How-To: Setup a Development Machine

- SMACK PRINT -

FALL 2015

Compiled By: Kyle Derby MacInnis

## Introduction

The purpose of this document is to outline the steps necessary to launch a fully configured and ready-to-use development machine. This machine will be used to access and configure the various models and simulations as well as used to draft scripts for automation purposes. The steps outline the necessary software that must be installed and includes some of the following:

- CentOS 6.5+
- Java JDK 7/8
- Python 2.7 & Python 3.3
- R 3.2.2 & Packages
- OpenStack Client Software
- Compilers
- Editors
- Utilities

The following will outline the steps used to install a development machine onto the OpenStack Cloud

****Note: Steps will be similar for installing onto a local machine (Assuming CentOS 6.5)***

## Prerequisites

This document assumes that you will launch the development machine as part of the OpenStack cloud. This will be done in two ways:

- Graphically
- Command Line Interface (CLI)

In order to use the command-line tools – the OpenStack Client software must be installed – this is actually done as part of the graphical method and should you wish to follow the CLI method, you must first install the tools yourself.

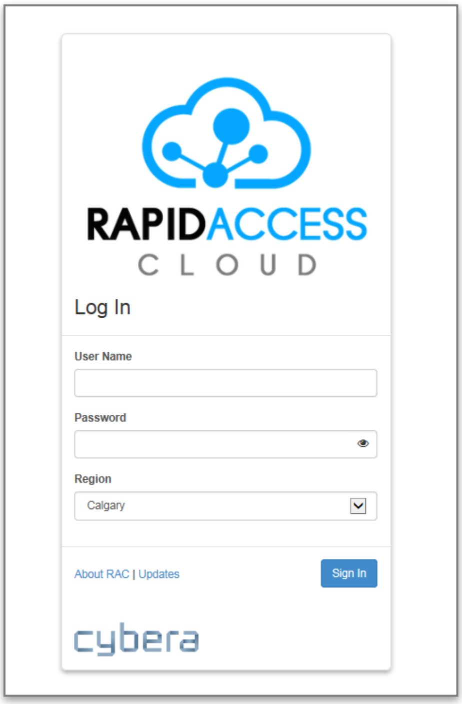
It is recommended that the Graphical Method is used as it requires fewer additional steps.

## Step-by-Step

The following two methods are done in a step-by-step procedure. Where necessary, some steps may not be included for brevity.

Graphical Method:

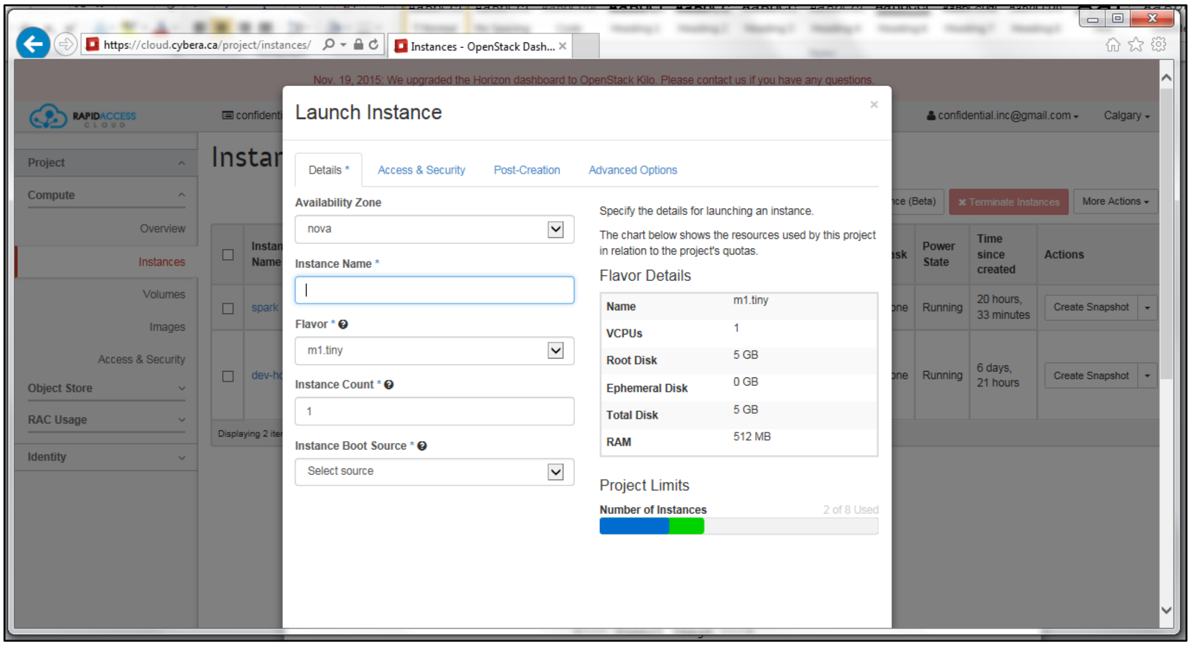
#Step 1: Login to *http://cloud.cybera.ca*



Region: Calgary

FIGURE 1 - LOGIN

#Step 2: Go to Instances & Launch Instance



Nov. 19, 2015: We upgraded the Horizon dashboard to OpenStack Kilo. Please contact us if you have any questions.

Launch Instance

Details * | Access & Security | Post-Creation | Advanced Options

Availability Zone: nova

Instance Name *: |

Flavor *: m1.tiny

Instance Count *: 1

Instance Boot Source *: Select source

Specify the details for launching an instance.

The chart below shows the resources used by this project in relation to the project's quotas.

Flavor Details

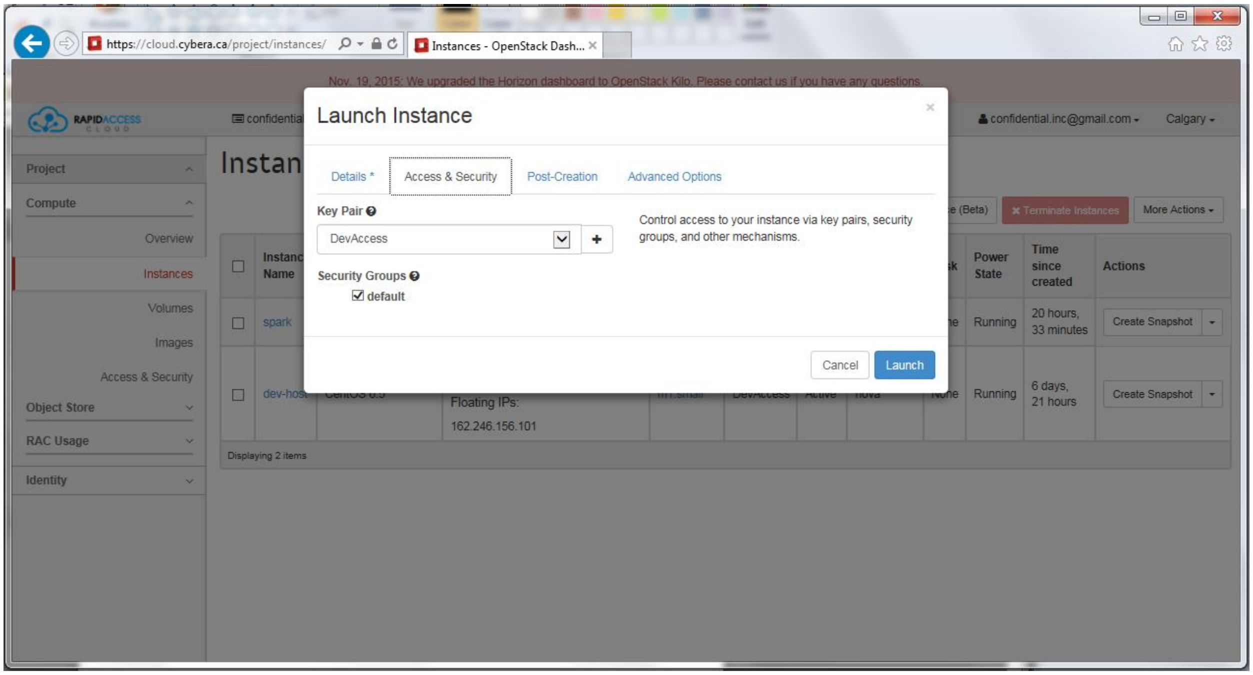
Name	m1.tiny
VCPUs	1
Root Disk	5 GB
Ephemeral Disk	0 GB
Total Disk	5 GB
RAM	512 MB

Project Limits

Number of Instances: 2 of 8 Used

FIGURE 2 - LAUNCH INSTANCE

#Step 3: Setup Authentication Profile



Nov. 19, 2015: We upgraded the Horizon dashboard to OpenStack Kilo. Please contact us if you have any questions.

Launch Instance

Details * | Access & Security | Post-Creation | Advanced Options

Key Pair: DevAccess

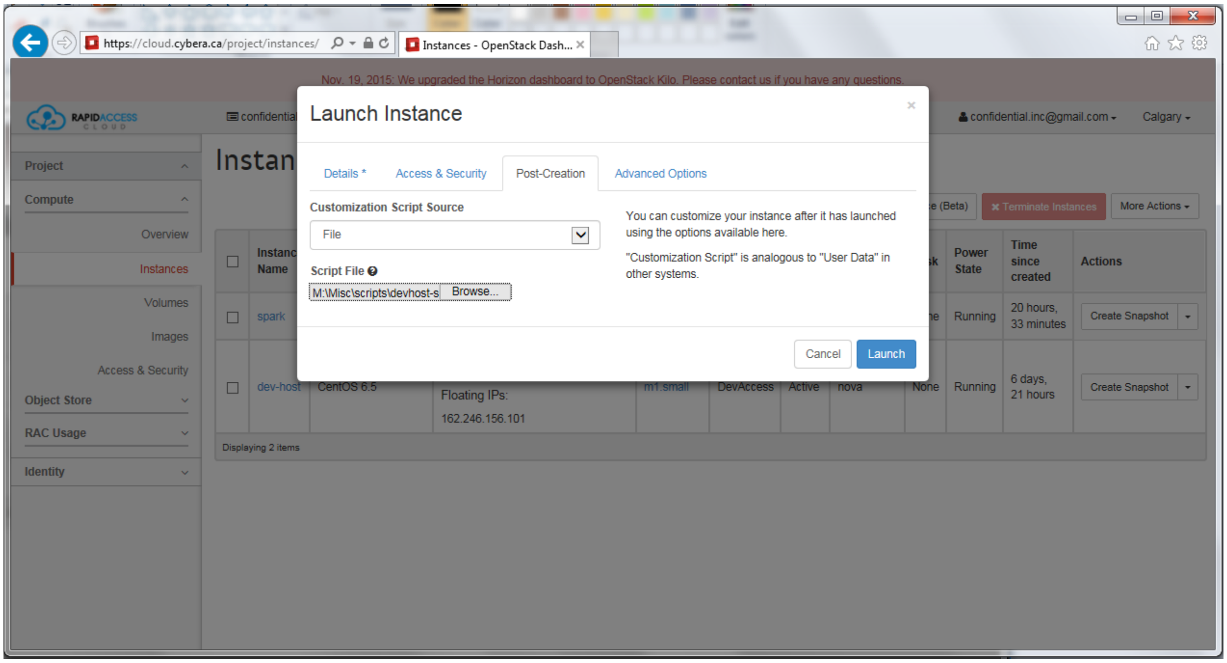
Security Groups: ☑ default

Control access to your instance via key pairs, security groups, and other mechanisms.

Cancel | Launch

FIGURE 3 - AUTHENTICATION INFO

#Step 4: Configure Instance w/ Cloud-Init Script



Nov. 19, 2015: We upgraded the Horizon dashboard to OpenStack Kilo. Please contact us if you have any questions.

Launch Instance

Details * | Access & Security | Post-Creation | Advanced Options

Customization Script Source: File

Script File: M:\Misc\scripts\devhost-s [Browse...]

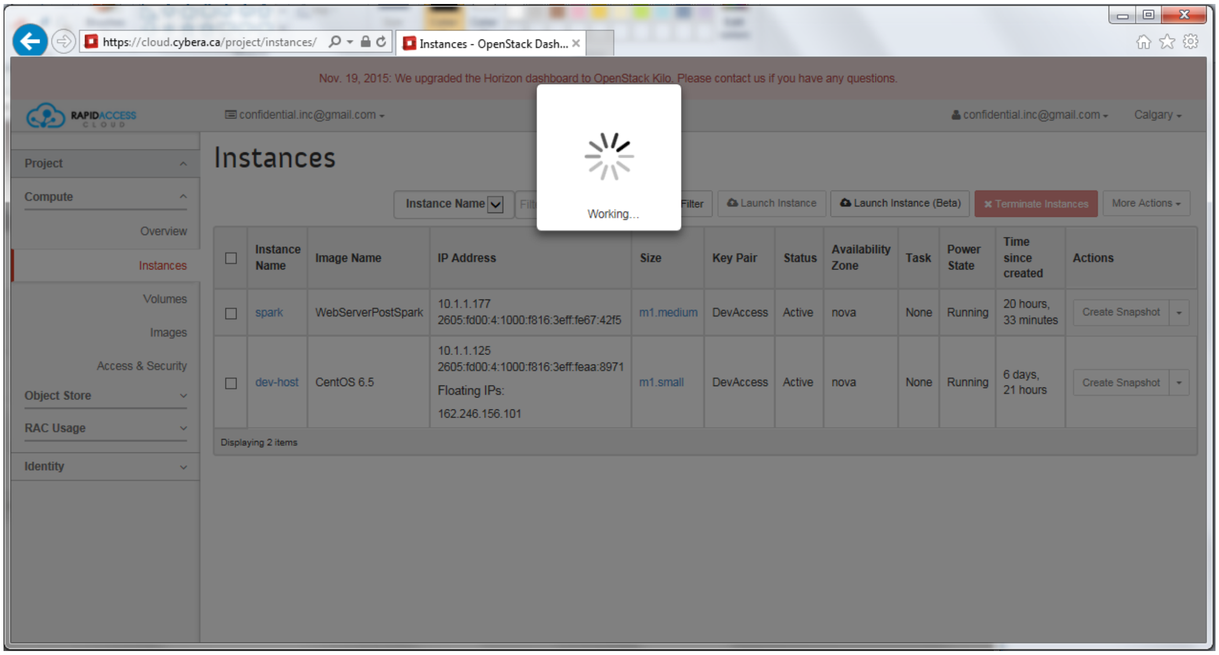
You can customize your instance after it has launched using the options available here.

"Customization Script" is analogous to "User Data" in other systems.

[Cancel] [Launch]

FIGURE 4 - CLOUD-INIT SCRIPT FILE

#Step 5: Launch



Nov. 19, 2015: We upgraded the Horizon dashboard to OpenStack Kilo. Please contact us if you have any questions.

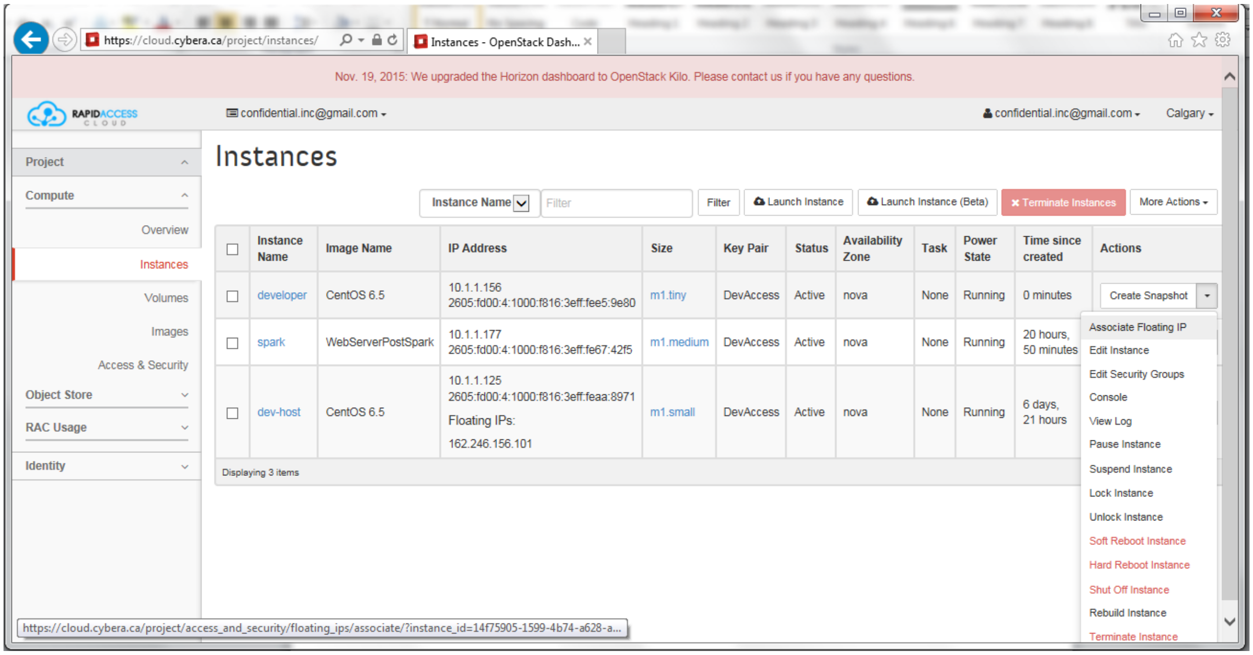
Instances

☐	Instance Name	Image Name	IP Address	Size	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Actions
☐	spark	WebServerPostSpark	10.1.1.177 2605:fd00:4:1000:f816:3eff:fe67:42f5	m1.medium	DevAccess	Active	nova	None	Running	20 hours, 33 minutes	Create Snapshot
☐	dev-host	CentOS 6.5	10.1.1.125 2605:fd00:4:1000:f816:3eff:feaa:8971 Floating IPs: 162.246.156.101	m1.small	DevAccess	Active	nova	None	Running	6 days, 21 hours	Create Snapshot

Displaying 2 items

FIGURE 5 - INSTANCE LAUNCHING

#Step 6: Associate Floating-IP



Nov. 19, 2015: We upgraded the Horizon dashboard to OpenStack Kilo. Please contact us if you have any questions.

Instances

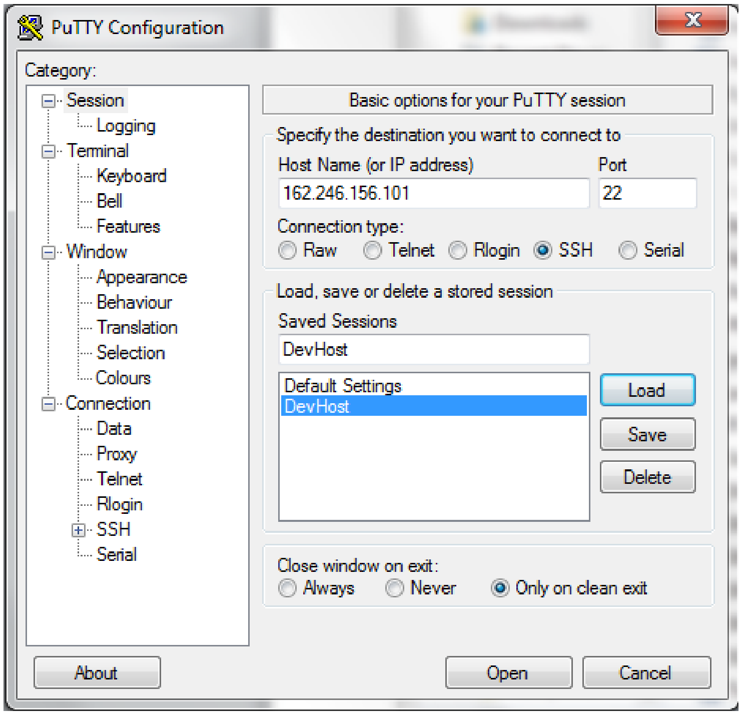
□	Instance Name	Image Name	IP Address	Size	Key Pair	Status	Availability Zone	Task	Power State	Time since created	Actions
□	developer	CentOS 6.5	10.1.1.156 2605:fd00:4:1000:f816:3eff:fee5:9e80	m1.tiny	DevAccess	Active	nova	None	Running	0 minutes	Create Snapshot
□	spark	WebServerPostSpark	10.1.1.177 2605:fd00:4:1000:f816:3eff:fe67:42f5	m1.medium	DevAccess	Active	nova	None	Running	20 hours, 50 minutes	
□	dev-host	CentOS 6.5	10.1.1.125 2605:fd00:4:1000:f816:3eff:feaa:8971 Floating IPs: 162.246.156.101	m1.small	DevAccess	Active	nova	None	Running	6 days, 21 hours	

Displaying 3 items

https://cloud.cybera.ca/project/access_and_security/floating_ips/associate/?instance_id=14f75905-1599-4b74-a628-a...

FIGURE 6 - ASSOCIATING FLOATING IP

#Step 7: SSH into Instance



Specify the destination you want to connect to

Host Name (or IP address): 162.246.156.101

Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

Load, save or delete a stored session

Saved Sessions: DevHost

Default Settings
DevHost

Close window on exit: ○ Always ○ Never ● Only on clean exit

FIGURE 7 - SSH INTO INSTANCE

CLI Method:

#Step 1: Login to Cloud

Before creating an instance – login to the OpenStack Cloud

```
$>: smack-login
 ____  __  __    _    ____ _  __  _____
/ ___||  \/  |  / \  / ___| |/ / | ____|_ __   ___ _ __ __ _ _   _
\___ \| |\/| | / _ \| |   | ' /  |  _| | '_ \ / _ \ '__/ _` | | | |
 ___) | |  | |/ ___ \ |___| . \  | |___| | | |  __/ | | (_| | |_| |
|____/|_|  |_/_/   \_\____|_|\_\ |_____|_| |_|\___|_|  \__, |\__, |
                                                       |___/ |___/
         _____                            _   _
        |  ___|__  _ __ ___  ___ __ _ ___| |_(_)_ __   __ _
        | |_ / _ \| '__/ _ \/ __/ _` / __| __| | '_ \ / _` |
        |  _| (_) | | |  __/ (_| (_| \__ \ |_| | | | | (_| |
        |_|  \___/|_|  \___|\___\__,_|___/\__|_|_| |_|\__, |
                                                      |___/
                      +-+-+-+-+-+ +-+-+-+-+-+
                      |C|l|o|u|d| |L|o|g|i|n|
                      +-+-+-+-+-+ +-+-+-+-+-+
Please enter your SMACK Openstack username: <enter your username>
Please enter your SMACK Openstack password: <enter your password>
Please enter your Project: <blank or 'SMACK'>
```

#Step 2: Launch an Instance

At the command line – enter the following to launch an instance. It will ask you for a variety of information – such as Name, and Flavour(size), etc..

```
$>: smack-mknode
```

#Step 3: Configure Instance

Answer all the questions as you feel are necessary. For defaults, you can leave all answers blank and it will use the most common setting.

```
$>: smack-mknode
 ____  __  __    _    ____ _  __  _____
/ ___||  \/  |  / \  / ___| |/ / | ____|_ __   ___ _ __ __ _ _   _
\___ \| |\/| | / _ \| |   | ' /  |  _| | '_ \ / _ \ '__/ _` | | | |
 ___) | |  | |/ ___ \ |___| . \  | |___| | | |  __/ | | (_| | |_| |
|____/|_|  |_/_/   \_\____|_|\_\ |_____|_| |_|\___|_|  \__, |\__, |
                                                       |___/ |___/
         _____                            _   _
        |  ___|__  _ __ ___  ___ __ _ ___| |_(_)_ __   __ _
        | |_ / _ \| '__/ _ \/ __/ _` / __| __| | '_ \ / _` |
        |  _| (_) | | |  __/ (_| (_| \__ \ |_| | | | | (_| |
        |_|  \___/|_|  \___|\___\__,_|___/\__|_|_| |_|\__, |
                                                      |___/
For Defaults Just Press Enter at Prompt.
```

```
Name (*default):        <enter instance name>

Flavour (*m1.tiny):     <enter instance size>

Key (*DevAccess):       <enter keypair name>

Setup Script (*devhost-setup.sh): <enter cloud-init script>


Launching VM Instance: default
  OS Type: linux
  Flavour: m1.tiny
  Image:    CentOS 6.5
  Security Group: Default
  Key: DevAccess
  Launch Script: devhost-setup.sh
      ------------
```

#Step 4: Associate Floating-IP

Enter the following into the command line to associate the floating IP to the instance.

```
$>: nova floating-ip-associated <instance name> <floating-ip>
```

To see a list of available floating IP addresses, use the following:

```
$>: nova floating-ip-list
```

#Step 5: SSH into Instance

Use the following to SSH into the instance from the command line.

```
$>: ssh –i <access_key.pem> centos@<floating ip address>
```

devhost-setup.sh

```
#!/bin/bash
#-----------------------------------------------------------#
#                 SMACK ENERGY FORECASTING                  #
#-----------------------------------------------------------#
#           - Development Host Init Script for Nodes -      #
#-----------------------------------------------------------#
#     Draft Cloud-Init Script for Setting up a development  #
#     host. This is the host that will be used to interact  #
#     with the development and handle the deployment of     #
#     node clusters via CLI.                                #
#                                                           #
#     Think of this computer as the point of interaction    #
#     for development - not production                      #
#-----------------------------------------------------------#
#                 - MODIFIED: Nov 24, 2015 -                #
#-----------------------------------------------------------#


# DIRECTORY AND FILE INFORMATION
#-----------------------------------
SMACK_DIR=/usr/local/smack
SMACK_DIR_BIN=/usr/local/smack/bin
SMACK_DIR_LOG=/usr/local/smack/log
SMACK_LOAD=$SMACK_DIR_LOG/smack_loaded
SMACK_INSTALL_LOG=$SMACK_DIR_LOG/install_log
# Log Reporting
echo -e "\n### INSTALL BEGINNING ###" >> $SMACK_INSTALL_LOG


# GENERATE DIRECTORY STORAGE
#-----------------------------------
mkdir $SMACK_DIR
mkdir $SMACK_DIR_BIN
mkdir $SMACK_DIR_LOG
# Log Reporting
echo -e "\nDIRECTORIES: COMPLETE" >> $SMACK_INSTALL_LOG


# UPDATE PACKAGES AND TOOLCHAIN
#-----------------------------------
yum -y update
yum -y install gcc-c++ wget curl curl-devel figlet python
yum -y install make binutils git nmap man maven
yum -y install nano python-devel python-pip links
yum -y groupinstall "Development Tools"
# Log Reporting
echo -e "\nTOOLCHAIN: COMPLETE" >> $SMACK_INSTALL_LOG


# INSTALL OPENSTACK CLIENT SOFTWARE
#-----------------------------------
pip install python-openstackclient
pip install python-swiftclient
pip install --upgrade setuptools
```

```
# Log Reporting
echo -e "\nOPENSTACK CLIENTS: COMPLETE" >> $SMACK_INSTALL_LOG


# ENSURE ONLY RUNS ONCE
#-----------------------------------
if ! [ -e $SMACK_LOAD ]; then

# INSTALL PYTHON 3.5
#-----------------------------------
mkdir /tmp/python3
cd /tmp/python3
wget "https://www.python.org/ftp/python/3.5.0/Python-3.5.0.tgz"
tar -xzvf Python-3.5.0.tgz
cd Python-3.5.0
./configure
make
make install
cd /
rm -rf /tmp/python3
# Log Reporting
echo -e "\nPYTHON 3.5: COMPLETE" >> $SMACK_INSTALL_LOG


# INSTALL JAVA RUNTIME VERSION (7/8)
#------------------------------------
#
#

# INSTALL ADDITIONAL SOFTWARE BELOW
#-----------------------------------
#
#


# ADD WELCOME MESSAGE TO INSTANCE
#-----------------------------------
cat << EOT >> /etc/bashrc
# Add Executable Path
echo -e "\nAdding Path: $SMACK_DIR_BIN\n"
export PATH=\$PATH:$SMACK_DIR_BIN
# SMACK Command Aliases
alias smack-login="source $SMACK_DIR_BIN/smack-login"
alias smack-logout="source $SMACK_DIR_BIN/smack-logout"
# Display Title Screen for SMACK
figlet -c SMACK Energy Forecasting
echo -e "\t\tSMACK Energy Forecasting - Making an Impact\n"
echo -e "Welcome to the Development Machine:\n"
echo -e "\n#TIP---For a list of commands type smack and press tab.\n"
EOT
# Log Reporting
echo -e "\nWELCOME: COMPLETE" >> $SMACK_INSTALL_LOG


# LIST NODES COMMAND (smack-lsnode)
#-----------------------------------
cat << EOF >> $SMACK_DIR_BIN/smack-lsnode
```

```
#!/bin/bash
#
#   NAME:   smack-lsnodes:
#
#     USAGE:      Displays Nodes in the cluster and their status
#
# Output Welcome Screen
if [ -z "\$OS_USERNAME" ] || [ -z "\$OS_PASSWORD" ]; then
      echo -e "Error: You are not logged in.\n\tPlease run 'smack-
login' and then try again."
      exit 1
else
      figlet -c SMACK Energy Forecasting
fi
# List Nodes in Cloud
figlet -cf digital NODES LISTING
nova --os-user-name \$OS_USERNAME \\
          --os-project-name \$OS_PROJECT_NAME \\
            --os-password \$OS_PASSWORD \\
            --os-region-name \$OS_REGION \\
            --os-auth-url \$OS_AUTH_URL \\
            --os-auth-url \$OS_AUTH_URL \\
            list
EOF
# Log Reporting
if [ -e "$SMACK_DIR_BIN/smack-lsnode" ]; then
      echo -e "\nLIST NODES: COMPLETE" >> $SMACK_INSTALL_LOG
else
      echo -e "\nLIST NODES: ERROR" >> $SMACK_INSTALL_LOG
fi


# CREATE NODE IN CLOUD COMMAND (smack-mknode)
#--------------------------------------------
cat << EOF > $SMACK_DIR_BIN/smack-mknode
#!/bin/bash
#
#   NAME:   smack-lsnodes:
#
#     USAGE:      Displays Nodes in the cluster and their status
#
### Definitions and Auth Information

# Output Welcome Screen
if [ -z "\$OS_USERNAME" ] || [ -z "\$OS_PASSWORD" ]; then
      echo -e "Error: You are not logged in. \n\tPlease run 'smack-
login' and then try again."
      exit 1
else
      figlet -c SMACK Energy Forecasting
fi

# Default Instance Information
INT_NAME=default
INT_FLAVOR=m1.tiny
INT_IMAGE=907f21d1-305c-4dee-a64a-43fc1a3701a4
INT_OS=linux
```

```
INT_KEY=DevAccess
INT_SECURITY=Default
INT_SCRIPT=setup-node.sh

# Boot and Launch Instance
echo "For Defaults Just Press Enter at Prompt."
echo -e "\tName (*default):"
read NAME
echo -e "\tFlavour (*m1.tiny):"
read FLAVOUR
echo -e "\tKey (*DevAccess):"
read KEY
echo -e "\tSetup Script (*setup-node.sh):"
read SCRIPT

#  Check for new name and change if necessary
if ! [ -z "\$NAME" ]; then
          INT_NAME=\$NAME
fi
if ! [ -z "\$FLAVOUR" ]; then
      INT_FLAVOR=\$FLAVOUR
fi
if ! [ -z "\$KEY" ]; then
      INT_KEY=\$KEY
fi
if ! [ -z "\$SCRIPT" ]; then
      INT_SCRIPT=\$SCRIPT
fi

# Display instance information
echo -e "Launching VM Instance: \$INT_NAME"
echo -e "\tOS Type: \$INT_OS\n\tFlavour: \$INT_FLAVOR"
echo -ne "\tImage: \$(nova --os-user-name \$OS_USERNAME \\
            --os-project-name \$OS_PROJECT_NAME \\
            --os-password \$OS_PASSWORD \\
            --os-region-name \$OS_REGION \\
            --os-auth-url \$OS_AUTH_URL \\
            image-list | grep \$INT_IMAGE | sed 's/ACTIVE//' | sed
s/\$INT_IMAGE// | sed 's/|//g')"
echo -e "\n\tSecurity Group: \$INT_SECURITY"
echo -e "\tKey: \$INT_KEY"
echo -e "\tLaunch Script: \$INT_SCRIPT"
echo -e "\t------------\n"

# Boot up new instance in cloud
nova --os-user-name \$OS_USERNAME \\
    --os-project-name \$OS_PROJECT_NAME \\
    --os-password \$OS_PASSWORD \\
    --os-region-name \$OS_REGION \\
    --os-auth-url \$OS_AUTH_URL \\
      boot \\
      --flavor \$INT_FLAVOR \\
      --image \$INT_IMAGE \\
      --key-name \$INT_KEY \\
      --user-data \$INT_SCRIPT \\
      --security-group \$INT_SECURITY \\
      \$INT_NAME
```

```
EOF
# Log Reporting
if [ -e "$SMACK_DIR_BIN/smack-mknode" ]; then
      echo -e "\nMAKE NODE: COMPLETE" >> $SMACK_INSTALL_LOG
else
      echo -e "\nMAKE NODE: ERROR" >> $SMACK_INSTALL_LOG
fi


# LOGIN COMMAND (smack-login)
#-----------------------------------
cat << EOF > $SMACK_DIR_BIN/smack-login
#!/bin/bash
#
#     Login to Cluster for Easy Use
#

# Output Welcome Screen
figlet -c SMACK Energy Forecasting
figlet -cf digital Cloud Login
# Login and Set Variables
read -p "Please enter your SMACK Openstack username: " UNAME
stty -echo
read -p "Please enter your SMACK Openstack password: " PASSWD
stty echo
read -p "Please enter your Project (ie. blank for personal or enter
'SMACK'): " PROJECT

# URLs for API Access (may need to change)
export KEYSTONE_URL="https://keystone-yyc.cloud.cybera.ca:5000/v2.0"
export NOVA_URL="https://nova-
yyc.cloud.cybera.ca:8774/v2/2b86ecd5b18f4fafb1d55adb79072def"
export CINDER_URL="https://cinder-
yyc.cloud.cybera.ca:8776/v1/2b86ecd5b18f4fafb1d55adb79072def"
export CINDER2_URL="https://cinder-
yyc.cloud.cybera.ca:8776/v2/2b86ecd5b18f4fafb1d55adb79072def"
export GLANCE_URL="http://glance-yyc.cloud.cybera.ca:9292"
export EC2_URL="https://nova-yyc.cloud.cybera.ca:8773/services/Cloud"
export SWIFT_URL="https://swift-
yyc.cloud.cybera.ca:8080/v1/AUTH_2b86ecd5b18f4fafb1d55adb79072def"
# Login Info
export OS_USERNAME=\$UNAME
export OS_PASSWORD=\$PASSWD
if [ -z "\$PROJECT" ]; then
      export OS_PROJECT_NAME=\$UNAME
else
      export OS_PROJECT_NAME=\$PROJECT
fi
export OS_AUTH_URL=\$KEYSTONE_URL
export OS_REGION=Calgary
export OS_ZONE=Nova
EOF
# Log Reporting
if [ -e "$SMACK_DIR_BIN/smack-login" ]; then
      echo -e "\nLOGIN: COMPLETE" >> $SMACK_INSTALL_LOG
else
      echo -e "\nLOGIN: ERROR" >> $SMACK_INSTALL_LOG
```

```
fi


# LOGOUT COMMAND (smack-logout)
#-----------------------------------
cat << EOF > $SMACK_DIR_BIN/smack-logout
#!/bin/bash
#
#     Easy Logout for Cluster
#
figlet -c SMACK Energy Forecasting
figlet -cf digital Logging Out...
# Unset Login Details
unset OS_USERNAME
unset OS_PASSWORD
unset OS_PROJECT_NAME
unset OS_REGION
unset OS_AUTH_URL
unset OS_ZONE
# Unset URLs
unset KEYSTONE_URL
unset GLANCE_URL
unset CINDER_URL
unset CINDER2_URL
unset SWIFT_URL
unset NOVA_URL
unset EC2_URL
EOF
# Log Reporting
if [ -e "$SMACK_DIR_BIN/smack-logout" ]; then
      echo -e "\nLOGOUT: COMPLETE" >> $SMACK_INSTALL_LOG
else
      echo -e "\nLOGOUT: ERROR" >> $SMACK_INSTALL_LOG
fi

# TERMINATE INSTANCE COMMAND
#-----------------------------------
#
#

# ASSOCIATE FLOATING IP COMMAND
#-----------------------------------
#
#

# UPLOAD FILE TO CONTAINER COMMAND
#-----------------------------------
#
#

# DOWNLOAD FILE FROM CONTAINER COMMAND
#-----------------------------------
#
#
```

```
# ADD ADDITIONAL COMMANDS BELOW
#-----------------------------------
#
#
#
#
#


# SET PERMISSIONS FOR COMMANDS
#-----------------------------------
chmod 755 $SMACK_DIR_BIN
chmod 600 $SMACK_DIR_LOG/*
chmod +x $SMACK_DIR_BIN/*
# Log Reporting
echo -e "\nPERMISSIONS: COMPLETE" >> $SMACK_INSTALL_LOG


# SET FILE FOR COMPLETION
#-----------------------------------
touch $SMACK_LOAD
echo -e "\n### INSTALL: COMPLETE ###" >> $SMACK_INSTALL_LOG

fi

# FINISHED
```