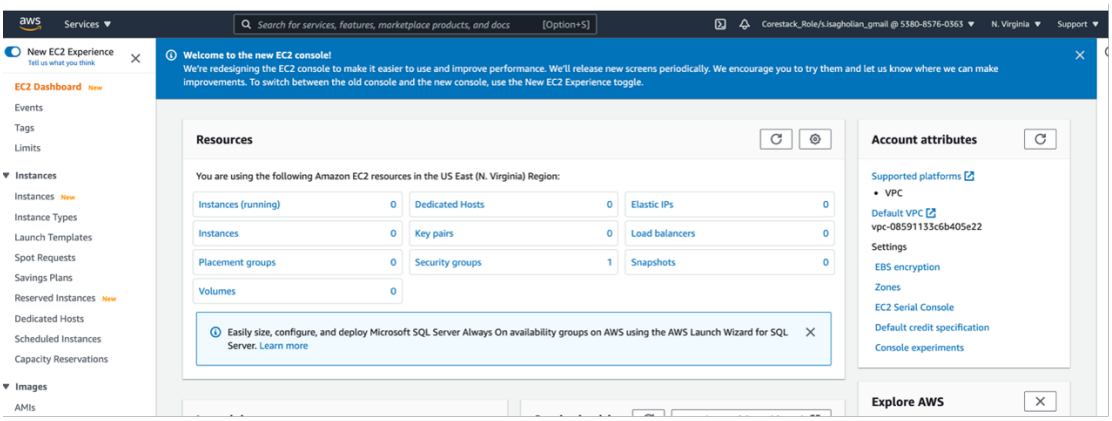
Screenshots Agenda

All steps are described in project readme file.
All source codes are included in project.

Project Github: https://github.com/serjikisagholian/capstone-k8s

1. AWS EC2 dashboard before beginning
2. Provisioning AWS using Ansible
3. Provisioning Kubernetes cluster on AWS
4. Provisioning Vagrant instances, Kubernetes, FlaskApp deployment and custom service account
5. Take snapshot from ETCD
6. Check Custom Service Account permissions
7. Testing HorizontalPodAutoscaler and number of pods with CPU usage increase

Start with a clean dashboard



Provision AWS with ansible

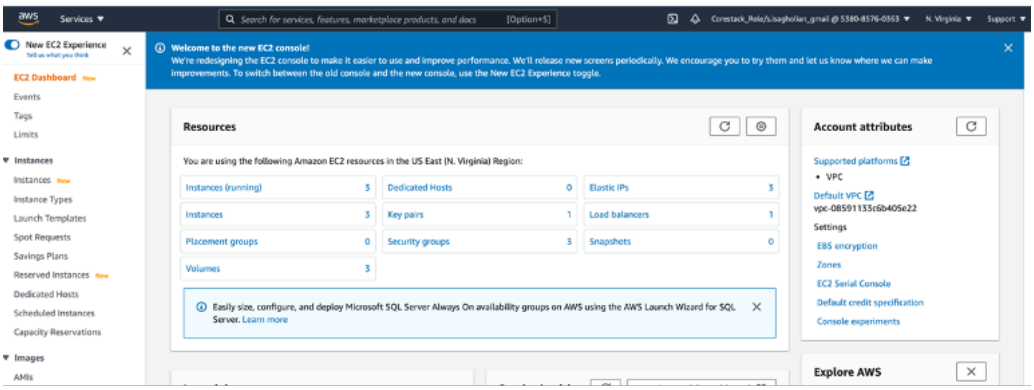
```
>> export AWS_ACCESS_KEY_ID=<AWS_ACCESS_KEY>
>> export AWS_SECRET_ACCESS_KEY=<AWS_SECRET_KEY>
>> export AWS_SECURITY_TOKEN=<AWS_SECURITY_TOKEN>
>> export AWS_REGION=<AWS_REGION>
>> cd aws-version
>> ansible-playbook provision.yml
```

Provision AWS with ansible (output)

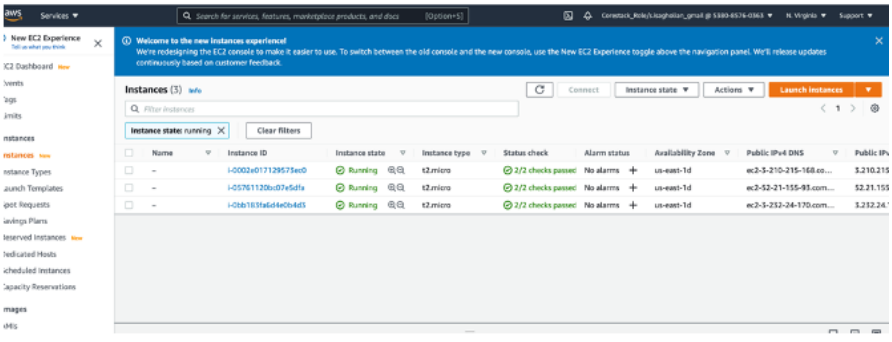
```
[WARNING]: No inventory was parsed, only implicit localhost is available
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [localhost] ***************************************************************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************************************************
ok: [localhost]
TASK [create a new ec2 key pair, returns generated private key] ****************************************************************************************************
changed: [localhost]

TASK [Save private key on local] ***********************************************************************************************************************************
changed: [localhost]

TASK [Create EC2 group] ********************************************************************************************************************************************
[WARNING]: One of your CIDR addresses (10.0.0.0/0) has host bits set. To get rid of this warning, check the network mask and make sure that only network bits are
set: 0.0.0.0/0.
changed: [localhost]

TASK [Create ELB] **************************************************************************************************************************************************
changed: [localhost]

TASK [provision master] ********************************************************************************************************************************************
changed: [localhost]

TASK [associate new elastic IPs with master] ***********************************************************************************************************************
changed: [localhost] => (item=i-0002e017129573ec0)

TASK [output the k8s_master] ***************************************************************************************************************************************
ok: [localhost] => {
    "msg": {
        "changed": true,
        "failed": false,
        "instance_ids": [
            "i-0002e017129573ec0"
        ],
        "instances": [
            {
                "ami_launch_index": "0",
                "architecture": "x86_64",
                "block_device_mapping": {
                    "/dev/sda1": {
                        "delete_on_termination": true,
                        "status": "attached",
                        "volume_id": "vol-0bc34b568930c3255"
                    }
                },
                "dns_name": "ec2-54-210-134-184.compute-1.amazonaws.com",
                "ebs_optimized": false,
                "groups": {
                    "sg-0ac101e3ee1f7ba33": "ec2group01",
                    "sg-0e22a51bed7c7da61": "default_elb_9832fbe7-68b3-3f05-b5bf-59b60f9c9f76"
                },
                "hypervisor": "xen",
                "id": "i-0002e017129573ec0",
                "image_id": "ami-0747bdcabd34c712a",
                "instance_type": "t2.micro",
                "kernel": null,
                "key_name": "my_keypair",
                "launch_time": "2021-05-14T19:09:15.000Z",
                "placement": "us-east-1d",
                "private_dns_name": "ip-172-31-23-208.ec2.internal",
                "private_ip": "172.31.23.208",
                "public_dns_name": "ec2-54-210-134-184.compute-1.amazonaws.com",
                "public_ip": "54.210.134.184",
                "ramdisk": null,
                "region": "us-east-1",
                "root_device_name": "/dev/sda1",
                "root_device_type": "ebs",
                "state": "running",
                "state_code": 16,
                "tags": {},
                "tenancy": "default",
                "virtualization_type": "hvm"
            }
        ],
        "tagged_instances": []
    }
}

TASK [output master_ips] *******************************************************************************************************************************************
ok: [localhost] => {
    "msg": {
        "changed": true,
        "msg": "All items completed",
        "results": [
            {
                "allocation_id": "eipalloc-04f7c0f5006f784c3",
                "ansible_loop_var": "item",
                "changed": true,
                "failed": false,
                "invocation": {
                    "module_args": {
                        "allow_reassociation": false,
                        "aws_access_key": null,
                        "aws_ca_bundle": null,
                        "aws_config": null,
                        "aws_secret_key": null,
                        "debug_botocore_endpoint_logs": false,
                        "device_id": "i-0002e017129573ec0",
                        "ec2_url": null,
                        "in_vpc": true,
                        "private_ip_address": null,
                        "profile": null,
                        "public_ip": null,
                        "public_ipv4_pool": null,
                        "region": null,
                        "release_on_disassociation": false,
                        "reuse_existing_ip_allowed": false,
                        "security_token": null,
                        "state": "present",
                        "tag_name": null,
                        "tag_value": null,
                        "validate_certs": true,
                        "wait_timeout": null
                    }
                },
                "item": "i-0002e017129573ec0",
                "public_ip": "3.210.215.168"
            }
        ]
    }
}

TASK [Create a fresh inventory] ************************************************************************************************************************************
changed: [localhost]

TASK [Add master to local host group] ******************************************************************************************************************************
changed: [localhost] => (item={'changed': True, 'public_ip': '3.210.215.168', 'allocation_id': 'eipalloc-04f7c0f5006f784c3', 'invocation': {'module_args': {'device_id': 'i-0002e017129573ec0',
'in_vpc': True, 'debug_botocore_endpoint_logs': False, 'validate_certs': True, 'state': 'present', 'reuse_existing_ip_allowed': False, 'release_on_disassociation': False, 'allow_reassociation':
False, 'ec2_url': None, 'aws_access_key': None, 'aws_secret_key': None, 'security_token': None, 'aws_ca_bundle': None, 'profile': None, 'aws_config': None, 'region': None, 'public_ip': None,
'wait_timeout': None, 'private_ip_address': None, 'tag_name': None, 'tag_value': None, 'public_ipv4_pool': None}}, 'failed': False, 'item': 'i-0002e017129573ec0', 'ansible_loop_var': 'item'})

TASK [Wait for servers to come online] *****************************************************************************************************************************
ok: [localhost] => (item={'changed': True, 'public_ip': '3.210.215.168', 'allocation_id': 'eipalloc-04f7c0f5006f784c3', 'invocation': {'module_args': {'device_id': 'i-0002e017129573ec0',
'in_vpc': True, 'debug_botocore_endpoint_logs': False, 'validate_certs': True, 'state': 'present', 'reuse_existing_ip_allowed': False, 'release_on_disassociation': False, 'allow_reassociation': False, 'ec2_url': None,
'aws_access_key': None, 'aws_secret_key': None, 'security_token': None, 'aws_ca_bundle': None, 'profile': None, 'aws_config': None, 'region': None, 'public_ip': None, 'wait_timeout': None, 'private_ip_address': None, 'tag_name':
None, 'tag_value': None, 'public_ipv4_pool': None}}, 'failed': False, 'item': 'i-0002e017129573ec0', 'ansible_loop_var': 'item'})
```

```
TASK [Add master as known hosts] ***********************************************************************************************************************************
changed: [localhost] => (item={'changed': True, 'public_ip': '3.210.215.168', 'allocation_id': 'eipalloc-04f7c0f5006f784c3', 'invocation': {'module_args': {'device_id': 'i-0002e017129573ec0',
'in_vpc': True, 'debug_botocore_endpoint_logs': False, 'validate_certs': True, 'state': 'present', 'reuse_existing_ip_allowed': False, 'release_on_disassociation': False, 'allow_reassociation':
False, 'ec2_url': None, 'aws_access_key': None, 'aws_secret_key': None, 'security_token': None, 'aws_ca_bundle': None, 'profile': None, 'aws_config': None, 'region': None, 'public_ip': None,
'wait_timeout': None, 'private_ip_address': None, 'tag_name': None, 'tag_value': None, 'public_ipv4_pool': None}}, 'failed': False, 'item': 'i-0002e017129573ec0', 'ansible_loop_var': 'item'})

TASK [provision nodes] *********************************************************************************************************************************************
changed: [localhost]

TASK [associate new elastic IPs with nodes] ************************************************************************************************************************
changed: [localhost] => (item=i-05761120bc07e5dfa)
changed: [localhost] => (item=i-0bb183fa6d4e0b4d3)

TASK [Add nodes section to inventory] ******************************************************************************************************************************
changed: [localhost]

TASK [Add nodes to local host group] *******************************************************************************************************************************
changed: [localhost] => (item={'changed': True, 'public_ip': '52.21.155.93', 'allocation_id': 'eipalloc-0067f0820a5f5d2d7', 'invocation': {'module_args': {'device_id': 'i-05761120bc07e5dfa',
'in_vpc': True, 'debug_botocore_endpoint_logs': False, 'validate_certs': True, 'state': 'present', 'reuse_existing_ip_allowed': False, 'release_on_disassociation': False, 'allow_reassociation':
False, 'ec2_url': None, 'aws_access_key': None, 'aws_secret_key': None, 'security_token': None, 'aws_ca_bundle': None, 'profile': None, 'aws_config': None, 'region': None, 'public_ip': None,
'wait_timeout': None, 'private_ip_address': None, 'tag_name': None, 'tag_value': None, 'public_ipv4_pool': None}}, 'failed': False, 'item': 'i-05761120bc07e5dfa', 'ansible_loop_var': 'item'})
changed: [localhost] => (item={'changed': True, 'public_ip': '3.232.24.170', 'allocation_id': 'eipalloc-0c29084297192d25f', 'invocation': {'module_args': {'device_id': 'i-0bb183fa6d4e0b4d3',
'in_vpc': True, 'debug_botocore_endpoint_logs': False, 'validate_certs': True, 'state': 'present', 'reuse_existing_ip_allowed': False, 'release_on_disassociation': False, 'allow_reassociation':
False, 'ec2_url': None, 'aws_access_key': None, 'aws_secret_key': None, 'security_token': None, 'aws_ca_bundle': None, 'profile': None, 'aws_config': None, 'region': None, 'public_ip': None,
'wait_timeout': None, 'private_ip_address': None, 'tag_name': None, 'tag_value': None, 'public_ipv4_pool': None}}, 'failed': False, 'item': 'i-0bb183fa6d4e0b4d3', 'ansible_loop_var': 'item'})

TASK [Wait for servers to come online] *****************************************************************************************************************************
ok: [localhost] => (item={'changed': True, 'public_ip': '52.21.155.93', 'allocation_id': 'eipalloc-0067f0820a5f5d2d7', 'invocation': {'module_args': {'device_id': 'i-05761120bc07e5dfa',
'in_vpc': True, 'debug_botocore_endpoint_logs': False, 'validate_certs': True, 'state': 'present', 'reuse_existing_ip_allowed': False, 'release_on_disassociation': False, 'allow_reassociation':
False, 'ec2_url': None, 'aws_access_key': None, 'aws_secret_key': None, 'security_token': None, 'aws_ca_bundle': None, 'profile': None, 'aws_config': None, 'region': None, 'public_ip': None,
'wait_timeout': None, 'private_ip_address': None, 'tag_name': None, 'tag_value': None, 'public_ipv4_pool': None}}, 'failed': False, 'item': 'i-05761120bc07e5dfa', 'ansible_loop_var': 'item'})
ok: [localhost] => (item={'changed': True, 'public_ip': '3.232.24.170', 'allocation_id': 'eipalloc-0c29084297192d25f', 'invocation': {'module_args': {'device_id': 'i-0bb183fa6d4e0b4d3',
'in_vpc': True, 'debug_botocore_endpoint_logs': False, 'validate_certs': True, 'state': 'present', 'reuse_existing_ip_allowed': False, 'release_on_disassociation': False, 'allow_reassociation':
False, 'ec2_url': None, 'aws_access_key': None, 'aws_secret_key': None, 'security_token': None, 'aws_ca_bundle': None, 'profile': None, 'aws_config': None, 'region': None, 'public_ip': None,
'wait_timeout': None, 'private_ip_address': None, 'tag_name': None, 'tag_value': None, 'public_ipv4_pool': None}}, 'failed': False, 'item': 'i-0bb183fa6d4e0b4d3', 'ansible_loop_var': 'item'})

TASK [Add nodes as known hosts] ************************************************************************************************************************************
changed: [localhost] => (item={'changed': True, 'public_ip': '52.21.155.93', 'allocation_id': 'eipalloc-0067f0820a5f5d2d7', 'invocation': {'module_args': {'device_id': 'i-05761120bc07e5dfa',
'in_vpc': True, 'debug_botocore_endpoint_logs': False, 'validate_certs': True, 'state': 'present', 'reuse_existing_ip_allowed': False, 'release_on_disassociation': False, 'allow_reassociation':
False, 'ec2_url': None, 'aws_access_key': None, 'aws_secret_key': None, 'security_token': None, 'aws_ca_bundle': None, 'profile': None, 'aws_config': None, 'region': None, 'public_ip': None,
'wait_timeout': None, 'private_ip_address': None, 'tag_name': None, 'tag_value': None, 'public_ipv4_pool': None}}, 'failed': False, 'item': 'i-05761120bc07e5dfa', 'ansible_loop_var': 'item'})
changed: [localhost] => (item={'changed': True, 'public_ip': '3.232.24.170', 'allocation_id': 'eipalloc-0c29084297192d25f', 'invocation': {'module_args': {'device_id': 'i-0bb183fa6d4e0b4d3',
'in_vpc': True, 'debug_botocore_endpoint_logs': False, 'validate_certs': True, 'state': 'present', 'reuse_existing_ip_allowed': False, 'release_on_disassociation': False, 'allow_reassociation':
False, 'ec2_url': None, 'aws_access_key': None, 'aws_secret_key': None, 'security_token': None, 'aws_ca_bundle': None, 'profile': None, 'aws_config': None, 'region': None, 'public_ip': None,
'wait_timeout': None, 'private_ip_address': None, 'tag_name': None, 'tag_value': None, 'public_ipv4_pool': None}}, 'failed': False, 'item': 'i-0bb183fa6d4e0b4d3', 'ansible_loop_var': 'item'})

PLAY RECAP *********************************************************************************************************************************************************
localhost                  : ok=19   changed=14   unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

(ans01) c-serjik.isagholian@MACC02Z20DDLVDQ aws-version %
```

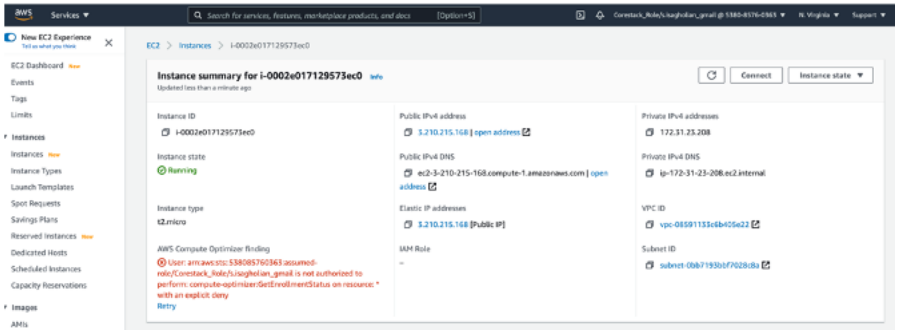
Dashboard after provisioning



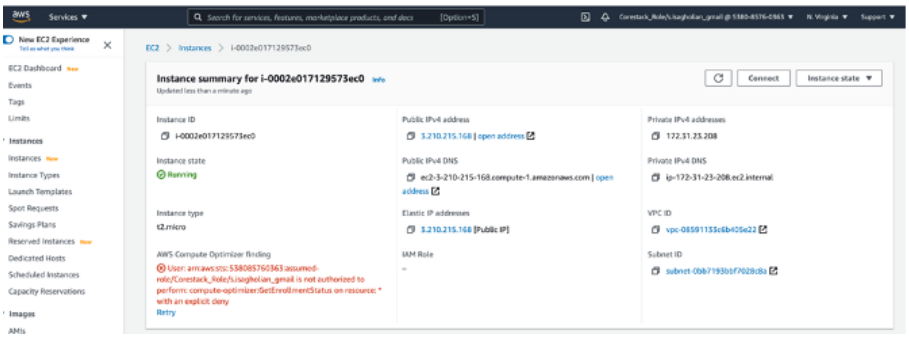
Instances



Master node



EC2 Group Assigned



Ansible hosts file after provisioning

```
[k8s_master]
3.210.215.168 ansible_user=ubuntu ansible_ssh_private_key_file=~/.aws/my_keypair.pem

[k8s_nodes]
52.21.155.93 ansible_user=ubuntu ansible_ssh_private_key_file=~/.aws/my_keypair.pem
3.232.24.170 ansible_user=ubuntu ansible_ssh_private_key_file=~/.aws/my_keypair.pem
```

Testing connectivity

```
>> ansible -m ping k8s_master -i ansible_hosts
>> ansible -m ping k8s_nodes -i ansible_hosts
```

```
(ans01) c-serjik.isagholian@MACC02ZZ20DDLVDQ aws-version % cat ansible_hosts
[k8s_master]
3.210.215.168 ansible_user=ubuntu ansible_ssh_private_key_file=~/.aws/my_keypair.pem

[k8s_nodes]
52.21.155.93 ansible_user=ubuntu ansible_ssh_private_key_file=~/.aws/my_keypair.pem
3.232.24.170 ansible_user=ubuntu ansible_ssh_private_key_file=~/.aws/my_keypair.pem
(ans01) c-serjik.isagholian@MACC02ZZ20DDLVDQ aws-version % ansible k8s_master -m ping -i ansible_hosts
3.210.215.168 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
(ans01) c-serjik.isagholian@MACC02ZZ20DDLVDQ aws-version % ansible k8s_nodes -m ping -i ansible_hosts
3.232.24.170 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
52.21.155.93 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
(ans01) c-serjik.isagholian@MACC02ZZ20DDLVDQ aws-version %
```

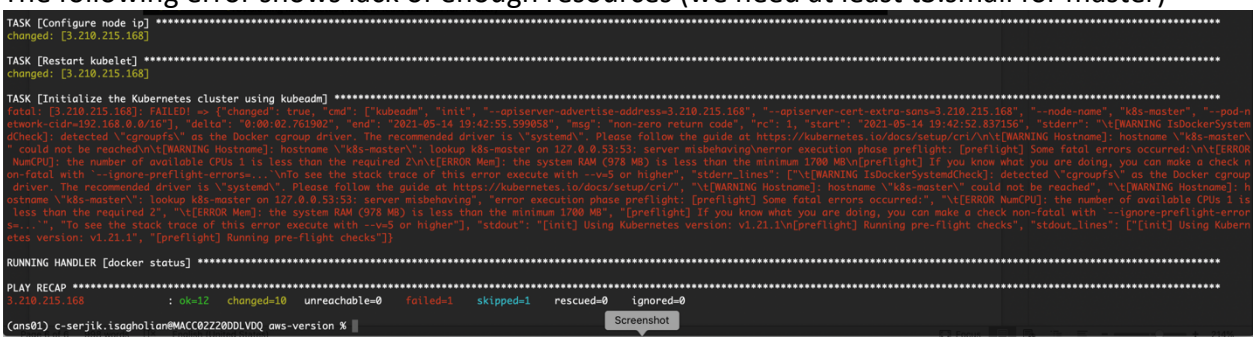
Provisioning Kubernetes (need at least t3.small for master – 2 cores, 1700 Mib)

```
>> ansible-playbook master-playbook.yml -i ansible_hosts
>> ansible-playbook node-playbook.yml -i ansible_hosts
```

```
(ans01) c-serjik.isagholian@MACC02ZZ20DDLVDQ aws-version % ansible-playbook master-playbook.yml -i ansible_hosts

PLAY [k8s_master] ***********************************************************************************************************************************************

TASK [Gathering Facts] ******************************************************************************************************************************************
ok: [3.210.215.168]

TASK [Install packages that allow apt to be used over HTTPS] ****************************************************************************************************
changed: [3.210.215.168]

TASK [Add an apt signing key for Docker] ************************************************************************************************************************
changed: [3.210.215.168]

TASK [Add apt repository for stable version] ********************************************************************************************************************
changed: [3.210.215.168]

TASK [Install docker and its dependecies] ***********************************************************************************************************************
changed: [3.210.215.168]

TASK [Add "ubuntu" user to docker group] ************************************************************************************************************************
changed: [3.210.215.168]

TASK [Remove swapfile from /etc/fstab] **************************************************************************************************************************
ok: [3.210.215.168] => (item=swap)
ok: [3.210.215.168] => (item=none)

TASK [Disable swap] *********************************************************************************************************************************************
skipping: [3.210.215.168]

TASK [Add an apt signing key for Kubernetes] ********************************************************************************************************************
changed: [3.210.215.168]

TASK [Adding apt repository for Kubernetes] *********************************************************************************************************************
changed: [3.210.215.168]

TASK [Install Kubernetes binaries] ******************************************************************************************************************************
changed: [3.210.215.168]

TASK [Configure node ip] ****************************************************************************************************************************************
changed: [3.210.215.168]

TASK [Restart kubelet] ******************************************************************************************************************************************
changed: [3.210.215.168]
```

rovisioning Kubernetes (need at least t2.small for master)

The following error shows lack of enough resources (we need at least t3.small for master)



Ansible output

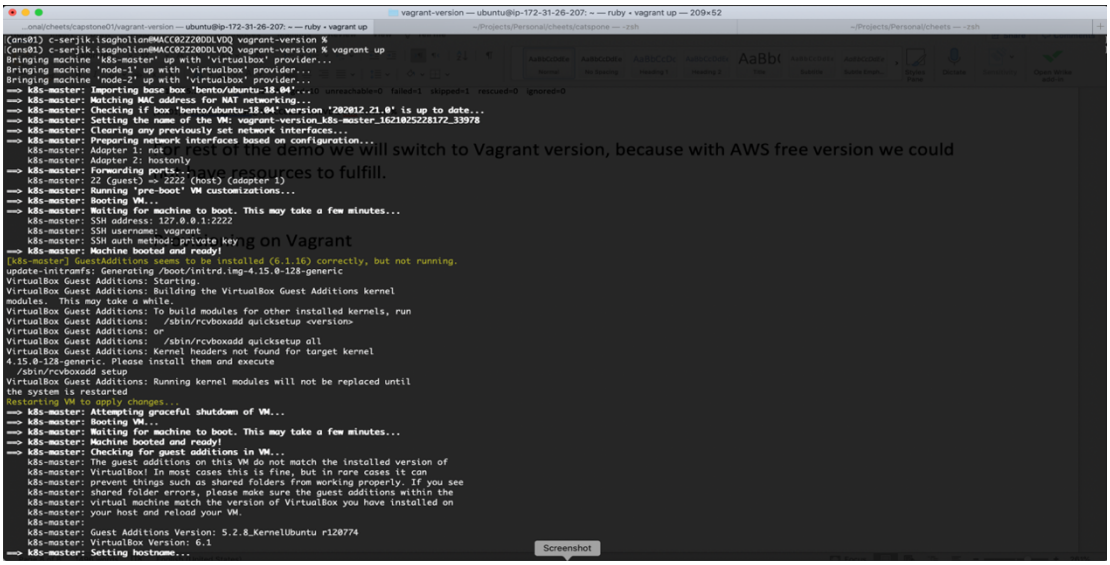
For rest of the demo we will switch to Vagrant version, because with AWS free version we could not have resources to fulfill.

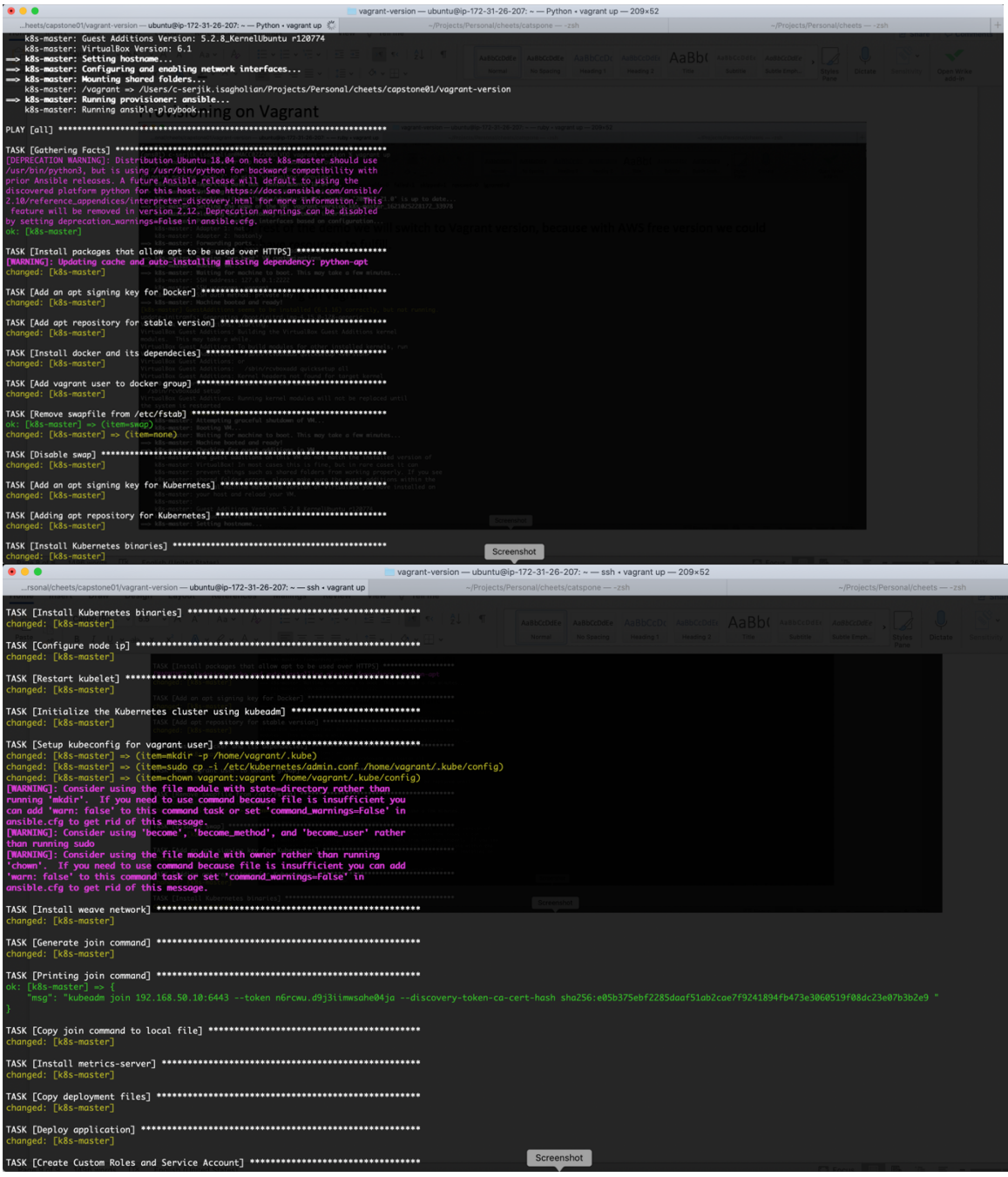
Vagrant file

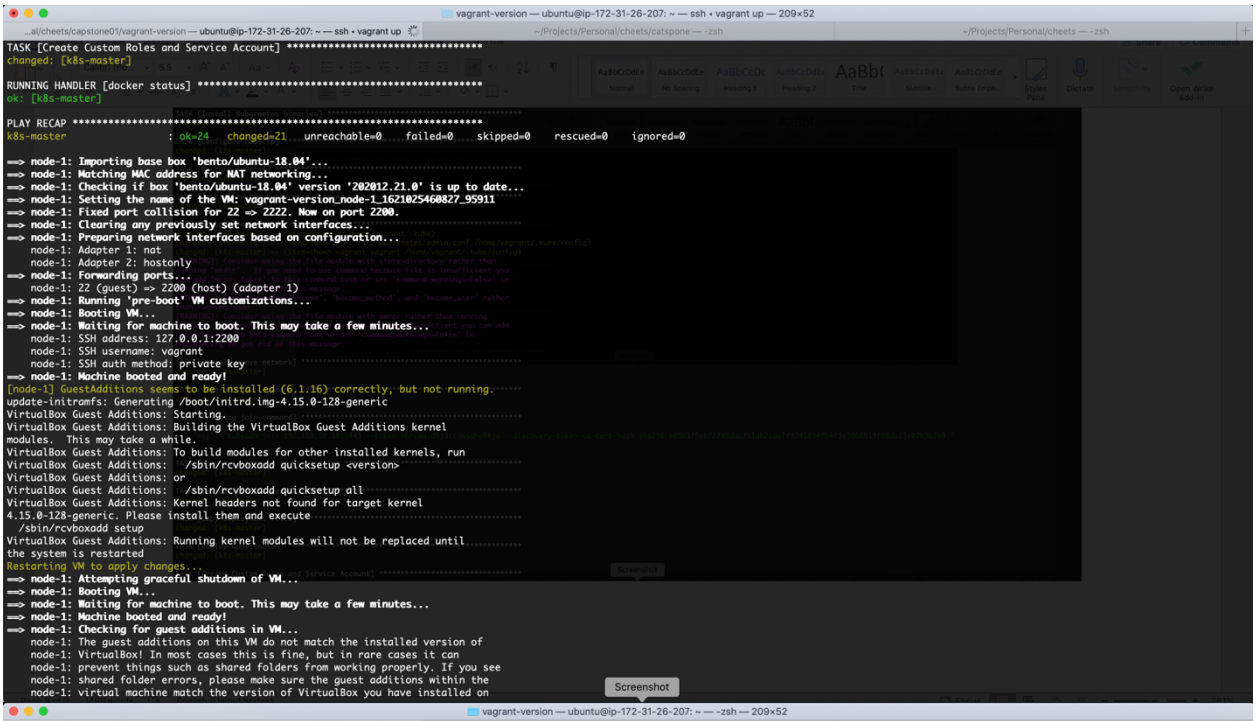
```
IMAGE_NAME = "bento/ubuntu-18.04"
N = 2

Vagrant.configure("2") do |config|
    config.ssh.insert_key = false

    config.vm.provider "virtualbox" do |v|
        v.memory = 6144
        v.cpus = 4
    end

    config.vm.define "k8s-master" do |master|
        master.vm.box = IMAGE_NAME
        master.vm.network "private_network", ip: "192.168.50.10"
        master.vm.hostname = "k8s-master"
        master.vm.provision "ansible" do |ansible|
            ansible.playbook = "kubernetes-setup/master-playbook.yml"
            ansible.extra_vars = {
                node_ip: "192.168.50.10",
            }
        end
    end

    (1..N).each do |i|
        config.vm.define "node-#{i}" do |node|
            node.vm.box = IMAGE_NAME
            node.vm.network "private_network", ip: "192.168.50.#{i + 10}"
            node.vm.hostname = "node-#{i}"
            node.vm.provision "ansible" do |ansible|
                ansible.playbook = "kubernetes-setup/node-playbook.yml"
                ansible.extra_vars = {
                    node_ip: "192.168.50.#{i + 10}",
                }
            end
        end
    end

end
```

Provisioning on Vagrant

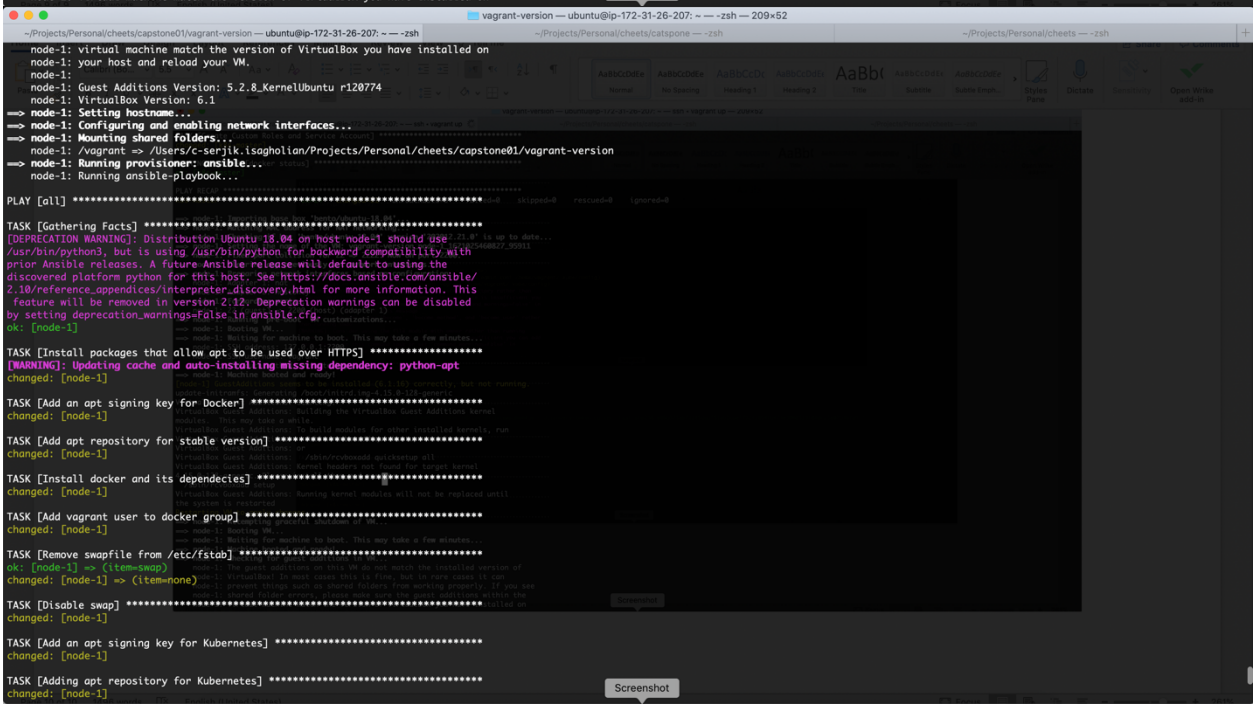
```
>> cd vagrant-version
>> vagrant up
```



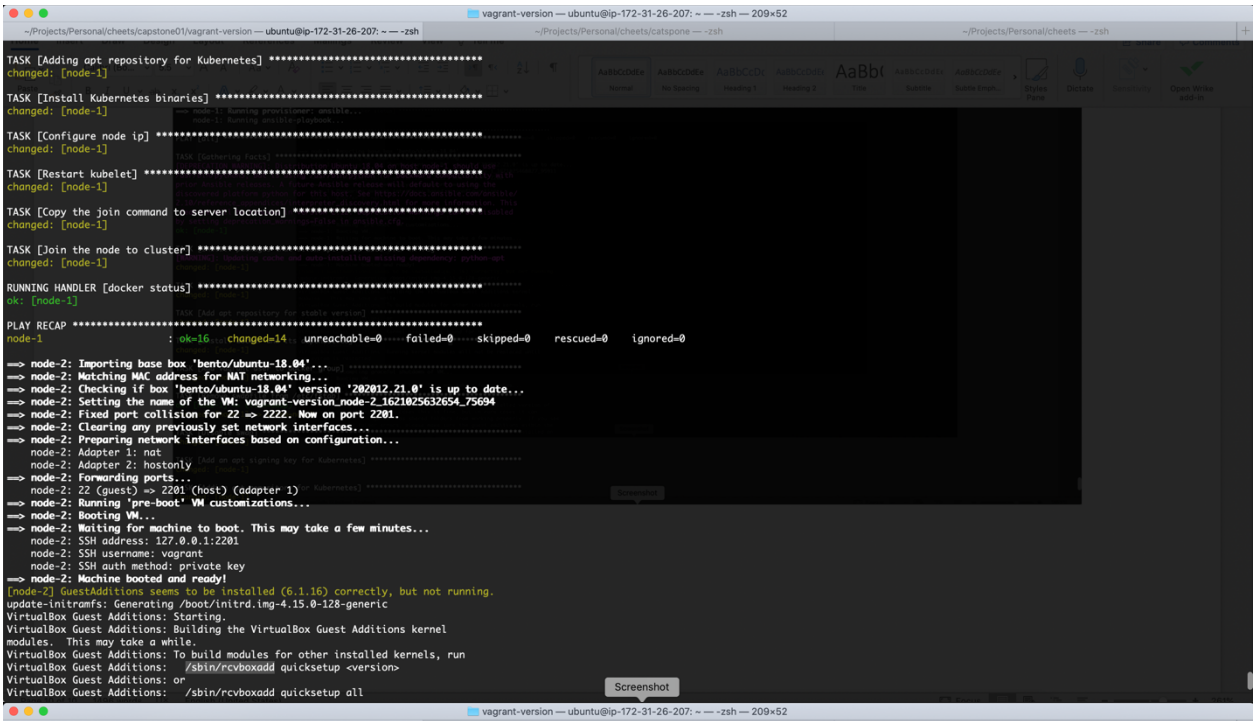

vagrant-version — ubuntu@ip-172-31-26-207: ~ — Python • vagrant up — 209×52
k8s-master: Guest Additions Version: 5.2.8_KernelUbuntu r120774
k8s-master: VirtualBox Version: 6.1
==> k8s-master: Setting hostname...
==> k8s-master: Configuring and enabling network interfaces...
==> k8s-master: Mounting shared folders...
k8s-master: /vagrant => /Users/c-serjik.isagholian/Projects/Personal/cheets/capstone01/vagrant-version
==> k8s-master: Running provisioner: ansible...
k8s-master: Running ansible-playbook...
PLAY [all] ***
TASK [Gathering Facts] ***
[DEPRECATION WARNING]: Distribution Ubuntu 18.04 on host k8s-master should use
/usr/bin/python3, but is using /usr/bin/python for backward compatibility with
prior Ansible releases. A future Ansible release will default to using the
discovered platform python for this host. See https://docs.ansible.com/ansible/
2.10/reference_appendices/interpreter_discovery.html for more information. This
feature will be removed in version 2.12. Deprecation warnings can be disabled
by setting deprecation_warnings=False in ansible.cfg.
ok: [k8s-master]
TASK [Install packages that allow apt to be used over HTTPS] ***
[WARNING]: Updating cache and auto-installing missing dependency: python-apt
changed: [k8s-master]
TASK [Add an apt signing key for Docker] ***
changed: [k8s-master]
TASK [Add apt repository for stable version] ***
changed: [k8s-master]
TASK [Install docker and its dependecies] ***
changed: [k8s-master]
TASK [Add vagrant user to docker group] ***
changed: [k8s-master]
TASK [Remove swapfile from /etc/fstab] ***
ok: [k8s-master] => (item=swap)
changed: [k8s-master] => (item=none)
TASK [Disable swap] ***
changed: [k8s-master]
TASK [Add an apt signing key for Kubernetes] ***
changed: [k8s-master]
TASK [Adding apt repository for Kubernetes] ***
changed: [k8s-master]
TASK [Install Kubernetes binaries] ***
changed: [k8s-master]
Screenshot
vagrant-version — ubuntu@ip-172-31-26-207: ~ — ssh • vagrant up — 209×52
TASK [Install Kubernetes binaries] ***
changed: [k8s-master]
TASK [Configure node ip] ***
changed: [k8s-master]
TASK [Restart kubelet] ***
changed: [k8s-master]
TASK [Initialize the Kubernetes cluster using kubeadm] ***
changed: [k8s-master]
TASK [Setup kubeconfig for vagrant user] ***
changed: [k8s-master] => (item=mkdir -p /home/vagrant/.kube)
changed: [k8s-master] => (item=sudo cp -i /etc/kubernetes/admin.conf /home/vagrant/.kube/config)
changed: [k8s-master] => (item=chown vagrant:vagrant /home/vagrant/.kube/config)
[WARNING]: Consider using the file module with state=directory rather than
running 'mkdir'. If you need to use command because file is insufficient you
can add 'warn: false' to this command task or set 'command_warnings=False' in
ansible.cfg to get rid of this message.
[WARNING]: Consider using 'become', 'become_method', and 'become_user' rather
than running sudo
[WARNING]: Consider using the file module with owner rather than running
'chown'. If you need to use command because file is insufficient you can add
'warn: false' to this command task or set 'command_warnings=False' in
ansible.cfg to get rid of this message.
TASK [Install weave network] ***
changed: [k8s-master]
TASK [Generate join command] ***
changed: [k8s-master]
TASK [Printing join command] ***
ok: [k8s-master] => {
"msg": "kubeadm join 192.168.50.10:6443 --token n6rcwu.d9j3iimwsahe04ja --discovery-token-ca-cert-hash sha256:e05b375ebf2285daaf51ab2cae7f9241894fb473e3060519f08dc23e07b3b2e9 "
}
TASK [Copy join command to local file] ***
changed: [k8s-master]
TASK [Install metrics-server] ***
changed: [k8s-master]
TASK [Copy deployment files] ***
changed: [k8s-master]
TASK [Deploy application] ***
changed: [k8s-master]
TASK [Create Custom Roles and Service Account] ***
Screenshot

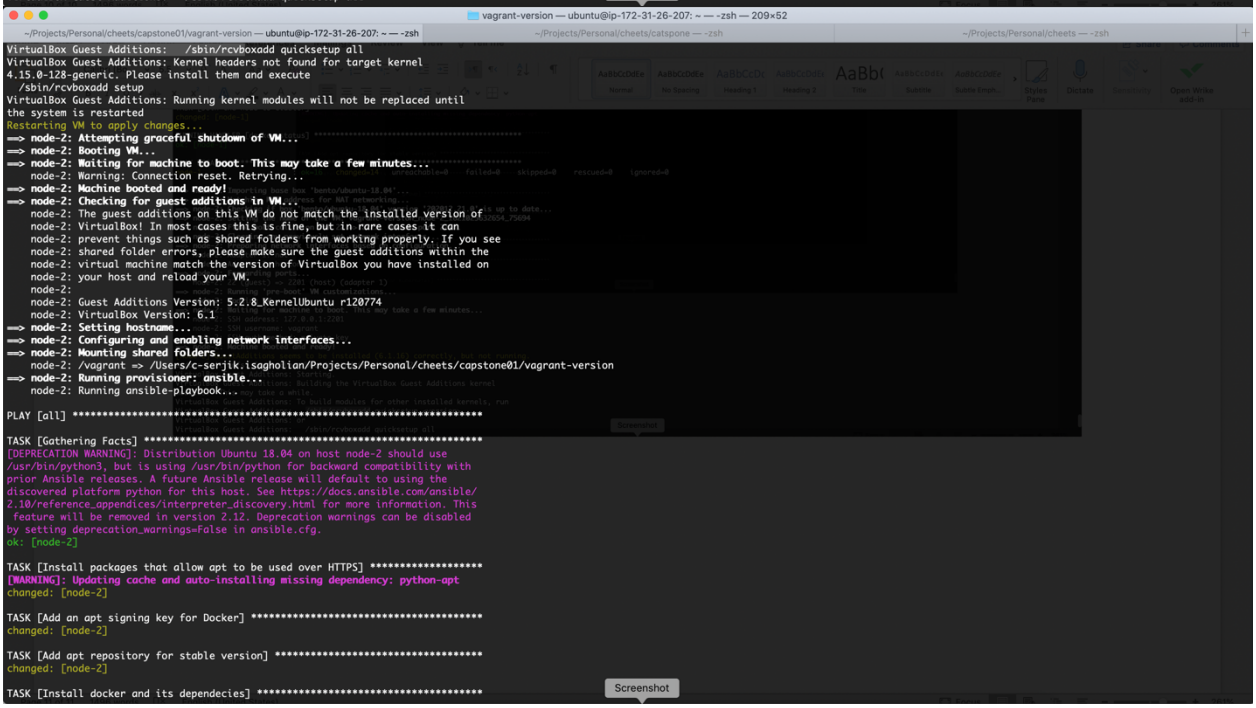


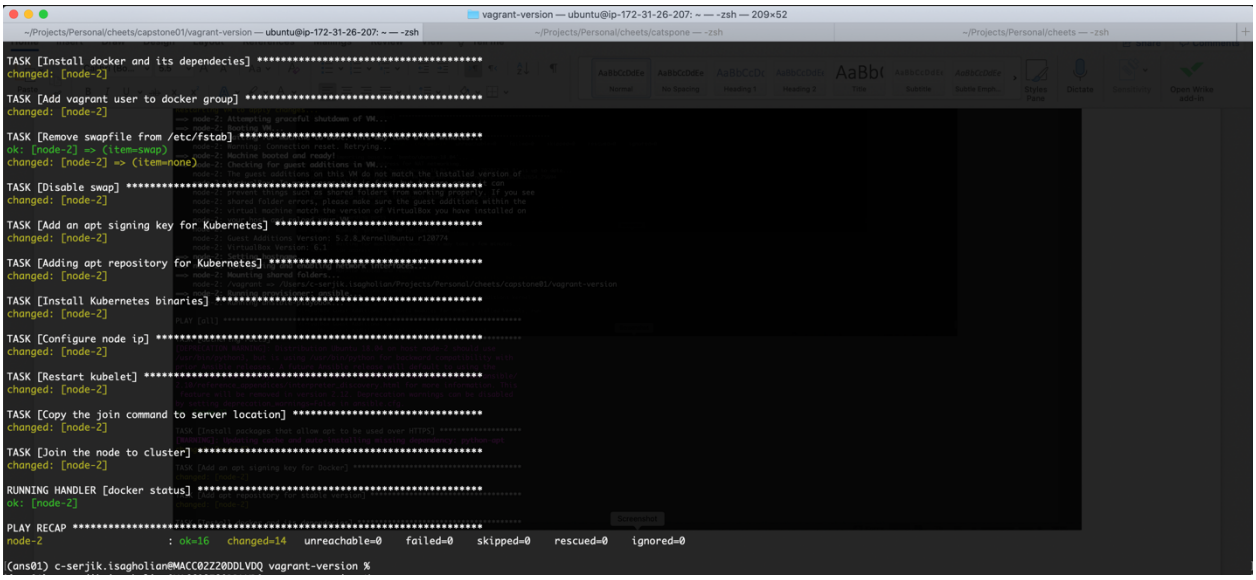
Screenshot



Screenshot

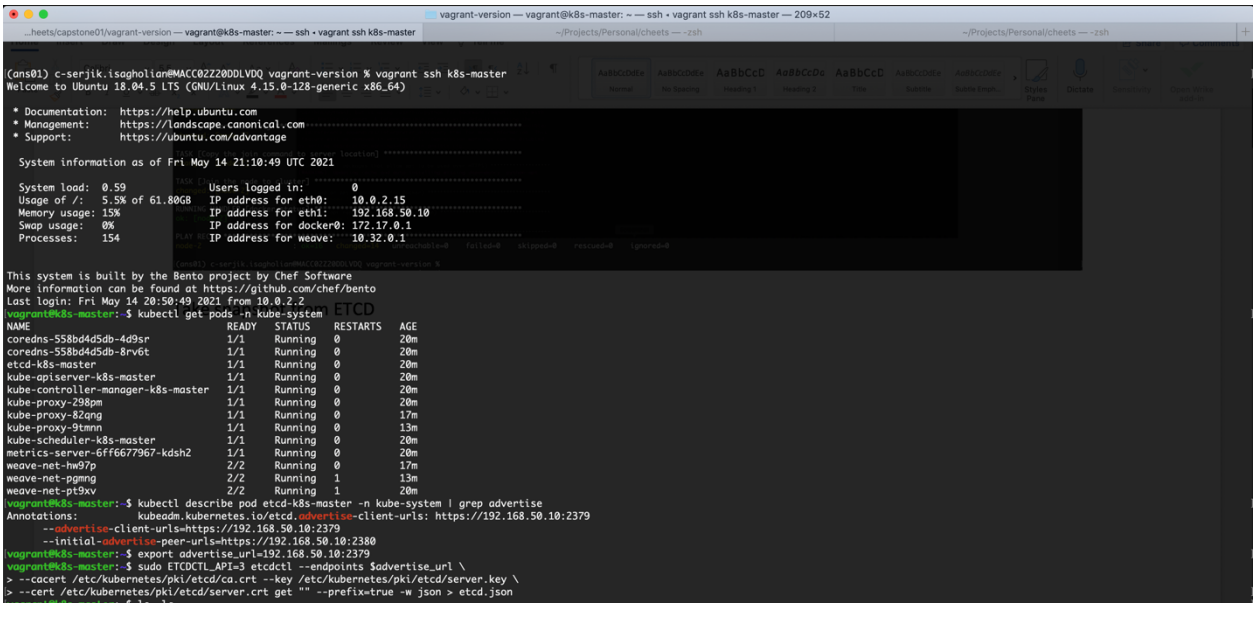

vagrant-version — ubuntu@ip-172-31-26-207: ~ — -zsh — 209×52
~/Projects/Personal/cheets/capstone01/vagrant-version — ubuntu@ip-172-31-26-207: ~ — -zsh
~/Projects/Personal/cheets/catspone — -zsh
~/Projects/Personal/cheets — -zsh
TASK [Adding apt repository for Kubernetes] ************************************
changed: [node-1]
TASK [Install Kubernetes binaries] ***
changed: [node-1]
TASK [Configure node ip] ***
changed: [node-1]
TASK [Restart kubelet] ***
changed: [node-1]
TASK [Copy the join command to server location] ********************************
changed: [node-1]
TASK [Join the node to cluster] **
changed: [node-1]
RUNNING HANDLER [docker status] **
ok: [node-1]
PLAY RECAP ***
node-1 : ok=16 changed=14 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
==> node-2: Importing base box 'bento/ubuntu-18.04'...
==> node-2: Matching MAC address for NAT networking...
==> node-2: Checking if box 'bento/ubuntu-18.04' version '202012.21.0' is up to date...
==> node-2: Setting the name of the VM: vagrant-version_node-2_1621025632654_75694
==> node-2: Fixed port collision for 22 => 2222. Now on port 2201.
==> node-2: Clearing any previously set network interfaces...
==> node-2: Preparing network interfaces based on configuration...
node-2: Adapter 1: nat
node-2: Adapter 2: hostonly
==> node-2: Forwarding ports...
node-2: 22 (guest) => 2201 (host) (adapter 1)
==> node-2: Running 'pre-boot' VM customizations...
==> node-2: Booting VM...
==> node-2: Waiting for machine to boot. This may take a few minutes...
node-2: SSH address: 127.0.0.1:2201
node-2: SSH username: vagrant
node-2: SSH auth method: private key
==> node-2: Machine booted and ready!
[node-2] GuestAdditions seems to be installed (6.1.16) correctly, but not running.
update-initramfs: Generating /boot/initrd.img-4.15.0-128-generic
VirtualBox Guest Additions: Starting.
VirtualBox Guest Additions: Building the VirtualBox Guest Additions kernel
modules. This may take a while.
VirtualBox Guest Additions: To build modules for other installed kernels, run
VirtualBox Guest Additions: /sbin/rcvboxadd quicksetup <version>
VirtualBox Guest Additions: or
VirtualBox Guest Additions: /sbin/rcvboxadd quicksetup all
Screenshot

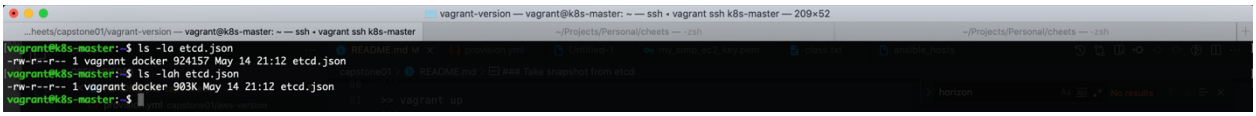

vagrant-version — ubuntu@ip-172-31-26-207: ~ — -zsh — 209×52
~/Projects/Personal/cheets/capstone01/vagrant-version — ubuntu@ip-172-31-26-207: ~ — -zsh
~/Projects/Personal/cheets/catspone — -zsh
~/Projects/Personal/cheets — -zsh
VirtualBox Guest Additions: /sbin/rcvboxadd quicksetup all
VirtualBox Guest Additions: Kernel headers not found for target kernel
4.15.0-128-generic. Please install them and execute
/sbin/rcvboxadd setup
VirtualBox Guest Additions: Running kernel modules will not be replaced until
the system is restarted
Restarting VM to apply changes...
==> node-2: Attempting graceful shutdown of VM...
==> node-2: Booting VM...
==> node-2: Waiting for machine to boot. This may take a few minutes...
node-2: Warning: Connection reset. Retrying...
==> node-2: Machine booted and ready!
==> node-2: Checking for guest additions in VM...
node-2: The guest additions on this VM do not match the installed version of
node-2: VirtualBox! In most cases this is fine, but in rare cases it can
node-2: prevent things such as shared folders from working properly. If you see
node-2: shared folder errors, please make sure the guest additions within the
node-2: virtual machine match the version of VirtualBox you have installed on
node-2: your host and reload your VM.
node-2:
node-2: Guest Additions Version: 5.2.8_KernelUbuntu r120774
node-2: VirtualBox Version: 6.1
==> node-2: Setting hostname...
==> node-2: Configuring and enabling network interfaces...
==> node-2: Mounting shared folders...
node-2: /vagrant => /Users/c-serjik.isagholian/Projects/Personal/cheets/capstone01/vagrant-version
==> node-2: Running provisioner: ansible...
node-2: Running ansible-playbook...
PLAY [all] ***
TASK [Gathering Facts] ***
[DEPRECATION WARNING]: Distribution Ubuntu 18.04 on host node-2 should use
/usr/bin/python3, but is using /usr/bin/python for backward compatibility with
prior Ansible releases. A future Ansible release will default to using the
discovered platform python for this host. See https://docs.ansible.com/ansible/
2.10/reference_appendices/interpreter_discovery.html for more information. This
feature will be removed in version 2.12. Deprecation warnings can be disabled
by setting deprecation_warnings=False in ansible.cfg.
ok: [node-2]
TASK [Install packages that allow apt to be used over HTTPS] *******************
[WARNING]: Updating cache and auto-installing missing dependency: python-apt
changed: [node-2]
TASK [Add an apt signing key for Docker] ***************************************
changed: [node-2]
TASK [Add apt repository for stable version] ***********************************
changed: [node-2]
TASK [Install docker and its dependecies] **************************************
Screenshot



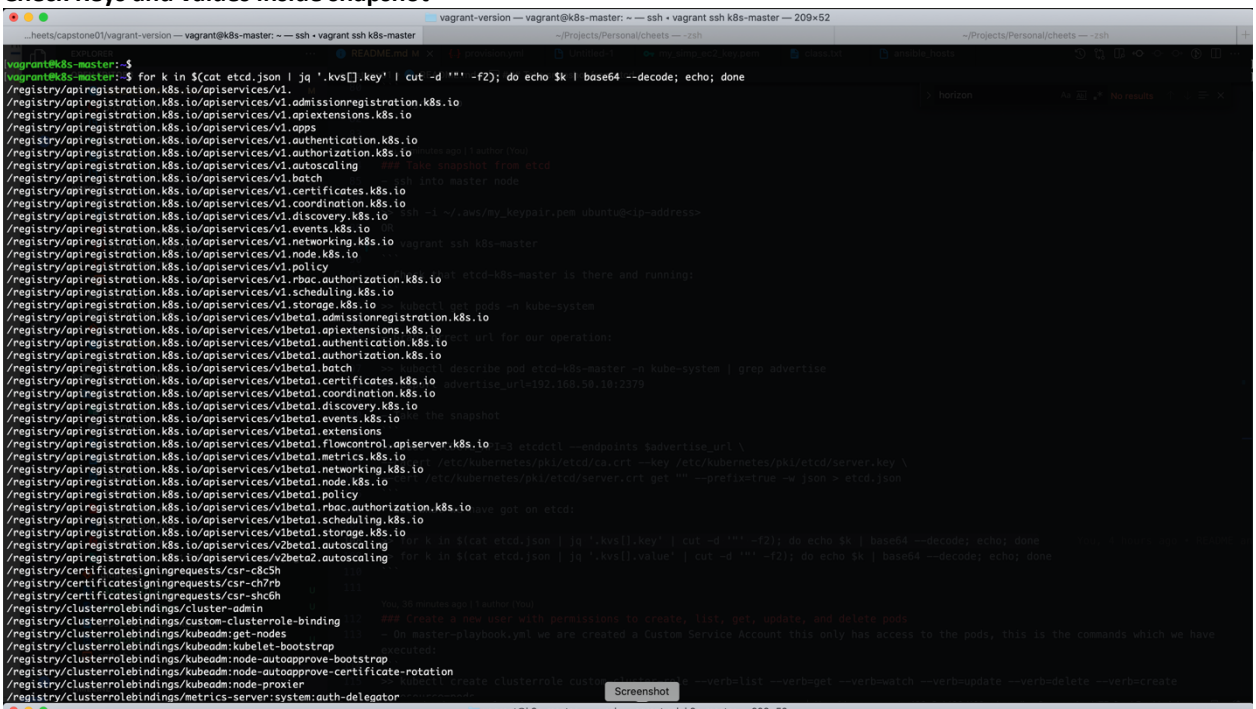
Take snapshot from ETCD

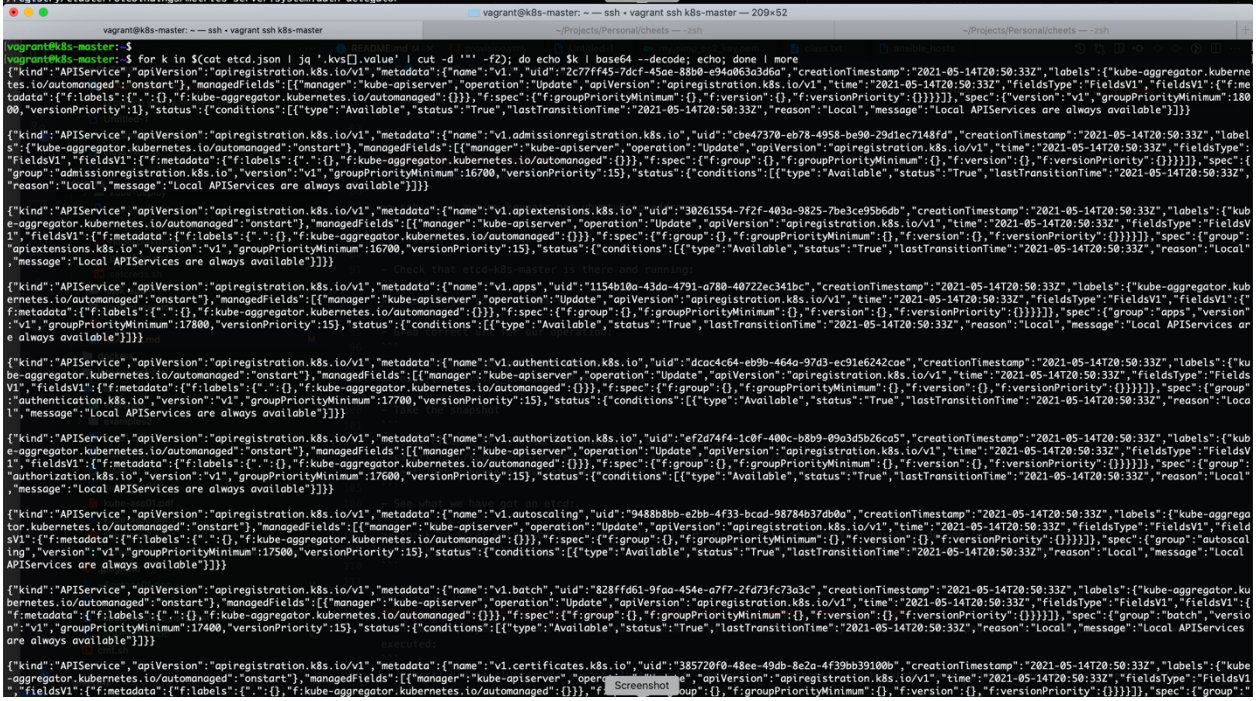
```
>> vagrant ssh k8s-master    # ssh to master node
>> kubectl get pods -n kube-system   # Check that etcd-k8s-master is there and running
# Grab correct url for our operation
>> kubectl describe pod etcd-k8s-master -n kube-system | grep advertise
>> export advertise_url=192.168.50.10:2379
# Take the snapshot
>> sudo ETCDCTL_API=3 etcdctl --endpoints $advertise_url \
--cacert /etc/kubernetes/pki/etcd/ca.crt --key /etc/kubernetes/pki/etcd/server.key \
--cert /etc/kubernetes/pki/etcd/server.crt get "" --prefix=true -w json > etcd.json
# See what we have got on etcd:
>> for k in $(cat etcd.json | jq '.kvs[].key' | cut -d '"' -f2); do echo $k | base64 --
decode; echo; done
>> for k in $(cat etcd.json | jq '.kvs[].value' | cut -d '"' -f2); do echo $k | base64 --
decode; echo; done
```





Check Keys and values inside snapshot

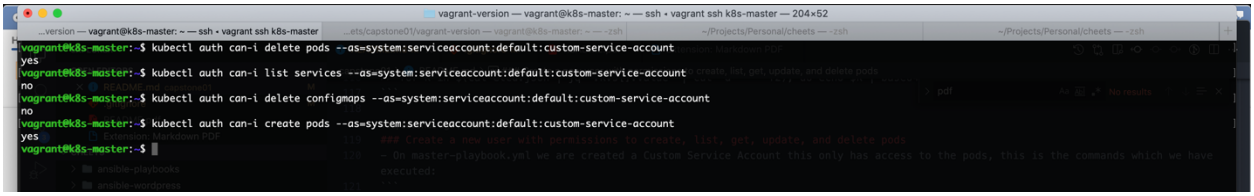




Check Custom Service Account permissions

We have created our custom-cluster-role and custom-service-account as part of master-playbook.yml.

```
  - name: Create Custom Roles and Service Account
    become: false
    shell: |
      kubectl create clusterrole custom-cluster-role --verb=list --verb=get --verb=watch --verb=update --verb=delete --verb=create  --
resource=pods
      kubectl create serviceaccount custom-service-account
      kubectl create clusterrolebinding custom-clusterrole-binding --clusterrole=custom-cluster-role --serviceaccount=default:custom-service-
account
```



Load Generate and Auto Scaling

We have created a 2 deployments for flaskapp and mysql, plus services for each, NetworkPolicy for mysql deployment and HorizontalPodAutoScaler for flaskapp. The all run as part of our master-playbook.yml.

```
  - name: Copy deployment files
    become: false
    copy:
      src: ./kube-deploy
      dest: ./

  - name: Deploy application
    become: false
    shell: |
     cd $(pwd)/kube-deploy
     kubectl apply -k ./
```

The kube-deploy contains yaml files for Kubernetes.

Kustomization.yaml

```
secretGenerator:
- name: mysql-pass
  literals:
  - password=centos
resources:
  - mysql-deployment.yaml
  - flaskapp-deployment.yaml
```

mysql-deployment.yaml

```
apiVersion: apps/v1 # for k8s versions before 1.9.0 use apps/v1beta2  and before 1.8.0 use extensions/v1beta1
kind: Deployment
metadata:
  name: flask-mysql
  labels:
    app: fk-mysql
spec:
  selector:
    matchLabels:
      app: fk-mysql
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        app: fk-mysql
    spec:
      containers:
      - image: mysql:5.6
        name: mysql
        env:
        - name: MYSQL_ROOT_PASSWORD
          valueFrom:
            secretKeyRef:
              name: mysql-pass
              key: password
        livenessProbe:
          tcpSocket:
            port: 3306
        ports:
        - containerPort: 3306
          name: mysql

---

apiVersion: v1
kind: Service
metadata:
```

```
  name: flask-mysql
  labels:
    app: fk-mysql
spec:
  ports:
    - port: 3306
  selector:
    app: fk-mysql
  clusterIP: None

---

kind: NetworkPolicy
apiVersion: networking.k8s.io/v1
metadata:
  name: flask-mysql
spec:
  podSelector:
    matchLabels:
      app: fk-mysql
  ingress:
  - from:
      - podSelector:
          matchLabels:
            app: fk-frontend
```

flaskapp-deployment.yaml

```
apiVersion: apps/v1 #  for k8s versions before 1.9.0 use apps/v1beta2  and before 1.8.0 use extensions/v1beta1
kind: Deployment
metadata:
  name: flaskapp
  labels:
    app: fk-frontend
spec:
  replicas: 1
  selector:
    matchLabels:
      app: fk-frontend
  strategy:
    type: Recreate
  template:
    metadata:
      labels:
        app: fk-frontend
    spec:
      containers:
      - image: serjik1024/flaskapp:1.0
        name: flaskapp
        env:
        - name: DB_HOST
          value: flask-mysql
        - name: DB_PASSWORD
          valueFrom:
            secretKeyRef:
              name: mysql-pass
              key: password
        ports:
        - containerPort: 5000
          name: flaskapp
        resources:
          limits:
            cpu: 500m
          requests:
            cpu: 200m
status: {}

---

apiVersion: v1
kind: Service
metadata:
  name: flaskapp
  labels:
    app:  fk-frontend
spec:
  ports:
  - port: 5000
    protocol: TCP
    targetPort: 5000
  selector:
    app: fk-frontend
  type: NodePort
status:
  loadBalancer: {}

---

apiVersion: autoscaling/v1
kind: HorizontalPodAutoscaler
metadata:
  creationTimestamp: null
  name: flaskapp
spec:
  maxReplicas: 3
  minReplicas: 1
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: flaskapp
```

```
  targetCPUUtilizationPercentage: 50
status:
  currentReplicas: 0
  desiredReplicas: 0
```

this is the output of our application with load generator: