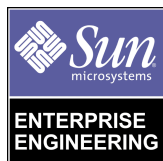




SSP Best Practices

By John S. Howard - Enterprise Engineering

Sun BluePrints™ OnLine - March 2000



<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.: 806-5094-10
Revision 01, March 2000

Copyright 2000 Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto, California 94303 U.S.A. All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, The Network Is The Computer, Sun BluePrints, Sun Enterprise, Sun Enterprise, Sun Enterprise Volume Manager, Sun StorEdge, Sun Explorer, Solstice DiskSuite, SunPS, Starfire, AnswerBook, SunFDDI, SunATM, Sun StorEdge Enterprise NetBackup, Solstice Backup, StorTools, JumpStart, and Solaris are trademarks, registered trademarks, or service marks of Sun Microsystems, Inc. in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

RESTRICTED RIGHTS: Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015(b)(6/95) and DFAR 227.7202-3(a).

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.

Copyright 2000 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, Californie 94303 Etats-Unis. Tous droits réservés.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, The Network Is The Computer, Sun BluePrints, Sun Enterprise, Sun Enterprise, Sun Enterprise Volume Manager, Sun StorEdge, Sun Explorer, Solstice DiskSuite, SunPS, Starfire, AnswerBook, SunFDDI, SunATM, Sun StorEdge Enterprise NetBackup, Solstice Backup, StorTools, JumpStart, et Solaris sont des marques de fabrique ou des marques déposées, ou marques de service, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" ET AUCUNE GARANTIE, EXPRESSE OU IMPLICITE, N'EST ACCORDEE, Y COMPRIS DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DE LA PUBLICATION A REPOUDRE A UNE UTILISATION PARTICULIERE, OU LE FAIT QU'ELLE NE SOIT PAS CONTREFAISANTE DE PRODUIT DE TIERS. CE DENI DE GARANTIE NE S'APPLIQUERAIT PAS, DANS LA MESURE OU IL SERAIT TENU JURIDIQUEMENT NUL ET NON AVENU.



Please
Recycle



Adobe PostScript

SSP Best Practices

Introduction

The System Service Processor (SSP) is a key component of the Sun Enterprise™ 10000 server. The proper functioning of the SSP can make or break the successful and timely recovery of a domain from a catastrophic or near catastrophic event. This paper will present best practice procedures and configurations for the SSP and backup SSP.

In addition, an overview of the Miniroot Recovery (MR) system for rapid recovery of systems or domains will be presented. Recovery tactics using the standard tools from Sun and Sun's suppliers (Netbackup, Networker, Sun Enterprise Volume Manager™, etc.) are quite tedious and very time consuming. This is especially true on the Sun Enterprise 10000 platform because of the longer boot cycle between stages of a full recovery. The MR system takes advantage of the architectural features of the Sun Enterprise 10000 server (especially the domain's relationship with the SSP) in order to simplify a large portion of a system recovery and maintenance.

General SSP Best Practices

The Solaris™ Operating Environment standard installation includes installation of the SSP software packages. As such, the SSP should be administered and maintained as any UNIX® system. However due to the real-time nature of the SSP software and the relationship that the SSP has to the Sun Enterprise 10000 platform, there are some SSP best practices and administrative techniques that may not be immediately apparent.

One of the key responsibilities of the SSP is to control and monitor the Sun Enterprise 10000 hardware. As such the SSP is used as a loghost, logging all `syslog` messages from the Sun Enterprise 10000 server domains serviced by that SSP. Depending on the number and size of the domains, the size of `/var/adm/messages` may increase at a rate greater than you may be accustomed to. The size of the message files should be monitored frequently. Also some devices such as a Sun StorEdge™ A5000 system with an intermittently failing hardware component, such as a GBIC or disk, can get into an error condition where they are logging hundreds of messages per second. A component or device in this failure mode can easily overrun the SSP's network interface and degrade performance of both the SSP and domain.

The SSP also acts as a repository for `arbstop` dumps, `xfstate` dumps and `recordstop` dumps from the Sun Enterprise 10000 server. An `arbstop` is when the Sun Enterprise 10000 server interconnect detects an error such as an ECC memory error, parity error, hardware queue overflow or a grant time-out. An arbitration stop request is sent to the global arbiters via the local arbiter. The global arbiters broadcast the arbitration stop to all system boards in the domain. If the problem exists in the global arbiters (a steering parity error), or is a phase lock loop error or a domain transgression error, then the arbitration stop is broadcast to all system boards and all domains are stopped. An `arbstop` on a running domain is logged by the SSP event detection daemon process (`edd`) and the ASIC history registers are captured to an `arbstop` dump file on the SSP. If an `arbstop` is requested while `hpost` is running, `hpost` will capture the ASIC history directly and dump the information to an `xfstate` dump.

A `recordstop` occurs when the Sun Enterprise 10000 hardware detects either a correctable or uncorrectable data ECC error on the data interconnect or a correctable ECC error is detected on a global address bus. It is important to note that `recordstop` are completely undetected by the Solaris Operating Environment and the software running on the domain which encounters the `recordstop` is unaffected. A domain that encounters a `recordstop` does not panic, crash or hang. As with an `arbstop`, `edd` logs the `recordstop` and writes the ASIC history registers to a `recordstop` dump on the SSP.

Individually, `arbstop` and `recordstop` dumps are not large. However, given that `recordstops` are undetected by Solaris Operating Environment and do not affect a running domain, over the course of years an inadequately monitored SSP may accumulate a large number of these files. `arbstop` and `recordstop` dumps are written to the directory:

```
/var/opt/SUNWssp/adm/$SUNW_HOSTNAME
```

(\$SUNW_HOSTNAME evaluates to the applicable domain name) on the SSP. This directory should be routinely monitored for dumps. Any `recordstop`, `arbstop` or `xfstate` dumps should be immediately sent to Sun for analysis and the files archived off the SSP.

Because of the real-time nature of several of the SSP's processes, Sun does not recommend that you install or use any third-party applications or unbundled Sun products on the SSP. As with all mission critical systems Sun™ Explorer should be regularly run on the SSP and the resulting output sent to Sun.

Domain Booting, Boot Service and the JumpStart™ Software

The SSP is also responsible for downloading POST and OpenBoot Prom (OBP) to domains, acting as a tftp boot server for the control board firmware, and as a Solaris Operating Environment boot server for the domains. Additionally, you should use the JumpStart software to provide a readily accessible fail-safe bootimage and to insure that domains and systems are installed and configured in an efficient and consistent manner.

Given all these responsibilities, you should subnet the network shared by the SSPs and the domains off the site or production backbone. If there are sufficient network interfaces available on the domains, you should create an administrative network comprised of the SSPs and the first network interface on each domain of the platform. This has the added security of the SSP not being reachable from any host except the domains serviced by the SSPs.

The Spare SSP and Fail Over of SSPs

Sun strongly recommends that all Sun Enterprise 10000 server's be configured and installed with a main SSP and a spare SSP. Equally crucial is that both the main and the spare SSPs are kept up and running on the network. Both the main and secondary SSPs should be maintained at the same version of the Solaris Operating Environment and the same version of the SSP software.

With a main and spare SSP, there is no need to use Solstice DiskSuite™ software to mirror the system disk of the main or spare SSP. In the event of a system disk failure in the main SSP, the SSP services can simply be failed over to the spare SSP. Because the SSP has only one system board and one SCSI disk controller (which are both single points of failure), the complete physical redundancy of SSPs gives you a greater protection than mirroring with Solstice DiskSuite software on the main SSP.

The fail over of main SSP to spare SSP is a relatively straightforward procedure to do manually and SunPSSM program has developed an automatic procedure for fail over of main to spare SSP. Although the manual SSP fail over is not a complicated task it does require practice to achieve a graceful and error-free fail over. The Systems Administration or Operations staff should test the fail over procedure on a monthly basis.

The procedure to manually fail over from the main SSP to the spare SSP follows. In this procedure, the primary SSP is the currently the main SSP and the secondary SSP is the spare SSP:

On the Primary SSP

Backup these directories and files to tape or similar backup media on the primary SSP:

```
/etc/hosts
/etc/ethers
/var/opt/SUNWssp
/etc/opt/SUNWssp
/export/home/ssp
/var/opt/SUNWssp/
/ssp_private/eprom_save
```

To unconfigure the primary SSP as the main SSP, log into the primary SSP as root and enter the following command:

```
/opt/SUNWssp/bin/ssp_config
```

and then respond **n** (no) to the prompt:

Are you currently configuring the MAIM SSP (y/n)

The primary SSP has now been reconfigured as a spare SSP.

Rename ethers to prevent the main SSP from responding to tftp and boot requests.

```
mv /etc/ethers /etc/
ethers.mainSSP
```

Reboot the primary SSP.

On Each Domain

For each domain on the platform served by the SSP, `rlogin` or `telnet` into the domain as root and perform the following steps:

Edit `/etc/ssphostname` and `/etc/opt/SUNWxntp/ntp.conf`, replacing the hostname of the primary SSP with the name of the secondary SSP.

Switch console communications from the primary to the secondary:

```
# ps -aef |grep cvcd
# kill -9 cvcd_pid
# /sbin/cvcd
```

On the Secondary SSP

On the secondary SSP, login as root and perform the following tasks:

Edit:

```
/var/opt/SUNWssp/.ssp_private/ssp_to_domain_hosts
```

to change the name of the primary SSP to the name of the secondary SSP

Configure the secondary SSP as the main SSP by typing the following:

```
# /opt/SUNWssp/bin/ssp_config spare
```

Restore `/etc/ethers`:

```
# mv /etc/ethers.mainSSP /etc/ethers
```

Reboot the secondary SSP.

Patching the SSPs

The SSP patches need to be updated to the latest revisions in a timely manner. Also, the recommended SSP patch cluster is more a requirement than a recommendation. To provide a non disruptive SSP patch installation and test method, use the procedure below to patch your SSPs.

Patch the secondary SSP to the latest SSP patches, but do not update the primary SSP patches.

Run a month in this configuration.

Fail over to the secondary.

Run a month in this configuration.

Patch the primary but do not fail over to it.

Run a month in this configuration.

Fail over, back to primary.

This constant rotation of SSPs provides you with a ready fall-back mechanism if a problem is encountered with the latest patch revisions; a patched up-to-current-revisions SSP to fail over to if you encounter a problem that has been fixed in the latest SSP patches; and a regularly scheduled test of your SSP fail over procedures.

The MR System Overview

The core idea of the MR system is to put the necessary recovery tools on the miniroot boot image such that the domain can boot from this augmented miniroot boot image to get everything it needs at one time (SEVM drivers, recovery client, etc). This is not difficult, but it requires careful planning and manipulation of the system configuration.

A moderately sized Starfire™ software domain can take 45 minutes to boot using default HPOST levels, and substantially longer if a more stringent selftest is performed. Traditional recovery tactics on these systems are frustratingly time consuming, in principal because multiple reboots are often required.

The time consuming aspect of recovery procedures stems from the fact that the tools necessary to manage and run a system or domain are not in the failsafe boot image, typically a CD-ROM. The persistent storage on these mission critical systems is almost always managed using add-on or unbundled products not found in a straight installation of the Solaris Operating Environment. The use of volume managers such as Solstice DiskSuite software or Sun Enterprise Volume Manager (SEVM) software to manage the disk space add further complexity and time to an already complex and time consuming recovery procedure. Infrastructure tools of this sort are mandatory for the operation of any important UNIX system or data center, yet are not available on the CD-ROM so often used as fail-safe boot media to effect emergency recoveries.

It is this lack of tools in the fail-safe boot image, combined with the resulting need to manipulate, work around, or re-configure these tools from read-only CD media, that mandates multiple reboots and complexity in most recovery situations.

A conventional approach for a repair as straightforward as a tape recovery of `/etc/passwd` will require several reboots, involving a number of less-than-ideal procedures that are highly susceptible to error; such as having to temporarily un-mirror `rootvol`. The multiple reboots necessary to disable altogether or work around the infrastructure tools would make this recovery intolerably long. A fail-safe boot image which permitted the operation of volume managers and backup/recover clients would drastically reduce the number of reboots required to perform any sort of recovery.

The configuration and tools described herein permit just that. The MR system grew from a need to have a fail-safe boot device available for Starfire software domains which provided access to volume manager objects and the restore client GUI. The MR system has evolved somewhat away from a fixed set of tools for recovery - although they are certainly available. MR is more about the mechanism by which one may incorporate site or product specific tools into the fail-safe Solaris Operating Environment boot media.

MR configuration layers on top of a standard JumpStart software boot image found on the Solaris 2.6 Operating Environment installation media. This mini-root is then modified in a few key ways, the most important of which is the boot sequence. MR provides extra processing for keywords supplied as arguments to the OBP `boot` command. These arguments control which services start when the kernel reaches run level two before optionally running either a Korn shell or the auto-recover application. These keywords are used much the same way that `install` is used with JumpStart software:

```
ok boot net - install
... JumpStart Install ...

ok boot net - recover
... Solaris miniroot boots with MR in
interactive recovery mode ...
```

The `recover` keyword causes the booting client to start network services not normally started on a JumpStart software boot. In addition, the client will travel into run level two without starting `pinstall`, the default behavior for JumpStart software. A `recover` boot leaves the system at a Korn shell prompt.

Setting Up MR

The setup for an MR server begins as if for a JumpStart server. In fact, the two functions may be provided using the same boot image. MR is very careful to modify the JumpStart software boot image in such a way as to preserve the default behavior.

The modified image can still be used normally to boot, install and upgrade clients without requiring special procedures. In other words, JumpStart software still works.

Setting up for MR is merely a matter of installing a JumpStart server, then splicing in the behaviors you would like from this fail-safe boot image. There are four basic steps to accomplish this:

Copy the JumpStart server image to writable media.

Modify the start-up routines to accept extra boot arguments.

Install the required device drivers.

Install the MR application software.

Copy the JumpStart Server Image

MR modifies the JumpStart software mini-root file system, so that image must be on writable media. The easiest way to accomplish this is to simply copy the JumpStart CD to a file system on the administrative system which is to be the recovery master. In a Sun Enterprise 10000 environment, the SSP is the natural choice for this. Any server which satisfies the fundamental services required of the JumpStart server (RARP responder, bootparam responder, NFS service) may function as the MR server, even laptops running the x86 version of the Solaris Operating Environment. In fact virtually any UNIX variant, including Linux and Free-BSD, will satisfy these requirements. We will only discuss using a Solaris host as the JumpStart server.

```
# BASEDIR=/path/to/JS_Server/base
# export BASEDIR
# cd /cdrom/cdrom0/Solaris_2.6/Tools
# ./setup_install_server $BASEDIR
```

See the JumpStart AnswerBook™ for details as to how to configure a JumpStart server. There are a number of options available, but sticking with the defaults will give you the best results.

Modify the Boot Start-up

The processing for keywords specified at the end of the boot command line is embedded in the shell scripts launched by init. In specific, `/sbin/rcS` and `/sbin/startup` handle this. MR modifies these scripts such that additional

keywords are accepted and triggers set for the functions that they control. This is necessary because you want the host to travel beyond single user mode, yet not start the JumpStart install process `pfinstall`.

In addition, MR requires network services such as `inetd`. These are not started in a JumpStart software install, MR configures them to start in `/sbin/sysconfig`.

Install the Device Drivers

A number of additional device drivers are employed by MR in a recovery boot. The SEVM devices, for example, are activated in order to facilitate the manipulation of volumes while recover booted. Other useful drivers can also be installed to permit booting from additional network devices. SunFDDI™ is a popular example.

Most of these device drivers can be added using `add_drv`. Some additional configuration may be necessary to a few system and driver config files to make added drivers function correctly.

Install the Application Software

Any application software necessary for a recovery should be installed on the JumpStart software/MR image as if its root directory were the base of the OS installed in `$BASEDIR`. For generic applications, this will work very easily. A number of work arounds may be necessary to accommodate software which requires host-specific licenses, writable work space, or a work space in `/var`.

When booted off of the mini-root `/tmp` is the only filesystem mounted as a writable device by default. Every file or directory that must be writable at any stage during the boot is placed in a subdirectory of the temporary file system, usually as a subdirectory of `/tmp/root`. The original pathname (such as `/etc/hosts`) is a symbolic link to its writable equivalent in `/tmp/root`. This means that each such file must be built by a start script before it is needed.

UNIX expects `/var` (and in particular `/var/tmp`) to be a writable space. To provide this on a read-only root file system with no sub-mount point for `/var`, the mini-root uses a symbolic link approach similar to that used for writable files. `/var` is a symlink to `/tmp/root/var`, which must be built early in the boot process. This is accomplished by making a subtree copy of `/var` from a prototype stored in `/.tmp_proto`. This is copied into `/var/tmp` (using `cpio`) very early in `rcS` execution.

Pre-Packaged Modifications

The MR suite of packages is designed to handle the majority of the setup and customizations necessary in a consistent and automatic way. The MR system consists of the base software package (MRbase) and the applicable application packages. Each application is bundled into its own separately installable package in order to allow modular addition to the fail-safe boot image. You only need install those components necessary for your environment. However, installation of MRbase is required to use any of the MR application packages.

MRbase

MRbase modifies the mini-root start scripts described above. MRbase is a prerequisite for all other packages. The modifications that MRbase makes to the start-up scripts provides additional argument processing during run level 2 to detect and process the MR arguments to the booting kernel. MRbase retains the default ability to boot for installation services.

The second modification made to the boot image by MRbase is the addition of extra network device drivers. The standard boot image can only boot from a few network interface types. With the exception of SunATM™, this restriction is not due to any technical reasons, but because the non-bootable network interface types do not have drivers on the install CD. The network device drivers MRbase adds to the boot image is the `nfe` driver for the SunFDDI card. Booting from this network interface type still requires the adherence to the rules governing placement of RARP servers on the same subnet or having directed broadcasts configured on the router for that subnet.

Note that as the Solaris Operating Environment does not support booting from an SunATM interface, MRbase does not support booting from the SunATM interface either.

MR Application Packages

Currently, the MR application packages consist of:

MRsnb

MRsbu

MRphoton

MRfw

MRvx

MRsnb

MRsnb consists of the client portions of Sun StorEdge Enterprise NetBackup™ software, Version 3.1.1GA, necessary to recover files from a specified NetBackup storage server. Some modifications to the JumpStart software `/etc/inetd.conf` and `/etc/services` are made in order to support the NetBackup architecture.

MRsnb includes the client applications only, installed into `/usr/opensv` (relative to the JumpStart software OS root). No `man` pages or storage node software is installed.

MRsbu

MRsbu consists of Solstice™ Backup software (a.k.a. Networker), Version 5.1.Build.103. All client software necessary to recover files from a specified Networker storage server is installed in `/usr/lib/nsr`. No `man` pages or storage server software is included.

MRphoton

MRphoton packages a slightly modified version of StorTools™ software, Version 3.2 Beta, which will operate from a mini-root file system, allowing you to run its diagnostics to troubleshoot a FC-AL loop containing your usual boot device.

The mini-root read-only root filesystem necessitates that the device instance file be built and placed someplace other than `/etc`. Hence, the location of the device instance file is `/tmp/root/path_to_inst` when booted off of the mini-root image. StorTools has been modified to obtain the location of the instance file from the environment, rather than assuming a location of `/etc/path_to_inst`. Also, some subroutines in StorTools assume the presence of utilities not available in the mini-root. These subroutines were changed to allow MRphoton to operate.

MRfw

MRfw includes the firmware patches and download utilities for a number of disk devices. This permits easy upgrade of the system boot device while booted from fail-safe media. Some of Sun's disks and arrays have firmware fixes or upgrades in patch form. Because these firmware downloads usually require that the disk or device receiving the download be quiesced, downloading firmware to the boot disk can be a dangerous and frustrating exercise, if possible at all. MRfw provides the common firmware packages in `/opt/MRfw` of the mini-root, simplifying the upgrade of system disk firmware.

MRvx

MRvx provides the boot image with the ability to directly access and manipulate Veritas Volume Manager objects.

MRvx includes Sun Enterprise Volume Manager software, Version 2.6 (patched with 106606-02). This package includes everything normally found in the `SUNWvxvm` or `VRTSvxvm` packages installed on a host. However, no `man` pages or GUI utilities are included. MRvx includes the drivers and utilities normally found in SEVM, relocated into the correct places for the JumpStart image. In addition, MRvx provides a setup script which facilitates starting and using SEVM on a read-only root filesystem.

It is important to note that because the SEVM setup is done in the mini-root, some devices may be accessible via a different controller number than they are when booted off of the system's usual system disk, or they may be completely inaccessible. In general, any device that required changes to the `sd.conf` will be by usable by MRvx above lun 0 or may not be visible at all. Devices such as EMC or A3500's configured for multi-lun support are the most common devices that fall into this category.

MRvx provides a quick and ready method to boot off of a device other than the system disk and be able to mount and modify the root and `/usr` filesystems without having to un-encapsulate. Not only does this save you the time of having to un-encapsulate and then re-encapsulate and the several associated reboots involved in that procedure, it also saves you the time of a mirror resync. A resync is no longer necessary since MRvx allows you to work with the mirrored devices directly. Additionally, use of MRvx avoids the potential of human error during the complicated and often manual un-encapsulation/re-encapsulation procedure.

MRvx is also crucial for those instances when un-encapsulating is impossible or dangerous. If the system administrator has done anything that has invalidated the saved VTOC, such as resizing a filesystem on the system disk, replacing the encapsulated boot device, or moving a subdisk on the system disk, then the VTOC

saved by the encapsulation process no longer matches the slice boundaries on the system disk. In these situations un-encapsulation is impossible or runs a high probability of data loss or corruption.

Using MR

The MR software is, in most cases, used in precisely the same way from the fail-safe booted client as it is from any other client. The various MR packages should be installed on the JumpStart server much as the original CD image was. When the JumpStart server image was copied, a base directory was handed to `setup_install_server` in order to specify the location. That same location is given to MRbase during its installation so it may modify the correct files. Simply `pkgadd MRbase`, specifying that location when it is prompted for. The `$BASEDIR` value will then be stored in a state file so that subsequent MR installations may suggest an intelligent default for this location. See Appendix 1 for an example of the installation.

After all components are installed, you must add an install client just as you would for the JumpStart software, then boot that client with the correct argument to invoke the MR special features. See Appendix 1 for a sample boot in which MRvx is also started.

Acknowledgments

The MR system was conceived and developed by Gene Trantham. Thanks to Ken Kambic for suggesting the main SSP/spare SSP patching technique.

Appendix 1: Installing MR

```
# cd /cdrom/cdrom0/s0/Solaris_2.6/Tools
# ./setup_install_server /export/Install/2.6
Verifying target directory...
Calculating the required disk space for the Solaris_2.6 produc ...
Copying the CD image to disk...
Install Server setup complete
# cd /var/tmp
```

```

# ls
MRbase.pkg MRphoton.pkg MRsnb.pkg MRfw.pkg MRsbu.pkg MRvx.pkg
# pkgadd -d ./MRbase.pkg all
Processing package instance <MRbase> from </var/tmp/MRbase.pkg>
Recovery Tools for Solaris_2.6 CD Image
(sparc,i386) 1.1.8
I need to know the base directory of your Solaris 2.6 CD image.
This is the same directory that you specified when you created the install server
using 'setup_install_server'.
What is the base directory for your CD image [?,q] /export/Install/2.6
## Executing checkinstall script.
Using </export/Install/2.6/Solaris_2.6/Tools/Boot> as the package base directory.
## Processing package information.
## Processing system information. 5 package pathnames are already properly
installed.
## Verifying disk space requirements.
## Checking for conflicts with packages already installed.
## Checking for setuid/setgid programs.
This package contains scripts which will be executed with super-user permission
during the process of installing this package.
Do you want to continue with the installation of <MRbase> [y,n,?] y
Installing Recovery Tools for Solaris_2.6 CD Image as <MRbase>
## Installing part 1 of 1.
/export/Install/2.6/Solaris_2.6/Tools/Boot/kernel/drv/fddi
/export/Install/2.6/Solaris_2.6/Tools/Boot/kernel/drv/nf
[ verifying class <none> ]
Modifying /etc/.MR_BASEDIR_2.6
[ verifying class <build> ]
Modifying /export/Install/2.6/Solaris_2.6/Tools/Boot/sbin/rcS
Modifying /export/Install/2.6/Solaris_2.6/Tools/Boot/sbin/sysconfig [ verifying
class <sed> ]
## Executing postinstall script.
Reboot client to install driver.
Note: major number maximum based on server, not client
Installation of <MRbase> was successful.
# pkgadd -d ./MRvx.pkg all
Processing package instance <MRvx> from </var/tmp/MRvx.pkg>
VxVM Drivers for Solaris_2.6 CD Image (sparc,i386) 1.1.1
I need to know the base directory of your Solaris 2.6 CD image.
This is the same directory that you specified when you created the install server
using 'setup_install_server'.
What is the base directory for your CD image?
Default=/export/Install/2.6 [?,q]
## Executing checkinstall script.
Using </export/Install/2.6/Solaris_2.6/Tools/Boot> as the package base directory.
## Processing package information.
## Processing system information.

```

```
7 package pathnames are already properly installed.
## Verifying package dependencies.
## Verifying disk space requirements.
## Checking for conflicts with packages already installed.
## Checking for setuid/setgid programs.
The following files are being installed with setuid and/or setgid permissions: /
export/Install/2.6/Solaris_2.6/Tools/Boot/usr/sbin/vxprint <setuid root>
Do you want to install these as setuid/setgid files [y,n,?,q] y
This package contains scripts which will be executed with super-user permission
during the process of installing this package.
Do you want to continue with the installation of <MRvx> [y,n,?] y
Installing VxVM Drivers for Solaris_2.6 CD Image as <MRvx>
## Installing part 1 of 1.
```

... Long list of files installed ...

```
[ verifying class <none> ]
Modifying /export/Install/2.6/Solaris_2.6/Tools/Boot/etc/devlink.tab
Modifying /export/Install/2.6/Solaris_2.6/Tools/Boot/etc/system
[ verifying class <sed> ]
## Executing postinstall script.
Reboot client to install driver.
Note: major number maximum based on server, not client
Installation of <MRvx> was successful.
#
```

Sample Boot

```
<0> ok boot net - recover
Boot device: /sbus@3,0/SUNW,hme@3,8c00000 File and args: - recover 2ec00
hostname: argus
domainname: Itasca.Central.Sun.COM
root server: rtfm
root directory: /export/Install/2.6/Solaris_2.6/Tools/Boot
SunOS Release 5.6 Version Generic [UNIX(R) System V Release 4.0]
Copyright (c) 1983-1997, Sun Microsystems, Inc.
socal1: port 0: Fibre Channel Loop is ONLINE
socal0: port 0: Fibre Channel Loop is ONLINE
Configuring devices...
The system is coming up. Please wait.
Starting remote procedure call (RPC) services: sysidnis done.
Launching K shell...
#
# /sbin/vxstartup.sh
```

```

Enter the VxVM host ID: [argus]
If you do not have a photon
or SSA on this host, you
will need a temporary
license. Temp licenses may
be had at
http://storage.central/
nw-storage/html/vxvm/
vxvm.html
Do you need a license key? [yes] yes
Please enter your key: 1036 9944 3416 4989 0183 456
vxvm:vxserial: INFO: Feature name: CURRSET [95]
vxvm:vxserial: INFO: Number of licenses: 1 (non-floating
vxvm:vxserial: INFO: Expiration date: Sun Aug 01 03:00:00 1999 (35.7 days from
now)
vxvm:vxserial: INFO: Release Level: 20
vxvm:vxserial: INFO: Machine Class: All
vxvm:vxserial: INFO: Key successfully installed in /etc/vx/elm/95.
Do you need another license key? [no] no
Starting up VxVM . . . . done
Do you wish to start volumes in the rootdg disk group? [yes] yes
dg rootdg usetype fsgen: start var
dg rootdg usetype root: start rootvol
dg rootdg usetype swap: start swapvol
Do you wish to start volumes in all other disk groups? [no] no
You can start all volumes in all known disk groups using the command:
    vxrecover -svn
# fsck /dev/vx/rdisk/rootdg/rootvol
** /dev/vx/rdisk/rootdg/rootvol ** Last Mounted on /
** Phase 1 - Check Blocks and Sizes
** Phase 2 - Check Pathnames
**Phase 3 - Check Connectivity
** Phase 4 - Check Reference Counts
** Phase 5 - Check Cyl groups
28054 files, 449629 used, 661858 free (1226 frags, 82579 blocks, 0.1% fragmentation)
# mount /dev/vx/dsk/rootdg/rootvol
/a

```