

Oracle Cloud

Sign Up

Network Setup

1. 메뉴 → 네트워킹 → 개요
2. 인터넷 접속을 통한 VCN 생성; VCN 마법사 시작
 1. vcn 이름
 2. vcn cidr 블록; 10.0.0.0/16
 3. 공용 서브넷 cidr 블록; 10.0.0.0/24
 4. 전용 서버넷 cidr 블록; 10.0.1.0/24
 5. 생성
 6. 가상 클라우드 네트워크 보기
 7. 리소스 → 보안목록 → Default Security List for {VCN 이름}
 1. 수신규칙추가
 1. 소스유형; cidr
 2. 소스; 0.0.0.0/0
 3. IP프로토콜; TCP
 4. 대상 포트 범위; 80, 443
 5. 수신규칙추가

Create SSH Key

```
$ ssh-keygen
$ pbcopy < ~/.ssh/id_rsa.pub
or
$ cat ~/.ssh/id_rsa.pub
```

```
$ vi ~/.ssh/config
Host {alias for connect}
HostName {ip address}
User {user for login}
IdentityFile ~/.ssh/id_rsa
```

In the Oracle Virtual Machine

```
$ vi ~/.ssh/authorized_key
{paste generated ssh key}
```

VM 인스턴스 생성

1. 메뉴 → 컴퓨트 → 인스턴스
2. 인스턴스 생성
3. 배치 및 하드웨어 구성 → 편집
 1. 이미지 → 이미지 변경
 1. 플랫폼 이미지에서 운영체제 선택
 2. 이미지 선택
4. SSH 키 입력
5. 생성

Connect Compute Instance to Network(VNIC)

1. 메뉴 → 네트워킹 → 개요
2. IP관리 → 예약된 공용 IP 주소
 1. 예약된 공용 IP 주소 이름;
 2. 예약된 공용 IP
3. 메뉴 → 컴퓨트 → 인스턴스
 1. 리소스 → 연결된 VNIC
 1. 리소스 → IP주소
 1. ... → 편집
 1. 공용 IP 유형; 공용 IP 없음
 2. 업데이트
 2. ... → 편집
 1. 공용 IP 유형; 예약된 공용 IP → 기존 예약된 IP 주소 선택;

Server Settings

timezone

```
$ sudo dpkg-reconfigure tzdata
```

update and upgrade

```
$ sudo apt update  
$ sudo apt upgrade
```

Install for Web Server

Apache

basic installation

```
$ sudo apt install apache2
$ sudo apt upgrade
$ sudo apache2ctl configtest
$ sudo vi /etc/apache2/apache2.conf
ServerName {IP or domain}
$ sudo systemctl restart apache2
```

When connection refused

```
$ sudo iptables-save > ~/iptables-rules
$ sudo iptables -P INPUT ACCEPT
$ sudo iptables -P OUTPUT ACCEPT
$ sudo iptables -P FORWARD ACCEPT
$ sudo iptables -F
$ sudo systemctl restart apache2
```

apply HTTP/2 , HTTPS TLS v1.3

```
$ sudo add-apt-repository ppa:ondrej/apache2
$ sudo apt update
$ sudo apt upgrade
$ sudo apt install apache2 libapache2-mod-fcgid libapache2-mod-proxy-uwsgi
libapache2-mod-xforward openssl
```

version check

```
$ /usr/sbin/apache2 -v or -V
$ /usr/bin/openssl version
```

~~MySQL~~ mariaDB

install

```
$ sudo apt install mysql-server mysql-client
```

```
$ sudo apt install mariadb-server
```

change login method to password(optional)

```
mysql> UPDATE mysql.user SET plugin='mysql_native_password',  
authentication_string=PASSWORD('{password}') WHERE User='root';  
mysql> commit;  
mysql> FLUSH PRIVILEGES;  
mysql> quit
```

settings

```
$ sudo vi /etc/mysql/mysql.conf.d/mysqld.cnf  
$ sudo vi /etc/mysql/mariadb.conf.d/50-server.cnf  
  
bind-address = 0.0.0.0  
  
$ sudo systemctl restart mysql
```

add user

```
$ sudo mysql  
  
mysql> use mysql;  
mysql> create user {id}('@%' identified by '{password}';  
mysql> grant all privileges on *.* to '{id}'('@%';  
mysql> flush privileges;
```

Php

```
$ sudo apt update  
$ sudo apt upgrade  
$ sudo apt install php php-gd php-xml php-json
```



2021년 1월 22일, 금요일 현재 dokuwiki는 php 8과 호환되지 않는 듯.

```
<code> $ sudo add-apt-repository ppa:ondrej/php $ sudo apt update $ sudo apt upgrade $ sudo apt
```

```
install php8.0-common php8.0-cli php8.0-fpm libapache2-mod-php8.0 php8.0-
{bcmath,bz2,cgi,cli,curl,dba,dev, enchant,fpm,gd,gmp,imap,interbase,intl,ldap,mbstring,mysql,odbc,o
pcache,pgsql,phpdbg,pspell,readline,snmp,soap,sqlite3,sybase,tidy,xml,xmllrpc,zip,,xsl} php-imagick
</code> <code> $ sudo vi /etc/php/8.0/fpm/php.ini date.timezone = Asia/Seoul cgi.fix_pathinfo=0
session.cookie_httponly = 1 session.cookie_secure = 1 memory_limit = 256M post_max_size = 56M
upload_max_filesize = 1024M max_file_uploads = 50 zlib.output_compression = off
max_execution_time = 180 opcache.memory_consumption = 128 opcache.interned_strings_buffer =
8 opcache.max_accelerated_files = 50000 opcache.revalidate_freq = 60 opcache.enable_cli = 1
opcache.enable = 1 opcache.jit_buffer_size = 100M opcache.jit = tracing </code> <code> $ sudo
a2enmod proxy_fcgi setenvif $ sudo a2enconf php8.0-fpm $ sudo systemctl restart apache2 </code>
```

```
$ sudo usermod -a -G www-data ubuntu
```

Let's Encrypt 클라이언트 Certbot 설치

```
$ sudo apt update
$ sudo apt upgrade
$ sudo apt install software-properties-common
$ sudo add-apt-repository universe
$ sudo apt update
$ sudo apt install certbot python3-certbot-apache
```

```
$ sudo certbot --apache
email;
ACME (A)gree/(C)ancel; A
receive email (Y)es/(N)o; N
Which names would you like to activate HTTPS for?;
1: No redirect, 2: Redirect; 2

if not appear domain name, try this
$ sudo certbot --apache -d "{domain address}"
1: Attempt to reinstall this existing certificate, 2: Renew & replace the
cert (limit ~5 per 7 days);
1: No redirect, 2: Redirect;
```

인증서 갱신이 정상인 확인

```
$ sudo certbot renew --dry-run
$ sudo systemctl restart apache2
```

```
$ certbot renew
```

```
$ sudo crontab -e
# renew cert; every month 1st day 4
0 4 1 * * /usr/bin/certbot renew --renew-hook="sudo systemctl restart
apache2"
```

```
$ sudo certbot delete --cert-name {cert name}
```

phpMyAdmin

```
$ sudo apt install phpmyadmin
```

```
$ sudo vi /etc/apache2/apache2.conf

Include /etc/phpmyadmin/apache.conf

$ sudo systemctl restart apache2
```

Web Server Settings

migrate

dokuwiki

wordpress

dns settings

set multi domain

ssl setting

automations

ssl update

db backups

wiki backups

References

- 2. 오라클 클라우드에서 우분투 LEMP - 인스턴스 서버 생성하기 (네트워크 셋팅을 포함)
- macOS: 맥에서 SSH 키 생성하고 사용하기
- 오라클 프리티어 계정 생성 및 인스턴스 생성하기
- Connect to Oracle Cloud with SSH and VNC
- Oracle Cloud SSH Key 여러 개 등록하기 (여러 PC에서 Cloud Access)
- 오라클 클라우드 VM 인스턴스 2개 생성할 때 깔끔하게 구조화하는 방법
- 오라클 클라우드 방화벽 설정에 대한 모든 것
- 우분투(Ubuntu) APM - Apache PHP Mysql 구성 및 설정
- 우분투(Ubuntu) 18.04 서버 초기 설정
- PHP 25년 역사의 새장을 여는 PHP 8 설치 방법 – 우분투 및 데비안 기준
- How to Install PHP 8.0 on Ubuntu 20.04 / 18.04
- Ubuntu 18.04 / phpMyAdmin 설치하는 방법
- [Ubuntu] 쉽게 따라할 수 있는 Apache2, Mysql, PHP 설치 하기
- [MySQL] 사용자 추가 및 권한 설정

From:

<http://dev.theta5912.net/> - **reth.dev**

Permanent link:

http://dev.theta5912.net/doku.php?id=public:computer:oracle_cloud&rev=1611296665

Last update: **2021/01/22 15:24**

