An Introduction

• Once upon a time, I used to be a UNIX systems administrator

• Now I’m the one who:
  – researches, procures, orders
  – installs, configures, patches
  – tests, validates
  – uses
  – and writes the check for all of the hardware and software
An Introduction

• I somehow just don’t seem to have much time anymore

I need to simplify!

• Need a solution that is simple, cost effective… and one that I can Google

Enter Solaris Zones

• Zones allow you to slice a single server up into separate and protected virtual machines
• Zones are the “meat” of the larger Solaris container technology
• A Solaris container is the combination of a Solaris zone with resource management features
Advanced Features

- Many advanced features and deployment options:
  - “resource pools” to allocated resources like CPUs
  - Fair-share scheduler to distribute resources
  - Zones implemented with ZFS file systems
  - Zones can run selected operating systems (branded)
  - Solaris zones can even work within other virtualization systems such as VMWare
- A lot of high-level design concepts and low-level technical tweaks are possible

But that’s not this talk…

- Solution that is simple:
  - Want something that I can implement in less than ten steps
- Solution that is cost effective:
  - Don’t want to duplicate hardware or buy additional software
- Solution that I can Google:
  - No time for courses or books
  - “Solaris Zones” reports 98,800 hits
  - High quality information can be found on the first page from reputable sources
Not So Advanced Features

- Each zone can have:
  - its own node name, virtual network interfaces, and storage assigned to it
  - a security boundary surrounding it which prevents a process associated with one zone from interacting with or observing processes in other zones
  - its own separate user list
- A zone is either the global zone or a non-global zone
- Think of a regular Solaris install as having one zone, the global zone

Solaris Zones Guidelines

- Applications that:
  - access the network and files, and performs no other I/O, should work correctly
  - require direct access to certain devices, e.g., a disk partition, will usually work, but may increase security risks
  - require direct access to some devices may not work

- DataFax can fit all of these guidelines
Why You Might Need Zones

• In the beginning there was only one…

![DataFax]

• But there might be reasons to have more than one server for DataFax

To Create Development or Test Servers

• It might be nice to have a test server as well…

![DataFax Production](DataFax) ![DataFax Development](DataFax)

• And then there were two
Development or Test Servers

• Not uncommon to have a separate production and development server for DataFax
• Some organizations require this as part of a regulatory standard process
• But keeping both servers in sync is difficult
• At least doubles the initial hardware costs

To Run Multiple DataFax Releases

• And then DataFax 3.8 came out…

• And then there were four
Multiple DataFax Releases

• Maybe you want to run legacy studies under previous releases, and new studies under the new system
• Typically some form of validation process for new releases
• May not be convenient to switch all studies completely over to the new release all at the same time

To Separate Studies or Projects

• Maybe a project doesn’t want their data somehow mixed up with that other data
• And then there were five

My data
Separate Studies or Projects

- Different studies or projects may have different security or other needs
  - Study numbers may overlap
  - Security needs, real or perceived
  - Remote access differences
  - Other issues such as usernames, processes, and permissions
- May just want to compartmentalize by study or project

To Run Different Tasks

- Maybe it would be convenient if other servers were closer to DataFax
- And then there were six
Different Servers for Different Tasks

- Not unusual to have various task-based servers
  - Web/data portal server
  - SSH/SFTP server
  - Sun Global Desktop/remote access server

- Convenient to have these “close” to DataFax

- But still want them to be separate for security and logical reasons

So where does this leave you?
And what does this cost you?

Server Cost Implications

<table>
<thead>
<tr>
<th>Multiple Servers</th>
<th>Multiple Zones</th>
</tr>
</thead>
<tbody>
<tr>
<td>– Procurement cost (research, purchase)</td>
<td>– All of the items to the left</td>
</tr>
<tr>
<td>– Installation cost (setup, configuration, testing, validation)</td>
<td></td>
</tr>
<tr>
<td>– Overhead costs (rack, power, cooling)</td>
<td>• X one server</td>
</tr>
<tr>
<td>– Maintenance (upgrades, security, administration, patches)</td>
<td>+ Minimal extra administration costs</td>
</tr>
</tbody>
</table>

• X each server
DataFax Cost Implications

<table>
<thead>
<tr>
<th>Multiple Servers</th>
<th>Multiple Zones</th>
</tr>
</thead>
<tbody>
<tr>
<td>• DataFax license X each server</td>
<td>• DataFax license X each zone</td>
</tr>
<tr>
<td>• For licensing purposes, each zone is its own computer</td>
<td></td>
</tr>
<tr>
<td>• Consistent with other applications (e.g. SAS, Oracle)</td>
<td></td>
</tr>
</tbody>
</table>

Building a Very Simple Zone

• Gather information:
  – Hostname for the zone (datafax-test)
  – Directory in the global zone where all of the zone's operating system files will be (/datafax-test)
  – IP address of the zone (192.168.1.149)
  – Name of the network device that the zone should use (ipge0)

• Use the `zonecfg(1M)` command to configure the zone

• Then the `zoneadm(1M)` to install and boot the zone
# zonecfg -z datafax-test

datafax-test: No such zone configured
Use 'create' to begin configuring a new zone.
zonecfg:datafax-test> create
zonecfg:datafax-test> set zonepath=/datafax-test
zonecfg:datafax-test> add net
zonecfg:datafax-test:net> set address=192.168.1.149
zonecfg:datafax-test:net> set physical=ipge0
zonecfg:datafax-test:net> end
zonecfg:datafax-test> commit
zonecfg:datafax-test> exit
#

## Configuring the Zone

## Installing the Zone

# zoneadm -z datafax-test install

Preparing to install zone <datafax-test>.
Creating list of files to copy from the global zone.
Copying <9123> files to the zone.
Initializing zone product registry.
Determining zone package initialization order.
Preparing to initialize <1048> packages on the zone.
   Initialized <1048> packages on zone.
Zone <datafax-test> is initialized.
The file
   </datafax-test/root/var/sadm/system/logs/install_log>
   contains a log of the zone installation.
#

## Installing the Zone
Booting the Zone

```bash
# zoneadm -z datafax-test boot

# zoneadm list
global
datafax-test

# zlogin datafax-test
Connected to zone 'datafax-test' pts/2]
Sun Microsystems Inc. SunOS 5.10 Generic January 2005
```

Using the Zone

- Use the `zlogin(1M)` command to login to the zone as root from the global zone
- Configure the system as you would any other system
  - Create local users, groups, or use various name services
  - NFS mount directories (datafax, home directories) from the global zone
  - Install DataFax or other applications
Deleting the Zone

```
# zoneadm -z datafax-test halt

# zoneadm -z datafax-test uninstall
Are you sure you want to uninstall zone datafax-test (y/[n])? y

# zonecfg -z datafax-test
zonecfg:datafax-test> delete
Are you sure you want to delete zone datafax-test (y/[n])? y
zonecfg:datafax-test> exit
```

Limitations of Solaris Zones

- Provides some software fault tolerance, but no additional hardware redundancy
- May have issues with local DataFax modems
  - Control modems via global zone, direct email to datafax@zone-name.yourcompany.com
  - Use fax service like Protus
- Most OS patches get applied to all zones, not specific zones
Limitations of Solaris Zones

- Must be careful when sharing/not sharing directories for studies and DataFax across zones
- User “datafax” might be different on each zone

Online Resources

- OpenSolaris FAQ
- Solaris online documentation:
- Wikipedia
- Sun BluePrints

And about 97,998 more places
Questions?

Darryl Pahl
Vice President,
DF/Net Research, Inc.