

UnetLab교육자료		
Dockerfile작성하여 이미지만들기		
문서번호: 20170205-01	버전: 1.0	Date: 2017/02/05

본 문서는 향후 UnetLab에서 사용할 리눅스 클라이언트와 서버머신들에 대한 이미지를 직접 만들어서 Lab에서 자유롭게 사용하려는 목적으로 만들어진 문서입니다. 문서에서 설명하는 서버머신들에 대한 Dockerfile은 실제 우분투 시스템에서 해당 서버기능을 설치할 때 동일한 방법으로 설정을 하는것이므로 Dockerfile에 나열하는 설치순서에 대하여 이해를 하려는 노력이 필요합니다. 해당 내용의 Dockerfile을 작성하여 직접 서버머신에 대한 도커 이미지를 만들어보세요. 앞으로 Lab에 서버머신에 대한 노드가 나타나면 자신들이 만든 서버이미지를 선택하면 됩니다.

## 1. Dockerfile작성과 빌드방법

- Dockerfile은 Docker이미지 설정파일로 설정된 내용대로 이미지를 생성한다.
- Dockerfile을 작성하여 이미지를 생성할때는 작업할 빈 디렉토리를 생성하고 그 디렉토리 안에서 Dockerfile을 작성하여 이미지를 생성해야한다. Dockerfile이 위치한 디렉토리에 다른 파일이 존재하는 경우에 해당파일들도 이미지에 넣는 작업이 수행되기 때문에 이미지생성에 문제가 발생한다.
- 파일이름이 반드시 Dockerfile으로 되어야 한다.
- Dockerfile은 <명령> <매개변수> 형식으로 작성되고 #은 주석이다. 명령은 대소문자를 구분하지 않지만 보통 대문자로 작성한다.

```
# 주석
FROM scratch
```

- 명령은 항상 FROM으로부터 시작해야한다. FROM이 없거나 앞에 다른 명령이 있는 경우에 이미지가 만들어지지 않는다.
- 이미지를 생성할 때는 Dockerfile이 있는 디렉토리에서 docker build명령을 사용한다. --tag(-t)옵션으로 이미지 이름을 설정할 수 있다. 이미지 이름앞에 사용자이름을 추가할 수도 있다.

```
# docker -H=tcp://127.0.0.1:4243 build --no-cache --tag openssh_client .
# docker -H=tcp://127.0.0.1:4243 build --no-cache --tag sghan/openssh_client .
```

- 이미지를 build할 때에 -no-cache옵션을 주어서 불필요한 파일이 남지 않도록 한다.

## 2. 우분투 클라이언트 만들기

- 우분투 클라이언트는 LAB에서 리눅스시스템으로 사용하기 위해서 만드는 이미지이다.
- 우분투 기본 이미지로 컨테이너를 생성하는 경우에 기본적으로 사용해야하는 명령어(ifconfig, ping, telnet, ssh등등)들이 존재하지 않기 때문에 패키지를 설치해주어야 한다. 하지만, LAB이 인터넷 연결구성을 하지 않은 상태이면 클라이언트에 패키지를 설치하지 못한다.
- 아래와 같이 우리가 우분투 리눅스 클라이언트에서 주로 사용해야할 명령들이 포함된 이미지를 만들어서 LAB에서 리눅스시스템으로 자유롭게 사용한다.

```
root@unl01:/data/docker-build# mkdir openssh_client
root@unl01:/data/docker-build# cd openssh_client/
root@unl01:/data/docker-build/openssh_client# vi Dockerfile
# 우분투 최신버전을 기본 이미지로 사용
FROM ubuntu
```

UnetLab교육자료 Dockerfile작성하여 이미지만들기		
문서번호: 20170205-01	버전: 1.0	Date: 2017/02/05

```
# 이미지 작성자에 대한 기본정보
MAINTAINER Han SangGu <sgghan@hinetworks.kr>

# 우분투 저장소를 국내미러사이트를 사용하도록 변경
RUN cd /etc/apt && \
    sed -i 's/archive.ubuntu.com/ftp.daum.net/g' sources.list

# 우분투 패키지 저장소로부터 패키지목록을 받아오고,
# 일반적으로 사용할 명령어를 포함하는 패키지를 설치하고,
# 다운로드했던 패키지파일을 삭제하여 공간 확보
RUN apt-get update && \
    apt-get install -y net-tools iputils-ping openssh-client ftp telnet dnsutils traceroute expect mysql-client
    openssl wget
    lynx vim lsof && \
    apt-get clean

# 해당 이미지를 컨테이너로 만들어서 실행했을때 마지막으로 구동되는 명령.
# 리눅스시스템에 사용자가 로그인하면 bash 이 구동되는것과 마찬가지로이다.
CMD ["/bin/bash"]

root@unl01:/data/docker-build/openssh_client# ls -al
total 12
drwxr-xr-x 2 root root 4096 Feb  5 15:03 .
drwxr-xr-x 3 root root 4096 Feb  5 14:57 ..
-rw-r--r-- 1 root root  789 Feb  5 15:03 Dockerfile

root@unl01:/data/docker-build/openssh_client# docker -H=tcp://127.0.0.1:4243 build --no-cache --tag
ubuntu_client .

Sending build context to Docker daemon  2.56 kB

Step 1 : FROM ubuntu
latest: Pulling from library/ubuntu
8aec416115fd: Pull complete
695f074e24e3: Pull complete
946d6c48c2a7: Pull complete
bc7277e579f0: Pull complete
2508cbcd94b: Pull complete
Digest: sha256:71cd81252a3563a03ad8daee81047b62ab5d892ebbf71cf53415f29c130950
Status: Downloaded newer image for ubuntu:latest
```

## UnetLab교육자료

### Dockerfile작성하여 이미지만들기

문서번호: 20170205-01

버전: 1.0

Date: 2017/02/05

```
---> f49eec89601e
Step 2 : MAINTAINER Han SangGu <sghan@hinetworks.kr>
---> Running in 20c742a6322a
---> 9f87654c4d94
Removing intermediate container 20c742a6322a
Step 3 : RUN cd /etc/apt && sed -i 's/archive.ubuntu.com/ftp.daum.net/g' sources.list
---> Running in 59affaa2ec60
---> e501d9540c4a
Removing intermediate container 59affaa2ec60
Step 4 : RUN apt-get update && apt-get install -y net-tools iputils-ping openssh-client ftp telnet dnsutils
tracertoute expect mysql-client openssl wget lynx vim lsof && apt-get clean
---> Running in ec9793c6b7b3
Get:1 http://ftp.daumkakao.com/ubuntu xenial InRelease [247 kB]
Get:2 http://ftp.daumkakao.com/ubuntu xenial-updates InRelease [102 kB]
Get:3 http://ftp.daumkakao.com/ubuntu xenial-security InRelease [102 kB]
...(중략)...
173 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
Processing triggers for sgml-base (1.26+nmu4ubuntu1) ...
---> 527da55bd7f9
Removing intermediate container 0a666483c511
Step 5 : CMD /bin/bash
---> Running in 8a28100d2d8d
---> 762dff72d5e9
Removing intermediate container 8a28100d2d8d
Successfully built 762dff72d5e9
root@unl01:/data/docker-build/openssh_client# docker -H=tcp://127.0.0.1:4243 images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
ubuntu_client        latest              becdf2f9f65e       About a minute ago 435.2 MB
ubuntu               latest             f49eec89601e       2 weeks ago       129.5 MB
```

### 3. 우분투 SSH서버 만들기

- 우분투 SSH서버는 LAB상의 클라이언트에서 ssh와 sftp로 연결테스트를 수행할 대상서버이다.
- 기본 명령어가 포함되도록 구성한 ubuntu\_client이미지를 기본이미지로 사용하여 openssh-server패키지를 추가로 설치하여 서버를 구동한다.
- 서버연결시험을 위하여 기본계정(아이디:admin, 암호:unl)을 생성한다.

## □ 우분투 SSH서버의 통신포트를 설정한다. (EXPOSE 22)

```

root@unl01:/data/docker-build# mkdir openssh_server
root@unl01:/data/docker-build# cd openssh_server/
root@unl01:/data/docker-build/openssh_server# vi Dockerfile
# 우리가 만든 우분투 클라이언트 이미지를 기본이미지로 사용
FROM ubuntu_client

# 이미지 작성자에 대한 기본정보
MAINTAINER Han SangGu <sghan@hinetworks.kr>

# 우분투 저장소를 국내미러사이트를 사용하도록 변경
RUN cd /etc/apt && \
    sed -i 's/archive.ubuntu.com/ftp.daum.net/g' sources.list

# 우분투 패키지 저장소로부터 패키지목록을 받아오고,
# openssh-server 패키지를 설치하고,
# 다운로드했던 패키지파일을 삭제하여 공간확보
RUN apt-get update && \
    apt-get install -y openssh-server && \
    apt-get clean

# sshd 에 대한 PID 를 생성할 디렉토리를 만들어준다
RUN mkdir /var/run/sshd

# 테스트할 계정을 생성하고 암호설정
RUN groupadd -r admin && useradd -r -g admin admin && \
    echo "admin:unl" | chpasswd

# 해당 이미지를 컨테이너로 만들어서 실행했을때 마지막으로 구동되는 명령.
CMD ["/usr/sbin/sshd", "-D"]

# 통신포트를 정의
EXPOSE 22

root@unl01:/data/docker-build/openssh_server# ls -al
total 12
drwxr-xr-x 2 root root 4096 Feb  5 15:36 .

```

## UnetLab교육자료

Dockerfile작성하여 이미지만들기

문서번호: 20170205-01

버전: 1.0

Date: 2017/02/05

```
drwxr-xr-x 4 root root 4096 Feb  5 15:29 ..
-rw-r--r-- 1 root root 1022 Feb  5 15:36 Dockerfile

root@unl01:/data/docker-build/openssh_server# docker -H=tcp://127.0.0.1:4243 build --no-cache --tag
openssh_server .

Sending build context to Docker daemon  2.56 kB

Step 1 : FROM ubuntu_client
----> becdf2f9f65e

Step 2 : MAINTAINER Han SangGu <sghan@hinetworks.kr>
----> Running in 8d6df041ece8
----> 9a435d5bc848

Removing intermediate container 8d6df041ece8

Step 3 : RUN cd /etc/apt &&      sed -i 's/archive.ubuntu.com/ftp.daum.net/g' sources.list
----> Running in d559e69e3de0
----> b205ce3edb02

Removing intermediate container d559e69e3de0

Step 4 : RUN apt-get update &&      apt-get install -y openssh-server &&      apt-get clean
----> Running in c41ca17555df

Hit:1 http://ftp.daumkakao.com/ubuntu xenial InRelease
Hit:2 http://ftp.daumkakao.com/ubuntu xenial-updates InRelease
...(중략)...

Processing triggers for systemd (229-4ubuntu13) ...
----> e5144d0af9e2

Removing intermediate container c41ca17555df

Step 5 : RUN mkdir /var/run/sshd
----> Running in e400e27bb21d
----> ccff4043cc47

Removing intermediate container e400e27bb21d

Step 6 : RUN groupadd -r admin && useradd -r -g admin admin &&      echo "admin:unl" | chpasswd
----> Running in 842d37024244
----> 0bd2e5da6ba0

Removing intermediate container 842d37024244

Step 7 : CMD /usr/sbin/sshd -D
----> Running in ed9acd4ce5ad
----> b78a5c9805ad

Removing intermediate container ed9acd4ce5ad

Step 8 : EXPOSE 22
```

```

---> Running in a2a6f40fd38a
---> 6c5f72758c56

Removing intermediate container a2a6f40fd38a
Successfully built 6c5f72758c56

root@unl01:/data/docker-build/openssh_server# docker -H=tcp://127.0.0.1:4243 images

REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
openssh_server       latest              6c5f72758c56       36 seconds ago     457.7 MB
ubuntu_client        latest              becdf2f9f65e       17 minutes ago     435.2 MB
ubuntu               latest              f49eec89601e       2 weeks ago        129.5 MB

```

#### 4. 우분투 exim4 메일서버 만들기

- ☐ 우분투 exim4 메일서버는 LAB상의 클라이언트에서 메일시험을 수행할 대상서버이다.
- ☐ 기본 명령어가 포함되도록 구성한 ubuntu\_client이미지를 기본이미지로 사용하여 exim4패키지를 추가로 설치하여 서버를 구동한다.
- ☐ 우분투 exim4 메일서버의 통신포트를 설정한다. (EXPOSE 25)

```

root@unl01:/data/docker-build# mkdir exim4_smtp
root@unl01:/data/docker-build# cd exim4_smtp/
root@unl01:/data/docker-build/exim4_smtp# vi Dockerfile
# 우리가 만든 우분투 클라이언트 이미지를 기본이미지로 사용
FROM ubuntu_client

# 이미지 작성자에 대한 기본정보
MAINTAINER Han SangGu <sgan@hinetworks.kr>

# 우분투 저장소를 국내미러사이트를 사용하도록 변경
RUN cd /etc/apt && \
    sed -i 's/archive.ubuntu.com/ftp.daum.net/g' sources.list

# 우분투 패키지 저장소로부터 패키지목록을 받아오고,
# exim4 패키지를 설치하고,
# 다운로드했던 패키지파일을 삭제하여 공간확보
RUN apt-get update && \
    apt-get install -y exim4 && \
    apt-get clean

```

```
# exim4 conf 파일을 작성한다.

RUN mv /etc/exim4/exim4.conf.template /etc/exim4/exim4.conf

# 해당 이미지를 컨테이너로 만들어서 실행했을때 마지막으로 구동되는 명령.
CMD /usr/sbin/exim4 -bd -q30m

# 통신포트를 정의
EXPOSE 25

root@unl01:/data/docker-build/exim4_smtp# ls -al
total 12
drwxr-xr-x 2 root root 4096 Feb  5 15:46 .
drwxr-xr-x 5 root root 4096 Feb  5 15:43 ..
-rw-r--r-- 1 root root  881 Feb  5 15:46 Dockerfile
root@unl01:/data/docker-build/exim4_smtp# docker -H=tcp://127.0.0.1:4243 build --no-cache --tag exim4_smtp .
Sending build context to Docker daemon  2.56 kB
Step 1 : FROM ubuntu_client
----> becdf2f9f65e
Step 2 : MAINTAINER Han SangGu <sghan@hinetworks.kr>
----> Running in 7cbedb62682d
----> 4639d04c6df2
Removing intermediate container 7cbedb62682d
Step 3 : RUN cd /etc/apt &&      sed -i 's/archive.ubuntu.com/ftp.daum.net/g' sources.list
----> Running in fb4967f5efcf
----> e85ac0333875
Removing intermediate container fb4967f5efcf
Step 4 : RUN apt-get update &&      apt-get install -y exim4 &&      apt-get clean
----> Running in 6e5a5c006379
Hit:1 http://ftp.daumkakao.com/ubuntu xenial InRelease
Hit:2 http://ftp.daumkakao.com/ubuntu xenial-updates InRelease
Hit:3 http://ftp.daumkakao.com/ubuntu xenial-security InRelease
...(중략)...
Processing triggers for systemd (229-4ubuntu13) ...
Processing triggers for libc-bin (2.23-0ubuntu5) ...
----> a78ae86d5345
Removing intermediate container 6e5a5c006379
Step 5 : RUN mv /etc/exim4/exim4.conf.template /etc/exim4/exim4.conf
```

```

---> Running in 2833b45a8712
---> a51c24b7a6f4
Removing intermediate container 2833b45a8712
Step 6 : CMD /usr/sbin/exim4 -bd -q30m
---> Running in f56e2f94f137
---> 5ac929cba0ac
Removing intermediate container f56e2f94f137
Step 7 : EXPOSE 25
---> Running in 7d8a81c1cfa4
---> b0a438400d39
Removing intermediate container 7d8a81c1cfa4
Successfully built b0a438400d39
root@unl01:/data/docker-build/exim4_smtp# docker -H=tcp://127.0.0.1:4243 images
REPOSITORY          TAG             IMAGE ID         CREATED          SIZE
exim4_smtp           latest          b0a438400d39    7 seconds ago   481.7 MB
openssh_server       latest          6c5f72758c56    11 minutes ago  457.7 MB
ubuntu_client        latest          becdf2f9f65e    28 minutes ago  435.2 MB
ubuntu               latest          f49eec89601e    2 weeks ago     129.5 MB

```

## 5. 우분투 apache2 웹서버 만들기

- ☐ 우분투 apache2 웹서버는 LAB상의 클라이언트에서 웹서버 연결테스트를 수행할 대상서버이다.
- ☐ 기본 명령어가 포함되도록 구성한 ubuntu\_client이미지를 기본이미지로 사용하여 apache2패키지를 추가로 설치하여 서버를 구동한다.
- ☐ 클라이언트에서 웹서버를 연결할때는 lynx 텍스트기반 브라우저를 사용한다.
- ☐ 우분투 apache2 웹서버의 통신포트를 설정한다. (EXPOSE 80)

```

root@unl01:/data/docker-build# mkdir apache2_web
root@unl01:/data/docker-build# cd apache2_web/
root@unl01:/data/docker-build/apache2_web# vi Dockerfile
# 우리가 만든 우분투 클라이언트 이미지를 기본이미지로 사용
FROM ubuntu_client

# 이미지 작성자에 대한 기본정보
MAINTAINER Han SangGu <sghan@hinetworks.kr>

# 우분투 저장소를 국내미러사이트를 사용하도록 변경

```



## UnetLab교육자료

### Dockerfile작성하여 이미지만들기

문서번호: 20170205-01

버전: 1.0

Date: 2017/02/05

```
RUN cd /etc/apt && \
    sed -i 's/archive.ubuntu.com/ftp.daum.net/g' sources.list

# 우분투 패키지 저장소로부터 패키지목록을 받아오고,
# apache2 패키지를 설치하고,
# 다운로드했던 패키지파일을 삭제하여 공간확보
RUN apt-get update && \
    apt-get install -y apache2 && \
    apt-get clean

# 해당 이미지를 컨테이너로 만들어서 실행했을때 마지막으로 구동되는 명령.
CMD ["/usr/sbin/apachectl","-DFOREGROUND"]

# 통신포트를 정의
EXPOSE 80

root@unl01:/data/docker-build/apache2_web# ls -al
total 12
drwxr-xr-x 2 root root 4096 Feb  5 15:57 .
drwxr-xr-x 6 root root 4096 Feb  5 15:55 ..
-rw-r--r-- 1 root root  791 Feb  5 15:57 Dockerfile

root@unl01:/data/docker-build/apache2_web# docker -H=tcp://127.0.0.1:4243 build --no-cache --tag apache2_web .
Sending build context to Docker daemon  2.56 kB
Step 1 : FROM ubuntu_client
---> becdf2f9f65e
Step 2 : MAINTAINER Han SangGu <sghan@hinetworks.kr>
---> Running in 7b274ca022c1
---> 306f585cf955
Removing intermediate container 7b274ca022c1
Step 3 : RUN cd /etc/apt &&    sed -i 's/archive.ubuntu.com/ftp.daum.net/g' sources.list
---> Running in 64fa88647dc9
---> b67f96c7558e
Removing intermediate container 64fa88647dc9
Step 4 : RUN apt-get update &&    apt-get install -y apache2 &&    apt-get clean
---> Running in 9dbe48022f82
Hit:1 http://ftp.daumkakao.com/ubuntu xenial InRelease
Hit:2 http://ftp.daumkakao.com/ubuntu xenial-updates InRelease
```

```

Hit:3 http://ftp.daumkakao.com/ubuntu xenial-security InRelease
Reading package lists...
...(중략)...
Processing triggers for libc-bin (2.23-0ubuntu5) ...
Processing triggers for systemd (229-4ubuntu13) ...
---> 9001ea12e74f
Removing intermediate container 9dbe48022f82
Step 5 : CMD /usr/sbin/apachectl -DFOREGROUND
---> Running in 971be3d8c8b5
---> d62ad93cc5fb
Removing intermediate container 971be3d8c8b5
Step 6 : EXPOSE 80
---> Running in 66b4d7536f10
---> f60fec959945
Removing intermediate container 66b4d7536f10
Successfully built f60fec959945
root@unl01:/data/docker-build/apache2_web# docker -H=tcp://127.0.0.1:4243 images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
apache2_web         latest             f60fec959945       About a minute ago 485 MB
exim4_smt           latest            b0a438400d39       10 minutes ago    481.7 MB
openssh_server      latest            6c5f72758c56       22 minutes ago    457.7 MB
ubuntu_client       latest            becdf2f9f65e       38 minutes ago    435.2 MB
ubuntu              latest            f49eec89601e       2 weeks ago       129.5 MB

```

## 6. 우분투 MySQL DB서버 만들기

- ☐ 우분투 MySQL DB서버는 LAB상의 클라이언트에서 mysql DB 연결테스트를 수행할 대상서버이다.
- ☐ 기본 명령어가 포함되도록 구성한 ubuntu\_client이미지를 기본이미지로 사용하여 mysql-server패키지를 추가로 설치하여 서버를 구동한다.
- ☐ mysql-server패키지 설치하는 과정에 데이터베이스의 root암호를 설정하는 부분이 있는데 암호(unl123)를 환경설정값에 넣어주어서 자동으로 설치가 진행되도록 한다.
- ☐ mysql config파일에서 모든 네트워크에서 데이터베이스를 접근할 수 있도록 설정파일을 변경한다.
- ☐ mysql이 실행될 때 PID와 데이터베이스 파일을 저장하는 디렉토리를 생성한다.
- ☐ 우분투 MySQL DB서버의 통신포트를 설정한다. (EXPOSE 3306)

```

root@unl01:/data/docker-build# mkdir mysqld_server
root@unl01:/data/docker-build# cd mysqld_server/
root@unl01:/data/docker-build/mysqld_server# vi Dockerfile
# 우리가 만든 우분투 클라이언트 이미지를 기본이미지로 사용

```

```

FROM ubuntu_client

# 이미지 작성자에 대한 기본정보
MAINTAINER Han SangGu <sgghan@hinetworks.kr>

# 우분투 저장소를 국내미러사이트를 사용하도록 변경
RUN cd /etc/apt && \
    sed -i 's/archive.ubuntu.com/ftp.daum.net/g' sources.list

# 패키지 설치관련 환경설정값에 mysql root 암호를 설정한다.
ENV DEBIAN_FRONTEND noninteractive
RUN echo "mysql-server mysql-server/root_password password un123" | debconf-set-selections
RUN echo "mysql-server mysql-server/root_password_again password un123" | debconf-set-selections

# 우분투 패키지 저장소로부터 패키지목록을 받아오고,
# mysql-server 패키지를 설치하고,
# 다운로드했던 패키지파일을 삭제하여 공간확보
RUN apt-get update && \
    apt-get install -y mysql-server && \
    apt-get clean

# config 파일에 모든 네트워크에서 연결가능하도록 수정
WORKDIR /etc/mysql/mysql.conf.d
RUN sed -i "s/127.0.0.1/0.0.0.0/g" mysqld.cnf

# PID 와 데이터베이스 파일이 저장될 디렉토리 생성
RUN mkdir /var/run/mysqld
RUN chown mysql:mysql /var/run/mysqld

# 해당 이미지를 컨테이너로 만들어서 실행했을때 마지막으로 구동되는 명령.
CMD ["/usr/sbin/mysqld"]

# 통신포트를 정의
EXPOSE 3306

```

```

root@unl01:/data/docker-build/mysqlld_server# ls -al
total 12
drwxr-xr-x 2 root root 4096 Feb  5 16:12 .
drwxr-xr-x 7 root root 4096 Feb  5 16:02 ..
-rw-r--r-- 1 root root 1375 Feb  5 16:12 Dockerfile
root@unl01:/data/docker-build/mysqlld_server# docker -H=tcp://127.0.0.1:4243 build --no-cache --tag mysqlld_server .
Sending build context to Docker daemon 3.072 kB
Step 1 : FROM ubuntu_client
----> becdf2f9f65e
Step 2 : MAINTAINER Han SangGu <sgshan@hinetworks.kr>
----> Running in a293327bcf56
----> d27f6763d896
Removing intermediate container a293327bcf56
Step 3 : RUN cd /etc/apt &&      sed -i 's/archive.ubuntu.com/ftp.daum.net/g' sources.list
----> Running in cd850e736d1c
----> 5d9832a78cb3
Removing intermediate container cd850e736d1c
Step 4 : ENV DEBIAN_FRONTEND noninteractive
----> Running in 751359a0ea94
----> d9f0f91a02a3
Removing intermediate container 751359a0ea94
Step 5 : RUN echo "mysql-server mysql-server/root_password password unl123" | debconf-set-selections
----> Running in e674dd5fa681
----> 7c87109965ad
Removing intermediate container e674dd5fa681
Step 6 : RUN echo "mysql-server mysql-server/root_password_again password unl123" | debconf-set-selections
----> Running in c912d6ae832e
----> 3ebc86e222db
Removing intermediate container c912d6ae832e
Step 7 : RUN apt-get update &&      apt-get install -y mysql-server &&      apt-get clean
----> Running in 4bc0910326a8
Hit:1 http://ftp.daumkakao.com/ubuntu xenial InRelease
Hit:2 http://ftp.daumkakao.com/ubuntu xenial-updates InRelease
Hit:3 http://ftp.daumkakao.com/ubuntu xenial-security InRelease
Reading package lists...
...(중략)...
```

```

Processing triggers for initramfs-tools (0.122ubuntu8.8) ...
---> 6c44bdba6ed2
Removing intermediate container 4bc0910326a8
Step 8 : WORKDIR /etc/mysql/mysql.conf.d
---> Running in 7cbc31ef9146
---> 0dcdb1f408e7
Removing intermediate container 7cbc31ef9146
Step 9 : RUN sed -i "s/127.0.0.1/0.0.0.0/g" mysqld.cnf
---> Running in 4a4883944a1d
---> 63c8785686dc
Removing intermediate container 4a4883944a1d
Step 10 : RUN mkdir /var/run/mysqld
---> Running in 4bc55f4b19e0
---> abb05b63092f
Removing intermediate container 4bc55f4b19e0
Step 11 : RUN chown mysql:mysql /var/run/mysqld
---> Running in 384fc6567737
---> 9f2b45b45c43
Removing intermediate container 384fc6567737
Step 12 : CMD /usr/sbin/mysqld
---> Running in d474739ea66a
---> 8cff15e83ce5
Removing intermediate container d474739ea66a
Step 13 : EXPOSE 3306
---> Running in 538a4157c41d
---> 099506724124
Removing intermediate container 538a4157c41d
Successfully built 099506724124
root@unl01:/data/docker-build/mysqld_server# docker -H=tcp://127.0.0.1:4243 images
REPOSITORY          TAG          IMAGE ID          CREATED           SIZE
mysqld_server        latest       099506724124     About a minute ago 731.9 MB
apache2_web          latest       f60fec959945     16 minutes ago   485 MB
exim4_smtp           latest       b0a438400d39     26 minutes ago   481.7 MB
openssh_server       latest       6c5f72758c56     38 minutes ago   457.7 MB
ubuntu_client        latest       becdf2f9f65e     54 minutes ago   435.2 MB

```

ubuntu

latest

f49eec89601e

2 weeks ago

129.5 MB

## 7. 우분투 vsftpd FTP서버 만들기

- 우분투 vsftpd FTP서버는 LAB상의 클라이언트에서 ftp로 연결테스트를 수행할 대상서버이다.
- 기본 명령어가 포함되도록 구성한 ubuntu\_client이미지를 기본이미지로 사용하여 vsftpd패키지를 추가로 설치하여 서버를 구동한다.
- 서버연결시험을 위하여 기본계정(아이디:admin, 암호:unl)을 생성한다.
- PID파일을 저장할 디렉토리와 vsftpd설정파일을 생성한다.
- 우분투 SSH서버의 통신포트를 설정한다. (EXPOSE 21)

```

root@unl01:/data/docker-build# mkdir vsftpd_server
root@unl01:/data/docker-build# cd vsftpd_server/
root@unl01:/data/docker-build/vsftpd_server# vi Dockerfile
# 우리가 만든 우분투 클라이언트 이미지를 기본이미지로 사용
FROM ubuntu_client

# 이미지 작성자에 대한 기본정보
MAINTAINER Han SangGu <sghan@hinetworks.kr>

# 우분투 저장소를 국내미러사이트를 사용하도록 변경
RUN cd /etc/apt && \
    sed -i 's/archive.ubuntu.com/ftp.daum.net/g' sources.list

# 우분투 패키지 저장소로부터 패키지목록을 받아오고,
# vsftpd 패키지를 설치하고,
# 다운로드했던 패키지파일을 삭제하여 공간확보
RUN apt-get update && \
    apt-get install -y vsftpd && \
    apt-get clean

# vsftpd 에 대한 PID 를 생성할 디렉토리를 만들어준다
RUN mkdir /var/run/vsftpd

# 테스트할 계정을 생성하고 암호설정
RUN groupadd -r admin && useradd -r -g admin -s /bin/bash admin && \
    echo "admin:unl" | chpasswd && \
    mkdir /home/admin
  
```

# vsftpd config 파일을 작성한다.

```
RUN mv /etc/vsftpd.conf /etc/vsftpd.conf.org && \
    echo 'anonymous_enable=NO' >> /etc/vsftpd.conf && \
    echo 'local_enable=YES' >> /etc/vsftpd.conf && \
    echo 'write_enable=YES' >> /etc/vsftpd.conf && \
    echo 'local_umask=022' >> /etc/vsftpd.conf && \
    echo 'dirmmessage_enable=YES' >> /etc/vsftpd.conf && \
    echo 'xferlog_enable=YES' >> /etc/vsftpd.conf && \
    echo 'connect_from_port_20=YES' >> /etc/vsftpd.conf && \
    echo 'xferlog_file=/var/log/xferlog' >> /etc/vsftpd.conf && \
    echo 'xferlog_std_format=YES' >> /etc/vsftpd.conf && \
    echo 'chroot_local_user=NO' >> /etc/vsftpd.conf && \
    echo 'secure_chroot_dir=/var/run/vsftpd' >> /etc/vsftpd.conf && \
    echo 'listen=YES' >> /etc/vsftpd.conf && \
    echo 'pam_service_name=vsftpd' >> /etc/vsftpd.conf && \
    echo 'userlist_enable=NO' >> /etc/vsftpd.conf && \
    echo 'tcp_wrappers=YES' >> /etc/vsftpd.conf
```

# 해당 이미지를 컨테이너로 만들어서 실행했을때 마지막으로 구동되는 명령.

CMD ["/usr/sbin/vsftpd"]

# 통신포트를 정의

EXPOSE 21

root@unl01:/data/docker-build/vsftpd\_server# ls -al

total 12

drwxr-xr-x 2 root root 4096 Feb 5 16:36 .

drwxr-xr-x 8 root root 4096 Feb 5 16:26 ..

-rw-r--r-- 1 root root 1986 Feb 5 16:36 Dockerfile

root@unl01:/data/docker-build/vsftpd\_server# docker -H=tcp://127.0.0.1:4243 build --no-cache --tag vsftpd\_server .

Sending build context to Docker daemon 3,584 kB

Step 1 : FROM ubuntu\_client

---> becdf2f9f65e

Step 2 : MAINTAINER Han SangGu <sghan@hinetworks.kr>

---> Running in a44d2b70f5b3

---> efe6fd08a21a

Removing intermediate container a44d2b70f5b3

```

Step 3 : RUN cd /etc/apt &&      sed -i 's/archive.ubuntu.com/ftp.daum.net/g' sources.list
---> Running in c30e06dc0971
---> 1bbc637aab55
Removing intermediate container c30e06dc0971
Step 4 : RUN apt-get update &&      apt-get install -y vsftpd &&      apt-get clean
---> Running in ec81dfee8fd1
Hit:1 http://ftp.daumkakao.com/ubuntu xenial InRelease
Hit:2 http://ftp.daumkakao.com/ubuntu xenial-updates InRelease
Hit:3 http://ftp.daumkakao.com/ubuntu xenial-security InRelease
Reading package lists...
...(중략)...
Processing triggers for libc-bin (2.23-0ubuntu5) ...
---> 48904f27cb15
Removing intermediate container ec81dfee8fd1
Step 5 : RUN mkdir /var/run/vsftpd
---> Running in 095525a10157
---> 9acc8de7022c
Removing intermediate container 095525a10157
Step 6 : RUN groupadd -r admin && useradd -r -g admin -s /bin/bash admin &&      echo "admin:un1" | chpasswd &&      mkdir /home/admin
---> Running in 8ecdb084feeb
---> 11d3b3d65e9c
Removing intermediate container 8ecdb084feeb
Step 7 : RUN mv /etc/vsftpd.conf /etc/vsftpd.conf.org &&      echo 'anonymous_enable=NO' >> /etc/vsftpd.conf &&      echo 'local_enable=YES' >> /etc/vsftpd.conf &&      echo 'write_enable=YES' >> /etc/vsftpd.conf &&      echo 'local_umask=022' >> /etc/vsftpd.conf &&      echo 'dirmesssage_enable=YES' >> /etc/vsftpd.conf &&      echo 'xferlog_enable=YES' >> /etc/vsftpd.conf &&      echo 'connect_from_port_20=YES' >> /etc/vsftpd.conf &&      echo 'xferlog_file=/var/log/xferlog' >> /etc/vsftpd.conf &&      echo 'xferlog_std_format=YES' >> /etc/vsftpd.conf &&      echo 'chroot_local_user=NO' >> /etc/vsftpd.conf &&      echo 'secure_chroot_dir=/var/run/vsftpd' >> /etc/vsftpd.conf &&      echo 'listen=YES' >> /etc/vsftpd.conf &&      echo 'pam_service_name=vsftpd' >> /etc/vsftpd.conf &&      echo 'userlist_enable=NO' >> /etc/vsftpd.conf &&      echo 'tcp_wrappers=YES' >> /etc/vsftpd.conf
---> Running in cd9a36692765
---> 3abbd447d75b
Removing intermediate container cd9a36692765
Step 8 : CMD /usr/sbin/vsftpd
---> Running in 0d67bc3c0e66
---> 09b43783ae27
Removing intermediate container 0d67bc3c0e66
Step 9 : EXPOSE 21

```



## UnetLab교육자료

*Dockerfile작성하여 이미지만들기*

문서번호: 20170205-01

버전: 1.0

Date: 2017/02/05

```
---> Running in 7276372c8b3b
---> e32b768d5fe6
Removing intermediate container 7276372c8b3b
Successfully built e32b768d5fe6
root@unl01:/data/docker-build/vsftpd_server# docker -H=tcp://127.0.0.1:4243 images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
vsftpd_server	latest	e32b768d5fe6	About a minute ago	439 MB
mysqld_server	latest	099506724124	24 minutes ago	731.9 MB
apache2_web	latest	f60fec959945	40 minutes ago	485 MB
exim4_smtp	latest	b0a438400d39	49 minutes ago	481.7 MB
openssh_server	latest	6c5f72758c56	About an hour ago	457.7 MB
ubuntu_client	latest	becdf2f9f65e	About an hour ago	435.2 MB
ubuntu	latest	f49eec89601e	2 weeks ago	129.5 MB