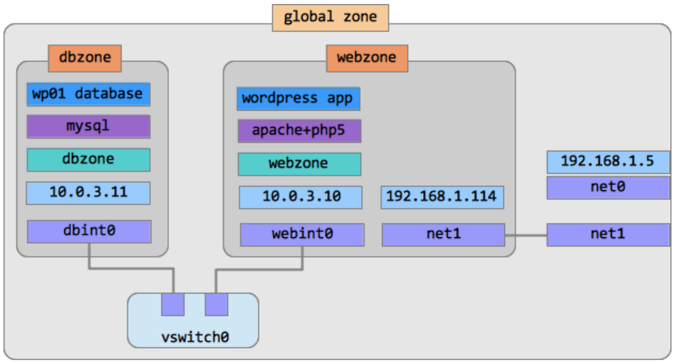
Oracle Solaris 11 Cloud in a Box

Task: You have to configure a secure multi-layered application environment in Oracle Solaris 11.

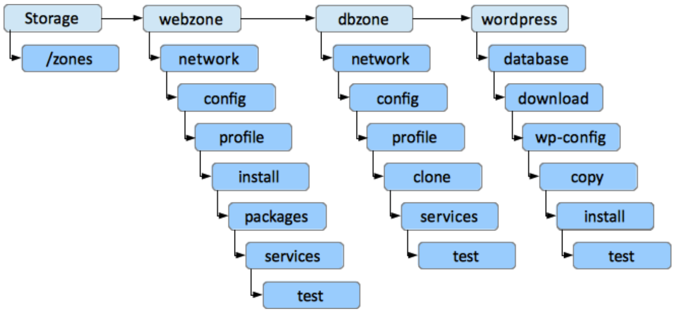
Lab: In this lab we will install a Wordpress blog application in Oracle Solaris 11 zones. We will create two Solaris zones: one for the database server and another for the web server. We will connect the zones via an internal virtual switch to make sure the database zone is connected to the web zone, but not visible from the outside world. The web zone will have two network interfaces: one for internal connection to the database zone and another for external connection.

Here is what we are going to build.



The final configuration will have a database layer, a web layer, an internal virtual network inside the Solaris box to connect them, and an outside connection for clients. It's pretty typical in today's applications and we are going to use this simple model to learn some modern Oracle Solaris 11 features. You can always extend this simple example to more complex architectures.

And here is our action plan:



We are going to perform the following steps to configure the application layer by layer.

1. Storage Layer
   - Create a ZFS file system where we will keep zones' filesystems. Mount it at `/zones`
2. Web Layer
   - Create network infrastructure for the zone: virtual switch and VNIC
   - Configure the zone with `zonecfg`
   - Create a system profile for the zone
   - Install the zone from the default package repository
   - Install the necessary packages into the zone
   - Enable the webserver (Apache) service in the zone
   - Test the webserver layer
3. Database Layer
   - Create a VNIC for the zone
   - Configure the zone with zonecfg
   - Create a system profile for the zone
   - Clone the zone from `webzone`
   - Enable the database (MySQL) service in the zone
   - Test the database layer
4. Application Layer
   - Create a database for the application
   - Download and unpack the Wordpress distribution
   - Configure the application using wp-config.php file
   - Copy the application files into the webserver directory
   - Perform the installation
   - Test the installation

Looks scary? Don't worry. You never know what you can do till you try. Let's begin!

Storage Layer

We'll start with creating a separate ZFS filesystem to store zone root directories. It will be created in our root pool and mounted at `/zones`.

```
root@solaris:~# zfs create -o mountpoint=/zones rpool/zones
```

That was all for the Storage Layer.

Web Layer

Network

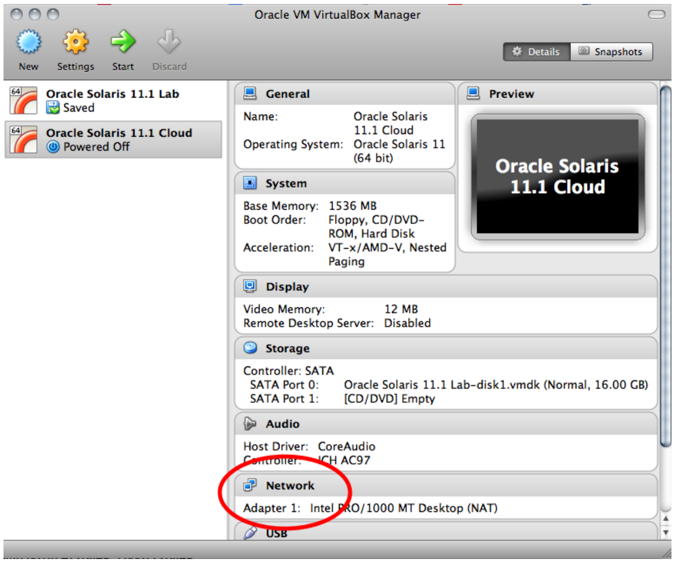
We begin with creating a network infrastructure for the zone. It will include a virtual internal switch and two network interfaces. First network interface will be for internal communication with the database zone. Second network interface will be used for outside communications.

Let's start with the outside connection. In our example we will use the second VirtualBox network interface which is called `net1` in Solaris. Check if you have two network interfaces configured in your virtual machine:

```
root@solaris:~# dladm show-phys
LINK              MEDIA              STATE      SPEED  DUPLEX    DEVICE
net0              Ethernet           up         1000   full      e1000g0
net1              Ethernet           up         1000   full      e1000g1
```

If there is only one network interface you have to *power off* your virtual machine and add the second network interface using VirtualBox Manager. Configure it as "Bridged Adapter".

Use the following screenshots as a guidance.



Oracle Solaris 11.1 Lab
Saved

Oracle Solaris 11.1 Cloud
Powered Off

General
Name: Oracle Solaris 11.1 Cloud
Operating System: Oracle Solaris 11 (64 bit)

System
Base Memory: 1536 MB
Boot Order: Floppy, CD/DVD-ROM, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging

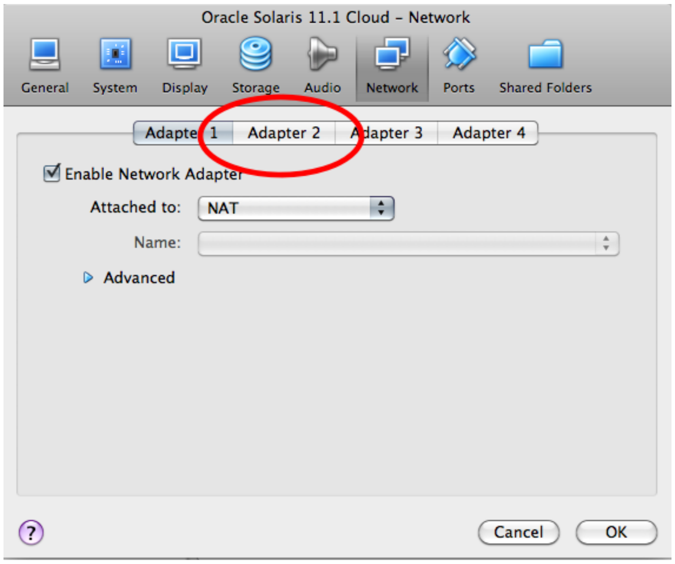
Display
Video Memory: 12 MB
Remote Desktop Server: Disabled

Storage
Controller: SATA
SATA Port 0: Oracle Solaris 11.1 Lab-disk1.vmdk (Normal, 16.00 GB)
SATA Port 1: [CD/DVD] Empty

Audio
Host Driver: CoreAudio
Controller: ICH AC97

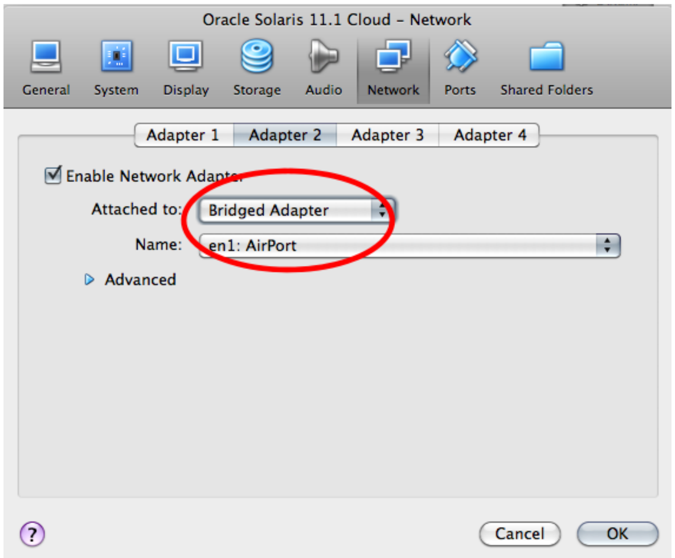
Network
Adapter 1: Intel PRO/1000 MT Desktop (NAT)

Here is only one network configured. Press "Network".



☑ Enable Network Adapter
Attached to: NAT
Name:

Only one NIC is enabled. Press "Adapter 2".



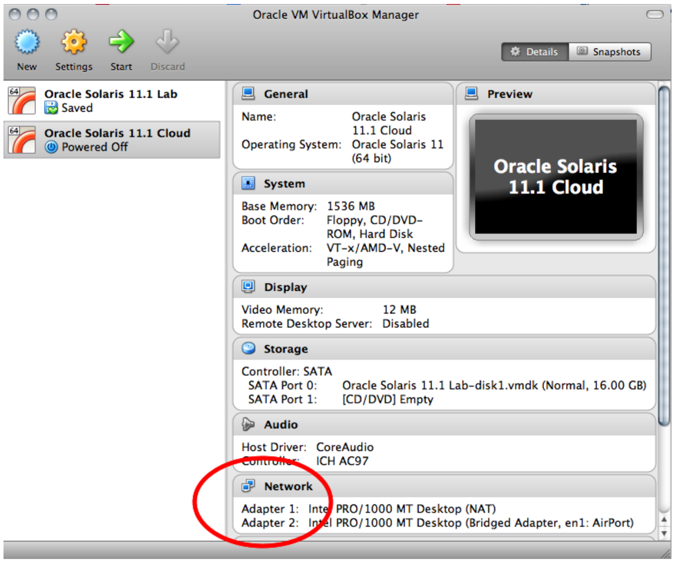
☑ Enable Network Adapter

Attached to: Bridged Adapter

Name: en1: AirPort

Advanced

Make sure you've chosen "Bridged Ethernet" for the second adapter.



Oracle Solaris 11.1 Lab
Saved

Oracle Solaris 11.1 Cloud
Powered Off

General

Name: Oracle Solaris 11.1 Cloud
Operating System: Oracle Solaris 11 (64 bit)

System

Base Memory: 1536 MB
Boot Order: Floppy, CD/DVD-ROM, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging

Display

Video Memory: 12 MB
Remote Desktop Server: Disabled

Storage

Controller: SATA
SATA Port 0: Oracle Solaris 11.1 Lab-disk1.vmdk (Normal, 16.00 GB)
SATA Port 1: [CD/DVD] Empty

Audio

Host Driver: CoreAudio
Controller: ICH AC97

Network

Adapter 1: Intel PRO/1000 MT Desktop (NAT)
Adapter 2: Intel PRO/1000 MT Desktop (Bridged Adapter, en1: AirPort)

Here is how it should look like with two networks.

After this is done, boot your Oracle Solaris 11 virtual machine again and check with 'dladm show-phys' that you have two network interfaces now.

Most likely, now net1 is already used by the global zone because it had received its IP address from your local DHCP server. Check it:

```
root@solaris:~# ipadm show-addr net1
ADDROBJ           TYPE     STATE        ADDR
net1/v4           dhcp     ok           192.168.1.114/24
net1/v6           addrconf ok           fe80::a00:27ff:fe42:ee04/10
```

To be able to use the datalink net1 in our webzone we have to disable the IP stack associated with it:

```
root@solaris:~# ipadm delete-ip net1
root@solaris:~# ipadm show-addr net1
ipadm: cannot get address information: No such interface
```

At the same time net1 still exists as datalink:

```
root@solaris:~# dladm show-link net1
LINK                CLASS     MTU    STATE    OVER
net1                phys      1500   up       --
```

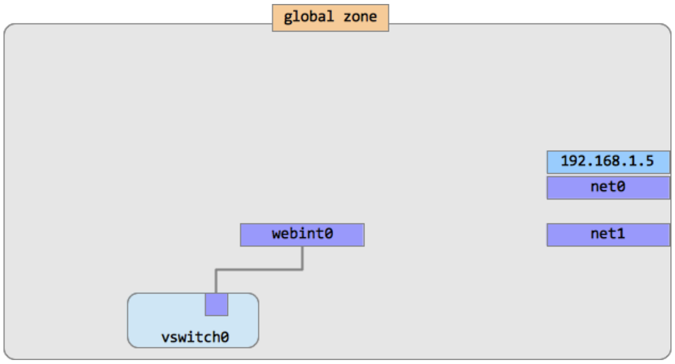
OK, net1 is ready for use by our zone. Now we will create a virtual Ethernet switch, or *etherstub*.

```
root@solaris:~# dladm create-etherstub vswitch0
```

Then we create a Virtual NIC (VNIC) for internal connections which is connected to that virtual switch.

```
root@solaris:~# dladm create-vnic -l vswitch0 webint0
```

Now we have two network interfaces for use in this zone.



Zone configuration

At this step we'll configure our web layer zone which is called `webzone`.

```
root@solaris:~# zonecfg -z webzone
zonecfg:webzone> create
zonecfg:webzone> set zonepath=/zones/webzone
zonecfg:webzone> add net
zonecfg:webzone:net> set physical=webint0
zonecfg:webzone:net> end
zonecfg:webzone> add net
zonecfg:webzone:net> set physical=net1
zonecfg:webzone:net> end
zonecfg:webzone> exit
```

Zone is configured, but not installed yet. We can install it right now, but in that case after the first boot we will have to login into it and create a system profile which includes hostname, time zone, root password etc. Not very convenient. We want to have our zone ready to use right after the first boot, so we will create its profile before the installation and then use it. This way it is also pretty easy to automate the process of future zone creation.

Zone's system profile

Use the `sysconfig` command to create a system profile for `webzone`:

```
root@solaris:~# sysconfig create-profile -o /root/web-zone-profile.xml
```

In the dialog screens set the following system's parameters:

```
Computer Name: webzone
Network connection: Automatically
Time Zone: select your time zone (hint: when selecting the country, press its first letter, e.g. U for Unites States)
Root password: solaris1
User real name: Zone User
User id: zuser
User password: oracle1
everything else: accept the defaults, just press F2 or ESC-2 to continue.
```

Zone installation

And now install the zone:

```
root@solaris:~# zoneadm -z webzone install -c /root/webzone-profile.xml
```

It will take about 10 minutes depending on your computer and network connection.

When the installation is completed we have to 1) configure the zone's networking and 2) install the necessary packages. Let's start with the networking part.

```
root@solaris:~# zoneadm -z webzone boot
...wait a couple of minutes...
root@solaris:~# zlogin -C webzone
[Connected to zone 'webzone' console]
Hostname: webzone
Apr 25 16:19:39 webzone sendmail[3606]: My unqualified host name (webzone) unknown; sleeping for retry

webzone console login: zuser
Password: oracle1 (not displayed)
Last login: Thu Apr 25 11:54:37 on console
Oracle Corporation      SunOS 5.11      11.1    September 2012
zuser@webzone:~$ su -
```

```
root@webzone:~#
```

Let's take a look at what we've got with the Automatic network configuration.

```
root@webzone:~# ipadm
NAME              CLASS/TYPE STATE        UNDER      ADDR
lo0               loopback   ok           --         --
   lo0/v4         static     ok           --         127.0.0.1/8
   lo0/v6         static     ok           --         ::1/128
net0              ip         ok           --         --
   net0/v4        dhcp       ok           --         10.0.2.16/24
   net0/v6        addrconf   ok           --         fe80::8:20ff:fe22:adfa/10
net1              ip         ok           --         --
   net1/v4        dhcp       ok           --         192.168.1.114/24
   net1/v6        addrconf   ok           --         2600:1005:b01b:3b24:a00:27ff:fe42:ee04/64
   net1/v6        addrconf   ok           --         fe80::a00:27ff:fe42:ee04/10
webint0           ip         ok           --         --
   webint0/v4     dhcp       ok           --         ?
   webint0/v6     addrconf   ok           --         fe80::8:20ff:fe35:c8f5/10
```

OK, we have `net1` configured with the IP address `192.168.1.114` it has received from our local DHCP server (e.g. you home router). Most likely we have received the DNS address as well and can connect to the outside world. Check it:

```
root@webzone:~# svccfg -s dns/client listprop config/nameserver
config/nameserver net_address 192.168.1.1
root@webzone:~# ping oracle.com
oracle.com is alive
```

We have `webint0` datalink available but it didn't receive any IP address. Why? Because it's connected only to our internal virtual switch and there is no way to get IP address automatically. Let's fix that. We will re-assign its IP address, but instead of relying on DHCP we will assign the IP address manually. Let's choose `10.0.3.0` as our internal subnet and the address will be `10.0.3.10`.

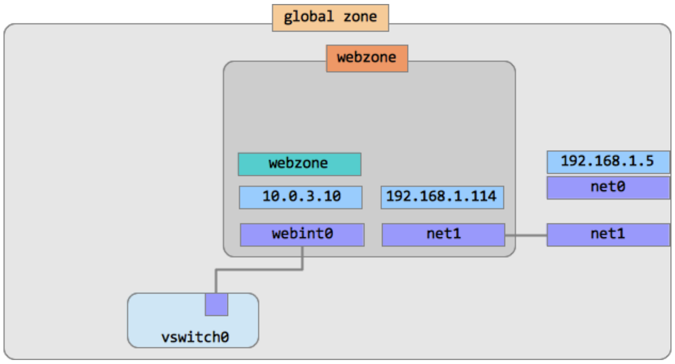
```
root@webzone:~# ipadm delete-addr webint0
root@webzone:~# ipadm create-addr -a local=10.0.3.10 webint0/v4
```

In addition to these two network interfaces we have the third one: `net0`. This is an 'automatic' virtual network interface, or 'anet' which is created every time the zone is booted. Actually you could have the zone configured without any network interfaces and still it gets connectivity by using this 'anet' virtual NIC. As you can see it has received its IP address from the internal VirtualBox's DHCP server. We don't need it for our exercise, so we can easily delete it.

```
root@webzone:~# ipadm delete-ip net0
```

To finish the networking part, let's add hostnames of our zones to `/etc/hosts`. We will need this for the Wordpress installation.

```
root@webzone:~# echo '10.0.3.10 webzone' >> /etc/hosts
root@webzone:~# echo '10.0.3.11 dbzone' >> /etc/hosts
```



Packages

Now we can install packages. We could install separate sets of packages into `webzone` and `dbzone`, but for simplicity sake we will install the full set of AMP (Apache, MySQL, PHP) packages into `webzone` and then clone it into `dbzone`. We will use the 'pkg' command which is new in Solaris 11.

```
root@webzone:~# pkg install amp
```

Check:

```
root@webzone:~# pkg list *apache*
root@webzone:~# pkg list *php*
root@webzone:~# pkg list *mysql*
```

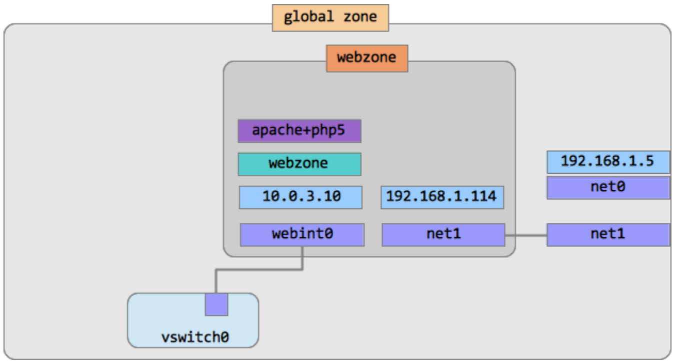
Make sure all the packages have the letter 'i' in the right column, which means "installed".

Start the web server service and check if it's running OK.

```
root@webzone:~# svcadm enable apache22
root@webzone:~# svcs apache22
STATE          STIME    FMRI
online         12:59:06 svc:/network/http:apache22
```

In your Solaris desktop open the Firefox browser and enter the webzone's external IP address (the one assigned to `net1`, in my case it's `192.168.1.114`) into the address line. The page should read **"It works!"**.

Now our `webzone` is fully configured.



```
root@webzone:~# exit
root@solaris:~# zoneadm -z webzone halt
```

Database Layer

At the next step we will create a database server zone. Instead of installing it (and waiting another 10 minutes) we will just clone our existing zone to save time and disk space. So, we repeat the first two steps (configuration and profile creation) and then replace the third one with the cloning operation. But before that we have to create a virtual NIC to use inside the zone.

Network

We want the dbzone to be connected only internally to the webzone, so we'll create a virtual NIC which is connected to the virtual switch we have created before.

```
root@solaris:~# dladm create-vnic -l vswitch0 dbint0
```

Zone configuration

As this zone will be running the database part of our application, we called it `dbzone`.

```
root@solaris:~# zonecfg -z dbzone
zonecfg:dbzone> create
zonecfg:dbzone> set zonepath=/zones/dbzone
zonecfg:dbzone> add net
zonecfg:dbzone:net> set physical=dbint0
zonecfg:dbzone:net> end
zonecfg:dbzone> exit
```

Zone's system profile

Next, we have to create the zone's profile where we configure networking, passwords, time zone etc. It's very similar to the one we have created for `webzone`. The only difference is that this time we will configure network manually.

```
root@solaris:~# sysconfig create-profile -o /root/dbzone-profile.xml
```

In the dialog screens set the following system's parameters (use F2 or ESC-2 to go from screen to screen):

```
Computer Name: dbzone
Network connection: Manually
NIC: dbint0
IP address: 10.0.3.11
Netmask: 255.255.255.0
Router: 10.0.3.10 (in our case it doesn't matter)
DNS server: 192.168.1.1 (consult your global zone setting by using the following command in another window: svccfg -s dns/client listprop
Time Zone: select your time zone (hint: when selecting a country, press its first letter, e.g. U for Unites States)
Root password: solaris1
User real name: Zone User
User id: zuser
User password: oracle1
everything else: accept the defaults, just press F2 or ESC-2 to continue.
```

Zone cloning

Now we can clone `webzone` into `dbzone`:

```
root@solaris:~# zoneadm -z dbzone clone -c /root/dbzone-profile.xml webzone
```

How long did it take? It was much faster than initial installation, wasn't it?

You may want to check the disk sizes of both zones:

```
root@solaris:~# zfs list rpool/zones rpool/zones/dbzone rpool/zones/webzone
NAME                  USED  AVAIL  REFER  MOUNTPOINT
rpool/zones           499M  8.73G    33K  /zones
rpool/zones/webzone   459M  8.73G    33K  /zones/webzone
rpool/zones/dbzone   39.4M  8.73G    34K  /zones/dbzone
```

Isn't it cool to have two zones installed using disk space of only one?

Boot the zone and then login into it to check if everything is all right.

Testing

```
root@solaris:~# zoneadm -z dbzone boot
...You may need to wait for a couple of minutes or more...
root@solaris:~# zlogin -C dbzone
[Connected to zone 'dbzone' console]
Hostname: dbzone
Apr 25 16:19:39 dbzone sendmail[3606]: My unqualified host name (dbzone) unknown; sleeping for retry

dbzone console login: zuser
Password: oracle1 (not displayed)
Last login: Thu Apr 25 11:54:37 on console
Oracle Corporation      SunOS 5.11      11.1    September 2012
zuser@dbzone:~$ su -
root@dbzone:~#
```

Check what was cloned from `webzone`. Remember, we have installed `apache`, `php` and `mysql` paskages in `webzone`? Check if they are installed here:

```
root@dbzone:~# pkg list *apache*
root@dbzone:~# pkg list *php*
root@dbzone:~# pkg list *mysql*
```

Yes, they are! But wait, there is more! Even the webserver is up and running in our cloned zone:

```
root@dbzone:~# svcs apache22
STATE          STIME    FMRI
online         13:07:06 svc:/network/http:apache22
```

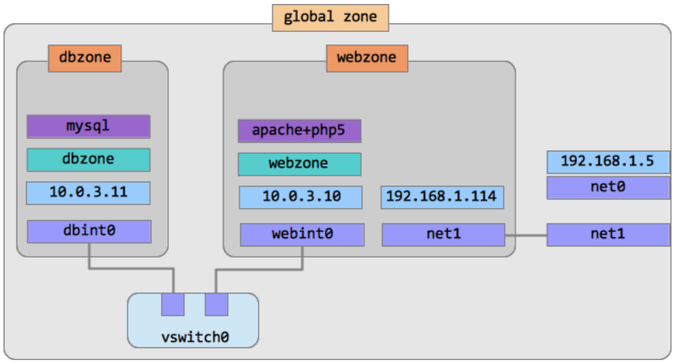
We don't need it here, in the database zone. Stop it:

```
root@dbzone:~# svcadm disable apache22
```

Check if networking is OK:

```
root@dbzone:~# ipadm
NAME              CLASS/TYPE STATE        UNDER      ADDR
dbint0            ip         ok           --         --
   dbint0/v4      static     ok           --         10.0.3.11/24
   dbint0/v6      addrconf   ok           --         fe80::8:20ff:fe5b:1e9a/10
lo0               loopback   ok           --         --
   lo0/v4         static     ok           --         127.0.0.1/8
   lo0/v6         static     ok           --         ::1/128
root@dbzone:~# ping webzone
webzone is alive
```

That's all we need--to be able to talk to `webzone`. Note that we can access `webzone` by name: that means we have cloned the `/etc/hosts` file as well. Very convenient!



Services

We have to start the database service in this zone, similarly to the web server we have started in `webzone`.

```
root@dbzone:~# svcadm enable mysql
root@dbzone:~# mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1
Server version: 5.1.37 Source distribution

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Great, the database server works too!

Wordpress

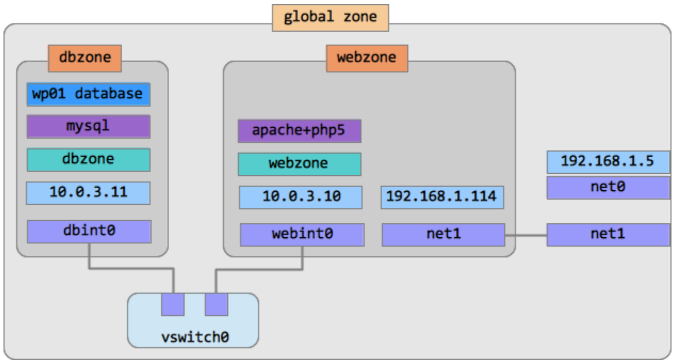
Time to install the application. While we are still in the database zone and logged into the MySQL monitor, let's create the database we need for Wordpress (we call it `wp01` here).

```
mysql> create database wp01;
```

Add access privileges for user '`wordpress`' from host '`webzone`' with password '`oracle1`':

```
mysql> grant all privileges on wp01.* to 'wordpress'@'webzone' identified by 'oracle1';
mysql> flush privileges;
mysql> quit;
root@dbzone:~#
```

That's all for database.



```
root@dbzone:~# exit
root@solaris:~#
```

Now log in to webzone, download and unpack the Wordpress distribution.

```
root@solaris:~# zlogin webzone
root@webzone:~# wget http://wordpress.org/latest.zip
root@webzone:~# unzip latest.zip
```

Now we have to enter our database parameters into Wordpress configuration. Rename the `wp-config-sample.php` file to `wp-config.php`.

```
root@webzone:~# cd wordpress
root@webzone:~/wordpress# mv wp-config-sample.php wp-config.php
```

Enter database details in `wp-config.php`

```
root@solaris:~/wordpress# vi wp-config.php
```

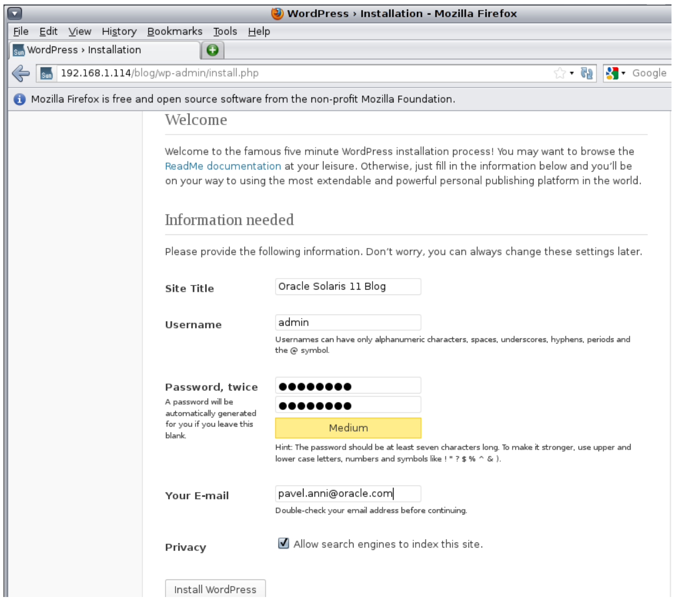
Edit the following lines:

```
// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define('DB_NAME', 'wp01');

/** MySQL database username */
define('DB_USER', 'wordpress');

/** MySQL database password */
define('DB_PASSWORD', 'oracle1');

/** MySQL hostname */
define('DB_HOST', 'dbzone');
```

Now we have to copy the Wordpress distribution to the webserver document root. We can do it from the global zone:

```
root@webzone:~/wordpress# mkdir /var/apache2/2.2/htdocs/blog/
root@webzone:~/wordpress# cp -r * /var/apache2/2.2/htdocs/blog/
```

Now your can install Wordpress from your browser, either from your global zone or from other computer in the same network. Open the browser and enter the URL `http://192.168.1.114/blog/wp-admin/install.php` (replace `192.168.1.114` with your `net1` address). The rest is easy. Enter the name of the blog, your administrator's login name ('admin' by default) and the password ('solaris1', for example).


WordPress › Installation - Mozilla Firefox
File Edit View History Bookmarks Tools Help
WordPress › Installation
Google
Install WordPress

192.168.1.114/blog/wp-admin/install.php

Mozilla Firefox is free and open source software from the non-profit Mozilla Foundation.

Welcome

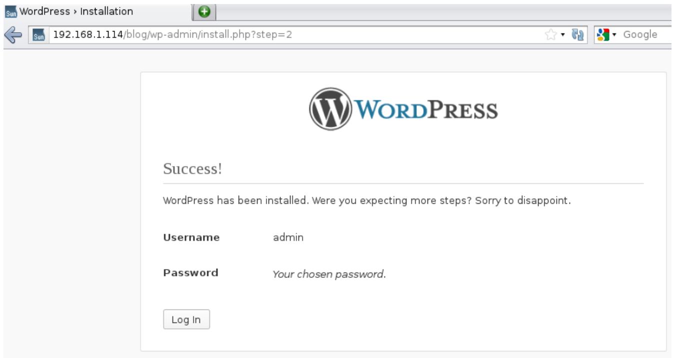
Welcome to the famous five minute WordPress installation process! You may want to browse the ReadMe documentation at your leisure. Otherwise, just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

Information needed

Please provide the following information. Don't worry, you can always change these settings later.

Site Title	Oracle Solaris 11 Blog
Username	admin Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods and the @ symbol.
Password, twice A password will be automatically generated for you if you leave this blank.	●●●●●●●● ●●●●●●●● Medium Hint: The password should be at least seven characters long. To make it stronger, use upper and lower case letters, numbers and symbols like ! " ? $ % ^ &).
Your E-mail	pavel.anni@oracle.com Double-check your email address before continuing.
Privacy	☑ Allow search engines to index this site.

Now you have to wait for a while (couple of minutes or so) until you see this:



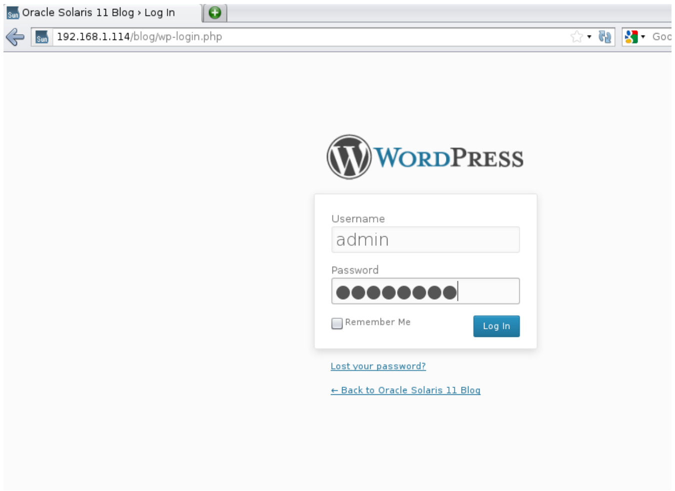
192.168.1.114/blog/wp-admin/install.php?step=2

Success!

WordPress has been installed. Were you expecting more steps? Sorry to disappoint.

Username	admin
Password	*Your chosen password.*

Now check if it works. Log in with your admin login name and password:

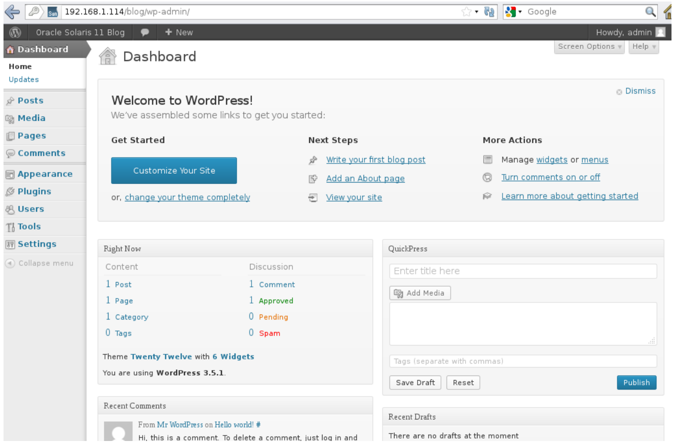

Oracle Solaris 11 Blog › Log In

192.168.1.114/blog/wp-login.php

Username: admin

Password: ●●●●●●●●

After the login you should see your blog's dashboard:


192.168.1.114/blog/wp-admin/

Welcome to WordPress!

We've assembled some links to get you started:

Right Now

Content		Discussion	
1	Post	1	Comment
1	Page	1	Approved
1	Category	0	Pending
0	Tags	0	Spam

Theme Twenty Twelve with 6 Widgets

You are using WordPress 3.5.1.

Recent Comments

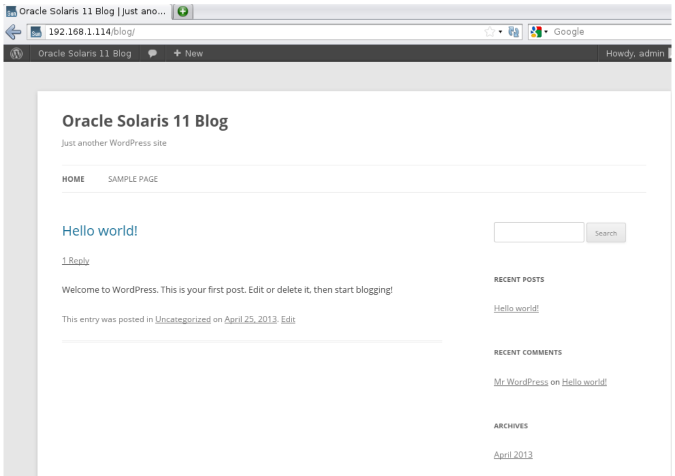
From Mr WordPress on Hello world! #

Hi, this is a comment. To delete a comment, just log in and

Recent Drafts

There are no drafts at the moment

Click on "Oracle Solaris 11 Blog" in the top left corner and you will see your blog:


Oracle Solaris 11 Blog | Just ano...

192.168.1.114/blog/

Oracle Solaris 11 Blog

Just another WordPress site

Hello world!

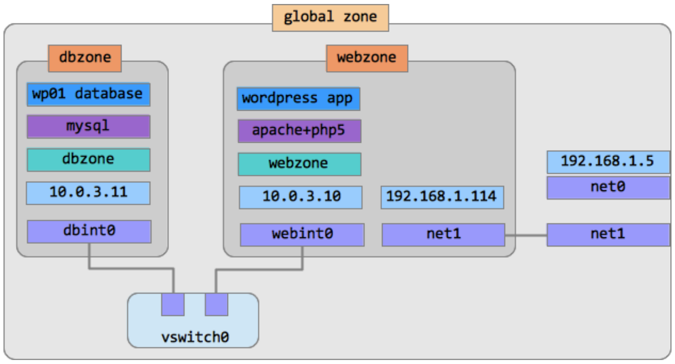
1 Reply

Welcome to WordPress. This is your first post. Edit or delete it, then start blogging!

This entry was posted in Uncategorized on April 25, 2013. Edit

Success!

Here is our final architecture:



Conclusion

In this introductory lab we have build a basic two-layer application architecture using Oracle Solaris 11 virtualization features. So far we have learned:

- How to create a ZFS file system and mount it (in one command)
- How to create a virtual switch and virtual network interface card (VNIC) using Solaris Network Virtualization.
- How to configure a zone and create its system profile and why we need that
- How to install a zone from the default Oracle Solaris IPS repository
- How to install packages using Image Packaging System (IPS)
- How to enable services in Solaris using Service Management Facility (SMF)
- How to clone a zone and inherit all the installed packages and running services
- How to connect zones using internal virtual switch and make sure they are not visible from outside
- How to create databases and assign user rights in MySQL
- How to install Wordpress application

Now you are ready to explore more advanced Oracle Solaris features. In the following labs we will learn:

- How to manage zone's resources
- How to secure a zone and protect it from unauthorized changes
- How to scale out web server zones and use Solaris load balancer
- How to configure network address translation (NAT) in `webzone` to be able to access the outside world from `dbzone`
- How to migrate a zone between hosts
- ...and many more