Clustering Solaris 10 Zones with *RSF-I*

High-Availability

Technical Whitepaper

Clustering Solaris 10 Zones with *RSF-1* from High-Availability

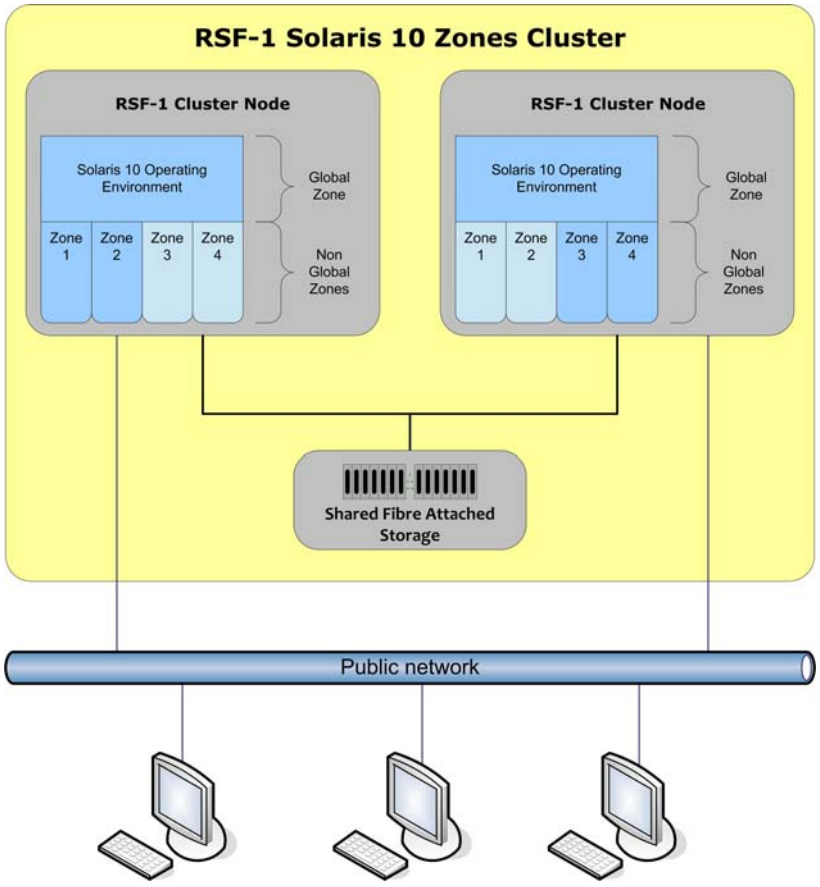
INTRODUCTION

The Solaris 10 Operating System implements a partitioning and virtualisation technology known as *zones* which provides a mini operating environment for running applications securely in isolation. This virtualisation is particularly suited to server consolidation projects, where each application can be configured to run concurrently in a different zone on a single server.

The main instance of the Solaris 10 Operating System which boots is referred to as the global zone; it is always running while the server is up, regardless of if zones are configured or not. Any zones created to run applications are non-global zones by their scope; the terms zone and non-global zone are used interchangeably.

For sites wishing to cluster zones, RSF-1 version v3.0.25 and above fully supports zone failover between separate instances of Solaris 10 global zones running on different servers to provide High Availability of critical applications running inside non-global zones.

Conceptually an RSF-1 zone cluster can be viewed as follows:



In clustered zones configurations, RSF-1 is installed on each server in the global zone. Each non-global zone is then configured as a separate RSF-1 zone service which can be independently failed over between global zones inside the cluster framework as required.

The zones themselves can be configured as either whole root or sparse, the cluster itself is not concerned with the specific model taken when building the clustered zone configurations. Whole root zones give maximum configuration flexibility but require the most amount of disk space, although this space requirement can be reduced by deploying ZFS filesystems[1].

To further increase resiliency, non-global zone IP addresses can optionally be placed inside IP Multipathing groups configured in the global zone to provide Network Adapter Failover functionality provided the server has an additional physical network interface installed and available specifically to be configured as a standby interface for redundancy.

ZONE CONFIGURATION

The following zone configuration is for a pair of RSF-1 clustered servers; `haserver01` and `haserver02`, each with a single non-global whole root zone installed locally; `vcprod01` on `haserver01` and `vcprod02` on `haserver02` respectively. Both zones in the pair are controlled by an RSF-1 zone service called `prod` running an application called Production.

The physical and logical network interfaces and associated IP addresses used for `haserver01`, `haserver02`, `vcprod01` and `vcprod02` including the `prod` service's `vhprod` virtual hostname are as follows:

haserver01, hme0: 193.168.10.10	haserver02, hme0: 192.168.10.20
vcprod01, hme0:1: 192.168.10.11	vcprod02, hme0:1: 192.168.10.21
vhprod, hme0:2: 192.168.10.30	

The first step is for the global zone administrator to create the `vcprod01` and `vcprod02` whole root zone configurations on `haserver01` and `haserver02` respectively. This also includes the `vhprod` virtual hostname ultimately used by clients to access the `prod` service over the network while running, plus the filesystem for the Production application which resides on shared storage, mounted under `/app` in the non-global zone on the server currently running the zone service.

The shared storage itself can be either SCSI-based or Fibre Channel, typically using either software-based mirroring or hardware RAID for redundancy, which can be mutipathed as appropriate for resiliency.

To avoid any virtual hostname or shared storage conflicts, RSF-1 guarantees the zone service will only ever be started on one server concurrently.

To begin creating the `vcprod01` zone configuration, run the `zonecfg` command from the global zone on `haserver01` as the root user, specifying the `zonepath` and `autoboot` property as follows:

```
[haserver01:/]# zonecfg -z vcprod01
vcprod01: No such zone configured
Use 'create' to begin configuring a new zone.
zonecfg:vcprod01> create
zonecfg:vcprod01> set zonepath=/cluster/vcprod01
zonecfg:vcprod01> set autoboot=false
```

The `zonepath` is the location in the `haserver01` global zone filesystem hierarchy where the `vcprod01` zone will be installed, in this example all clustered zones reside under the `/cluster` directory structure for clarity.

[1] For more information on ZFS, please refer to the documentation at http://docs.sun.com.

The autoboot property is comparable to the OBP[2] auto-boot? parameter and determines if the zone is to be booted automatically when the global zone is booted. As RSF-1 itself will control vcprod01 booting on haserver01 when starting the prod service, the autoboot property for all cluster controlled zones must explicitly be set to false.

Create the IP address configured by the global zone on haserver01 for network access to the vcprod01 zone as follows:

```
zonecfg:vcprod01> add net
zonecfg:vcprod01:net> set physical=hme0
zonecfg:vcprod01:net> set address=192.168.10.11
zonecfg:vcprod01:net> end
```

Add the vhprod virtual hostname to be used by clients to access the prod service when running to the configuration as follows:

```
zonecfg:vcprod01> add net
zonecfg:vcprod01:net> set physical=hme0
zonecfg:vcprod01:net> set address=192.168.10.30
zonecfg:vcprod01:net> end
```

When each IP address defined above for the vcprod01 zone is configured, Solaris will create a separate hme0 logical interface in the global zone for the zone to use, i.e. hme0:1, hme0:2, etc.

Add the Production application's /app filesystem residing on shared storage; this definition includes mount point, device special and raw files, filesystem type and mount options as follows:

```
zonecfg:vcprod01> add fs
zonecfg:vcprod01:fs> set dir=/app
zonecfg:vcprod01:fs> set special=/dev/dsk/c2t0d0s0
zonecfg:vcprod01:fs> set raw=/dev/rdsk/c2t0d0s0
zonecfg:vcprod01:fs> set type=ufs
zonecfg:vcprod01:fs> set options=logging
zonecfg:vcprod01:fs> end
```

The vcprod01 zone configuration on haserver01 now complete; run the verify command to verify the configuration is syntactically correct, followed by the commit command to write the in-memory configuration to disk as follows:

```
zonecfg:vcprod01> verify
zonecfg:vcprod01> commit
zonecfg:vcprod01> exit
```

From the global zone on haserver01, run the zoneadm command again to view the newly configured vcprod01 zone in the zone database as follows:

```
[haserver01:/]# zoneadm list -icv
  ID NAME             STATUS         PATH
   0 global           running        /
   - vcprod01         configured     /cluster/vcprod01
```

Next, from the global zone on haserver02, run the zonecfg command as the root user to begin creating the vcprod02 zone configuration as follows:

```
[haserver02:/]# zonecfg -z vcprod02
vcprod02: No such zone configured
Use 'create' to begin configuring a new zone.
zonecfg:vcprod02> create
```

[2] OpenBoot PROM, please refer to the documentation at http://docs.sun.com.

```
zonecfg:vcprod02> set zonepath=/cluster/vcprod02
zonecfg:vcprod02> set autoboot=false
zonecfg:vcprod02> add net
zonecfg:vcprod02:net> set physical=hme0
zonecfg:vcprod02:net> set address=192.168.10.21
zonecfg:vcprod02:net> end
zonecfg:vcprod02> add net
zonecfg:vcprod02:net> set physical=hme0
zonecfg:vcprod02:net> set address=192.168.10.30
zonecfg:vcprod02:net> end
zonecfg:vcprod02> add fs
zonecfg:vcprod02:fs> set dir=/app
zonecfg:vcprod02:fs> set special=/dev/dsk/c2t0d0s0
zonecfg:vcprod02:fs> set raw=/dev/rdsk/c2t0d0s0
zonecfg:vcprod02:fs> set type=ufs
zonecfg:vcprod02:fs> set options=logging
zonecfg:vcprod02:fs> end
zonecfg:vcprod02> verify
zonecfg:vcprod02> commit
zonecfg:vcprod02> exit
```

Finally, from the global zone on `haserver02`, run the `zoneadm` command to view the newly configured `vcprod02` zone in the zone database as follows:

```
[haserver02:/]# zoneadm list -icv
  ID NAME             STATUS         PATH
   0 global           running        /
   - vcprod02         configured     /cluster/vcprod02
```

ZONE INSTALLATION

Once the `vcprod01` and `vcprod02` zone configurations have been created on `haserver01` and `haserver02` respectively, the zones need to be installed, i.e. files copied from the global zone, before they can be booted.

From the global zone on `haserver01`, run the `zoneadm` command as the root user to install the `vcprod01` zone as follows:

```
[haserver01:/]# zoneadm -z vcprod01 install
Preparing to install zone <vcprod01>.
Checking <ufs> file system on device </dev/rdsk/c2t0d0s0> to be
mounted at </cluster/vcprod01/root>
Creating list of files to copy from the global zone.
Copying <77290> files to the zone.
Initializing zone product registry.
Determining zone package initialization order.
Preparing to initialize <1096> packages on the zone.
Initialized <1096> packages on zone.
Zone <vcprod01> is initialized.
The file </cluster/vcprod01/root/var/sadm/system/logs/install_log>
contains a log of the zone installation.
```

It can take a long time for zone installation to complete, especially as the zone is configured as a whole root zone.

Once the `vcprod01` zone installation is complete, run the `zoneadm` command again from the global zone on `haserver01` to view the newly installed `vcprod01` zone in the zone database as follows:

```
[haserver01:/]# zoneadm list -icv
  ID NAME             STATUS         PATH
   0 global           running        /
```

```
   - vcprod01         installed      /cluster/vcprod01
```

Next, from the global zone on haserver02, run the zoneadm command as the root user to install the vcprod02 zone as follows:

```
[haserver02:/]# zoneadm -z vcprod02 install
Preparing to install zone <vcprod02>.
Checking <ufs> file system on device </dev/rdsk/c2t0d0s0> to be
mounted at </cluster/vcprod02/root>
Creating list of files to copy from the global zone.
Copying <77290> files to the zone.
Initializing zone product registry.
Determining zone package initialization order.
Preparing to initialize <1096> packages on the zone.
Initialized <1096> packages on zone.
Zone <vcprod02> is initialized.
The file </cluster/vcprod02/root/var/sadm/system/logs/install_log>
contains a log of the zone installation.
```

Finally, once the vcprod02 zone installation is complete, run the zoneadm command again from the global zone on haserver02 to view the newly installed vcprod02 zone in the zone database as follows:

```
[haserver02:/]# zoneadm list -icv
  ID NAME             STATUS         PATH
   0 global           running        /
   - vcprod02         installed      /cluster/vcprod02
```

ZONE BOOTING

Once the vcprod01 and vcprod02 zones have been installed by the global zone administrator on haserver01 and haserver02 respectively, the zones are in an unconfigured state, similar to when the global zone is first installed.

The zones can now be booted with the zoneadm command to finish the zone configuration. Since this is the first time the zones have been booted after installation, the normal Solaris sysid tools need to be run to complete the zone configuration[3].

From the global zone on haserver01, run the zlogin command to access the vcprod01 zone console as follows[4]:

```
[haserver01:/]# zoneadm -z vcprod01 boot
[haserver01:/]# zlogin -C vcprod01
[Connected to zone 'vcprod01' console]
SunOS Release 5.10 Version Generic 64-bit
Copyright 1983-2005 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.
Hostname: vcprod01
```

At this point, the normal system identification process for a freshly installed Solaris OS instance is started on the vcprod01 zone. After system identification is complete and the non-global zone root user's password is set, the zone is up and ready for use.

To disconnect from the console, use the command sequence: ~. (tilde dot). The vcprod01 zone can now be accessed over the network using the telnet, rlogin, or ssh commands, just like a standard Solaris OS system.

[3] For more information, please refer to the sysidcfg man page.
[4] Unless the ${zonepath}/root/etc/default/login file is modified, only the root user can login at the console.

From the global zone on haserver01, run the zoneadm command again to view the vcprod01 zone running in the zone database as follows:

```
[haserver01:/]# zoneadm list -icv
  ID NAME             STATUS         PATH
   0 global           running        /
   - vcprod01         running        /cluster/vcprod01
```

Shutdown the vcprod01 zone from the global zone on haserver01 in preparation for booting the vcprod02 zone on haserver02 with the following command:

```
[haserver01:/]# zoneadm -z vcprod01 halt
```

Next, from the global zone on haserver02, run both the zoneadm and zlogin commands again above to boot the vcprod02 zone and repeat the post installation configuration procedure as follows:

```
[haserver02:/]# zoneadm -z vcprod02 boot
[haserver02:/]# zlogin -C vcprod02
[Connected to zone 'vcprod02' console]
SunOS Release 5.10 Version Generic 64-bit
Copyright 1983-2005 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.
Hostname: vcprod02
# ~.
```

From the global zone on haserver02, run the zoneadm command again to view the vcprod02 zone running in the zone database as follows:

```
[haserver02:/]# zoneadm list -icv
  ID NAME             STATUS         PATH
   0 global           running        /
   - vcprod02         running        /cluster/vcprod02
```

Finally, shutdown the vcprod01 zone from the global zone on haserver01 with the following command:

```
[haserver02:/]# zoneadm -z vcprod02 halt
```

RSF-1 Integration

Once the zone configuration and installation is complete, the next task is to integrate the vcprod01 and vcprod02 zones into the RSF-1 prod zone service to provide automatic failover capability for the Production application.

Referring to the RSF-1 Administrators and Quick Start guides as appropriate, install and license RSF-1 in the global zone on both haserver01 and haserver02, then create the RSF-1 configuration file by referring to the relevant sections in the documentation as necessary.

When defining the RSF-1 zone data service, the corresponding section in the configuration file will be as follows:

```
SERVICE prod vhprod "RSF-1 Production Zone Service"
  INITIMEOUT 60
  RUNTIMEOUT 60
  IPDEVICE none
 SERVER haserver01
 SERVER haserver02
```

Once the RSF-1 configuration file is complete, the next step is to create the `prod` service script directory on both `haserver01` and `haserver02` for RSF-1 to use when starting, stopping or failing over the zone service as follows:

```
[haserver01:/]# mkdir /opt/HAC/RSF-1/etc/rc.prod.d
```

To control zone booting and shutdown, copy the example zone control script `/opt/HAC/RSF-1/etc/service/scripts/S30zone.zonename` to the zone service script directory created above, renaming the script to `S30zone.vcprod01` on `haserver01` and `S30zone.vcprod02` on `haserver02`, modifying the zone name contained within the script to the same name as the zone being controlled as appropriate.

Finally on both `haserver01` and `haserver02`, run the RSF-1 `rsfklink` command to automatically generate the corresponding `K70zone.<zonename>` symbolic links to shut down the corresponding zone and also add entries for the `vhprod` virtual hostname to `/etc/hosts` in the global zone.

IPMP Integration – Optional

Regardless of whether zones are configured or not, a server with only a single physical network interface installed is more vulnerable to certain types of network outage, such as switch or local link failure. These outages can prevent clients from accessing services running on the server over the network.

Solaris IP Multipathing is a technology designed to increase a server's resiliency against network outages by monitoring interfaces and failing over IP addresses provided an additional standby network interface is installed and available specifically to be used for redundancy[5].

If IPMP is configured on a server, all non-global zone IP addresses will be automatically failed over to the standby interface if corresponding IPMP groups in the global zone exist, even though the standby interface used for Network Adapter Failover does not appear in the associated zone configurations.

To setup IPMP from the global zone on `haserver01`, assuming two physical network interfaces, `hme0` and `hme1` (standby interface), configured in an IPMP group called `public` with the corresponding test addresses `192.168.1.12` and `192.168.1.13` on the public network respectively, the corresponding IPMP network interface configuration would appear as follows:

```
[haserver01:/]# cat /etc/hostname.hme0
haserver01 group public up \
addif 192.168.1.12 deprecated -failover up

[haserver01:/]# cat /etc/hostname.hme1
192.168.1.13 group public deprecated -failover standby up
```

Conversely, the network interface configuration files on `haserver02` using the corresponding IPMP test addresses `192.168.1.21` and `192.168.1.22` would appear as follows:

```
[haserver2:/]# cat /etc/hostname.hme0
haserver02 group public up \
addif 192.168.1.22 deprecated -failover up

[haserver02:/]# cat /etc/hostname.hme1
192.168.1.23 group public deprecated -failover standby up
```

Zone Maintenance

As each RSF-1 zone service consists of a pair of zones, one zone per server, it's often desirable to be able to boot the non-running zone in the pair for maintenance purposes while

[5] For further information, please refer to the IP Network Multipathing Administration Guide.

the zone service is already running on the other server, i.e. to apply patches or to install new software – otherwise the running zone service would have to be stopped and manually failed over to allow the other zone in the pair to be booted before patching could be performed, etc.

The problem here is without additional safeguards, booting a non-running zone manually while the zone service is already running on the other server would effectively configure the associated virtual hostname and application filesystems residing on shared storage twice, leading to the very real possibility of data corruption occurring[6].

To boot non-running zones for maintenance purposes without affecting the other zone in the pair which may already be running as a zone service, High-Availability provide a zone maintenance utility which can be integrated as an additional service on each server in the cluster.

When booting a zone on a server, either under cluster control as an RSF-1 zone service or manually using the `zoneadm` command, the zone configuration is parsed from a corresponding XML file located in `/etc/zones` which lists all properties and attributes for the zone, including the associated zone service's virtual hostname and application filesystems residing on shared storage.

When the zone maintenance utility is installed, a second set of zone XML configuration files are created by the global zone administrator with the virtual hostname and shared storage application filesystems explicitly removed, these maintenance versions are then swapped in by the server's zone maintenance service for all non-running zones locally before booting them safely into maintenance mode with all shared resources excluded[7]. This removes any possibility of conflicting with the other zone in the pair which may already be running as a zone service on the other server under cluster control.

To safeguard the integrity of both sets of XML configuration files, part of the zone maintenance service's installation procedure requires checksums to be calculated on both versions and stored in a table for later reference, this ensures only the correct XML file, i.e. maintenance version, is ever used to boot a zone outside of the cluster framework.

When each server is booted, RSF-1 sets it's own zone maintenance service explicitly to blocked mode to prevent it from starting automatically[8]; this is to ensure maintenance mode is only entered when specifically desired and never when a server reboots in automatic mode, requiring an administrator to unblock the service before maintenance mode can be entered for all non-running zones locally.

In addition, when starting the zone maintenance service on a server, all zones which are booted locally into maintenance mode also have their corresponding RSF-1 zone service set to blocked mode; this is to explicitly prevent any attempts being made to restart those zones locally as a zone service under cluster control while they are already running in maintenance mode outside the cluster framework[9].

For assistance in installing and configuring the zone maintenance utility and associated services, please contact High-Availability.

[6] RSF-1 explicitly prevents a cluster controlled zone service from starting each zone concurrently on both servers via the cluster framework.
[7] Additional zone maintenance service scripts can be created to disable any non-global zone application start scripts via the `zonepath` from the global zone.
[8] For more information on blocked mode, please refer to the RSF-1 Administrators guide.
[9] It's important to note the blocked mode explicitly prevents an RSF-1 service from starting regardless of being set to automatic switchover mode.