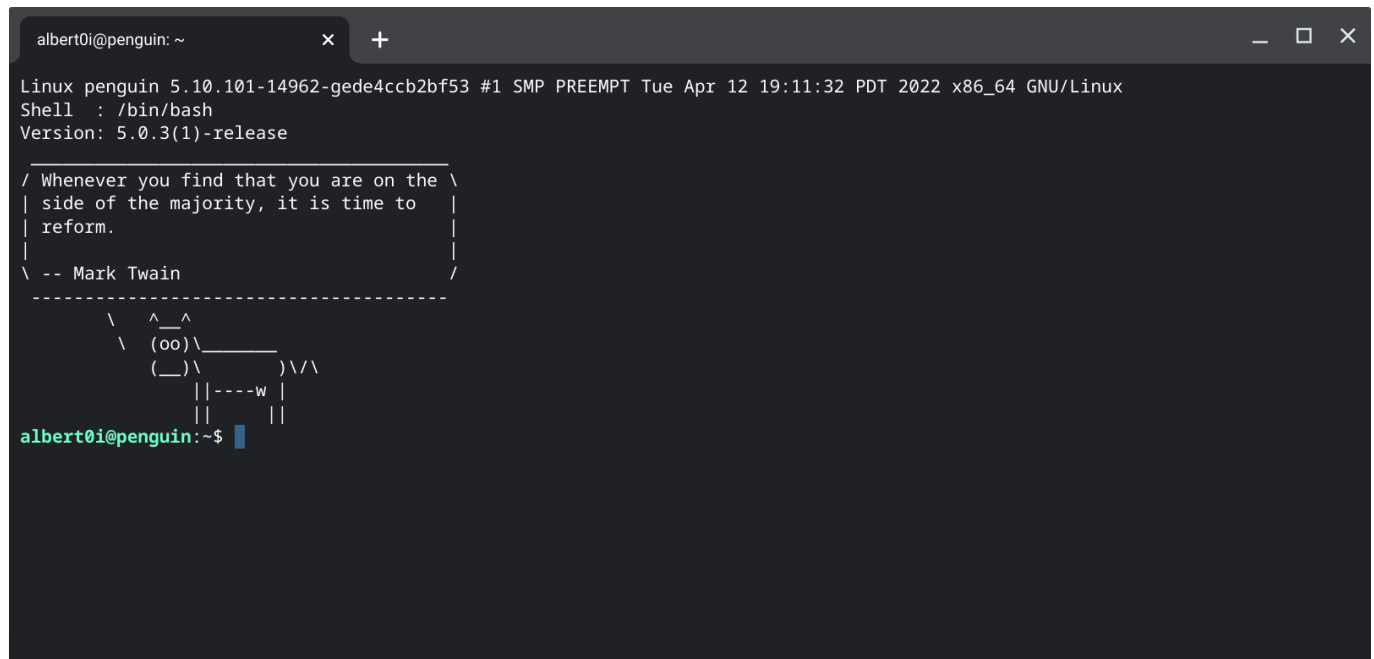


Docker swarm on Oracle Cloud (Free Tier)

— 窮L的後現代生活



```
albert0i@penguin: ~
Linux penguin 5.10.101-14962-gede4ccb2bf53 #1 SMP PREEMPT Tue Apr 12 19:11:32 PDT 2022 x86_64 GNU/Linux
Shell : /bin/bash
Version: 5.0.3(1)-release

/ Whenever you find that you are on the \
| side of the majority, it is time to   |
| reform.                             |
\ -- Mark Twain                        /

-----
\   ^__^
\  (oo)\_______
    (__)\       )\/\
       ||----w |
       ||     ||

albert0i@penguin:~$
```

I. Oracle Cloud Infrastructure (OCI)

Oracle Cloud Free Tier allows you to explore a wide range of Oracle Cloud Infrastructure products and a set of Always Free offers. Regarding to the number of compute instances available to Oracle Free Tier account. Following are excerpts from [\[1\]](#).

Depending on the size of the boot volume and the number of OCPUs that you allocate to each Ampere A1 Compute instance, you can create up to four compute instances. The minimum boot volume size for each instance is 47 GB, regardless of shape. Your account comes with 200 GB of Always Free block volume storage which you use to create the boot volumes for your compute instances.

For example, using the default boot volume size of 47 GB, you could provision two instances using the VM.Standard.E2.1.Micro shape, and two Ampere A1 Compute instances that each have 2 OCPUs. Or, you could provision four Ampere A1 Compute instances with 1 OCPU each, and zero instances using the VM.Standard.E2.1.Micro shape. Many combinations are possible, depending on how you allocate your block storage and Ampere A1 Compute OCPUs. See [Details of the Always Free compute instances](#) for more information on allocating OCPU and memory resources when creating Ampere A1 Compute instances.

For Virtual Cloud Networks (VCNs):

Free Tier tenancies (tenancies that are not paid and do not have Free Trial credits) can have up to 2 virtual cloud networks (VCNs). A VCN is a software-defined network that you set up in the Oracle Cloud Infrastructure data centers in a particular region. VCNs include IPv4 and IPv6 support.

In a word,

- Two **Micro instances (AMD processor)** [Always Free] VM instances using the VM.Standard.E2.1.Micro shape; with AMD EPYC (霄龍) 7551 2GHz dual CPU, 1GB memory and 47GB hard disk each;

```
ubuntu@instance-20220508-0913-wc x ubuntu@instance-20220509-0853-wc x +
ubuntu@instance-20220508-0913-worker:~$ sudo lshw -c cpu
*-cpu
  description: CPU
  product: AMD EPYC 7551 32-Core Processor
  vendor: Advanced Micro Devices [AMD]
  physical id: 400
  bus info: cpu@0
  version: pc-i440fx-4.2
  slot: CPU 0
  size: 2GHz
  capacity: 2GHz
  width: 64 bits
  capabilities: fpu fpu_exception wp vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clf
lsh mmx fxsr sse sse2 ht syscall nx mmxext fxsr_opt pdpe1gb rdtscp x86-64 rep_good nopl cpuid extd_apicid tsc_kn
own_freq pni pclmulqdq ssse3 fma cx16 sse4_1 sse4_2 x2apic movbe popcnt tsc_deadline_timer aes xsave avx f16c rdr
and hypervisor lahf_lm cmp_legacy cr8_legacy abm sse4a misalignsse 3dnowprefetch osvw topoext perfctr_core ssbd i
bpb vmcall fsgsbase tsc_adjust bmi1 avx2 smep bmi2 rdseed adx smap clflushopt sha_ni xsaveopt xsavec xgetbv1 xsa
ves clzero xsaveerptr virt_ssbd arat arch_capabilities
  configuration: cores=1 enabledcores=1 threads=2
ubuntu@instance-20220508-0913-worker:~$ sudo dmidecode -t processor | grep Speed
    Max Speed: 2000 MHz
    Current Speed: 2000 MHz
```

- Two **Ampere A1 Compute instances (Arm processor)** VM instances using the VM.Standard.A1.Flex shape; with Neoverse-N1 2GHz single CPU, 6GB memory and 47GB hard disk each;

```
ubuntu@instance-20220508-0913-wc x ubuntu@instance-20220509-0853-wc x +
ubuntu@instance-20220509-0853-worker3:~$ sudo lshw -c cpu
*-cpu
  description: CPU
  vendor: QEMU
  physical id: 400
  bus info: cpu@0
  version: virt-4.2
  slot: CPU 0
  size: 2GHz
  capacity: 2GHz
  configuration: cores=1 enabledcores=1 threads=1
ubuntu@instance-20220509-0853-worker3:~$ sudo dmidecode -t processor | grep Speed
    Max Speed: 2000 MHz
    Current Speed: 2000 MHz
ubuntu@instance-20220509-0853-worker3:~$
```

in favour of more powerful CPUs.

II. Docker swarm ingress network

Docker swarm makes use a special overlay network **ingress** to facilitate load balancing among a service's nodes. The ingress network is created automatically when you initialize or join a swarm. Most users do not need to customize its configuration, but Docker allows you to do so.[\[2\]](#)


By default, ingress network resides in "10.0.0.0/24" which collides with default Virtual Cloud Networks (VCN), therefore the ingress network doesn't appear to be working properly! [3]

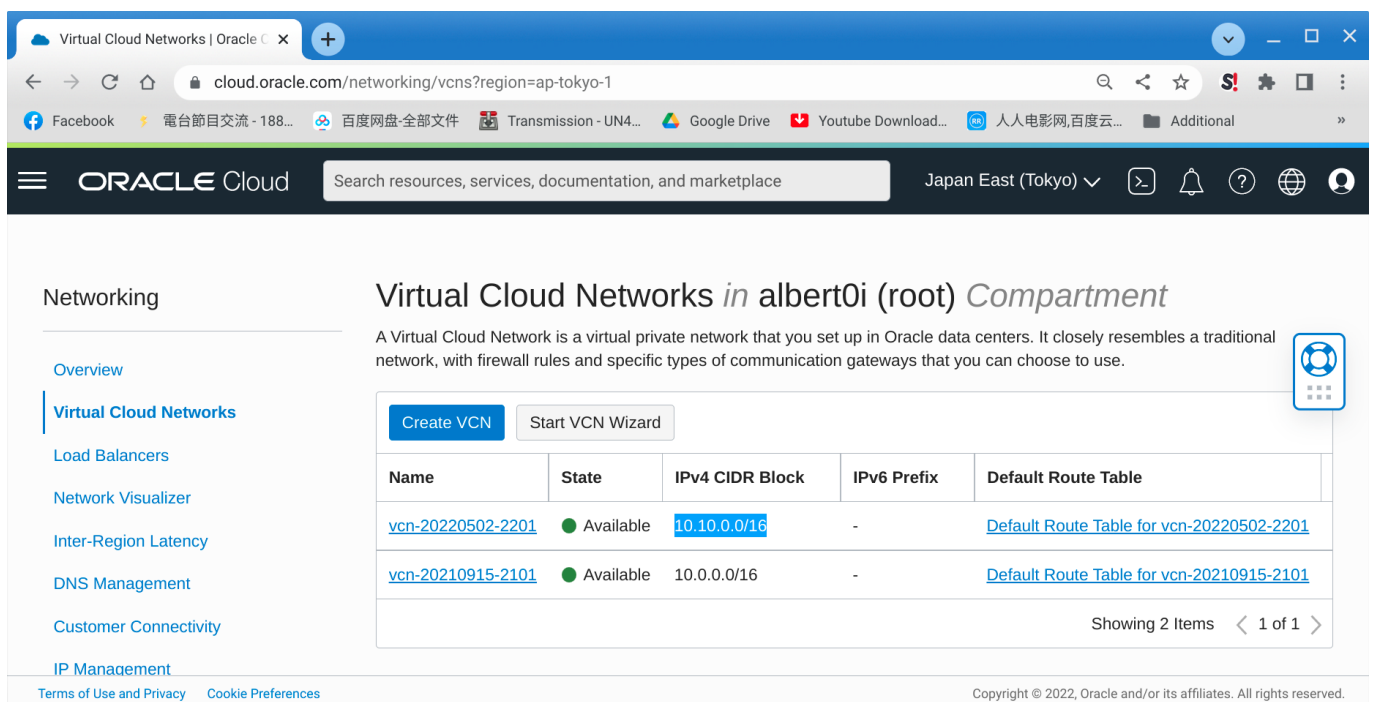
```
ubuntu@instance-20220502-1505: ~/app$ docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
25636aeb5790        bridge             bridge              local
0f592551f43c        docker_gwbridge    bridge              local
a55b7009a0e5        host               host                local
ms8hzhg49dey        ingress            overlay             swarm
7445b6699b8c        none               null                local
ubuntu@instance-20220502-1505: ~/app$ docker network inspect ingress
[
  {
    "Name": "ingress",
    "Id": "ms8hzhg49deyayqg63je0kl3w",
    "Created": "2022-05-02T07:37:04.144292097Z",
    "Scope": "swarm",
    "Driver": "overlay",
    "EnableIPv6": false,
    "IPAM": {
      "Driver": "default",
      "Options": null,
      "Config": [
        {
          "Subnet": "10.0.0.0/24",
```

In addition, all traffic between clients is blocked by default in Oracle cloud. The solution is to add rules for your subnet to the security list of the VCN the VMs are connected to. [4]

III. Virtual Cloud Networks (VCN)

Free Tier tenancies can have up to 2 virtual cloud networks (VCNs).

- Open Oracle Cloud Menu by clicking the left upper 
- Choose Networking → Virtual Cloud Networks → Create VCN
- Give it a name **vcn-20220502-2201** for example, the subnet can be "10.10.1.0/24" or something else



Virtual Cloud Networks in albert0i (root) *Compartment*

A Virtual Cloud Network is a virtual private network that you set up in Oracle data centers. It closely resembles a traditional network, with firewall rules and specific types of communication gateways that you can choose to use.

[Create VCN](#) [Start VCN Wizard](#)

Name	State	IPv4 CIDR Block	IPv6 Prefix	Default Route Table
vcn-20220502-2201	Available	10.10.0.0/16	-	Default Route Table for vcn-20220502-2201
vcn-20210915-2101	Available	10.0.0.0/16	-	Default Route Table for vcn-20210915-2101

Showing 2 Items < 1 of 1 >

- On the public subnet created, click to edit the **Default Security List**
- Add **Ingress Rules** for port 80, 8080, 3000, 8000, and 443 etc
- Add two more **Ingress Rules** for subnet '10.10.0.0/16' TCP and UDP traffics

Virtual Cloud Networks | Oracle Cloud

cloud.oracle.com/networking/vcns/ocid1.vcn.oc1.ap-tokyo-1.amaaaaaahr3lrzqafgzrno4dn6nn7nifwjbkansw75csl...

You are using a Free Tier account. To access all services and resources, [upgrade](#) to a paid account.

ORACLE Cloud docker swarm Japan East (Tokyo)

Enabled	Direction	Source	Protocol	Destination	Ports	Actions
<input type="checkbox"/>	No	0.0.0.0/0	ICMP	All	ICMP traffic for port: All	
<input type="checkbox"/>	No	0.0.0.0/0	TCP	All	TCP traffic for port: 443 HTTPS	
<input type="checkbox"/>	No	10.10.0.0/16	TCP	All	TCP traffic for port: All	
<input type="checkbox"/>	No	10.10.0.0/16	UDP	All	UDP traffic for port: All	

0 Selected Showing 11 Items < 1 of 1 >

Terms of Use and Privacy Cookie Preferences Copyright © 2022, Oracle and/or its affiliates. All rights reserved.

- Once created, subsequent creation of instances will use this customized network
- OCI instances can't attach more than one networks
- You can choose network settings ONLY in instance creation
- OCI instance can't change network settings

Instances | Oracle Cloud Infrastructure

cloud.oracle.com/compute/instances/create?region=ap-tokyo-1

ORACLE Cloud Search resources, services, documentation, and marketplace Japan East (Tokyo)

Create compute instance

[Networking](#) is how your instance connects to the internet and other resources in the Console. To make sure you can [connect to your instance](#), assign a public IP address to the instance.

Primary network

☒ Select existing virtual cloud network ☐ Create new virtual cloud network ☐ Enter subnet OCID

Virtual cloud network in **albert0i (root)** [\(Change Compartment\)](#)

vcn-20220502-2201

Subnet

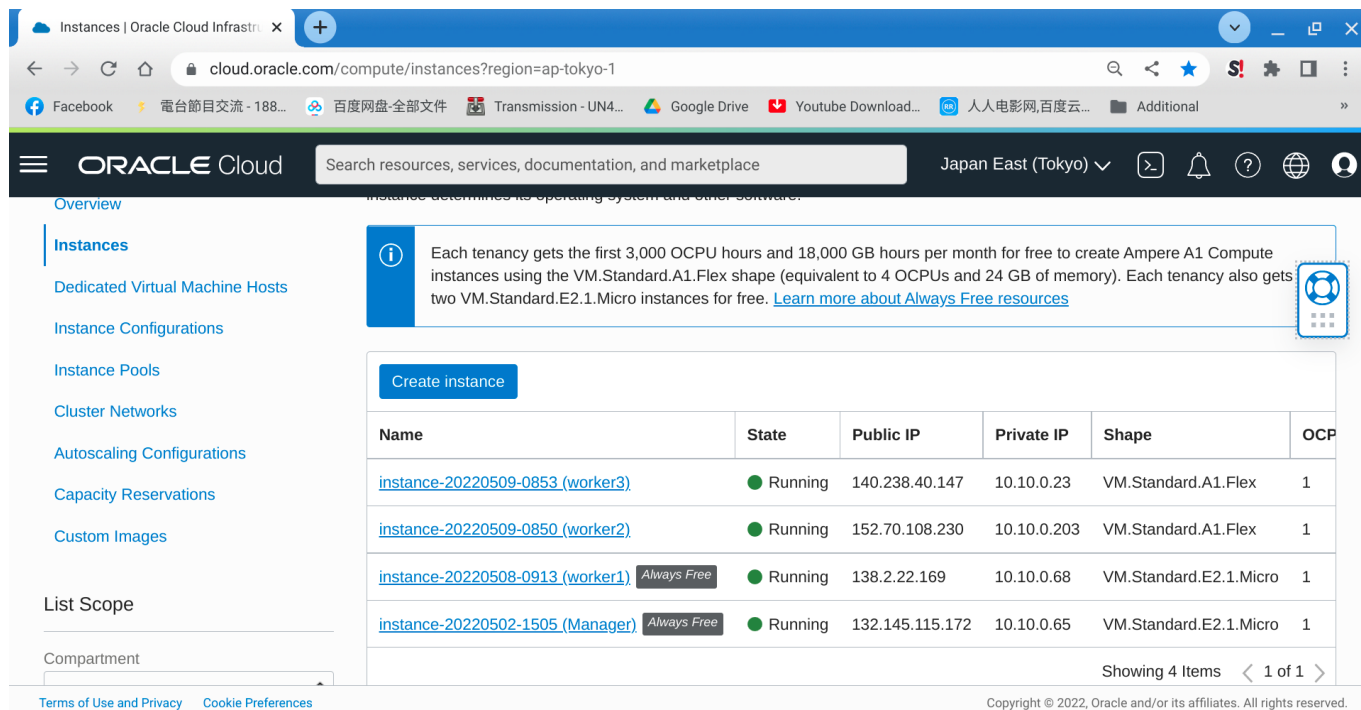
☒ Select existing subnet ☐ Create new public subnet

Subnet in **albert0i (root)** [\(Change Compartment\)](#)

Create Save as stack Cancel

Terms of Use and Privacy Cookie Preferences Copyright © 2022, Oracle and/or its affiliates. All rights reserved.

IV. OCI Instances



Instances | Oracle Cloud Infrastr

cloud.oracle.com/compute/instances?region=ap-tokyo-1

Overview

Instances

Dedicated Virtual Machine Hosts

Instance Configurations

Instance Pools

Cluster Networks

Autoscaling Configurations

Capacity Reservations

Custom Images

List Scope

Compartment

Terms of Use and Privacy

Cookie Preferences

Copyright © 2022, Oracle and/or its affiliates. All rights reserved.

Each tenancy gets the first 3,000 OCPU hours and 18,000 GB hours per month for free to create Ampere A1 Compute instances using the VM.Standard.A1.Flex shape (equivalent to 4 OCPUs and 24 GB of memory). Each tenancy also gets two VM.Standard.E2.1.Micro instances for free. [Learn more about Always Free resources](#)

Create instance

Name	State	Public IP	Private IP	Shape	OCP
instance-20220509-0853 (worker3)	Running	140.238.40.147	10.10.0.23	VM.Standard.A1.Flex	1
instance-20220509-0850 (worker2)	Running	152.70.108.230	10.10.0.203	VM.Standard.A1.Flex	1
instance-20220508-0913 (worker1) Always Free	Running	138.2.22.169	10.10.0.68	VM.Standard.E2.1.Micro	1
instance-20220502-1505 (Manager) Always Free	Running	132.145.115.172	10.10.0.65	VM.Standard.E2.1.Micro	1

Showing 4 Items < 1 of 1 >

V. Scripts

.profile

```
#
# Add to the end
# some more ls aliases
alias ll='ls -l'
alias la='ls -A'
#alias l='ls -CF'
alias ver='uname -a;echo "Shell : " $SHELL; echo "Version:" $BASH_VERSION'
alias cls='clear'

# some more environment variables
export PATH=$PATH:.
set -o allexport;source /home/ubuntu/.env;set +o allexport

# replace accordingly
echo '...'
echo '...'
echo '...'
echo '...'
echo '...'
echo '...'
ver
```

.env

```
NODE_ENV=production
```

```
MONGO_USER=root

MONGO_PASSWORD=root

SESSION_SECRET=secret

MONGO_INITDB_ROOT_USERNAME=root

MONGO_INITDB_ROOT_PASSWORD=root

PUBLIC_IP=132.145.115.172 #replace accordingly

INT_NAME="ens3" #replace accordingly
PRIVATE_IP=$(ip addr show $INT_NAME | grep "inet\b" | awk '{print $2}' |
cut -d/ -f1)
```

.bash_logout

```
## Add to the end
echo
echo Goodbye $(whoami), have a nice day!
sleep 1
clear
```

install packages

```
sudo apt-get update -y
sudo apt install docker.io -y
sudo apt install docker-compose -y
sudo apt install fortune-mod -y
sudo apt install cowsay -y
sudo apt install mc -y
```

disable ufw and iptables

```
sudo systemctl status ufw
sudo systemctl disable ufw
sudo systemctl stop ufw
sudo systemctl status ufw

sudo systemctl status iptables
sudo systemctl disable iptables
sudo systemctl stop iptables
sudo systemctl status iptables
```

configure docker

```
sudo chmod 666 /var/run/docker.sock  
sudo usermod -aG docker ${USER}
```

configure docker swarm

```
docker swarm init --advertise-addr 10.10.0.65
```

To add a worker to this swarm, run the following command:

```
docker swarm join --token SWMTKN-1-  
0t7xb0wo9ttl75ofimvlikhgprmqtvz8f8yfyarub5b0que9g3-  
bgl7zox94yqhwpogb2eln5e4d 10.10.0.65:2377
```

To add a manager to this swarm, run the following command:

```
docker swarm join --token SWMTKN-1-  
0t7xb0wo9ttl75ofimvlikhgprmqtvz8f8yfyarub5b0que9g3-  
782k5kk40a2itv8fmbrrxmorn 10.10.0.65:2377
```

VI. Deployment

[illegible]

```
ubuntu@instance-20220502-1505: ~/ x ubuntu@instance-20220508-0913-wc x +
ubuntu@instance-20220502-1505:~/app$ docker stack deploy -c docker-compose.yml -c docker-compose.prod.yml myapp
Creating network myapp_default
Creating service myapp_redis
Creating service myapp_mongo
Creating service myapp_nginx
Creating service myapp_node-app
ubuntu@instance-20220502-1505:~/app$ docker stack ps myapp
ID              NAME                IMAGE                NODE                DESIRED STATE  CURREN
T STATE        ERROR                PORTS
sk887rn87h11    myapp_mongo.1       mongo:4.4            instance-20220508-0913-worker  Running        Prepar
ing 23 seconds ago
nonat5l209r7    myapp_nginx.1       nginx:stable-alpine  instance-20220502-1505        Running        Runnin
g 10 seconds ago
r7p57taivuru    \_ myapp_nginx.1    nginx:stable-alpine  instance-20220502-1505        Shutdown      Failed
18 seconds ago    "task: non-zero exit (1)"
zggxww6k0nyx    myapp_node-app.1    albert0i/node-app:1.0  instance-20220508-0913-worker  Running        Runnin
g 9 seconds ago
eftsxmr18o86    myapp_node-app.2    albert0i/node-app:1.0  instance-20220502-1505        Running        Runnin
g 15 seconds ago
rtiiv28vstum    myapp_redis.1       redis:6.2.6          instance-20220502-1505        Running        Runnin
g 24 seconds ago
ubuntu@instance-20220502-1505:~/app$
```



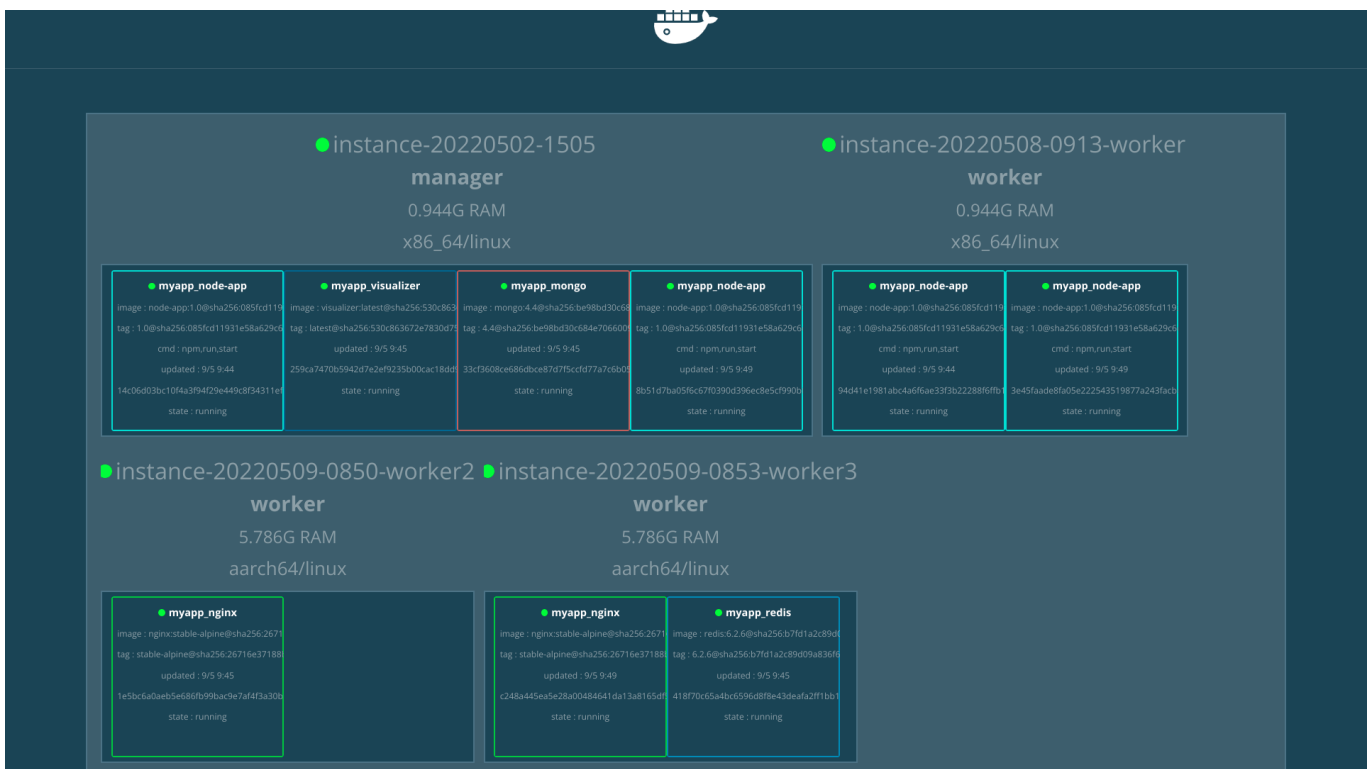
```

albert0i@penguin: ~
ubuntu@instance-20220502-1505: ~/
albert0i@penguin: ~

ubuntu@instance-20220502-1505:~/app$ docker service ls
ID                NAME                MODE                REPLICAS            IMAGE                PORTS
ppml2nfhs350     myapp_mongo         replicated          1/1                 mongo:4.4
paoat4g9cqt2     myapp_nginx         replicated          0/1                 nginx:stable-alpine *:80->80/tcp
v98u8z3vvsjy     myapp_node-app      replicated          2/2                 albert0i/node-app:1.0
h6ujmmtgk75     myapp_redis         replicated          0/1                 redis:6.2.6
y4cx4cbawh32     myapp_visualizer    replicated          1/1                 dockersamples/visualizer:latest *:8080->8080/tcp

ubuntu@instance-20220502-1505:~/app$ docker service scale myapp_node-app=4
myapp_node-app scaled to 4
overall progress: 4 out of 4 tasks
1/4: running [=====>]
2/4: running [=====>]
3/4: running [=====>]
4/4: running [=====>]
verify: Service converged
ubuntu@instance-20220502-1505:~/app$ docker service scale myapp_nginx=2
myapp_nginx scaled to 2
overall progress: 2 out of 2 tasks
1/2: running [=====>]
2/2: running [=====>]
verify: Service converged
ubuntu@instance-20220502-1505:~/app$

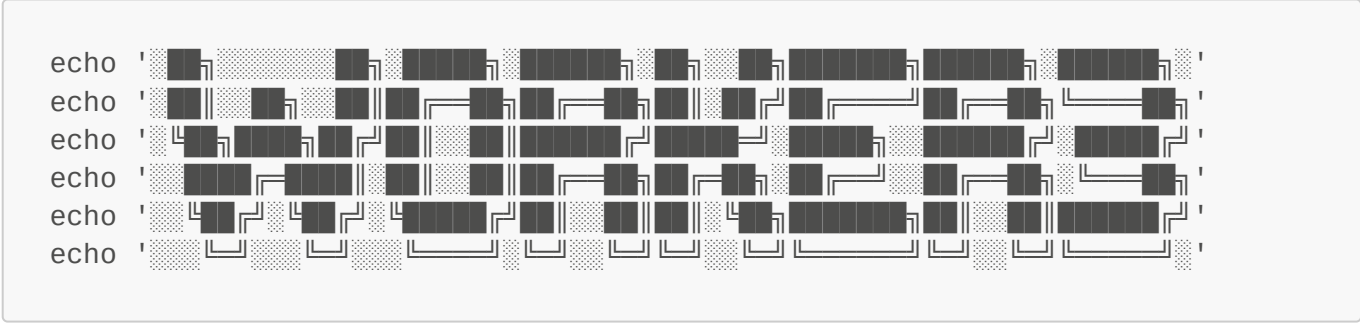
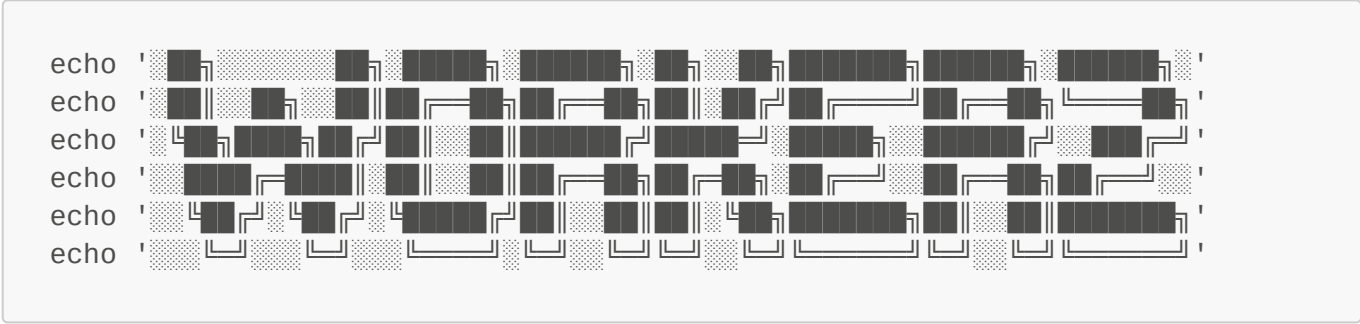
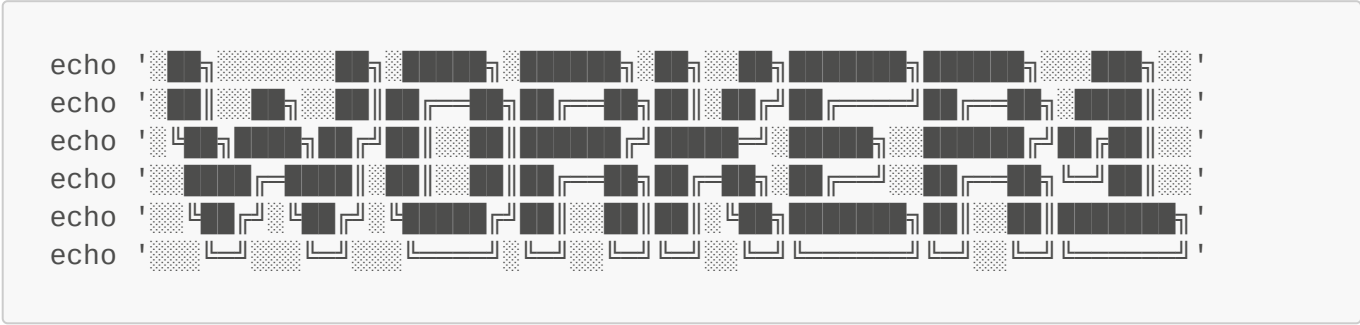
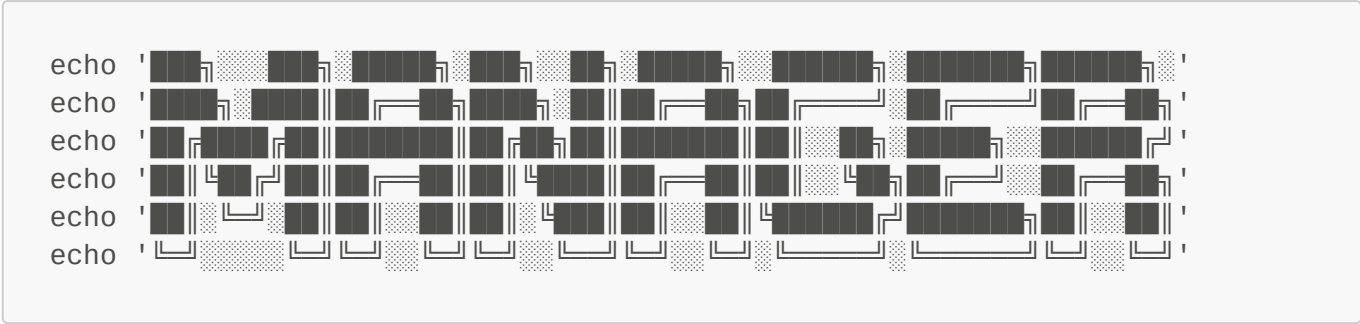
```



VII. Reference

1. [Oracle Cloud's Free Tier|Always Free Resources](#)
2. [Manage swarm service networks](#)
3. [Docker Swarm on Oracle always free tier nodes: no route to host](#)
4. [Unable to join docker swarm using Oracle Cloud VMs connected to the same VCN](#)
5. [Got permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get http://%2Fvar%2Frun%2Fdocker.sock/v1.24/images/json?all=1: dial unix /var/run/docker.sock: connect: permission denied code example](#)
6. [dockersamples/docker-swarm-visualizer](#)
7. [Text Art](#)

VII. Text Art



EOF (2022/05/21)