

# Assisted Practice Project4-

## Problem 1

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with links like EC2 Dashboard, EC2 Global View, Events, Tags, Limits, Instances (selected), and Images. The main area displays a table of instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 D...
JenkinsInstance	i-04360505948a652bb	Terminated	t2.micro	-	No alarms	us-east-1a	-
kubernetesinst...	i-03c27fd08eeae5287	Running	t2.micro	2/2 checks passed	No alarms	us-east-1a	ec2-54-242-4...

Below the table, the details for the running instance are shown:

**Instance: i-03c27fd08eeae5287 (kubernetesinstance)**

**Details** | Security | Networking | Storage | Status checks | Monitoring | Tags

**Instance summary**

Instance ID i-03c27fd08eeae5287 (kubernetesinstance)	Public IPv4 address 54.242.48.221   open address	Private IPv4 addresses 172.31.24.97
IPv6 address -	Instance state Running	Public IPv4 DNS ec2-54-242-48-221.compute-1.amazonaws.com   open address
Hostname type IP name: ip-172-31-24-97.ec2.internal	Private IP DNS name (IPv4 only) ip-172-31-24-97.ec2.internal	

Feedback: Looking for language selection? Find it in the new Unified Settings.

The screenshot shows a terminal window titled "ubuntu@ip-172-31-24-97:~". The terminal output shows the installation of Kubernetes components:

```
Preparing to unpack .../4-socat_1.7.4.1-Subuntu4_amd64.deb ...
Unpacking socat (1.7.4.1-Subuntu4) ...
Preparing previously unselected package kubelet.
Preparing to unpack .../5-kubelet_1.22.8-00_amd64.deb ...
Unpacking kubelet (1.22.8-00) ...
Setting up kubelet (1.22.8-00) ...
Preparing to unpack .../6-kubectl_1.22.8-00_amd64.deb ...
Unpacking kubectl (1.22.8-00) ...
Selecting previously unselected package kubeadm.
Preparing to unpack .../7-kubeadm_1.22.8-00_amd64.deb ...
Unpacking kubeadm (1.22.8-00) ...
Setting up conctrack (1:1.4.6-2build2) ...
Setting up kubelet (1.22.8-00) ...
Setting up ebtables (2.0.11-4build2) ...
Setting up iproute (4.19.100-1Subuntu4) ...
Setting up cri-tools (1.25.0-00) ...
Setting up kubernetes-cni (1.1.1-00) ...
Setting up kubelet (1.22.8-00) ...
Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service → /lib/systemd/system/kubelet.service.
Setting up kubeadm (1.22.8-00) ...
Processing triggers for man-db (2.10.2-1) ...
Scanning processes ...
Scanning Linux images ...

Running kernel seems to be up-to-date.
No services need to be restarted.
No containers need to be restarted.
No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-24-97:~$ sudo apt-mark hold kubelet kubeadm kubectl
Kubelet set on hold.
kubeadm set on hold.
kubectl set on hold.
ubuntu@ip-172-31-24-97:~$ kubectl version
Client Version: version.Info{Major:"1", Minor:"22", GitVersion:"v1.22.8", GitCommit:"7061dbbf75f0f82e8ab21f9be7e8ffcaae8e0d44", GitTreeState:"clean", BuildDate:"2022-03-16T14:10:06Z", GoVersion:"go1.16.15", Compiler:"gc", Platform:"linux/amd64"}
The connection to the server localhost:8080 was refused - did you specify the right host or port?
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

## Problem 2

**New EC2 Experience** Tell us what you think

EC2 Dashboard  
EC2 Global View  
Events  
Tags  
Limits

Instances Instances New

Instance Types  
Launch Templates  
Spot Requests  
Savings Plans  
Reserved Instances Reserved Instances  
Dedicated Hosts  
Scheduled Instances  
Capacity Reservations

Images IP name: ip-172-31-24-97.ec2.internal

Feedback Looking for language selection? Find it in the new Unified Settings

Instances (1/2) Info Connect Instance state Actions Launch instances

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 D
JenkinsInstance	i-04360505948a652bb	Terminated	t2.micro	-	No alarms	us-east-1a	-
kubernetesinst...	i-03c27fd08eeae5287	Running	t2.micro	2/2 checks passed	No alarms	us-east-1a	ec2-54-242-48-221.compute-1.amazonaws.com   open address

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

ubuntu@ip-172-31-85-68: ~

```
ubuntu@ip-172-31-85-68: ~$ curl -H "Content-Type: application/json" -X POST https://cloud.weave.works/k8s/net?k8s-version=$kubever --data-binary @- > /tmp/weave-ingress.yaml
ubuntu@ip-172-31-85-68: ~$ kubectl apply -f https://cloud.weave.works/k8s/net?k8s-version=$kubever
Error from server: failed to find 'https://cloud.weave.works/k8s/net?k8s-version=$kubever': refused to connect
ubuntu@ip-172-31-85-68: ~$ kubectl get pods -n kube-system
NAME          READY   STATUS    RESTARTS   AGE
coredns-78fcf09978-rx5pc   0/1     Pending   0          9m15s
coredns-7cd-ip-172-31-85-68 1/1     Running   0          9m19s
kube-apiserver-ip-172-31-85-68 1/1     Running   0          9m22s
kube-controller-manager-ip-172-31-85-68 1/1     Running   0          9m19s
kube-dns-ip-172-31-85-68 1/1     Running   0          9m15s
kube-scheduler-ip-172-31-85-68 1/1     Running   0          9m19s
ubuntu@ip-172-31-85-68: ~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://download.docker.com/linux/ubuntu jammy-backports InRelease
Hit:4 https://security.cloudflare.com/ubuntu jammy-security InRelease
Get:5 http://security.google.com/api kubernetes-xenial InRelease [110 kB]
Get:6 http://archive.ubuntu.com/ubuntu jammy InRelease [9303 B]
Fetched 120 kB in 197 kB/s (Reading package lists... Done)
ubuntu@ip-172-31-85-68: ~$ sudo apt-get install maven -y
sudo: apt-get install : command not found
ubuntu@ip-172-31-85-68: ~$
```

ubuntu@ip-172-31-26-218: ~

```
ubuntu@ip-172-31-26-218: ~$ curl -H "Content-Type: application/json" -X POST https://cloud.weave.works/k8s/net?k8s-version=$kubever --data-binary @- > /tmp/weave-ingress.yaml
ubuntu@ip-172-31-26-218: ~$ kubectl apply -f https://cloud.weave.works/k8s/net?k8s-version=$kubever
Error from server: failed to find 'https://cloud.weave.works/k8s/net?k8s-version=$kubever': refused to connect
ubuntu@ip-172-31-26-218: ~$ kubectl get pods -n kube-system
NAME          READY   STATUS    RESTARTS   AGE
coredns-78fcf09978-rx5pc   0/1     Pending   0          9m15s
coredns-7cd-ip-172-31-85-68 1/1     Running   0          9m19s
kube-apiserver-ip-172-31-85-68 1/1     Running   0          9m22s
kube-controller-manager-ip-172-31-85-68 1/1     Running   0          9m19s
kube-dns-ip-172-31-85-68 1/1     Running   0          9m15s
kube-scheduler-ip-172-31-85-68 1/1     Running   0          9m19s
ubuntu@ip-172-31-26-218: ~$ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://download.docker.com/linux/ubuntu jammy-backports InRelease
Hit:4 https://security.cloudflare.com/ubuntu jammy-security InRelease
Get:5 http://security.google.com/api kubernetes-xenial InRelease [9303 B]
Fetched 120 kB in 197 kB/s (Reading package lists... Done)
ubuntu@ip-172-31-85-68: ~$ sudo apt-get install maven -y
sudo: apt-get install : command not found
ubuntu@ip-172-31-85-68: ~$
```

[control-plane] Creating static Pod manifest for "kube-scheduler"
[etcd] Creating static Pod manifest for local etcd in "/etc/kubernetes/manifests"
[wait-control-plane] Waiting for the kubelet to boot up the control plane as static Pods from directory "/etc/kubernetes/manifests". This can take up to 4m0s
[apiservice] All control plane components are healthy after 15.505138 seconds
[upload-certs] Storing certificates in "/var/lib/kubelet/pki" using ConfigMap "kubeadm-config" in the "kube-system" Namespace
[kubelet] Creating ConfigMap "kubelet-config-1.22" in namespace kube-system with the configuration for the kubelets in the cluster
[mark-control-plane] Skipping phase. Please see --upload-certs
[mark-control-plane] Marking the node ip-172-31-26-218 as control-plane by adding the labels: [node-role.kubernetes.io/master(deprecated) node-role.kubernetes.io/control-plane node.kubernetes.io/exclude-from-external-load-balancers]
[mark-control-plane] Marking the node ip-172-31-26-218 as control-plane by adding the taints [node-role.kubernetes.io/master:NoSchedule]
[bootstrap-token] Using token: z253nl.hnfqz4sgw9b4hf
[bootstrap-token] Configuring bootstrap tokens, cluster-info ConfigMap, RBAC Roles
[bootstrap-token] configured RBAC rules to allow Node Bootstrap tokens to get nodes
[bootstrap-token] configured RBAC rules to allow Node Bootstrap tokens to post CSRs in order for nodes to get long term certificate credentials
[bootstrap-token] configured RBAC rules to allow certificates rotation for all node client certificates issued by a Node Bootstrap Token
[bootstrap-token] Creating the "cluster-info" ConfigMap in the "kube-public" namespace
[kubelet-finalize] Updating "/etc/kubernetes/kubelet.conf" to point to a rotatable kubelet client certificate and key
[addons] Applied essential addon: CoreDNS
[addons] Applied essential addon: kube-proxy

Your Kubernetes control-plane has initialized successfully!

To start using your cluster, you need to run the following as a regular user:

mkdir -p \$HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf \$HOME/.kube/config
sudo chown \$(id -u):\$(id -g) \$HOME/.kube/config

Alternatively, if you are the root user, you can run:

export KUBECONFIG=/etc/kubernetes/admin.conf

You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
https://kubernetes.io/docs/concepts/cluster-administration/addons/

Then you can join any number of worker nodes by running the following on each as root:
kubeadm join 172.31.26.218:6443 --token z253nl.hnfqz4sgw9b4hf \
--discovery-token-ca-cert-hash sha256:88b1712f9b0b598b62121aae5ec0ad889d3536e0c6370ddd7cd224f4791d7b
ubuntu@ip-172-31-26-218: ~\$ export kubever=\$(kubectl version | base64 | tr -d '\n')

## Problem 3

The screenshot shows the AWS Management Console EC2 Instances page and a MobaXterm terminal window.

**AWS EC2 Instances Page:**

- Instances (1/2) Info
- Instances (1/2) table:
  - JenkinsInstance (terminated)
  - kubernetesinstance (Running, t2.micro, 2/2 checks passed)
- Actions dropdown
- Launch instances button

**MobaXterm Terminal Session:**

- Session: 5 ubuntu@ip-172-31-85-68: ~\$ SpringBootDockerApp\$
- Output log:

```
[INFO] No sources to compile
[INFO] ... maven-surefire-plugin:2.22.2:test (default-test) @ demo-docker ...
[INFO] ... maven-jar-plugin:3.2.0:jar (default-jar) @ demo-docker ...
[INFO] Building jar: /home/ubuntu/SpringBootDockerApp/target/demo-docker-0.0.1-SNAPSHOT.jar
[INFO] ... spring-boot-maven-plugin:2.5.6:repackage (repackage) @ demo-docker ...
[INFO] Replacing main artifact with repackaged archive
[INFO] ...
[INFO] ... maven-install-plugin:3.5.2:install (default-install) @ demo-docker ...
[INFO] Installing /home/ubuntu/SpringBootDockerApp/target/demo-docker-0.0.1-SNAPSHOT.jar to /home/ubuntu/.m2/repository/com/example/demo-docker/0.0.1-SNAPSHOT/demo-docker-0.0.1-SNAPSHOT.jar
[INFO] ...
[INFO] BUILD SUCCESS
[INFO] Total time: 11.367 s
[INFO] Finished at: 2022-10-07T06:05:02Z
[INFO] ...
ubuntu@ip-172-31-85-68:~/SpringBootDockerApp$ sudo docker build -t springboot .
sending build context to Docker daemon 17.62MB
Step 1/5 : FROM java
manifest for java:latest not found: manifest unknown
ubuntu@ip-172-31-85-68:~/SpringBootDockerApp$ sudo docker images
REPOSITORY          TAG      IMAGE ID      CREATED     SIZE
k8s.gcr.io/kube-apiserver   v1.22.15   bd16c7ea581a  2 weeks ago  128MB
k8s.gcr.io/kube-controller-manager   v1.22.15   c9f099a9d222  2 weeks ago  128MB
k8s.gcr.io/kube-scheduler   v1.22.15   b40e0521513a  2 weeks ago  52.7MB
k8s.gcr.io/kube-proxy   v1.22.15   8fe9e16d5d5d  2 weeks ago  104MB
k8s.gcr.io/etcd      3.5.0-0   004811815584  5 months ago  295MB
k8s.gcr.io/coredns/credentials   v1.8.4    8d147337d794  7 months ago  47.9MB
k8s.gcr.io/coredns   v1.8.4    d24203e445b  18 months ago  45.8MB
ubuntu@ip-172-31-85-68:~/SpringBootDockerApp$ kubectl run springbootapp --image=springboot --port=8080
pod/springbootapp created
ubuntu@ip-172-31-85-68:~/SpringBootDockerApp$
```

## Problem 4

The screenshot shows the AWS Management Console EC2 Instances page and a MobaXterm terminal window.

**AWS EC2 Instances Page:**

- Instances (1/2) Info
- Instances (1/2) table:
  - JenkinsInstance (terminated)
  - kubernetesinstance (Running, t2.micro, 2/2 checks passed)
- Actions dropdown
- Launch instances button

**MobaXterm Terminal Session:**

- Session: 5 ubuntu@ip-172-31-85-68: ~\$ SpringBootDockerApp\$
- Output log:

```
[INFO] No sources to compile
[INFO] ... maven-surefire-plugin:2.22.2:test (default-test) @ demo-docker ...
[INFO] ... maven-jar-plugin:3.2.0:jar (default-jar) @ demo-docker ...
[INFO] Building jar: /home/ubuntu/SpringBootDockerApp/target/demo-docker-0.0.1-SNAPSHOT.jar
[INFO] ... spring-boot-maven-plugin:2.5.6:repackage (repackage) @ demo-docker ...
[INFO] Replacing main artifact with repackaged archive
[INFO] ...
[INFO] ... maven-install-plugin:3.5.2:install (default-install) @ demo-docker ...
[INFO] Installing /home/ubuntu/SpringBootDockerApp/target/demo-docker-0.0.1-SNAPSHOT.jar to /home/ubuntu/.m2/repository/com/example/demo-docker/0.0.1-SNAPSHOT/demo-docker-0.0.1-SNAPSHOT.jar
[INFO] ...
[INFO] BUILD SUCCESS
[INFO] Total time: 11.367 s
[INFO] Finished at: 2022-10-07T06:05:02Z
[INFO] ...
ubuntu@ip-172-31-85-68:~/SpringBootDockerApp$ sudo docker build -t springboot .
sending build context to Docker daemon 17.62MB
Step 1/5 : FROM java
manifest for java:latest not found: manifest unknown
ubuntu@ip-172-31-85-68:~/SpringBootDockerApp$ sudo docker images
REPOSITORY          TAG      IMAGE ID      CREATED     SIZE
k8s.gcr.io/kube-apiserver   v1.22.15   bd16c7ea581a  2 weeks ago  128MB
k8s.gcr.io/kube-controller-manager   v1.22.15   c9f099a9d222  2 weeks ago  128MB
k8s.gcr.io/kube-scheduler   v1.22.15   b40e0521513a  2 weeks ago  52.7MB
k8s.gcr.io/kube-proxy   v1.22.15   8fe9e16d5d5d  2 weeks ago  104MB
k8s.gcr.io/etcd      3.5.0-0   004811815584  5 months ago  295MB
k8s.gcr.io/coredns/credentials   v1.8.4    8d147337d794  7 months ago  47.9MB
k8s.gcr.io/coredns   v1.8.4    d24203e445b  18 months ago  45.8MB
ubuntu@ip-172-31-85-68:~/SpringBootDockerApp$ kubectl run springbootapp --image=springboot --port=8080
pod/springbootapp created
ubuntu@ip-172-31-85-68:~/SpringBootDockerApp$
```

```
[INFO] Installing /home/ubuntu/SpringBootDockerApp/pom.xml to /home/ubuntu/.m2/repository/com/example/demo-docker/0.0.1-SNAPS
HOT/demo-docker-0.0.1-SNAPSHOT.pom
[INFO] BUILD SUCCESS
[INFO]   Total time: 11.367 s
[INFO]   Finished at: 2022-10-07T06:05:39Z
[INFO]
[INFO] [17:31:45] $ sudo docker build -t springboot .
[INFO] Sending build context to Docker daemon 17.62MB
[INFO] Step 1/5 : FROM openjdk:8-jdk-alpine
[INFO] manifest for java@latest not found: manifest unknown: manifest unknown
[INFO] Step 2/5 : RUN curl -fSL https://repo1.maven.org/maven2/com/example/demo-docker/0.0.1-SNAPSHOT/demo-docker-0.0.1-SNAPSHOT.jar > /app/demo-docker.jar
[INFO] Step 3/5 : EXPOSE 8082
[INFO] Step 4/5 : CMD ["java", "-jar", "/app/demo-docker.jar"]
[INFO] Step 5/5 : ENTRYPOINT ["java", "-jar", "/app/demo-docker.jar"]
[INFO] Successfully built 3e3a5d81a584 (17.62MB)
[INFO] Successfully tagged springboot:latest
[INFO] [17:31:45] $ sudo docker images
REPOSITORY          TAG      IMAGE ID      CREATED     SIZE
k8s.gcr.io/kube-apiserver   v1.22.15   ed2103ea1a581a  2 weeks ago  128MB
k8s.gcr.io/kube-controller-manager   v1.22.15   c0f0999a4f22  2 weeks ago  12MB
k8s.gcr.io/kube-scheduler   v1.22.15   b409e97b215  2 weeks ago  52.7MB
k8s.gcr.io/kube-proxy   v1.22.15   8f111918584  2 weeks ago  29MB
k8s.gcr.io/etcd   v3.5.0-0-   ed2103ea1a584  2 weeks ago  29MB
k8s.gcr.io/coredns/coredns   v1.8.4    8d147537fb7d  16 months ago  47.6MB
k8s.gcr.io/pause   3.5      ed2103ea1a581a  16 months ago  683kB
[INFO] [17:31:45] $ ./start.sh
[INFO] [17:31:45] $ kubectl get pods
NAME        READY   STATUS    RESTARTS   AGE
springbootapp   0/1     Pending   0          10s
[INFO] [17:31:45] $ kubectl expose pod/seringbootapp --port=8082 --target-port=8082 --type=LoadBalanc
er
[INFO] [17:31:45] $ kubectl get pods
[INFO] [17:31:45] $ kubectl get services
[INFO] [17:31:45] $ kubectl get svc
[INFO] [17:31:45] $ kubectl get service
NAME        TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
springbootapp   LoadBalancer   10.96.0.1    <none>       443/TCP   47m
[INFO] [17:31:45] $ kubectl get svc
[INFO] [17:31:45] $ kubectl get service
NAME        TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
springbootapp   LoadBalancer   10.103.197.20  <pending>   8082:30950/TCP   86s
```

```
Session Servers Tools Games Sessions View Split Multi-Terminal Packages Settings Help
X server Exit
Quick connect...
Name Size (B)
home/ubuntu/ + Name
  - cache
  - .ssh
  - bash_logout
  - bashrc
  - profile
  - xauthORITY
  1
k8s.gcr.io/etcd 3.5-0-0 004911015584 15 months ago 295MB
k8s.gcr.io/coredns v1.8.4 8d14537fb7d 16 months ago 47.6MB
k8s.gcr.io/pause 3.5 ed210e3ed4a5b 18 months ago 683kB
ubuntu@ip-172-31-85-68:~/SpringBootDockerApp$ kubectl get pods
  NAME          READY   STATUS    RESTARTS   AGE
springbootapp  0/1     Pending   0          15s
ubuntu@ip-172-31-85-68:~/SpringBootDockerApp$ kubectl expose pod/springbootapp --port=8082 --target-port=8082 --type=LoadBalancer
Error from server (NotFound): pods "springbootapp" not found
ubuntu@ip-172-31-85-68:~/SpringBootDockerApp$ kubectl expose pod/springbootapp --port=8082 --target-port=8082 --type=LoadBalancer
services/springbootapp exposed
ubuntu@ip-172-31-85-68:~/SpringBootDockerApp$ kubectl get service
  NAME           TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)          AGE
kubernetes     ClusterIP   10.101.197.28   <none>        443/TCP         47m
springbootapp  LoadBalancer 10.101.197.28   <pending>    8082:30950/TCP  86s
ubuntu@ip-172-31-85-68:~/SpringBootDockerApp$ kubectl describe svc springbootapp
Name:           springbootapp
Namespace:      default
Labels:          run=springbootapp
Annotations:    <none>
Selector:       run=springbootapp
Type:           LoadBalancer
IP Family Policy: Single Stack
IP Families:   IPv4
IP:             10.101.197.28
Ports:
  Port:          <unset>  8082/TCP
  TargetPort:    8082/TCP
  NodePort:     <unset>  30950/TCP
  Endpoints:    <none>
  Session Affinity: None
  External Traffic Policy: Cluster
Events:         <none>
ubuntu@ip-172-31-85-68:~/SpringBootDockerApp$
```