Creating and Using Oracle® Solaris Zones

Part No: E54752
April 2019

ORACLE

Creating and Using Oracle Solaris Zones

Part No: E54752

Référence: E54752

Accès aux services de support Oracle

Les clients Oracle qui ont souscrit un contrat de support ont accès au support électronique via My Oracle Support. Pour plus d'informations, visitez le site `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` ou le site `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` si vous êtes malentendant.

Contents

Using This Documentation

- **Overview** – Describes how to set up and use the Oracle Solaris Zones feature and related resource management capabilities
- **Audience** – Technicians, system administrators, and authorized service providers
- **Required knowledge** – Experience administering Oracle Solaris environments. Experience with virtualized environments is a plus.

Product Documentation Library

Documentation and resources for this product and related products are available at http://www.oracle.com/pls/topic/lookup?ctx=E53394-01.

Feedback

Provide feedback about this documentation at http://www.oracle.com/goto/docfeedback.

♦ ♦ ♦ CHAPTER 1

How to Plan and Configure Non-Global Zones

This chapter describes what you need to do before you can configure a zone on your system. This chapter also describes how to configure a zone, modify a zone configuration, and delete a zone configuration from your system.

For an introduction to the zone configuration process, see Chapter 1, “Configuration Resources for Non-Global Zones” in *Oracle Solaris Zones Configuration Resources*.

For information about `solaris10` branded zone configuration, see *Creating and Using Oracle Solaris 10 Zones*.

Evaluating the Current System Setup

Zones can be used on any machine that runs the Oracle Solaris 10 or later release. The following primary system considerations are associated with the use of zones.

- The performance requirements of the applications running within each zone.
- The availability of disk space to hold the files that are unique within each zone.

Disk Space Requirements

There are no limits on how much disk space can be consumed by a zone. The global administrator or a zone administrator with appropriate authorizations is responsible for space restriction. The global administrator must ensure that local or shared storage is sufficient to hold a non-global zone's root file system. Even a small uniprocessor system can support a number of zones running simultaneously.

The nature of the packages installed in the non-global zone affects the space requirements of the zone. The number of packages is also a factor.

The disk requirements are determined by the disk space used by the packages currently installed in the global zone and the installed software.

A zone requires a minimum of 150 megabytes of free disk space per zone. However, the free disk space needed is generally from 500 megabytes to 1 gigabyte when the global zone has been installed with all of the standard Oracle Solaris packages. That figure can increase if more software is added.

An additional 40 megabytes of RAM for each zone are suggested, but not required on a system with sufficient swap space.

Restricting Zone Size

You can use ZFS dataset quotas with zones that have `zonepaths` backed by ZFS datasets to restrict zone size. Administrators that can access `zonepath` datasets can modify the datasets' `quota` and`reservation` properties to control the maximum amount of disk space that each zone can consume. These properties are described in the `zfs`(1M) man page.

Administrators can also create ZFS volumes with fixed sizes and install zones in the volume's datasets. The volumes limit the sizes of the zones installed within them.

Determine the Zone Host Name and the Network Requirements

You must determine the host name for the zone.

Inside an exclusive-IP zone, you configure addresses as you do for the global zone.

For a shared-IP zone that will have network connectivity, you must do one of the following:

- Assign an IPv4 address for the zone
- Manually configure and assign an IPv6 address for the zone

For more information on exclusive-IP and shared-IP types, see “Zone Network Interfaces” in *Oracle Solaris Zones Configuration Resources*.

Zone Host Name

If you are using the NIS or DNS name services, or the LDAP directory service, then the host information is stored in a database, such as *hosts.byname*, on a server.

If you use local files for the naming service, the `hosts` database is maintained in the `/etc/inet/hosts` file. The host names for zone network interfaces are resolved from the local `hosts` database in `/etc/inet/hosts`. Alternatively, for shared-IP zones, the IP address itself can be specified directly when configuring a zone so that no host name resolution is required. See the `hosts(4)` and `nodename(4)` man pages for more information. Also see Chapter 3, "Configuring and Administering IP Interfaces and Addresses in Oracle Solaris" in *Configuring and Managing Network Components in Oracle Solaris 11.3*.

Shared-IP Zone Network Address

Each shared-IP zone that requires network connectivity has one or more unique IP addresses. Both IPv4 and IPv6 addresses are supported.

IPv4 Zone Network Address

If you are using IPv4, obtain an address and assign the address to the zone. When you assign addresses to the zone, you can specify the address by using CIDR notation, such as `192.0.2.0/24`.

For shared-IP zones, the IP address itself can be specified directly when configuring a zone so that no host name resolution is required.

For more information, see `hosts(4)`, `netmasks(4)`, and `nodename(4)`.

IPv6 Zone Network Address

If you are using IPv6, you must manually configure the address. Typically, at least the following two types of addresses must be configured:

Link-local address

A link-local address is of the form `fe80::`*64-bit interface ID*`/10`. The `/10` indicates a prefix length of 10 bits.

Global unicast address

A global unicast address is based off a 64–bit prefix that the administrator configures for each subnet, and a 64-bit interface ID. The prefix can be obtained by running the `ipadm show-addr` command on any system on the same subnet that has been configured to use IPv6.

The 64–bit interface ID is typically derived from a system's MAC address. For zones use, an alternate address that is unique can be derived from the global zone's IPv4 address by using the following convention:

```
16 bits of zero:upper 16 bits of IPv4 address:lower 16 bits of IPv4
address:a zone-unique number
```

Assume that the global zone's IPv4 address is 192.0.2.10. This address is converted to hexadecimal as follows:

- 192 = c0
- 0 = 0
- 2 = 2
- 10 = 0a

Thus, a suitable link-local address for a non-global zone using a zone-unique number of `1` is `fe80::c00:c80a:1/10`.

If the global prefix in use on that subnet is `2001:0db8:aabb:ccdd/64`, a unique global unicast address for the same non-global zone is `2001:0db8:aabb:ccdd::c0a8:c80a:1/64`. Note that you must specify a prefix length when configuring an IPv6 address.

For more information about link-local and global unicast addresses, see the `ipadm`(1M) and `inet6`(7P) man pages.

Exclusive-IP Zone Network Address

Inside an exclusive-IP zone, configure addresses as you do for the global zone. Note that DHCP and IPv6 stateless address autoconfiguration can be used to configure addresses. For information about IP address configuration, see Chapter 3, “Configuring and Administering IP Interfaces and Addresses in Oracle Solaris” in *Configuring and Managing Network Components in Oracle Solaris 11.3*.

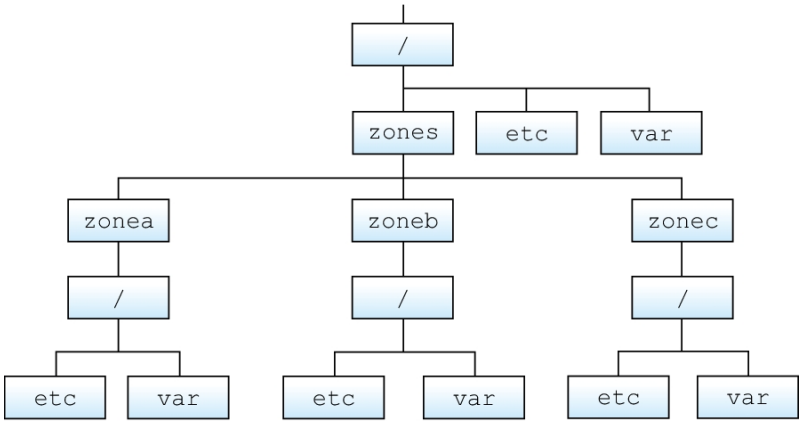
File System Configuration

You can specify a number of mounts to be performed when the virtual platform is set up. File systems that are loopback-mounted into a zone by using the loopback virtual file system

(LOFS) file system should be mounted with the `nodevices` option. For information on the `nodevices` option, see "File Systems and Non-Global Zones" on page 127.

LOFS lets you create a new virtual file system so that you can access files by using an alternative path name. In a non-global zone, a loopback mount makes the file system hierarchy look as though it is duplicated under the zone's root. In the zone, all files will be accessible with a path name that starts from the zone's root. LOFS mounting preserves the file system name space.

FIGURE 1 Loopback-Mounted File Systems



See the `lofs`(7FS) man page for more information.

Creating, Revising, and Deleting Non-Global Zone Configurations

Task	Description	For Instructions
Configure a non-global zone.	Use the `zonecfg` command to create a zone, verify the configuration, and commit the configuration. You can also use a script to configure and boot multiple zones on your system.	"Configuring, Verifying, and Committing a Zone" on page 22

Task	Description	For Instructions
	You can use the `zonecfg` command to display the configuration of a non-global zone.	
Modify a zone configuration.	Use these procedures to modify a resource type in a zone configuration, modify a property type such as the name of a zone, or add a dedicated device to a zone.	"Using the `zonecfg` Command to Modify a Zone Configuration" on page 30
Revert a zone configuration or delete a zone configuration.	Use the `zonecfg` command with the `revert` subcommand to undo a resource setting made to a zone configuration or to delete a zone configuration.	"Using the `zonecfg` Command to Revert or Remove a Zone Configuration" on page 34
Delete a zone configuration.	Use the `zonecfg` command with the `delete` subcommand to delete a zone configuration from the system.	"How to Delete a Zone Configuration" on page 36

Configuring, Verifying, and Committing a Zone

The `zonecfg` command described in the `zonecfg(1M)` man page is used to perform the following actions.

- Create the zone configuration.
- Verify that all required information is present.
- Commit the non-global zone configuration.

The `zonecfg` command can also be used to persistently specify the resource management settings for the global zone.

While configuring a zone with the `zonecfg` utility, you can use the `revert` subcommand to undo the setting for a resource. See "How to Revert a Zone Configuration" on page 35.

To display a non-global zone's configuration, see "How to Display the Configuration of a Non-Global Zone" on page 29.

▼ How to Configure the Zone

Note that the only required elements to create a non-global zone are the `zonename` and `zonepath` properties for zones with a `rootzpool` resource. Other resources and properties are optional. Some optional resources also require choices between alternatives, such as the decision to

use either the `dedicated-cpu` resource or the `capped-cpu` resource. See "Zone Configuration Data" in *Oracle Solaris Zones Configuration Resources* for information on available `zonecfg` properties and resources.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Set up a zone configuration with the zone name you have chosen.**

   The name `my-zone` is used in this example procedure.

   ```
   global$ zonecfg -z my-zone
   ```

   If this is the first time you have configured this zone, you will see the following system message:

   ```
   Use 'create' to begin configuring a new zone.
   ```

3. **Create the new zone configuration.**

   This procedure uses the default settings.

   ```
   zonecfg:my-zone> create
   create: Using system default template 'SYSdefault'

   zonecfg:my-zone> info
       zonename: my-zone
        zonepath.template: /system/zones/%{zonename}
        zonepath: /system/zones/my-zone
     ...
   ```

   Note - If you choose a path for your zone by using `set zonepath=`, the zone must reside on a ZFS dataset. The ZFS dataset will be created automatically when the zone is installed or attached. If a ZFS dataset cannot be created, the zone will not install or attach. Note that if the parent directory of the zone path exists, it must be the mount point of a mounted dataset.

4. **Set the autoboot value.**

   If set to `true`, the zone is automatically booted when the global zone is booted. The default value is `false`. Note that for the zones to autoboot, the zones service `svc:/system/zones:default` must also be enabled. This service is enabled by default.

   ```
   zonecfg:my-zone> set autoboot=true
   ```

5. **Set persistent boot arguments for a zone.**

   ```
   zonecfg:my-zone> set bootargs="-m verbose"
   ```

6. **Dedicate one CPU to this zone.**

```
zonecfg:my-zone> add dedicated-cpu
```

   a. **Set the number of CPUs.**

```
zonecfg:my-zone:dedicated-cpu> set ncpus=1-2
```

   b. **(Optional) Set the importance.**

```
zonecfg:my-zone:dedicated-cpu> set importance=10
```

   The default is 1.

   c. **End the specification.**

```
zonecfg:my-zone:dedicated-cpu> end
```

7. **Revise the default set of privileges.**

```
zonecfg:my-zone> set limitpriv="default,sys_time"
```

   This line adds the ability to set the system clock to the default set of privileges.

8. **Set the scheduling class to `FSS`.**

```
zonecfg:my-zone> set scheduling-class=FSS
```

9. **Add a memory cap.**

```
zonecfg:my-zone> add capped-memory
```

   a. **Set the memory cap.**

```
zonecfg:my-zone:capped-memory> set physical=1g
```

   b. **Set the swap memory cap.**

```
zonecfg:my-zone:capped-memory> set swap=2g
```

   c. **Set the locked memory cap.**

```
zonecfg:my-zone:capped-memory> set locked=500m
```

   d. **End the memory cap specification.**

```
zonecfg:my-zone:capped-memory> end
```

Note - To use the `capped-memory` resource, the `resource-cap` package must be installed in the global zone.

10. Add a file system.

```
zonecfg:my-zone> add fs
```

a. Set the mount point for the file system, `/usr/local` in this procedure.

```
zonecfg:my-zone:fs> set dir=/usr/local
```

b. Specify that `/opt/local` in the global zone is to be mounted as `/usr/local` in the zone being configured.

```
zonecfg:my-zone:fs> set special=/opt/local
```

In the non-global zone, the `/usr/local` file system will be readable and writable.

c. Specify the file system type, `lofs` in this procedure.

```
zonecfg:my-zone:fs> set type=lofs
```

The type indicates how the kernel interacts with the file system.

d. End the file system specification.

```
zonecfg:my-zone:fs> end
```

This step can be performed more than once to add more than one file system.

11. Set the `hostid` if necessary.

```
zonecfg:my-zone> set hostid=80f0c086
```

12. Add a ZFS dataset named *sales* in the storage pool *tank*.

```
zonecfg:my-zone> add dataset
```

a. Specify the path to the ZFS dataset *sales*.

```
zonecfg:my-zone> set name=tank/sales
```

b. End the dataset specification.

```
zonecfg:my-zone> end
```

The only dataset type that should be used with a dataset resource is a ZFS file system. The zone administrator can create child file systems and clones of its descendants. The zone administrator can modify properties of the dataset, and control compression and encryption.

13. **Create an exclusive-IP zone with an automatic VNIC.**

```
zonecfg:my-zone> set ip-type=exclusive
zonecfg:my-zone> add anet
```

    a. **Specify `auto` as the underlying link for the link to be created.**

```
zonecfg:my-zone:anet> set lower-link=auto
```

    The `zoneadmd` daemon automatically selects the link over which the VNIC will be created each time the zone boots. The IPoIB links are skipped when selecting the datalink.

    b. **End the specification.**

```
zonecfg:my-zone:anet> end
```

14. **Add a device.**

```
zonecfg:my-zone> add device
```

    a. **Set the device match, `/dev/sound/*` in this procedure.**

```
zonecfg:my-zone:device> set match=/dev/sound/*
```

    b. **End the device specification.**

```
zonecfg:my-zone:device> end
```

This step can be performed more than once to add more than one device.

15. **Add Open Fabrics User Verbs (OFUV) devices for components of OFUV other than IB diagnostic tools.**

```
zonecfg:my-zone> add device
```

    a. **Set the device match, `infiniband/ofs/*` in this procedure.**

```
zonecfg:my-zone:device> set match=infiniband/ofs/*
```

    b. **End the device specification.**

```
zonecfg:my-zone:device> end
```

IB diagnostic tools are not supported in non-global zones. Devices added can be used with components of OFUV, such as verbs and `rdma_cm`.

This step can be performed more than once to add more than one device.

16. **Add OFUV devices for components of OFUV other than IB diagnostic tools.**

```
zonecfg:my-zone> add device
```

   a. **Set the device match, `infiniband/hca/*` in this procedure.**

```
zonecfg:my-zone:device> set match=infiniband/hca/*
```

   b. **End the device specification.**

```
zonecfg:my-zone:device> end
```

IB diagnostic tools are not supported in non-global zones. Devices added can be used with components of OFUV, such as verbs and `rdma_cm`.

This step can be performed more than once to add more than one device.

17. **To allow disk labeling with the `format` command, an entire disk/LUN should be delegated to a zone, and the `allow-partition` property should be set.**

```
zonecfg:my-zone> add device
```

   a. **Set the device match, `/dev/*dsk/c2t40d3*` in this procedure.**

```
zonecfg:my-zone:device> set match=/dev/*dsk/c2t40d3*
```

   b. **Set `allow-partition` to be `true`.**

```
zonecfg:my-zone:device> set allow-partition=true
```

   c. **End the device specification.**

```
zonecfg:my-zone:device> end
```

This step can be performed more than once to add more than one device.

18. **To allow `uscsi` operations on a disk, the `allow-raw-io` property should be set.**

```
zonecfg:my-zone> add device
```

a. **Set the device match, `/dev/*dsk/c2t40d3*` in this procedure.**

```
zonecfg:my-zone:device> set match=/dev/*dsk/c2t40d3*
```

b. **Set `allow-raw-io` to be `true`.**

```
zonecfg:my-zone:device> set allow-raw-io=true
```

c. **End the device specification.**

```
zonecfg:my-zone:device> end
```

Caution - Allowing a zone to perform `uscsi` operations on a disk also allows the zone to access any other device connected to the same bus as the disk. Therefore, enabling this capability could create a security risk and allow for attacks against the global zone or other zones that use resources on the same bus. For more information, see `uscsi`(7I).

This step can be performed more than once to add more than one device.

19. **Add a zone-wide resource control by using the property name.**

```
zonecfg:my-zone> set max-sem-ids=10485200
```

This step can be performed more than once to add more than one resource control.

20. **Add a comment by using the `attr` resource type.**

```
zonecfg:my-zone> add attr
```

a. **Set the name to `comment`.**

```
zonecfg:my-zone:attr> set name=comment
```

b. **Set the type to `string`.**

```
zonecfg:my-zone:attr> set type=string
```

c. **Set the value to a comment that describes the zone.**

```
zonecfg:my-zone:attr> set value="This is my work zone."
```

d. **End the `attr` resource type specification.**

```
zonecfg:my-zone:attr> end
```

21. **Verify the zone configuration for the zone.**

```
zonecfg:my-zone> verify
```

22. **Commit the zone configuration for the zone.**

```
zonecfg:my-zone> commit
```

23. **Exit the `zonecfg` command.**

```
zonecfg:my-zone> exit
```

Note that even if you did not explicitly type `commit` at the prompt, a `commit` is automatically attempted when you type `exit` or an EOF occurs.

Using Multiple Subcommands From the Command Line

Tip - The `zonecfg` command also supports multiple subcommands, quoted and separated by semicolons, from the same shell invocation.

```
global$ zonecfg -z my-zone "create ; set zonepath=/zones/my-zone"
```

For shared-IP zones, a static address can only be assigned in a `zonecfg net` resource. It cannot be supplied on the command line.

Where to Go From Here

See "Installing and Booting Zones" on page 50 to install your committed zone configuration.

▼ How to Display the Configuration of a Non-Global Zone

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Display the configuration of a zone.**

```
global$ zonecfg -z zonename info
```

Using the zonecfg Command to Modify a Zone Configuration

You can also use the zonecfg command to do the following:

- Modify a resource type in a zone configuration
- Clear a property value in a zone configuration
- Add a dedicated device to a zone
- Modify a zone's privilege set
- Add and remove storage

▼ How to Modify a Resource Type in a Zone Configuration

You can select a resource type and modify the specification for that resource.

1. **Become a zone administrator.**
   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Select the zone to be modified, `my-zone` in this procedure.**

   ```
   global$ zonecfg -z my-zone
   ```

3. **Select the resource type to be changed, for example, a resource control.**

   ```
   zonecfg:my-zone> select rctl name=zone.cpu-shares
   ```

4. **Remove the current value.**

   ```
   zonecfg:my-zone:rctl> remove value (priv=privileged,limit=20,action=none)
   ```

5. **Add the new value.**

   ```
   zonecfg:my-zone:rctl> add value (priv=privileged,limit=10,action=none)
   ```

6. **End the revised `rctl` specification.**

   ```
   zonecfg:my-zone:rctl> end
   ```

7. **Commit the zone configuration for the zone.**

   ```
   zonecfg:my-zone> commit
   ```

8. **Exit the `zonecfg` command.**

```
zonecfg:my-zone> exit
```

Note that even if you did not explicitly type `commit` at the prompt, a `commit` is automatically attempted when you type `exit` or an EOF occurs.

Committed changes made through `zonecfg` take effect the next time the zone is booted.

How to Clear a Property in a Zone Configuration

Use this procedure to reset a standalone property.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Select the zone to be modified, `my-zone` in this procedure.**

   ```
   global$ zonecfg -z my-zone
   ```

3. **Clear the property to be changed, the existing pool association in this procedure.**

   ```
   zonecfg:my-zone> clear pool
   ```

4. **Commit the zone configuration for the zone.**

   ```
   zonecfg:my-zone> commit
   ```

5. **Exit the `zonecfg` command.**

   ```
   zonecfg:my-zone> exit
   ```

   Note that even if you did not explicitly type `commit` at the prompt, a `commit` is automatically attempted when you type `exit` or an EOF occurs.

   Committed changes made through `zonecfg` take effect the next time the zone is booted.

How to Rename a Zone by Using the `zonecfg` Command

Use this procedure to rename zones that are in either the configured state or the installed state.

Note that zones with either `rootzpool` or `zpool` resources cannot be renamed in the installed state because the `zonename` is part of the existing zpool name. To rename these zones, see "*Renaming Zones on Shared Storage*" at the end of this procedure.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Select the zone to be renamed, `my-zone` in this procedure.**

   ```
   global$ zonecfg -z my-zone
   ```

3. **Change the name of the zone, for example, to `newzone`.**

   ```
   zonecfg:my-zone> set zonename=newzone
   ```

4. **Commit the change.**

   ```
   zonecfg:newzone> commit
   ```

5. **Exit the `zonecfg` command.**

   ```
   zonecfg:newzone> exit
   ```

   Committed changes made through `zonecfg` take effect the next time the zone is booted.

▼ How to Add a Dedicated Device to a Zone

Use the following specification to place a scanning device in a non-global zone configuration.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Select the zone for the device.**

   ```
   global$ zonecfg -z my-zone
   ```

3. **Add a device.**

   ```
   zonecfg:my-zone> add device
   ```

4. **Set the device match, `/dev/scsi/scanner/c3t4*` in this procedure.**

```
zonecfg:my-zone:device> set match=/dev/scsi/scanner/c3t4*
```

5. **Set the device match, `/dev/scsi/scanner/c3t4*` in this procedure.**

```
zonecfg:my-zone:device> set match=/dev/scsi/scanner/c3t4*
```

6. **End the device specification.**

```
zonecfg:my-zone:device> end
```

7. **Exit the `zonecfg` command.**

```
zonecfg:my-zone> exit
```

How to Set `zone.cpu-shares` in the Global Zone

Use this procedure to persistently set shares in the global zone.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Ensure that the resource-mgmt service is enabled.**

```
$ svcs svc:/system/resource-mgmt:default
STATE          STIME    FMRI
disabled       Jan_08   svc:/system/resource-mgmt:default
```

   If the service is not online, enable it and check again.

```
$ svcadm enable resource-mgmt:default
$ svcs svc:/system/resource-mgmt:default
STATE          STIME    FMRI
online         Jan_12   svc:/system/resource-mgmt:default
```

3. **Use the `zonecfg` command.**

```
$ zonecfg -z global
```

4. **Set 5 shares for the global zone.**

```
zonecfg:global> set cpu-shares=5
```

5. **Exit `zonecfg`.**

```
zonecfg:global> exit
```

How to Modify Zone Privileges

Use this procedure to change privileges in a non-global zone. The default Oracle Solaris privileges and optional privileges you can specify for a zone are shown in "Privileges in a Non-Global Zone" on page 144.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Select the zone to be modified, `my-zone` in this procedure.**

   ```
   global$ zonecfg -z my-zone
   zonecfg:my-zone>
   ```

3. **Revise the set of privileges.**

   ```
   zonecfg:my-zone> set limitpriv="default,file_flag_set"
   ```

   This line adds the ability to set `immutable`, `nounlink` or `appendonly` file attributes on files from within the non-global zone.

4. **Commit the change.**

   ```
   zonecfg:my-zone> commit
   ```

5. **Exit the `zonecfg` command.**

   ```
   zonecfg:my-zone> exit
   ```

   Committed changes made through `zonecfg` take effect the next time the zone is booted.

Using the `zonecfg` Command to Revert or Remove a Zone Configuration

Use the `zonecfg` command described in `zonecfg`(1M) to revert a zone's configuration or to delete a zone configuration.

How to Revert a Zone Configuration

While configuring a zone with the `zonecfg` utility, use the `revert` subcommand to undo a resource setting made to the zone configuration.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **While configuring a zone called `tmp-zone`, type `info` to view your configuration:**

```
zonecfg:tmp-zone> info
```

   The `net` resource segment of the configuration displays as follows:

```
…
fs:
        dir: /tmp
        special: swap
        type: tmpfs
net:
        address: 192.0.2.0
        physical: eri0
device
        match: /dev/pts/*
…
```

3. **Remove the net address.**

```
zonecfg:tmp-zone> remove net address=192.0.2.0
```

4. **Verify that the `net` entry has been removed.**

```
zonecfg:tmp-zone> info

…
fs:
        dir: /tmp
        special: swap
        type: tmpfs
device
        match: /dev/pts/*
…
```

5. **Type `revert`.**

```
zonecfg:tmp-zone> revert
```

6. **Answer yes to the following question:**

```
Are you sure you want to revert (y/[n])? y
```

7. **Verify that the net address is once again present:**

```
zonecfg:tmp-zone> info

…
fs:
        dir: /tmp
        special: swap
        type: tmpfs
net:
        address: 192.0.2.0
        physical: eri0
device
        match: /dev/pts/*
…
```

How to Delete a Zone Configuration

Use `zonecfg` with the `delete` subcommand to delete a zone configuration from the system.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Delete the zone configuration for the zone `a-zone` by using one of the following two methods:**

   - Use the `-F` option to force the action:

     ```
     global$ zonecfg -z a-zone delete -F
     ```

   - Delete the zone interactively by answering yes to the system prompt:

     ```
     global$ zonecfg -z a-zone delete
     Are you sure you want to delete zone a-zone (y/[n])? y
     ```

♦ ♦ ♦ CHAPTER 2

About Installing, Shutting Down, Halting, Uninstalling, and Cloning Non-Global Zones

This chapter discusses zone installation on the Oracle Solaris operating system. It also describes the two processes that manage the virtual platform and the application environment, `zoneadmd` and `zsched`. Information about halting, rebooting, cloning, and uninstalling zones is also provided.

The following topics are addressed in this chapter:

To clone a non-global zone, install and boot a non-global zone, or to halt or uninstall a non-global zone, see Chapter 3, "Installing, Booting, Shutting Down, Halting, Uninstalling, and Cloning Non-Global Zones".

For information about `solaris10` branded zone installation, see Chapter 5, "Installing the solaris10 Branded Zone" in *Creating and Using Oracle Solaris 10 Zones*.

Zone Installation and Administration Concepts

The `zoneadm` command, described in the `zoneadm`(1M) man page is the primary tool used to install and administer non-global zones. Run operations using the `zoneadm` command from the

global zone. If zone administration is assigned to non-root users, subcommands that make a copy of another zone require the authorization `solaris.zone.clonefrom/`*source_zone*.

Perform the following tasks by using the `zoneadm` command:

- Verify a zone
- Install a zone
- Attach a zone
- Change the state of an installed zone to incomplete
- Boot a zone, which is similar to booting a regular Oracle Solaris system
- Display information about a running zone
- Shut down a zone
- Halt a zone
- Reboot a zone
- Uninstall a zone
- Relocate a zone from one point on a system to another point on the same system
- Provision a new zone based on the configuration of an existing zone on the same system
- Migrate a zone, used with the `zonecfg` command

For zone installation and verification procedures, see Chapter 3, "Installing, Booting, Shutting Down, Halting, Uninstalling, and Cloning Non-Global Zones" and the `zoneadm`(1M) man page. Also refer to the `zoneadm`(1M) man page for supported options to the `zoneadm list` command. For zone configuration procedures, see Chapter 1, "How to Plan and Configure Non-Global Zones" and the `zonecfg`(1M) man page. Zone states are described in "Non-Global Zone State Model" in *Introduction to Oracle Solaris Zones*.

To produce auditing records for zones, read "Using Oracle Solaris Auditing in Zones" on page 148 before you install non-global zones.

Zone Construction

This section applies to initial non-global zone construction, and not to the cloning of existing zones.

The zone is installed using the packages specified by the manifest passed to the `zoneadm install -m` command. If no manifest is provided, the default manifest uses `pkg:/group/system/solaris-small-server`. A new zone has the default `solaris` configuration and logs (SMF repository, `/etc`, `/var`), which are only modified by the profile(s) passed to `zoneadm install -s`, and the networking information specified in any `zonecfg add net` entries.

The system repository, the zone's configured publishers, and packages kept in sync with the global zone are discussed in Chapter 8, “About Automatic Installation and Packages on an Oracle Solaris System With Zones Installed”.

The files needed for the zone's root file system are installed by the system under the zone's root path.

A successfully installed zone is ready for booting and initial login.

Data from the following are not referenced or copied when a zone is installed:

- Non-installed packages
- Data on CDs and DVDs
- Network installation images

In addition, the following types of information that can be present in the global zone are not copied into a zone that is being installed:

- New or changed users in the `/etc/passwd` file
- New or changed groups in the `/etc/group` file
- Configurations for networking services such as DHCP address assignment
- Customizations for networking services such as `sendmail`
- Configurations for network services such as naming services
- New or changed `crontab`, printer, and mail files
- System log, message, and accounting files

If Oracle Solaris Auditing is used, modifications to files might be required. For more information, see “Using Oracle Solaris Auditing in Zones” on page 148.

The resources specified in the configuration file are added when the zone transitions from installed to ready. A unique zone ID is assigned by the system. File systems are mounted, network interfaces are set up, and devices are configured. Transitioning into the ready state prepares the virtual platform to begin running user processes.

In the ready state, the `zsched` and `zoneadmd` processes are started to manage the virtual platform.

- `zsched`, a system scheduling process similar to `sched`, is used to track kernel resources associated with the zone.
- `zoneadmd` is the zones administration daemon.

A zone in the ready state does not have any user processes executing in the zone. At least one process must be executing in a zone for the zone to be in the running state. See the `init`(1M) man page for more information.

How Zones Are Installed

The `solaris` brand installer supports the following zone installation methods:

- The publisher origin.

  To install a non-global zone, the repository that you set as the `solaris` publisher origin must contain at least the same software that is installed in the global zone. The repository can also contain older or newer software, but it must contain the same software that is installed in the global zone.
- A Unified Archive file image of an installed system running the Oracle Solaris release or a `solaris` non-global zone.
- A zone BE, using `zoneadm install -z` *zbe*. The system performs a package update if necessary.

The installer options are as follows. See "How to Install a Configured Zone" on page 51 for example command lines.

`-a` *archive*

The path to an archive used to install a non-global zone. Archives can be compressed using `gzip` or `bzip`. The `-d` and the `-a` options are incompatible.

When using the `-a` *archive* option, a package update is performed if necessary. The `zoneadm attach` subcommand can be used to reattach the zone to its original host if wanted.

`-c` *profile|dir*

Provides a profile or a directory of profiles to apply during configuration. The file argument must be specified with an absolute path. If a profile is applied, the configuration step occurs non-interactively.

If no profile is provided, the interactive system configuration tool is used for the configuration of the system. All profiles must have an `.xml` file extension. If you supply a directory option to `-c`, all profiles in that directory must be valid, correctly formed configuration profiles.

`-d` *path*

The path to the root directory of an installed system or a non-global zone. A package update is performed if necessary.

If *path* is a hyphen (`-`), the `zonepath` is assumed to be already be populated with the system image. The `-d` and the `-a` options are incompatible.

`-m` *manifest*

The AI manifest is an XML file that defines how to install a zone. The file argument must be specified with an absolute path.

`-p`

Preserve system identity after installing the zone. The `-p` and the `-u` options are incompatible.

`-s`

Install silently. The `-s` and the `-v` options are incompatible.

`-u`

Unconfigure the zone after installing it, and prompt for a new configuration on zone boot. The `-p` and the `-u` options are incompatible.

`-U`

Update all packages to the latest versions if necessary to be compatible with the packages installed in the global zone.

`-v`

Verbose output from the install process. The `-s` and the `-v` options are incompatible.

`-x`

Use `force-zpool-import` with the `-x` option to forcibly import any zpools that appear to be in use.

If a storage object contains any preexisting partitions, zpools, or UFS file systems, the `install` fails and an error message is displayed. The `-x` option to `zoneadm install` is used to continue the installation and overwrite any preexisting data. This option is similar to the `zpool create -f` command.

Use `force-zpool-create-all` with the `-x` option to forcibly create all zpool resources. Use `force-zpool-create=`*zpoolname* to limit the option to a specific zpool or set of zpools. See the `zoneadm`(1M) man page for usage.

The `zoneadmd` Daemon

The zones administration daemon, `zoneadmd`, is the primary process for managing the zone's virtual platform. The daemon is also responsible for managing zone booting and shutting down. There is one `zoneadmd` process running for each active (ready, running, or shutting down) zone on the system.

The `zoneadmd` daemon sets up the zone as specified in the zone configuration. This process includes the following actions:

- Allocating the zone ID and starting the `zsched` system process

- Setting zone-wide resource controls
- Preparing the zone's devices as specified in the zone configuration
- Setting up network interfaces
- Mounting loopback and conventional file systems
- Instantiating and initializing the zone console device

If the zoneadmd daemon is not already running, it is automatically started by zoneadm. Thus, if the daemon is not running for any reason, any invocation of zoneadm to administer the zone restarts zoneadmd.

The man page for the zoneadmd daemon is zoneadmd(1M).

The zsched Zone Scheduler

An active zone is a zone that is in the ready state, the running state, or the shutting down state. Every active zone has an associated kernel process, zsched. Kernel threads doing work on behalf of the zone are owned by zsched. The zsched process enables the zones subsystem to keep track of per-zone kernel threads.

Zone Application Environment

The zoneadm command is used to create the zone application environment.

The internal configuration of the zone is specified by using the sysconfig interface. The internal configuration specifies a naming service to use, the default locale and time zone, the zone's root password, and other aspects of the application environment. The sysconfig interface is described in “System Configuration Interactive Tool” on page 69 and the sysconfig(1M) man page. Note that the default locale and time zone for a zone can be configured independently of the global settings.

About Shutting Down, Halting, Rebooting, and Uninstalling Zones

This section provides an overview of the procedures for halting, rebooting, uninstalling, and cloning zones.

Shutting Down a Zone

The `zoneadm shutdown c` command is used to cleanly shut down a zone. The action is equivalent to running `/usr/sbin/init 0` in the zone. If the `-r` option is also specified, the zone is then rebooted. See "Zone Boot Arguments" on page 44 for supported boot options.

The `svc:/system/zones` service uses the `zoneadm shutdown` to cleanly shut down zones when the global zone shuts down.

The `shutdown` subcommand waits until the zone is successfully shut down. If the action doesn't complete within a reasonable amount of time, `zoneadm halt` can be used to forcibly halt the zone. See "How to Halt a Zone" on page 58 for more information.

Halting a Zone

The `zoneadm halt` command is used to terminate all processes running in a zone and remove the virtual platform. The zone is then brought back to the installed state. All processes are killed, devices are unconfigured, network interfaces are destroyed, file systems are unmounted, and the kernel data structures are destroyed.

The `zoneadm halt` command is used to terminate all processes running in a zone and remove the virtual platform. The zone is then brought back to the installed state. All processes are killed, devices are unconfigured, network interfaces are destroyed, file systems are unmounted, and the kernel data structures are destroyed.

The `halt` command does *not* run any shutdown scripts within the zone. To shut down a zone, see "Shutting Down a Zone" on page 43. Alternatively, you can log in to the zone and run shutdown. See "How to Use `zlogin` to Shut Down a Zone" on page 84.

If the halt operation fails, see "Zone Does Not Halt" on page 190.

Rebooting a Zone

The `zoneadm reboot` command is used to reboot a zone. The zone ID will change when the zone is rebooted.

Zone Boot Arguments

You can use the following boot arguments with the `zoneadm boot` and `reboot` subcommands:

- `-i` *altinit*
- `-m` *smf_options*
- `-s`

The following definitions apply:

`-i` *altinit*

Selects an alternative executable to be the first process. *altinit* must be a valid path to an executable. The default first process is described in `init`(1M).

`-m` *smf_options*

Controls the boot behavior of SMF. There are two categories of options, recovery options and messages options. Message options determine the type and number of messages that displays during boot. Service options determine the services that are used to boot the system.

Recovery options include the following:

debug

Prints standard per-service output and all `svc.startd` messages to log.

`milestone=`*milestone*

Boot to the subgraph defined by the given milestone. Legitimate milestones are `none`, `single-user`, `multi-user`, `multi-user-server`, and `all`.

Message options include the following:

`quiet`

Prints standard per-service output and error messages requiring administrative intervention

`verbose`

Prints standard per-service output and messages providing more information.

`-s`

Boots only to milestone `svc:/milestone/single-user:default`. This milestone is equivalent to `init` level `s`.

For usage examples, see “How to Boot a Zone” on page 56 and “How to Boot a Zone in Single-User Mode” on page 57.

For information on the Oracle Solaris service management facility (SMF) and `init`, see *Managing System Services in Oracle Solaris 11.3*, `svc.startd`(1M) and `init`(1M).

Zone `autoboot` Property Value

To automatically boot a zone when the global zone is booted, set the `autoboot` resource property in a zone's configuration to `true`. The default setting is `false`.

Note that for zones to automatically boot, the zones service `svc:/system/zones:default` must also be enabled. This service is enabled by default.

See “Zones Packaging Overview” on page 113 for information on the `autoboot` setting during `pkg update`.

Uninstalling a Zone

The `zoneadm uninstall` command is used to uninstall all of the files under the zone's root file system. Before proceeding, the command prompts you to confirm the action, unless the `-F` (force) option is also used. Use the `uninstall` command with caution, because the action is irreversible.

About Cloning Non-Global Zones

Cloning allows you to copy an existing configured and installed zone on your system to rapidly provision a new zone on the same system. Note that at a minimum, you must reset properties and resources for the components that cannot be identical for different zones. Thus, the `zonepath` must always be changed. In addition, for a shared-IP zone, the IP addresses in any net resources must be different. For an exclusive-IP zone, the physical property of any net resources must be different. Application-specific configurations generally must be re-configured in the clone. For example, if you have a database instance in a zone and you clone that zone, you might have to reconfigure the database instance in the clone so that it recognizes itself as a different instance.

- Cloning a zone is a faster way to install a zone.
- The new zone will include any changes that have been made to customize the source zone, such as added packages or file modifications.

You can clone a zone by using one of the following methods:

- Clone a zone using the `zoneadm clone` command. This method is recommended if you are cloning a small number of zones.

  When the source `zonepath` and the target `zonepath` both reside on ZFS and are in the same pool, the `zoneadm clone` command automatically uses ZFS to clone the zone. When using ZFS clone, the data is not actually copied until it is modified. Thus, the initial clone takes very little time. The `zoneadm` command takes a ZFS snapshot of the source `zonepath`, and sets up the target `zonepath`. The `zonepath` of the destination zone is used to name the ZFS clone.

  Note - You can specify that a ZFS `zonepath` be copied instead of ZFS cloned, even though the source could be cloned in this way.

  See "Cloning a Non-Global Zone on the Same System" on page 61 for more information.
- Cloning a zone using a Unified Archive file. This method is recommended when you must clone multiple zones for a large deployment. Cloning a zone by this method requires the following steps:

  1. Creating a Unified Archive file. A Unified Archive file can contain all zones or selected zones.
  2. Using the `zonecfg` and `zoneadm` commands to configure and install the new zone or zones. When you create a new zone using an archive as a reference source, the new zone will mimic the original system's configuration.

     See Chapter 1, "Oracle Solaris System Recovery and Cloning (Overview)" in *Using Unified Archives for System Recovery and Cloning in Oracle Solaris 11.3* for further information.

Creating a Golden Zone

```
# zonecfg -z z1-golden 'create -t z1; set zonepath=/zones/z1-golden'
# zoneadm -z z1 shutdown
# zoneadm -z z1-golden clone z1
# zoneadm -z z1 boot
```

To create a copy of your golden zone, *z1-golden*:

```
# zonecfg -z z2 'create -t z1-golden; set zonepath=/zones/z2'
# zoneadm -z z2 clone z1-golden
```

You do not have to shut down the source zone during cloning. Because you have never booted `z1-golden`, it has diverged very little from the snapshots used to create it. Thus, the cost of `z1-golden` is likely less than 1 megabyte of disk space. As new boot environments are created, `z1-golden` will get new boot environments at only the cost of the disk space used by the updated packages.

♦ ♦ ♦ CHAPTER 3

Installing, Booting, Shutting Down, Halting, Uninstalling, and Cloning Non-Global Zones

This chapter describes how to install and boot a non-global zone. A method for using cloning to install a zone on the same system is also provided. Other tasks associated with installation, such as halting, rebooting, and uninstalling zones, are addressed. Move an existing non-global zone to a new location on the same machine. Procedures to move an existing non-global zone to a new location on the same machine and to completely delete a zone from a system are also included.

For general information about zone installation and related operations, see Chapter 2, “About Installing, Shutting Down, Halting, Uninstalling, and Cloning Non-Global Zones”.

For information about `solaris10` branded zone installation and cloning, see Chapter 5, “Installing the solaris10 Branded Zone” in *Creating and Using Oracle Solaris 10 Zones*.

Zone Installation

Task	Description	For Instructions
(Optional) Verify a configured zone prior to installing the zone.	Ensure that a zone meets the requirements for installation. If you skip this procedure, the verification is performed automatically when you install the zone.	“How to Verify a Configured Zone Before It Is Installed” on page 50
Install a configured zone.	Install a zone that is in the configured state.	“How to Install a Configured Zone” on page 51
Obtain the universally unique identifier (UUID) for the zone.	This separate identifier, assigned when the zone is installed, is an alternate way to identify a zone.	“How to Obtain the UUID of an Installed Non-Global Zone” on page 53
(Optional) Transition an installed zone to the ready state.	You can skip this procedure if you want to boot the zone and use it immediately.	“How to Transition the Installed Zone to the Ready State” on page 55
Boot a zone.	Booting a zone places the zone in the running state. A zone can be booted from the ready state or from the installed state.	“How to Boot a Zone” on page 56

Task	Description	For Instructions
Boot a zone in single-user mode.	Boots only to milestone `svc:/milestone/single-user:default`. This milestone is equivalent to `init` level `s`. See the `init`(1M) and `svc.startd`(1M) man pages.	"How to Boot a Zone in Single-User Mode" on page 57

Installing and Booting Zones

Use the `zoneadm` command described in the `zoneadm`(1M) man page to perform installation tasks for a non-global zone. You must be the global administrator or a zone administrator with appropriate rights to perform the zone installation. The examples in this chapter use the zone name and zone path established in "Configuring, Verifying, and Committing a Zone" on page 22.

▼ How to Verify a Configured Zone Before It Is Installed

You can verify a zone prior to installing it. One of the checks performed is a check for sufficient disk size. If you skip this procedure, the verification is performed automatically when you install the zone.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Verify a configured zone named `my-zone` by using the `-z` option with the name of the zone and the `verify` subcommand.**

```
global$ zoneadm -z my-zone verify
```

   This message regarding verification of the zone path will be displayed:

```
WARNING: /zones/my-zone does not exist, so it could not be verified.
    When 'zoneadm install' is run, 'install' will try to create
    /zones/my-zone, and 'verify' will be tried again,
    but the 'verify' may fail if:
    the parent directory of /system/zones/my-zone is group- or other-writable
    or
    /system/zones/my-zone overlaps with any other installed zones
    or
    /system/zones/my-zone is not a mountpoint for a zfs file system.
```

However, if an error message is displayed and the zone fails to verify, make the corrections specified in the message and try the command again.

If no error messages are displayed, you can install the zone.

Verifying Zones on Shared Storage

For zones configured on shared storage, `zonecfg verify` verifies that none of the configured zpool resources are already online on the system, for a zone in the configured state.

For zones configured on shared storage, the `zoneadm verify` command confirms that all zpools configured as `zpool` and `rootzpool` resources are online on the system, for a zone in the installed state. If the resources are not available, the `verify` fails and information about the failed zpools is displayed.

▼ How to Install a Configured Zone

This procedure is used to install a configured non-global zone. For information on installation options, see "How Zones Are Installed" on page 40.

The zone must reside on its own ZFS dataset. Only ZFS is supported. The `zoneadm install` command automatically creates a ZFS file system (dataset) for the `zonepath` when the zone is installed. If a ZFS dataset cannot be created, the zone is not installed.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Install the configured zone `my-zone` by using the `zoneadm` command with the `install` subcommand, automatically creating a ZFS dataset for the `zonepath` ZFS.**

   Note - The parent directory of the zone path must also be a dataset or the file system creation will fail.

   Use one of the following installation methods:

   - **Install the zone.**

     ```
     global$ zoneadm -z my-zone install
     ```

   - **Install the zone from the repository.**

```
global$ zoneadm -z my-zone install -m manifest -c [ profile | dir ]
```

- **Install the zone from an image.**

  ```
  global$ zoneadm -z my-zone install -a archive -s -u
  ```

- **Install the zone from a directory.**

  ```
  global$ zoneadm -z my-zone install -d path -p -v
  ```

The system will display that a ZFS file system has been created for this zone.

You will see various messages as the files and directories needed for the zone's root file system are installed under the zone's root path.

3. **(Optional) If an error message is displayed and the zone fails to install, type the following to get the zone state:**

```
global$ zoneadm list -v
# zoneadm list -cvd
  ID NAME             STATUS      PATH                       BRAND    IP
   0 global           running     /                          solaris  shared
   - my-zone          configured  /zones/my-zone             solaris  excl
```

- If the state is listed as `configured`, make the corrections specified in the message and try the `zoneadm install` command again.
- If the state is listed as `incomplete`, first execute this command:

  ```
  global$ zoneadm -z my-zone uninstall
  ```

  Make the corrections specified in the message, and try the `zoneadm install` command again.

4. **(Optional) If a storage object contains any preexisting partitions, zpools, or UFS file systems, the `install` fails and an error message is displayed.**

The source zone must be in the `uninstalled` state before the `force` subcommand can be used:

```
global$ zoneadm -z my-zone uninstall
```

Then, continue the installation and overwrite any preexisting data by using one of the following `-x` options to `zoneadm install`:

- `-x force-zpool-import`
- `-x force-zpool-create=`*zpoolname*
- `-x force-zpool-create=`*zpoolname1*,*zpoolname2*,*zpoolname3*

- `-x force-zpool-create-all`

  This option is similar to the `zpool create -f` command.

  `-x force-zpool-create=`*zpoolname* can be used one or more times.

5. **When the installation completes, use the `list` subcommand with the `-i` and `-v` options to list the installed zones and verify the status.**

   ```
   global$ zoneadm list -iv
   ```

   You will see a display that is similar to the following:

   ```
   ID  NAME     STATUS       PATH                          BRAND      IP
    0  global   running      /                             solaris    shared
    -  my-zone  installed    /zones/my-zone                solaris    excl
   ```

Troubleshooting If a zone installation is interrupted or fails, the zone is left in the `incomplete` state. Use `uninstall -F` to reset the zone to the `configured` state.

Next Steps This zone was installed with the Oracle Solaris Service Management Facility (SMF) framework. You can enable or disable individual services as described in Chapter 3, "Administering Services" in *Managing System Services in Oracle Solaris 11.3*.

▼ How to Obtain the UUID of an Installed Non-Global Zone

A universally unique identifier (UUID) is assigned to a zone when it is installed. Obtain the UUID by using `zoneadm` with the `list` subcommand and the `-c -p` options. The UUID is the fifth field of the display.

1. **Become a zone administrator.**
   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **View the UUIDs for zones that have been installed.**

   ```
   global$ zoneadm list -cp
   ```

   You will see a display similar to the following:

   ```
   0:global:running:/::solaris:shared:-:none:
   3:test_zone:running:/system/volatile/zones/test_zone/zonepath:95180a6d-fab2-4363-ee33-
   81ba6e84a84f:solaris-kz:excl:R:solaris-kz:
   ```

```
-:zone123:installed:/system/zones/zone123:96972ce7-d41d-4fec-ff4b-8f14123e0974:solaris:
excl:-::
```

Example 1 How to Obtain the UUID for a Specific Zone

Use the following command to obtain the UUID for *test_zone*:

```
$ zoneadm list -cp | grep test_zone | cut -f 5 -d:
```

You will see a display similar to the following:

```
95180a6d-fab2-4363-ee33-81ba6e84a84f
```

Example 2 How to Use the *test_zone* UUID in a Command

```
global$ zoneadm -z test_zone -u 95180a6d-fab2-4363-ee33-81ba6e84a84f list -v
```

You will see a display similar to the following:

```
ID NAME             STATUS      PATH                          BRAND      IP
3 test_zone         running     -                             solaris-kz excl
```

If both -u *uuid-match* and -z *zonename* are present, the match is done based on the UUID first. If a zone with the specified UUID is found, that zone is used, and the -z parameter is ignored. If no zone with the specified UUID is found, then the system searches by the zone name.

About the Zone UUID

Zones can be uninstalled and reinstalled under the same name with different contents. Zones can also be renamed without the contents being changed. For these reasons, the UUID is more reliable than the zone name.

See Also For more information, see `zoneadm`(1M) and `libuuid`(3LIB).

▼ How to Mark an Installed Non-Global Zone Incomplete

If administrative changes on the system have rendered a zone unusable or inconsistent, it is possible to change the state of an installed zone to `incomplete`.

1. **Become a zone administrator.**

For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Mark the zone `testzone` incomplete.**

```
global# zoneadm -z testzone mark incomplete
```

3. **Use the `list` subcommand with the `-i` and `-v` options to verify the status.**

```
global$ zoneadm list -iv
```

You will see a display that is similar to the following:

```
ID  NAME     STATUS       PATH                           BRAND      IP
 0  global   running      /                              solaris    shared
 -  my-zone  installed    /zones/my-zone                 solaris    excl
 -  testzone incomplete   /zones/testzone                solaris    excl
```

Marking a Zone Incomplete

The `-R` *root* option can be used with the `mark` and `list` subcommands of `zoneadm` to specify an alternate boot environment. See `zoneadm`(1M) for more information.

Note - Marking a zone incomplete is irreversible. The only action that can be taken on a zone marked incomplete is to uninstall the zone and return it to the configured state. See “How to Uninstall a Zone” on page 60.

▼ How to Transition the Installed Zone to the Ready State

Transitioning into the `ready` state prepares the virtual platform to begin running user processes. Zones in the `ready` state do not have any user processes executing in them.

You can skip this procedure if you want to boot the zone and use it immediately. The transition through the `ready` state is performed automatically when you boot the zone.

1. **Become a zone administrator.**

For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Use the `zoneadm` command with the `-z` option, the name of the zone, which is `my-zone`, and the `ready` subcommand to transition the zone to the ready state.**

```
global$ zoneadm -z my-zone ready
```

3. **At the prompt, use the `zoneadm list` command with the `-v` option to verify the status.**

```
global$ zoneadm list -v
```

You will see a display that is similar to the following:

```
ID  NAME     STATUS       PATH                           BRAND      IP
 0  global   running      /                              solaris    shared
 1  my-zone  ready        /zones/my-zone                 solaris    excl
```

Note that the unique zone ID `1` has been assigned by the system.

How to Boot a Zone

Boot a zone to place the zone in the `running` state. You can boot from the `ready` state or from the `installed` state. A zone in the `installed` state that is booted transparently transitions through the `ready` state to the `running` state. Zone login is allowed for zones in the `running` state.

You must be the global administrator or a user with appropriate authorizations in the global zone to perform this procedure.

1. **Become a zone administrator.**

For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Use the `zoneadm` command with the `-z` option, the name of the zone, which is `my-zone`, and the `boot` subcommand to boot the zone.**

```
global$ zoneadm -z my-zone boot
```

3. **When the boot completes, use the `list` subcommand with the `-v` option to verify the status.**

```
global# zoneadm list -v
```

You will see a display that is similar to the following:

```
ID  NAME     STATUS       PATH                           BRAND      IP
 0  global   running      /                              solaris    shared
```

```
 1  my-zone  running     /zones/my-zone                solaris    excl
```

Example 3 Specifying Boot Arguments for Zones

Boot a zone using the `-m verbose` option:

```
global$ zoneadm -z my-zone boot -- -m verbose
```

Reboot a zone using the `-m verbose` boot option:

```
global$ zoneadm -z my-zone reboot -- -m verbose
```

Reboot a zone inside the zone using the `-m verbose` option:

```
my-zone# reboot -- -m verbose
```

▼ How to Boot a Zone in Single-User Mode

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Boot the zone in single-user mode.**

   ```
   global$ zoneadm -z my-zone boot -- -s
   ```

Where to Go From Here

To log in to the zone and perform the initial internal configuration, see Chapter 4, "About Non-Global Zone Login" and Chapter 5, "Logging In to Non-Global Zones".

Shutting Down, Halting, Rebooting, and Uninstalling Zones

▼ How to Shut Down a Zone

The shutdown procedure cleanly shuts down the zone.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **List the zones running on the system.**

   ```
   global$ zoneadm list -v
   ```

   You will see a display that is similar to the following:

   ```
   ID  NAME     STATUS       PATH                           BRAND      IP
    0  global   running      /                              solaris    shared
    1  my-zone  running      /zones/my-zone                 solaris    excl
   ```

3. **Use the `zoneadm` command with the `-z` option, the name of the zone, for example, `my-zone`, and the `shutdown` subcommand shut down the given zone.**

   ```
   global$ zoneadm -z my-zone shutdown
   ```

4. **Also specify the `-r` option to reboot the zone.**

   ```
   global$ zoneadm -z my-zone shutdown -r boot_options
   ```

   See Example 3, "Specifying Boot Arguments for Zones," on page 57.

5. **List the zones running on the system to confirm that the zone has been shut down.**

   ```
   global$ zoneadm list -v
   ```

How to Halt a Zone

The halt procedure is used to remove both the application environment and the virtual platform for a zone. To cleanly shut down a zone, see "How to Use `zlogin` to Shut Down a Zone" on page 84.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **List the zones running on the system.**

   ```
   global$ zoneadm list -v
   ```

   You will see a display that is similar to the following:

```
ID  NAME     STATUS       PATH                           BRAND      IP
 0  global   running      /                              solaris    shared
 1  my-zone  running      /zones/my-zone                 solaris    excl
```

3. **Use the `zoneadm` command with the `-z` option, the name of the zone, for example, `my-zone`, and the `halt` subcommand to halt the given zone.**

```
global$ zoneadm -z my-zone halt
```

4. **List the zones on the system again, to verify that `my-zone` has been halted.**

```
global$ zoneadm list -iv
```

You will see a display that is similar to the following:

```
ID  NAME     STATUS       PATH                           BRAND      IP
 0  global   running      /                              solaris    shared
 -  my-zone  installed    /zones/my-zone                 solaris    excl
```

5. **Boot the zone if you want to restart it.**

```
global$ zoneadm -z my-zone boot
```

Troubleshooting If the zone does not halt properly, see "Zone Does Not Halt" on page 190 for troubleshooting tips.

▼ How to Reboot a Zone

1. **Become a zone administrator.**

For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **List the zones running on the system.**

```
global$ zoneadm list -v
```

You will see a display that is similar to the following:

```
ID  NAME     STATUS       PATH                           BRAND      IP
 0  global   running      /                              solaris    shared
 1  my-zone  running      /zones/my-zone                 solaris    excl
```

3. **Use the `zoneadm` command with the `-z reboot` option to reboot the zone `my-zone`.**

```
global$ zoneadm -z my-zone reboot
```

4. **List the zones on the system again to verify that `my-zone` has been rebooted.**

```
global# zoneadm list -v
```

You will see a display that is similar to the following:

```
ID  NAME     STATUS       PATH                           BRAND      IP
 0  global   running      /                              solaris    shared
 2  my-zone  running      /zones/my-zone                 solaris    excl
```

Tip - Note that the zone ID for `my-zone` has changed. The zone ID generally changes after a reboot.

How to Use the `zoneadm` Command to Rename a Zone

Use the `zoneadm` command with the `rename` subcommand to rename a zone.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Change the name of the zone.**

```
global$ zoneadm -z old_name rename new_name
```

How to Uninstall a Zone

Caution - Use this procedure with caution. The action of removing all of the files in the zone's root file system is irreversible.

The zone cannot be in the `running` state. The `uninstall` operation is invalid for running zones.

You must be the global administrator or a user with appropriate authorizations in the global zone to perform this procedure.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **List the zones on the system.**

```
global$ zoneadm list -v
```

You will see a display that is similar to the following:

```
ID  NAME     STATUS       PATH                           BRAND      IP
 0  global   running      /                              solaris    shared
 -  my-zone  installed    /zones/my-zone                 solaris    excl
```

3. **Use the `zoneadm` command with the `-z uninstall` option to remove the zone `my-zone`.**

You can also use the `-F` option to force the action. If this option is not specified, the system will prompt for confirmation.

```
global$ zoneadm -z my-zone uninstall -F
```

Note that when you uninstall a zone that has its own ZFS file system for the `zonepath`, the ZFS file system is destroyed.

4. **List the zones on the system again, to verify that `my-zone` is no longer listed.**

```
global$ zoneadm list -iv
```

You will see a display that is similar to the following:

```
ID  NAME     STATUS       PATH                           BRAND      IP
 0  global   running      /                              solaris    shared
```

Troubleshooting If a zone uninstall is interrupted, the zone is left in the `incomplete` state. Use the `zoneadm uninstall` command to reset the zone to the `configured` state.

If the `zonepath` is not removed, this could be an indication that this zone is installed in another boot environment. The `zonepath` and various datasets that exist within the `zonepath` dataset are not removed while a boot environment exists that has an installed zone with a given `zonepath`. See `beadm`(1M) for more information about boot environments.

Use the `uninstall` command with caution because the action is irreversible.

Cloning a Non-Global Zone on the Same System

Use cloning to provision a new zone on a system by copying the data from a source `zonepath` to a target `zonepath`.

When the source `zonepath` and the target `zonepath` both reside on ZFS and are in the same pool, the `zoneadm clone` command automatically uses ZFS to clone the zone. However, you can specify that the ZFS `zonepath` be copied and not ZFS cloned.

How to Clone a Zone

You must configure the new zone before you can install it. The parameter passed to the `zoneadm create` subcommand is the name of the zone to clone. This source zone must be halted.

1. **Become a zone administrator.**
   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Halt the source zone to be cloned, which is `my-zone` in this procedure.**

   ```
   global$ zoneadm -z my-zone halt
   ```

3. **Start configuring the new zone by exporting the configuration of the source zone `my-zone` to a file, for example, `master`.**

   ```
   global$ zonecfg -z my-zone export -f /zones/master
   ```

   Note - You can also create the new zone configuration using the procedure "How to Configure the Zone" on page 22 instead of modifying an existing configuration. If you use this method, skip ahead to Step 6 after you create the zone.

4. **Edit the file `master`.**
   Set different properties and resources for the components that cannot be identical for different zones. For example, you must set a new `zonepath`.

   - For a shared-IP zone, the IP addresses in any net resources must be changed.
   - For an exclusive-IP zone, the physical property of any net resource must be changed.

5. **Create the new zone, `zone1`, by using the commands in the file *master*.**

   ```
   global# zonecfg -z zone1 -f /zones/master
   ```

6. **Install the new zone, `zone1`, by cloning `my-zone`.**

   ```
   global# zoneadm -z zone1 clone my-zone
   ```

   The system displays:

   ```
   Cloning zonepath /zones/my-zone...
   ```

7. **(Optional) If a storage object contains any preexisting partitions, zpools, or UFS file systems, the `clone` fails, and an error message is displayed use the appropriate `-x` option to `zoneadm clone`.**

   This continues the operation and overwrites any preexisting data. This option is similar to the `zpool create -f` command.

   Note - The source zone must be halted and uninstalled before the `-x force` subcommand can be used.

   - `-x force-zpool-import`
   - `-x force-zpool-create=`*zpoolname*
   - `-x force-zpool-create=`*zpoolname1*`,`*zpoolname2*`,`*zpoolname3*
   - `-x force-zpool-create-all`

   The `-x force-zpool-create=`*zpoolname* option can be used multiple times.

8. **List the zones on the system.**

```
ID  NAME     STATUS       PATH                           BRAND      IP
 0  global   running      /                              solaris    shared
 -  my-zone  installed    /zones/my-zone                 solaris    excl
 -  zone1    installed    /zones/zone1                   solaris    excl
```

Example 4 Applying a System Configuration Profile to a Cloned Zone

Provide an absolute path to the configuration file:

```
# zoneadm -z zone1 clone -c /path/config.xml my-zone
```

Note that you must provide an absolute path to the configuration file.

Moving a Non-Global Zone

This procedure is used to move the zone to a new location on the same system by changing the `zonepath`. The zone must be halted. The normal `zonepath` criteria described in "Resource Types and Properties" in *Oracle Solaris Zones Configuration Resources* apply.

This information also applies to moving `solaris10` branded zones. For information on `solaris10` branded zones, see *Creating and Using Oracle Solaris 10 Zones*.

Note - You cannot move a zone in one of the following conditions:

- The zone is present in other BEs. You can either delete those BEs first, or create a new zone at the new path by cloning the zone.
- The zone is on shared storage with a `rootzpool` resource to a different location on the system. A rename of the `zonepath` is supported.

▼ How to Move a Zone That Is Not on Shared Storage

1. **Become a zone administrator.**
   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Halt the zone to be moved, `db-zone` in this procedure.**

   ```
   global$ zoneadm -z db-zone halt
   ```

3. **Use the `zoneadm` command with the `move` subcommand to move the zone to a new `zonepath`, `/zones/db-zone`.**

   ```
   global$ zoneadm -z db-zone move /zones/db-zone
   ```

4. **Verify the path.**

   ```
   ID  NAME     STATUS       PATH                           BRAND      IP
    0  global   running      /                              solaris    shared
    -  my-zone  installed    /zones/my-zone                 solaris    excl
    -  db-zone  installed    /zones/db-zone                 solaris    excl
   ```

Deleting a Non-Global Zone From the System

The procedure described in this section completely deletes a zone from a system.

▼ How to Remove a Non-Global Zone

1. **Become a zone administrator.**

For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Shut down the zone `my-zone` using one of the following methods.**

   The `zoneadm shutdown` method is preferred.

   - **Using `zoneadm`:**

     ```
     global$ zoneadm -z my-zone shutdown
     my-zone
     ```

   - **Using `zlogin`:**

     ```
     global$ zlogin my-zone shutdown
     my-zone
     ```

3. **Remove the root file system for `my-zone`.**

   ```
   global$ zoneadm -z my-zone uninstall -F
   ```

   The `-F` option to force the action generally isn't required.

4. **Delete the configuration for `my-zone`.**

   ```
   global$ zonecfg -z my-zone delete -F
   ```

   The `-F` option to force the action generally isn't required.

5. **List the zones on the system, to verify that `my-zone` is no longer listed.**

   ```
   global$ zoneadm list -iv
   ```

   You will see a display that is similar to the following:

   ```
   ID  NAME     STATUS       PATH                           BRAND      IP
    0  global   running      /                              solaris    shared
   ```

♦ ♦ ♦ CHAPTER 4

About Non-Global Zone Login

This chapter discusses logging in to zones from the global zone. The following topics are covered in this chapter:

For procedures and usage information, see Chapter 5, "Logging In to Non-Global Zones". For the full list of available options, see the `zlogin`(1) man page.

`zlogin` Command

If zone administration is assigned to non-root users, the authorization `solaris.zone.manage/`*zonename* is required. A specific *zonename* suffix, preceded by the slash character (`/`), is optional. When omitted, the authorization matches any zone.

Unless the `-C` option is used to connect to the zone console, logging in to a zone using `zlogin` starts a new task. A task cannot span two zones.

The `zlogin` command is used to log in from the global zone to any zone that is in the running state or the ready state.

Note - Only the `zlogin` command with the `-C` option can be used to log in to a zone that is not in the running state.

As described in "How to Use Non-Interactive Mode to Access a Zone" on page 82, you can use the `zlogin` command in non-interactive mode by supplying a command to run inside a

zone. However, the command or any files the command acts upon cannot reside on NFS. The command will fail if any of its open files or any portion of its address space resides on NFS. The address space includes the command executable itself and the command's linked libraries.

The `zlogin` command can only be used by the global administrator or a zone administrator with the appropriate rights, operating in the global zone. See the `zlogin(1)` man page for more information.

Internal Zone Configuration

The system configuration data can exist as either a single profile, `sc_profile.xml` or as a directory, `profiles`, of SMF profiles. The single file or directory both describe the zones system configuration data that will be passed to the automated installer during zone installation. If no `sc_profile.xml` file or `profiles` directory is given during zone installation, the `sysconfig` interactive tool will query the administrator for the data the first time the console `zlogin` command is used.

To provide a profile or a directory of profiles to apply after installation from the repository, use `-z` *ZBE* with the `zoneadm attach` command. The specified existing zone boot environment is attached. If the specified zone boot environment is associated with a different global zone, the specified *ZBE* is cloned and a clone of the *ZBE* is attached. All profiles must have an `.xml` extension.

Oracle Solaris uses SMF to centralize the configuration information.

An Oracle Solaris instance is created and configured during installation. An Oracle Solaris instance is defined as a boot environment in either a global or a non-global zone. You can use the `sysconfig` utility to perform configuration tasks on an Oracle Solaris instance, or to unconfigure an Oracle Solaris instance and reconfigure the instance. The `sysconfig` command can be used to create an SMF profile.

After the installation or creation of an Oracle Solaris instance in a global or non-global zone, where system configuration is needed, system configuration will happen automatically. System configuration is not needed in the case of a `zoneadm clone` operation in which the `-p` option to preserve system identity is specified, or in the case of an `attach` operation in which the `-c` *profile.xml* `sysconfig` file option is not specified.

You can do the following:

- Use the `sysconfig configure` command to configure that Oracle Solaris instance and cause the SCI tool to start on the console.

  ```
  # sysconfig configure
  ```

- Use the `sysconfig configure` command to configure an Oracle Solaris instance in the global or a non-global zone.

  ```
  # sysconfig configure -c sc_profile.xml
  ```

  If you specify an existing configuration profile with the command, a non-interactive configuration is performed. If you do not specify an existing configuration profile with the command, the System Configuration Interactive (SCI) Tool runs. The SCI Tool enables you to provide specific configuration information for that Oracle Solaris instance.
- You can use the `sysconfig create-profile` command to create a new system configuration profile.

The `sysconfig` interface is described in "System Configuration Interactive Tool" on page 69 and in the `sysconfig`(1M) man page.

System Configuration Interactive Tool

The System Configuration Interactive (SCI) Tool enables you to specify configuration parameters for a newly installed Oracle Solaris instance.

The `sysconfig create-profile` command queries the administrator and creates an SMF profile file in `/system/volatile/scit_profile.xml`. Parameters include system hostname, time zone, user and root accounts, name services.

To navigate in the tool:

- Use the function keys listed at the bottom of each screen to move through the screens and to perform other operations. If your keyboard does not have function keys, or if the keys do not respond, press the Esc key. The legend at the bottom of the screen changes to show the Esc keys for navigation and other functions.
- Use the up and down arrow keys to change the selection or to move between input fields.

For more information, see the `sysconfig`(1M) man page.

Example Zone Configuration Profiles

Exclusive-IP zone with automatic configuration

```
<!DOCTYPE service_bundle SYSTEM "/usr/share/lib/xml/dtd/service_bundle.dtd.1">
<service_bundle type="profile" name="sysconfig">
  <service version="1" type="service" name="system/config-user">
    <instance enabled="true" name="default">
      <property_group type="application" name="root_account">
```

```
        <propval type="astring" name="login" value="root"/>
        <propval type="astring" name="password" value="$5$KeNRy1zU
$lqzy9rIsNloUhfvJFIWmVewE75aB5/EBA77kY7EP6F0"/>
        <propval type="astring" name="type" value="role"/>
      </property_group>
      <property_group type="application" name="user_account">
        <propval type="astring" name="login" value="admin1"/>
        <propval type="astring" name="password" value="$5$/g353K5q$V8Koe/XuAeR/
zpBvpLsgVIqPrvc.9z0hYFYoyoBkE37"/>
        <propval type="astring" name="type" value="normal"/>
        <propval type="astring" name="description" value="admin1"/>
        <propval type="count" name="gid" value="10"/>
        <propval type="astring" name="shell" value="/usr/bin/bash"/>
        <propval type="astring" name="roles" value="root"/>
        <propval type="astring" name="profiles" value="System Administrator"/>
        <propval type="astring" name="sudoers" value="ALL=(ALL) ALL"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/timezone">
    <instance enabled="true" name="default">
      <property_group type="application" name="timezone">
        <propval type="astring" name="localtime" value="UTC"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/environment">
    <instance enabled="true" name="init">
      <property_group type="application" name="environment">
        <propval type="astring" name="LC_ALL" value="C"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/identity">
    <instance enabled="true" name="node">
      <property_group type="application" name="config">
        <propval type="astring" name="nodename" value="my-zone"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/keymap">
    <instance enabled="true" name="default">
      <property_group type="system" name="keymap">
        <propval type="astring" name="layout" value="US-English"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/console-login">
```

```
    <instance enabled="true" name="default">
      <property_group type="application" name="ttymon">
        <propval type="astring" name="terminal_type" value="vt100"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="network/physical">
    <instance enabled="true" name="default">
      <property_group type="application" name="netcfg">
        <propval type="astring" name="active_ncp" value="Automatic"/>
      </property_group>
    </instance>
  </service>
</service_bundle>
```

Exclusive-IP zone with static configuration using NIS without DNS

```
<!DOCTYPE service_bundle SYSTEM "/usr/share/lib/xml/dtd/service_bundle.dtd.1">
<service_bundle type="profile" name="sysconfig">
  <service version="1" type="service" name="system/config-user">
    <instance enabled="true" name="default">
      <property_group type="application" name="root_account">
        <propval type="astring" name="login" value="root"/>
        <propval type="astring" name="password" value="$5$m8OR3zqK
$Ox5XGubRJdi4zj0JzNSmVJ3Ni4opDOGpxi2nK/GGzmC"/>
        <propval type="astring" name="type" value="normal"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/timezone">
    <instance enabled="true" name="default">
      <property_group type="application" name="timezone">
        <propval type="astring" name="localtime" value="UTC"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/environment">
    <instance enabled="true" name="init">
      <property_group type="application" name="environment">
        <propval type="astring" name="LC_ALL" value="C"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/identity">
    <instance enabled="true" name="node">
      <property_group type="application" name="config">
        <propval type="astring" name="nodename" value="my-zone"/>
      </property_group>
    </instance>
```

```
  </service>
  <service version="1" type="service" name="system/keymap">
    <instance enabled="true" name="default">
      <property_group type="system" name="keymap">
        <propval type="astring" name="layout" value="US-English"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/console-login">
    <instance enabled="true" name="default">
      <property_group type="application" name="ttymon">
        <propval type="astring" name="terminal_type" value="vt100"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="network/physical">
    <instance enabled="true" name="default">
      <property_group type="application" name="netcfg">
        <propval type="astring" name="active_ncp" value="DefaultFixed"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="network/install">
    <instance enabled="true" name="default">
      <property_group type="application" name="install_ipv4_interface">
        <propval type="astring" name="address_type" value="static"/>
        <propval type="net_address_v4" name="static_address" value="10.10.10.13/24"/
>
        <propval type="astring" name="name" value="net0/v4"/>
        <propval type="net_address_v4" name="default_route" value="10.10.10.1"/>
      </property_group>
      <property_group type="application" name="install_ipv6_interface">
        <propval type="astring" name="stateful" value="yes"/>
        <propval type="astring" name="stateless" value="yes"/>
        <propval type="astring" name="address_type" value="addrconf"/>
        <propval type="astring" name="name" value="net0/v6"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/name-service/switch">
    <property_group type="application" name="config">
      <propval type="astring" name="default" value="files nis"/>
      <propval type="astring" name="printer" value="user files nis"/>
      <propval type="astring" name="netgroup" value="nis"/>
    </property_group>
    <instance enabled="true" name="default"/>
  </service>
  <service version="1" type="service" name="system/name-service/cache">
```

```
    <instance enabled="true" name="default"/>
  </service>
  <service version="1" type="service" name="network/dns/client">
    <instance enabled="false" name="default"/>
  </service>
  <service version="1" type="service" name="network/nis/domain">
    <property_group type="application" name="config">
      <propval type="hostname" name="domainname" value="example.net"/>
      <property type="host" name="ypservers">
        <host_list>
          <value_node value="192.0.2.0"/>
        </host_list>
      </property>
    </property_group>
    <instance enabled="true" name="default"/>
  </service>
  <service version="1" type="service" name="network/nis/client">
    <instance enabled="true" name="default"/>
  </service>
</service_bundle>
```

Exclusive-IP zone with dynamic configuration with NIS

```
<!DOCTYPE service_bundle SYSTEM "/usr/share/lib/xml/dtd/service_bundle.dtd.1">
<service_bundle type="profile" name="sysconfig">
 <service version="1" type="service" name="system/config-user">
   <instance enabled="true" name="default">
     <property_group type="application" name="root_account">
       <propval type="astring" name="login" value="root"/>
       <propval type="astring" name="password" value="$5$Iq/.A.
K9$RQyt6RqsAY8TgnuxL9i0/84QwgIQ/nqcK8QsTQdvMy/"/>
       <propval type="astring" name="type" value="normal"/>
     </property_group>
   </instance>
 </service>
 <service version="1" type="service" name="system/timezone">
   <instance enabled="true" name="default">
     <property_group type="application" name="timezone">
       <propval type="astring" name="localtime" value="UTC"/>
     </property_group>
   </instance>
 </service>
 <service version="1" type="service" name="system/environment">
   <instance enabled="true" name="init">
     <property_group type="application" name="environment">
       <propval type="astring" name="LC_ALL" value="C"/>
     </property_group>
   </instance>
 </service>
```

```
<service version="1" type="service" name="system/identity">
  <instance enabled="true" name="node">
    <property_group type="application" name="config">
      <propval type="astring" name="nodename" value="my-zone"/>
    </property_group>
  </instance>
</service>
<service version="1" type="service" name="system/keymap">
  <instance enabled="true" name="default">
    <property_group type="system" name="keymap">
      <propval type="astring" name="layout" value="US-English"/>
    </property_group>
  </instance>
</service>
<service version="1" type="service" name="system/console-login">
  <instance enabled="true" name="default">
    <property_group type="application" name="ttymon">
      <propval type="astring" name="terminal_type" value="sun-color"/>
    </property_group>
  </instance>
</service>
<service version="1" type="service" name="system/name-service/switch">
  <property_group type="application" name="config">
    <propval type="astring" name="default" value="files nis"/>
    <propval type="astring" name="printer" value="user files nis"/>
    <propval type="astring" name="netgroup" value="nis"/>
  </property_group>
  <instance enabled="true" name="default"/>
</service>
<service version="1" type="service" name="system/name-service/cache">
  <instance enabled="true" name="default"/>
</service>
<service version="1" type="service" name="network/dns/client">
  <instance enabled="false" name="default"/>
</service>
<service version="1" type="service" name="network/nis/domain">
  <property_group type="application" name="config">
    <propval type="hostname" name="domainname" value="special.example.com"/>
    <property type="host" name="ypservers">
      <host_list>
        <value_node value="192.0.2.0"/>
      </host_list>
    </property>
  </property_group>
  <instance enabled="true" name="default"/>
</service>
<service version="1" type="service" name="network/nis/client">
  <instance enabled="true" name="default"/>
```

```
 </service>
</service_bundle>
```

Non-Global Zone Login Methods

This section describes the methods you can use to log in to a zone.

Zone Console Login

Each zone maintains a virtual console, `/dev/console`. Performing actions on the console is referred to as console mode. Console login to a zone is available when the zone is in the installed state. The zone console is closely analogous to a serial console on a system. Connections to the console persist across zone reboots. To understand how console mode differs from a login session such as `telnet`, see "Remote Login" on page 76.

The zone console is accessed by using the `zlogin` command with the `-C` option and the *zonename*. The zone does not have to be in the running state.

The `-d` option can also be used. The option specifies that if the zone halts, the zone disconnects from the console. This option can only be specified with the `-C` option.

Processes inside the zone can open and write messages to the console. If the `zlogin -C` process exits, another process can then access the console.

If role-based access control (RBAC) is in use, access to the zone console requires the authorization `solaris.zone.manage/`*zonename*. A specific *zonename* suffix, preceded by the slash character (`/`), is optional. When omitted, the authorization matches any zone.

To bring up the System Configuration Interactive (SCI) Tool upon boot, type the following:

```
root@test2:~# sysconfig configure -s
```

User Login Methods

To log in to the zone with a user name, use the `zlogin` command with the `-l` option, the user name, and the *zonename*. For example, the administrator of the global zone can log in as a normal user in the non-global zone by specifying the `-l` option to `zlogin`:

```
global$ zlogin -l user zonename
```

To log in as user `root`, use the `zlogin` command without options.

Failsafe Mode

If a login problem occurs and you cannot use the `zlogin` command or the `zlogin` command with the `-C` option to access the zone, an alternative is provided. You can enter the zone by using the `zlogin` command with the `-S` (safe) option. Only use this mode to recover a damaged zone when other forms of login are not succeeding. In this minimal environment, it might be possible to diagnose why the zone login is failing.

Remote Login

In Oracle Solaris, the secure by default (SBD) feature is implemented automatically at installation. With this feature, Secure Shell is the only enabled remote login to an Oracle Solaris system. Use `ssh` to enter a non-global zone. Other remote login services such as `rlogin` or `telnet` are insecure and can expose your network to unauthorized access. For more information, see the `ssh(1)` man page.

Interactive and Non-Interactive Modes

Two other methods for accessing the zone and for executing commands inside the zone are also provided by the `zlogin` command. These methods are interactive mode and non-interactive mode.

Interactive Mode

In interactive mode, a new pseudo-terminal is allocated for use inside the zone. Unlike console mode, in which exclusive access to the console device is granted, an arbitrary number of `zlogin` sessions can be open at any time in interactive mode. Interactive mode is activated when you do not include a command to be issued. Programs that require a terminal device, such as an editor, operate correctly in this mode.

If zone administration is assigned to non-root users and you are a zone administrator logging in interactively, you must be assigned the authorization `solaris.zone.login/`*zonename* for the zone. Password authentication takes place in the zone.

Non-Interactive Mode

Non-interactive mode is used to run shell scripts which administer the zone. Non-interactive mode does not allocate a new pseudo-terminal. Non-interactive mode is enabled when you supply a command to be run inside the zone.

For non-interactive logins, or to bypass password authentication, the authorization `solaris.zone.manage/`*zonename* is required.

♦ ♦ ♦ CHAPTER 5

Logging In to Non-Global Zones

This chapter provides procedures for completing the configuration of an installed zone, logging into a zone from the global zone, and shutting down a zone. This chapter also shows how to use the `zonename` command to print the name of the current zone.

For an introduction to the zone login process, see Chapter 4, “About Non-Global Zone Login”.

Initial Zone Boot and Zone Login Procedures

Task	Description	For Instructions
Perform the internal configuration or unconfigure a zone.	System configuration can occur either interactively by using a text user interface, or non-interactively by using a profile.	See “System Configuration Interactive Tool” on page 69 and the `sysconfig`(1M) man page.
Log in to the zone.	You can log into a zone through the console, by using interactive mode to allocate a pseudo-terminal, or by supplying a command to be run in the zone. Supplying a command to be run does not allocate a pseudo-terminal. You can also log in by using failsafe mode when a connection to the zone is denied.	“Logging In to a Zone” on page 80
Exit a non-global zone.	Disconnect from a non-global zone.	“How to Exit a Non-Global Zone” on page 83
Shut down a zone.	Shut down a zone by using the `shutdown` utility or a script.	“How to Use `zlogin` to Shut Down a Zone” on page 84
Print the zone name.	Print the zone name of the current zone.	“Displaying the Name of the Current Zone” on page 85

Logging In to a Zone

Use the `zlogin` command to log in from the global zone to any zone that is running or in the ready state. See the `zlogin`(1) man page for more information.

You can log in to a zone in various ways, as described in the following procedures. You can also log in remotely, as described in "Remote Login" on page 76.

▼ How to Create a Configuration Profile

Caution - You must supply all required data. If you provide a profile with missing data, then the zone is configured with missing data. This configuration might prevent the user from logging in or getting the network running.

1. **Assume the `root` role.**

   For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Create the profile using the `sysconfig` tool.**

   - **For an exclusive-IP zone**

     ```
     # sysconfig create-profile -o /path/sysconf.xml
     ```

   - **For a shared-IP zone:**

     ```
     # sysconfig create-profile -o /path/sysconf.xml -g
      location,identity,naming_services,users
     ```

3. **Use the created profile during zone install, clone, or attach operations.**

   ```
   # zoneadm -z my-zone install -c /path/sysconf.xml
   ```

   If the configuration file is used, the system will *not* start the System Configuration Interactive (SCI) Tool on the console at initial `zlogin`. The file argument must be specified with an absolute path.

▼ How to Log In to the Zone Console to Perform the Internal Zone Configuration

If a `config.xml` file was passed to the `zoneadm clone`, `attach`, or `install` commands, this configuration file is used to configure the system. If no `config.xml` file was provided during the `clone`, `attach`, or `install` operation, then the first boot of the zone will start the SCI Tool on the console.

To avoid missing the initial prompt for configuration information, it is recommended that two terminal windows be used, so that `zlogin` is running before the zone is booted in a second session.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Use the `zlogin` command with the `-C` option and the name of the zone, for example, `my-zone`.**

   ```
   global$ zlogin -C my-zone
   ```

3. **From another terminal window, boot the zone.**

   ```
   global$ zoneadm -z my-zone boot
   ```

   You will see a display similar to the following in the `zlogin` terminal window:

   ```
   [NOTICE: Zone booting up]
   ```

4. **Respond to the series of questions about configuration parameters for your newly installed zone.**

   Parameters include system host name, time zone, user and root accounts, and name services. By default, the SCI Tool produces an SMF profile file in `/system/volatile/scit_profile.xml`.

Troubleshooting If the initial SCI screen doesn't appear, you can type **Ctrl L** to refresh the SCI screen.

▼ How to Log In to the Zone Console

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Use the `zlogin` command with the `-C` option, the `-d` option and the name of the zone, for example, `my-zone`.**

   ```
   global$ zlogin -C -d my-zone
   ```

   Using the `zlogin` command with the `-C` option starts the SCI Tool if the configuration has not been performed.

3. **When the zone console displays, log in as `root`, press Return, and type the root password when prompted.**

   ```
   my-zone console login: root
   Password:
   ```

How to Use Interactive Mode to Access a Zone

In interactive mode, a new pseudo-terminal is allocated for use inside the zone.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **From the global zone, log in to the zone, for example, `my-zone`.**

   ```
   global$ zlogin my-zone
   ```

   Information similar to the following will display:

   ```
   [Connected to zone 'my-zone' pts/2]
   Last login: Wed Jul  3 16:25:00 on console
   ```

3. **Type `exit` to close the connection.**

   You will see a message similar to the following:

   ```
   [Connection to zone 'my-zone' pts/2 closed]
   ```

How to Use Non-Interactive Mode to Access a Zone

Non-interactive mode is enabled when the user supplies a command to be run inside the zone. Non-interactive mode does not allocate a new pseudo-terminal.

Note - The command or any files that the command acts upon cannot reside on NFS.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **From the global zone, log in to the `my-zone` zone and supply a command name.**

   The command `zonename` is used here.

   ```
   global$ zlogin my-zone zonename
   ```

   You will see the following output:

   ```
   my-zone
   ```

How to Exit a Non-Global Zone

- **To disconnect from a non-global zone, use one of the following methods.**

  - **To exit the zone non-virtual console:**

    ```
    zonename$ exit
    ```

  - **To disconnect from a zone virtual console, use the tilde (~) character and a period:**

    ```
    zonename$ exit
    ```

    Your screen will look similar to this:

    ```
    [Connection to zone 'my-zone' pts/6 closed]
    ```

  - **To exit from a Secure Shell session:**

    ```
    zonename$ ~~.
    ```

Note - The default escape sequence for `ssh` is also ~, which causes the `ssh` session to exit.

See Also For more information about `zlogin` command options, see the `zlogin(1)` man page.

▼ How to Use Failsafe Mode to Enter a Zone

When a connection to the zone is denied, the `zlogin` command can be used with the `-S` option to enter a minimal environment in the zone.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **From the global zone, use the `zlogin` command with the `-S` option to access the zone, for example, `my-zone`.**

   ```
   global$ zlogin -S my-zone
   ```

▼ How to Use `zlogin` to Shut Down a Zone

Note - Running `init 0` in the global zone to cleanly shut down a Oracle Solaris system also runs `init 0` in each of the non-global zones on the system. Note that `init 0` does not warn local and remote users to log off before the system is taken down.

Use this procedure to cleanly shut down a zone. To halt a zone without running shutdown scripts, see "How to Halt a Zone" on page 58.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Log in to the zone to be shut down, for example, `my-zone`, and specify `shutdown` as the name of the utility and `init 0` as the state.**

   ```
   global$ zlogin my-zone shutdown -i 0
   ```

   Your site might have its own shutdown script, tailored for your specific environment.

Enabling a Service

You can enable or disable individual services in the zone.

Displaying the Name of the Current Zone

The `zonename` command described in the `zonename(1)` man page displays the name of the current zone. The following example shows the output when `zonename` is used in the global zone.

```
$ zonename
global
```

♦ ♦ ♦ CHAPTER 6

Live Zone Reconfiguration

Live Zone Reconfiguration reconfigures or reports on the live configuration of non-global zones while the zones are running.

About Live Zone Reconfiguration

Use Live Zone Reconfiguration to do the following tasks:

- Report on and inspect the current live zone configuration
- Make changes to the live zone configuration
- Apply changes to the live zone configuration

Changes can be made either on a temporary or on a persistent basis. The standard tools `zonecfg` and `zoneadm` are used to administer Live Zone Reconfiguration. Temporary changes are active until the next reboot. You do not need to reboot for changes to be applied to the persistent configuration.

There is no downtime in service availability within the zone when the following configuration changes are made:

- Changing resource controls
- Changing network configuration
- Changing the CPU resource pool
- Adding or removing file systems
- Adding or removing virtual and physical devices

About Making Temporary Changes

Parameters changed in the live mode take effect immediately after they are committed. Temporary changes made through the `zonecfg` command are valid until the next zone reboot. The live mode is available for a running zone only.

About Making Changes to the Configuration

Use the `zonecfg` command edit modes to make configuration changes to the zone. You can change either the persistent stored configuration or the running live configuration. The `zonecfg` command supports the following the edit modes for use with Live Reconfiguration:

Default mode

Create, modify, and list the persistent zone configuration stored on the stable storage. Parameters changed in the default edit mode do not affect a running zone at the time the changes are made. The default edit mode is the primary way to maintain the zone configuration. This mode is backward compatible. The zone must be reconfigured by one of the following methods for the changes to take effect:

- `zoneadm apply`
- `zoneadm reboot`

Live mode

Retrieve, inspect and edit the running live configuration. The live mode is available for a running zone only. Parameters changed in the live mode take effect immediately after they are committed. Applied changes remain active until the next zone reboot. The live mode is enabled by the `-r` option, which is used to retrieve and to edit the live configuration instead of the persistent configuration.

```
# zonecfg -z zonename -r
```

About Applying Changes to the Configuration

You can use the `zoneadm apply` command to apply changes made by you to the live or the persistent zone configuration. You do not have to reboot for the changes to be made permanent. You can perform a trial run by using the `-n` before the changes are made permanent. You can use the following options to the `apply` subcommand:

`-n`

Trial run using the “no execute” option, `-n`. This trial run, referred to as dry run mode, uses the real reconfiguration but no changes are applied to the running zone. Use the dry run mode to review actions that would be performed by the real reconfiguration.

`-q`

Quiet mode. This mode suppresses all system messages and returns a status code only.

Live Zone Reconfiguration Examples

Use the examples in this section to perform common zones reconfiguration tasks.

▼ How to Inspect the Live Configuration of the Running Zone

All brands can inspect the configuration.

1. **Become a zone administrator.**

   For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Display information about the live zone configuration of the zone** *my-zone***.**

   ```
   $ zonecfg -z my-zone -r info
   ```

3. **(Optional) Export the live configuration.**

   ```
   $ zonecfg -z my-zone -r export -f exported.cfg
   ```

   All brands can export the configuration.

▼ How to View a Possible Configuration by Using a Dry Run

A dry run is also known as a trial run.

1. **Become a zone administrator.**

For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Configure changes using the `zonecfg` command.**

```
$ zonecfg -z my-zone -r
```

3. **Use the `commit` subcommand with the `-n` option to view the actions that would be performed by the actual reconfiguration.**

```
zonecfg:my-zone> commit -n
```

How to Make Persistent Configuration Changes and Apply the Changes

1. **Become a zone administrator.**

   For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Make changes to a zone named *zone1*, and apply the changes.**

```
$ zonecfg -z zone1 "set cpu-shares=4;clear pool;add anet;…"
$ zoneadm -z zone1 apply
           zone1: Checking: set property cpu-shares=4
           zone1: Checking: clear property pool
           zone1: Checking: add anet linkname=myanet0
           zone1: Applying changes
```

How to Temporarily Add an `anet` Resource to the Running Zone

1. **Become a zone administrator.**

   For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Make the addition and apply the changes.**

```
$ zonecfg -z zone1 -r "add anet;set linkname=anet1;set lower-link=net1;end;commit"
           zone1: Checking: add anet linkname=anet1
           zone1: Applying changes
```

   The `commit` subcommand in the example is not required. The `zonecfg` command commits the changes when the command exits.

How to Make Temporary Changes to the Running Zone

1. **Become a zone administrator.**

   For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Make a configuration change, such as adding a disk.**

```
$ zonecfg -z zone1 -r 'add device; set match=/dev/rdsk/c1t3d0*;end; \
        add device; set match=/dev/dsk/c1t3d0*; end;'
   zone1: Checking: Adding device match=/dev/rdsk/c1t3d0*
   zone1: Checking: Adding device match=/dev/dsk/c1t3d0*
   zone1: Applying the changes
```

3. **When you no longer need the disk you added, return the zone to the persistent configuration.**

```
$ zoneadm -z zone1 apply
   zone 'zonename': Checking: Removing device match=/dev/rdsk/c1t3d0*
   zone 'zonename': Checking: Removing device match=/dev/dsk/c1t3d0*
   zone 'zonename': Applying changes
```

How to Recover From a Failure While Committing Temporary Changes

The configuration of a running zone can change externally while the live configuration is being edited. When this conflict occurs, the `commit` subcommand returns an error. You can reload the configuration to show the updated version, and make the edits.

1. **Become a zone administrator.**

   For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Perform the following steps to recover from a failure by using `reload`.**

```
$ zonecfg -z zone1 -r
zonecfg:zone1> add anet;set linkname=anet1;set lower-link=net1;end
zonecfg:zone1> commit
zone1: error: Failed to commit. The live configuration of the zone
   'zone1' changed externally.
zonecfg:zone1> reload
zonecfg:zone1> add anet;set linkname=anet1;set lower-link=net1;end
```

```
zonecfg:zone1> commit
zone1: Checking: add anet linkname=anet1
zone1: Applying changes
```

♦ ♦ ♦ CHAPTER 7

Migrating and Converting Oracle Solaris Zones

This chapter describes how to migrate an existing `solaris` branded zone and its data from a source Oracle Solaris 11 system onto a target Oracle Solaris 11 system.

This chapter also describes zone conversions. Zone conversion is used in the following processes:

- Transforming global zones into non-global zones, also known as physical to virtual (P2V) conversion
- Transforming non-global zones into global zones

Migration can also involve data transfer using archives. These archives can be Oracle Solaris Unified Archives or legacy archive types such as `tar`, `cpio`, and `pax`. For Oracle Solaris 11, zone conversions using archives can be part of a zone migration strategy.

Zone migration is used for server consolidation, workload rebalancing, and disaster recovery.

About Migrating `solaris` Branded Zones

A *zone migration* is the process of transferring an existing zone configuration and data to another host system. A source zone is shut down, moved, and rebooted on a target host. You can migrate both global and non-global zones. See "About Zone Migration" in *Introduction to Oracle Solaris Zones* for more information on zone migration and zone migration types.

Note - `solaris10`, `solaris`, and `solaris-kz` zones all support migration. For migration requirements, see the documentation about the brand of zone being migrated.

For Oracle Solaris 11, migrations use zones on shared storage (ZOSS) to transfer data. Migrations using ZOSS take approximately the same time as a zone reboot. For additional information on ZOSS, see Chapter 13, "Getting Started With Oracle Solaris Zones on Shared Storage".

If ZOSS is not available, archives created by zone conversions can be used as part of a migration data transfer strategy. See “About Zone Conversions and Converting Oracle Solaris Zones” on page 94.

About Zone Conversions and Converting Oracle Solaris Zones

Zone conversion is the process of creating an archive of an Oracle Solaris global zone or non-global zone and deploying it as an Oracle Solaris global zone or non-global zone. The source zone type (global or non-global) may be the same as or different than the deployed zone type (global or non-global).

You can convert a zone as follows:

- A global zone can be transformed into a non-global zone. This process is also referred to as a *physical to virtual* or *P2V* conversion.
- A non-global zone can be transformed into a non-global zone. This process is sometimes referred to as a *virtual to virtual* or *V2V* conversion.
- A non-global zone may be transformed into a global zone. This process is sometimes referred to as a *virtual to physical conversion*.

On an Oracle Solaris 10 system, you can use archives to transform Oracle Solaris 10 `native` zones into `solaris10` branded zones on an upgraded host.

Note - Oracle Solaris supports multiple types of virtualization. For example, both Logical Domains and Zones are types of virtualization. Conversion of a Logical Domain to a Zone is a type of V2V conversion. From the view of zone conversion it is a global to non-global conversion, or a P2V conversion. The model of zone conversion on Oracle Solaris 11 releases is discussed in terms of global versus non-global instead of physical versus virtual.

The following archive types are supported by zone conversions:

- Unified Archives. See “Using Oracle Solaris Unified Archives for Zone Conversions” on page 95
- Legacy Archives. See “Using Legacy Archives for Zone Conversions” on page 95

Unified Archives are recommended for zone conversions for host systems running Oracle Solaris 11.2. For Oracle Solaris 11 releases prior to Oracle Solaris 11.2, use legacy archives.

Note - Virtual to physical conversions are only supported when using Unified Archives. See *Using Unified Archives for System Recovery and Cloning in Oracle Solaris 11.3* for more information on Unified Archives.

Prior to converting a global zone to a non-global zone, you can verify that the software running in the global zone is compatible with non-global zones. The `zonep2vchk`(1M) tool evaluates a global zone's configuration before conversion to a non-global zone.

Using Oracle Solaris Unified Archives for Zone Conversions

You can use the `zonecfg` and `zoneadm` commands to configure and to install new zones directly from a Unified Archive file. Unified Archive files contain both zone configuration and zone data. On the destination system, a zone can be configured and installed from the archive.

For this release, see *Using Unified Archives for System Recovery and Cloning in Oracle Solaris 11.3* for a full description of Unified Archives, including usage for system and zone cloning and recovery.

Using Legacy Archives for Zone Conversions

Legacy archives include archive files created with `zfs send`, `cpio`, or `pax`.

Legacy archives for zone conversions should only be considered:

- If the source system is running an Oracle Solaris release earlier than Oracle Solaris 11.2.
- If zones on shared storage (ZOSS) is not available.

See the `zfs`(1M), `cpio`(1), or `pax`(1) man pages for additional information.

Using the `zonep2vchk` Tool to Prepare for Global to Non-Global Conversions

The `zonep2vchk`(1M) tool evaluates a global zone's configuration before conversion to a non-global zone. The primary documentation for the tool is the `zonep2vchk`(1M) man page.

About the `zonep2vchk` Tool

The `zonep2vchk` evaluates a global zone, or physical instance, when it is converted to a non-global zone, or virtual instance. The physical to virtual (P2V) conversion process consists of archiving a global zone (source), and then installing a non-global zone (target) using that archive. The `zonep2vchk` utility must be run with an effective user `id` of `0`.

The utility does the following:

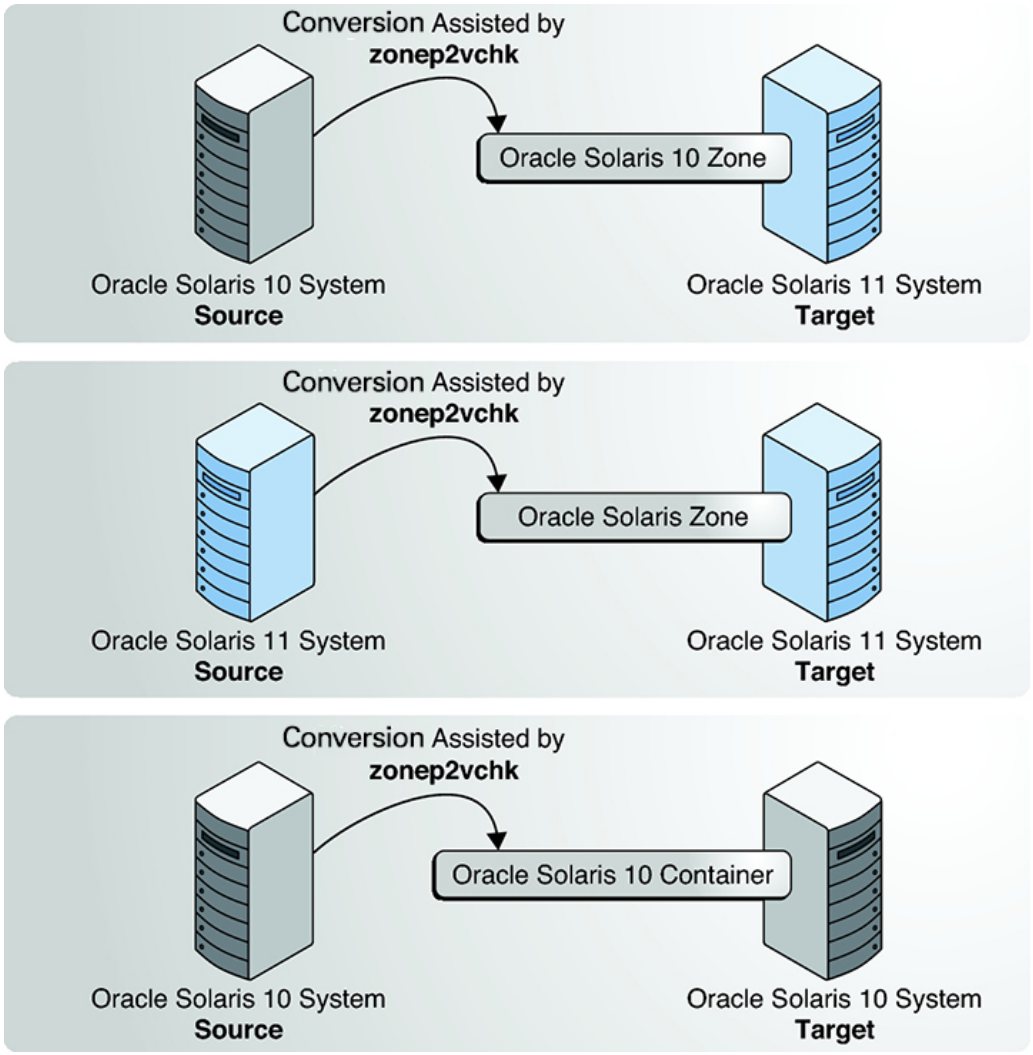
- Identifies problem areas in the source system's configuration
- Minimizes the manual reconfiguration effort required
- Supports conversion of both Oracle Solaris 10 and Oracle Solaris 11 system images into zones on Oracle Solaris 11 releases
- Supports complex network configurations in the original system image, including multiple IP interfaces, IP multipathing, and VLANs

This tool can be used to assist in the conversion of an Oracle Solaris 11 or Oracle Solaris 10 global zone to a non-global zone on this release. This tool results in the following conversion types:

- Converts an Oracle Solaris 11 system into a `solaris` brand zone
- Converts an Oracle Solaris 10 system into a `solaris10` brand zone

For Oracle Solaris 11 target systems, an `anet` resource (VNIC) is included in the `zonecfg` output for each network resource on the source system. By default, exclusive-IP is the network type when migrating either an Oracle Solaris 11 system or an Oracle Solaris 10 system into a non-global zone on an Oracle Solaris 11 system.

FIGURE 2 `zonep2vchk` Utility



Types of Zone Analyses

Basic analysis, the -b option, checks for Oracle Solaris features in use that might be impacted by global to non-global conversion.

Static analysis, the -s option, inspects binaries for system and library calls that might not function in a zone.

Runtime analysis, the -r option, inspects the currently executing applications for operations that might not function in a zone.

Information Produced From Zone Analyses

Two main categories of information are presented by the analysis:

- Issues that can be addressed with a specific zone configuration or with configuration changes in the global zone
- Identification of functions that cannot work inside a zone

For example, if an application sets the system clock, that can be enabled by adding the appropriate privilege to a zone, but if an application accesses kernel memory, that is never allowed inside a zone. The output distinguishes between these two classes of issues.

By default, the utility prints messages in human readable form. To print messages in machine parsable form, the -P option is used. For complete information on available options as well as command invocation and output, see the zonep2vchk(1M) man page.

Migration and Zone Conversion Requirements For solaris Branded Zones

Both zone migration and zone conversion on solaris branded zones have the following requirements:

- The global zone on the target system must be running an Oracle Solaris 11 release that is equal to or higher than that on the original source host.
- To ensure that the zone will run properly, the target system must have the same or later versions of the required operating system packages as those installed on the original source host.

  Other packages, such as those for third-party products, can be different.

Note - solaris10 branded zones and solaris-kz branded zones have additional brand-specific requirements for zone conversion and zone migration. For solaris10 branded zones, see Chapter 2, "Assessing an Oracle Solaris 10 System and Creating an Archive" in *Creating and Using Oracle Solaris 10 Zones*. For solaris-kz branded zones, see *Creating and Using Oracle Solaris Kernel Zones*.

Tasks to Migrate solaris Zones

This section describes migration administration and related tasks for solaris branded zones. The following topics are discussed:

About Orphaned Zone Boot Environments

When migrated to different hosts, native non-global zones can accumulate zone boot environments that are not associated with an existing global zone. A non-global zone boot environment is known as an *orphaned boot environment*.

If a zone boot environment that is not associated with any global zone is selected, it is cloned. This cloned boot environment will be updated and re-parented to the active global boot environment. The orphaned boot environment will continue to exist.

This section provides the following information to manage zone boot environments during zoneadm attach operations of a zone:

Note - The force-zbe-clone and the deny-zbe-clone are mutually exclusive options.

The options attach-last-booted-zbe and destroy-orphan-zbes can be used with the force-zbe-clone or deny-zbe-clone options.

destroy-orphan-zbes Zone Migration Option

This option destroys all zone boot environments that are not associated with any global zone.

To destroy all orphan zone boot environments during attach, use:

```
-x destroy-orphan-zbes
```

force-zbe-clone Zone Migration Option

If the selected zone boot environment is associated with the currently active global zone boot environment, the selected zone boot environment is attached. This behavior can be changed by using -x force-zbe-clone.

To select a different global zone boot environment, use:

```
-x force-zbe-clone
```

deny-zbe-clone Zone Migration Option

By default, if either of the following are true, the selected zone boot environment is cloned and the clone is attached.

- The selected zone boot environment is associated with another global zone boot environment.
- The selected zone boot environment is an orphaned boot environment that is not associated with any global zone boot environment.

  An attach operation that uses deny-zbe-clone fails when you try to attach an orphaned zone boot environment that is marked active on reboot. The zone boot environment must be marked inactive to use this option.

The original zone boot environment continues to exist.

To deny clone and attach, use:

```
-x deny-zbe-clone
```

beadm list Changes

Orphaned boot environments are indicated by beadm using the O flag in the active column. In the following output, solaris-0, solaris-1, and solaris-2 are orphaned.

```
root@t1: # beadm list
BE        Active Mountpoint Space   Policy Created
--        ------ ---------- -----   ------ -------
solaris-0 !RO    -          3.40M   static 2014-01-17 15:04
solaris-1 !RO    -          3.85M   static 2014-02-09 18:11
solaris-2 !RO    -          39.0K   static 2014-02-10 04:54
solaris-3 !R     -          2.19G   static 2014-02-12 16:56
solaris-5 NR     /          2.43G   static 2014-03-20 04:38
```

Boot environments that are shown as orphaned can be destroyed with `beadm destroy`. To destroy all orphaned boot environments, use the `-O` with `beadm destroy`.

For more information on boot environments, see *Creating and Administering Oracle Solaris 11.3 Boot Environments* and the `beadm`(1M) man page.

Migrating a Non-Global Zone to a Different System

This section provides the following procedures:

- "How to Migrate a Non-Global Zone by Using Shared Storage" on page 101
- "How to Convert a Non-Global Zone by Using Unified Archives" on page 102
- "How to Use ZFS Archives to Convert and to Deploy a Non-Global Zone on a Different Host" on page 103
- "How to Migrate Zones from One ZFS Pool to Another ZFS Pool in a Different System" on page 106

▼ How to Migrate a Non-Global Zone by Using Shared Storage

If a zone's storage is configured by using a `rootzpool` resource, has no dataset resources, and optionally contains one or more `rpool` resources, migration is quick and simple. For this procedure, both the source system and target system (in the examples shown in this procedure, `host1` and `host2`) must have access to the storage referenced in the `rootzpool` and `zpool` resources.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Configure the zone on the destination host.**

   source`$` **`zonecfg -z`** *zonename* **`export -f /net/`***hostname*`/`*zonename*`.cfg` *target*

```
$ zonecfg -z zonename -f /net/hostname/zonename.cfg
```

For example:

```
host1$ zonecfg -z my-zone export -f /net/my-host/my-zone.cfg
host2$ zonecfg -z my-zone -f /net/my-host/my-zone.cfg
```

3. **Shut down the zone.**

```
source$ zoneadm -z zonename shutdown
```

For example:

```
host1$ zoneadm -z my-zone shutdown
```

4. **Detach the zone from the source host.**

```
source$ zoneadm -z zonename detach
```

For example:

```
host1$ zoneadm -z my-zone detach
```

5. **Attach the zone to the destination host.**

The `-u` and `-U`options might be needed.

```
target$ zoneadm -z zonename attach
```

For example:

```
host2$ zoneadm -z my-zone attach
```

6. **Boot the zone.**

```
target$ zoneadm -z zonename boot
```

For example:

```
host2$ zoneadm -z my-zone boot
```

▼ How to Convert a Non-Global Zone by Using Unified Archives

1. **Become a zone administrator.**

To create an archive, you must be assigned the Unified Archive Administration rights profile. The `root` role has all of these rights.

For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Create a recovery archive.**
   Use this procedure on the source system to create a recovery archive of the zone to be migrated.

   source`$` **`archiveadm create -r -z`** *zonename archive-name*

   For example:

   `host1$` **`archiveadm create -r -z my-zone /net/server/my-zone-archive.uar`**

3. **Uninstall the zone on the source global zone, or set `autoboot` on the zone to false.**

   source`$` **`zonecfg -z`** *zonename* **`set autoboot=false`**

4. **Configure the target destination global zone from the archive.**

   target`$` **`zonecfg -z`** *zonename* **`create -a /net/server/`***zonename*`.uar`

   For example:

   `host2$` **`zonecfg -z my-zone create -a /net/server/my-zone-archive.uar`**

5. **Install the zone using the archive.**

   target`$` **`zoneadm -z`** *zonename* **`install -a`** *archive-name*

6. **Boot the migrated zone.**

   target`$` **`zoneadm -z`** *zonename* **`boot`**

See Also For additional information about creating and deploying Unified Archives, refer to Chapter 2, "Working With Unified Archives" in *Using Unified Archives for System Recovery and Cloning in Oracle Solaris 11.3*.

▼ How to Use ZFS Archives to Convert and to Deploy a Non-Global Zone on a Different Host

Note - Unified Archives are recommended as an archive zone conversion strategy for systems running Oracle Solaris 11.2 or higher.

This example describes how to create an archive of a non-global zone using the `zfs` command. The archive is then attached and deployed to another system.

This example assumes that the administrator on the source and target systems are able to access a shared NFS server for temporary file storage. In the event that shared temporary space is not available, other means, such as the `scp` remote secure copy program, can be used to

copy the files between the source and target systems. The `scp` program requests passwords or passphrases if they are needed for authentication.

1. **Assume the `root` role.**

   For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Shut down the zone to be migrated, `my-zone` in this procedure.**

   ```
   source# zoneadm -z my-zone shutdown
   ```

3. **(Optional) Detach the zone.**

   ```
   source# zoneadm -z my-zone detach
   ```

   The detached zone is now in the configured state. The zone will not automatically boot when the global zone next boots.

4. **Export the zone configuration.**

   ```
   source# mkdir /net/server/zonearchives/my-zone
   source# zonecfg -z my-zone export> /net/server/zonearchives/my-zone/my-zone.zonecfg
   ```

5. **Create a `gzip` ZFS archive.**

   ```
   source# zfs list -H -o name /zones/my-zone
   rpool/zones/my-zone
   source# zfs snapshot -r rpool/zones/my-zone@v2v
   source# zfs send -rc rpool/zones/my-zone@v2v | gzip > /net/server/zonearchives/my-zone/
   my-zone.zfs.gz
   ```

   Use of compression is optional, but it is generally faster because less I/O is performed while writing and subsequently reading the archive. For more information, see *Managing ZFS File Systems in Oracle Solaris 11.3*.

6. **On the target system, configure the zone.**

   ```
   target# zonecfg -z my-zone -f /net/server/zonearchives/my-zone/my-zone.zonecfg
   ```

   You will see the following system message:

   ```
   my-zone: No such zone configured
   Use 'create' to begin configuring a new zone.
   ```

7. **(Optional) View the configuration.**

   ```
   target# zonecfg:my-zone> info
   ```

```
zonename: my-zone
zonepath: /zones/my-zone
autoboot: false
pool:
net:
         address: 192.0.2.0
         physical: net0
```

8. **Make any required adjustments to the configuration.**

   For example, the network physical device is different on the target system, or devices that are part of the configuration might have different names on the target system.

```
target# zonecfg -z my-zone
zonecfg:my-zone> select net physical=net0
zonecfg:my-zone:net> set physical=net100
zonecfg:my-zone:net> end
```

9. **Commit the configuration and exit.**

```
zonecfg:my-zone> commit
zonecfg:my-zone> exit
```

10. **Install the zone on the target destination by using one of the following methods.**

    Use of the `install` subcommand is recommended.

    - **Install the zone, performing the minimum updates required to allow the `install` to succeed:**

```
target# zoneadm -z my-zone install -p -a /net/server/zonearchives/my-zone/my-zone.zfs.gz
```

      In this release, you can also `attach` the zone, performing the minimum updates required to allow the attach to succeed. If updates are allowed, catalogs from publishers are refreshed during a `zoneadm attach`.

```
target# zoneadm -z my-zone attach -u -a /net/server/zonearchives/my-zone/my-zone.zfs.gz
```

    - **Install the zone, updating all packages in the zone to the latest version that is compatible with the global zone.**

```
target# zoneadm -z my-zone install -U -p -a /net/server/zonearchives/my-zone/my-zone.zfs.gz
```

      In this release, you can also `attach` the zone, updating all in the zone to the latest version that is compatible with the global zone.

```
target# zoneadm -z my-zone install -U -a /net/server/zonearchives/my-zone/my-zone.zfs.gz
```

- **Attach the zone to the new host without updating any software.**

```
target# zoneadm -z my-zone attach -a /net/server/zonearchives/my-zone/my-zone.zfs.gz
```

Note - Use of the `install` subcommand is recommended. In this release, you can also attach the zone, updating all packages in the zone to the latest version that is compatible with the global zone.

Troubleshooting If a storage object contains any preexisting partitions, zpools, or UFS file systems, the `install` fails and an error message is displayed. To continue the installation and overwrite any preexisting data, use the `-x force-zpool-create` option to `zoneadm install`.

▼ How to Migrate Zones from One ZFS Pool to Another ZFS Pool in a Different System

The following procedure shows how to migrate all the zones from one ZFS pool to another ZFS pool in a different system.

In this procedure, the zones in source `SysA` will be moved to a pool in target `SysB`. The ZFS pool in `SysA` is `rpool/zones` and mounted at `/zones`. In `SysB`, the target pool is `newpool/zones`. The new zone path for the migrated zone is `/newpool/zones`.

The procedure also assumes that the target pool already exists.

For more information about creating ZFS pools, see Chapter 3, "Creating and Destroying Oracle Solaris ZFS Storage Pools" in *Managing ZFS File Systems in Oracle Solaris 11.3*.

Before You Begin Before you begin, you must ensure the following:

- All the compatibility requirements for performing a zones migration are met.
- Secure Shell has been configured to enable key-based authentication in both systems and root login. For more information about Secure Shell authentication, see "How to Generate a Public/Private Key Pair for Use With Secure Shell" in *Managing Secure Shell Access in Oracle Solaris 11.3*

1. **Become a zone administrator.**

   You must also be assigned the ZFS File System Management and ZFS Storage Management rights profiles. The `root` role has all of these rights.

For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Shut down and detach all zones that use `rpool/zones`.**
   Perform this step for all the zones that are using `rpool/zones`.

```
sysA$ zonename=zone1
sysA$ zoneadm -z $zonename shutdown
sysA$ zoneadm -z $zonename detach
sysA$ zonecfg -z $zonename export -f /zones/$zonename.cfg
```

3. **Create a snapshot of the source pool.**

```
sysA$ zfs snapshot -r rpool/zones@send-to-sysB
```

4. **Replicate the storage for the zones to the target pool in the system.**

```
sysA$ zfs send -R rpool/zones@send-to-sysB | ssh sysB zfs receive -d newpool
```

5. **On the target system, attach and boot the zones. Perform this step on all the zones that have been migrated.**

```
sysB$ zonename=zone1
sysB$ recvmountpoint=/newpool/zones
sysB$ zonecfg -z $zonename -f /newpool/zones/$zonename.cfg
sysB$ zonecfg -z $zonename 'set zonepath=/newpool/zones/%{zonename}'
sysB$ zoneadm -z $zonename attach -u
sysB$ zoneadm -z $zonename boot
```

where `%{zonename}` is a token as described in the Tokens section of the `zonecfg(1M)` man page.

Migrating a Zone From a Machine That Is Not Usable

A machine that hosts a non-global zone can become unusable. However, if the storage that the zone lives on, such as a SAN, is still usable, you might be able to migrate the zone to a new system. You can move the `zonepath` for the zone to the new system. In some cases, such as a SAN, the `zonepath` data might not actually move. The SAN might simply be reconfigured so the `zonepath` is visible on the new system. Because the zone was not properly detached, you must first create the zone on the new system by using the `zonecfg` command. After you create the zone, attach it on the new system.

The procedure for this task is described in “How to Use ZFS Archives to Convert and to Deploy a Non-Global Zone on a Different Host” on page 103.

Converting a Global Zone Into a Non-Global Zone

Because zones do not nest, the P2V process makes any existing zones inside the converted system image unusable in the destination zone. Unified archives bypass this limitation by using the `archiveadm create -r -z` command. For more information, see "Recovery Archives" in *Using Unified Archives for System Recovery and Cloning in Oracle Solaris 11.3*.

This section provides the following information:

About Converting an Oracle Solaris System (Global Zone) Into a `solaris` Brand Non-Global Zone

An existing Oracle Solaris 11 system can be directly migrated into a `solaris` brand zone on an Oracle Solaris 11 system. Use the Unified Archive `archiveadm` command on systems running Oracle Solaris 11.2 or higher. Use legacy archives on systems running the Oracle Solaris 11 or Oracle Solaris 11.1 releases.

For all systems, use the `zonep2vchk` command on the source system to prepare for converting and archiving the system image. Use the `zonecfg` and `zoneadm` commands to configure and to install the archive in the destination zone on the target system.

The following restrictions apply to migrating a global zone to a non-global zone:

- The global zone on the target system must be running an Oracle Solaris 11 release that is equal to or higher than the source system.
- To ensure that the zone runs properly, the target system must have the same or a later version of required operating system packages. Other packages, such as packages for third-party products, can be different.

For more information, see the `zonep2vchk`(1M), `archiveadm`(1M), `zfs`(1M), `zonecfg`(1M), and `zoneadm`(1M), and `solaris`(5) man pages.

▼ Scanning the Source System With `zonep2vchk`

1. **Assume the `root` role.**

For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Run the `zonep2vchk` tool with the `-b` option to perform a basic analysis that checks for Oracle Solaris features in use that might be impacted by a P2V migration.**

```
source# zonep2vchk -b
```

3. **Run the `zonep2vchk` tool with the `-s` option to perform a static analysis of application files.**

   This inspects ELF binaries for system and library calls that might affect operation inside a zone.

```
source# zonep2vchk -s /opt/myapp/bin,/opt/myapp/lib
```

4. **Run the `zonep2vchk` tool with the `-r` option to perform runtime checks that look for processes that could not be executed successfully inside a zone.**

```
source# zonep2vchk -r 2h
```

5. **Run the `zonep2vchk` tool with the `-c` option on the source system to generate a template `zonecfg` script, named `s11-zone.config` in this procedure.**

```
source# zonep2vchk -c > /net/somehost/p2v/s11-zone.config
```

   This configuration will contain resource limits and network configuration based on the physical resources and networking configuration of the source host.

▼ How to Create an Archive of the System Image on a Network Device

Archive the file systems in the global zone. Verify that no non-global zones are installed on the source system. The examples in this section use the `archiveadm` command for creating archives.

1. **Become an administrator who is assigned the Unified Archive Administration rights profile.**

   For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Create a recovery archive of the global zone on network storage.**

   In this procedure, the recovery archive of the global zone is named `/net/somehost/p2v/s11-zone-config.uar`.

```
source$ archiveadm create -r -z global /net/somehost/p2v/s11-zone-config.uar
```

See Also The Unified Archive generated with `archiveadm` contains a zone configuration that serves as a starting point for the converted zone. Additional configuration may be needed to configure the zone in a way that is optimal for the target system. For more information, see "How to Create a Recovery Archive" in *Using Unified Archives for System Recovery and Cloning in Oracle Solaris 11.3* and the `archiveadm`(1M) man page.

▼ How to Configure and Install a Migrated Zone on the Target System

The template `zonecfg` script generated by the `zonep2vchk` tool defines aspects of the source system's configuration that must be supported by the destination zone configuration. Additional target system dependent information must be manually provided to fully define the zone.

In this procedure, the zone is named `s11-zone` and the configuration file is named `/net/somehost/p2v/s11-zone-config.uar`.

1. **Become a zone administrator.**

   You must also be assigned the Network Management rights profile. The `root` role has all of these rights.

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Configure the zone from the archive.**

```
target$ zonecfg -z s11-zone
Use 'create' to begin configuring a new zone.
zonecfg:s11-zone> create -a /net/somehost/p2v/s11-zone-config.uar
zonecfg:s11-zone> info
```

3. **Add the zone's storage resources.**

```
zonecfg:s11-zone> set zonepath=/system/zones/%{zonename}
zonecfg:s11-zone> add rootzpool
zonecfg:s11-zone:rootzpool> add storage iscsi://zfssa/luname.naa.
600144F0DBF8AF19000052E820D60003
zonecfg:zonename:rootzpool> end
```

   Best practice is to use ZOSS for a storage resource when configuring zones that were deployed using Unified Archives. See Chapter 13, "Getting Started With Oracle Solaris Zones on Shared Storage".

4. **Use the following commands in the global zone of the target system to view the current link configuration.**

```
target$ dladm show-link
```

```
target$ dladm show-phys
target$ ipadm show-addr
```

By default, the `zonecfg` script defines an exclusive-IP network configuration with an `anet` resource for every physical network interface that was configured on the source system. The target system automatically creates a VNIC for each `anet` resource when the zone boots.

The use of VNICs make it possible for multiple zones to share the same physical network interface. The lower-link name of an `anet` resource is initially set to *change-me* by the `zonecfg` command. You must manually set this field to the name of one of the data links on the target system. Any link that is valid for the lower link of a VNIC can be specified.

5. **Commit and exit the zone configuration.**

```
zonecfg:s11-zone> commit
zonecfg:s11-zone> exit
target#
```

Example 5 Installing a Zone From a Unified Archive File

Install the zone using the Unified Archive file created on the source system. This example does not alter the original system configuration during the installation.

```
target$ zoneadm -z s11-zone install -a s11-zone-config.uar
```

♦ ♦ ♦ CHAPTER 8

About Automatic Installation and Packages on an Oracle Solaris System With Zones Installed

You can specify installation and configuration of non-global zones as part of an AI client installation. The Image Packaging System (IPS) is supported for this release. This chapter discusses installing and maintaining the operating system by using IPS packaging when zones are installed.

For information about SVR4 packaging and patching used in `solaris10` and `native` zones, see "Chapter 25, About Packages on an Oracle Solaris System With Zones Installed (Overview)" and "Chapter 26, Adding and Removing Packages and Patches on an Oracle Solaris System With Zones Installed (Tasks)" in *System Administration Guide: Oracle Solaris Containers-Resource Management and Oracle Solaris Zones*. This is the Oracle Solaris 10 version of the guide.

Image Packaging System Software on Systems Running the Oracle Solaris Release

Graphical and command line tools enable you to download and install packages from repositories. This chapter provides information about adding packages to the installed non-global zone. Information about removing packages is also included. The material in this chapter supplements the existing Oracle Solaris installation and packaging documentation. For more information, see Chapter 3, “Installing and Updating Software Packages” in *Adding and Updating Software in Oracle Solaris 11.3*.

Zones Packaging Overview

The `solaris` packaging repository is used in administering the zones environment.

The zones automatically update when you use the `pkg` command to upgrade the system to a new version of Oracle Solaris.

The Image Packaging System (IPS), described in `pkg`(5), is a framework that provides for software lifecycle management such as installation, upgrade, and removal of packages. IPS can be used to create software packages, create and manage packaging repositories, and mirror existing packaging repositories.

After an initial installation of the Oracle Solaris operating system, you can install additional software applications from a packaging repository through the Image Packaging System CLI and GUI (Package Manager) clients.

After you have installed the packages on your system, the IPS clients can be used to search, upgrade, and manage them. The IPS clients can be also used to upgrade an entire system to a new release of Oracle Solaris, create and manage repositories, and mirror an existing repository.

If the system on which IPS is installed can access the Internet, then the clients can access and install software from the Oracle Solaris 11.3 Package Repository (default `solaris` publisher), `http://pkg.oracle.com/solaris/release/`.

The zone administrator can use the packaging tools to administer any software installed in a non-global zone, within the limits described in this document.

The following general principles apply when zones are installed:

- If a package is installed in the global zone, then the non-global zone can install the package from the system-repository service in the global zone and does not have to use the network to install that package. If that package has not been installed in the global zone, then the zone will need to use the zones-proxy service to access the publishers to install the package over the network, using the global zone.
- The global administrator or a zone administrator with the appropriate rights can administer the software on every zone on the system.
- The root file system for a non-global zone can be administered from the global zone by using the Oracle Solaris packaging tools. The Oracle Solaris packaging tools are supported within the non-global zone for administering co-packaged (bundled), standalone (unbundled), or third-party products.
- The packaging tools work in a zones-enabled environment. The tools allow a package to also be installed in a non-global zone.

Note - While certain package operations are performed, a zone is temporarily locked to other operations of this type. The system might also confirm a requested operation with the administrator before proceeding.

About Packages and Zones

The software installed in `solaris` branded zones, as described in `brands(5)`, must be compatible with the software that is installed in the global zone. The `pkg` command automatically enforces this compatibility. If the `pkg update` command is run in the global zone to update software, zones are also updated, to keep the zones in sync with the global zone. The non-global zone and global zone can have different software installed. The `pkg` command can also be used in a zone to manage software within that zone.

If the `pkg update` command (with no FMRIs specified) is run in the global zone, `pkg` will update all the software in both the global zone and any non-global zones on the system.

You can use the trial-run, also called dry-run, installation capability of `pkg install` in Oracle Solaris Zones. To perform a trial run installation, use the `pkg-install -n` command syntax. If the system generates reject messages during the trial run installation, see "Package Cannot Be Installed" in *Adding and Updating Software in Oracle Solaris 11.3* for possible resolutions to the issues.

Using a zone package variant, the various components within a package are specifically tagged to only be installed in either a global zone (`global`) or a non-global zone (`nonglobal`). A given package can contain a file that is tagged so that it will not be installed into a non-global zone.

Only a subset of the Oracle Solaris packages installed in the global zone are completely replicated when a non-global zone is installed. For example, many packages that contain the Oracle Solaris kernel are not needed in a non-global zone. All non-global zones implicitly share the same kernel from the global zone.

For more information, see "Working with Non-Global Zones" in *Adding and Updating Software in Oracle Solaris 11.3* and *Installing Oracle Solaris 11.3 Systems*.

Note - When updating the global zone on a system with non-global zones, the system might appear to display package download information twice for the zones. However, the packages are only downloaded once.

About Adding Packages in Systems With Zones Installed

On the Oracle Solaris 11 release, use the `pkg install` command.

```
# pkg install package_name
```

Using `pkg` in the Global Zone

The `pkg install` command is used in the global zone to add the package to the global zone only. The package is not propagated to any other zones.

Using the `pkg install` Command in a Non-Global Zone

The `pkg install` command is used by the zone administrator in the non-global zone to add the package to the non-global zone only. To add a package in a specified non-global zone, execute the `pkg install` command as the zone administrator.

Package dependencies are handled automatically in IPS.

Adding Additional Packages in a Zone by Using a Custom AI Manifest

The process of adding extra software in a zone at installation can be automated by revising the AI manifest. The specified packages and the packages on which they depend will be installed. The default list of packages is obtained from the AI manifest. The default AI manifest is `/usr/share/auto_install/manifest/zone_default.xml`. See *Adding and Updating Software in Oracle Solaris 11.3* for information on locating and working with packages.

EXAMPLE 6 Revising the Manifest

The following procedure adds `mercurial` and a full installation of the `vim` editor to a configured zone named *my-zone*.

Note that only the minimal `vim-core` from `solaris-small-server` is installed by default.

1. Copy the default AI manifest to the location where you will edit the file and make it writable.

```
# cp /usr/share/auto_install/manifest/zone_default.xml ~/my-zone-ai.xml
# chmod 644 ~/my-zone-ai.xml
```

2. Edit the file, adding the `mercurial` and `vim` packages to the `software_data` section as follows:

```
<software_data action="install">
```

```
    <name>pkg:/group/system/solaris-small-server</name>
    <name>pkg:/developer/versioning/mercurial</name>
    <name>pkg:/editor/vim</name>
  </software_data>
```

3. Install the zone.

```
# zoneadm -z my-zone install -m ~/my-zone-ai.xml
```

The system displays:

```
A ZFS file system has been created for this zone.
Progress being logged to /var/log/zones/zoneadm.20111113T004303Z.my-zone.install
       Image: Preparing at /zones/my-zone/root.

 Install Log: /system/volatile/install.15496/install_log
 AI Manifest: /tmp/manifest.xml.XfaWpE
  SC Profile: /usr/share/auto_install/sc_profiles/enable_sci.xml
    Zonename: my-zone
Installation: Starting ...

              Creating IPS image
              Installing packages from:
                  solaris
                      origin:  http://localhost:1008/
solaris/54453f3545de891d4daa841ddb3c844fe8804f55/

DOWNLOAD                                  PKGS       FILES    XFER (MB)
Completed                              169/169 34047/34047  185.6/185.6

PHASE                                        ACTIONS
Install Phase                              46498/46498

PHASE                                          ITEMS
Package State Update Phase                 169/169
Image State Update Phase                       2/2
Installation: Succeeded
...
```

About Removing Packages in Zones

Use the `pkg uninstall` command to remove packages on a system with zones installed.

```
# pkg uninstall package_name
```

Package Information Query

Use the `pkg info` command to query the software package database on a system with zones installed.

The command can be used in the global zone to query the software package database in the global zone only. The command can be used in a non-global zone to query the software package database in the non-global zone only.

Proxy Configuration on a System That Has Installed Zones

Persistent proxies should be set in an image by using the `--proxy` option as described in Chapter 5, "Configuring Installed Images" in *Adding and Updating Software in Oracle Solaris 11.3*. If a persistent image proxy configuration is not used, and `http_proxy` and `https_proxy` environment variables are always used to access repositories when running the `pkg` command, then the `system-repository` services should also be configured to use those same proxies through SMF `system-repository` service properties. See the `pkg`(1) man page.

Access to repositories configured in the global zone is provided to non-global zones using the `system-repository` service. Any updates to proxies for origins in the global zone are automatically made to the `system-repository` configuration. Using this method, no modifications are required to the `system-repository` SMF service.

It is also possible to configure the proxies used by the `system-repository` SMF service, overriding any proxies configured on publishers in the global zone. The `system-repository` proxies can be set by using the `config/http_proxy` or `config/https_proxy` SMF properties.

For more information, see the `pkg.sysrepo`(1M) man page and *Adding and Updating Software in Oracle Solaris 11.3*.

Configuring the Proxy in the Global Zone

You can configure the proxy directly in the global zone, and any updates to proxies for origins in the global zone are automatically made to the `system-repository` configuration. The `system-repository` service does not require modification.

EXAMPLE 7 Configuring the Proxy in the Global Zone

```
# pkg set-publisher --proxy http://www-proxy -g http://pkg-server pub
```

No port specification is required unless the proxy accepts connections on a port other than `80`.

If zones are on the system, the system-repository service is restarted, and the proxy is used to provide access to *pkg-server*.

Overriding `system-repository` Proxies by Using `https_proxy` and `http_proxy`

Proxies should be set in an image, and only the `system-repository` service proxy should be set. The `https_proxy` and `http_proxy` should be set in the environment when running the `pkg` command.

The procedures in this section are used to set proxies in the `system-repository` service on an internal subnet that does not have a direct connection to the IPS publisher repository. Use of this procedure overrides any proxies configured by the `pkg` command in the global zone. Non-global zones communicate with the `system-repository` over HTTP. The `system-repository` then accesses the publishers using the protocol for that repository as configured in the global zone.

This configuration allows the `solaris` non-global zones to contact the publisher set in the global zone as well. Recursive `pkg` operations into the `solaris` zones will succeed.

EXAMPLE 8 Using `https_proxy` and `http_proxy` To Override Global Zone Proxies

For example, assume that the software on a system running `solaris` non-global zones is managed by IPS and requires the use of the proxy server `http_proxy=http://`*192.0.2.0* for access to both `http` and `https` URLs. The following steps show how to use `http_proxy` and `https_proxy` environment variables and SMF service properties to allow the global zone and non-global zones to access the IPS repositories.

Note that these variables override any proxy configuration set on the origin unless the user is running the `pkg` command from a non-global zone to connect to the Universal Resource Identifier (URI) for a system publisher. In this case, the command goes through the `system-repository`.

A host name that can be resolved can also be used.

1. Type the following lines to set the proxy in the `shell` for the global zone:

```
# export http_proxy=http://192.0.2.0
# export https_proxy=http://192.0.2.0
```

Setting the proxy allows `pkg` commands to reach the publisher through the proxy server. This affects `pkg` operations that use an `https` or `http` URL and do not go through the `system-repository` for the global zone.

2. To allow the `solaris` zones on the system to use the configured system publishers directly accessible from the global zone, execute the following commands:

```
# svccfg -s system-repository:default setprop config/http_proxy = http://192.0.2.0
# svccfg -s system-repository:default setprop config/https_proxy = http://192.0.2.0
```

3. To make the change take effect in the live SMF repository, run:

```
# svcadm refresh system-repository
```

4. To confirm that the setting is in effect, run:

```
# svcprop -p config/http_proxy system-repository
# svcprop -p config/https_proxy system-repository
```

For more information about the `pkg` command, see the `pkg(1)` man page.

Parallel Zone Updates

Zones can be configured to be updated in parallel instead of serially. The parallel update provides a significant improvement in the time required to update all the zones on a system. For additional information and a configuration example, see "Updating Multiple Non-Global Zones Concurrently" in *Adding and Updating Software in Oracle Solaris 11.3*.

How Zone State Affects Package Operations

The following table describes what will happen when packaging commands are used on a system with non-global zones in various states.

Zone State	Effect on Package Operations
Configured	Package tools can be run. No software has been installed yet.
Incomplete	If `zoneadm` is operating on the zone, package tools should not be used. If no `zoneadm` process is operating on the zone, package operations are safe to run, but no software in this zone will be changed and any software in the zone will not affect dependency resolution.

Zone State	Effect on Package Operations
Unavailable	The software image within the zone is not accessible. The software image will not be changed, nor will it affect dependency resolution.
Installed	Package tools can be run. Note that immediately after `zoneadm -z` *zonename* `install` has completed, the zone is also moved to the installed state.
Ready	Package tools can be run.
Running	Package tools can be run.

A non-global zone transitions to the unavailable state when the storage for the zone is not accessible or when the image of the zone, which is described in `pkg`(5), is out of sync with the global zone's image. This state transition occurs to prevent a problem that is affecting a non-global zone from blocking package operations in the global zone.

When a zone's storage is temporarily unavailable and package operations that change the version of installed software occur, it is likely that after fixing the storage problem, the zone might need to be attached by using one of the `solaris` brand's `attach` options that allow for updates. For example, `zoneadm -z` *zonename* `attach -u` might be required to synchronize versions of critical software between the global zone and a non-global zone that is in the unavailable state.

♦ ♦ ♦ CHAPTER 9

About Oracle Solaris Zones Administration

This chapter covers these general zone administration topics:

For information about `solaris10` branded zones, see *Creating and Using Oracle Solaris 10 Zones*.

Global Zone Visibility and Access

The global zone acts as both the default zone for the system and as a zone for system-wide administrative control. There are administrative issues associated with this dual role. Since applications within the zone have access to processes and other system objects in other zones, the effect of administrative actions can be wider than expected. For example, service shutdown scripts often use `pkill` to signal processes of a given name to exit. When such a script is run from the global zone, all such processes in the system will be signaled, regardless of zone.

The system-wide scope is often needed. For example, to monitor system-wide resource usage, you must view process statistics for the whole system. A view of just global zone activity would miss relevant information from other zones in the system that might be sharing some or all of the system resources. Such a view is particularly important when system resources such as CPU are not strictly partitioned using resource management facilities.

Thus, processes in the global zone can observe processes and other objects in non-global zones. This allows such processes to have system-wide observability. The ability to control or send signals to processes in other zones is restricted by the privilege `PRIV_PROC_ZONE`. The privilege is similar to `PRIV_PROC_OWNER` because the privilege allows processes to override the restrictions placed on unprivileged processes. In this case, the restriction is that unprivileged processes in the global zone cannot signal or control processes in other zones. This is true even when the user IDs of the processes match or the acting process has the `PRIV_PROC_OWNER` privilege. The `PRIV_PROC_ZONE` privilege can be removed from otherwise privileged processes to restrict actions to the global zone.

For information about matching processes by using a `zoneidlist`, see the `pgrep(1)` `pkill(1)` man pages.

Process ID Visibility in Zones

Only processes in the same zone will be visible through system call interfaces that take process IDs, such as the `kill` and `priocntl` commands. For information, see the `kill(1)` and the `priocntl(1)` man pages.

System Observability in Zones

When the `ps` command is run in the global zone, user and group names are resolved using the global zone's name services. Processes running in a non-global zone on the system display user and group names that match the global zone's name services. These global zone names might be different than the names configured in name services in the non-global zones.

The `ps` command has the following modifications:

- The `-o` option is used to specify output format. This option allows you to print the zone ID of a process or the name of the zone in which the process is running.
- The `-z` *zonelist* option is used to list only processes in the specified zones. Zones can be specified either by zone name or by zone ID. This option is only useful when the command is executed in the global zone.
- The `-Z` option is used to print the name of the zone associated with the process. The name is printed under the column heading `ZONE`.

For more information, see the `ps`(1) man page.

A `-z` *zonename* option has been added to the following Oracle Solaris utilities. You can use this option to filter the information to include only the zone or zones specified.

- `ipcs` (see the `ipcs`(1) man page)
- `pgrep` (see the `pgrep`(1) man page)
- `ptree` (see the `proc`(1) man page)
- `prstat` (see the `prstat`(1M) man page)

See Table 5, “Commands Modified for Use on an Oracle Solaris System With Zones Installed,” on page 154 for the full list of changes made to commands.

Reporting Active Zone Statistics with the `zonestat` Utility

To use the `zonestat` utility, see the `zonestat`(1) man page and “Using the `zonestat` Utility in a Non-Global Zone” on page 163.

The `zonestat` utility reports on the CPU, memory, and resource control utilization of the currently running zones. The `zonestat` utility prints a series of reports at specified intervals. Optionally, the utility can print one or more summary reports.

The zonestat utility also reports on network bandwidth utilization in exclusive-IP zones. An exclusive-IP zone has its own IP-related state and one or more dedicated data-links.

When run from within a non-global zone, only processor sets visible to that zone are reported. The non-global zone output will include all of the memory resources, and the limits resource.

The zonestat service in the global zone must be online to use the zonestat service in the non-global zones. The zonestat service in each non-global zone reads system configuration and utilization data from the zonestat service in the global zone.

The zonestatd system daemon is started during system boot. The daemon monitors the utilization of system resources by zones, as well as zone and system configuration information such as psrset processor sets, pool processor sets, and resource control settings. There are no configurable components.

Monitoring Non-Global Zones Using the fsstat Utility

The fsstat utility collects and prints kstats per zone, including aggregations. By default, the utility reports an aggregate of all running zones. A per-fstype kstat is produced for each zone. The global zone kstat reports its exclusive activity. The global zone can see the kstats of all zones on the system. Non-global zones only see the kstats associated with the zone in which the utility is run. A non-global zone cannot monitor file system activity in other zones.

For more information, see the fsstat(1M) man page and “Reporting Per-Zone fstype Statistics for all Zones” on page 167.

Non-Global Zone Node Name

The node name is the local source for the system name. The node name must be unique, such as the zone name. The node name can be set by the zone administrator.

```
$ hostname myhostname
```

To view the hostname, type hostname.

```
$ hostname
…
```

myhostname

Running an NFS Server in a Zone

The NFS server package `svc:/network/nfs/server:default` must be installed in the zone to create NFS shares in a zone.

The `sys_share` privilege can be prohibited in the zone configuration to prevent NFS sharing within a zone. See Table 1, “Status of Privileges in Zones,” on page 144.

Restrictions and limitations include the following:

- Cross-zone LOFS mounts cannot be shared from zones.
- File systems mounted within zones cannot be shared from the global zone.
- NFS over Remote Direct Memory Access (RDMA) is not supported in zones.
- Oracle Sun Cluster HA for NFS (HANFS) failover is not supported in zones.

See *Introduction to Oracle Solaris 11.3 Network Services*.

File Systems and Non-Global Zones

This section provides information about file system issues on an Oracle Solaris system with zones installed. Each zone has its own section of the file system hierarchy, rooted at a directory known as the zone `root`. Processes in the zone can access only files in the part of the hierarchy that is located under the zone root. The `chroot` utility can be used in a zone, but only to restrict the process to a root path within the zone. For more information about `chroot`, see `chroot(1M)`.

The `-o nosuid` Option

The `-o nosuid` option to the `mount` utility has the following functionality:

- Processes from a `setuid` binary located on a file system that is mounted using the `nosetuid` option do not run with the privileges of the `setuid` binary. The processes run with the privileges of the user that executes the binary.

  For example, if a user executes a `setuid` binary that is owned by `root`, the processes run with the privileges of the user.

- Opening device-special entries in the file system is not allowed. This behavior is equivalent to specifying the `nodevices` option.

This file system-specific option is available to all Oracle Solaris file systems that can be mounted with `mount` utilities, as described in the `mount`(1M) man page. In this guide, these file systems are listed in "Mounting File Systems in Zones" on page 128. Mounting capabilities are also described. For more information about the `-o nosuid` option, see Chapter 7, "Accessing Network File Systems" in *Managing Network File Systems in Oracle Solaris 11.3*.

Mounting File Systems in Zones

When file systems are mounted from within a zone, the `nodevices` option applies. For example, if a zone is granted access to a block device (`/dev/dsk/c0t0d0s7`) and a raw device (`/dev/rdsk/c0t0d0s7`) corresponding to a UFS file system, the file system is automatically mounted `nodevices` when mounted from within a zone. This rule does not apply to mounts specified through a `zonecfg` configuration.

Options for mounting file systems in non-global zones are described in the following table. Procedures for these mounting alternatives are provided in "Configuring, Verifying, and Committing a Zone" on page 22 and "Mounting File Systems in Running Non-Global Zones" on page 170.

Any file system type not listed in the table can be specified in the configuration if it has a mount binary in `/usr/lib/`*`fstype`*`/mount`.

To mount file system types other than HSFS and NFS from inside the non-global zone, also add the file system type to the configuration by using the `zonecfg fs-allowed` property.

Allowing file system mounts other than the default might allow the zone administrator to compromise the system.

File System	Mounting Options in a Non-Global Zone
AutoFS	Cannot be mounted using `zonecfg`. Can be mounted from within the zone.
CacheFS	Cannot be used in a non-global zone.
FDFS	Can be mounted using `zonecfg`, can be mounted from within the zone.
HSFS	Can be mounted using `zonecfg`, can be mounted from within the zone.
LOFS	Can be mounted using `zonecfg`, can be mounted from within the zone.
MNTFS	Cannot be mounted using `zonecfg`. Can be mounted from within the zone.
NFS	Cannot be mounted using `zonecfg`. V2, V3, and V4, which are the versions currently supported in zones, can be mounted from within the zone.
PCFS	Can be mounted using `zonecfg`, can be mounted from within the zone.

File System	Mounting Options in a Non-Global Zone
PROCFS	Cannot be mounted using `zonecfg`. Can be mounted from within the zone.
TMPFS	Can be mounted using `zonecfg`, can be mounted from within the zone.
UDFS	Can be mounted using `zonecfg`, can be mounted from within the zone.
UFS	Can be mounted using `zonecfg`, can be mounted from within the zone. **Note -** The `quota` command documented in `quota(1M)` cannot be used to retrieve quota information for UFS file systems added through the `zonecfg add fs` resource. The `system/file-system/ufs` package must be installed in the global zone if `add fs` is used. To use UFS file systems in a non-global zone through the `zonecfg` command, the package must be installed into the zone after installation or through the AI manifest script. The following is typed as one line: `global$` **`pkg -R /tank/zones/my-zone/root \`** **`install system/file-system/ufs`**
VxFS	Can be mounted using `zonecfg`, can be mounted from within the zone.
ZFS	Can be mounted using the `zonecfg dataset` and `fs` resource types.

For more information, see "How to Configure the Zone" on page 22, "Mounting File Systems in Running Non-Global Zones" on page 170, and the `mount(1M)` man page.

Unmounting File Systems in Zones

The ability to unmount a file system will depend on who performed the initial mount. If a file system is specified as part of the zone's configuration using the `zonecfg` command, then the global zone owns this mount and the non-global zone administrator cannot unmount the file system. If the file system is mounted from within the non-global zone, for example, by specifying the mount in the zone's `/etc/vfstab` file, then the non-global zone administrator can unmount the file system.

Security Restrictions and File System Behavior

There are security restrictions on mounting certain file systems from within a zone. Other file systems exhibit special behavior when mounted in a zone. The list of modified file systems follows.

AutoFS

AutoFS is a client-side service that automatically mounts the appropriate file system. When a client attempts to access a file system that is not presently mounted, the AutoFS

file system intercepts the request and calls `automountd` to mount the requested directory. AutoFS mounts established within a zone are local to that zone. The mounts cannot be accessed from other zones, including the global zone. The mounts are removed when the zone is halted or rebooted. For more information on AutoFS, see "How Autofs Works" in *Managing Network File Systems in Oracle Solaris 11.3*.

Each zone runs its own copy of `automountd`. The auto maps and timeouts are controlled by the zone administrator. You cannot trigger a mount in another zone by crossing an AutoFS mount point for a non-global zone from the global zone.

Certain AutoFS mounts are created in the kernel when another mount is triggered. Such mounts cannot be removed by using the regular `umount` interface because they must be mounted or unmounted as a group. Note that this functionality is provided for zone shutdown.

MNTFS

MNTFS is a virtual file system that provides read-only access to the table of mounted file systems for the local system. The set of file systems visible by using `mnttab` from within a non-global zone is the set of file systems mounted in the zone, plus an entry for root (`/`) . Mount points with a special device that is not accessible from within the zone, such as `/dev/rdsk/c0t0d0s0`, have their special device set to the same as the mount point. All mounts in the system are visible from the global zone's `/etc/mnttab` table. For more information on MNTFS, see "Mounting File Systems" in *Managing Network File Systems in Oracle Solaris 11.3*.

NFS

NFS mounts established within a zone are local to that zone. The mounts cannot be accessed from other zones, including the global zone. The mounts are removed when the zone is halted or rebooted.

From within a zone, NFS mounts behave as though mounted with the `nodevices` option.

The `nfsstat` command output only pertains to the zone in which the command is run. For example, if the command is run in the global zone, only information about the global zone is reported. For more information about the `nfsstat` command, see `nfsstat(1M)`.

PROCFS

The `/proc` file system, or PROCFS, provides process visibility and access restrictions as well as information about the zone association of processes. Only processes in the same zone are visible through `/proc`.

Processes in the global zone can observe processes and other objects in non-global zones. This allows such processes to have system-wide observability.

From within a zone, `procfs` mounts behave as though mounted with the `nodevices` option. For more information about `procfs`, see the `proc(4)` man page.

LOFS

The scope of what can be mounted through LOFS is limited to the portion of the file system that is visible to the zone. Hence, there are no restrictions on LOFS mounts in a zone.

UFS, UDFS, PCFS, and other storage-based file systems

When using the `zonecfg` command to configure storage-based file systems that have an `fsck` binary, such as UFS, the zone administrator must specify a `raw` parameter. The parameter indicates the raw (character) device, such as `/dev/rdsk/c0t0d0s7`. The `zoneadmd` daemon automatically runs the `fsck` command in preen mode (`fsck -p`), which checks and fixes the file system non-interactively, before it mounts the file system. If the `fsck` fails, `zoneadmd` cannot bring the zone to the ready state. The path specified by `raw` cannot be a relative path.

It is an error to specify a device to `fsck` for a file system that does not provide an `fsck` binary in `/usr/lib/fs/`*fstype*`/fsck`. It is also an error if you do not specify a device to `fsck` if an `fsck` binary exists for that file system.

For more information, see "The `zoneadmd` Daemon" on page 41 and the `fsck`(1M) command.

ZFS

In addition to the default dataset described in "File Systems Mounted in Zones" in *Oracle Solaris Zones Configuration Resources*, you can add a ZFS dataset to a non-global zone by using the `zonecfg` command with the `add dataset` resource. The dataset is visible and mounted in the non-global zone, and also visible in the global zone. The zone administrator can create and destroy file systems within that dataset, and modify the properties of the dataset.

The `zoned` attribute of `zfs` indicates whether a dataset has been added to a non-global zone.

```
# zfs get zoned tank/sales
NAME          PROPERTY    VALUE      SOURCE
tank/sales    zoned       on         local
```

Each dataset that is delegated to a non-global zone through a dataset resource is aliased. The dataset layout is not visible within the zone. Each aliased dataset appears in the zone as if it were a pool. The default alias for a dataset is the last component in the dataset name. For example, if the default alias is used for the delegated dataset `tank/sales`, the zone will see a virtual ZFS pool named `sales`. The alias can be customized to be a different value by setting the alias property within the dataset resource.

A dataset named `rpool` exists within each non-global zone's `zonepath` dataset. For all non-global zones, this zone `rpool` dataset is aliased as `rpool`.

```
my-zone# zfs list -o name,zoned,mounted,mountpoint
```

```
NAME                 ZONED  MOUNTED  MOUNTPOINT
rpool                   on       no  /rpool
rpool/ROOT              on       no  legacy
rpool/ROOT/solaris      on      yes  /
rpool/export            on       no  /export
rpool/export/home       on       no  /export/home
```

Dataset aliases are subject to the same name restrictions as ZFS pools. These restrictions are documented in the `zpool`(1M) man page.

If you want to share a dataset from the global zone, you can add an LOFS-mounted ZFS file system by using the `zonecfg` command with the `add fs` subcommand. The global administrator or a user granted the appropriate authorizations is responsible for setting and controlling the properties of the dataset.

For more information on ZFS, see Chapter 10, "Oracle Solaris ZFS Advanced Topics" in *Managing ZFS File Systems in Oracle Solaris 11.3*.

Non-Global Zones as NFS Clients

Zones can be NFS clients. Version 2, version 3, and version 4 protocols are supported. For information on these NFS versions, see "Features of the NFS Service" in *Managing Network File Systems in Oracle Solaris 11.3*.

The default version is NFS version 4. You can enable other NFS versions on a client by using one of the following methods:

- You can use `sharectl`(1M) to set properties. Set `NFS_CLIENT_VERSMAX=`*number* so that the zone uses the specified version by default. See "Setting Up the NFS Service" in *Managing Network File Systems in Oracle Solaris 11.3*.
- You can manually create a version mount. This method overrides `sharectl` setting. See "Setting Up the NFS Service" in *Managing Network File Systems in Oracle Solaris 11.3*.

Use of `mknod` Prohibited in a Zone

Note that you cannot use the `mknod` command documented in the `mknod`(1M) man page to make a special file in a non-global zone.

Traversing File Systems

A zone's file system namespace is a subset of the namespace accessible from the global zone. Unprivileged processes in the global zone are prevented from traversing a non-global zone's file system hierarchy through the following means:

- Specifying that the zone root's parent directory is owned, readable, writable, and executable by root only
- Restricting access to directories exported by `/proc`

Note that attempting to access AutoFS nodes mounted for another zone will fail. The global administrator must not have auto maps that descend into other zones.

Restriction on Accessing A Non-Global Zone From the Global Zone

After a non-global zone is installed, the zone must never be accessed directly from the global zone by any commands other than system backup utilities. Moreover, a non-global zone can no longer be considered secure after it has been exposed to an unknown environment. An example would be a zone placed on a publicly accessible network, where it would be possible for the zone to be compromised and the contents of its file systems altered. If there is any possibility that compromise has occurred, the global administrator should treat the zone as untrusted.

Any command that accepts an alternative root by using the `-R` or `-b` options (or the equivalent) must *not* be used when the following are true:

- The command is run in the global zone.
- The alternative root refers to any path within a non-global zone, whether the path is relative to the current running system's global zone or the global zone in an alternative root.

An example is the `-R` *root_path* option to the `pkgadd` utility run from the global zone with a non-global zone root path.

The list of commands, programs, and utilities that use `-R` with an alternative root path include the following:

- `auditreduce`
- `bart`
- `installf`
- `localeadm`
- `makeuuid`

- `metaroot`
- `pkg`
- `prodreg`
- `removef`
- `routeadm`
- `showrev`
- `syseventadm`

The list of commands and programs that use `-b` with an alternative root path include the following:

- `add_drv`
- `pprosetup`
- `rem_drv`
- `roleadd`
- `update_drv`
- `useradd`

Networking in Shared-IP Non-Global Zones

Zones installed on an Oracle Solaris system can communicate with each other over the network. The zones have separate bindings, or connections, and can run their own server daemons. These daemons can listen on the same port numbers without any conflict. The IP stack resolves conflicts by considering the IP addresses for incoming connections. The IP addresses identify the zone.

In Oracle Solaris, the network configuration of a specific system is managed by an active network configuration profile (NCP) that is enabled either automatically during an Oracle Solaris installation or manually by a system administrator. Only one NCP can be active on the system at one time.

To use shared-IP type zones, the operative NCP on the global zone must be the `DefaultFixed` NCP. To check which NCP is currently active on the system, type the following command:

```
# netadm list
TYPE   PROFILE         STATE
ncp    DefaultFixed    online
ncp    Automatic       disabled
loc    Automatic       offline
loc    NoNet           offline
```

```
loc     DefaultFixed    online
```

The NCP with the `online` state is the operative or active network configuration profile on the system. If `DefaultFixed` is offline, then enable the profile with the following command:

```
# netadm enable DefaultFixed
# svcprop -p netcfg/active_ncp svc:/network/physical:default
DefaultFixed
```

Shared-IP Zone Partitioning

Shared-IP is not the default, but this type is supported.

The IP stack in a system supporting zones implements the separation of network traffic between zones. Applications that receive IP traffic can only receive traffic sent to the same zone.

Each logical interface on the system belongs to a specific zone, the global zone by default. Logical network interfaces assigned to zones though the `zonecfg` utility are used to communicate over the network. Each stream and connection belongs to the zone of the process that opened it.

Bindings between upper-layer streams and logical interfaces are restricted. A stream can only establish bindings to logical interfaces in the same zone. Likewise, packets from a logical interface can only be passed to upper-layer streams in the same zone as the logical interface.

Each zone has its own set of binds. Each zone can be running the same application listening on the same port number without binds failing because the address is already in use. Each zone can run its own version of various networking service such as the followings:

- Internet services daemon with a full configuration file (see the `inetd`(1M) man page)
- `sendmail` (see the `sendmail`(1M) man page)
- `apache`

Zones other than the global zone have restricted access to the network. The standard TCP and UDP socket interfaces are available, but `SOCK_RAW` socket interfaces are restricted to Internet Control Message Protocol (ICMP). ICMP is necessary for detecting and reporting network error conditions or using the `ping` command.

Shared-IP Network Interfaces

Each non-global zone that requires network connectivity has one or more dedicated IP addresses. These addresses are associated with logical network interfaces that can be placed in a

zone. Zone network interfaces configured by `zonecfg` will automatically be set up and placed in the zone when it is booted. The `ipadm` command can be used to add or remove logical interfaces when the zone is running. Only the global administrator or a user granted the appropriate authorizations can modify the interface configuration and the network routes.

Within a non-global zone, only that zone's interfaces are visible to the `ipadm` command.

For more information, see the `ipadm`(1M) and `if_tcp`(7P) man pages.

IP Traffic Between Shared-IP Zones on the Same Machine

A shared-IP zone can reach any given IP destination if there is a usable route for that destination in its routing table. To view the routing table, use the `netstat` command with the `-r` option from within the zone. The IP forwarding rules are the same for IP destinations in other zones or on other systems.

Oracle Solaris IP Filter in Shared-IP Zones

Oracle Solaris IP Filter provides stateful packet filtering and network address translation (NAT). A stateful packet filter can monitor the state of active connections and use the information obtained to determine which network packets to allow through the firewall. Oracle Solaris IP Filter also includes stateless packet filtering and the ability to create and manage address pools. See Chapter 6, “IP Filter Firewall in Oracle Solaris” in *Securing the Network in Oracle Solaris 11.3* for additional information.

Oracle Solaris IP Filter can be enabled in non-global zones by turning on loopback filtering as described in Chapter 7, “Configuring IP Filter Firewall” in *Securing the Network in Oracle Solaris 11.3*.

Oracle Solaris IP Filter is derived from open source IP Filter software.

IP Network Multipathing in Shared-IP Zones

In Oracle Solaris, IP network multipathing (IPMP) ensures continuous network availability by grouping multiple interfaces on the same IP link. These underlying interfaces back up one another such that the network remains available if one of the underlying interfaces fail. IPMP also provides load spreading of packets for systems with multiple interfaces.

IPMP is implemented in Oracle Solaris in the following manner:

- Multiple interfaces such as `net0`, `net1`, and `net2` are configured to form an IPMP interface `ipmp0`.
- The IPMP interface `ipmp0` is configured with multiple IP addresses called data addresses. These addresses are used to host network traffic.
- IP addresses can also be configured directly on the underlying interfaces `net`*N*. These addresses are not used for network traffic but for failure detection to determine whether an underlying interface has failed. Thus, these IP addresses on the underlying interfaces are called test addresses.

Because the data addresses reside on `ipmp0`, the network remains available even if one underlying IP interface fails. Traffic continues to flow through the other addresses in `ipmp0`.

As with all network configuration tasks, you configure IPMP on the global zone. Then, you extend the functionality to non-global zones. The functionality is extended by assigning one of the IPMP interface's data address to the zone.

In a given non-global zone, only the interfaces associated with the zone are visible through the `ipadm` command.

See "How to Extend IP Network Multipathing Functionality to Shared-IP Non-Global Zones" on page 175. The zones configuration procedure is covered in "How to Configure the Zone" on page 22. For information on IPMP features, components, and usage, see Chapter 2, "About IPMP Administration" in *Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.3*.

Networking in Exclusive-IP Non-Global Zones

An exclusive-IP zone has its own IP-related state. The zone is assigned its own set of datalinks when the zone is configured.

Packets are transmitted on the physical link. Then, devices like Ethernet switches or IP routers can forward the packets toward their destination, which might be a different zone on the same system as the sender.

For virtual links, the packet is first sent to a virtual switch. If the destination link is over the same device, such as a VNIC on the same physical link or etherstub, the packet will go directly to the destination VNIC. Otherwise, the packet will go out the physical link underlying the VNIC.

For information on features that can be used in an exclusive-IP non-global zone, see "Exclusive-IP Non-Global Zones" in *Oracle Solaris Zones Configuration Resources*.

Exclusive-IP Zone Partitioning

Exclusive-IP zones have separate TCP/IP stacks, so the separation reaches down to the datalink layer. One or more datalink names, which can be a NIC or a VLAN on a NIC, are assigned to an exclusive-IP zone by the global administrator. The zone administrator can configure IP on those datalinks with the same flexibility and options as in the global zone.

Exclusive-IP Datalink Interfaces

A datalink name must be assigned exclusively to a single zone.

The `dladm show-link` command can be used to display datalinks assigned to running zones.

```
sol-t2000-10{pennyc}1: dladm show-link
LINK                CLASS     MTU    STATE    OVER
vsw0                phys      1500   up       --
net0                phys      1500   up       --
netg2               phys      1500   up       --
netg1               phys      1500   up       --
netg3               phys      1500   up       --
zoneA/net0          vnic      1500   up       net0
zoneB/net0          vnic      1500   up       net0
aggr1               aggr      1500   up       net2 net3
vnic0               vnic      1500   up       net1
zoneA/vnic0         vnic      1500   up       net1
vnic1               vnic      1500   up       net1
zoneB/vnic1         vnic      1500   up       net1
vnic3               vnic      1500   up       aggr1
vnic4               vnic      1500   up       aggr1
zoneB/vnic4         vnic      1500   up       aggr1
```

For more information, see `dladm`(1M).

IP Traffic Between Exclusive-IP Zones on the Same System

There is no internal loopback of IP packets between exclusive-IP zones. All packets are sent down to the datalink. Typically, this means that the packets are sent out on a network

interface. Then, devices like Ethernet switches or IP routers can forward the packets toward their destination, which might be a different zone on the same system as the sender.

Oracle Solaris IP Filter in Exclusive-IP Zones

You have the same IP Filter functionality that you have in the global zone in an exclusive-IP zone. IP Filter is also configured the same way in exclusive-IP zones and the global zone.

IP Network Multipathing in Exclusive-IP Zones

IP network multipathing (IPMP) provides physical interface failure detection and transparent network access failover for a system with multiple interfaces on the same IP link. IPMP also provides load spreading of packets for systems with multiple interfaces.

The datalink configuration is done in the global zone. First, multiple datalink interfaces are assigned to a zone using `zonecfg`. The multiple datalink interfaces must be attached to the same IP subnet. IPMP can then be configured from within the exclusive-IP zone by the zone administrator.

Device Use in Non-Global Zones

The set of devices available within a zone is restricted to prevent a process in one zone from interfering with processes running in other zones. For example, a process in a zone cannot modify kernel memory or modify the contents of the root disk. Thus, by default, only certain pseudo-devices that are considered safe for use in a zone are available. Additional devices can be made available within specific zones by using the `zonecfg` utility.

`/dev` and the `/devices` Namespace

The `devfs` file system described in the `devfs`(7FS) man page is used by the Oracle Solaris system to manage `/devices`. Each element in this namespace represents the physical path to a hardware device, pseudo-device, or nexus device. The namespace is a reflection of the device tree. As such, the file system is populated by a hierarchy of directories and device special files.

Devices are grouped according to the relative `/dev` hierarchy. For example, all of the devices under `/dev` in the global zone are grouped as global zone devices. For a non-global zone, the

devices are grouped in a `/dev` directory under the zone's root path. Each group is a mounted `/dev` file system instance that is mounted under the `/dev` directory. Thus, the global zone devices are mounted under `/dev`, while the devices for a non-global zone named `my-zone` are mounted under `/my-zone/root/dev`.

The `/dev` file hierarchy is managed by the `dev` file system described in the `dev(7FS)` man page.

Caution - Subsystems that rely on `/devices` path names are not able to run in non-global zones. The subsystems must be updated to use `/dev` path names.

Caution - If a non-global zone has a device resource with a match that includes devices within `/dev/zvol`, it is possible that namespace conflicts can occur within the non-global zone. For more information, see the `dev(7FS)` man page.

Exclusive-Use Devices

You might have devices that you want to assign to specific zones. Allowing unprivileged users to access block devices could permit those devices to be used to cause system panic, bus resets, or other adverse effects. Before making such assignments, consider the following issues:

- Before assigning a SCSI tape device to a specific zone, consult the `sgen(7D)` man page.
- Placing a physical device into more than one zone can create a covert channel between zones. Global zone applications that use such a device risk the possibility of compromised data or data corruption by a non-global zone.

Device Driver Administration

In a non-global zone, you can use the `modinfo` command described in the `modinfo(1M)` man page to examine the list of loaded kernel modules.

Most operations concerning kernel, device, and platform management will not work inside a non-global zone because modifying platform hardware configurations violates the zone security model. These operations include the following:

- Adding and removing drivers
- Explicitly loading and unloading kernel modules
- Initiating dynamic reconfiguration (DR) operations
- Using facilities that affect the state of the physical platform

Utilities That Do Not Work or Are Modified in Non-Global Zones

Utilities That Do Not Work in Non-Global Zones

The following utilities do not work in a zone because they rely on devices that are not normally available:

- `add_drv` (see the `add_drv`(1M) man page)
- `disks` (see the `disks`(1M) man page)
- `prtconf` (see the `prtconf`(1M) man page)
- `prtdiag` (see the `prtdiag`(1M) man page)
- `rem_drv` (see the `rem_drv`(1M) man page)

SPARC: Utility Modified for Use in a Non-Global Zone

The `eeprom` utility can be used in a zone to view settings. The utility cannot be used to change settings. For more information, see the `eeprom`(1M) and `openprom`(7D) man pages.

Allowed Utilities With Security Implications

If `allowed-raw-io` is enabled, the following utilities can be used in a zone. Note that security considerations must be evaluated. Before adding devices, see "Device Use in Non-Global Zones" on page 139, "Running Applications in Non-Global Zones" on page 141, and "Privileges in a Non-Global Zone" on page 144 for restrictions and security concerns.

- `cdrecord` (see the`cdrecord`(1) man page).
- `cdrw` (see the `cdrw`(1) man page).
- `rmformat` (see the `rmformat`(1) man page).

Running Applications in Non-Global Zones

In general, all applications can run in a non-global zone. However, the following types of applications might not be suitable for this environment:

- Applications that use privileged operations that affect the system as a whole. Examples include operations that set the global system clock or lock down physical memory.

- The few applications dependent upon certain devices that do not exist in a non-global zone, such as `/dev/kmem`.
- In a shared-IP zone, applications that are dependent upon devices in `/dev/ip`.

Resource Controls Used in Non-Global Zones

For additional information about using a resource management feature in a zone, also refer to the chapter that describes the capability in *Administering Resource Management in Oracle Solaris 11.3*.

Any of the resource controls and attributes described in the resource management chapters can be set in the global and non-global zone `/etc/project` file, NIS map, or LDAP directory service. The settings for a given zone affect only that zone. A project running autonomously in different zones can have controls set individually in each zone. For example, Project A in the global zone can be set `project.cpu-shares=10` while Project A in a non-global zone can be set `project.cpu-shares=5`. You could have several instances of `rcapd` running on the system, with each instance operating only on its zone.

The resource controls and attributes used in a zone to control projects, tasks, and processes within that zone are subject to the additional requirements regarding pools and the zone-wide resource controls.

A non-global zone can be associated with one resource pool, although the pool need not be exclusively assigned to a particular zone. Multiple non-global zones can share the resources of one pool. Processes in the global zone, however, can be bound by a sufficiently privileged process to any pool. The resource controller `poold` only runs in the global zone, where there is more than one pool for it to operate on. The `poolstat` utility run in a non-global zone displays only information about the pool associated with the zone. The `pooladm` command run without arguments in a non-global zone displays only information about the pool associated with the zone.

Zone-wide resource controls do not take effect when they are set in the `project` file. A zone-wide resource control is set through the `zonecfg` utility.

Fair Share Scheduler on a System With Zones Installed

This section describes how to use the fair share scheduler (FSS) with zones.

FSS Share Division in a Global or Non-Global Zone

FSS CPU shares for a zone are hierarchical. The shares for the global and non-global zones are set by the global administrator through the zone-wide resource control `zone.cpu-shares`. The `project.cpu-shares` resource control can then be defined for each project within that zone to further subdivide the shares set through the zone-wide control.

To assign zone shares by using the `zonecfg` command, see "How to Set `zone.cpu-shares` in the Global Zone" on page 33. For more information on `project.cpu-shares`, see "Available Resource Controls" in *Administering Resource Management in Oracle Solaris 11.3*. Also see "Using the Fair Share Scheduler on an Oracle Solaris System With Zones Installed" on page 180 for example procedures that show how to set shares on a temporary basis.

Share Balance Between Zones

You can use `zone.cpu-shares` to assign FSS shares in the global zone and in non-global zones. If FSS is the default scheduler on your system and shares are not assigned, each zone is given one share by default. If you have one non-global zone on your system and you give this zone two shares through `zone.cpu-shares`, that defines the proportion of CPU which the non-global zone will receive in relation to the global zone. The ratio of CPU between the two zones is 2:1.

Extended Accounting on a System With Zones Installed

The extended accounting subsystem collects and reports information for the entire system (including non-global zones) when run in the global zone. The global administrator can also determine resource consumption on a per-zone basis.

The extended accounting subsystem permits different accounting settings and files on a per-zone basis for process-based and task-based accounting. The `exacct` records can be tagged with the zone name `EXD PROC ZONENAME` for processes, and the zone name `EXD TASK ZONENAME` for tasks. Accounting records are written to the global zone's accounting files as well as the per-zone accounting files. The `EXD TASK HOSTNAME`, `EXD PROC HOSTNAME`, and `EXD HOSTNAME` records contain the `uname -n` value for the zone in which the process or task executed instead of the global zone's node name.

Privileges in a Non-Global Zone

Processes are restricted to a subset of privileges. Privilege restriction prevents a zone from performing operations that might affect other zones. The set of privileges limits the capabilities of privileged users within the zone.

To display the list of privileges available for a given zone, use the `ppriv` utility. From within the zone, use the `ppriv -l` command. From the global zone, add the zone name to the command: `ppriv -l` *zonename*.

The following table lists all of the Oracle Solaris privileges and the status of each privilege with respect to zones. Optional privileges are not part of the default set of privileges but can be specified through the `limitpriv` property. Required privileges must be included in the resulting privilege set. Prohibited privileges cannot be included in the resulting privilege set.

TABLE 1 Status of Privileges in Zones

Privilege	Status	Notes
`cpc_cpu`	Optional	Access to certain `cpc`(3CPC) counters
`dtrace_proc`	Optional	`fasttrap` and `pid` providers; `plockstat`(1M)
`dtrace_user`	Optional	`profile` and `syscall` providers
`file_flag_set`	Optional	Allows a process to set `immutable`, `nounlink` or `appendonly` file attributes; can be used to mark files immutable in the global zone and the non-global zone cannot remove the files
`graphics_access`	Optional	`ioctl`(2) access to `agpgart_io`(7I)
`graphics_map`	Optional	`mmap`(2) access to `agpgart_io`(7I)
`net_rawaccess`	Optional in shared-IP zones Default in exclusive-IP zones	Raw `PF_INET/PF_INET6` packet access
`proc_clock_highres`	Optional	Use of high resolution timers
`proc_priocntl`	Optional	Scheduling control; `priocntl`(1)
`sys_ipc_config`	Optional	Increase IPC message queue buffer size
`dtrace_kernel`	Prohibited	Currently unsupported
`proc_zone`	Prohibited	Currently unsupported
`sys_config`	Prohibited	Currently unsupported
`sys_devices`	Prohibited	Currently unsupported
`sys_dl_config`	Prohibited	Currently unsupported
`sys_linkdir`	Prohibited	Currently unsupported

Privilege	Status	Notes
sys_net_config	Prohibited	Currently unsupported
sys_res_config	Prohibited	Currently unsupported
sys_smb	Prohibited	Currently unsupported
sys_suser_compat	Prohibited	Currently unsupported
file_read	Required, Default	Allows a process to read a file or directory whose permission or ACL allow the process read permission
file_write	Required, Default	Allows a process to write a file or directory whose permission or ACL allow the process write permission
net_access	Required, Default	Allows a process to open a TCP, UDP, SDP or SCTP network endpoint
proc_exec	Required, Default	Used to start init(1M)
proc_fork	Required, Default	Used to start init(1M)
sys_mount	Required, Default	Needed to mount required file systems
sys_flow_config	Required, Default in exclusive-IP zones Prohibited in shared-IP zones	Needed to configure flows
sys_ip_config	Required, Default in exclusive-IP zones Prohibited in shared-IP zones	Required to boot zone and initialize IP networking in exclusive-IP zone
sys_iptun_config	Required, Default in exclusive-IP zones Prohibited in shared-IP zones	Configure IP tunnel links
contract_event	Default	Used by contract file system
contract_identity	Default	Set service FMRI value of a process contract template
contract_observer	Default	Contract observation regardless of UID
file_chown	Default	File ownership changes
file_chown_self	Default	Owner/group changes for own files
file_dac_execute	Default	Execute access regardless of mode/ACL
file_dac_read	Default	Read access regardless of mode/ACL
file_dac_search	Default	Search access regardless of mode/ACL
file_dac_write	Default	Write access regardless of mode/ACL
file_link_any	Default	Link access regardless of owner
file_owner	Default	Other access regardless of owner
file_setid	Default	Permission changes for setid, setgid, setuid files

Privilege	Status	Notes
ipc_dac_read	Default	IPC read access regardless of mode
ipc_dac_write	Default	Allow a process to write a System V IPC message queue, semaphore set, or shared memory segment in which the permission bits would not otherwise allow the process write permission
ipc_dac_owner	Default	IPC write access regardless of mode
ipc_owner	Default	IPC other access regardless of mode
net_icmpaccess	Default	ICMP packet access: ping(1M)
net_observability	Default	Allow a process to open a device for receiving network traffic; sending traffic is disallowed
net_privaddr	Default	Binding to privileged ports
proc_audit	Default	Generation of audit records
proc_chroot	Default	Changing of root directory
proc_info	Default	Process examination
proc_lock_memory	Default	Locking memory; shmctl(2)and mlock(3C) If this privilege is assigned to a non-global zone by the system administrator, consider also setting the zone.max-locked-memory resource control to prevent the zone from locking all memory.
proc_owner	Default	Process control regardless of owner
proc_session	Default	Process control regardless of session
proc_setid	Default	Setting of user/group IDs at will
proc_taskid	Default	Assigning of task IDs to caller
sys_acct	Default	Management of accounting
sys_admin	Default	Simple system administration tasks
sys_audit	Default	Management of auditing
sys_nfs	Default	NFS client support
sys_ppp_config	Default in exclusive-IP zones Prohibited in shared-IP zones	Create and destroy PPP (sppp) interfaces, configure PPP tunnels (sppptun)
sys_resource	Default	Resource limit manipulation
sys_share	Default	Allows sharefs system call needed to share file systems. Privilege can be prohibited in the zone configuration to prevent NFS sharing within a zone.
sys_time	Default	System time manipulation

The following table lists all of the Oracle Solaris Trusted Extensions privileges and the status of each privilege with respect to zones. Optional privileges are not part of the default set of privileges but can be specified through the `limitpriv` property.

Note - Trusted Extensions privileges are interpreted only if the system is configured with Trusted Extensions.

TABLE 2 Status of Oracle Solaris Trusted Extensions Privileges in Zones

Trusted Extensions Privilege	Status	Notes
`file_downgrade_sl`	Optional	Set the sensitivity label of file or directory to a sensitivity label that does not dominate the existing sensitivity label
`file_upgrade_sl`	Optional	Set the sensitivity label of file or directory to a sensitivity label that dominates the existing sensitivity label
`sys_trans_label`	Optional	Translate labels not dominated by sensitivity label
`win_colormap`	Optional	Colormap restrictions override
`win_config`	Optional	Configure or destroy resources that are permanently retained by the X server
`win_dac_read`	Optional	Read from window resource not owned by client's user ID
`win_dac_write`	Optional	Write to or create window resource not owned by client's user ID
`win_devices`	Optional	Perform operations on input devices.
`win_dga`	Optional	Use direct graphics access X protocol extensions; frame buffer privileges needed
`win_downgrade_sl`	Optional	Change sensitivity label of window resource to new label dominated by existing label
`win_fontpath`	Optional	Add an additional font path
`win_mac_read`	Optional	Read from window resource with a label that dominates the client's label
`win_mac_write`	Optional	Write to window resource with a label not equal to the client's label
`win_selection`	Optional	Request data moves without confirmer intervention
`win_upgrade_sl`	Optional	Change sensitivity label of window resource to a new label not dominated by existing label
`net_bindmlp`	Default	Allows binding to a multilevel port (MLP)
`net_mac_aware`	Default	Allows reading down through NFS

To change which privileges are available in a non-global zone, see "How to Modify Zone Privileges" on page 34.

For more information about privileges, see the `ppriv`(1) man page and "Process Rights Management" in *Securing Users and Processes in Oracle Solaris 11.3*.

Using IP Security Architecture in Zones

The Internet Protocol Security Architecture (IPsec), which provides IP datagram protection, is described in "IPsec Reference" in *Securing the Network in Oracle Solaris 11.3*. The Internet Key Exchange (IKE) protocol is used to manage the required keying material for authentication and encryption automatically.

For more information, see the `ipsecconf(1M)` and `ipseckey(1M)` man pages.

IP Security Architecture in Shared-IP Zones

IPsec can be used in the global zone. However, IPsec in a non-global zone cannot use IKE. Therefore, you must manage the IPsec keys and policy for the non-global zones by using the Internet Key Exchange (IKE) protocol in the global zone. Use the source address that corresponds to the non-global zone that you are configuring.

IP Security Architecture in Exclusive-IP Zones

IPsec can be used in exclusive-IP zones.

Using Oracle Solaris Auditing in Zones

An audit record describes an event, such as logging in to a system or writing to a file. Oracle Solaris Auditing provides the following two auditing models on systems that are running zones:

- All zones are audited identically from the global zone. This model is used when all zones are administered by the global zone, for example, to achieve service isolation through zones.
- Each zone is audited independently of the global zone. This model is used when each zone is administered separately, for example, to achieve server consolidation by zone.

Auditing is described in Chapter 1, "About Auditing in Oracle Solaris" in *Managing Auditing in Oracle Solaris 11.3*. For zones considerations associated with auditing, see "Auditing on a System With Oracle Solaris Zones" in *Managing Auditing in Oracle Solaris 11.3* and "Configuring the Audit Service in Zones" in *Managing Auditing in Oracle Solaris 11.3*. For additional information, also see the `auditconfig(1M)`, `auditreduce(1M)`, `usermod(1M)`, and `user_attr(4)` man pages.

Note - It is also possible to use audit policies that are activated on a temporary basis, but not set in the repository.

For additional information, see the example that follows "How to Change Audit Policy" in *Managing Auditing in Oracle Solaris 11.3*.

Core Files in Zones

The `coreadm` command is used to specify the name and location of core files produced by abnormally terminating processes. Core file paths that include the *zonename* of the zone in which the process executed can be produced by specifying the `%z` variable. The path name is relative to a zone's root directory.

For more information, see the `coreadm`(1M) and `core`(4) man pages.

Running DTrace in a Non-Global Zone

DTrace programs that require only the `dtrace_proc` and `dtrace_user` privileges can be run in a non-global zone. To add these privileges to the set of privileges available in the non-global zone, use the `zonecfg limitpriv` property. For instructions, see "How to Use DTrace in a Non-Global Zone" on page 168.

The providers supported through `dtrace_proc` are `fasttrap` and `pid`. The providers supported through `dtrace_user` are `profile` and `syscall`. DTrace providers and actions are limited in scope to the zone.

Also see "Privileges in a Non-Global Zone" on page 144 for more information.

Setting Time Values in Non-Global Zones

You can set time values in non-global zones that are different from the value in the global zone. The ability to set the time in the zone is available through the default `sys_time` privilege. This privilege allows a non-global zone process to set either the virtual zone time or the system time, depending on the value of the `global-time` property for the zone, if set. The ability to set

different times in non-global zones is not independent of time changes made in the global zone. If the time is changed in the global zone, the non-global zone time is offset by the same amount.

Network Time Protocol (NTP) can be run from any zone, affecting only the zone in which the command is run. When running NTP across a system with non-global zones that have different times, run NTP in the global zone. Running NTP in the global zone will synchronize all the non- global zone clocks that just run at an offset. The effect of NTP changing the time through clock modulation in the global zone will transfer to a non global-zone as well.

When NTP is run inside the zone with `global-time` set to `false`, the `ntp_adjtime` and `adjtime` system calls cannot be used to make corrections to the zone time. When `global-time` is set to `false`, NTP keeps time in sync by adjusting the clock to a given value through setting the time, which maintains time synchronization at a second level of granularity. NTP can make zone time go forward or backward to maintain synchronization.

When NTP is run inside the zone when `global-time` is set to `true`, NTP is allowed to modulate the system clock through the `ntp_adjtime` and `adjtime` system calls. Then the ability to run NTP inside the zone to keep system time in tight synchronization is preserved.

For more information about the `global-time` property, see *Oracle Solaris Zones Configuration Resources*. For information about setting the time, see `date(1)`. For more information about privileges, see "Privileges in a Non-Global Zone" on page 144. For more information about `adjtime` and `ntp_adjtime`, see the `adjtime(2)` and `ntp_adjtime(2)` man pages.

About Backing Up an Oracle Solaris System With Zones Installed

You can perform backups in individual non-global zones, or back up the entire system from the global zone.

Backing Up Loopback File System Directories

Do not back up the loopback file systems (`lofs`) from within non-global zones.

If you back up and restore `read/write` loopback file systems from within a non-global zone, note that these file systems are also writable from the global zone, and from any other zones in which they are `read/write` mounted. Back up and restore these file systems from the global zone only, to avoid multiple copies.

Backing Up Your System From the Global Zone

You might choose to perform your backups from the global zone in the following cases:

- You want to back up the configurations of your non-global zones as well as the application data.
- Your primary concern is the ability to recover from a disaster. If you need to restore everything or almost everything on your system, including the root file systems of your zones and their configuration data as well as the data in your global zone, backups should take place in the global zone.
- You have commercial network backup software.

Note - Your network backup software should be configured to skip all inherited `lofs` file systems if possible. The backup should be performed when the zone and its applications have quiesced the data to be backed up.

Backing Up Individual Non-Global Zones on Your System

You might decide to perform backups within the non-global zones in the following cases.

- The non-global zone administrator needs the ability to recover from less serious failures or to restore application or user data specific to a zone.
- You use the backup software of a particular application or service running in a zone. It might be difficult to execute the backup software from the global zone because application environments, such as directory path and installed software, would be different between the global zone and the non-global zone.

  If the application can perform a snapshot on its own backup schedule in each non-global zone and store those backups in a writable directory exported from the global zone, the global zone administrator can pick up those individual backups as part of the backup strategy from the global zone.

Creating Oracle Solaris Unified Archive Backups

You can use the `zonecfg` and `zoneadm` commands to configure and to install new zones directly from a Unified Archive file. Unified Archive files contain both zone configuration and zone data. On the destination server, a zone can be configured and installed from the archive.

For more information, see *Using Unified Archives for System Recovery and Cloning in Oracle Solaris 11.3* for a full description of Unified Archives, including usage for system and zone cloning and recovery.

Determining What to Back Up in Non-Global Zones

You can back up everything in the non-global zone, or, because a zone's configuration changes less frequently, you can perform backups of the application data only.

Backing Up Application Data Only

If application data is kept in a particular part of the file system, you might decide to perform regular backups of this data only. The zone's root file system might not have to be backed up as often because it changes less frequently.

You will have to determine where the application places its files. Locations where files can be stored include the following:

- Users' home directories
- `/etc` for configuration data files
- `/var`

Assuming the application administrator knows where the data is stored, it might be possible to create a system in which a per-zone writable directory is made available to each zone. Each zone can then store its own backups, and the global administrator or user granted the appropriate authorizations can make this location one of the places on the system to back up.

General Database Backup Operations

If the database application data is not under its own directory, the following rules apply:

- Ensure that the databases are in a consistent state first.

  Databases must be quiesced because they have internal buffers to flush to disk. Make sure that the databases in non-global zones have come down before starting the backup from the global zone.
- Within each zone, use file system capabilities to make a snapshot of the data, then back up the snapshots directly from the global zone.

This process will minimize elapsed time for the backup window and remove the need for backup clients/modules in all of the zones.

Tape Backups

Each non-global zone can take a snapshot of its private file systems when it is convenient for that zone and the application has been briefly quiesced. Later, the global zone can back up each of the snapshots and put them on tape after the application is back in service.

This method has the following advantages:

- Fewer tape devices are needed.
- There is no need for coordination between the non-global zones.
- There is no need to assign devices directly to zones, which improves security.
- Generally, this method keeps system management in the global zone, which is preferred.

About Restoring Non-Global Zones

In the case of a restore where the backups were done from the global zone, the global administrator or a user granted the appropriate authorizations can reinstall the affected zones and then restore that zone's files. Note that this assumes the following:

- The zone being restored has the same configuration as it did when the backup was done.
- The global zone has not been updated between the time when the backup was done and the time when the zone is restored.

Otherwise, the restore could overwrite some files that should be merged by hand.

Note - If all file systems in the global zone are lost, restoring everything in the global zone restores the non-global zones as well, as long as the respective root file systems of the non-global zones were included in the backup.

Commands Used on a System With Zones Installed

The commands identified in Table 3, “Commands Used to Administer and Monitor Zones,” on page 154 provide the primary administrative interface to the zones facility. To run

zone administration commands in a non-root role, see “Assigning Limited Rights to Zone Administrators” on page 159.

TABLE 3 Commands Used to Administer and Monitor Zones

Command Reference	Description
zlogin(1)	Used to log in to a non-global zone
zonename(1)	Prints the name of the current zone
zonestat(1)	Used to observe zone resource usage.
zoneadm(1M)	Used to administer zones on a system
zonecfg(1M)	Used to set up a zone configuration
getzoneid(3C)	Used to map between zone ID and name
zones(5)	Provides description of zones facility
zcons(7D)	Zone console device driver

The `zoneadmd` daemon is the primary process for managing the zone's virtual platform. The man page for the `zoneadmd` daemon is `zoneadmd`(1M). The daemon does not constitute a programming interface.

The commands in the following table are used with the resource capping daemon.

TABLE 4 Commands Used With `rcapd`

Command Reference	Description
rcapstat(1)	Monitors the resource utilization of capped projects.
rcapadm(1M)	Configures the resource capping daemon, displays the current status of the resource capping daemon if it has been configured, and enables or disables resource capping.
rcapd(1M)	The resource capping daemon.

The commands identified in the following table have been modified for use on an Oracle Solaris system with zones installed. These commands have options that are specific to zones or present information differently. The commands are listed by man page section.

TABLE 5 Commands Modified for Use on an Oracle Solaris System With Zones Installed

Command Reference	Description
ipcrm(1)	Added -z *zone* option. This option is only useful when the command is executed in the global zone.
ipcs(1)	Added -z *zone* option. This option is only useful when the command is executed in the global zone.
pgrep(1)	Added -z *zoneidlist* option. This option is only useful when the command is executed in the global zone.

Command Reference	Description
ppriv(1)	Added the expression `zone` for use with the `-l` option to list all privileges available in the current zone. Also use the option `-v` after `zone` to obtain verbose output.
priocntl(1)	Zone ID can be used in *idlist* and `-i` *idtype* to specify processes. You can use the `priocntl -i` *zoneid* command to move running processes into a different scheduling class in a non-global zone.
proc(1)	Added `-z` *zone* option to `ptree` only. This option is only useful when the command is executed in the global zone.
ps(1)	Added *zonename* and *zoneid* to list of recognized `format` names used with the `-o` option. Added `-z` *zonelist* to list only processes in the specified zones. Zones can be specified either by zone name or by zone ID. This option is only useful when the command is executed in the global zone. Added `-Z` to print the name of the zone associated with the process. The name is printed under an additional column header, `ZONE`.
renice(1)	Added *zoneid* to list of valid arguments used with the `-i` option.
sar(1)	If executed in a non-global zone in which the pools facility is enabled, the `-b`, `-c` `-g`, `-m`, `-p`, `-u`, `-w`, and `-y` options display values only for processors that are in the processor set of the pool to which the zone is bound.
auditconfig(1M)	Added `zonename` token.
auditreduce(1M)	Added `-z` *zone-name* option. Added ability to get an audit log of a zone.
coreadm(1M)	Added variable `%z` to identify the zone in which process executed.
df(1M)	Added `-Z` option to display mounts in all visible zones. This option has no effect in a non-global zone.
dladm(1M)	Added `-Z` option to `show` subcommands, which adds a zone column to the default command output. The zone column indicates the zone to which the resource is currently assigned.
dlstat(1M)	Added `-Z` option to `show` subcommands, which adds a zone column to the default command output. The zone column indicates the zone to which the resource is currently assigned.
fsstat(1M)	Added `-z` option to report on file system activity per zone. Multiple `-z` options can be used to monitor activity in selected zones. The option has no effect if only used to monitor `mountpoints` and not `fstypes`. Added `-A` option to report aggregate file system activity for the specified `fstypes` across all zones. This is the default behavior if neither `-z` or the `-Z` option is used. The `-A` option has no effect if only used to monitor `mountpoints` and not `fstypes`. When used with either the `-z` or the `-Z` option, the `-A` option displays the aggregate for the specified `fstypes` across all zones on a separate line. Added `-Z` option to report file system activity in all zones on the system. This option has no effect if used with `-z` option. The option has no effect if only used to monitor `mountpoints` and not `fstypes`.

Command Reference	Description
iostat(1M)	If executed in a non-global zone in which the pools facility is enabled, information is provided only for those processors that are in the processor set of the pool to which the zone is bound.
ipadm(1M)	Configure Internet Protocol network interfaces and TCP/IP tunables. The `from-gz` type is only displayed in non-global zones, and indicates that the address was configured based on the `allowed-address` property configured for the non-global exclusive-IP zone from the global zone. The `zone` address property specifies the zone in which all the addresses referenced by `allowed-address` should be placed. The zone must be configured as a shared-IP zone.
kstat(1M)	If executed in the global zone, `kstat`s are displayed for all zones. If executed in a non-global zone, only `kstat`s with a matching *zoneid* are displayed.
mpstat(1M)	If executed in a non-global zone in which the pools facility is enabled, command only displays lines for the processors that are in the processor set of the pool to which the zone is bound.
ndd(1M)	When used in the global zone, displays information for all zones. `ndd` on the TCP/IP modules in an exclusive-IP zone only displays information for that zone.
netstat(1M)	Displays information for the current zone only.
nfsstat(1M)	Displays statistics for the current zone only.
poolbind(1M)	Added *zoneid* list. Also see "Resource Pools Used in Zones" in *Administering Resource Management in Oracle Solaris 11.3* for information about using zones with resource pools.
prstat(1M)	Added `-z` *zoneidlist* option. Also added `-Z` option. If executed in a non-global zone in which the pools facility is enabled, the percentage of recent CPU time used by the process is displayed only for the processors in the processor set of the pool to which the zone is bound. Output of the `-a`, `-t`, `-T`, `-J`, and `-Z` options displays a SWAP instead of a SIZE column. The swap reported is the total swap consumed by the zone's processes and `tmpfs` mounts. This value assists in monitoring the swap reserved by each zone, which can be used to choose a reasonable `zone.max-swap` setting.
psrinfo(1M)	If executed in a non-global zone, only information about the processors visible to the zone is displayed.
traceroute(1M)	Usage change. When specified from within a non-global zone, the `-F` option has no effect because the "don't fragment" bit is always set.
vmstat(1M)	When executed in a non-global zone in which the pools facility is enabled, statistics are reported only for the processors in the processor set of the pool to which the zone is bound. Applies to output from the `-p` option and the `page`, `faults`, and `cpu` report fields.
priocntl(2)	Added `P_ZONEID` *id* argument.
processor_info(2)	If the caller is in a non-global zone and the pools facility is enabled, but the processor is not in the processor set of the pool to which the zone is bound, an error is returned.
p_online(2)	If the caller is in a non-global zone and the pools facility is enabled, but the processor is not in the processor set of the pool to which the zone is bound, an error is returned.

Command Reference	Description
pset_bind(2)	Added P_ZONEID as *idtype*. Added zone to possible choices for P_MYID specification. Added P_ZONEID to valid *idtype* list in EINVAL error description.
pset_info(2)	If the caller is in a non-global zone and the pools facility is enabled, but the processor is not in the processor set of the pool to which the zone is bound, an error is returned.
pset_list(2)	If the caller is in a non-global zone and the pools facility is enabled, but the processor is not in the processor set of the pool to which the zone is bound, an error is returned.
pset_setattr(2)	If the caller is in a non-global zone and the pools facility is enabled, but the processor is not in the processor set of the pool to which the zone is bound, an error is returned.
sysinfo(2)	Changed PRIV_SYS_CONFIG to PRIV_SYS_ADMIN.
umount(2)	ENOENT is returned if file pointed to by *file* is not an absolute path.
getloadavg(3C)	If the caller is in a non-global zone and the pools facility is enabled, the behavior is equivalent to calling with a psetid of PS_MYID.
getpriority(3C)	Added zone IDs to target processes that can be specified. Added zone ID to EINVAL error description.
priv_str_to_set(3C)	Added “zone” string for the set of all privileges available within the caller's zone.
pset_getloadavg(3C)	If the caller is in a non-global zone and the pools facility is enabled, but the processor is not in the processor set of the pool to which the zone is bound, an error is returned.
sysconf(3C)	If the caller is in a non-global zone and the pools facility enabled, sysconf (_SC_NPROCESSORS_CONF) and sysconf(_SC_NPROCESSORS_ONLN) return the number of total and online processors in the processor set of the pool to which the zone is bound.
ucred_get(3C)	Added ucred_getzoneid() function, which returns the zone ID of the process or -1 if the zone ID is not available.
core(4)	Added n_type: NT_ZONENAME. This entry contains a string that describes the name of the zone in which the process was running.
pkginfo(4)	Now provides optional parameters and an environment variable in support of zones.
proc(4)	Added capability to obtain information on processes running in zones.
audit_syslog(5)	Added in *zone name* field that is used if the zonename audit policy is set.
privileges(5)	Added PRIV_PROC_ZONE, which allows a process to trace or send signals to processes in other zones. See zones(5).
if_tcp(7P)	Added zone ioctl() calls.
cmn_err(9F)	Added zone parameter.
ddi_cred(9F)	Added crgetzoneid(), which returns the zone ID from the user credential pointed to by cr.

♦ ♦ ♦ CHAPTER 10

Administering Oracle Solaris Zones

This chapter covers general administration tasks and provides usage examples.

See Chapter 9, "About Oracle Solaris Zones Administration" for general zone administration topics.

Assigning Limited Rights to Zone Administrators

The `root` role has all administrative rights. By default, the global zone administrator (`root`) can administer all non-global zones.

For security, the `root` role should delegate rights that enable trusted users to administer zones with just the rights that are required. Rights in Oracle Solaris are privileged commands and authorizations, collected into rights profiles for specific areas of responsibility. For zones, Oracle Solaris provides several rights profiles.

The root role has another way to limit administrative rights in a zone. Each non-global zone has an admin resource. By specifying user names and required authorizations in the admin resource, root can limit access to the zone. When site security requires separation of duty, this method can prevent root from accessing the non-global zone.

The root role can assign a rights profile directly to a user or create an administrative role and assign the rights profile to that role. When users are assigned a role, they must assume it to administer zones.

For more information about Oracle Solaris rights and the admin resource for zones, review the following:

- Chapter 1, "About Using Rights to Control Users and Processes" in *Securing Users and Processes in Oracle Solaris 11.3*
- "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*
- "admin Resource for Zones" in *Oracle Solaris Zones Configuration Resources*

Assigning Rights to Non-Root Users to Manage Zones

For various reasons, the global zone administrator might want to distribute the rights to manage zones to selected users. Possible reasons include:

- To enable root to concentrate on other administration operations
- To assign a specialist to zone administration
- To reduce the number of entry points for an attack
- To enable separation of duty where root has limited powers
- To implement site security requirements, such as PCI-DSS or HIPAA compliance

The rights that you in the root role can assign to delegate zone administration include the following:

Zone Security rights profile

For administrators who will create and configure zones.

The Zone Security rights profile includes the zonecfg or txzonemgr commands and every solaris.zone.* authorization. The assignee can delegate zone administration. For information about txzonemgr, see "Creating Labeled Zones" in *Trusted Extensions Configuration and Administration*.

If the `auths` property of the `admin` resource is configured in the managed zone, this rights profile is not sufficient to create, log in, and configure zones. The zone administrator must be named in the `user` property of the `admin` resource and be assigned the `solaris.zone.*` authorizations.

Note - This rights profile permits the user to create or modify or delete any zone configuration on the host.

Zone Configuration rights profile

For administrators who will create and modify zones.

The Zone Configuration rights profile enables a zone administrator to configure a zone. For a migrated zone, the administrator must be granted this rights profile on the target system to complete the migration if a configuration for the zone does not already exist on the target system. The Zone Configuration rights profile includes the `zonecfg` command only.

If the `auths` property of the `admin` resource is configured in the managed zone, this rights profile is not sufficient to configure zones. The zone administrator must be named in the `user` property of the `admin` resource and be assigned the `solaris.zone.config` authorization. If login is restricted, the zone administrator must also be assigned the `solaris.zone.login` authorization.

Note - This rights profile permits the user to create or modify or delete any zone configuration on the host.

Zone Management rights profile

For administrators who will manage existing zones.

The Zone Management rights profile includes the `zlogin` and `zoneadm` commands.

If the `auths` property of the `admin` resource is configured in the managed zone, this rights profile is not sufficient to manage zones. The zone administrator must be named in the `user` property of the `admin` resource and be assigned the `solaris.zone.*` authorizations to log in and manage the zone.

Zone Migration rights profile

For administrators who will migrate any type of zone.

The Zone Migration rights profile enables a zone administrator to perform migration of an `installed` or `running` zone. A zone administrator who is assigned this profile can perform live or warm migrations. The Zone Migration rights profile includes the `zoneadm` and `zonecfg` commands.

If the `auths` property of the `admin` resource is configured in the managed zone, this rights profile is not sufficient to migrate zones. The zone administrator must be named as a `user` in the `admin` resource and be assigned the `solaris.zone.migrate` authorization. If

login is restricted, the zone administrator must also be assigned the `solaris.zone.login` authorization.

Common Commands When Limiting Rights to Administer Zones

Use the following commands to determine the contents of rights profiles and to assign them.

- To list the contents of a rights profile, run the following command in a terminal window:

  ```
  $ profiles -p "Rights Profile Name" info
  ```

- To see the rights or privileges that are granted to the commands in a Zone rights profile, run the following command:

  ```
  $ getent exec_attr | grep Zone
  ```

  This command finds the privileged commands in the rights profiles that contain the word "Zone".

- To assign a rights profile to a user, run a `usermod` command similar to the following:

  ```
  # usermod -K profiles+="Zone Management" username
  ```

EXAMPLE 9 Using the `admin` Resource to Limit Zone Access

To assign administrative rights to an account for a particular zone, the role name or user name must exist in the global zone.

1. From a terminal in the global zone, specify who has administrative rights in the named zone:

   ```
   # zonecfg -z zonename
   zonecfg:zonename> add admin
   zonecfg:zonename:admin> set user=account-name
   zonecfg:zonename:admin> set auths=login,manage
   zonecfg:zonename:admin> end
   zonecfg:zonename> commit
   ```

2. Verify that *account-name* has the required authorizations in the non-global zone.

   ```
   global1# zonecfg -z zone info admin
   admin:
           user: account-name
           auths: login,manage
   ```

The *account-name* account can now log in and manage the named zone.

For more examples, including how to assign authenticated rights profiles and how to create and assign roles, see "Assigning Rights to Users" in *Securing Users and Processes in Oracle Solaris 11.3*.

Using the zonestat Utility in a Non-Global Zone

The zonestat utility reports on the CPU, memory, network, and resource control utilization of the currently running zones. Usage examples follow.

For complete information, see zonestat(1).

The zonestat network component shows the usage of virtual network (VNIC) resources on PHYS, AGGR, Etherstub, and SIMNET datalinks by zones. Information on other datalinks, such as bridges and tunnels, can be obtained by using the networking utilities described in the dladm(1M) and dlstat(1M) man pages.

All zonestat options and resource types can also be invoked within a non-global zone to display statistics for that zone.

```
root@zoneA:~# zonestat -z global -r physical-memory 2
```

Note - When zonestat is used in a non-global zone, the combined resource usage of all other zones, including the global zone, is reported as used by the global zone. Non-global zone users of zonestat are not aware of the other zones sharing the system.

▼ How to Use the zonestat Utility to Display a Summary of CPU and Memory Utilization

1. **Assume the root role.**
   For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Display a summary of CPU and memory utilization every 5 seconds.**

```
# zonestat -z global -r physical-memory 5
Collecting data for first interval...
Interval: 1, Duration: 0:00:05
PHYSICAL-MEMORY             SYSTEM MEMORY
```

```
mem_default                          2046M
                               ZONE  USED %USED   CAP  %CAP
                            [total] 1020M 49.8%     -     -
                           [system]  782M 38.2%     -     -
                             global  185M 9.06%     -     -

Interval: 2, Duration: 0:00:10
PHYSICAL-MEMORY              SYSTEM MEMORY
mem_default                          2046M
                               ZONE  USED %USED   CAP  %CAP
                            [total] 1020M 49.8%     -     -
                           [system]  782M 38.2%     -     -
                             global  185M 9.06%     -     -
...
```

How to Use the `zonestat` Utility to Report on the Default `pset`

1. **Assume the `root` role.**

   For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Report on the default `pset` once a second for 1 minute:**

```
# zonestat -r default-pset 1 1m
Collecting data for first interval...
Interval: 1, Duration: 0:00:01
PROCESSOR_SET                   TYPE  ONLINE/CPUS     MIN/MAX
pset_default           default-pset          2/2         1/-
                               ZONE  USED   PCT   CAP  %CAP   SHRS  %SHR %SHRU
                            [total]  0.02 1.10%     -     -      -     -     -
                           [system]  0.00 0.19%     -     -      -     -     -
                             global  0.01 0.77%     -     -      -     -     -
                              zone1  0.00 0.07%     -     -      -     -     -
                              zone2  0.00 0.06%     -     -      -     -     -

...
Interval: 60, Duration: 0:01:00
PROCESSOR_SET                   TYPE  ONLINE/CPUS     MIN/MAX
pset_default           default-pset          2/2         1/-
                               ZONE  USED   PCT   CAP  %CAP   SHRS  %SHR %SHRU
                            [total]  0.06 3.26%     -     -      -     -     -
                           [system]  0.00 0.18%     -     -      -     -     -
                             global  0.05 2.94%     -     -      -     -     -
```

```
zone1  0.00 0.06%     -     -      -     -     -
zone2  0.00 0.06%     -     -      -     -     -
```

▼ Using zonestat to Report Total and High Utilization

1. **Assume the root role.**

   For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Monitor silently at a 10-second interval for 3 minutes, then produce a report on the total and high utilizations.**

```
# zonestat -q -R total,high 10s 3m 3m
Report: Total Usage
    Start: Fri Aug 26 07:32:22 PDT 2011
      End: Fri Aug 26 07:35:22 PDT 2011
    Intervals: 18, Duration: 0:03:00
SUMMARY                   Cpus/Online: 2/2   PhysMem: 2046M  VirtMem: 3069M
                     ---CPU----  --PhysMem-- --VirtMem-- --PhysNet--
                ZONE  USED %PART  USED %USED  USED %USED PBYTE %PUSE
             [total]  0.01 0.62% 1020M 49.8% 1305M 42.5%    14 0.00%
            [system]  0.00 0.23%  782M 38.2% 1061M 34.5%     -     -
              global  0.00 0.38%  185M 9.06%  208M 6.77%     0 0.00%
               test2  0.00 0.00% 52.4M 2.56% 36.6M 1.19%     0 0.00%

Report: High Usage
    Start: Fri Aug 26 07:32:22 PDT 2011
      End: Fri Aug 26 07:35:22 PDT 2011
    Intervals: 18, Duration: 0:03:00
SUMMARY                   Cpus/Online: 2/2   PhysMem: 2046M  VirtMem: 3069M
                     ---CPU----  --PhysMem-- --VirtMem-- --PhysNet--
                ZONE  USED %PART  USED %USED  USED %USED PBYTE %PUSE
             [total]  0.01 0.82% 1020M 49.8% 1305M 42.5%  2063 0.00%
            [system]  0.00 0.26%  782M 38.2% 1061M 34.5%     -     -
              global  0.01 0.55%  185M 9.06%  207M 6.77%     0 0.00%
               test2  0.00 0.00% 52.4M 2.56% 36.6M 1.19%     0 0.00%
```

How to Obtain Network Bandwidth Utilization for Exclusive-IP Zones

The `zonestat` command used with the `-r` option and `network` resource type shows the per-zone utilization of each network device.

Use this procedure to view how much datalink bandwidth in the form of VNICs is used by each zone. For example, `zoneB` displayed under net0 indicates that this zone consumes resources of net0 in the form of VNICs. The specific VNICs can be displayed by also adding the `-x` option.

1. **Assume the `root` role.**

   For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Use the `network` resource type to the `zonestat` command with the `-r` option to display the utilization one time.**

```
# zonestat -r network 1 1
Collecting data for first interval...
Interval: 1, Duration: 0:00:01

 NETWORK-DEVICE                  SPEED        STATE        TYPE
 aggr1                       2000mbps           up        AGGR
                    ZONE TOBYTE  MAXBW %MAXBW PRBYTE %PRBYTE POBYTE %POBYTE
                  global  1196K      -      -   710K   0.28%   438K   0.18%

 net0                        1000mbps           up        PHYS
                    ZONE TOBYTE  MAXBW %MAXBW PRBYTE %PRBYTE POBYTE %POBYTE
                 [total]  7672K      -      -  6112K   4.89%  1756K   1.40%
                  global  5344K  100m*  42.6%  2414K   1.93%  1616K   1.40%
                   zoneB   992K   100m  15.8%  1336K   0.76%   140K   0.13%
                   zoneA  1336K    50m  10.6%   950K   1.07%      0   0.00%

 net1                        1000mbps           up        PHYS
                    ZONE TOBYTE  MAXBW %MAXBW PRBYTE %PRBYTE POBYTE %POBYTE
                  global   126M      -      -    63M   6.30%    63M   6.30%

 etherstub1                    n/a          n/a   ETHERSTUB
                    ZONE TOBYTE  MAXBW %MAXBW PRBYTE %PRBYTE POBYTE %POBYTE
                 [total]  3920K      -      -      0       -      0       -
                  global  1960K  100M*  1.96%      0       -      0       -
                   zoneA  1960K    50M  3.92%      0       -      0       -
```

Reporting Per-Zone fstype Statistics for all Zones

Use the -z option to report on file system activity per zone. Multiple -z options can be used to monitor activity in selected zones.

Use the -A option to report aggregate file system activity for the specified fstypes across all zones. This is the default behavior if neither -z or the -Z option is used.

When used with either the -z or the -Z option, the -A option displays the aggregate for the specified fstypes across all zones on a separate line.

Use the -Z option to report file system activity in all zones on the system. This option has no effect if used with the -z option. The option has no effect if only used to monitor mountpoints and not fstypes.

▼ How to Use the -z Option to Monitor Activity in Specific Zones

1. **Assume the root role.**

   For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Use multiple -z options to monitor activity in zones s10 and s10u9.**

```
# fsstat -z s10 -z s10u9 zfs tmpfs
 new  name   name  attr  attr lookup rddir  read read  write write
 file remov  chng   get   set    ops   ops   ops bytes   ops bytes
   93    82     6  163K   110   507K   148 69.7K 67.9M 4.62K 13.7M zfs:s10
  248   237   158  188K   101   612K   283 70.6K 68.6M 4.71K 15.2M zfs:s10u9
12.0K 1.90K 10.1K 35.4K    12  60.3K     4 25.7K 29.8M 36.6K 31.0M tmpfs:s10
12.0K 1.90K 10.1K 35.6K    14  60.2K     2 28.4K 32.1M 36.5K 30.9M tmpfs:S10u9
```

▼ How to Display Per-Zone fstype Statistics for All Zones

1. **Assume the root role.**

For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Obtain per-zone statistics for file system types `tmpfs` and `zfs`for each zone running on the system, and also display a system-wide aggregate for file system types `tmpfs` and `zfs`:**

```
# fsstat -A -Z zfs tmpfs
 new  name   name  attr  attr lookup rddir  read read  write write
 file remov  chng   get   set    ops   ops   ops bytes   ops bytes
 360K 1.79K 20.2K 4.20M 1.02M  25.0M  145K 5.42M 2.00G 1.07M 8.10G zfs
 359K 1.48K 20.1K 4.04M 1.02M  24.5M  144K 5.31M 1.88G 1.06M 8.08G zfs:global
   93    82     6 74.8K   107   250K   144 54.8K 60.5M 4.61K 13.7M zfs:s10
  248   237   158 90.2K   101   336K   283 53.0K 58.3M 4.71K 15.2M zfs:s10u9
60.0K 41.9K 17.7K  410K   515   216K   426 1022K 1.02G  343K  330M tmpfs
49.4K 38.1K 11.0K  366K   489   172K   420  968K  979M  283K  273M tmpfs:global
5.28K 1.90K 3.36K 21.9K    12  21.7K     4 25.7K 29.8M 29.9K 28.3M tmpfs:s10
5.25K 1.90K 3.34K 22.1K    14  21.6K     2 28.4K 32.1M 29.8K 28.2M tmpfs:s10u9
```

In the output, the non-global zones on the system are `S10` and `S10u9`.

Using DTrace in a Non-Global Zone

Perform the following steps to use DTrace functionality as described in "Running DTrace in a Non-Global Zone" on page 149.

▼ How to Use DTrace in a Non-Global Zone

1. **Become a zone administrator.**

   You must also be assigned the Network Management rights profile. The `root` role has all of these rights.

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Use the `zonecfg limitpriv` property to add the `dtrace_proc` and `dtrace_user` privileges.**

```
global$ zonecfg -z my-zone
zonecfg:my-zone> set limitpriv="default,dtrace_proc,dtrace_user"
zonecfg:my-zone> exit
```

Note - Depending on your requirements, you can add either privilege, or both privileges.

3. **Boot the zone.**

```
global$ zoneadm -z my-zone boot
```

4. **Log in to the zone.**

```
global$ zlogin my-zone
```

5. **Run the DTrace program.**

```
my-zone$ dtrace -l
```

Checking the Status of SMF Services in a Non-Global Zone

To check the status of SMF services in a non-global zone, use the `zlogin` command.

▼ How to Check the Status of SMF Services From the Command Line

1. **Become a zone administrator.**
   You must be assigned the Zone Management rights profile. For more information, see "Assigning Rights to Non-Root Users to Manage Zones" on page 160.

2. **From the command line, type the following to show all services, including disabled ones.**

```
global$ zlogin my-zone svcs -a
```

See Also For more information, see Chapter 5, "Logging In to Non-Global Zones" and `svcs(1)`.

▼ How to Check the Status of SMF Services From Within a Zone

1. **Become a zone administrator.**

You must be assigned the Zone Management rights profile. For more information, see "Assigning Rights to Non-Root Users to Manage Zones" on page 160.

2. **Log in to the zone.**

```
global$ zlogin my-zone
```

3. **Run the `svcs` command with the `-a` option to show all services, including disabled ones.**

```
my-zone$ svcs -a
```

See Also For more information, see Chapter 5, "Logging In to Non-Global Zones" and `svcs(1)`.

Mounting File Systems in Running Non-Global Zones

You can mount file systems in a running non-global zone. The following procedures are covered.

- As the global administrator or a zone administrator with the appropriate rights, you can import raw and block devices into a non-global zone. After the devices are imported, the zone administrator has access to the disk. The zone administrator can then create a new file system on the disk and perform one of the following actions:
  - Mount the file system manually
  - Place the file system in `/etc/vfstab` so that it will be mounted on zone boot
- As the global administrator or a user granted the appropriate authorizations, you can also mount a file system from the global zone into the non-global zone.

  Before mounting a file system from the global zone into a non-global zone, note that the non-global zone should be in the ready state or be booted. Otherwise, the next attempt to ready or boot the zone will fail. In addition, any file systems mounted from the global zone into a non-global zone will be unmounted when the zone halts.

▼ How to Use LOFS to Mount a File System

You can share a file system between the global zone and non-global zones by using LOFS mounts. This procedure uses the `zonecfg` command to add an LOFS mount of the global zone `/export/datafiles` file system to the `my-zone` configuration. This example does not customize the mount options.

1. **Become a zone administrator.**
   You must also be assigned the ZFS File System Management and ZFS Storage Management rights profiles. The `root` role has all of these rights.
   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Use the `zonecfg` command.**

   ```
   global$ zonecfg -z my-zone
   ```

3. **Add a file system to the configuration.**

   ```
   zonecfg:my-zone> add fs
   ```

4. **Set the mount point for the file system, `/datafiles` in `my-zone`.**

   ```
   zonecfg:my-zone:fs> set dir=/datafiles
   ```

5. **Specify that `/export/datafiles` in the global zone is to be mounted as `/datafiles` in `my-zone`.**

   ```
   zonecfg:my-zone:fs> set special=/export/datafiles
   ```

6. **Set the file system type.**

   ```
   zonecfg:my-zone:fs> set type=lofs
   ```

7. **End the specification.**

   ```
   zonecfg:my-zone:fs> end
   ```

8. **Verify and commit the configuration.**

   ```
   zonecfg:my-zone> verify
   zonecfg:my-zone> commit
   ```

▼ How to Delegate a ZFS Dataset to a Non-Global Zone

Use this procedure to delegate a ZFS dataset to a non-global zone.

1. **Assume the `root` role.**
   For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **From the global zone, create a new ZFS file system named `fs2` on an existing ZFS pool named `poolA`:**

```
global# zfs create poolA/fs2
```

3. **(Optional) Set the `mountpoint` property for the `poolA/fs2` file system to `/fs-del/fs2`.**

```
global# zfs set mountpoint=/fs-del/fs2 poolA/fs2
```

Setting the `mountpoint` is not required. If the `mountpoint` property is not specified, the dataset is mounted at `/alias` within the zone by default. Non-default values for the `mountpoint` and the `canmount` properties alter this behavior, as described in the `zfs`(1M) man page.

4. **Verify that the source of the `mountpoint` property for this file system is now `local`.**

```
global# zfs get mountpoint poolA/fs2
NAME       PROPERTY    VALUE        SOURCE
poolA/fs2  mountpoint  /fs-del/fs2  local
```

5. **Delegate the `poolA/fs2` file system or specify an aliased dataset:**

- **Delegate the `poolA/fs2` file system to the zone:**

```
# zonecfg -z my-zone
zonecfg:my-zone> add dataset
zonecfg:my-zone:dataset> set name=poolA/fs2
zonecfg:my-zone:dataset> end
```

- **Specify an aliased dataset:**

```
# zonecfg -z my-zone
zonecfg:my-zone> add dataset
zonecfg:my-zone:dataset> set name=poolA/fs2
zonecfg:my-zone:dataset> set alias=delegated
zonecfg:my-zone:dataset> end
```

6. **Reboot the zone and display the `zoned` property for all `poolA` file systems:**

```
global# zfs get -r zoned poolA
NAME      PROPERTY  VALUE   SOURCE
poolA     zoned     off     default
poolA/fs2 zoned     on      default
```

Note that the `zoned` property for `poolA/fs2` is set to `on`. This ZFS file system was delegated to a non-global zone, mounted in the zone, and is under zone administrator control. ZFS uses the `zoned` property to indicate that a dataset has been delegated to a non-global zone at one point in time.

Adding Non-Global Zone Access to Specific File Systems in the Global Zone

▼ How to Add Access to CD or DVD Media in a Non-Global Zone

This procedure enables you to add read-only access to CD or DVD media in a non-global zone. The loopback file system `lofs` is used to make the CD or DVD content available to the non-global zone. The media is mounted into the non-global zone. A CD or DVD can then be used to install a product in the non-global zone. This procedure uses a DVD named `jes_05q4_dvd`.

1. **Assume the `root` role.**

   For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Insert the media.**

3. **Check for media in the drive.**

   ```
   global# volcheck
   ```

4. **Test whether the DVD is auto mounted.**

   ```
   global# ls /cdrom
   ```

   You will see a display similar to the following:

   ```
   cdrom   cdrom1   jes_05q4_dvd
   ```

5. **Loopback mount the file system with the options `ro,nodevices` (read-only and no devices) in the non-global zone.**

   ```
   global# zonecfg -z my-zone
   zonecfg:my-zone> add fs
   zonecfg:my-zone:fs> set dir=/cdrom
   zonecfg:my-zone:fs> set special=/cdrom
   zonecfg:my-zone:fs> set type=lofs
   zonecfg:my-zone:fs> add options [ro,nodevices]
   zonecfg:my-zone:fs> end
   ```

```
zonecfg:my-zone> commit
zonecfg:my-zone> exit
```

6. **Reboot the non-global zone.**

```
global# zoneadm -z my-zone reboot
```

7. **Use the `zoneadm list` command with the `-v` option to verify the status.**

```
global# zoneadm list -v
```

You will see a display that is similar to the following:

```
ID  NAME     STATUS       PATH                           BRAND      IP
 0  global   running      /                              solaris   shared
 1  my-zone  running      /zones/my-zone                 solaris   excl
```

8. **Log in to the non-global zone.**

```
global# zlogin my-zone
```

9. **Verify the DVD-ROM mount.**

```
my-zone# ls /cdrom
```

You will see a display similar to this:

```
cdrom   cdrom1   jes_05q4_dvd
```

10. **Install the product as described in the product installation guide.**

11. **Exit the non-global zone.**

```
my-zone# exit
```

Tip - You might want to retain the `/cdrom` file system in your non-global zone. The mount will always reflect the current contents of the CD-ROM drive, or an empty directory if the drive is empty.

---

12. **(Optional) If you want to remove the `/cdrom` file system from the non-global zone, perform the following steps.**

```
global# zonecfg -z my-zone
zonecfg:my-zone> remove fs dir=/cdrom
zonecfg:my-zone> commit
zonecfg:my-zone> exit
```

Using IP Network Multipathing on an Oracle Solaris System With Zones Installed

▼ How to Use IP Network Multipathing in Exclusive-IP Non-Global Zones

IP Network Multipathing (IPMP) in an exclusive-IP zone is configured as it is in the global zone. To use IPMP, an exclusive-IP zone requires at least two `zonecfg add net` resources. IPMP is configured from within the zone on these datalinks.

You can configure one or more physical interfaces into an IP multipathing group, or IPMP group. After configuring IPMP, the system automatically monitors the interfaces in the IPMP group for failure. If an interface in the group fails or is removed for maintenance, IPMP automatically migrates, or fails over, the failed interface's IP addresses. The recipient of these addresses is a functioning interface in the failed interface's IPMP group. The failover component of IPMP preserves connectivity and prevents disruption of any existing connections. Additionally, IPMP improves overall network performance by automatically spreading out network traffic across the set of interfaces in the IPMP group. This process is called load spreading.

1. **Become a zone administrator.**

   For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Configure IPMP groups as described in “Configuring IPMP Groups” in *Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.3*.**

▼ How to Extend IP Network Multipathing Functionality to Shared-IP Non-Global Zones

Use this procedure to configure IPMP in the global zone and extend the IPMP functionality to non-global zones.

Each address, or logical interface, should be associated with a non-global zone when you configure the zone. See “About Using the zonecfg Command” in *Oracle Solaris Zones Configuration Resources* and “How to Configure the Zone” on page 22 for instructions.

This procedure accomplishes the following:

- The cards `net0` and `net1` are configured together in an IPMP group whose interface is `ipmp0`.
- A data address of `ipmp0`, `192.0.2.0`, is associated with the non-global zone `my-zone`.
- The `net0` card is set as the zone's physical interface.

In a running zone, you can use the `ipadm` command to make the association. See "Shared-IP Network Interfaces" on page 135 and the `ipadm`(1M) man page for more information.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **In the global zone, configure IPMP groups.**

   Follow procedures in "Configuring IPMP Groups" in *Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.3*.

3. **Use the `zonecfg` command to configure the zone.**

   When you configure the `net` resource, add address `192.0.2.0` and physical interface `net0` to the zone *my-zone*:

   ```
   zonecfg:my-zone> add net
   zonecfg:my-zone:net> set address=192.0.2.0
   zonecfg:my-zone:net> set physical=net0
   zonecfg:my-zone:net> end
   ```

   Only `net0` would be visible in non-global zone `my-zone`.

If `net0` Subsequently Fails

If net0 subsequently fails, because `192.0.2.0` is assigned to `ipmp0`, then that address continues to be available through `net1`. The availability of the address also applies to `my-zone`. However, after `net0` has failed, then `net1` becomes the visible interface of `my-zone`.

Administering Datalinks in Exclusive-IP Non-Global Zones

The `dladm` command is used from the global zone to administer datalinks.

How to Use the dladm show-linkprop Command

The dladm show-linkprop command shows the assignment of datalinks to running exclusive-IP zones.

1. **Become a network administrator.**

   You must be assigned the Network Management rights profile. The root role has all of these rights.

   For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Show the assignment of datalinks on the system.**

```
global$ dladm show-linkprop
```

Example 10 Viewing Zone Datalink Assignment

1. `global$ dladm show-linkprop`
2. Note that the net0 link is assigned to the zone vzl-100.

```
global$ dladm show-linkprop
LINK     PROPERTY        PERM VALUE        EFFECTIVE    DEFAULT   POSSIBLE
net2     speed           r-   10           10           10        --
...
vzl-100/net0 autopush    rw   --           --           --        --
vzl-100/net0 zone        rw   vzl-100      vzl-100      --        --
vzl-100/net0 state       r-   up           up           up        up,down
vzl-100/net0 mtu         rw   1500         1500         1500      576-1500
vzl-100/net0 maxbw       rw   --           --           --        --
vzl-100/net0 cpus        rw   --           0-3,8-11     --        --
vzl-100/net0 rxfanout    rw   --           8            1         --
vzl-100/net0 pool        rw   --           --           --        --
vzl-100/net0 priority    rw   high         high         high      low,medium,
                                                                  high
vzl-100/net0 tagmode     rw   vlanonly     vlanonly     vlanonly  normal,
                                                                  vlanonly
vzl-100/net0 protection  rw   mac-nospoof  mac-nospoof  --        mac-nospoof,
                                                                  restricted,
                                                                  ip-nospoof,
                                                                  dhcp-nospoof
vzl-100/net0 mac-address rw   0:16:3e:86:11:f5 0:16:3e:86:11:f5 0:16:3e:86:11:f5 --
vzl-100/net0 allowed-ips rw   --           --           --        --
```

```
vzl-100/net0 allowed-dhcp-cids rw --       --           --        --
vzl-100/net0 rxrings     rw   --           --           --        --
vzl-100/net0 txrings     rw   --           --           --        sw,hw
vzl-100/net0 txringsavail r-  0           0            --        --
vzl-100/net0 rxringsavail r-  0           0            --        --
vzl-100/net0 rxhwclntavail r- 0           0            --        --
vzl-100/net0 txhwclntavail r- 0           0            --        --
vzl-100/net0 vsi-typeid  rw   --          116          --        --
vzl-100/net0 vsi-vers    rw   --          0            --        --
vzl-100/net0 vsi-mgrid   rw   --          ::           --        --
vzl-100/net0 vsi-mgrid-enc rw --          oracle_v1    oracle_v1 none,
                                                                 oracle_v1
vzl-100/net0 lro         rw   off         off          auto      on,off,auto
vzl-100/net0 cos         rw   --          --           0         --
vzl-100/net0 etsbw-lcl   rw   --          --           0         --
vzl-100/net0 etsbw-rmt   r-   --          --           --        --
vzl-100/net0 etsbw-lcl-advice r- --       --           --        --
vzl-100/net0 etsbw-rmt-advice rw --       --           0         --
```

Example 11 How to Display the Datalink Name and Physical Location When Using Vanity Naming

Device physical locations are shown in the LOCATION field. To view the datalink name and physical location information for a device, use the -L option.

```
global$ dladm show-phys -L
LINK         DEVICE        LOCATION
net0         net0       MB
net1         net1       MB
net2         net2       MB
net3         net3       MB
net4         ibp0          MB/RISER0/PCIE0/PORT1
net5         ibp1          MB/RISER0/PCIE0/PORT2
net6         eoib2         MB/RISER0/PCIE0/PORT1/cloud-nm2gw-2/1A-ETH-2
net7         eoib4         MB/RISER0/PCIE0/PORT2/cloud-nm2gw-2/1A-ETH-2
```

▼ How to Use dladm to Assign Temporary Datalinks

The dladm command can be used with the set-linkprop subcommand to temporarily assign datalinks to running exclusive-IP zones. Persistent assignment must be made through the zonecfg command.

1. **Become a network administrator.**

You must be assigned the Network Management rights profile. The root role has all of these rights.

For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Use dladm set-linkprop with the -t to add net0 to a running zone called zoneA.**

```
global$ dladm set-linkprop -t -p zone=zoneA net0
LINK     PROPERTY        PERM   VALUE        DEFAULT        POSSIBLE
net0     zone            rw     zoneA        --             --
```

Tip - The -p option produces a display using a stable machine-parsable format.

How to Use the dladm reset-linkprop Command

The dladm reset-linkprop command resets the net0 link value to unassigned.

1. **Become a network administrator.**

You must be assigned the Network Management rights profile. The root role has all of these rights.

For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Use dladm reset-linkprop with the -t option to undo the zone assignment of the net0 device.**

```
global$ dladm reset-linkprop -t -p zone net0
LINK     PROPERTY        PERM   VALUE        DEFAULT        POSSIBLE
net0     zone            rw     zoneA        --             --
```

Tip - The -p option produces a display using a stable machine-parsable format.

Troubleshooting If the running zone is using the device, the reassignment fails and an error message is displayed. See "Exclusive-IP Zone Is Using Device, so dladm reset-linkprop Fails" on page 189.

Using the Fair Share Scheduler on an Oracle Solaris System With Zones Installed

Limits specified through the `prctl` command are not persistent. The limits are only in effect until the system is rebooted. To set shares in a zone permanently, see "How to Configure the Zone" on page 22 and "How to Set `zone.cpu-shares` in the Global Zone" on page 33.

▼ How to Set FSS Shares in the Global Zone Using the `prctl` Command

The global zone is given one share by default. You can use this procedure to change the default allocation. Note that you must reset shares allocated through the `prctl` command whenever you reboot the system.

1. **Assume the `root` role.**

   For more information, see "Using Your Assigned Administrative Rights" in *Securing Users and Processes in Oracle Solaris 11.3*.

2. **Use the `prctl` utility to assign two shares to the global zone:**

   ```
   # prctl -n zone.cpu-shares -v 2 -r -i zone global
   ```

3. **(Optional) To verify the number of shares assigned to the global zone, type:**

   ```
   # prctl -n zone.cpu-shares -i zone global
   ```

See Also For more information on the `prctl` utility, see the `prctl(1)` man page.

▼ How to Change the `zone.cpu-shares` Value in a Zone Dynamically

This procedure can be used in the global zone or in a non-global zone.

1. **Become a zone administrator.**

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Use the `prctl` command to specify a new value for `cpu-shares`.**

   `# prctl -n zone.cpu-shares -r -v` *value* `-i zone` *zonename*

   idtype is either the *zonename* or the *zoneid*. *value* is the new value.

Backing Up an Oracle Solaris System With Installed Zones

The following procedures can be used to back up files in zones. Remember to also back up the configuration files of the zones.

▼ Backing Up an Oracle Solaris System by Using Unified Archives

1. **Become a zone administrator.**

   You must also be assigned the Unified Archive Administration rights profile. The `root` role has all of these rights.

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Create a recovery archive.**

   source-host`$ archiveadm create -r -z` *zonename archive-name*

   For example, using the zone name `my-zone` and the archive path `/net/server/my-zone-archive.uar`:

   ```
   host1$ archiveadm create -r -z my-zone /net/server/my-zone-archive.uar
   ```

▼ x64: How to Print a Copy of a Zone Configuration

You should create backup files of your non-global zone configurations. You can use the backups to recreate the zones later if necessary. Create the copy of the zone's configuration after you have logged in to the zone for the first time and have responded to the `sysidtool` questions. This procedure uses a zone named `my-zone` and a backup file named `my-zone.config`.

1. **Become a zone administrator.**

For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Print the configuration for the zone `my-zone` to a file named `my-zone.config`.**

```
global$ zonecfg -z my-zone export > my-zone.config
```

Recreating a Non-Global Zone

▼ How to Recreate an Individual Non-Global Zone

You can use the backup files of your non-global zone configurations to recreate non-global zones, if necessary. This procedure uses a zone named `my-zone` and a backup file named `my-zone.config` to illustrate the process of recreate a zone.

1. **Become a zone administrator.**

   For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Specify that `my-zone.config` be used as the `zonecfg` command file to recreate the zone `my-zone`.**

```
global$ zonecfg -z my-zone -f my-zone.config
```

3. **Install the zone.**

```
global$ zoneadm -z my-zone install -a /path/to/archive  options
```

4. **If you have any zone-specific files to restore, such as application data, manually restore (and possibly manually merge) files from a backup into the newly created zone's root file system.**

♦ ♦ ♦ CHAPTER 11

Configuring and Administering Immutable Zones

This chapter describes how to configure immutable zones by specifying the zone's security policy and how to maintain immutable zones by configuring administrative access. All zones: the physical global zone, non-global zones, and virtual global zones (called kernel zones) can be made immutable.

This chapter covers the following topics:

About Immutable Zones

An *immutable zone* is a zone with a read-only `root` file system. The read-only root preserves the zone's configuration. Also, additional restrictions to the runtime environment extend the zone's secure runtime boundary. Maintenance operations are possible, but you as administrator must take deliberate steps to access the zone for maintenance. The mandatory write access control (MWAC) security policy blocks modifications to system binaries or system configurations.

MWAC is used to enforce file system write privilege through an SMF property, `file-mac-profile`. You can specify the MWAC security policy by modifying the `file-mac-profile` value with the `zonecfg` command. The policy is enforced in the kernel. Because the global zone is not subject to the MWAC policy of a non-global zone, the global zone can write to a non-global zone's file system for installation, image updates, and maintenance.

The MWAC policy is downloaded when the zone enters the ready state. The policy is enabled at zone boot. To perform post-install assembly and configuration, a temporary writable root-file

system boot sequence is used. Modifications to the zone's MWAC configuration only take effect when you reboot the zone.

Configuring Immutable Zones

Mutable and immutable zones are differentiated by their MWAC security policy, which you specify with the `file-mac-profile` property of the `zonecfg` command.

Setting the MWAC Security Policy

By default, the `file-mac-profile` property is not set and the zone has a writable root dataset.

Several values for `file-mac-profile` restrict access to all or part of the runtime environment from inside the zone. All of the profiles except `none` will cause the `/var/pkg` directory and its contents to be read-only from inside the zone. The `none` MWAC security policy is equivalent to an unset MWAC security policy.

The following MWAC values restrict access to all or part of the runtime environment from inside the zone:

`strict`

Read-only file system, no exceptions.

- IPS packages cannot be installed.
- Persistently enabled SMF services are fixed.
- SMF manifests cannot be added from the default locations.
- Logging and auditing configuration files are fixed. Data can only be logged remotely.
- Running an NFS server inside an immutable zone with this profile is not supported. You must use the `fixed-configuration` profile to run an NFS server.

`fixed-configuration`

Permits updates to `/var/*` directories, with the exception of directories that contain system configuration components.

- IPS packages, including new packages, cannot be installed.
- Persistently enabled SMF services are fixed.
- SMF manifests cannot be added from the default locations.
- Logging and auditing configuration files can be local. `syslog` and audit configuration are fixed.

`flexible-configuration`

Permits modification of files in `/etc/*` directories, changes to root's home directory, and updates to `/var/*` directories. This configuration provides the closest functionality to the Oracle Solaris 10 `native` sparse root zone documented in the Oracle Solaris 10 guide, *System Administration Guide: Oracle Solaris Containers-Resource Management and Oracle Solaris Zones*.

- IPS packages, including new packages, cannot be installed.
- Persistently enabled SMF services are fixed.
- SMF manifests cannot be added from the default locations.
- Logging and auditing configuration files can be local. `syslog` and audit configuration can be changed.

`dynamic-zones`

Is valid for global zones, including the global zone of a kernel zone. Permits the creation and the destroying of kernel zones and non-global zones.

Is equivalent to `fixed-configuration`, but adds the ability to create and destroy kernel zones and non-global zones.

Is similar to `flexible-configuration`, but `dynamic-zones` cannot write to files in the `/etc` directory.

EXAMPLE 12 Setting the MWAC Security Policy for a Zone

In this example, you are assigned the Zone Security rights profile and create an immutable zone. In this zone, the zone administrator can create and destroy kernel and non-global zones. Otherwise, the zone is immutable.

```
$ zonecfg -z global set file-mac-profile=dynamic-zones
```

After the MWAC security policy is set and you reboot the immutable zone, the zone boots transient read-write until it reaches the `self-assembly-complete` milestone and then reboots in read-only mode.

Zone Resource Exceptions to MWAC Security Policy

Datasets that you add to a zone through the `zonecfg add dataset` command are not subject to MWAC policy. Zones have full control over added datasets. The platform datasets are visible, but their data and their properties are read-only unless the zone is booted read/write.

File systems that you add to a zone through the `zonecfg add fs` command are not subject to MWAC policy. To maintain the policy, mount the file systems read-only.

SMF Commands Exception to MWAC Security Policy

Changes that you make to an immutable zone through the SMF commands described in the `svcadm`(1M) and `svccfg`(1M) man pages are only applicable to the temporary, live SMF database, not to the on-disk SMF database. If you modify the zone's MWAC security policy, the changes take effect when the zone is rebooted.

Administering Immutable Non-Global Zones

If you do not configure administrative access, you can administer the on-disk non-global zone from the global zone only. Within a running non-global zone, you can only administer the runtime state of an immutable zone. Therefore, modifying MWAC policy in a running zone is temporary. For more information, see "SMF Commands Exception to MWAC Security Policy" on page 186.

Determining Whether a Non-Global Zone Is Immutable

The parsable output of the `zoneadm list -p` command from the global zone displays an `R/W` column, and a `file-mac-profile` column. In the following output, the `fixed-configuration` running zones, `testzone2` and `testzone3`, are read-only, while the `testzone1` running zone is a read-write `fixed-configuration` zone.

```
global$ zoneadm list -p
0:global:running:/:UUID:solaris:shared:-:none
 5:testzone2:running:/export/zones/testzone2:UUID \
     :solaris:shared:R:fixed-configuration
 12:testzone3:running:/export/zones/testzone3:UUID \
     :solaris:shared:R:fixed-configuration
 13:testzone1:running:/export/zones/testzone1:UUID \
     :solaris:excl:W:fixed-configuration
 -:testzone:installed:/export/zones/testzone:UUID \
     :solaris:excl:-:fixed-configuration
```

Administering an Immutable Zone by Making It Writable

The `zoneadm boot` subcommand provides two options that allow the global zone administrator to manually boot an immutable zone with either a writable root file system or with a transient writable root file system. The zone is in writable mode only until the next reboot.

`-w`

Manually boot the zone with a writable `root` file system.

`-W`

Manually boot the zone with a transient writable `root` file system. The system is rebooted automatically when the `self-assembly-complete` milestone is reached. The reboot places the zone under control of the MWAC policy again. This option is permitted when the zone has an MWAC policy of `none`.

Both the `-W` and `-w` options are ignored for zones that are not immutable zones.

The `zlogin` command provides the `-U` option for actions such as editing an immutable file or adding a new package. Use of this option requires the authorization `solaris.zone.manage/`*zonename*. This option operates in unsafe mode, where unprotected files can be modified. You use this option for zones with the `flexible-configuration` MWAC security policy.

Note - These options cannot be used with console login and are ignored for zones that are not immutable zones.

Immutable Global Zones

Immutable global zones extend immutable non-global zones to global zones.

Configuring an Immutable Global Zone

To configure an immutable global zone is similar to configuring an immutable non-global zone. The MWAC security policy is set with the `zonecfg` command, as “Setting the MWAC

Security Policy" on page 184 describes. After committing the zone configuration, the boot information is written and the boot archive is updated. The global zone becomes immutable immediately. No reboot is necessary.

The following information is specific to immutable global zones:

- If the global zone uses DHCP to set network interfaces, the `flexible-configuration` MWAC policy must be selected.
- The `rpool` dataset is restricted.

  You can add an unrestricted sub-dataset by using the `zonecfg add dataset` command. An immutable global zone can only run zones in unrestricted datasets. All the children of an unrestricted dataset are also unrestricted.

Maintaining an Immutable Global Zone

When you run a package update on the immutable global zone, the first boot is read-write. The system needs these permissions to perform the required self-assembly steps. When the self-assembly steps have been performed, the system becomes immutable again.

♦ ♦ ♦ CHAPTER 12

Troubleshooting Miscellaneous Oracle Solaris Zones Problems

This chapter contains zones troubleshooting information.

Installation Fails Due to Unmatched Allowable Packages

The following error message when you attempt to install a native zone indicates a publisher mismatch between the global zone and the non-global zone:

```
The following pattern(s) did not match any allowable packages. Try
using a different matching pattern, or refreshing publisher information:
```

The global zone currently might be configured with a publisher that does not contain the same version of the installed system software. To install a non-global zone, the repository that you set as the `solaris` publisher origin must contain at least the same system software that is installed in the global zone where you are installing the non-global zone.

See "Non-Global Zone Cannot Be Installed" in *Adding and Updating Software in Oracle Solaris 11.3* for more information.

Exclusive-IP Zone Is Using Device, so `dladm reset-linkprop` Fails

The following error message indicates an attempt to use the `dladm reset-linkprop` command failed.

```
dladm: warning: cannot reset link property 'zone' on 'net0': operation failed
```

See “How to Use the dladm reset-linkprop Command” on page 179. The running zone excl is using the device.

You can reset the value:

1. Type:

```
global$ ipadm delete-ip net0
```

2. Rerun the dladm command.

Incorrect Privilege Set Specified in Zone Configuration

If the zone's privilege set contains a disallowed privilege, is missing a required privilege, or includes an unknown privilege name, an attempt to verify, ready, or boot the zone will fail with an error message such as the following:

```
zonecfg:zone5> set limitpriv="basic"
global$ zoneadm -z zone5 boot
  required privilege "sys_mount" is missing from the zone's privilege set
  zoneadm: zone zone5 failed to verify
```

To fix the problem, see “How to Modify Zone Privileges” on page 34.

Zone Does Not Halt

In the event that the system state associated with the zone cannot be destroyed, the halt operation will fail halfway. This leaves the zone in an intermediate state, somewhere between running and installed. In this state there are no active user processes or kernel threads, and none can be created. When the halt operation fails, you must manually intervene to complete the process.

The most common cause of a failure is the inability of the system to unmount all file systems. Unlike a traditional Oracle Solaris system shutdown, which destroys the system state, zones must ensure that no mounts performed while booting the zone or during zone operation remain once the zone has been halted. Even though zoneadm makes sure that there are no processes executing in the zone, the unmount operation can fail if processes in the global zone have open files in the zone.

Use the tools described in the proc(1) (see pfiles(1)) and fuser(1M) man pages to find these processes and take appropriate action. After these processes have been dealt with, rerun zoneadm halt to completely halt the zone.

♦ ♦ ♦ CHAPTER 13

Getting Started With Oracle Solaris Zones on Shared Storage

You can use Oracle Solaris Zones on shared storage in Oracle Solaris to transparently access and manage shared storage resources in zones. These automated capabilities simplify deployment, administration, and migration of zones and their corresponding shared storage resources in Oracle Solaris systems.

You can describe the corresponding shared storage resources in a host-independent format in the zone configuration. Zones installations using this feature are encapsulated into dedicated ZFS storage pools hosted on shared storage devices.

The Oracle Solaris Zones framework will automatically configure and unconfigure shared storage resources. Any ZFS storage pool management tasks required throughout the various zone management activities will be performed automatically.

About Shared Storage Resources Using Storage URIs

Storage URIs are used to describe shared storage resources in a host-independent format. Storage URIs uniquely identify storage objects across different nodes. They follow the well-known principles and format of URIs commonly used in the Internet. The following storage URIs have been introduced with Oracle Solaris.

Local Device URI

The local device storage URI type describes a storage device by its local device path. The path must refer to a device in the `/dev` name space. These devices are usually direct-attached storage (DAS) resources that are unique to a particular system, and their device name and path are generally not portable. However, they can also refer to shared storage resources for technologies that already provide a unified name space under `/dev` across multiple nodes.

The following is a list of URI syntax.

- `dev:`*local-path-under-/dev*
- `dev:///`*path-with-dev*
- `dev:`*absolute-path-with-dev*

Examples of the use of the URI syntax are as follows:

- `dev:dsk/c0t0d0s0`
- `dev:///dev/dsk/c0t0d0`
- `dev:/dev/dsk/c0t0d0`
- `dev:chassis/SYS/HD1/disk`
- `dev:dsk/c0t60A98000564C303132302D6F72613939d0`

The local device storage URI can refer to an entire disk or to a particular slice or partition. However, the use of slices or partitions is generally not recommended with ZFS storage pools.

Logical Unit URI

The logical unit URI type describes fibre channel (FC) or serial-attached SCSI (SAS) storage devices. It refers to a logical unit (LU) based on its device ID (WWN). The logical unit storage URI always represents an entire disk.

The following is a list of URI syntax.

- `lu:luname.naa.`*ID*
- `lu:luname.eui.`*ID*
- `lu:initiator.naa.`*ID*`,target.naa.`*ID*`,luname.naa.`*ID*
- `lu:initiator.naa.`*ID*`,target.naa.`*ID*`,luname.eui.`*ID*

The following examples show how to use the URI syntax:

- `lu:luname.naa.5000c5000288fa25`
- `lu:luname.eui.0021280001cf80f6`
- `lu:initiator.naa.2100001d38089fb0,target.naa.2100001d38089fb0,luname.naa.5000c5000288fa25`
- `lu:initiator.naa.2100001d38089fb0,target.naa.2100001d38089fb0,luname.eui.0021280001cf80f6`

In the `luname`-only URI form, the ID describes a logical unit name.

In the `initiator,target,luname` form, an initiator specifies an initiator port and a target specifies a target port. Together, they specify a path to the logical unit. The logical unit name in the second syntax must match the `luname` of the URI in the first syntax, the `luname`-only form.

It is best to use multipathing in conjunction with `luname`-only URIs. If multipathing is disabled and a `luname`-only URI is used, a random path to the specified logical unit will be chosen. To avoid random path assignment an `initiator,target,luname` URI form can be used to select a specific path to a logical unit. If multipathing is enabled and an `initiator,target,luname` URI form is used, then the multipathing framework controls which paths are used to access the logical unit, and the URI is only used to identify the unit, not the access path.

iSCSI URI

The iSCSI URI type describes storage devices accessed by using the iSCSI network-based storage protocol. It always refers to an entire disk.

For more information on iSCSI Qualified Name (IQN), see the `suri(5)` man page.

The following is a list of URI syntax.

- `iscsi:///luname.naa.`*ID*
- `iscsi:///luname.eui.`*ID*
- `iscsi://`*host*`[:`*port*`]/luname.naa.`*ID*
- `iscsi://`*host*`[:`*port*`]/luname.eui.`*ID*
- `iscsi:///target.`*IQN*`,lun.`*LUN*
- `iscsi://`*host*`[:`*port*`]/target.`*IQN*`,lun.`*LUN*

The following examples show how to use the URI syntax:

- `iscsi:///luname.eui.0021280001cf80f6`
- `iscsi:///luname.naa.600144f03d70c80000004ea57da10001`
- `iscsi://[::1]/luname.naa.600144f03d70c80000004ea57da10001`
- `iscsi://127.0.0.1/luname.naa.600144f03d70c80000004ea57da10001`
- `iscsi://127.0.0.1:3260/luname.naa.600144f03d70c80000004ea57da10001`
- `iscsi://hostname:3260/luname.eui.0021280001cf80f6`
- `iscsi://hostname:3260/luname.naa.600144f03d70c80000004ea57da10001`
- `iscsi://[::1]/target.iqn.com.sun:02:d0f2d311-f703,lun.2`
- `iscsi://hostname:3620/target.iqn.com.sun:4db41b76-e3d7-cd2f-bf2d-9abef784d76c,lun.0`
- `iscsi:///target.iqn.com.sun:02:d0f2d311-f703,lun.6`
- `iscsi://[::1]/target.iqn.com.sun:02:d0f2d311-f703,lun.2`
- `iscsi://hostname:3620/target.iqn.com.sun:4db41b76-e3d7-cd2f-bf2d-9abef784d76c,lun.0`

See “Logical Unit URI” on page 192 for an explanation of the `luname`-only URI form and the ID.

Optionally, the `hostname[:`*port*`]` authority section provides information to automatically configure the iSCSI initiator using an unauthenticated `SendTargets` discovery address. IPv6 addresses must be enclosed in square brackets (`[]`).

Use the `luname`-only URI form only for static or iSNS based configurations, or when using authentication. These functions must be configured outside of the zones framework before they can be used.

Note that if you want to use iSCSI over iSER transports, the iSCSI storage URI transparently allows for this as well. To use iSER, the target and initiator must go through the address assigned to the InfiniBand (IB) partition link. In the global zone, you must set up the IB partitions with the `dladm create-part` command and assign the target and initiator addresses to the IB partitions using the `ipadm create-ip` and `ipadm create-addr` commands. This address is then used in the authority section of the iSCSI storage URI to specify the target's discovery address.

If InfiniBand (IB) hardware is present and an InfiniBand reliable-connected (RC) connection can be established, then an iSER-enabled initiator uses iSER connections to iSER-enabled targets. If the RC connection cannot be established, the connection is established using IP-based connectivity.

Managing Storage URIs and Shared Storage Resources

To generate and verify storage URIs or administer shared storage resources based on storage URIs, use the `suriadm` command.

You can use the `suriadm` command to verify storage URIs that were created manually, or to create storage URIs automatically, based on existing system device paths. Depending on the storage URI type, the `suriadm` command allows you to configure and unconfigure the corresponding storage subsystem. Given a storage URI, the same command can also identify possible instantiated device instances for the storage object described by the storage URI. The following examples demonstrate common use cases. For more information, see the `suriadm`(1M) man page.

EXAMPLE 13 Verifying Storage URIs With the `suriadm parse` Syntax

The following examples show how to use the `suriadm` command to verify storage URIs. In these examples, the storage URI string has been parsed and its corresponding properties are

displayed. Once verified, the storage URI can be used later with either the `zonecfg` command or the `suriadm` command.

```
root@initiator:~# suriadm parse iscsi://target/luname.naa.
600144F035FF8500000050C884E50001
PROPERTY    VALUE
uri-type    iscsi
hostname    target
port        -
luname      naa.600144F035FF8500000050C884E50001

root@host:~# suriadm parse dev:/dev/dsk/c4t1d0
PROPERTY    VALUE
uri-type    dev
path        /dev/dsk/c4t1d0
```

EXAMPLE 14 Producing Storage URIs Based on Device Path With the `suriadm lookup-uri` Syntax

The following example shows how to use the `suriadm` command to produce storage URIs. In the example, based on the existing local device path, the `suriadm` command output suggests valid storage URIs for later use with either the `suriadm` or `zonecfg` commands.

```
root@target:~# suriadm lookup-uri -t iscsi /dev/dsk/
c0t600144F035FF8500000050C884E50001d0
iscsi://target/luname.naa.600144f035ff8500000050c884e50001

root@host:~# suriadm lookup-uri /dev/dsk/c4t1d0
dev:dsk/c4t1d0

root@host:~# suriadm lookup-uri /dev/dsk/c0t600144F0DBF8AF190000510979640005d0
lu:luname.naa.600144f0dbf8af190000510979640005
lu:initiator.naa.10000000c9991d8c,target.naa.21000024ff3ee89f,luname.naa.
600144f0dbf8af190000510979640005
dev:dsk/c0t600144F0DBF8AF190000510979640005d0
```

EXAMPLE 15 Configuring iSCSI based Storage Resources With the `suriadm map` Syntax

The following example shows how to use the `suriadm map` command to configure iSCSI based storage resources. In this example, for an iSCSI storage URI, the `suriadm` configures the iSCSI initiator's send-targets discovery address and instantiates a local device representing the iSCSI target. The local device path from the mapped-dev storage URI property can now be used with utilities such as the `zpool`, `format`, and `mkfs` commands.

```
root@initiator:~# suriadm map iscsi://target/luname.naa.600144F035FF8500000050C884E50001
PROPERTY     VALUE
mapped-dev   /dev/dsk/c0t600144F035FF8500000050C884E50001d0s0
```

EXAMPLE 16 Locate a Configured Storage Resource With `suriadm lookup-mapping` Syntax

In this example, the command is used to show the local system device currently associated with the given storage URI.

```
root@initiator:~# suriadm lookup-mapping iscsi://target/luname.naa.
600144F035FF8500000050C884E50001
PROPERTY     VALUE
mapped-dev   /dev/dsk/c0t600144F035FF8500000050C884E50001d0s0
```

EXAMPLE 17 Unconfigure iSCSI-Based Storage Resources With `suriadm unmap`

For an iSCSI storage URI, the `suriadm` command removes the iSCSI initiator's send-targets discovery address, and unconfigures the shared storage resource.

```
root@initiator:~# suriadm unmap iscsi://target/luname.naa.
600144F035FF8500000050C884E50001
root@initiator:~# suriadm lookup-mapping iscsi://target/luname.naa.
600144F035FF8500000050C884E50001
Failed to lookup mapping for URI: "iscsi://target/luname.naa.
600144F035FF8500000050C884E50001": No such logical
 unit name found: "naa.600144F035FF8500000050C884E50001"
```

Assigning Shared Storage Resources to Oracle Solaris Zones

Assign shared storage resources to zones in the zone configuration by using storage URIs to describe the location of a storage object.

Two zone configuration resource types, `rootzpool`, and `zpool`, and a property type, `storage`, are used to assign shared storage resources to a particular Oracle Solaris Zone. These are configured and maintained with the `zonecfg`.

`storage` Property for Zones

When using shared storage resources, the `storage` property defines the location of the storage object in a host-independent format by using storage URIs. The following storage URIs are currently supported by the Oracle Solaris Zones framework in Oracle Solaris:

- `dev`: local device path storage URI, DAS

- `iscsi`: iSCSI storage URI
- `lu`: Fibre Channel (FC) and Serial Attached SCSI (SAS)

The storage property is managed using the following `zonecfg` subcommands from within a `rootzpool` or `zpool` resource scope:

```
zonecfg:zonename:zpool> add storage URI string
zonecfg:zonename:zpool> remove storage URI string
```

`rootzpool` Resource

The `rootzpool` resource is a dedicated ZFS storage pool for a zone. The entire zone installation is encapsulated into its own dedicated ZFS storage pool. This ZFS storage pool will be composed of shared storage resources.

The `rootzpool` resource must specify at least one storage property. Multiple storage properties can be specified to describe redundant ZFS storage pool configurations. Note that there can be only one `rootzpool` resource per zone configuration.

The ZFS storage pool name for a `rootzpool` resource will be automatically assigned as *zonename*_`rpool`. The name cannot be changed. The `rootzpool` resource is managed by using the following `zonecfg` subcommands from the global resource scope:

```
zonecfg:zonename> add rootzpool
zonecfg:zonename:rootzpool> add storage URI string
zonecfg:zonename:rootzpool> end

zonecfg:zonename> remove rootzpool

zonecfg:zonename> select rootzpool storage=URI string

zonecfg:zonename> info rootzpool
```

`zpool` Resource

The `zpool` resource describes a ZFS storage pool composed of shared storage resources that is delegated to the Oracle Solaris Zone. The `zpool` resource will specify at least one storage property. Multiple storage properties can be specified to describe redundant ZFS storage pool configurations. There can be multiple `zpool` resources defined for a zone configuration.

The ZFS storage pool name for a `zpool` resource is assigned by combining the zone name and the specified name property, as *zonename_name*. For the name property, the `zonecfg` will verify

that the string is eligible for a ZFS storage name and a ZFS dataset name. The string `rpool` is not permitted and cannot be used for this property.

The `zpool` resource is managed using the following `zonecfg` subcommands from the global resource scope:

```
zonecfg:zonename> add zpool
zonecfg:zonename:zpool> add storage URI string
zonecfg:zonename:zpool> set name=name string
zonecfg:zonename:zpool> end

zonecfg:zonename> remove zpool
zonecfg:zonename> remove zpool name=name string
zonecfg:zonename> remove zpool storage=URI string

zonecfg:zonename> select zpool storage=URI string

zonecfg:zonename> info zpool
zonecfg:zonename> info zpool name=name string
zonecfg:zonename> info zpool storage=URI string
```

Renaming Zones

You can use the `zoneadm` command to rename a zone in either the configured or the installed state.

Restrictions on Zone Configuration

To alter a storage resource URI within a `rootzpool` or `zpool` resource when the location description changes for an existing shared storage resource, use the `zonecfg remove storage` *old URI* command syntax followed by the `add storage` *new URI* command.

Automated ZFS Storage Pool Management for Oracle Solaris Zones on Shared Storage Resources

When using the zones on the shared storage support feature in Oracle Solaris, the zones framework will automatically manage all of the ZFS storage pools associated with the `rootzpool` or `zpool` resources for a particular zone.

To facilitate zone and storage resource migration, ZFS storage pools associated with `rootzpool` or `zpool` resources are not persistently configured on the system. They will not appear in the system's global ZFS storage pool repository `/etc/zfs/zpool`.cache.

The zones framework creates corresponding ZFS storage pools when you install or clone a zone. You can configure more than one shared storage resource with the `zonecfg add storage` command per `rootzpool` or `zpool` resource. In this case, a mirrored ZFS storage pool will be created by default.

You can create ZFS storage pools in advance by using custom configurations such as encryption), or different redundancy levels like `raidz` or `raidz2`. After first configuring all the required shared storage resources in the zone configuration, use the `zpool` utility to export the pre-created ZFS storage pool again. The zones framework will first attempt to import and use this pre-created ZFS storage pool during zone installation and cloning.

When you use the `zoneadm attach` syntax, the zones framework will initially configure all shared storage resources and then import all configured ZFS storage pools.

For the `zoneadm detach` command syntax, the zones framework will first export all configured ZFS storage pools and then unconfigure all shared storage resources.

When uninstalling a zone with the `zoneadm uninstall` command, the zones framework will first export all configured ZFS storage pools and then unconfigure all shared storage resources by default. However, you can explicitly request that ZFS storage pools be destroyed by using the `-x force-zpool-destroy` option with the `zoneadm uninstall` command.

During system boot, the zones SMF service `svc:/system/zones:default` is run. The SMF service will configure shared storage resources and import ZFS storage pools for all zones in the installed state that have `rootzpool` or `zpool` resources. Any failures during this stage will be logged to the corresponding SMF services log file `/var/svc/log/system-zones:default.log`.

ZFS storage pools configured with `rootzpool` or `zpool` resources will appear as virtualized ZFS storage pools inside the zone, but they cannot be managed directly by the zone administrator.

While in use by a running zone, ZFS storage pools cannot be destroyed or exported by the global zone.

From the global zone, you can use the `zpool` command for administrative actions on ZFS storage pools managed by the zones framework. Administrative actions include bringing a specific device online or taking it offline, replacing faulted devices, and adding or removing devices. To keep the zone configuration in sync, update the corresponding storage URIs accordingly to reflect changes made to the shared storage resources being used.

About the Unavailable Zone State

In Oracle Solaris, the zone state Unavailable, indicates that the zone is installed but cannot be booted. This state is displayed in the output of the zoneadm list -p command syntax.

```
 root@initiator:~# zoneadm list -cp
0:global:running:/::solaris:shared:-:none
-:iscsi:unavailable:/iscsi:a0a4ba0d-9d6d-cf2c-cc42-f123a5e3ee11:solaris:excl:-:
```

An installed zone with shared storage resources can enter the Unavailable state if the zones framework experiences either failures during the configuration of shared storage resources or failures during ZFS storage pool management.

To move the zone out of the Unavailable state, you must first identify and possibly fix any problems related to shared storage resource connectivity or zone misconfiguration. You can then use the zoneadm attach command to reattach the zone properly and move it into the installed zone state. It is also possible to uninstall a zone with the zoneadm uninstall command to move the zone back into the configured zone state.

For more information about zones states, see "Non-Global Zone State Model" in *Introduction to Oracle Solaris Zones*.

Additional zoneadm Subcommand Options

Where appropriate, the zoneadm CLI subcommands have been enhanced with options specific to ZFS storage pool management.

Options for Installing, Cloning, and Attaching Zones

For the install, clone, and attach subcommands of zoneadm, three available options are discussed in the following sections:

-x force-zpool-import Option

This option applies to all zpool resources specified in the zone configuration and instructs the zones framework to forcibly import any ZFS storage pools that might appear to be in use, such as by another system. This mimics the behavior of zpool import -f command.

This option facilitates importing ZFS storage pools onto a new system during zone migration when the ZFS storage pools were never properly exported on another system, for example, by using the zoneadm detach or zoneadm uninstall commands. In these situations, the forcible import must be applied to all ZFS storage pools configured for a particular zone.

-x force-zpool-create= Option

This option instructs the zones framework to forcibly create a new ZFS storage pool, either over an existing old pool or onto a device that appears to be currently in use for other purposes. This mimics the behavior of the zpool create -f command syntax.

This option's scope is limited to the ZFS storage pools specified in a comma-separated list of zpool names or specified by using the option multiple times, for example:

```
-x force-zpool-create=rpool,pool2,pool3
-x force-zpool-create=pool1 -x force-zpool-create=pool2
```

The ZFS storage pool name to be used here is the name property of the corresponding zpool resource in the zone configuration. To specify the ZFS storage pool described by the rootzpool resource, use the name rpool.

-x force-zpool-create-all Option

This option instructs the zones framework to forcibly create new ZFS storage pools for all zpool and rootzpool resources specified in the zone configuration.

Options for Uninstalling Zones

For the zoneadm uninstall subcommand, the following three options are available:

-x force-zpool-destroy=zpoolname Option

This option instructs the zones framework to destroy a ZFS storage pool while uninstalling a zone. This mimics the behavior of `zpool destroy poolname` command. This option takes a *zpoolname* list:

`force-zpool-destroy=zpoolname`*{,zpoolname,zpoolname,...}*

The scope of this option is limited to the ZFS storage pools specified in the comma-separated list of `zpool` names, or specified by using the option multiple times, for example:

```
-x force-zpool-destroy=rpool,pool2,pool3
-x force-zpool-destroy=pool1 -x force-zpool-destroy=pool2
```

The name arguments correspond to the name property specified in the zone configuration for the desired `zpool` resource. The name `rpool` is used to specify the ZFS storage pool associated with the `rootzpool` resource.

-x force-zpool-destroy-all Option

This option instructs the zones framework to destroy the ZFS storage pools of all zpool and `rootzpool` resources specified in the zone configuration when uninstalling the zone.

Note that the options to force the zones framework to destroy ZFS storage pools when uninstalling a zone should be used with caution. Even though you might want to uninstall a zone in the current, active boot environment (BE), there could be other zone boot environments (ZBEs) within this ZFS storage pool that belong to currently inactive boot environments. Destroying the ZFS storage pool associated with a `rootzpool` resource will make the zone unavailable to all boot environments that carry this zone in the installed state. For more information about zone boot environments, refer to the `beadm`(1M) man page.

-x force-storage-destroy-all Option

Specify this option to destroy storage. Note that not all storage URI types support this operation. You can destroy storage referenced by file or NFS storage URIs, but not storage referenced by `dev`, iSCSI, or `lu` storage URIs.

Restrictions in the Use of the `zoneadm` Command

For zones configured with a `rootzpool` resource, the `zoneadm move` command syntax is restricted to only permit renaming the `zonepath`. It is not possible to move a zone with a `rootzpool` resource out of its corresponding ZFS storage pool into a different place in the file system hierarchy or into another ZFS storage pool.

Implementing Oracle Solaris Zones Hosted on Shared Storage Resources

The following sections discuss deploying Oracle Solaris Zones on shared storage resources.

`zpool` Configuration Considerations for Zones on Shared Storage

When configuring ZFS storage pools, you might want to consider using redundant pool configurations such as mirror, `raidz`, or `raidz2`. Even if the backend storage hosting the shared storage resources provides resilience and data protection, in the form of hardware RAID for example, redundant configurations enable ZFS storage pools to use their inherent self-healing capabilities.

There are two ways to use redundant ZFS storage pool configurations with Oracle Solaris Zones hosted on shared storage resources.

- You can use mirrored ZFS storage pool configurations, which by default will be created automatically if you specify multiple storage properties for a `rootzpool` or `zpool` resource in the zone configuration.
- Alternatively, you can create more complex ZFS storage pool configurations such as `raidz` or `raidz2` in advance, with the `zpool create` command. Note that you must export the new created ZFS storage pool again using `zpool export`. Next, all corresponding storage resource URIs should be added to the `rootzpool` or `zpool` resource in the zone configuration with the `zonecfg` command. The zones framework will now import this ZFS storage pool during zone installation or clone rather then attempt to create a new ZFS storage pool.

Sample Scenarios for Zones on Shared Storage

This section provides examples for deploying Oracle Solaris Zones on shared storage resources.

EXAMPLE 18 Oracle Solaris Zones Using iSCSI-Based Shared Storage Devices

This exercise sets up a sample configuration on an Oracle Solaris server that will provide shared storage through an iSCSI target. A zone is configured and installed on a second server running Oracle Solaris, using the iSCSI-based shared storage resources to host a zone.

First, install the corresponding package, using one of the following `pkg install` commands. The first command installs the entire multi-protocol storage-server group package. The second command installs only the target support for iSCSI within the common multi-protocol SCSI target (COMSTAR) framework, as described in the `itadm`(1M) and `stmfadm`(1M) man pages.

```
root@target:~# pkg install group/feature/storage-server
root@target:~# pkg install system/storage/iscsi/iscsi-target
```

Then, create the backing store for the iSCSI targets to be exported from this server. Create four ZFS volumes as the backing store for four iSCSI target logical units, each 10GB in size, stored in the target servers `rpool/export` dataset with the `zfs` command.

```
root@target:~# zfs create -V 10G rpool/export/zonevol1
root@target:~# zfs create -V 10G rpool/export/zonevol2
root@target:~# zfs create -V 10G rpool/export/zonevol3
```

After setting up the backing store, use the `stmfadm` command to create target logical units for each ZFS volume. This gives the corresponding device ID (WWN) for each, which will be used later in the storage URI for iSCSI target discovery on the client host.

```
root@target:~# stmfadm create-lu  /dev/zvol/rdsk/rpool/export/zonevol1
Logical unit created: 600144F035FF8500000050C884E50001
root@target:~# stmfadm create-lu  /dev/zvol/rdsk/rpool/export/zonevol2
Logical unit created: 600144F035FF8500000050C884E80002
root@target:~# stmfadm create-lu  /dev/zvol/rdsk/rpool/export/zonevol3
Logical unit created: 600144F035FF8500000050C884EC0003
```

You can view configured logical units with the `stmfadm list-lu` syntax.

```
root@target:~# stmfadm list-lu
LU Name: 600144F035FF8500000050C884E50001
LU Name: 600144F035FF8500000050C884E80002
LU Name: 600144F035FF8500000050C884EC0003
LU Name: 600144F035FF8500000050C884EC0004
```

You can query for details about configured logical units with the `stmfadm list-lu -v` syntax.

```
root@target:~# stmfadm list-lu -v
LU Name: 600144F035FF8500000050C884E50001
    Operational Status     : Online
    Provider Name          : sbd
    Alias                  : /dev/zvol/rdsk/rpool/export/zonevol1
    View Entry Count       : 0
    Data File              : /dev/zvol/rdsk/rpool/export/zonevol1
    Meta File              : not set
    Size                   : 10737418240
    Block Size             : 512
    Management URL         : not set
    Software ID            : not set
    Vendor ID              : SUN
    Product ID             : COMSTAR
    Serial Num             : not set
    Write Protect          : Disabled
    Write Cache Mode Select: Enabled
    Writeback Cache        : Enabled
    Access State           : Active
```

To make the logical unit available to iSCSI initiators, add a logical unit view to the target server with the `stmfadm add-view` command.

```
root@target:~# stmfadm add-view 600144F035FF8500000050C884E50001
root@target:~# stmfadm add-view 600144F035FF8500000050C884E80002
root@target:~# stmfadm add-view 600144F035FF8500000050C884EC0003
```

Now configure the iSCSI target on the target server. First, enable the iSCSI target SMF service with `svcadm enable`.

```
root@target:~# svcadm enable -r svc:/network/iscsi/target:default
```

Then, create the iSCSI target itself using `itadm create-target`.

```
root@target:~# itadm create-target
Target iqn.1986-03.com.sun:02:b62a8291-b89e-41ba-9aef-e93836ad0d6a successfully created
```

You can query for the details about configured iSCSI targets using either `itadm list-target` or `stmfadm list-target`.

```
root@target:~# itadm list-target -v
TARGET NAME                                                  STATE    SESSIONS
iqn.1986-03.com.sun:02:b62a8291-b89e-41ba-9aef-e93836ad0d6a  online   0
        alias:                  -
        auth:                   none (defaults)
        targetchapuser:         -
        targetchapsecret:       unset
        tpg-tags:               default
```

```
root@target:~# stmfadm list-target -v
Target: iqn.1986-03.com.sun:02:b62a8291-b89e-41ba-9aef-e93836ad0d6a
    Operational Status     : Online
    Provider Name          : iscsit
    Alias                  : -
    Protocol               : iSCSI
    Sessions               : 0
```

This completes the tasks performed on the sample server that provides the iSCSI target storage.

The next steps are to configure and install a zone on the second server by using this shared storage provided over iSCSI.

The first step is to install the corresponding package on the client server selected to be the iSCSI initiator.

```
root@initiator:~# pkg install pkg:/system/storage/iscsi/iscsi-initiator
```

Next, use the `zonecfg` command to configure a zone with a `rootzpool` and a `zpool` resource. Use the three iSCSI target logical units configured as shared storage resources to host the zone. Use the iSCSI storage URIs obtained previously with `suriadm`, on the target server.

```
root@initiator:~# zonecfg -z iscsi
Use 'create' to begin configuring a new zone.
zonecfg:iscsi> create
create: Using system default template 'SYSdefault'
zonecfg:iscsi> set zonepath=/iscsi
zonecfg:iscsi> add rootzpool
zonecfg:iscsi:rootzpool> add storage iscsi://target/luname.naa.
600144F035FF8500000050C884E50001
zonecfg:iscsi:rootzpool> end
zonecfg:iscsi> add zpool
zonecfg:iscsi:zpool> set name=data
zonecfg:iscsi:zpool> add storage iscsi://target/luname.naa.
600144F035FF8500000050C884E80002
zonecfg:iscsi:zpool> add storage iscsi://target/luname.naa.
600144F035FF8500000050C884EC0003
zonecfg:iscsi:zpool> end
zonecfg:iscsi> commit
zonecfg:iscsi> exit
```

Install the zone using `zoneadm install`.

```
root@initiator:~# zoneadm -z iscsi install
Configured zone storage resource(s) from:
       iscsi://target/luname.naa.600144F035FF8500000050C884E50001
Created zone zpool: iscsi_rpool
Configured zone storage resource(s) from:
       iscsi://target/luname.naa.600144F035FF8500000050C884E80002
```

```
       iscsi://target/luname.naa.600144F035FF8500000050C884EC0003
Created zone zpool: iscsi_data
Progress being logged to /var/log/zones/zoneadm.20130125T112209Z.iscsi.install
       Image: Preparing at /iscsi/root.

 AI Manifest: /tmp/manifest.xml.pmai7h
  SC Profile: /usr/share/auto_install/sc_profiles/enable_sci.xml
    Zonename: iscsi
Installation: Starting ...

              Creating IPS image
Startup linked: 1/1 done
              Installing packages from:
                  solaris
                      origin:  http://pkg.oracle.com/solaris/release/
DOWNLOAD                                PKGS         FILES    XFER (MB)   SPEED
Completed                            183/183   33556/33556  222.2/222.2  3.4M/s

PHASE                                          ITEMS
Installing new actions                   46825/46825
Updating package state database                 Done
Updating image state                            Done
Creating fast lookup database                   Done
Installation: Succeeded

        Note: Man pages can be obtained by installing pkg:/system/manual

 done.
        Done: Installation completed in 266.487 seconds.

  Next Steps: Boot the zone, then log into the zone console (zlogin -C)
              to complete the configuration process.

Log saved in non-global zone as /iscsi/root/var/log/zones/
zoneadm.20130125T112209Z.iscsi.install
root@initiator:~#
```

With the zone installation completed, verify that the zone has been properly installed with `zoneadm list`.

```
root@initiator:~# zoneadm list -cp
0:global:running:/::solaris:shared:-:none
-:iscsi:installed:/iscsi:a0a4ba0d-9d6d-cf2c-cc42-f123a5e3ee11:solaris:excl:-:
```

Finally, observe the newly created ZFS storage pools associated with this zone by using the `zpool` command.

```
root@initiator:~# zpool list
NAME          SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT
```

```
iscsi_data   9.94G  83.5K  9.94G   0%  1.00x  ONLINE  -
iscsi_rpool  9.94G   436M  9.51G   4%  1.00x  ONLINE  -

root@initiator:~# zpool status -v iscsi_rpool
  pool: iscsi_rpool
 state: ONLINE
  scan: none requested
config:

       NAME                                     STATE     READ WRITE CKSUM
       iscsi_rpool                              ONLINE       0     0     0
         c0t600144F035FF8500000050C884E50001d0  ONLINE       0     0     0

root@initiator:~# zpool status -v iscsi_data
  pool: iscsi_data
 state: ONLINE
  scan: none requested
config:

       NAME                                      STATE     READ WRITE CKSUM
       iscsi_data                                ONLINE       0     0     0
         mirror-0                                ONLINE       0     0     0
           c0t600144F035FF8500000050C884E80002d0  ONLINE       0     0     0
           c0t600144F035FF8500000050C884EC0003d0  ONLINE       0     0     0
```

The zone installation will be entirely contained within this ZFS storage pool. The ZFS dataset layout for this zone follows.

```
root@initiator:~# zfs list -t all|grep iscsi
iscsi_data                                          83.5K  9.78G    31K  /iscsi_data
iscsi_rpool                                          436M  9.36G    32K  /iscsi
iscsi_rpool/rpool                                    436M  9.36G    31K  /rpool
iscsi_rpool/rpool/ROOT                               436M  9.36G    31K  legacy
iscsi_rpool/rpool/ROOT/solaris                       436M  9.36G   390M  /iscsi/root
iscsi_rpool/rpool/ROOT/solaris@install                64K      -   390M  -
iscsi_rpool/rpool/ROOT/solaris/var                  46.1M  9.36G  45.4M  /iscsi/root/var
iscsi_rpool/rpool/ROOT/solaris/var@install           644K      -  45.4M  -
iscsi_rpool/rpool/VARSHARE                            31K  9.36G    31K  /var/share
iscsi_rpool/rpool/export                              62K  9.36G    31K  /export
iscsi_rpool/rpool/export/home                         31K  9.36G    31K  /export/home
```

The new zone hosted on iSCSI-based shared storage resources has been successfully installed and can now be booted using `zoneadm boot`.

After the zone has been booted, the zone administrator observes virtualized ZFS datasets and storage pools from within the zone.

```
root@iscsi:~# zpool list
NAME    SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT
```

```
data   9.94G    85K  9.94G   0%  1.00x  ONLINE  -
rpool  9.94G   449M  9.50G   4%  1.00x  ONLINE  -

root@iscsi:~# zpool status -v
  pool: data
 state: ONLINE
  scan: none requested
config:

        NAME                                        STATE     READ WRITE CKSUM
        data                                        ONLINE       0     0     0
          mirror-0                                  ONLINE       0     0     0
            c0t600144F035FF8500000050C884E80002d0  ONLINE       0     0     0
            c0t600144F035FF8500000050C884EC0003d0  ONLINE       0     0     0

  pool: rpool
 state: ONLINE
  scan: none requested
config:

        NAME                                       STATE     READ WRITE CKSUM
        rpool                                      ONLINE       0     0     0
          c0t600144F035FF8500000050C884E50001d0  ONLINE       0     0     0

root@iscsi:~# zfs list -t all
NAME                             USED  AVAIL  REFER  MOUNTPOINT
data                              85K  9.78G    31K  /data
rpool                            464M  9.33G    31K  /rpool
rpool/ROOT                       464M  9.33G    31K  legacy
rpool/ROOT/solaris               464M  9.33G   416M  /
rpool/ROOT/solaris@install      1.83M      -   390M  -
rpool/ROOT/solaris/var          46.2M  9.33G  45.6M  /var
rpool/ROOT/solaris/var@install   674K      -  45.4M  -
rpool/VARSHARE                    39K  9.33G    39K  /var/share
rpool/export                    96.5K  9.33G    32K  /export
rpool/export/home               64.5K  9.33G    32K  /export/home
rpool/export/home/user          32.5K  9.33G  32.5K  /export/home/user
```

EXAMPLE 19 Example Oracle Solaris Zones Using DAS Storage Devices

This exercise uses direct attached local storage devices to configure and install a zone on Oracle Solaris. Note that this method is usually not portable across different hosts.

First, discover the available local disks with the `format` command. Then, use `suriadm lookup-uri` to construct the corresponding storage URIs to be used within the zone configuration.

```
root@host:~# format
```

```
Searching for disks...done
AVAILABLE DISK SELECTIONS:
       1. c4t1d0 <SEAGATE-ST336704LSUN36G-0326-33.92GB>
          /pci@0,0/pci1022,7450@a/pci17c2,20@4/sd@1,0
       2. c4t2d0 <FUJITSU-MAT3073NC-0104-68.49GB>
          /pci@0,0/pci1022,7450@a/pci17c2,20@4/sd@2,0
       3. c4t3d0 <SEAGATE-ST336704LSUN36G-0326-33.92GB>
          /pci@0,0/pci1022,7450@a/pci17c2,20@4/sd@3,0
       4. c4t4d0 <FUJITSU-MAW3073NC-0103-68.49GB>
          /pci@0,0/pci1022,7450@a/pci17c2,20@4/sd@4,0

root@host:~# suriadm lookup-uri -t dev /dev/dsk/c4t1d0
dev:dsk/c4t1d0
root@host:~# suriadm lookup-uri -t dev /dev/dsk/c4t2d0
dev:dsk/c4t2d0
root@host:~# suriadm lookup-uri -t dev /dev/dsk/c4t3d0
dev:dsk/c4t3d0
root@host:~# suriadm lookup-uri -t dev /dev/dsk/c4t4d0
dev:dsk/c4t4d0
```

Using those storage URIs, configure a zone with a `rootzpool` and a `zpool` resource, both representing mirrored ZFS storage pools.

```
root@host:~# zonecfg -z disk
Use 'create' to begin configuring a new zone.
zonecfg:disk> create
create: Using system default template 'SYSdefault'
zonecfg:disk> set zonepath=/disk
zonecfg:disk> add rootzpool
zonecfg:disk:rootzpool> add storage dev:dsk/c4t1d0
zonecfg:disk:rootzpool> add storage dev:dsk/c4t3d0
zonecfg:disk:rootzpool> end
zonecfg:disk> add zpool
zonecfg:disk:zpool> set name=dpool
zonecfg:disk:zpool> add storage dev:dsk/c4t2d0
zonecfg:disk:zpool> add storage dev:dsk/c4t4d0
zonecfg:disk:zpool> end
zonecfg:disk> commit
zonecfg:disk> exit
```

Now install the zone.

```
root@host:~# zoneadm -z disk install
Created zone zpool: disk_rpool
Created zone zpool: disk_dpool
Progress being logged to /var/log/zones/zoneadm.20130213T132236Z.disk.install
       Image: Preparing at /disk/root.

 AI Manifest: /tmp/manifest.xml.rOaOhe
```

```
  SC Profile: /usr/share/auto_install/sc_profiles/enable_sci.xml
    Zonename: disk
Installation: Starting ...

              Creating IPS image
Startup linked: 1/1 done
              Installing packages from:
                  solaris
                      origin:  http://pkg.oracle.com/solaris/release/
DOWNLOAD                                  PKGS         FILES    XFER (MB)   SPEED
Completed                               183/183   33556/33556  222.2/222.2  2.0M/s

PHASE                                          ITEMS
Installing new actions                   46825/46825
Updating package state database                 Done
Updating image state                            Done
Creating fast lookup database                   Done
Installation: Succeeded

        Note: Man pages can be obtained by installing pkg:/system/manual

 done.
        Done: Installation completed in 308.358 seconds.

  Next Steps: Boot the zone, then log into the zone console (zlogin -C)
              to complete the configuration process.

Log saved in non-global zone as /disk/root/var/log/zones/
zoneadm.20130213T132236Z.disk.install
root@host:~#
```

After zone installation, the following two new ZFS storage pools will be online.

```
root@host:/# zpool list
NAME         SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT
disk_dpool    68G  83.5K  68.0G   0%  1.00x  ONLINE  -
disk_rpool  33.8G   434M  33.3G   1%  1.00x  ONLINE  -

root@host:/# zpool status -v disk_rpool
  pool: disk_rpool
 state: ONLINE
  scan: none requested
config:

       NAME        STATE     READ WRITE CKSUM
       disk_rpool  ONLINE       0     0     0
         mirror-0  ONLINE       0     0     0
           c4t1d0  ONLINE       0     0     0
           c4t3d0  ONLINE       0     0     0
```

```
root@host:/# zpool status -v disk_dpool
  pool: disk_dpool
 state: ONLINE
  scan: none requested
config:

       NAME        STATE     READ WRITE CKSUM
       disk_dpool  ONLINE       0     0     0
         mirror-0  ONLINE       0     0     0
           c4t2d0  ONLINE       0     0     0
           c4t4d0  ONLINE       0     0     0
```

The zone installation will be entirely contained within this ZFS storage pool. The following ZFS dataset layout for this zone is present.

```
root@host:~# zfs list -t all|grep disk
disk_dpool                                          83.5K  66.9G    31K  /disk_dpool
disk_rpool                                           434M  32.8G    32K  /disk
disk_rpool/rpool                                     433M  32.8G    31K  /rpool
disk_rpool/rpool/ROOT                                433M  32.8G    31K  legacy
disk_rpool/rpool/ROOT/solaris                        433M  32.8G   389M  /disk/root
disk_rpool/rpool/ROOT/solaris@install                 63K      -   389M  -
disk_rpool/rpool/ROOT/solaris/var                   43.8M  32.8G  43.2M  /disk/root/var
disk_rpool/rpool/ROOT/solaris/var@install            584K      -  43.2M  -
disk_rpool/rpool/VARSHARE                             31K  32.8G    31K  /var/share
disk_rpool/rpool/export                               62K  32.8G    31K  /export
disk_rpool/rpool/export/home                          31K  32.8G    31K  /export/home
```

The new zone hosted on local device storage resources has been successfully installed and can now be booted using the `zoneadm boot` command.

After the zone has been booted, the zone administrator can observe virtualized ZFS datasets and storage pools from inside the zone.

```
root@disk:~# zpool list
NAME    SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT
dpool    68G  83.5K  68.0G   0%  1.00x  ONLINE  -
rpool  33.8G   472M  33.3G   1%  1.00x  ONLINE  -

root@disk:~# zpool status -v
  pool: dpool
 state: ONLINE
  scan: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        dpool       ONLINE       0     0     0
```

```
          mirror-0  ONLINE       0     0     0
            c4t2d0  ONLINE       0     0     0
            c4t4d0  ONLINE       0     0     0

  pool: rpool
 state: ONLINE
  scan: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        rpool       ONLINE       0     0     0
          mirror-0  ONLINE       0     0     0
            c4t1d0  ONLINE       0     0     0
            c4t3d0  ONLINE       0     0     0

root@disk:~# zfs list -t all
NAME                             USED  AVAIL  REFER  MOUNTPOINT
dpool                           83.5K  66.9G    31K  /dpool
rpool                            465M  32.8G    31K  /rpool
rpool/ROOT                       465M  32.8G    31K  legacy
rpool/ROOT/solaris               465M  32.8G   416M  /
rpool/ROOT/solaris@install      5.60M      -   389M  -
rpool/ROOT/solaris/var          43.9M  32.8G  43.3M  /var
rpool/ROOT/solaris/var@install   618K      -  43.2M  -
rpool/VARSHARE                    39K  32.8G    39K  /var/share
rpool/export                    96.5K  32.8G    32K  /export
rpool/export/home               64.5K  32.8G    32K  /export/home
rpool/export/home/user          32.5K  32.8G  32.5K  /export/home/user
```

EXAMPLE 20 Oracle Solaris Zones Using Fibre Channel-Based Storage Devices

This exercise uses a shared storage device provided over fibre channel to configure and install a zone on Oracle Solaris.

First, discover the fibre channel logical units currently visible to our host by using the `fcinfo lu` command.

```
root@host:~# fcinfo lu -v
OS Device Name: /dev/rdsk/c0t600144F0DBF8AF190000510979640005d0s2
        HBA Port WWN: 10000000c9991d8c
                Remote Port WWN: 21000024ff3ee89f
                        LUN: 5
        Vendor: SUN
        Product: ZFS Storage 7120
        Device Type: Disk Device
```

Use `suriadm lookup-uri` to construct a storage URI based on the device path. Remove the slice portion of the device name for the query to retrieve a storage URI representing an entire LU.

```
root@host:~# suriadm lookup-uri /dev/dsk/c0t600144F0DBF8AF190000510979640005d0
lu:luname.naa.600144f0dbf8af190000510979640005
lu:initiator.naa.10000000c9991d8c,target.naa.21000024ff3ee89f,luname.naa.
600144f0dbf8af190000510979640005
dev:dsk/c0t600144F0DBF8AF190000510979640005d0
```

From the three URIs displayed, select the `luname`-only form of the logical unit storage URI for use in the zone configuration.

```
root@host:~# zonecfg -z fc
Use 'create' to begin configuring a new zone.
zonecfg:fc> create
create: Using system default template 'SYSdefault'
zonecfg:fc> set zonepath=/fc
zonecfg:fc> add rootzpool
zonecfg:fc:rootzpool> add storage lu:luname.naa.600144f0dbf8af190000510979640005
zonecfg:fc:rootzpool> end
zonecfg:fc> commit
zonecfg:fc> exit
```

Install the zone.

```
root@host:~# zoneadm -z fc install
Created zone zpool: fc_rpool
Progress being logged to /var/log/zones/zoneadm.20130214T045957Z.fc.install
       Image: Preparing at /fc/root.

AI Manifest: /tmp/manifest.xml.K9aaow
SC Profile: /usr/share/auto_install/sc_profiles/enable_sci.xml
  Zonename: fc
Installation: Starting ...

            Creating IPS image
Startup linked: 1/1 done
            Installing packages from:
                solaris
                    origin:   http://pkg.oracle.com/solaris/release/
DOWNLOAD                                  PKGS         FILES    XFER (MB)   SPEED
Completed                               190/190   34246/34246  231.3/231.3  7.2M/s

PHASE                                          ITEMS
Installing new actions                   48231/48231
Updating package state database                 Done
Updating image state                            Done
Creating fast lookup database                   Done
```

```
Installation: Succeeded

        Note: Man pages can be obtained by installing pkg:/system/manual

 done.

        Done: Installation completed in 104.318 seconds.

  Next Steps: Boot the zone, then log into the zone console (zlogin -C)

              to complete the configuration process.

Log saved in non-global zone as /fc/root/var/log/zones/
zoneadm.20130214T045957Z.fc.install
root@host:~#
```

After zone installation, the following new ZFS storage pool will be online.

```
root@host:~# zpool list
NAME       SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT
fc_rpool  39.8G   441M  39.3G   1%  1.00x  ONLINE  -

root@host:~# zpool status -v fc_rpool
  pool: fc_rpool
 state: ONLINE
  scan: none requested
config:

       NAME                                      STATE     READ WRITE CKSUM
       fc_rpool                                  ONLINE       0     0     0
         c0t600144F0DBF8AF190000510979640005d0  ONLINE       0     0     0
```

The zone installation will be entirely contained within this ZFS storage pool. The zone has the following ZFS dataset layout.

```
root@host:~# zfs list -t all|grep fc
fc_rpool                                  440M  38.7G    32K  /fc
fc_rpool/rpool                            440M  38.7G    31K  /rpool
fc_rpool/rpool/ROOT                       440M  38.7G    31K  legacy
fc_rpool/rpool/ROOT/solaris               440M  38.7G   405M  /fc/root
fc_rpool/rpool/ROOT/solaris@install        67K      -   405M  -
fc_rpool/rpool/ROOT/solaris/var          34.3M  38.7G  33.6M  /fc/root/var
fc_rpool/rpool/ROOT/solaris/var@install   665K      -  33.6M  -
fc_rpool/rpool/VARSHARE                    31K  38.7G    31K  /var/share
fc_rpool/rpool/export                      62K  38.7G    31K  /export
fc_rpool/rpool/export/home                 31K  38.7G    31K  /export/home
```

The new zone hosted on shared storage provided from a fibre channel target has been successfully installed. This zone can now be booted using `zoneadm boot`.

After the zone has been booted, the zone administrator can observe virtualized ZFS datasets and storage pools from inside the zone.

```
root@fc:~# zpool list
NAME    SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT
rpool  39.8G   451M  39.3G   1%  1.00x  ONLINE  -

root@fc:~# zpool status -v
  pool: rpool
 state: ONLINE
  scan: none requested
config:

        NAME                                        STATE     READ WRITE CKSUM
        rpool                                       ONLINE       0     0     0
          c0t600144F0DBF8AF190000510979640005d0  ONLINE       0     0     0

root@fc:~# zfs list -t all
NAME                             USED  AVAIL  REFER  MOUNTPOINT
rpool                            467M  38.7G    31K  /rpool
rpool/ROOT                       467M  38.7G    31K  legacy
rpool/ROOT/solaris               467M  38.7G   430M  /
rpool/ROOT/solaris@install      1.90M      -   405M  -
rpool/ROOT/solaris/var          34.4M  38.7G  33.7M  /var
rpool/ROOT/solaris/var@install   703K      -  33.6M  -
rpool/VARSHARE                    39K  38.7G    39K  /var/share
rpool/export                    96.5K  38.7G    32K  /export
rpool/export/home               64.5K  38.7G    32K  /export/home
rpool/export/home/user          32.5K  38.7G  32.5K  /export/home/user
```

Migrating Oracle Solaris Zones Hosted on Shared Storage Resources

Migration of zones hosted on shared storage using storage URIs in the zone configuration is a straightforward and simple process. The only commands required for this process are the `zoneadm` and `zonecfg` commands. No additional commands must be run to migrate zones with storage resources or ZFS storage pools.

EXAMPLE 21 Migrating a Zone Based on iSCSI Shared Storage

This example shows the steps to migrate the zone based on iSCSI shared storage from its current host to a new host.

```
root@initiator:/# zoneadm list -cp
0:global:running:/::solaris:shared:-:none
-:iscsi:installed:/iscsi:a0a4ba0d-9d6d-cf2c-cc42-f123a5e3ee11:solaris:excl:-:
```

The first step is to detach the zone on the current host. The zone will move from the installed state into the configured state. All the ZFS storage pools will be exported and the shared storage resources will be unconfigured automatically.

```
root@initiator:/# zoneadm -z iscsi detach
Exported zone zpool: iscsi_rpool
Unconfigured zone storage resource(s) from:
        iscsi://target/luname.naa.600144F035FF8500000050C884E50001
Exported zone zpool: iscsi_data
Unconfigured zone storage resource(s) from:
        iscsi://target/luname.naa.600144F035FF8500000050C884E80002
        iscsi://target/luname.naa.600144F035FF8500000050C884EC0003

root@initiator:/# zoneadm list -cp
0:global:running:/::solaris:shared:-:none
-:iscsi:configured:/iscsi::solaris:excl:-:
```

The last step required on the current host is to export the zone configuration with `zonecfg export`, and transfer the resulting file onto the new host.

```
root@initiator:/# zonecfg -z iscsi export -f /export/iscsi.cfg
```

On the new host, instantiate the zone configuration first from the saved file using `zonecfg`. The zone will be in the configured zone state.

```
root@newhost:/# zonecfg -z iscsi -f /export/iscsi.cfg

root@newhost:/# zoneadm list -cp
0:global:running:/::solaris:shared:-:none
-:iscsi:configured:/iscsi::solaris:excl:-:
```

Then, attach the zone on the new host with `zoneadm attach`. The zone will move into the installed state. All shared storage resources will be configured and the corresponding ZFS storage pools will be imported automatically.

```
root@newhost:/# zoneadm -z iscsi attach
Configured zone storage resource(s) from:
       iscsi://target/luname.naa.600144F035FF8500000050C884E50001
Imported zone zpool: iscsi_rpool
Configured zone storage resource(s) from:
       iscsi://target/luname.naa.600144F035FF8500000050C884E80002
       iscsi://target/luname.naa.600144F035FF8500000050C884EC0003
Imported zone zpool: iscsi_data
Progress being logged to /var/log/zones/zoneadm.20130214T145001Z.iscsi.attach
```

```
    Installing: Using existing zone boot environment
      Zone BE root dataset: iscsi_rpool/rpool/ROOT/solaris
                     Cache: Using /var/pkg/publisher.
  Updating non-global zone: Linking to image /.
Processing linked: 1/1 done
  Updating non-global zone: Auditing packages.
No updates necessary for this image.

  Updating non-global zone: Zone updated.
                    Result: Attach Succeeded.
Log saved in non-global zone as /iscsi/root/var/log/zones/
zoneadm.20130214T145001Z.iscsi.attach

root@newhost:/# zoneadm list -cp
0:global:running:/::solaris:shared:-:none
-:iscsi:installed:/iscsi:a19fbb45-4af3-670f-c58e-ee48757c75d6:solaris:excl:-:
```

The zone has been migrated to the new host and is now ready to be booted with `zoneadm boot`.

This process is basically the same for the three different types of storage URIs supported in Oracle Solaris. For iSCSI-based or fibre channel-based storage resources, you must also confirm that the new host has access to the same logical unit and target ports.

Moving Existing Zones In and Out of Shared Storage Zone Configurations

In Oracle Solaris, it is possible to convert existing zone installations into shared storage-based zone configurations. It is also possible to convert an installed zone using shared storage resources into a traditional configuration. Note that there are manual steps involved in the process.

▼ How To Move an Existing Zone Into a Shared Storage Configuration

The following steps are required to convert an existing, installed zone into a shared storage zone configuration with a `rootzpool` resource.

1. **Become a zone administrator.**

You must also be assigned the Unified Archive Administration rights profile. The `root` role has all of these rights.

For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Shut down the zone with `zoneadm shutdown` command.**

3. **Create an archive of the installed zone.**

   Refer to the `solaris(5)` man page for additional information.

4. **Uninstall the zone with `zoneadm uninstall`.**

   This step removes the current zone installation from the system and move the zone back into the configured zone state.

5. **Add a `rootzpool` resource and corresponding shared storage resources to the zone configuration with the `zonecfg` command.**

6. **Install the zone again from the archive by using `zoneadm install -a`.**

   Refer to the `solaris(5)` man page for more details about this option.

   During the archive-based installation, the zones framework will configure the shared storage resources and create or import the desired ZFS storage pools.

7. **The zone can now be booted again with `zoneadm boot`.**

How To Move an Existing Zone out of a Shared Storage Configuration

To move an installed zone out of its shared storage configuration, perform the following steps.

1. **Become a zone administrator.**

   You must also be assigned the ZFS Storage Management rights profile. The `root` role has all of these rights.

   For more information, see "Assigning Limited Rights to Zone Administrators" on page 159.

2. **Shut down the zone with `zoneadm shutdown`.**

3. **If you are removing a `rootzpool` resource from a zone, create an archive of the installed zone.**

Refer to the solaris(5) man page for additional information.

4. **Detach the zone using `zoneadm detach`.**

   The zone will move into the configured state. During zone detach, the ZFS storage pools will be exported and corresponding storage resources will be unconfigured.

5. **Remove the desired `rootzpool` and `zpool` resources from the Zone configuration with `zonecfg`.**

6. **Perform one of the following steps depending on the resource that has been removed.**

   - If a `rootzpool` resource has been removed, re-install the zone from the archive using `zoneadm install -a`.
   - When removing a `zpool` resource only, just re-attach the zone with `zoneadm attach`. This step moves the zone back into the installed state again.

   The archive based installation process will create a new, local zonepath containing the zone installation.

7. **Manually destroy the ZFS storage pools with `zpool` if necessary.**

How To Add Additional ZFS Storage Pools to an Installed Zone

It is possible to add additional ZFS storage pools based on shared storage resources to an installed zone. To assign an already existing ZFS storage pool to a zone, perform the following manual steps. Note that the existing ZFS storage pool must be exported first with the `zpool export` command.

This procedure can also be used to migrate ZFS storage pools from one installed zone to another. In addition to these steps, the source zone must be detached first using `zoneadm detach` to export the ZFS storage pool properly, and its zone configuration must be altered to remove the corresponding zpool resource with `zonecfg`.

1. **Become a zone administrator.**

   For more information, see “Assigning Limited Rights to Zone Administrators” on page 159.

2. **Shut down the zone with `zoneadm shutdown`.**

3. **Detach the installed zone using `zoneadm detach`.**

   The zone will move into the configured zone state.

4. **Add a new zpool resource and corresponding shared storage resources to the zone configuration with `zonecfg`.**

5. **Re-attach the zone using `zoneadm attach`.**

   The zone will move back into the installed state.

   During attach, the zones framework will configure the shared storage resources, import the existing ZFS storage pool, and assign it to the installed zone.

6. **The zone can now be rebooted with `zoneadm boot`.**

Zones References for Shared Storage Resources

For more information, refer to the following sources of information.

Oracle Solaris Zones Man Page References

The following man pages provide information about the commands to use to configure Oracle Solaris Zones on shared resources.

- `beadm`(1M)
- `fcinfo`(1M)
- `iscsiadm`(1M)
- `iser`(7D)
- `itadm`(1M)
- `sasinfo`(1M)
- `solaris`(5)
- `stmfadm`(1M)
- `suri`(5)
- `suriadm`(1M)
- `zoneadm`(1M)
- `zonecfg`(1M)
- `zones`(5)

- `zpool`(1M)

Oracle Solaris Administration Guides

Oracle Solaris Zones Configuration Resources in the Oracle Solaris Documentation Library provides additional information about configuring zones on shared resources.

Zones Glossary

brand An instance of the BrandZ functionality, which provides non-global zones that contain non-native operating environments used for running applications.

branded zone An isolated environment in which to run non-native applications in non-global zones.

cap A limit that is placed on system resource usage.

capping The process of placing a limit on system resource usage.

datalink An interface at Layer 2 of the OSI protocol stack, which is represented in a system as a STREAMS DLPI (v2) interface. This interface can be plumbed under protocol stacks such as TCP/IP. In the context of Oracle Solaris 10 zones, datalinks are physical interfaces, aggregations, or VLAN-tagged interfaces . A datalink can also be referred to as a physical interface, for example, when referring to a NIC or a VNIC.

default pool The pool created by the system when pools are enabled.

See also resource pool.

default processor set The processor set created by the system when pools are enabled.

See also processor set.

dynamic configuration Information about the disposition of resources within the resource pools framework for a given system at a point in time.

dynamic reconfiguration On SPARC based systems, the ability to reconfigure hardware while the system is running. Also known as DR.

extended accounting A flexible way to record resource consumption on a task basis or process basis in the Solaris operating system.

fair share scheduler A scheduling class, also known as FSS, that allows you to allocate CPU time that is based on shares. Shares define the portion of the system's CPU resources allocated to a project.

FSS See fair share scheduler.

global administrator The root user or an administrator with the `root` role. When logged in to the global zone, the global administrator or a user granted the appropriate authorizations can monitor and control the system as a whole.

See also zone administrator.

global scope Actions that apply to resource control values for every resource control on the system.

global zone The zone contained on every Oracle Solaris system. When non-global zones are in use, the global zone is both the default zone for the system and the zone used for system-wide administrative control.

See also non-global zone.

immutable zone A zone configured with a read-only root.

local scope Local actions taken on a process that attempts to exceed the control value.

locked memory Memory that cannot be paged.

memory cap enforcement threshold The percentage of physical memory utilization on the system that will trigger cap enforcement by the resource capping daemon.

naming service database In the Projects and Tasks Overview chapter of this document, a reference to both LDAP containers and NIS maps.

native zone A `solaris` branded non-global zone. Zones of other brands, such as the `labeled` and `solaris10`, or `solaris-kz` brands are not native zones.

The operating system level within a running native zone always matches the level of the global zone. Because the same kernel is used, no system-call translation is performed.

non-global zone A virtualized operating system environment created within a single instance of the Oracle Solaris operating system. The Oracle Solaris Zones software partitioning technology is used to virtualize operating system services.

non-global zone administrator See zone administrator.

Oracle Solaris Zones A software partitioning technology used to virtualize operating system services and provide an isolated, secure environment in which to run applications.

pool See resource pool.

pool daemon The `poold` system daemon that is active when dynamic resource allocation is required.

processor set A disjoint grouping of CPUs. Each processor set can contain zero or more processors. A processor set is represented in the resource pools configuration as a resource element. Also referred to as a pset.

project A network-wide administrative identifier for related work.

resident set size The size of the resident set. The resident set is the set of pages that are resident in physical memory.

resource An aspect of the computing system that can be manipulated with the intent to change application behavior.

resource capping daemon A daemon that regulates the consumption of physical memory by processes running in projects that have resource caps defined.

resource consumer Fundamentally, an Oracle Solaris process. Process model entities such as the project and the task provide ways of discussing resource consumption in terms of aggregated resource consumption.

resource control A per-process, per-task, or per-project limit on the consumption of a resource.

resource management A functionality that enables you to control how applications use available system resources.

resource partition An exclusive subset of a resource. All of the partitions of a resource sum to represent the total amount of the resource available in a single executing Oracle Solaris instance.

resource pool A configuration mechanism that is used to partition machine resources. A resource pool represents an association between groups of resources that can be partitioned.

resource set A process-bindable resource. Most often used to refer to the objects constructed by a kernel subsystem offering some form of partitioning. Examples of resource sets include scheduling classes and processor sets.

RSS See resident set size.

scanner A kernel thread that identifies infrequently used pages. During low memory conditions, the scanner reclaims pages that have not been recently used.

static pools configuration A representation of the way in which an administrator would like a system to be configured with respect to resource pools functionality.

task In resource management, a process collective that represents a set of work over time. Each task is associated with one project.

whole root zone A type of non-global zone in which all of the required system software and any additional packages are installed into the private file systems of the zone.

working set size The size of the working set. The working set is the set of pages that the project workload actively uses during its processing cycle.

workload An aggregation of all processes of an application or group of applications.

zone administrator The rights of a zone administrator are confined to a non-global zone.

See also global administrator.

zone state The status of a non-global zone. The zone state is one of configured, incomplete, installed, ready, unavailable, running, or shutting down.

Index

M

N

O

P

R

S

T

U

V

Z