network interface `dman0` is not configured at installation time. Even if the `dman0` interface is manually configured, the domain cannot communicate with the SC because the domain is excluded from the SC perspective. This solution provides excellent protection for the Sun Fire 12K or 15K chassis against malicious domains attempting to attack either the SC or other domains in the chassis. We recommend this solution for environments that require strongly enforced separation between domains and SCs.

Be aware that when you disable all IP traffic on the I1 MAN network, some functionality is no longer available. The services that are unavailable are as follows:

- Dynamic reconfiguration (DR) from the SC: commands such as `addboard`, `removeboard`, `deleteboard`, and `rcfgadm` cannot be used for domains excluded from the I1 MAN network
- Network time protocol (NTP) from the SC for the domains

Domain-side DR is still available for domains that are excluded from the MAN network. Also, console access to the domains is available because console traffic does not have to use the internal I1 MAN network. Console access can use the Sun Fire 12K or 15K server's IOSRAM, or "mailbox," connection to reach the domains. The IOSRAM interface is not TCP/IP based. Services using the IOSRAM interface, such as domain booting, remain available even if IP traffic to one or more domains is disabled.

Ultimately, security policy and enterprise application requirements may be the deciding factor as to which option is most suitable. Disabling ARP on the I1 MAN network provides some protection for domains against ARP attacks, but it still leaves all the functionality present in the MAN network. If more stringent security is required, disable all IP traffic between the SCs and one or more individual domains on the MAN network.

To enforce strict separation between a domain and all other domains and SCs in a Sun Fire 12K or 15K chassis, we recommend that the domain be excluded from the MAN network. This change can only be performed on the SC and is described later in this article.

Security Recommendations

The recommendations made in other Sun BluePrint OnLine articles apply to the Solaris OE configuration of the Sun Fire SCs. This article uses these recommendations and SC-specific security recommendations to improve the overall security posture of Sun Fire SCs by dramatically reducing potential access points to the SC and installing secure access mechanisms.

The recommendations for securing the SC follow closely with the hardening described in the "Solaris Operating Environment Security - Updated for Solaris 8 Operating Environment" Sun BluePrints OnLine article.

To improve the overall security posture of a Sun Fire 12K or 15K SC, we strongly recommend that you install all required patches for Solaris OE and SMS software. Refer to the "Solaris Operating Environment Security - Updated for Solaris 8 Operating Environment" article for instructions on how to incorporate patch installation with the Solaris Security Toolkit.

We made the following exceptions to the recommendations provided in the previously mentioned article, due to functionality that is required by the SC and supportability constraints:

- The `in.rshd`, `in.rlogind`, and `in.rexecd` daemon entries listed in the `/etc/inetd.conf` are not disabled because the failover management daemon (`fomd`) requires them.
- For `fomd` to effectively use the `in.rshd`, `in.rlogind`, and `in.rexecd` daemons, a `/.rhosts` file must be present on both SCs. This file contains the `scman1` network hostname of the other SC and allows `fomd` to access the SC, as `root`, without requiring a password.

Note – Beginning with SMS 1.2 and patch 112481-05, you can use Secure Shell as an alternative transport mechanism for `fomd`, removing the requirement for `in.rshd`, `in.rlogind`, `in.rexecd`, and `/.rhosts` on the SC. Using this alternative transport is strongly recommended. The use of any Secure Shell implementation is possible, although the examples in this article are based on OpenSSH running on Solaris 8 OE.

- The remote procedure call (RPC) system startup script is not disabled because RPC is used by `fomd`.

- The Solaris Basic Security Module (Solaris BSM) is not enabled. The Solaris BSM subsystem is difficult to optimize for appropriate logging levels, and its logs are difficult to interpret. This subsystem should only be enabled in those sites that have the expertise and resources to manage the generation and data reconciliation tasks required to use Solaris BSM effectively.
- Solaris OE minimization of the SC is not described in this article.
- The SC cannot be configured as a network time protocol (NTP) client.

The creation of user accounts and their associated privileges are not addressed in this article. Adding new users to an SC requires that the users be provided with privileges not only in the Solaris OE but also with SMS domain and platform privileges. Refer to the *System Management Services (SMS) 1.2 Administrator Guide* for instruction on how to define user access to the SMS software.

Securing the System Controller

In order to effectively secure an SC, changes are required to both the Solaris OE software running on the SC and the configuration of the Sun Fire 12K or 15K platform. To simplify the Solaris OE installation and deployment of these recommendations, we added customized modules to versions 0.3.8 and later of the Solaris Security Toolkit software. These modules automate the implementation of the security recommendations.

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 – We recommend that you disable failover before hardening either of the SCs. Re-enable failover only after both SCs are hardened *and* tested.

The Sun Fire 12K and 15K SC module `sunfire_15k_sc-secure.driver` exclusively performs hardening tasks. No minimization of the Solaris OE is performed.

You can use the Sun Fire 12K and 15K SC module of the Solaris Security Toolkit in either standalone or JumpStart™ mode to secure an SC.

Note – Configuration modifications for performance enhancements and software configuration are not addressed by the Solaris Security Toolkit.

To harden the SCs, perform the following tasks:

- “Adding Security Software” on page 20
- “Customizing the Solaris Security Toolkit Driver” on page 26 (optional)
- “Overriding Solaris Security Toolkit Defaults” on page 36 (optional)
- “Installing Downloaded Software and Implementing Modifications” on page 37

Adding Security Software

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

- “Install Solaris Security Toolkit Software” on page 20
- “Download Recommended Patch Cluster Software” on page 21
- “Download FixModes Software” on page 23
- “Download OpenSSH Software” on page 24
- “Download the MD5 Software” on page 25

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 Secure Shell, available from a variety of vendors. You must install a Secure Shell product on the SCs.

Install Solaris Security Toolkit Software

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

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 0.3.8 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-0.3.8.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 by using the** uncompress **command**:

   ```
   # uncompress SUNWjass-0.3.8.pkg.Z
   ```

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

   ```
   # pkgadd -d SUNWjass-0.3.8.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-jass-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 SC, this section describes how to download the latest patch cluster.

Downloading the latest patch cluster does not require a SunSolve OnLine^{SM} 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 SC with the `scp` command, or `ftp` if Secure Shell is not available.**

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

```
% scp 8_Recommended.zip sun15-sc0:/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 will 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 SC. 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® technology. The file name is `FixModes.tar.Z`.

   Note – Only certain versions of FixModes are supported for use on Sun Fire SCs. The correct FixModes version must have `secure-modes.c` version 1.41 and `exempt-pkgs.h` version 1.1. Newer versions of either file are acceptable. Earlier version of FixModes must not be used to secure Sun Fire SCs.

   ---

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

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

   ```
   % scp FixModes.tar.Z sun15-sc0:/var/tmp
   ```

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

   ```
   # cd /opt/SUNWjass/Packages
   # mv /var/tmp/FixModes.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, network access to the SC must be encrypted.

The tool most commonly used to implement encryption is Secure Shell software, whether a version bundled with Solaris, a third-party 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 a Secure Shell software product.

Note – With the release of Solaris 9 OE, a version of Solaris Secure Shell is included. If using Solaris 9 OE, we strongly recommend using this Secure Shell version.

Information on where to obtain commercial versions of Secure Shell is provided in "Related Resources" on page 43.

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

Access to the system can be gained with Secure Shell similarly to what is provided by Telnet and ftp.

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

▼ To Download OpenSSH Software

Note – If the SC is running Solaris 9 OE, you can use the Solaris Secure Shell software and skip the OpenSSH installation steps in this section.

- **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 SC 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 Secure Shell, then no compiling is required.

Download the MD5 Software

The MD5 software validates MD5 digital fingerprints on the SC. 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 backdoor access onto a system; they hide their presence and cause systems to operate in unstable manners.

▼ To Install the MD5 Software

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 SC with the `scp` command, or `ftp` if `scp` is not available.**

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

```
% scp md5.tar.Z sun15-sc0:/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.

Customizing the Solaris Security Toolkit Driver

If you determine that your system requires some of the services and daemons disabled by the Solaris Security Toolkit, or you want to enable any of the inactive scripts available in the Solaris Security Toolkit, do so before executing the Solaris Security Toolkit.

As described earlier in this article, the SMS 1.2 software provides new capabilities for securing the MAN network:

- Disable ARP on the MAN network.
- Disable all I1 IP traffic between the SCs and specific domains.

Also, you can use Solaris Secure Shell as an alternative transport mechanism for `fomd`, removing the absolute requirement for `in.rshd`, `in.rlogind`, `in.rexecd` and `/.rhosts` on the SC. We strongly recommend that you use this alternative transport. To use this functionality with SMS 1.2, patch number 112481-05 or newer must be installed on the SC.

We strongly recommend that you disable ARP on the MAN network. For multi-domain system configurations where domain separation is of critical concern, we also recommend disabling IP connectivity between the SC and those domains that require separation.

Disabling ARP on the MAN network can only be done for an entire chassis. It is not possible to make this change only for certain domains. It must be done on all domains having IP connectivity to the I1 network.

Caution – When disabling ARP on a SunFire 12K or 15K system, it is critical that the necessary configuration changes be made to *all domains and both SCs* at the same time. Making the changes only on certain domains or SCs causes the system to malfunction.

Implementing any of these modifications to the SC requires modifying the files included with the Solaris Security Toolkit, as well as domain side modifications when disabling ARP on the MAN network. In addition, if you are implementing Secure Shell as an alternative transport for `fomd`, then additional manual steps are required.

The following sections provide instructions for using each of these options. The instructions include the required modifications to the Solaris Security Toolkit driver in addition to any manual modifications required. The modifications described are cumulative; if you want to use all three options, perform all the steps described in each of the three sections.

▼ To Disable ARP

1. **To add the necessary features or customize the hardening required for your system, edit a copy of the** `sunfire_15k_sc-hardening.driver file`**..**

```
# cd /opt/SUNWjass/Drivers
# vi sunfire_15k_sc-hardening.driver
```

Caution – To preserve your changes for future updates and prevent the Solaris Security Toolkit from overriding your changes, modify only a copy of the driver. Keep the original Solaris Security Toolkit driver as a master.

2. **If static ARP configuration is required for this SC, uncomment** `s15k-static-arp.fin` **from the driver by removing the # symbol in front of the script.**

   After editing the line, it should appear as follows in the `JASS_SCRIPTS` definition:

```
 s15k-static-arp.fin
```

3. **Review the domain hostname to MAC address mapping in the** `sms_sc_arp` **file.**

   This file is in the `/opt/SUNWjass/Files/etc` directory. The Solaris Security Toolkit uses the following initial values (`sun15-a` through `sun15-r`) in this file:

```
sun15-a        08:00:20:d5:c6:09
sun15-b        08:00:20:fc:7e:4e
sun15-c        08:00:20:b2:a8:c7
sun15-d        08:00:20:cf:ad:7d
sun15-e        08:00:20:ce:de:e5
sun15-f        08:00:20:cc:c9:b7
sun15-g        08:00:20:9d:02:e1
sun15-h        08:00:20:2a:7f:f6
sun15-i        08:00:20:f4:39:37
sun15-j        08:00:20:6e:aa:31
sun15-k        08:00:20:90:67:88
sun15-l        08:00:20:ce:e6:5e
sun15-m        08:00:20:5a:27:7f
sun15-n        08:00:20:55:e5:36
sun15-o        08:00:20:67:78:73
sun15-p        08:00:20:c6:84:b5
sun15-q        08:00:20:c0:37:cb
sun15-r        08:00:20:41:9d:68
```

   a. **If your site configuration for the MAN network uses different domain hostnames, replace the** `sun15-a` **through** `sun15-r` **values with your hostnames.**

   b. **If your site configuration requires different MAC addresses, replace them with MAC addresses that match your domain hostnames in this file for both SCs.**

4. **Review the IP Address for the I1 MAN interface of the main SC and matching MAC address in the** `sms_domain_arp` **file.**

   This file is in the `/opt/SUNWjass/Files/etc` directory. The Solaris Security Toolkit uses the following initial values in this file:

```
192.168.103.1           08:00:20:63:49:1e
```

   c. **If your site configuration for the MAN network uses a different IP Address for the I1 MAN interface, replace the** `192.168.103.1` **value with the IP address of the I1 MAN interface used in your environment.**

   d. **If your site configuration requires a different MAC address than the initial** `08:00:20:63:49:1e` **value, replace it with the MAC address that matches the IP Address for the I1 MAN interface on all domains and both SCs.**

   All the domains must use the same `/etc/sms_domain_arp` file.

Caution – The IP Address of the main SC in this file must match the IP address chosen as the IP Address of the SC on the I1 MAN network. Any mismatches cause MAN network failures. These failures can adversely affect the reliability, availability, and serviceability (RAS) of the platform.

Note – If a domain is configured to have both s15k-exclude-domains.fin and s15k-static-arp.fin applied to it, the result is equivalent to s15k-exclude-domains.fin. It is possible, however, to have some domains excluded while others use static ARPs. We strongly recommend that you disable ARP on the I1 MAN network to protect against ARP-spoofing attacks. Note that disabling ARP on the I1 MAN network is a modification that affects the entire chassis and all the domains in the chassis. It is not possible to disable ARP only between certain domains and the SC.

▼ To Disable I1 Traffic

1. **To add the necessary features or customize the hardening required for your system, edit a copy of the** sunfire_15k_sc-hardening.driver file**..**

```
# cd /opt/SUNWjass/Drivers
# vi sunfire_15k_sc-hardening.driver
```

Caution – To preserve your changes for future updates and prevent the Solaris Security Toolkit from overriding your changes, modify only a copy of the driver. Keep the original Solaris Security Toolkit driver as a master.

2. **If domain exclusion is required for this SC, uncomment** s15k-exclude-domains.fin **from the driver by removing the # symbol in front of the script.**

   After editing the line, it should appear as follows in the JASS_SCRIPTS definition:

```
  s15k-exclude-domains.fin
```

   The default configuration of the s15k-exclude-domains.fin script is to disable I1 IP connectivity for all possible domains on the system.

3. **If you do not want the default configuration, then edit the** s15k-exclude-domains.fin **script.**

The s15k-exclude-domains.fin script is in the /opt/SUNWjass/Files directory and includes a variable, domain_RE, which specifies the domains to have their I1 IP connectivity disabled. To modify the domains that are impacted by this script, the definition of domain_RE must be changed. For example, if the following regular expression were used, then all domains would have their IP I1 connectivity disabled except for domain D:

```
domain_RE='D[A-CE-R]-I1'
```

Note – If a domain is configured to have both s15k-exclude-domains.fin and s15k-static-arp.fin applied to it, the result is equivalent to s15k-exclude-domains.fin. It is possible, however, to have some domains excluded while others use static ARPs. We strongly recommend that you disable ARP on the I1 MAN network to protect against ARP-spoofing attacks. Note that disabling ARP on the I1 MAN network is a modification that affects the entire chassis and all the domains in the chassis. It is not possible to disable ARP only between certain domains and the SC.

▼ To Use fomd With Secure Shell Instead of r*

Note – We recommend that you disable the failover mechanism before hardening the SCs. Re-enable failover only after you harden and test both SCs.

Using fomd with Secure Shell involves performing the following set up procedures:

- "Verify Installation or Install Patch and Secure Shell Software" on page 31
- "Configure Secure Shell" on page 31
- "Reboot the SCs and Verify Configurations" on page 34
- "Add Features and Customize the Hardening" on page 35

Caution – Although any version of Secure Shell may be used by fomd, the Secure Shell binaries scp and ssh must be available in one of the following locations: /usr/bin, /opt/SUNWSMS/SMS/bin, or /opt/OBSDssh/bin. If the binaries, or links to the binaries, are not found in any of these three locations, then fomd reverts to r*. The fomd generates log messages when reverting to r*. If the r* services are disabled and fomd cannot fall back, then fomd generates file propogation errors for each file it cannot copy.

Verify Installation or Install Patch and Secure Shell Software

1. **If patch 112481-05 or newer is not installed, download it from SunSolve OnLine and install it on both SCs.**

2. **Review the contents of the patch README file before continuing.**

3. **Verify that a Secure Shell version is installed, configured, and running appropriately.**

Note – For `fomd` to work properly over Secure Shell, it must be possible for `root` to `ssh` from one SC to the other SC without requiring a password or command line options.

Configure Secure Shell

If using the Open BSD-based OpenSSH Secure Shell package, perform the following steps to configure Secure Shell properly between the two SCs.

We recommend creating 2048-bit keys using RSA for Secure Shell protocol version 2. Be aware that using 2048-bit keys may introduce a slight performance reduction.

The defaults for OpenSSH on Solaris 8 (2/02) are 1024-bit keys with RSA for protocol version 1, that is, `-t rsa1`. You must specify `-t rsa` to get RSA for protocol version 2.

Note – The following steps require that you have `root` access and know the correct hostnames for the SCs on the I2 MAN network. These are the hostnames that SC failover uses; therefore, Secure Shell must be configured properly for SC failover to work.

In the following steps, we use the filename `/.ssh/id_rsa` for the Secure Shell keys and the default I2 MAN hostnames, `sun15-sc0-i2` and `sun15-sc1-i2`, for the SCs.

1. **If using Solaris 9 OE, change the default Secure Shell configuration to allow root log in on both SCs:**

   a. **On both** `SC0` **and** `SC1`**, edit the file** `/etc/ssh/sshd_config` **to allow root log in.**

      SC failover does not work if this step is omitted.

b. Change the PermitRootLogin parameter from `no` **to** `yes`**.**

c. Reboot the SCs or restart the Secure Shell server to implement the changes.

2. **On the main SC (**`SC0`**) in a terminal window command line, perform the following steps:**

Note – We recommend that you display terminal windows side by side for both SCs (main and spare) throughout this procedure. Doing so makes it easier to enter the data and perform sequential steps on each SC.

---

a. Generate a host key by entering the following command:

```
# /opt/OBSDssh/bin/ssh-keygen -b 2048 -t rsa
```

b. Accept the default filename without entering a pass phrase.

c. Transfer the public key file to a temporary file on `SC1` **using either** `scp` **(if available) or** `rcp`**:**

```
# scp  /.ssh/id_rsa.pub  sun15-sc1-i2:/.ssh/id_rsa.pub-sc0
```

or

```
# rcp  /.ssh/id_rsa.pub  sun15-sc1-i2:/.ssh/id_rsa.pub-sc0
```

Caution – Do not omit the remote filename or use `/.ssh/id_rsa.pub` as the filename on `SC1`. If there is already a public key file on `SC1`, it may be overwritten and invalidate `SC1`'s private key.

---

3. **On the spare SC (**`SC1`**) in a terminal window command line, perform the following steps:**

a. Generate a host key by entering the following command:

```
# /opt/OBSDssh/bin/ssh-keygen -b 2048 -t rsa
```

b. Accept the default filename without entering a pass phrase.

c. **Transfer the public key file to a temporary file on** SC0 **using either** scp **(if available) or** rcp**:**

```
# scp  /.ssh/id_rsa.pub  sun15-sc0-i2:/.ssh/id_rsa.pub-sc1
```

or

```
# rcp  /.ssh/id_rsa.pub  sun15-sc0-i2:/.ssh/id_rsa.pub-sc1
```

Caution – Do not omit the remote filename or use /.ssh/id_rsa.pub as the filename on SC0. If there is already a public key file on SC0, it may be overwritten and invalidate SC0's private key.

4. **Return to the main SC and perform the following steps:**

a. **Append** SC1**'s public key to the** /.ssh/authorized_keys2 **or** /.ssh/authorized_keys **file.**

Note – If you are using OpenSSH, the file to use is /.ssh/authorized_keys2. If you are using Sun's Secure Shell on Solaris 9, the file to use is /.ssh/authorized_keys. This command creates the file if it does not exist.

```
# cat  /.ssh/id_rsa.pub-sc1 >> /.ssh/authorized_keys2
```

or

```
# cat  /.ssh/id_rsa.pub-sc1 >> /.ssh/authorized_keys
```

b. **Remove the temporary file.**

```
# rm  /.ssh/id_rsa.pub-sc1
```

5. **Return to the spare SC and perform the following steps:**

   a. **Append** `SC0`**’s public key to the** `/.ssh/authorized_keys2` **or** `/.ssh/authorized_keys` **file.**

```
# cat  /.ssh/id_rsa.pub-sc0 >> /.ssh/authorized_keys2
```

or

```
# cat  /.ssh/id_rsa.pub-sc0 >> /.ssh/authorized_keys
```

   b. **Remove the temporary file.**

```
# rm  /.ssh/id_rsa.pub-sc0
```

6. **Due to a known incompatibility between the SMS software and Secure Shell startup script, move the Secure Shell startup script from UNIX run-level 3 to run-level 2 on both the main and spare SC.**

```
# mv  /etc/rc3.d/S*ssh*  /etc/rc2.d/.
```

Reboot the SCs and Verify Configurations

1. **Reboot each of the SCs by entering the following command from the platform shell:**

```
# reboot -y
```

Note – You can reboot the SCs while the domains are running.

---

2. **Verify the configuration on** `SC0` **works properly.**

   a. **Log into** `SC1` **from** `SC0` **using** `ssh`:

```
# ssh  sun15-sc1-i2
```

   b. **When prompted to add this host to the list of known hosts, answer** `yes`**.**

c. Log out of `SC1` **with** "`exit`."

d. Log into `SC1` **from** `SC0` **again with** `ssh`**, and verify that no prompts are generated and that the login without a password is successful.**

3. Verify the configuration on `SC1` **works properly.**

a. Log into `SC0` **from** `SC1` **using** `ssh`:

```
# ssh  sun15-sc0-i2
```

b. When prompted to add this host to the list of known hosts, answer `yes`**.**

c. Log out of `SC0` **with** "`exit`."

d. Log into `SC0` **from** `SC1` **again with** `ssh`**, and verify that no prompts are generated and that the login without a password is successful.**

Add Features and Customize the Hardening

1. To add necessary features and customize the hardening required for your system, edit a copy of the `sunfire_15k_sc-hardening.driver file`**..**

```
# cd /opt/SUNWjass/Drivers
# vi sunfire_15k_sc-hardening.driver
```

Caution – To preserve your changes for future updates and prevent the Solaris Security Toolkit from overriding your changes, modify only a copy of the driver. Keep the original Solaris Security Toolkit driver as a master.

2. **Uncomment the** s15k-sms-secure-failover.fin **from the driver by removing the # symbol in front of the script.**

   After editing the line, it should appear as follows in the JASS_SCRIPTS definition:

   ```
   s15k-sms-secure-failover.fin
   ```

   This script disables the r* services in the /etc/inetd.conf file automatically and removes the /.rhosts file. The five services disabled are as follows:

   ```
   #shell stream  tcp  nowait  root /usr/sbin/in.rshd      in.rshd
   #shell stream  tcp6 nowait  root /usr/sbin/in.rshd      in.rshd
   #login stream  tcp6 nowait  root /usr/sbin/in.rlogind   in.rlogind
   #exec  stream  tcp  nowait  root /usr/sbin/in.rexecd    in.rexecd
   #exec  stream  tcp6 nowait  root /usr/sbin/in.rexecd    in.rexecd
   ```

Overriding Solaris Security Toolkit Defaults

If there are some services that must remain enabled, and the Solaris Security Toolkit automatically disables them, you can override the defaults before executing the driver.

To prevent the toolkit from disabling a service, comment out the call to the appropriate finish script in the driver.

For example, if your environment requires Network File System (NFS)-based services, you can leave them enabled. Comment out the disable-nfs-server.fin and disable-rpc.fin scripts by appending a # sign before them in the copy of the sunfire_15k_domain-hardening.driver script.

For more information about editing and creating driver scripts, refer to the Sun BluePrints OnLine article titled “The Solaris™ Security Toolkit - Installation, Configuration, and Usage Guide: Updated for version 0.3.”

Installing Downloaded Software and Implementing Modifications

The Solaris Security Toolkit version 0.3.8 and later provides a driver (`sunfire_15k_sc-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.”

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

▼ To Install Downloaded Software and Implement Changes

Note – Even if no patches are installed by the Solaris Security Toolkit, it is critical that the `sunfire_15k_sc-secure.driver` be called. Calling either the `-harden` or `-config` drivers may result in an unsupported configuration.

- **Execute the** `sunfire_15k_sc-secure.driver` **script as follows:**

```
# cd /opt/SUNWjass
# ./jass-execute -d sunfire_15k_sc-secure.driver
./jass-execute: NOTICE: Executing driver,
sunfire_15k_sc-secure.driver

============================================================
sunfire_15k_sc-secure.driver: Driver started.
============================================================
[...]
```

Note – The hardening described in this article is performed in standalone mode, not JumpStart mode, because the SC 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.

▼ 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.

Note – By default, the Solaris Security Toolkit overwrites any files backed up during earlier runs being undone. In some cases, this action overwrites changes made to files since the run was performed. If you have concerns about overwriting changes, use the -n (no force) option to prevent modified files from being overwritten. Please refer to the Solaris Security Toolkit documentation for more details about this option.

● **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.

Note – You cannot use the undo feature on `install-fixmodes.fin`, `install-openssh.fin`, and `install-strong-permissions.fin` Solaris Security Toolkit scripts. Refer to the Solaris Security Toolkit documentation for additional information.

Verifying SC Hardening

Note – We recommend that you disable the failover mechanism before hardening the SCs. Re-enable failover only after you harden and test both SCs.

After performing the procedures in this article to harden the SC, test the configuration and hardening.

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

- TCP IPv4 services listed by `netstat` went from 31 to 6
- UDP IPv4 services listed by `netstat` went from 57 to 5

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

```
# netstat -a

UDP: IPv4
   Local Address         Remote Address     State
-------------------- -------------------- -------
      *.sunrpc                              Idle
      *.32771                               Idle
      *.32773                               Idle
      *.syslog                              Idle
      *.32776                               Idle
      *.*                                   Unbound

TCP: IPv4
   Local Address        Remote Address    Swind Send-Q Rwind Recv-Q  State
-------------------- -------------------- ----- ------ ----- ------ -----
      *.sunrpc           *.*                0      0 24576      0 LISTEN
      *.32771            *.*                0      0 24576      0 LISTEN
      *.sun-dr           *.*                0      0 24576      0 LISTEN
      *.32772            *.*                0      0 24576      0 LISTEN
      *.32773            *.*                0      0 24576      0 LISTEN
      *.22               *.*                0      0 24576      0 LISTEN
      *.*                *.*                0      0 24576      0 IDLE
```

▼ To Test the Main SC

1. **Disable the failover mechanism.**
2. **Reboot the SC.**
3. **Place the hardened SC in the main SC role.**
4. **Verify that the SC takes control of the frame.**
5. **Verify that the SMS controls the platform and functions properly.**
6. **Validate that the number of daemons and services running on the SC are significantly lower than before hardening.**
7. **After verifying that the main SC is hardened and functioning properly, perform all of the same procedures in this article (all software installation and hardening processes) on the spare SC.**

   The spare SC must not be hardened until the main SC is tested.

8. **Manually define the newly hardened and tested main SC as the default main SC.**

▼ To Test the Spare SC

After hardening the main SC, testing it, and manually defining it as the main, harden and test the spare SC.

Caution – Do not harden the spare SC until you verify that the hardened main SC functions properly in your environment.

---

1. **Disable the failover mechanism.**
2. **Reboot the SC.**
3. **Place the hardened SC in the spare SC role.**
4. **Verify that the spare SC takes control of the frame by becoming the main SC, and that the SMS controls the platform and functions properly.**
5. **Validate that the number of daemons and services running on the SC are significantly lower than before hardening.**
6. **Enable failover only after you harden and test both SCs.**
7. **Test failover and verify that each SC can assume the main role when appropriate.**

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 OE, 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.

Dina K. Nimeh

Dina Nimeh is a Senior Software Engineer with 15 years of experience in many areas from device drivers to databases. For the past four years, Dina has focused on secure software development and the deployment of security system solutions such as vulnerability assessment tools, intrusion detection systems, and public key infrastructures. Currently, she works with the Enterprise Systems Group at Sun Microsystems.

Related Resources

Publications

- Lowman, Jacob, and Anderson, Dan. *Sun Fire 15K Open System Controller (OpenSC)* white paper, `http://www.sun.com/servers/wp/docs/opensc.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/minimization-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_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 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 Nimeh, Dina K. “Securing Sun Fire 12K and 15K Domains - Updated for SMS 1.2,” Sun BluePrints OnLine, July 2002. `http://sun.com/blueprints/0702/`
- 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`
- Sun Microsystems, Inc. *System Management Services (SMS) 1.2 Administrator Guide*, Part No 816-2527-10, Sun Microsystems, Inc., February 2002, Revision A. `http://docs.sun.com`
- Sun Microsystems, Inc. *System Management Services (SMS) 1.2 Reference Guide*, Sun Microsystems, Part No 816-2528-10, Sun Microsystems, Inc., February 2002, Revision A. `http://docs.sun.com`

- 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/0401/network-updt1.pdf`

Web Sites

- Commercial versions of SSH are available from:

  `http://www.ssh.com`

  `http://www.fsecure.com`
- The Solaris Security Toolkit software is available from:

  `http://www.sun.com/security/jass`
- SUNSOLVE ONLINE Web site: `http://sunsolve.sun.com`
- SecurityFocus Web site: `http://online.securityfocus.com/tools/142`