GlassFish ESB v2.x Field Notes
Preparing Basic JeOS Appliance for GlassFish ESB LB and HA Testing

Michael.Czapski@sun.com
December 2009, Release 1.0.0.3

Table of Contents

Introduction

It seems frequently assumed that architecting and deploying Highly Available (HA) solutions requires Application Server and/or Operating System clustering. When it comes to SOA and Integration solutions this is frequently a wrong assumption. Load Balanced (LB) and Highly Available HA) SOA and Integration solutions may not require that degree of complexity and sophistication. Testing LB and HA solutions requires infrastructure consisting of multiple hosts and the ability to "crash" hosts at will. With virtualization technologies available now it is far easier to use multiple virtual machines then to use physical machines. It is also easier and potentially less destructive to "crash" virtual machines then it is to do so with physical machines.

This note walks through the process of building a Base OpenSolaris-based VMware Virtual Appliance, based on the JeOS Prototype. It will be used as the underlying infrastructure in future Notes discussing building GlassFish ESB, Java MQ, MySQL and other appliances used in LB and HA testing. The major advantage of this infrastructure is that, as well as being fully functional, if is free (as in free beer).

At the end of the Note we will have a basic VMware Appliance, running the minimal, headless OpenSolaris Operating System (based on June 2009 JeOS Prototype), configured to use NAT networking, and ready to use as the basis of GlassFish ESB and other infrastructure. To accomplish this we will download the JeOS Prototype, configure machine name and NAT networking and test network connectivity. We will also discuss the steps required to clone this appliance.
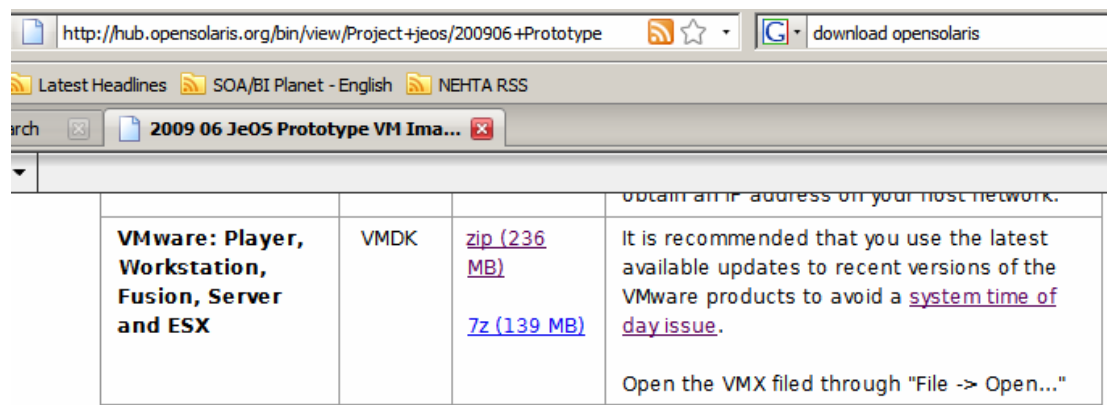
Obtain JeOS 2009/06

The OpenSolaris Project, www.OpenSolaris.org, builds the Free and Open Source OpenSolaris Operating System. This Operating System, which includes Gnome-based

Desktop and other creature comforts, is available for download from the Project site. This distribution is fairly large as it contains a lot of software which will not be used in LB and HA testing.

The distribution that will be used in the Note is built through the OpenSolaris JeOS Prototype Project, http://hub.opensolaris.org/bin/view/Project+jeos/. The JeOS Project "… is focused on delivering examples of heavily reduced, text-based, headless server-oriented forms of the Sun-managed OpenSolaris distribution such that developers and administrators can more easily realize application-tailored installations of OpenSolaris in virtual appliance, cloud and bare metal environments.". The appliance configured in this Note is based on the 2009.06 Prototype Image. This image comes in a variety of form factors, http://hub.opensolaris.org/bin/view/Project+jeos/200906+Prototype. This Note discusses configuration of the VMware image.
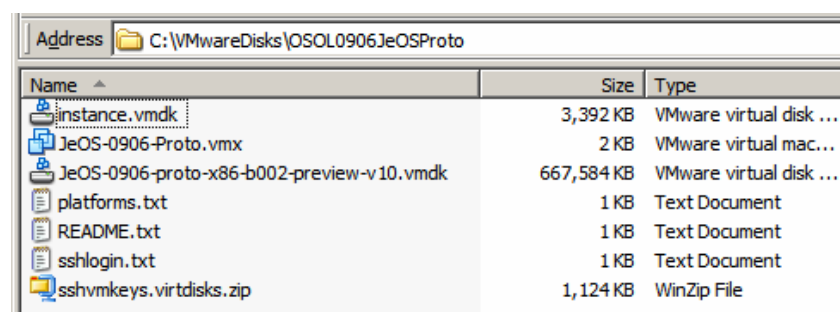
Let's download the 2009.06 JeOS VMare image from the JeOS Prototype Download page: http://hub.opensolaris.org/bin/view/Project+jeos/200906+Prototype.

Direct link, http://dlc.sun.com/osol/jeos/2009/osol-0906-jeos-proto-vmware.zip, points to the `osol-0906-jeos-proto-vmware.zip` archive.

Unzip the archive into a suitable directory. I used `C:\VMwareDisks`. Directory `OSOL0906JeOSProto` was created by the Unzip process.

Please note that this distribution gives a minimal, headless installation. There is no graphical environment. All work needs to be done through the console window or using a SSH Client.

VMware Environment

The Host environment for this Note is Windows XP SP3. I see no reason why the discussion should not be applicable to any other platform on which VMware Player, VMware Workstation or VMware Server of one sort or another is supported.

If you have a VMware Player 2.5 or better, VMware Worksstation 5.5 or better or a recent VMware Server infrastructure (other then ESX) you are ready to boot the VM. If not, one of the VMware infrastructure components is required.

I use VMware Player 3.0 for this work. It can be downloaded from http://www.vmware.com/products/player/. It is a free download, though it requires registration and responses to a questionnaire. The Player is available for 32 and 64 bit Windows and 32 and 64 bit Linux. See FAQ for additional information if this is of interest to you: https://www.vmware.com/products/player/faqs.html. See VMware Player 3.0. See Release Notes for supported Hosts and Guests: http://www.vmware.com/support/player30/doc/releasenotes_player3.html.

Installation of VMware Player will require re-boot on Windows.

Re-configure JeOS VM Image

This discussion is VMware Player 3.0-specific. Older versions of VMware Player don't support GUI-based reconfiguration. VMware Workstation and VMware Servers do. To re-configure the VM for VMware Player 2.5 or earlier consider using the VMX Builder tool, http://vmxbuilder.com/.

Start VMware Player, "File" → "Open Virtual Machine…", locate the configuration file, `JeOS-0906-Proto.vmx`, in the directory to which JeOS was unzipped.

Click the "Edit Virtual Machine Settings" link.
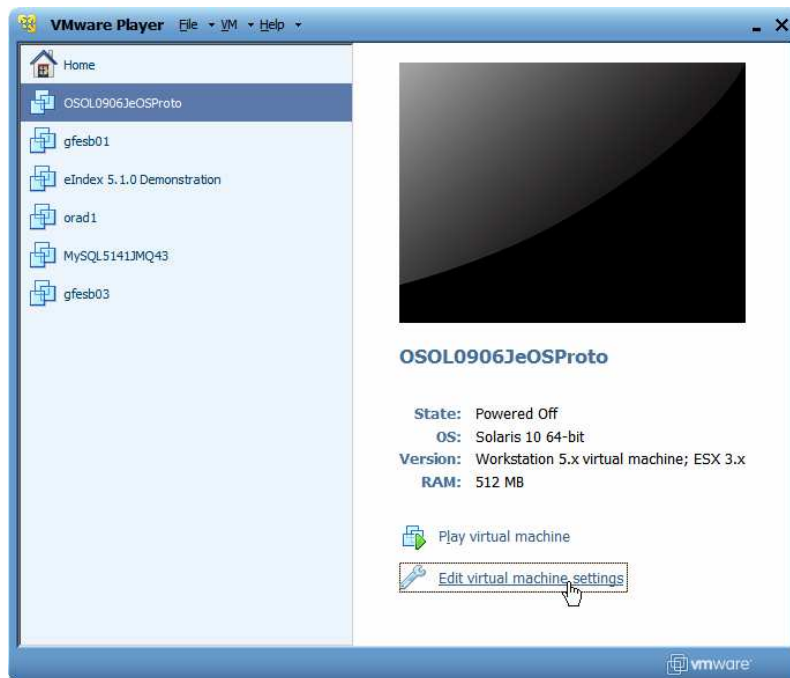
Increase the amount of memory allocated to the machine if you have sufficient host memory. Changes the number of processors the machine is to use, if you have more then one. Change the networking from Bridged to NAT.
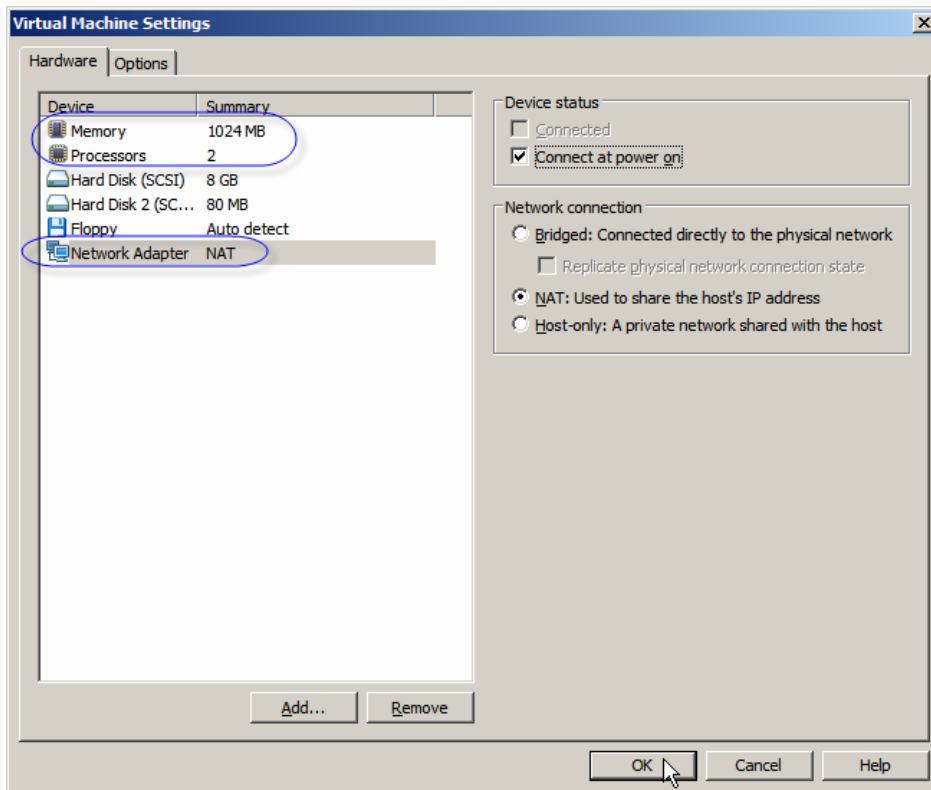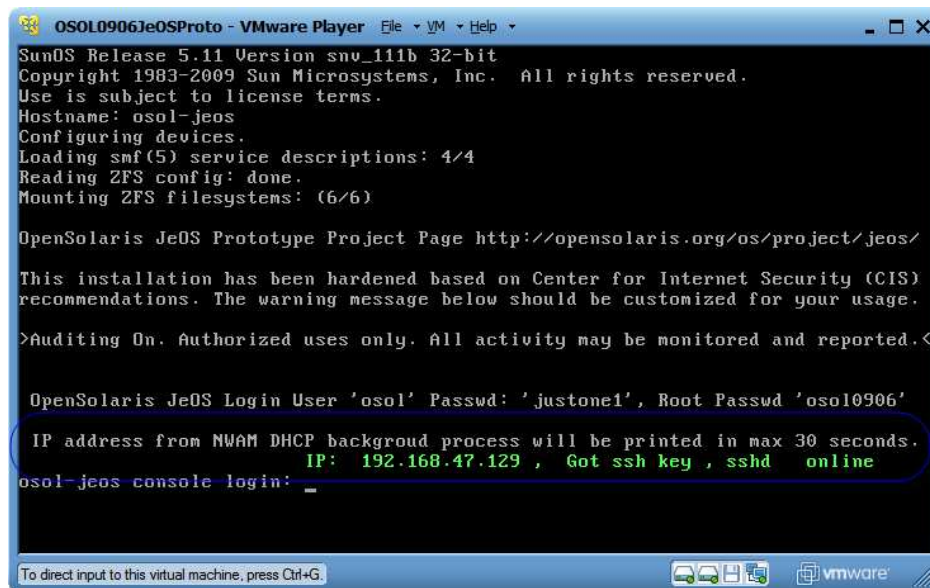

Make any other changes you may need to make, for example to switch from 64-bit to 32-bit architecture if this is the host architecture.

Click the "Play the virtual machine" link. When the GRUB menu appears choose the first option.
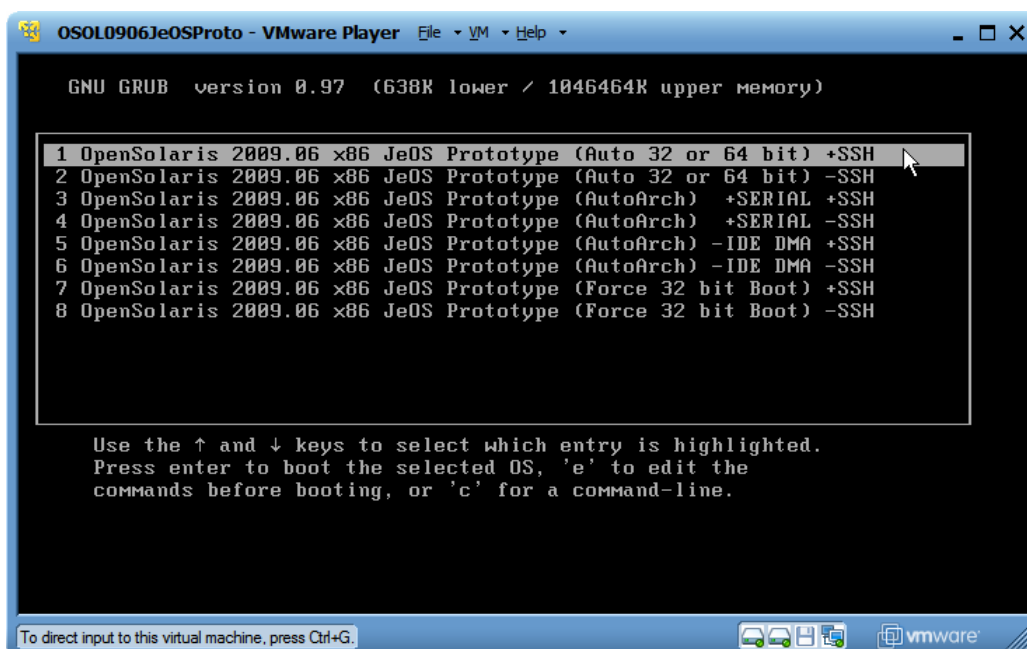
Wait for the machine to boot and display its DHCP-acquired IP address. Note down this address, for me 192.168.47.129. We will use it later to configure the PuTTY SSH Client.

Log in as osol/justone1

As root, using pfexec, edit the GRUB boot menu to remove "Press any key to continue" screen by deleting the two lines indicated in the file.

```
$ pfexec vi /rpool/boot/grub/menu.lst
```


Reduce the number of options in the boot menu by commenting out or deleting those boot entries that you don't need.

Reduce boot timeout value by setting menu timeout to a lower value, for example 5 seconds.

```
$ pfexec bootadm set-menu timeout=5
```

Reboot the machine to test the changes.

```
$ pfexec reboot
```

Configure SSH Terminal Client

While the JeOS can be managed through the local Console, remote access to the standard JeOS installation can be gained using SSH as user osol. SSH root access is disabled. SSH access will make it easier to work with the machine.

I use PuTTY, Release 0.6 for Windows, http://www.chiark.greenend.org.uk/~sgtatham/putty/. Feel free to use a SSL Client of your choice. The configuration discussion in this section assumes the use of PuTTY. If you use a different SSH client you will need to figure this out for yourself.

Assume PuTTY is installed to "C:\Program Files\PuTTY".
Assume JeOS is unzipped to "C:\VMwareDisks\OSOL0906JeOSProto".

To configure access to the JeOS for PuTTY it is necessary to convert the JeOS distribution-provided OpenSSH sample key, sshvmrsa.key, to the PuTTY ppk format using the PuTTY puttygen tool.

Extract the OpenSSH key, sshvmkeyrsa.key, from the JeOS-provided archive sshvmkeys.virtdisks.zip, to the same directory whether the archive is located. Run the PuTTY puttygen application. Click the Load button.
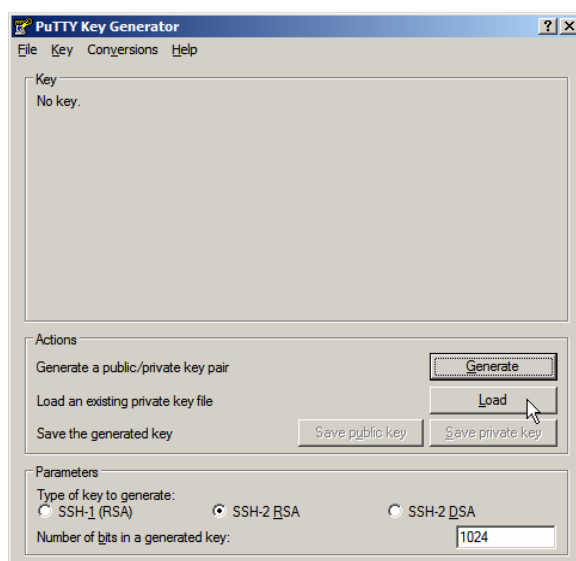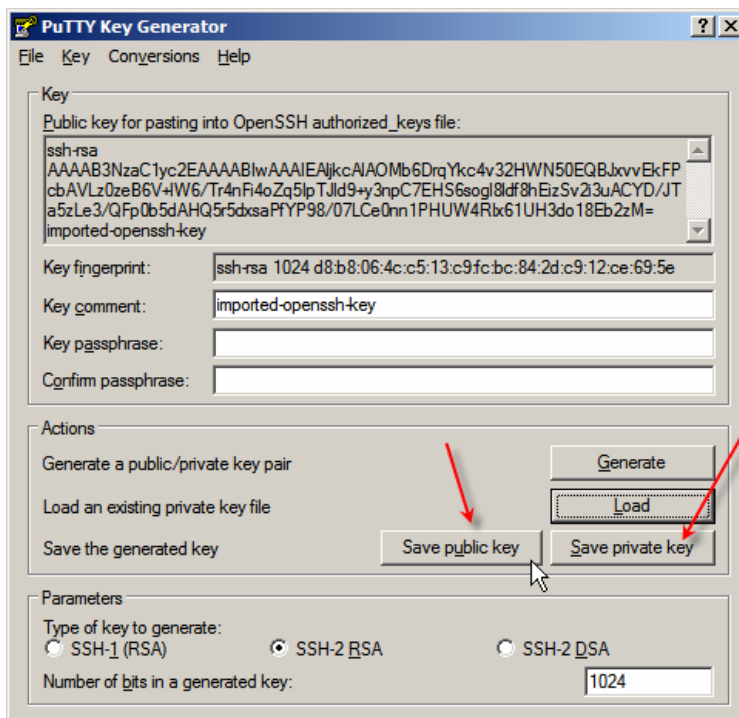

Locate the sshvmkeyrsa.key, which was just extracted to the JeOS directory.
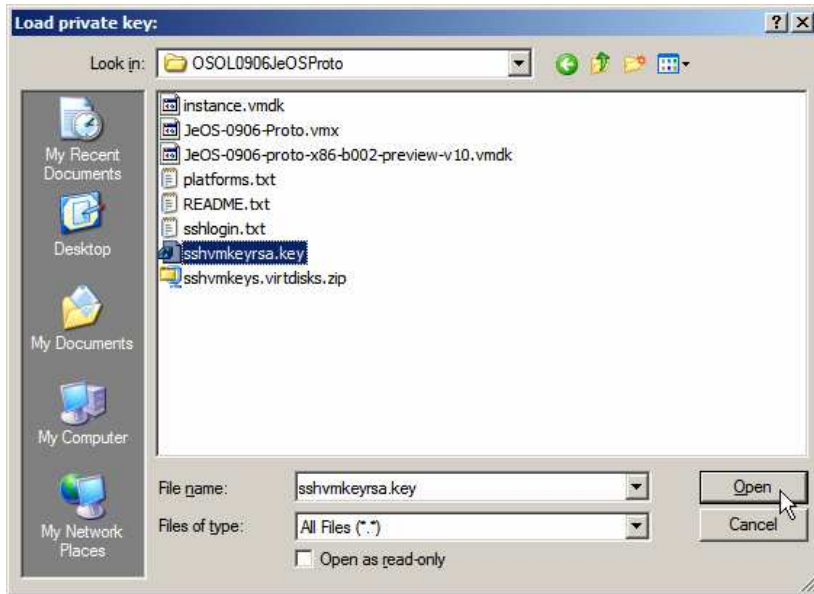
A dialogue box will confirm successful import.
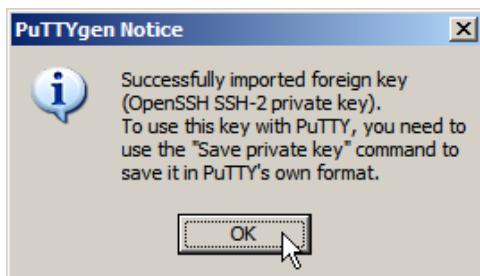
Click "Save public key" and "Save private key" buttons to save the keys in PuTTY-acceptable format.

Name the public key something like "sshvmkeyrsa.key.pub" and the private key something like "sshvmkeyrsa.key.ppk", ignoring the security warning. This Note is intended to get a VM configured, running and accessible as easily as possible. Security implications and other matters are beyond the scope of this note.

Exit puttygen once both keys are saved.

Start PuTTY to create a Session Document, which save session configuration for future sessions.

Enter Host Address and session name. Click the Save button.
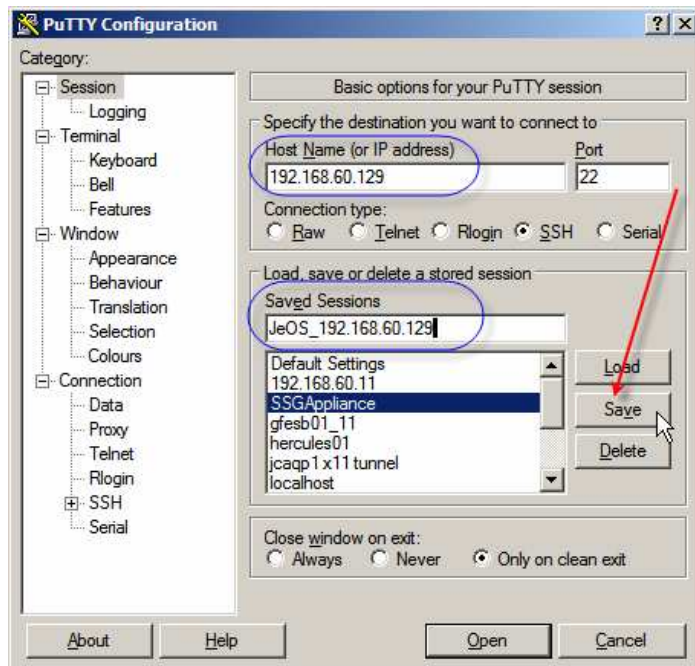
Expand the SSH → Auth configuration category, Browse to where the sshvmkeyrsa.key is located and select it.
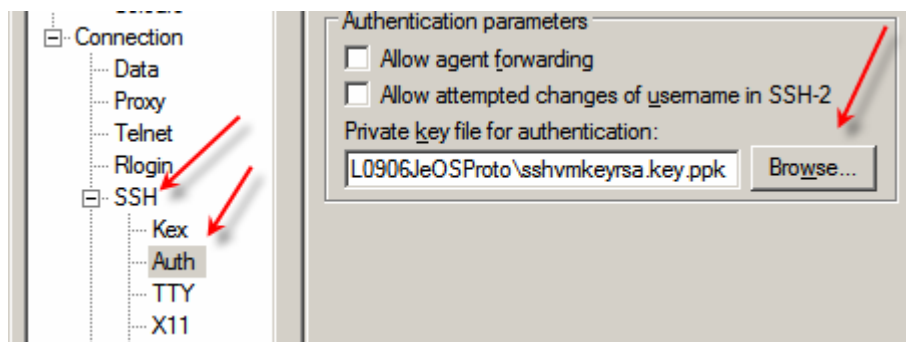
Select the Session category and click the Save button to save configuration again.

Click the Open button.

The first time PuTTY connects to the virtual machine an Alert dialogue box may appear.

Provide the username of osol to log in.

Reconfigure PuTTY session appearance as you see fit.

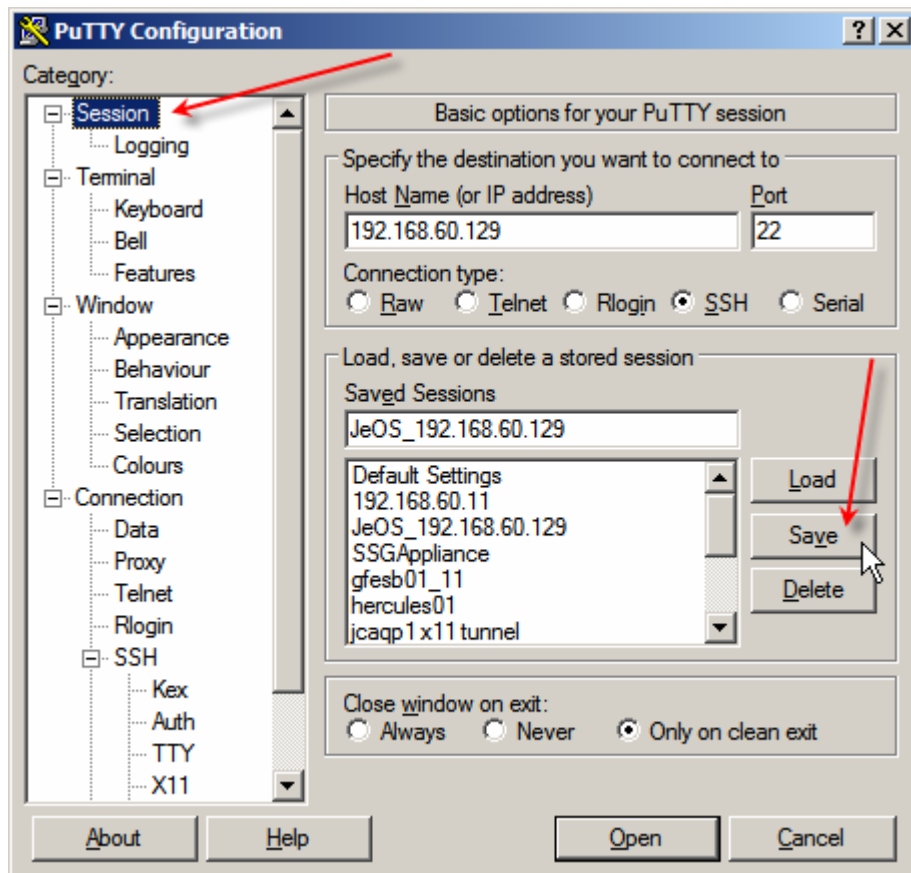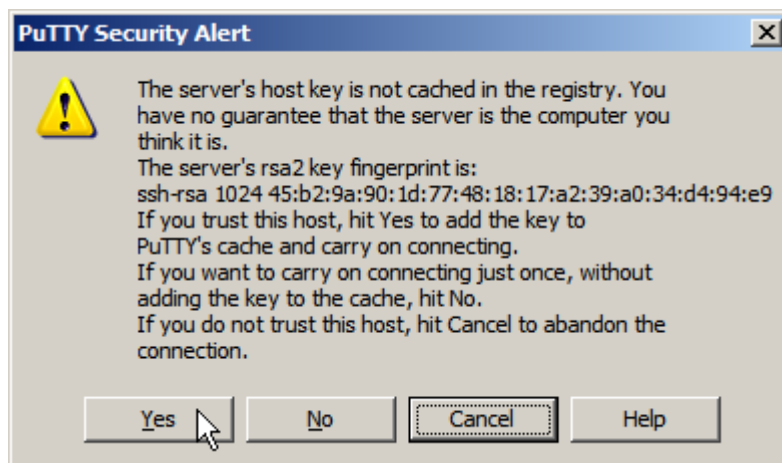On Windows, create a shortcut to a command similar to the following, replacing the host address and session name as appropriate:

```
"C:\Program Files\PuTTY\putty.exe" -load "JeOS_192.168.47.129" osol@192.168.47.129
```

This shortcut will start PuTTY with the correct saved session, the correct username and the correct (at least for a while) address.

Configure SSH SFTP/SCP Client

At some point \one might need to exchange files between the host and the guest. While the ftp client in the JeOS can used for this, there is a need to have an FTP server on the host. Since the JeOS guest already has a SSH server it is more convenient, perhaps, on the host, to use the SCP to transfer files to/from guest.

I use WinSCP, Version 4.2.4 for Windows, http://winscp.net/eng/index.php. Feel free to use another SSH file transfer client.

Assume WinSCP is installed to "C:\Program Files\WinSCP3".
Assume JeOS is unzipped to "C:\VMwareDisks\OSOL0906JeOSProto".

Start WinSCP and click New button.
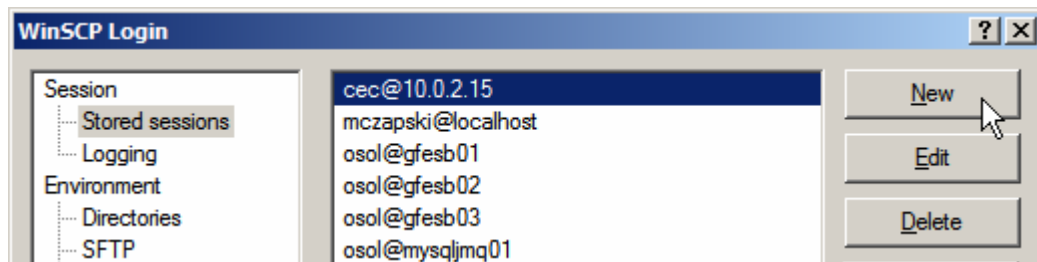
Enter host name/address, username: osol, password: justone1 and browse to the keyfile, sshvmkeyrsa.key.ppk, in the JeOS directory.
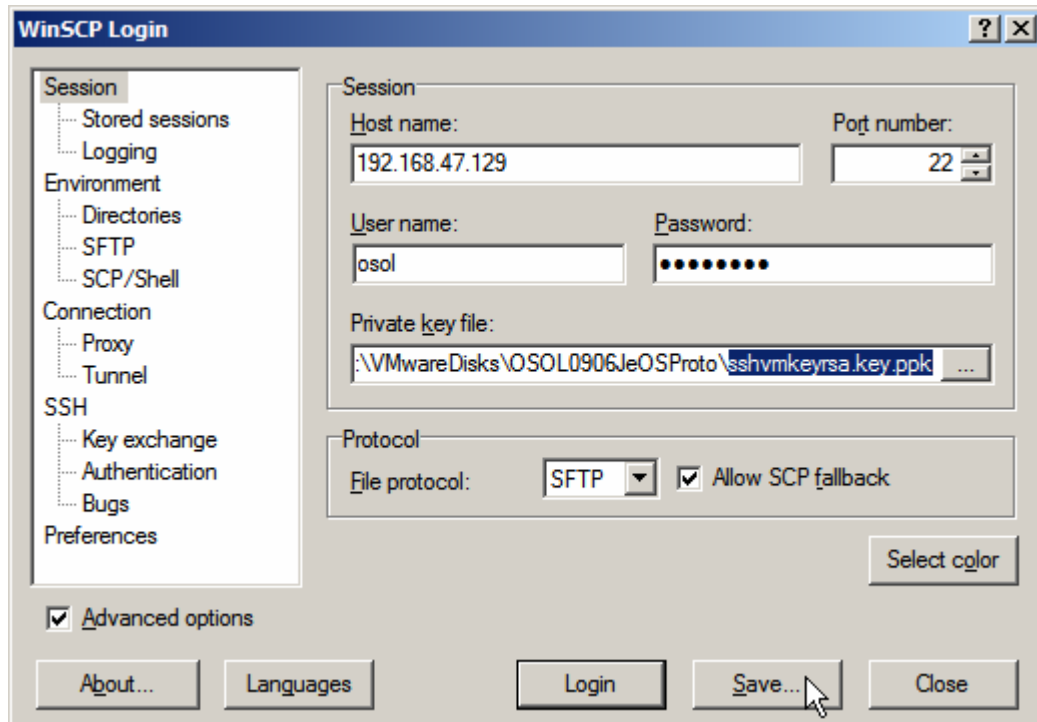
Click the Save button and save the session properties.
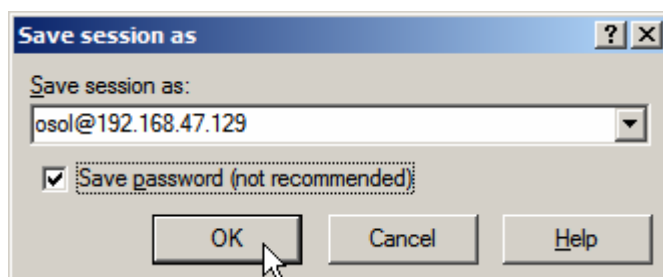
Choose the session and click Login.

You may get an Alert dialogue warning of a potential security breach. Choose to Add the key.

Now we can conveniently transfer files between Windows and JeOS using WinSCP.

Increase swap space on Guest

Almost inevitably the VM will run out of swap space once anything of substance gets installed, for example GlassFish ESB runtime. To anticipate this add an extra swap file using commands like the following, where the extra swap space will be 1G.

```
pfexec swap -sh # find out how big your swapdevice is
pfexec zfs get volsize rpool/swap # find out how big your ZFS swapdevice is
pfexec zfs set volsize=2G rpool/swap # enlarge it to (eg:) 2G, 2048M
pfexec reboot
```

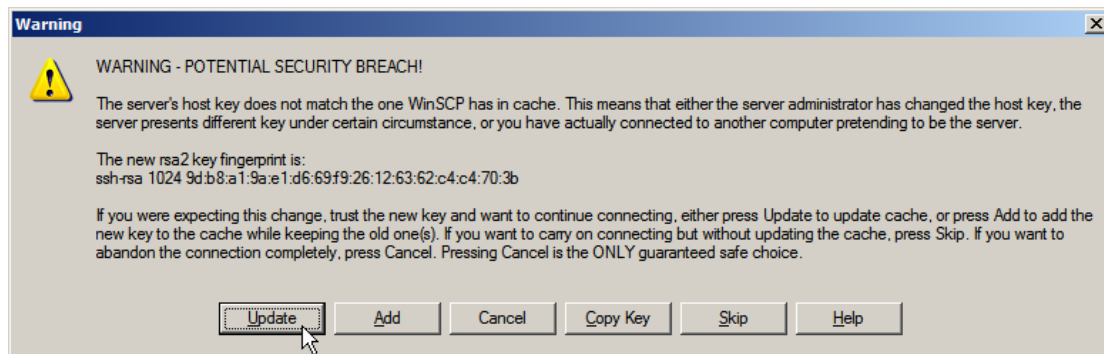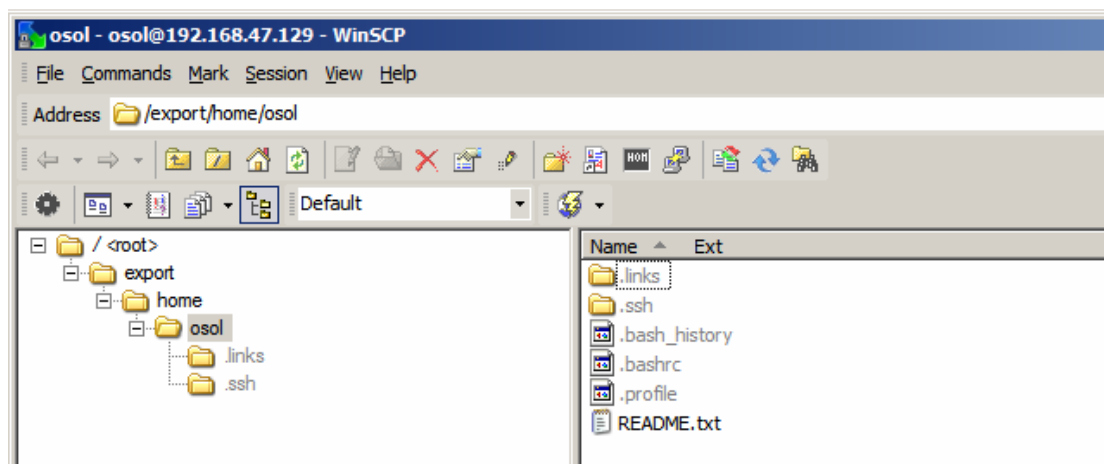# Install JDK 1.6.0_16 on Guest

In most cases this virtual machine will be used to host software which requires Java Virtual Machine to run, and possibly, the JDK. Let's download JDK 1.6.0_16 for Solaris x86 or x64 (depending on your host platform architecture) from the Sun download site, https://cds.sun.com/is-bin/INTERSHOP.enfinity/WFS/CDS-CDS_Developer-Site/en_US/-/USD/ViewProductDetail-Start?ProductRef=jdk-6u16-oth-JPR@CDS-CDS_Developer, to a convenient directory.

In this Note the shell variant is used - http://cds.sun.com/is-bin/INTERSHOP.enfinity/WFS/CDS-CDS_Developer-Site/en_US/-/USD/VerifyItem-Start/jdk-6u16-solaris-i586.sh?BundledLineItemUUID=uR9IBe.oEgEAAAEl_lsxcEip&OrderID=.7JIBe.oL

Start a WinSCP session to the VM. Create a new directory called "downloads" in osol user's home directory.
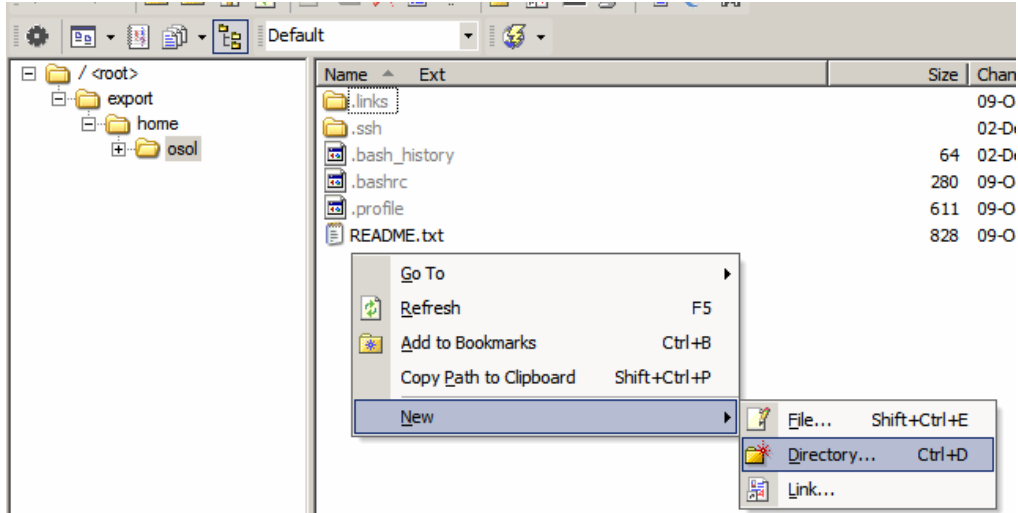
Copy the JDK download to the "downloads" directory.
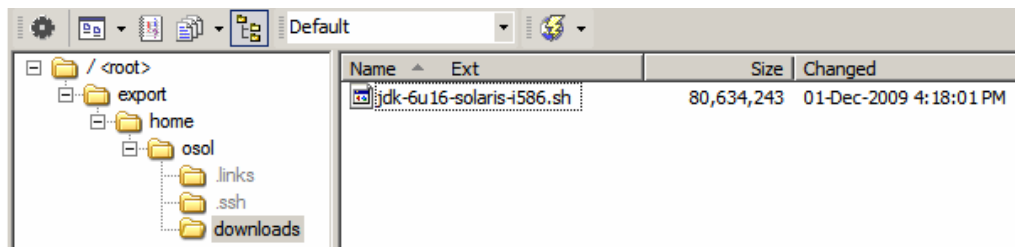
Terminate the WinSCP session and start a SSH Client (PuTTY) session to the virtual machine.

Execute the following in the SSH Client window:

```
cd /
pfexec /bin/sh /export/home/osol/downloads/jdk-6u16-solaris-i586.sh
```

Answer yes to the License question. Once the installation is finished execute the following command to verify installation:

```
/jdk1.6.0_16/bin/java -version
```

Edit osol user's .bashrc and add the path to the JDK bin directory to the osol user's path.

Shut down the machine using a command like:

```
pfexec shutdown -y -i0 -g0
```

The VM image is now ready for cloning.

Clone the VM Image

What we have so far is a template base JeOS appliance image which can be cloned as needed.

Let's create clone called gfesbv21, with the host name of gfesbv21, based on the template.

Create a directory called gfesbv21 in a convenient location.
Copy all file form directory OSOL0906JeOSProto to that new directory.
Rename the following:

From	To
JeOS-0906-proto-x86-b002-preview-v10.vmdk	gfesbv21.vmdk
JeOS-0906-Proto.vmsd	gfesbv21.vmsd
JeOS-0906-Proto.vmx	gfesbv21.vmx
JeOS-0906-Proto.vmxf	gfesbv21.vmxf
instance.vmdk	instance.vmdk

Using a text editor, for example Notepad, edit gfesbv21.vmx and change the following strings:

From	To
JeOS-0906-proto-x86-b002-preview-v10.vmdk	gfesbv21.vmdk
OSOL0906JeOSProto	gfesbv21
JeOS-0906-Proto.vmxf	gfesbv21vmxf

Delete or comment out (using # as a comment line marker) the lines starting with the following:

```
ethernet0.generatedAddress
uuid.location
uuid.bios
```

Save the gfesbv21.vmx configuration file.

Use this configuration file to open the virtual machine in the VMware Player.

Take note of the assigned IP address.

```
OpenSolaris JeOS Login User 'osol' Passwd: 'justone1', Root Passwd 'osol0906'

 IP address from NWAM DHCP backgroud process will be printed in max 30 seconds.
                      IP:  192.168.47.130 ,  Got ssh key , sshd   online
osol-jeos console login: _
```

Create a new SSH Client session document to point to this machine or use the Console to modify host name in various startup files.

Log in as user osol.

As root, edit /etc/nodename and set the new host name there:

`pfexec vi /etc/nodename`

As root, edit /etc/hosts and set host name there. Make sure that the non-FQDN host name <u>does not</u> resolve to 127.0.0.1

`pfexec vi /etc/hosts`

By default the hosts file has the following content:

```
# Internet host table
#
::1 osol-jeos osol-jeos.local localhost loghost
127.0.0.1 osol-jeos osol-jeos.local localhost loghost
~
```

Change it to read:

```
# Internet host table
#
::1 gfesbv21.local localhost loghost
127.0.0.1 gfesbv21.local localhost loghost
192.168.47.130   gfesbv21
~
```

192.168.47.130 is the DHCP address that was displayed on the Console at boot. This address is likely to be retained until you remove the assigned `ethernet0.generatedAddress` entry from the vmx file or reinstall VMware infrastructure. Add a fully-qualified domain name variant, for example gfesbv21.aus.sun.com, following gfesbv21, if you know what your domain name is, if you care for DNS-based name resolution and if sendmail host name errors don't concern you.
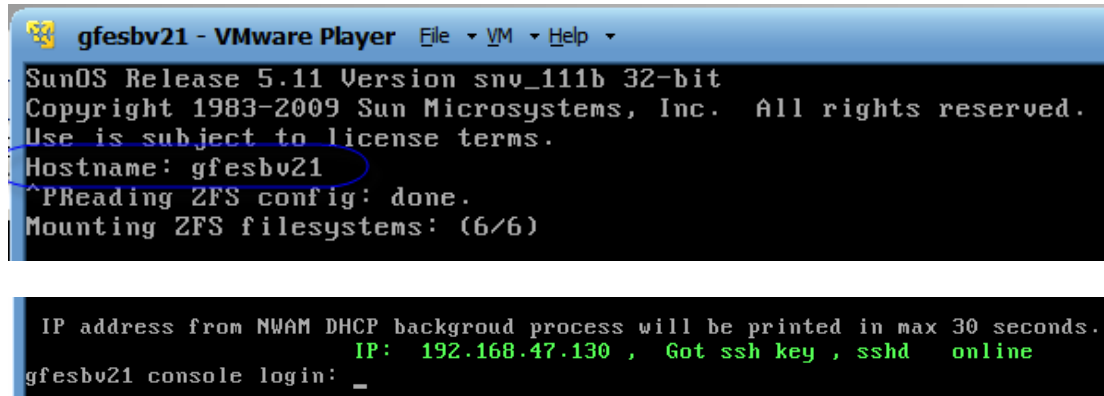
As root, create a file `/etc/hostname.e1000g0` and add to it a single line containing the literal "`inet gfesbv21`", where gfesbv21 is the host name

```
pfexec vi /etc/hostname.e1000g0
```

Reboot the virtual machine, for example using this command:

```
pfexec reboot
```

Notice the changed host name and the same IP address:

The JeOS clone, gfesbv21, is ready.

Use this method to create as many clones as you need for LB and HA testing, and whatever else you need a basic OpenSolaris-based server image.

When the IP Address Changes

VMware generates the MAC address for the VM image. DHCP infrastructure "remembers" this MAC address and endeavors to issue the same IP address each time a vm image is booted. If you re-install VMware Player, or in similar circumstances, the next time you boot your vm image it may be issued a different IP Address. This tends to cause problems as /etc/hosts has the IP Address embedded.

As root, edit /etc/hosts and change the IP Address to what the console shows at boot, if the two are different.

Summary

This note walked through the process of building a Base OpenSolaris-based VMware Virtual Appliance, based on the JeOS Prototype. To accomplish this we downloaded the JeOS Prototype, configured machine name and NAT networking and tested network connectivity. We also discussed the steps required to clone this appliance.

We now have a basic VMware Appliance template, running the minimal, headless OpenSolaris Operating System (based on June 2009 JeOS Prototype), configured to use NAT networking, and ready to use as the basis of GlassFish ESB and other infrastructure. We also have a single clone of the template, ready for installation of additional software.