**LIVE MIGRATION**

**Multi node devstack setup:**

* [**http://docs.openstack.org/developer/devstack/guides/multinode-lab.html**](http://docs.openstack.org/developer/devstack/guides/multinode-lab.html)
* Ignore the Network configuration part.
* For controller, change  HOST\_IP and floating IP range with IP of the controller node in local.conf.
* For compute, change  HOST\_IP and floating IP range with IP of the compute node in local.conf. And SERVICE\_HOST as controller IP.
* After completing the setup, check the list of available nodes using: nova service-list
* Enable/Disable services on a node using: nova-manage service enable/disable --host=<<hostname>> --service=<<service-name>>

Ex: nova-manage service disable --host=s2controller --service=nova-compute

* Verify that compute1 and compute2 are listed.
* Add compute1 and compute2 with IPs in /etc/hosts

**Live Migrating Instances:**

* Command to live-migrate instances:

nova live-migration <<instance>> <<compute node>>

* For instances without shared storage use –block-migrate.
* **Issues you may face:**
* **NoLiveMigrationForConfigDriveInLibVirt**
* Comment out force\_config\_drive in /etc/nova/nova.conf. It is set to true by default. Either change it to false or comment.
* Restart nova services
* Create a new instance and live migrate. Already existing instances don’t work as the force\_config\_drive was true for that instance.
* **Unacceptable CPU info: CPU doesn't have compatibility**.
* This exception occurs when instance CPU model is different from destination host CPU model.
* In nova.conf change the cpu\_mode and cpu\_model.

cpu\_mode = custom

cpu\_model = <<model name>>

* There are a number of cpu models listed in /usr/share/libvirt/cpu\_map.xml
* Select a model having features common to both instance CPU and destination host CPU.
* On rackspace cloud servers, changing model to gate64 in all the compute nodes worked.
* Restart the nova services and create a new instance and then live-migrate.
* **Live Migration failure: operation failed: Failed to connect to remote libvirt URI qemu+ssh://ubuntu@node-01/system: Cannot recv data**
* Source host is not able to ssh destination without password.
* Generate ssh keypair and add the public key to the destination host authorized keys.
* <http://www.tecmint.com/ssh-passwordless-login-using-ssh-keygen-in-5-easy-steps/>
  + For root user: On compute1:
    - ssh-keygen -t rsa
    - This saves private and public key in .ssh folder.
    - cat .ssh/id\_rsa.pub | ssh root@<<compute2IP>> 'cat >> .ssh/authorized\_keys'
* For stack user:
  + - ssh-keygen -t rsa
    - This saves private and public key in .ssh (/home/stack/.ssh) folder.
    - cat .ssh/id\_rsa.pub | ssh stack@<<compute2IP>> 'cat >> .ssh/authorized\_keys'
* Do the same for compute node2.
* Summary to provide password less ssh access:

1. First exchange ssh keys between users who launched ./stack.sh as mentioned above.  
2. Second exchange ssh keys between root users

3. Third allow root users to ssh without password on users mentioned in step 1.

For example.  
user@host:~$ sudo su  
root@host:~ ssh-copy-id user@compute-1  
root@host:~ ssh-copy-id user@compute-2

* Test:
  + (root)Compute1: ssh roubuntuot@compute2
  + (root)Compute1: ssh stack@compute2
  + (stack)Compute1: ssh stack@compute2
  + (stack)Compute1: ssh root@compute2
  + Same for compute2