

ปฏิบัติการที่ 5 การสร้าง Container

5.1 การติดตั้ง Docker Engine

Docker Engine มีทั้งบน Windows/Mac/Linux

รู้จัก Docker file จะเป็นแอป nodeJS ซึ่งมีรายละเอียด

- node:4.6 เวอร์ชัน
- ตำแหน่งคือ ./app
- ให้รัน npm install เป็นการติดตั้งส่วนที่เกี่ยวข้อง
- expose คือพอร์ต 3000
- CMD เป็นการสั่งให้เปิด
- index.js เป็นไฟล์ที่อ่านเมื่อเปิดเครื่อง ซึ่งมีโค้ดให้แสดง hello world ในพอร์ตที่ระบุ และล็อก
- package.json มีชื่อ เวอร์ชัน ระบุเป็นส่วนตัว และระบุสคริปต์ที่ทำงาน ระบุเวอร์ชันที่ต้องการรัน

การสร้างโครงการจะสามารถสร้าง docker build หรือสร้างด้วยมือ หรือซอฟต์แวร์อื่นๆ เช่น jenkins

1.1.1 การติดตั้ง docker บน Ubuntu

1. เปิด Terminal บน Linux: Ubuntu พิมพ์ว่า

sudo apt-get install docker.io

ติดตั้ง docker ซึ่งจะมี ubuntu เพื่อติดตั้ง

2. ตรวจสอบค่าติดตั้งด้วยคำสั่ง

```
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /lib/systemd/system/docker.socket.  
docker.service is a disabled or a static unit, not starting it.  
Processing triggers for systemd (237-3ubuntu10.41) ...  
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...  
Processing triggers for ureadahead (0.100.0-21) ...  
vagrant@ubuntu-bionic:~$ docker --version  
Docker version 19.03.6, build 369ce74a3c  
vagrant@ubuntu-bionic:~$
```

docker -version

ดูเวอร์ชัน

3. การติดตั้งโปรแกรม git พิมพ์ว่า

sudo apt-get install git

4. ดำเนินการโคลน docker-demo มาเพื่อใช้งาน

git clone <https://github.com/khajorn/docker-demo>

5. เข้าดูไฟล์ที่ตั้งมา

```
cd docker-demo
```

เข้าไปดูว่ามีอะไรบ้างใน dockerfile และดูไฟล์อื่นๆใช้คำสั่ง cat เพื่อให้ดูว่า docker สร้างอะไรบ้าง

6. นำชื่อผู้ใช้ Ubuntu และนำมาใส่กลุ่ม docker

```
sudo usermod -G docker Ubuntu
```

7. ออกจากระบบ แล้วเข้าใหม่เพื่อให้อยู่ในกลุ่ม docker

```
logout
```

8. เข้ามาใน Secure Shell ใหม่

```
vagrant ssh
```

9. เข้าไปในตำแหน่ง docker-demo

```
cd docker-demo/
```

10. ทำการสร้าง docker ด้วยคำสั่ง

```
sudo docker build .
```

```

vagrant@ubuntu-bionic:~/docker-demo$ sudo docker build .
Sending build context to Docker daemon 212.5kB
Step 1/6 : FROM node:12
12: Pulling from library/node
81fc19181915: Pull complete
ee49ee6a23d1: Pull complete
828510924538: Pull complete
a8f58c4fcca0: Pull complete
33699d7df21e: Pull complete
923705ffa8f8: Pull complete
5fa4fa08e10b: Pull complete
7f3e893e40e2: Pull complete
1b42cd4fbc0: Pull complete
Digest: sha256:2b85f4981f92ee034b51a3c8bb22dbb451d650d5c12b6439a169f8adc750e4b6
Status: Downloaded newer image for node:12
---> f5be1883c8e0
Step 2/6 : WORKDIR /app
---> Running in dba09ec696a7
Removing intermediate container dba09ec696a7
---> 45ac3389da76
Step 3/6 : ADD . /app
---> 1231076e3205
Step 4/6 : RUN npm install
---> Running in 52bdf3aa0abc
added 94 packages from 485 contributors and audited 95 packages in 4.763s
found 1 low severity vulnerability
  run `npm audit fix` to fix them, or `npm audit` for details
Removing intermediate container 52bdf3aa0abc
---> 5fd8af874914
Step 5/6 : EXPOSE 3000
---> Running in 07905616f971
Removing intermediate container 07905616f971
---> ccb096b0e7cd
Step 6/6 : CMD npm start
---> Running in b973549264fe
Removing intermediate container b973549264fe
---> 913db8908edf
Successfully built 913db8908edf
vagrant@ubuntu-bionic:~/docker-demo$

```

พบว่าระบบสามารถสร้างจนเสร็จอธิบายในแต่ละรายละเอียด

ดูรายละเอียดสิ่งที่เกี่ยวข้องจดค่าหลัง Successfully built

11. พิมพ์คำสั่ง

docker run -p 3000:3000 -it 913db8908edf

```

vagrant@ubuntu-bionic:~/docker-demo$ sudo docker run -p 3000:3000 -it 913db8908edf
f

> myapp@0.0.1 start /app
> node index.js

Example app listening at http://:::3000

```

12. เปิด Terminal อีกหน้าจอพิมพ์ว่า

vagrant ssh

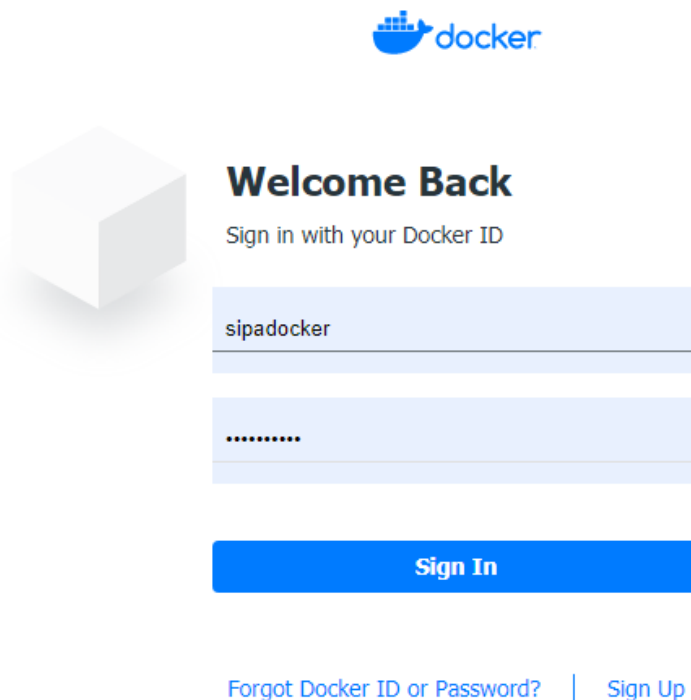
ต้องอยู่ในตำแหน่ง Vagrantfile ที่ถูกต้อง

curl localhost 3000

```
Last login: Fri Jun 19 07:39:07 2020 from 10.0.2.2
vagrant@ubuntu-bionic:~$ curl http://localhost:3000
Hello World!vagrant@ubuntu-bionic:~$
```

5.2 การบริหาร Docker registry

การบริหารงาน Docker hub จะต้องไปดำเนินการสร้างค่า repository ไว้ใน Docker Hub และสมัครบัญชีรายชื่อไว้ก่อน



คำสั่งที่ใช้ docker run เพื่อรันในระบบทดสอบ

ถ้าต้องการส่งอิมเมจไปใช้ Docker Registry เหมือน Docker Hub

5.2.1 ขั้นตอนล็อกอิน Docker Hub

1. เปิด Terminal
2. พิมพ์ว่า

sudo docker login


```
vagrant@ubuntu-bionic:~$ sudo docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't
have a Docker ID, head over to https://hub.docker.com to create one.
Username: sipadocker
Password:
WARNING! Your password will be stored unencrypted in /home/vagrant/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
vagrant@ubuntu-bionic:~$
```

โดยต้องไปลงทะเบียนก่อน

3. สลับเข้าไปที่ hub.docker.com แล้วไปสร้าง docker-demo ไว้

Thank you for attending DockerCon 2020! [Watch the recordings.](#)

 dockerhub [Explore](#) [Repositories](#)



[Repositories](#) [Create](#)


Create Repository

Docker Demonstration

Visibility

Using 0 of 1 private repositories. [Get more](#)

☒ **Public**   Public repositories appear in Docker Hub search results

☐ **Private**  Only you can view private repositories

[Build Settings \(optional\)](#)

4. คลิกคำสั่ง Create
5. การเข้าไปใช้อิมเมจใน Docker Hub
sudo docker tag 913db8908edf sipadocker/k8s-demo
6. ตรวจสอบค่าอิมเมจ
sudo docker images

```

vagrant@ubuntu-bionic:~$ sudo docker tag 913db8908edf sipadocker/k8s-demo
vagrant@ubuntu-bionic:~$ sudo docker images

```

REPOSITORY	TAG	IMAGE ID	CREATED
sipadocker/k8s-demo	latest	913db8908edf	50 minutes ago
node	12	f5be1883c8e0	12 hours ago

```

vagrant@ubuntu-bionic:~$

```

7. การส่งไฟล์ขึ้น

sudo docker push sipadocker/k8s-demo


```

vagrant@ubuntu-bionic:~$ sudo docker push sipadocker/k8s-demo
The push refers to repository [docker.io/sipadocker/k8s-demo]
e7626d00c271: Pushed
2711c6b8fb7d: Pushed
05f1a0a22041: Pushed
dc48ece44f3c: Mounted from library/node
798326960eac: Mounted from library/node
dacaab4534e4: Mounted from library/node
bc17cd405095: Mounted from library/node
ee854067fbbd: Mounted from library/node
740ffea5d5c3: Mounted from library/node
eac9ead92b24: Mounted from library/node
23bca356262f: Mounted from library/node
8354d5896557: Mounted from library/node

```



8. ตรวจสอบใน Docker Hub

Thank you for attending DockerCon 2020! [Watch the recordings.](#)


 **dockerhub** [Explore](#) [Repositories](#)

Repositories > sipadocker / k8s-demo

[General](#) [Tags](#) [Builds](#) [Timeline](#) [Collaborators](#) [Webhooks](#) [Settings](#)

 **sipadocker / k8s-demo**
Kubernetes for Docker 
⌚ Last pushed: in 17 hours

Tags
This repository contains 1 tag(s).

latest		⌚ in 17 hours
--------	---	---------------

[See all](#)

Docker comma
To push a new tag
[docker push sipa](#)

Recent builds
[Link a source prov](#)

9. ถ้าต้องการตรวจสอบ tag ล่าสุดให้พิมพ์ว่า

```
sudo docker tag 913db8908edf sipadocker/k8s-demo:latest
```

ปฏิบัติกา