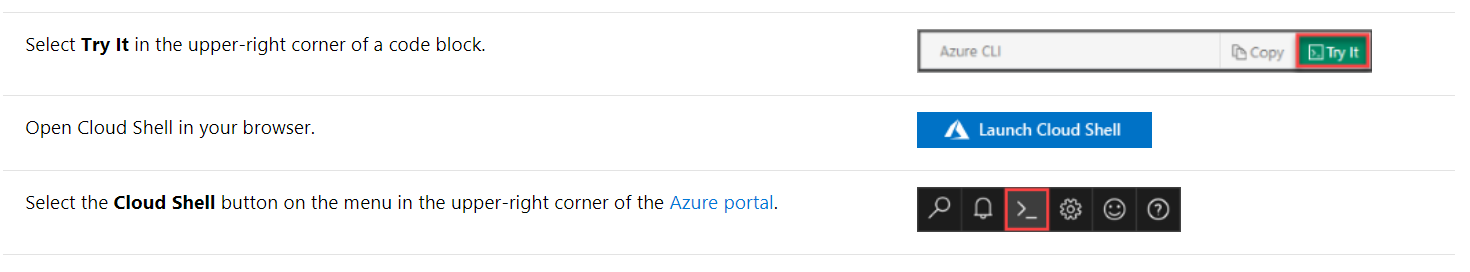
Prepare Lab Environment

1. Create and Manage Linux VMs with the Azure CLI 2.0
   1. Open Azure Cloud Shell

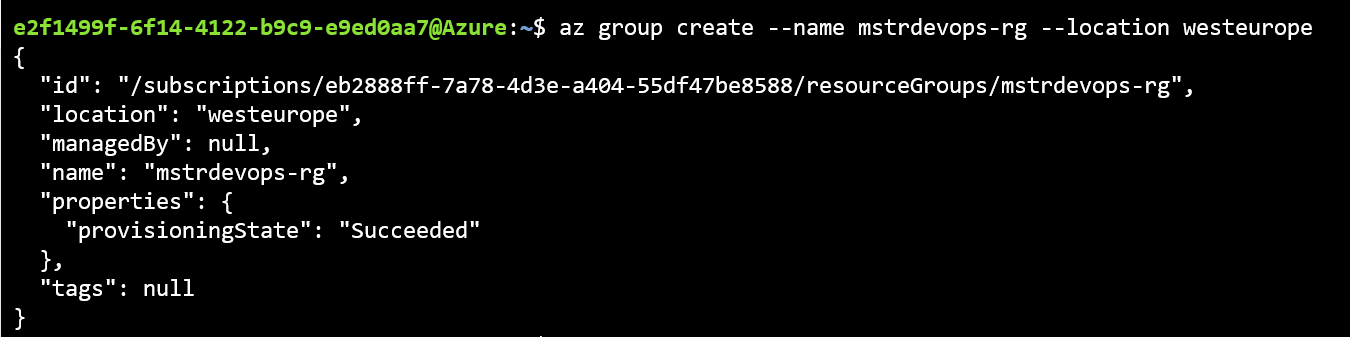


If you choose to install and use the CLI locally, this tutorial requires that you are running the Azure CLI version 2.0.30 or later. Run az --version to find the version. If you need to install or upgrade, see [Install Azure CLI 2.0.](https://docs.microsoft.com/en-us/cli/azure/install-azure-cli?view=azure-cli-latest)

* 1. Create resource group

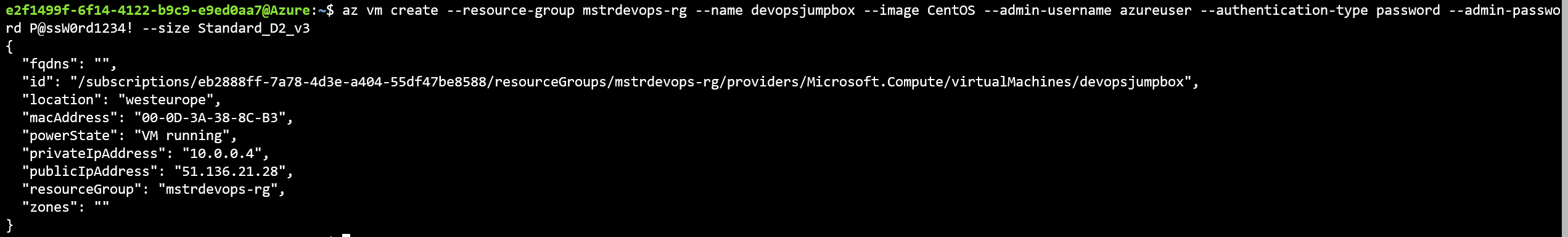
Create a resource group with the [az group create](https://docs.microsoft.com/cli/azure/group" \l "az_group_create) command.

az group create --name mstrdevops-rg --location westeurope

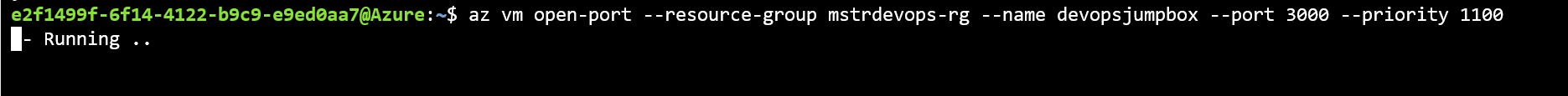


* 1. Create Linux VM and Allow Firewall Ports

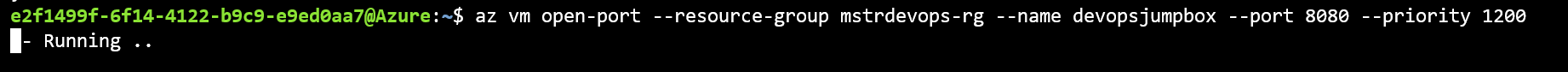
az vm create --resource-group mstrdevops-rg --name devopsjumpbox --image CentOS --admin-username azureuser --authentication-type password --admin-password P@ssW0rd1234! --size Standard\_D2\_v3



az vm open-port --resource-group mstrdevops-rg --name devopsjumpbox --port 3000 --priority 1100



az vm open-port --resource-group mstrdevops-rg --name devopsjumpbox --port 8080 --priority 1200



az vm open-port --resource-group mstrdevops-rg --name devopsjumpbox --port 8081 --priority 1300

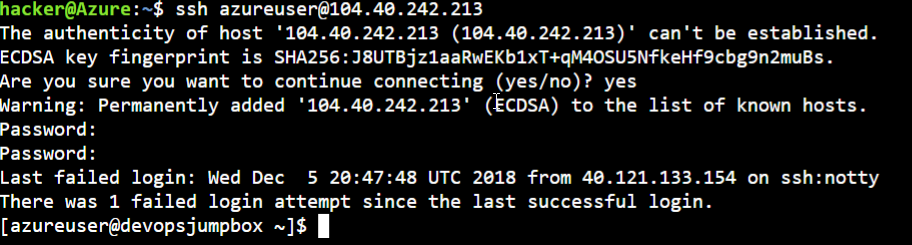
* 1. Connect to VM

You can now connect to the VM with SSH from your local computer. Replace the example IP address with the publicIpAddress. To get public IP Address run the command.

az vm list-ip-addresses -n devopsjumpbox --query [0].virtualMachine.network.publicIpAddresses[0].ipAddress -o tsv

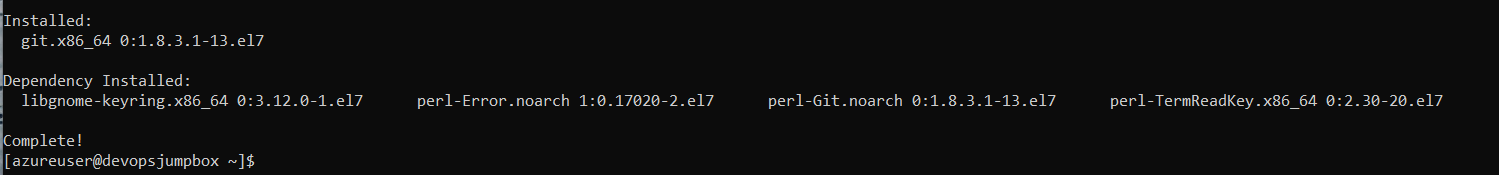


ssh [azureuser@104.40.242.213](mailto:azureuser@104.40.242.213)

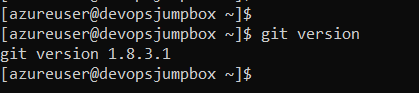


1. Initial Setup and Running App on Local Machine
   1. Install Git on Linux VM (Jumpbox)

sudo yum install git

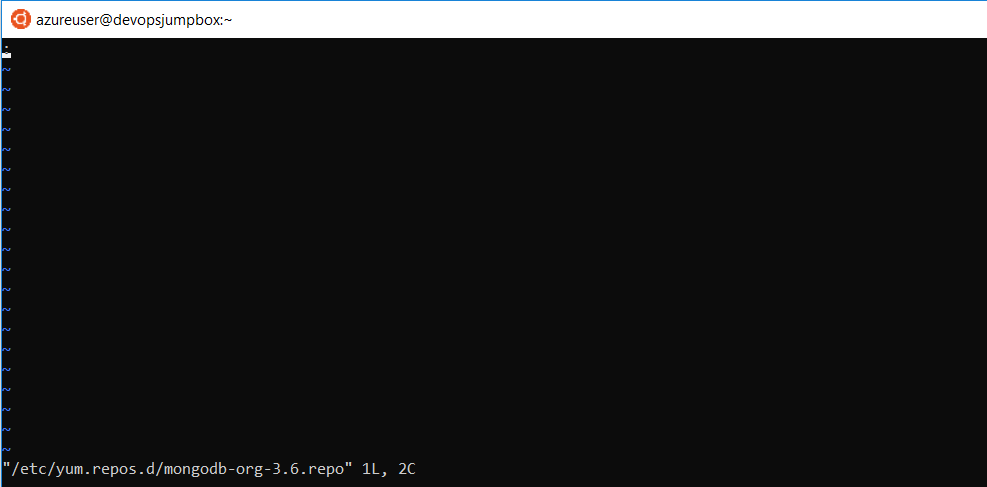


git version

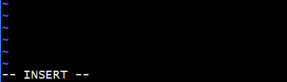


* 1. Install MongoDB on Jumpbox

sudo vi /etc/yum.repos.d/mongodb-org-3.6.repo



Select “i” on your keyboard. You’ll see the bottom of the window showing INSERT mode.



**NOTE: Type the following into the editor, as you may have errors with copying and pasting.**

[mongodb-org-3.6]

name=MongoDB Repository

baseurl=https://repo.mongodb.org/yum/redhat/$releasever/mongodb-org/3.6/x86\_64/

gpgcheck=1

enabled=1

gpgkey=https://www.mongodb.org/static/pgp/server-3.6.asc

When you are finished typing, hit the Esc key and type “:wq” and hit the Enter key to save the changes and close the file.

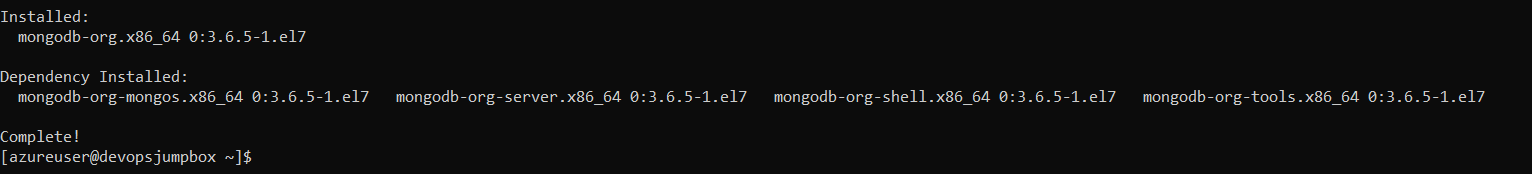
<Esc>

:wq!

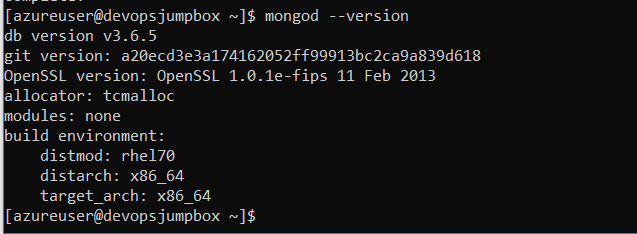
<Enter>



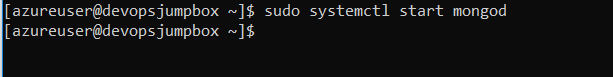
sudo yum install -y mongodb-org



mongod -version

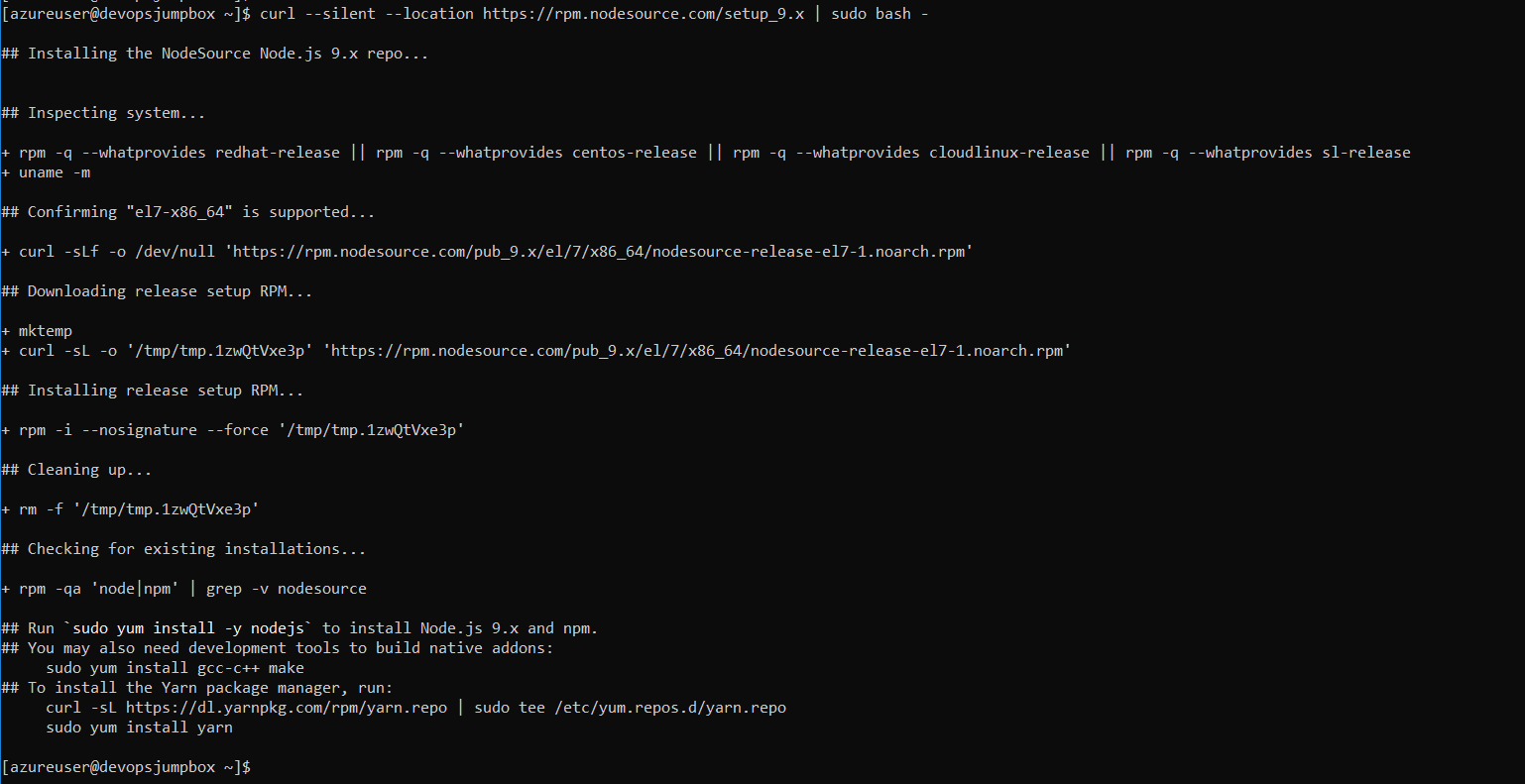


sudo systemctl start mongod

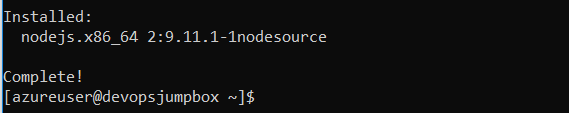


* 1. Install NodeJS on Jumpbox

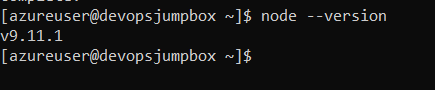
curl --silent --location https://rpm.nodesource.com/setup\_9.x | sudo bash -



sudo yum -y install nodejs

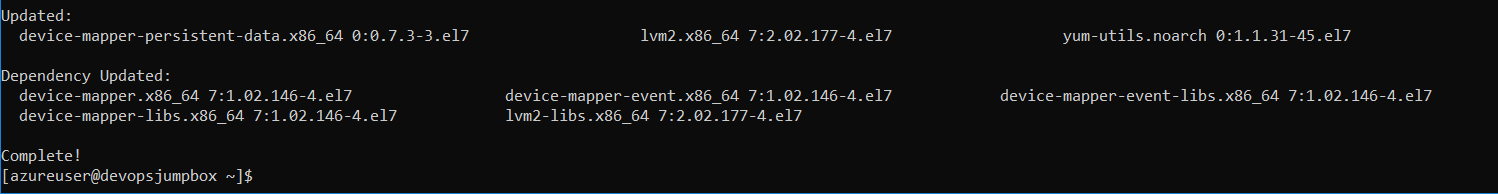


node --version

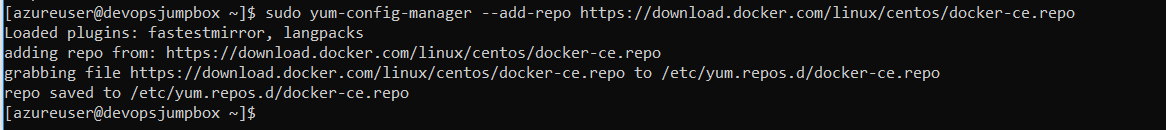


* 1. Install DockerCE on Jumpbox

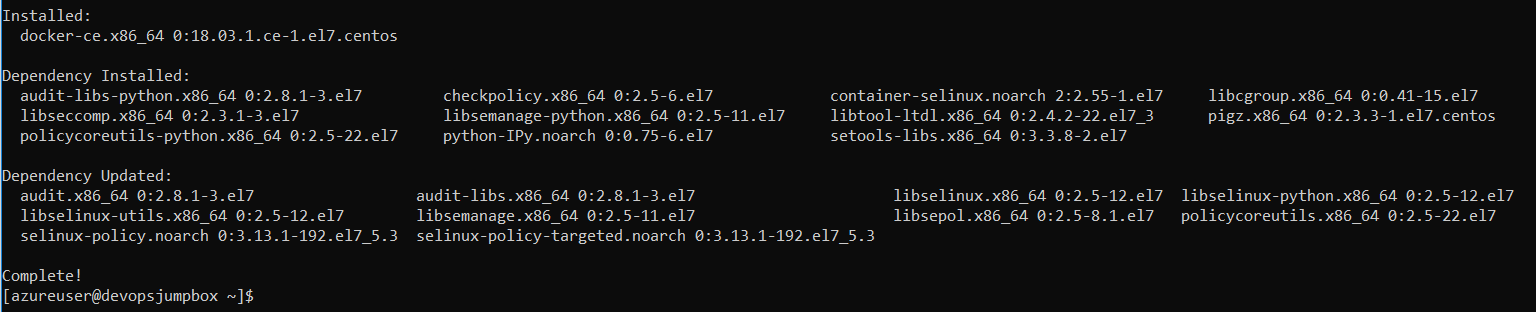
sudo yum install -y yum-utils device-mapper-persistent-data lvm2



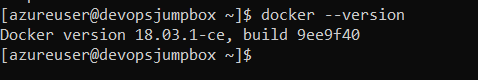
sudo yum-config-manager --add-repo <https://download.docker.com/linux/centos/docker-ce.repo>



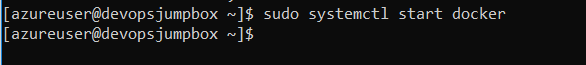
sudo yum install -y docker-ce



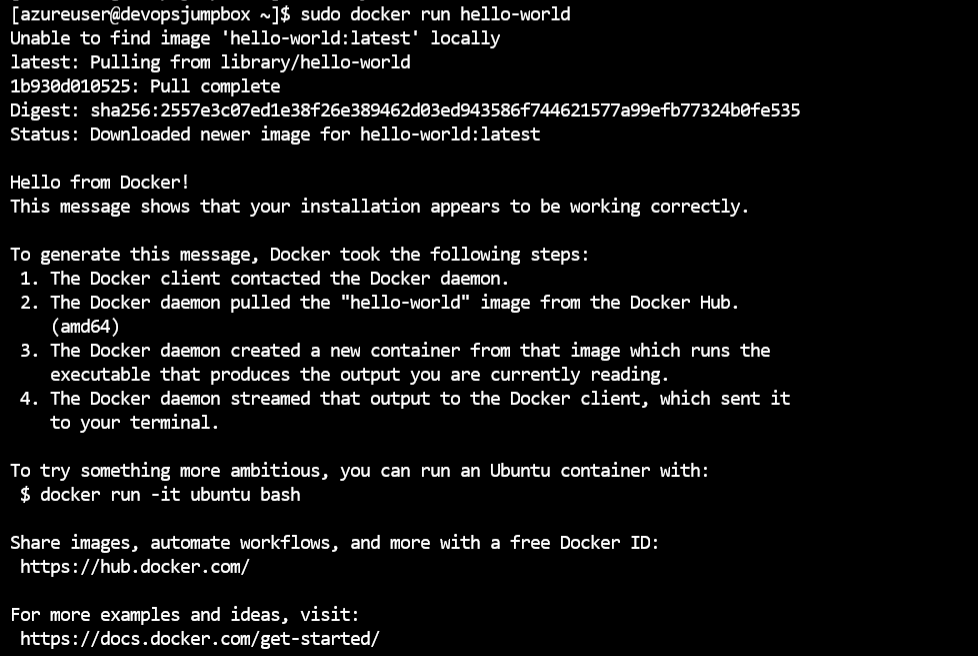
docker --version



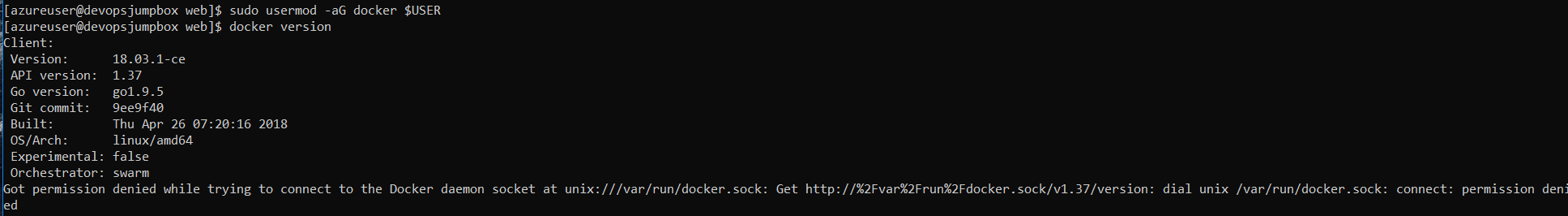
sudo systemctl start docker



sudo docker run hello-world



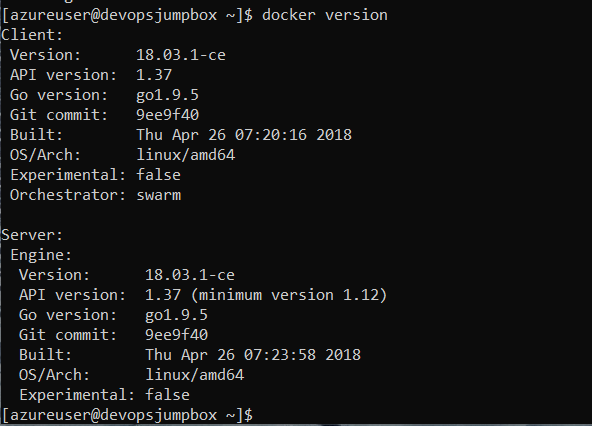
sudo usermod -aG docker $USER



In order for the user permission changes to take effect, exit the SSH session by typing ‘exit’, then press <Enter>. Repeat connect from establish the SSH session. P@ssW0rd1234!

ssh [azureuser@104.40.242.213](mailto:azureuser@104.40.242.213)

docker version

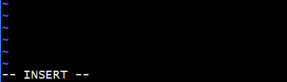


* 1. Install Kubectl on Jumpbox

sudo vi /etc/yum.repos.d/kubernetes.repo



Select “i” on your keyboard. You’ll see the bottom of the window showing INSERT mode.



[kubernetes]

name=Kubernetes

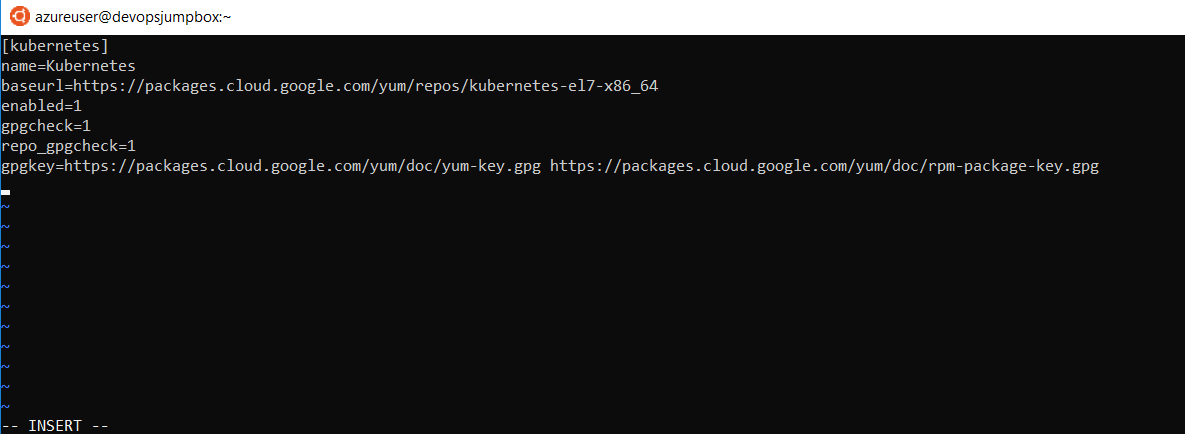
baseurl=https://packages.cloud.google.com/yum/repos/kubernetes-el7-x86\_64

enabled=1

gpgcheck=1

repo\_gpgcheck=1

gpgkey=https://packages.cloud.google.com/yum/doc/yum-key.gpg <https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg>



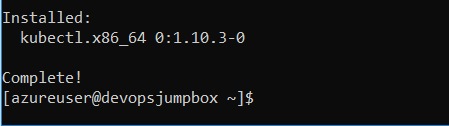
When you are finished typing, hit the Esc key and type “:wq!” and hit the Enter key to save the changes and close the file.

<Esc>

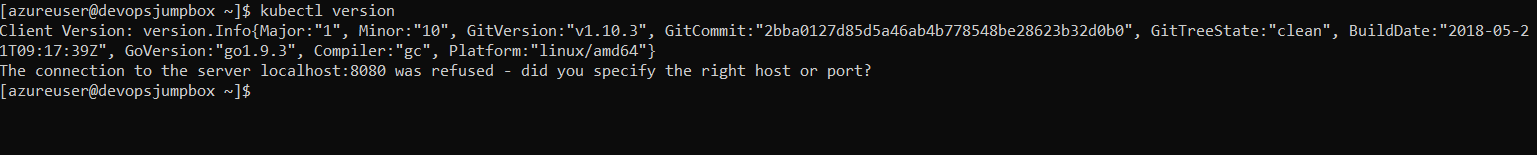
:wq!

<Enter>

sudo yum install kubectl



kubectl version

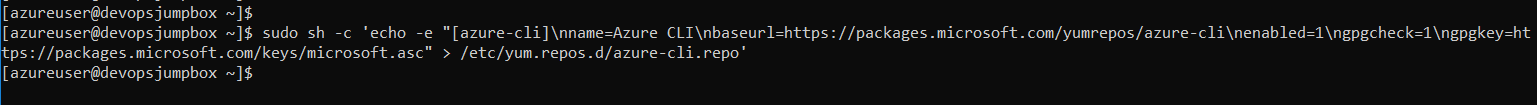


* 1. Install AzureCLI on Jumpbox

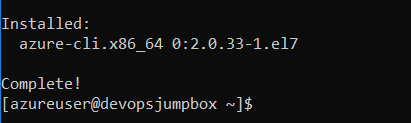
sudo rpm --import <https://packages.microsoft.com/keys/microsoft.asc>



sudo sh -c 'echo -e "[azure-cli]\nname=Azure CLI\nbaseurl=https://packages.microsoft.com/yumrepos/azure-cli\nenabled=1\ngpgcheck=1\ngpgkey=https://packages.microsoft.com/keys/microsoft.asc" > /etc/yum.repos.d/azure-cli.repo'

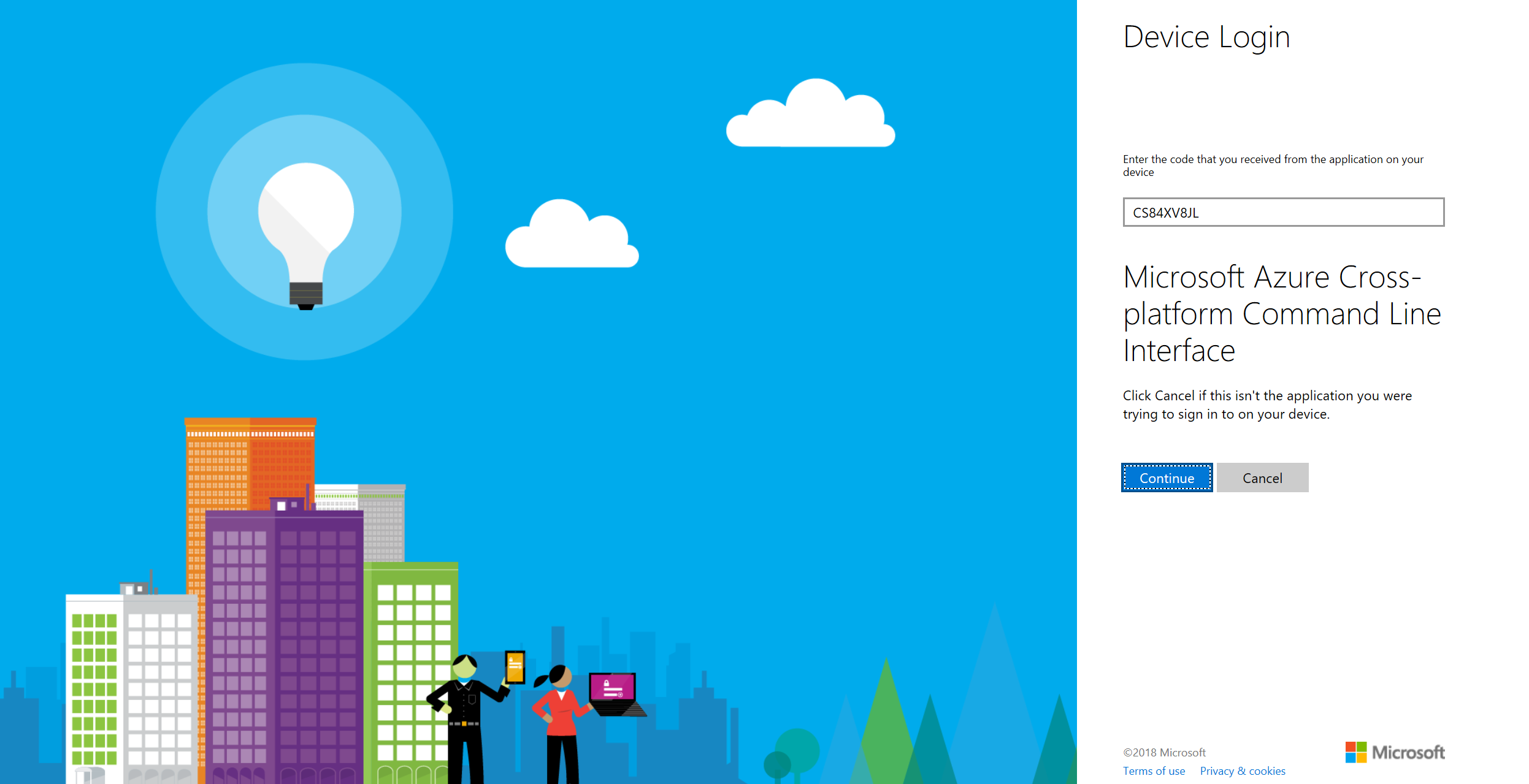


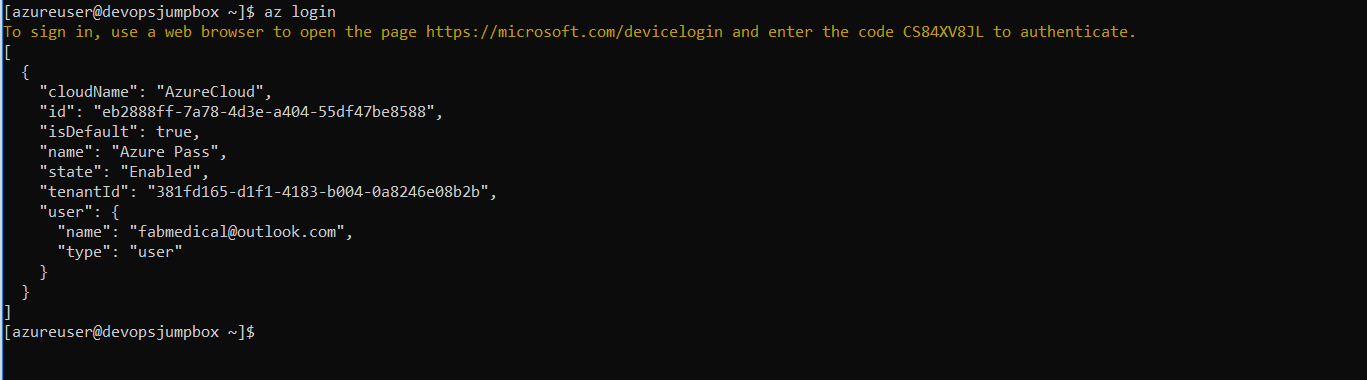
sudo yum install -y azure-cli



az login

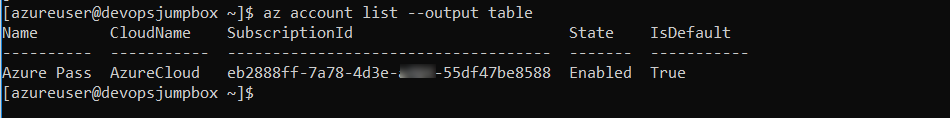




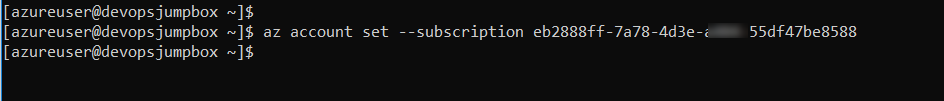


az account list --output table

Please note the subscriptionid value.



az account set --subscription "use the subscriptionid"



* 1. Install Jenkins on Jumpbox

Jenkins is a Java application, so the first step is to install Java. Run the following command to install the OpenJDK 8 package:

sudo yum install -y java-1.8.0-openjdk-devel

Enable the Jenkins repository:

curl --silent --location http://pkg.jenkins-ci.org/redhat-stable/jenkins.repo | sudo tee /etc/yum.repos.d/jenkins.repo

And add the repository to your system:

sudo rpm --import <https://jenkins-ci.org/redhat/jenkins-ci.org.key>

Install the latest stable version of Jenkins:

sudo yum install -y jenkins

To Change Jenkins port open Jenkins configuration File:

sudo vi /etc/sysconfig/jenkins

Change http port number:

JENKINS\_PORT="8081"

Give permission to Jenkins user

sudo usermod -aG wheel jenkins

Give Jenkins user to docker deamon

sudo usermod -a -G docker jenkins

sudo systemctl restart docker

Make jenkins user sudoer for the lab environment.

Run visudo and at the bottom add a line similar to this;

sudo visudo

jenkins ALL=(ALL) NOPASSWD: ALL

Start Jenkins:

sudo systemctl start jenkins

Check the status:

systemctl status jenkins

Enable the Jenkins service to start on system boot:

sudo systemctl enable jenkins

Get initialAdminPassword:

sudo cat /var/lib/jenkins/secrets/initialAdminPassword

You should see a 32-character long alphanumeric password as shown below:

Output:

2115173b548f4e99a203ee99a8732a32

Copy the password from your terminal, open your browser, type <http://104.40.242.213:8081> (Replace the example IP address with the **publicIpAddress** noted before) in your browser and paste copied password into the Administrator password field. Once you paste it click **continue**.

To Complete initial setup follow these steps:

* Choose Select plugins to install
* Search for GitHub in the text box across the top. Check the box for GitHub, then select Install
* Create the first admin user. Enter a username, such as admin, then provide your own secure password. Finally, type a full name and e-mail address.
* Select Save and Finish
* Once Jenkins is ready, select Start using Jenkins.