National College of Ireland

Reducing instance acquisition lag to improve scaling out in the kubernetes cluster

MSc Research Project
Cloud Computing

Bharath Raj Kanthimathinathan
Student ID: 20225024

School of Computing
National College of Ireland

Supervisor: Shivani Jaswal

National College of Ireland
Project Submission Sheet
School of Computing

National College of Ireland

Student Name:	Bharath Raj Kanthimathinathan
Student ID:	20225024
Programme:	Cloud Computing
Year:	2022
Module:	MSc Research Project
Supervisor:	Shivani Jaswal
Submission Due Date:	15/08/2022
Project Title:	Reducing instance acquisition lag to improve scaling out in the kubernetes cluster
Word Count:	858
Page Count:	7

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Bharath raj
Date:	18th September 2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Reducing instance acquisition lag to improve scaling out in the kubernetes cluster

Bharath Raj Kanthimathinathan
20225024

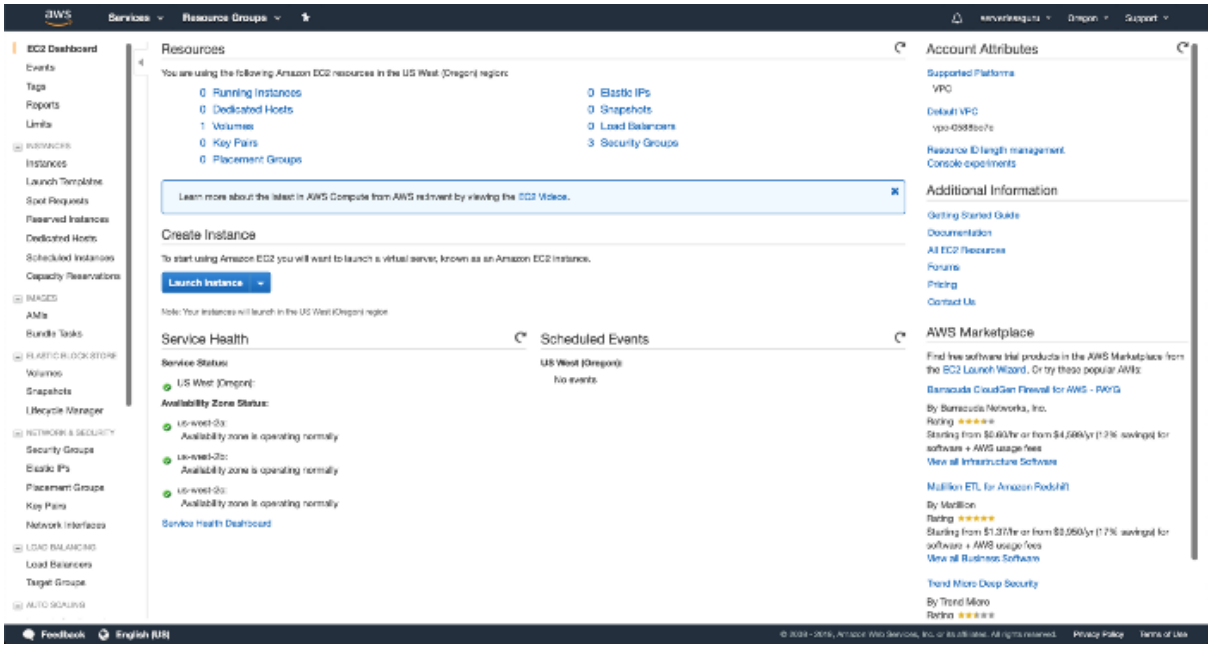
1 Introduction

The document outlines the steps that are required to be carried out to setup and perform the research. It includes the steps required to install the kubernetes cluster setup for testing the ANA autoscaler. Followed by it the steps are covered to install the ANA autoscaler and finally the steps that are required to setup the EKS cluster.

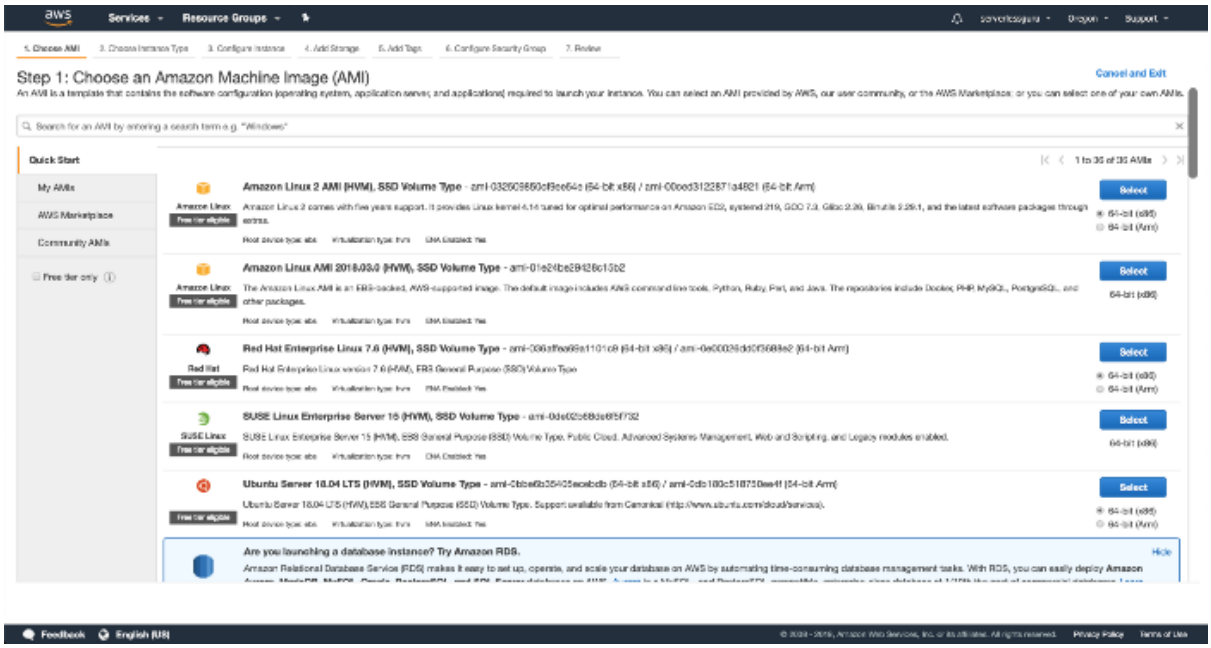
2 Kubernetes cluster setup

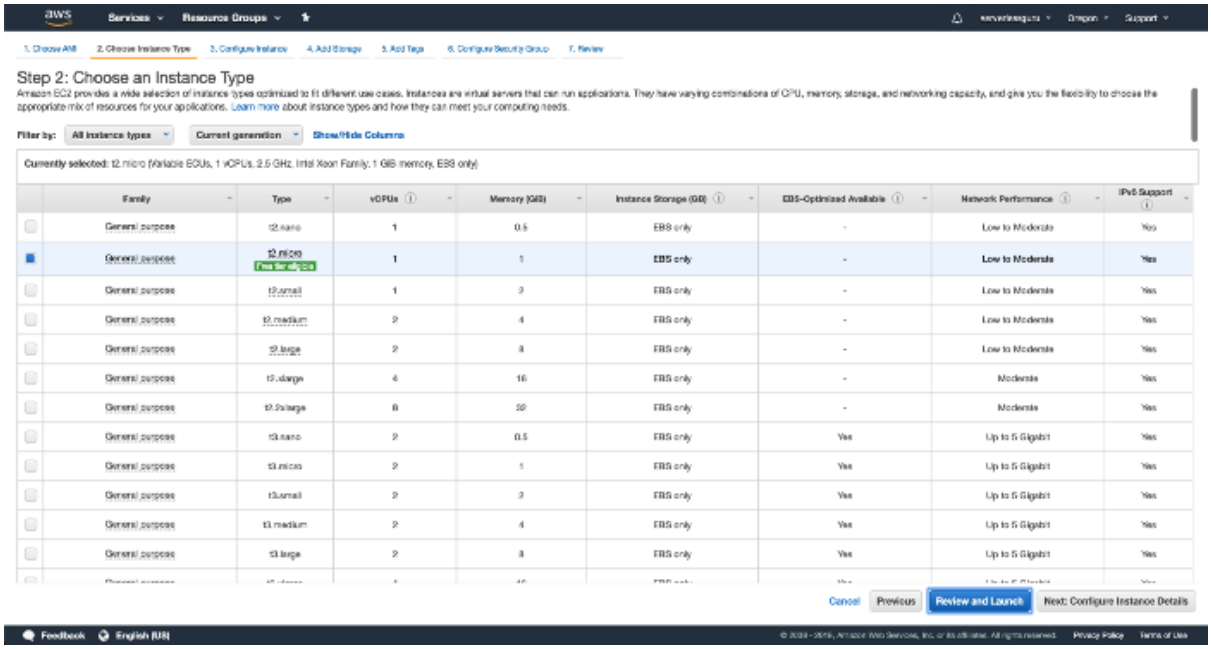
The kubernetes cluster is installed in the EC2 instances created in the AWS cloud using the kubeadm tool.

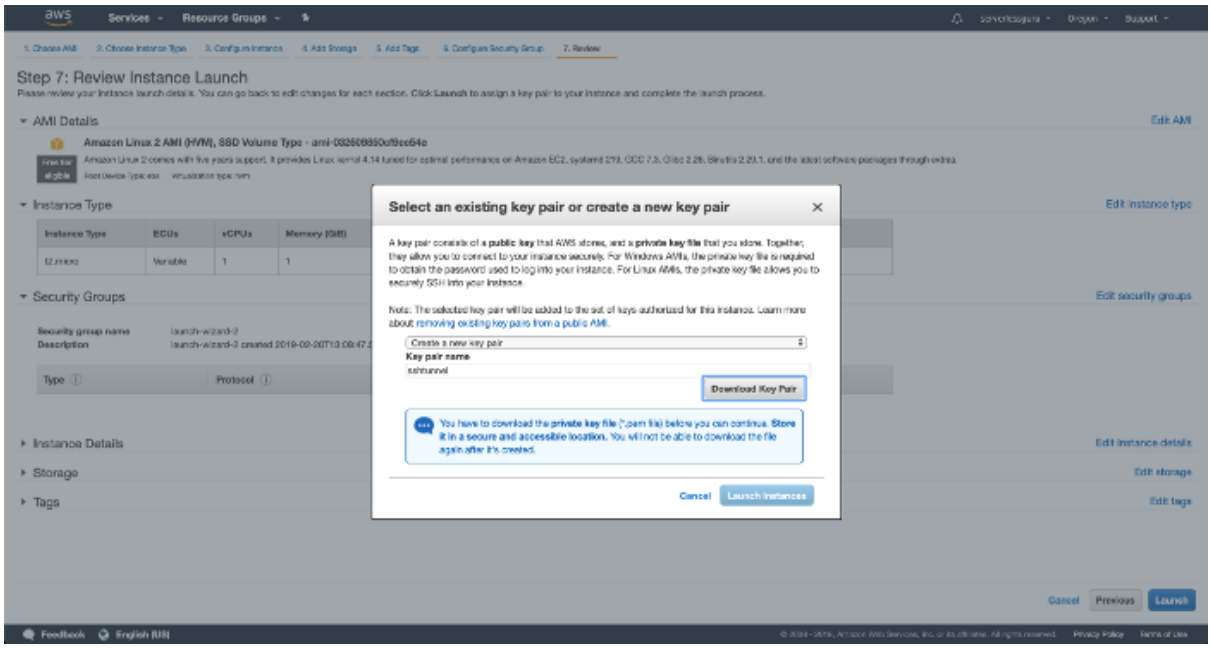
- AWS account is created first, then after logging into the console EC2 service is selected.
- Three instances are required to be created, one for the master node and two for the worker node.
- Launch instance is selected.
- Ubuntu 20.04 image is selected for the EC2 instance.
- t2.micro instance type is selected for the EC2 instance.
- A key pair is created which is used to login into the master node and install the ANA autoscaler scripts.
- The steps are followed for the creation of the three EC2 instances.
- It needs to be ensured that all the EC2 instances are present in the same VPC and security group.
- Use **apt update** and **apt-get install** to install the required packages as shown in **figure 1**.
- Add the docker repository and install the docker runtime as shown in **figure 2**.
- Change network settings to run kuberentes as shown in **figure 3**.



- Install kubernetes by adding the repository and installing the packages as seen in **figure 4** Kub (2022).
- Repeat the above steps from apt update on all the EC2 instances and ensure kubernetes is installed on all of them. Following this step the commands are only ran on the master node.
- Initialize the master node by logging into one of the server using the private key that was used to create the EC2 instance and run the command **kubeadm init –pod-network-cidr=10.244.0.0/16**.
- Copy the kubeconfig to enable and run kubectl commands easily as shown in **figure 5**.
- Install the flannel pod network for enabling the communication between the pods using kube-flannel.yaml file present as part of the solution artifact using command **kubectl apply -f kube-flannel.yaml**.
- Install krew plugin which helps in installing kubectl plugin and enhancing the functionality of the kubectl command line.
- Install resource-capacity plugin using the krew plugin.
- Install the Jquery package using the **apt-get install jq**.
- Install the metrics server which is part of the solution artifact using command **kubectl apply -f metrics-server.yaml**.


Services
Resource Groups
Step 1: Choose an Amazon Machine Image (AMI)
Cancel and Exit
Quick Start
My AMIs
AWS Marketplace
Community AMIs
Free tier only
Amazon Linux 2 AMI (HVM), SSD Volume Type
Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras.
Select
64-bit (x86)
64-bit (Arm)
Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type
Red Hat Enterprise Linux 7.6 (HVM), SSD Volume Type
SUSE Linux Enterprise Server 15 (HVM), SSD Volume Type
Ubuntu Server 18.04 LTS (HVM), SSD Volume Type
Are you launching a database instance? Try Amazon RDS.
Amazon Relational Database Service (RDS) makes it easy to set up, operate, and scale your database on AWS by automating time-consuming database management tasks. With RDS, you can easily deploy Amazon
Feedback
English (US)


Services
Resource Groups
Step 2: Choose an Instance Type
Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.
Filter by:
All instance types
Current generation
Show/Hide Columns
Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)
Family
Type
vCPUs
Memory (GiB)
Instance Storage (GB)
EBS-Optimized Available
Network Performance
IPv6 Support
Cancel
Previous
Review and Launch
Next: Configure Instance Details
Feedback
English (US)




```
$ sudo apt update -y && sudo apt upgrade -y
$ sudo apt-get install -y ca-certificates curl gnupg
lsb-release
```

Figure 1: Install required packages

```
$ curl -fsSL
https://download.docker.com/linux/debian/gpg | sudo
gpg --dearmor -o /usr/share/keyrings/docker-archive-
keyring.gpg

$ echo "deb [arch=$(dpkg --print-architecture) signed-
by=/usr/share/keyrings/docker-archive-keyring.gpg]
https://download.docker.com/linux/debian
$(lsb_release -cs) stable" | sudo tee
/etc/apt/sources.list.d/docker.list > /dev/null

$ sudo apt-get install docker.io -y
$ systemctl enable docker.service
$ systemctl start docker.service
```

Figure 2: Install docker

```
$ echo 1 > /proc/sys/net/ipv4/ip_forward
$ lsmod | grep br_netfilter
$ sudo modprobe br_netfilter
$ cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf
br_netfilter
EOF
$ cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf
net.bridge.bridge-nf-call-ip6tables = 1
net.bridge.bridge-nf-call-iptables = 1
EOF
$ sudo sysctl --system
$ cat <<EOF | sudo tee /etc/docker/daemon.json
{
  "exec-opts": ["native.cgroupdriver=systemd"],
  "log-driver": "json-file",
  "log-opts": {
    "max-size": "100m"
  },
  "storage-driver": "overlay2"
}
EOF
```

Figure 3: Change network settings

```
$ sudo apt-get update -y
$ sudo apt-get install -y apt-transport-https ca-
certificates curl
$ sudo curl -fsSLo /usr/share/keyrings/kubernetes-
archive-keyring.gpg
https://packages.cloud.google.com/apt/doc/apt-key.gpg
$ echo "deb [signed-by=/usr/share/keyrings/kubernetes-
archive-keyring.gpg] https://apt.kubernetes.io/
kubernetes-xenial main" | sudo tee
/etc/apt/sources.list.d/kubernetes.list
$ sudo apt-get update -y
$ sudo apt-get install -y kubelet kubeadm kubectl
$ sudo apt-mark hold kubelet kubeadm kubectl
```

Figure 4: Install kubernetes

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

Figure 5: Setup the kubeconfig

3 ANA autoscaler setup

- Copy the ana.sh bash script into the master node and make it executable using **chmod +x ana.sh**.
- Register the ana.sh bash script in crontab to run the script every minute.
- Copy the create_dormant.sh bash script into the master node and edit the script to input the awscli credentials taken from the aws console. Also make the script executable **chmod +x create_dormant.sh**.
- Register the create_dormant.sh bash script in crontab to run the script every minute.
- Copy the ec2-user-data file into the master node.
- Get the output of this command **kubeadm token create –print-join-command** and append it as the last of ec2-user-data file.
- Copy the logger bash script into the master node and make it executable **chmod +x logger**.
- Register the logger script in crontab of to run the script every minute.

4 EKS cluster setup

- Setup the awscli credentials in the command line in any of the EC2 instance server used in the ANA setup to run the commands for the creation of the EKS cluster only.
- Run the command to create the EKS cluster **eksctl create cluster –name eksana –region eu-west-1 –version 1.22 –asg-access** AWS (2022).
- Set the kubeconfig to access the cluster **aws eks update-kubeconfig –region eu-west-1 –name eksana**.
- Create the the policy for the eks cluster **aws iam create-policy –policy-name EksPolicy –policy-document file://cluster-autoscaler-policy.json** with the policy file present in the solution artifact.
- Take note of the arn of the policy created and use it in the creation of the iam service account **eksctl create iamserviceaccount –cluster=eksana –name=cluster-autoscaler –attach-policy-arn=XXX**
- Deploy the cluster autoscaler **kubectl apply -f cluster-autoscaler.yaml**.
- Annotate with the ARN of the created service account **kubectl annotate serviceaccount cluster-autoscaler -n kube-system eks.amazonaws.com/role-arn=XXX**

5 Load generation setup

- For both the setups the same load generation setup is installed.
- The php apache application is installed **kubectl apply -f php-apache.yaml**.
- The load_generator.sh bash script is copied into the worker nodes and made executable **chmod +x load_generator.sh**. Edit the IP with the clusterip used by the php-apache service **kubectl get svc php-apache**. Start the script to perform the load testing.

References

AWS (2022). EKS Manual, `https://docs.aws.amazon.com/eks/latest/userguide/create-cluster.html`. [Online; accessed 10-Aug-2022].

Kub (2022). Kubernetes Manual, `https://kubernetes.io/docs/home/`. [Online; accessed 10-Aug-2022].