

Creating and Using Oracle Solaris Kernel Zones



Part No: E37629-03
December 2014

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS. Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Copyright © 2014, Oracle et/ou ses affiliés. Tous droits réservés.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf disposition de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, breveter, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délivre la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique:

U.S. GOVERNMENT END USERS. Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer des dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour ce type d'applications.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. UNIX est une marque déposée d'The Open Group.

Ce logiciel ou matériel et la documentation qui l'accompagne peuvent fournir des informations ou des liens donnant accès à des contenus, des produits et des services émanant de tiers. Oracle Corporation et ses affiliés déclinent toute responsabilité ou garantie expresse quant aux contenus, produits ou services émanant de tiers. En aucun cas, Oracle Corporation et ses affiliés ne sauraient être tenus pour responsables des pertes subies, des coûts occasionnés ou des dommages causés par l'accès à des contenus, produits ou services tiers, ou à leur utilisation.

Contents

Using This Documentation	5
1 Planning and Configuring Oracle Solaris Kernel Zones	7
About Oracle Solaris Kernel Zones	7
Kernel Zones and General Zones Concepts	8
Hardware and Software Requirements for Oracle Solaris Kernel Zones	8
Verifying Hardware and Software Support on Kernel Zone Hosts	9
Tuning the Host ZFS ARC Cache on a Kernel Zone Host	11
Configuring the Oracle Solaris Kernel Zone	11
▼ How to Configure a Kernel Zone	11
Configuring and Customizing Kernel Zone Resources	12
Managing Kernel Zone CPUs	13
Managing Kernel Zone Memory	14
Managing Kernel Zone Storage Devices and the <code>bootpri</code> Resource Property	15
Managing Kernel Zone Network Devices and Configuration	17
2 Installing, Shutting Down, Cloning, and Migrating Oracle Solaris Kernel Zones	19
Installing a Kernel Zone	19
Installing a Kernel Zone by Using Direct Installation	20
Using AI Manifests and <code>sysconfig</code> Profiles in Kernel Zone Installations	22
Installing a Kernel Zone from Installation Media	25
Uninstalling a Kernel Zone	26
Shutting Down, Rebooting, and Halting a Kernel Zone	26
Cloning a Kernel Zone	27
Suspending and Resuming a Kernel Zone	28
Migrating a Kernel Zone by Using Suspend and Resume	29
▼ How to Migrate a Kernel Zone	30
3 Administering Oracle Solaris Kernel Zones	33

Working in the Kernel Zone Environment	33
Process ID Visibility in Zones	33
Duplicate Process IDs in Kernel Zones	34
Kernel Zone Zonepath	34
Resource Management Functionality in Kernel Zones	34
Working With Kernel Zones and Immutable Zones	34
Configuring Immutable Kernel Zones	34
▼ How to Configure a Kernel Zone as an Immutable (Read-Only) Zone	35
Maintaining and Administering Read-Only Kernel Zones	35
Managing Removable Devices on the Kernel Zone	36
▼ How to Add a Virtual CD-ROM Device to a Kernel Zone	36
Working With Kernel Zone Auxiliary States	38
Managing Nested Zones	39
▼ How to Add Multiple MAC Addresses to a Kernel Zone	40
Nested Zones and New Non-Global Zone Configuration	41
Kernel Zone Host Data and Host ID	41
Working With the Kernel Zone Boot Loader	42
▼ How to Specify Alternate Boot Environments in a Kernel Zone	43
Index	47

Using This Documentation

- **Overview** – Describes how to plan, configure, install, and administer Oracle Solaris Kernel Zones
- **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

Late-breaking information and known issues for this product are included in the documentation library at <http://www.oracle.com/pls/topic/lookup?ctx=E36784>.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Feedback

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

Planning and Configuring Oracle Solaris Kernel Zones

This chapter discusses how to plan and configure Oracle Solaris Kernel Zones, also known as `solaris-kz` branded zones. It provides procedures for verifying hardware support, zone configuration, and applying zone resources specific to kernel zones.

This chapter covers the following planning and configuration topics:

- [“About Oracle Solaris Kernel Zones” on page 7](#)
- [“Hardware and Software Requirements for Oracle Solaris Kernel Zones” on page 8](#)
- [“Configuring the Oracle Solaris Kernel Zone” on page 11](#)
- [“Configuring and Customizing Kernel Zone Resources” on page 12](#)

About Oracle Solaris Kernel Zones

An Oracle Solaris Kernel Zone, also called a `solaris-kz` branded zone, uses the branded zones framework to run a zone with a separate kernel and operating system (OS) installation from the global zone. The separate kernel and OS installation provide for greater independence and enhanced security of operating system instances and applications.

The administrative and structural content of a kernel zone is entirely independent from that of the global zone. For example, a kernel zone does not share system packaging with the global zone, or kernel zone host. Package updates on the kernel zone host are not linked images and do not affect kernel zones. Similarly, packaging commands such as `pkg update` are fully functional from inside of a kernel zone. See [Chapter 3, “Installing, Removing, and Updating Software Packages,”](#) in [“Packaging and Delivering Software With the Image Packaging System in Oracle Solaris 11.2”](#) for additional information on packaging commands.

System processes are handled in the kernel zone's separate process ID table and are not shared with the global zone. Resource management in kernel zones is also different. Resource controls such as `max-processes` are not available when configuring a kernel zone.

Use the existing `zlogin`, `zonecfg`, and `zoneadm` commands to manage and to administer kernel zones on the global zone.

For more information about the branded zones framework, see the [brands\(5\)](#) man page.

See [Chapter 1, “Oracle Solaris Zones Introduction,”](#) in [“Introduction to Oracle Solaris Zones ”](#) for additional overview in regard to kernel zones concepts.



Caution - An Oracle Solaris x86 system might panic if Oracle VM VirtualBox and Oracle Solaris Kernel Zones are run at the same time on the system.

Kernel Zones and General Zones Concepts

This book assumes that you are familiar with the following resource management and zones concepts:

- Resource controls that determine how applications use available system resources
- Commands used to configure, install, and administer zones, primarily `zonecfg`, `zoneadm`, and `zlogin`
- `zonecfg` resources and property types
- Global zones and non-global zones
- The whole-root non-global zone model
- Authorizations granted through the `zonecfg` utility
- Global administrator and zone administrator
- The zone state model
- Zone isolation characteristics
- Network concepts and configuration
- Zone shared-IP and exclusive-IP types

See [“Introduction to Oracle Solaris Zones ”](#) and [“Creating and Using Oracle Solaris Zones ”](#) for more information about these concepts.

Hardware and Software Requirements for Oracle Solaris Kernel Zones

To use kernel zones, you must be running Oracle Solaris 11.2 on your host operating system.

The physical host system must have the following components:

- SPARC systems:
 - A SPARC T4 system with at least System Firmware 8.5.1.
 - A SPARC T5, SPARC M5, or SPARC M6 system with at least System Firmware 9.2.1.
 - A Fujitsu M10/SPARC M10 system with at least XCP Firmware 2230.
- You can download the latest system firmware by logging in to [My Oracle Support](#) and clicking the Patches & Updates link. Use Search to locate the firmware download for your system type.
- x86 systems:
 - Nehalem+ (Intel) or Barcelona+ (AMD) based x86 machines with CPU virtualization (for example, VT-x) enabled in the BIOS. Specifically, kernel zones on x86 require Extended/Nested Page Table support, also referred to as EPT, NPT, or RVI (Rapid Virtualization Indexing).

In addition, both SPARC and x86 systems require the following:

- A minimum of 8 GB of physical RAM
- The kernel zone brand package software package, `brand/brand-solaris-kz`.
For information on obtaining and installing software packages, see [Chapter 3, “Installing, Removing, and Updating Software Packages,”](#) in [“Packaging and Delivering Software With the Image Packaging System in Oracle Solaris 11.2”](#).
- Sufficient tuning of the ZFS Adaptive Replacement Cache (ARC) on the host to prevent memory errors. See [“Tuning the Host ZFS ARC Cache on a Kernel Zone Host”](#) on [page 11](#).

Oracle Solaris Kernel Zones can run in guests on Oracle VM Server for SPARC. Each Oracle VM Server for SPARC domain has an independent limit for the number of kernel zones that you can run. The limit is 768 for SPARC T4 or SPARC T5 systems, and 512 for SPARC M5 or SPARC M6 systems.

Kernel zones cannot run in Oracle VM Server for x86 guests or on Oracle VM VirtualBox.

Verifying Hardware and Software Support on Kernel Zone Hosts

Before planning and deploying a kernel zone, you must verify that the kernel zone host has the hardware and software requirements as described in [“Hardware and Software Requirements for Oracle Solaris Kernel Zones”](#) on [page 8](#). You can use the `virtinfo` command to verify the hardware requirements, firmware or BIOS requirements, and kernel zone brand package software requirements on the kernel zone host.

▼ How to Verify Kernel Zone Support on a Host

1. On the kernel zone host, become an administrator.

For more information, see [“Using Your Assigned Administrative Rights”](#) in [“Securing Users and Processes in Oracle Solaris 11.2”](#).

2. Verify that the Oracle Solaris operating system version is at least 11.2.

```
# uname -a
```

For example, on the host `global`:

```
global# uname -a
SunOS global 5.11 11.2 sun4v sparcsun4v
```

3. Verify the installation of the kernel zone brand package, `brand/brand-solaris-kz`.

```
# pkg list brand/brand-solaris-kz
```

The following example shows that the kernel zone brand package is installed on the host `global`.

```
global# pkg list brand/brand-solaris-kz
NAME (PUBLISHER)          VERSION          IFO
system/zones/brand/brand-solaris-kz 0.5.11-0.175.2.0.0.36.22321  i--
```

4. Run the `virtinfo` command.

```
# virtinfo
```

The following example output shows that kernel zones are supported on the host `global`.

```
global# virtinfo
NAME          CLASS
logical-domain current
non-global-zone supported
kernel-zone   supported
```

See Also For further information, see the [`virtinfo\(1M\)`](#) man page.

Tuning the Host ZFS ARC Cache on a Kernel Zone Host

To ensure efficient performance, you must configure the kernel zone host to limit the system ZFS Adaptive Replacement Cache (ARC). This value needs to be set only once on the host when you are planning your kernel zone configuration.



Caution - Failure to limit the host ZFS ARC cache can lead to low memory failures.

To limit the ZFS ARC cache on the host, as an administrator, set the `zfs_arc_max` value to the ZFS ARC cache value in bytes. The suggested value is one-half of what you would like the host ZFS resources to use. For example, if you want ZFS to use less than 2 GB of memory, set the ARC cache to 1 GB, or `0x40000000`. See [“ZFS ARC Parameters”](#) in [“Oracle Solaris 11.2 Tunable Parameters Reference Manual ”](#) and Chapter 11, [“Recommended Oracle Solaris ZFS Practices,”](#) in [“Managing ZFS File Systems in Oracle Solaris 11.2 ”](#) for further information.

You must reboot the kernel zone host to have the changes take effect.

Configuring the Oracle Solaris Kernel Zone

This section describes how to configure an Oracle Solaris kernel zone.

▼ How to Configure a Kernel Zone

This procedure describes how to configure a kernel zone using the kernel zone template, `SYSsolaris-kz`. For an overview of zone template properties, see [“zonecfg template Property”](#) in [“Introduction to Oracle Solaris Zones ”](#). For general information regarding zone configuration, see Chapter 1, [“How to Plan and Configure Non-Global Zones,”](#) in [“Creating and Using Oracle Solaris Zones ”](#).

Before You Begin Before you begin to configure a kernel zone, you must confirm kernel zone hardware support, software support, and memory configuration on your host system. See [“Verifying Hardware and Software Support on Kernel Zone Hosts”](#) on page 9 and [“Tuning the Host ZFS ARC Cache on a Kernel Zone Host”](#) on page 11.

1. **Become an administrator.**

For more information, see [“Using Your Assigned Administrative Rights”](#) in [“Securing Users and Processes in Oracle Solaris 11.2”](#).

2. Create a new kernel zone configuration.

The solaris-kz branded zone template is SYSsolaris-kz. For example, on the host global, to create a new kernel zone configuration for the kernel zone, kzone1:

```
global# zonecfg -z kzone1
Use 'create' to begin configuring a new zone.
zonecfg:kzone1> create -t SYSsolaris-kz
```

The remaining configuration steps in this procedure use the kernel zone kzone1.

3. Add any additional kernel zone resources.

You can set some kernel zone resources now or after the zone is configured. For more information, see [“Configuring and Customizing Kernel Zone Resources”](#) on page 12.

4. Commit the zone configuration.

```
zonecfg:kzone1> commit
```

5. Exit zonecfg.

```
zonecfg:kzone1> exit
```

6. (Optional) Verify the zone configuration.

You can verify a zone prior to installation. If you skip this step, the verification is performed automatically when you install the zone. See [“\(Optional\) How to Verify a Configured Zone Before It Is Installed”](#) in [“Creating and Using Oracle Solaris Zones”](#).

```
# zoneadm -z zonename verify
```

For example, to verify the kernel zone kzone1 on the host global:

```
global# zoneadm -z kzone1 verify
```

If you see an error message 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.

Configuring and Customizing Kernel Zone Resources

Zone resources are mechanisms for managing machine, system, and CPU resources. Resources are set when planning a zone configuration. Note that some resources on the kernel zone differ

from what is available in `solaris` and `solaris10` zones. For example, there is no support for the `max-processes`, `fs-allowed`, and `ip-type` resources in `solaris-kz` zones.

This section describes how to configure resources to add additional support for the following components:

- Kernel zone CPUs. See [“Managing Kernel Zone CPUs” on page 13](#).
- Kernel zone memory. See [“Managing Kernel Zone Memory” on page 14](#).
- Kernel zone storage devices. See [“Managing Kernel Zone Storage Devices and the `bootpri` Resource Property” on page 15](#).
- Kernel zone network devices and network configuration. See [“Managing Kernel Zone Network Devices and Configuration” on page 17](#).

You use the `zonecfg` command on the global zone to set or to modify kernel zone resources.

Note - You must be the global administrator or a user with appropriate authorization in the global zone to use the `zonecfg` command.

See [“About Resources in Zones” in “Introduction to Oracle Solaris Zones ”](#) and the `solaris-kz(5)` man page for additional information about kernel zone resources.

Managing Kernel Zone CPUs

By default, a kernel zone is given one virtual CPU upon creation. You can alter the number of virtual CPUs by adding and modifying the `virtual-cpu` resource.

Use the `dedicated-cpu zonecfg` resource property to dedicate a host CPU to the kernel zone.

Note that if you have already defined the `dedicated-cpu zonecfg` resource, the default number of virtual CPUs configured in the virtual platform matches the lower value of the `ncpus` range inside the `dedicated-cpu zonecfg` resource. If both resources exist, they are cross-checked for consistency. See the `zonecfg(1M)` man page for further information.

See [Chapter 1, “How to Plan and Configure Non-Global Zones,” in “Creating and Using Oracle Solaris Zones ”](#) for general information on how to set the `virtual-cpu` and `dedicated-cpu` zone resources.

EXAMPLE 1-1 Adding Additional Virtual CPUs to a Kernel Zone

This example shows how to add additional virtual CPUs to the kernel zone `kzone1`.

```
global# zonecfg -z kzone1
```

```
zonecfg:kzone1> info virtual-cpu
zonecfg:kzone1> add virtual-cpu
zonecfg:kzone1:virtual-cpu> set ncpus=8
zonecfg:kzone1:virtual-cpu> end
zonecfg:kzone1> info virtual-cpu
virtual-cpu:
    ncpus: 8
zonecfg:kzone1> exit
```

EXAMPLE 1-2 Adding a Dedicated CPU to a Kernel Zone

This example shows how to add a dedicated CPU to the kernel zone `kzone1`.

```
global# zonecfg -z kzone1
zonecfg:kzone1> info dedicated-cpu
zonecfg:kzone1> add dedicated-cpu
zonecfg:kzone1:dedicated-cpu> set ncpus=8
zonecfg:kzone1:dedicated-cpu> end
zonecfg:kzone1> info dedicated-cpu
    ncpus: 8
zonecfg:kzone1> exit

zonecfg:kzone1> info dedicated-cpu
dedicated-cpu:
    ncpus: 8
```

Managing Kernel Zone Memory

You must allocate a fixed amount of physical RAM to the kernel zone virtual platform. You can define this amount by setting the kernel zone `capped-memory` resource.

The physical memory assigned to a kernel zone is allocated in its entirety when it is configured. The memory allocated is only for the exclusive use of the kernel zone. For example, once a kernel zone is booted, all of the memory as specified in the `capped-memory` resource appears to be in use to the host operating system.

On an x86 system, the `capped-memory` resource must be set in increments of 2 megabytes (MB).

On a SPARC system, the `capped-memory` resource must be set in increments of 256 megabytes (MB).

The zone allocates the `capped-memory` resource when the zone boots. This amount remains fixed while the zone is running.

See [Chapter 1, “How to Plan and Configure Non-Global Zones,”](#) in [“Creating and Using Oracle Solaris Zones”](#) for general information on how to set the `capped-memory` zone resource.

If kernel zone memory size is increased prior to installation, you must also increase the kernel zone root disk size to account for the larger swap and dump devices. If a kernel zone does not have a disk explicitly added, a zvol is created and used as the root disk. By default, the zvol is 16GB in size. If a different root disk size is required, use the `zoneadm install -x install-size` command to modify the disk size. For example, to specify a 32GB root disk size on the kernel zone, `kzone1`:

```
global# zoneadm -z kzone1 install -x install-size=32G
```

For additional information on setting the capped-memory zone resource, see [Chapter 1, “How to Plan and Configure Non-Global Zones,”](#) in [“Creating and Using Oracle Solaris Zones”](#). For information on modifying the disk size using the `zoneadm` command, see the [`zoneadm\(1M\)`](#) man page.

EXAMPLE 1-3 Setting the capped-memory Resource on a SPARC System

This example shows how to set the capped-memory resource on a SPARC system.

```
global# zonecfg -z kzone1
zonecfg:kzone1> select capped-memory
zonecfg:kzone1:capped-memory> set physical=2048m
zonecfg:kzone1:capped-memory> end
zonecfg:kzone1> exit
```

EXAMPLE 1-4 Setting the capped-memory Resource on an x86 System

This example shows how to set the capped-memory resource on an x86 system.

```
global# zonecfg -z kzone1
global# zonecfg -z kzone1
zonecfg:kzone1> select capped-memory

zonecfg:kzone1:capped-memory> set physical=16g
zonecfg:kzone1:capped-memory> end
zonecfg:kzone1> exit
```

Managing Kernel Zone Storage Devices and the `bootpri` Resource Property

A kernel zone root is always accessible. You can add additional storage devices to a kernel zone by using the `add device` resource. Additional kernel zone storage devices have the following requirements:

- The full storage device path (for example, `/dev/rdisk/c9t0d0`) must be specified.

- The storage device must be defined by only one of the following:
 - The `add device match` resource property. If you specify a storage device for the `add device match` resource property, you must specify a device that is present in `/dev/rdisk`, `/dev/zvol/rdisk`, or `/dev/did/rdisk`.
 - A valid storage URI.
- The storage device must be a whole disk or LUN.

Use the `bootpri` resource property to specify the boot order of each storage device. The `bootpri` resource property must be set to any positive integer value.



Caution - The `bootpri` resource property must be set only if the device is to be used as a boot device. If the `bootpri` resource property is set on devices other than boot devices, data corruption might result.

To unset the `bootpri` resource property, use the `clear bootpri` resource property.

If multiple bootable devices are present during installation, the devices will be used for a mirrored ZFS pool in the zone.

The default boot order of each device is determined by sorting devices first by `bootpri`, then by `id` if multiple devices have the same `bootpri`.

EXAMPLE 1-5 Adding Additional Storage Devices to a Kernel Zone

This example shows how to add the additional storage device `/dev/rsk/c9t0d0` to the kernel zone `kzone1`.

```
global# zonecfg -z kzone1
zonecfg:kzone1> add device
zonecfg:kzone1:device> set match=/dev/rdisk/c9t0d0
zonecfg:kzone1:device> set bootpri=4
zonecfg:kzone1:device> end
```

EXAMPLE 1-6 Changing the Kernel Zone Default Boot Device to Use a Storage URI:

This example shows how to change the default boot device on the kernel zone `kzone1` to use a storage URI located at `iscsi://zfssa/luname.naa.600144F0DBF8AF19000052E820D60003`.

```
global# zonecfg -z kzone1
zonecfg:kzone1> select device id=0
zonecfg:kzone1:device> set storage=iscsi://zfssa/
luname.naa.600144F0DBF8AF19000053482CC00029
zonecfg:kzone1:device> end
zonecfg:kzone1> info device
device:
```



```

match not specified
storage: iscsi://zfssa/luname.naa.600144F0DBF8AF19000052E820D60003
id: 0
bootpri: 0

```

Managing Kernel Zone Network Devices and Configuration

Kernel zones provide network access in kernel zones by adding `net` or `anet` resources. See [“Zone Network Interfaces”](#) in [“Introduction to Oracle Solaris Zones”](#) for further information about these two resource types.

Exclusive-IP zones must be used for kernel zones. See [“Exclusive-IP Zone Network Address”](#) in [“Creating and Using Oracle Solaris Zones”](#) for more information about exclusive-IP zones.

You can supply additional MAC addresses to support nested zones, or zones where a kernel zone hosts non-global `solaris` and `solaris10` branded zones. See [“Managing Nested Zones”](#) on page 39 for more information about nested zones.

You can optionally specify a network device ID to identify the VNIC address from inside the zone and determine the order in which the network interfaces are presented to the kernel zone. This process is similar to moving a NIC from one physical slot to another.

See [Chapter 1, “How to Plan and Configure Non-Global Zones,”](#) in [“Creating and Using Oracle Solaris Zones”](#) for general information on how to set network zone resources.

EXAMPLE 1-7 Adding Additional Network Resources to a Kernel Zone

This example shows how to add a network resource to the kernel zone `kzone1`. The ID is set to 3 to determine the order in which the new `anet` interface is presented to the kernel zone.

```

global# zonecfg -z kzone1
zonecfg:kzone1> add anet
zonecfg:kzone1:anet> set id=3
global# zonecfg:kzone1:anet> end
zonecfg:kzone1> exit

```

EXAMPLE 1-8 Removing Network Devices From a Kernel Zone

This example shows how to remove a network device from the kernel zone `kzone1`. The information on the existing `anet` resources is listed and the `anet` device with the value of 1 is deleted.

```

global# zonecfg -z kzone1 info anet
anet:

```

```
lower-link: auto
allowed-address not specified
allowed-dhcp-cids not specified
link-protection: mac-nospoof
mac-address: random
mac-prefix not specified
mac-slot not specified
vlan-id not specified
priority not specified
rxrings not specified
txrings not specified
mtu not specified
maxbw not specified
rxfanout not specified
vsi-typeid not specified
vsi-vers not specified
vsi-mgrid not specified
etsbw-lcl not specified
cos not specified
id: 0
anet:
lower-link: auto
allowed-address not specified
allowed-dhcp-cids not specified
link-protection: mac-nospoof
mac-address: default
mac-prefix not specified
mac-slot not specified
vlan-id not specified
priority not specified
rxrings not specified
txrings not specified
mtu not specified
maxbw not specified
rxfanout not specified
vsi-typeid not specified
vsi-vers not specified
vsi-mgrid not specified
etsbw-lcl not specified
cos not specified
id: 1
global# zonecfg -z kzone1 remove anet id=1
```

Installing, Shutting Down, Cloning, and Migrating Oracle Solaris Kernel Zones

This chapter describes different methods for installing a kernel zone and describes how to halt, shut down, restart, and clone a kernel zone. This chapter discusses the following topics:

- [“Installing a Kernel Zone” on page 19](#)
- [“Uninstalling a Kernel Zone” on page 26](#)
- [“Shutting Down, Rebooting, and Halting a Kernel Zone” on page 26](#)
- [“Cloning a Kernel Zone” on page 27](#)
- [“Suspending and Resuming a Kernel Zone” on page 28](#)
- [“Migrating a Kernel Zone by Using Suspend and Resume” on page 29](#)

For general information about zone installation and zone cloning concepts, see [“Introduction to Oracle Solaris Zones ”](#).

Installing a Kernel Zone

You install a kernel zone by using the `zoneadm install` command. All zones must be configured before installation. For more information, see [Chapter 2, “Non-Global Zone Configuration Overview,”](#) in [“Introduction to Oracle Solaris Zones ”](#).

You can install a kernel zone through one of the following methods:

- A kernel zone direct installation. See [“Installing a Kernel Zone by Using Direct Installation” on page 20](#).
- An Automated Installation (AI) manifest or an Oracle Solaris system configuration (`sysconfig`) profile. See [“Using AI Manifests and `sysconfig` Profiles in Kernel Zone Installations” on page 22](#)
- Oracle Solaris media. See [“Installing a Kernel Zone from Installation Media” on page 25](#)

Installing a Kernel Zone by Using Direct Installation

Direct installation is the default kernel zone installation method. In a direct installation, the installer runs on the global zone. By default, the installer creates and formats the kernel zone boot disk and installs Oracle Solaris packages on that disk using the global zone's pkg publishers.

Note - In a kernel zone direct installation, the installer can recognize and install only the exact version of Oracle Solaris that is running on the global zone. To install a kernel zone with a different Oracle Solaris version than the global zone, you must use an AI Media installation or interactive text installation. See [“Installing a Kernel Zone from Installation Media” on page 25](#).

A kernel zone direct installation occurs when you do not specify the `-b` option during a `zoneadm install` operation.

▼ How to Install a Kernel Zone Using Direct Installation

1. Become an administrator.

For more information, see [“Using Your Assigned Administrative Rights” in “Securing Users and Processes in Oracle Solaris 11.2”](#).

2. Install the kernel zone.

```
# zoneadm -z zonename install
```

For example, to install the kernel zone `kzone1` on the host `global`:

```
global# zoneadm -z kzone1 install
```

Note - If a direct installation fails after zone verification, confirm that the publishers on the global zone have all of the required package components. See [“Copying and Creating Package Repositories in Oracle Solaris 11.2”](#) for more information.

3. Boot the kernel zone.

```
# zoneadm -z zonename boot
```

For example, to boot the kernel zone `kzone1` on the host `global`:

```
global# zoneadm -z kzone1 boot
```

4. Log in to the kernel zone console to complete the zone configuration process.

```
# zlogin -C zonename
```

For example, to log in to the console on the kernel zone kzone1:

```
global# zlogin -C kzone1
```

Example 2-1 Installing a Kernel Zone Using Direct Installation

This example shows a successful direct installation of the kernel zone kzone1.

```
global# zoneadm -z kzone1 install
Progress being logged to /var/log/zones/zoneadm.20146T195713Z.kzone1.install
pkg cache: Using /var/pkg/publisher.
Install Log: /system/volatile/install.778521/install_log
AI Manifest: /tmp/zoneadm777933.spq5FV/devel-ai-manifest.xml
SC Profile: /usr/share/auto_install/sc_profiles/enable_sci.xml
Installation: Starting ...

Creating IPS image
Startup: Retrieving catalog 'nightly' ... Done
Startup: Caching catalogs ... Done
Startup: Refreshing catalog 'nightly' ... Done
Startup: Refreshing catalog 'solaris' ... Done
Startup: Refreshing catalog 'extra' ... Done
Startup: Caching catalogs ... Done
Installing packages from:
solaris
origin: http://ipkg.us.oracle.com/solaris11/dev/
Startup: Linked image publisher check ... Startup: Refreshing catalog 'nightly' ...
Done
Startup: Refreshing catalog 'solaris' ... Done
Startup: Refreshing catalog 'extra' ... Done
Planning: Solver setup ... Done
Planning: Running solver ... Done
Planning: Finding local manifests ... Done
Planning: Fetching manifests: 0/477 0% complete
Planning: Fetching manifests: 477/477 100% complete
Planning: Package planning ... Done
Planning: Merging actions ... Done
Planning: Checking for conflicting actions ... Done
Planning: Consolidating action changes ... Done
Planning: Evaluating mediators ... Done
Planning: Planning completed in 29.49 seconds
The following licenses have been accepted and not displayed.
Please review the licenses for the following packages post-install:
consolidation/osnet/osnet-incorporation
Package licenses may be viewed using the command:
pkg info --license <pkg_fmri>
```

```
Download: 0/52325 items 0.0/535.0MB 0% complete
Download: 1024/52325 items 30.8/535.0MB 5% complete
Download: 2233/52325 items 42.7/535.0MB 7% complete
...
Download: 46744/52325 items 518.8/535.0MB 96% complete (6.4M/s)
Download: Completed 534.98 MB in 79.80 seconds (5.0M/s)
Actions: 1/74042 actions (Installing new actions)
Actions: 17036/74042 actions (Installing new actions)
...
Actions: 72796/74042 actions (Installing new actions)
Actions: Completed 74042 actions in 97.96 seconds.
Done
Installation: Succeeded
Done: Installation completed in 359.901 seconds.
```

Using AI Manifests and `sysconfig` Profiles in Kernel Zone Installations

You can use an AI manifest or `sysconfig` profile when you need to install multiple kernel zones with specific resource and package configurations different from that of the global zone.

Use the `zoneadm install` command with the `-m` option to specify an alternate AI manifest:

```
# zoneadm -z zonename install -m manifest
```

Use the `-c` option to specify a `sysconfig` profile:

```
# zoneadm -z zonename install -c sysconfig-profile
```

For example, to use the AI manifest `/data/archives/kzone-manifest.xml` to install the kernel zone `kzone1`:

```
global# zoneadm -z kzone1 install -m /data/archives/kzone-manifest.xml
```

Note the following guidelines when installing an alternate AI manifest or `sysconfig` profile to a kernel zone:

- For a successful installation, the AI manifest and `sysconfig` files must include the full path and `.xml` suffix.
- You cannot apply custom disk references in an AI manifest to a kernel zone installation. Because a kernel zone root disk is not available to the global zone, the kernel zone installation script automatically assigns a labeled loopback file, or `lofi`, device during configuration to allow for root disk creation. See [“Removable lofi Device in Non-Global Zones”](#) in [“Introduction to Oracle Solaris Zones”](#) for information about how to configure `lofi` devices.

For additional information about developing and customizing an AI manifest, see [Chapter 9, “Customizing Installations,”](#) in [“Installing Oracle Solaris 11.2 Systems”](#). See [“Zone Installation and Administration Concepts”](#) in [“Creating and Using Oracle Solaris Zones”](#) for further information about zone root disk creation.

EXAMPLE 2-2 Installing a Kernel Zone by Using a Separate Automated Installer (AI) Manifest

This example shows an installation of the kernel zone `kzone1` using the non-default Automated Install (AI) manifest `/var/tmp/kz_manifest.xml`.

```
global# zoneadm -z kzone1 install -m /var/tmp/manifest/kz_manifest.xml
Progress being logged to /var/log/zones/zoneadm.20146T195713Z.kzone1.install
pkg cache: Using /var/pkg/publisher.
Install Log: /system/volatile/install.10708/install_log
AI Manifest: /tmp/zoneadm10343.51a4Vu/devel-ai-manifest.xml
SC Profile: /usr/share/auto_install/sc_profiles/enable_sci.xml
Installation: Starting ...

    Creating IPS image
    Startup: Retrieving catalog 'solaris' ... Done
    Startup: Caching catalogs ... Done
    Startup: Refreshing catalog 'solaris' ... Done
Installing packages from:
    solaris
        origin: http://pkg.oracle.com/solaris/release/
    Startup: Linked image publisher check ... Startup: Refreshing catalog 'solaris' ...
Done
    Planning: Solver setup ... Done
    Planning: Running solver ... Done
    Planning: Finding local manifests ... Done
    Planning: Fetching manifests: 0/501 0% complete
    Planning: Fetching manifests: 501/501 100% complete
    Planning: Package planning ... Done
    Planning: Merging actions ... Done
    Planning: Checking for conflicting actions ... Done
    Planning: Consolidating action changes ... Done
    Planning: Evaluating mediators ... Done
    Planning: Planning completed in 32.07 seconds
    The following licenses have been accepted and not displayed.
    Please review the licenses for the following packages post-install:
        consolidation/osnet/osnet-incorporation
    Package licenses may be viewed using the command:
        pkg info --license <pkg_fmri>

    Download: 0/64687 items 0.0/569.3MB 0% complete
    Download: 931/64687 items 5.8/569.3MB 1% complete (1.2M/s)
    ...
    Download: 64589/64687 items 569.2/569.3MB 99% complete (825k/s)
    Download: Completed 569.25 MB in 358.54 seconds (1.6M/s)
    Actions: 1/88614 actions (Installing new actions)
    Actions: 19471/88614 actions (Installing new actions)
    ...
```

```
Actions: 86994/88614 actions (Installing new actions)
Actions: 87128/88614 actions (Installing new actions)
Actions: Completed 88614 actions in 73.71 seconds.
Installation: Succeeded
Done: Installation completed in 342.508 seconds.
```

```
Log saved in non-global zone as /zones/kzone1/root/var/log/zones/
zoneadm.20146T195713Z.kzone1.install
global#
```

EXAMPLE 2-3 Installing a Kernel Zone by Using an Alternate sysconfig Profile

This example installs the kernel zone `kzone1` using the sysconfig profile `/var/tmp/kzone1.sysconfig.xml`.

```
global# zoneadm -z kzone1 install -c /var/tmp/kzone1-sysconfig.xml
The following ZFS file system(s) have been created:
    rpool/zones/kzone1
Progress being logged to /var/log/zones/zoneadm.20146T195713Z.kzone1.install
pkg cache: Using /var/pkg/publisher.
AI Manifest: /tmp/zoneadm124827.zQWo0h/devel-ai-manifest.xml
SC Profile: /var/tmp/kzone1-sysconfig.xml
Installation: Starting ...
```

```
Creating IPS image
Startup: Retrieving catalog 'nightly' ... Done
Startup: Caching catalogs ... Done
Startup: Refreshing catalog 'nightly' ... Done
Startup: Refreshing catalog 'solaris' ... Done
Startup: Refreshing catalog 'extra' ... Done
Startup: Caching catalogs ... Done
Installing packages from:
    nightly
        origin: file:///server/nightly
    solaris
        origin: file:///server/solaris
    extra
        origin: http://server/extra
Startup: Refreshing catalog 'nightly' ... Done
Startup: Refreshing catalog 'solaris' ... Done
Startup: Refreshing catalog 'extra' ... Done
Planning: Solver setup ... Done
Planning: Running solver ... Done
Planning: Finding local manifests ... Done
...
Planning: Fetching manifests: 552/552 100% complete
Planning: Package planning ... Done
Planning: Merging actions ... Done
Planning: Checking for conflicting actions ... Done
Planning: Consolidating action changes ... Done
Planning: Evaluating mediators ... Done
Planning: Planning completed in 56.62 seconds
```



```
...
Download: 9746/65597 items 143.6/661.7MB 21% complete
Download: 35018/65597 items 370.8/661.7MB 56% complete
Download: 62181/65597 items 654.5/661.7MB 98% complete
Download: Completed 661.67 MB in 40.57 seconds (0B/s)
...
Actions: 87940/89672 actions (Installing new actions)
Actions: 88107/89672 actions (Installing new actions)
Actions: 88745/89672 actions (Installing new actions)
Actions: Completed 89672 actions in 108.50 seconds.
Done
Installation: Succeeded
Done: Installation completed in 342.508 seconds.
```

```
Log saved in non-global zone as /zones/kzone1/root/var/log/zones/
zoneadm.20146T195713Z.kzone1.install
global#
```

Installing a Kernel Zone from Installation Media

You can install kernel zones from Oracle Solaris installation media.

Note the following considerations:

- Interactive media text installations and automated installations from media are both supported. Live media installation is not supported for kernel zones. See [Part II, “Installing Using Installation Media,”](#) in [“Installing Oracle Solaris 11.2 Systems”](#) for additional information about these installation methods.
- The installation media must have support for kernel zones. See [“Verifying Hardware and Software Support on Kernel Zone Hosts”](#) on page 9.
- During installation, you must include the complete path to the installation media image. Otherwise, the Oracle Solaris media installation will fail.

During an Oracle Solaris media installation, the kernel zone is booted and you are connected to the zone console. For information about how to use the zone console, see [“Zone Console Login”](#) in [“Creating and Using Oracle Solaris Zones”](#).



Caution - If you exit or disconnect from the kernel zone console before an Oracle Solaris media installation completes, the installation will fail.

▼ How to Perform an Oracle Solaris Media Installation for a Kernel Zone

1. Become an administrator.

For more information, see [“Using Your Assigned Administrative Rights”](#) in [“Securing Users and Processes in Oracle Solaris 11.2”](#).

2. Install the kernel zone using the Oracle Solaris installation media.

```
global# zoneadm -z zonename install -b media-file
```

For example, to install the kernel zone media located at `/var/tmp/solaris-media.iso` to the kernel zone, `kzone1`:

```
global# zoneadm -z kzone2 install -b /var/tmp/solaris-media.iso
```

Uninstalling a Kernel Zone

Use the `zoneadm uninstall` command to uninstall a kernel zone, for example, before you install a new or updated zone configuration. Note that the zone cannot be in the running state when you perform this operation. See [“Shutting Down, Halting, Rebooting, and Uninstalling Zones”](#) in [“Creating and Using Oracle Solaris Zones”](#) for information about zone uninstall procedures.

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

Shutting Down, Rebooting, and Halting a Kernel Zone

Use the `zoneadm shutdown`, `zoneadm reboot`, and `zoneadm halt` commands to shutdown, reboot, and halt a kernel zone. See [“About Shutting Down, Halting, Rebooting, and Uninstalling Zones”](#) in [“Creating and Using Oracle Solaris Zones”](#) for information about using these commands.

If you want a zone to boot automatically upon the host system reboot, set the `autoboot zonecfg` resource. See [Chapter 1, “How to Plan and Configure Non-Global Zones,”](#) in [“Creating and Using Oracle Solaris Zones”](#) for additional information about how to set this resource.

You must be the global administrator or a user with appropriate authorizations in the global zone to shut down, reboot, or halt a zone.

Cloning a Kernel Zone

Cloning enables you to copy an existing configured and installed zone on your system to a new zone on the same system. The cloned zone includes any customizations of the existing zone. For example, added packages, modified zone resources, and file modifications on the source zone will appear in each cloned zone. Cloning a zone is an efficient way to add additional zones with a similar customized zone configuration.

You can clone a kernel zone in the following ways:

- Use the `zoneadm clone` command if you need to clone a small number of zones. See [Example 2-4](#).
- Use a Unified Archive file if you need to clone multiple zones for a large deployment, such as in a data center environment. See [Example 2-5](#).

Note - A Unified Archive file can include only kernel zones that are in the running state. During Unified Archive creation, you can exclude any kernel zones that are not running. See [Chapter 2, “Working With Unified Archives,”](#) in [“Using Unified Archives for System Recovery and Cloning in Oracle Solaris 11.2 ”](#) for more information.

After a kernel zone is cloned, you can boot and log in to the new zone.

EXAMPLE 2-4 Cloning a Kernel Zone by Using the `zoneadm clone` Command.

This example demonstrates how to clone the kernel zone `kzone1` to the kernel zone `kzone2` on the host `global`. For a step-by-step procedure, see [“Cloning a Non-Global Zone on the Same System”](#) in [“Creating and Using Oracle Solaris Zones ”](#).

```
global# zoneadm -z kzone1 halt
global# zonecfg -z kzone2 create -t kzone1
global# zoneadm -z kzone2 clone kzone1
Progress being logged to /var/log/zones/zoneadm.20140327T223951Z.kzone2.clone
Install Log: /system/volatile/install.100847/install_log
  AI Manifest: /system/shared/ai.xml
Installation: Starting ...

    Creating direct clone image...
    Registering dynamic archive transfer
    Pre-validating manifest targets before actual target selection
    Pre-validation of manifest targets completed
    Validating combined manifest and archive origin targets
    Commencing transfer of stream: ...
    Completed transfer of direct stream: ...
    Archive transfer completed
Installation: Succeeded
```

EXAMPLE 2-5 Cloning and Deploying a Kernel Zone by Using a Unified Archive

The following example demonstrates cloning and deploying the kernel zone `kzone1` by using the `archiveadm` command. A Unified Archive is created for the kernel zone `kzone1`. The archive info is verified and the kernel zone `kzone2` is cloned with the modified zone configuration from `kzone1`. For a step-by-step procedure, please see [“Using Unified Archives for System Recovery and Cloning in Oracle Solaris 11.2”](#).

```
global# archiveadm create -z kzone1 /var/tmp/kzone1.uar
Unified Archive initialized: /var/tmp/kzone1.uar.
\
Logging to: /system/volatile/archive_log.26248
Dataset discovery completed...
/
Media creation complete for zone(s)...
-
Archive stream creation completed...
-
Archive creation completed...
global# zoneadm list -cv
  ID NAME          STATUS    PATH                                BRAND    IP
  0 global          running   /                                    solaris  shared
  2 kzone1          running   -                                    solaris-kz  excl

global# archiveadm info /var/tmp/kzone1.uar
Archive Information
  Creation Time: 2014-04-10T17:12:12Z
  Source Host: global
  Architecture: i386
  Operating System: Oracle Solaris 11.2 X86
  Deployable Systems: kzone1
global# zonecfg -z kzone2 create -a /var/tmp/kzone1.uar
global# zoneadm -z kzone2 install -a /var/tmp/kone1.uar
global# zoneadm list -cv
  ID NAME          STATUS    PATH                                BRAND    IP
  0 global          running   /                                    solaris  shared
  2 kzone1          running   -                                    solaris-kz  excl
  - kzone2          configured -                                    solaris-kz  excl
```

Ssuspending and Resuming a Kernel Zone

You can suspend a kernel zone to disk by using the `zoneadm suspend` command, which is unique to kernel zones.

The `zoneadm suspend` command places a kernel zone in the suspended state. A suspend image is created and then compressed and encrypted by using the advanced encryption standard AES-128-CCM. An encryption key is automatically generated by `/dev/random`. After compression and encryption, the running state of the zone is written to the kernel zone's zone path.

The compressed and encrypted suspend image will generally be smaller than the zone's RAM. Writing the running state of the zone takes up the entire amount of the RAM used by the zone. Therefore, a `zoneadm suspend` operation can potentially use a significant amount of system resources.

You can resume a suspended zone (make all zone operations active again) by invoking the `zoneadm boot` command.

Suspend and resume are supported for a kernel zone only if a kernel zone has a `suspend` resource property in its configuration. The kernel zone template `SYSsolaris-kz` provides the `suspend` resource property by default. For example, the suspend information for the kernel zone `kzone1` is as follows:

```
global# zonecfg -z kzone1 info suspend
suspend:
    path: /system/zones/kzone1/suspend
    storage not specified
```

The `suspend` resource must have either the full file path or the storage location specified. See the [solaris-kz\(5\)](#) man page for more information on `suspend` resource property requirements.

If the suspend image and the rest of the zone storage are accessible by multiple hosts on shared storage, you can use the suspend image to support a zone migration. See [“Migrating a Kernel Zone by Using Suspend and Resume” on page 29](#).

Migrating a Kernel Zone by Using Suspend and Resume

You can migrate a kernel zone to another host by using the `zoneadm suspend` command. For zone migrations, a zone configuration must be portable across machines. For example, you cannot use a `zvol` on the source host.

A zone source and destination must be on the same platform for a zone migration. On x86 systems, the vendor as well as the CPU revision must be identical. On SPARC, the hardware platform must be the same. For example, you cannot migrate a kernel zone from a T4 host to a T3 host.

Note - To prevent loss of the encryption key during a kernel zone migration, use the `zonecfg` `export` command on the source system to generate a command file to be used on the destination system. For example, to generate a command file for a zone migrated from `global1` to `global2`:

```
global1# zonecfg -z kzone1 export -f /net/.../kzone1.cfg
global2# zonecfg -z kzone1 -f /net/.../kzone1.cfg
```

See [“Kernel Zone Host Data and Host ID”](#) on page 41 for additional information.

▼ How to Migrate a Kernel Zone

1. Become an administrator.

For more information, see [“Using Your Assigned Administrative Rights”](#) in [“Securing Users and Processes in Oracle Solaris 11.2”](#).

2. On the global zone, suspend the file system on the kernel zone to be migrated.

```
#zoneadm -z zonename suspend
```

For example, to suspend the kernel zone `kzone1` on the global zone `global`:

```
global# zoneadm -z kzone1 suspend
```

3. Detach the kernel zone file system on the global zone.

```
#zoneadm -z zonename detach
```

For example, to detach the kernel zone `kzone1` on the global zone `global`:

```
global#zoneadm -z kzone1 detach
```

4. Export the zone configuration and transfer the file to the new host.

```
# zonecfg -z zonename export | ssh root@newhost zonecfg -z zonename -f -
```

For example:

```
global# zonecfg -z kzone1 export | ssh root@global2 zonecfg -z kzone1 -f -
```

5. Attach the zone on the new host.

```
# zoneadm -z zonename attach
```

For example:

```
global2# zoneadm -z kzone1 attach
```

6. Boot the kernel zone on the new host to resume the migrated zone.

```
#zoneadm -z zonename boot
```

For example:

```
global2# zoneadm -z kone1 boot
```

See Also For additional information about zones on shared storage and zone migrations, see [“Migrating a Non-Global Zone to a Different Machine”](#) in [“Creating and Using Oracle Solaris Zones ”](#).

◆◆◆ CHAPTER 3

Administering Oracle Solaris Kernel Zones

This chapter covers the following Oracle Solaris Kernel Zone administration topics:

- [“Working in the Kernel Zone Environment” on page 33](#)
- [“Working With Kernel Zones and Immutable Zones” on page 34](#)
- [“Managing Removable Devices on the Kernel Zone” on page 36](#)
- [“Working With Kernel Zone Auxiliary States” on page 38](#)
- [“Managing Nested Zones” on page 39](#)
- [“Kernel Zone Host Data and Host ID” on page 41](#)
- [“Working With the Kernel Zone Boot Loader” on page 42](#)

For information about administrative topics for `solaris` and `solaris10` branded zones, see [Chapter 10, “About Oracle Solaris Zones Administration,” in “Creating and Using Oracle Solaris Zones”](#).

Working in the Kernel Zone Environment

Working in a kernel zone environment is very similar to working in a global zone. This section describes the major differences between the kernel zone administrative environment and working with a global zone.

Process ID Visibility in Zones

Kernel zone processes are not directly visible to the kernel zone host. You must use the `zlogin` command followed by a process management command to view any process information about a kernel zone. For example, to see process information about `syslogd` on the kernel zone `kzone1` from the kernel zone host `global`:

```
global# zlogin kzone1 ps -ef |grep syslogd
root 1520      1   0 20:23:08 ?          0:00 /usr/sbin/syslogd
```

Duplicate Process IDs in Kernel Zones

The global zone and each kernel zone manage their own process ID space. The same numeric process ID might identify different system processes in the global zone and in one or more kernel zones. For example, on the same physical host, you can have the numeric process 5678 running `syslogd` on the global zone and running `sendmail` on a kernel zone.

To kill process 5678 with the `ps` command in `kzone1`, use the `zlogin` command followed by the `kill` command.

```
global# zlogin kzone1 kill 5678
```

Kernel Zone Zonepath

A kernel zone's `zonpath`, by design, cannot be set. It contains no persistent or otherwise serviceable data.

Resource Management Functionality in Kernel Zones

Resource controls such as `max-processes` are not available when configuring a kernel zone. Because a kernel zone has an independent kernel from the global zone, a process running inside a kernel zone cannot take up a process table slot in the global zone.

Working With Kernel Zones and Immutable Zones

Immutable Zones provide read-only, or immutable, file system profiles. Immutable zones are supported on both `solaris` branded zones (in non-global zones) and on kernel zones. For detailed information regarding immutable zones, see [“Read-Only Zone Overview”](#) in [“Creating and Using Oracle Solaris Zones”](#).

Configuring Immutable Kernel Zones

To configure a kernel zone as an immutable zone, set the `file-mac-profile` property from within the kernel zone. See [“Configuring Read-Only Zones”](#) in [“Creating and Using Oracle Solaris Zones”](#) for additional information about the `file-mac-profile` property.

▼ How to Configure a Kernel Zone as an Immutable (Read-Only) Zone

1. Become an administrator.

For more information, see [“Using Your Assigned Administrative Rights”](#) in [“Securing Users and Processes in Oracle Solaris 11.2”](#).

2. Boot the kernel zone.

```
# zoneadm kernel-zone-name boot
```

3. Log into the kernel zone.

```
# zlogin kernel-zone-name
```

4. Set the immutable zone from within the kernel zone.

```
kernel-zone-name# zonecfg -z global set file-mac-profile=value
```

For example:

```
kzone1# zonecfg -z global set file-mac-profile=flexible-configuration
updating /platform/i86pc/amd64/boot_archive
```

5. Reboot the kernel zone.

```
kernel-zone-name# reboot
```

Maintaining and Administering Read-Only Kernel Zones

Use the Trusted Path mode in the `zlogin` command to perform maintenance and administration operations on read-only kernel zones. The Trusted Path mode has the following options:

- T Allows administrators to enter an immutable zone with the Trusted Path attribute `PRIV_PROC_TPD` set. This session can modify files that are normally immutable. The session cannot read unprotected files.
- U Performs the same process as the `-T` option, but in unsafe mode. This option is required for zones with the `flexible-configuration` setting of the `file-mac-profile` property.

Note - These options cannot be used with console login sessions. Both the `-T` and `-U` `zlogin` command options are ignored for zones that are not immutable zones.

See the [zlogin\(1\)](#) man page for more information.

Managing Removable Devices on the Kernel Zone

You can configure a removable loopback file `lofi` device, which works as a CD-ROM device, on the kernel zone. See [“Removable lofi Device in Non-Global Zones”](#) in [“Introduction to Oracle Solaris Zones”](#) for additional information about how to configure `lofi` devices.

▼ How to Add a Virtual CD-ROM Device to a Kernel Zone

1. Become an administrator.

For more information, see [“Using Your Assigned Administrative Rights”](#) in [“Securing Users and Processes in Oracle Solaris 11.2”](#).

2. Create an empty removable read-only `lofi` device.

```
# lofiadm -r
```

The following example shows sample output.

```
global# lofiadm -r
/dev/lofi/1
```

3. Add the `lofi` device to the kernel zone.

```
# zonecfg -z zonename
```

The following example demonstrates adding the `lofi` device located at `/dev/rlofi/1` to the kernel zone `kzone1`:

```
global# zonecfg -z kzone1
zonecfg:kzone1> add device
zonecfg:kzone1:device> set match=/dev/rlofi/1
zonecfg:kzone1:device> end
zonecfg:kzone1> exit
```

4. Reboot the kernel zone to have the configuration changes take effect.

```
# zoneadm -z zonename reboot
```

5. On the kernel zone, update the device file system (devfs) and hardware abstraction layer (hal) so that the hal will see the new cdrom device.

```
# devfsadm -i zvblk
```

For example:

```
global# devfsadm -i zvblk
global# svcadm restart hal
global# rmformat -l
...
  2. Logical Node: /dev/rdisk/c1dp0
     Physical Node: /zvnex/zvblk@1
     Connected Device: kz          vCDROM          0
     Device Type: CD Reader
     Bus: <Unknown>
     Size: 0.0 MB
     Label: <Unknown>
     Access permissions: Medium is write protected.
```

6. Restart the hal service on the kernel zone.

```
# svcadm restart hal
```

7. List the removable devices on the zone.

```
# rmformat -l
```

For example, the following example lists the removable devices on the kernel zone kzone1:

```
kzone1# rmformat -l
Looking for devices...
  1. Logical Node: /dev/rdisk/c1d0p0
     Physical Node: /zvnex/zvblk@0
     Connected Device: kz          vDisk          0
     Device Type: Removable
     Bus: <Unknown>
     Size: 16.4 GB
     Label: <Unknown>
     Access permissions: <Unknown>
  2. Logical Node: /dev/rdisk/c1d1p0
     Physical Node: /zvnex/zvblk@1
     Connected Device: kz          vCDROM          0
     Device Type: CD Reader
     Bus: <Unknown>
     Size: 0.0 MB
     Label: <Unknown>
     Access permissions: <Unknown>
```

8. Insert a CD-ROM into the CD-ROM drive on the kernel zone.

```
# lofiadm -r device-name device-path
```

The following example demonstrates inserting the CD-ROM image `/root/sol-11_3-repo.full.iso` in the `lofi` device `/dev/lofi/1` on the kernel zone `kzone1`:

```
kzone1# lofiadm -r /root/sol-11_2-repo-full.iso /dev/lofi/1
kzone1# lofiadm
Block Device      File              Options
/dev/lofi/1      /root/sol-11_2-repo-full.iso  Removable,ReadOnly
```

9. Mount the CD-ROM.

```
# mount -F hsfs device-location /mnt
```

The following example mounts the CD-ROM device located at `/dev/dsk/c1d1p0`.

```
kzone1# mount -F hsfs /dev/dsk/c1d1p0 /mnt
```

10. Unmount the CD-ROM device.

```
# unmount /mnt
```

11. Eject the CD-ROM.

```
# eject cdrom
```

12. Verify that the CD-ROM device is no longer present on the kernel zone host.

```
# lofiadm
```

For example:

```
global# lofiadm
Block Device      File              Options
/dev/lofi/1      -                 Removable,ReadOnly
```

Working With Kernel Zone Auxiliary States

Kernel zones use *auxiliary states* to communicate supplementary state information to the global zone. A kernel zone does not have an auxiliary state set by default. Auxiliary states are set only when you initiate debugging and kernel maintenance operations.

To view the global zone current state and the kernel zone auxiliary states, use the `zoneadm list -s` command.

```
global# zoneadm list -s
```

NAME	STATUS	AUXILIARY STATE
global	running	
kzone1	running	
kzone2	running	
kzone3	running	debugging

The available kernel zone auxiliary states are as follows:

suspended	The zone has been suspended and will resume on the next boot. Note that the zone must be attached before this state is visible. A kernel zone appears in a suspended auxiliary state when undergoing a migration. See “Migrating a Kernel Zone by Using Suspend and Resume” on page 29 .
debugging	The kernel zone is in the kernel debugger, kmdb. Although the zone is in the running state, the zone cannot service any network requests. You must connect to the zone console to interact with kmdb. For information about how to connect to the zone console, see Chapter 4, “About Non-Global Zone Login,” in “Creating and Using Oracle Solaris Zones” .
panicked	The zone is in the running state but has panicked. The host is not affected. You must use zone console access to log in to a kernel zone in the panicked auxiliary state.

For information about zone states, see [Chapter 1, “Oracle Solaris Zones Introduction,”](#) in [“Introduction to Oracle Solaris Zones”](#). For additional information about kernel zone auxiliary states, see the [solaris-kz\(5\)](#) man page. For information about the kernel debugger see the [kmdb\(1\)](#) man page.

Managing Nested Zones

A *nested zone* is a non-global zone that is installed and booted from within a kernel zone. In nested zones, the kernel zone serves as the global zone. A nested zone can be a new solaris branded zone or a migrated solaris or solaris10 branded zone. Kernel zones are not supported as a nested zone.

Nested zones have the following requirements:

Operating System	All nested zones must have support for Oracle Solaris 11.2 or higher. <ul style="list-style-type: none"> ■ solaris branded zones running Oracle Solaris 11 or Oracle Solaris 11.1 must be updated to Oracle Solaris 11.2. See Chapter 3, “Installing and Updating Software Packages,” in “Adding and Updating Software in Oracle Solaris 11.2” for information about updating system software packages.
-------------------------	--

- You can migrate a `solaris10` branded zone to a non-global zone running at least Oracle Solaris 11.2. See [“Migrating a solaris10 Branded Zone to Another Host”](#) in [“Creating and using Oracle Solaris 10 Zones”](#) for procedures on how to migrate a `solaris10` zone to an Oracle Solaris 11.2 host.

Network Configuration

A `solaris` or `solaris10` branded zone that runs as a nested zone can use exclusive-IP or shared-IP. If you need an exclusive-IP configuration, you need to configure the kernel zone to allow for additional MAC addresses.

System Resources

Nested zones only can use system resources available to the kernel zone. These resources include virtual disks and iSCSI disks.

Cloning

If a kernel zone containing a nested configuration is cloned, only the outside kernel zone will be cloned. Any zones inside the kernel zone are not cloned during the zone cloning process. See [“Cloning a Kernel Zone”](#) on page 27.

▼ How to Add Multiple MAC Addresses to a Kernel Zone

1. Become an administrator.

For more information, see [“Using Your Assigned Administrative Rights”](#) in [“Securing Users and Processes in Oracle Solaris 11.2”](#).

2. Add the new MAC addresses.

```
global# zonecfg:kz-1> select anet id=0
zonecfg:kz-1:anet> add mac
zonecfg:kz-1:anet:mac> set mac-address=0:16:3e:86:19:70
zonecfg:kz-1:anet:mac> end
zonecfg:kz-1:anet> info
anet:
  lower-link: net0
  ...
  mac-address: 0:16:3e:86:11:70
  ...
  id: 0
  mac:
    mac-address: 0:16:3e:86:19:70
    mac-prefix not specified
  id: 0
```


Nested Zones and New Non-Global Zone Configuration

You can configure, install, and boot a new `solaris` branded zone from within a kernel zone using `zonecfg` and `zoneadm` commands. For example:

```
kzone1# zonecfg -z zone1
Use 'create' to begin configuring a new zone.
zonecfg:zone1> create -t SYSsolaris
zonecfg:zone1> commit
zonecfg:zone1> exit
```

See [“Creating and Using Oracle Solaris Zones”](#) for additional information about planning, configuring, and installing non-global zones.

Kernel Zone Host Data and Host ID

Each kernel zone bootable device contains state information known as *host data*. A kernel zone's host data monitors kernel zone state information including:

- Zone usage
- Zone suspends, as described in [“Suspending and Resuming a Kernel Zone”](#) on page 28
- Time of day offset between the kernel zone clock and the global zone clock
- OpenBoot variables (SPARC only)

Kernel zone host data is encrypted and authenticated with the advanced encryption standard AES-128-CCM, using the same encryption key used for the kernel zone suspend image.

When a kernel zone is configured or booted, the host data is read to determine whether the kernel zone's boot storage is in use on another system. If the boot storage is in use on another system, the kernel zone will enter the `unavailable` state and an error message will indicate which system is using the boot storage. For example:

```
global# zoneadm -z kzone1 attach
zone 'kzone1': error: ERROR: zone kzone1 is in use by host with  hostid 848611d4
zone 'kzone1': error:      last known state: installed
zone 'kzone1': error:      hostname: global2
zone 'kzone1': error:  boot environment name: solaris-1
zone 'kzone1': error:  boot environment uuid: 69ed2e6a-e25a-6d36-e022-ed7261ed8899
zone 'kzone1': error:      last update time: Sun Apr 13 20:08:13 2014
zone 'kzone1': error: To fix, detach the zone from the other host then attach it to this host
zone 'kzone1': error: If the zone is not active on another host, attach it with
zone 'kzone1': error:  zoneadm -z kzone1 attach -x force-takeover
```

If the boot storage is not in use by the other system, you can repair the kernel zone by using the `zoneadm attach -x force-takeover` command.



Caution - Forcing a takeover or reinitialization of the host data makes it impossible to detect if the zone is in use on any other system. Running multiple instances of a zone that reference the same storage leads to unrepairable corruption of the zone's file systems.

If a zone's encryption key is not accessible, the host data and any suspend image will not be readable. In such circumstances, any attempt to ready or boot the zone will cause the zone to enter the unavailable state. If recovery of the zone's encryption key is not possible, use the `zoneadm attach -x initialize-hostdata` command to generate a new encryption key and host data.

To prevent loss of the encryption key during a kernel zone migration, use the `zonecfg export` command on the source system to generate a command file to be used on the destination system. For example:

```
global# zonecfg -z kzone1 export -f /net/.../kzone1.cfg
global# zonecfg -z kzone1 -f /net/.../kzone1.cfg
```

Working With the Kernel Zone Boot Loader

The kernel zone boot loader manages booting operations on the kernel zone. To invoke the boot loader, the kernel zone must be in the ready or installed state. You can use the kernel zone boot loader to perform the following operations:

- List available boot environments
- Boot the zone to an alternate boot environment

Use the `zoneadm boot` command to invoke the kernel zone boot loader. You must also invoke the zone console when you invoke the kernel zone boot loader. The boot loader output will appear in the zone console.

Note - The command sequence to exit from the zone console is `~..`. See [“How to Log In to the Zone Console”](#) in [“Creating and Using Oracle Solaris Zones”](#) for additional information.

For information about creating and managing boot environments on the operating system level, see [Chapter 1, “Introduction to Managing Boot Environments,”](#) in [“Creating and Administering Oracle Solaris 11.2 Boot Environments”](#). Additional information for managing zones and boot environments is available in [Chapter 2, “beadm Zones Support,”](#) in [“Creating and Administering Oracle Solaris 11.2 Boot Environments”](#).

▼ How to Specify Alternate Boot Environments in a Kernel Zone

1. Become an administrator.

For more information, see [“Using Your Assigned Administrative Rights”](#) in [“Securing Users and Processes in Oracle Solaris 11.2”](#).

2. Log into the zone console.

```
# zlogin -C zonename
```

For example, to log into the console on kzone1:

```
global# zlogin -C kzone1
```

3. In a separate terminal window, list the available kernel zone boot environments.

```
# zoneadm -z zonename boot -- -L
```

The following example shows sample output.

```
global# zoneadm -z kzone2 boot -- -L
[Connected to zone 'kzone2' console]
1 kz-130118 (rpool/ROOT/kz-130118)
2 kz-1 (rpool/ROOT/kz-1)
3 solaris-5 (rpool/ROOT/solaris-5)
4 solaris-7 (rpool/ROOT/solaris-7)
Select environment to boot: [ 1 - 4 ]:
```

4. Boot to a selected boot environment.

```
# zoneadm -z zonename boot -- -Z boot-environment
```

For example:

```
global# zoneadm -z kzone1 boot -- -Z rpool/ROOT/solaris-backup-1
```

Example 3-1 Selecting and Booting Alternate Boot Environments on a SPARC Host

The following example shows the zone console output for alternate boot environments for the kernel zone kzone1. The kernel zone host hardware is a SPARC system.

```
[Connected to zone 'kzone1' console]
NOTICE: Entering OpenBoot.
NOTICE: Fetching Guest MD from HV.
NOTICE: Starting additional cpus.
NOTICE: Initializing LDC services.
NOTICE: Probing PCI devices.
```

```
NOTICE: Finished PCI probing.
```

```
SPARC T4-2, No Keyboard  
Copyright (c) 1998, 2014, Oracle and/or its affiliates. All rights reserved.  
OpenBoot 4.36.0.build_05, 2.0000 GB memory available, Serial #1845652596.  
Ethernet address 0:0:0:0:0:0, Host ID: 6e026c74.
```

```
Boot device: disk0 File and args: -L  
1 Oracle Solaris 11.2 SPARC  
2 bootenv123  
3 bootenv456  
Select environment to boot: [ 1 - 3 ]: 2
```

```
To boot the selected entry, invoke:  
boot [<root-device>] -Z rpool/ROOT/bootenv123
```

```
Program terminated  
ok boot -Z rpool/ROOT/bootenv123
```

```
[NOTICE: Zone rebooting]  
NOTICE: Entering OpenBoot.  
NOTICE: Fetching Guest MD from HV.  
NOTICE: Starting additional cpus.  
NOTICE: Initializing LDC services.  
NOTICE: Probing PCI devices.  
NOTICE: Finished PCI probing.
```

```
SPARC T4-2, No Keyboard  
Copyright (c) 1998, 2014, Oracle and/or its affiliates. All rights reserved.  
OpenBoot 4.36.0.build_05, 2.0000 GB memory available, Serial #1845652596.  
Ethernet address 0:0:0:0:0:0, Host ID: 6e026c74.
```

```
...  
Hostname: kzone1  
kzone1 console login:
```

Example 3-2 Selecting and Booting Alternate Boot Environments on an x86 Host

The following example shows the zone console output for alternate boot environments for the kernel zone kzone1. The kernel zone host hardware is an x86 system.

```
[Connected to zone 'kzone1' console]  
1 boot-2 (rpool/ROOT/boot-2)  
2 Oracle Solaris 11.2 x86 (rpool/ROOT/solaris)  
3 boot-1 (rpool/ROOT/boot-1)  
Select environment to boot: [ 1 - 3 ]:2  
Boot device: disk0 File and args:  
reading module /platform/i86pc/amd64/boot_archive...done.  
reading kernel file /platform/i86pc/kernel/amd64/unix...done.
```

```
SunOS global 5.11 11.2 i86pc i386 i86pc
Copyright (c) 1983, 2014, Oracle and/or its affiliates. All rights reserved.
Hostname: kzone1
...
kzone1 console login:
```


Index

A

- add-device resource property, 15
- adding memory, 14
- adding network devices, 17
- adding storage devices, 15
- anet resource, 17
- archiveadm command, 27
- Automated Installation (AI) manifests
 - using for kernel zone installations, 22
- auxiliary states, 38

B

- boot environments, specifying, 43
- boot loader, 42
- bootpri resource property, 15
- brand, 7
- branded zone, 7
- BrandZ, 7

C

- capped-memory resource property, 14
- cloning a kernel zone, 27
- configuring a kernel zone, 11
- configuring a read-only kernel zone, 34
- configuring an immutable kernel zone, 34
- configuring kernel zone resources, 12
- CPUs
 - managing, 13

D

- dedicated-cpu resource property, 13
- default kernel zone installation method, 20

- direct installation, 20
- duplicate process IDs
 - kernel zones and, 34

G

- general zones concepts, 8

H

- hardware requirements, 8
- hierarchical zones
 - configuration, 39
 - requirements, 39
- host data, 41
- host ID, 41
- host requirements, 8

I

- immutable zone, 34
- installation media
 - using for kernel zone installations, 25

K

- kernel zone boot loader, 42
- kernel zone configuration, 11
- kernel zone installation
 - direct installation, 20
- kernel zone installations
 - Automated Installation (AI) manifests, 22
 - installation media, 25
 - sysconfig profiles, 22
- kernel zone migration, 29

kernel zone root disks, 15

L

lofi devices, 36

M

memory

 managing, 14

migrating a kernel zone, 29

N

nested zones, 39

net resource, 17

network device id, 17

network devices

 adding, 17

 removing, 17

O

Oracle Solaris Kernel Zones, 8

See also kernel zone, solaris-kz branded zone,

 operating system requirement

 definition of, 7

 related concepts, 8

P

process IDs

 visibility in kernel zones, 33

R

read-only kernel zone, 34

removable device configuration, 36

resuming a kernel zone, 28

S

shutting down a kernel zone, 26

software requirements, 8

solaris-kz branded zone, 7

SPARC firmware requirements, 8

suspending a kernel zone, 28

sysconfig profiles

 kernel zone installations, 22

U

uninstalling a kernel zone, 26

V

verifying support on a host, 9

virtinfo command, 9

virtual-cpu resource property, 13

W

warm migration, 29

X

x86 BIOS requirements, 8

Z

ZFS ARC cache requirements, 8

 tuning, 11

zone

 branded *See* solaris-kz branded zone

zoneadm boot command, 42

zoneadm clone command, 27

zoneadm halt command, 26

zoneadm install command, 19

zoneadm list -s command, 38

zoneadm reboot command, 26

zoneadm resume command, 28

zoneadm shutdown command, 26

zoneadm suspend command, 28

zoneadm uninstall command, 26

zonepath, 34