

Network Administration/System Administration (NTU CSIE, Spring 2024)

Homework #5 - Virsh, Docker & Kubernetes

B12902110 呂承諺

March 29, 2024

Virsh

1. Command

```
virt-install \
  --name b12902110 \
  --vcpus 8 \
  --memory 8192 \
  --disk /tmp2/b12902110/nasahw5/ubuntu.qcow2,format=qcow2,size=20 \
  --network user,mac=52:54:00:90:21:10 \
  --graphics type=vnc,password=902110,port=5950,listen=0.0.0.0 \
  --cdrom /tmp2/nasa-hw5/ubuntu.iso
```

Result

virsh list:

```
b12902110@ws1: ~
$ virsh list
  Id    Name         State
  ----  -
  4     b12902110    running
```

After boot up:

```
Ubuntu 22.04.4 LTS nasa-hw5 tty1
nasa-hw5 login: [ 19.593706] cloud-init[937]: Cloud-init v. 23.3.3-0ubuntu0~22.04.1 running 'modules:config' at Fri, 29 Mar 2024 07:54:16 +0000. Up 19.52 seconds.
[ 20.081263] cloud-init[970]: Cloud-init v. 23.3.3-0ubuntu0~22.04.1 running 'modules:final' at Fri, 29 Mar 2024 07:54:16 +0000. Up 20.02 seconds.
[ 20.151548] cloud-init[970]: Cloud-init v. 23.3.3-0ubuntu0~22.04.1 finished at Fri, 29 Mar 2024 07:54:16 +0000. Datasource DataSourceNone. Up 20.14 seconds
[ 20.151989] cloud-init[970]: 2024-03-29 07:54:16,666 - cc_final_message.py[WARNING]: Used fallback datasource
```

After login:

```
Ubuntu 22.04.4 LTS nasa-hw5 tty1
nasa-hw5 login: [ 19.593706] cloud-init[937]: Cloud-init v. 23.3.3-0ubuntu0~22.04.1 running 'modules:config' at Fri, 29 Mar 2024 07:54:16 +0000. Up 19.52 seconds.
[ 20.081263] cloud-init[970]: Cloud-init v. 23.3.3-0ubuntu0~22.04.1 running 'modules:final' at Fri, 29 Mar 2024 07:54:16 +0000. Up 20.02 seconds.
[ 20.151548] cloud-init[970]: Cloud-init v. 23.3.3-0ubuntu0~22.04.1 finished at Fri, 29 Mar 2024 07:54:16 +0000. Datasource DataSourceNone. Up 20.14 seconds
[ 20.151989] cloud-init[970]: 2024-03-29 07:54:16,666 - cc_final_message.py[WARNING]: Used fallback datasource
b12902110
Password:
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-101-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Fri Mar 29 07:54:54 AM UTC 2024

System load:        1.208984975
Usage of /:         29.8% of 9.75GB
Memory usage:       3%
Swap usage:         0%
Processes:          185
Users logged in:    0
IPv4 address for enp1s0: 10.0.2.15
IPv6 address for enp1s0: fec0::5054:ff:fe90:2110

Expanded Security Maintenance for Applications is not enabled.

11 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Fri Mar 29 07:51:12 UTC 2024 on tty1
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

b12902110@nasa-hw5:~$ _
```

ip a:

```
b12902110@nasa-hw5:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 52:54:00:90:21:10 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 metric 100 brd 10.0.2.255 scope global dynamic enp1s0
        valid_lft 86338sec preferred_lft 86338sec
    inet6 fec0::5054:ff:fe90:2110/64 scope site dynamic mngtmpaddr noprefixroute
        valid_lft 86340sec preferred_lft 14340sec
    inet6 fe80::5054:ff:fe90:2110/64 scope link
        valid_lft forever preferred_lft forever
b12902110@nasa-hw5:~$
```

2. Command (on VM)

```
sudo systemctl enable --now serial-getty@ttyS0.service
sudo systemctl enable --now ssh.service
```

Command (on host)

```
virsh qemu-monitor-command b12902110 --hmp 'hostfwd_add ::11022-:22'
```

Result

```
b12902110@ws1: /tmp2/b12902110/nasahw5
$ virsh console b12902110
Connected to domain 'b12902110'
Escape character is ^] (Ctrl + J)

Ubuntu 22.04.4 LTS nasa-hw5 ttyS0

nasa-hw5 login: b12902110
Password:
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-101-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

System information as of Fri Mar 29 09:02:50 AM UTC 2024

System load:          0.6572265625
Usage of /:            30.0% of 9.75GB
Memory usage:         3%
Swap usage:           0%
Processes:            187
Users logged in:      0
IPv4 address for enp1s0: 10.0.2.15
IPv6 address for enp1s0: fec0::5054:ff:fe90:2111

Expanded Security Maintenance for Applications is not enabled.

11 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Fri Mar 29 08:32:42 UTC 2024 on ttyS0
b12902110@nasa-hw5:~$

b12902110@ws1: /tmp2/b12902110/nasahw5
$ virsh qemu-monitor-command b12902110 --hmp 'hostfwd_add ::11022-:22'

b12902110@ws1: /tmp2/b12902110/nasahw5
$ ssh -p 11022 localhost
The authenticity of host '[localhost]:11022 ([127.0.0.1]:11022)' can't be established.
ED25519 key fingerprint is SHA256:PTwkjHy9TchdrrrSGMnDxwJ0BmifD2GJ+Et1+PAtYW0.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[localhost]:11022' (ED25519) to the list of known hosts.
b12902110@localhost's password:
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-101-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

System information as of Fri Mar 29 09:16:16 AM UTC 2024

System load:          0.4453125
Usage of /:            30.3% of 9.75GB
Memory usage:         3%
Swap usage:           0%
Processes:            188
Users logged in:      0
IPv4 address for enp1s0: 10.0.2.15
IPv6 address for enp1s0: fec0::5054:ff:fe90:2110

Expanded Security Maintenance for Applications is not enabled.

11 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Fri Mar 29 09:16:16 2024 from 10.0.2.2
b12902110@nasa-hw5:~$
```

References

- [man virt-install, man virsh](#)
- [libvirt: Network XML format](#)
- [libvirt: Creating a NAT Virtual Network](#)
- [libvirt: NAT forwarding \(aka "virtual networks"\)](#)
- [libvirt: Domain XML format](#)
- [qemu - How to port forward SSH in virt-manager? - Unix & Linux Stack Exchange](#)
- [Invocation —QEMU documentation](#)
- [host port forward with qemu through libvirt in user-mode networking - Server Fault](#)
- [Structured Procrastination port redirection from kvm host to guest -Structured Procrastination](#)

Docker

Set Up

3. Command

```
# Add Docker's official GPG key:
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg \
  -o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc

# Add the repository to Apt sources:
# Add the repository to Apt sources:
echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] \
  https://download.docker.com/linux/ubuntu \
  $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update

# Install the Docker packages.
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin \
  docker-compose-plugin docker-compose
```

Result

```
b12902110@nasa-hw5:~$ sudo docker version
Client: Docker Engine - Community
Version: 26.0.0
API version: 1.45
Go version: go1.21.8
Git commit: 2ae903e
Built: Wed Mar 20 15:17:48 2024
OS/Arch: linux/amd64
Context: default

Server: Docker Engine - Community
Engine:
Version: 26.0.0
API version: 1.45 (minimum version 1.24)
Go version: go1.21.8
Git commit: 8b79278
Built: Wed Mar 20 15:17:48 2024
OS/Arch: linux/amd64
Experimental: false
containerd:
Version: 1.6.28
GitCommit: ae07eda36dd25f8a1b98dfbf587313b99c0190bb
runc:
Version: 1.1.12
GitCommit: v1.1.12-0-g51d5e94
docker-init:
Version: 0.19.0
GitCommit: de40ad0
b12902110@nasa-hw5:~$ sudo docker-compose version
docker-compose version 1.29.2, build unknown
docker-py version: 5.0.3
CPython version: 3.10.12
OpenSSL version: OpenSSL 3.0.2 15 Mar 2022
b12902110@nasa-hw5:~$
```

Docker Basics

4. Cases to use Docker:

- Host OS is Linux.
- Application runs on Linux.
- When performance and efficiency is important, because Docker shares resources with the host.
- When we want to deploy the environment elsewhere, Docker offers better portability.

Cases to use VM:

- Host OS is not Linux.
- Application doesn't run on Linux.
- Need control over hardware resources (CPU, memory, etc.).

References

- [Docker vs VM - Difference Between Application Deployment Technologies - AWS](#)
- [Reddit - Dive into anything](#)

5. The Docker Engine runs on a Linux kernel. On Windows and macOS, Docker is run in a virtual machine (WSL2 or LinuxKit VM), which hurts performance.

References

- [Instantly Improve Docker Performance on Mac](#)
- [Docker on MacOS is slow and how to fix it | CNCF](#)
- [Install Docker Desktop on Windows | Docker Docs](#)

6. (a) `docker stop $(docker ps -a -q)`
“`docker ps -a -q`” list IDs of all containers. Substitute it into argument of “`docker stop`”.

- (b) `docker image rm $(docker image -a -q)`
“`docker image -a -q`” list IDs of all images. Substitute it into argument of “`docker image rm`”.

- (c) `docker system prune`
Remove all unused containers, networks, images (both dangling and unused), and optionally, volumes.

- (d)
- ```
docker inspect \
 --format='{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}' \
 5b0f1ed0dcb8
```

Return low-level information on Docker objects.

- (e) `docker container stats`  
Display a live stream of container(s) resource usage statistics.

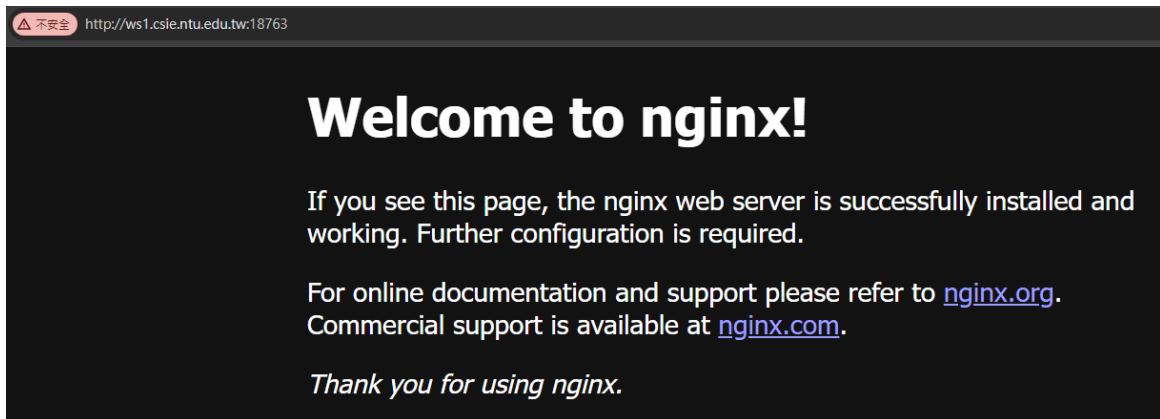
## References

- [Stop and remove all docker containers - Stack Overflow](#)
- [docker system prune | Docker Docs](#)
- [How to get a Docker container's IP address from the host - Stack Overflow](#)
- [docker inspect | Docker Docs](#)
- [docker container stats | Docker Docs](#)

## 7. Command `docker run --name nginx-1 -d -p 8763:80 nginx:1.24.0`

- `docker run`: Create and run a new container from an image.
- `--name nginx-1`: Assign name `nginx-1` to container.
- `-d`: Run in background.
- `-p 8763 80`: Forward port 8763 on host to port 80 in container.
- `nginx:1.24.0`: Image to run.

### Result



Note: Port 18763 on ws1 is forwarded to port 8763 on Ubuntu VM.

### References

- `docker run --help`

## 8. Command `docker exec -it nginx-1 /bin/bash`

- `docker exec`: Execute a command in a running container.
- `-i`: Keep STDIN open even if not attached.
- `-t`: Allocate a pseudo-TTY.
- `nginx-1`: Container name.
- `/bin/bash`: Command to execute.

### Result

```
b12902110@nasa-hw5:~/nasa-hw5/b12902110$ docker exec -it nginx-1 /bin/bash
root@4284f880e2a8:/# |
```

### References

- `docker exec --help`

9. **Command** `docker exec nginx-1 cat /etc/nginx/nginx.conf`

**Result**

```
b12902110@nasa-hw5:~/nasa-hw5/b12902110$ docker exec nginx-1 cat /etc/nginx/nginx.conf

user nginx;
worker_processes auto;

error_log /var/log/nginx/error.log notice;
pid /var/run/nginx.pid;

events {
 worker_connections 1024;
}

http {
 include /etc/nginx/mime.types;
 default_type application/octet-stream;

 log_format main '$remote_addr - $remote_user [$time_local] "$request" '
 '$status $body_bytes_sent "$http_referer" '
 '"$http_user_agent" "$http_x_forwarded_for"';

 access_log /var/log/nginx/access.log main;

 sendfile on;
 #tcp_nopush on;

 keepalive_timeout 65;

 #gzip on;

 include /etc/nginx/conf.d/*.conf;
}
```

10. (a) **ENTRYPOINT** defines the executable that the container runs, **CMD** defines default arguments for **ENTRYPOINT**.
- (b) **ENTRYPOINT** is overridden with `docker run --entrypoint NEW_ENTRY_POINT ....`  
**CMD** is overridden with `docker run ... NEW_COMMAND`.
- (c) If using **exec** form, **CMD** is the arguments for **ENTRYPOINT**.

**References**

- [Dockerfile reference | Docker Docs](#)

11. Docker Compose is a tool to manage *multi-container* applications. We define services, network, volumes to be used in an YAML configuration file. On the other hand, Docker is the underlying virtualization engine that handles containers, images, networks, etc.

**References**

- [Docker Engine overview | Docker Docs](#)
- [Docker Compose overview | Docker Docs](#)

12. (a) **Command** `docker-compose up -d`
- **docker-compose up**: Builds, (re)creates, starts, and attaches to containers for a service.
  - **-d**: Detached mode: Run containers in the background.
- (b) **Command** `docker-compose pause`  
Pause services.
- (c) **Command** `docker-compose down -v`
- **docker-compose down**: Stops containers and removes containers, networks, volumes, and images created by 'up'.





## 14. mysql-b12902110.yaml

```
services:
 db:
 image: mysql:latest
 networks:
 - nasa-net
 environment:
 - MYSQL_HOST=b12902110
 - MYSQL_ROOT_PASSWORD=secret

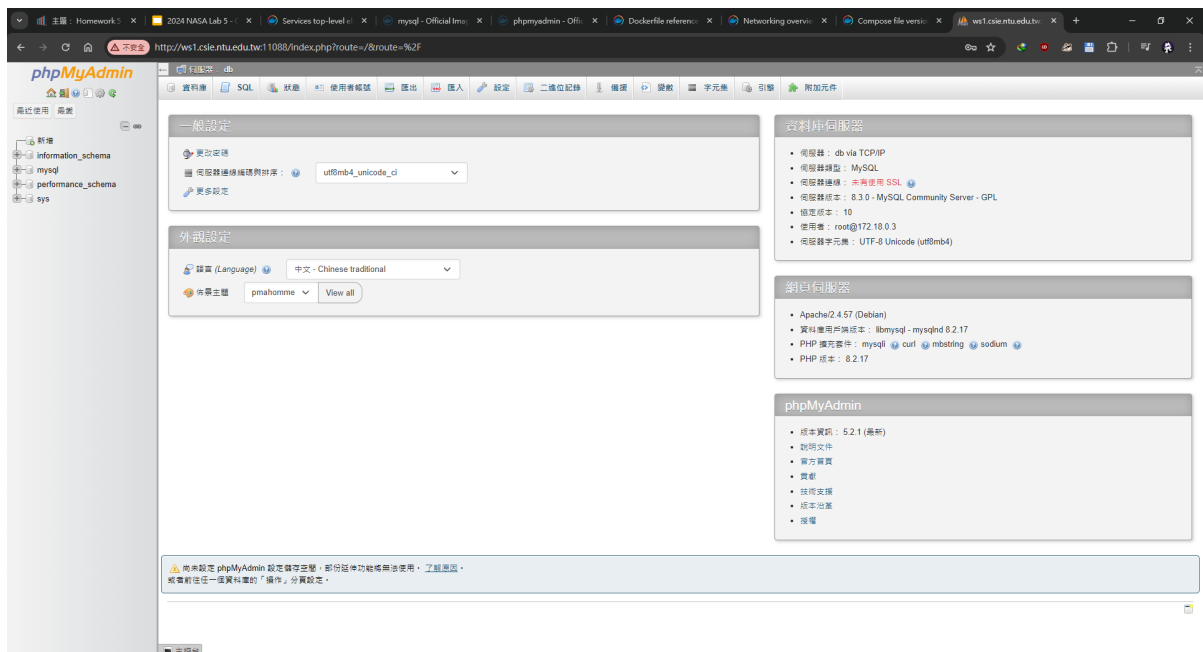
 web:
 image: phpmyadmin:latest
 depends_on:
 - db
 ports:
 - 8888:80
 networks:
 - nasa-net

networks:
 nasa-net:
```

## Command

```
sudo docker-compose -f mysql-b12902110.yaml up
```

## Result



## References

- [Try Docker Compose | Docker Docs](#)
- [Networking in Compose | Docker Docs](#)
- [mysql - Official Image | Docker Hub](#)
- [phpmyadmin - Official Image | Docker Hub](#)

- [MySQL :: MySQL 5.7 Reference Manual :: 4.9 Environment Variables](#)
- [Ways to set environment variables with Compose | Docker Docs](#)

## Kubernetes

15.

16. (a) **Steps**

- Reuse the VM installed in step 1.
- Run the following commands:

```
curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64
sudo install minikube-linux-amd64 /usr/local/bin/minikube && rm minikube-linux-amd64
alias kubectl="minikube kubectl --"
kubectl
```

(b) **nginx-b12902110.yaml**

```
apiVersion: apps/v1
kind: Deployment
metadata:
 name: nginx-b12902110
spec:
 replicas: 1
 selector:
 matchLabels:
 app: nginx-b12902110
 template:
 metadata:
 labels:
 app: nginx-b12902110
 spec:
 containers:
 - name: nginx
 image: nginx:latest
 ports:
 - containerPort: 8888
```

**mysql-b12902110.yaml**

```
apiVersion: apps/v1
kind: Deployment
metadata:
 name: mysql-b12902110
spec:
 replicas: 1
 selector:
 matchLabels:
 app: mysql-b12902110
 template:
 metadata:
 labels:
 app: mysql-b12902110
 spec:
 containers:
 - name: mysql
 image: mysql:latest
 env:
```

```
- name: MYSQL_HOST
 value: b12902110
- name: MYSQL_ROOT_PASSWORD
 value: $(MYSQL_ROOT_PASSWORD)
```

### phpmyadmin-b12902110.yaml

```
apiVersion: apps/v1
kind: Deployment
metadata:
 name: phpmyadmin-b12902110
spec:
 replicas: 1
 selector:
 matchLabels:
 app: phpmyadmin-b12902110
 template:
 metadata:
 labels:
 app: phpmyadmin-b12902110
 spec:
 containers:
 - name: phpmyadmin
 image: phpmyadmin:latest
```

(c)