SO304 - Zone Administration

Solaris Containers

The term Solaris **Containers** is often confused with that of Solaris **Zones**. In fact, there is a slight difference. The definition of a container is:

Solaris container = Solaris Zone + Solaris Resource Manager (SRM)

The **SRM** is responsible for workload and resource management.

Solaris Zones are **not** full guest operating system kernels and as such they are similar in concept to FreeBSD **Jails**. Zones share the same kernel and as such cannot be live migrated to another host.

There are two types of zones:

- a single **Global** zone,
- one or more **Non-global** or **local** zones.

Each local zone requires about 400 Mb of disk space and 15 Mb of RAM.

The Global Zone

The global zone:

- has a zone **ID** of **0**,
- provides a unique instance of the Solaris kernel,
- contains all packages installed by IPS,
- can contains other software not installed by IPS,

- contains a database of all applications installed in the global zone,
- contains all configuration data concerning the global zone such as its host name,
- knows about all *devices* and all *file systems*,
- is aware of all local zones as well as their configuration,
- is the zone in which local zones can be created, installed, configured, managed, un-installed and deleted.

Non-global or Local Zones

A local zone:

- is given a zone ID when it is booted,
- shares the kernel with the global zone,
- contains a some of the installed packages,
- shares packages with the global zone,
- can contain other software and files not present in the global zone,
- contains a database of all locally installed applications as well as all applications shared by the global zone,
- has no knowledge of the other local zones,
- cannot be used to manage or to un-install local zones, including itself,
- contains all configuration data concerning the local zone such as its host name and IP address,

For those familiar with Solaris 10 zones, there were two types of local zones:

- ***Small zones*** or *Sparse Root zones* where the zone shared the following global zone directories:
  - /usr
  - /lib
  - /platform
  - /sbin
- ***Big zones*** or *Whole Root zones* that contained a complete Solaris installation.

In Solaris 11 only Whole Root Zones remain.

Lab #1 - Installing a Non-global Zone

In this lab you will be installing a **Local Zone** into a ZFS file system. Start by looking at where you can create the directory that will contain future zones:

```
root@solaris:~# zfs list
NAME                            USED  AVAIL  REFER  MOUNTPOINT
mypool                          103M  51.5M    31K  /mypool
mypool/iscsi                    103M   155M    16K  -
rpool                          7.40G  11.9G  4.58M  /rpool
rpool/ROOT                     5.22G  11.9G    31K  legacy
rpool/ROOT/solaris             5.22G  11.9G  4.08G  /
rpool/ROOT/solaris-backup-1    2.47M  11.9G  1.98G  /
rpool/ROOT/solaris-backup-1/var  46K  11.9G   758M  /var
rpool/ROOT/solaris-backup-2     127K  11.9G  3.92G  /
rpool/ROOT/solaris-backup-2/var  58K  11.9G   266M  /var
rpool/ROOT/solaris/var          980M  11.9G   209M  /var
rpool/VARSHARE                  102K  11.9G   102K  /var/share
rpool/dump                     1.03G  12.0G  1.00G  -
rpool/export                    110M  11.9G    32K  /export
rpool/export/home               110M  11.9G    32K  /export/home
rpool/export/home/trainee       110M  11.9G   110M  /export/home/trainee
rpool/swap                     1.03G  12.0G  1.00G  -
```

You **cannot** create zone datasets under the **rpool/ROOT** dataset.

Configuring the Zone's Dataset

It seems that the best option is to create a new file system just for zones:

```
root@solaris:~# zfs create -o mountpoint=/zones rpool/zones
root@solaris:~# zfs list
NAME                              USED  AVAIL  REFER  MOUNTPOINT
mypool                            103M  51.5M    31K  /mypool
mypool/iscsi                      103M   155M    16K  -
rpool                            7.40G  11.9G  4.58M  /rpool
rpool/ROOT                       5.22G  11.9G    31K  legacy
rpool/ROOT/solaris               5.22G  11.9G  4.08G  /
rpool/ROOT/solaris-backup-1      2.47M  11.9G  1.98G  /
rpool/ROOT/solaris-backup-1/var    46K  11.9G   758M  /var
rpool/ROOT/solaris-backup-2       127K  11.9G  3.92G  /
rpool/ROOT/solaris-backup-2/var    58K  11.9G   266M  /var
rpool/ROOT/solaris/var            980M  11.9G   209M  /var
rpool/VARSHARE                    102K  11.9G   102K  /var/share
rpool/dump                       1.03G  12.0G  1.00G  -
rpool/export                      110M  11.9G    32K  /export
rpool/export/home                 110M  11.9G    32K  /export/home
rpool/export/home/trainee         110M  11.9G   110M  /export/home/trainee
rpool/swap                       1.03G  12.0G  1.00G  -
rpool/zones                        31K  11.9G    31K  /zones
```

Now create your zone using the **zonecfg** command:

```
root@solaris:~# zonecfg -z myzone
Use 'create' to begin configuring a new zone.
zonecfg:myzone> create
create: Using system default template 'SYSdefault'
zonecfg:myzone> set zonepath=/zones/myzone
zonecfg:myzone> set autoboot=true
zonecfg:myzone>
```

The **-z** switch stands for the **zonename**.

Zones are represented by **XML** files. As you can see above, when created, the zonecfg uses the default template **/etc/zones/SYSdefault.xml**:

```
root@solaris:~# cat /etc/zones/SYSdefault.xml
<?xml version="1.0"?>

<!--
 Copyright (c) 2010, 2011, Oracle and/or its affiliates. All rights reserved.

    DO NOT EDIT THIS FILE.  Use zonecfg(1M) instead.
-->

<!DOCTYPE zone PUBLIC "-//Sun Microsystems Inc//DTD Zones//EN" "file:///usr/share/lib/xml/dtd/zonecfg.dtd.1">

<zone name="default" zonepath="" autoboot="false" brand="solaris"
      ip-type="exclusive">
  <automatic-network lower-link="auto" linkname="net0"
             link-protection="mac-nospoof" mac-address="random"/>
</zone>
```

Note that you also have set the **autoboot** property to **yes** so that the zone starts on system boot.

To show the configuration that zonecfg has already filled in for you by using the **/etc/zones/SYSdefault.xml** file, use the command **info** :

```
zonecfg:myzone> info
zonename: myzone
zonepath: /zones/myzone
brand: solaris
autoboot: true
bootargs:
file-mac-profile:
pool:
limitpriv:
scheduling-class:
ip-type: exclusive
```

```
hostid:
fs-allowed:
anet:
    linkname: net0
    lower-link: auto
    allowed-address not specified
    configure-allowed-address: true
    defrouter 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
    pkey not specified
    linkmode not specified
zonecfg:myzone>
```

Finally, **commit** the configuration, **verify** the newly created **XML** file and **quit**:

```
zonecfg:myzone> commit
zonecfg:myzone> verify
zonecfg:myzone> exit
```

```
root@solaris:~#
```

Your zone's configuration is now in its own **XML** file:

```
root@solaris:~# cat /etc/zones/myzone.xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE zone PUBLIC "-//Sun Microsystems Inc//DTD Zones//EN" "file:///usr/share/lib/xml/dtd/zonecfg.dtd.1">
<!--
    DO NOT EDIT THIS FILE.  Use zonecfg(1M) instead.
-->
<zone name="myzone" zonepath="/zones/myzone" autoboot="true" brand="solaris" ip-type="exclusive">
  <automatic-network lower-link="auto" linkname="net0" link-protection="mac-nospoof" mac-address="random"/>
</zone>
```

Display the zones on your system by using the **list** subcommand of the **zoneadm** command:

```
root@solaris:~# zoneadm list -cv
  ID NAME             STATUS     PATH                           BRAND    IP
   0 global           running    /                              solaris  shared
   - myzone           configured /zones/myzone                  solaris  excl
```

The switches used with the **list** subcommand are:

Switch	Description
-c	Display all zones
-v	Display verbose output

Installing a Zone

Now you are ready to install the zone with the following command:

```
root@solaris:~# zoneadm -z myzone install
```

Go grab a cup of coffee or juice ! The installation process can take upto 20 minutes.

When installation is complete, you will see something similar to the following:

```
root@solaris:~# zoneadm -z myzone install
The following ZFS file system(s) have been created:
    rpool/zones/myzone
Progress being logged to /var/log/zones/zoneadm.20121214T123059Z.myzone.install
       Image: Preparing at /zones/myzone/root.

 AI Manifest: /tmp/manifest.xml.UWaiVk
  SC Profile: /usr/share/auto_install/sc_profiles/enable_sci.xml
    Zonename: myzone
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  154k/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 3389.965 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 /zones/myzone/root/var/log/zones/zoneadm.20121214T123059Z.myzone.install
```

Now use zonesadm's **list** subcommand to display the zones:

```
root@solaris:~# zoneadm list -cv
  ID NAME             STATUS     PATH                           BRAND    IP
   0 global           running    /                              solaris  shared
   - myzone           installed  /zones/myzone                  solaris  excl
```

Note that the myzone **STATUS** is now **installed** as opposed to **configured**.

A Zone's First Boot

Verify myzone and then boot it:

```
root@solaris:~# zoneadm -z myzone verify
root@solaris:~# zoneadm -z myzone boot
```

Check if the zone status is now **running**:

```
root@solaris:~# zoneadm list -cv
  ID NAME             STATUS     PATH                           BRAND    IP
   0 global           running    /                              solaris  shared
   1 myzone           running    /zones/myzone                  solaris  excl
```

Now you can login to the zone using the **zlogin** command:

```
root@solaris:~# zlogin -C myzone
[Connected to zone 'myzone' console]
```

Hit ↵ Enter and configure the zone:

- host name = myzone.fenestros.loc
- time zone = Europe/Paris
- root password = Wind0w$
- user name and login = myzone
- user password = fenestr0$

Once configured you will see messages similar to the following:

```
SC profile successfully generated.
Exiting System Configuration Tool. Log is available at:
/system/volatile/sysconfig/sysconfig.log.7316
```

Use the tilde-dot (~.) shortcut to leave the zone and return to your global zone:

```
~.
[Connection to zone 'myzone' console closed]
root@solaris:~#
```

Logging into a Zone Directly as Root

Log back into the zone as root using the **-S** switch of the **zlogin** command:

```
root@solaris:~# zlogin -S myzone
[Connected to zone 'myzone' pts/4]
@myzone:~$ whoami
root
@myzone:~$ ~.
[Connection to zone 'myzone' pts/4 closed]
root@solaris:~#
```

Logging into a Zone as a specific User

To log into the zone as the **myzone** user that you previously created, use the following command:

```
root@solaris:~# zlogin -l myzone myzone
[Connected to zone 'myzone' pts/4]
No directory! Logging in with home=/
Oracle Corporation  SunOS 5.11    11.1    September 2012
-bash-4.1$ whoami
myzone
-bash-4.1$ ~.
[Connection to zone 'myzone' pts/4 closed]
root@solaris:~#
```

LAB #2 - Administering Zones

Sharing Files between the Global and Local Zones

To share files between the two zones, you need to configure a LOFS mount.

In the global zone, create a directory for sharing files:

```
root@solaris:~# mkdir -p /root/zones/myzone
```

Now use the **zonecfg** command to configure the share:

```
root@solaris:~# zonecfg -z myzone
zonecfg:myzone> add fs
zonecfg:myzone:fs> set dir=/root/share
zonecfg:myzone:fs> set special=/root/zones/myzone
zonecfg:myzone:fs> set type=lofs
zonecfg:myzone:fs> add options [rw,nodevices]
zonecfg:myzone:fs> end
zonecfg:myzone> exit
root@solaris:~#
```

Note that **dir** indicates the mount point in the local zone whilst **special** indicates the share in the global zone.

Now create a file in **/root/zones/myzone**:

```
root@solaris:~# touch /root/zones/myzone/testshare
```

Reboot myzone, check it is up and running, log into myzone as root and check you can see the share. Finally, create a file in **/root/share**:

```
root@solaris:~# zoneadm -z myzone reboot
root@solaris:~# zoneadm list -cv
  ID NAME             STATUS     PATH                           BRAND    IP
   0 global           running    /                              solaris  shared
   2 myzone           running    /zones/myzone                  solaris  excl
root@solaris:~# zlogin -S myzone
```

```
[Connected to zone 'myzone' pts/4]
@myzone:~$ cd /root
@myzone:~root$ ls
share
@myzone:~root$ ls share
testshare
@myzone:~root$ touch share/shareback
@myzone:~root$ ls share
shareback  testshare
```

Leave myzone and check if you can see the **shareback** file from the global zone:

```
@myzone:~root$ ~.
[Connection to zone 'myzone' pts/4 closed]
root@solaris:~# ls /root/zones/myzone
shareback  testshare
```

You can also share the global zone's DVD-ROM drive. However do **not** use the process explained above since it creates a permanent LOFS mount which will stop you ejecting the DVD from the global zone's DVD-ROM drive whilst the local zone is running:

```
root@solaris:~# mkdir /zones/myzone/root/globaldvdrom
root@solaris:~# ls /cdrom/cdrom0
32Bit                           autorun.sh                      runasroot.sh
VBoxWindowsAdditions-amd64.exe
64Bit                           cert                            VBoxLinuxAdditions.run
VBoxWindowsAdditions-x86.exe
AUTORUN.INF                     OS2                             VBoxSolarisAdditions.pkg
VBoxWindowsAdditions.exe
root@solaris:~# mount -F lofs /cdrom/cdrom0 /zones/myzone/root/globaldvdrom
```

Now check you can see the contents of the DVD from within the local zone:

```
root@solaris:~# zlogin myzone ls /globaldvdrom
32Bit
```

```
64Bit
AUTORUN.INF
autorun.sh
cert
OS2
runasroot.sh
VBoxLinuxAdditions.run
VBoxSolarisAdditions.pkg
VBoxWindowsAdditions-amd64.exe
VBoxWindowsAdditions-x86.exe
VBoxWindowsAdditions.exe
```

Finally unmount the DVD-Rom and eject it:

```
root@solaris:~# umount /zones/myzone/root/globaldvdrom
root@solaris:~# eject cdrom
cdrom /dev/dsk/c7t1d0s2 ejected
```

Removing the Share

In order to remove the LOFS share, proceed as follows:

```
root@solaris:~# zonecfg -z myzone
zonecfg:myzone> info fs
fs:
    dir: /root/share
    special: /root/zones/myzone
    raw not specified
    type: lofs
    options: [rw,nodevices]
zonecfg:myzone> remove fs dir=/root/share
zonecfg:myzone> exit
```

```
root@solaris:~# zoneadm -z myzone reboot
```

Allocating CPU Resources

First, lets see what the non-global zone currently sees as available processors:

```
root@solaris:~# zlogin myzone psrinfo -v
Status of virtual processor 0 as of: 12/15/2012 06:54:05
  on-line since 12/15/2012 07:25:33.
  The i386 processor operates at 2271 MHz,
    and has an i387 compatible floating point processor.
Status of virtual processor 1 as of: 12/15/2012 06:54:05
  on-line since 12/15/2012 07:25:34.
  The i386 processor operates at 2271 MHz,
    and has an i387 compatible floating point processor.
```

As you can see, the zone has *grabbed* both of the processors available in the global zone. In order to limit the availabilty to just 1 processor, you need to change the zone's configuration:

```
root@solaris:~# zonecfg -z myzone
zonecfg:myzone> add dedicated-cpu
zonecfg:myzone:dedicated-cpu> set ncpus=1
zonecfg:myzone:dedicated-cpu> end
zonecfg:myzone> exit
root@solaris:~# zoneadm -z myzone reboot
root@solaris:~# zlogin myzone psrinfo -v
Status of virtual processor 0 as of: 12/15/2012 07:12:29
  on-line since 12/15/2012 07:25:33.
  The i386 processor operates at 2271 MHz,
    and has an i387 compatible floating point processor.
```

The dedicated cpu is now invisible to all other non-global zones. You can also define a range of CPUs, such as **1-3**, in which case, when the non-global zone boots, the system will allocate 1 CPU as a minimum and 2 or 3 CPUs if they are available.

Before proceeding further, remove the dedicated CPU:

```
root@solaris:~# zonecfg -z myzone "remove dedicated-cpu"
root@solaris:~# zoneadm -z myzone reboot
```

Fair Share Scheduler

Another way of sharing resources is to use the Fair Share Scheduler. Firstly, you need to set that scheduler as the default for the system:

```
root@solaris:~# dispadmin -d FSS
root@solaris:~# dispadmin -l
CONFIGURED CLASSES
==================

SYS (System Class)
TS  (Time Sharing)
SDC (System Duty-Cycle Class)
FX  (Fixed Priority)
IA  (Interactive)
RT  (Real Time)
FSS (Fair Share)
```

Next set the FSS scheduler as the default for your zone:

```
root@solaris:~# zonecfg -z myzone "set scheduling-class=FSS"
```

Now you can give your global zone 75% of your processors leaving 25% for your zone:

```
root@solaris:~# zonecfg -z global
zonecfg:global> set cpu-shares=75
zonecfg:global> exit
root@solaris:~# zonecfg -z myzone
zonecfg:myzone> set cpu-shares=25
zonecfg:myzone> exit
```

Finally use the **prstat** command to display the CPU resource balancing:

```
   PID USERNAME  SIZE   RSS STATE   PRI NICE      TIME  CPU PROCESS/NLWP
  3725 trainee   518M  180M sleep    49    0   0:31:48 0.4% firefox/21
  3738 trainee   129M   18M sleep    59    0   0:00:37 0.4% gnome-terminal/2
 11208 daemon     14M 3380K sleep    59    0   0:00:04 0.3% rcapd/1
  1159 trainee   169M  152M sleep    59    0   0:04:51 0.1% Xorg/3
  3661 trainee   227M  149M sleep    59    0   0:03:32 0.1% java/23
     5 root        0K    0K sleep    99  -20   0:01:11 0.1% zpool-rpool/136
  3683 trainee    13M  724K sleep    59    0   0:01:25 0.0% VBoxClient/3
 12971 root       11M 3744K cpu1     59    0   0:00:00 0.0% prstat/1
 12134 netadm   5340K 2808K sleep    59    0   0:00:00 0.0% nwamd/7
 12308 root     5880K 2296K sleep    59    0   0:00:00 0.0% nscd/25
  3645 trainee   150M   35M sleep    59    0   0:00:06 0.0% gnome-panel/2
  3644 trainee   128M   15M sleep    59    0   0:00:12 0.0% metacity/1
 12400 root     3964K 1008K sleep    59    0   0:00:00 0.0% syslogd/11
  3658 trainee    61M   23M sleep    12   19   0:00:03 0.0% updatemanagerno/1
   814 root       16M 6288K sleep    59    0   0:00:05 0.0% nscd/37
   932 root       11M  600K sleep    59    0   0:00:05 0.0% VBoxService/7
   957 root       11M 1144K sleep    59    0   0:00:00 0.0% syslogd/11
   818 root        0K    0K sleep    99  -20   0:00:00 0.0% zpool-mypool/136
   637 root     9408K 1036K sleep    59    0   0:00:00 0.0% dhcpagent/1
   881 daemon   3356K    4K sleep    59    0   0:00:00 0.0% rpcbind/1
    95 netadm   4296K  680K sleep    59    0   0:00:00 0.0% ipmgmtd/6
   104 root     9692K  388K sleep    59    0   0:00:00 0.0% in.mpathd/1
    85 daemon     16M 2940K sleep    59    0   0:00:00 0.0% kcfd/3
    50 root       16M 1560K sleep    59    0   0:00:00 0.0% dlmgmtd/7
```

```
ZONEID    NPROC  SWAP   RSS MEMORY      TIME  CPU ZONE
     0      124 4095M  973M    48%   0:53:57 1.4% global
     3       28  137M   46M   2.2%   0:00:08 0.0% myzone




Total: 152 processes, 956 lwps, load averages: 0.08, 0.12, 0.12
```

Allocating Memory

Three types of memory capping are possible within a zone:

Cap	Description
Physical	Total amount of physical memory available to the zone. Once past the cap, memory pages are paged out
Locked	Amount of memory that can be allocated to a greedy application by a zone
Swap	Amount of swap space that can be used by a zone

To cap the physical memory of a zone, you need to add and correctly configure the **capped-memory** property:

```
root@solaris:~# zonecfg -z myzone
zonecfg:myzone> add capped-memory
zonecfg:myzone:capped-memory> set physical=50m
zonecfg:myzone:capped-memory> end
zonecfg:myzone> exit
root@solaris:~# zonecfg -z myzone info capped-memory
capped-memory:
    physical: 50M
```

Zone Statistics

Zone statistics can be displayed by using the **zonestat** command:

```
root@solaris:~# zonestat 5 3
Collecting data for first interval...
Interval: 1, Duration: 0:00:05
SUMMARY                   Cpus/Online: 2/2   PhysMem: 2047M  VirtMem: 3071M
                     ---CPU----  --PhysMem-- --VirtMem-- --PhysNet--
               ZONE  USED %PART  USED %USED  USED %USED PBYTE %PUSE
            [total]  0.23 11.9% 1495M 73.0% 1967M 64.0%     0 0.00%
           [system]  0.19 9.68%  309M 15.1%  891M 29.0%     -     -
             global  0.04 4.52% 1134M 55.4% 1024M 33.3%     0 0.00%
             myzone  0.00 0.10% 51.3M 2.50% 51.5M 1.67%     0 0.00%

Interval: 2, Duration: 0:00:10
SUMMARY                   Cpus/Online: 2/2   PhysMem: 2047M  VirtMem: 3071M
                     ---CPU----  --PhysMem-- --VirtMem-- --PhysNet--
               ZONE  USED %PART  USED %USED  USED %USED PBYTE %PUSE
            [total]  0.06 3.47% 1495M 73.0% 1967M 64.0%     0 0.00%
           [system]  0.02 1.00%  310M 15.1%  891M 29.0%     -     -
             global  0.04 4.80% 1134M 55.4% 1024M 33.3%     0 0.00%
             myzone  0.00 0.14% 51.3M 2.50% 51.5M 1.67%     0 0.00%

Interval: 3, Duration: 0:00:15
SUMMARY                   Cpus/Online: 2/2   PhysMem: 2047M  VirtMem: 3071M
                     ---CPU----  --PhysMem-- --VirtMem-- --PhysNet--
               ZONE  USED %PART  USED %USED  USED %USED PBYTE %PUSE
            [total]  0.07 3.83% 1494M 72.9% 1963M 63.9%     0 0.00%
           [system]  0.02 1.10%  308M 15.0%  891M 29.0%     -     -
             global  0.05 5.34% 1134M 55.4% 1020M 33.2%     0 0.00%
             myzone  0.00 0.12% 51.3M 2.50% 51.5M 1.67%     0 0.00%
```

Non-global Zone Privileges

Certain things cannot be done from within a non-global zone. The list of privileges assigned to a zone can be displayed as follows:

```
root@solaris:~# zlogin myzone ppriv -l
contract_event
contract_identity
contract_observer
cpc_cpu
dtrace_kernel
dtrace_proc
dtrace_user
file_chown
file_chown_self
file_dac_execute
file_dac_read
file_dac_search
file_dac_write
file_downgrade_sl
file_flag_set
file_link_any
file_owner
file_read
file_setid
file_upgrade_sl
file_write
graphics_access
graphics_map
ipc_dac_read
ipc_dac_write
ipc_owner
net_access
net_bindmlp
```

```
net_icmpaccess
net_mac_aware
net_mac_implicit
net_observability
net_privaddr
net_rawaccess
proc_audit
proc_chroot
proc_clock_highres
proc_exec
proc_fork
proc_info
proc_lock_memory
proc_owner
proc_priocntl
proc_session
proc_setid
proc_taskid
proc_zone
sys_acct
sys_admin
sys_audit
sys_config
sys_devices
sys_ipc_config
sys_linkdir
sys_mount
sys_iptun_config
sys_flow_config
sys_dl_config
sys_ip_config
sys_net_config
sys_nfs
sys_ppp_config
```

```
sys_res_bind
sys_res_config
sys_resource
sys_share
sys_smb
sys_suser_compat
sys_time
sys_trans_label
win_colormap
win_config
win_dac_read
win_dac_write
win_devices
win_dga
win_downgrade_sl
win_fontpath
win_mac_read
win_mac_write
win_selection
win_upgrade_sl
```

Changing a Zone's Name

To change the name of a zone, it first has to be shutdown:

```
root@solaris:~# zoneadm -z myzone halt
```

Now you can change the zone name:

```
root@solaris:~# zonecfg -z myzone "set zonename=myzone1"
root@solaris:~# zoneadm list -cv
  ID NAME             STATUS     PATH                           BRAND    IP
   0 global           running    /                              solaris  shared
```

```
   - myzone1          installed  /zones/myzone                  solaris  excl
```

Changing a Zone's Root Dataset

To change the underlying root dataset of your **myzone1** zone use the following command:

```
root@solaris:~# zoneadm -z myzone1 move /zones/myzone1
root@solaris:~# zoneadm list -cv
  ID NAME             STATUS     PATH                           BRAND    IP
   0 global           running    /                              solaris  shared
   - myzone1          installed  /zones/myzone1                 solaris  excl
```

Backing Up a Zone

Backing up a zone includes backing up the zone configuration **and** the application data in it. You can use any kind of backup software to backup data within the zone and then export it such that it may be re-injected after a zone restore. The zone configuration is backed up as follows:

```
root@solaris:~# zonecfg -z myzone1 export -f myzone1.config
root@solaris:~# cat myzone1.config
create -b
set brand=solaris
set zonepath=/zones/myzone1
set autoboot=true
set scheduling-class=FSS
set ip-type=exclusive
add anet
set linkname=net0
set lower-link=auto
set configure-allowed-address=true
set link-protection=mac-nospoof
set mac-address=random
```

```
end
add capped-memory
set physical=50M
end
add rctl
set name=zone.cpu-shares
add value (priv=privileged,limit=25,action=none)
end
```

Now backup the zone's xml file:

```
root@solaris:~# cp /zones/myzone1/root/etc/svc/profile/site/scit_profile.xml /root/myzone1.xml
```

Restoring a Zone

Disaster has struck! Uninstall and delete **myzone1**:

```
root@solaris:~# zoneadm -z myzone1 uninstall
Are you sure you want to uninstall zone myzone1 (y/[n])? y
Progress being logged to /var/log/zones/zoneadm.20121218T170820Z.myzone1.uninstall
root@solaris:~#
root@solaris:~# zonecfg -z myzone1 delete
Are you sure you want to delete zone myzone1 (y/[n])? y
```

Now restore myzone1 as follows:

```
root@solaris:~# zonecfg -z myzone1 -f myzone1.config
root@solaris:~# zoneadm -z myzone1 install -c /root/myzone1.xml
The following ZFS file system(s) have been created:
    rpool/zones/myzone1
Progress being logged to /var/log/zones/zoneadm.20121218T171621Z.myzone1.install
       Image: Preparing at /zones/myzone1/root.
```

```
 AI Manifest: /tmp/manifest.xml.9BaOQP
  SC Profile: /root/myzone1.xml
    Zonename: myzone1
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  674k/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 678.453 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 /zones/myzone1/root/var/log/zones/zoneadm.20121218T171621Z.myzone1.install
```

Log in as root and check the zone is running correctly:

```
root@solaris:~# zlogin -S myzone1
[Connected to zone 'myzone1' pts/3]
@myzone.solaris.loc:~$ ls
bin     dev     etc     export  home    lib     mnt     net     nfs4    opt     proc    root    rpool   sbin
system  tmp     usr     var
```

Cloning a Local Zone

In this section you are going to create a template zone that you can clone as necessary every time you need to install a new zone. Start by creating a zone called **cleanzone**:

```
root@solaris:~# zonecfg -z cleanzone
Use 'create' to begin configuring a new zone.
zonecfg:cleanzone> create
create: Using system default template 'SYSdefault'
zonecfg:cleanzone> set zonepath=/zones/cleanzone
zonecfg:cleanzone> set autoboot=true
zonecfg:cleanzone> verify
zonecfg:cleanzone> commit
zonecfg:cleanzone> exit
```

Install the zone:

```
root@solaris:~# zoneadm -z cleanzone install
The following ZFS file system(s) have been created:
    rpool/zones/cleanzone
Progress being logged to /var/log/zones/zoneadm.20121218T143129Z.cleanzone.install
       Image: Preparing at /zones/cleanzone/root.

 AI Manifest: /tmp/manifest.xml.vAaqcB
  SC Profile: /usr/share/auto_install/sc_profiles/enable_sci.xml
    Zonename: cleanzone
```

```
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  552k/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 797.979 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 /zones/cleanzone/root/var/log/zones/zoneadm.20121218T143129Z.cleanzone.install
```

Boot the zone to import the zone's manifest:

```
root@solaris:~# zoneadm -z cleanzone boot
```

Login to the zone and hit ↵ Enter then immediately leave the zone by using the **~.** shortcut:

```
root@solaris:~# zlogin -C cleanzone
```

To clone a zone, it first needs to be shutdown:

```
root@solaris:~# zoneadm -z cleanzone halt
root@solaris:~# zoneadm list -cv
  ID NAME             STATUS     PATH                           BRAND    IP
   0 global           running    /                              solaris  shared
   3 myzone1          running    /zones/myzone1                 solaris  excl
   - cleanzone        installed  /zones/cleanzone               solaris  excl
```

Now create a clone of **cleanzone**:

```
root@solaris:~# zonecfg -z myzone2 "create -t cleanzone"
root@solaris:~# zonecfg -z myzone2 "set zonepath=/zones/myzone2"
root@solaris:~# zoneadm -z myzone2 clone cleanzone
The following ZFS file system(s) have been created:
    rpool/zones/myzone2
Progress being logged to /var/log/zones/zoneadm.20121218T174936Z.myzone2.clone
Log saved in non-global zone as /zones/myzone2/root/var/log/zones/zoneadm.20121218T174936Z.myzone2.clone
```
