

Cloud Computing Report

Student Number: 190336989

Student Name: XINQUAN LI

- Introduction

Docker is a set of platform as service product (PaaS), it deliver software by using containers. Different containers are isolated from each other, containers have different software, libraries and files, but container can communicate with other containers by using channels. Docker is a kind of OS-level virtualisation, but docker is light well than virtual machines. In this project, it is required that use Docker platform to set up a Docker host, host an application stack on Docker environment and analyse the results of benchmark.

- Task 1 Pull the web-application image and test

During this task, i use the command line to pull the image in Ubuntu and Mac OS. when the image is pulled, the application is running by using “docker run -d -p 127.0.0.1:8080:8080 nclcloudcomputing/javabenchmarkapp”, it direct run on my computer and virtual machine. The method of checking the state is to enter <http://localhost:8080/primecheck>. After entering it, the page will show the time, it is the time that perform a random prime number checking calculation. From this task, using command line may be a good way to learn Docker because it is more direct for users.

- Task 2 Deploy a multi service application in a Docker environment

There is two methods to create a Docker service on a single node Docker swarm with two instances of Docker images(nclcloudcomputing/javabenchmarkapp and MongoDB).I use two methods (command line and edit docker-compose.yml) to finish this task, there are many differences between these two methods.

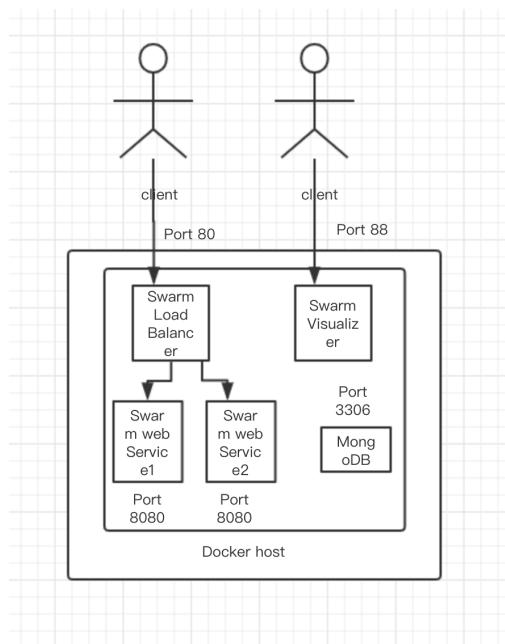


Fig.1.Docker Swarm UML Diagram

According to the UML diagram of Docker swarm, i deploy a multi service application in the environment of Docker. As it is shown, using command line or edit the file(docker-compose.yml) can build such service. I use these two methods to finish this task. The UML diagram gives a direct structure of this application. The port 3306 is to visit MongoDB and get the service of MongoDB. The port is to represent the service of specific process in a computer, it means that you can get different service from same port at same time, or it will be conflicted. By using visualiser, we can verify swarm graphical view.

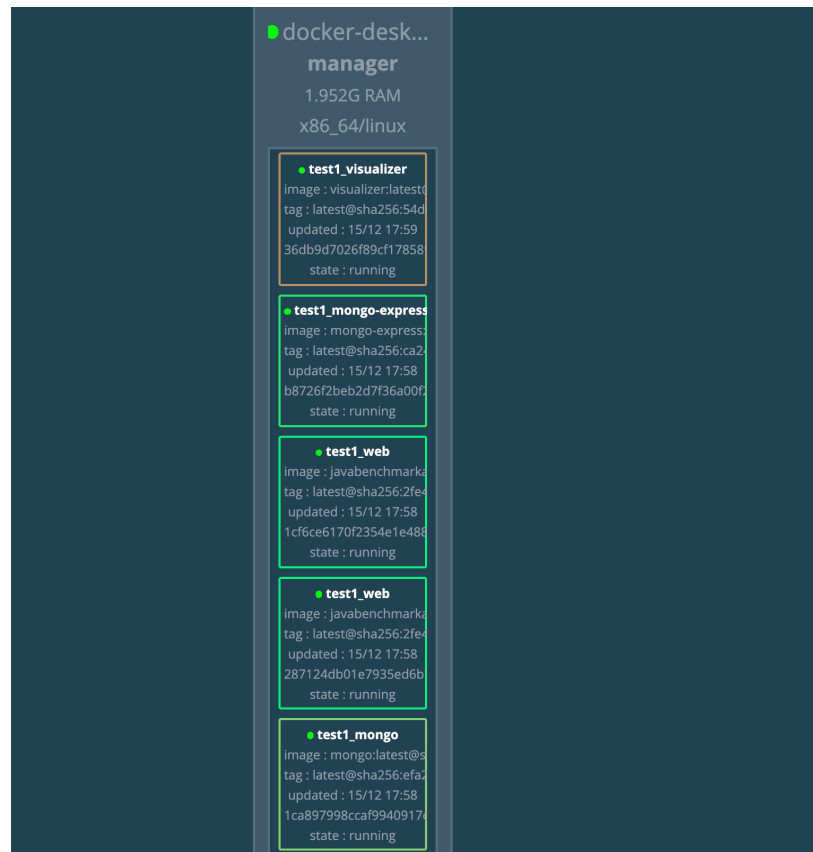


Fig.2.Docker Swarm Visualiser

• Task 3 Load Generator

In this task, it is required to create a program to generate load on the web application. The program need to visit the url of the web application several times and accept parameters. We need enter some parameters to run this program, because the request distribution contains two kinds of mathematical distribution.(Normal and Poisson) This parameters includes different values μ and σ in case of the Normal distribution and λ in the case of the Poisson distribution.

In the theory of mathematics, a normal distribution(or Gaussian) a type of continuous probability distribution for a real-valued random variable. And the two parameters are the mean(expectation) and standard deviation in the normal distribution.

Poisson distribution is a discrete probability distribution commonly found in statistics and probability. The parameter of Poisson distribution is the average number of occurrence of random events per unit time(or unit area).The Poisson distribution is suitable for describing the number of random events that occur per unit time. In the program of this task , i let users enter these parameters and they can choose the two different distribution. But sometimes, if the parameter of

Poisson distribution are bigger than 20, it is usually thought that the Poisson distribution are equivalent to the Normal distribution. As the parameter of Poisson increases, the Poisson distribution gradually becomes symmetrical, which is different from Normal distribution. Sometimes when i run the program , it will generate some negative numbers, and the IDE will warn. So i choose use absolute value of the number to let the time sleep. Here is the UML diagram of task 3.

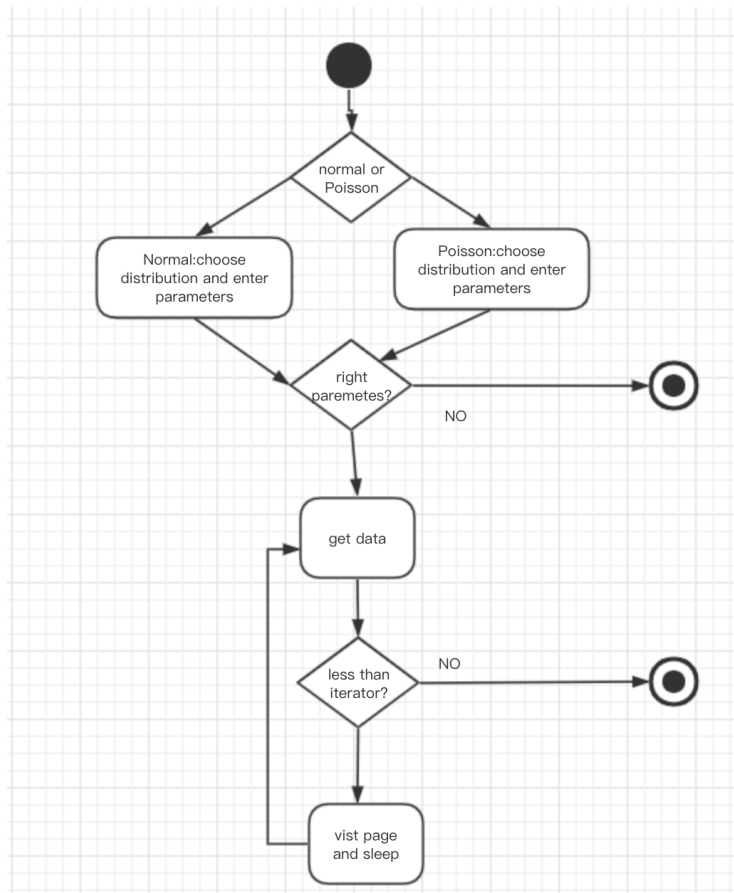


Fig.3. The UML of Load Generator

The attributes of Normal distribution and Poisson distribution are linked to their parameters very closely. If we collect the data from load generator , it should be meaning to verify the mathematical nature of Normal distribution and Poisson distribution.

- Task 4 Add a Docker monitoring tool

In this task, it is required that we need to pull a specific image to add a monitoring tool in the host of Docker environment. First we need pull the image named google/cadvisor, it is convenient to get the image from the docker hub.

The essence of monitoring tool is that we must ensure the performance of program, process and containers is on the way that expected. It is very important to check quality during the running time of a process, a program and a container. Luckily, there are many use tools to do that.

By checking the method of installing cadvisor, we run the monitoring tool and verify if it can show the information of containers or not.

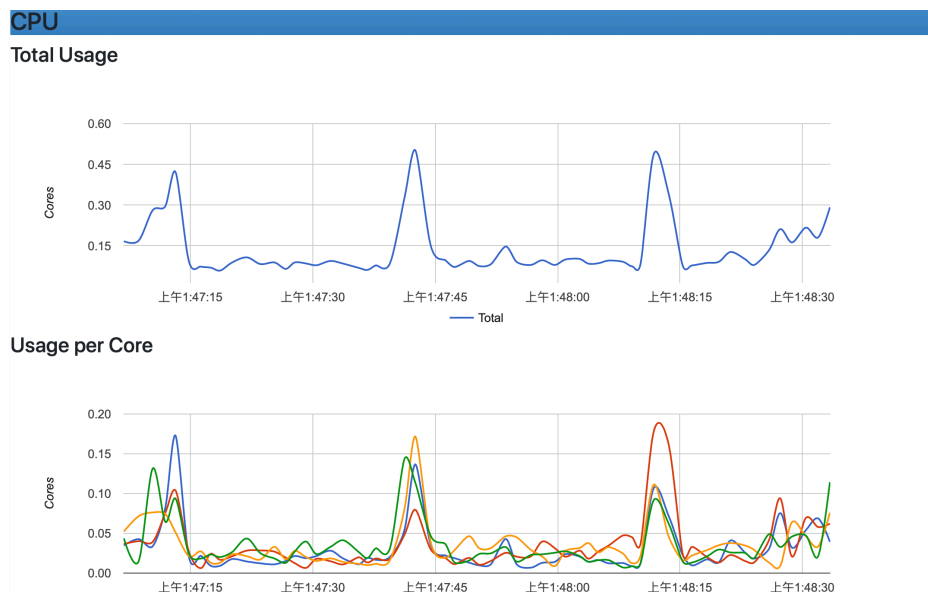


Fig.4. The Monitoring Tool:Cadvisor

If we run the program in the task 3, it will generate load to the website and the monitor will record the information of usage of each container, such as CPU utilisation , Memory utilisation and I/O. These information will directly show on the page. If we enter different parameters or choose different distribution it will generate different load for the website, the graph in the monitor tool should be very same to the math image. Later, if we could get the data from the monitor by using API, we could verify it for the second time.

- Task 5 Insert benchmark result into database

In this task, we need to write a program to get the data of container instance of the web application by using the API of the monitor, and then it is very important to insert these data into the database(MongoDB). This data are generated from the program in the task3 that generate load by visiting the URL of the website(<http://localhost:8080/primecheck>). I use python to write the program, and change the data that from the monitor into Json format then put them into the database.

During this task, i realise the strength of Docker. Compared to virtual machines, Docker is much simpler. Because users do not need to run a bloated operating system. For example, the image in Docker is much more lightweight than traditional application on the operating or virtual machine.

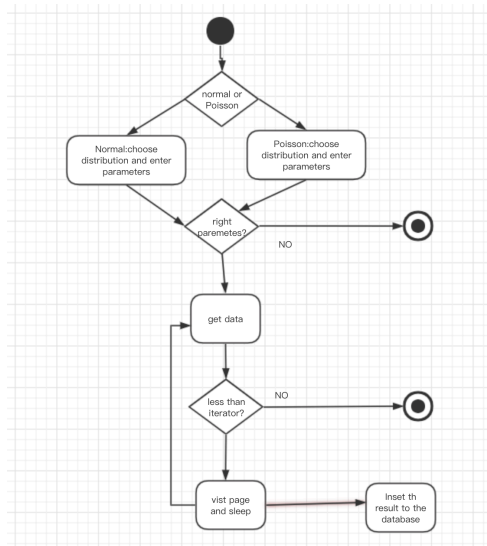


Fig.5. The process of inserting data into database

- Task 6 Deploy the components developed in Task 1 - Task 5 on AWS Cloud Platform
- AWS(Amazon Web Service) is a professional cloud computing service provided by Amazon company. It provide cloud computing platform, service and infrastructure, and these are all based on the demand of users. For example, Amazon Elastic allow users to use a virtual cluster of computers based on the need of uses at any time.

IaaS: Infrastructure as a Service

PaaS: Platform as a Service

SaaS: Software as a Service

During this task, i have gained many experience of using AWS. First you need to choose EC2 service, then choose an instance(such as Ubuntu or Cent OS). By using a SSH, i connect to the instance that i create. And it is quite similar to the the job that i have finished on my computer.

Edit inbound rules						
Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ		Description ⓘ	
Custom TCP I ↓	TCP	8080	Custom ↓	0.0.0.0/0	primecheck	✕
SSH ↓	TCP	22	Custom ↓	0.0.0.0/0	e.g. SSH for Admin Desktop	✕
Custom TCP I ↓	TCP	5000	Custom ↓	0.0.0.0/0	visualiser	✕
Custom TCP I ↓	TCP	8080	Custom ↓	0.0.0.0/0	monitor	✕
Add Rule						

Fig.6. Inbound rules on AWS

There are many benefits of AWS. For example, the AWS platform is used to use, same and reliable. And it is also scalable and high performance. In the business world, it is very convenient for many companies.

- Analysis and discussion

In this task, it is required that we need to graphic the results of benchmark. I use python and some three party libraries to finish this job. For example, numpy, matplotlib and mongoclient.

When I am doing this task, it is reasonable to consider that the results must have many correlation between the load generator in the task 3. It means the graph of the results may be very similar to the two distributions in the task 3. Because if we run the program, the graph of CPU, Memory may change in the monitor. So it is considerable to draw these two distributions with giving different parameters. And then the graph of two distributions can compare to the data in database(MongoDB). We can verify the results if it match the distribution or not. At last, we may discuss the results and try to find the reason behind it. So I consider there should be more groups of two distribution.

Normal Distribution	mu	sigma
group1	0	1
group2	1	2
group3	3	1.5

Table.1. Three group of Normal distribution with different parameters.

Here are the graph of these three groups:

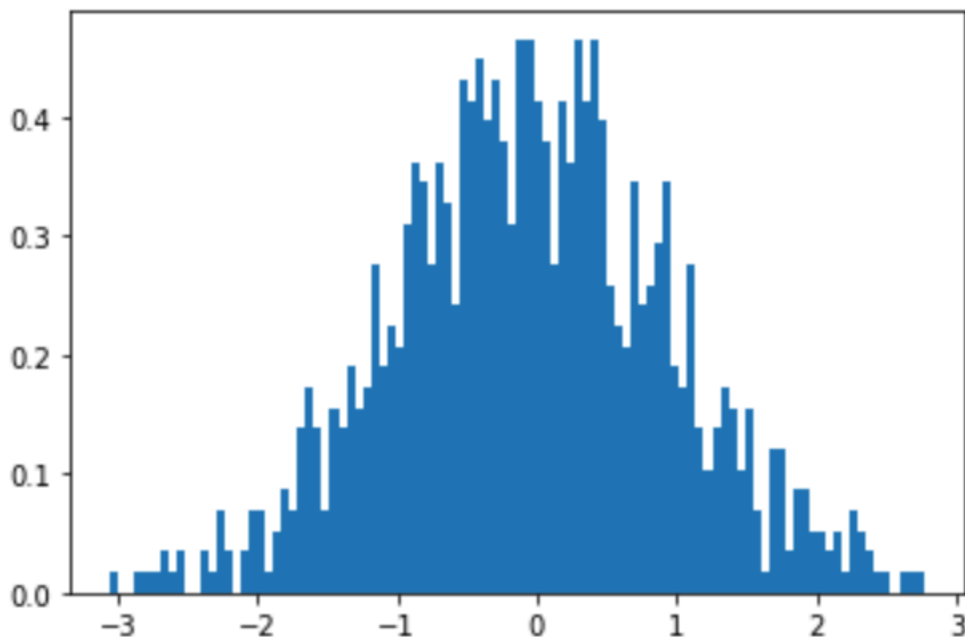
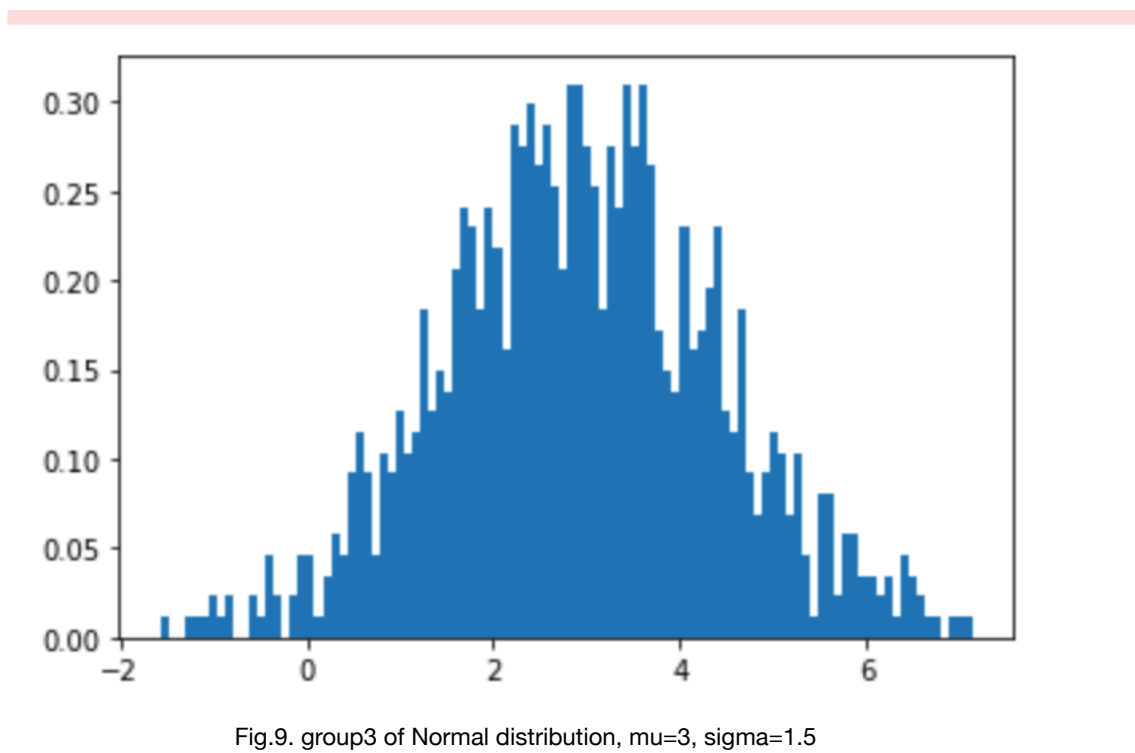
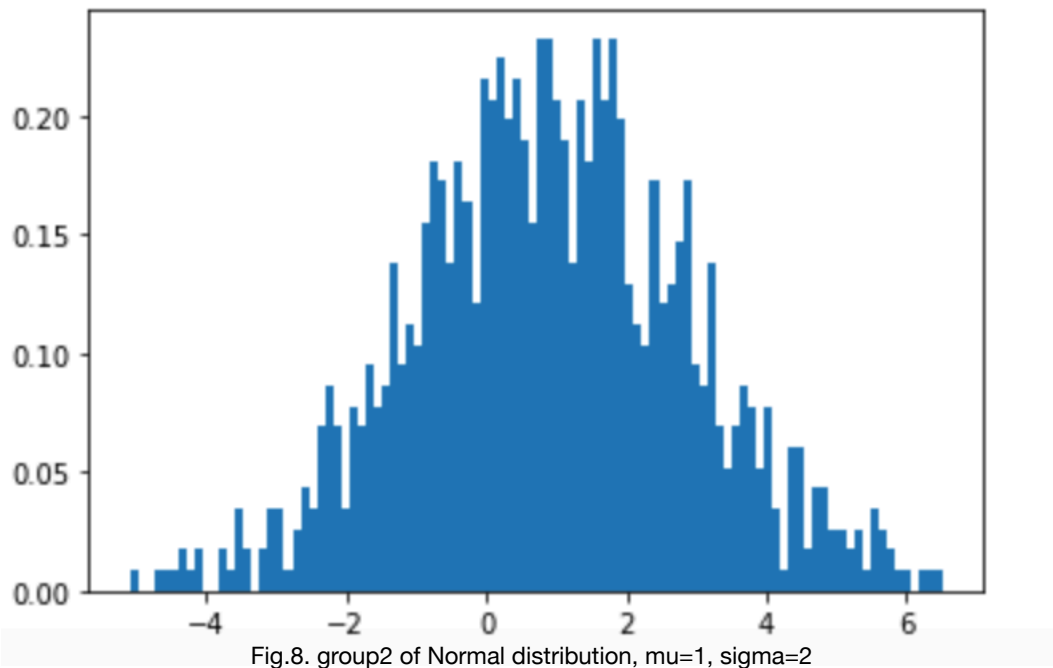


Fig.7. group1 of Normal distribution, $\mu=0$, $\sigma=1$



Similarly, there are three groups of Poisson distribution.

Poisson distribution	lamda
group1	1
group2	2.5
group3	3.5

Table.2.Three group of Poisson distribution with different parameters.

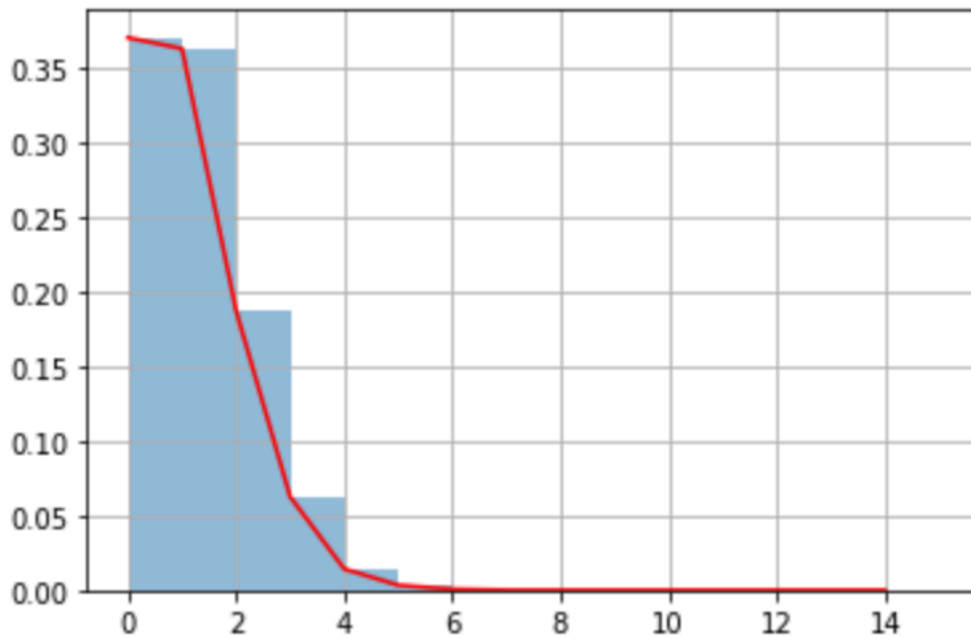


Fig.10. group1 of Poisson distribution, lamda=1

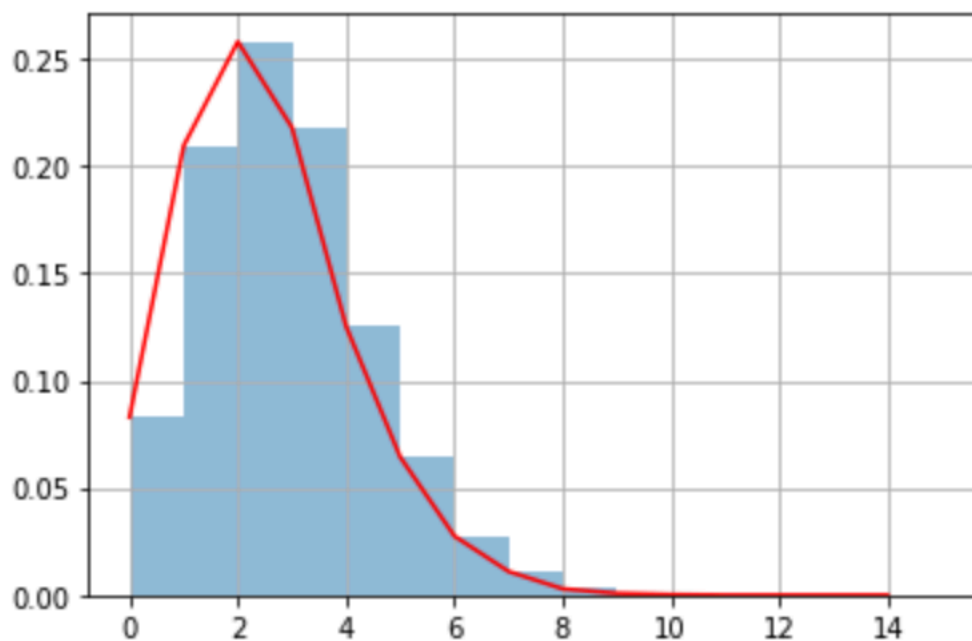


Fig.11. group2 of Poisson distribution, lamda=2.5

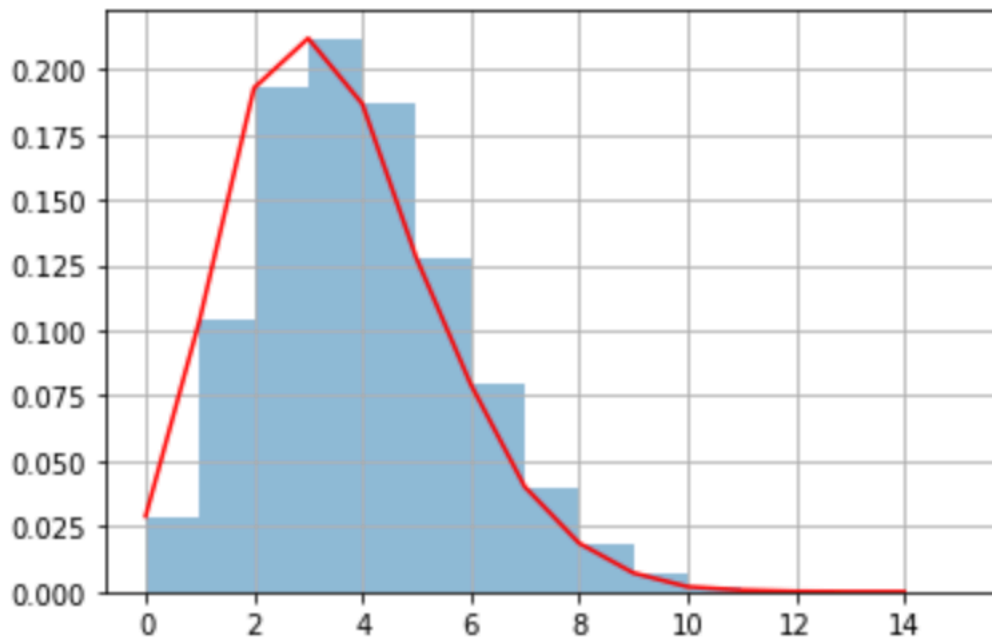


Fig.12. group3 of Poisson distribution, lamda=3.5

If we run task3, the task 3 will generate load for the website. We can get the data from the database and compare data with the graph in the monitor tool.

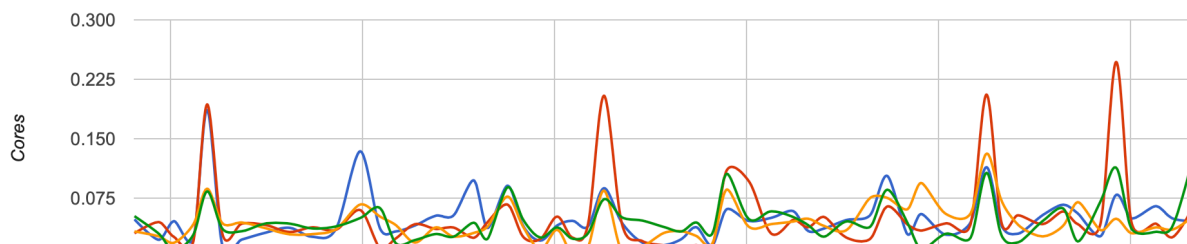


Fig.13. the cpu graph in the Cadvisor

Compared to the data in the database, it is quite similar to Cadvisor.

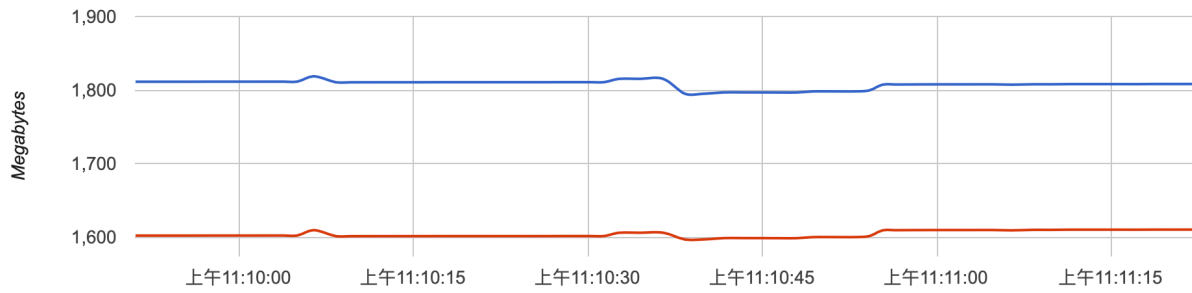


Fig.14. the memory graph in the Cadvisor

The graph of CPU is quite similar to the data in the database and the graph of distribution. It will be a little difference between the gaps of different groups. Maybe the application and docker need some time to adjust.

The use of memory is quite small, as we mentioned, Docker is more lightweight.

The net is always a straight line, I guess it should be worked well if it is a straight line. And i change the time of sleep, it is still a straight line.