Installation Procedure

This document contains instructions to do complete setup of Xen, KVM hosts and OpenStack along with integrated RPCore, MtWilson and KMS.

# OpenStack Setup

This section describes how to setup OpenStack components and upgrade it to have all required code changes.

**Note:** This is a physical machine. We will refer this machine as Controller in the remaining document.

## Controller Setup

1. Create a bootable media using the controller iso
2. Install the OS

### Apply Patches

1. Make sure after booting machine has an IP or provide a static IP
2. Checkout 'Controller' directory (from GIT\_REPO\_DEV\_BRANCH/mysteryhill/Setup-Standardization/Controller) and go inside it

# cd Controller

1. Run the ‘controller-config.sh’ as below

# ./controller-config.sh <Controller’s IP\_Address>

Eg. # ./controller-config.sh 192.168.0.2

It will start the all controller services

# Xen Host Setup

This section describes how to setup Xen host and upgrade it to have all required components and code changes.

**Note:** This is a physical machine. We will refer this machine as Xen-Host / dom0 in the remaining document.

## Base Xen Host Setup

First of all install Xen and dom0 using below mentioned steps:

1. Create a bootable media using the dom0 iso (available at … Check with Andrew/Kitty for location ) using the ‘Steps to create bootable USB in Windows’ doc
2. Install the OS with manual partitioning
   1. We need one partition of 100GB to create SR

## RPCore and other integration related changes

Follow below mentioned steps to install RPCore and other code changes once base Xen host is setup:

1. Checkout ‘Dom0’ directory (from GIT\_REPO\_DEV\_BRANCH/mysteryhill/Setup-Standardization/Dom0 ) and go inside it

# cd Dom0

1. Run the ‘setup-xcp-xapi.sh’ script to install and configure the xcp-xapi

# ./ setup-xcp-xapi.sh

1. Edit the /etc/network/interfaces and put your IPs

---------------------------------------

*# interfaces(5) file used by ifup(8) and ifdown(8)*

*auto lo*

*iface lo inet loopback*

*auto eth0*

*iface eth0 inet manual*

*# Xenbr0 config*

*auto xenbr0*

*iface xenbr0 inet dhcp*

*bridge\_ports eth0*

-----------------------------------------

1. Reboot the machine
2. Check for xcp service status

# service xcp-xapi status

# xe vm-list

1. Use the ‘kernel-upgrade.sh’ script to upgrade the kernel to 3.11.0-19-generic
2. Reboot the machine
3. Check the blktap module

# modinfo blktap

1. mkdir /opt/RP
2. Clone rpcore folder from GIT to /opt/RP and build it.
3. Run ‘dom0-config.sh’ to run the nontpm rpcore and xapi\_access\_control\_proxy.

xapi\_access\_control\_proxy expects configuration file at path-/etc/intel\_rpcore\_plugin.conf

It will be automatically set by dom0-config.sh.

# ./dom0-config.sh

1. To block port 80 for outside request

# iptables -A INPUT -d <dom0\_ip> -p tcp --dport 80 -j DROP

1. Create SR

Either You can use script to create Encrypted SR- (device encrypted with DM\_CRYPT)

or you can setup non encrypted SR manually.

For Encrypted SR:

a. Execute script /opt/RP/setup\_encrypted\_sr.sh It will ask you for device on which you want to create SR: for example /dev/sda4. And it will generate file: /etc/sr\_encryption\_key

b. After system is rebooted, execute script /opt/RP/setup\_encrypted\_sr\_after\_reboot.sh to create encrypted lvm once again, and activate SR.

For Non Encrypted SR:

assuming /dev/sda4 is the device to be used for SR

# xe sr-create device-config:device=/dev/sda4 host-uuid=<host-uuid> name-label=LocalStorage shared=true type=ext content-type=user

The output will be sr-uuid -- 814394a8-80a7-a0d7-2a16-480cd902a42f

# xe pool-param-set uuid=<pool-uuid> default-SR=<sr-uuid created in above step>

Get the pool-uuid from- xe pool-list

and get sr-uuid from - xe sr-list

1. Create the symbolic link for /boot/guest

# ln -s /var/run/sr-mount/7af1113e-7fcf-ea6c-8a7e-5912b63dbfba/os\_image /boot/guest

## Compute Node Setup

Follow below steps to launch compute node using provided image:

1. Go to the Dom0 machine
2. Import the provided compute image (available at … Check with Andrew/Kitty for location ) using ‘xe’ command as below

# xe vm-import filename=base\_image\_compute\_new

1. # xe vm-start uuid=<uuid return in above step>
2. # xe console uuid=<vm-uuid>

Configure compute VM using below mentioned steps:

1. Configure the network, if needed
2. Check out ‘Compute’ directory (from GIT\_REPO\_DEV\_BRANCH/mysteryhill/Setup-Standardization/Compute) in /root/ directory
3. Go to the /root/Compute/ directory and run ‘compute-config.sh’

## Controller Machine Configuration

~~Once controller is setup as mentioned in the “OpenStack Setup” section, execute following command on controller machine. Make sure that all compute services are running on Compute VM:~~

~~# nova network-create vmnet --fixed-range-v4=10.35.34.0/24 --bridge-interface=xenbr0 --multi-host=T~~

nova network-create test --fixed-range-v4=192.168.1.0/24 --multi-host=T --bridge=xenbr0 --bridge-interface=eth1 --gateway=192.168.1.254 --dns1=192.168.0.4

Note: Value of fixed-range-v4 in the above command should be the range specified in /etc/dhcp/dhcpd.conf file in DHCP server

# KVM Host Setup

This section describes how to setup KVM host and upgrade it to have all required components and code changes.

**Note:** This is a physical machine. We will refer this machine as KVM-Host in the remaining document.

## Base KVM Host Setup

Follow below mentioned steps to setup KVM Host from iso image:

1. Create a bootable media using the KVM iso (available at … Check with Andrew/Kitty for location) using the ‘Steps to create bootable USB in Windows’ doc
2. Install the OS

## RPCore and other integration related changes

Follow below steps to apply RPCore and other integration related changes:

1. Get the KVM directory from git
2. mkdir /opt/RP
3. Get the rpcore directory from git, copy it to /opt/RP directory and build it.
4. Go to the 'kvm' directory

# cd kvm

1. Upgrade the kernel to 3.11.0-19-generic using script 'kernel-upgrade.sh' and reboot system

# ./kernel-upgrade.sh

# reboot

1. Run the 'kvm-setup.sh', it will install the kvm packages.

# ./kvm-setup.sh

1. Run the 'kvm-config.sh', it will configures kvm setup. It will also start nontpmrpcore and rplistener

If you want to run the tpm rpcore

a. Set up rpcore manually first.

b. Comment the line number 176 from kvm-config.sh which a function call to start nontpm rpcore.

then run the script

# ./kvm-config.sh

1. Edit the ‘/etc/network/interfaces’ to look like as below

---------------------------------------

*# interfaces(5) file used by ifup(8) and ifdown(8)*

*auto lo*

*iface lo inet loopback*

*auto eth0*

*iface eth0 inet manual*

*auto br100*

*iface br100 inet dhcp*

*bridge\_ports eth0*

-----------------------------------------

1. Reboot the machine
2. After reboot execute the following command

# chown -R nova:nova /var/run/libvirt/

1. Restart the nova services

# service nova-api-metadata restart

# service nova-network restart

# service nova-compute restart

## Controller Machine Configuration

~~When all compute services are running on KVM Host, run the nova network-create command on Controller machine:~~

~~# nova network-create vmnet --fixed-range-v4=10.35.35.0/24 --bridge-interface=br100 --multi-host=T~~

nova network-create test --fixed-range-v4=192.168.1.0/24 --multi-host=T --bridge=xenbr0 --bridge-interface=eth1 --gateway=192.168.1.254 --dns1=192.168.0.4

Note: Value of fixed-range-v4 in the above command should be the range specified in /etc/dhcp/dhcpd.conf file in DHCP server

# MtWilson Setup

TBD

# MH Server (KMS) Setup

TBD

# MH Client Setup

TBD

# Important Scripts

This section contains information related to useful scripts.

## Installation Related Scripts

TBD

## Scripts to perform operations on different components

TBD

# Basic Test of Setup

Following processes needs to be running on mentioned systems:

Component Name: Xen – dom0

|  |  |  |
| --- | --- | --- |
| Process Name | Port | Useful Information |
| Tcsd | 30003 | tcsd daemon needs to be started before starting RPCore  # sudo service trousers start/stop/restart |
| RPCore | 16005 | Tcsd needs to be started before starting RPCore.  To start RPCore service execute following command from $rpcore\_root/bin/debug/:  # sudo ./tcsd-rpcoresvc |
| XAPI | 80 | # sudo service xcp-xapi start/stop/restart |
| xapi\_access\_control\_proxy | 8080 | XAPI needs to be started before starting xapi-access-control-proxy  Start XAPI Access Control proxy using following command from xapi-access-control-proxy:  # sudo ./xapi-access-control.py |
| OVS – TBD -- | 6633 |  |
| Trust Agent – TBD -- | 9999 |  |

Component Name: KVM host

|  |  |  |
| --- | --- | --- |
| Process Name | Port | Useful Information |
| Tcsd | 30003 | tcsd daemon needs to be started before starting RPCore  # sudo service trousers start/stop/restart |
| RPCore | 16005 | Tcsd needs to be started before starting RPCore.  To start RPCore service execute following command from $rpcore\_root/bin/debug/:  # sudo ./tcsd-rpcoresvc |
| Libvertd |  | To start libvertd execute following command:  # sudo /usr/sbin/libvirtd –d |
| rp\_listner | 16004 | To start rp\_listner execute following command under directory $rpcore\_root/src/rpproxy/kvm\_proxy/:  ./rp\_listner |
| Nova Compute |  | # sudo service nova-compute start/stop/restart |
| OVS | 6633 | # sudo service neutron-plugin-openvswitch-agent start/stop/restart |
| Trust Agent – TBD -- | 9999 |  |

Component Name: Controller

|  |  |  |
| --- | --- | --- |
| Process Name | Port | Useful Information |
| MySQL | 3306 | # sudo service mysql start/stop/restart |
| MemCache | 11211 | # sudo service memcached start/stop/restart |
| RabbitMQ | 5672 (default) | # sudo service rabbitmq-server start/stop/restart/status |
| Keystone |  | # sudo service keystone start/stop/restart |
| nova-api |  | # sudo service nova-api start/stop/restart |
| nova-cert |  | # sudo service nova-cert start/stop/restart |
| nova-conductor |  | # sudo service nova-conductor start/stop/restart |
| nova-consoleauth |  | # sudo service nova-consoleauth start/stop/restart |
| neutron-server |  | # sudo service neutron-server start/stop/restart |
| neutron-dhcp-agent |  | # sudo service neutron-dhcp-agent start/stop/restart |
| neutron-metadata-agent |  | # sudo service neutron-metadata-agent start/stop/restart |
| neutron-ns-metadata-proxy |  | # sudo service neutron-ns-metadata-proxy start/stop/restart |
| neutron-openvswitch-agent |  | This needs only if OVS is enabled.  # sudo service neutron-openvswitch-agent start/stop/restart |
| glance-api |  | # sudo service glance-api start/stop/restart |
| glance-registry |  | # sudo service glance-registry start/stop/restart |

## Basic Testcases

|  |  |  |
| --- | --- | --- |
| Testcase | Steps | Expected Output |
| TBD |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |