Murano Deployment Guide

Murano Deployment Guide

v0.1

Publication date 2013-04-04
Copyright © 2013 Mirantis, Inc.

Abstract

This document is intended for individuals who wish to prepare development environment for Murano.

Table of Contents

List of Tables

Chapter 1. System Configuration

Table 1.1. Hardware configuration

HDD	2 x 500 GB (2 x 320 GB min) with software RAID
CPU	4 core
RAM	8 GB

Table 1.2. Software configuration

OS	Ubuntu Server 12.10

Software RAID Configuration

Software RAID is used to provide additional redundancy to the host system and add some performance for the OpenStack. The configureation is the following:

```
$ lsblk -i
NAME      MAJ:MIN RM   SIZE RO TYPE  MOUNTPOINT
sda        8:0     0 465.8G  0 disk
|-sda1     8:1     0   7.5G  0 part  [SWAP]
|-sda2     8:2     0 279.4G  0 part
| `-md1    9:1     0 558.8G  0 raid0 /opt
`-sda3     8:3     0 178.8G  0 part
  `-md0    9:0     0 178.7G  0 raid1 /
sdb        8:16    0 465.8G  0 disk
|-sdb1     8:17    0   7.5G  0 part  [SWAP]
|-sdb2     8:18    0 279.4G  0 part
| `-md1    9:1     0 558.8G  0 raid0 /opt
`-sdb3     8:19    0 178.8G  0 part
  `-md0    9:0     0 178.7G  0 raid1 /
sr0       11:0     1  1024M  0 rom
```

/ets/fstab should be modified in order to allow the system use both swap partitions simultaneously: add **pri=0** option

```
# swap was on /dev/sda1 during installation
UUID=c8a3ca9e-2942-4cf0-bbba-dfa6aa631b55    none    swap    sw,pri=0   0    0
# swap was on /dev/sdb1 during installation
UUID=4d7254b9-1461-424d-b129-a8cdb31fbf17    none    swap    sw,pri=0   0    0
```

Warning

Please **DO NOT** store any valuable data under the **/opt** folder as it is located at RAID1 partition. It provides better performance but it's also vulnerable to any single drive fault.

Chapter 2. Prepare Reference Windows Image

Table 2.1. Server-side

Prerequisite	Link
Ubuntu Server with KVM installed	Ubuntu Server 12.04.1 amd64 [http://releases.ubuntu.com/12.04.1/ubuntu-12.04.1-server-amd64.iso]
VirtIO drivers for Windows (.iso file, ~44 MB)	VirtIO Windows Drivers [http://alt.fedoraproject.org/pub/alt/virtio-win/latest/images/bin/]
Windows Server 2012 installation image (.iso file, ~3.5 GB)	Windows Server 2012 Evaluation [http://care.dlservice.microsoft.com/dl/download/6/D/A/6DAB58BA-F939-451D-9101-7DE07DC09C03/9200.16384.WIN8_RTM.120725-US-HRM_SSS_X64FREE_EN-US_DV5.ISO?lcid=1033&cprod=winsvr2012rtmisotn_p]

Table 2.2. Workstation-side

Prerequesite	Link
VNC client	Depends on your workstation system

Prepare the system

Install KVM

Nothing special, just a typical KVM installation

```
apt-get install kvm qemu-kvm
```

Install and configure SAMBA

The SAMBA servier will be used as an untermediate storage for prerequesites and scripts that will be copied to the VM.

Install samba server

```
apt-get intall samba
```

Next, let's create the folder where we will store files

```
mkdir -p /opt/samba/share
chown nobody:nogroup /opt/samba/share
```

Edit the /etc/samba/smb.conf

```
...
```

```
[global]
...
security = user
...
[share]
comment = Deployment Share
path = /opt/samba/share
browsable = yes
guest ok = yes
guest account = nobody
read only = no
create mask = 0755
```

Restart the samba services

```
restart smbd && restart nmdb
```

Get all the files

Create a directory where all installation files will be stored. For example, let's use folder **/opt/kvm**.

Copy ISO files to that folder, and then change directory to that folder.

Note

Be sure that MD5 checksums are correct for all ISO images!

Next, rename ISO files for convinience. I assume that Windows Server installation ISO file is renamed to **ws-2012.iso** and VirtIO drivers ISO file is renamed to **virtio-win.iso**.

Open the folder shared with the samba package (**//host-server-ip/share**). Create subfolder Files. Copy the following files to that folder

File	URL for download
Sysinternals Suite	Sysinternals Suite web site
Far file manager	

Start VM installation

Now let's create hard disk image file with RAW format as it allows better performance.

```
kvm-img create -f raw ws-2012.img 40G
```

And start the VM

```
sudo kvm \
  -m 2048 \
  -drive file=ws-2012.img,if=virtio \
  -cdrom ws-2012.iso \
  -drive file=virtio-win.iso,index=3,media=cdrom \
  -boot d \
  -net user \
  -net nic,model=virtio \
```

```
-redir tcp:3389::3389 \
-nographic \
-usbdevice tablet \
-vnc :10
```

The terminal window where you issues this command will be locked while kvm is running, so you may need to have second terminal opened.

Open VNC viewer and connect to the **<your server address>:10**. 10 is the number of our VNC screen (it corresponds to the port 5910), as we used it earlier when started virtual instance. After a small delay a Windows Setup window should appear.

Choose your language, time zone and keyboard layout, then click Next, and on the next screen click Install Now.

Select required installation type (Windows Server 2012 Standard Edition (Server with a GUI)).

Accept the EULA, and then select type of installation Custom: Install Windows only (advanced).

Now we should select the drive to install Windows on. By default Windows won't see any drives as there are no appropriate drivers.

Click Load driver, then Browse, and expand the CDROM drive (drive E:), following the path WIN8/ AMD64. Click OK.

Setup will scan this folder and should find two drivers (for Ethernet and SCSI devices):

```
Red Hat VirtIO Ethernet Adapter (E:\WIN8\AMD64\NETKVM.INF)
Red Hat VirtIO SCSI controller (E:\WIN8\AMD64\VIOSTOR.INF)
```

Select both and click Next.

The **Drive 0 Unallocated Space** record now should appear. Ensure that it is selected and click Next.

When installation complete, provide new password for Administrator (I use P@ssw0rd [mailto:P@ssw0rd] by default, as it fit password requirements.)

Post-install configuration

Log in to the system.

In the Server Manager window select the Local Server tab (it's on the left side).

Under Properties find the Remote Desktop property and click it's value string Disabled. Then, select Allow remote connections to this computer and uncheck the checkbox below it. Click OK. Click Refresh to ensure that the setting was changed.

Start PowerShell and type **ipconfig** to found your Ethernet adapter's default Gateway IP. Next, open Explorer and navigate to folder **\\gateway-ip\share\Files**.

Install Far Manager.

Create folder **C:\Program Files (x86)\Sysinternals Suite and extract SysinternalsSuite.zip** there.

Copy the Far Manager link file from **"C:\Users\All Users\Start Menu\Programs\Far Manager 3 x64\Far Manager 3 x64.lnk"** to the folder **"C:\Users\All Users\Start Menu\Programs\Administrative Tools"**. Now **Far Manager** could be opened from **Tools** menu in **System Manager**.

Open PowerShell window and shutdown the VM:

```
Stop-Computer -Force
```

Wait until the system shuts down.

Convert image to QCOW2 format

Convert hard drive image from RAW to QCOW2 format:

```
kvm-img convert -O qcow2 ws-2012.img ws-2012-refimage.qcow2
```

Note

The **ws-2012-refimage.qcow2** file is your reference image. You shoud **NEVER** modify it, and just make a copy to create a new VM.

The best way is to convert it back to the RAW format with a new name like this

```
kvm-img convert -O raw ws-2012-refimage.qcow2 new-vm-name.img
```

The qcow2 file should be about 7 Gb in size.

It's a good idea to reflect Windows Server type in image's name. For example

```
ws-2012-full-refimage.qcow2 for GUI version
ws-2012-core-refimage.qcow2 for Core version
```

Now we have a Reference Windows Server image, which we will use in further image preparation steps.

Chapter 3. Install OpenStack (DevStack)

Prepare Windows Image

Ensure that you have Reference Image of the Windows Server 2012. Make a copy of the Reference Image and configure it. Convert the final image to QCOW2 format.

Prepare the system for devstack installation

Create group *stack* and user *stack*

```
# groupadd stack
# useradd -g stack -s /bin/bash -m stack
```

Allow the user to become *root* without password

```
# echo 'stack ALL=(ALL) NOPASSWD: ALL' > /etc/sudoers.d/stack
# chmod 0440 /etc/sudoers.d/stack
```

Become user *stack* and clone Murano code to the **/opt** directory

```
# su stack
$ cd /opt
$ git clone git://github.com/...
```

Next, clone devstack to *stack* home directory

```
$ cd
$ git clone git://github.com/openstack-dev/devstack.git
```

Create a link to *devstack-scripts* folder

```
$ cd
$ ln -s devstack-scripts /opt/murano/Deployment/devstack-scripts
```

Edit *localrc* file

```
$ cd ~/devstack-scripts
$ vim localrc
```

Correct the following variables as shown below

```
...
MYSQL_DB_TMPFS=true
...
NOVA_CACHE_TMPFS=false
...
GLANCE_IMAGE_LIST="/path/to/the/image.qcow2"
COMPUTE_NODE_LIST=""
...
```

You may set the **GLANCE_IMAGE_LIST** to an empty string (**GLANCE_IMAGE_LIST=""**) and import the image name manually later.

Edit **~/devstack-scripts/standalone/devstack.localrc** file. This file will replace the **~/devstack/localrc** file so make all the necessary changes here.

```
$ cd ~/devstack-scripts/standalone
$ vim devstack.localrc
```

You might need to change the following variables:

```
...
HOST_IP=
...
FLAT_INTERFACE=
...
```

Start devstack

Start devstack using our helper script

```
$ cd ~/devstack-scripts
$ ./start-devstack.sh standalone
```

Wait until script finishes, then try to connect to web-interface.

To stop services, use another helper script

```
$ cd ~/devstack-scripts
$ ./stop-devstack.sh
```

Chapter 4. Log Files Locations

Component	Log files folder
Windows Agent	**C:\Murano\Agent\log.txt**
Windows PowerShell Modules	**%TEMP%** folder by default **C:\Windows\Temp** for Local System account **C:\Users\%username%\AppData\Local\Temp** for regular user Separate path also might be specified in module's code.
CloudBase-Init	**C:\Program Files (x86)\Cloudbase-Init\logs\cloudbase-init.log**
Devstack	**\opt\stack\logs***
Portas	
Conductior	