Exercise 3.2: Add a Compute Host

Overview

In a previous exercise you deployed an All-In-One DevStack instance, running on **Ubuntu**. Then configured a project, user and deployed a new virtual machine.

Connect to the terminal of your cloud controller, **devstack-cc**, via the provided link for lab3.2. You will be presented a new **Katacoda** environment. The new instance may have a different public IP address and URL for BUI access. Use the **ip** command, as shown in a previous task, to determine the IP address for **eth0** for the new instance and reference the file /opt/host for the URL to the **Horizon** BUI. You can also use the **OpenStack Dashboard** tab on the **Katacoda** page.

Install Software on the New Compute Node

In this exercise we will grow our cloud by adding a Nova compute node. Connect to the terminal via the browser. The only way to connect to **compute-node** is via **devstack-cc node**.

An SSH public key for the **Ubuntu** user has been implemented and the compute-node has been pre-populated. If asked to accept the SSH fingerprint choose yes. Use **exit** to return to devstack-cc when necessary. For example:

```
ubuntu@devstack-cc:~$ ssh compute-node
ubuntu@compute-node:~$ exit
ubuntu@devstack-cc:~$
```

Find the IP address of your controller node (devstack-cc). We will use this IP address in later stages

```
ubuntu@devstack-cc: $ ip addr show ens3
    2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000

       link/ether 02:42:ac:11:00:0b brd ff:ff:ff:ff:ff:ff

       inet 172.17.0.11/16 brd 172.17.255.255 scope global ens3

          valid_lft forever preferred_lft forever

       inet6 fe80::42:acff:fe11:b/64 scope link

          valid_lft forever preferred_lft forever
```

The backslash in the **git** command following is to indicate that the command should be on one line.

1. Install the **git** command and pull down the **DevStack** software.

```
ubuntu@devstack-cc:~$ ssh compute-node

ubuntu@compute-node:~$ sudo apt-get update

    ubuntu@compute-node:~$ sudo apt-get install git vim
    <output_ommited>
    After this operation, 21.6 MB of additional disk space will be used. Do you want
    to continue? [Y/n] y
    <output_omitted>

    ubuntu@compute-node:~$ git clone \ https://git.openstack.org/openstack-
    dev/devstack -b stable/pike <output_omitted>
```

2. Find the private IP address of the compute node. Update the table at the beginning of the lab for future reference. Your IP may be different than the example below.

```
ubuntu@compute-node:~$ ip addr show ens3
2:     ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc mq state UP group default qlen 1000
         link/ether 02:1f:91:1e:db:18 brd ff:ff:ff:ff:ff:ff
         inet 192.168.97.2/20 brd 172.31.47.255 scope global ens3
            valid_lft forever preferred_lft forever
         inet6 fe80::1f:91ff:fe1e:db18/64 scope link valid_lft
            forever preferred_lft forever
```

3. We need to create another local.conf file, similar but different from the first node. This file will point to the IP Address of the first node so that the script can sign in to the various services. We will also limit which services are enabled on the new node. Note the flat interface may be different. Nodes dedicated to compute services don't need access to the same networks as a head node or the network node and may use a data network instead.

NOTE: Here assign the SERVICE_HOST with IP address of controller node (devstack-cc) as found in first step.
But, HOST_IP and HOST_IPV6 will have the IPv4 and IPv6 addresses of current compute-node, as found in previous step.

```
ubuntu@compute-node:~$ cd devstack ; vim local.conf
[[local|localrc]]
HOST_IP=192.168.97.2 # IP for compute-node
SERVICE_HOST=192.168.97.1 # devstack-cc IP, first node you used
FLAT_INTERFACE=ens3
FIXED_RANGE=10.4.128.0/20
FIXED_NETWORK_SIZE=4096
FLOATING_RANGE=192.168.42.128/25
MULTI_HOST=1
LOGFILE=/opt/stack/logs/stack.sh.log
ADMIN_PASSWORD=openstack
DATABASE_PASSWORD=db-secret
RABBIT_PASSWORD=rb-secret
SERVICE_PASSWORD=sr-secret
DATABASE_TYPE=mysql
MYSQL_HOST=$SERVICE_HOST
RABBIT_HOST=$SERVICE_HOST
GLANCE_HOSTPORT=$SERVICE_HOST:9292
ENABLED_SERVICES=n-cpu,q-agt,n-api-meta,c-vol,placement-client
NOVA_VNC_ENABLED=True
NOVNCPROXY_URL="http://$SERVICE_HOST:6080/vnc_auto.html"
VNCSERVER_LISTEN=$HOST_IP
VNCSERVER_PROXYCLIENT_ADDRESS=$VNCSERVER_LISTEN
```

4. Before running the stack.sh script, save the output of the **ip** command for later comparison: ubuntu@compute-node:~devstack$ ip addr show > ~/ip.before.out

5. Install the **DevStack** software on the second node. If there are issues, double-check and edit the local.conf configuration file, run ./unstack.sh and ./clean.sh and try again. Ask for assistance if you continue to receive errors.

```
ubuntu@compute-node:~devstack$ ./stack.sh
<output_omitted>
```

6. Once the script has finished check to see if you have a second hypervisor.As admin, navigate to Admin -> Compute -> Hypervisors The Hypervisor tab should show two hostnames, as does the Compute Host tab.

If not you will need to use a five step process to enable the new node. You may see some output about python code deprecation. This can be ignored if the node is added. Your hostnames and IP addresses may be different. Below we find only one hypervisor after adding the compute node.

```
ubuntu@compute-node:~devstack$ source openrc admin
```

```
ubuntu@compute-node:~devstack$ openstack hypervisor list
+        ----+                      --------------------  +                    ----------------+--------------          +-------       +
| ID |                         Hypervisor Hostname |  Hypervisor Type | Host IP                                       | State |
+        ----+                      --------------------  +                    ----------------+--------------          +-------       +
|            1 |                      devstack-cc                  |  QEMU                              | 192.168.71.1             | up             |
+        ----+                      --------------------  +                    ----------------+--------------          +-------       +
```

7. Return to the ~~compute-node~~ controller node (devstack-cc) and enable the new hypervisor.

(a) **Return to the devstack-cc node**. Source the config file as admin ubuntu@devstack-cc:~$ cd devstack/

```
ubuntu@devstack-cc:~/devstack$ source openrc admin
WARNING: setting legacy OS_TENANT_NAME to support cli tools.
```

~~(b) Verify the compute-host was added and is up.~~

```
ubuntu@devstack-cc:~/devstack$ ~~nova service-list --binary nova-compute~~

ubuntu@devstack-cc:~/devstack$ openstack compute service list
+----+-----------------+-------------+----------+---------+-------+----------------------------+
| ID | Binary          | Host        | Zone     | Status  | State | Updated At                 |
+----+-----------------+-------------+----------+---------+-------+----------------------------+
[...]
|  2 | nova-compute    | devstack-cc  | nova     | enabled | up    | 2019-06-08T07:29:43.000000
|
|  3 | nova-compute    | compute-node | nova     | enabled | up    | 2019-06-
08T07:29:41.000000 |
+----+-----------------+-------------+----------+---------+-------+----------------------------+
+------------------------------------                                +--------------          +-------------- +------+           ----
+-------                         +--------------------------- +         ----------------          +------------- +
| Id                                                              | Binary                | Host                          |  Zone |           Stat
| State | Updated_at                                             |  Disabled Reason | Forced down                  |
+------------------------------------                                +--------------          +-------------- +------+           ----
-+-------                          +---------------------------          +----------------          +-------------+
                     | 32fa0ccd-45a8-45f5-b2e7-2d84e7377eb3 | nova-compute          | devstack-cc                 |  nova |           enabl
| up                          | 2017-12-19T19:49:35.000000 |  -                                    | False                 |
                       | 9fc15015-d150-478f-ba4f-764fe4ed03c9 | nova-compute          | compute-node               |  nova |           enabl
| up                          | 2017-12-19T19:49:35.000000 |  -                                    | False                 |
+------------------------------------                                +--------------          +-------------- +------+           ----
-+-------                          +---------------------------          +----------------          +-------------+
```

(c) Verify the hypervisor has not yet been added.

```
ubuntu@devstack-cc:~/devstack$ openstack hypervisor list
+        ----+                      --------------------  +                    ---------------- +        --------------        +-------
| ID |                         Hypervisor Hostname |                    Hypervisor Type |  Host IP                        | State |
+        ----+                      --------------------  +                    ---------------- +        --------------        +-------
|            1 |                      devstack-cc                  |               QEMU                   |         192.168.97.1        | up
+        ----+                      --------------------  +                    ---------------- +        --------------        +-------
```

(d) Use a script to join the hypervisor to the cloud.

```
ubuntu@devstack-cc:~/devstack$ ./tools/discover_hosts.sh
/usr/local/lib/python2.7/dist-packages/pymysql/cursors.py:166: Warning: (1287, u"'@@t x_isolation'
is deprecated and will be removed in a future release. Please use '@@tra nsaction_isolation'
instead")

  result = self._query(query)
Found 2 cell mappings.
Skipping cell0 since it does not contain hosts.
Getting compute nodes from cell 'cell1': 79bd3053-a007-469d-ba72-d7b106d08568
Found 1 unmapped computes in cell: 79bd3053-a007-469d-ba72-d7b106d08568
Checking host mapping for compute host 'compute-node': b3caa6f3-fe33-49af-839a-375813 8af2b1

Creating host mapping for compute host 'compute-node': b3caa6f3-fe33-49af-839a-375813 8af2b1
```

(e) Verify the compute-host has been added.

```
ubuntu@devstack-cc:~/devstack$ openstack hypervisor list
+    ----+---------------------                +-----------------          +    --------------    +-------
| ID     | Hypervisor Hostname                 | Hypervisor Type |            Host IP              | State |
+    ----+---------------------                +-----------------          +    --------------    +-------
|       1| devstack-cc                         | QEMU                      |      192.168.97.1     | up
|       2| compute-node                        | QEMU                      |      192.168.97.2     | up
+    ----+---------------------                +-----------------          +    --------------    +-------
```

8. Return to the compute-node. Save the output of the **ip** command again to a new file. Compare how the networking on the node has changed. Note the new bridges and interfaces created.

Ssh compute-node

```
ubuntu@compute-node:~devstack$ ip addr show > ~/ip.after.out

ubuntu@compute-node:~devstack$ diff ~/ip.before.out ~/ip.after.out
```

9. What new network interfaces do you see ? ____

Answer:

```
5: ovs-system: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1
   link/ether 06:3e:2a:b7:5e:2e brd ff:ff:ff:ff:ff:ff
8: br-int: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1
   link/ether 4a:c8:2c:01:a2:41 brd ff:ff:ff:ff:ff:ff
9: br-ex: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1
   link/ether 0e:fa:be:92:2f:44 brd ff:ff:ff:ff:ff:ff
10: br-tun: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1
   link/ether 26:08:6d:37:b9:44 brd ff:ff:ff:ff:ff:ff
11: vxlan_sys_4789: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 65485 qdisc noqueue master ovs-system state UNKNOWN group default qlen 1000
   link/ether d2:a5:b4:fb:2c:ee brd ff:ff:ff:ff:ff:ff
   inet6 fe80::d0a5:b4ff:fefb:2cee/64 scope link
      valid_lft forever preferred_lft forever
```

10. We will create another instance from the BUI. After it has finished spawning run the **ip** command again and view the differences again.

On your local system open a browser and point it at the public IP Address of your **devstack-cc** node. Log into BUI as developer1 with the password openstack.

Navigate to Project -> Compute -> Instances. Select Launch Instance.

Use the name devOS3 and boot from the available cirros image. Select the m1.tiny flavor. When the fields are filled select Launch

When it finishes spawning check the differences in IP information on the new compute host.

```
ubuntu@compute-node:~devstack$ ip addr show > ~/ip.devos3.out

ubuntu@compute-node:~devstack$ diff ~/ip.after.out \
~/ip.devos3.out
```

You may see something like this

```
12. qbr396a293f-65: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1450 qdisc noqueue state UP group default qlen
1000
   link/ether aa:63:90:1d:da:73 brd ff:ff:ff:ff:ff:ff
13: qvo396a293f-65@qvb396a293f-65: <BROADCAST,MULTICAST,PROMISC,UP,LOWER_UP> mtu 1450 qdisc noqueue
master ovs-system state UP group default qlen 1000
   link/ether 8e:df:4b:06:9d:a5 brd ff:ff:ff:ff:ff:ff
   inet6 fe80::8cdf:4bff:fe06:9da5/64 scope link
      valid_lft forever preferred_lft forever
14: qvb396a293f-65@qvo396a293f-65: <BROADCAST,MULTICAST,PROMISC,UP,LOWER_UP> mtu 1450 qdisc noqueue
master qbr396a293f-65 state UP group default qlen 1000
   link/ether aa:63:90:1d:da:73 brd ff:ff:ff:ff:ff:ff
   inet6 fe80::a863:90ff:fe1d:da73/64 scope link
      valid_lft forever preferred_lft forever
15: tap396a293f-65: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1450 qdisc pfifo_fast master qbr396a293f-65 state
UNKNOWN group default qlen 1000
   link/ether fe:16:3e:a6:00:72 brd ff:ff:ff:ff:ff:ff
   inet6 fe80::fc16:3eff:fea6:72/64 scope link
      valid_lft forever preferred_lft forever
```

10. Log into the BUI as the user admin with the password openstack

11. Navigate to the Admin -> Compute -> Hypervisors page.

12. Select the hypervisor tab. You should see a second hypervisor listed. Also a second compute host listed under the Compute Host tab.

13. Navigate to the Admin -> Compute -> Instances page. You should find that each compute host has one instance running.

14. Return to the command line. Use **exit** to return to the **devstack-cc** system. Using the same command and namespace as before, but with the IP Address for **devOS3** try to log into the new instance. Your instance IP may be different.

```
ubuntu@compute-node:~/devstack$ exit
logout
Connection to compute-node closed.
```

14.a) You should also check the route ID using the dashboard.
Network > Routers > net-router. Click on properties and read the ID field

ID: 27793189-8ffd-4eab-9a43-aee5e8f81a8d

14.b) Get the route-UID for the current deployment. It may not be same as previous class.

```
ubuntu@devstack-cc: sudo ip netns list
qrouter-27793189-8ffd-4eab-9a43-aee5e8f81a8d
qdhcp-82f2dbc7-c535-4328-97a3-5d1dd5f894b5
```

```
qrouter-a9c28fc7-bda5-4fe5-9df5-37eeff3023d5
qdhcp-f06a5984-b87d-47c7-a6e2-7cd12ded6672
```

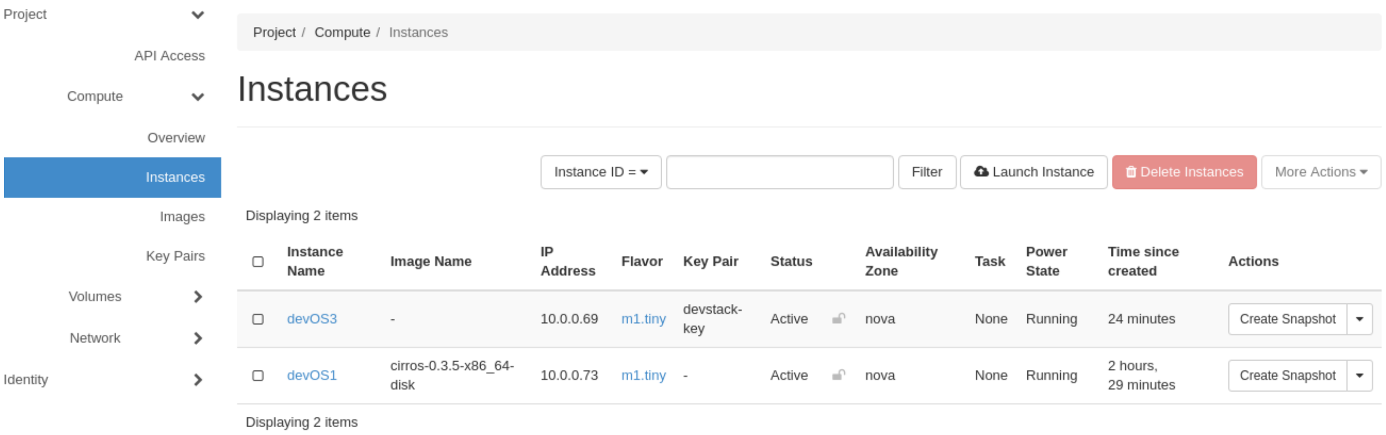
15. Now try to login to devOS3 instance

```
ubuntu@devstack-cc:~/devstack$ sudo ip netns exec \
qrouter-4fd279c4-b125-4611-956d-adc67432e0d2 \ ssh
cirros@10.0.0.
The authenticity of host ’10.0.0.10 (10.0.0.10)’ can’t be established. RSA key
fingerprint is f8:3f:2a:07:d4:31:51:66:ee:a7:00:5c:22:f8:ce:c3. Are you sure you
want to continue connecting (yes/no)? yes
Warning: Permanently added ’10.0.0.10’ (RSA) to the list of known hosts.
cirros@10.0.0.10’s password: cubswin:) $
exit
Connection to 10.0.0.10 closed.
```

NOTE: username login=cirros , IP address and password can also be seen from the “log” view of the instance devOS3 under Compute > Instances tab

16. Goto Compute > Instances using on your Dashboard.

Why is “Image Name” empty under the devOS3 instance ? --- Question (unanswered)



Displaying 2 items

□	Instance Name	Image Name	IP Address	Flavor	Key Pair	Status		Availability Zone	Task	Power State	Time since created	Actions
□	devOS3	-	10.0.0.69	m1.tiny	devstack-key	Active	🔓	nova	None	Running	24 minutes	Create Snapshot ▾
□	devOS1	cirros-0.3.5-x86_64-disk	10.0.0.73	m1.tiny	-	Active	🔓	nova	None	Running	2 hours, 29 minutes	Create Snapshot ▾

Displaying 2 items

17. You can delete the devOS3 instance, as well, to conserve resources.

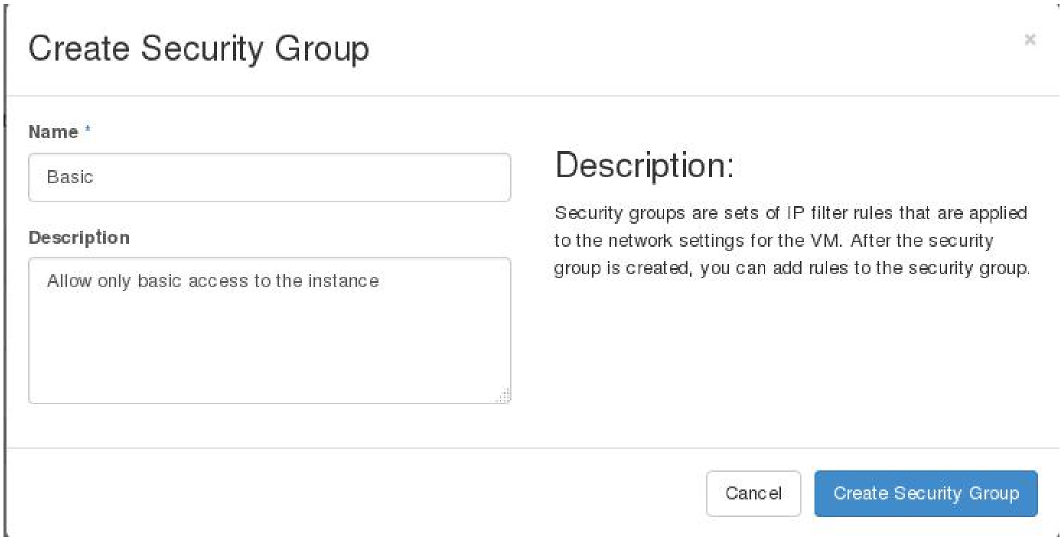
Create a Security Group

The private IP Address allows access to an instance from the host machine. In order to allow outside access to an instance a new security group must be created and rules for access added.

1. Log into the BUI as developer1.

2. Navigate to the Project -> Network -> Security Groups page.

3. Select the +Create Security Group button. Fill it out as found in the following graphic, then select the Create Security Group button.



Create Security Group

Name *

Basic

Description

Allow only basic access to the instance

Description:

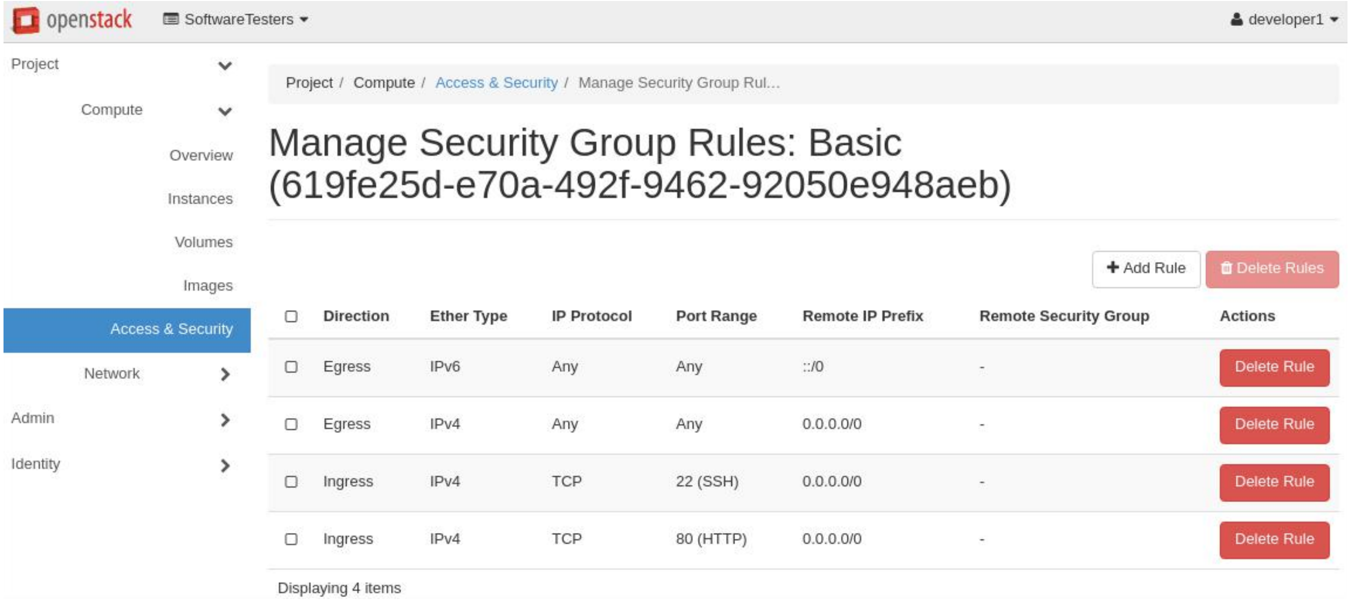
Security groups are sets of IP filter rules that are applied to the network settings for the VM. After the security group is created, you can add rules to the security group.

Figure 3.20: **Create a Security Group**

4. Select the button Manage Rules under the Actions column on the right of the newly created line for the Basic group.

5. Select the +Add Rule button. Add rules for **ssh** and HTTP access. To add **ssh** access, under the top drop-down scroll to the bottom and select SSH, then the Add button.

6. Follow the same steps to add a rule for HTTP. After adding the rule your page should look something like this:



Project / Compute / Access & Security / Manage Security Group Rul...

Manage Security Group Rules: Basic (619fe25d-e70a-492f-9462-92050e948aeb)

☐	Direction	Ether Type	IP Protocol	Port Range	Remote IP Prefix	Remote Security Group	Actions
☐	Egress	IPv6	Any	Any	::/0	-	Delete Rule
☐	Egress	IPv4	Any	Any	0.0.0.0/0	-	Delete Rule
☐	Ingress	IPv4	TCP	22 (SSH)	0.0.0.0/0	-	Delete Rule
☐	Ingress	IPv4	TCP	80 (HTTP)	0.0.0.0/0	-	Delete Rule

Displaying 4 items

Figure 3.21: **Rules for SSH and HTTP**

7. After adding the rules navigate back to the Project -> Compute -> Instances page.

8. Click on the drop-down under the Actions field under your longest running, instance, devOS1, and select Edit Security Groups.

9. Select the blue plus sign to add the Basic group to this instance, then Save.

Use a Floating IP Address

Now that we have associated a new security group which allows **ssh**, let’s test our work. First we add a gateway so our private network can access the public network, allocate an IP to the Project, then associate with a port of an instance.

1. Navigate to the Project -> Network -> Routers page. Select the Set Gateway button. Choose the drop-down and select **public** as the External Network. Then select Submit.

2. Navigate to the Project -> Network -> Floating IPs page.

3. Select the Allocate IP to Project button. Use the drop-down to select the public pool. Then the Allocate IP button. A new address should be listed, but in a **Down** status.

4. Navigate to the Project -> Compute -> Instances page.

5. Click on the drop-down under the Actions field for devOS1 and select Associate Floating IP.

6. Use the drop-down to select the newly allocated IP address. Then the Associate button.

7. When the BUI updates, write down the newly assigned floating IP address: ____________________________

8. Return to the command line of your cloud controller and log into the instance, but without using a namespace. Instead using the newly assigned floating IP address. Your IP address will be different than the example following.

```
ubuntu@compute-node:~/devstack$ ssh cirros@192.168.42.141
<output_omitted>
cirros@192.168.42.141’s password: cubswin:) $
uname -a
Linux devos1 3.2.0-80-virtual #116-Ubuntu SMP Mon Mar 23 17:28:52 UTC 2015 x86_64 GNU/Linux $ exit

Connection to 192.168.42.141 closed.
```

Solution 3.2

Use a Floating IP Address

2. 192.168.42.141