

Mobius with AWS

AWS

무료 계정 (프리티어)

Amazon Elastic Compute Cloud (Amazon EC2)

Run Mobius with AWS

1. EC2 Instance

1.1. create **Instance**

1.2. Security group rule settings

1.3. Connect to EC2 Instance

2. Mobius 구동환경 구성

2.1. Install Node.js

2.2. Install MQTT Server

2.3. Install **mysql**

3. Install Mobius

3.1. Mobius Download

3.2. Create Database

4. 환경설정

5. 실행

서버로 AWS에서 Mobius 구동하기

pm2

Elastic IP (탄력적 IP)

AWS

무료 계정 (프리티어)

- AWS 프리티어는 처음 AWS에 가입한 날부터 12개월 동안 사용이 가능하다.
- 프리티어가 사용할 수 있는 리소스, 시간 등이 제한되어 있으며 이게 초과될 시에는 요금이 부과될 수 있다.

Amazon Elastic Compute Cloud (**Amazon EC2**)

EC2는 **인스턴스** 라고 하는 가상 컴퓨팅 환경을 제공합니다. 인스턴스에 서버를 구축해서 관리할 수 있다. 운영체제, 인스턴스 계산 능력, 메모리 등 다양한 구성을 제공한다.

인스턴스는 시간 단위로 계산되며 **[콘솔]**로 요금을 잘 확인하는게 좋다

EC2 프리티어 요금

- 1년간 매달 750 시간의 linux/windows t2.micro 인스턴스 사용시간을 제공받는다.
- 프리 티어를 계속 유지하려면 EC2 마이크로 인스턴스만 사용해야한다

Run Mobius with AWS

1. EC2 Instance

1.1. create Instance

인스턴스 시작 버튼을 누르면 인스턴스를 생성할 수 있다. 아래 옵션으로 지정하여 인스턴스를 생성한다. 프리티어 계정이므로 고려할 사항이 그닥 없다 (저 외의 옵션은 기본 값을 사용).

- ubuntu server 18.04 LTS (64비트 x86)
- t2.micro

키페어 는 생성/설정 후, 파일을 저장한다 (이후 **ssh** 연결 시 필요하다)

1.2. Security group rule settings

ssh 로 instance에 접속해 mobius를 설치하고, 설치한 mobius를 테스트하려면 외부 포트에 대한 접속을 허용해 주어야 한다.

- **ssh** : 22 번 포트 사용
- **mobius** : 7579 번 포트 사용

[네트워크 및 보안 > 보안 그룹] 에서 새로운 보안 그룹을 생성하여 아래와 같은 규칙을 적용해준다.

Inbound rules

	Name	보안 그룹 ID	보안 그룹 이름	VPC ID	설명	소유자	인바운드 규칙 수	아웃바운드 규칙 수
<input checked="" type="checkbox"/>	-	sg-0a17cf37dd6eb99a1	mobius-rule	vpc-3231ab59	ssh access and mobius ...	386515813601	4 권한 항목	2 권한 항목
<input type="checkbox"/>	-	sg-122e9f62	default	vpc-3231ab59	default VPC security gr...	386515813601	6 권한 항목	3 권한 항목

sg-0a17cf37dd6eb99a1 - mobius-rule

세부 정보

인바운드 규칙

아웃바운드 규칙

태그

Inbound rules (4)

Filter security group rules

< 1 >

<input type="checkbox"/>	Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
<input type="checkbox"/>	-	sg-r-02f48f7b375629b49	IPv6	사용자 지정 TCP	TCP	7579	::/0	-
<input type="checkbox"/>	-	sg-r-08e020b92fa75d332	IPv6	SSH	TCP	22	::/0	-
<input type="checkbox"/>	-	sg-r-012a72c4d022890...	IPv4	사용자 지정 TCP	TCP	7579	0.0.0.0/0	-
<input type="checkbox"/>	-	sg-r-03f2923b60659afac	IPv4	SSH	TCP	22	0.0.0.0/0	-

Outbound rules

보안 그룹 (1/2) 정보

Q 보안 그룹 필터링

Name	보안 그룹 ID	보안 그룹 이름	VPC ID	설명	소유자	인바운드 규칙 수	아웃바운드 규칙 수
-	sg-0a17cf37dd6eb99a1	mobius-rule	vpc-3231ab59	ssh access and mobius ...	386515813601	4 권한 항목	2 권한 항목
-	sg-122e9f62	default	vpc-3231ab59	default VPC security gr...	386515813601	6 권한 항목	3 권한 항목

sg-0a17cf37dd6eb99a1 - mobius-rule

세부 정보 | 인바운드 규칙 | **아웃바운드 규칙** | 태그

Outbound rules (2)

Q Filter security group rules

Name	Security group rule...	IP version	Type	Protocol	Port range	Destination	Description
-	sgr-02e69893aa85d1b...	IPv4	모든 트래픽	전체	전체	0.0.0.0/0	-
-	sgr-0eb3823e1d6773...	IPv6	모든 트래픽	전체	전체	::/0	-

1.3. Connect to EC2 Instance

ssh 접속을 통해 instance에 접근하여, mobius 를 설치/실행할 수 있다.

SSH client 를 사용해 Linux instance 연결

ssh client 를 사용하여 인스턴스 연결한다. 아래는 터미널에서 **ssh** 명령어로 접속하는 예시이다.

- **[key path]** : private key 경로
- **[username]** : instance user name
 - Linux 인스턴스의 기본 user name은 인스턴스를 시작할 때 지정된 AMI에 의해 결정된다 (ex. ubuntu: **ubuntu**)
- **[public dns name]** : IPV4 public dns name
 - IPV6 는 문서 참조

```
ssh -i "[key path]" [username]@[public dns name]
```

[Error] 다른 ssh client의 경우는 모르겠으나, vs code 의 ssh client 확장 프로그램의 경우 timeout error가 계속 발생하였다.

2. Mobius 구동환경 구성

Mobius 구동환경 구성 / 설치 는 [IoTKETI/Mobius/doc](#) 내의 문서에서 상세히 다루고 있다. 해당 문서에는 명령어, 오류 위주로만 간단히 정리해두었다.

2.1. Install Node.js

`nvm` 은 node.js version을 관리해주는 도구로, node.js 설치와 버전 전환을 쉽게해준다.

```
# install nvm
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.34.0/install.sh | bash

# activate nvm
. ~/.nvm/nvm.sh

# install nodejs
nvm install node
```

`node -v`, `npm -v` 으로 버전 확인

2.2. Install MQTT Server

Mobius 서버는 디바이스와의 통신을 위해 MQTT 프로토콜을 지원하며, 이를 위해 mosquitto (MQTT broker server) 를 설치해야 한다.

```
sudo apt-get install mosquitto
sudo apt-get install mosquitto-clients
mosquitto -v
```

2.3. Install mysql

1. install mysql

`5.7` 버전으로 설치 진행하였다. (8.x으로 설치해도 실행이 가능하나 설정 값이 다른 것으로 함)

```
sudo apt-get install mysql-server-5.7
```

2. 설치 후에, 비밀번호를 설정한다.

```
sudo mysql -u root -p
```

3. Install Mobius

3.1. Mobius Download

1. 아래 명령어를 통해 Mobius server platform 파일을 다운로드한다.

```
git clone https://github.com/IoTKETI/Mobius
```

2. package 다운로드

Mobius 디렉토리 내에서 `npm install` 명령어 실행하여 모듈들을 다운로드한다.

```
cd Mobius
npm install # or sudo npm install
```

3.2. Create Database

`sudo mysql -u root -p` 명령어로 root로 로그인하여 진행한다.

1. DB 생성

create 명령어로 db생성 후, show 명령어로 db가 생성되었는지 확인한다.

```
mysql> create database mobiusdb;
mysql> show databases;
```

2. 테이블 생성

a. 테이블을 생성하기 위해, 다운로드 받았었던 Mobius server platform 폴더 내의 sql 파일을 import 한다.

```
cd [your-download-folder]/Mobius/mobius
sudo mysql -u root -p mobiusdb < mobiusdb.sql
```

b. 테이블이 생성되었는지 확인한다.

```
use mobiusdb;
show tables;
```

[error] `ERROR 1273 (HY000) at line 249: Unknown collation: 'utf8mb4_0900_ai_ci'`

- **문제:** sql 파일 import시 오류 발생
- **해결:** `sed` 명령어로 파일 편집
 - `utf8mb4_0900_ai_ci` > `utf8mb4_general_ci`
 - `CHARSET=utf8mb4` > `CHARSET=utf8``

```
sed -i 's/utf8mb4_0900_ai_ci/utf8mb4_general_ci/g' mobius.sql
sed -i 's/CHARSET=utf8mb4/CHARSET=utf8/g' mobius.sql
```

[error] `[db.connect] No Connection`

- **문제:** mobius 실행 시, 해당 출력 이후 동작 정지

```
sudo node mobius.js
...
[db.connect] No Connection
```

- 해결: root 사용자에게 대한 인증방법 변경
- reference
 - <https://github.com/loTKETI/Mobius/issues/33>
 - <https://vitux.com/how-to-install-and-configure-mysql-in-ubuntu-18-04-lts/>

4. 환경설정

Mobius의 환경설정은 `conf.json` 파일을 통해 변경할 수 있다. `conf.json` 은 Mobius의 포트번호와 db 접근 비밀 번호를 저장하고 있다. db 접근 비밀번호는 개인이 설정한 비밀번호로 변경해야 한다.

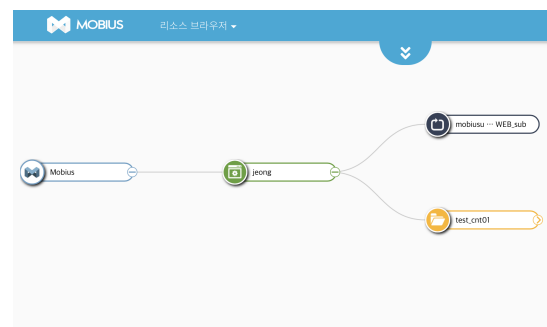
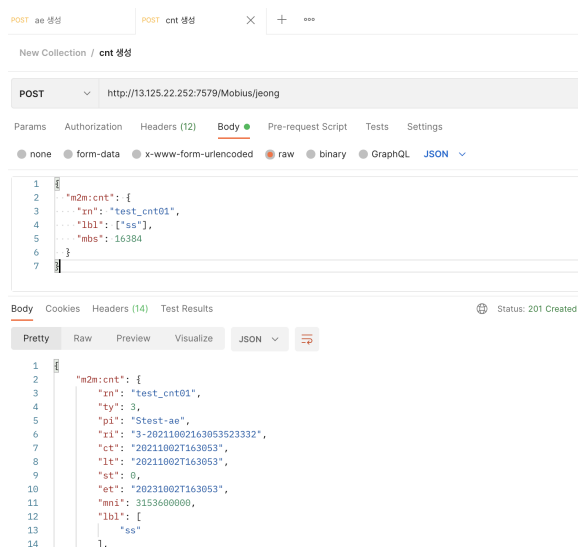
```
// conf.json
{
  "csebaseport": "7579",
  "dbpass": "qwer1234" # Your DB Password
}
```

Mobius 디렉토리 내에 해당 파일을 만들어도 되고, `mobius.js` 에서 `conf.json` 파일을 작성하는 코드를 수정해도 된다.

5. 실행

`sudo node mobius.js` 명령어를 통해 서버를 구동한 뒤, postman, web browser 등으로 API Testing 을 진행하면 된다.

외부에서 Mobius에 접근할 때는 instance의 "퍼블릭 IPv4 주소"를 사용해야 한다.



서버로 AWS에서 Mobius 구동하기

아래의 글들은 실제로 실행해본 글이 아니라, 위의 과정들을 찾아보는 과정에서 도움이 될까 싶어 같이 작성한 글입니다.

pm2

시도해보지는 않았으나, pm2 프로그램을 설치하면 터미널을 종료해도 서버를 계속 구동시킬 수 있다고 합니다.

Elastic IP (탄력적 IP)

ec2 생성 후, 기본으로 생성되는 public ip는 ec2가 재부팅 될 때마다 새로운 ip로 재할당 됩니다. elastic ip는 고정 ip를 생성해주는 서비스로 이를 사용하여 서버의 ip를 고정적으로 유지할 수 있습니다.

다만, **elastic ip는 생성 후, 사용하지 않는 동안에는 요금이 부과될 수 있으므로** 주의해서 사용해야 할 것 같습니다. 예시로, 프리티어 계정으로 ec2 인스턴스에 elastic ip를 연동한 뒤, 인스턴스를 실행하지 않으면 요금이 부과됩니다. 자세한 내용은 [aws 페이지](#)를 참고하면 좋을 것 같습니다.